

## Absolventská práce

**Prohlašuji, že jsem absolventskou práci na téma Oficiální WWW prezentace firmy s využitím technologie PHP a MySQL vypracoval samostatně a použil jen pramenů, uváděných v bibliografii.**

**V Českých Budějovicích dne 1. května 2006**

**podpis autora**

**Studijní obor: Výpočetní technika a programování**

**Oficiální WWW prezentace firmy s využitím technologie**

**PHP a MySQL**

**Absolventská práce**

**Autor: Petr JAROLÍMEK**

**Vedoucí absolventské práce: PaedDr. Petr PEXA**

**Tímto bych chtěl poděkovat PaedDr. Petru Pexovi za odborné vedení,  
cenné rady a připomínky při vypracování absolventské práce.**

---

<b>1 ÚVOD</b> .....	<b>6</b>
<b>2 POUŽITÉ TECHNOLOGIE</b> .....	<b>7</b>
2.1. PROGRAMOVACÍ JAZYKY PRACUJÍCÍ NA STRANĚ KLIENTA .....	7
2.2. PROGRAMOVACÍ JAZYKY PRACUJÍCÍ NA STRANĚ SERVERU .....	8
2.3. SKRIPTOVACÍ JAZYK PHP .....	9
2.4. XHTML .....	11
2.5. CSS .....	14
2.6. JAVASCRIPT .....	15
2.7. MYSQL .....	17
<b>3 JAKÝ INTERNETOVÝ PROHLÍŽEČ UPŘEDNOSTNIT?</b> .....	<b>19</b>
<b>4 UŽIVATELSKÉ ROZHRANÍ</b> .....	<b>21</b>
4.1. GRAFICKÁ STRUKTURA UŽIVATELSKÉHO ROZHRANÍ.....	21
4.2. PŘIHLÁŠENÍ UŽIVATELE .....	22
4.3. SYSTÉMOVÁ NAVIGACE .....	23
4.3.1. <i>Systémová navigace administrátora</i> .....	24
4.3.1.1. Práce s Osobními daty.....	25
4.3.1.2. Nastavení kurzů a účtů.....	25
4.3.1.3. Sponzoři .....	25
4.3.1.4. Uživatelé (distributoři).....	26
4.3.1.5. Uživatelé (zákazníci).....	26
4.3.1.6. Aktuality.....	26
4.3.1.7. Akce .....	27
4.3.1.8. Kontakty.....	27
4.3.1.9. Diskuse.....	28
4.3.1.10. Bady .....	28
4.3.1.11. Kosmetika .....	29
4.3.1.12. Přístroje .....	30
4.3.1.13. Ostatní .....	30
4.3.1.14. Kúry .....	31
4.3.1.15. Seznam objednávek distributorů.....	32
4.3.1.16. Seznam objednávek zákazníků .....	32
4.3.2. <i>Systémová navigace distributora</i> .....	33
4.3.2.1. Osobní data .....	33
4.3.2.2. Objednávka .....	33
4.3.2.3. Seznam objednávek.....	34
4.4. ODHLÁŠENÍ UŽIVATELE .....	34
<b>5 ZÁVĚR</b> .....	<b>35</b>
<b>LITERATURA</b> .....	<b>36</b>
<b>REJSTŘÍK (DODELAT)</b> .....	<b>CHYBA! ZÁLOŽKA NENÍ DEFINOVÁNA.</b>
<b>PŘÍLOHY</b> .....	<b>37</b>

# 1 Úvod

Spolu s rozvojem zobrazování informací na internetu vznikla potřeba upravovat a přidávat nové informace. Současná podoba internetových sídel by se upravovala bez pomocných nástrojů velmi složitě. Vzhledem k nutnosti zjednodušit práci s daty internetové presentace na takovou úroveň, aby s nimi mohli pracovat zaměstnanci s běžnou znalostí práce na počítači, začali vznikat správci internetového obsahu.

Existuje několik druhů správců internetového obsahu. Ve své diplomové práci se zabývám aplikací, která by měla být jednoduše rozšiřitelná a upravitelná, aby systém neomezoval rozmanitost spravovaných presentací.

Aplikace by se měla rozdělit do dvou částí, správy textu a správy produktů. Správa textu bude sloužit k práci s texty. Správa produktů by měla sloužit ke snadnému vytváření katalogových listů a elektronického obchodu. Systém by měl být schopný udržovat hierarchii informací a vytvářet navigaci presentace.

Práce se systémem by měla být umožněna pouze oprávněným osobám. Dalším požadavkem na systém je také možnost omezit použití některé funkce jen na určitou skupinu uživatelů.

## 2 Použité technologie

### 2.1. Programovací jazyky pracující na straně klienta

Jedná se o jazyky, které jsou schopny změnit některé části internetové prezentace aniž by došlo ke komunikaci se serverem, např. změna obrázku po najetí kurzoru. Internetová prezentace musí přijmout všechny informace ze serveru (např. z databáze), které by mohli být potřebné při běhu programu. Bohužel se pak často stává, že uživatel přijme i nepotřebné informace. Pokud by se tento způsob programování využil k programování aplikace, která pracuje s velkým množstvím informací, nastává pak příliš dlouhé nahrávání informací do lokální paměti uživatele, s tím je spojen i výskyt chyb při přenosu a netrpělivost uživatele. Další nevýhodou, která zřejmě nejvíce ovlivnila rozvoj programovacích jazyků na straně klienta je závislost výstupu aplikace na druhu klienta. V historii vzniklo několik verzí programovacích jazyků tohoto druhu, ale verze nebyly navzájem kompatibilní, ani jazyky vyvíjené různými společnostmi nebyly kompatibilní.

Mezi významné představitele této kategorie patří JavaScript, Java, JScript a Visual Basic. Jazyky pracující na straně klienta, prošly velice nepřehledným vývojem. V současnosti je situace taková, že veškeré aplikace musí být testovány a upravovány pro jednotlivé internetové prohlížeče, vzniká tak několik verzí jedné aplikace, což neprospívá možnému rozšíření tohoto druhu jazyků. Obvykle je vytvořena verze pro IE 6.0 a 5.0, Firefox a prohlížeče používané uživateli jiných operačních systémů, zejména Linux a Macintosh. Vývoj neustále pokračuje a již existují obecně platné standarty a je otázkou času a vůle firem, kdy se situace zlepší.

Jazyky JScript ( Java Script ) a VBScript ( Visual Basic Script ) jsou vyvinuty firmou Microsoft a zaručují tak kompatibilitu s programem Internet Explorer, ale ani jeden jazyk se nestal dominantním na trhu. JavaScript od společnosti Netscape je dodnes velmi populární a nadále vznikají nové verze, ačkoliv společnost

Netscape ukončila svou činnost jednak na poli internetových prohlížečů, jednak mezi programovacími jazyky. Vytvářet aplikace typu správce obsahu internetových presentací pouze pomocí jazyka JavaScript, nebo JScript je zřejmě nemožné, ale tyto jazyky jsou používány jako krátké procedury či funkce, které pomáhají zpřehlednit ovládání celého systému. Jiná situace nastává při použití jazyka Java od společnosti Sun. Java je jednak velice výkonný jazyk při použití na straně klienta, ale i na straně serveru, jeho rozšíření JSP (Java Server Page). Použití jazyka Java přesahuje tvorbu internetových aplikací.

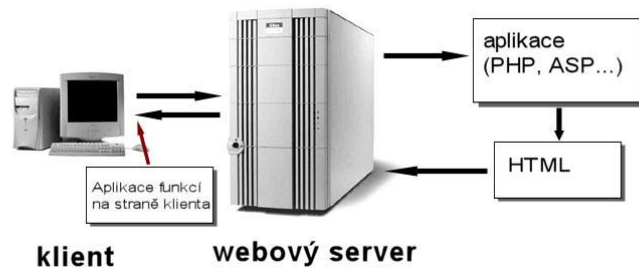
Velkou výhodou programování na straně klienta je možnost okamžité reakce na požadavek uživatele. Pokud jsou všechna data aplikace v lokální paměti, pak je práce s takovou aplikací srovnatelná s běžnými programy, psanými bez síťové komunikace.

## **2.2. Programovací jazyky pracující na straně serveru**

Tyto jazyky pracují tak, že každý požadavek uživatele je odeslán na server, tam je tento požadavek zpracován a výsledek poslán zpět do prohlížeče v podobě statické stránky, proto jsou tyto jazyky nezávislé na používaném internetovém prohlížeči či zařízení. Jazyky pracující na straně serveru jsou používány zejména pro komunikaci s databázemi. Použití těchto jazyků může být limitováno operačním systémem na serveru, i když v současné době většina jazyků je napsána pro několik významných systémů, a tak toto omezení není již tolik významné.



získávání informací pomocí internetové aplikace



**Obrázek 1: Komunikace webservru včetně obsluhy internetových programovacích jazyků**

Komunikace webservru včetně obsluhy internetových programovacích jazyků

Klient odesílá serveru nějaký příkaz, či dotaz, ten je zpracován aplikací, která vrátí odpovídající výsledek, který musí být v nějakém přijatelném formátu, například HTML. Tento způsob práce zvyšuje bezpečnost internetových aplikací, klient obdrží prostý dokument HTML, který nemůže samotnou aplikaci nijak ohrozit. Výhodou je nezávislost napsané aplikace na použitém operačním systému klienta, výsledkem je dokument HTML. V této kategorii existuje několik programovacích jazyků z nichž každý je vhodný na určitý druh projektu.

## 2.3. Skriptovací jazyk PHP

PHP patří mezi nejpoužívanější programovací jazyky svého druhu, pracuje na webservru Apache, který je obvykle nainstalován na operačním systému Linux. PHP je možné bez problémů používat na mnoha dalších operačních systémech včetně Windows, což je jeho nespornou výhodou. PHP je šířen jako open-source, může být tedy modifikován podle potřeb programátora, stejně tak i mnoho podpůrných knihoven a nástrojů. Na internetu existuje dostatečné množství návodů a významná podpora programátorské komunity. PHP snadno komunikuje s databázemi jako je MySQL, PostgreSQL a řada dalších.

Za velkou nevýhodu PHP můžeme považovat možnost nahlédnutí do zdrojových kódů kýmkoliv, kdo má přímý přístup k serveru. Proto PHP nazýváme ne kompilovaný jazyk.

Výhodou možnosti jednoduchého prolínání kódu HTML a PHP, kde PHP je odděleno značkami `<?php ?>`, je vytvoření stránek dynamických, kde je využit jak klasický statický jazyk HTML, tak možnost část kódu generovat pomocí skriptu PHP a reagovat tak na akce uživatele. Těmito akcemi může být například požadavek na zobrazení určitého článku uloženého v databázi, nebo naopak zadávání dat do databáze.

Nejčastější využití PHP nalezneme v souvislosti s dynamickými webovými stránkami a to i s podporou databází. Mezi nejčastější aplikace psané v PHP patří například:

- **internetové obchody**
- **podnikové informační systémy**
- **diskusní fóra**
- **redakční systémy**
- **firemní prezentace**
- **vyhledávače a katalogy**

```
<div class="stred">
  <?php
    if(!$_GET['page'])
    {
      $_GET['page']="main";
    }
    include"data/" . $_GET['page'] . ".php";
  ?>
</div>
```

**Obrázek 2: PHP kód**

`$_GET[„promena“]`

superglobální proměnná, která zjišťuje hodnotu z proměnné

`include()`

příkaz vloží a ohodnotí specifikovaný soubor

## 2.4. XHTML

Předchůdce jazyka XHTML (eXtensible HyperText Markup Language) je jazyk HTML. XHTML je jazyk, ve kterém se webové stránky vytváří. K jeho vytvoření totiž nepotřebujete mít žádný speciální software. Vystačí si s jakýmkoliv textovým editorem jako je třeba Poznámkový blok ve Windows. Přestože právě Windows je nejrozšířenějším operačním systémem, není jediný.

Dokument XHTML se řídí definicí DTD standardu XHTML, tento formát se právem stává nejpopulárnějším formátem dokumentů na internetu. Nejpodstatnější informací plynoucí z jeho názvu je, že jde o jazyk *značkovací*. To znamená, že se v něm používají speciální značky nazvané tagy, které určují význam textu v dokumentu.

Některé důležité změny oproti HTML

- **Názvy tagů musí být psány malými písmeny.**
- **Všechny tagy musí mít také buď koncovou značku nebo musí končit `</>`. př. tag `<br>` v XHTML vypadá takto `<br />`**
- **Hodnoty atributů musí být psány v uvozovkách. To platí i pro číselné a jednoslovné hodnoty.**

```
<table width="860" border="0" class="patka" cellpadding="4" cellspacing="0">
<tr>
<td>
<div class="powered">
Copyright © 2005 BAdY.CZ, všechna práva vyhrazena || Powered by <p class="email">
<a href="mailto:admin@bady.cz">JeremyCB</a>
</p>
</div>
</td>
</tr>
</table>
```

*Obrázek 3: XHTML kód*

**<table></table>**

tabulka, uzavírá celou strukturu tabulky

**width**

minimální šířka tabulky (délka nebo procento)

**border**

šířka rámečku buněk (pixely)

**class**

Třída elementu používaná pro CSS. Umožňuje jednoduché označení elementu, který má mít jiný vzhled. V zápisu stylu se jménu třídy předřazuje tečka

**cellpadding**

vnitřní okraj buněk (pixely)

**cellspacing**

vnější okraj buněk (pixely)

**<tr></tr>**

Řádek tabulky, musí být uvnitř elementu `<table>`, smí obsahovat pouze buňky (`<td>` a `<th>`). Přípustnými (ale zastaralými a neužívanými) atributy jsou `height`, `background`, `bgcolor`. Jde o tag párový, ale pokud se neuvede koncová značka, v prohlížeči se nic zlého nestane, protože to je stejně vždycky před začátkem nového řádku nebo před koncem tabulky. Slušností je ovšem tag `<tr>` ukončovat, už pro přehlednost.

**<td></td>**

Buňka tabulky. Musí být umístěna v řádku `<tr>`. `<td>` je párový tag, ale pokud se neuzavře, tak si s tím prohlížeče poradí.

**<div></div>**

Div se nejčastěji překládá jako oddíl. Zahrnuje v sobě libovolně velkou oblast textu včetně nadpisů, obrázků a tabulek. Nezřídka se celá stránka vyskytuje uvnitř jednoho elementu `<div>`.

Před a za oddílem se zalomí řádek. V tomto smyslu je `<div>` velmi podobný odstavci (`<p>`). Narozdíl od odstavce ale `<div>` nedělá před a za sebou vizuální mezery (aneb má nulový výchozí margin).

Tag `<div>` byl do HTML přidán zejména kvůli kaskádovým stylům. V tom se zase podobá tagu `<span>`, který také přibyl kvůli stylům. Rozdíl mezi `<div>` a `<span>` je v tom, že `<div>` je blokový element (způsobuje zalomení řádku před a po), kdežto `<span>` je in-line element (řádkový).

### `<p></p>`

Odstavec, základní formátovací tag. Značka `<p>` se nachází na začátku každého odstavce. Původně šlo o tag nepárový, ale postupně se začala používat i značka `</p>`, která tag ukončila. Novější specifikace vyžadují ukončovací tag.

Text nového odstavce vždy začíná nový řádek a konec odstavce končí řádkem. Klasický odstavec v HTML má před sebou a za sebou navíc vertikální mezeru, která většinou přesně odpovídá výšce jednoho řádku, což odpovídá anglosaské normě. (Této mezery se lze zbavit pomocí CSS stylu margin, například `<p style="margin-top: 0px; margin-bottom: 0px">`.) Pokud se vyskytnou za sebou dva odstavce nebo jiné elementy s vertikální mezerou, výška mezery se nesčítá. Tag `<p>` je velmi často používán HTML editory jako základní prvek dokumentu.

Texty, které mají odstavce značené pomocí tagu `<p>`, jsou dobře transformovatelné do jiných formátů a dobře se s nimi pracuje pomocí CSS.

### `<a></a>`

Odkaz, vše, co je mezi značkami `<a>` a `</a>` bude sloužit jako odkaz (tzn. bude klikatelný).

Href, nejdůležitější atribut, cíl odkazu. Zjednodušeně pojato tato hodnota říká, jaká stránka se objeví po kliknutí.

## 2.5. CSS

Je to zkratka Cascading Style Sheets, česky "kaskádové styly". Kaskádové, protože se na sebe mohou vrstvit definice stylu, ale platí jenom ta poslední. Mnoho lidí se domnívá ze CSS je skriptovací jazyk. Není tomu tak. Je to souhrn značek a metod pro grafickou úpravu stránek. Tyto vlastnosti a metody se vpisují přímo do HTML a také mají svou vlastní (avšak jednoduchou) syntaxi.

```
body{
font-family:verdana, arial, serif;
font-size:11px;
text-align:center;
}

p{display:inline;}

.logo{
margin:10px 10px 0px 10px;
}

.menuitemtop{
font-size:11px;
font-weight:bold;
text-align:center;
background-color:#464646;
color:#4BC651;
}
```

**Obrázek 4: CSS kód**

### **body**

Tělo dokumentu, obsahuje veškerý zobrazovaný obsah stránky

### **font-family**

Určuje font, který se použije k vykreslení písma (*jméno fontu1, jméno fontu2*)

*nejčastěji používané fonty:* Arial, Times, Verdana, Tahoma, serif, sans-serif

### **font-size**

Určuje velikost písma (*velikost v jednotkách, klíčové slovo*)

*příklady různých jednotek:* px, pt, mm, in, em, ex, en a procenta

*klíčová slova:* xx-small, x-small, small, medium, large, x-large, xx-large

**text-align**

Určuje zarovnání odstavce nebo jiného bloku (*left, center, right, justify*)

**p**

odstavec

**display**

Určuje způsob zobrazení prvku (*block, inline, list-item, none*)

**.logo**

Vlastní třída, vztahuje se na každý tag, který má uvedeno správné class

**margin**

Určuje šířku vnějšího okraje prvku (*délka, procento, auto, a b c d, bez hodnoty*)

**.menuItemtop**

Vlastní třída, vztahuje se na každý tag, který má uvedeno správné class

**font-size**

Určuje velikost písma (*velikost v jednotkách, klíčové slovo*)

**font-weight**

Určuje tučnost písma neboli duktus

*příklad:* (normal, bold, 100, 400, 700, násobky 100 do 900, bolder, lighter)

**text-align**

Určuje zarovnání odstavce nebo jiného bloku (*left, center, right, justify*)

**background-color**

Určuje barvu pozadí (*barva, transparent*)

**color**

Určuje barvu textu a rámečku (*barva*)

## 2.6. JavaScript

JavaScript je klientský skriptovací jazyk. Příkazy tohoto jazyka jsou pak provedeny na straně klienta a ne na straně serveru, jak je tomu například u PHP. Tento systém má výhody i nevýhody. Výhodou je mnohem menší zatěžování serveru, na kterém jsou stránky umístěny, a prostředí klienta, které je odlišné od

prostředí serveru a lze na něm provádět i věci, které na straně serveru nelze. Velkou nevýhodou je viditelnost kódu v HTML, zdrojový kód si může každý snadno zobrazit a tak vlastně „opsat“ pracně vymyšlené skripty.

Co se týče bezpečnosti v JavaScriptu. JavaScript na straně klienta neumožňuje čtení a zapisování souborů z důvodů, které jsou zřejmé. Nebyl by pak problém jednoduchým programkem naprosto znehodnotit obsah pevného disku. Rovněž nepodporuje práci se sítí, až na jednu důležitou výjimku: může donutit webovský prohlížeč k načtení libovolného URL.

JavaScript však není jediný skriptovací jazyk, který se používá. Jeho největším konkurentem je asi určitě VBScript, ten však rozhodně není tak rozšířen.

```
function diskuse()
{
  if (document.dis.name.value == "")
  {
    document.dis.name.focus();
    alert ("Nezadal(a) jste vaše jméno");
    return false;
  }
  if (document.dis.email.value != "")
  {
    re = new RegExp ("^[^.]+(\\.[^.]*)*@[^[^.]++[a-z]{2,3}$");
    if (!re.test(document.dis.email.value))
    {
      alert ("Zadaná adresa není správnou adresou elektronické pošty!");
      return false;
    }
  }
}
```

### ***Obrázek 5: JavaScript kód***

Funkce diskuse() nám hlídá vyplnění údajů pro přidání nového téma. Pokud není vyplněno políčko jméno, sdělíme uživateli „Nezadal(a) jste vaše jméno“. Pokud je zadaná emailová adresa, testujeme zda má platný tvar. Formulář, na kde bude tato funkce aplikována musí obsahovat hodnotu name=“dis“.

```
document.dis.name.value == ""
```

Zjišťuje, zda hodnota prvku name je prázdná.



`document.dis.name.focus()`

Metoda `focus()` nastaví aktivní textové pole.

`alert`

Zobrazí upozorňovací okénko s příslušným textem.

`document.dis.email.value == ""`

Zjišťuje, zda hodnota prvku `email` není prázdná.

`re = new RegExp("regulární výraz")`

Proměnné `re` přiřadíme masku regulárního výrazu.

`re.test`

Metoda `test()` porovnává vložený textový řetězec s maskou regulárního výrazu a vrací `TRUE` pokud podmínice vyhovuje.

## 2.7. MYSQL

Populární databáze, která je volně distribuovaná pro nekomerční účely. MySQL patří do standardní nabídky databází pro virtuální servery. Databázový systém je založený na tabulkách. Řádky zastupují jednotlivé záznamy, ve sloupcích jsou pak obsaženy jednotlivé hodnoty. Tento způsob zápisu je ideální pro různé seznamy, ceníky, zkrátka jakákoli data zapisovatelná pomocí tabulky. Velkou výhodou MySQL je rychlost a jednoduchost. Běží totiž jako samostatný server a k práci s daty se většinou využívá skriptování na straně serveru pomocí PHP, nebo ASP. I proto našel velké využití právě na internetu.

S MySQL se pracuje pomocí příkazů, které se zakončují středníkem. Základním prvkem MySQL je databáze. Ta obsahuje tabulky, které mají definovaný počet sloupců. Do řádků v tabulkách se pak ukládají samotná data. Ta mohou mít různý formát, který ale musí odpovídat parametrům daného sloupce.

```
<?php
$dbname = "db_bady";
$dbserver = "localhost";
$dbuser = "root";
$dbpass = "";

@$db = mysql_connect("$dbserver", "$dbuser", "$dbpass") or die ("Nelze navázat spojení s databází");
@mysql_select_db("$dbname", $db) or die ("Databáze nenalezena");
?>
```

**Obrázek 6: MySQL kód**

#### **mysql\_connect**

**resource mysql\_connect (string server, string uziv, string heslo)**

Funkce `mysql_connect()` vytvoří spojení s databázovým serverem MySQL, je-li spojení úspěšně navázáno, je vrácen identifikátor spojení. Nejsou-li zadány nepovinné parametry, použijí se následující hodnoty: `server = $dbserver`, `uziv = $dbuser`, pod kterým běží právě spuštěný skript a `heslo = $dbpass`. Bude-li funkce `mysql_connect()` zavolána podruhé se stejnými parametry, nebude vytvořeno nové spojení, ale použije se stávající se stejným identifikátorem spojení. Spojení se serverem bude ukončeno automaticky po skončení běhu skriptu, nebude-li dříve zavolána funkce `mysql_close()`. Pokud se navázání s databází nepodaří, upozorníme uživatele chybovým hlášením „Nelze navázat spojení s databází“.

#### **mysql\_select\_db**

**bool mysql\_select\_db (string jmeno\_databaze, resource spojeni)**

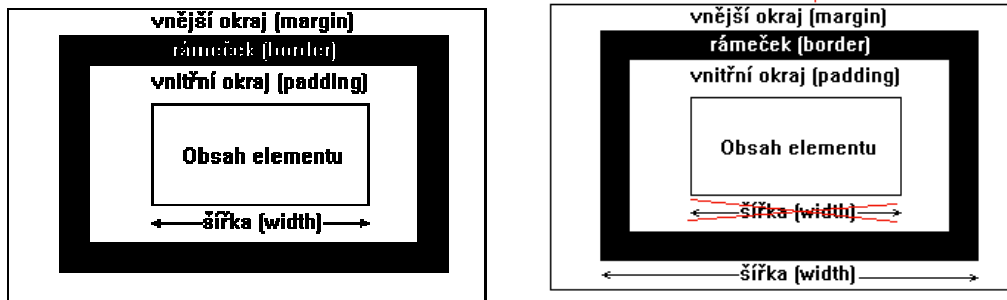
Funkce `mysql_select_db()` nastaví jako aktivní tu databázi v serveru, která je spojená se specifickým identifikátorem `spojeni`. Pro spojení použijeme proměnnou `$db`, ve které je uložen identifikátor spojení a `jmeno_databaze = $dbname`. Funkce vrací `TRUE` při úspěchu, `FALSE` při selhání, upozorníme uživatele chybovým hlášením „Databáze nenalezena“.

@

V PHP existuje operátor řízení chyb, který blokuje výpis chybových zpráv. Je to znak zavináč `@`, který můžete předřadit před jakýkoli výraz, kde získáváte hodnotu. V tomto výrazu jsou pak blokovány jakékoli i kritické chybové zprávy.

### 3 Jaký internetový prohlížeč upřednostnit?

Jakmile je vytvořen soubor CSS, pak by měla být prezentace zobrazena na jakémkoliv médiu shodně, není to tak docela pravda. Nejlépe tento problém vidíme u internetových prohlížečů, existuje skupina významných prohlížečů (Internet Explorer, Mozilla Firefox, Opera), v níž sice jasně dominuje v oblíbenosti Internet Explorer následovaný Firefoxem, ale přesto nelze vytvářet prezentace pro jeden určitý druh prohlížeče. Některé prezentace či aplikace udávají poměr Internet Explorer a Firefoxu 1:1, ale naopak jiné udávají až 98% podíl Internet Explorer, z toho lze usoudit, že prezentace by měla být optimalizována pro co největší počet internetových prohlížečů, tvůrce nemůže předpovídat, jaká skupina uživatelů bude převládat. Zdrojem problémů je rozdílné chápání grafických vlastností v CSS, nejznámější je nejednotnost definovaných rozměrů bloků v prohlížečích. Dva nejvýznamnější internetové prohlížeče Internet Explorer a Firefox, pak Internet Explorer interpretuje šířku elementu jako šířku obsahu, naopak Firefox chápe šířku elementu jako celek, včetně ohraničení a neviditelného okraje za ohraničením, tak jak to také definuje standart CSS. Vzhledem k rozšířenosti Internet Exploreru nelze chybu v Internet Exploreru přehlédnout a uživatelů Internet Exploreru se tyto chyby netýkají, proto není špatný výklad standardů CSS v Internet Exploreru důvodem pro uživatele, aby Internet Explorer nahradili. Důsledkem tohoto konkrétního problému je dvojitá verze definice rozměrů elementů v CSS. Nejčastěji je řešeno CSS hackem, kdy zápis `width: 100px;` je určen pro Firefox a `width: 100px;` je pro Internet Explorer, který je nepochopitelně schopen tento syntakticky nesprávný zápis rozeznat.



Obrázek 7: Chybná interpretace vlastnosti width v IE

## 4 Uživatelské rozhraní

Celý systém správy internetového obsahu lze rozdělit do dvou nejdůležitějších částí. Systém se dělí na obecný systém a redakční systém, elektronický obchod se generuje z redakčního systému automaticky. Systém může obsahovat více částí, které jsou v současnosti nestandardní, stejně tak může obsahovat buď redakční systém, a nebo správu elektronického obchodu. Obecný systém je nutnou součástí každé instalace systému.

### 4.1. Grafická struktura uživatelského rozhraní

Grafická stránka uživatelského rozhraní je vytvořena s pomocí formátování XHTML tabulkami. V designu stránek je použito minimum grafiky, aby bylo dosaženo co nejvyšší rychlosti při jejich načítání. Ostatní grafické prvky jsou tvořeny pomocí CSS stylů, které umožňují velkou variabilitu designu stránek. Barvy webové prezentace jsou navrženy tak, aby stránky byly přehledné a srozumitelné.

Redakční systém má dvě podoby, které se liší právy uživatelů. Jedná se o uživatele administrátora a o uživatele distributora. Nemůžeme si dovolit, aby distributor měl přístup k citlivým datům nebo je mohl měnit. K tomu nám slouží účet administrátora, který má kompletní správu celé aplikace. Administrátor má přístup do celého systému a k databázi. Administrátor má možnost měnit jakákoliv data v databázi.

Copyright © 2005-2006 BAdy.CZ, všechna práva vyhrazena || Powered by JeremyCB

Obrázek 8: Redakční systém po přihlášení administrátora

## 4.2. Přihlášení uživatele

Přihlášením uživatele se rozumí ověření identifikace uživatele a zpřístupnění odpovídajících částí systému. Před samotným zpřístupněním uživatele do systému je nutné ověřit, zda uživatel má oprávnění se systémem pracovat. K jednoznačné identifikaci uživatele slouží přihlašovací jméno a heslo. Jako přihlašovací jméno slouží osmiznakové číslo a heslo, které je po zavedení účtu automaticky generováno. Přihlašovací údaje jsou ukládány do databáze. Porovnají se informace zadané uživatelem s informacemi uložené v databázi. Pokud jsou

informace identické, vrátí funkce hodnotu TRUE, jinak funkce vrací hodnotu FALSE.

The image shows a login form with a dark grey header containing the word 'PŘIHLÁŠENÍ' in green. Below the header is a green background area. There are two white input fields: the first is labeled 'Jméno:' and the second is labeled 'Heslo:'. To the right of the 'Heslo:' field is a small button with the text 'OK'.

*Obrázek 9: Přihlašovací formulář do redakčního systému*

### 4.3. Systémová navigace

Informace systémové navigace jsou uchovány ve stromu, který má několik kořenů. Každá položka menu odpovídá části systému. Při změně uživatele se systémová navigace změní podle uživatelských práv uživatele. Systémová navigace je v obou rozhraní pevně stanovená. Pomocí PHP funkce `include` přiřadíme každé položce systémového menu příslušný soubor, ve kterém se nacházejí příslušná data. Systémová navigace je přizpůsobená tak, aby bylo možné co nejjednodušeji a nejrychleji nalézt příslušná data.



*Obrázek 10: Systémová navigace administrátora a distributora*

#### 4.3.1. Systémová navigace administrátora

Systémová navigace administrátora umožňuje přidávat, odebírat a mazat např. uživatelské účty, aktuality, kontakty, produkty, objednávky....



#### 4.3.1.1. Práce s Osobními daty

Osobní data se skládají z mnoha údajů a uchovávají nám informace o našem účtu. Identifikační číslo je jedinečné v celém systému a pro účet administrátora je neměnné. Ostatní informace o účtu si uživatel může změnit jako např. jméno a příjmení, heslo, email.... Po dokončení a uložení změn údajů nám slouží tlačítko „Editovat“.

#### 4.3.1.2. Nastavení kurzů a účtů

Nastavení kurzů a účtů nám slouží pro nastavení bankovních účtů, poštovních poplatků a měnového kurzu. Systém je nastaven pro práci s dvěma měnami: českou a slovenskou. Pro přidání bankovního účtu vyplníme všechny jeho údaje. Jako typ účtu máme možnost zadat „d“ nebo „z“. Tyto znaky nám určují zda se jedná o účet určený distributorům nebo zákazníkům.

Nastavení poštovních poplatků rozdělujeme na dvě skupiny. První skupina nám určuje jakému uživateli bude zboží doručeno (distributor, zákazník) a druhá skupina nastavuje možnosti, do jaké země a jak velké poštovní poplatky bude muset uživatel uhradit.

Kurzy měn se mění každý den, proto jsou zde zavedeny dva formuláře, které nám nastavují částku kurzu. Kurz USD reprezentuje americký dolar, od kterého se nám automaticky vypočítávají české ceny produktů. Kurz SK reprezentuje slovenskou korunu, která vypočítává slovenské ceny produktů. Tyto ceny se nám promítají do všech produktů v celém systému.

#### 4.3.1.3. Sponzoři

Prvním kořenem je „Seznam“, který nám zobrazuje seznam sponzorů v systému. Pro přidání sponzora použijeme druhý kořen „Přidat“, kde jednoduše vybereme jméno sponzora a tlačítkem „Přidat“ jej přidáme sponzora do systému.

*Pozn.: Pokud chcete přidat sponzora, musí být nejprve přidán jako uživatel!!!*

Pro odebrání sponzora nám slouží poslední kořen „Odebrat“, kde opět jednoduše vybereme jméno sponzora a potvrdíme jej tímto tlačítkem.

#### 4.3.1.4. Uživatelé (distributoři)

Uživatelé systému se rozumí uživatelé distributorů, pro které je velká část systému vytvořena. Kořen „Přidat“ nám po vyplnění všech údajů o distributorovi přidá uživatele do systému. Heslo uživatele je šestimístné a je generováno automaticky pomocí funkce `keygen()`. Máme také možnost přiřadit uživateli sponzora. Pokud již uživatel nemá sponzora, nebudeme sponzora vybírat ze seznamu. Po přidání do systému bude zaslán uživateli email s přihlašovacími údaji do systému.

Pro úpravu dat uživatelského účtu použijeme kořen „Editovat“. Po jednoduchém vybrání uživatele máme možnost měnit jednotlivé údaje. Pro změnu údajů použijeme tlačítko „Editovat“.

Pro odebrání uživatelského účtu využijeme kořen „Odebrat“, kde vybereme název uživatele a po stisknutí tlačítka „Odebrat“ odstraníme uživatelský účet ze systému.

#### 4.3.1.5. Uživatelé (zákazníci)

Obsahuje jeden kořen „Seznam“, který nám zobrazí seznam všech registrovaných zákazníků do elektronického obchodu. Zde je vidět jméno, město, telefon a email. Pro podrobnější informace nebo pro změnu některých údajů klikneme na jméno zákazníka. Zde máme podrobnější údaje o zákazníkovi jako jsou fakturační a doručovací údaje. U zákazníka je možné měnit heslo zákazníka, u fakturačních údajů je možné změnit ulici a město. Po provedených změnách klikneme na tlačítko „Editovat“.

#### 4.3.1.6. Aktuality

Aktuality jsou v tomto systému založeny z důvodu informovanosti spotřebitele. Zadané aktuality se uloží do databáze a poté jsou promítnuty na hlavní stránku.

Aktuality jsou zobrazované v pravé části stránek. Kořen „Přidat“ nám přidá aktualitu na hlavní stránku po vyplnění všech údajů na formuláři.

Pro editování aktualit použijeme kořen „Editovat“, vybereme název aktuality, aktualizujeme data a použijeme tlačítko „Editovat“.

Pro smazání aktualit vybereme kořen „Odebrat“, vybereme aktualitu a pro dokončení smazání použijeme tlačítko „Odebrat“.

### **4.3.1.7. Akce**

Akce se nám zobrazují na titulní stránce po pravé straně. Poskytují informace uživatelům, kdy mají možnost se s námi setkat a na jakém místě.

Pro přidání nové akce použijeme kořen „Přidat“, ve kterém vyplníme potřebné informace. Např. místo, datum, čas, text akce. Po stisknutí tlačítka „Přidat“ se data uloží do databáze a promítnou se na hlavní straně.

Pro upravení použijeme kořen „Editovat“, který nám umožní vybrat název akce a poté vlastní úpravu. Pro dokončení úprav použijeme tlačítko „Editovat“.

Odebírání je opět velmi jednoduché, použijeme kořen „Odebrat“. Vybereme akci, kterou hodláme vymazat a potvrdíme tlačítkem „Odebrat“.

### **4.3.1.8. Kontakty**

Po přidání kontaktu se nám kontakt zobrazí v hlavní stránce. Ke kontaktu můžeme nastavit postavení ve firmě. Např. vedoucí, distributor, péče o zákazníky....

Editace kontaktů provedeme kořenem „Editovat“. Pro editaci kontaktu vybereme jméno a pro změnění údajů použijeme tlačítko „Editovat“.

Mazání kontaktů je velice jednoduché. Stačí jen použít kořen „Odebrat“, kde vybereme jméno kontaktu a potvrdíme tlačítkem „Odebrat“.

### 4.3.1.9. Diskuse

Diskuse žádný kořen neobsahuje. Pokud klikneme na slovo „Kořen“, zobrazí se nám seznam všech diskusí, které jsou založeny. Každá diskuse má své jméno, datum založení a autora. Každou diskusi nebo více diskusí je možné odstranit z databáze pomocí zaškrťovacího tlačítka a po následném potvrzení tlačítka „Smazat“.

Z důvodu vulgárních výrazů byl zaveden systém na odstranění záznamů. Po kliknutí na název diskuse se zobrazí jednotlivé příspěvky. Každý příspěvek má svého autora, text, IP adresu a datum. Pro odebrání příspěvku použijeme opět zaškrťovací tlačítko a potvrdíme tlačítkem „Smazat“.

### 4.3.1.10. Bady

Produkty bady přidáváme pomocí kořenu „Přidat“. Pro korektní přidání produktu do databáze je vhodné vyplnit všechny údaje. Jako první údaj se nám nabízí název produktu, který reprezentuje celý název produktu. Dalším údajem je název produktu pro menu. To znamená název produktu, který se nám zobrazí na hlavní stránce vlevo. Pak si můžeme všimnout údajů složení, popisu, základních komponentů, vlastností a kontraindikací produktu. Do těchto údajů vkládáme specifické vlastnosti produktu. Balení nám určuje v jakém množství je produkt dodáván. Dále narážíme na cenu pro distributory a na cenu pro zákazníky. První cena je určena, jak z názvu plyne, pro distributory a druhá cena je určena pro běžné zákazníky. Obě ceny jsou zadávány v dolarech. Jako poslední můžeme, (ale i nemusíme), přiřadit produktu obrázek stisknutím tlačítka „Procházet“. Pak se zobrazí se dialogové okno, kde si najdeme a následně vybereme vhodný obrázek a potvrdíme tlačítkem „Otevřít“. Obrázek může mít libovolnou velikost, protože při přidávání je použita funkce, která ho automaticky zmenší na požadovaný formát. Když obrázek nemáme k dispozici, přiřadí se produktu obrázek s nápisem „Obrázek není k dispozici“. Pro přidání produktu do systému klikneme na tlačítko „Přidat“.

*Pozn. Minimální výška obrázku je 80px.*

Pro editaci použijeme kořen „Editovat“, ve kterém vybereme produkt, který chceme editovat. Po vybrání produktu se nám kolonky vyplní a můžeme s editací začít. Název produktu a název produktu pro menu nám zajistí název produktu a název produktu pro menu. Popis, vlastnosti, kontraindikace produktu je také možné editovat, ale je tu jedna zásada, které se musíme držet. Tyto texty se zobrazí i se zdrojovým kódem, který je opatřen znaky „<zdroj.kód>“. Je tedy třeba upravovat informace tak, aby nebyl poškozen zdrojový kód např. umazání, vymazání, změna kódu, atd. Jako poslední možnost změny je zařazení produktu do jiné kategorie to zajistíme vybráním příslušné kategorie z nabídky. Pro uložení změněných dat využijeme tlačítko „Editovat“

Odebrání produktu zajistíme kořenem „Odebrat“, kde vybereme název produktu a potvrdíme odebrání tlačítkem „Odebrat“.

### **4.3.1.11. Kosmetika**

Při přidávání produktu do „Kosmetiky“ použijeme kořen „Přidat“, kde vyplníme potřebné informace: název produktu, název produktu pro menu, popis, vlastnosti, balení, ceny a obrázek. Názvem produktu pro menu se rozumí název, který se bude vyskytovat na hlavní stránce v pravé části. Popisem produktu myslíme obecný popis výrobku. Dále je možné vyplnit vlastnosti daného produktu. Tyto vlastnosti jsou detailnější popis výrobku. Při přidávání obrázku můžeme vybrat libovolně velký obrázek, systém si jej sám upraví podle již nastavených hodnot. Po vyplnění všech potřebných údajů potvrdíme přidání výrobku do systému tlačítkem „Přidat“.

*Pozn. Minimální výška obrázku je 80px.*

Pro editování výrobku klikneme na kořen „Editovat“, kde vybereme název výrobku. Po vybrání výrobku se můžeme pustit do úprav. Je možné měnit název výrobku, název produktu pro menu, popis a vlastnosti produktu, balení, cenu a kategorii. U popisu a vlastností produktu je potřeba dávat pozor, jelikož text, který budeme upravovat obsahuje meta znaky. Dbejte na to, aby meta znak zůstal

uzavřený. Jedná se o znak „<br />“. Po změně údajů potvrdíme změnu tlačítkem „Editovat“.

Mazání je velmi jednoduché. Vybereme kořen „Odebrat“, kde vybereme výrobek a pro odebrání použijeme tlačítko „Odebrat“.

### 4.3.1.12. Přístroje

Při přidávání produktu „Přístroje“ použijme kořen „Přidat“, kde vyplníme potřebné informace. Název produktu, název produktu pro menu, popis, vlastnosti, balení, ceny a obrázek. Název produktu pro menu rozumíme název, který se bude vyskytovat na hlavní stránce v pravé části. Popisem produktu myslíme obecný popis výrobku. Dále je možné vyplnit vlastnosti daného produktu. Tyto vlastnosti jsou detailnější popis výrobku. Při přidávání obrázku můžeme vybrat libovolně velký obrázek, systém si jej sám upraví podle již nastavených hodnot. Po vyplnění všech potřebných údajů potvrdíme přidání výrobku do systému tlačítkem „Přidat“.

*Pozn. Minimální výška obrázku je 80px.*

Pro editování výrobku klikneme na kořen „Editovat“, kde vybereme název výrobku. Po vybrání výrobku se můžeme pustit do úprav. Je možné měnit název výrobku, název produktu pro menu, popis a vlastnosti produktu, balení, cenu a kategorii. U popisu a vlastností produktu je potřeba dávat pozor, jelikož text, který budeme upravovat, obsahuje meta znaky. Dbejte na to, aby meta znak zůstal uzavřený. Jedná se o znak „<br />“. Po změně údajů potvrdíme změnu tlačítkem „Editovat“.

Mazání je velmi jednoduché, vybereme kořen „Odebrat“, kde vybereme výrobek a pro odebrání použijeme tlačítko „Odebrat“.

### 4.3.1.13. Ostatní

Při přidávání produktu „Ostatní“ použijme kořen „Přidat“, kde vyplníme potřebné informace. Název produktu, název produktu pro menu, popis, vlastnosti, balení, ceny a obrázek. Název produktu pro menu se rozumí název, který se bude

vyskytovat na hlavní stránce v pravé části. Popisem produktu myslíme obecný popis výrobku. Dále je možné vyplnit vlastnosti daného produktu. Tyto vlastnosti jsou detailnější popis výrobku. Při přidávání obrázku můžeme vybrat libovolně velký obrázek, systém si jej sám upraví podle již nastavených hodnot. Po vyplnění všech potřebných údajů potvrdíme přidání výrobku do systému tlačítkem „Přidat“.

*Pozn. Minimální výška obrázku je 80px.*

Pro editování výrobku klikneme na kořen „Editovat“, kde vybereme název výrobku. Po vybrání výrobku se můžeme pustit do úprav. Je možné měnit název výrobku, název produktu pro menu, popis a vlastnosti produktu, balení, cenu a kategorii. U popisu a vlastností produktu je potřeba dávat pozor, jelikož text, který budeme upravovat obsahuje meta znaky. Dbejte na to, aby meta znak zůstal uzavřený. Jedná se o znak „<br />“. Po změně údajů potvrdíme změnu tlačítkem „Editovat“.

Mazání je velmi jednoduché, vybereme kořen „Odebrat“, kde vybereme výrobek a pro odebrání použijeme tlačítko „Odebrat“.

### **4.3.1.14. Kúry**

„Kúry“ jinak je také nazýváme léčebné kúry, se skládají z různých produktů a jsou zařazeny do etap. Každá etapa má svůj specifický název i význam.

Pro přidávání kúry použijeme kořen „Přidat“. Na první stránce máme možnost nastavit počet etap a počet produktů v jednotlivé etapě. Maximální počet etap je 8. Po zaškrtnutí etap a nastavení počtu produktů v etapě použijeme tlačítko „OK“ pro potvrzení nastavení.

Podle počtu etap a počtu produktů se nám vygeneruje stránka, ve které vyplníme následující údaje. Název kúry a název kúry pro menu, dále vyplníme název etapy, do které vybereme produkt, který má etapa obsahovat. Vyplníme potřebné informace o dávkování, přípravě, všeobecné podmínky a upozornění. V poli „další text“ je možno použít odrážku „•“, kterou v textu vyvoláme znakem

„#“. Pro odřádkování textu použijeme znak „^“. Po vyplnění všech údajů potvrdíme zapsání kúry do systému tlačítkem „Přidat“.

Editování kúry nebo kúr zajistíme kořenem „Editovat“, kde vybereme název kúry. Po vybrání kúry se nám pole vyplní uloženými informacemi, které je možné editovat. Pro uložení změněných dat využijeme tlačítko „Editovat“.

Odebrat kúru je opět jednoduché, použijeme kořen „Odebrat“, vybereme kúru, kterou hodláme odstranit a stiskneme tlačítko „Odebrat“.

### **4.3.1.15. Seznam objednávek distributorů**

Seznam objednávek distributorů nám zobrazuje seznam všech objednávek, které distributoři objednali. Položka obsahuje číslo objednávky, jméno distributora, datum a čas objednávky a celkovou cenu. Po vybrání objednávky se nám zobrazí celá objednávka s údaji o příjemci, o sponzorovi, a s adresou doručení. Ve spodní části se dozvíme, jakou formou bude objednávka doručena. Tabulka na konci objednávky reprezentuje seznam objednaného zboží. Pod touto tabulkou je vyjádřena celková finanční částka objednaného zboží.

Tlačítko „vyřízeno“ použijeme po vyřízení objednávky, které nám zaručí odebrání objednávky z databáze.

### **4.3.1.16. Seznam objednávek zákazníků**

Seznam objednávek zákazníků nám udává seznam o objednávkách učiněných prostřednictvím E-Shopu. Po zobrazení seznamu objednávek máme možnost vidět celou objednávku prostřednictvím kliknutí na požadovanou objednávku. Na objednávce nalezneme veškeré údaje odběratele, dodavatele, objednaného zboží a ceny. Pokud je objednávka odeslána zákazníkovi na určenou adresu, použijeme odkaz „vyřízeno“, který nám zajistí odebrání objednávky ze systému.



### 4.3.2. Systémová navigace distributora

Systémová navigace distributora se velice liší od systémové navigace administrátora. Systém je navržen tak, aby po přihlášení distributora nebyla ovlivněna ostatní data systému. Jednodušeji řečeno distributor má omezená práva používání tohoto systému.

Po přihlášení distributora do systému uživatele přivítáme. Po levé straně je systémová navigace, po pravé straně se nalézá odhlášení ze systému.

#### 4.3.2.1. Osobní data

Slouží k editaci osobních dat daného uživatele. Uživatel má možnost měnit údaje a ve spodní části formuláře může vidět jakého používá sponzora. Pro ukončení a uložení změn uživatele použijeme tlačítko „Editovat“. Pokud nebudou vyplněny všechny údaje označené „\*“, zobrazíme uživateli upozornění, kde není políčko vyplněné a neumožníme mu data uložit. Po opravení nebo doplnění chybějících dat uložíme data do databáze.

#### 4.3.2.2. Objednávka

Objednávka pro distributora je nejdůležitějším prvkem systému. Distributor zde objednává vybrané produkty. Objednávka je rozdělena do 4 kategorií a to kategorie bady, kosmetika, přístroje a ostatní. V každé kategorii jsou příslušné produkty, u kterých je uvedena kolonka pro objednané množství a cena za jeden kus. Ve spodní části máme možnost vyplnit, a tím zaslat objednávku na jinou adresu doručení. Po vyplnění objednávkového formuláře musíme vybrat způsob dodání. Máme možnost dobírkou s platbou při předání, nebo dobírkou s platbou na účet. Pokračujeme tlačítkem „Spočítat“, které nám zajistí náhled objednávky s objednávaným zbožím, dílčí, konečnou cenu, způsob platby a místo dodání. Pokud jsme na objednávce udělali chybu, použijeme tlačítko „zrušit“. Jinak použijeme tlačítko „objednat“, které nám zajistí odeslání objednávky. Uživateli je doručen e-mail, který obsahuje objednávku a další pokyny.

#### 4.3.2.3. Seznam objednávek

Udává nám seznam všech objednávek daného uživatele. Po kliknutí na objednávku uvidíme náhled objednávky, která obsahuje osobní údaje, údaje sponzora, jinou adresu doručení, způsob dodání a objednávané zboží s cenou. Objednávku nelze měnit ani mazat, slouží k přehledu o objednaném zboží.

### 4.4. Odhlášení uživatele

Odhlášení uživatele provedeme jednoduchým kliknutím na odkaz „Odhlášení“, v pravé části stránky, který nám zajistí bezpečné opuštění redakčního systému a návrat na hlavní stranu.



***Obrázek 11: Odhlášení uživatele administrátor a distributor***

Pokud se vyskytne chyba, upozorníme uživatele o této události a sdělíme, jak postupovat dále.

## 5 Závěr

System je schopen plnohodnotně ovládat informace prezentované na internetu, stejně tak i vytvářet elektronický obchod.

Jako programovací jazyk jsem zvolil PHP s databází MySQL. Vytvořený systém je rozdělen na administrátorskou a distributorskou část. Administrátorská část umožňuje spravovat celý systém. Elektronický obchod je generován na základě dat uložených v administrátorské sekci. Distributorská část slouží pro distributory, ve které mají možnost objednávat výrobky. Ten vytváří možnost vkládat do systému popis produktů, jejich cenu a další informace, které jsou nutné k prodeji produktů na internetu. Práce se systémem je umožněna pouze oprávněným uživatelům, proto je nutná databáze, která uchovává informace o uložených uživateli a manipulaci s jejich přihlašování do systému.

Elektronický obchod spravuje informace o produktech. Podobně jako články jsou i produkty zařazovány do databáze. Elektronický obchod obsahuje správu zákazníků a jejich objednávek.

Výsledkem je aplikace umožňující pohodlnou práci s celým systémem. Aplikace je přenositelná mezi platformami Windows a Linux. Aplikace je optimalizována pro prohlížeč Mozilla Firefox, tak i pro Internet Explorer 6.0. Aplikace je plně funkční i pro alternativní prohlížeče www stránek jako jsou Netscape, Opera.

## Literatura

- [1] BAKKEN, S.S. a kol. : Manuál PHP, PHP Documentation Group, 1997 - 2004 (aktualizováno 20.02.2005). URL: <http://mirrors.inway.cz/manual/cs/>.
- [2] JANOVSKEÝ, D. : Jak psát web, [www.jakpsatweb.cz](http://www.jakpsatweb.cz), 2006. URL: <http://www.jakpsatweb.cz>. ISSN 1801-0458
- [3] KOSEK, J. : PHP – Tvorba interaktivních aplikací, 1.vyd., Praha, Grada Publishing, 1998. ISBN 80-7169-373-1.
- [4] PEXA, P. : Tvorba WWW a WAP, 1. vyd., České Budějovice, Kopp, 2001. ISBN 80-7232-161-7.
- [5] ŠKULTÉTY, R. : JavaScript Programujeme internetové aplikace, 1. vyd., Praha, Computer Press, 2001, strany 168 -169. ISBN 80-7226-457-5.
- [6] <http://interval.cz> : Webdegign, 2006. URL: <http://interval.cz/?idcategory=14>. ISSN 1212-8651
- [7] <http://www.linuxsoft.cz> : PHP serial, 2004 – 2006. URL: <http://www.linuxsoft.cz/php/>. ISSN 1801-3805.
- [8] <http://www.owebu.cz> : Články o databázích, Články o CSS, XHTML, 2004 – 2005. URL: <http://www.owebu.cz/>. ISSN 1214-7958
- [9] <http://www.pcsvet.cz> : Programování, 2002 - 2004. URL: <http://www.pcsvet.cz/prog/>. ISSN 1213-6042
- [10] <http://www.zive.cz> : Programování, 2006. URL: <http://www.zive.cz/h/Programovani/>. ISSN 1214-1887

## **Přílohy**

**Příloha A** - CD se zdrojovým kódem celé aplikace.

**Příloha B** - Struktura adresářů a souborů v aplikaci.