



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA PODNIKATELSKÁ
ÚSTAV INFORMATIKY

FACULTY OF BUSINESS AND MANAGEMENT
INSTITUTE OF INFORMATICS

NÁVRH INTERNETOVÝCH STRÁNEK

WEB PAGE DESIGN

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

DITA BLAŽKOVÁ

VEDOUCÍ PRÁCE
SUPERVISOR

doc. Ing. MILOŠ KOCH, CSc.

BRNO 2011

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Blažková Dita

Manažerská informatika (6209R021)

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách, Studijním a zkušebním řádem VUT v Brně a Směrnicí děkana pro realizaci bakalářských a magisterských studijních programů zadává bakalářskou práci s názvem:

Návrh internetových stránek

v anglickém jazyce:

Website Design

Pokyny pro vypracování:

Úvod
Vymezení problému a cíle práce
Teoretická východiska práce
Analýza problému a současné situace
Vlastní návrhy řešení, přínos návrhů řešení
Závěr
Seznam použité literatury
Přílohy

Seznam odborné literatury:

CEDERHOLM, Dan. Flexibilní webdesign. 1.vyd. Brno: Computer Press, a.s., 2006.232 s.

ISBN: 80-251-1018-4.

DOMES, Martin. Tvorba webových stránek. 1.vyd. Brno: Computer Press, a. s., 2006. 192 s.

ISBN: 80-251-0920-8.

GUTMANS, Andi. Mistrovství v PHP 5 . 1.vyd. Brno: Computer Press, a.s.,2005. 520 s. ISBN 80-251-0799-X.

KRUG, Steve. Webdesign - Nenuťte uživatele přemýšlet. 2.vyd. Brno: Computer Press, a.s., 2006. 168 s. ISBN: 80-7226-892-9.

Vedoucí bakalářské práce: doc. Ing. Miloš Koch, CSc.

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2010/2011.

L.S.

Ing. Jiří Kříž, Ph.D.
Ředitel ústavu

doc. RNDr. Anna Putnová, Ph.D., MBA
Děkan fakulty

V Brně, dne 27.05.2011

Abstrakt

Práce se zabývá návrhem a tvorbou internetových stránek a technologiemi k tomuto cíli potřebnými. Vymezuje pojmy podstatné pro orientaci v problematice webových stránek a sumarizuje základní znalosti potřebné pro vytvoření funkční webové prezentace.

Abstract

This thesis describes web sites design and creation and appropriate technologies. It defines the essential terms and summarizes the basic knowledge needed for creation a functional website.

Klíčová slova

Internetová stránka, (X)HTML, CSS, JavaScript, PHP, SQL, optimalizace, webdesign, layout

Klíčová slova

Website, (X)HTML, CSS, JavaScript, PHP, SQL, optimization, web design, layout

Bibliografie

BLAŽKOVÁ, D. Návrh internetových stránek. Brno: Vysoké učení technické v Brně, Fakulta podnikatelská, 2011. 79 s. Vedoucí bakalářské práce doc. Ing. Miloš Koch, CSc..

Čestné prohlášení

Prohlašuji, že předložená bakalářská práce je původní a zpracovala jsem ji samostatně. Prohlašuji, že citace použitých pramenů je úplná, že jsem ve své práci neporušila autorská práva (ve smyslu Zákona č. 121/2000 Sb., o právu autorském a o právech souvisejících s právem autorským).

V Brně dne 3. června 2011

.....
Podpis

Poděkování

Především děkuji svému vedoucímu bakalářské práce doc. Ing. Miloši Kochovi, CSc. za vedení, věcné rady a připomínky při realizaci práce a odbornou pomoc, což přispělo ke zvýšení kvality a odbornosti mé práce.

Dále děkuji divadelnímu souboru Tak Jo za poskytnutí zpětné vazby běžného uživatele při tvorbě stránek a určení tak nejvhodnější cesty ke konečné podobě webových stránek.

Rovněž děkuji svému bratrovi za inspirující připomínky.

A v neposlední řadě děkuji celé své rodině za podporu a poskytnuté zázemí.

Obsah

Úvod.....	10
1 Cíl práce	12
2 Teoretická východiska práce	13
2.1 Internet a jeho služby	13
2.2 Technologie tvorby webových stránek	16
2.2.1 Struktura dokumentu	17
2.2.2 HTML.....	19
2.2.3 XHTML.....	21
2.2.4 XML	22
2.2.5 CSS.....	23
2.2.6 Javascript	25
2.2.7 PHP.....	26
2.3 Základní soubory webových stránek.....	27
2.3.1 Favicon.ico	27
2.3.2 Robots.txt	28
2.3.3 Sitemap.xml.....	29
2.3.4 .htaccess.....	30
2.4 Optimalizace.....	31
2.4.1 Optimalizace pro vyhledávače	31
2.4.2 Optimalizace pro prohlížeče.....	36
3 Analýza problému	38
3.1 Základní informace o zadavateli	38
3.2 Požadavky na webové stránky	39
4 Vlastní návrh řešení	40
4.1 Adresářová struktura	40
4.2 Grafický návrh.....	41
4.3 Zdrojový kód.....	43
4.3.1 Struktura dokumentu	43
4.3.2 ID hlavička	44
4.3.3 ID stranka	47
4.3.4 ID Sloupec	48
4.3.5 ID paticka	54
4.4 Využití PHP a SQL	55
4.5 Administrace	59
4.6 Ostatní soubory	61
4.7 Optimalizace pro prohlížeče a vyhledávače.....	61
4.8 Prezentace na internetu.....	63
4.9 Ekonomické zhodnocení	63
4.9.1 Potenciální náklady a výnosy	66
Závěr.....	68

5	Seznam použitých zdrojů.....	69
5.1	Internetové zdroje.....	69
5.2	Knihy.....	71
6	Seznam použitých zkratek.....	72
	Seznam grafů	73
	Seznam tabulek.....	73
	Seznam obrázků	73
	Seznam rovnic.....	73
	Seznam příloh	73
	Přílohy	74
6.1	Příloha 1: Clenove.php (neaktivní)	74
6.2	Příloha 2: Clenove.php (po najetí myši na sloupec).....	74
6.3	Příloha 3: Odpočet do dalšího představení.....	74
6.4	Příloha 4: Akce.php.....	75
6.5	Příloha 5: Nabídka.php.....	76
6.6	Příloha 6: Fotogalerie.php	77
6.7	Příloha 7: Program.php	78
6.8	Příloha 8: Zdrojové soubory.....	79

Úvod

"Are you receiving this?"¹

Tato věta odstartovala nový věk. V srpnu roku 1969 bylo těchto několik slov posláno z University of California v Los Angeles po vůbec první síti na světě, ARPANET. (1) O tři roky později už Ray Tomlinson vyvíjí první e-mailový program. A v následujícím roce se na svět dostává dnes tolik známý protokol TCP/IP, který se stává základním předpokladem pro vytvoření sítě celosvětových rozměrů. (23)

Zatímco začátkem roku 2000 užívalo internet jen 250 milionů z tehdejších 6 miliard lidí na Zemi (21), v roce 2011 přesáhl počet uživatelů internetu 2 miliardy ze 7 miliard lidí planety. (4) To znamená, že téměř třetina lidské populace se baví, informuje, či pracuje s daty na celosvětové počítačové síti. Příčinou, ale i důsledkem toho se neustále vyvíjí nové technologie. 2. února 2011 byly vyčerpány poslední adresy IPv4 a v současné době se již zavádí nová verze protokolu IPv6. S takovým kvantem uživatelů se nevyhnutelně zkvalitňuje i zobrazování dat ve webových prezentacích. Zrychlují se, jsou interaktivnější a jejich podoba se stává natolik variabilní, že je možné vytvořit jakýkoli design, který vývojáře napadne.

Je těžké odhadnout, kudy se bude v následujících letech tato technologie ubírat. Budou se nadále rozvíjet a zdokonalovat stávající verze programovacích jazyků a technika zobrazování? Nebo jedna z velkých IT společností vystoupí na trh s převratnou a již dlouho studovanou a vyvíjenou holografií? Možná bude od klasických monitorů upuštěno a nahradí je prostorový hologram. Softwarová technika zobrazování dat úzce souvisí s jejím hardwarovým zázemím, proto programovací jazyky, díky nimž uživatel internetu může vidět jen přehledné uživatelské rozhraní, musí držet krok a souběžně s technologiemi hardwaru se rozvíjet.

¹ překl. z ang.: "Došlo ti to?"

V tuto chvíli však stále převládají klasické webové prezentace na dvourozměrných zobrazovacích zařízeních. Proto smyslem následující práce je sumarizovat informace potřebné pro vytvoření a efektivní zprovoznění webové prezentace.

1 Cíl práce

Cílem teoretické části mé práce je sumarizovat informace potřebné při vytváření internetových stránek, jako jsou základní rozdíly mezi programovacími jazyky, přidávané soubory, které neobsahují vlastní kód stránek a informace o optimalizaci pro vyhledávače a prohlížeče.

V praktické části práce vytvářím webové stránky pro divadelní soubor Tak Jo. Zpracovávám vhodný design a zajímavou funkcionalitu a optimalizuji stránky pro dostupné internetové prohlížeče a fulltextové vyhledávače.

2 Teoretická východiska práce

2.1 Internet a jeho služby

Internet pochází ze dvou slov: *inter* a *net*, *mezi sítí*. Původně tak byla označena jedna ze sítí připojených k internetu, ale název se zobecnil a dnes tak můžeme nazývat celosvětovou síť sestávající z velkého množství menších navzájem propojených počítačových sítí. Jednotlivé sítě spojují síťové uzly. Uzlem může být počítač, ale i nemusí - jako síťový uzel může fungovat i např. router. Internet byl vytvořen primárně pro komunikaci a předávání a sdílení dat. Aby to bylo možné, je každému počítači, připojenému k internetu, přidělena unikátní IP adresa, v rámci protokolů TCP/IP. (1) (23)

Až donedávna byl používán výhradně protokol IPv4, který nabízel 2^{32} adres. Ukázalo se, že tento počet není dostačující a proto se v současnosti inovuje protokol na IPv6, který bude nabízet až 2^{128} IP adres. Díky tomu se změní i zápis adres. Zatímco v IPv4 měly adresy jen 32 bitů a bylo tak možné psát v desítkové soustavě, u IPv6 by 128 bitový systém mohl vést k nepohodlně dlouhým zápisům, proto bude adresa v hexadecimální soustavě. (22)

Internet by se netěšil takové popularitě nebýt několika základních služeb, na jejichž základě fungují jednotlivé uživatelské aplikace. Základním předpokladem pro využívání služeb internetu je program zvaný *klient*. Klientem může být ftp klient, poštovní program, www prohlížeč ap. Klient se, pomocí připojení k internetu, spojí se serverem poskytujícím konkrétní službu a uživateli tak zprostředkuje požadovanou funkci. Základními službami jsou WWW, E-mail, IM, VoIP, FTP, DNS, sdílení souborů, připojení ke vzdálenému počítači, aj. (8)

Služba WWW je systém webových stránek využívající obvykle protokol HTTP, což je internetový protokol určený pro výměnu hypertextových dokumentů². (8)

Další neméně známou službou je E-mail, což je elektronická pošta. Pro přenos zpráv využívá protokol SMTP a z druhé strany: pro komunikaci s poštovními

² hypertextový dokument je forma prezentace, kde videa, zvuky, text, akce ap., nejsou umístěny přímo v dokumentu, ale jsou s ním spojeny soustavou vazeb (odkazů)

programy zase protokoly POP3 nebo IMAP. Rozdíl mezi těmito dvěma protokoly je v možnosti připojení několika uživatelů zároveň (IMAP), a způsobem přístupu k poště, kdy POP3 stahuje celé zprávy na lokální disk počítače na rozdíl od IMAP, který nechává poštu na serveru a stahuje pouze hlavičky zpráv. (5) (8)

IM je služba umožňující komunikovat v reálném čase. Navíc většina IM systémů umožňuje uživateli nastavit status přítomnosti (pomocí *away message*) a další funkce. Mezi nejoblíbenější komunikační nástroje patří ICQ, Skype, QIP, Windows Live Messenger, Facebook, Google Talk, ap. (8)

Služba VoIP umožňuje telefonovat prostřednictvím internetu. Technologii VoIP využívá např. Skype, X-Lite, Microsoft NetMeeting, ap. (8)

Další službou internetu je FTP, který využívá stejnojmenný protokol nebo protokol HTTP pro sdílení a přenos souborů mezi počítači a správu účtů internetových stránek. (8) Tato možnost posílání dat se preferuje v případech, kdy je třeba odeslat nebo přijmout velké množství dat, neboť objem přenášených dat není omezen. Pro přístup ke službě není nutné znát speciální příkazy, stačí si nainstalovat jeden z celé řady specializovaných programů (CuteFTP, WS_FTP, SmartFTP, aj.) nebo si vybrat jinak zaměřenou aplikaci s podporou služby FTP (Total Commander, PSPad, NetBeans, Windows Průzkumník, aj.).

DNS je další službou internetu, která využívá stejnojmenný protokol. Na DNS servery se ukládají informace, které umožní uživateli používat snadněji zapamatovatelné názvy jmen počítačů (v alfanumerických znacích). (8) To znamená, že jeho hlavní úlohou je překládat IP adresy uzlů sítě na *doménová jména*. Začátkem každé URL je tak doménové jméno skládající se ze 3 vrstev:

- doména 3. úrovně (www),
- doména 2. úrovně (volitelný řetězec znaků),
- doména nejvyššího řádu (cz). (1)

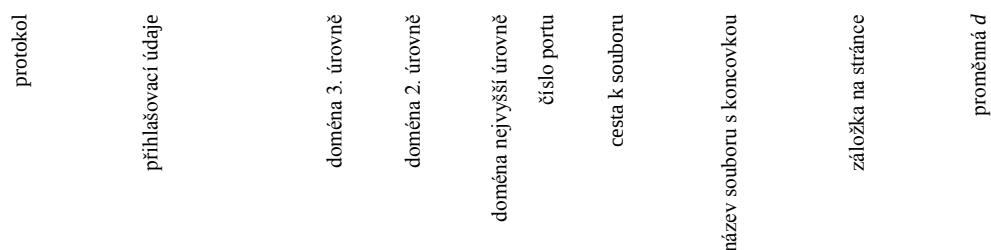
Domén nejvyšší úrovně je mnoho, ale jsou buď tematické (*edu* pro vzdělávací instituce, *com* pro komerci, aj.), nebo státní (*cz* pro Českou republiku, *sk* pro Slovensko, *hu* pro Maďarsko, a další.).

Jednotlivé úrovně domény neboli *subdomény*, se oddělují tečkami a mohou mít až 63 znaků. Celé jméno domény pak může nabýt až 255ti znaků až ve 127 úrovních. Za doménou nejvyšší úrovně se nachází prostor pro cestu a název s koncovkou zobrazovaného souboru v rámci struktury adresářů serveru, na kterém je webová prezentace umístěna.

Na samém začátku, ještě před doménou třetí úrovně se vkládá protokol, přes který je možné o dotazovanou stránku server požádat. Nejčastěji lze nalézt začátek s protokolem *HTTP* v podobě *http://*.

Celý řetězec slouží pro jednoznačnou identifikaci dokumentů na internetu a nazývá se URL. URL má ovšem ještě více možností, než jen alokaci souborů. Pomocí URL lze například předávat různé parametry mezi stránkami v rámci jedné domény - jsou umístěny za názvem stránky a oddělují se otazníkem. Za doménu nejvyššího řádu lze umístit identifikaci požadované služby - číslo portu, který se odděluje dvojtečkou. Nebo někdy na úplném konci je řetězec začínající znakem mřížky (#) - tím se označuje umístění na stránce, záložka. Naopak na začátek URL, za protokol, se vkládají přihlašovací údaje ve tvaru *uživatel:heslo@*. (1) Celé URL pak může vypadat následovně:

http://uzivatel:heslo@www.takjo.com:80/hra_1/detail.php#premiere?d=1



Uvedené URL je absolutní. Naproti tomu existují i URL relativní, které jsou založeny na principu teček. Výchozí umístění je to, kde se nachází aktuální dokument (stránka, která obsahuje odkaz). (1) Je-li třeba odkázat na dokument ve stejném adresáři, použije se pouze název souboru:

divadlo/hry/hra_1.php → divadlo /hry/hra_2.php

hra_2.php

Pokud se však dokument nachází výše, použije se zápis dvou teček oddělených lomítkem od názvu hledaného dokumentu, pro nadřazený adresář:

divadlo /hry/hra_1.php → divadlo /pokladna.html

../pokladna.html

Celé URL se vkládá do adresního řádku ve zvoleném internetovém prohlížeči. Takto sepsané URL odkáže na konkrétní místo v hledané webové stránce. Aby ovšem adresa vůbec odkázala na vytvořené webové stránky, je třeba mít zaregistrovanou doménu, tedy vymezený prostor na některém ze serverů, kam si zakladatel stránek uloží data potřebná pro správné zobrazení. K tomuto účelu se nejčastěji využívají webhostingy, což jsou pronájmy prostorů na cizích serverech.

2.2 Technologie tvorby webových stránek

Hlavním předpokladem pro designově přitažlivé a funkční webové stránky je znalost několika základních programovacích jazyků. Existují více variant jak vytvořit webové stránky, například pomocí nástrojů ze sady MS Office - publikováním Wordovského dokumentu nebo vytvořením webové prezentace specializovanějším nástrojem MS FrontPage. Dále lze webové stránky tvořit přímo na internetu - tuto službu nabízí třeba blog.cz nebo eshop-rychle.cz a mnoho dalších. Úskalím této tvorby je poněkud nízká variabilita a žádná originalita. Vzhledem k požadavkům moderní doby je vhodnější vytvořit si webové stránky od základů. Pro tuto variantu existuje velké množství freeware programů.

Universální editory pro vývojáře pracující s prostým textem jsou například PSPad, NetBean nebo Bluefish. Dobrým pomocníkem mohou být i aplikace instalované jako doplňky k prohlížečům: pro Firefox Web Developer a Firebug, nebo jejich protějšek v Opeře: Web Developer Toolbar, atp. Pro optimalizaci na MSIE se využívá např. IETester a obecněji pro všechny prohlížeče Adobe BrowserLab.

Pokud vývojář programuje webové stránky, které zahrnují i PHP a SQL a nechce je testovat na síti, má možnost kontroly na localhostu díky některému s balíčků PHP Triad, Vertrigo, WAMP nebo XAMPP. Každý z těchto balíčků v sobě mimo jiné spojuje Apache, PHP a MySQL a vytváří tak kompletní prostředí pro webové programátory.

2.2.1 Struktura dokumentu

Specializované programy jako jsou například zmíněný PSPad nebo NetBeans při otevření nového dokumentu automaticky předpřipraví strukturu dokumentu tak, aby vývojář mohl už jen vpisovat kód. V případě obvyklých textových editorů jako je NotePad nebo WordPad, v nichž lze kód rovněž vytvářet, taková možnost neexistuje, proto je dobré si tuto strukturu zapamatovat.

Základním kamenem pro webové prezentace je jazyk HTML. I když se do HTML kódu vkládá PHP, CSS, Javascript aj., podklad obvykle tvoří HTML. Hned nahoře takového dokumentu lze narazit na řádek:

```
<!DOCTYPE html PUBLIC " //W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1 transitional.dtd">
```

Tento řetězec prohlížeči řekne, jakým způsobem vykreslit stránku. Onen způsob je definován v řetězci slovem *transitional*, které může být nahrazeno slovem *strict*, anebo i *quirks* ap. Přičemž se mění i adresa na konci.

Způsob vykreslování *strict* předpokládá čistý HTML dokument bez CSS stylování, které je vloženo do externího dokumentu. *Transitional* povoluje interní stylování. Nestandardní *quirks* režim odpovídá postupům, jež se používaly ve starších prohlížečích. (24) (9)

Přestože se tento řádek může zdát, při programování jednodušších stránek, bezvýznamný, u optimalizace složitějších webových prezentací je naopak velice podstatný. Každý z prohlížečů si výslednou stránku pojme vlastním stylem zobrazování a tak nakonec prezentace perfektně vykreslená v Opeře, je v Internetu

Exploreru úplně rozsypaná. Právě optimalizace pro Internet Explorer bývá nejobtížnější, zvláště pak pro jeho starší verze. V takovém případě se hodí klíčové slovo *quirks*.

Za DOCTYPE následuje specifikace jazyka stránky. Celý další text je tedy uzavřen do párové značky `<html>`. Mezi `<html>` a `</html>` je dokument dále rozdělen na *hlavičku* a *tělo* dokumentu.

Tělo dokumentu nabízí prostor pro vlastní programování. Ohraničuje se párovou značkou `<body>`.

Hlavička, ohraničená párovou značkou `<head>`, v sobě zahrnuje *titulek* (`<title>`), *informace o dokumentu* (`<meta>`), *volání jiného souboru*, se kterým má aktuální dokument vztah (`<link>`) a málo používané stanovení *základu odkazů* (`<base>`). Až na titulek jsou všechny výše zmíněné tagy nepárové. (14)

Nejrozsáhlejší část v hlavičce obvykle zabírají informace o dokumentu, kde lze specifikovat autora stránek, popis stránky, klíčová slova, přístup vyhledávacích robotů, jazyk a znakovou sadu, atd.

Další četnější části jsou odkazy na jiné soubory. U složitějších stránek lze mezi odkazy najít volání externího CSS, Javascriptu, faviconu (ikony webové prezentace) a případné další použité knihovny jako je JQuery.

Finální struktura dokumentu tedy může vypadat takto:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Quirks//EN">
<html>
  <head>
    <meta http-equiv="Content-Language" content="cs">
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
    <meta name="description" content="Oficiální stránky amatérského
                                     divadelního souboru Tak Jo.">
    <meta name="copyright" content="All Rights Reserved,
                                     www.takjo.com">
    <meta name="author" content="Rawn">
    <meta name="keywords" content="Tak Jo, divadlo, soubor, amatérské,
                                     herec, pohádka, představení, hra">
    <meta name="robots" content="all">
```

```

<meta name="googlebot" content="snippet, noarchive">

<link href="favicon.ico" rel="shortcut icon" type="image/x-icon">
<link rel="StyleSheet" href="knihovny/style.css" type="text/css"
  media="screen, projection">

<script src="knihovny/jquery-1.4.2.js" type="text/javascript"
  language="javascript"></script>
<script src="knihovny/style_selektor.js"
  type="text/javascript"></script>
<script src="knihovny/script.js" type="text/javascript"
  language="javascript"></script>

<title>DS Tak Jo</title>
</head>
<body>
  vlastní kód stránky
</body>
</html>

```

Výsledný soubor se ukládá s koncovkou .html nebo .htm neobsahuje-li jazyk PHP. Pokud ano, koncovka se mění na .php. Externí soubory s CSS nebo Javascriptem mají vlastní koncovky: .css a .js. Tyto však neobsahují žádnou předepsanou strukturu, nýbrž se do nich zapisuje, jako do volané knihovny, pouze kód. Stejně tak lze i PHP psát samostatně, čehož se využívá především v případech, kdy vývojář potřebuje v jedné zobrazené stránce spustit PHP funkci vícekrát či ne bezprostředně při úvodním načítání.

2.2.2 HTML

Nejzákladnějším programovacím jazykem pro webové stránky je HTML - hypertextový značkovací jazyk. Hypertextový znamená *odkazovací* a slovem značkovací je míněn způsob vytváření kódu. HTML je non-case sensitivní, to znamená, že při psaní kódu nezáleží na velikosti písmen. Jazyk sestává z množiny značek neboli *tagů*, což jsou vlastně klíčová slova, která mají předdefinovaný význam a mnohé i stylování. Tagy se uzavírají do lomených závorek. Dovnitř některých tagů (uvnitř lomených závorek, za klíčovým slovem) lze vpisovat další kód, tím může být stylování anebo atributy tagu. Atribut znamená vlastnost, a

převážná většina³ atributů má hodnoty. Například tag *a*, což je odkaz, vyžaduje povinný atribut *href*, který musí mít definovanou hodnotu v podobě cesty k odkazovanému souboru.

Hodně z tagů je *párových*, to znamená, že jsou používány ve dvojicích:

<div> BLOK TEXTU </div>

Druhý z párového tagu před sebou nese znak lomítka, což uvozuje právě koncovou část tagu. V rámci zápisu tagů v lomených závorkách se v některých případech vyskytuje i prostor pro volitelné řetězce, například u odkazů:

** ODKAZ **

V první části tagu je vloženo klíčové slovo *href*, za kterým následuje rovnítko a v uvozovkách se očekává absolutní nebo relativní adresa souboru, na který je odkazováno. Text (nebo cokoli jiného, např. obrázek) mezi *<a>* a ** se stává odkazem. Prakticky tedy uživatel ve výsledku spatří pouze slovo *ODKAZ*, na které když klikne, prohlížeč ho přesměruje na domovskou stránku Google.cz.

Tagy se dále dají dělit na blokové elementy (např. *div*) a řádkové (např. *a*). Blokové způsobí zalomení řádku, naopak řádkové tok textu neovlivní. Tagy lze do sebe vkládat, aniž by se však jejich působnost křížila, nelze ale kombinovat elementy blokové a řádkové v hierarchii blokový uvnitř řádkového:

<div> ODKAZ NA GOOGLE </div>



** ODKAZ NA <div>GOOGLE</div> **



**<div> ODKAZ NA GOOGLE</div> **



(10)

³ např. ve formuláři u tlačítka `<input type="submit" value="OK" disabled>` může být mimo atributů *typu* a *hodnoty* použit i atribut *disabled*, který se tak stane sám sobě hodnotou a znefunkční tlačítko

Mimo dělení na párové a nepárové, blokové a řádkové lze ještě elementy rozdělit na logické a fyzické. Fyzické formátování určuje, jak má text vypadat, naopak logické formátování vymezuje význam elementu.

Tabulka 1: Příklad typů formátování

tag	funkce	formátování
big	<i>zmenšení textu</i>	fyzické formátování
u	<i>podtržení textu</i>	fyzické formátování
code	<i>výpis kódu</i>	logické formátování
span	<i>úsek textu</i>	logické formátování

(18)

2.2.3 XHTML

XHTML je novější forma jazyka HTML. HTML ukočilo svůj vývoj na verzi 4.01 a právě tam navazuje XHTML. V souvislosti s tímto jazykem se lze setkat s koncovkou souboru .xhtml. X na začátku zkratky HTML znamená eXtensible, v překl. rozšiřitelný, což paradoxně neznamená, že XHTML přináší nové metody, postupy či další klíčová slova, ale naopak jen omezení. Zatímco u HTML prohlížeče tolerovaly některé chyby, XHTML striktně vyžaduje tyto nedostatky odstraňovat. Týká se to například zmíněného křížení tagů nebo pravidla, že všechny atributy mají mít hodnoty v uvozovkách. Další změny oproti HTML se týkají následujícího:

- tagy a atributy jsou povinně psány malými písmeny (case sensitivní jazyk),
- nepárové tagy končí lomítkem,
- párové tagy se musí skutečně vyskytovat v párech,
- všechny atributy musí mít hodnotu,
- interní javascript a styly se zapisují jiným způsobem,
- dokument má mít XML prolog (deklarace),
- dokument vyžaduje správný DOCTYPE. (17)

Z výše jmenovaného je třeba upřesnit snad jen odlišný způsob zápisu interního javascriptu a stylu. Místo obvyklé deklarace daného jazyka se v XHTML musí kód javascriptu a CSS ohraničit ještě zakomentovaným řetězcem CDATA:

```
<script type="text/javascript">
    /*  */
        vlastní kód stránky
    /*  */
</script> (17)
```

Vzhledem k četným omezením, které mají za účel donutit vývojáře, aby psal kód korektně a vyvaroval se tak chyb v zobrazování, však XHTML není příliš používaným jazykem. Zvláště když moderní prohlížeče tyto vývojářovy chyby stále tolerují.

2.2.4 XML

Další ze značkovacích jazyků je XML, který je přímým následovníkem jazyka SGML. Co se týče syntaxe, je XML velice podobný HTML, avšak s jedním markantním rozdílem: umožňuje definovat si vlastní tagy. Tento způsob zápisu kódu je přehlednější nejen pro vývojáře, ale i pro vyhledávací služby (Google atp.).

příklad kódu HTML

```
<div class="hra">
    <span class="nazev">Záměna</span>
    <span class="cas">, 90 min</span>
</div>
```

příklad kódu XML

```
<hra>
    <nazev>Záměna</nazev>
    <cas jednotka="min">, 90 min</cas>
</hra>
```

XML se nestaví k datům jako k zaměnitelným řetězcům, ale dává jim význam. Na rozdíl od HTML však XML nemá předdefinované formáty některých tagů, takže je třeba použít CSS stylování.

Na druhé straně od této volnosti je tu několik pravidel, která je třeba dodržovat:

- každý element se musí vyskytovat v páru,
- elementy se nesmějí křížit,
- obsah dokumentu musí být uzavřen v jednom kořenovém elementu,
- veškeré hodnoty atributů musí být ohraničeny uvozovkami,
- element nesmí začínat tečkou, pomlčkou nebo podtržítkem. (10)

2.2.5 CSS

Aby webová prezentace získala požadovaný vzhled, nestačí ovládat výše popsané značkovací jazyky, ale je nutné použít CSS, tedy Cascading Style Sheets, kterýžto jazyk je určen výhradně na stylování dokumentu. (10) CSS je non-case sensitivní.

CSS kód lze vkládat několika způsoby:

- přímo do HTML tagů

```
<div style="font-weight: bold;">TUČNÉ</div>
```

- jako interní styl

```
<div id="tucne">TUČNÉ</div>  
<style type="text/css">#tucne { font-weight: bold; }</style>
```

- do externího souboru

odkaz na soubor style.css v hlavičce dokumentu:

```
<link href="style.css" rel="stylesheet" type="text/css" />
```

v těle dokumentu:

```
<div id="tucne">TUČNÉ</div>
```

v externím souboru style.css:

```
#tucne { font-weight: bold; }
```

Pokud se styl nevloží přímo do tagu, může se použít označení daného tagu pomocí identifikátoru (*id="idecko"*) nebo třídy (*class="trida"*). Rozdíl mezi těmito dvěma značkami spočívá v četnosti jejich použití. Identifikátor se na dané stránce smí použít jen jednou a je tudíž unikátní, zato u třídy se předpokládá frekventovanější užití. Samozřejmě v některých případech toto značkování není nutné, umožňuje-li to kód, stačí se odkazovat přímo na klíčová slova tagů.

Aby CSS vědělo, která část kódu je stylována, používají se tzv. selektory. Odkazuje-li se v CSS kódu na identifikátor, napíše se znak mřížky (#) a hodnota identifikátoru: *#idecko*. Je-li cílem odkazu třída, použije se tečka: *.trida*. Při odkazu na tag je název tagu bez označení: *div*. Selektory lze i hierarchicky konkretizovat, což se provádí pomocí mezer mezi danými selektory: *div .trida* - v tomto případě se CSS zaměří na tag označený *class="trida"*, který se nachází mezi párovým tagem *<div>*. Pokud je potřeba, je možné definovat styl i pro více tagů najednou, v takovém případě se selektory oddělují čárkou: *#idecko, .trida*.

Za vhodnou kombinací selektorů už přichází samotná definice stylu, která je uzavřena do složených závorek, její syntaxe je: *{atribut: hodnota; atribut: hodnota;}*, kdy mezi atributem a hodnotou je nutná dvojtečka a za hodnotou středník.

CSS v neposlední řadě umožňuje i definování tagů pomocí pseudoelementů:

a:link {styl nenavštíveného odkazu}

a:visited {styl navštíveného odkazu}

a:active {styl aktivního odkazu}

a:hover {styl odkazu po přejetí myši}

a:first-line {styl prvního řádku odstavce}

a:first-child {styl prvního elementu ve skupině}

atp. (10)

2.2.6 Javascript

Javascript je skriptovací jazyk, který umožňuje stylování, i programování funkčnosti webu. Javascript lze do stránky vkládat stejným způsobem jako CSS: interně, externě i přímo dovnitř tagů:

odkaz na externí soubor script.js v hlavičce dokumentu:

```
<script language="JavaScript" type="text/javascript" src="script.js" />
```

interně vložený skript (jedna z možností):

```
<script>function ztucni ()  
{ document.getElementById("ztucnit").style.fontWeight = "bold";}</script>  
<div id="ztucnit" onClick="ztucni();">ZTUČNÍ SE</div>
```

skript vložený dovnitř tagu:

```
<div id="ztucnit" onClick="this.style.fontWeight: bold;">ZTUČNÍ SE</div>
```

Pokud javascript není v souboru s koncovkou .js (javascriptový soubor), ohraničuje se klíčovým slovem *script*:

```
<script language="JavaScript" type="text/ javascript"> vlastní kód </script>
```

Javascript má rozsáhlé možnosti a svým způsobem je velice podobný jazyku PHP. Javascriptem lze zapisovat do dokumentu, reagovat na činnost uživatele, měnit styly, atd.

Často se spolu s Javascriptem používá knihovna JQuery, která je díky předdefinovaným funkcím jeho rozšířením. Knihovna JQuery zdokonalila selekci, přidala hodně vizuálních efektů (především oblíbená funkce *animate*) a mimo jiné umožňuje i snadné volání externích PHP funkcí.

2.2.7 PHP

PHP datuje svůj vznik od roku 1996, kdy se zkratka vykládala jako Personal Home Page, od té doby pokročila vpřed a význam PHP se změnil na Hypertext Preprocessor. (10)

Jazyk PHP zajišťuje na stránkách především funkčnost, ale dokáže využít ostatní jazyky i ke stylování. Jeho velkou výhodou je i podpora SQL, díky čemuž je programátor schopen pracovat s databázemi. Kód PHP lze vkládat i do HTML, avšak koncovka souboru, kde se PHP vyskytuje je vždy .php. Kód se uzavírá mezi: `<?php vlastní kód ?>`.

Na rozdíl od výše popisovaných jazyků, PHP má jednu zásadní odlišnost: pracuje na straně serveru. Což může být výhodou, ale i nevýhodou. Nevýhodou je, že PHP kód se při načítání webové prezentace přeloží a tím veškeré jeho fungování končí. Pokud by tedy bylo potřeba např. zaktualizovat výpis z databáze a nebylo by možné použít refresh, musí se zapojit např. javascript - PHP se vloží do externího souboru a javascriptem je pak libovolně volán. Na druhé straně, PHP dokáže skrýt velkou část kódu, právě díky tomu, že se k uživateli dostane jen překlad kódu. To lze využít mnoha způsoby, jedním z nich je např. jednoduché ověřování hesla:

```
<?php
    if ($heslo == "tajne") {echo "Správně!";} else {echo "Špatně!";}
?>
```

kdy uživatel při zobrazení zdrojového kódu stránky uvidí jen "Správně!" nebo "Špatně!", nikoli však celou podmínku s obsaženým heslem.

Tento způsob je použitelný jen na velice jednoduchých stránkách - pro složitější a hlavně souborově četnější webové stránky se pro přihlašování často využívají SESSIONS, které jsou nejsnadněji ovladatelné právě díky PHP.

Vyjmenované jazyky nejsou jedinou možností jak vytvořit funkční a nastylované webové stránky. Zvláště v současné době, kdy se internet stal hlavním zdrojem

informací a tudíž nejsledovanějším médiem současnosti, se programovací jazyky posouvají neustále vpřed. Například VML (Vector Markup Language), který umožňuje vytváření vektorové grafiky přímo na stránce. Flashové aplikace. Nebo další a další knihovny (Raphaël).

2.3 Základní soubory webových stránek

Mezi soubory se samotným kódem webové prezentace, které se na serveru vyskytují s koncovkami .html, .htm., .xhtml, .php, .js, .css, ap., lze najít i další, které výše popisované jazyky neobsahují a přesto jsou nutnou součástí internetových stránek.

2.3.1 Favicon.ico

Nejzákladnějším z nich je *favicon.ico*, což je vlastně malý obrázek, který se uživateli zobrazuje v některých prohlížečích vedle adresního řádku a ve všech prohlížečích v oblíbených položkách vedle názvu.

Favicon není příliš náročný na vytvoření: stačí jakýkoli formát, který se po vytvoření přejmenuje na .ico a jakákoliv velikost, neboť prohlížeč si již obrázek zmenší. Každopádně oficiální rozměry pro ikonu jsou 16px². Protože je ikona tak malá, je vhodné použít co nejjednodušší obrázek v 16 barvách s výrazným tvarem, aby se z faviconu nestala jen barevná skvrnka.

Pro zobrazení ikony se do hlavičky vloží řádek:

```
<link href="favicon.ico" rel="shortcut icon" type="image/x-icon">
```

přičemž atribut *type* je nepovinný. V poslední době některé prohlížeče načítají soubor *favicon* automaticky a to i když na něj není v hlavičce odkazováno a nezáleží ani v jakém formátu je ikona uložena. Jediný význam tohoto zápisu je záměr, kdy vývojář požaduje na každé stránce jinou ikonu.

2.3.2 Robots.txt

Další z přidávaných souborů je *robots.txt*. Slouží k tomu, aby se vyhledávacím robotům zakázal přístup do webové prezentace, nebo na její část. Zákaz přístupu lze provést třemi způsoby, které se liší oblastí působnosti:

- souborem *robots.txt* (pro celý web najednou)
- tagem *meta* v hlavičce (pro každý soubor zvlášť)
- v tagu *a* atributem *rel="nofollow"* (pro konkrétní odkaz)

V případě volby souboru *robots*, je třeba, aby byl umístěn v kořeni adresářové struktury a název a koncovka souboru byly napsány malými písmeny. Obsah souboru pak vypadá velice jednoduše:

User-agent: *

Disallow: /tajne/

První řádek definuje "který robot" a následující "kam nesmí". Takovýchto řádků může být v souboru libovolně, zvláště, pokud hvězdičku (znak pro "vše") nahradí konkrétní robot (Googlebot, Jyrobot, Seznambot, Slurp, Gulliver, atd.). Stejně tak pro jednoho robota lze vyjmenovat více zákazů. Uvedený zákaz (řetězec ohraničený lomítky z obou stran) říká, že je zakázán vstup do složky *tajne*. Pokud by lomítko bylo jen před řetězcem, význam zákazu by se změnil na "zákaz vstupu na stránky, jejichž názvy začínají řetězcem *tajne*". Kdyby naopak zápis vypadal takto: *Disallow: /*, robot nemá přístup nikam. A naopak nevyplněné *Disallow* značí volný vstup. Jestliže je třeba zakázat vstup jen pro některé koncovky souborů (např. *.php*), zápis vypadá následovně: */*.php\$*, kde dolar (\$) značí konec adresy. Existuje i možnost všechny stránky zakázat a jen některé povolit, pro povolení se využívá klíčové slovo *Allow*. (20)

V případě volby zákazu pro jednotlivé stránky pomocí *meta* tagu se do hlavičky vloží řádek:

`<meta name="robots" content="noindex, nofollow">`

Atribut *content* může nabýt hodnot, popř. jejich kombinací:

- all (volný vstup - defaultní hodnota, odpovídá kombinaci: index, follow),
- noindex (obsah stránky nebude indexován),
- index (obsah stránky bude indexován),
- nofollow (odkazy nebudou sledovány),
- follow (odkazy budou sledovány). (20)

2.3.3 Sitemap.xml

Sitemap funguje jako mapa stránek pro vyhledávací roboty, kteří se na strukturálně složitějších webech nemusí dostat všude. Ukládá se do kořenového adresáře a obvykle bývá ve formátu .xml, ale není to podmínkou. Sitemap standardně vypadá takto:

```
<?xml version="1.0" encoding="utf-8"?>
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">
  <url>
    <loc>http://www.takjo.com/program.php</loc>
    <lastmod>2011-05-15</lastmod>
    <changefreq>weekly</changefreq>
    <priority>0.8</priority>
  </url>
</urlset>
```

Tag *urlset* ohraničuje celý dokument. Pro vymezení jednotlivých položek je určen *url*. Uvnitř *url* je povinné *loc*, které definuje adresu stránky (je nutné zadat URL včetně protokolu), a nepovinné *lastmod*, *changefreq* a *priority*, kde první z nich určuje poslední modifikaci stránky, druhý frekvenci změn na stránce (může nabýt hodnot: *always*, *hourly*, *daily*, *weekly*, *monthly*, *yearly*, *never*) a poslední prioritu stránky v rámci serveru (rozsah hodnot je 0.0 - 1.0, kde 1.0 značí nejvyšší prioritu), defaultní hodnota *priority* je nastavena na 0.5.

2.3.4 .htaccess

Stejně jako v předchozích případech se i s tímto souborem lze setkat v kořenovém adresáři. Tečka před řetězcem není umístěna náhodou, tečka před názvem na serveru značí skrytý soubor. Některé servery tento soubor nepodporují, především freehostingy a na některých je třeba tuto funkci povolit.

Htaccess umožní vývojáři nastavit chybové stránky, přesměrování stránek, vypínání a zapínání indexů, omezení přístupu, aj. Pravděpodobně nejatraktivnější funkcí souboru je nadefinování vlastních chybových stránek.

Nejběžnějšími chybami při vyhledávání jsou chyby s číselným kódem v řádu 400 a to především 404. Pokud vyhledávání proběhne v pořádku, vygeneruje se kód 200 a uživateli se zobrazí požadovaná stránka. Možných chyb existuje velké množství, kdy chyby s číselným kódem v řádu 100 jsou informační, 200 o úspěšném zpracování, 300 o přesměrování, 400 chyby na straně klienta, 500 chyby na straně serveru. Nejběžnější chyby a jejich vysvětlení jsou shrnuty v následující tabulce:

Tabulka 2: Nejběžnější kódy chyb

Kód	Název chyby	Popis
400	Bad Request	Požadavek byl syntakticky nesprávně zadán.
401	Unauthorized	Neautentizovaný přístup.
402	Payment required	Rezervováno pro budoucí účely.
403	Forbidden	Přístup odmítnut.
404	Not Found	Požadovaný dokument nebyl nalezen.
500	Internal Server Error	Při zpracování požadavku došlo k blíže nespecifikované chybě.

Chce-li tedy programátor vytvořit vlastní chybovou stránku například pro "Požadovaný dokument nebyl nalezen.", vytvoří nový soubor (např. 404.php) s

požadovaným vzhledem a do souboru .htaccess jen přidá řádek s adresou dané chybové stránky:

```
ErrorDocument 404 http://www.takjo.com/chyby/404.php
```

Kvůli vyhledávačům, které je dobré informovat o neexistující stránce, je dobré vložit, na začátek kódu v souboru 404.php, informaci do hlavičky:

```
<?php
    header ("HTTP/1.1 404 Not Found");
?>
```

(10)

2.4 Optimalizace

2.4.1 Optimalizace pro vyhledávače

Hlavním důvodem, proč umisťovat informace na internet je jejich zveřejnění uživatelům. Protože však v dnešní době je na internetu umístěno více než 4mld. webových stránek, vyvstává otázka, jak uživatele upozornit právě na konkrétní webové stránky. V rámci internetu existují dva legální způsoby: reklama formou odkazů a dobrá pozice ve vyhledávačích. Reklama obvykle bývá placená (SEM⁴ kampaň), proto se programátoři menších stránek zaměřují spíše na druhý způsob. A bitva o přední pozice ve vyhledávačích začíná u dobré optimalizace.

Optimalizace pro vyhledávače, zkratka SEO (Search Engine Optimization), je souhrn doporučení, jak upravit webové stránky pro vyhledávače. Základní body optimalizace se shrnují pod termíny *on-page* a *off-page* faktory:

- údaje v hlavičce,
- volba domény,
- názvy souborů,
- duplicita obsahu,
- zpětné odkazy,

⁴ Search Engine Marketing (sponzorované odkazy, zápis do databází jako jsou firmy.cz nebo mapy.cz, ap.)

- ranky,
- čistota kódu. (11)

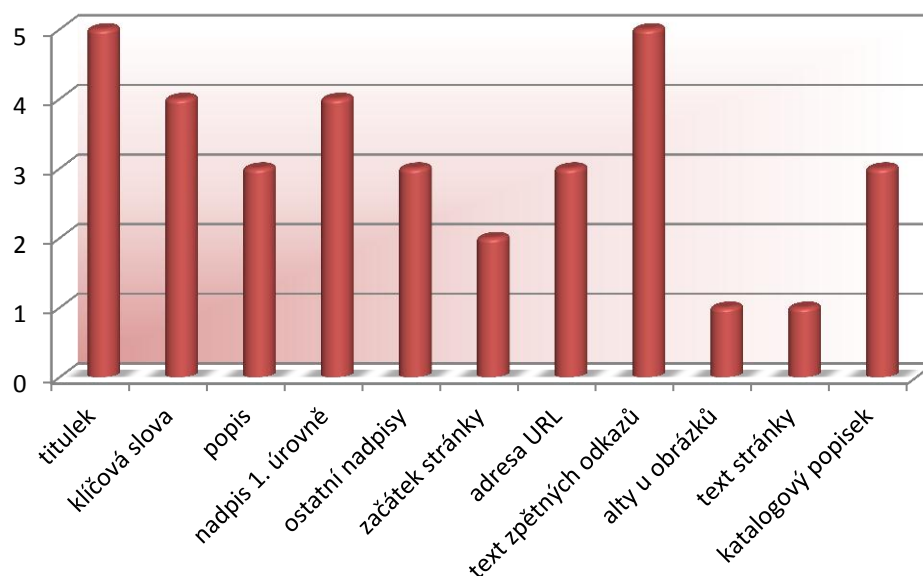
Předchozí vyjmenované patří, až na zpětné odkazy, mezi *on-page* faktory, čili opatření v rámci vytvářené webové stránky.

Existuje více možností, jak si ověřit úroveň optimalizace vytvořených stránek. Nejobecnější nástroj se nachází na adrese <http://seo-servis.cz/source-zdrojovy-kod/>, kde lze stránky prověřit z hlediska kódu stránky (informace v hlavičce, chyby v HTML kódu, sémantiku obsahu, přístupnost). Pro ověření korektnosti kódu jsou však vhodnější stránky <http://validator.w3.org/> (css, html, ap.) a http://www.javascriptlint.com/online_lint.php (javascript). Dalšími pomocníky mohou být stránky:

- <http://www.ranks.nl/tools/spider.html>
měří hustotu klíčových slov na stránce
- <https://adwords.google.com/select/KeywordToolExternal>
počítá frekvenci zadávání klíčových slov ve vyhledávačích (Google)
- <http://martin.m-core.net/project/seo-tools/search-engine-locator/>
počítá pozici ve vyhledávačích Google, Jyxo a Morfeo
- <http://www.jyxo-position.eu/>
počítá pozici ve vyhledávačích Jyxo
- <http://www.svist21.cz/seo/sep.php>
počítá pozici ve vyhledávačích Seznam, Google, Jyxo a Morfeo

Zmiňování vyhledávací roboti, někdy též boti, crawleři, nebo jen roboti, procházejí internetovou sítí a schraňují informace pro vyhledávače, které se dále zpracovávají do databáze (indexování), aby uživatelé na svoje dotazy dostali odpovídající výsledky (webmodul). Roboti začínají většinou v některém z katalogů, najdou odkaz a sledují ho. Nalezenou stránku stáhnou pro zaindexování a vydají se po dalším odkazu na nalezené stránce dál. Z toho vyplývá, že robot najde a stáhne jen ty stránky, na které existuje vnější odkaz (klasický odkaz bez použití javascriptu). (19)

Jakmile je webová stránka zaindexovaná, je třeba už jen zajistit, aby se po zadání dotazu odkaz nezobrazil např. až na sté pozici, kam uživatel obvykle ani nedorazí. Každý webmodul (uživatelské rozhraní umožňující přijmout od uživatele dotaz, zpracovat ho a zobrazit odpověď) řadí výsledky trochu jiným způsobem, obecně však přikládá na internetové stránce tzv. *váhu* na jednotlivé prvky následujícím způsobem (na pomyslné stupnici od 0 do 5, kde 5 je nejvyšší):



Graf 1: Váha elementů na stránce

Největší váhu pro vyhledávače má text odkazů mířících na stránku z jiného serveru (platí u Google a Jyxo) a v každém případě titulek. Dále hodně přihlíží ke klíčovým slovům (Google neváží vůbec) a nadpisům první úrovně. Různě se bere v potaz popis, adresa URL, katalogový popis (zadáva se do spřízněných katalogů) a ostatní úrovně nadpisů. Větší než malou váhu vyhledávače přiznávají začátku stránky (čili řetězci bezprostředně umístěnému za tagem *body*). Malou pozornost věnují hodnotám atributů *alt* u obrázků a samotnému textu stránky. Přestože se posledním elementům přikládá malá váha, stále může mít rozhodující vliv na dobré umístění odkazu na stránky mezi výsledky. (19)

Dalším prvkem, který ovlivňuje pozici mezi výsledky je *Page Rank*, který vyjadřuje věrohodnost a důležitost stránky. Google počítá *Page Rank* podle

zpětných odkazů (odkazů směřujících na stránky z jiného serveru). Jyxo, který poskytuje vyhledávání i Atlasu a Seznamu, Page Rank počítá podle odkazů z domén druhé úrovně vedoucích na stránky. Page Rank stránek lze zjistit např. na následujících odkazech:

- <http://pagerank.jklir.net/>
- <http://rank.isecure.cz/>
- <http://pagerank.yuhu.cz/>
- <http://www.google.com/intl/cs/toolbar/ff/index.html>

(odkaz ke stažení Google Toolbaru, kde mezi jinými lze vypočítat GTPR⁵)

Nejpoužívanějším fulltextovým⁶ vyhledávačem je bezkonkurenčně Google. Ten řadí své výsledky v první řadě podle textu stránky a titulku, dále podle Page Ranku, který počítá podle počtu odkazů na stránky z jiných serverů. Neméně důležitou položkou pro dosažení dobré pozice je text odkazů zvenčí. Bohužel, stejně jako je tomu i u ostatních vyhledávačů, přesné parametry a algoritmus vyhledávání Google, ze zřejmých důvodů, pečlivě tají. (13) Je však alespoň znám vzorec pro počítání Google PageRanku:

Rovnice 1: Vzorec pro vypočítání Google PageRanku

$$PR(takjo.com) = \frac{1-d}{m} + d * \left[\frac{PR(stránka_1)}{C(stránka_1)} + \dots + \frac{PR(stránka_n)}{C(stránka_n)} \right] \quad (15)$$

kde d je *dampening faktor* (nastavený pravděpodobně na 0,85), m je celkový počet zaindexovaných stránek a $C(stránka_n)$ je počet odkazů vedoucích ze stránky_n. (15)

Hodnoty všech stránek podrobených tomuto výpočtu se budou pohybovat těsně nad nulou, což může zmást, neboť při stažení zmíněného Google Toolbaru, má programátor k dispozici rovněž funkci pro výpočet PageRanku a tato zobrazuje výsledky v rozsahu od 0 do 10. PageRank vypočítaný Toolbarem není skutečným

⁵ Google Toolbar PageRank

⁶ fulltextové vyhledávání je způsob vyhledávání informací pomocí databází nebo textových souborů s indexovanými informacemi

v tom smyslu, že sice odráží hodnotu, s níž vyhledávač při pozicování pracuje, avšak hodnota je převedena do přehlednější stupnice. (15)

Vyvstává zřejmá otázka: Jak stránkám Page Rank zvýšit? Při pohledu na vzorec je odpověď jasná, zajistit, aby na stránky směřovalo co nejvíc odkazů ze stránek s vysokým Page Rankem. (15)

Kromě těchto, dá se říci legálních, technik existuje ještě mnoho podvodných, které si pozici mezi výsledky zajišťují zneužíváním znalostí o parametrech vyhledávání. Jsou jimi například:

- Cloaking
(poskytování různého obsahu vyhledávacím robotům a uživatelům)
- Doorway page
(hlavní stránka vytvořená pouze pro roboty)
- Deceptive redirect
(klamné přesměrování, čili uživatel je po vstupu na stránku přesměrován na jinou, která obsahem neodpovídá vyhledávaným parametrům)
- Opakovaná matoucí slova
(do textu jsou uměle a opakovaně přidána klíčová slova)
- Skrytý obsah
(stránka obsahuje skrytou oblast, v níž se vyskytují matoucí klíčová slova)
- Odkazované farmy
(technika spočívá ve vytvoření velkého počtu stránek vytvořených pouze pro roboty, na nichž je uveden odkaz a umístěn algoritmus pro přesměrování na cílovou stránku; takto vytvořená "farma" pak nabývá vysokého hodnocení ve výpočtu Page Ranku)
- Vyhledávací obsahové farmy
(vylepšená verze Odkazové farmy, kde se odkazovací stránky vygenerují ze seznamu nejčastěji hledaných slov a na cílové stránce je, kromě vydělávající reklamy, tento seznam vypsán do prostého textu)

Podvodné techniky jsou detekovány buď automatickým, nebo ručním rozpoznáváním. Kupříkladu Google sestavil formulář, kde lze takovéto stránky nahlásit:

<https://www.google.com/webmasters/tools/spamreport?hl=en&pli=1>

Všechny rozpoznané podvodné techniky jsou penalizovány. Ať už významným propadem v hodnocení stránky nebo úplným bojkotem. (16)

2.4.2 Optimalizace pro prohlížeče

Doposud uvedené informace o optimalizaci se týkaly pouze vyhledávačů. Neméně základnou problematikou však je i optimalizace pro prohlížeče. Rozdíl mezi nimi je zásadní: cíl optimalizování pro vyhledávače spočívá ve zlepšení pozice mezi výsledky dotazu uživatele, kdežto optimalizace pro prohlížeče je otázkou stability designu.

Při vytváření stránek by programátor měl myslet na validitu kódu. Kód je validní právě tehdy, je-li v souladu s technickými pravidly pro psaní daného značkovacího jazyka. (6) Ani zcela validní kód však není zárukou úspěchu. Potíže spočívají i v pojetí některých stylových vlastností prvků v různých prohlížečích. Problematické oblasti zahrnují výpočet šířky prvku, s tím související *marginy*, anebo např. *z-indexy* při absolutním pozicování. Obecně nejmarkantnější rozdíl bývá mezi MSIE a ostatními prohlížeči. Pro ulehčení práce s optimalizací bylo vyvinuto několik způsobů, jak design sjednotit pro všechny prohlížeče.

První z nich je již zmíněná volba *DOCTYPE* na začátku dokumentu. Kromě korektního kódu (kterémužto požadavku není vždy snadné vyhovět) je to jedno z mála oficiálních řešení.

Další možností je využití tzv. CSS hacků. Jsou to vlastně techniky využívající chyb v prohlížečích. CSS hacků existuje velké množství a jejich podoba i počet se neustále mění. Důvodem toho je zdokonalování prohlížečů a jejich pravidelný

upgrade. Nejběžnější technika z rodiny CSS hacků pro sjednocování designu pro různé prohlížeče spočívá v úpravě klíčových slov jazyka CSS, kdy se například přidáním podtržítka (_) před atribut způsobí, že atribut přečte jen Internet Explorer. Anebo dodáním řetězce *!important* před středník daného atributu, vlastnost přečte zase jen Firefox. Atd. Nevýhodou CSS hacků je, že nejsou validní.

Technika, která validitu neovlivní a přitom funguje stejným způsobem jako CSS hacky je knihovna `CSS_browser_selector.js`. Knihovna obsahuje překvapivě krátký javascriptový kód, na který se odkazuje v hlavičce dokumentů a v CSS jazyce přidáním krátkého řetězce před daný element.

```
.ie .priklad {background-color: yellow;}  
.ie7 .priklad {background-color: orange;}  
.gecko .priklad {background-color: gray;}  
.win.gecko .priklad {background-color: red;}  
.linux.gecko .priklad {background-color: pink;}  
.opera .priklad {background-color: green;}  
.konqueror .priklad {background-color: blue;}  
.webkit .priklad {background-color: black;}
```

V takto sestaveném kódu bude mít prvek třídy *priklad* růžovou barvu v Internetu Exploreru 7, zelenou v Opeře, černou v Safari, atd.

Návod i kód ke stažení lze nalézt na stránce:

http://rafael.adm.br/css_browser_selector/

Další nesporná výhoda této techniky spočívá v její nadčasovosti - vzhledem k tomu, že se zde nevyužívá chyb prohlížečů, je to způsob časově trvalejší.

3 Analýza problému

Internet, jakožto největší masmédiu na světě představuje obrovský potenciál v rozvoji každé společnosti, ať už zprostředkovávající obchod, poskytující služby nebo výrobky. Prvním krokem k úspěchu je vejít ve známost a k tomu je reklama na internetu perfektním nástrojem. Reklamou však není míněna pouze plošná, textová či jiná reklama, ale základem každé reklamy je vlastní identita, přičemž samotná identita se stává reklamou, kde identitu tvoří webové stránky.

Jen v únoru roku 2011 navštívilo internet 5.857.421 českých uživatelů, kteří si zobrazili 7.245.152.461 webových stránek. (7) A to nejsou zdaleka rekordní počty, neboť návštěvnost ovlivňuje spousta faktorů, které se každým dnem mění - svátky, počasí, ap.

Organizace, pro kterou, v této práci, vytvořím webovou prezentaci, spadá do sektoru nevýdělečných organizací poskytujících služby. Veškeré výnosy vzniklé z poskytování služeb organizace investuje zpět do své činnosti, je tedy existenčně závislá na zákaznících. V takovém případě by bylo výraznou chybou nevyužít možnosti propagace na internetu tou nejzákladnější formou.

3.1 Základní informace o zadavateli

Zadavatelem webových stránek je **Občanské Sdružení Umění Divadelního**, konkrétně amatérský divadelní soubor **Tak Jo** s patnáctiletou tradicí s hlavním působištěm v Brně. Divadelní soubor nemá pravidelný program ani stálé působiště, je proto pro jeho diváky velice důležité vždy včasné a na konkrétním místě informovat o nadcházejících představení. Tímto stálým místem se v posledních sedmi letech staly internetové stránky.

Divadelní soubor funguje v sestavě třinácti členů. V průměru si divadlo dokáže nasmlouvat 1 až 2 představení měsíčně v závislosti na roční době. Repertoár souboru činí 10 divadelních her (5 dětských a 5 dospělých her), přičemž každoročně soubor nazkouší 1 až 2 nové hry a po 7 až 10 letech je hra vypuštěna z

repertoáru. Z každé hry si divadlo přeje prezentovat fotografie, a z některých videa.

Neboť si členové budou stránky sami aktualizovat, ocenili by i administraci webu.

3.2 Požadavky na webové stránky

- Při prvním pohledu na webovou prezentaci by mělo být znát, že uživatel zavítal na stránky divadla.
- Obsah stránek by měl být interaktivní a co nejvíce vzájemně provázaný.
- Celkový vzhled by neměl působit složitě, ale přesto by měl být graficky zajímavý.
- Menu musí obsahovat stávající položky z minulých stránek:
 - Návštěvní kniha
 - Kontakt
 - Nabídka
 - Program
 - Fotogalerie
 - Videogalerie
 - Členové
 - Slovo o souboru
- Nově by do obsahu chtěl soubor zahrnout slevové akce a náhledy do novin a článků, kde se o Tak Jo psalo.
- Na stránce musí být vymezena oblast pro běžící text nebo obdobné okno s informacemi.
- Na viditelném místě by mělo být umístěno logo a název souboru.
- Administrace alespoň pro program a nabídku.

4 Vlastní návrh řešení

4.1 Adresářová struktura

Než jsem vytvořila první složku nebo soubor, rozhodla jsem se, že veškeré pojmenovávání povedu v češtině bez diakritiky. S tímto předsevzetím jsem vytvořila následující adresářovou strukturu:

- chyby
- knihovny
- obrazky
 - clenove
 - fotogalerie
 - hry
- soukrome
- video
 - nahledy
 - prehravac

Kde *chyby* obsahuje chybové stránky a *knihovny* soubory:

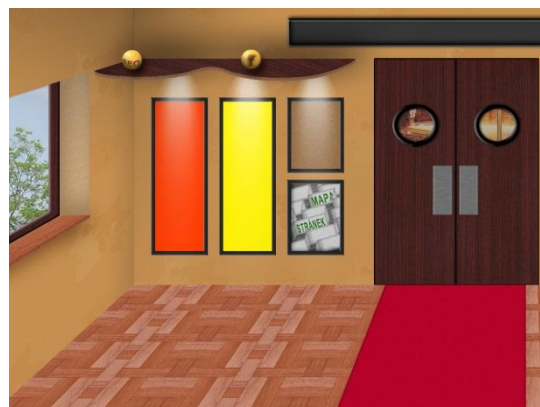
- jquery-1.4.2.js (knihovna JQuery)
- PIE.htc (knihovna pro zakulacené rohy - IE)
- CSS_browser_selector.js (knihovna pro snazší optimalizaci pro prohlížeče)
- script.js (Javascript pro webové stránky)
- style.css (CSS pro webové stránky)

Soubory s vlastním kódem, favicon, .htaccess, atd. jsem nechala v kořenovém adresáři.

4.2 Grafický návrh

Současné grafické podobě předcházelo několik návrhů, které byly postupně zavrženy pro malý potenciál, nezajímavost, nebo jen pro grafickou nevýmluvnost.

První uvedená pracovní verze měla základní myšlenku v imaginárních prostorách divadla. Index by byl náhledem do předsálí, předměty a plakáty v místnosti by sloužily jako menu. Utajený poklop v podlaze by suploval odkaz do administrace,



Obrázek 1: Grafický návrh v1.3

koberec jako plocha pro aktuality a na informačním displeji nade dveřmi by běžela upozornění. Dveře do sálu by představovaly zdvojený odkaz do nabídky, programu a galerií. Okno v levé části mělo sloužit jako dynamický grafický prvek, který by se měnil v závislosti na denní a roční době, popř. svátcích.



Obrázek 2: Grafický návrh v3.0

grafický návrh měl být uplatněn pro index a obdobná verze pak pro další stránky.

První víceméně použitelná verze byla inspirovaná aplikací Total

Další pracovní verze tvořila dřevěnou desku s měnícími se fotografiemi z divadelních her, aktualitami, běžícím upozorněním v horní části a položkami menu. Samotný obsah by se zobrazil po kliknutí na některý z odkazů a to formou animace, kdy by se dřevěná deska rozdělila a "odplula" do stran. Čili uvedený rozpracovaný

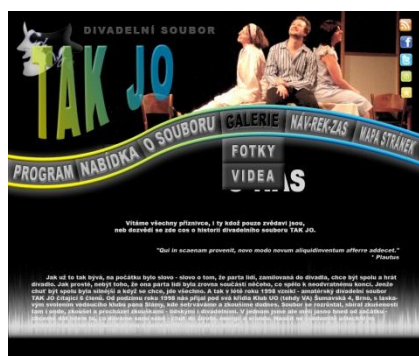


Commander, kdy má uživatel k dispozici 2 okna, do nichž si načte zvolený obsah. Purpurová barva by



Obrázek 3: Grafický návrh v4.0

byla kombinována zlatým a stříbrným písmem a černým pozadím. Layout stránky by tvořila fotografie v hlavičce, informace běžící v pruhu pod ní a tři sloupce: obsah, menu, obsah.



Obrázek 5: Grafický návrh v5.0

Duší předposledního návrhu měl být volný prostor. Menu oddělovalo hlavičku s měnící se fotografií a identifikačními údaji v horní části a zbytek stránky věnovaný obsahu. Barevné schéma mělo základ ve žluté a modré na černém pozadí.

Poslední návrh, který byl i částečně zrealizován se nejvíce podobá současné verzi stránek.

Layout obsahuje hlavičku v horním pruhu, boční sloupec a obsahovou část. Hlavička má k dispozici grafické prvky měnící se v závislosti na denní a roční době. Menu funguje jako kruhová nabídka sestávající z malých okének. Informační pás je nahrazen náhodně se vysunujícím oknem "VĚDĚLI JSTE, ŽE...". Údaje o návštěvnosti, aktuality, atp. jsou zahrnuty v pravém sloupci.

I přes slibný poslední návrh, kdy bylo evidentně třeba ubrat červené, jsem se nakonec přiklonila k modernímu trendu - užšímu okno zarovnanému na střed.



Obrázek 4: Grafický návrh v6.3

Na pozadí stránky se rýsuje divadelní opona ztrácející se s vertikálním rozměrem stránky. Nad celým webem dominuje název a logo divadelního souboru. Pod ním se nachází klasický dvousloupcový layout s menu v horní liště. Rozložení na

pravý a levý sloupec, kdy levý je věnován obsahu, jsem ponechala z předchozí verze.



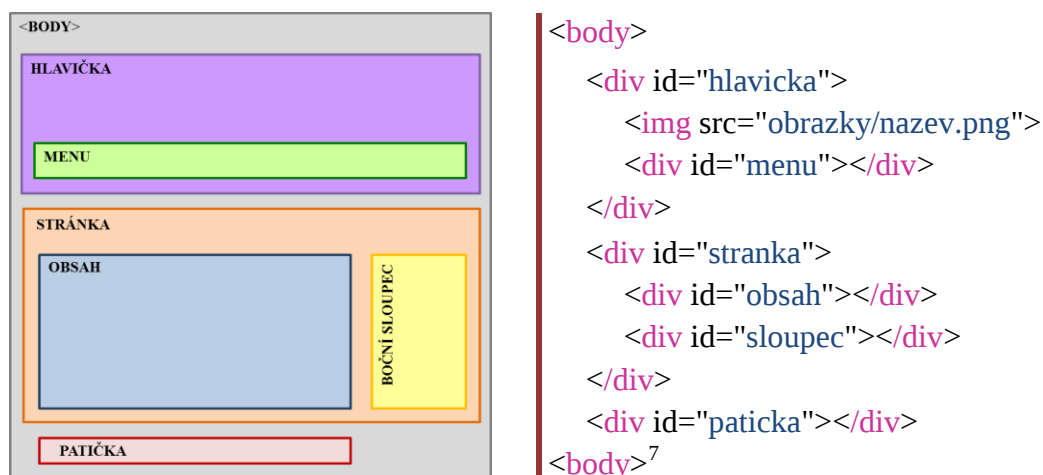
Obrázek 6: Grafický návrh v7.0

4.3 Zdrojový kód

4.3.1 Struktura dokumentu

V celém kódu jsem se snažila separovat CSS od HTML, PHP ap. Čili v každé hlavičce *head* lze najít *link* odkazující na externí soubor.

Samotnou strukturu dokumentu lze znázornit následujícím náčrtem a kódem:



Obrázek 7: Náčrt struktury dokumentu

⁷ ukázka kódu je velice zjednodušená, ale co se týče struktury, odpovídá

4.3.2 ID hlavicka

První *div* v dokumentu nese *id="hlavicka"*. V tomto elementu se skrývá název a logo formou obrázku, který při najetí myši "vyvstane s pozadí" a menu. Název zároveň funguje jako odkaz na index.



Obrázek 9: Název (onMouseOut)



Obrázek 8: Název (onMouseOver)

```
<a href="index.php">  
    
</a>
```

V tagu obrázku definuji akce při jednotlivých událostech myši - při přejetí myší a při kliknutí. Javascript tak zajistí, aby se zaměňovala cesta k obrázku, čímž se obrázky vyměňují. Uživatel již vidí pouze jak nápis "poskakuje".

Do menu jsem vložila 10 položek:

- MÉDIA O NÁS
- AKCE A SLEVY
- NÁVŠTĚVNÍ KNIHA
- KONTAKT
- NABÍDKA
- PROGRAM
- FOTOGALERIE
- VIDEOGALERIE
- ČLENOVÉ
- SLOVO O SOUBORU

Protože by se všechny odkazy do horizontálního menu vedle sebe nevešly, přidala jsem na strany menu šipky, aby si uživatel s menu posouval, přičemž při načtení stránky se mu zobrazí 4 celá tlačítka a 2 půlky po stranách. Zobrazená tlačítka jsou odkazy, které by uživatel neměl přehlédnout (kontakt, nabídka, program).

```
<div id="nabidka">
  
  <div id="menu">
    <ul>
      <li><a href="mediaonas.php">MÉDIA O NÁS</a></li>
      <li><a href="akce.php">AKCE A SLEVY</a></li>
      <li><a href="kniha.php">NÁVŠTĚVNÍ KNIHA</a></li>
      <li><a href="kontakt.php">KONTAKT</a></li>
      <li><a href="nabidka.php">NABÍDKA</a></li>
      <li><a href="program.php">PROGRAM</a></li>
      <li><a href="fotogalerie.php">FOTOGALERIE</a></li>
      <li><a href="videogalerie.php">VIDEOGALERIE</a></li>
      <li><a href="clenove.php">ČLENOVÉ</a></li>
      <li><a href="slovo.php">SLOVO O SOUBORU</a></li>
    </ul>
  </div>
  
</div>

<script>
  $("#sp, #sl").click (
    function () {
      smer = $(this).attr("id");
      kolik = $("#menu ul").css("right");
      kolik = kolik.substring(0, kolik.indexOf("px"));
      kolik = parseInt(kolik);
      if (smer=="sp") {
        if (navigator.userAgent.indexOf("MSIE")>=0) {
          if (kolik>=640) {
            kolik = 900;
            $("#sp").css("visibility", "hidden");
          } else {
            kolik = kolik + 250;
            $("#sl").css("visibility", "visible");}
        } else {
          if (kolik>=700) {
```

```

        kolik = 950;
        $("#sp").css("visibility", "hidden");
    } else {
        kolik = kolik + 250;
        $("#sl").css("visibility", "visible");
    }
    } else {
    if (kolik<=270) {
        kolik = 30;
        $("#sl").css("visibility", "hidden");
    } else {
        kolik = kolik - 250;
        $("#sp").css("visibility", "visible");}}
    $("#menu ul").animate({right: kolik+"px"}, 500);
});
</script>

```

Celý javascriptový kód ve funkci

```

$("#sp, #sl").click (function () {});

```

způsobí posun menu buď nalevo, nebo napravo. Když se menu dostane na poslední položku (např. zde: položka nejvíce vlevo MÉDIA O NÁS), zmizí i šipka v daném směru (např. zde: levá šipka). To znamená, že při každém kliknutí testuji polohu menu, abych zjistila, zda se již nenachází na konci.

Mezi řádky lze najít podmínku, která ošetřuje odlišné počítání Internet Exploreru od jiných prohlížečů:

```

if (navigator.userAgent.indexOf("MSIE")>=0) {}

```

Na závěr je použita funkce *animate* (plynulý pohyb s proměnlivou rychlostí dle fáze posunu), kterou definuje knihovna JQuery načítaná v hlavičce dokumentu.

`$("#menu ul").animate({right: kolik+"px"}, 500);`

sektor

volání funkce

prováděná akce
(posun vlevo o hodnotu
proměnné *kolik*)

dobu provedení (ms)

Aby posuny správně fungovaly je třeba je podpořit CSS kódem a HTML strukturou.

V HTML struktuře *div "nabidka"* drží celé menu pohromadě i s šípkami, *div "menu"* zajišťuje rámeček se zakulacenými rohy, element *ul* je ten, který se posunuje, a tagy *li* jsou samotnými tlačítky nesoucí pozadí.

V CSS je, mimo jiné, nutné specifikovat, u elementu *ul*, vlastnost *position: relative*, protože bez ní by posuny, založené na vlastnosti *right*, nefungovaly. Stejně tak je důležitá vlastnost *overflow: hidden* u elementu *menu*, která skryje přesahující položky menu.

Kosmetickou záležitostí je použití CSS3 na zaoblené rohy. Použila jsem jen základní čtvercové zaoblení 20px na 20px. Pro spolehlivé zobrazení je třeba ošetřit každý z prohlížečů (a verzi prohlížečů) zvlášť:

```
border: 5px solid #8C3333;
border-radius: 20px;           /* Firefox, Safari/Chrome, Opera, IE9 */
-webkit-border-radius: 20px;  /* Chrome */
-moz-border-radius: 20px;     /* Firefox */
behavior: url(knihovny/PIE.htc); /* Internet Explorer */
```

(2)

4.3.3 ID stranka

Druhým velkým blokem v dokumentu je *div* označený jako *stranka*. Tento element však jen drží vnořené *divy* pohromadě. Vnořenými elementy jsou *obsah* a *slopec*. Základními vlastnostmi, které zajišťují podobu layoutu, jsou jen:

```
body {min-width: 800px;}
#stranka {display: table; margin: auto;}
#obsah, #sloupec {display: table-cell; margin: 20px 10px 0px 10px;}
#obsah {width: 580px; padding-bottom: 60px /* kvůli patičce */;}
#sloupec {width: 180px;}
```

Plus, samozřejmě, definice vlastností pro zaoblené rohy atp.

4.3.4 ID Sloupec

Jako rychlý průřez stránkami a zároveň reklamní poutač jsem vedle obsahu přidala další oblast, která se na každé stránce opakuje.

Do obsahu sloupce jsem zahrnula:

- ikony tradičních ikonových funkcí,
- odpočet do dalšího divadelního představení,
- aktuality,
- aktuální datum a přání k svátku,
- návštěvnost stránek,
- odkazy na divadelní hry a členy souboru,
- náhodné video,
- datum poslední aktualizace.

V pořadí odshora, podle důležitosti, jsem nejdříve umístila ikony pro:

- zařazení mezi oblíbené položky,
- nastavení stránky jako domovské,
- rychlý kontakt (e-mail),
- rss,
- přihlášení do administrace,
- odkaz na facebook.

Pod ikonami může uživatel zjistit, kolik zbývá do dalšího divadelního představení. Odpočítávání je ve formátu: dd:hh:mm:ss a řídí se podle tabulky *program* v databázi.

```
<a href="program.php"
  onMouseOver="$(#odpocet, #cas, this).css('textDecoration', 'none');">
<div id="odpocet"
```

```
onMouseOver="window.document.getElementById('cas').innerHTML=
'<?php echo $nazev; ?>'; window.clearInterval(intodp);"
onMouseOut="intodp=window.setInterval('cas()', 1000);">
```

```
<div class="nadpis">Do dalšího divadelního představení zbývá</div>
<span id="cas" onMouseOver="this.style.cursor='pointer';"></span>
```

```
<script type="text/javascript">
function cas() {
    var nyní = new Date();
    var n = nyní.getMonth();
    var konec = new Date(<?php echo (int)$dat[0]; ?>,
                        <?php echo (int)$dat[1]-1; ?>,
                        <?php echo (int)$dat[2]; ?>,
                        <?php echo (int)$cas[0]; ?>,
                        <?php echo (int)$cas[1]; ?>,
                        <?php echo (int)$cas[2]; ?>);
    var k = konec.getMonth();

    function pocet_dni(p) {
        if (p==2) {
            if ((konec.getYear()%4)>0) {return 28;} else {return 29;}
        } else {
            if (p==1 || p==3 || p==5 || p==7 || p==8 || p==10 || p==12)
                {return 31;} else {return 30;}
        }
    }

    if (n==k) {
        k = konec.getDate();
        n = nyní.getDate();
        dny = k - n + 1;
    } else {
        dny = konec.getDate();
        var a = k - 1;
        while (a>=n) {
            if (a==n) {
                dny = dny+(pocet_dni(a)-nyni.getDate());
            } else {
                dny = dny+pocet_dni(a);}
            a--;
        }
    }
    if (dny!=0) {zbyva_dny = dny - 1;} else {zbyva_dny = dny;}
}
```

```

k = konec.getHours();
n = nyní.getHours();
if (k==0) {k = 24;}
if (n==0) {n = 24;}
if (n>k) {zbyva_hodiny = (24-n)+k;} else {
    if(n==k) {zbyva_hodiny = 23;} else {
        zbyva_hodiny = k-n-1;}}

k = konec.getMinutes();
n = nyní.getMinutes();
if (n<=k) {zbyva_minuty = (k-n);} else {zbyva_minuty = (60-n);}
if (zbyva_minuty!=0) {zbyva_minuty--;}
k = konec.getSeconds();
n = nyní.getSeconds();
if (n<=k) {zbyva_sekundy = (k-n);} else {zbyva_sekundy = (60-n);}

if ((zbyva_hodiny+"").length==1)
    {h = "0"+zbyva_hodiny;} else {h = zbyva_hodiny;}
if ((zbyva_minuty+"").length==1)
    {m = "0"+zbyva_minuty;} else {m = zbyva_minuty;}
if ((zbyva_sekundy+"").length==1)
    {s = "0"+zbyva_sekundy;} else {s = zbyva_sekundy;}
zbyva = zbyva_dny+"d:" +h+"h:" +m+"m:" +s+"s";

window.document.getElementById("cas").innerHTML = zbyva;}
intodp = window.setInterval("cas()", 1000);
</script>

</div></a>

```

Další položkou ve sloupci je výpis aktualit, kdy jednotlivé zprávy i s nadpisky jsou ořezány na pevně stanovaný počet znaků a celkově slouží jako odkaz na stránku *aktuality.php*.

V rámci podpory osobního přístupu k uživateli stránek je ve sloupci umístěno i blahopřání k svátku v daném dni. Prakticky je přiřazování jména ke dni realizováno pomocí rozsáhlého dvourozměrného pole.

Pod datem a přáním k svátku uživatel najde statistiku návštěvnosti. Z daného dne a celkový počet návštěvníků. Počítadlo do součtu zahrnuje i roboty.

```

<div id="navstevnost">
  <div class="nadpis">NÁVŠTĚVY</div>

  <?php
    require "setup.php";
    $ip = $_SERVER["REMOTE_ADDR"];
    $cele = $_SERVER["HTTP_USER_AGENT"];
    $dat = date("Y-m-d");
    $cas = date("H:i:s");
    if (strpos($cele, "Netscape") || strpos($cele, "Navigator"))
      {$prohlizec = "Netscape"; }
    else if (strpos($cele, "Firefox")) {$prohlizec = "Firefox"; }
    else if (strpos($cele, "Safari")) {$prohlizec = "Safari"; }
    else if (strpos($cele, "konqueror")) {$prohlizec = "Konqueror"; }
    else if (strpos($cele, "pera")) {$prohlizec = "Opera"; }
    else if (strpos($cele, "MSIE")) {$prohlizec = "Internet Explorer"; }
    else {$prohlizec = "Jiný"; }

    $sql = "SELECT COUNT(ip) AS pocet FROM ip
           WHERE dat='".$dat.'" AND ip='".$ip.'"";
    $exis = mysql_result(mysql_query($sql), 0);
    if ($exis==0) {
      $sql = "INSERT INTO ip(ip, prohlizec, dat, cas, hlavicka) VALUES
             ('".$ip."', '".$prohlizec."', '".$dat."', '".$cas."', '".$cele."')";
      $result = mysql_query($sql) or die(mysql_error());}
    echo "<table>";
    $sql = "SELECT COUNT(distinct ip, dat) AS pocet FROM ip
           WHERE dat='".$dat.'"";
    $dnes = mysql_result(mysql_query($sql), 0);
    echo "<tr><td>Dnes</td><td><b>".$dnes."</b></td></tr>";
    $sql = "SELECT COUNT(distinct ip, dat) AS pocet FROM ip";
    $celkem = mysql_result(mysql_query($sql), 0);
    $celkem = $celkem + 84338;
    echo "<tr><td>Celkem</td><td><b>".$celkem."</b></td></tr>
         </table>";
    MySQL_Close($connect);
  ?>

</div>

```

⁸ před zavedením databáze (k 3. 3. 2011) bylo 8.433 návštěv

Další položkou ve sloupci je 15 čtverečků ve dvou různých barvách. Při přejetí myši se barevná plocha čtverce změní na obrázek - světlé na náhodně vybranou divadelní hru a tmavé na náhodně vybraného člena souboru. Náhodný výběr her a členů se provádí při načtení stránky, čili když uživatel ze čtverce sjede a pak se myši vrátí zpět, zobrazí se mu stejný obrázek. Světlé čtverce fungují zároveň jako odkazy do stránky *nabidka.php* a tmavé na *program.php*.

```
<div id="pexeso">
  <?php
    function rd() {
      $c = rand(0,1);
      if ($c==0) {return "tmave";} else {return "svetle";}
    }
    $bylt = "-";
    $byls = "-";
    for ($i=1; $i<=15; $i++) {
      $naz = rd();
      if ($naz=="tmave") {
        do {$fot = rand(1,18);} while (substr_count($bylt, $fot)>=2);
        $bylt = $bylt."-".$fot;
        echo "<a href='clenove.php'>
              <img alt='Člen souboru' src='obrazky/bok/pexeso-"
              .$naz.
              ".png' onMouseOver='this.src=\"obrazky/bok/pexeso-fotky/"
              .$fot.
              ".png\"' onMouseOut='this.src=\"obrazky/bok/pexeso-"
              .$naz.
              ".png\"'></a>";
      } else {
        do {$fot = rand(1,11);} while (substr_count($byls, $fot)>=2);
        $byls = $byls."-".$fot;
        echo "<a href='nabidka.php'>
              <img alt='Divadelní hra' src='obrazky/bok/pexeso-"
              .$naz.
              ".png' onMouseOver='this.src=\"obrazky/bok/pexeso-hry/"
              .$fot.
              ".png\"' onMouseOut='this.src=\"obrazky/bok/pexeso-"
              .$naz.
              ".png\"'></a>";
      }
    }
  }
```

```

        $prohlizec = $_SERVER["HTTP_USER_AGENT"];
        if (($i%5)==0 && strpos($prohlizec, "pera")>0) {echo "<br>";}
    }
?>
</div>

```

Druhou položkou odspodu je náhodné video. Pro videogalerii i pro náhled v bočním sloupci používám FLV Player MAXI nahraný na serveru spolu s videi.

```

<div class="nadpis">VIDEO</div>
<?php
    $cesta = "video";
    if ($adresar = opendir($cesta)) {
        $a = 0;
        while (false !== ($soubor = readdir($adresar))) {
            if ($soubor != "." &&
                $soubor != ".." &&
                strpos($soubor, ".flv")!="" &&
                $soubor != "reportaz.flv") {
                $soubor = str_replace(".flv", "", $soubor);
                $vypis[$a] = $soubor;
                $a++;
            }
        }
        closedir($adresar);
        $cis = rand(1, (count($vypis)));
        $cis--;
    }
?>

<div id="video">
    <object type="application/x-shockwave-flash"
        data="video/prehravac/player_flv_maxi.swf">
        <param name="movie" value="video/prehravac/player_flv_maxi.swf">
        <param name="allowFullScreen" value="true">
        <param name="FlashVars"
            value="config=config.txt&flv=./<?php echo $vypis[$cis] ?>
                .flv&startimage=./<?php echo $vypis[$cis] ?>.jpg">
    </object>
    <a href="videogalerie.php">Podívejte se na další naše videa...</a>
</div>

```

Konečným údajem je informace o poslední aktualizaci. Tento údaj se neaktualizuje automaticky, v administraci je tato možnost pouze nabídnuta.

4.3.5 ID paticka

Div paticka je poměrně stručný, obsahuje pouze 3 *spany* a jednoduché stylování:

```
<div id="paticka">
  <span>&copy; 2011 Tak Jo</span>
  <span>Všechna práva vyhrazena</span>
  <span>Vytvořila Rawn</span>
</div>

<style>
  #paticka {
    width: 580px;
    margin: auto;
    position: relative;
    bottom: 40px;
    right: 110px;      /* zarovnání na střed divu obsah, tzn. zarovnání na
                        střed stránky a odsud o (šířka divu sloupec +
                        marginy) vpravo */
  }

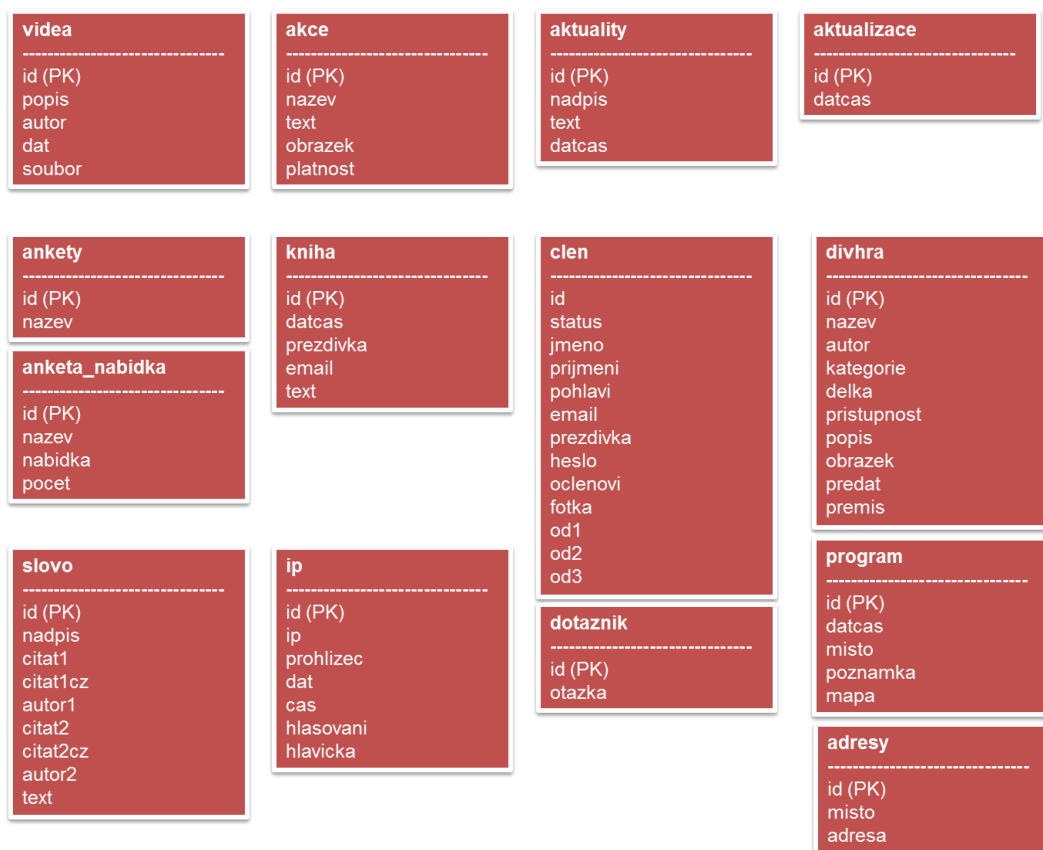
  #paticka span {
    padding: 0px 20px 0px 20px;
    font-size: 1.5ex;
    color: #757575;
  }
</style>
```

Díky takto sestavenému CSS se *div paticka* zarovná na střed *divu obsah* a posune se navíc dovnitř, čímž bude působit jako vnořený, ale bude vždy na spodu nezávisle na obsahu.

4.4 Využití PHP a SQL

Protože by bylo kontraproduktivní a, v konečném důsledku, i nepřehledné, vkládat do každé stránky znovu hlavičku, patičku, boční sloupec a další opakující se bloky, použila jsem příkaz *include* z jazyka PHP. Místo rozsáhlého kódu tak na stránce pouze zavolám externí soubor, kam byl kód přesunut. Když tedy bude třeba změnit údaje např. v patičce, nebude se změna provádět desetkrát na každé jednotlivé stránce, ale pouze jednou v externím souboru.

Díky PHP lze vytvářet nejen zajímavé funkce, ale i ovládat databázi. Vytvořila jsem si databázi s následujícími tabulkami:

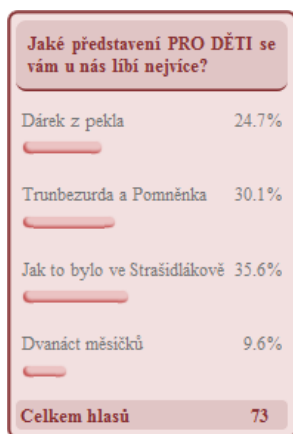


Neboť ošetření vazeb a vztahů mezi daty provádím pomocí PHP, není třeba vytvářet složité popropojovanou síť tabulek cizími klíči.

Tabulka *video* slouží jen pro doplnění informací ke konkrétnímu video souboru, informace se zobrazují na stránce *videogalerie.php*.

Tabulka *akce* shromažďuje data o současných i minulých slevových akcích, přičemž aktuální akce jsou z tabulky vypisovány do stránky *akce.php*.

Aktuality se vypisují do bočního *sloupce* na každé stránce, stejně tak jako údaj poslední aktualizace stránek z tabulky *aktualizace*.



Obrázek 10: Příklad ankety

Tabulky *ankety* a *anketa_nabidka* spolu úzce souvisí, neboť v *ankety* jsou vypsány názvy anket a v *anketa_nabidka* pak jejich zaškrťovací možnosti, počty hlasů a ID z tabulky *ankety* - jedná se tedy v zásadě o cizí klíč. Ankety jsou vypisovány na stránce *akce.php*. Ankety nemají vlastní položku v menu, neboť pro divadelní soubor nejsou výsledky z anket nijak zvlášť důležité, zahrnula jsem je tedy alespoň pod poutavý název položky AKCE A SLEVY.

Z tabulky *kniha* jsou vypisovány příspěvky uživatelů zveřejňované na stránce *kniha.php*.

V tabulce *Člen* lze najít veškeré veřejné údaje o členech divadelního souboru, které jsou vypisovány v souboru *clenove.php*. S těmito údaji je spjata tabulka *dotaznik* obsahující otázky, na něž jednotliví členové odpovídají a jejich reakce jsou uloženy do atributů *od1*, *od2*, atd.

Další podstatnou tabulkou je *divhra* (divadelní hra), v níž obsažené informace slouží pro výpis do *nabidka.php* a v kombinaci s tabulkou *program* a *adresy* navíc do souboru *program.php*. Tabulku *adresy* jsem založila pro snadnější zadávání v administraci, vzhledem k tomu, že divadelní soubor hostuje v některých lokacích opakovaně, je v administraci možnost přidat si takovou adresu (spolu s rozsáhlým odkazem na *mapy.cz*) do databáze, aby ji příště již nemusel zadávat, ale jen vybrat z roletky.

Posledními tabulkami jsou *slovo* a *ip*. První se využívá pro informace vkládané do souboru *slovo.php* a tabulka *ip* shromažďuje údaje o návštěvnicích webových stránek. Záznamy o počtech z *ip* jsou využity do funkce NÁVŠTĚVNOST v

bočním *sloupci*, dále pro omezení v hlasování "jednou denně" a zbytek pro mne, jako programátora.

V běžném provozu stránky pak pomocí javascriptové funkce

```
$.post("vypsat_vzkazy.php", {}, function(data) {  
    $("#list").html(data);});
```

zavolám externí PHP soubor (např. pro vypsaní záznamů z návštěvní knihy):

```
<?php  
    require "setup.php";  
    $a = 1;  
    $sql = "SELECT * FROM kniha ORDER BY datcas DESC";  
    $result = MySQL_Query($sql) or die(mysql_error());  
    while ($data = mysql_fetch_array($result)) {  
        $datc = $data["datcas"];  
        $prez = $data["prezdivka"];  
        $semai = $data["email"];  
        $text = $data["text"];  
        if (($a%2)==0) {$pol = "lichapol";} else {$pol = "sudapol";}  
        echo "<div class='". $pol. "'>  
            <span class='cas'>". $datc. "</span>  
            <div class='udaje'>  
                <span>". $prez. "</span><br>  
                ". $semai. "</div>  
            <div class='oblast'>". $text. "</div>  
        </div>";  
        $a++;  
    }  
    MySQL_Close($connect);  
?>
```

a data se na stránce zaktualizují. Toto je použitelné například po vložení nového příspěvku, kdy nechci nutit uživatele refreshovat. Zmíněné vložení do databáze pracuje obdobným způsobem, kdy se použije SQL příkaz INSERT a v javascriptové funkci se zavolají hodnoty z formuláře:

```

<div id="vzkazinfo"></div>                                <!-- CSS: display: none -->
<div id="list"></div>
<form action="#" method="POST" id="prispevek">
  <input type="button" id="ok" value="Přidat" onClick="pridej()">
  Odesílatel <input class="pole" type="text" name="odesilatel" /><br>
  E-mail <input class="pole" type="text" name="email" /><br>
  <textarea id="text" name="txt"></textarea><br>
</form>

<script>
function pridej() {
  ds = $("input[name=odesilatel]").attr("value");
  ml = $("input[name=email]").attr("value");
  tx = $("textarea[name=txt]").attr("value");
  if (ds=="" || tx=="") {
    alert("Povinná pole (odesílatel, text) nesmí být prázdná!");
  } else {
    $.post("vlozit_vzkaz.php", {odesilatel: ds, email: ml, tex: tx},
    function(data) {
      if(data == "0") {
        $("#vzkazinfo").css("display", "block");
        $("#vzkazinfo").html("POLE ODESÍLATEL A TEXT NESMÍ  
ZŮSTAT PRÁZDNÁ.");
      } else {
        $.post("vypsati_vzkazy.php", {},
        function(data) {
          $("#list").html(data);
        });
        $("body").animate({scrollTop: "350px"}, 0);
        $("input[name=odesilatel]").attr("value", "");
        $("input[name=email]").attr("value", "");
        $("textarea[name=txt]").attr("value", "");
        $("#vzkazinfo").css("display", "block");
        $("#vzkazinfo").html("VZKAZ BYL ÚSPĚŠNĚ PŘIDÁN!");
      }
    });
  }
}
</script>

```

V externím souboru *vlozit_vzkaz.php*:

```

<?php
    require "setup.php";
    $ode = $_POST["odesilatel"];
    $ema = $_POST["email"];
    $tex = $_POST["tex"];
    $nyni = StrFTime("%Y-%m-%d %H:%M:%S", Time());
    $sql = "INSERT INTO kniha(datcas, prezdivka, email, text)
            VALUES('".$nyni."', '".$ode."', '".$ema."', '".$tex."');";
    $result = mysql_query($sql) or die(mysql_error());
    if ($result=="1") {echo "1";} else {echo "0";}
    MySQL_Close($connect);
?>

```

4.5 Administrace

Pro snazší správu stránek, aby do budoucna nevyžadovala tak časté zásahy programátora, jsem na stránky vložila jednoduchý administrační systém pro správu dat.

Do administrace mají povolen vstup jen členové souboru, bylo tedy třeba zajistit přihlašování. Podmíněný vstup jsem zajistila pomocí SESSIONS. Při pokusu vstoupit na zabezpečené stránky se spustí jednoduché ověření:

```

<?
    session_start();
    if(!empty($_SESSION["prihlasen"])) && ($_SESSION["prihlasen"]==1)
    {
        $kdo = $_SESSION["uzivatel"];
        echo $kdo;
    } else {
        echo "0";
    }
?>

```

kde při zjištění přihlášeného uživatele funkce vrátí uživatelské jméno přihlášeného a v opačném případě znak nuly. Při návratu nuly je uživatel vyzván k zadání uživatelského jména a hesla a posléze je spuštěna funkce pro přihlášení:

```

<?
    $fjmeno = $_POST["jmeno"];
    $fheslo = $_POST["heslo"];
    require "setup.php";
    $sql = "SELECT prezdivka, heslo FROM clen
            WHERE prezdivka='".$fjmeno.'";";
    $result = mysql_query($sql) or die(mysql_error());
    $data = mysql_fetch_array($result);
    $jmeno = $data["prezdivka"];
    $heslo = $data["heslo"];
    if(($jmeno==$fjmeno) && ($heslo==$fheslo)) {
        session_start();
        $_SESSION["prihlasen"] = 1;
        $_SESSION["uzivatel"] = $jmeno;
        header("Location: admin_data.php");
        exit;
    } else {
        $spatne = "Přihlašovací údaje nejsou platné.";
        echo $spatne;
    }
?>

```

Poté dostane uživatel práva ke vstupu na zabezpečené stránky a ověřování přihlášení bude, až do odhlášení nebo vypnutí prohlížeče, probíhat na pozadí stránky.

V momentě, kdy uživatel využije tlačítka pro odhlášení, se spustí funkce pro vymazání záznamu v session:

```

<?
    session_start();
    if($_SESSION["prihlasen"]==1) {
        session_destroy();
        echo "1";
    } else {
        echo "0";
    }
?>

```

4.6 Ostatní soubory

Pominu-li soubory s vlastním kódem, vytvořila jsem i několik souborů pro podporu správné funkcionality stránek.

Mezi tyto soubory patří:

- .htaccess
- favicon.ico
- robots.txt
- rss.php
- sitemap.xml

V *.htaccess* jsem sepsala odkazy pro nejběžnější chyby: 400, 401, 402, 403, 404 a 500, na odpovídající vytvořené chybové stránky.

Pro *favicon* jsem využila logo divadelního souboru. Odebrala jsem maskám úkos, reliéf a stín a zmenšila ho do velikosti 16px na 16px. Logo v tomto případě výborně posloužilo, protože sestává jen ze tří barev: černé, bílé a modré.



Obrázek 11: Logo

Pomocí souboru *robots* jsem botům zakázala přístup do složky *soukrome*, která je určena pro soubory, jež si mezi sebou členové divadla vyměňují.

Pro *RSS* jsem zvolila klasickou koncovku *.xml* a so souboru vypsala první RSS zprávy.

A nakonec *sitemap*, kde jsem vypsala všechny stránky, přiřadila jim priority a povětšinou definovala frekvenci změn na týden.

4.7 Optimalizace pro prohlížeče a vyhledávače

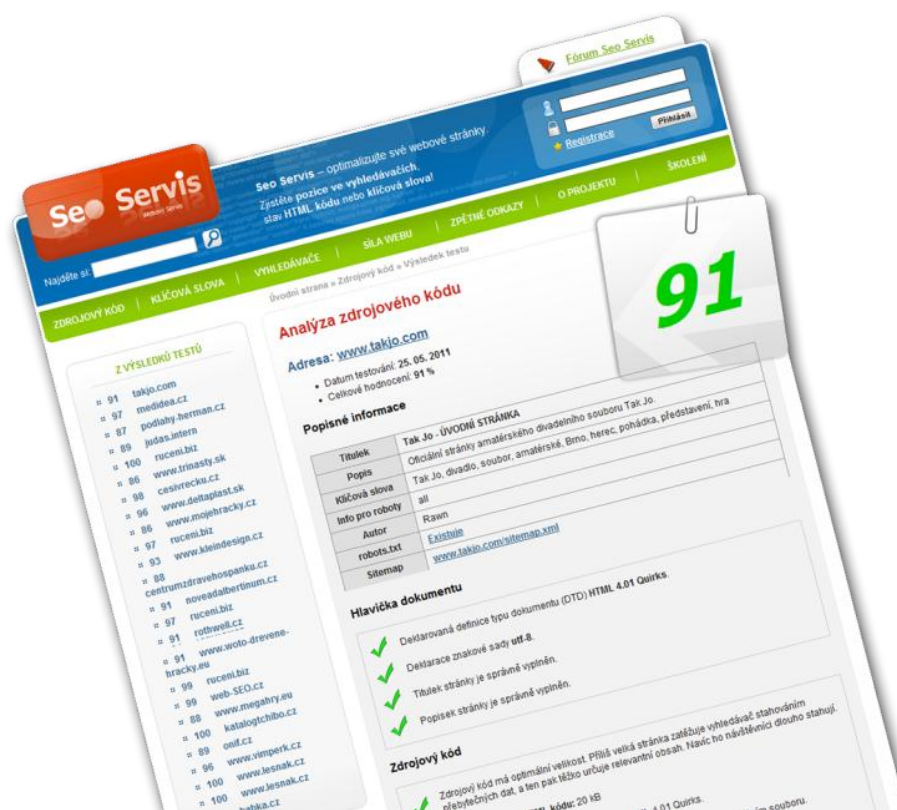
Při optimalizaci pro prohlížeče jsem v *DOCTYPE* použila režim vykreslování *quirks* a zbylé odlišnosti sjednotila pomocí knihovny *CSS_browser_selector.js*.

V rámci optimalizace pro vyhledávače jsem zvolila následující klíčová slova:

Tabulka 3: Klíčová slova zhodnocena na portálu Google AdWords

Klíčové slovo	Konkurence	Místní objem vyhledávání za měsíc
tak jo	0	260
divadlo brno	0,01	14.800
tak	0,01	40.500
jo	0,04	14.800
divadlo	0,01	246.000
soubor	0,01	40.500
amatérské	0,02	9.900
brno	0,17	1.500.000
herec	0,01	22.200
herečka	0,01	9.900
pohádka	0,01	33.100
představení	0,04	5.400
hra	0,08	246.000
hry	0,18	1.000.000

Webovou prezentaci jsem nechala zanalyzovat na stránkách <http://seo-servis.cz/source-zdrojovy-kod> a vyšel mi pěkný výsledek 91%.



Obrázek 12: Výsledek analýzy zdrojového kódu

Web byl koncipován tak, aby do budoucna nepotřeboval další zásahy odborného pracovníka. Administrace umožňuje mazání záznamů, přidávání i jejich úpravy. V nejbližší době na databázi hodlám napojit i soubor RSS, jemuž tak změním koncovku na .php.

4.8 Prezentace na internetu

Současná podoba stránek je již provozu schopná, přesto bych se jí ráda ještě nějakou dobu věnovala. Poměrně-li postupné napojování všech částí obsahu na databázi, je třeba více zintuitivnit administraci. Síť odkazů, jimiž je web protkaný se postupem času zaměří i směrem ven ze stránek. Velká část práce náleží i grafice. Současná podoba je jen jednoduchým základem, takováto strohost by návštěvníky dlouho neuspokojila. Mám v úmyslu napojit stránky osobněji na jednotlivé členy divadelního souboru. V tuto chvíli se na zápatí stránek náhodně generuje postava člena souboru ve formě obrázku - později bych chtěla statické obrázky změnit na video, popřípadě i animaci. Chabý pokus o dynamičnost, na stránce *clenove.php*, nahradí Flash a vůbec celou webovou prezentaci postupně probudím k životu. Cílem je, aby uživatel, který na stránky zavítá, měl touhu si s funkcemi a grafikou pohrát, nejen získat informace, které se mu tím snadněji podsunou. Dobrou optimalizací se sice dobře přilákají návštěvníci a tím i potenciální diváci, ale pokud je na stránkách nic nezaujme, nebudou se vracet. Záměrem je vytvořit web natolik dynamický, aby v návštěvníkovi vznikl pocit, že když ho jeden den nenavštíví, unikne mu důležitá informace.

Červencem divadelní sezóna končí, a tudíž poklesne i návštěvnost stránek. Jejich jednoduchost by tak prozatím neměla mít extrémní dopady a v září letošního roku již webové stránky budou v ostrém provozu se všemi grafickými i funkčními prvky.

4.9 Ekonomické zhodnocení

Divadelní soubor spustil svou první webovou prezentaci v roce 2004, od té doby prošla třemi obměnami, avšak vždy se jednalo o statický a pro běžného uživatele špatně ovladatelný web. Celková návštěvnost čítala od roku 2008 do 3. 3. 2011,

kdy jsem spustila svoji verzi stránek, 8.433 uživatelů. K 26. 5. 2011 návštěvnost stoupla na 9.942 uživatelů⁹:

Tabulka 4: Návštěvnost

Časové rozpětí	Počet návštěvníků
1. 1. 2008 - 2. 3. 2011¹⁰	8.433
3. 3. 2011 - 26. 5. 2011	1.509
1. 1. 2008 - 26. 5. 2011	9.942

Jednoduchým výpočtem lze tedy zjistit průměrnou návštěvnost na den za provozu starých verzí stránek a nové:

Tabulka 5: Průměrná denní návštěvnost

Verze webu	Výpočet	Průměrná denní návštěvnost (zaokrouhleno)
staré verze	$\frac{8433}{366 + 365 + 365 + 64}$	7
nová verze	$\frac{1509}{28 + 30 + 26}$	18

Jak je z výsledných hodnot patrné, průměrná denní návštěvnost se se spuštěním nových stránek zvýšila ze 7 na 18.

Hodnoty návštěvnosti vstupující do výpočtu jsou navýšeny o roboty, kteří stránky pravidelně kontrolují, avšak v předchozích verzích stránek nebylo možné oddělit fyzického uživatele od robota, pro porovnání tedy ve výpočtech s novými hodnotami roboty rovněž nevylučuji. Skutečná návštěvnost vypadá následovně:

⁹ hodnoty návštěvnosti nezahrnují duplicity (denně je možné přidat IP adresu pouze jednou)

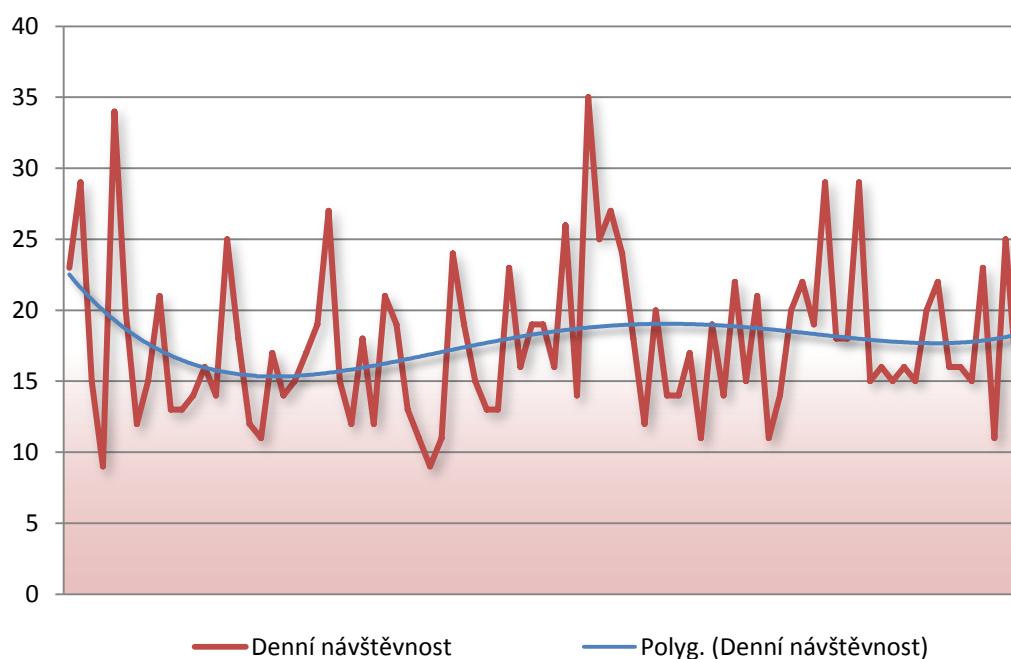
¹⁰ není znám přesný datum spuštění první verze stránek

Tabulka 6: Rozdělení návštěvnosti na uživatele a roboty

Měsíc	Počet robotů	Počet uživatelů	Návštěvnost
březen (od 3. v měsíci)	x	y	501
duben	255	275	530
květen (do 26. v měsíci)	226	252	478
	x+481	y+527	1509

V březnu jsem, bohužel, nezaznamenávala hlavičky návštěvníků, takže u prvních 260ti záznamů nelze určit, zda se jednalo o bota nebo fyzického uživatele.

V následujícím grafu je znázorněna celková denní návštěvnost pro jednotlivé dny (červená křivka) a polynomická spojnice trendu (modrá křivka):



Graf 2: Denní návštěvnost od 3. 3. 2011 do 26. 5. 2011

Divadelní soubor udává ceny vstupného podle typu představení:

Typ představení	cena vstupného	
	student/dítě	dospělý
pro dospělého diváka	90,-Kč	130,-Kč
pro dětského diváka	30,-Kč	60,-Kč

Od září 2011 bude mít divadelní soubor stálou scénu, kde cena pronájmu na neděli činí 750,-Kč a na ostatní dny 7000,-Kč. V plánu je hrát jednou měsíčně v neděli dětské představení a jednou měsíčně v pátek hru pro dospělé. Aby představení nebyla ztrátová, musí je navštívit odpovídající počet diváků. V případě dětského představení 13 - 25 diváků a u dospělých her 53 - 77 diváků, v závislosti na typu vstupenky.

Nejčastěji na dospělá představení chodí páry, čili 2 dospělí a na dětská představení rodič s dítětem, tedy dospělý a dítě. Aby představení nebylo ztrátové, mělo by hru pro dospělé shlédnout alespoň 27 párů a dětské 9 rodičů s dítětem. Při současné návštěvnosti, průměrně 263 fyzických uživatelů za měsíc, by tak stačilo, aby se jen 11% uživatelů rozhodlo vzít partnera a vyrazit do divadla na dospělé představení souboru. Nebo pouze 4% uživatelů, kteří by zavedli dítě na pohádku. V porovnání s předchozí návštěvností, kdy by pohádku muselo shlédnout alespoň 5% rodičů s dítětem a dospělé představení 13% párů z celkového průměrného počtu návštěvníků za měsíc, se jistě jedná o zlepšení, ke kterému přispěla právě nová verze stránek.

4.9.1 Potenciální náklady a výnosy

Pro celkové zhodnocení projektu jsem využila svých dvouletých zkušeností v této oblasti. Použila jsem obvyklou cenovou relaci a vyčíslila částky pro jednotlivé fáze projektu:

Činnost	Částka
Předprojektová konzultace	600,-Kč
Návrh grafického designu	7.600,-Kč
Projektová konzultace designu	400,-Kč
Programování šablon stránek	16.500,-Kč
Programování administrace	8.700,-Kč
Vložení dat do databáze a statického obsahu do stránek	13.400,-Kč
Závěrečná konzultace	zdarma
Drobné úpravy	zdarma
Celkový součet	47.150,-Kč

Celkové náklady tedy činí 47.150,-Kč. Částku jsem ovšem souboru neúčtovala, mohou je tím pádem vyčíslit jako výnos. V položce Dodatečné úpravy zahrnuji další drobná vylepšení funkcí a designu prováděných do začátku následující divadelní sezóny.

Díky administraci nebude třeba platit si externího pracovníka za 4.400,-Kč/měs. na správu dat, jediné výdaje tak případnou na reklamu a stávající webhosting, což činí 210,-Kč/měs. Průměrně na divadelní hry souboru chodí kolem 50ti diváků, nepočítaje objednaná představení, kdy objednavatel nabídne zdarma prostor a zaplatí za představení předem smluvenou částku. Spolu s příspěvky členů si tak divadelní soubor vydělává na vlastní existenci. Webové stránky, jež jsem souboru vypracovala tak nejen ušetří náklady, ale i jako reklama přispějí velkým podílem k návštěvnosti a tím i vyšším ziskům.

Závěr

Cílem mé práce bylo sumarizovat základní teoretické informace potřebné při tvorbě praktické části práce. Aplikovala jsem všechny dosavadní dostupné znalosti a informace, aby webová prezentace, která tím vznikla, byla funkční a s originálním designem, čímž jsem vytvořila dobrý základ pro další vývoj a růst. Nadále budu s projektem spjata správou dat a rozvojem funkcí i grafické složky stránek.

Cením si příležitosti, kterou mi divadelní soubor poskytl, a to, abych mohla na tak velkém projektu uplatnit své teoretické znalosti a konfrontovat je s požadavky, postřehy, připomínkami a názory členů divadelního souboru Tak Jo. Práce na tomto projektu mne obohatila a byla mi velkým přínosem, což mi v budoucnu pomůže vyvarovat se mnoha chyb.

5 Seznam použitých zdrojů

5.1 Internetové zdroje

- (1) *Adaptic* [online]. c2011 [cit. 2011-04-03]. URL. Dostupné z WWW: <<http://www.adaptic.cz/znalosti/slovnicek/url/>>.
- (2) *CSS3.info* [online]. c2009 [cit. 2011-05-22]. Border-radius: create rounded corners with CSS! Dostupné z WWW: <<http://www.css3.info/preview/rounded-border/>>.
- (3) *Internet* [online]. 2008-04-29 [cit. 2011-03-25]. Historie Internetu ve světě a v ČR. Dostupné z WWW: <http://userweb.pedf.cuni.cz/kcj/studenti/heclova/internet.html#psani_s_lova_I>.
- (4) *Finance.cz* [online]. 2011-01-26 [cit. 2011-03-27]. ITU: Světový počet uživatelů internetu vloni dosáhl dvou miliard. Dostupné z WWW: <<http://www.finance.cz/zpravy/finance/296146-itu-svetovy-pocet-uzivatelu-internetu-vloni-dosahl-dvou-miliard/>>.
- (5) *IDOC* [online]. c2007 [cit. 2011-03-30]. Protokoly POP3 / IMAP / Exchange. Dostupné z WWW: <<http://idoc.vsb.cz/cit/uzivatel/email/protokoly/>>.
- (6) *Symbio : Internetová agentura* [online]. c2011 [cit. 2011-05-16]. Validita WWW stránek. Dostupné z WWW: <<http://www.symbio.cz/slovník/validita-www-stranek.html>>.
- (7) *NetMonitor : Sdružení pro internetovou reklamu* [online]. c2011 [cit. 2011-05-22]. Návštěvnost českého internetu v únoru 2011. Dostupné z WWW: <<http://www.netmonitor.cz/tz-navstevnost-ceskeho-internetu-v-unoru-2011>>.
- (8) *Webnode* [online]. 2009-04-11 [cit. 2011-03-30]. Základní služby Internetu. Dostupné z WWW: <<http://itv.webnode.cz/news/zakladni-sluzby-internetu/>>.
- (9) BIEN, Jan. *Mraveniště* [online]. 2006-04-04 [cit. 2011-05-14]. Pracovní režimy prohlížečů. Dostupné z WWW: <<http://interval.cz/podklady/1999-2008/bien/730/rezimy.htm>>.
- (10) GRIMMICH, Šimon. *Tvorba-webu.cz* [online]. c2008 [cit. 2011-04-03]. HTML. Dostupné z WWW: <<http://www.tvorba-webu.cz/html/>>.
- (11) GRIMMICH, Šimon. *Tvorba-webu.cz* [online]. c2008 [cit. 2011-05-15]. SEO - Optimalizace pro vyhledávače. Dostupné z WWW: <<http://www.tvorba-webu.cz/seo/>>.

- (12) HEJL, Zdeněk. *Web o webu : html, css, php, htaccess, seo, reklama* [online]. 2006-04-04 [cit. 2011-05-15]. Úvod do htaccess. Dostupné z WWW: <<http://web.nejmedia.net/htaccess-1-chybove-stranky.htm>>.
- (13) JANOVSKEÝ, Dušan. *Jak psát web : o tvorbě, údržbě a zlepšování internetových stránek* [online]. 2003, 25. října 2010 [cit. 2011-05-16]. Google PageRank. Dostupné z WWW: <<http://www.jakpsatweb.cz/seo/pagerank.html>>.
- (14) JANOVSKEÝ, Dušan. *Jak psát web : o tvorbě, údržbě a zlepšování internetových stránek* [online]. 2003, 25. října 2010 [cit. 2011-05-14]. Hlavička. Dostupné z WWW: <<http://www.jakpsatweb.cz/html/hlavicka.html>>.
- (15) JANOVSKEÝ, Dušan. *Jak psát web : o tvorbě, údržbě a zlepšování internetových stránek* [online]. 2003, 25. října 2010 [cit. 2011-05-16]. O Goole. Dostupné z WWW: <<http://www.jakpsatweb.cz/google.html>>.
- (16) JANOVSKEÝ, Dušan. *Jak psát web : o tvorbě, údržbě a zlepšování internetových stránek* [online]. 2003, 25. října 2010 [cit. 2011-05-16]. Podvodné optimalizační techniky. Dostupné z WWW: <<http://www.jakpsatweb.cz/clanky/podvodne-seo-techniky.html>>.
- (17) JANOVSKEÝ, Dušan. *Jak psát web : o tvorbě, údržbě a zlepšování internetových stránek* [online]. 2003, 25. října 2010 [cit. 2011-05-14]. Syntaxe XHTML. Dostupné z WWW: <<http://www.jakpsatweb.cz/html/xhtmll.html>>.
- (18) JANOVSKEÝ, Dušan. *Jak psát web : o tvorbě, údržbě a zlepšování internetových stránek* [online]. 2003, 25. října 2010 [cit. 2011-05-14]. Úprava textu. Dostupné z WWW: <<http://www.jakpsatweb.cz/html/text.html>>.
- (19) JANOVSKEÝ, Dušan. *Jak psát web : o tvorbě, údržbě a zlepšování internetových stránek* [online]. 2003, 25. října 2010 [cit. 2011-05-15]. Vyhledávače. Dostupné z WWW: <<http://www.jakpsatweb.cz/vyhledavace.html>>.
- (20) JANOVSKEÝ, Dušan. *Jak psát web : o tvorbě, údržbě a zlepšování internetových stránek* [online]. 2003, 25. října 2010 [cit. 2011-05-15]. Zakázání přístupu vyhledávačům. Dostupné z WWW: <<http://www.jakpsatweb.cz/robots-txt.html>>.
- (21) LEVINE, David. *World Population* [online]. 1995-05-26 [cit. 2011-03-27]. World Population Counter. Dostupné z WWW: <<http://www.ibiblio.org/lunarbin/worldpop/>>.

- (22) LUDVIG, Michal. *LINUXZONE* [online]. 2003-02-18 [cit. 2011-03-30]. IPv6 krok za krokem (1). Dostupné z WWW: <<http://www.linuxzone.cz/index.phtml?ids=4&idc=566>>.
- (23) MUSIL, Marek. *Internet* [online]. 2003-07-21 [cit. 2011-03-27]. Dostupné z WWW: <<http://ihistory.webzdarma.cz/>>.
- (24) VÍCHA, Ondřej. Tvorba www stránek [online]. c2010 [cit. 2011-05-14]. DOCTYPE. Dostupné z WWW: <<http://www.stranektvorba.cz/doctype.html>>.

5.2 Knihy

- (25) CEDERHOLM, Dan. Flexibilní webdesign. 1.vyd. Brno: Computer Press, a.s., 2006. 232 s. ISBN: 80-251-1018-4.
- (26) DOMES, Martin. Tvorba webových stránek. 1.vyd. Brno: Computer Press, a. s., 2006. 192 s. ISBN: 80-251-0920-8.
- (27) GUTMANS, Andi. Mistrovství v PHP 5 . 1.vyd. Brno: Computer Press, a.s., 2005. 520 s. ISBN 80-251-0799-X.
- (28) KRUG, Steve. Webdesign - Nenut'te uživatele přemýšlet. 2.vyd. Brno: Computer Press, a.s., 2006. 168 s. ISBN: 80-7226-892-9.
- (29) CASTRO, Elisabeth. HTML, XHTML a CSS : Názorný průvodce tvorbou WWW stránek. Vyd. 1. Brno : Computer Press, a.s., 2009. 438 s. ISBN 978-80-251-1531-2.

6 Seznam použitých zkratek

IT	Information Technology
TCP/IP	Transmission Control Protocol/Internet Protocol
IP	Internet Protocol
WWW	World Wide Web
IM	Instant messaging
VoIP	Voice over Internet Protocol
FTP	File Transfer Protocol
DNS	Domain Name System
SMTP	Simple Mail Transfer Protocol
POP3	Post Office Protocol version 3
IMAP	Internet Message Access Protocol
ICQ	I seek you
QIP	Quiet Internet Pager
NFS	Network File System
GFS	Global File System
AFS	Andrew File System
SSH	Secure Shell
VNC	Virtual Network Computing
URL	Uniform Resource Locator
HTML	HyperText Markup Language
XHTML	Extended HyperText Markup Language
XML	eXtensible Markup Language
PHP	Personal Home Page, Hypertext Preprocessor
CSS	Cascading Style Sheets
SGML	Standard Generalized Markup Language
VML	Vector Markup Language
MSIE	Microsoft Internet Explorer
IE	Internet Explorer
SEO	Search engine optimization
SEM	Search Engine Marketing
GTPR	Google Toolbar PageRank

Seznam grafů

Graf 1: Váha elementů na stránce	33
Graf 2: Denní návštěvnost od 3. 3. 2011 do 26. 5. 2011	65

Seznam tabulek

Tabulka 1: Příklad typů formátování.....	21
Tabulka 2: Nejběžnější kódy chyb	30
Tabulka 3: Klíčová slova zhodnocena na portálu Google AdWords	62
Tabulka 4: Návštěvnost	64
Tabulka 5: Průměrná denní návštěvnost	64
Tabulka 6: Rozdělení návštěvnosti na uživatele a roboty	65

Seznam obrázků

Obrázek 1: Grafický návrh v1.3	41
Obrázek 2: Grafický návrh v3.0	41
Obrázek 3: Grafický návrh v4.0	42
Obrázek 4: Grafický návrh v6.3	42
Obrázek 5: Grafický návrh v5.0	42
Obrázek 6: Grafický návrh v7.0	43
Obrázek 7: Náčrt struktury dokumentu	43
Obrázek 8: Název (onMouseOver).....	44
Obrázek 9: Název (onMouseOut).....	44
Obrázek 10: Příklad ankety	56
Obrázek 11: Logo	61
Obrázek 12: Výsledek analýzy zdrojového kódu.....	62

Seznam rovnic

Rovnice 1: Vzorec pro vypočítání Google PageRanku	34
---	----

Seznam příloh

6.1	Příloha 1: Clenove.php (neaktivní)	74
6.2	Příloha 2: Clenove.php (po najetí myši na sloupec).....	74
6.3	Příloha 3: Odpočet do dalšího představení	74
6.4	Příloha 4: Akce.php	75
6.5	Příloha 5: Nabidka.php	76
6.6	Příloha 6: Fotogalerie.php	77
6.7	Příloha 7: Program.php	78
6.8	Příloha 8: Zdrojové soubory	79

Přílohy

6.1 Příloha 1: Clenove.php (neaktivní)



6.2 Příloha 2: Clenove.php (po njetí myši na sloupec)



6.3 Příloha 3: Odpočet do dalšího představení



6.4 Příloha 4: Akce.php

AKCE A SLEVY

ANKETY

Jaké představení PRO DĚTI se vám u nás líbí nejvíce?

Dárek z pekla24.7%

Trunbezurda a Pomněnka30.1%

Jak to bylo ve Strašidlákově35.6%

Dvanáct měsíčků9.6%

Celkem hlasů73

Jaké představení PRO DOSPĚLÉ se vám u nás líbí nejvíce?

Tři v tom29.2%

Záměna21.3%

Liga proti nevěře14.6%

Ona a On23.6%

Vydrž, miláčku!11.2%

Celkem hlasů89

AKCE A SOUTĚŽE



Slevová akce 2+1. Kupte si 2 vstupenky na naše představení a získáte od nás poukaz pro jednu osobu na jakékoliv naše představení zdarma.

Nárok na poukaz máte při koupi vstupenek nejpozději 48 hodin do daného představení. Při koupi vstupenek těsně před představením poukazy nevydáváme. Chcete-li uplatnit akci 2+1, pošlete nám e-mail na adresu rejza@takjo.com, nebo zavolejte na telefonní číslo 775 17 38 34 a my Vám vstupenky rádi přivezeme (týká se objednávek v rámci Brna).

© 2011 Tak Jo

Vytvořila Rawn

Všechna práva vyhrazena

★🏠🇪🇺📶📺🌐

Do dalšího divadelního představení zbývá

75d:07h:16m:43s

AKTUALITY

Byla spuštěna nová pod...
Spustili jsme pro vás nové stránky v beta provozu. Pomozte nám zdo...

Bylo zprovozněno RSS
Můžete si nachávat zasílat novinky a aktualizace provedené na str...

Do programu bylo přidán...
Chci zobrazit program!

NÁVŠTĚVY

Dnes11

Celkem9906

Najed' myši a odhal kartičku

VIDEO



Podívejte se na další naše videa...

Poslední aktualizace

25. 5. 2011

6.5 Příloha 5: Nabidka.php

NABÍDKA DIVADELNÍCH HER

PŘIPRAVOVANÉ HRY

[Fotogalerie](#)

[Videogalerie](#)

V neblízkosti době toto představení nehrajeme, ale můžete si nás pozvat.



Nevěsta pro Lucipera

Jana Blažková

Výpravná kostýmovaná pohádka o tom, jak to dopadne, když se Luciper rozhodne oženit a ne každému v pekle se líbí s kým.

Představení pro dospělé.
Délka: 80 minut

HRY PRO DOSPĚLÉ DIVÁKY

[Fotogalerie](#)

[Videogalerie](#)

V neblízkosti době toto představení nehrajeme, ale můžete si nás pozvat.



Mr. Miss Olymp

Jana Blažková

Bláznivá situační komedie o tom, že ani na Olympu nebylo všechno tak, jak si všichni myslí.

Představení pro dospělé.
Délka: 70 minut

[Fotogalerie](#)

[Videogalerie](#)

V neblízkosti době toto představení nehrajeme, ale můžete si nás pozvat.



Záměna

Jiří Just

Lechtivá, moderní komedie s hudebními vsuvkami o tom, jak požádat manželku bližního svého a zůstat přitom bližním.

Představení pro dospělé.
Délka: 80 minut

[Fotogalerie](#)

[Videogalerie](#)

V neblízkosti době toto představení nehrajeme, ale můžete si nás pozvat.



Liga proti nevěře

Zdeněk Podskalský

Situační komedie s překvapením o tom, že manželství je hra, ve které je zdravý rozum jen na překážku.

Představení pro dospělé.
Délka: 80 minut

[Fotogalerie](#)

[Videogalerie](#)

V neblízkosti době toto představení nehrajeme, ale můžete si nás pozvat.



ONA A ON, včera, dnes a zítra...

Jana a Dita Blažkovi, Jana Blažková

Tři jednoaktovky v duchu konverzačních komedií o manželství, jaké bylo včera, je dnes, ale bude i zítra.

Představení pro dospělé.
Délka: 80 minut

[Fotogalerie](#)

[Videogalerie](#)

V neblízkosti době toto představení nehrajeme, ale můžete si nás pozvat.



Vydrž, miláčku!

Ray Cooney a John Chapman

Konverzační komedie o tom, co způsobí úmysl zabalit nevěru do norka.

Představení pro dospělé.



Do dalšího divadelního představení zbývá

75d:08h:15m:55s

AKTUALITY

Byla spuštěna nová pod...
Spustili jsme pro vás nové stránky v beta provozu. Pomozte nám zdo...
Bylo zprovozněno RSS
Můžete si nacházet zasílat novinky a aktualizace provedené na str...
Do programu bylo přidán...
Chci zobrazit program!

Dnes je čtvrtek 26. 5.

Vše nejlepší k svátku přejeme Filipovi!

NÁVŠTĚVY

Dnes	11
Celkem	9906

Najed' myši a odhal kartičku



VIDEO



Podívejte se na další naše videa...

Poslední aktualizace

25. 5. 2011

6.6 Příloha 6: Fotogalerie.php

amatéřský
divadelní
soubor

TAK JO



IA

KONTAKT

NABÍDKA

PROGRAM

FOTOGALERIE

VIDEO

FOTOGALERIE



ONA A ON, včera, dnes a zítra...



Vydrž, miláčku!



Dárek z pekla



Trunbezurda a Pomněnka



Jak to bylo ve Strašidlákově



Dvanáct měsíčků



Záměna



Liga proti nevěře

★ 🏠 🇬🇧 📶 📺 📺

Do dalšího divadelního představení zbývá

75d:08h:14m:54s

AKTUALITY

Byla spuštěna nová pod...
Spustili jsme pro vás nové stránky v beta provozu. Pomozte nám zdo...

Bylo zprovozněno RSS
Můžete si nacházet zasílat novinky a aktualizace provedené na str...

Do programu bylo přidán...
Chci zobrazit program!

Dnes je čtvrtek 26. 5.

Vše nejlepší k svátku přejeme Filipovi!

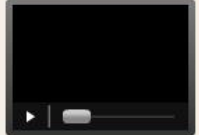
NÁVŠTĚVY

Dnes11

Celkem9906

Najed' myši a odhal kartičku

VIDEO



Podívejte se na další naše videa...

Poslední aktualizace

25. 5. 2011

© 2011 Tak Jo

Vytvořila Rawin

Všechna práva vyhrazena

The image is a screenshot of a website for an amateur theater group named "TAK JO". The website has a dark red background with a subtle pattern. At the top, the group's name "TAK JO" is written in large, white, stylized letters. To the left of the name is the text "amatérský divadelní soubor" (amateur theater group). To the right is a logo featuring a white silhouette of a person's head and shoulders, with a black cat-like face superimposed on it. Below the header is a navigation bar with several tabs: "Hlavní stránka" (Home), "KONTAKT" (Contact), "NABÍDKA" (Offer), "PROGRAM" (Program), "FOTOGALERIE" (Photo Gallery), and "VIDEO" (Video). The "PROGRAM" tab is currently selected. The main content area is divided into two columns. The left column is titled "PROGRAM" and lists the theater's schedule. It is organized by month, with headers for "Srpén 2011", "Květen 2011", "Duben 2011", "Březen 2011", and "Únor 2011". Each month's section lists the date, time, play title, author, and location. For example, in August 2011, there are two plays: "Trunbezurda a Pomněnka" on August 23rd at 18:00 and "Dárek z pekla" on August 11th at 18:00. In May 2011, there is "Záměna" on May 7th at 19:00. In April 2011, there are "Mr. Miss Olymp" on April 19th at 19:00 and "Dvanáct měšičků" on April 1st at 9:00. In March 2011, there is "Dvanáct měšičků" on March 13th at 10:00. In February 2011, there is "Liga proti nevěře" on February 3rd at 18:30. The right column contains several sections. At the top, there are social media icons (Facebook, Twitter, YouTube, etc.) and a section titled "Do dalšího divadelního představení zbývá" (Until the next theatrical performance remains) with a countdown timer showing "75d:08h:15m:10s". Below this is a section titled "AKTUALITY" (Actualities) with text about a new website version and RSS feed. Further down is a section titled "NÁVŠTĚVY" (Visits) showing the number of visitors for the day (11) and total (9906). At the bottom of the right column is a section titled "VIDEO" with a video player showing a scene from a play. The footer of the website contains the copyright notice "© 2011 Tak Jo", the creator's name "Vytvořila Rawn", and the statement "Všechna práva vyhrazena" (All rights reserved).

6.8 Příloha 8: Zdrojové soubory

