

Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačních technologií



Bakalářská práce

Web design pomocí CSS

Jan Švarc

© 2009 ČZU v Praze

Čestné prohlášení

Prohlašuji, že svou bakalářskou práci "Web design pomocí CSS" jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu literatury na konci práce. Jako autor uvedené bakalářské práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Příbrami dne:

Poděkování

Rád bych touto cestou poděkoval doc. Ing Zdeňkovi Havlíčkovi, CSc. za odborné vedení a rady, které mi velmi pomohly při zpracování této práce.

Web design pomocí CSS

Web design by the help CSS

Souhrn

Cílem práce je charakteristika web designu jako takového a také pozice technologie kaskádových stylů CSS v tomto pojmu. V práci budou rozebrány jednotlivé technologie web designu a následně bude proveden rozbor těchto hlavních technologií se svými klady i zápory. Další kapitoly jsou věnovány především kaskádovým stylům CSS, kde jsou popsány nejnovější trendy těchto stylů. V případové studii jsou nastíněny možnosti a největší výhody použití kaskádových stylů CSS.

Klíčová slova

Web design, kaskádové styly CSS, internetové stránky, pozicování, seo, JavaScript, layout, fluidní layout, fixní layout.

Summary

The purpose of the paper is characterize Web design as such and position of technology Cascading Style Sheets CSS in web design. Single technologies of the web design will be characterized and then main technologies will be analysed with their advantages and disadvantages. Next chapters are mainly paid to Cascading Style Sheets and the newest trends of Cascading Style Sheets are charactered there. The possibilities and the biggest advantages of using Cascading Style Sheets are characterized in the chapter case study.

Key words

Web design, Cascading Style Sheets CSS, internet sites, seo, JavaScript, layout, fluid layout, fixed layout.

Obsah

1. Úvod.....	- 3 -
2. Cíl a metodika práce	- 4 -
3. Web design.....	- 5 -
3.1 Technologie web designu.....	- 5 -
3.1.1 CSS.....	- 5 -
3.1.2 (X)HTML.....	- 7 -
3.1.3 JavaScript	- 8 -
3.1.4 SEO	- 10 -
3.2 Pravidla web designu	- 13 -
3.1.5 PHP	- 15 -
4. Současné trendy CSS	- 17 -
4.1 Pozicování webu pomocí CSS	- 18 -
4.1.1 Normální tok	- 19 -
4.1.2 Plovoucí prvky	- 20 -
4.1.3 Absolutní pozicování	- 20 -
4.2 Nejpoužívanější layouty pomocí CSS.....	- 21 -
4.2.1 Fluidní layout	- 22 -
4.2.2 Fixní layout	- 23 -
5. Případová studie.....	- 26 -
5.1 Design webu 1.....	- 26 -
5.2 Design webu 2.....	- 27 -
5.3 Design webu 3.....	- 28 -
6. Závěr.....	- 29 -
7. Použitá literatura	- 30 -
8. Příloha	- 31 -
8.1 Design webu 1	- 31 -
8.2 Design webu 2.....	- 34 -
8.3 Design webu 3.....	- 37 -

1. Úvod

Je to přesně 20 let, kdy pomocí jazyka HTML (Hyper Text Markup Language) propojovala skupina fyziků své dokumenty do skupiny serverů CERN. Bylo novinkou, že se text dal číst kdekoliv na světě pomocí jednoduchého programu a že byl umožněn přesun na jiné související informace. Nyní očekáváme více, mnohem více. Očekáváme webové sídlo s mnoha informacemi a dobrou prezentací, ale nechceme být zatěžováni složitým rozhraním. Informace musí být snadno k vyhledání a měly by být aktuální. Čisté a dynamické sídlo je dobrým aktivem pro uživatele a poskytovatele informací. Dnes jsou již pryč oslnivá webových sídla s blikajícími věcmi, mnoha animovanými obrázky, které obvykle zakrývaly malou hloubku obsahu. Požadujeme informace, chceme je o pět minut dříve a takovým způsobem, který se nám líbí. Proto vznikla činnost, která se nazývá web design, při které jsou vytvářeny webové stránky a webové aplikace.

2. Cíl a metodika práce

Cílem této práce je charakteristika web designu jako takového a také pozice technologie kaskádových stylů CSS v tomto pojmu. Nejprve bude v práci provedena charakteristika web designu a technologií spadající pod tento termín. Následně budou provedeny rozbor těchto hlavních technologií se svými klady i zápory.

Nedílnou součástí tvorby stránek je dnes technologie CSS, které by se web designér neměl vyhýbat. Proto se bude kapitola čtyři zabývat současnými trendy vývoje internetových stránek pomocí CSS stylů. Bude zde dále proveden rozbor jednotlivých webových layoutů pomocí CSS stylů se svými výhodami a nevýhodami.

V případové studii budou nastíněny hlavní přednosti kaskádových stylů CSS na praktických příkladech, jenž ukáží především flexibilitu CSS stylů, kdy změnou souboru s CSS se změní vzhled stránky aniž by se měnilo formátování www stránky.

Metodický postup se bude zakládat na sběru informací z odborné literatury a internetových zdrojů, které se zabývají tématem web design a kaskádových stylů CSS. Při zpracování bude také využito poznatků, které byly získány při praktickém studiu jednotlivých technologií web designu a při vlastní tvorbě www stránek.

3. Web design

Web design je nejednoznačný termín. Různí weboví profesionálové jej definují různě. Clint Echer ve své knize Profesionální web design uvádí dvě definice. Web design lze definovat jako proces programování podpůrných funkcí webu. Web design lze popsat jako návrh vzhledu, jehož pomocí se identifikuje společnost nebo jednotlivec, kterému daný web patří. Obě tyto definice jsou ve své podstatě správné. [1]

V první fázi vývoje internetový stránek se pod termínem web design skrývalo vytváření statických stránek ve formátu HTML s propojením textem a grafikou. V současnosti jsou ale vytvářeny také dynamické weby, které pomocí různých programovacích jazyků (JavaScript, PHP, ASP) umožňují kromě použití stránek HTML, grafiky a stylů CSS také interakci s databázemi a prohlížeči.

3.1 Technologie web designu

Tato část kapitoly se bude věnovat základním technologiím, které byly uvedeny výše, jako jsou:

- CSS
- (X)HTML
- JavaScript
- SEO
- PHP jako zástupce programovacího jazyka pro programování dynamických internetových stránek

3.1.1 CSS

CSS neboli Cascading Style Sheets, česky kaskádové styly, vznikly někdy kolem roku 1997. Je to kolekce metod pro grafickou úpravu webových stránek.

HTML nikdy nebyl zamýšlen jako jazyk vzhledu a interaktivity. Byl navržen, aby tvůrcům umožnil dokumenty řádně označit a uspořádat. Jenže mezitím, vyhovující rychle rostoucím potřebám autorů i zadavatelů, se postarali výrobci prohlížečů o množství doplňků a rozšíření jazyka HTML nad rámec standardu. HTML se tak roztránil na mnoho druhů a poddruhů, které odpovídaly pouze jednotlivým verzím konkrétního prohlížeče. S přibývajícými nestandardními rozšířeními se z dokumentů postupně začala vytrácet jejich struktura. Kód se více a více zaměřoval na popis výsledného vzhledu. S nástupem tabulek, coby mohutného nástroje pro pozicování prvků, zmizela struktura z dokumentů HTML takřka úplně. Vytrácejí se některé půvaby původního HTML, především přenositelnost a kompatibilita. Stránky jsou formátovány pro jeden typ prohlížeče a uživatelé, vlastníci jiný typ prohlížeče, stránku nemusejí vůbec přečíst. [2]

Řešením jsou standardy. Před lety nabrala na síle standardizační organizace W3C (World Wide Web Consortium). Tato organizace začala zastřešovat projekty související s publikováním na webu a řídit nové sjednocující standardy. Živelně působící výrobci prohlížečů se do projektu zapojili a postupně začínají standardy zavádět do praxe a prosazovat ve svých prohlížečích.

Jedním ze standardů jsou CSS. Umožňují formátovat dokumenty, definuje způsob prezentace na koncových zařízeních, popisuje podobu stránek a styl jednotlivých prvků a přitom nijak neovlivňuje obsah dokumentů samotných. Když definice stylu odstraníme nebo nepoužijeme, zůstane nezměněný původní dokument. Kompletní přechod, kdy bez CSS nebude možné formátovat dokumenty vůbec, je prozatím budoucnost, ale patrně nezvratná. [2]

Hlavní výhody používání CSS:

- rozsáhlejší možnosti formátování
- konzistentní styl
- oddělení struktury a stylu
- dynamická práce se styly

- kratší doba načítání stránky – v dnešní době vysokorychlostních připojení k internetu není tato výhoda ta hlavní.

Nevýhody používání CSS:

Hlavní, podle mého mínění dnes již jedinou, nevýhodou je ne vždy dostatečná podpora v majoritních prohlížečích. Prohlížeče obsahují v implementaci CSS chyby a je někdy nesnadné napsat kód tak, aby se ve všech (resp. v běžných) prohlížečích výsledek zobrazil stejně. Situace se však poslední dobou značně zlepšuje.

Vývoj u CSS jde stále dopředu, nyní již organizace W3C doporučuje třetí verzi CSS, tedy CCS3. Ta i její předchůdce CSS2 přinášejí podporu zejména zařízením typu PDA a podobných.

3.1.2 (X)HTML

Jak bylo zmíněno v předcházející kapitole, tak vývoj značkovacího jazyka HTML (HyperText Markup Language) byl ukončen a nástupce tohoto jazyka je jazyk založený na xml, tedy (X)HTML.

Rozlišujeme 3 druhy XHTML:

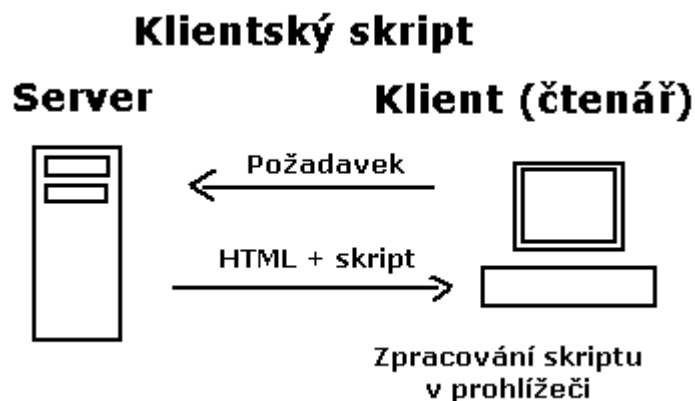
- *HTML 1.0 Strict* - čistě strukturální značkování, neobsahuje žádné značky spojené s formátováním vzhledu
- *XHTML 1.0 Transitional* - povoluje atributy pro formátování textu a odkazů v elementu body a některé další atributy
- *XHTML 1.0 Frameset* - používá se při použití rámců pro rozdělení okna prohlížeče na dvě nebo více částí

Rozdíly XHTML oproti HTML:

- Všechny atributy mají hodnoty v uvozovkách
- Zákaz křížení značek
- Značky a atributy jsou malými písmeny
- Nepárové značky končí lomítkem
- Párové značky jsou párové povinně
- Všechny atributy musejí mít hodnotu
- Interní javascript a styly se zapisují jiným způsobem
- Dokument má mít XML prolog.
- Dokument požaduje správný doctype.

3.1.3 JavaScript

Programovací jazyk JavaScript vytvořila koncem minulého století společnost Netscape. Zpočátku tento jazyk nazývala LiveScript, ale když nastal ve světě boom okolo programovacího jazyka Java, usoudili, že z marketingového hlediska bude lepší když se nový programovací jazyk přejmenuje a bude se nazývat JavaScript, přestože s Javou nemá téměř nic společného. Jednou z nejdůležitějších vlastností JavaScriptu, od kterých se odvíjí i další řada vlastností, je fakt, že jde o programovací jazyk, jehož činnost zabezpečuje prohlížeč webových stránek návštěvníka stránky a ne server, na kterém jsou stránky uloženy. K provádění javascriptového kódu dochází na počítači návštěvníka stránky až po jejím načtení. Řečeno v odborné terminologii, JavaScript je programovací jazyk, který je interpretovaný na straně klienta.[4]



Obrázek 1 - klientský skript

Z toho pro tvůrce stránek vyplývá několik omezení. Kvůli zachování bezpečnosti je pochopitelné, že autoři programovacího jazyka JavaScript do něj nezabudovali funkce, které by mu umožňovaly zapisovat a načítat údaje ze souborů. Pro návštěvníka stránky je to pozitivum, jelikož se nemusí obávat toho, že by mu někdo pomocí tohoto jazyka dokázal smazat obsah disku nebo zneužít diskrétní informace, které se na něm nachází. Na druhou stranu to, že je JavaScript jazyk interpretovaný na straně klienta, do značné míry svazuje ruce tvůrcům stránek, jelikož je ochuzuje o možnost zpracovávat a centrálně uchovávat údaje týkajících se provozu webové stránky. Jinými slovy, jen pomocí JavaScriptu nelze naprogramovat aplikace typu: počítadlo evidující celkový počet návštěv na stránce, knihu návštěv, diskusní fórum nebo redakční systém. Uvedená omezení významně zužují okruh možné aplikace JavaScriptu, ale i přesto je JavaScript v rukou tvůrců stránek silným nástrojem přinášejícím jednu z možností, jak zinteraktivnit webovou stránku. [4]

V současné době se JavaScript nejčastěji používá k např.:

- záměně obrázků na stránce v závislosti na poloze kurzoru myši,
- vytváření prvků ulehčujících navigaci v podobě hierarchických a roletových nabídek,

- kontrole správnosti údajů zadaných návštěvníkem stránky do formuláře (před jejich odesláním na server) a jejich korekce,
- práci s okny prohlížeče stránek,
- využití údajů o aktuálním datu a času a jejich zobrazení.

3.1.4 SEO

Internetové vyhledávače se v poslední době stávají stále častějším způsobem hledání informací. Tím jak zviditelnit webovou stránku ve vyhledávačích se zabývá Search Engine Marketing (SEM). Skládá se ze dvou základních částí, placené a neplacené. Rozdíl mezi nimi samozřejmě není nutně v celkových finančních nákladech, ale v tom že v placené části SEM se platí za zviditelnění přímo vyhledávači, kdežto v neplacené se platí pouze člověku (firmě), který optimalizaci provádí a ne vyhledávači. Neplacená část se nazývá optimalizace pro vyhledávače (SEO - Search Engine Optimization), někdy také pouze přirozená optimalizace.

Co jsou vyhledávače. Třeba Google, Jyxo, Morfeo, fulltext Seznamu nebo třeba Yahoo.com. Servery, které mají velikou databázi stránek s informacemi o tom, která stránka internetu obsahuje jaké slovo. Protože takové vyhledávače umějí hledat v celém textu stránek, říká se jim fulltextové. Takových serverů existují řádově stovky. Programům, které procházejí web a schraňují pro vyhledávače informace, se říká roboti, boti nebo crawleři.

Robot prochází internetem. Většinou někde začne, třeba v nějakém katalogu. Najde odkaz a sleduje ho. Nalezenou stránku si stáhne pro zaindexování, zároveň hledá další odkazy, které případně opět sleduje, aby si to zase postahoval. Z toho vyplývá, že robot může odhalit pouze stránku, na kterou vede nějaký odkaz. Musí přitom jít o klasický odkaz. Pokud na stránky nebudou roboti chodit, žádný čtenář ty stránky nemůže najít fulltextovým vyhledávačem. V praxi ale roboti najdou jenom část stránek, které existují, protože to prostě

všechno nestíhají. Jiné roboty musíte popohnat. Většinou je potřeba na stránkách vyhledávače někde najít nenápadný odkaz [přidej odkaz](#), [add a link](#), [submit site](#), [add site](#) a podobně. Zadá se URL adresa, a roboti časem přijdou.

Indexování. Čili zpracování dat do databáze. Každý vyhledávač si přebírá nasbírané informace po svém. Nejčastěji si vypisuje všechna slova, počítá jejich váhu a dává je do relace s adresou stránky.

Váha (důležitost). Při hledání slova vyplivne vyhledávač první adresy stránek, na kterých má hledané slovo velkou váhu. Váhu vypočítá tak, že se dané slovo vyskytuje v titulku stránky, v klíčových slovech, v popisu a v nadpisech. Je nutno poznamenat, že každý vyhledávač to počítá jinak.

Co vyhledávače sledují	V HTML	Důležitost (váha)
titulek	<title>text titulku</title>	obrovská
klíčová slova	<meta name=keywords content="slovo, slovo">	značná, někdy žádná (Google)
popis (description)	<meta name=description content="Stručný popis">	různá
nadpis 1. úrovně	<H1>Nadpis</h1>	značná
ostatní nadpisy	<Hn>Nadpis</hn>	sporná
začátek stránky	<body>Několik prvních slov ...	větší než malá
adresa URL	jméno souboru včetně cesty	různá
text odkazů mířících na tu stránku z jiného serveru	text odkazu	u některých vyhledávačů obrovská (Google, Jyxo)

alty u obrázků	<code></code>	malá
text stránky	prostě text	malá
katalogový popis	co zadáte do případného spřízněného katalogu	různá (pouze na Seznamu)

Řazení výsledků. Různé vyhledávače řadí výsledky různě. Obecně se dá říci, že čím více se hledaná slova na stránce vyskytují a čím mají významnější pozici (titulky, nadpisy), tím je stránka řazena výše. Sleduje se i text odkazů mířících na stránku, zda obsahuje hledané slovo. Algoritmy, ale nikdo přesně nezná a liší se nejen od vyhledávače k vyhledávači např. na Google prý každý měsíc.

Page Rank. Další metodou je takzvaný PageRank (PR), který používá Google (GPR) a poslední dobou mnoho jiných vyhledávačů. PageRank vyjadřuje něco jako věrohodnost nebo důležitost stránky. Page Rank ovlivňuje řazení výsledků, nikdo přesně neví jak (asi podstatně). Jaké mají vaše stránky Page Rank, se můžete dozvědět, pokud si stáhnete a nainstalujete Google Toolbar. GPR se počítá podle toho, kolik stránek danou stránku odkazuje -- čím více, tím lepší PageRank. Navíc odkazy ze stránek s vyšším PageRankem mají větší váhu. Google Page Rank se počítá postupným přepočítáváním (iterací). Existuje na to vzoreček. Nejlepší český vyhledávač Jyxo má navíc i JyxoRank, který se počítá podle toho, z kolika různých domén druhé úrovně vedou na stránku odkazy. Podle JyxoRanku řadí výsledky svého hledání.

3.2 Pravidla web designu

Web design má několik nepsaných pravidel, kterých by se profesionální web designér měl držet.

- 1. To, že něco můžeme udělat, ještě neznamena, že by jsme to udělat měli.** Web design nabízí technologie a mnohé možnosti a nástroje k vytváření webů. Přidání určité technologie může však často web zpomalit nebo být pro uživatele nevhodná, což přinutí návštěvníka web opustit. Při přidávání určité technologie je nutno si položit otázku, zda přidává technologie webu na hodnotě nebo ji přidáváme pouze jako módní novinku. Pro příklad tohoto pravidla zde uvedu blikající text. Ten zejména v polovině 90. let 20. století byl využíván až příliš často a většinu uživatelů brzy otrávil.[1]
- 2. Výjimka potvrzuje pravidlo.** Pro web designéry téměř neexistují absolutní pravidla. Určitou technologii či metodu bychom neměli zavrhnout proto, že se nedala uplatnit na nějakém jiném webu, či již není módní záležitostí. Příkladem zde může opět být blikající text, který v době svého používání byl sice na 99 procentech webů spíše na škodu, ale i dnes by se daly najít případy, kdy by mohl být užitečný. Například situace kdy uživatel ve formuláři zapomene či nezadá správně některé údaje. Nejčastěji bývá taková chyba zvýrazněna barevně (například červeně), aby ji uživatel mohl snadněji najít a opravit. Na některých webových stránkách však bývá mnoho barevných kombinací, takže pro uživatele může být složitější, aby si všiml červeného textu označující danou chybu. V tomto případě by mohl být blikající text dostatečně výrazný, takže uživatel by mohl chybu rychle zjistit a opravit.[1]
- 3. Soudcem a porotou jsou koncoví uživatelé.** Když je design webu vytvářen, najde se spousta lidí ochotných přispěchat s hodnocením. Web designér může mít o vzhledu a funkčnosti webu jinou představu než jeho

nadřazený. Skutečným soudcem a porotou by měli ale být uživatelé. Jejich spokojenost je tím proč je web vytvářen, a proto by se na webu nemělo vyskytovat nic, co by se většině uživatelů nelíbilo či čemu by nerozuměli.[1]

- 4. Web designér by měl vždy usilovat o získání co nejširších znalostí a zkušeností.** Profesionální web designér by se měl vždy snažit pochopit a ve svých dílech využívat nejvhodnější postupy, které existují a to bez ohledu na osobní představu o technických a estetických pravidlech a standardech pro web design. Při řešení libovolného problému bude web designér vždy ve výhodě, pokud mu bude důkladně rozumět po technické stránce. Příkladem zde může být různá zobrazení některých elementů webu v různých prohlížečích, kde je důležitá právě zkušenost a široké znalosti.[1]
- 5. Nejlepším přístupem je skromnost.** Na poli web designu hraje důležitou roli spousta maličkostí. Vždy se někde najde web designér, jehož weby jsou atraktivnější, je na nich použita novější technologie nebo starší technologii využívá kreativnějším způsobem. Pokud web designér se nenechá uspokojit prvním úspěchem, má spoustu příležitostí jak se poučit od jiných.[1]
- 6. Není na světě člověk, který by se zvděčil všem.** Ať jako vzorek poslouží 100 milionů uživatelů webu po celém světě nebo 10 lidí v malé firmě, dá se očekávat, že design se nebude líbit každému. Existuje jemná hranice mezi tím, že se design bude líbit většině lidí, a tím, že se pokusíme uspokojit všechny, ale ve skutečnosti neuspokojíme nikoho.[1]
- 7. Udržování si přehled o specifikacích a standardech.** Webové specifikace a standardy se neustále vyvíjí a jistě se vyvíjet budou i v budoucnu. Web designér by ale měl chápat alespoň základní principy nejnovějších technik, protože mohou mít vliv na jeho budoucí práci.[1]

3.1.5 PHP

Programovací jazyk PHP byl vyvinut k tomu, aby se statické stránky HTML obohatily o dynamické skriptování. PHP bylo původně označení pro Personal Home Page, tedy osobní domácí stránky. Vše začalo v roce 1994, kdy byla napsána binární část Common Gateway Interface (CGI) v programovacím jazyku C. Tuto prvotní část napsal Dánský/Grónský Programátor Rasmus Lerdorf. Lerdorf zpočátku vytvořil tyto nástroje pro osobní domácí stránky (Personál Home Page) za účelem možné záměny s malou skupinou skriptů v Perlu, které chtěl používat pro údržbu osobní domovské stránky. Nástroje měly zajistit běh úloh jako například zobrazení résumé (obsahu) a zaznamenávání návštěvnosti jeho stránek. Tento binární kód ještě tentýž rok skloubil s jiným programem, který sám napsal. Po spojení s Form Interpreter tak vznikla kombinace PHP/FI, která měla mnohem větší funkčnost. PHP/FI obsahovala širokou implementaci pro programovací jazyk C a navíc tato verze mohla komunikovat s databázemi, což umožnilo tvorbu prvních jednoduchých dynamických webových aplikací. Lerdorf veřejně vydal PHP 8. června 1995, aby mohl najít co nejvíce chyb a tak zdokonalil kód. Tato verze byla pojmenována jako PHP verze 2 a již měla základní funkčnost jako má dnešní PHP. To zahrnuje například proměnné ve stylu Perlu, zpracování formulářů a možnost vložit HTML kód. Syntaxe byla obdobná jako u Perlu, ale byla omezenější, jednodušší a méně konzistentní. Zeev Suraski a Andi Gutmans, dva Izraelští vývojáři na Technion IIT, přepsali parser v roce 1997, vytvořili tak základ PHP 3 a změnili název jazyka na rekurzivní zkratku PHP = PHP: Hypertext Preprocesso. V roce 2008 se stává PHP 5 jedinou stabilní verzí, která se vyvíjí. PHP 6 se bude zároveň vyvíjet s verzí PHP 5.[5]

Výhody PHP:

- PHP je specializované na webové stránky

- multiplatformost (zejména Linux, Windows)
- poměrně slušná dokumentace
- obrovské množství projektů a kódů, které lze zdarma využít (WordPress, phpBB a další)
- vynikající, velmi svobodná licence – svobodnější, než GPL

Nevýhody PHP:

- jazyk PHP není nikde definován, je popsán pouze jeho implementací
- ač jazyk výborně podporuje výjimky, jeho knihovna je používá jen zřídka
- ve standardní distribuci chybí ladící (debugovací) nástroj
- slabší podpora Unicode, pouze přes PHP knihovnu (v PHP 6 má být Unicode řetězec jako základní typ)

4. Současné trendy CSS

Tahanice mezi zastánci a odpůrci formátování WWW stránek s CSS a pomocí tabulek jsou nekonečné. Avšak v dnešní době si dovolím tvrdit, že u profesionálních web designérů převládá formátování a hlavně i pozicování pomocí technologie CSS. Pro ukázkou zde budou uvedeny výhody a hlavně nevýhody a tedy i proč se již tabulkové či dokonce rámcové pozicování nepoužívá.

Layout pomocí rámu - Obsah každého rámu je tvořen jedním HTML dokumentem.

Výhody rámcového layoutu:

- Jeden kód použitelný na více stránkách (lze lépe nahradit skripty (PHP) nebo šablonami)
- Obsah rámce je stále přístupný (vhodné pro menu, ale zase stále překáží)
- Velikost rámců lze přizpůsobit požadavkům uživatele

Nevýhody rámcového layoutu:

- Komplikovaný tisk ve většině prohlížečů
- Nelze vytvořit odkaz na konkrétní konfiguraci rámu
- Problémy s bookmarkováním...
- Některé vyhledavače rámy neindexují (Pokud indexují, tak po jednotlivých dokumentech)
- Nedostaneme se na kompletní frameset
- Omezuje design stránky (části jsou viditelně odděleny)
- Některá zařízení rámy nezobrazují (lynx, hlas. čtečky, PDA)

Tabulkový layout - Zneužití tabulek pro vytvoření layoutu. Donedávna jediný způsob, proto značně rozšířené.

Výhody:

- Jednoduchost návrhu
- Kompatibilita
- Mnoho možností manipulace s obsahem, zarovnávání, ...

Nevýhody:

- Vzhled stránky daný přímo v HTML kódu
- Dlouhý a nepřehledný kód (<tr><td><tr><td>)
- Tabulka se zobrazí, až se celá načte.
- I při malé změně designu je někdy nutné předělat celý HTML kód

4.1 Pozicování webu pomocí CSS

V dnešní době máme několik možností, jak řešit layout webových stránek. Každý má své problémy, a tak vznikají různé CSS frameworky, které se některé z nich snaží řešit. CSS layout, občas nazývaný beztabulkový, nebo div layout, je rozvržení stránky, přičemž využíváme stylování blokového elementu div přes css. V CSS se používají tři různá poziční schémata, každé z nich nakládá s prvky odlišným způsobem:

- normální tok
- plovoucí prvky
- absolutní pozicování

4.1.1 Normální tok

Normální tok je výchozím zobrazovacím schématem pro dokumenty. Jednoduše zobrazuje obsahy jednotlivých elementů (boxů), jak jdou za sebou. Vzdálenosti mezi nimi závisejí na jejich vlastnostech margin, přičemž se tyto okraje slučují. Levá hrana každého boxu (tj. až vnější hrana oblasti margin) se dotýká levé hrany omezujícího bloku. Řádkové boxy se umísťují vedle sebe od horní hrany omezujícího bloku. Obdélník, který obsahuje boxy tvořící jednu řádku, se nazývá box řádky. Jeho šířka se rovná šířce omezujícího bloku. Jeho výška je přinejmenším taková, aby pojal všechny boxy, které jej tvoří. Pokud se všechny řádkové boxy nevejdou do jediného boxu řádky, jsou rozděleny do několika boxů řádek. Ty se pak řadí svisle přímo pod sebe. Odstavec textu je tvořen takovou posloupností boxů řádek.[2]

I přes tuto zdánlivou triviálnost existují "kouzla", která můžeme s elementem v normálním toku provádět. Těto magii se říká relativní pozicování (relative positioning). Prvek v normálním toku, který má vlastnost `position:relative`, může být posunut ze své pozice. Velikost a směr posunu určují vlastnosti `left`, `right`, `top` a `bottom`. Vlastnost `left` určuje posun vpravo, `right` posun vlevo. Vlastnost `top` posouvá prvek dolů, `bottom` nahoru. Platí přitom, že pokud zadáme obě protilehlé vlastnosti a nebudou odpovídat této rovnosti, jedna z nich bude ignorována. Element, který je relativně pozicován, je nejprve zformátován v dokumentu, jako by pozicován nebyl, a teprve nakonec je posunut ze své původní pozice o zadanou vzdálenost. Znamená to tedy, že relativní pozicování nemá žádný vliv na vzhled a rozměry prvku. Posun také nijak neovlivňuje formátování okolních prvků.[2]

4.1.2 Plovoucí prvky

Prvky v normálním toku se stávají plovoucími, pokud mají vlastnost `float:left`, respektive `float:right`. Prvek je z místa, kde by byl normálně zobrazen, posunut k levé (respektive pravé) hraně omezujícího bloku, z toku dokumentu je vyňat a ostatní obsah jej obtéká. To znamená, že následující prvky jsou zobrazeny ve zbývajícím prostoru podél něj, prostor pro jejich obsah je zúžen a v plné šíři jsou zobrazeny zase až za tímto plovoucím prvkem. Pokud má však následující obsah nastavenou příslušnou hodnotu vlastnosti `Clear`, plovoucí prvek obtékat nebude a zobrazí se až pod ním. Protože je plovoucí prvek vyňat z toku dokumentu, boxy před i za ním jsou zobrazeny, jako by zde vůbec nebyl. Každý plovoucí prvek musí mít specifikovanou šířku vlastností `width` nebo svými skutečnými rozměry. Plovoucí prvek se automaticky stává blokem. Jeho horní hrana je zarovnána s horní hranou aktuální řádky. Horní a dolní okraje plovoucích prvků se již se sousedícími okraji ostatních boxů neslučují. Pokud je plovoucí prvek relativně pozicován, stejně jako v normálním toku dokumentu jsou i jeho okolí nejprve zformátovány bez posunutí a teprve následně se prvek odsune podle zadaných hodnot. Pokud je tedy relativně posunut do prostoru ostatních prvků, bude je překrývat.[2]

4.1.3 Absolutní pozicování

Při absolutním pozicování (`position` je `absolute` nebo `fixed`) je prvek zcela vyňat z normálního toku dokumentu a vytváří nový hlavní blokový box a tím i nový normální tok pro své potomky. Všechny prvky, které jsou obsaženy uvnitř tohoto boxu se pozicují již vzhledem k tomuto bloku. Absolutně pozicované prvky, které nejsou navzájem vnořené, spolu žádným způsobem neinteragují a mohou se překrývat - stejně jako s obsahem normálního toku dokumentu. Pozice prvku se odvozuje od polohy a rozměru jeho omezujícího bloku. Ten je určen

nejbližším nadřazeným absolutně, nebo relativně pozicovaným prvkem, omezující blok tvoří jeho hrana výplně. [2]

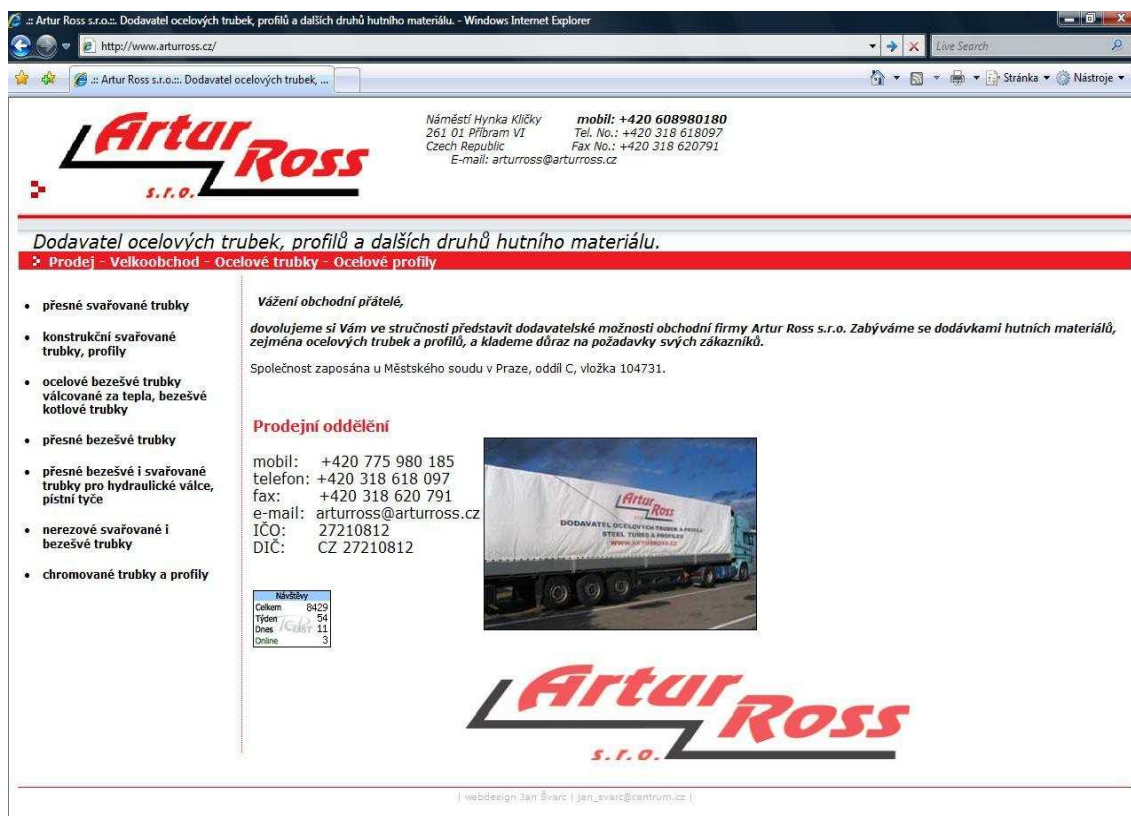
Tím, že se absolutně pozicované prvky vyjímají z ostatního toku dokumentu, vznikají vlastně dvě paralelní, na sobě nezávislá zobrazení obsahu dokumentu. Normální tok dokumentu tvoří základní vrstvu, nad ní se zakládají další vrstvy pro každý absolutně pozicovaný prvek. Do vrstvy vytvořené pro pozicovaný prvek se umísťuje jeho obsah v normálním toku. Pro jeho potomky, kteří jsou také absolutně pozicováni, se opět vytváří další podvrstvy. Všechny podvrstvy vytvořené pro potomky jsou vytvořeny v rámci původní vrstvy prvku. [2]

4.2 Nejpoužívanější layouty pomocí CSS

Mezi nejpoužívanější layouty pomocí CSS patří fixní, procentní a fluidní CSS layout. Shodou okolností se na toto téma konala přednáška na letošním Webexpo2008, které se konalo v říjnu roku 2008 na České zemědělské Universitě v Praze. Přednášku vedl pan Jaroslav Knap, z internetu známý jako ASH721. Jaroslav Knap v přednášce Fixní, procentní a fluidní CSS layout porovnával výhody a nevýhody různých přístupů pro tvorbu layoutu. Na příkladech některých webů předváděl problémy, které mohou nastat při nestandardním nastavení prohlížečů nebo při změně velikosti písma. Zaujala mne také položená otázka: „Má cenu ještě tvořit weby za pomoci procentního layoutu (Webová stránka je navržena tak, že se natahuje podle šířky okna prohlížeče.), když většina prohlížečů dnes umí zoomovat?“ (zoomování je vlastnost prohlížeče, kterou vyvoláte u Firefoxu a IE kombinací kláves Ctrl++(-)). Souhlasím s odpovědí pana Knapa, kdy říkal že procentní layout už velkou budoucnost s touto vlastností prohlížečů nemá, jelikož si uživatel stránku v prohlížeči prostě přiblíží. Proto v této části kapitoly bude uveden pouze příklad fluidního a fixního layoutu.

4.2.1 Fluidní layout

Webová stránka je navržena tak, že se natahuje podle šířky okna prohlížeče. S rozšiřujícími se velkými monitory a HDTV se rozdíly mezi rozlišeními čím dál víc prohlubují. Někteří lidé se rozhodnou svoje stránky udělat tak, aby se rozšiřovaly podle toho, jaké má návštěvník rozlišení (inteligentní člověk ovšem vytyčí nějaké limity, aby se stránky třeba příliš nezúžily nebo neroztáhly). Na jedné straně je to asi to nejlepší, co můžete udělat v boji s mnoha různými rozlišeními, co návštěvník může mít. Na straně druhé není lehké podobný web udělat tak, aby tím neutrpělo grafické zpracování. Díky roztahujícímu se obsahu (menu asi roztahovat nebudete) může na některých větších rozlišeních web vypadat opravdu divně a hlavně špatně se doplňovat s hlavičkou.



Obrázek 2 - fluidní layout

Na obrázku 2 je uveden příklad fluidního layout z vlastní tvorby pro firmu Artur Ross s.r.o.. Tento web je tvořen hlavičkou, dvěma sloupci a patičkou. Pozadí hlavičky se natahuje podle šířky okna prohlížeče. Pozadí hlavičky je tvořeno milimetr tenkými pruhy vedle sebe, kde se střídá bílá, červená a šedá. Dále je v hlavičce umístěno logo, které je umístěno absolutně, aby nedocházelo k deformaci u příliš širokého nebo naopak úzkého okna prohlížeče. V červeném a šedém pruhu už je klasický text. Pod hlavičkou jsou zmiňované dva sloupce, kde levý sloupec, v němž se nachází menu, má pevně danou šířku a pravý sloupec, v němž je obsahová část webu a jeho šířka je fluidní. Web je zakončen patičkou, která ukončuje web úzkou červenou linkou.

Výhody takto tvořeného webu:

- web je zobrazen stále v celém okně prohlížeče
- nízký objem dat

Nevýhody takto tvořeného webu:

- není lehké podobný web udělat tak, aby tím neutrpělo grafické zpracování
- díky roztahovacímu obsahu na některých větších rozlišeních může web vypadat opravdu divně a špatně se doplňovat s hlavičkou

4.2.2 Fixní layout

Fixní pozicování s prvky `position:fixed` jsou podkategorií absolutně pozicovaných prvků. Jediným rozdílem je, že jejich výchozí omezující blok je určen průzorem dokumentu. Znamená to, že pozicované prvky jsou umístěny relativně k okrajům tohoto průzoru (okna dokumentu v prohlížeči). Posun průzoru po dokumentu neovlivňuje jejich pozici a při posouvání tak zůstávají stále na původním místě.

Třísloupcový fixní layout. V současnosti je to jeden z nepoužívanějších layoutů, většina kodérů již rezignovala na fluidní layouts díky množství problémů, které přinášely. Pro příklad uveden obrázek webu firmy Stavus a.s. opět z vlastní tvorby.



Obrázek 3 - Třísloupcový fixní layout

Na obrázku 3 je uveden třísloupcový layout, který ještě není zcela dokončen. Jak již název napovídá skládá se ze tří sloupců, které mají fixní, tedy neměnnou šířku. Tento styl jsem vybral u tohoto webu proto, že grafika hlavičky není neměnná. Od loga firmy jsou v modré barvě vytvořeny tenké bílé linky, které se nedají vedle sebe opakovat.

Výhody takto tvořeného webu:

- Udělat layout s fixní šířkou je jednodušší

- Při složitém grafickém podkladu prakticky jediná možnost

Nevýhody takto tvořeného webu:

- při velké rozlišení vyplňuje malou část okna prohlížeče, což jak je naznačeno na začátku této kapitoly, lze pomocí zoomování internetového prohlížeče vyřešit
- Naopak při malém rozlišení se objeví dole scroll bar a uživatel je nucen si v oknu prohlížeče posouvat obsah webu, což není pro uživatele pohodlné. Na druhou stranu v dnešní době LCD monitorů s velkým rozlišením je uživatelů jenž stále používají rozlišení 800 x 600 a nižší naprosté minimum.

5. Případová studie

V této kapitole jak již zaznělo na začátku bude nastíněna hlavní výhoda CSS a to tak, že úpravou souboru s CSS se změní vzhled webu aniž by se zasáhlo do formátování stránky.

5.1 Design webu 1

Na obrázku 4 je vidět první vzhled stránky, která je také z vlastní tvorby a byla navržena pro firmu Lama s.r.o., jenž se zabývá malířskými pracemi. Web je dělán fixně a zarovnan na střed okna prohlížeče. Šířka webu je 790 pixelů a výška 585 pixelů. Web byl optimalizován pro uživatele používající rozlišení o šíři 1024px a vyšší, což má v dnešní době snad 99 procent uživatelů.

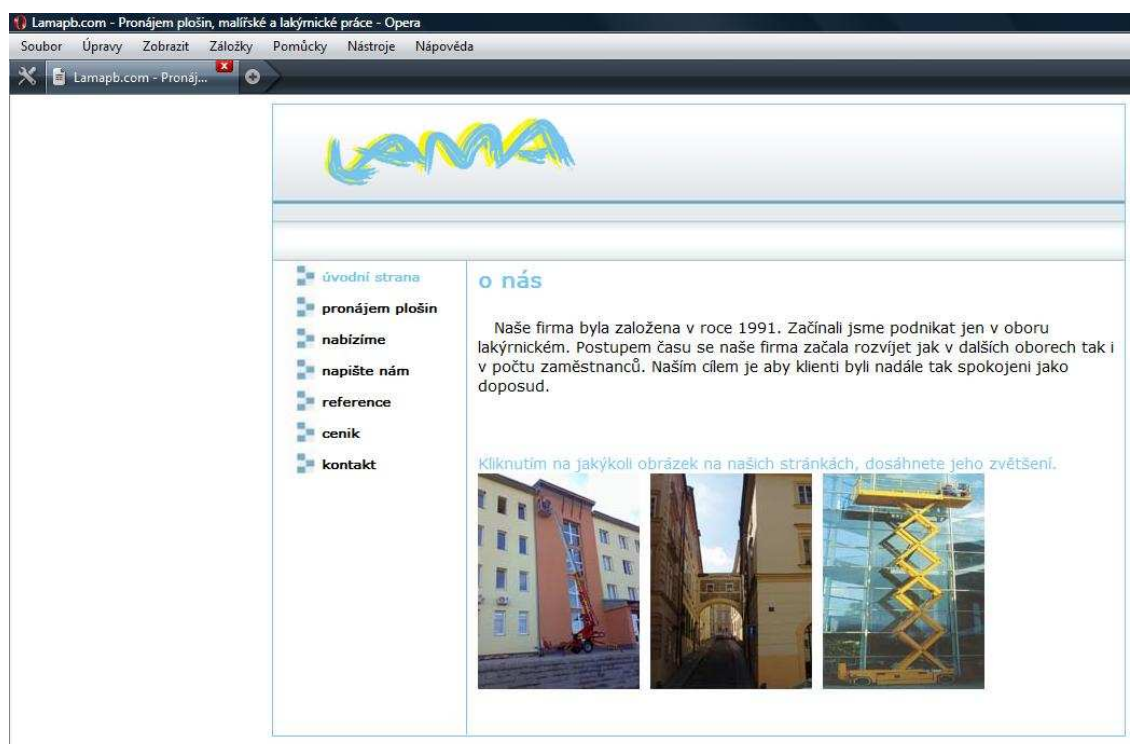


Obrázek 4 - Web firmy Lama s.r.o. - design 1

Pro srovnání jednotlivých zdrojových kódů jednotlivých verzí designu webu jsou zdrojové kódy přiloženy v příloze.

5.2 Design webu 2

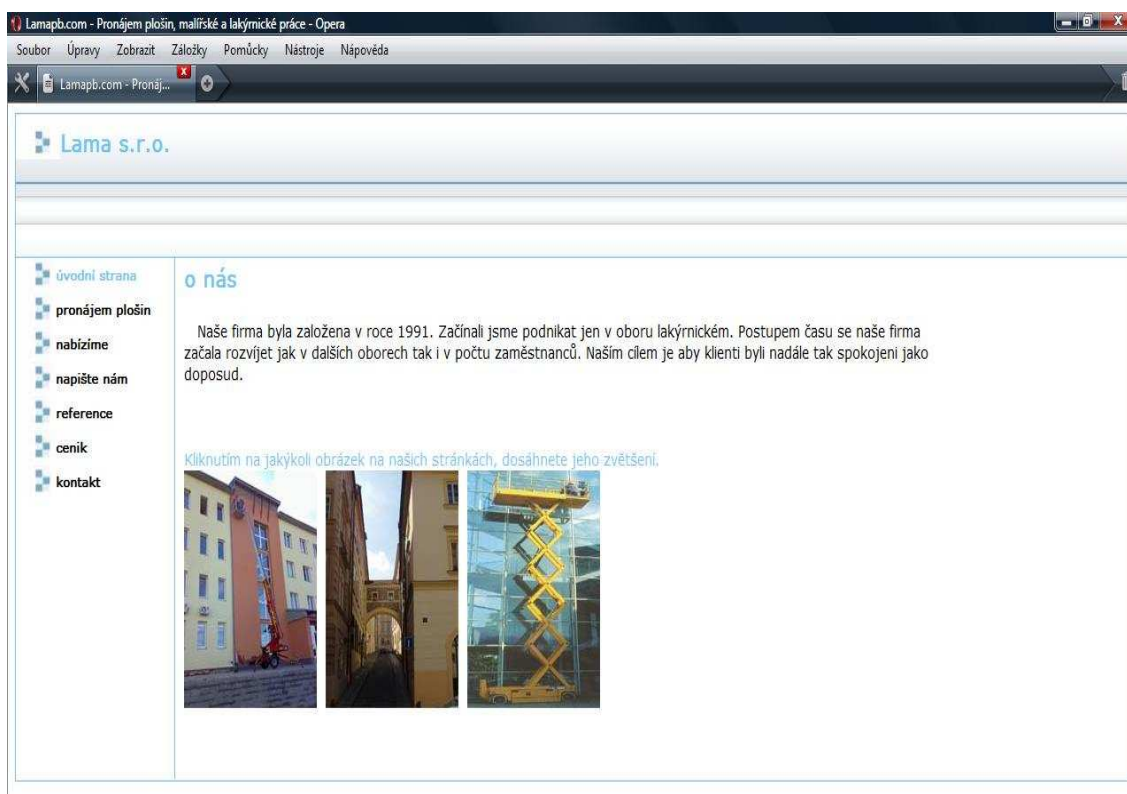
Na obrázku 5 je vidět design 2. Tento design má naprosto podobné rysy jako design 1. Opět fixní layout s šířkou 790 a výškou 585 pixelů vycentrovaný do středu okna prohlížeče. Do formátování stránky jsem nezasáhl a design stránek je jiný. To vše díky CSS stylům, jenž jsou uložené v externím souboru a pouze jejich úpravou vznikl design 2. Zdrojový kód stránky i souboru s CSS styly je opět přiložen v příloze pro srovnání.



Obrázek 5 - Web firmy Lama s.r.o. - design 2

5.3 Design webu 3

Design číslo 3 je podobný jako design číslo 2, ale je na rozdíl od ostatních designů fluidní. To znamená, že se přizpůsobuje rozlišení uživatele tak jak je vysvětleno výše. Rozdíl mezi designem 2 je především v hlavičce, kde je pozadí dělá pouze pruh široký 1 pixel, který se neustále opakuje na 100% šíři stránky. Tento design se neobešel bez zásahu do formátování. Jedná se však o jediný zásah a to zásah do hlavičky stránky, kde byl doplněn nadpisem jméno firmy. Vše ostatní bylo opět upraveno pouze v souboru s CSS styly.



Obrázek 6 - Web firmy Lama s.r.o. - design 3

6. Závěr

V předložené práci bylo provedeno nastínění činnosti web design, která se vytvářením stránek zabývá. Byla provedena charakteristika jednotlivých technologií tvořících tento pojem. Zejména web design tvořený CSS styly, jednotlivé webové layouty pomocí CSS se svými výhodami nevýhodami. Všechny sdělené informace jsou datovány ke květnu 2009.

CSS styly a celý web design je v neustálém vývoji, který se snaží usměrňovat standardizační organizace W3C. V případové studii byly nastíněny hlavní přednosti CSS stylů, zejména flexibilita, díky které lze říci, že web design bez CSS stylů si nelze v blízké budoucnosti představit. Nyní již organizace W3C doporučuje web designérům a tvůrcům majoritních prohlížečů třetí verzi CSS, tedy CSS3.

7. Použitá literatura

1. Echer, Clint. Profesionální webdesign Techniky a vzorová řešení. 1. vydání. Brno: CP Books, a.s., 2005. 421s. ISBN 80-251-0547-4.
2. Staníček, Petr. CSS Kaskádové styly Kompletní průvodce. 2.vydání. Brno: CP Books, a.s., 2003. 178 s. ISBN 80-7226-872-4.
3. Krug, Steve. Webdesign. 2.vydání. Brno: CP Books, a.s., 2007. 166s. ISBN 80-251-1291-8.
4. Škultéty, Rastislav. JavaScript Programujeme internetové aplikace. 2. vydání. Brno: CP Books, a.s., 2004. 224 s. ISBN 80-251-0144-4.
5. Kotlet, Michael. PHP 5 a MySQL 5 Průvodce webového programátora. 1. vydání. Brno: CP Books, a.s., 2007. 607 s. ISBN 978-80-251-1813-9.

8. Příloha

8.1 Design webu 1

HTML soubor

```
<?
$prohlizec_info=$_SERVER['HTTP_USER_AGENT'];
if(ereg("MSIE", $prohlizec_info)){ $prohlizec="msie";}
if(!$menu){
$menu="uvod";
}
if(!$obsah){
$obsah="malirskeprace";
}?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="content-type" content="text/html; charset=windows-1250">
<meta name="generator" content="PSPad editor, www.pspad.com">
<meta name="description" content="Lama, Malířské práce, Malíř, Malíř pokojů, ">
<meta name="keywords" content="Lama, Malířské práce, Malíř, Malíř pokojů,..
</title>
<link rel="stylesheet" type="text/css" href="styly_lama.css">
</head>
<body>
<div id="all">
<div id="logo"></div>
<div id="levy_sloupec_s_odkazy">
<div id="odkaz1">
<b><a href="index.php?menu=uvod" id="odkazy_text" style="<?
if($menu=="uvod"){echo("color:#75c6f2");}?>">úvodní strana</a></b>
</div>
<div id="odkaz2">
<b><a href="index.php?menu=plosiny" id="odkazy_text" style="<?
if($menu=="plosiny"){echo("color:#75c6f2");}?>">pronájem plošin </a></b>
</div>
<div id="odkaz2">
<b><a href="index.php?menu=nabidka" id="odkazy_text" style="<?
if($menu=="nabidka"){echo("color:#75c6f2");}?>">nabízíme </a></b>
</div>
<div id="odkaz2">
<b><a href="index.php?menu=napiste" id="odkazy_text" style="<?
if($menu=="napiste"){echo("color:#75c6f2");}?>">napište nám</a></b>
</div>
```

```

<div id="odkaz2">
<b><a href="index.php?menu=reference" id="odkazy_text" style="<?
if($menu=="reference"){echo("color:#75c6f2");}?>">reference</a></b>
</div>
<div id="odkaz2">
<b><a href="index.php?menu=cenik" id="odkazy_text" style="<?
if($menu=="cenik"){echo("color:#75c6f2");}?>">cenik</a></b>
</div>
<div id="odkaz2">
<b><a href="index.php?menu=kontakt" id="odkazy_text" style="<?
if($menu=="kontakt"){echo("color:#75c6f2");}?>">kontakt</a></b>
</div>
<div id="pravyroh">
</div>
</div>
<div id="obsah">
<?php
if(isset($menu)){
$adresa= $menu . ".php";
include ("$adresa");
}?>
</div>
</div>
</body>
</html>

```

Soubor s CSS

```

a {color: black; text-decoration: none;margin-top:3px;font-size:11px;
a:visited {color: black; text-decoration: none }
a:link {color: black; text-decoration: none }
a:active {color: #75c6f2; text-decoration: none }
a:hover {color: #75c6f2; text-decoration: none;}
body {text-align: center;
background-image:url("lama_17.jpg");
font-family:MS Reference Sans Serif;
font-size:14px;
scrollbar-face-color: #ffffff; /* Barva scroll-panelu: barva těla */
scrollbar-track-color: #FFFFFF; /* barva plochy v pozadí */
scrollbar-arrow-color: #75c6f2; /* barva šipek */
scrollbar-highlight-color: #ffffff; /* barva levého a horního stínu */
scrollbar-3dlight-color: #75c6f2; /* barva levého a horního okraje */
scrollbar-shadow-color: #75c6f2; /* barva pravého a dolního stínu */
scrollbar-darkshadow-color: #ffffff; /* barva pravého a dolního okraje */ }
h1 {color:#75c6f2;
margin:5px;
margin-left:10px;
font-size:20px;}

```

```
h2 {color:#75c6f2;
font-size:15px;}
#all {width: 790px;
text-align: left;
margin: 0 auto;}
#logo {background-image:url("logo4.jpg");
width:790px;
height:145px;
background-repeat:no-repeat;
margin:0px;
padding:0px;}
#levy_sloupec_s_odkazy {float:left; margin-top:-1px;}
#odkaz1 {margin-left:0px;
margin-top:0px;
background-image:url("odkaz.jpg");
width:184px;
height:28px;
padding-top: 7px;
background-repeat:no-repeat;}
#odkaz2 {margin-left:0px;
margin-top:-6px;
background-image:url("odkaz.jpg");
width:184px;
height:28px;
padding-top: 7px;
background-repeat:no-repeat;}
#odkazy_text {margin-left:40px;}
#pravyroh {background-image:url("logoroh.jpg");
margin-left:0px;
width:178px;
height:231px;
float:left;}
#obsah {background-image:url("obsah.jpg");
width:607px;
height:440px;
float:left;
margin-top:-1px;
margin-left:-1px;
padding-right:5px;
overflow: auto;}
#cara {color:#75c6f2;}
.nabidka a {font-size:15px;}
```

8.2 Design webu 2

HTML soubor

```
<?
$prohlizec_info=$_SERVER['HTTP_USER_AGENT'];
if(eregi("MSIE", $prohlizec_info)){ $prohlizec="msie";}
if(!$menu){
$menu="uvod";
}
if(!$obsah){
$obsah="malirskeprace";
}
?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="content-type" content="text/html; charset=windows-1250">
<meta name="generator" content="PSPad editor, www.pspad.com">
<meta name="description" content="Lama, Malířské práce, Malíř, Malíř pokojů, ">
<meta name="keywords" content="Lama, Malířské práce, Malíř, Malíř pokojů, ">
<meta name="author" content="Jan Švarc">
<title>Lamapb.com - Pronájem plošin, malířské a lakýrnické práce</title>
<link rel="stylesheet" type="text/css" href="styly_lama.css">
</head>
<body>
<div id="all">
<div id="logo"></div>
<div id="levy_sloupec_s_odkazy">
<div id="odkaz1">
<b><a href="index.php?menu=uvod" id="odkazy_text" style="color:#75c6f2;">
if($menu=="uvod"){echo("color:#75c6f2");} ?>">úvodní strana</a></b>
</div>
<div id="odkaz2">
<b><a href="index.php?menu=plosiny" id="odkazy_text" style="color:#75c6f2;">
if($menu=="plosiny"){echo("color:#75c6f2");} ?>">pronájem plošin </a></b>
</div>
<div id="odkaz2">
<b><a href="index.php?menu=nabidka" id="odkazy_text" style="color:#75c6f2;">
if($menu=="nabidka"){echo("color:#75c6f2");} ?>">nabízíme </a></b>
</div>
<div id="odkaz2">
<b><a href="index.php?menu=napiste" id="odkazy_text" style="color:#75c6f2;">
if($menu=="napiste"){echo("color:#75c6f2");} ?>">napište nám</a></b>
</div>
<div id="odkaz2">
```

```

<b><a href="index.php?menu=reference" id="odkazy_text" style="<?
if($menu=="reference"){echo("color:#75c6f2");}?>">reference</a></b>
</div>
<div id="odkaz2">
<b><a href="index.php?menu=cenik" id="odkazy_text" style="<?
if($menu=="cenik"){echo("color:#75c6f2");}?>">cenik</a></b>
</div>
<div id="odkaz2">
<b><a href="index.php?menu=kontakt" id="odkazy_text" style="<?
if($menu=="kontakt"){echo("color:#75c6f2");}?>">kontakt</a></b>
</div>
<div id="pravyroh">
</div>
</div>
<div id="obsah">
<?php
if(isset($menu)){
$adresa= $menu . ".php";
include ("$adresa");
}
?>
</div>
</div>
</body>
</html>

```

Soubor s CSS

```

a {color: black; text-decoration: none;margin-top:3px;font-size:11px;
a:visited {color: black; text-decoration: none }
a:link {color: black; text-decoration: none }
a:active {color: #75c6f2; text-decoration: none }
a:hover {color: #75c6f2; text-decoration: none;}
body {text-align: center;
background-image:url("lama_17.jpg");
font-family:MS Reference Sans Serif;
font-size:14px;
scrollbar-face-color: #ffffff; /* Barva scroll-panelu: barva těla */
scrollbar-track-color: #FFFFFF; /* barva plochy v pozadí */
scrollbar-arrow-color: #75c6f2; /* barva šipek */
scrollbar-highlight-color: #ffffff; /* barva levého a horního stínu */
scrollbar-3dlight-color: #75c6f2; /* barva levého a horního okraje */
scrollbar-shadow-color: #75c6f2; /* barva pravého a dolního stínu */
scrollbar-darkshadow-color: #ffffff; /* barva pravého a dolního okraje */ }
h1 {color:#75c6f2;
margin:5px;
margin-left:10px;
font-size:20px;}

```

```
h2 {color:#75c6f2;
     font-size:15px;}
#all {width: 790px;
      text-align: left;
      margin: 0 auto;
      border:1px solid #75c6f2;}
#logo {background-image:url("pozadi2.png");
       width:790px;
       height:145px;
       background-repeat:no-repeat;
       margin:0px;
       padding:0px;}
#levy_sloupec_s_odkazy {float:left; margin-top:-1px;}
#odkaz1 {margin-left:0px;
         margin-top:1px;
         background-image:url("odkaz_2.jpg");
         width:184px;
         height:28px;
         padding-top: 7px;
         background-repeat:no-repeat;}
#odkaz2 {margin-left:0px;
         margin-top:-6px;
         background-image:url("odkaz_2.jpg");
         width:184px;
         height:28px;
         padding-top: 7px;
         background-repeat:no-repeat;}
#odkazy_text {margin-left:45px;}
#pravyroh {background-image:url("logoroh.jpg");
           margin-left:0px;
           width:178px;
           height:231px;
           float:left;}
#obsah {width:607px;
        height:440px;
        float:left;
        margin-top:-1px;
        margin-left:-5px;
        overflow: auto;
        border-left: 1px solid #75c6f2;}
#cara {color:#75c6f2;}
.nabidka a {font-size:15px;}
```

8.3 Design webu 3

HTML soubor

```
<?
$prohlizec_info=$_SERVER['HTTP_USER_AGENT'];
if(ereg("MSIE", $prohlizec_info)){ $prohlizec="msie";}
if(!$menu){
$menu="uvod";
}
if(!$obsah){
$obsah="malirskeprace";
}
?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="content-type" content="text/html; charset=windows-1250">
<meta name="generator" content="PSPad editor, www.pspad.com">
<meta name="description" content="Lama, Malířské práce, Malíř, Malíř pokojů, ">
<meta name="keywords" content="Lama, Malířské práce, Malíř, Malíř pokojů, ">
<meta name="author" content="Jan Švarc">
<title>Lamapb.com - Pronájem plošin, malířské a lakýrnické práce</title>
<link rel="stylesheet" type="text/css" href="styly_lama.css">
</head>
<body>
<div id="all">
<div id="logo">&nbsp;&nbsp;&nbsp;<h1 style="margin-top:-7px;">Lama s.r.o.</h1></div>
<div id="levy_sloupec_s_odkazy">
<div id="odkaz1">
<b><a href="index.php?menu=uvod" id="odkazy_text" style="<?
if($menu=="uvod"){echo("color:#75c6f2");} ?>">úvodní strana</a></b>
</div>
<div id="odkaz2">
<b><a href="index.php?menu=plosiny" id="odkazy_text" style="<?
if($menu=="plosiny"){echo("color:#75c6f2");} ?>">pronájem plošin </a></b>
</div>
<div id="odkaz2">
<b><a href="index.php?menu=nabidka" id="odkazy_text" style="<?
if($menu=="nabidka"){echo("color:#75c6f2");} ?>">nabízíme </a></b>
</div>
<div id="odkaz2">
<b><a href="index.php?menu=napiste" id="odkazy_text" style="<?
if($menu=="napiste"){echo("color:#75c6f2");} ?>">napište nám</a></b>
</div>
<div id="odkaz2">
```



```

<b><a href="index.php?menu=reference" id="odkazy_text" style="<?
if($menu=="reference"){echo("color:#75c6f2");}?>">reference</a></b>
</div>
<div id="odkaz2">
<b><a href="index.php?menu=cenik" id="odkazy_text" style="<?
if($menu=="cenik"){echo("color:#75c6f2");}?>">cenik</a></b>
</div>
<div id="odkaz2">
<b><a href="index.php?menu=kontakt" id="odkazy_text" style="<?
if($menu=="kontakt"){echo("color:#75c6f2");}?>">kontakt</a></b>
</div>
<div id="pravyroh">
</div>
</div>
<div id="obsah">
<?php
if(isset($menu)){
$adresa= $menu . ".php";
include ("$adresa");
}
?>
</div>
</div>
</body>
</html>

```

Soubor s CSS

```

a          {color: black; text-decoration: none;margin-top:3px;font-size:11px;}
a:visited  {color: black; text-decoration: none}
a:link     {color: black; text-decoration: none}
a:active   {color: #75c6f2; text-decoration: none}
a:hover    {color: #75c6f2; text-decoration: none;}
body {text-align: center;
background-image:url("lama_17.jpg");
font-family:MS Reference Sans Serif; font-size:14px;
scrollbar-face-color: #ffffff; /* Barva scroll-panelu: barva těla */
scrollbar-track-color: #FFFFFF; /* barva plochy v pozadí */
scrollbar-arrow-color: #75c6f2; /* barva šipek */
scrollbar-highlight-color: #ffffff; /* barva levého a horního stínu */
scrollbar-3dlight-color: #75c6f2; /* barva levého a horního okraje */
scrollbar-shadow-color: #75c6f2; /* barva pravého a dolního stínu */
scrollbar-darkshadow-color: #ffffff; /* barva pravého a dolního okraje */}
h1 {color:#75c6f2;
margin:5px;
margin-left:10px;

```

```
font-size:20px;}
h2 {color:#75c6f2;
font-size:15px;}
#all {width: 100%;
text-align: left;
margin: 0 auto;
border:1px solid #75c6f2;}
#logo {background-image:url("pozadi.jpg");
width:100%;
height:120px;
border-bottom:1px solid #75c6f2;
margin:0px;
padding:0px;}
#levy_sloupec_s_odkazy {float:left; margin-top:-1px;}
#odkaz1 {margin-left:0px;
margin-top:1px;
background-image:url("odkaz_2.jpg");
width:184px;
height:28px;
padding-top: 7px;
background-repeat:no-repeat;}
#odkaz2 {margin-left:0px;
margin-top:-6px;
background-image:url("odkaz_2.jpg");
width:184px;
height:28px;
padding-top: 7px;
background-repeat:no-repeat;}
#odkazy_text {margin-left:45px;}
#pravyroh {background-image:url("logoroh.jpg");
margin-left:0px;
width:178px;
height:231px;
float:left;}
#obsah {width:70%;
height:440px;
float:left;
margin-top:-1px;
margin-left:-5px;
padding-right:5px;
overflow: auto;
border-left: 1px solid #75c6f2;}
#cara {color:#75c6f2;}
.nabidka a {font-size:15px;}
```