

ZÁPADOČESKÁ UNIVERZITA V PLZNI
FAKULTA PEDAGOGICKÁ
KATEDRA VÝPOČETNÍ A DIDAKTICKÉ TECHNIKY

**NÁVRH A REALIZACE ŠABLON V REDAKČNÍM SYSTÉMU
Joomla!**
BAKALÁŘSKÁ PRÁCE

Michael Vaník

Informatika se zaměřením na vzdělávání, obor IT-TV

Vedoucí práce: Mgr. Tomáš Přibáň, Ph.D.

Plzeň, 2014

Prohlašuji, že jsem bakalářskou práci vypracoval samostatně s použitím uvedené literatury a zdrojů informací.

V Plzni, 15. dubna 2014

.....

Poděkování

Rád bych poděkoval vedoucímu práce, panu Mgr. Tomášovi Přibáňovi Ph.D., za odborné vedení, věcné rady a připomínky.

OBSAH

ÚVOD	1
1 REDAKČNÍ SYSTÉM JOOMLA!	2
1.1 REDAKČNÍ SYSTÉMY	2
1.2 REDAKČNÍ SYSTÉM JOOMLA!	3
1.3 FUNKCE	4
1.4 JAK JOOMLA! FUNGUJE	5
1.5 VÝVOJOVÉ VERZE	7
1.5.1 Joomla! 1.0 vs 1.5	7
1.5.2 Joomla! 1.5 vs 2.5	8
1.5.3 Joomla! 2.5 vs 3.0	9
1.6 MOŽNOSTI VYUŽITÍ	11
2 JOOMLA! – INSTALACE A ZÁKLADNÍ ČÁSTI	12
2.1 INSTALACE	12
2.1.1 Možnosti instalace	12
2.1.2 Instalace XAMPP	13
2.1.3 Instalace Joomla!	14
2.1.4 Instalace češtiny	16
2.2 ZÁKLADNÍ ČÁSTI SYSTÉMU	17
2.2.1 Front end	17
2.2.2 Back end	19
3 VYTVOŘENÍ VLASTNÍ ŠABLONY	22
3.1 CO JE TO ŠABLONA?	22
3.2 STRUKTURA ŠABLONY	22
3.3 INDEX.PHP	25
3.3.1 Zakázání přímého přístupu	25
3.3.2 DOCTYPE	25
3.3.3 <html>	25
3.3.4 <jdoc:include ... >	26
3.3.5 zdrojový kód hlavičky	26
3.4 TEMPLATE.CSS	28
3.4.1 wrapper	29
3.4.2 Header	29
3.4.3 Menu, content	30
3.4.4 Footer	30
3.4.5 Aktuální stav šablony	30
3.5 TEMPLATEDETAILS.XML	31
3.5.1 Znaková sada	31
3.5.2 Doctype	31
3.5.3 Rozšíření	31
3.6 NÁVRH GRAFICKÉHO LAYOUTU	33
3.6.1 Volba barevného schématu	33
3.6.2 Vytvoření grafiky	33
3.6.3 Nastýlování pomocí CSS3	34

3.7	ZOBRAZENÍ OBSAHU	38
3.7.1	Modules, component	38
3.7.2	Systemová hlášení	39
3.7.3	Dynamická změna rozvržení	39
3.8	DYNAMICKÉ PARAMETRY	41
3.8.1	Úprava textu zobrazovaného v hlavičce	42
3.8.3	Změna obrázku pod horizontálním menu	43
4	PRAKTICKÉ POUŽITÍ VYTVOŘENÉ ŠABLONY	44
4.1	VALIDACE	44
4.2	KONTROLA ZOBRAZENÍ V PROHLÍŽEČÍCH	45
4.3	NÁHLEDOVÉ OBRÁZKY	46
4.4	IKONA FAVICON.....	46
4.5	VYTVOŘENÍ ARCHIVU ZIP	46
4.6	INSTALACE	47
4.7	ODINSTALACE.....	48
4.8	NÁHLED S POZICEMI MODULŮ	48
4.9	ÚPRAVA SOUBORŮ	48
4.10	OVERRIDES	49
	ZÁVĚR.....	50
	RESUMÉ	51
	SEZNAM LITERATURY	52
	SEZNAM OBRÁZKŮ	56
	PŘÍLOHY	I

Úvod

Redakční systémy přinesly revoluční řešení pro tvůrce webu. Díky redakčním systémům odpadá nutnost znát HTML kód a kaskádové styly, které jsou při klasické tvorbě webu nezbytnou znalostí. Uživatel se tak může plně soustředit na tvorbu obsahu.

Téma bakalářské práce „Návrh a realizace šablon v redakčním systému Joomla!“ jsem si zvolil zejména proto, abych vyzdvihl jednu z největších výhod redakčních systémů – aplikaci šablon. Pomocí šablon lze snadno změnit nejen grafický vzhled webu, ale i jeho strukturu a rozložení. Uživatel tak nemusí překreslovat jednotlivé grafické prvky, jednoduše aplikuje požadovanou šablonu a web má během okamžiku zcela nový vzhled. Toto téma je aktuální z důvodu nárůstu použití redakčních systémů jak mezi profesionály, tak i laiky.

Bakalářská práce je rozdělena do dvou částí. Úvodní teoretickou část tvoří první kapitola, která je z části věnována redakčním systémům obecně a následně je pozornost zaměřena na redakční systém Joomla! a jeho teoretický rozbor. Jsou zde popsány základní funkce, vývojové verze a možnosti využití.

Druhá kapitola podrobně popisuje, jak úspěšně nainstalovat Joomla! a další aplikace potřebné k bezproblémovému provozu. Dále je zde popsána administrativní i uživatelská část webu, jejich jednotlivé prvky a využití.

Praktickou část tvoří třetí kapitola, která je věnována stěžejnímu tématu bakalářské práce – tvorbě šablon. Je zde popsána základní struktura každé šablony, její nejdůležitější soubory a náležitosti, bez kterých by se tyto soubory neobešly. Dále je zde popsáno, jakým způsobem rozčlenit šablonu na jednotlivé bloky a jak nastavit jejich vlastnosti. Za jednu z nejdůležitějších informací je považován návod, jak zobrazit vlastní obsah a moduly zaručující prezentaci dat.

Čtvrtá, závěrečná kapitola práce, je věnována praktickému využití vytvořené šablony. Jedná se zejména o validaci šablony, kontrolu správného zobrazení v prohlížečích, vytvoření instalačního souboru a samotnou instalaci.

1 REDAKČNÍ SYSTÉM JOOMLA!

1.1 REDAKČNÍ SYSTÉMY

V nedávné historii, kdy se ještě redakční systémy nevyužívaly, byli programátoři nuceni tvořit webové prezentace pomocí HTML kódu. Vytvořili několik stránek, které propojili odkazy a naformátovali na požadovaný vzhled. Když chtěli obsah prezentace změnit, museli otevřít stránku s HTML kódem a vše ručně přepisovat. Takovéto úpravy mnohdy trvaly déle než tvorba celé prezentace. [1] [2]

Revoluční řešení přinesly tzv. redakční systémy (CMS). Zkratka CMS pochází z anglických slov Content management system, což přeloženo do češtiny znamená „Systém pro správu obsahu“. Tento dlouhý název se většinou nepoužívá, obvyklejší je používat český termín redakční systém. Redakční systémy jsou programy, které umožňují jednoduchou a efektivní správu obsahu. Nejčastěji se využívají pro tvorbu a úpravu internetových stránek.

Hlavní výhodou je, že oproti klasické tvorbě stránek uživatel nemusí nutně znát značkovací jazyk HTML a může se plně soustředit na obsah. Obsah lze jednoduše vytvářet a upravovat v administrátorském rozhraní nazývaném back end. Výslednou podobu webu uživatel vidí v uživatelské části webu, v takzvaném front endu. Mezi administrátorskou a uživatelskou částí webu lze snadno přepínat.

Redakční systémy mají speciální strukturu, do které je z databáze nahráván text a vzhled je navržený pomocí šablony. Šablony je možné upravovat, instalovat nové, či si tvořit vlastní. Pokud chcete změnit celkový vzhled webu, stačí nahrát novou šablonu. Není třeba měnit strukturu a stylování jednotlivých elementů, což ušetří mnoho času. [1] [2]

Redakční systémy můžeme rozdělit do dvou skupin. První skupinu tvoří systémy vytvořené pomocí technologie ASP.NET (DotNetDuke, Kentico, Nemesis...). Druhou, podstatně rozsáhlejší skupinu, tvoří systémy vytvořené pomocí skriptovacího jazyka PHP (Drupal, WordPress, TextPattern, Contao, Umbraco).

Drtivá většina redakčních systémů potřebuje ke své činnosti databázový server. Nejpoužívanějším řešením je sada svobodného softwaru používaného jako platforma pro implementaci dynamických webových stránek tzv. LAMP. Jedná se o zkratku složenou ze slov:

- Linux – operační systém.
- Apache – webový server.
- MySQL – databázový systém.
- PHP – skriptovací jazyk.

Není nutností využívat přímo tyto výše uvedené technologie. Alternativou pro PHP je Perl nebo Python. MySQL se nechá nahradit pomocí systémů Oracle či PostgreSQL. Počáteční písmeno zkratky LAMP je závislé na používaném operačním systému. Pokud se jedná o Windows, kombinace zní WAMP, pokud o MacOS, jde o MAMP. Z toho všeho vyplývá, že při použití redakčních systémů nejsme vázáni na konkrétní operační systém. Ovládání a přístup k obsahu zajistí internetový prohlížeč.

[1] [2]

Redakční systémy nemusíme rozdělovat čistě z technologického hlediska, ale můžeme je rozdělit i z pohledu finančního na: komerční a nekomerční systémy. Hlavní výhodou komerčních redakčních systémů je to, že nabízí neustálou podporu a servis, avšak bývají poměrně drahé a soukromým osobám se vyplatí nahradit je nekomerčním (bezplatným) systémem. Nekomerční tzv. svobodné systémy se kvalitou zpracování vyrovnají systémům komerčním, mnohdy je kvalita a výčet nabízených funkcí dokonce vyšší. [1] [2]

1.2 REDAKČNÍ SYSTÉM JOOMLA!

Joomla! patří mezi jeden z nejrozšířenějších redakčních systémů. První verze byla zveřejněna v roce 2005, avšak kořeny tohoto systému sahají až do roku 2001. V dubnu 2001 uvolnila australská společnost Miro Corporation open-source program s názvem Mambo, ze kterého Joomla! vychází. [1] [2]

Systém Mambo byl na svou dobu velmi kvalitní a pokročilý software. Oblíbenost systému Mambo byla opravdu vysoká, vznikla komunita tisíce uživatelů, kteří mezi sebou volně šířili zdrojové kódy. Avšak v roce 2005 byl vývoj systému Mambo kvůli neshodám uvnitř organizace ukončen. [1] [2]

Netrvalo dlouho a světlo světa spatřil nový redakční systém – Joomla! 1.0. Jádro bylo tvořeno základními kódy systému Mambo, velké zlepšení přinesla modernizace uživatelského rozhraní, které působilo přívětivějším dojmem a přehledností. Joomla! zaznamenala rapidní nárůst počtu stahování, kdy během prvního roku si Joomla! stáhlo více než 2.5 milionů lidí. [1] [2]

V široké nabídce redakčních systémů lze najít mnoho a mnoho různých redakčních systémů. Joomla! ovšem disponuje jednou obrovskou výhodou a tou je její jednoduchost a uživatelské rozhraní, které zvládne ovládat i úplný začátečník. Můžete si stáhnout tisíce volně dostupných pluginů, s nimiž můžete své stránky vylepšovat a dodávat jim nové funkce.

Další obrovskou výhodou jsou dokonale navržené šablony, díky nimž bude Váš web působit dojmem, že jej tvořil profesionální webdesigner. Správa systému se provádí přes webový prohlížeč. Není zapotřebí instalovat editační programy, svůj web můžete spravovat odkudkoli, kde je k dispozici připojení k internetu. [1] [2]

V případě redakčního systému Joomla! se jedná o tzv. open-source program, neboli program s otevřeným zdrojovým kódem, ke kterému má uživatel plný přístup. Může jej upravovat a dál šířit, to vše legální cestou. Joomla! je šířena pod GNU GPL licenci (GNU General Public License). Kódy je možné upravovat a volně šířit pod podmínkou, že software bude šířen bezplatně. Toto opatření se týká i softwaru, který je od takto licencovaných produktů odvozen. [4]

1.3 FUNKCE

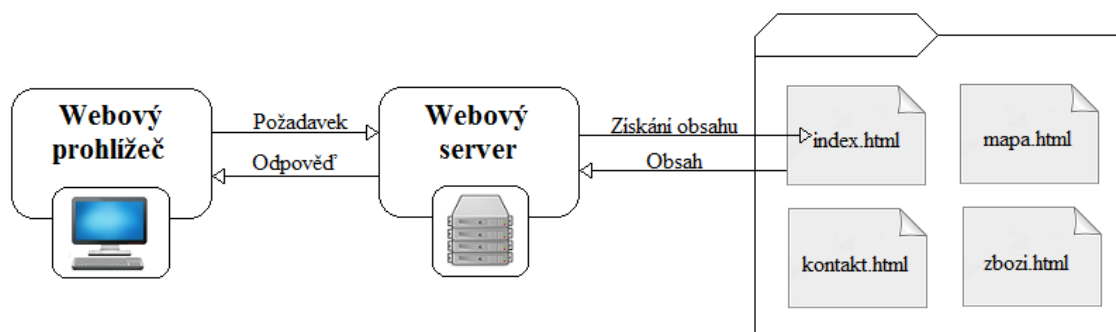
Joomla! nabízí mnoho vestavěných funkcí, díky nimž se řadí mezi nejprřednější redakční systémy ohledně použitelnosti a oblíbenosti. Mezi nejdůležitější funkce patří:

- Webové rozhraní, přes které se celý web spravuje.

- Grafický design, který se také spravuje přes webový prohlížeč – pro změnu vzhledu celého webu stačí zvolit jinou již vytvořenou šablonu, nebo si vytvořit šablonu vlastní. Jednotlivé sekce webu není nutné upravovat zvlášť, jedna šablona změní vzhled celého webu.
- Správa schvalování článků – každý článek, který přispěvatel napíše, je možné nejprve schválit, až poté je na webu zveřejněn. Lze tak zabránit vkládání nevhodného obsahu.
- Správa skupin uživatelů – uživatelům je možné přidělovat, upravovat, popřípadě odebírat práva k jednotlivým sekcím webu.
- Možnost automatického publikování článků – stačí nastavit datum a čas, kdy má být článek zveřejněn a o více není nutné se starat.
- Komunikace s protokoly FTP a LDAP.
- Podpora informačních kanálů RSS a Atom – technologie RSS a Atom (nástupce RSS) umožňují uživatelům přihlášení k odběru novinek z určité části webu.
- Možnost implementace reklamních bannerů.
- Tvorba vícejazykových webů. [1]

1.4 JAK JOOMLA! FUNGUJE

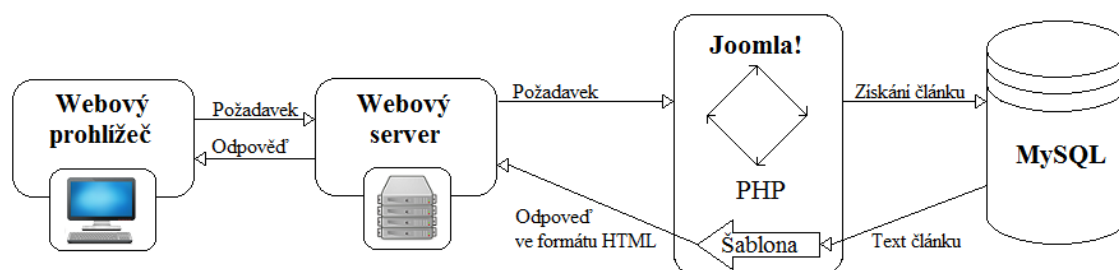
Joomla! funguje komplikovanějším způsobem než weby psané v HTML kódu. Běžné weby se v prohlížeči zobrazují poměrně jednoduchým způsobem. Webový prohlížeč požádá server o vydání stránky. Server získá statický obsah (HTML kód) a zobrazí jej v prohlížeči. Obsah uložený na serveru je totožný s obsahem vráceným prohlížečem. Princip fungování běžného webu psaného v HTML kódu je znázorněn na obrázku 1. [1]



Obrázek 1 - princip funkčnosti běžného webu [1]

Největší rozdíl mezi běžným webem a webem postaveným na redakčním systému je využití šablony, která je od obsahu zcela oddělená. Jedná se o základní design zobrazující webovou stránku, do které je načítán obsah z databáze. Pokud chcete použít pro web nový vzhled a zachovat stejný obsah, stačí zvolit jinou šablonu. [1]

V adresním řádku může adresa běžné webové stránky a stránky běžící na systému Joomla! vypadat stejně, ovšem načtení webu postaveného na redakčním systému je zcela odlišné a složitější. Webový prohlížeč si z webového serveru vyžádá stránku. Webový server načte část systému Joomla! a provede interpretaci. Systém Joomla! určí, jaký konkrétní obsah je požadován a podle toho si vyžádá článek z databáze. Dle zvolené šablony se obsah naformátuje. Systém Joomla! vytvoří obsah ve formátu HTML a odešle ho prohlížeči ve výsledné podobě, která je uživateli prezentována. Dynamicky tak poskytuje obsah načtený z databáze. Princip funkčnosti je znázorněn na obrázku 2. [1]



Obrázek 2 - průběh požadavku na zobrazení stránky v systému Joomla! [1]

1.5 VÝVOJOVÉ VERZE

Redakční systém Joomla! prošel několika vývojovými verzemi. Každá z verzí s sebou přinesla určité vylepšení, rozšíření funkcí či upravení uživatelského rozhraní. Verze se dělí na dvě základní skupiny. Na hlavní verze s dlouhodobou podporou označované zkratkou LTS (long term support) mající dobu podpory minimálně dva roky od data zveřejnění a na verze s krátkodobou podporou označované jako STS (short term support). Podpora verzí znamená vydávání rozšíření, bezpečnostních balíčků, hotfixů a různých aktualizací. Rozdíly mezi klíčovými verzemi jsou uvedeny v následující podkapitole. [5]

1.5.1 JOOMLA! 1.0 vs 1.5

Historicky první verze nesla označení Joomla! 1.0. Netrvalo dlouho a vývojáři zveřejnili druhou verzi – Joomla! 1.5. Funkčnost těchto dvou verzí se příliš neliší, obě nabízejí prakticky to samé. Po instalaci je k dispozici uživatelská a administrátorská část, která umožňuje správu veškerého obsahu. Obě verze obsahují vestavěné WISIWYG editory podobné kancelářským programům (Microsoft Word, OpenOffice...) sloužící ke snadné úpravě textů. Samozřejmostí je možnost rozšíření pomocí komponent a šablon. Zde má Joomla! 1.0 značnou výhodu – má k dispozici větší množství doplňků. [6]

Joomla! 1.5 přinesla velkou novinku – podporu kódování UTF-8. Na scénu se tak dostaly i menší evropské jazyky, například čeština a slovenština nebo východní „obrázkové“ jazyky (korejština, japonština, čínština). Verze 1.0 kódování UTF-8 také podporovala, ale často se vyskytly problémy, neboť se do překladu míchaly i jiné znakové sady. Čeština byla ve verzi 1.5 přidána do seznamu nabízených jazyků již v procesu instalace, tudíž je pro nezkušené uživatele instalace snadnější. [6]

Při provozu Joomla! 1.0 na webhostingu je možné narazit na četné potíže. Některé hostingové společnosti neumožňují měnit doporučená nastavení pro provoz, což může znamenat jisté problémy, mezi ně patří například nefunkční instalační proces, nutnost ruční instalace doplňků, potíže s ukládáním nastavení a celkový chod jako takový.

Joomla! 1.0 si často zabrala značnou část strojového výkonu serveru, což vyvrcholilo zákazem Joomla! 1.0 na některých webhostinzích používat. Tento problém byl řešen ve verzi 1.5, která dokázala využívat FTP vrstvu, která umožňuje uživatelům manipulaci se soubory ve svém datovém prostoru. [6]

1.5.2 JOOMLA! 1.5 vs 2.5

Verzím 1.6 a 1.7 není věnována taková pozornost, jednalo se pouze o krátkodobé projekty, které daly podnět ke vzniku stabilní verze 2.5. Joomla! 2.5 přinesla mnoho nových funkcí, mezi něž patří například:

Systém ACL (accesscontrol list) – seznam oprávnění k určitému objektu. V seznamu je určeno, kteří uživatelé mají práva přistupovat ke konkrétním objektům a jaké operace s nimi mohou provádět. Správce má nad uživateli plnou kontrolu a možnost jejich práva jednoduše upravovat. Tento systém odstranil jedno z největších omezení předchozích verzí. [7] [8]

Neomezený počet úrovní – předchozí verze umožňovaly obsah členit pouze do dvou úrovní (sekce a kategorie). Toto omezení verze 2.5 odstranila a je možné definovat tolik podkategorií, kolik je třeba. Toho lze efektivně využít například při tvorbě katalogu obsahujícího velké množství produktů. [7]

Nová správa aktualizací – není zapotřebí aktualizace vyhledávat a ručně instalovat, známkou toho, že jsou nové aktualizace k dispozici, zobrazuje ovládací panel. Postačí jedno kliknutí a aktualizace se sama nainstaluje. [7]

Zdokonalené vyhledávání disponující automatickým dokončováním slov. Při zadání slova v nesprávném tvaru je zobrazen návrh slova, které měl uživatel nejspíše na mysli. Podobné inteligentní vyhledávání je známé například z vyhledávače google.com. Výsledky hledání jsou řazeny dynamicky na základě jejich četnosti, nejvýznamnější obsah je uveden v horní části výsledků. [7] [9]

reCAPTCHA – bezplatná služba, která pomáhá digitalizovat tištěné dokumenty. CAPTCHA je metoda sloužící k odlišení uživatelů od robotů, které napadají webové stránky za účelem spamu. Využívá toho, že lidé dokážou přečíst i zkreslený text, roboti nikoliv. CAPTCHA je denně využívána milióny lidí po celém světě. Čas mezi přečtením zkresleného textu, jeho rozluštěním a vepsáním do formuláře činí jen několik vteřin, ale v celosvětovém souhrnu jsou to tisíce hodin. Tento čas právě reCAPTCHA využívá. Tištěné dokumenty jsou skenovány a transformovány do textové podoby pomocí OCR (optical character recognition) systémů. OCR systémy nejsou dokonalé a některá slova nedokážou správně rozpoznat. Slovo, které nelze správně rozpoznat je uživateli zobrazeno spolu s jiným známým slovem. Pokud uživatel zadá známé slovo ve správném tvaru, předpokládá se, že správně zadal i slovo nerozpoznané systémem OCR. Pro vyšší spolehlivost jsou tyto dvě slova poskytnuta dalším uživatelům a po několikanásobné shodě je nenaskenované slovo prohlášeno za rozpoznané. [10] [11]

1.5.3 JOOMLA! 2.5 vs 3.0

Joomla! 3.0 přinesla další vylepšení, které většina uživatelů zajisté ocení. Mezi ně patří například:

Responsivní technologie – jedná se o technologie, které design stránky přizpůsobí podle toho, na jakém zařízení je zobrazován. Počet uživatelů využívajících smartphony, iPady, tablety a podobná zařízení ke správě svých webů vybudovaných v redakčním systému Joomla! rok od roku roste. U předchozích verzí Joomla! byla správa webů z mobilních zařízení také možná, ale administrativní ani uživatelská část nebyla zobrazována v optimálním rozlišení a v ideální struktuře. Tento problém Joomla! 3.0 odstranila, tudíž je správa webu možná z jakéhokoli zařízení připojeného k internetu. [12] [13] [14]

LESS CSS – LESS je knihovna, která je od Joomla! verze 3.0. začleněna do její struktury. U běžných CSS stylů může u rozsáhlejších projektů nastat problém, jelikož se programátor v nich začne ztrácet. Klasické CSS styly nenabízí totiž žádné možnosti, které by umožňovaly strukturování uvnitř souboru a mohou tak vznikat dlouhé bloky s výčtem vlastností, které chceme konkrétnímu selektoru nastavit. LESS zavedl do CSS některé vlastnosti známé z jiných programovacích jazyků, například makra, výrazy, proměnné, funkce apod. Nepřidává do CSS nové možnosti popisu vzhledu, ale značně zjednodušuje a hlavně zpřehledňuje jejich zápis. Všechny šablony nasazené jak pro administraci, tak i pro uživatelskou část, obsahují díky LESS CSS méně stylovacích souborů, což vede k ušetření datového prostoru. [13] [14] [15] [16]

Rychlejší instalace – celý instalační proces Joomla! 3.0 je zjednodušen do tří kroků oproti verzi 2.5, jejíž instalace obsahovala kroků sedm. Nový je i design instalace nabízející jednoduché a přehledné instalační prostředí. Instalátor umí rozpoznat Váš jazyk, pokud je dostupný jazykový balíček, je Vám automaticky nabídnut. Celá instalace trvá zhruba minutu, což je minimum s ohledem na to, kolik funkcí, možností a rozšíření Joomla! 3.0 poskytuje. [13]

Drag and drop – Joomla! 3.0 nabízí používání oblíbené technologie drag and drop (česky „táhni a pusť“). Lze tak v administrátorské části pohodlně přeskupovat jakýkoli obsah. Použití je opravdu jednoduché, daný objekt se nejdříve uchopí stiskem levého tlačítka myši a tažením se přemístí (drag), poté se pustí (drop) na požadovaném místě. Není nutné používat přednastavené filtry řazení (titulek sestupně, kategorie vzestupně, autor sestupně atd.), obsah si může každý uspořádat dle svých představ. Jedná se o užitečnou technologii, která zvyšuje použitelnost aplikace a zpříjemňuje její užívání. [13] [17] [18]

Prázdné články – Joomla! 3.0 umožňuje uložení prázdných článků. To nebylo ve starších verzích možné a každý článek musel obsahovat nějaký text, i kdyby se mělo jednat o jediný znak. [13] [14]

1.6 MOŽNOSTI VYUŽITÍ

Možnosti využití systému Joomla! jsou opravdu široké. Ať je třeba vytvořit jakkoli zaměřený web či webovou aplikaci, v drtivé většině případů to Joomla! umožní. Zde je výčet jen několika možných využití:

E-shop – internetová aplikace umožňující komunikaci mezi vlastníkem webu a zákazníkem sloužící k prodeji jakýchkoliv produktů. Joomla! umožňuje vytvoření e-shopu ve zlomku času oproti klasickému programování. Při procházení katalogem zboží je možné využít již vytvořené filtrování, které usnadňuje a urychluje vyhledávání.

Školní web – pokud si potřebuje webové stránky vytvořit základní, mateřská či vysoká škola, Joomla! poskytuje ideální řešení a web může vytvořit jakýkoli technicky zdatnější zaměstnanec nebo i student.

Osobní stránky – jakkoli zaměřené osobní stránky lze vytvořit za několik málo okamžiků. Problémem nejsou ani profesionálně vypadající fotogalerie, jejichž tvorbu Joomla! nabízí.

Online časopis – pokud nechcete shánět vydavatele pro svůj časopis a zároveň riskovat neúspěch ve vysoké konkurenci, zvolte online časopis. Oproti klasickému tištěnému časopisu nabízí značnou výhodu – čtenáři mohou přidávat ke článkům komentáře, což zajišťuje přímou komunikaci mezi autorem a čtenářem.

Aukční web – Joomla! umožňuje vytvoření webu sloužícího jako internetová aukce. Zájemci v reálném čase podávají cenové nabídky, čímž se určuje konečná cena produktu.

Sociální síť – Joomla! nabízí možnost vytvoření vlastní sociální sítě, jenž se v posledních letech těší značného rozmachu. [1]

2 Joomla! – instalace a základní části

2.1 instalace

2.1.1 Možnosti instalace

Instalaci Joomla! nelze vykonat bez instalačních souborů. Ty jsou ke stáhnutí na domovských stránkách www.joomla.org. Po prostudování výhod vydaných verzí jsem se rozhodl pro nejnovější dostupnou verzi – Joomla! 3.2.1.

Samotnou instalaci lze provést dvěma způsoby:

Instalace na webový hosting – tento způsob je určen pro reálný provoz stránek. Celý web je umístěn na webovém serveru, který provozovateli pronajímá datový prostor. Před výběrem hostingu je nutné si ověřit, zdali podporuje redakční systém Joomla!. Pro budoucí spolupráci se dále doporučuje zjistit, jestli disponuje kvalitní technickou podporou, která je především pro nezkušené uživatele k nezaplacení.

Seznam několika kvalitních a ověřených hostingů, které nabízejí bezproblémový provoz webů vytvořených pomocí Joomla!:

- websupport.cz
- active24.cz
- cesky-hosting.cz
- onebit.cz
- webhosting-c4.cz
- savana.cz
- hostime.cz

Instalace na lokálním počítači – tuto možnost, která se používá pro vývoj a úpravu stránek, využiji v této bakalářské práci. [19]

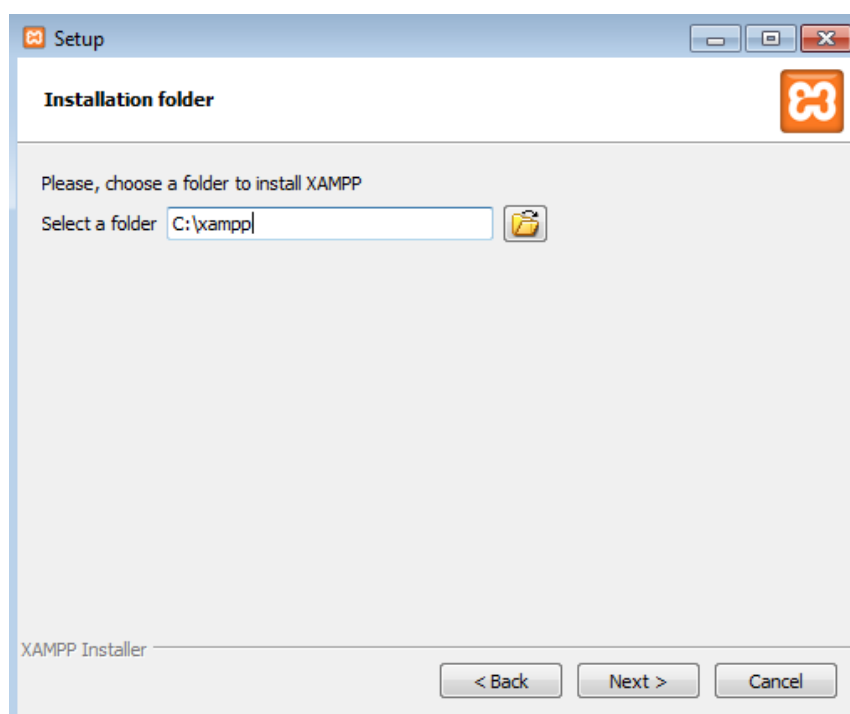
2.1.2 INSTALACE XAMPP

Jelikož v úvodu bylo uvedeno, že k provozu redakčního systému je zapotřebí použít tzv. LAMP (operační systém, webový server, databázový systém a skriptovací jazyky), je výhodné využít multiplatformní balík XAMPP, který umožňuje rychlou a snadnou instalaci Apache (webový server), MySQL (databázový systém), PHP či Perl (skriptovací jazyk) a dalších doplňků. Název XAMPP je tvořen z: X – multiplatformita, A – Apache, M – MySQL, P – PHP, P – Perl. [33]

XAMPP stáhneme z oficiálních stránek www.apachefriends.org. Zvolíme nejnovější dostupnou verzi – XAMPP 1.8.3. Instalaci provedeme v několika krocích, nezabere více než několik minut.

Spustíme instalační soubor `xampp-win32-1.8.3-1-VC11-installer.exe`. V dialogovém okně zvolíme komponenty, které chceme nainstalovat (defaultně jsou zaškrtnuté všechny komponenty) a klikneme na tlačítko „Next“. Pro správnou funkci Joomla! nejsou všechny komponenty potřebné, ale jejich nainstalování funkčnost nijak neovlivní a později se mohou hodit v jiných projektech.

Dalším krokem je výběr složky, kam se má XAMPP nainstalovat. V našem případě ponecháme adresář `C:\xampp` (ukázka je na obrázku 3). Základní nastavení je hotové, klikneme na tlačítko „Next“. Instalátor se už o vše postará za nás.



Obrázek 3 - instalace XAMPP

2.1.3 INSTALACE JOOMLY!

Prvním krokem je stažení Joomla!. Na stránkách www.joomla.org rozklikneme sekci "Download" a stáhneme instalační balíček (Joomla_3.2.1-Stable-Full_Package.zip).

Soubor s příponou .zip rozbalíme například pomocí programu TotalCommander do připravené dočasné složky, kterou můžeme pojmenovat „Joomla“. Složka musí být umístěna v adresáři C:/xampp/htdocs/název_složky. V našem případě konkrétně C:/xampp/htdocs/Joomla. Do složky htdocs se umísťují soubory, které chceme spouštět na lokálním serveru.

Joomla! 3.0 vyžaduje minimálně PHP 5.3.1 a MySQL 5.1, tyto podmínky nemusíme více řešit, jelikož jsme nainstalovali XAMPP v nejnovější verzi, která zahrnuje MySQL5.6.14 a PHP 5.5.6. [21]

Samotnou instalaci spustíme přes internetový prohlížeč zadáním následující adresy: localhost/Joomla/installation/. Instalace se skládá ze tří kroků:

1) Instalace nepodporuje češtinu, takže ponecháme zvolený anglický jazyk, nebo zvolíme jakoukoli jinou nabízenou možnost. Tato volba záleží na uživateli a jeho znalostech cizích jazyků. Vytvoříme administrátorský účet a vyplníme základní informace o stránkách. Hlavní konfigurace obsahuje následující položky:

- Sitename – název stránek.
- Description – krátký popis stránek nebo klíčová slova.
- Admin Email – kontaktní e-mail na administrátora.
- Admin Username – jméno, kterým se bude administrátor přihlašovat ke správě webu.
- Admin Password – heslo pro přihlášení do administrátorské sekce.
- Confirm Admin Password – stejné heslo jako v položce „Admin Password“. Tato kontrola slouží k zamezení problémů při přihlašování, kdyby se uživatel při vyplňování hesla překlepl a nenapsal ho správně. [22]

Po vyplnění povinných údajů klikneme na tlačítko „Next“.

- 2) Tento krok obsahuje přihlašovací údaje k databázi. Databázi vytvoříme přes systém phpMyAdmin, umožňuje jednoduchou správu databází a nastavování práv uživatelům. Přístup zajišťuje internetový prohlížeč zadáním adresy: localhost/phpmyadmin. Vyplníme následující položky:

Database Type – zvolíme MySQLi. MySQL i MySQLi je pořád jedna a ta samá databáze, pouze se liší používané PHP funkce. MySQLi je nejnovější ovladač umožňující i procedurální použití. V jednom SQL dotazu může být zahrnuto více dotazů, což má pozitivní vliv na rychlost. [20]

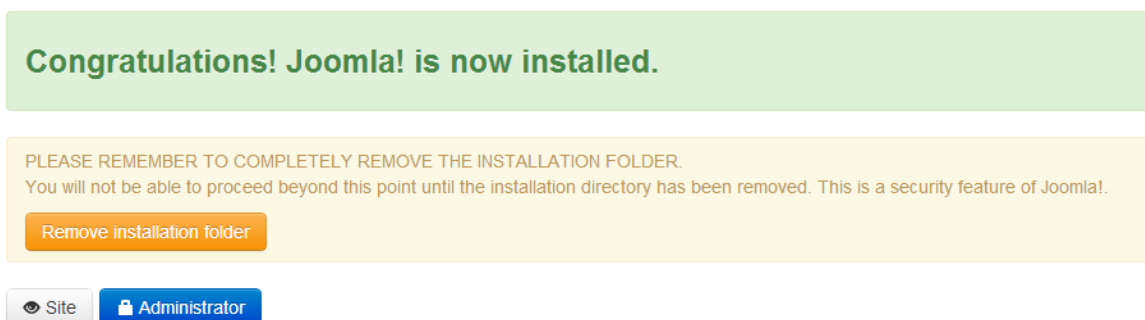
- Host Name – v našem případě ponecháme „localhost“.
- Username – jméno uživatele, který bude k databázi přistupovat.
- Password – heslo pro přihlášení k databázi.
- Database Name – název databáze.

Table Prefix – nejsnazší je ponechat instalátorem náhodně vygenerovanou hodnotu. Můžeme zvolit i vlastní hodnotu obsahující pouze alfanumerické znaky a končící podtržítkem. Prefix je předpona umožňující v jedné databázi rozlišit jednotlivé konkrétní tabulky.

Old Database Process – pokud již databáze zvolená v poli „Database Name“ obsahuje nějaká data, můžeme je zálohovat (Backup) nebo vymazat (Remove).

- 3) V části „Finalisation“ můžeme zvolit, zda chceme nahrát ukázková data. Pokud nechceme, ponecháme defaultně vybranou hodnotu „None“. V části „Overview“ si můžeme na mail nechat zaslat naše nastavení a hesla. Nastavení je vhodné zaslat pro případ, že bychom zapomněli, jaké údaje jsme v 1. kroku zadávali. Z bezpečnostních důvodů se zasílání hesla nedoporučuje, po prolomení našeho emailu by měl útočník všechny potřebné údaje k napadení webu. Další části zobrazují námi zadané údaje, pokud vše souhlasí, klikneme na tlačítko „Install“, čímž instalaci spustíme. [22]

Po dokončení instalace se zobrazí hlášení oznamující úspěšné nainstalování systému Joomla!. Klikneme na tlačítko „Remove installation folder“, čímž se automaticky smaže instalační složka (viz obrázek 4). Pro vstup do administrátorské části klikneme na tlačítko „Administrator“ a přihlásíme se pomocí údajů zadaných při instalaci.



Obrázek 4- odstranění instalační složky

2.1.4 INSTALACE ČEŠTINY

Instalační balíček pro naši verzi stáhneme na adrese <http://forge.joomla.org/gf/project/joomlacz/frs/>. Do internetového prohlížeče vložíme adresu pro přihlášení do administrátorské části, v našem případě (localhost/Joomla/administrator) a přihlásíme se pomocí údajů zadaných při instalaci. V horním menu zvolíme záložku „Extensions – Extension Manager“. Dále v sekci „Upload Package File“ klikneme na tlačítko procházet a vybereme stažený balíček. Kliknutím na tlačítko „Upload & Install“ spustíme nahrávání. Po dokončení nahrávání se zobrazí hlášení „Installing package was succesfull“ oznamující úspěšnou instalaci češtiny.

Opět se vrátíme do horního menu, klikneme na „Extensions“ a poté na záložku „Language Manager“, které umožňuje správu jazyků. V sekci "Installed - Site" a "Installed - Administrator" nastavíme češtinu jako výchozí jazyk (kliknutím na tlačítko „Default“ – viz obrázek 5). Tím je instalace češtiny dokončena a uživatelská i administrátorská část webu bude plně česky. [23]

Jazyk	Značka jazyka	Umístění	Výchozí	Verze	Datum
<input type="radio"/> Czech (Czech Republic)	cs-CZ	Veřejná část	<input checked="" type="checkbox"/>	3.2.1.1	2013-11-18
<input type="radio"/> English (United Kingdom)	en-GB	Veřejná část	<input type="checkbox"/>	3.2.2	2013-03-07

Obrázek 5 - výběr jazyka

2.2 ZÁKLADNÍ ČÁSTI SYSTÉMU

2.2.1 FRONT END

Front end je uživatelské rozhraní, které vidí běžní návštěvníci webu. Vzhled závisí na použité šabloně a funkcionalitu ovlivňují nainstalovaná rozšíření. Návštěvníci nedisponují žádnými pokročilými funkcemi a mohou dle jejich konkrétních práv psát články a zasílat je správci ke schválení. [24]

Obsah

Obsah je většinou to, kvůli čemu webové stránky navštěvujeme. Může nabývat nejrůznějších forem, nejčastěji se jedná o text, ale obsahem je i obrázek, video, hudba, mapa atd.

Obsah může být statický (neměnný) nebo dynamický (umožňující úpravy). Starší webové stránky byly tvořeny staticky, ale dnešní společnost si svůj život nedokáže bez dynamických webů představit. V případě dynamických stránek lze obsah odeslat emailem, převést do formátu pdf, zvětšit velikost písma apod. [25]

Menu

Menu je nabídka o libovolném množství položek sloužící ke snadné navigaci na webu. Uživatelské rozhraní obsahuje většinou horní a hlavní menu.

- Horní menu – je umístěno v horní části stránky a je navrženo tak, aby uživateli umožňovalo rychlý přístup k nejdůležitějšímu obsahu.
- Hlavní menu – nejdůležitější navigační část webu. Mělo by se na všech stránkách zobrazovat na stejné pozici a obsahovat odkaz na úvodní stránku. [25]

Nejnovější a nejoblíbenější články

V této části můžeme články zobrazovat mnoha způsoby, například vypsát pouze titulek nebo titulek s několika větami článku. Joomla! tuto část automaticky aktualizuje. Nejnovější články jsou zobrazovány na základě data zveřejnění a nejoblíbenější dle toho, kolikrát byly přečteny.

Přihlašovací formulář

Formulář slouží k přihlášení návštěvníků webu. Každý uživatel disponuje určitými právy, která mu umožňují vkládat, upravovat, mazat články atd. Přihlašovací formulář umožňuje rozdělit web do veřejné a administrátorské oblasti. Měl by obsahovat odkaz na založení nového uživatelského účtu a možnost zaslat zapomenuté heslo na email. [25]

Vyhledávání

Užitečná funkce, která uživateli umožňuje vyhledat konkrétní obsah. Po zadání hledaného slova zkoumá vyhledávací algoritmus všechna slova ve všech dokumentech a porovnává je se slovem hledaným. Výsledky vyhledávání je možné filtrovat dle typu, autora, kategorie a jazyku. [26]

Ostatní

Možnosti redakčního systému Joomla! jsou takřka neomezené. Výše uvedené části jsou výčtem toho hlavního, co se v uživatelském rozhraní nachází. Dále je možné na web umístit například anketu, počítadlo přístupů, změnit dle libosti banner, vložit placenou reklamu atd.

Uživatelé

Každý uživatel, který chce na web přidávat články, se musí registrovat. Registrace se provádí přes odkaz „Vytvořit účet“, který se nachází v dolní části přihlašovacího formuláře. Registrace je opravdu jednoduchá, stačí vyplnit přihlašovací jméno, e-mail a dvakrát stejné heslo. Na zadanou e-mailovou adresu bude zaslána zpráva obsahující unikátní odkaz, kterým se provede aktivace účtu. Uživatelská část webu (front end) obsahuje následující skupiny uživatelů s různými právy: [27]

- Registrovaní – běžní uživatelé, kteří mohou po přihlášení číst články s omezeným přístupem. Nemohou vkládat nové články a jakkoli měnit obsah webu.
- Autor – uživatelé této skupiny mají právo vkládat nové články a nastavovat, kdy se provede jejich zveřejnění. Před zveřejněním musí být schváleny uživatelem s vyššími právy. Dále pak mohou své dosavadní články upravovat. [1]

- Editor – tito uživatelé mohou vkládat nové články a upravovat své články a navíc mohou upravovat články ostatních uživatelů. Nové články musí být schváleny uživatelem s vyššími právy, aby byly zveřejněny.
- Publisher – uživatelé skupiny „Publisher“ mají stejná práva jako „Editor“ a navíc mohou schvalovat či zamítnout články předchozích skupin a tím ovlivnit jejich zveřejnění. [1]

2.2.2 BACK END

Back end neboli administrátorské rozhraní slouží k veškeré administraci webu. Tato část by měla být dobře zabezpečena, protože neoprávněný uživatel by mohl pár kliky dovézt celý web k záhubě. Back end otevřeme zadáním následující adresy do prohlížeče: `název_domény/administrator`. V našem případě je `název_domény` „localhost“, neboť neprovozujeme web na webovém serveru, ale na virtuálním lokálním serveru vytvořeném na použitém počítači. [24]

Menu

Stejně jako v uživatelské části, i zde je menu nejdůležitějším prvkem stránky. Zajišťuje přístup ke všem nástrojům pro správu webu. Vpravo od menu se nachází odkaz na uživatelskou část webu sloužící k rychlému náhledu a kontrole provedených změn a tlačítko sloužící k ohlášení či úpravě účtu. Pod menu jsou zobrazení přihlášení uživatelé, nejčtenější a nejnovější články. Díky tomuto přehledu může administrátor okamžitě po přihlášení vidět, zda se na webu děje něco zásadního.

Komponenty

Komponenty jsou nejdůležitějším rozšířením Joomla!. Jedná se o speciální rozšiřující prvek napsaný nejčastěji v jazyce PHP, který má za úkol přidat nějakou novou funkci. Na komponenty můžeme pohlížet jako na miniaplikace, které generují celou stránku. Často obsahují složité a rozsáhlé administrační rozhraní. Mohou být tvořeny vývojovými týmy Joomla!, anebo si je vytváří programátoři na základě toho, jakou konkrétní funkci potřebují svému webu dodat. Každá stránka může obsahovat pouze jednu komponentu. To je jeden z největších rozdílů oproti modulům. [1]

Moduly

Moduly s komponenty často spolupracují. Zajišťují zobrazení obsahu, který je komponenty zpracováván. Jedna stránka může obsahovat více modulů, každý z nich má v šabloně přesně definované umístění. Pomocí komponenty můžeme například naprogramovat diskusní fórum, jehož správné zobrazení nám zajistí právě moduly. Dalším typickým příkladem mohou být moduly zobrazující počet přihlášených uživatelů, poslední přečtený článek či informaci o nejprodávanějším produktu. [1] [24]

Pluginy

Pluginy, neboli zásuvné moduly, jsou kusy programového kódu připojeného k Joomla! za účelem rozšíření funkčnosti. Pluginy mohou být využity například pro formátování textu článků či mohou určovat, v jakém formátu bude zobrazován datum v kalendáři. Dalším využitím může být filtrování obsahu pro přihlášené či nepřihlášené uživatele. [24] [25]

Šablony

Šablony poskytují celému webu komplexní vizuální úpravu. Určují barvu, velikost a typ písma, jaké obrázky budou použity jako pozadí, mezery mezi slovy, zarovnání textu, umístění jednotlivých částí webu, jinými slovy vše, co ovlivňuje vzhled stránek. Šablony se skládají minimálně z jednoho HTML souboru definujícího strukturu stránky a jednoho CSS souboru popisujícího design. Pro změnu designu celého webu stačí v administraci změnit použitou šablonu. Žádné přepisování kódu a nastavování dílčích vlastností, o vše se postará šablona. Pokud nevyhovuje některá z dostupných šablon, můžete si vytvořit vlastní přesně podle svých představ. [24] [25]

Články

Důležitou částí webu jsou samozřejmě články. Ty utváří obsah, který se bude návštěvníkům zobrazovat. Články jsou tvořeny převážně texty, ale mohou obsahovat i obrázky či odkazy. Články je do databáze možné uložit dvěma způsoby, buď vložením z uživatelského rozhraní, nebo přímým uložením přes administrativní prostředí. Každý článek musí mít přiřazenou určitou kategorii a sekci. To nám umožňuje přehledné třídění při hledání konkrétního článku. Sekce dělí články na nejvyšší úrovni a obsahuje jednotlivé kategorie. Kategorie zahrnují články příbuzných témat. Příkladem může být

sekce „Osobní automobily“ obsahující kategorie „Škoda“, „Fiat“, „Renault“. Třetí úroveň by pak tvořily jednotlivé názvy automobilů.

S články úzce souvisí WYSIWYG editory umožňující vkládání textu podobně jako pomocí textových procesorů. Bez těchto editorů bychom byli nuceni formátovat text klasickými HTML značkami a redakční systémy by zde ztrácely svůj význam. [24] [28]

Uživatelé

Uživatelé administrační části disponují vyššími oprávněními než uživatelé části „front end“. Uživatelé „back end“ se dělí na tři typy správců s oprávněními upravovat šablony, vytvářet nové sekce a kategorie či přidělovat práva ostatním uživatelům.

- Manažer – má ze správců nejnižší oprávnění. Mohou potvrzovat nově registrované uživatele a provádět základní údržbu webu, ale mají omezený přístup k ovládacímu panelu.
- Administrátor – druhé nejvyšší postavení v celé hierarchii uživatelů. Administrátoři mohou instalovat nová rozšíření, měnit rozvržení webu, upravovat oprávnění všech uživatelů (kromě Superadministrátorů) atd.
- Superadministrátor – má nejvyšší postavení ze všech uživatelů. Jedná se o jakéhosi krále webu bez jakýchkoli omezení. Tento účet se vytvoří při instalaci systému Joomla!. [1]

3 VYTVOŘENÍ VLASTNÍ ŠABLONY

3.1 CO JE TO ŠABLONA?

Šablona (anglicky template) je několik souborů, které spolu spolupracují a udávají celkový vzhled internetové stránky. Estetická podoba webu je velmi důležitá. Vzhled musí návštěvníka zaujmout a zároveň umožňovat snadnou orientaci. V každé šabloně je přesně nadefinováno, které prvky se na stránce zobrazí a jaké bude jejich umístění. Každá šablona je zcela oddělená a nezávislá na obsahu a pouze určuje, jak bude obsah prezentován. Jedním kliknutím tak můžeme stejný obsah zobrazit ve zcela odlišném grafickém rozvržení. Šablony se dělí na dva základní typy:

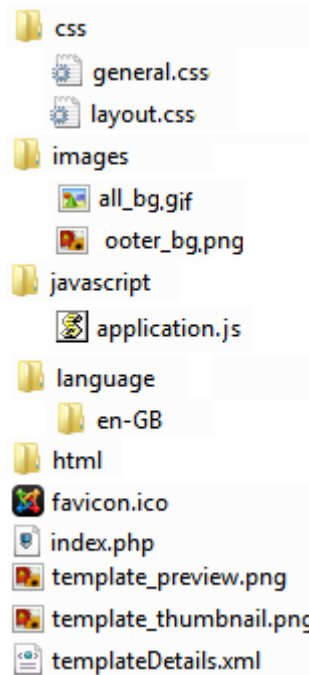
- Šablony pro front end – určují vzhled uživatelské části webu.
- Šablony pro back end – šablony pro administrátorské rozhraní.

Šablony můžeme získat několika způsoby. Nejjednodušším způsobem je stažení z Internetu. Existuje množství webů poskytujících šablony (jak zdarma, tak i placené). Dalším způsobem je vytvoření šablony na zakázku od specializované firmy, což ovšem bývá finančně náročnější. Jinou cestou k získání šablony je úprava šablony již hotové, kdy si přizpůsobíme zobrazovaný obsah a grafický design. Předposledním způsobem je vytvoření šablony pomocí WISIWYG editorů, kdy přímo při tvorbě vidíme, jak bude nový design vypadat. Mezi tyto editory patří například Adobe Dreamweaver a Microsoft Expression Web. Posledním způsobem je vytvoření vlastní šablony, kdy naše kroky od začátku směřují k určitému vzhledu a funkčnosti. Posledním zmiňovaným způsobem se budeme zabývat v této a následující kapitole a ukážeme, jak šablony vytvořit od úplného základu.

3.2 STRUKTURA ŠABLONY

Soubory šablon pro uživatelskou část (front end) se nachází v adresáři templates, kde má každá šablona svůj podadresář totožný s jejím názvem. Šablony pro administrátorskou část (back end) jsou uloženy v adresáři administrator/templates.

Všechny soubory tvořící šablonu jsou ve většině případů pro přehlednost členěny do složek. Stromová struktura šablony může vypadat jako na obrázku 6.



Obrázek 6 - příklad stromové struktury šablony

- Soubor *index.php* – základní soubor kombinující kód HTML a PHP. Sdružuje všechny používané soubory a řídí zobrazení textu, modulů a komponent.
- Složka *css* – v této složce se nachází CSS styly používané šablonou. Šablona může používat jeden styl obsahující deklarace všech vlastností a pozic, nebo zvláštní styl pro jednotlivé části webu.
- Soubor *templateDetails.xml* – soubor obsahující veškeré informace o šabloně, seznam souborů a pozic, které jsou v šabloně obsaženy. Bez tohoto souboru by nebyla instalace systému Joomla! možná.
- Složka *images* – zde jsou uloženy všechny obrázky zobrazované šablonou (kromě náhledových obrázků *template_preview.png* a *template_thumbnail.png*).
- Složka *javascript* – do této složky se nejčastěji ukládají skripty používané v šabloně. Skripty lze uložit do libovolné složky, důležité ovšem je uvedení správné cesty.

- Složka *language* – složka obsahující jazykové soubory šablony. Každý jazyk se skládá ze dvou souborů, jeden s příponou *.ini* (uživatelská část webu) a druhý s příponou *.sys.ini* (administrativní část webu).
- Složka *html* – složka, která obsahuje uživatelsky upravené výstupy systému Joomla! pomocí tzv. Overrides - přepsání. Kód umístěný v této složce přepisuje standardní výstup.
- Soubor *favicon.ico* – malá ikonka zobrazující se v adresním panelu vlevo od názvu stránky. Jedná se o obrázky o rozměrech 16x16 pixelů, které lze vytvořit v mnoha grafických programech.
- Soubor *template_thumbnail.png* – obrázek nejčastěji o rozměrech 206x150 pixelů, zobrazovaný ve Správci šablon jako malý náhled šablony. Rozměry nejsou pevně dány, ale je vhodné dodržovat jeden rozměr pro všechny šablony.
- Soubor *template_preview.png* – velké náhled šablony, který se zobrazí po kliknutí na malý náhled šablony (výše zmiňovaný soubor *template_thumbnail.png*). Obvyklé rozměry jsou 640x388 pixelů. [29]
- Naší vlastní šablonu pojmenujeme *moje_sablona*, tudíž adresář, ve kterém bude umístěna, bude pojmenovaný *moje_sablona*. Joomla! rozlišuje velká a malá písmena, proto je nutné formát názvů přesně dodržet (šablona *moje_sablona* nemůže být uložena například v adresáři *Moje_Sablona*). V adresáři *moje_sablona* vytvoříme tři základní soubory, ze kterých se šablona skládá:
 - *index.php*
 - *template.css*
 - *templateDetails.xml*

Můžeme využít libovolný textový editor nebo speciální vývojová prostředí určená pro jednoduchou editaci kódu (Eclipse, jEdit, PsPad atd.). Vývojová prostředí mají jednu obrovskou výhodu - barevně zvýrazňují klíčová slova, což vede k přehlednosti celého kódu. Tyto tři soubory tvoří kostru každé šablony. Dále se v kořenovém adresáři mohou nacházet javascripty, obrázky, náhledy, další upřesňující CSS styly atd. K těmto souborům se vrátíme před dokončením šablony. [1]

3.3 INDEX.PHP

Soubor *index.php* obsahuje veškeré funkce pro prezentaci dat. Je složen z kombinace kódu HTML a PHP. Kód HTML zajišťuje obecnou strukturu a kód PHP zodpovídá za dynamickou funkčnost obsahu. Pokud chceme například přidat další moduly a rozšířit tak funkčnost šablony, přidáváme je právě do tohoto souboru. [29]

Do souboru *index.php* vložíme následující kód:

```

1 <?php
2 defined('_JEXEC') or die ('chyba');
3 ?>
4 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
5 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
6 <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="<?php echo
7 $this->language; ?>" lang="<?php echo $this->language; ?>"
8 dir="<?php echo $this->direction; ?>" >
9 <head>
10 <jdoc:include type="head" />
11 <link rel="stylesheet" href="<?php echo $this->baseurl ?>/templates/
12 <?php echo $this->template ?>/template.css" type="text/css" />
13 </head>

```

3.3.1 ZAKÁZÁNÍ PŘÍMÉHO PŘÍSTUPU

Každá šablona by měla začínat PHP kódem `defined('_JEXEC')` `or die`;, který zakáže přímý přístup k souboru *index.php*. Používá se z hlediska bezpečnosti a zajišťuje, že se k souboru *index.php* může přistupovat pouze z prostředí Joomla!.

3.3.2 DOCTYPE

Na 4. a 5. řádku definujeme DOCTYPE. Jedná se o deklaraci standardů, které dávají prohlížeči informace o použitých elementech.

3.3.3 <HTML>

Následuje značka `<html>`, ve které definujeme jazyk dokumentu. Parametr `xmlns` přímo odkazuje na používaný značkovací jazyk, výraz `xml:lang` slouží pro definování jazyka používaného v dokumentu. Parametr `dir` (z anglického slova *direction*) udává směr čtení textu. Joomla! dokáže text zobrazovat i pro čtení zprava doleva, což je typické pro arabské země. Ruční zápis pro jednojazyčný web by mohl vypadat například takto: `xml:lang="cs" lang="cs"` (pro web v češtině). [29]

3.3.4 <JDOC:INCLUDE ... >

Následuje tagjdoc (JDocument), který umožňuje přístup k frameworku systému Joomla!. Jedná se o jakési rozhraní sloužící k prezentaci dat. Přidáním značky include můžeme přidat moduly, systémové zprávy, vlastní obsah anebo například hlavičku. [1]

3.3.5 ZDROJOVÝ KÓD HLAVIČKY

Hlavička obsahuje titulek, kódování, popis stránky atd. Všechny údaje jsou automaticky generovány. Při zobrazení zdrojového kódu šablony, najdeme například tyto údaje:

Znaková sada

```
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
```

Tímto příkazem je definována znaková sada použitá v dokumentu. Určení znakové sady je pro internetové prohlížeče důležité. Aby dokázal prohlížeč správně zobrazit znaky, musí být určeno, jakou znakovou sadu má použít.

Dříve mohly znakové sady obsahovat pouze 256 znaků. Proto muselo existovat mnoho znakových sad, aby se pokryly všechny světové jazyky. Nyní je tomu jinak. UTF-8 je univerzální znaková sada používaná systémem Joomla!. Velkou předností je interpretace až 1 114 112 znaků, což zcela pokryje znaky všech světových jazyků. [29]

Údaje ve značce meta:

```
<meta name="robots" content="index,follow">
```

Klíčové slovo index povoluje vyhledávacím robotům indexovat stránky. Klíčové slovo follow povoluje sledování odkazů.

Hodnota index,follow je výchozí, proto se na stránkách nemusí vyskytovat. Vyhledávací roboti se budou chovat, jako by na tento zápis narazili. Použití tohoto zápisu nastává v případě, že chceme indexování a sledování odkazů zakázat: noindex,nofollow. Hodnoty je možné libovolně kombinovat.

```
<meta name="keywords"content="Joomla!, template">
```

V tomto příkazu jsou definována klíčová slova použitá na stránce. Zde uvedená klíčová slova musí odpovídat obsahu stránky, jinak jim vyhledávače nepřisuzují žádný

význam, ba naopak, staví se k nim negativně. To z důvodu tzv. *KeywordSpammingu*. Jak již název vypovídá, jedná se o použití velkého množství klíčových slov za účelem zvýšení návštěvnosti webu. Vyhledávací roboti jsou již natolik „chytří“, že dokážou tento neetický způsob odhalit a webu hrozí penalizace.

```
<meta name="description" content="Popis stránky">
```

V tomto zápisu je definován popis obsahu internetových stránek. Popis by měl být co nejvýstižnější, obsahovat příslušná klíčová slova a nepřesáhnout dvě až čtyři věty. Věty tvořící popis mají za úkol vzbudit v návštěvníkovi zvědavost.

Titulek

```
<title>Titulek stránky</title>
```

Titulek dokumentu je jednou z nejdůležitějších částí hlavičky. Titulek je zobrazen v titulkovém pruhu prohlížeče a jeho úkolem je informovat uživatele, kde se právě nachází. Klíčová slova obsažená v titulku mají velkou váhu na hledanost webu.

Informační kanál RSS

```
<linkhref="/joomla3.2/index.php?format=feed& type=rss" rel="alternate" type="application/rss+xml" title="RSS 2.0" />
```

Pomocí informačního kanálu RSS (ReallySimpleSyndication) je v Joomla! automaticky tvořen obsah ve formátu RSS založený na XML. Informační kanály slouží k jednoduchému získání požadovaných informací – pomocí RSS čteček. Nástupcem informačního kanálu RSS je Atom. Integrace Atomu je založena na stejném příkazu, odlišností je pouze uvedení slova „Atom 1.0“ namísto „RSS 2.0“. [29]

<link>

Následující zápis začínající tagem<link> určuje cestu k CSS souboru. Cestu bychom mohli samozřejmě zapsat i ručně (například takto: <link rel="stylesheet" href="template.css" type="text/css" />), ale uvedený způsob pomocí PHP kódu je vhodný zejména tehdy, pokud bychom web přesouvali (například z lokálního počítače na hosting).

- \$this->baseurl – značí adresu domény.

- `$this->template` – určuje název použité šablony.

Cestu k CSS souboru bychom museli po změně domény ručně přepsat, takto za nás vše obstará Joomla! a o změnu cesty se nemusíme starat. Obdobným způsobem bychom integrovali i javascripty. `$this` je globální objekt, jehož pomocí přistupujeme k parametrům šablony.

<body>

Pro začátek si web rozvrhneme na čtyři základní části: hlavičku (header), menu (menu), obsah (content) a patičku (footer). Pro snadné umístění stránky na střed „zabalíme“ tyto části do divu s názvem „wrapper“. Vložení následující kódu zajistíme základní rozvržení:

```
16 <body>
17
18 <div id="wrapper">
19
20 <div id="header">
21 </div>
22 .
23 <div id="menu">
24 </div>
25
26 <div id="content">
27 </div>
28
29 <div id="footer">
30 </div>
31 .
32 </div>
33 </body>
```

3.4 TEMPLATE.CSS

Abychom mohli jednotlivé divy správně „poskládat“, je potřeba jim nastavit barevné pozadí (jinou možností je nastavení barevných rámečků), bez kterého by se zobrazovaly jako prázdná místa na stránce. K tomu poslouží již zmiňovaný CCS styl *template.css*. Následující kód nastaví divům odlišná barevná pozadí, rozměry a jejich umístění.

```

#wrapper{
margin:0 auto;
width:1000px;
min-height:900px;
}
#header{
background-color:#800000;
width:1000px;
height:170px;
}
#menu{
background-color:#0000FF;
position:relative;
float:left;
width:200px;
min-height:700px;
}

#content{
background-color:#FFD700;
position:relative;
float:left;
width:800px;
min-height:700px;
}
#footer{
background-color:#008000;
position:relative;
float:left;
width:1000px;
min-height:30px;
}

```

3.4.1 WRAPPER

Jako první nastavíme vlastnosti selektoru „wrapper“, který v našem případě představuje obal ostatních divů. Zápis `margin: 0 auto;` zajišťuje vycentrování stránky na střed monitoru. Pomocí vlastnosti `width` nastavíme požadovanou šířku divu, v našem případě zvolíme 1000 px. Jelikož se v posledních letech používají zejména monitory s vysokým rozlišením (např. 1366×768 px, 1600×1200 px, 1920×1080 px), mohli bychom nastavit šířku větší (např. 1200 px) a při zobrazení by nebyl žádný problém v podobě posuvníků, které zhoršují celkový estetický vzhled stránky. Ovšem i v dnešní době se najdou uživatelé využívající rozlišení 1024x768 px, u kterých by se stránka o šířce 1200 px zobrazovala s posuvníky. Další možností by bylo určení šířky pomocí procent (kolik procent obrazovky má stránka zaujímat), ale při vyšších rozlišeních by mohlo dojít k přílišnému roztažení stránky a zhoršení celkového vzhledu. Zápisem `min-height: 900 px;` nastavíme minimální výšku divu. Je důležité určit minimální výšku, protože div bude dynamicky měnit svou výšku dle zobrazovaného obsahu. Zápis `height: 900 px;` je nevhodný.

3.4.2 HEADER

Následuje nastavení vlastností selektoru „header“ představující hlavičku. Zápis `background-color: #800000;` určí barvu pozadí divu, v tomto případě jsme zvolili hexadecimální hodnotu #800000, což představuje hnědočervenou barvu. Dále určíme šířku (`width: 1000 px;`) a výšku (`height: 170 px;`). Výška hlavičky zůstane neměnná,

proto můžeme zvolit vlastnost `height`, nikoli `min-height`, jako tomu bylo u divu „wrapper“.

3.4.3 MENU, CONTENT

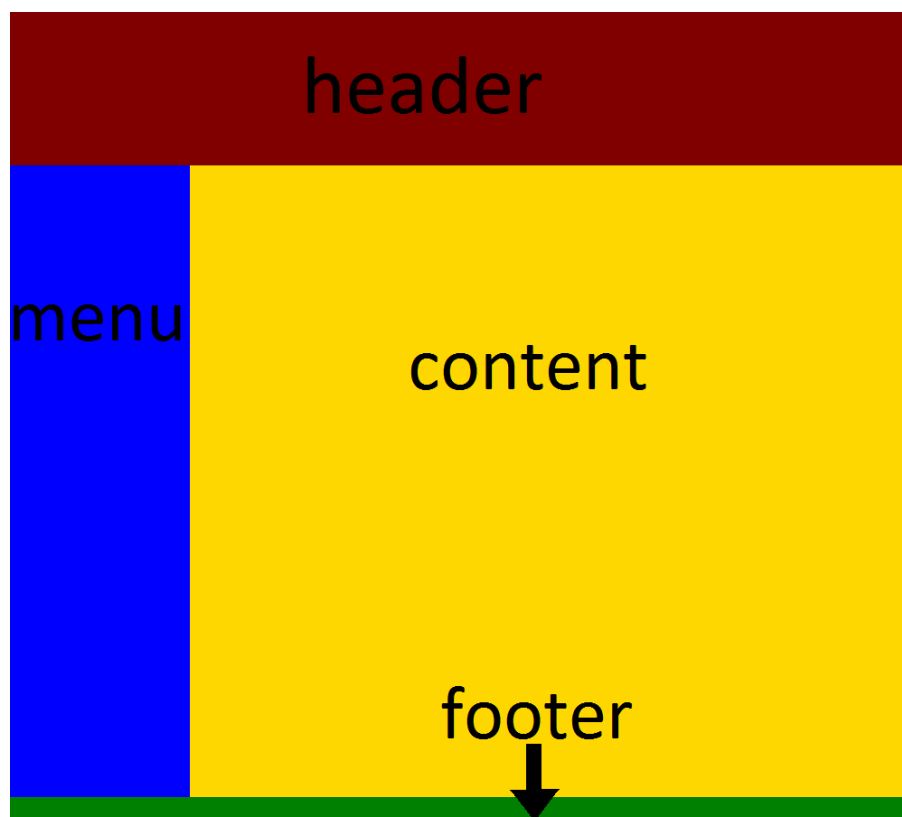
Selektory „menu“ a „content“ mají shodný zápis lišící se pouze v parametrech vlastností. Určujeme barvu pozadí, šířku a minimální výšku (tyto divy budou svou výšku dynamicky měnit) jako u selektoru „wrapper“. Navíc zde určujeme pozici divu. Zápis `position: relative; a float: left;` nastavuje relativní pozicování s obtékáním zleva.

3.4.4 FOOTER

Posledním selektorem je patička „footer“, u které určujeme totožné vlastnosti jako u předešlých selektorů. Opět můžeme nastavit pevnou šířku (`height: 30 px;`), protože patička svou výšku měnit nebude.

3.4.5 AKTUÁLNÍ STAV ŠABLONY

V tomto stádiu by naše šablona mohla vypadat jako na obrázku 7. Kontrolu správného „poskládání“ umožňují divy různých barev.



Obrázek 7 - současný stav šablony

3.5 TEMPLATEDETAILS.XML

Soubor *templateDetails.xml* obsahuje kompletní metadata šablony. Ve formátu XML jsou uvedeny informace o autorovi, název šablony, licence, obsažené soubory, umístění modulů, parametry šablony atd. Tento soubor musí bez výjimky obsahovat každá šablona, jinak by nemohla být nainstalována. [29]

Jazyk XML je podobně jako HTML také značkový jazyk, ovšem oproti HTML se liší tím, že jednotlivé elementy mohou mít jakýkoli název, což je v HTML pevně určeno. Jazyk XML je méně tolerantní vůči chybám. Každý otevřený element musí být uzavřen, jinak se bude zobrazovat chybové hlášení. [29]

Podobně jako předcházející soubory, vytvoříme i soubor *templateDetails.xml*, který by mohl obsahovat následující kód:

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <!DOCTYPE install PUBLIC "-//Joomla! 3.2//DTD template 1.0//EN"
3 "http://www.joomla.org/xml/dtd/3.2/template-install.dtd">
4 <extension version="3.2" type="template" client="site">
5
6 </extension>

```

3.5.1 ZNAKOVÁ SADA

Na 1. řádku uvedeme použitou verzi jazyka XML a znakovou sadu. Zvolíme znakovou sadu UTF-8, která plně podporuje české znaky.

3.5.2 DOCTYPE

Na 2. a 3. řádku definujeme interní typ dokumentu obsahující příkazy, které určují strukturu souboru.

3.5.3 ROZŠÍŘENÍ

Na 4. řádku definujeme, o jaký typ rozšíření se jedná. Určíme, že se jedná o rozšíření pro Joomla! verze 3.2, rozšíření je typu šablona a parametrem klient říkáme, že se jedná o šablonu pro uživatelskou část.

Mezi značky `<extension>` a `</extension>` vložíme následující kód:

```
7 <name>moje_sablona</name>
8 <creationDate>15.2.2014</creationDate>
9 <author>Michael Vaník</author>
10 <authorEmail>mvanik@students.zcu.cz</authorEmail>
11 <license>GNU General Public License</license>
12 <version>1.0</version>
13 <description>Ukázková šablona sloužící pro studijní účely
14 </description>
15
16 <files>
17 <folder>language</folder>
18 <filename>index.php</filename>
19 <filename>template.css</filename>
20 <filename>templateDetails.xml</filename>
21 </files>
22
23
24 <positions>
25 <position>nazev_pozice</position>
26 </positions>
27
28 <languages folder="language">
29 <language tag="cs-CS">cs-CS/cs-CS.moje_sablona.ini</language>
30 <language tag="cs-CS">cs-CS/cs-CS.moje_sablona.sys.ini</language>
31 </languages>
```

- name – tento řádek určuje název šablony. Zadaný název slouží k vytvoření stejnojmenného adresáře v adresáři šablon. Název by neměl obsahovat diakritiku a speciální znaky. My si již na začátku v adresáři templates vytvořili adresář moje_sablona, tudíž název šablony určený zde, je moje_sablona.
- creationDate – datum, kdy byla šablona vytvořena. Datum může být zadáno v libovolném formátu.
- author – jméno autora šablony.
- authorEmail – e-mailová adresa autora šablony. Může sloužit k tomu, aby se potencionální uživatelé šablony mohli spojit s autorem.
- licence – licence, pod kterou je šablona šířena.
- version – verze šablony. Kdybychom prováděli nějaká změny, je možné tento údaj upravit.
- description – popis šablony.
- files – seznam souborů použitých v šabloně.

- folder – výčet složek nacházejících se v kořenovém adresáři.
- filename – seznam jednotlivých HTML, PHP, XML, CSS, obrázků a případně dalších souborů.
- positions – seznam umístění modulů používaných v šabloně.
- position – názvy jednotlivých umístění. Tyto názvy se zobrazí ve Správci modulů v administrátorské části webu.
- languages – mezi tyto značky se umísťují používané jazykové soubory.
- language – cesta k jazykovým souborům. [35]

3.6 NÁVRH GRAFICKÉHO LAYOUTU

3.6.1 VOLBA BAREVNÉHO SCHÉMATU

Ideální sladění barev je pro web velmi důležité, neboť barvy jsou to první, co návštěvník uvidí. Základní podmínkou je dostatečný kontrast mezi barvou pozadí stránky a barvou textu. Při nedostatečném kontrastu je text špatně čitelný. Pozadí stránky by nemělo mít příliš syté barvy. Jejich použití strhává čtenářovo pozornost na pozadí a zhoršuje soustředěnost na text. Ideální kombinací jsou pastelové barvy spolu s barvou tmavou. [31]

Pro usnadnění vhodného výběru barev existuje mnoho programů. Příkladem mohou být ColorWizard (www.colorsontheweb.com), Adobe Kuler (www.kuler.adobe.com) nebo ColorSchemer (www.colorschemer.com).

3.6.2 VYTVOŘENÍ GRAFIKY

Pro vytvoření grafického vzhledu webu lze použít libovolný grafický editor.

Pro své jednodušší ovládání, plnou podporu češtiny a zejména bezplatné šíření pod licencí GNU/GPL lze doporučit oblíbený program Gimp (www.gimp.org).

Dalším oblíbeným, ale ovšem placeným nástrojem je Adobe Photoshop (www.adobe.com/cz/products/photoshop.html). Jedná se o profesionální bitmapový grafický editor disponující všemi funkcemi, které můžete při tvorbě webové grafiky

využít. Novinkou je jeho implementace do CreativeCloudu – uživatel neplatí za software, ale za službu, která mu je on-line poskytnuta.

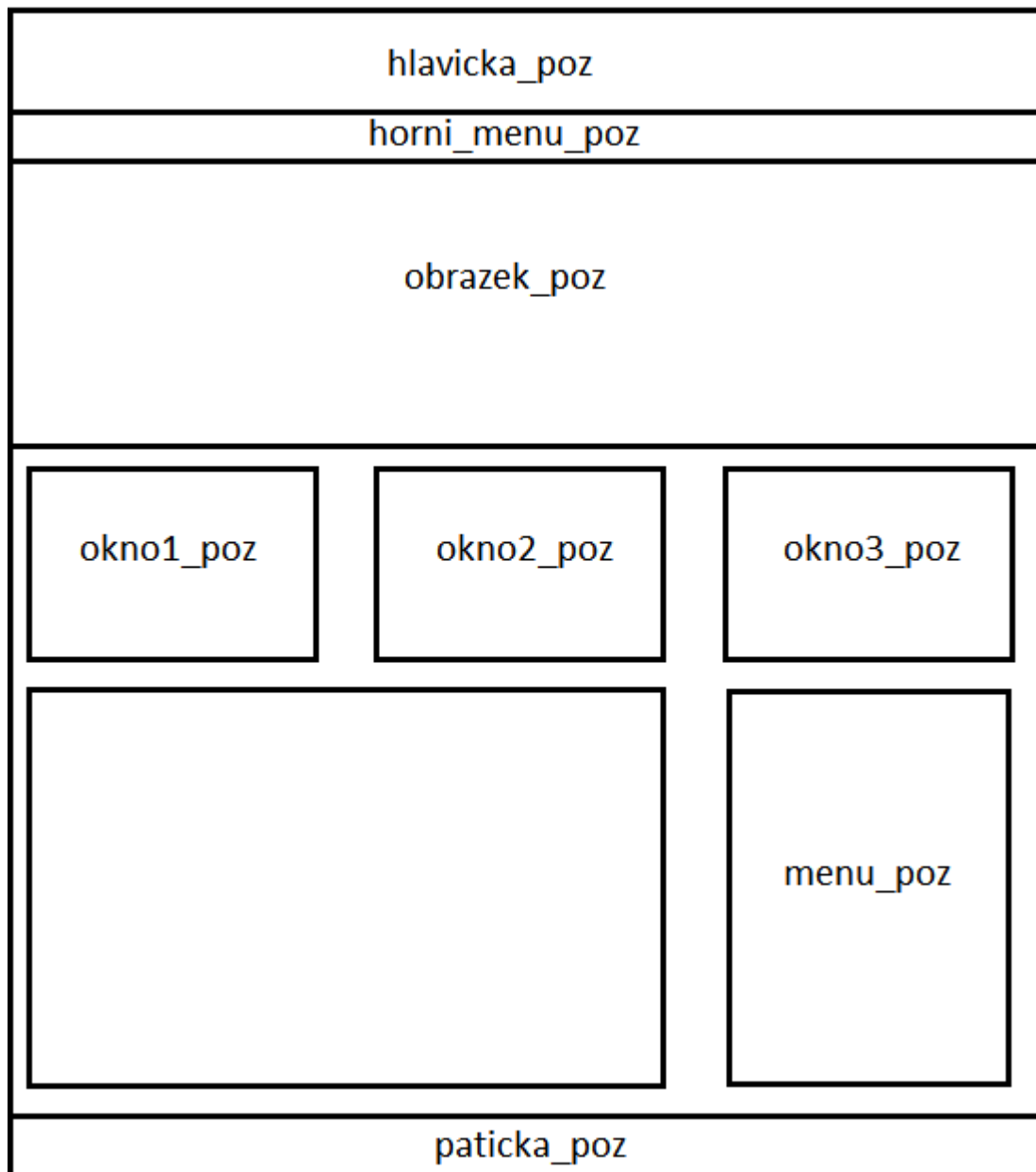
Pro tvorbu vektorové grafiky jsou oblíbenými nástroji Inkspace (www.inkscape.org) a Corel Draw (www.corel.com).

Je na každém uživateli, který grafický program si zvolí. Ohled bychom měli brát zejména na to, jak náročné funkce chceme využít.

Revoluci ve tvorbě webové grafiky přinesla třetí verze kaskádových stylů – CSS3. Díky CSS3 je možné například vytvoření zaoblených rohů, kdy pouze zadáme, který roh stylujeme a jaký má být rádius zaoblení. Další zajímavou funkcí jsou stíny objektů, stíny textu, lineární barevné přechody, natočení objektu o libovolný úhel, postupná změna barvy objektu po najetí myši a mnoho dalšího. Využití formátování pomocí CSS má velkou výhodu – při zobrazení stránky není potřeba načítat grafické soubory, což umožňuje rychlejší zobrazení webu. CSS3 využijeme při tvorbě naší grafiky.

3.6.3 NASTYLOVÁNÍ POMOCÍ CSS3

Před samotným „nastylováním“ jednotlivých objektů šablony je nutné si rozmyslet jejich rozmístění a pozice. Základní strukturu uvedenou v kapitole 3.5 upravíme na požadovanou podobu. Názvy pozic pro umístění modulů může každý uživatel zvolit dle svého uvážení, možná ukázka je na obrázku 8.

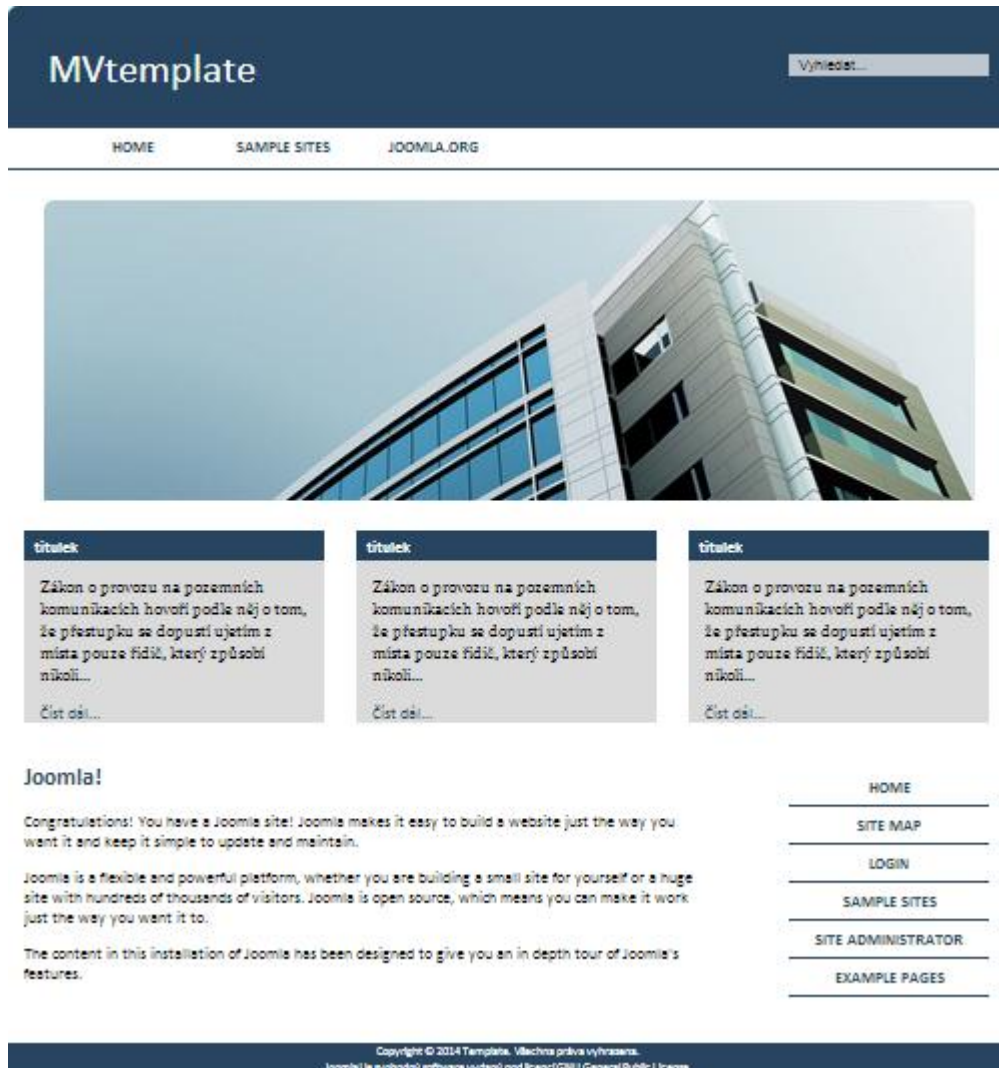


Obrázek 8 - rozvržení pozic pro umístění modulů

Pro většinu vlastností v CSS3 je možné využít on-line generátory, které značně urychlí a zjednoduší práci. Příklady CSS3 generátorů:

- www.css3generator.com
- www.css3gen.com
- www.generatecss.com
- www.createcss3.com

Po zobrazení obsahu a „nastylování“ pomocí CSS3 může naše šablona vypadat jako na obrázku 9. Jak zobrazit obsah si popíšeme v následující podkapitole.



Obrázek 9 - možný vzhled po „nastylování“

Jednotlivé kroky ve stylování si popisovat nebudeme, je na každém, jaký vzhled si zvolí. Pouze si řekneme, čemu je potřeba věnovat pozornost při použití CSS3, aby bylo zaručeno správné zobrazení ve všech prohlížečích. Pro zjištění názvu prvku můžeme využít aplikaci FireBug. Další možností je zobrazení zdrojového kódu šablony přímo v prohlížeči Google Chrome nebo Mozilla Firefox. Na požadovaný objekt klikneme pravým tlačítkem – zvolíme volbu Zkontrolovat prvek či Prozkoumat prvek. Ve spodní části obrazovky se otevře okno, kde vidíme kompletní HTML kód i zápis v CSS stylech. Zde je možná přímá úprava objektů. Po znovunačtení stránky se zobrazí web s původními vlastnostmi, provedená úprava se nikam neukládá, je možné si tak „nanečisto“ vyzkoušet úpravu vzhledu objektu.

Pro jednodušší orientaci v HTML kódu můžeme modulům přiřadit vlastní příponu třídy modulu. Ve Správci modulů vybereme požadovaný modul a klikneme na tlačítko Upravit. Zobrazí se nám základní informace o modulu, klikneme na Pokročilé -> Přípona třídy modulu, napíšeme požadovaný název třídy a klikneme na „Uložit“. Stylování můžeme provést s námi zadanou příponou třídy. Přípony tříd se hodí, pokud například v celém dokumentu definujeme vlastnosti nadpisu úrovně H3, ale pro jeden konkrétní modul chceme využít jiné stylování.

Některé moduly, například nabídky, umožňují nejen vytvoření vlastní přípony třídy modulu, ale i vlastní CSS třídu či CSS identifikátor. Uvedeme si příklad na nabídce Main Menu.

- class="nav menu" – název třídy bez jakéhokoli vlastní úpravy.
- class="nav menu_moje_trida" – název třídy po zadání „_moje_trida“ do pole CSS třída nabídky.
- id="muj_identifikátor" – název identifikátoru po zadání „muj_identifikator“ do pole CSS identifikátor nabídky.

Prefixy

Pro správné zobrazení objektů „nastylovaných“ pomocí CSS3 ve všech prohlížečích je nutno dodržet určitá pravidla zápisu. Každý prohlížeč má vlastní vykreslovací jádro, proto musíme standardní zápis doplnit o tzv. prefixy (-o, -ms, -moz, -webkit). Na posledním místě je vždy uveden neprefixový – standardní zápis, který „přebíjí“ předchozí zápisy. Tento zápis budou po určité standardizaci podporovat všechny prohlížeče. Ukážeme si příklad – postupná změna barvy pozadí odkazu po najetí myši:

```
-o-transition:color .2s ease-out, background .5s ease-in;
-ms-transition:color .2s ease-out, background .5s ease-in;
-moz-transition:color .2s ease-out, background .5s ease-in;
-webkit-transition:color .2s ease-out, background .5s ease-in;
transition:color .2s ease-out, background .5 ease-in;
```

- -o – tento prefix se používal u dnes již neexistujícího vykreslovacího jádra Opery.
- -ms – prefix pro prohlížeče Internet Explorer.

- -moz – prefix používaný pro prohlížeče Firefox.
- -webkit – prefix pro prohlížeče Safari, Android Browser, Google Chrome a Opera (od verze 15). [34]

3.7 ZOBRAZENÍ OBSAHU

3.7.1 MODULES, COMPONENT

Aby nám šablona zobrazovala nějaká data, musíme do těla dokumentu (mezi tagy `<body></body>`) vložit moduly zajišťující prezentaci dat. Moduly se do souboru *index.php* připojují pomocí metody `<jdoc:include type="modules" name="nazev_pozice" style="xhtml" />`.

Výše uvedenou metodou se nepřipojují pouze moduly (modules), ale i hlavička (head) či systémové zprávy (message). Co konkrétně se má připojit, určuje parametr `type`.

Parametr `name` určuje pozici v rámci šablony. Šablony nainstalované v základní verzi systému Joomla! používají nejednoznačné názvy pozic, které o skutečné umístění nijak nevyovídají (position-0, positon-1 apod.). Mnohem vhodnější je použít výstižnější názvy jako například `menu_poz`, `hlavicka_poz`, `obrazek_poz`. Názvy pozic doplníme do souboru *templateDetails.xml*.

```

23 <positions>
24   <position>hlavicka_poz</position>
25   <position>horni_menu_poz</position>
26   <position>obrazek_poz</position>
27   <position>okno1_poz</position>
28   <position>okno2_poz</position>
29   <position>okno3_poz</position>
30   <position>menu_poz</position>
31   <position>paticka_poz</position>
32 </positions>

```

Parametr `style` určuje, v jaké podobě se modul zobrazí. Uvedeme si příklady hodnot tohoto parametru:

- `style="xhtml"` – zobrazí kromě obsahu modulu i jeho titulek.
- `style="none"` – zobrazí pouze obsah modulu.

Pokud bychom chtěli formátování modulů uživatelsky upravit, je možné vytvořit vlastní styl.

Pro zobrazení samotného obsahu slouží podobná metoda jako pro zobrazení modulů: `<jdoc:include type="component" />`. Parametr `type="component"` říká, že se jedná o komponentu zobrazující obsah – nejčastěji článek. Tato metoda smí být použita pouze jednou, proto není nutný parametr `name`, jako tomu bylo u modulů. Obsah se zobrazí v divu, do kterého tuto metodu vložíme.

3.7.2 SYSTÉMOVÁ HLÁŠENÍ

Systémová hlášení mají jednoduchý úkol – poskytnou uživateli varování či informace o chybě. Nejčastějším systémovým hlášením je upozornění na chybně zadané heslo při přihlášení, nebo nevyplnění všech povinných údajů v registračním formuláři.

Systém Joomla! obsahuje tři typy systémových hlášení:

- Chyba – například chybné údaje při přihlášení.
- Varování – například při nevyplnění povinných údajů ve formuláři.
- Zpráva – například pokud se nepodařilo odeslat mail. [29]

Systémová hlášení se do souboru `index.php` připojují příkazem:

```
<jdoc:include type="message" />
```

Výše uvedený příkaz lze umístit kamkoli do souboru `index.php`. „Nastylování“ hlášení se provádí klasickým způsobem jako u všech ostatních objektů. Texty chybových hlášení jsou uloženy v jazykovém souboru příslušné komponenty, kde je možná jejich modifikace dle přání uživatele.

3.7.3 DYNAMICKÁ ZMĚNA ROZVRŽENÍ

Abychom zajistili dynamickou úpravu struktury, vytvoříme na začátku dokumentu `index.php` proměnou `$info_menu`, do které se uloží počet modulů, přiřazených na pozici `menu_poz`. `$this` je globální objekt obsahující množství dat týkajících se šablony (titulek, název, použitý jazyk a další parametry). Počet modulů zjistíme pomocí

funkce `countModules`. Jako parametr do závorky uvedeme, na jaké pozici chceme zjišťovat počet modulů. Celý zápis má následující podobu:

```
$info_menu=$this->countModules('menu_poz');
```

Pokud bychom chtěli zjišťovat počet přiřazených modulů i na dalších pozicích, nic nám nebrání tomu, vytvořit další proměnou, do které by se tyto údaje ukládaly.

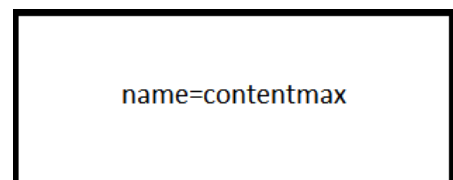
Pro názornost si vystačíme s výše uvedenou proměnou `$info_menu`, do které se uloží počet modulů, přiřazených pozici `menu_poz`. Pokud je pozici `menu_poz` přiřazen nějaký modul (`$info_menu >= 1`), nastaví se divu určenému pro obsah název `content`. Pokud není pozici `menu_poz` přiřazen žádný modul, ponese div určený pro obsah název `contentmax` a jeho šířku nastavíme tak, aby obsah vyplnil šířku celé stránky. Oba divy libovolně nastylujeme pomocí CSS. Podmínka zajišťující dynamické rozvržení může vypadat takto:

```
68 <?php if ($info_menu >= 1){ ?>
69 <div id="content">
70   <jdoc:include type="component"/>
71 </div>
72 <?php } else{
73 ?>
74
75 <div id="contentmax">
76   <jdoc:include type="component"/>
77 </div>
78 <?php } ?>
```

Pro názornost si uvedeme (obrázek 10 a 11), jak bude v obou případech vypadat struktura rozvržení.



Obrázek 11 - pozici "menu_poz" je přiřazen modul



Obrázek 10 - pozici "menu_poz" není přiřazen žádný modul

3.8 DYNAMICKÉ PARAMETRY

Parametry šablony zvyšují flexibilitu a zjednodušují dynamickou úpravu dle požadavků uživatele. Nejvýstižnějším příkladem může být možnost změny barevného provedení celé šablony. Uživatel může pomocí parametru volit mezi barevnými schémata a rozhodnout se pro to, které mu nejvíce vyhovuje. Ruční „nastylování“ barev by bylo zdlouhavé a nepraktické. Dynamické parametry umožňují jedním kliknutím načíst potřebný CSS styl a tím během okamžiku změnit barevné schéma celé šablony. Dalším příkladem může být změna loga, nastavení šířky stránky, úprava názvu, titulku, popisku stránky a mnoho dalšího.

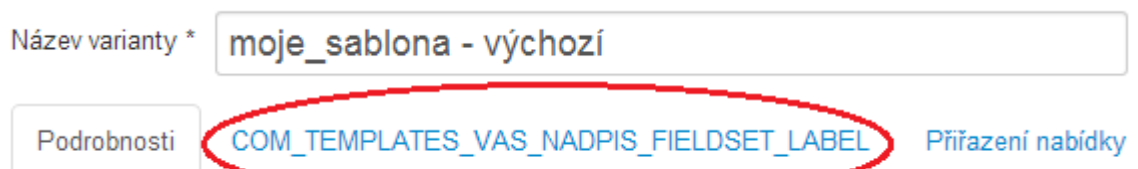
Jádro systému Joomla! je řízeno souborem *templateDetails.xml*. Potřebné příkazy se nacházejí uvnitř elementu config. Následující příkazy se uvedou zápisem `<fieldname="params">`. V tomto případě slouží parametr name k informování systému o použití polí. Následuje zápis `<fieldsetname="VAS_NADPIS">`. Element fieldset seskupuje příkazy, které uvidíme v administrátorské části webu pod možností Pokročilé nastavení. Řetězec uvedený v parametru name si můžeme sami zvolit. Je ovšem nutné zapsat jeho hodnotu velkými písmeny, bez speciálních znaků a bez diakritiky. [29]

```

36 <config>
37 <fields name="params">
38 <fieldset name="VAS_NADPIS">
39
40 </fieldset>
41 </fields>
42 </config>

```

Pokud bychom se nyní podívali do Správce šablon -> Varianty šablon a vybrali upravovanou šablonu, uvidíme to co je vidět na obrázku 12.



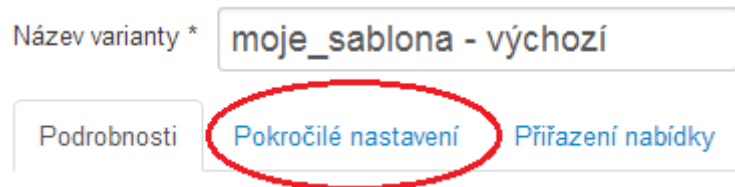
Obrázek 12 - pokročilé nastavení před „překladem“

Námi zadaný řetězec VAS_NADPIS se integroval do předdefinovaného konstrukturu. Aby se řetězec zobrazil ve tvaru, který požadujeme (například výše

uvedený Pokročilé nastavení), musíme do používaného jazykového souboru (*cs-CZ/cs-CZ.tpl_moje_sablona.ini*) vložit řetězec:

```
33 COM_TEMPLATES_VAS_NADPIS_FIELDSET_LABEL="Pokročilé nastavení"
```

Tento zápis zajistí „překlad“ řetězce VAS_NADPIS na řetězec Pokročilé nastavení (viz obrázek 13).



Obrázek 13 - pokročilé nastavení po "překladu"

Pro možnost změny parametrů šablony je nutné do souboru *templateDetails.xml* vložit formulářové prvky, ze kterých se budou získávat uživatelem zadané hodnoty. Mezi nepoužívanější formulářové prvky patří:

- type="text" – textové pole.
- type="list" – zaškrťovací políčka (výběr několika možností).
- type="radio" – přepínač (výběr pouze jedné možnosti).
- type="media" – výběr pomocí Správce médií (například výběr obrázku).[29]

3.8.1 ÚPRAVA TEXTU ZOBRAZOVANÉHO V HLAVIČCE

Mezi elementy `<fieldset></fieldset>` vložíme tento kód:

```
40 <field name="nazev_spolecnosti" type="text" default="defaultní text"
41 label="TPL_MOJE_SABLONA_FIELD_NAZEV_SPOL_LABEL"
42 description="TPL_MOJE_SABLONA_FIELD_NAZEV_SPOLECNOSTI_DESC"
43 filter="string" />
```

- name – název prvku.
- type – typ prvku.
- default – defaultně nastavená hodnota.
- label – popisek prvku.
- description – nápovědný text zobrazený při najetí myši.

- filter – filtrování zadaného řetězce dle požadavků uživatele. Hodnoty:
 - string – převod řetězce na text.
 - integer – převod zadaných údajů na celé číslo (při zadání pouze písmen bude hodnota 0, při zadání kombinace písmen i čísel se písmena odstraní).
 - alnum – povoleny jsou písmena i čísla.
 - word – odstranění mezer mezi znaky – vznikne jedno slovo. [29]

Aby se popisek (label) i nápovědný text (description) zobrazily jako požadovaný řetězec, vložíme do jazykového souboru:

```
31 TPL_MOJE_SABLONA_FIELD_NAZEV_SPOL_LABEL="Název společnosti"
32 TPL_MOJE_SABLONA_FIELD_NAZEV_SPOL_DESC="Napište název společnosti"
```

K zobrazení zadaného řetězce v šabloně použijeme funkci `get()` a jako parametr uvedeme název formulářového prvku. Zápis vložíme na požadované místo do souboru `index.php`.

```
26 <?php echo $this->params->get('nazev_spol'); ?>
```

Zobrazený text lze libovolně naformátovat.

3.8.3 ZMĚNA OBRÁZKU POD HORIZONTÁLNÍM MENU

Postup je podobný jako u předchozího příkladu (změna textu zobrazovaného v hlavičce) s tou změnou, že neuvedeme `type="text"`, ale `type="media"`. Do souboru `templateDetails.xml` vložíme následující kód:

```
48 <field name="headerImage" type="media"
49 label="TPL_MOJE_SABLONA_FIELD_HEADER_IMAGE_LABEL"
50 description="TPL_MOJE_SABLONA_FIELD_HEADER_IMAGE_DESC" />
```

V jazykovém souboru určíme, jaký text se bude zobrazovat jako popisek (label) a nápovědný text (description):

```
31 TPL_MOJE_SABLONA_FIELD_NAZEV_SPOL_LABEL="Obrázek"
32 TPL_MOJE_SABLONA_FIELD_NAZEV_SPOL_DESC="Vyberte obrázek"
```

Do souboru `index.php` vložíme cestu k obrázku, opět použijeme funkci `get()`:

```
48 
```

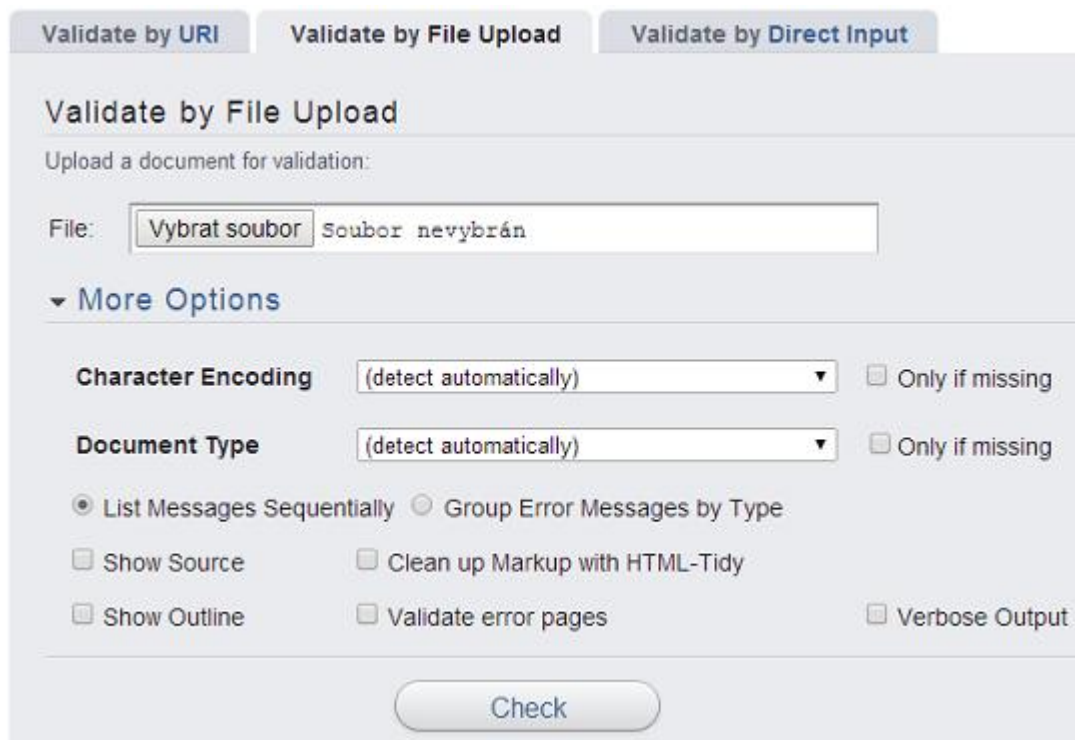

4 PRAKTICKÉ POUŽITÍ VYTVOŘENÉ ŠABLONY

4.1 VALIDACE

Validita webu znamená jeho soulad s webovými standardy – neobsahuje syntaktické chyby dle zvoleného jazyka.

Nevalidní kód může způsobit mnoho problémů. Stránky se můžou špatně zobrazovat, například některá část může překrývat jinou. Další problém může nastat při použití hlasové čtečky, která se může v nevalidním kódu snadno ztratit a interpretovat tak nesprávný obsah. Další problém může nastat při indexaci webu (uložení do databáze vyhledávače) vyhledávacím robotem. Nejpoužívanější prohlížeče jsou schopny drobné chyby v kódu opravit a zobrazit stránky nevalidní správně. Méně používané prohlížeče nevalidní stránky v mnoha případech správně zobrazit nedokážou, proto je validní kód důležitý. [30]

Systému pro validaci existuje několik, jedním z neoblíbenějších je validátor vytvořený mezinárodním konsorciem W3C dostupný na adrese www.validator.w3.org (viz obrázek 14). Použití je zcela zdarma a pomůže rychle odhalit případné chyby.



The image shows the 'Validate by File Upload' section of the W3C Validator interface. At the top, there are three tabs: 'Validate by URI', 'Validate by File Upload' (which is selected), and 'Validate by Direct Input'. Below the tabs, the text 'Upload a document for validation:' is followed by a file selection field labeled 'File:' containing the text 'Vybrat soubor' and 'Soubor nevybrán'. A 'More Options' section is expanded, showing several settings: 'Character Encoding' set to '(detect automatically)' with a checkbox for 'Only if missing'; 'Document Type' set to '(detect automatically)' with a checkbox for 'Only if missing'; radio buttons for 'List Messages Sequentially' (selected) and 'Group Error Messages by Type'; checkboxes for 'Show Source', 'Clean up Markup with HTML-Tidy', 'Show Outline', 'Validate error pages', and 'Verbose Output'. A 'Check' button is located at the bottom of the form.

Obrázek 14 - validace

Nejdříve provedeme validaci HTML souboru, ve kterém je z pravidla větší pravděpodobnost výskytu chyby. Soubor můžeme zadat pomocí URL adresy stránky (Validate by URL), nahrát soubor uložený v počítači (Validate by FileUpload) nebo do validátoru přímo zkopírovat kód (Validate by Direct Input).

Character encoding – určuje kódování znaků dokumentu. Pokud je v hlavičce dokumentu kódování správně nastavené, je nejjednodušší možností ponechat výchozí hodnotu tohoto pole – detect automatically (automatická detekce).

Document type – uvádí DOCTYPE dokumentu. Pokud není DOCTYPE v dokumentu uveden, je nutné ho uvést zde, aby validátor věděl, dle jakého předpisu má dokument kontrolovat.

Po provedení nastavení klikneme na tlačítko „Check“ a systém nám oznámí výsledek validace.

Pro validování CSS stylu využijeme podobný on-line nástroj dostupný na adrese www.jigsaw.w3.org/css-validator. Zde je nejdůležitějším nastavením volba úrovně CSS (CSS1, CSS2 nebo CSS3). Po kliknutí na tlačítko „Check“ se opět dozvíme výsledek validace.

4.2 KONTROLA ZOBRAZENÍ V PROHLÍŽEČÍCH

Kontrola zobrazení v různých prohlížečích je důležitou činností, neboť každý prohlížeč provádí zpracování a rendering (vykreslení) obsahu odlišně.

Ruční kontrola stránek v desítkách prohlížečů je časově nereálná, proto můžeme využít nástroje umožňující hromadnou kontrolu takřka ve všech světových prohlížečích.

Jedním z nejpoužívanějších a spolehlivých nástrojů pro kontrolu zobrazení je Adobe BrowserLab. Jedná se o on-line aplikaci, která umožňuje porovnávat layout webové stránky vykreslené různými prohlížeči. Použití je velice jednoduché, stačí zadat URL adresu webu, který chceme zkontrolovat, zvolit požadované prohlížeče a aplikace vygeneruje náhledy v podobě obrázků. Dalším online nástrojem fungujícím na podobném principu je například Browser Shots (www.browsershots.org).

4.3 NÁHLEDOVÉ OBRÁZKY

Náhledové obrázky se zobrazují v administrativní části systému ve Správci šablon. Jak již bylo v kapitole 3.2 řečeno, soubory s náhledovými obrázky se nachází v kořenovém adresáři šablony. Jde o soubory *template_thumbnail.png* – malý náhled a *template_preview.png* – velký náhled. Rozměry obrázků nejsou pevně dány, obvyklé rozměry jsou 206 x 150 px pro malý náhled a 640 x 388 px pro velký náhled. Pro jednotné zobrazení by se měly dodržovat stejné rozměry pro všechny šablony. Oba tyto obrázky vytvoříme velice jednoduše: zobrazíme šablonu v prohlížeči, provedeme printscreen obrazovky, obrázek zkopírujeme do libovolného grafického editoru a upravíme na požadovanou velikost. Obrázky uložíme pod uvedenými názvy do kořenového adresáře šablony.

4.4 IKONA FAVICON

Obrázek představující ikonu vytvoříme v libovolném grafickém editoru. Rozměry jsou pevně dány – 16 x 16 px. Pro tento obrázek existuje speciální formát .ico, ale můžeme použít například i formáty .png či .gif. Cestu k tomuto souboru se nejčastěji určuje těmito zápisy:

- `<link rel="icon" type="image/vnd.microsoft.icon" href="favicon.ico" />`
- `<link rel="icon" type="image/png" href="favicon.png" />`
- `<link rel="icon" type="image/gif" href="favicon.gif" />`

Kdybychom ikonu nevytvořili, nebo zapsali chybnou cestu, zobrazí se defaultní ikona systému Joomla!.

4.5 VYTVOŘENÍ ARCHIVU ZIP

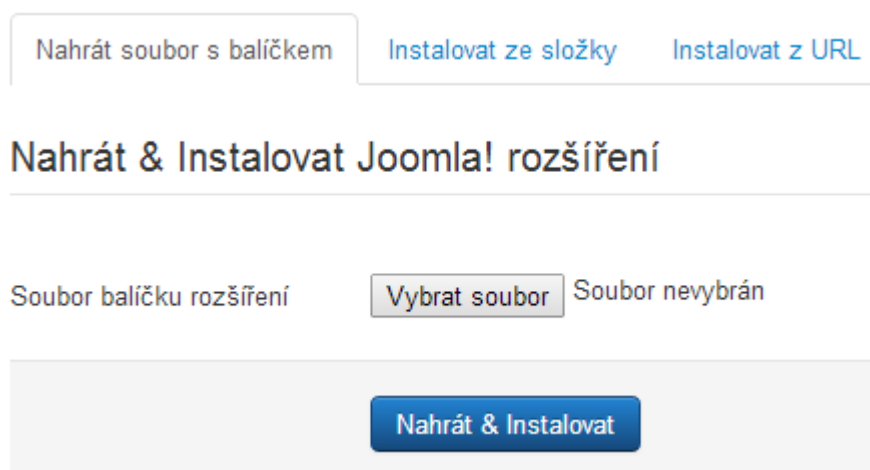
Šablony, stejně jako všechny rozšíření se instalují v podobě archivu ZIP. Po vytvoření náhledových obrázků, ikony a všech klíčových souborů se přesuneme do kořenového adresáře šablony a všechny soubory zkomprimujeme do archivu

ve formátu ZIP. Pro archivaci je možné použít libovolný archivační program, například TotalCommander.

4.6 INSTALACE

Šablony se v systému Joomla! spravují stejně jako ostatní typy rozšíření – přes Správce rozšíření. V administrátorské části klikneme na Rozšíření -> Správce rozšíření -> Instalace (viz obrázek 15). Máme tři možnosti, jak novou šablonu nainstalovat:

1. Nahrát soubor s balíčkem.
2. Instalovat ze složky.
3. Instalovat z URL.



Obrázek 15 - instalace šablony

Zvolíme možnost Nahrát soubor s balíčkem, klikneme na Vybrat soubor, vybereme již vytvořený archiv *moje_sablona.zip* a klikneme na tlačítko Nahrát & Instalovat. Systém Joomla! Archiv dekomprimuje a uloží potřebné parametry šablony do databáze. Pokud jsou všechny soubory správně pojmenované a obsahují potřebná data (zejména soubor *templateDetails.xml*), zobrazí se zpráva o úspěšné instalaci šablony. V opačném případě se zobrazí varovná zpráva o neúspěšné instalaci.

Pro nastavení nainstalované šablony jako výchozí, stačí kliknout na hvězdičku s popiskem „Nastavit jako výchozí“. Tímto způsobem lze jedním kliknutím měnit vzhled celého webu.

4.7 ODINSTALACE

Aby nepoužívaná šablona nezabírala zbytečně místo v paměti, je vhodné ji odinstalovat. Klikneme na Rozšíření -> Správce rozšíření -> Správa. Pomocí filtru nastavíme správu šablon, vybereme požadovanou šablonu a klikneme na tlačítko Odinstalovat. Nyní je šablona odebraná z databáze a zároveň z disku (serveru) smazány její soubory.

4.8 NÁHLED S POZICEMI MODULŮ

Náhled šablony zobrazující pozice modulů umožňuje rychle a jednoduše zjistit, na jakém místě se konkrétní modul v rámci šablony nachází. Zobrazení náhledu je v systému z důvodu bezpečnosti defaultně zakázáno. Ve Správci šablon u každé šablony vidíme poznámku „Náhled není k dispozici“. Povolení náhledu s pozicemi modulů nastavíme ve Správci šablon -> Možnosti -> Náhled pozic modulů a přepneme na „Povoleno“. „Náhled není k dispozici“ se změní na odkaz „Náhled“. Po kliknutí na tento odkaz se zobrazí náhled šablony s pozicemi jednotlivých modulů. Tato možnost je nejvíce užitečná v případě rozsáhlých internetových stránek s množstvím pozic

4.9 ÚPRAVA SOUBORŮ

System Joomla! nabízí možnost přímé úpravy souborů. Pro úpravu libovolného souboru se přesuneme do Správce šablon -> Šablony a zvolíme šablonu, kterou chceme upravovat. Zobrazí se všechny soubory, které šablona obsahuje a po kliknutí na konkrétní soubor se zobrazí jeho zdrojový kód. Přímá editace souborů přináší obrovskou výhodu – odpadá nutnost otevírání souborů v programech pro editaci kódu

a jejich nahrávání po každé úpravě znovu na server. Kromě editace souborů disponuje Joomla! i možnostmi správy složek.

4.10 OVERRIDES

Overrides, neboli přepsání, nabízí jednoduchou modifikaci modulů, komponent a pluginů dle potřeb uživatele. Pokud ve Správci šablon zvolíme možnosti Vytvořit přepsání, zobrazí se přehled všech dostupných objektů. Zvolený objekt se zkopíruje do složky `html` v kořenovém adresáři šablony. Pokud adresář `html` neexistuje, automaticky se vytvoří. Při načítání stránky systém nejdříve prohledá adresář `html` a zobrazí objekty v něm uložené, až poté začne s načítáním standardního výstupu.

Pokud například chceme, aby uživatel v přihlašovacím formuláři zadával mail místo uživatelského jména, vytvoříme přepsání přihlašovacího formuláře (`mod_login`) a modul upravíme. Systém tak bude načítat upravený modul ze složky `html`. Výhoda tohoto postupu je v tom, že při aktualizaci systému, šablony či komponenty není nutné znovu provádět úpravy kódu, které by se aktualizací přehrály, jelikož upravený kód zůstává v adresáři `html`.

ZÁVĚR

Redakční systémy jsou v současnosti rozšířeným nástrojem pro tvorbu a správu webu. Vzhledem k jejich existenci odpadá nutnost znalosti programovacích jazyků, uživatel se tedy může zcela soustředit na vytváření obsahu.

V práci byl uveden obecný popis redakčních systémů, podstatnější pozornost však byla věnována konkrétnímu systému s názvem Joomla!. Šlo zejména o popis funkcí, vlastností, rozdílů mezi vývojovými verzemi, hlavních částí systému, ovládání, doplňků a instalace samotného systému. Byla nastíněna jedna z největších výhod redakčních systémů – jednoduchá aplikace šablon, které umožňují rychle a jednoduše změnit vzhled a strukturu webu.

Stěžejní část obsahu práce byla věnována tvorbě vlastní šablony. Výstupem se tak stala „od nuly“ vytvořená šablona pro systém Joomla!. Byly použity základní technologie (HTML, PHP, CSS) a popsány kroky, jak zrealizovat vlastní návrh šablony. Jednalo se především o popis adresářové struktury, informace o základních souborech šablon a jejich vlastnímu kódu, návrh grafického layoutu a v neposlední řadě byl vytvořen návod na zobrazení obsahu a rozmístění modulů.

Práce může posloužit jako jakýsi návod na realizaci webu s vlastním vzhledem. Postupy využívané při tvorbě šablon jsou popsány tak, aby jejich realizaci zvládli i méně zkušení uživatelé. Pro náročnější uživatele je určena část zabývající se dynamickými parametry šablon a dynamická změna rozvržení. Vlastní šablony poskytují uživateli téměř neomezené možnosti co se vzhledu i funkčnosti týče, tudíž je možné web upravit na požadovaný vzhled.

RESUMÉ

The bachelor thesis is devoted to content management systems, particular the system Joomla. This is the description of the basic parts, uses and how to install. A significant part of the work includes instructions how to create a custom template. There are specific steps realizations from graphic design to the installation. This is particularly the description of the structure, the basic files and scope of the necessary codes. All steps are written so that they are understandable for beginners. The work is supplemented by useful images for easier orientation in the topic

Custom template offers to the user almost unlimited possibilities about design and functionality. Applications created template is very simple, only need a few clicks of the mouse and the website has a completely new look. In the creation own templates the user is not limited and can take a certain look and layout. If you need to add other features can be applied to countless extensions, or create your own.

SEZNAM LITERATURY

- [1] RAHMEL, Dan. *Joomla!: podrobný průvodce tvorbou a správou webů*. Vyd. 1. Překlad Ondřej Gibl. Brno: Computer Press, 2010, 382 s. ISBN 978-80-251-2714-8.
- [2] Redakční systém Joomla! - co je zač, pohled do historie. [online]. [cit. 2013-12-09]. Dostupné z: <http://www.linuxexpres.cz/software/redakcni-system-joomla-co-je-zac-pohled-do-historie>
- [3] Joomla! - historie verzí. [online]. [cit. 2013-12-09]. Dostupné z: <http://www.cms-systemy.cz/redakcni-systemy/joomla/ostatni/39-historie-verzi.html>
- [4] Co je Joomla!?. [online]. [cit. 2013-12-09]. Dostupné z: <http://www.joomlaportal.cz/index.php/clanky-a-novinky/zaciname-s-cms-joomla/493-bart>
- [5] Proč na web nenasazovat Joomla 3.0 a další STS verze?. [online]. [cit. 2013-12-13]. Dostupné z: <http://www.joomlaportal.cz/index.php/clanky-a-novinky/komunita-joomlaportalcz/526-proc-na-web-nenasazovat-joomla-3-0-a-dalsi-sts-verze>
- [6] VÍT, Svatopluk. Joomla! 1.0.x versus Joomla! 1.5.x. *Linux Expres* [online]. 2008 [cit. 2013-12-13]. Dostupné z: <http://www.linuxexpres.cz/duel/joomla-1-0-x-versus-joomla-1-5-x>
- [7] A summary of the ganges which have been introduced by Joomla 2.5. *Learn Joomla!* [online]. [cit. 2013-12-13]. Dostupné z: <http://www.learnjoomla.co.uk/blog/a-summary-of-the-changes-which-have-been-introduced-by-joomla-25>
- [8] Access control list. *Wikipedia* [online]. 2013, 11. 3. 2013 [cit. 2013-12-13]. Dostupné z: http://cs.wikipedia.org/wiki/Access_control_list
- [9] HAGEN, Graf. What'snew in Joomla! 2.5?. *Learn Joomla!* [online]. 2012 [cit. 2013-12-13]. Dostupné z: <http://magazine.joomla.org/issues/issue-jan-2012/item/654-What-s-new-in-Joomla-2-5>

- [10] Google. *ReCAPTCHA* [online]. 2013. vyd. 2013 [cit. 2013-12-13]. Dostupné z: <http://www.google.com/recaptcha>
- [11] CAPTCHA. *Wikipedia* [online]. 2013, 7.8.2013 [cit. 2013-12-13]. Dostupné z: <http://cs.wikipedia.org/wiki/CAPTCHA>
- [12] Co to znamená, když se řekne responsivní web – úvod do responsivního webdesignu. In: TOMAN, Jan. *Programujte.com* [online]. 2012 [cit. 2013-12-21]. Dostupné z: <http://programujte.com/clanek/2012012500-co-to-znamená-kdyz-se-rekne-responsivni-web-uvod-do-responsivniho-webdesignu/>
- [13] Joomla! 2.5 vs Joomla! 3.0. In: *Website 21* [online]. 2012 [cit. 2013-12-21]. Dostupné z: <http://www.website21.cz/joomla-2-5-vs-joomla-3-0>
- [14] DINH, Hung. 20 New Things: Joomla 2.5 vs. Joomla 3.0. *magazine.joomla.org* [online]. 2012 [cit. 2013-12-21]. Dostupné z: <http://magazine.joomla.org/issues/issue-nov-2012/item/933-20-new-things-joomla25-vs-joomla30>
- [15] MALÝ, Martin. LESS: stejné CSS za méně peněz. *Zdrojak.cz* [online]. 2010 [cit. 2013-12-21]. Dostupné z: <http://www.zdrojak.cz/clanky/less-stejne-css-za-mene-penez/>
- [16] HANÁK, Drahomír. LESS: CSS preprocesor pro pohodlnější stylování. *Devbook.cz* [online]. 2013 [cit. 2013-12-21]. Dostupné z: <http://www.devbook.cz/less-css-preprocesor>
- [17] KRCHA, Tomáš. Flex 4: Drag-and-drop. *Zdrojak.cz* [online]. 2010 [cit. 2013-12-21]. Dostupné z: <http://www.zdrojak.cz/clanky/flex-4-drag-and-drop/>
- [18] Drag and Drop. *RAYNET Cloud CMR* [online]. [cit. 2013-12-21]. Dostupné z: <http://raynet.cz/co-je-drag-and-drop.html>
- [19] HEROUT, Tomáš. Instalace Joomla 2.5 krok za krokem. *Helpmark.cz* [online]. [cit. 2013-12-24]. Dostupné z: <http://www.helpmark.cz/navody/joomla-2-5/143-instalace-joomla-25-krok-za-krokem>

- [20] MORAVEC, Zdeněk. PHP a MySQL – MySQLi - 1. díl. *Programujte.com* [online]. 2010 [cit. 2013-12-24]. Dostupné z: <http://programujte.com/clanek/2009103100-php-a-mysql-mysqli-1-dil/>
- [21] LINHART, Jan. Jaký webhosting vybrat pro Joomla?. *Escope.cz* [online]. 2013 [cit. 2013-12-24]. Dostupné z: <http://www.escope.cz/cz/blog/105-jaky-webhosting-vybrat-pro-joomla>
- [22] Instalace Joomla! 3.1.0. *Opensource.cesky-hosting.cz* [online]. 2013 [cit. 2013-12-24]. Dostupné z: <http://opensource.cesky-hosting.cz/redakcni-systemy/joomla/instalace-joomla-310/>
- [23] Joomla 3.0 - Instalace a čeština. *Navody.c4.cz* [online]. [cit. 2013-12-24]. Dostupné z: <http://navody.c4.cz/joomla-3-0-instalace-a-cestina>
- [24] VÍT, Svatopluk. Redakční systém Joomla - komponenty, moduly, šablony. *Linuxexpres.cz*[online]. 2008 [cit. 2013-12-27]. Dostupné z: <http://www.linuxexpres.cz/software/redakcni-system-joomla-komponenty-moduly-sablony>
- [25] GRAF, Hagen. *Building websites with Joomla! 1.5: the best-selling Joomla! Tutorial guide up dated for the final release* [online]. Birmingham, [England]: Packt, c2008, xii, 363 p. [cit. 2013-12-27]. From technologies to solutions. ISBN ISBN 978-1-847195-30-2. Dostupné z: http://meloun.upce.cz/docs/building_websites_with_joomla.pdf
- [26] Co nás potěšilo v Joomla 2.5. In: *Redakcni-systemy.com* [online]. 2008 [cit. 2013-12-27]. Dostupné z: <http://www.redakcni-systemy.com/joomla/navody/782-co-nas-potilo-v-joomla-25>
- [27] [Http://www.linuxexpres.cz/software/redakcni-system-joomla-sprava-uzivatelu](http://www.linuxexpres.cz/software/redakcni-system-joomla-sprava-uzivatelu). In: VÍT, Svatopluk. *Linuxexpres.cz* [online]. 2008 [cit. 2013-12-27]. Dostupné z: <http://www.linuxexpres.cz/software/redakcni-system-joomla-sprava-uzivatelu>
- [28] VÍT, Svatopluk. Redakční systém Joomla – píšeme články. In: *Linuxexpres.cz* [online]. 2008 [cit. 2013-12-28]. Dostupné z: <http://www.linuxexpres.cz/software/redakcni-system-joomla-piseme-clanky>

- [29] RADTKE, Angie. *Joomla!: tvorba a úprava šablon*. Vyd. 1. Brno: Computer Press, 2011, 342 s. ISBN 978-80-251-3696-6.
- [30] Validní kód. In: *Webtvorba.cz* [online]. [cit. 2014-03-17]. Dostupné z: <http://www.webtvorba.cz/pristupnost/validni-kod.html>
- [31] SATRAPA, Pavel. Jaké barvy zvolit. *Nti.tul.cz/* [online]. [cit. 2014-03-27]. Dostupné z: <http://www.nti.tul.cz/~satrapa/www/kurs/volbabarev.html>
- [32] Adobe BrowserLab preview 4: testování vzhledu webových stránek. *Grafika.cz* [online]. [cit. 2014-03-27]. Dostupné z: <http://www.grafika.cz/rubriky/software/adobe-browserlab-preview-4-testovani-vzhledu-webovych-stranek-137568cz>
- [33] XAMPP. *Itbiz.cz* [online]. [cit. 2014-03-29]. Dostupné z: <http://www.itbiz.cz/slovník/informacni-technologie-it/xampp>
- [34] CSS prefixy. MICHÁLEK, Martin. *Vzhůru dolů* [online]. [cit. 2014-04-02]. Dostupné z: <http://www.vzhurudolu.cz/prirucka/prefix>
- [35] Create a Template with Joomla: Step by Step. NORTH, Barrie. *Tutsplus* [online]. 2011 [cit. 2014-04-02]. Dostupné z: <http://code.tutsplus.com/tutorials/create-a-template-with-joomla-step-by-step--net-20048>

SEZNAM OBRÁZKŮ

Obrázek 1 - princip funkčnosti běžného webu	6
Obrázek 2 - průběh požadavku na zobrazení stránky v systému Joomla!.....	6
Obrázek 3 - instalace XAMPP.....	13
Obrázek 4 - odstranění instalační složky	16
Obrázek 5 - výběr jazyka.....	16
Obrázek 6 - příklad stromové struktury šablony	23
Obrázek 7 - současný stav šablony.....	30
Obrázek 8 - rozvržení pozic pro umístění modulů	35
Obrázek 9 - možný vzhled po „nastylování“	36
Obrázek 10 - pozici "menu_poz" není přiřazen žádný modul.....	40
Obrázek 11 - pozici "menu_poz" je přiřazen modul	40
Obrázek 12 - pokročilé nastavení před „překladem“	41
Obrázek 13 - pokročilé nastavení po "překladu"	42
Obrázek 14 - validace	44
Obrázek 15 - instalace šablony.....	47

PŘÍLOHY

Všechny zdrojové kódy vytvořené šablony jsou nahrány na přiloženém CD.

Dále na je šablona online k dispozici na adrese www.bp.moxo.cz