

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

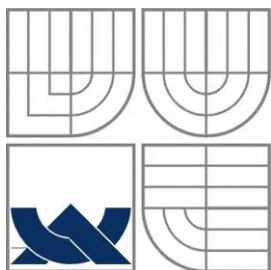
MODULOVÝ REDAKČNÍ SYSTÉM S VYUŽITÍM
TECHNOLIE AJAX

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

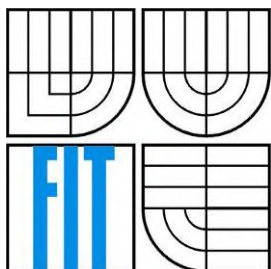
AUTOR PRÁCE
AUTHOR

PAVEL MACHOUREK

BRNO 2009



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

MODULOVÝ REDAKČNÍ SYSTÉM S VYUŽITÍM TECHNOLOGIE AJAX

MODUL MANAGEMENT SYSTEM WITH USING AJAX TECHNOLOGY

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

PAVEL MACHOUREK

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. PETR ČÁSTEK

BRNO 2009

Abstrakt

Tato práce se zabývá popisem redakčních systémů a jejich možnostmi. Představeny jsou všechny důležité technologie, které se v redakčních systémech běžně používají jako PHP, ASP.NET, Java, AJAX a jiné. Popsána je rovněž implementace vlastního redakčního systému s využitím PHP, MySQL a AJAX a možnosti dalšího rozšíření systému.

Abstract

This work is focused on description of Web Content Management Systems and its capabilities. All important technologies, which are usually used such as PHP, ASP.NET, Java, AJAX etc. are introduced. There is also described implementation of own Web Content Management System using PHP, MySQL and AJAX and possible future development.

Klíčová slova

PHP, MySQL, AJAX, Apache, HTML, CSS, WYSIWYG, Redakční systém, databáze, web, moduly, CMS, JavaScript

Keywords

PHP, MySQL, AJAX, Apache, HTML, CSS, WYSIWYG, Web Content Management System, database, web, modules, CMS, JavaScript

Citace

Machourek Pavel: Modulový redakční systém s využitím technologie AJAX, bakalářská práce, Brno, FIT VUT v Brně, 2009

Modulový redakční systém s využitím technologie AJAX

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením Ing. Petra Částka
Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Pavel Machourek

19. 5. 2009

Poděkování

Na tomto místě bych rád poděkoval svému vedoucímu Ing. Petrovi Částkovi za rady a připomínky,
které mi pomohly projekt vypracovat.

© Pavel Machourek, 2009

*Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních
technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je
nezákonné, s výjimkou zákonem definovaných případů.*

Obsah

Obsah	1
1 Úvod.....	3
2 Redakční systém.....	4
2.1 Vysvětlení pojmu.....	4
2.2 Princip funkce.....	6
2.3 Dělení redakčních systémů.....	7
2.3.1 Zpracování.....	8
2.3.2 Cena.....	8
2.3.3 Přístup	9
3 Platformy.....	10
3.1 PHP.....	10
3.2 ASP.NET	10
3.3 Java.....	11
3.4 Python.....	11
4 Ukládání dat	12
4.1 Soubory.....	12
4.2 Databáze	12
5 AJAX.....	14
5.1 Web 2.0.....	14
5.2 Nevýhody AJAXu	14
6 Vlastnosti a funkce redakčních systémů.....	16
6.1 Druh obsahu.....	16
6.2 Editace obsahu.....	16
6.3 Správa uživatelů	17
6.4 Bezpečnost.....	17
6.5 Uživatelské rozhraní	18
6.6 SEO.....	19
6.7 Moduly.....	19
7 Příklady redakčních systémů přílohy	20
7.1 Drupal.....	20
7.2 Joomla.....	21
7.3 Plone.....	21
7.4 CMS Made Simle	22
7.5 LoveCMS.....	22

7.6	NanoCMS	23
8	Implementační část.....	24
8.1	Požadavky.....	24
8.2	Návrh	24
8.3	Popis implementace.....	28
8.3.1	Aplikační logika	28
8.3.2	Uživatelské rozhraní.....	29
8.3.3	Administrátorské rozhraní.....	29
8.3.4	Moduly	30
9	Závěr.....	32

1 Úvod

Internet – pomocník nebo svůdce? Ať už si myslíme o něm cokoliv, jen málokdo si dnes umí život bez internetu představit. Denně na něm vyhledáme informace, komunikujeme s jinými lidmi, hrajeme hry, nebo jiným způsobem krátíme volnou chvíli. Abychom to vše a ještě mnohem víc mohli vykonávat, je třeba, aby se výše jmenované na internetu vyskytovalo. Jinými slovy řečeno, musí být lidé, kteří to na internet dají a musí existovat techniky, které jim to umožní.

V současné době se využívá připravených nástrojů, jež umožňují rychlé a pohodlné umístění potřebných informací na internet. Pracné ruční úpravy jsou vhodné jen pro projekty malého rozsahu. Správu větších projektů můžeme vykonávat prostřednictvím redakčního systému, který nám umožňuje snadné a rychlé zveřejnění data bez jakékoliv znalosti HTML, CSS, či PHP.

V současnosti je k dispozici nepřeberné množství redakčních systémů. Neustále vznikají nové a nové. Tento trend je určen především velmi rozdílnými požadavky provozovatelů. Jaké jsou jednotlivé požadavky? Jaké aspekty hrají důležitou roli při výběru nebo při tvorbě redakčního systému? Nejen tyto otázky se pokusím ve své práci zodpovědět.

Začneme však od začátku a seznámíme se s pojmem redakční systém. Přiblížíme si princip jeho funkce a uvedeme základní rozdělení redakčních systémů. Objasníme si problematiku samotného vymezení pojmu, neboť v názvosloví často dochází k chybnému zaměňování několika termínů. Od terminologie přejdeme pozvolna k praxi. Seznámíme se s platformami, na nichž jsou redakční systémy postaveny. Jedná se především o PHP, coby nejpoužívanější skriptovací jazyk šířený jako svobodný software, ASP.NET – zástupce komerční platformy od firmy Microsoft a některé další.

Hlavním úkolem redakčního systému je snadná manipulace s daty. Jaké máme možnosti při ukládání různých druhů dat, jaké jsou jejich výhody a nevýhody? To se stalo tématem čtvrté kapitoly.

AJAX – pojem, který dnes slyšíme ze všech stran. Jak vyplývá z názvu práce, vztahuje se i k redakčním systémům. Nejedná se pouze o jednu technologii, nýbrž o několik různých technik umožňujících dosáhnout požadovaného výsledku. AJAX zprostředkovává lepší interakci s uživatelem tím, že snižuje množství komunikace mezi serverem a klientským počítačem. Abychom AJAX lépe poznali, je mu věnována celá jedna kapitola.

Před čtením šesté kapitoly již budeme mít určitý přehled o principu fungování redakčního systému a o technologiích, které se běžně využívají. Budeme se tedy moci zaměřit na konkrétní funkce redakčních systémů a v následující kapitole si několik zajímavých a nejpoužívanějších systémů představíme.

Práce je zakončena návrhem a popisem implementace celého redakčního systému. Nachází se zde rovněž diagram případů použití popisující role uživatelských skupin v systému a ER diagram, který byl použit u návrhu uchování dat v databázi.

2 Redakční systém

2.1 Vysvětlení pojmu

Letmý pohled na českou Wikipedii prozradí, že redakční systém je totéž co systém pro správu obsahu. „Systém pro správu obsahu (CMS z anglického content management system) je software zajišťující správu dokumentů, nejčastěji webového obsahu. V dnešní době se jako CMS zpravidla chápou webové aplikace, někdy s případným doplňkovým programovým vybavením u klienta.“ [1]. Se stejným názorem se můžeme setkat i v jiných zdrojích. Komerční redakční systém WEBJump se zabývá rozdíly mezi termíny CMS, redakční systém, administrativní systém a publikační systém hned na titulní stránce. Dochází však k závěru, že se jedná o různé názvy pro stejnou věc. [2]. Rovněž tvůrci redakčního systému phpRS v tom mají jasno: „ ... pojem "redakční systém" má mnoho alternativních označení, které mají samozřejmě všechny shodný význam. Jmenovitě jde o následující slovní spojení a zkratky: RS, publikační systém, CMS a Content Management System.“ [3].

Je tomu ale doopravdy tak? Anglická wikipedie nabídne rozdělení systémů pro správu obsahu podle oblasti použití na:

- podnikový systém pro správu obsahu (Enterprise CMS, ECM – Enterprise Content Management)
- webový systém pro správu obsahu (Web CMS, WCM – Web content management)
- mobilní systém pro správu obsahu (Mobile CMS)
- komponentní systém pro správu obsahu (Component CMS)

Podnikový systém pro správu obsahu je podle Association for Information and Image Management (AIIM¹) systém, který zahrnuje strategie, metody a nástroje pro vytváření, správu, ukládání, ochranu a distribuci obsahu a dokumentů, týkajících se organizačních procesů. Jeho nástroje a strategie umožňují správu organizačních nestrukturovaných informací, ať jsou kdekoliv. [4]. Integrace s dalšími podnikovými systémy je pro něj rovněž typická.

Webový systém pro správu obsahu je obvykle implementován jako webová aplikace určená pro vytváření a správu HTML obsahu. Jeho použití je tedy mnohem specifičtější zaměřené než je tomu u ECM. [5].

Další oblasti použití systémů pro správu obsahu již nejsou pro zařazení redakčního systému důležité. Podstatné je, že význam těchto pojmů je velmi obtížné přesně vymezit, ani sama organizace

¹ AIIM je nezisková organizace založená již v roce 1943, která se zaměřuje na správu dokumentů, obsahu, nahrávek a firemních procesů.

AIIIM si jimi není jistá, když za poslední čtyři roky svou definici ECM třikrát změnila. Například podle Ulricha Kampffmeyera zahrnuje podnikový systém pro správu obsahu teoreticky i WCM. [6].

Vztah mezi redakčním systémem a systémem pro správu obsahu tedy není tak triviální, jak by se mohlo na první pohled zdát. Slovník internetové agentury Symbio uvádí: „CMS se vyvinuly z jednodušších tzv. redakčních systémů (RS). Tak označujeme aplikace vytvořené především za účelem oddělení obsahu od prezentace, které sloužily zpočátku zejména zpravodajským serverům pro publikaci článků (odtud tedy označení "redakční systémy)".“ [7]. Ondřej Vích a Václav Bittner ve své prezentaci představují redakční systém jako systém úzce zaměřený pouze na správu internetové prezentace. [8].

Asociace sdružující IT společnosti a internetové agentury specializující se na vývoj internetových a intranetových řešení Asociace.biz uvádí význam pojmu redakční systém takto: „Redakční systém umožňuje zaměstnancům snadno vytvářet a spravovat obsah internetové prezentace bez nutných odborných znalostí. Prezentace je pak tvořena přímo autory obsahu a ten je pak publikován automaticky, nikoliv IT oddělením.“ [9]. Dokonce se přímo věnuje porovnání redakčního systému a WCMS: „Redakční systém však není Web Content Management System (WCMS / CMS) a pro velkou firmu nebo instituci, která potřebuje systematickou správu veškerého digitálního obsahu určeného nejen pro internetovou prezentaci, ale také např. pro podnikový portál, extranet, wapovou prezentaci nebo tištěný katalog, by byl redakční systém nedostačující a tudíž špatnou investicí. WCMS disponuje mnohem sofistikovanějším a výkonnějším procesním řízením správy a publikace obsahu a také lepší podporou dalších firemních procesů než redakční systém. WCMS má široké portfolio možností publikace, využívá celou řadu integračních nástrojů – např. s ERP systémy (SAP, LCS Noris, Navision apod.) a umožňuje spravovat mnohem širší spektrum digitálního obsahu.“ [tamtéž]. U pojmu *Web Content Management System* pak dodává, že „Web Content Management System (WCMS) nastupuje tam, kde běžné redakční systémy nestačí. Veškerou funkcionalitu redakčních systémů přebírá, rozšiřuje a doplňuje pro použití v rámci rozsáhlých internetových prezentací a portálů, často s početným autorským prostředím. Oproti redakčnímu systému poskytuje široké možnosti v otázce integrací s externími aplikacemi, podporuje složitější workflow procesy a snadnou přípravu jazykových mutací jednotlivých prezentací.“ [10]. Zde už ovšem nejde o klasický WCMS, ale, dle dříve zmíněné definice z wikipedie, se jedná spíše o ECMS implementovaný pomocí webového rozhraní.

Je tedy zřejmé, že určit co je redakční systém není snadné. Hlavní problém je nejspíš způsoben rozdílností české a anglické terminologie. Napomáhá tomu i zažité označování běžných WCMS jako CMS. Anglickému pojmu *CMS* odpovídá český *systém pro správu obsahu*, který se běžně nepoužívá. Český redakční systém nemá anglický ekvivalent. V angličtině běžně používané *CMS* se pravděpodobně začalo do češtiny překládat jako *redakční systém* (místo nepoužívaného systému pro správu obsahu), čímž vznikl zmatek v pojmech. Proti tomu se snaží vývojáři Enterprise CMS bránit

důsledným používáním označení ECMS. Chtějí tím dát najevo, že se nejedná o běžný CMS, i když by označení CMS bylo pro ECMS mnohem přesnější než jak je zažité označovat tak běžné WCMS.

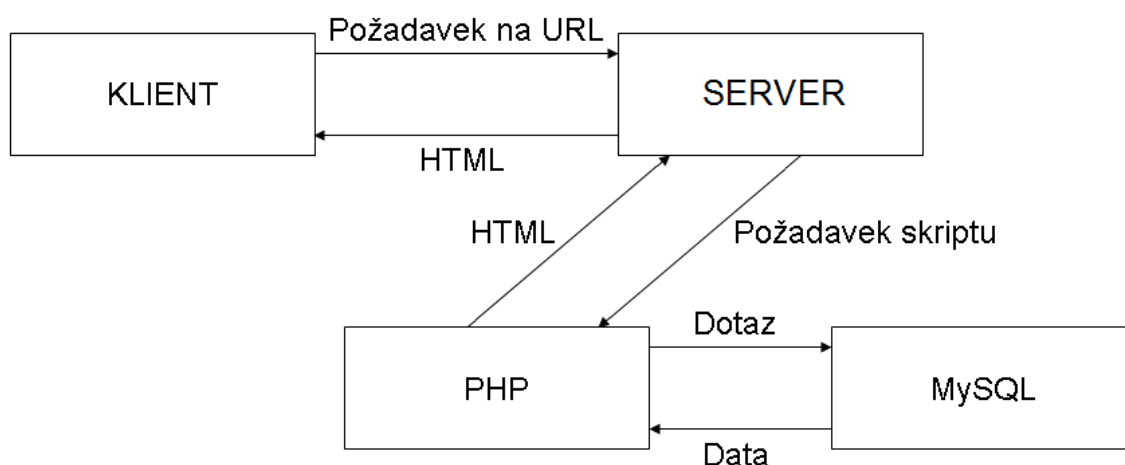
Redakční systém je *webový systém pro správu obsahu*, často je ovšem tento pojem chybně používán jako obecné označení systému pro správu obsahu (stejně jako v angličtině se označení CMS používá ve spojení s běžnými WCMS). Prakticky redakční systém představuje aplikaci, která spravuje data a informace různého charakteru a obsahu a zároveň se stará o jejich zobrazení. Jeho správa i výstup jsou realizovány přes webové rozhraní.

Dále v textu budou použity pojmy přímo související s redakcí. Jsou to:

- Článek – základní obsahová jednotka redakčního systému, která obsahuje textové informace. Může být doplněn i daty jiného než textového typu.
- Redaktor – uživatel systému, který má oprávnění vytvářet nové články.
- Šéfredaktor – uživatel systému, který definuje kategorie obsahu a schvaluje jednotlivé články před zveřejněním. Také může články vytvářet.

2.2 Princip funkce

Redakční systém je nejčastěji soubor skriptů umístěných na serveru. O spouštění těchto skriptů se stará interpret požadovaného jazyka. Tyto skripty reagují na veškeré podněty od uživatelů či správců systému a generují požadovaný výsledek. Samotný redakční systém neobsahuje data, ty jsou zpravidla uložena v externích souborech nebo v databázi. Veškerá obsluha systému by měla být prováděna přes webové rozhraní. Zásahy do zdrojových kódů by měly být buď zcela eliminovány, nebo by měly být potřebné maximálně při instalaci systému či při zásadních změnách jeho nastavení.



Obrázek 1: Příklad architektury klient – server s dynamickým generováním stránky

Redakční systém je typická aplikace typu klient-server. Serverová část je tvořena nejčastěji interpretem skriptovacího jazyka a stará se o příjem a zpracování požadavků od klienta (webový

prohlížeč). Komunikace probíhá pomocí protokolu HTTP, případně HTTPS. Od klienta směrem k serveru putují požadavky na zobrazení některé stránky nebo vložená data a server odešle klientovi požadovanou stránku v HTML či XHTML jazyce a případně zpracuje vstupní data. Na straně klienta může také docházet ke zpracování kódu – například pomocí Javascriptu. Vzhledem k potencionálnímu nebezpečí zde jsou značná omezení. Javascript je klíčová technologie pro AJAX, o kterém bude řeč později.

Serverová část je tedy od klienta oddělena a klient nemá přístup ke zdrojovým kódům tvořícím redakční systém, ale vidí pouze výsledek činnosti.

2.3 Dělení redakčních systémů

Redakční systémy se mohou dělit podle více kritérií – oblast použití, rozsah, cena, způsob přístupu nebo třeba použité platformy. Je možné narazit na minimalistické řešení stejně tak jako na mohutné komplexní systémy. Rovněž se velmi liší technologie, na kterých jsou tyto systémy postaveny.

Redakční systémy se nejčastěji dělí dle následujících kritérií:

Zpracování:

- online
- offline
- hybridní

Cena:

- volně dostupné
- komerční

Přístup:

- intranet
- internet

Platforma:

- PHP
- Java
- ASP.NET
- Python, Perl ...

Způsob ukládání dat:

- databázové
- souborové

2.3.1 Zpracování

Používají se tři základní druhy zpracování: offline, online a hybridní. Tyto druhy zpracování se liší tím, kdy jsou aplikovány prezentační šablony pro vytvoření výsledné webové stránky. Seth Gottlieb výstižně tyto způsoby přirovnává k pečení, smažení a předpečení chleba.¹

Systémy s *online zpracováním* aplikují šablony dle potřeby. Celá webová stránka je vytvořena v momentě, kdy přijde požadavek na její zobrazení („smažení“). Oproti tomu systémy s *offline zpracováním* mají veškerý obsah předpřipravený (například formou statického HTML – „pečení“). *Hybridní přístup* je kombinací předchozích, kdy je předem vygenerován spustitelný kód, který se spouští po příchodu požadavku („předpečení“).

Online zpracování poskytuje mnohem větší volnost v interakci s uživatelem. Zároveň však klade největší nároky na výkon serveru. Offline zpracování naopak umožňuje distribuovat vygenerovaný obsah na velké množství levných webových serverů. Tím zvládá obsluhovat podstatně větší množství uživatelů, ovšem za cenu menší interaktivity (bez zpracování informací na straně serveru). [11].

2.3.2 Cena

Není žádným překvapením, že redakční systémy jsou buď komerční, nebo zdarma. Systémů vyvíjených pod licenci GPL a distribuovaných zdarma je velké množství, přičemž kvalita těch nejrozšířenějších je na špičkové úrovni. Svědčí o tom například i fakt, že některé z nich se dočkaly komerční podpory².

Hlavní výhodou rozšířených nekomerčních systémů je obrovská komunita vývojářů a uživatelů. Vyskytne-li se nějaký problém, můžeme jej s touto komunitou konzultovat, případně zkusit vyhledat, jestli podobný problém již někdo jiný neřešil. Dostupná jsou i velká množství rozšíření těchto systémů v podobě modulů. Pokud systému některá funkce chybí, pravděpodobně existuje modul, který danou funkci do systému přidá.

Firmy nabízející komerční redakční systémy se nesnaží pouze o prodej svého systému, ale spolu s ním nabízejí kompletní řešení pro vytvoření a správu firemní intranetové/internetové prezentace. Mezi nabízené služby obvykle patří: analýzy, konzultace, instalace systému, grafický návrh, kustomizace, hosting, podpora a vývoj případných požadovaných funkcí. Je tedy patrné, že se orientují převážně na *B2B* sféru (*business to business*). Tam, kde by nekomerční redakční systém mohl nabídnout vyšší kvalitu systému, nemůže nabídnout služby spojené s kompletní tvorbou a podporou firemní prezentace.

¹ Proces pečení chleba: Přibližně v 80 % se proces přeruší a chleba se prudce zmrazí. V tomto stavu může být dlouho skladován při zachování požadovaných vlastností. V případě potřeby se snadno dopeče.

² Společnosti poskytující komerční podporu jsou *Acquia* (<<http://acquia.com/>>) a *Open Source Support Desk* (<<https://www.opensourcesupportdesk.com/>>).

2.3.3 Přístup

Redakční systém může být přístupný veřejnosti pomocí internetu, nebo pouze úzké skupině uživatel prostřednictvím intranetu. Na základě toho se budou pravděpodobně lišit požadavky na výkon a zabezpečení. Veřejný server s mnohem větším počtem uživatelů bude potencionálně víc ohrožen. Nejedná se samozřejmě o jediný ukazatel, od něhož se zabezpečení bude odvíjet, ale zcela jistě bude hrát podstatnou roli. To ale rozhodně neznamená, že neveřejný redakční systém nepotřebuje zabezpečení – důvěrná a citlivá data musí být i na internetu dobře chráněna.

3 Platformy

Platforma, neboli také framework, je hardwarové a softwarové prostředí umožňující činnost programů. V rámci redakčního systému se budeme dále zabývat především platformou softwarovou a to především ve významu programovacích jazyků s jejich knihovnamí. Podle použité platformy se odvíjí celá řada vlastností redakčního systému. Její výběr by rozhodně neměl být podceněn.

3.1 PHP

PHP je rekurzivní zkratka pro PHP: Hypertext Preprocessor (původně se používalo Personál Home Page tools). Jedná se o rozšířený univerzální skriptovací jazyk, který je zvláště vhodný pro vývoj webu. Jeho syntaxe je odvozena od jazyků C, Java a Perl. Hlavním cílem jazyka je umožnit webovým vývojářům rychle vytvářet dynamicky generované stránky. Od páté verze je PHP dobře vybaveno pro objektivě orientované programování.

PHP je možné používat k následujícím činnostem:

- Server-side scripting, neboli spouštění skriptů na straně serveru je tradiční využití na které je PHP především zaměřeno.
- Skriptování v příkazové řádce. Není zde potřeba server ani prohlížeč, stačí PHP parser.
- Vývoj klientských aplikací. Jedná se o okrajovou záležitost.

Další vlastností je multiplatformnost. PHP lze provozovat na všech běžných platformách a většině webových serverů. Umožňuje jak procedurální, tak i objektivě orientované programování, komunikaci pomocí mnoha protokolů a podporuje přes 20 různých databází. Dále PHP nabízí užitečné funkce pro zpracování textu (od regulárních výrazů až po parsování XML dokumentů).

Podle statistik se jedná o nejvyužívanější platformu již od roku 1996, kdy se PHP dostalo před NCSA HTTPd. Vrcholná léta PHP v letech 2004 až 2006 kdy se blížilo k 80% podílu na trhu, poté začal jeho podíl ukrajovat Microsoft s e svým serverem IIS a technologií ASP. Nejčastěji používaná verze je PHP 5.2 a podíl starších verzí neustále klesá. [12]

3.2 ASP.NET

ASP.NET je součástí frameworku .NET, vyvíjeného firmou Microsoft, a navazuje na předchozí technologii ASP, od níž se však značně liší. ASP.NET je objektivě orientované prostředí. Zdrojový kód se neinterpretuje, ale kompiluje. Tato kompilace probíhá ve dvou etapách. Nejdříve se kód zapsaný v C# zkompiluje do jazyka MSIL a ten je pak *Just In Time* kompilován až před zpracováním požadavku od klienta. Není však nutné používat C#, je možné si zvolit libovolný jazyk z rodiny

.NET. Přestože se jedná o řešení firmy Microsoft a mohlo by se zdát, že pro provoz aplikací postavených na ASP.NET bude potřeba operační systém Microsoft Windows, není tomu tak. ASP.NET lze provozovat díky projektu Mono i na operačním systému z rodiny GNU/Linux.

3.3 Java

Firma *Sun Microsystems* má taktéž své želízko v ohni. Je jím velmi univerzální a často používaný objektově orientovaný programovací jazyk Java.

Nasazení Javy je až neskutečné – od mobilních telefonů, přes běžné stolní počítače až po rozsáhlé distribuované systémy. V roce 2007 Sun uvolnil zdrojové kódy a od té doby je Java vyvíjena jako Open Source.

Java využívá ke svému běhu *Java Virtual Machine*, což je prostředí interpretující *bytecode*. Bytecode je vytvořen kompilací zdrojových kódů. Díky tomu je Java nezávislá na operačním systému. Jediné co je potřeba, je existence Java Virtual Machine na požadovaném operačním systému. Používá *Just In Time* kompilaci – tedy přímo za běhu programu.

Pro vývoj webových aplikací jsou určeny součásti této platformy *Java EE (Enterprise Edition)* a její části *Java Servlets* a *Java Server Pages*.

3.4 Python

Python je dynamický objektově orientovaný programovací jazyk, který lze použít v mnoha oblastech. Nabízí silnou podporu pro integraci s jinými jazyky a nástroji. Obsahuje mnoho standardních knihoven. Další předností je jeho rychlé pochopení a vysoká produktivita práce. [13].

Tak stručně by se dal Python představit. Jedná se o interpretovaný (někdy označován jako skriptovací) jazyk. Skriptovací jazyky neprodukují zrovna ty nejrychlejší programy, ale v případě pythonu jde především o efektivitu programování – a ta je na velmi dobré úrovni. I přesto ani výkon není špatný (jelikož jsou jeho kritické knihovny implementovány v jazyce C), je přibližně 3–5x výkonnější než PHP.

Začít v pythonu programovat není nijak náročné. Často můžeme narazit na doporučení, že se jedná o jeden z nejvhodnějších jazyků pro začátečníky. Klade totiž důraz na jednoduchou syntaxi a přehlednost zdrojových kódů. Rozhodně to ale neznamená, že by byl vhodný pouze pro začátečníky. Právě naopak. Python neztrácí nic ze své elegance ani u rozsáhlých projektů.

Pro tvorbu webových aplikací je důležitá implementace modulu *BaseHTTPServer* v základním balíku. Tento modul však obsahuje pouze základní prostředky pro komunikaci pomocí http protokolu. Python nabízí kompletní propracované frameworky jako *Pylons*, *TurboGears*, *Zope* a některé další.

4 Ukládání dat

Hlavním smyslem redakčního systému je snadná manipulace s daty. Data mohou být různého charakteru – od textu přes obrázky až po audio/video nebo binární soubory. Redakční systém musí být schopný manipulovat s tím druhem dat, který se v něm má používat. Důležitou otázkou tedy je, jaká data se mají kam ukládat. Možnosti uložení jsou v podstatě dvě. Jedná se buď o uložení do souborů, nebo do databáze.

4.1 Soubory

Ukládání rozsáhlejších textů do souborů je vhodné pro aplikace menšího rozsahu. Při práci s malým množstvím dat je manipulace se soubory jednodušší – není potřeba se zabývat přístupem do databáze. Tento přístup ovšem s přibývajícím množstvím dat brzy narazí na obtíže. Manipulace se stane pomalejší a pracnější než při užití databáze. Pokud se chystáme v systému použít ukládání dat do souborů, je potřeba danou situaci zvážit i s výhledem do budoucnosti. Z počátku vhodný systém se může nestát po několika měsících nepřehledným a nepoužitelným.

Soubory se často používají ve spojení s obrázky a jinými binárními daty. Situace, kdy jsou texty uloženy v databázi a binární data v souborech je velmi častá. Souboru mohou být uloženy buď všechny v jedné složce, nebo může mít každý uživatel svou vlastní složku. Rovněž, pokud se v redakčním systému běžně používá více obrázků v jednom článku, tak může mít každý článek svou složku. Jaký přístup je nejlepší není možné jednoznačně rozhodnout. Vždy záleží na konkrétním použití redakčního systému (např.: Kdo vytváří obsah – pouze úzká skupina autorů, nebo libovolný (registrovaný) uživatel? Jaký je druh obsahu?).

S tímto přístupem je mohou nastat problémy u webhostingů, kde bývají přísná omezení uživatelských práv a manipulace se soubory se stane buď velmi obtížnou, nebo téměř nemožnou.

4.2 Databáze

Databáze je uspořádaná množina dat. „Předchůdcem databází byly papírové kartotéky, které umožňovaly uspořádávání dat podle různých kritérií a zařídování nových položek. Veškeré operace s nimi prováděl přímo člověk. Správa takových kartoték byla v mnohém podobná správě dnešních databází. Dalším krokem bylo převedení zpracování dat na stroje.“ [14]. Existuje několik databázových modelů, ale ve vztahu k redakčnímu systému nás budou zajímat pouze databáze relační.

„*Relační databázový model* sdružuje data do tzv. relací (tabulek), které obsahují n-tice (řádky). Tabulky (relace) tvoří základ relační databáze. Tabulka je struktura záznamů s pevně stanovenými položkami (sloupci – atributy). Každý sloupec má definován jednoznačný název, typ a rozsah, neboli

doménu. Záznam se stává n-ticí (řádkem) tabulky. Pokud jsou v různých tabulkách sloupce stejného typu, pak tyto sloupce mohou vytvářet vazby mezi jednotlivými tabulkami. Tabulky se poté naplňují vlastním obsahem – konkrétními daty. Kolekce více tabulek, jejich funkčních vztahů, indexů a dalších součástí tvoří relační databázi.

Relační model přináší celou řadu výhod, zejména mnohdy přirozenou reprezentaci zpracovávaných dat, možnost snadného definování a zpracování vazeb apod.

Relační model klade velký důraz na zachování integrity dat. Zavádí pojmy referenční integrity, cizí klíč, primární klíč, normální tvar apod.

S relačními databázemi je úzce spojen pojem SQL (Structured Query Language), neboli strukturovaný dotazovací jazyk. Jeho základní model je obecně použitelný pro většinu relačních databází. Od svého vzniku prošel několika revizemi a poskytovatelé databázových produktů jej obohatili o různá lokální rozšíření. Tato rozšíření ale nejsou vzájemně kompatibilní.“ [15].

Drtivá většina redakčních systémů databázi využívá, a to nejčastěji MySQL. Redakčních systémů schopných využívat MySQL je přibližně dvojnásobek, než těch které umí využívat jiná řešení – SQL Server, Oracle a PostgreSQL. Zatímco Oracle a Microsoft SQL Server získávají body především u systémů s proprietární licencí, MySQL dominuje mezi systémy používajícími PHP. [16].

Použití databáze se přímo nabízí pro uchování textových dat v redakčním systému. Velmi snadno se z databáze získávají, filtrují a upravují. U každého článku je možné mít všechny doplňující informace jako jméno autora, datum zveřejnění, zda článek prošel schválením šéfredaktora/administrátora, do které kategorie obsahu článek spadá, počet zobrazení a mnohé další. Není pak problém zobrazit články od jednotlivých autorů, pouze některé kategorie, nejčtenější články... Všechny tyto možnosti databáze automaticky poskytuje, jejich implementace v redakčním systému, který databázi nepoužívá, by byla mnohem náročnější.

U binárních dat již není situace tak jednoduchá. Databáze samozřejmě jejich ukládání podporuje a je praktické mít o datech v databázi ještě další informace, ale jsou zde i nevýhody. Pokud v databázi budou pouze články, nebude příliš objemná, ale po přidání binárních dat velmi rychle nabude na objemu. To může být často problém u webhostingu, jelikož velikost databáze může být omezena víc než kapacita pro celou webovou prezentaci, takže v souborech by bylo možné uložit mnohem více dat. Také v případě zálohy databáze bude výsledná záloha objemnější a manipulace se zálohou komplikovanější.

5 AJAX

AJAX, neboli *Asynchronous JavaScript and XML* je skupina propojených programovacích technik používaná k vytváření interaktivních webových aplikací. Pomocí Ajaxu mohou webové aplikace získávat data ze serveru bez potřeby znovu načítat obsah celé stránky. Navzdory názvu, Ajax nemusí nutně využívat JavaScript a XML – nejedná se totiž o konkrétní technologii ale použití několika technologií k dosažení určitého cíle.

Technologie, které Ajax používá:

- HTML a CSS (HyperText Markup Language a Cascading Style Sheets)
- DOM (Dokument Object Model)
- JavaScript
- XMLHttpRequest

Ačkoli náznaky podobné technologie se objevily již v roce 1996, výrazného rozšíření se Ajax dočkal teprve před několika lety díky jeho masivnímu využití ve službách provozovaných společností Google (nejprve Gmail a brzy poté Google Maps). [17]. Nyní, když se orientace webu mění z pouhého úložiště informací na zdroje obsahující informace a funkcionalitu, získává Ajax na popularitě. Tento vývoj bývá označován jako „Web 2.0“ a Ajax je s tímto pojmem neodmyslitelně spojen, neboť je jeho klíčovou součástí.

5.1 Web 2.0

Ať už je tento pojem kritizován nebo vychvalován, velmi dobře popisuje možnosti a využití technologie Ajax. Termín Web 2.0 označuje další fázi vývoje webu, kde kromě již zmíněného přechodu od informací k funkčnosti je kladen důraz také na socializaci a lepší organizaci obsahu.

Pokud Netscape byl tahounem „Webu 1.0“, pak u Webu 2.0 je dominantou firma Google. Google odstartoval svou existenci jako webová aplikace, která nikdy nebyla prodávána. Nevydává ani neplánuje nové verze, jen se postupně vyvíjí. Je to dáno současnou koncepcí, která se přesouvá od zaměření na produkty ke službám. Zachytit tento trend se snaží všichni velcí hráči. Kromě Google to je třeba i Microsoft se svými službami Windows Live. [18].

5.2 Nevýhody AJAXu

AJAX ovšem nepřináší pouze samá pozitiva. Jeho použití musí být mezeno pouze na pomocné funkce. Nemělo by se jednat o zobrazování hlavních obsahových prvků. Ne každý prohlížeč

JavaScript podporuje (mobilní telefony) případně může být JavaScript záměrně zakázán. Uživatel by se tak ze stránky vůbec nic nedozvěděl.

Nesmíme zapomenout na největšího „čtenáře“ internetu – *GoogleBot*. Ten neustále prochází veškerý obsah internetu a indexuje ho. Na jeho výsledcích závisí umístění stránky ve vyhledávači Google. JavaScript však není nepodporován GoogleBotem, čímž stránka založená na AJAXu nemá šanci uspět v SEO (blíže viz kapitola 6).

Tím, že se mění pouze část stránky, zůstane jí její původní URL adresa a není tak možné za normálních okolností na akce provedené AJAXem aplikovat tlačítka prohlížeče zpět a vpřed, uložit stránku do oblíbených nebo někomu poslat odkaz.

Kvůli daným nedostatkům je možné AJAX použít pouze na okrajové, pomocné funkce. AJAX tím ztrácí na důležitosti. Jeho použití není nezbytné, jedná se pouze o zpříjemnění některých uživatelských akcí.

Stručně řečeno – hlavní nevýhodou AJAXu je jeho využitelnost. Využívá se pouze pro drobné doplňky, nebo naopak u velkých webových aplikací, které vyvíjejí obří společnosti jako Google či Microsoft.

6 Vlastnosti a funkce redakčních systémů

Možností, co všechno může redakční systém umět je nepřehledné množství. Najít nebo vytvořit redakční systém, který bude umět všechno a vyhovovat všem je utopie. Záleží pouze na požadavcích a očekávání. Ty určují, který systém bude pro danou situaci ten nejvhodnější. V této kapitole se podíváme na běžné vlastnosti a funkce redakčních systémů.

6.1 Druh obsahu

Práce s textem je triviální a v žádném případě by neměla dělat potíže ani nejjednoduššímu redakčnímu systému. U obrázků již záleží, jaký způsob manipulace vyžadujeme – zda stačí schopnost nahrávat je na server a poté zobrazovat, nebo jsou požadovány i editační funkce. Editace textů a obrázků se dá často vhodně řešit použitím již hotového osvědčeného řešení. Správným výběrem lze ušetřit spoustu času, který by zabrala tvorba vlastního WYSIWYG¹ editoru a editoru obrázků. K tomu můžeme získat i některé funkce navíc. Běžně používané WYSIWYG editory jsou například *FCKEditor* nebo *TinyMCE* a pro editaci obrázků *ImageEditor*, *AIE* nebo *Maya*.²

6.2 Editace obsahu

V případě, že textové informace do systému zadávají a upravují uživatelé se znalostí HTML a požadují větší volnost při tvorbě, je možné editor textu vůbec nevytvářet a nechat uživatele zadávat přímo HTML kód. Jelikož se tato situace vyskytuje zřídka, používají se WYSIWYG editory, které umožňují zadávaný text snadno formátovat. Přímě během tvorby má autor přehled o tom, jak bude výsledný článek vypadat. Používají se i jednodušší editory, které neformátují text do výsledné podoby během jeho tvorby, ale vkládají do textu místo uživatele formátovací značky. Některé z nich mohou být podobné HTML, jiné zase mohou zastupovat komplexnější formátovací styly. U těchto editorů pak je obvykle možnost zobrazit si náhled výsledného textu před jeho finálním odesláním a opravit případné nedostatky.

¹ WYSIWYG = what you see is what you get = „co vidíš, to dostaneš“.

² Jejich srovnání nalezneme na URL: < <http://geniisoft.com/showcase.nsf/WebEditors>>

6.3 Správa uživatelů

Pokud si nepřejeme, aby nám kdokoli měnil obsah celého webu, je důležité vymezit, co který uživatel může a co ne. Na druhou stranu existují i systémy, kde je editace kýmkoli žádoucí (např.: Wikipedie). U firemní prezentace, magazínu nebo na osobním blogu to však zajisté žádoucí není.

Správou uživatelů vymezujeme práva jednotlivým uživatelům v systému. Toto nastavení se může vztahovat buď na jednotlivé uživatele, nebo na skupiny uživatelů. U méně rozsáhlých projektů lze preferovat nastavení jednotlivých uživatelů, u projektů s řádově stovkami až tisíci uživateli by něco takového bylo neúnosné.

Další možnost určení přístupových práv je *vymezení práv* v redakčním systému, popřípadě v databázi. Nastavení práv v databázi je již složitější, a při dobře zabezpečeném redakčním systému není nutné, pokud jsou ale na bezpečnost kladeny zvýšené nároky, tak toto nastavení bezpečnosti jistě neuškodí.

6.4 Bezpečnost

Bezpečnost je bezesporu jednou z velmi důležitých vlastností redakčního systému. Není možné uchovávat citlivá data v systému se zanedbaným zabezpečením. Naopak je zbytečné řešit extrémním způsobem bezpečnost u neveřejného redakčního systému, který slouží k usnadnění komunikace uvnitř menší firmy a nejsou v něm žádná citlivá data.

Jedná o velmi rozsáhlý a komplikovaný obor informatiky, proto není možné se jí v této práci věnovat detailně.

Pokud je po redakčním systému požadováno důkladné zabezpečení, je potřeba mít tento fakt na mysli při celém procesu jeho vytváření. Způsobů, jak systém napadnout, je nespočet a i při maximální snaze zabezpečení, nebude systém s velkou pravděpodobností neprůstřelný. Dnešní běžně používané metody a technologie téměř vždy počítají s jistými předpoklady o útočnickovi a nedají se označit za absolutně bezpečné. Ať už se jedná o zabezpečení dat na serveru, jejich dostupnost skrze redakční systém nebo přenos dat skrze distribuční médium, všude mohou nastat problémy. Například u přenosu dat je možné dosáhnout *nepodmíněné bezpečnosti* (zabezpečení, které lze prokázat matematicky za neprolomitelné za všech okolností) pomocí Vermanovi šifry a kvantové kryptografie. Toto řešení má ovšem své nedostatky a pro běžný provoz je nepoužitelné – už jen nutnost spojení komunikujících stran souvislým kabelem vylučuje použití v libovolné počítačové síti o více než dvou počítačích.

Bezpečnost redakčního systému tedy nikdy nebude ideální. Jde pouze o to, jak obtížné bude ji prolomit. Pokud se vrátíme zpět k reálným podmínkám, a nebudeme předpokládat, že má útočník k dispozici nekonečný výpočetní výkon, spokojíme se serverem nepřístupným nepovolaným osobám,

správně nastavenými a aktualizovanými systémy (operační systém, webový server, klient, redakční systém), uvědomělým uživatelem a šifrovanou komunikací.

V redakčním systému můžeme uvažovat o ošetření všech dat zadaných uživateli systému, dostupnosti částí systému určeným pouze pro správce systému nebo jiné uživatele s přístupovými právy do některých těchto částí, zabezpečení dat v databázi a souborů.

Praktické příklady zabezpečení jsou např. obrana proti SQL Injection¹ nebo nastavení práv uživatele, přes kterého se přistupuje do databáze – redakční systém jen málokdy musí mazat tabulky či databáze.

6.5 Uživatelské rozhraní

Uživatelské rozhraní zajišťuje komunikaci člověka s redakčním systémem. Na jeho kvalitě může záviset úspěch celé internetové prezentace, případně magazínu. V případě nepovedeného a nepřehledného rozhraní mohou uživatelé web přestat navštěvovat. Více neznamena lépe – mnohdy to platí i o uživatelských rozhraních. Složitě rozhraní sice může usnadnit ovládání redakčního systému, ale snadno se může stát přeplácaným a nepřehledným. Příkladem takového přeplácaného webu je například stránka *www.viaarena.com*, kde jsou nabídky skryty mezi nespočtem reklam a orientace na stránce je velmi obtížná.

Jednoduchý design nenabízí všechny možné volby na jednom místě, zato je mnohem přehlednější. Extrémním příkladem v tomto směru je *www.google.com*, kde jsou jen nutné položky a svou funkčnost plní prvotřídně.

Kromě množství ovládacích prvků na stránce je důležité i jejich rozmístění. Navigační prvky se běžně umísťují nahoru nebo doleva. Méně časté je jejich umístění napravo, ale není problém na takové rozložení narazit. Často se takto umístěná navigace objevuje na blogu. V dolní části stránky by ji hledal asi málokdo, i když i tato varianta je možná. Nejedná se však o jedinou navigaci na stránce. Využívá se pouze v případě, že se ve spodní části stránky objeví stejná nabídka jako nahoře, aby nebylo nutné se vracet po dočtení delšího článku zpět na jeho začátek.

Samotné (X)HTML sice neoplývá mnoho možnostmi zpříjemnění uživatelského rozhraní, ale tyto nedostatky snadno dožene JavaScript nebo Flash.

JavaScript je interpretovaný klientský programovací jazyk, který se zapisuje přímo do HTML kódu. Obvykle jím jsou ovládány grafické prvky stránky nebo tvořeny animace.

Flash je počín společnosti Adobe. Jedná se o vektorový program, využívaný pro tvorbu animací, prezentací a her. Hojně se používá pro zobrazování reklamy a díky tomu upadl v nelibost mnoha uživatelů, kteří mají v prohlížeči zakázané zobrazování Flash animací.

¹ SQL Injection je technika napadení webových stránek, při které se do neošetřeného vstupu vloží kód se záměrem poškodit nebo ovládnout databázi

6.6 SEO

Zkratka SEO znamená *Search Engine Optimization*, česky optimalizace pro vyhledávače. Nachází-li se na některém webu dokonalý obsah, ale nikdo jej nenajde, je to vcelku k ničemu. Na tyto případy se zaměřuje právě SEO. Jedná se o techniky, pomocí kterých by měl web získat co nejlepší pozice ve vyhledávačích a tím i získat co největší návštěvnost. Tato optimalizace se týká fulltextových vyhledávačů, neboť u těch katalogových stačí se zaregistrovat a vhodně zvolit klíčová slova.

Nejvýznamnějším fulltextovým vyhledávačem je *Google*, který má celosvětově přes 80% podíl na trhu [19]. Na českém trhu pak má největší podíl *Seznam*, následovaný druhým, třetím a čtvrtým Googlem v různých jazykových mutacích [20].

Fulltextové vyhledávače využívají roboty, kteří neustále prochází celý web a indexují si stránky. Chování těchto robotů je ovlivnitelné a právě na aspekty, které roboti berou v úvahu, se zaměřuje SEO.

Mezi běžné praktiky patří vhodné používání (X)HTML značek, klíčových slov, souboru robots.txt, krátká a neměnná URL adresa atd. Některé jmenované praktiky se shodují s pravidly pro tvorbu přístupného webu. Nalézt je můžeme například na adrese <http://www.pravidla-pristupnosti.cz/>. Někdy tyto optimalizace zacházejí až příliš daleko a pak se již dá hovořit o jejich zneužívání.

6.7 Moduly

Stane-li se a redakční systém neobsahuje některou pro nás důležitou funkci, nemusí to nutně znamenat, že je pro nás nevhodný. Řešením mohou být právě rozšiřující moduly, které do systému přidávají nové funkce. Na první pohled primitivní a nezajímavý redakční systém se základní funkcionalitou se může díky modulům proměnit v mocný nástroj vyhovující spoustě uživatelů a naopak komplexní redakční systém bez jedné důležité funkce a nerozšiřitelný pomocí modulů vypadne ze hry.

Funkcí, které mohou moduly přidávat je nespočet – komentáře k článkům, vyhledávání, statistiky, ankety, novinky, pokročilá správa uživatelů, vkládání obrázků, obrázková galerie a mnohé další.

7 Příklady redakčních systémů přílohy

Nyní se podíváme na některé běžně používané redakční systémy. Následující výběr se řídí především výsledky hlasování o nejlepší Open Source CMS za rok 2008. Soutěž Open Source CMS Awards každoročně probíhá na serveru *www.packtpub.com* a má několik kategorií: Nejlepší PHP Open Source CMS, Nejlepší ne-PHP Open Source CMS, Nejslibnější Open Source CMS a nakonec Celkový vítěz. Nominace do soutěže probíhá prostřednictvím hlasování uživatelů. Těmi je vybráno 5 nejlepších systémů v každé kategorii. Poté komise sestavená z odborníků v oblasti Open Source a CMS provede výběr finálové trojce. Nyní přijde na řadu opět hlasování uživatelů. Na všechny finalisty čekají finanční odměny.

Zaměříme se na vítěze kategorie PHP Open Source CMS a také na celkového vítěze *Drupal*. Dále na systémy *Joomla* a *CMS Made Simple*, které se umístily na druhém místě v kategorii PHP Open Source CMS (v dané kategorii byla dvě druhá místa). Naši pozornost věnujeme i vítězi kategorie Nejlepší ne-PHP Open Source CMS, jímž se stal systém *Plone*. [21].

Vzhledem ke složitosti a rozsahu těchto systémů se podíváme i na systémy z druhého konce spektra – na minimalistické *LoveCMS* a *NanoCMS*.

Ukázky těchto redakčních systémů jsou v příloze 1.

7.1 Drupal

Drupal vytvořil holandský student Dries Buytaert. Jedná se o open source redakční systém, tedy o volně dostupný software, který v základní instalaci obsahuje moduly pro tvorbu článků, statických stránek, diskusních fór, blogů, přidávání komentářů k obsahu a mnoho dalších. Jeho funkcionalitu lze rozšířit pomocí stažených modulů. [22].

Jedná se o dnes nejpopulárnější redakční systém, jehož vývoj se již dostává za hranice běžných redakčních systémů. Má nepřehledné množství možností, ale to sebou také nese nevýhody v jeho složitosti.

Vlastnosti:

- jazyk – PHP
- podporované databáze – MySQL a PostgreSQL
- vysoká flexibilita
- propracovaný systém autorizace uživatelů
- globalizace a lokalizace
- monitorování a logování aktivity
- nezvyklá terminologie

- absence OOP
- málo graficky povedených výchozích skinů

7.2 Joomla

Redakční systém Joomla vznikl v roce 2005 při rozdělení vývojářů systému Mambo. „Joomla je slovo pocházející z arabštiny a znamená to buď „shluk slov, které dávají smysl“ nebo „dohromady.“ Slovo se často používá ve významu „Součet“ nebo „Suma“ tak, jak je znáte z různých obchodních tabulek apod. Na domovské stránce projektu sice najdete, že Joomla je slovo pocházející ze svahilštiny, ale není to tak úplně pravda. Svahilština je totiž dialektem arabštiny, když 95% slov je plně arabských.“ [23].

Joomla stejně jako drupal staví na modularitě a snadném rozšíření pomocí modulů. Na rozdíl od Drupalu je ovšem zaměřen víc na běžné uživatele v podobě přívětivějšího a snadnějšího ovládní. Nemá tak propracované přidělování uživatelských oprávnění a neumí pomocí jedné instalace obsluhovat více instancí (pro každý web je potřeba samostatná instalace).

Vlastnosti systému:

- jazyk – PHP
- databáze – MySQL
- instalační skript
- licence – GPL
- user management
- WYISWYG editor
- podpora SSL

7.3 Plone

Na rozdíl od předchozích systémů Plone nevyužívá ke svému běhu PHP. Používá aplikační server *Zope*, který je napsaný v jazyce *Python*.

Práce na projektu Plone započaly již v roce 1999 a o dva roky později byla vydána první verze. V roce 2004, dva měsíce po vydání druhé verze, byla založena nadace Plone Foundation, která má na starosti vývoj, marketing a ochranu Plone. Tato nadace má vlastnické právo ke zdrojovým souborům, obchodním značkám a doménám. Přesto je však projekt pořád Open Source.

Název aplikačního serveru *Zope* znamená „*Z Object Publishing Environment*“ = Prostředí pro publikaci objektů. Běžný webový server interpretuje adresu `http://www.example.com/tomas/kniha` jako soubor `kniha` v adresáři `tomas`. Na rozdíl tomu *Zope* zobrazí objekt `kniha`, který je uvnitř objektu `tomas`. Objekt `tomas` může mít například nadefinovanou HTML hlavičku, patičku a zelený text a objekt `kniha` zdědí tyto vlastnosti. Bude mít nastaven rovněž zelený text. Pokud přesuneme objekt

kniha do objektu petr, který má modrý text, bude kniha také vypsána modře. Tyto objekty jsou obvykle uloženy v objektové databázi ZODB¹. [24].

Snad největší přednost tohoto systému je v snadné rozšiřitelnosti, která plyne z jeho objektově orientovaného návrhu, a také v jeho bezpečnosti – v *National Vulnerability Database*² má Plone řádově méně záznamů než Drupal nebo Joomla. Díky multiplatformnosti pythonu a Zope je možné Plone provozovat na všech běžně rozšířených platformách. V praxi tento systém používá například CIA.

Vlastnosti systému:

- jazyk – Python
- databáze – ZODB
- licence – GPL
- user management
- verzování a evidence historie dokumentů

7.4 CMS Made Simle

CMS Made Simple je redakční systém s intuitivním ovládáním a vysokou přizpůsobitelností. Hlavní zaměření tohoto CMS je na projekty menšího rozsahu (desítky až stovky stránek), typicky prezentace korporace, webová stránka představující určitou skupinu nebo organizaci apod. Sami tvůrci přiznávají, že jejich systém není nejlepší pro tvorbu portálů a blogů. [25].

Vlastnosti systému:

- jazyk – PHP
- databáze – MySQL
- instalační skript
- licence – GPL
- user management
- WYISWYG editor

7.5 LoveCMS

LoveCMS je velmi jednoduchý redakční systém běžící na PHP a využívající MySQL databázi. Hlavním mottem LoveCMS je „být jednoduchý a efektivní“. Základní instalace obsahuje šablonovací systém a je možné ji rozšiřovat moduly. Má bohužel velmi špatnou, respektive žádnou dokumentaci.

¹ZODB je zkratka pro Zope Object Databáze a je to objektově orientovaná databáze pro transparentní a trvalé ukládání objektů jazyka Python.

²Odkaz na stránku *National Vulnerability Database*: <<http://web.nvd.nist.gov>>

Přestože je tento systém velmi jednoduchý, dá se v něm snadno vytvořit nenáročná prezentace. Bezpečnost systému ovšem není ideální.¹

Vlastnosti systému:

- jazyk – PHP
- databáze – MySQL
- instalační skript
- licence – BSD

7.6 NanoCMS

NanoCMS je představitel redakčního systému, který nepoužívá databázi. Jako své hlavní přednosti uvádí především snadné používání, instalaci pomocí pouhého zkopírování a minimální velikost (pouze několik desítek kilobytů). Oproti LoveCMS nabízí mnohem více šablon a také dokumentaci.

Vlastnosti systému:

- Jazyk – PHP
- Databáze – pouze soubory
- Nemá instalační skript
- licence – GPL

¹ Popis jedné neopravené kritické chyby najdeme na URL: <<http://secunia.com/advisories/31389>>

8 Implementační část

V této kapitole se budeme věnovat tvorbě vlastního redakčního systému. Na začátku se zaměříme na požadavky, které budeme na redakční systém mít. Provedeme návrh realizace požadovaného systému a nakonec se budeme věnovat jeho implementaci.

8.1 Požadavky

Redakční systém, který budeme implementovat, nebude v plánovaném rozsahu reálně použit. Cílem tvorby tohoto redakčního systému je seznámit se s problematikou redakčních systémů. Zjistit jaká úskalí se mohou při jeho tvorbě objevit. Po provedení implementace zhodnotíme, zda byl vybraný postup správný, zda by bylo možné redakční systém skutečně nasadit do praxe a jaké úpravy by případně vyžadoval.

Požadujeme systém, který staví na objektově orientovaném programování, jelikož všechny běžné programovací jazyky dnes OOP podporují. Díky tomu bude systém snadné rozšířit o další vlastnosti.

Jedny z hlavních požadavků budou *jednoduchost a přehlednost*. Redakční systém bude sloužit především k testovacím účelům a jako odrazový můstek do této oblasti. Ne jen orientace v uživatelském rozhraní by tudíž měla být snadná, nýbrž i orientace ve zdrojových kódech. Jestliže bude později nutné některou jeho část vyměnit nebo upravit, nebude obtížné se v systému zorientovat a požadované úkony provést.

Z hlediska nasazení by měl být co *nejuniverzálnější*. Univerzálnost je požadována jak po technické stránce, tak po obsahové. Potíže by mu neměl dělat přechod na jiný operační systém, ani by neměl být úzce zaměřený na konkrétní druh obsahu.

Další vyžadovanou vlastností je *modularita*. Tou se zajistí ještě snadnější rozšiřitelnost o nové funkce a bude jej možné snadno přizpůsobit pro konkrétní nasazení.

Redakční systém musí být *použitelný i pro běžné uživatele*, kteří nejsou schopní upravovat zdrojové soubory a neznají HTML či CSS.

Zpříjemnit uživatelů práci s redakčním systémem by měla technologie AJAX.

8.2 Návrh

Podle výše uvedených požadavků nyní vybereme vhodné technologie a navrheme, jak bude redakční systém fungovat.

Požadavek na objektově orientované programování splňují všechny zmíněné programovací jazyky/platformy, které splňují rovněž další požadavek – univerzálnost a nezávislost na operačním

systému. I ASP.NET, který byl původně zaměřený pouze na operační systém Microsoft Windows, je nyní možné provozovat na jiných operačních systémech. Vzhledem k požadavku na univerzálnost tedy zvolíme jako hlavní parametr pro výběr platformy její snadnou dostupnost u webhostingových společností. S tímto výběrem nám pomůže server *www.webhostingy.cz*, který vede databázi u nás dostupných webhostingů. Z celkových 166 webhostingových programů celých 147 podporuje *PHP*, ostatní zde uvedené platformy se pohybují přibližně na dvaceti webhostingových programech. Tímto je tedy rozhodnuta cílová platforma – neboť v případě volby jiné než *PHP* by byly značně omezeny možnosti umístění redakčního systému na internet.

Další důležitou otázkou, kterou je nutné rozhodnout, je kam ukládat data. Jelikož není dopředu známé cílové použití našeho redakčního systému, není možné zvolit soubory pro ukládání článků. Obrázky budou uloženy v *souborech* (databáze by se jejich ukládáním značně rozrostla a vzhledem k možným omezením ze strany webhostingu by redakční systém mohl brzy narazit). Výběr konkrétní databáze pak opět rozhodlo porovnání na *www.hostingy.cz*. Vítězství připadlo *MySQL*.

Velkou výhodou těchto dvou zvolených řešení bude jejich snadná dostupnost. Je možné sehnat kompletní balík *Apache* serveru, *MySQL* databáze a *PHP* programovacího jazyka. Takové balíky jsou označovány jak *LAMP* případně *WAMP* – *Linux Apache MySQL a PHP*, případně stejné nástroje pro operační systém Windows.

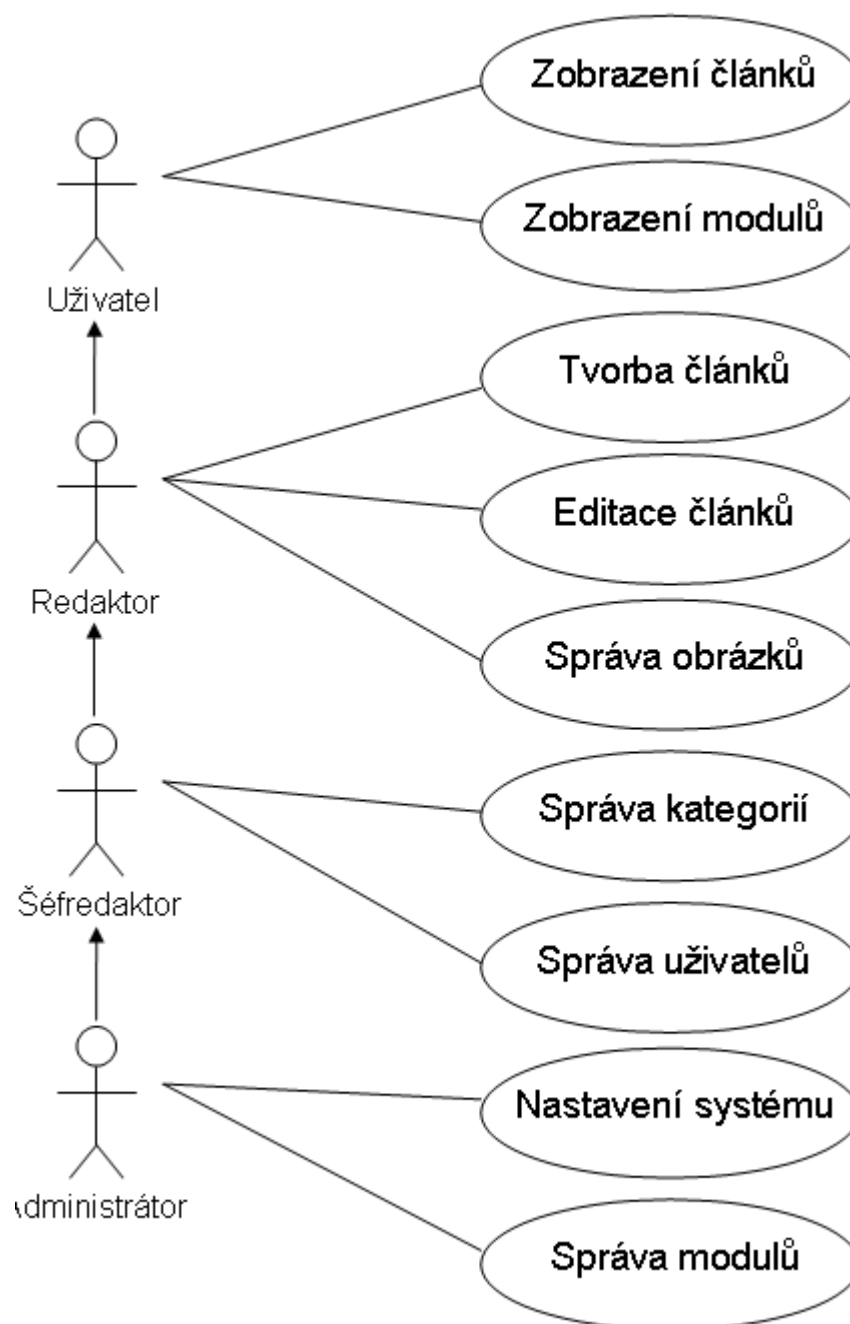
Technické aspekty implementovaného redakčního systému tím máme vyřešeny. Nyní se budeme věnovat aspektům uživatelským. Pro obecné použití bude nutné vytvořit systém, který nebude omezen na jednoho či několik málo autorů. Redakční systém bude obsahovat uživatelské skupiny a příslušnost uživatele do některé z těchto skupin určí, jaké části systému mu budou přístupné. Tyto role nám nejlépe přiblíží diagram případů použití (viz Obrázek 2).

Systém bude vést evidenci uživatelů. Ti sice nebudou mít v plánovaném systému žádná zvláštní oprávnění, ale pro usnadnění dalšího vývoje systému a také pro využití v modulech budou mít uživatelé možnost se zaregistrovat. O uživateli se budou uchovávat základní informace a jeho příslušnost ke skupině. U uživatele posléze postačí změnit příslušnost ke skupině, čímž se z něho stane redaktor.

Redaktor už bude mít v redakčním systému možnost vytvářet a editovat články. Aby nebyl redaktor omezen pouze na vkládání holého textu, bude moci do redakčního systému nahrávat obrázky a následně je používat v člancích.

Za provozní záležitosti redakčního systému bude odpovídat šéfredaktor. Tomu oproti redaktorovi přibude možnost spravovat uživatele, aby mohl určit členy redakce. Bude rovněž spravovat obsahové sekce. Články tedy budou podobně jako uživatelé kategorizovány do obsahových sekcí.

Veškeré funkce, které budou v redakčním systému implementovány, budou přístupné Administrátorovi. Ten bude oproti šéfredaktorovi spravovat redakční systém spíše z technického hlediska – instalovat a odebírat moduly a upravovat nastavení celého systému.



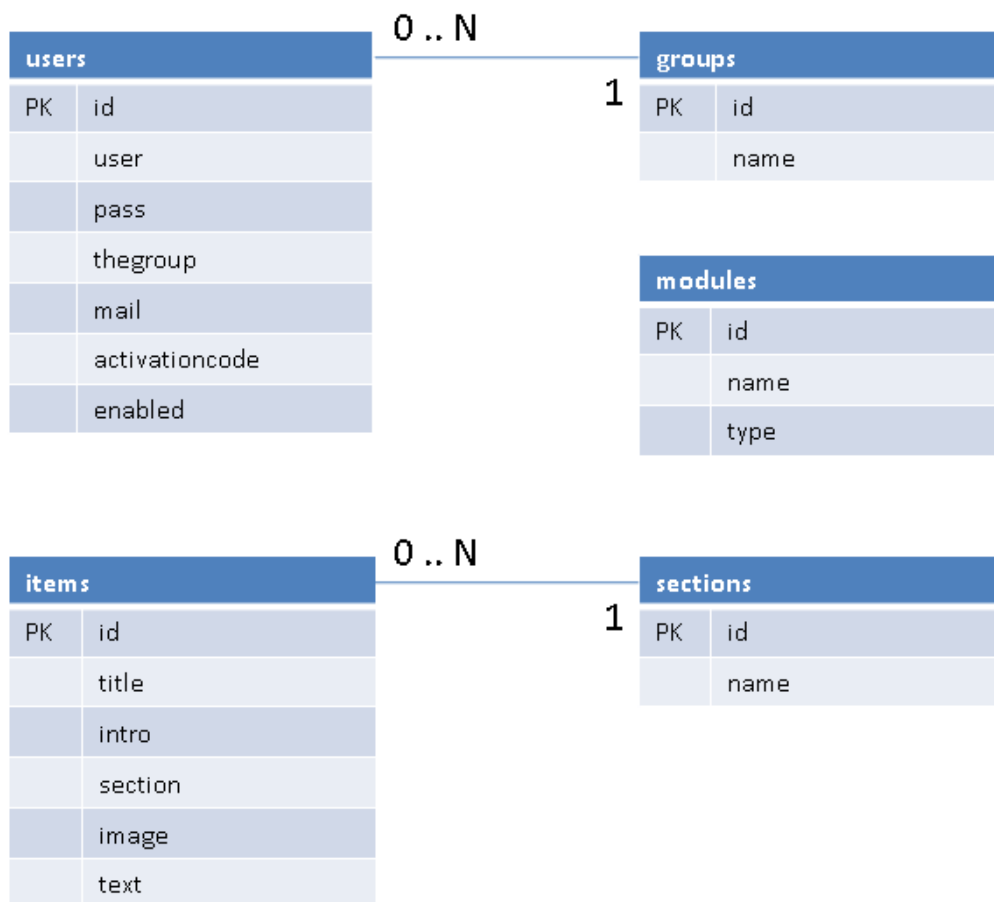
Obrázek 2: Diagram případů použití.

Požadavek na systém, který vyplývá z jeho univerzálnosti a jednoduchosti, je i snadná editace textu. Bude nutné umožnit i neznalým uživatelům formátovat obsah. Řešením požadavku se stane WYSIWYG editor.

Bylo již řečeno, že redakční systém bude muset počítat s rozšířením pomocí modulů. Moduly bude nutné do systému přidávat bez zásahů do zdrojových kódů. Jejich správa tak musí probíhat přes webové rozhraní. Pro moduly budou v klasickém rozložení stránky rezervovány oblasti, kde bude možné moduly umístit. Moduly se budou lišit podle svého plánovaného umístění – buď se bude jednat pouze o postraní doplňky sloužící k doplnění článků, nebo rozsáhlejší rozšíření, které potřebují

více prostoru. Tyto typy také bude možné zkombinovat. Modul tím bude moci ještě více ovlivnit funkčnost redakčního systému.

Podle dosavadního návrhu je sestaven následující ER diagram, znázorňující relační schéma databáze (viz Obrázek 3).



Obrázek 3: ER diagram.

Schéma databáze není nikterak složité (což splňuje požadavek jednoduchosti systému). Redakční systém nabízí pouze základní funkcionalitu, o jeho rozšíření se starají především moduly. Po vytvoření a integraci složitějších modulů se může schéma databáze značně zkomplikovat. Pro základní funkcionalitu redakčního systému a seznámení se s problematikou je však výše uvedené schéma dostatečné.

Implementace technologie AJAX bude použita na doplňkové funkce. Nabízí se tak například modul pro vyhledávání článků, který bude uživateli nabízet nalezené články již při zapisování hledaných výrazů. Tento modul zároveň ověří, zda je implementovaný redakční systém schopný integrovat moduly, které vyžadují přístup do databáze a použití JavaScriptu.

Další možnosti využití AJAXu jsou například sledování stavu při nahrávání obrázků, zobrazení náhledu článku a automatické doplnění psaného textu. Případné další použití vyplyne při implementaci.

8.3 Popis implementace

Poslední fází vývoje redakčního systému je implementace. Jelikož není možné detailně popsat implementaci každé funkcionality, pokusím se popsat nejdůležitější části.

Implementace probíhala na počítači s operačním systémem Microsoft Windows a jako server byl zvolen volně dostupný balík aplikací XAMPP. Důvodů pro tuto volbu bylo hned několik: 1. Jedná se o multiplatformní řešení. 2. Nepotřebuje instalaci, takže je možné jej spolu s redakčním systémem vypálit na CD a spustit kdekoliv jinde. 3. Měl jsem s ním dobré zkušenosti z dřívějšíka. Obsahuje všechny součásti zvolené v předcházející podkapitole a nic dalšího tedy není potřeba.

Jako první proběhlo vytvoření databáze podle ER diagramu, které nezáviselo na existenci kterékoliv další součásti systému, naopak samo bylo potřebné pro další práci.

Dalším krokem bylo promyšlení si struktury uložení zdrojových souborů v kořenovém adresáři. Přímo v něm musí být umístěn soubor `index.php`, dále zde budou soubory obsahující definice CSS stylů a veřejně přístupné části redakčního systému. Ve vlastní složce budou umístěny všechny ovládací prvky administrátorského rozhraní. Ve zvláštních složkách budou uloženy obrázky, soubory obsahující třídy, které zajišťují chod systému, a moduly.

Adresářová struktura je tedy následující:

- `/admin` - Administrátorské rozhraní
- `/images` - Obrázky
- `/includes` - Aplikační logika
- `/install` - Instalační skript
- `/modules` - Moduly
- `/` - Uživatelské rozhraní

Následně byla vytvořena aplikační logika, uživatelské rozhraní, administrátorské rozhraní a nakonec moduly. Nyní si podrobněji jednotlivé části popíšeme.

8.3.1 Aplikační logika

Základní třídou celého redakčního systému je třída *SystemComponent*. Její funkce není nikterak složitá. Poskytuje pouze ostatním třídám přístup k nastavením systému. Zpočátku bylo nastavení systému (jako např. adresa stránky či jméno databáze) uloženo přímo v tomto souboru. V pozdějších fázích se toto rozhodnutí však ukázalo jako neperspektivní a bylo potřeba nastavení přesunout někam, kde bude možné jej snadno modifikovat. V úvahu přicházely dvě možnosti. Tabulka v databázi nebo soubor. U databáze by byl problém s uložením vícera hodnot různého typu. U souboru by se však musel vymýšlet jeho formát, případně způsob načtení dat.

Dilema vyřešila posloupnost několika PHP funkcí (respektive dvojic funkcí). První z nich jsou *serialize()* a *unserialize()*. Ty umožní převést libovolnou proměnou na textový řetězec vhodný k uložení se zachováním typu a zase zpět.

Další dvojice funkcí vhodných pro daný účel je *file_get_contents()* a *file_put_contents()*, která zajišťuje snadné čtení a zápis hodnot do a ze souboru. Není problém mít nastavení v poli, převést jej na řetězec, uložit do souboru a obráceným postupem snadno získat původní pole. Toto řešení je pro uložení nastavení zvláště vhodné a jednoduché.

Použitím dvojice funkcí *base64_encode()* a *base64_decode()* se učiní obsah souboru zaručeně nečitelným a bez znalosti tohoto postupu nebude možné zjistit přístupové heslo do databáze. Použita však může být libovolná jiná dvojice funkcí, která zakóduje a poté zase dekoduje řetězec s nastavením.

Třída *SystemComponent* začala postrádat svůj původní smysl, ale je možné ji využít při dalších rozšířeních. Jelikož ke změně došlo až v pozdní fázi vývoje, bylo by její odstranění pracné.

Další třídou je *DbConnector*, který propojuje datovou vrstvu s aplikační logikou. Připojuje se k *MySQL* serveru a získává z něj data. *MySQL* server by bylo možné nahradit jiným SQL serverem, který by byl kompatibilní s dotazy *MySQL*.

Třída *sentry* kontroluje přihlášené uživatele, případně je přihlašuje. Je bezprostředně navázaná na správu uživatelských účtů. Umožňuje snadné zamezení přístupu na stránku, pokud uživatel není registrovaný a nepatří do požadované uživatelské skupiny.

Kontrolu validních vstupů ve formulářích provádí třída *Validator*.

Prezentační logiku představuje třída *insert*. Ta vytváří strukturu celé výsledné stránky a vkládá moduly. Na výsledné stránce zůstává prostor pouze pro její hlavní obsah. V závislosti na oprávnění uživatele zobrazuje položky v menu pro správu redakčního systému.

8.3.2 Uživatelské rozhraní

Vhodná aplikační logika redakčního systému usnadňuje a urychluje tvorbu souborů uživatelského rozhraní. Spočívá pouze ve volání tříd *sentry*, *insert* a vytvoření obsahu hlavní části stránky. Jedinou výjimkou je *index.php*. Ten, kromě vytváření obsahu hlavní části stránky, obsluhuje i přihlašování a odhlašování uživatelů. Obsah hlavní části stránky je vytvářen třídou *insert* a po doplnění modulů do systému zobrazí buď obsah některého nastaveného modulu, nebo klasický výpis posledních článků.

8.3.3 Administrátorské rozhraní

Ani u administrátorského rozhraní nenastaly žádné problémy. Vytváření jednotlivých jeho částí je shodné s vytvářením uživatelského rozhraní, jen je díky třídě *sentry* umožněn přístup pouze oprávněným uživatelům.

Pro tvorbu článků byl zvolen volně dostupný WYSIWYG editor *TinyMCE*,¹ jehož začlenění proběhlo lehce. Má dva módy (základní a pokročilý) lišící se počtem ovládacích prvků, které lze jednoduše přepínat.

Dalším integrovaným prvkem do systému je *ImageEditor*² sloužící k editaci obrázků. Ke své činnosti využívá technologie AJAX a umožňuje snadnou editaci obrázků. Oba dva editory jsou umístěny ve vlastních složkách v kořenovém adresáři.

Jeho integrace byla bezproblémová, byly však provedeny dvě drobné úpravy. *ImageEditor* pracoval pouze ve svých dočasných složkách a nebyl schopný po provedení změn přepsat původní obrázek. Proto byla, podle již hotových funkcí, vytvořena funkce pro zkopírování upraveného obrázku na místo původního. Druhou úpravou bylo přidání odkazu pro návrat zpět do redakčního systému. V případě plánovaného použití redakčního systému v reálném provozu by bylo nutné zajistit mazání dočasných souborů editoru. Ten totiž při úpravě každého obrázku vytvoří jeho kopie, které v systému zůstanou.

Při volbě rozložení stránky a umístění modulů je použita technologie AJAX pro okamžité umístění modulů na pozice kam patří. Pomocí AJAXU je rovněž zobrazován průběh nahrávání obrázků na server.

8.3.4 Moduly

Při tvorbě již prvního modulu se objevily problémy. V uživatelském rozhraní bylo vše v pořádku, ale v administrátorském modul chyběl. Příčinou byly rozdílné relativní cesty, dané umístěním administrátorského rozhraní do vlastní složky. Tento problém byl vyřešen vložením souboru *root.php*, který obsahuje pouze proměnou *\$ROOT* a v ní řetězec pro vytvoření relativní adresy do kořenového adresáře. V kořenovém adresáři je tedy tento řetězec prázdný, v podadresáři první úrovně obsahuje řetězec „../“ pro posunutí relativní cesty o adresář výš a spolu s hlouběji zanořenou strukturou adresářů se jeho délka násobí. V každém souboru je soubor *root.php* vložen hned jako první a podle něj pak není problém se navigovat v adresářové struktuře redakčního systému.

Moduly jsou tvořeny následujícím způsobem: ve složce *modules* má každý modul vlastní složku, která je shodná s jeho názvem, v ní je soubor *root.php*, *module.php* a případně další volitelné soubory. Moduly jsou trojího typu: pouze postraní modul (1), modul pro zobrazení pouze na hlavní stránce (2) a modul s oběma typy zobrazení (3). Soubor *module.php* obsahuje třídu, jejíž název je shodný s názvem modulu a předem dané metody. Jedná se o:

- *index()*
- *minor()*
- *install()*

¹ WYSIWYG editor TinyMCE je dostupný na WWW: <<http://tinymce.moxiecode.com/>>

² ImageEditor je dostupný na WWW: <<http://www.ajaxprogrammer.com/?p=9>>

- *uinstall()*

Metoda *index()* je volána při zobrazení obsahu modulů 2. nebo 3. typu do hlavní obsahové části. Metoda *minor()* je volána při zobrazení modulů typu 1 a 3 na postraní pozici. Metoda *install()* musí obsahovat SQL dotaz, který vloží jméno a typ modulu do databáze do tabulky *Modules*. *Uinstall()* slouží k odstranění modulu z databáze a navrácení případných dalších změn, které se provedly při instalaci nebo provozu modulu. Tyto dvě metody jsou volány při instalaci nebo odinstalaci modulů v administrátorském rozhraní.

Implementované moduly jsou „*google*“ pro fulltextové vyhledávání v celém redakčním systému pomocí vyhledávače google, „*Novinky*“ pro zobrazování stručných aktualit a „*suggest*“ pro snadné vyhledávání článků s pomocí AJAXu.

9 Závěr

Cílem této bakalářské práce bylo seznámit čtenáře s problematikou redakčních systémů, jejich možnostmi a požadavky, které na ně uživatelé mají, a s technologií AJAX. Po nastudování a popisu různých technologií jsem se rozhodl pro implementaci redakčního systému v jazyce PHP s využitím databáze MySQL. Tento výběr zaručuje snadnou přenositelnost mezi platformami a možnost umístění na většinu webhostingových služeb. Při implementaci redakčního systému se neobjevily žádné závažné potíže, které by znemožnily provést implementaci navrženého řešení.

Implementovaný redakční systém splňuje požadavky, které na něho byly kladené. Je velmi snadno rozšiřitelný, přehledný a jednoduchý. Výsledný HTML kód generovaný redakčním systémem je HTML 4.01 STRICT validní. Při změně umístění souborů redakčního systému na serveru, kde byl vytvářen, nedochází k žádným potížím. Při umístění na webhosting ovšem některé části systému nefungovali – například zobrazení článků nebo loga. Celkové chování redakčního systému bude prověřeno a případné úpravy popsány v článku na ukázkové veřejné verzi tohoto redakčního systému dostupné na xmacho05.medlov.eu nejpozději do 22. 5. 2009.

V redakčním systému bylo vyžadováno použití technologie AJAX. Tato technologie byla implementována na několika místech systému. Ze získané zkušenosti s implementací dané technologie nemám moc dobrý pocit. Vzhledem k omezením plynoucím z vlastností AJAXu (zmiňovaných v kapitole 5.2) nebyl AJAX použit pro manipulaci s obsahovými částmi redakčního systému, ale pouze pro doplňkové funkce. Množství času, které jsem musel věnovat implementaci těchto částí, ani zdaleka neodpovídá jejich přínosu. Pokud by byl tento čas věnován vývoji nových modulů nebo dalších funkcí systému, bylo by to na redakčním systému poznat mnohem výraznějším způsobem. Tato neefektivnost při tvorbě může být způsobena použitím dvou programovacích a jednoho značkovacího jazyka. Rychlost vývoje v porovnání s programováním v samotném PHP či Pythonu je několikanásobně nižší. Přínos AJAXu pro redakční systém menšího rozsahu není velký. Asi nejužitečnější z implementovaných funkcí AJAXu se mi jeví automatické našeptávání při psaní. Jeho implementace nebyla nikterak složitá. Jedná se o užitečnou funkci a podobného výsledku by bez AJAXu nebylo možné dosáhnout.

Jako perspektivní se mi AJAX jeví u komplexních webových aplikací, za nimiž stojí velké firmy, které na uživatelské pohodlí (umožněné AJAXem) lákají uživatele. Příklad vhodného užití je například Gmail. U projektů malého a středního rozsahu nemusí být použití AJAXu vždy výhodné.

Další vývoj redakčního systému vidím především ve tvorbě dalších modulů jako například diskuze k článkům, ankety, statistiky zobrazení jednotlivých článků, diskusní fórum nebo uživatelské soutěže.

Literatura

- [1] *Redakční systém* [online]. Wikipedie, aktualizováno 20. 4. 2009 [cit. 2009-04-29]. Dostupné z WWW: <http://cs.wikipedia.org/w/index.php?title=System_pro_spravu_obsahu&oldid=3869189>
- [2] *Redakční systém WEBJump* [online]. WebJump, [cit. 2009-04-29]. Dostupné z WWW: <<http://redakcni-system.webjump.cz/>>
- [3] Lukáš, J.: *Co je to redakční systém?* [online]. Super svět, aktualizováno 5. 5. 2005 [cit. 2009-04-29]. Dostupné z WWW: <<http://www.supersvet.cz/view.php?navezclanku=co-je-to-redakcni-system&cislocclanku=2005050501>>
- [4] Duhon, B., Patel, J., Tucker, T.: *What is ECM?* [online]. AIIM, 2008 [cit. 2009-04-30]. Dostupné z WWW: <<http://www.aiim.org/What-is-ECM-Enterprise-Content-Management.aspx>>
- [5] *Web content management system* [online]. Wikipedie, aktualizováno 14. 5. 2009 [cit. 2009-04-30]. Dostupné z WWW: <http://en.wikipedia.org/w/index.php?title=Web_content_management_system&oldid=289858971>
- [6] KAMPPFMEYER, U.: *ECM Enterprise Content Management* [online]. Hamburg. 2006 : Project Consult, 2006 [cit. 2006-09-19]. Dostupný z WWW: <http://www.project-consult.net/Files/ECM_White%20Paper_kff_2006.pdf>. ISBN 10: 3-936534-09-8.
- [7] *Systém pro správu obsahu (CMS)* [online]. Symbio, 1999-2009 [cit. 2009-04-30]. Dostupné z WWW: <<http://www.symbio.cz/slovník/system-pro-spravu-obsahu-cms.html>>
- [8] Vich, O., Bittner, V.: *Enterprise Content Management* [online]. Content Management, 2002 aktualizováno 13. 11. 2002 [cit. 2009-04-30]. Dostupné z WWW: <www.asociace.biz/_storage/Enterprise-Content-Management.ppt>
- [9] *Redakční systém* [online]. Asociace.biz, 2005 [cit. 2009-04-30]. Dostupné z WWW: <<http://asociace.biz/redakcni-system.htm>>
- [10] *Web Content Management System* [online]. Asociace.biz, 2005 [cit. 2009-04-30]. Dostupné z WWW: <<http://asociace.biz/web-content-management.htm>>
- [11] Gottlieb, S.: *CMS Deployment Patterns* [online]. Content here, aktualizováno 11. 6. 2007 [cit. 2009-05-01]. Dostupné z WWW: <<http://www.contenthere.net/2007/06/cms-deployment-patterns.html>>
- [12] *What can PHP do?* [online]. PHP, 2001-2009, aktualizováno 15. 5. 2009 [cit. 2009-05-01]. Dostupné z WWW: <<http://cz.php.net/manual/en/intro-whatcando.php>>
- [13] *Python Programming Language -- Official Website* [online]. Python, 1999-2009 [cit. 2009-05-01]. Dostupné z WWW: <<http://www.python.org/>>

- [14] *Databáze* [online]. Wikipedie, aktualizováno 22. 2. 2009 [cit. 2009-05-01]. Dostupné z WWW: <<http://cs.wikipedia.org/w/index.php?title=Databáze&oldid=3652805>>
- [15] *Relační model* [online]. Wikipedie, aktualizováno 25. 1. 2009 [cit. 2009-05-01]. Dostupné z WWW: <http://cs.wikipedia.org/w/index.php?title=Relační_model&oldid=3545573>
- [16] *List of content management systems* [online]. Wikipedie, aktualizováno 14. 5. 2009 [cit. 2009-05-01]. Dostupné z WWW: <http://en.wikipedia.org/w/index.php?title=List_of_content_management_systems&oldid=289866661>
- [17] O'Reilly, T.: *What Is Web 2.0* [online]. O'Reilly, 2008, aktualizováno 30. 9. 2005 [cit. 2009-05-01]. Dostupné z WWW: <<http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html?page=5>>
- [18] O'Reilly, T.: *What Is Web 2.0* [online]. O'Reilly, 2008, aktualizováno 30. 9. 2005 [cit. 2009-05-01]. Dostupné z WWW: <<http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html>>
- [19] *Search Engine Market Share* [online]. Market Share, aktualizováno 17. 5. 2009 [cit. 2009-05-02]. Dostupné z WWW: <<http://marketshare.hitslink.com/search-engine-market-share.aspx?qprid=4>>
- [20] *Podíly vyhledávačů v referech* [online]. Cose.cz, 2009 [cit. 2009-05-02]. Dostupné z WWW: <<http://vyhledavace.chose.cz/>>
- [21] *Open Source CMS Award* [online]. Packt Publishing, 2009, aktualizováno 31. 10. 2008 [cit. 2009-05-03]. Dostupné z WWW: <<http://www.packtpub.com/award>>
- [22] *O systému Drupal* [online]. Drupal.cz, 2009 [cit. 2009-05-03]. Dostupné z WWW: <<http://www.drupal.cz/o-systemu-drupal>>
- [23] Vít, S.: *Co je to Joomla?* [online]. Joomlaportal.cz, 2004–2009, aktualizováno 15. 10. 2004 [cit. 2009-05-03]. Dostupné z WWW: <<http://www.joomlaportal.cz/content/view/25/43/>>
- [24] *Plone (software)* [online]. Wikipedia, aktualizováno 15. 4. 2009 [cit. 2009-05-03]. Dostupné z WWW: <[http://en.wikipedia.org/w/index.php?title=Plone_\(software\)&oldid=284082542](http://en.wikipedia.org/w/index.php?title=Plone_(software)&oldid=284082542)>
- [25] *Why CMS Made Simple?* [online]. CMS made simple, [cit. 2009-05-03]. Dostupné z WWW: <<http://www.cmsmadesimple.org/support/faq/>>

Seznam příloh

Příloha 1. Ukázky redakčních systémů

Příloha 2. DVD

Příloha 1. Ukázky redakčních systémů

The screenshot shows the homepage of Drupal.cz. At the top, there is a blue header with the Drupal logo and the text 'Drupal.cz'. Below the header is a navigation menu with links: O Drupalu, Demo, Blogy, Fórum, Stáhnout, Katalog, Webhosting, and Služby. The main content area is divided into several sections:

- Drupal je populární systém pro správu obsahu (Content Management System - CMS).** Snadná rozšiřitelnost pomocí modulů a mnoho možností, které poskytuje, z Drupalu dělá jeden z nejlepších redakčních systémů. Navíc je zdarma i se zdrojovými kódy. Toto jsou české Drupal stránky. Jsou určeny k podpoře uživatelů a diskusi o jeho používání. Below this text are links for 'Více o systému Drupal' and 'Poradna pro začátečníky', and a search bar with the text 'Hledat'.
- Programujeme pro Drupal: bezpečně** (Jakub Suchý, 6 Květen, 2009 - 07:50). The article text discusses the popularity of Drupal in the Czech Republic and the world, and the importance of security. It mentions that while Drupal is easy to use, it's important to follow security guidelines. A 'Číst dál' link is present.
- Z čeho je poskládaný Drupal?** (Ema, 23 Duben, 2009 - 10:43). The article text explains that Drupal is like Lego, easy to assemble. It mentions that the code is open and can be modified. A 'Číst dál' link is present.
- Seznam nahrazovaných řetězců modulu Token pro modul Rules** (milos.kroulik, 20 Duben, 2009 - 18:18). The article text explains that the Token module is used for replacing strings in the Rules module. A 'Číst dál' link is present.

On the right side of the page, there are several widgets:

- Reklama**: A banner for 'HRAJE SE o 100 CEN' featuring a woman's face.
- Články**: A list of articles with titles like 'šablony a vzhled bezpečnost cck drupal drupal.cz image moduly navigace ostatní případové studie pro webmastery reklama setkání, akce taxonomy tema tipy & triky token tutorial upgrade vývoj drupalu views webdev workshop začátečníci'.
- Hledat**: A search bar with the text 'Google Vlastní vyhledávání' and a 'Hledat' button.
- Přihlášení**: A login form with fields for 'Username or e-mail: *' and 'Heslo: *', and a 'Přihlásit se' button. Below the form are links for 'Přihlásit pomocí OpenID', 'Vytvořit nový účet', and 'Zaslat nové heslo'.

Obrázek P1.1: Drupal

Joomla!
...because open source matters

Look Sensational with..... gear from the Joomla! Shop

Joomla! 1.5 - Experience the Freedom! It has never been easier to create your own dynamic Web site. Manage all your content from the best CMS admin interface and in virtually any language you speak.

Home search...

Main Menu

- Home
- Joomla! Overview
- Joomla! License
- More about Joomla!
- FAQ
- The News
- Web Links
- News Feeds

Who's Online
We have 2 guests online

Advertisement

Featured Links:
[Joomla! Extensions](#)
 Joomla! Components, Modules, Plugins and Languages by the bucket load.
[Joomla! Shop](#)
 For all your Joomla! merchandise.
[Ads by Joomla!](#)

Resources

- Joomla! Home
- Joomla! Forums
- Joomla! Documentation
- Joomla! Community
- Joomla! Magazine
- OSM Home
- Administrator

Key Concepts

- Extensions
- Content Layouts

Welcome to the Frontpage
Joomla! Community Portal
 Written by Administrator
 Saturday, 07 July 2007 09:54

The Joomla! Community Portal is now online. There, you will find a constant source of information about the activities of contributors powering the Joomla! Project. Learn about [Joomla! Events](#) worldwide, and see if there is a [Joomla! User Group](#) nearby.

The Joomla! Community Magazine promises an interesting overview of feature articles, community accomplishments, learning topics, and project updates each month. Also, check out [JoomlaConnect™](#). This aggregated RSS feed brings together Joomla! news from all over the world in your language. Get the latest and greatest by clicking [here](#).

Last Updated on Saturday, 07 July 2007 09:54

We are Volunteers
 Written by Administrator
 Saturday, 07 July 2007 09:54

The Joomla Core Team and Working Group members are volunteer developers, designers, administrators and managers who have worked together to take Joomla! to new heights in its relatively short life. Joomla! has some wonderfully talented people taking Open Source concepts to the forefront of industry standards. Joomla! 1.5 is a major leap forward and represents the most exciting Joomla! release in the history of the project.

Last Updated on Saturday, 07 July 2007 09:54

Joomla! Security Strike Team
 Written by Administrator
 Saturday, 07 July 2007 09:54

The Joomla! Project has assembled a top-notch team of experts to form the new Joomla! Security Strike Team. This new team will solely focus on investigating and resolving security issues. Instead of working in relative secrecy, the JSST will have a strong public-facing presence at the [Joomla! Security Center](#).

Last Updated on Saturday, 07 July 2007 09:54

[Read more... >>](#)

Millions of Smiles
 Written by Administrator
 Saturday, 07 July 2007 09:54

The Joomla! team has millions of good reasons to be smiling about the Joomla! 1.5. In its current incarnation, it's had millions of downloads, taking it to an unprecedented level of popularity. The new code base is

Joomla! License Guidelines
 Written by Administrator
 Wednesday, 20 August 2008 10:11

This Web site is powered by Joomla! The software and default templates on which it runs are Copyright 2005-2008 [Open Source Matters](#). The sample content distributed with Joomla! is licensed under the [Joomla! Electronic](#)

Obrázek P1.2: Joomla

Log in

Login Name

Password

[Forgot your password?](#)

[New user?](#)

Wmopko

— filed under: [Plone tutorial](#), [Plone trial](#), [Demo](#), [Plone test](#), [Plone demonstration](#), [Plone](#), [Plone demo](#)

Welcome to Nidelven IT's Plone hosting demo!

Also available in presentation mode...

Welcome to [Nidelven IT's](#) Plone hosting demo. Register, log in and play around to get a feel for what Plone is about. :)

If you're not sure what to do next, have a look at our [introduction to Plone](#) for information on how to sign up and create/publish objects.

[Send this](#) [Print this](#)

News
Buy my product! Mar 08, 2009
Click on save and you should get the following view with the information you Feb 02, 2009
Man Bites Dog Jun 22, 2008
test item Jun 10, 2008
Rune's test item Jun 01, 2008
More news...

« May 2009 »						
Mo	Tu	We	Th	Fr	Sa	Su
					1	2 3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

Obrázek P1.3: Plone

CMS Made Simple nabízí rychlý a jednoduchý způsob, jak vytvořit a následně spravovat jak Vaše rodinné webové stránky, tak prezentaci mezinárodní společnosti.

Stáhnout zde
» verze: 1.5.4

Vlastnosti | **Moduly** | **Design** | **Vývoj**

- Zdrojový kód pod [licencí GPL](#)
- Bohaté možnosti vytváření modulů s pomocí API
- Úplná dokumentace API
- [Smarty šablony](#)
- Databázová abstrakce

Více »

Zkušenosti uživatelů [Více »](#)

Pomocí CMSMS je jednoduché vytvořit stránky a pak je předat běžným uživatelům do správy. Na rozdíl od jiných redakčních systémů není CMSMS příliš složitý a také není pouze pro blogy. Rozbalovací menu jsou velmi snadná na používání. K dispozici je také výborné diskusní fórum.
Martin Johnson (IMBA-UK)

Podpořte nás [Více »](#)

Fórum - poslední příspěvky [Více »](#)

Re: Oprávnění v News
Tue, 19 May 2009 11:24:11 GMT

Re: Pekna URL
Tue, 19 May 2009 11:22:22 GMT

Oprávnění v News
Mon, 18 May 2009 15:38:05 GMT

Re: Pekna URL

Novinky

Verze 1.5.3 - Arcibio
Vložil kino.

Včera byla uvolněná nová verze CMS Made Simple 1.5.3. pod označením Arcibio. Obsahuje spoustu oprav, ať již v samotném CMS MS, ale i v modulech pro vyhledávání, novinky, editoru TinyMCE. Všechny změny naleznete v originálním [postu](#).

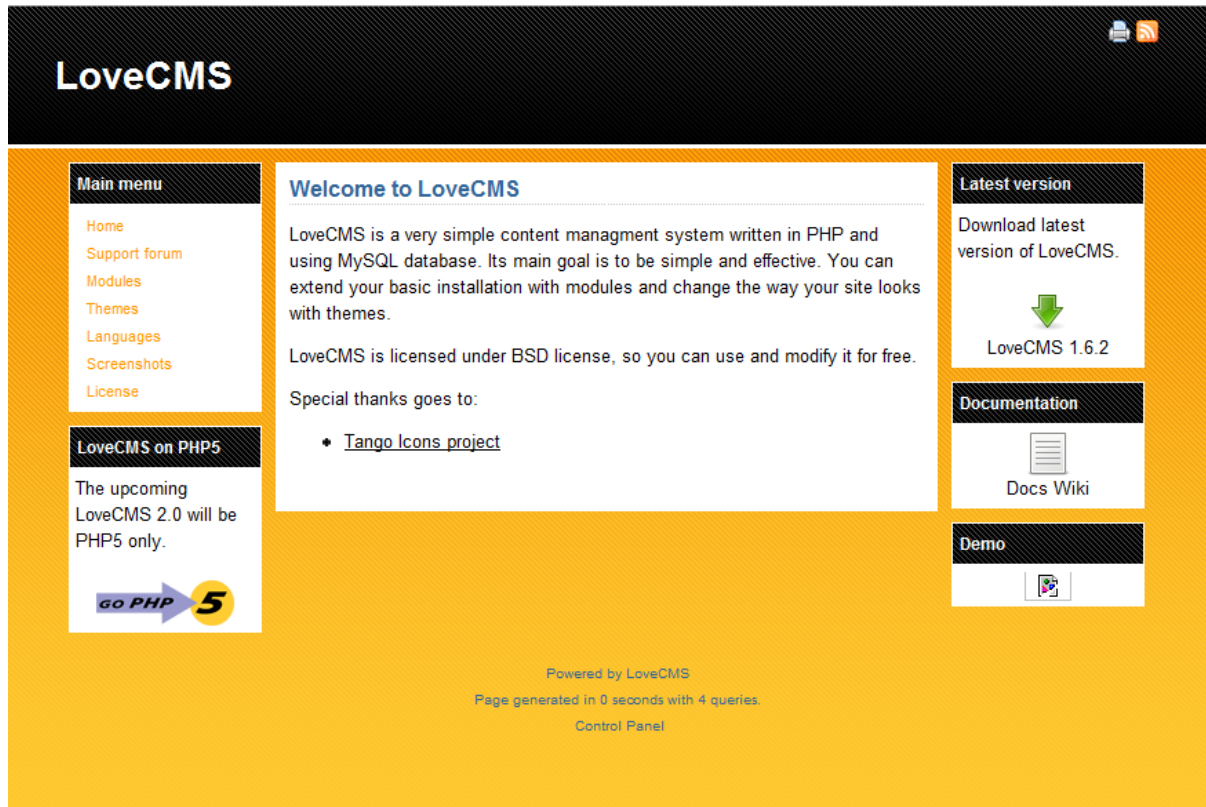
Aktuální verzi můžete stahovat přímo na oficiálních [stránkách systému](#), kde lze najít i tzv. diff (menší balíček sloužící pro přechod z verze 1.5.2 - stačí překopírovat na Váš web).

Vydání CMS Made Simple 1.5 a 1.5.1
Vložil milos.kroulik.

1. prosince vyšla nová verze CMS Made Simple - 1.5 "Puerto Rico". Mezi mnoha dalšími změnami nově umožňuje např. kopírování stránek. Ted Kulp, jeden z autorů CMS Made Simple, doporučuje co nejrychlejší upgrade na tuto verzi. Brzy bylo bohužel objeveno několik nepříjemných chyb, proto 4. prosince následovala opravná verze 1.5.1 "San Juan".
[Více](#)

Přizpůsobení výstupu modulů a administračních šablon
Vložil milos.kroulik.

Obrázek P1.4: CMS Made Simple



Obrázek P1.5: LoveCMS

Welcome Guest, [Login](#) or [Register](#)

Navigation

- [Home](#)
- [Features](#)
- [Downloads](#)
- [How to Install?](#)
- [Managing content](#)
- [FAQ](#)
- [License](#)
- [Contact](#)
- [Support NanoCMS](#)

Get notified

NanoCMS is actively being developed and maintained, please subscribe to stay tuned.

Support

If you ever feel my works were useful, a Donation would be greatly appreciated.

You can donate thru PayPal...

About NanoCMS v0.4

Nano CMS is the tiniest CMS you can find around. The user interface and the functionality are very very simple and extremely easy to use. The core feature of NanoCMS is that it is filebased and does not use any database at all, which makes it super easy to install - just extract and that's it.

Pages can be created through this CMS in a breeze without touching any code, and added to that links to those pages are generated automatically too.

Customizing the design/look and feel is very simple too, it just takes 5 minutes even for newbies.

NanoCMS is programmed by [Kalyan Chakravarthy](#)

Latest News ()

Title	Comments
French Translation for NanoCMS	(1)
Spanish translation for NanoCMS	(0)
NanoCMS v0.4 final - Automatic Upgrader	(2)
NanoCMS v0.4 stable released	(3)
Watchout for 14 feb	(2)

Recent Discussions

Topic	Views	Replies	Last Reply
Noob installation problem	43	4	Today 1:38 pm by Zorchenhimer
Template: Music Studio	44	0	Yesterday 1:07 pm by vishalhd
Tweak: Simple Search v1.3	2710	55	Yesterday 12:21 pm by Zorchenhimer
Tweak: News/Blog page - v1.1	3406	79	Yesterday 12:20 pm by Zorchenhimer
[tweak] Lightbox+	67	1	Yesterday 4:50 am by boshy

Downloads



[More downloads](#)

Information

- [Installation Instructions](#)
- [Tweakers, what and how to ?](#)
- [Language translations](#)
- [Full docs](#)

Obrázek P1.6: NanoCMS