

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INTELLIGENT SYSTEMS

INFORMAČNÍ SYSTÉM MALÉ OBCE
POMOCÍ PHP/MYSQL

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

VOJTĚCH BRTNÍK

BRNO 2008



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INTELLIGENT SYSTEMS

INFORMAČNÍ SYSTÉM MALÉ OBCE POMOCÍ PHP/MYSQL

VILLAGE WEB INFORMATION SYSTEM BASED ON PHP/MYSQL

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

VOJTĚCH BRTNÍK

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. PAVEL JURKA,

BRNO 2008

Abstrakt

Cílem práce je implementace informačního systému malé obce, který usnadní a zefektivní komunikaci a získávání informací občanům v interakci s obecním úřadem. Tento systém je implementován za pomoci PHP5, šablonovacího systému SMARTY, JavaScriptu, CSS a HTML. Systém bude využívat relační databáze MySQL 5.0.

Klíčová slova

Obecní úřad, elektronická podatelna, úřední deska, databáze, informační systém, PHP, CSS, SMARTY.

Abstract

The aim of this project is implementation of an information system for small cities, for data extraction and communication between community and the municipal office. This system is created in PHP5, with the use of SMARTY template system, JavaScript, CSS and HTML. The system uses MySQL 5.0 relation database.

Keywords

Municipal office, electronic registry, bulletin board, database, information system, PHP, CSS, Smarty

Citace

Vojtěch Brtník: Informační systém malé obce
pomocí PHP/MySQL, bakalářská práce, Brno, FIT VUT v Brně, 2008

Informační systém malé obce pomocí PHP/MySQL

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana ing.Pavla Jurky. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Vojtěch Brtník

23. ledna 2008

Poděkování

Tímto bych rád poděkoval ing.Pavlu Jurkovi, za odborné vedení a cenné rady v průběhu práce

© Vojtěch Brtník, 2008.

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

1	Úvod	3
2	Úvod do problematiky	4
2.1	E-Government	4
2.2	Uživatelé internetu	4
2.3	Města a obce na internetu	4
3	Cíle práce	6
3.1	Nároky kladené na systém	7
3.1.1	Povinný obsah	7
3.1.2	Nároky na funkčnost systému	8
3.1.3	Nároky uživatele	8
3.1.4	Bezpečnost	8
4	Návrh řešení	10
4.1	ER-diagram	10
4.2	USE case diagram	10
5	Implementace	13
5.1	Softwarové prostředky	13
5.1.1	HTML	13
5.1.2	PHP	13
5.1.3	Smarty	14
5.1.4	CSS	14
5.1.5	JavaScript	15
5.1.6	MySQL	15
5.2	Databáze	15

5.2.1	Tabulka uživatel	15
5.2.2	Tabulka hlavní odkazy	15
5.2.3	Tabulka vedlejší odkazy	15
5.2.4	Tabulka obsah	16
5.2.5	Tabulka témata	16
5.2.6	Tabulka příspěvky	16
5.2.7	Tabulka mapa	16
5.2.8	Tabulka foto	16
5.2.9	Další tabulky	16
5.3	Architektúra systému	16
5.4	Jednotlivé části systému	17
5.4.1	Obsah stránek	17
5.4.2	Struktura stránek	17
5.4.3	Aktuality	18
5.4.4	Úřední deska	18
5.4.5	E-podatelna	18
5.4.6	Forum	18
5.4.7	Mapa obce	19
5.4.8	Fotogalerie	19
5.4.9	Uživatelé	19
5.4.10	Validita odesílaných formulářů	19
6	Návrh možných rozšíření	20
7	Závěr	21
A	Příloha 1	23

Kapitola 1

Úvod

Uplynulý rok lze označit jako velmi rušný a v některých případech přelomový pro informatiku ve městech a obcích. Byl to důsledek vydání nových zákonů a vyhlášek kladoucích další požadavky na rozvoj v oblasti informatiky obcí. Obsahem této práce je návrh a implementace informačního systému, který by vyhovoval potřebám malé obce. Při jeho tvorbě je nutné zohlednit požadavky, které jsou kladeny na kvalitní informační systém. Těmi jsou především slučitelnost s používanými webovými prohlížeči, intuitivní a přehledná navigace systémem, jednoduchá údržba systému a bezpečnost uložených dat. Splnění těchto předpokladů je základním měřítkem úspěšného systému.

Kapitola 2

Úvod do problematiky

2.1 E-Government

E-Government je proces transformace vztahů veřejné zprávy pomocí komunikačních a informačních technologií, za účelem spolehlivěji a efektivněji poskytovat služby veřejné zprávy veřejnosti a nalezení cest k větší otevřenosti veřejné správy. Snahou E-Governmentu je zvýšení výkonnosti, které by především vedlo ke zjednodušení činností veřejnosti při styku s veřejnou správou.

2.2 Uživatelé internetu

Ve druhém čtvrtletí roku 2006 použilo 41%(1,44 milionu jednotlivců) všech uživatelů internetu tento druh komunikace ve vztahu k veřejné zprávě. 36% uživatelů použilo v tomto období internet k vyhledání informací na webových stránkách úřadů, 17% za účelem vytisknutí potřebného dokumentu, 7% k vyplnění formuláře přímo na webu obce a 13% využilo možnosti komunikovat s úředníky prostřednictvím elektronické pošty[1].

Uživatelé, kteří nemají zájem o elektronickou komunikaci s úřady ve 40% uvedli jako hlavní důvod potřebu osobního jednání s úředníkem, 16% chybí jistota že jejich problém bude při neosobním kontaktu řešen, 16% uživatelů nemá dostatek informací o možnostech této komunikace. U ekonomických subjektů byla situace v tomto období značně lepší.

2.3 Města a obce na internetu

Z průzkumu společnosti Qbizm technologies , provedeného v roce 2006, vyplývá, že již téměř každé město a obec vlastní své webové stránky (celých 92%), jejichž obsah je poměrně často

aktualizován. Průzkum dále odhalil fakt, že 48% úřadů nemá možnost přímo ovlivňovat obsah svých webů, a proto musí pokaždé využívat služeb externích společností. Přičemž 96% obcí, které se účastnily průzkumu udávalo, že svoje stránky aktualizují několikrát v rámci jednoho měsíce. U těchto je zřejmá potřeba pro modernizaci stránek, které by byly založeny na bázi redakčního a publikačního systému [2].

Dalším zajímavým zjištěním byl počet měst a obcí, které mají web bez vlastní domény. Tato situace byla zjištěna pouze v 7% případů z testovaného vzorku. Adresy založené na vlastní doméně jako například `www.nazevobce.cz` působí důvěryhodnějším dojmem, jsou lépe zapamatovatelné a také lépe reprezentují důstojnost dané obce nebo města.

Z průzkumu také vyplývá, že nejoblíbenější varianta e-mailů obsahuje název obce, města a jeho variace včetně nepraktických kombinací typu `ou.XY`, `obec.XY`, `ou-XY`, `urad.XY` apod. (34%), slovo `podatelna` (10%), nebo jméno a příjmení konkrétní osoby (7%). Intuitivnější by bylo striktně se omezit pouze na používání adresy `podatelna@yx.cz`, která nejlépe vyhovuje charakteru oficiální komunikace.

Kapitola 3

Cíle práce

V této kapitole jsou popisovány cíle které byly ustanoveny pro tento projekt, na jejich základě bude v následné kapitole proveden rozbor celého systému.

Cílem tohoto projektu je implementace informačního systému, sloužícího pro prezentaci obce v prostředí internetu, formou splňující všechny normy stanovené zákonem. Dále by měl systém splňovat všeobecné standardy zavedené v odvětví informačních systému. Informační systém musí být přehledný, s přívětivým prostředím k uživateli, podporující snadné získávání hledaných dat o obci a dění v ní.

Tento systém bude vystaven na principech redakčního systému. Redakční systém umožňuje každému uživateli bez znalosti programování publikovat informace na internetu nebo měnit stávající stránky. Nemusí přitom znát programovací jazyk (X)HTML, PHP, JavaScript a další. Pomocí redakčního systému uživatel upraví stránky do potřebné formy. Všechny publikační systémy pro správu obsahu mají nespornou řadu výhod. Redakční systémy urychlují a značně zjednodušují práci při správě a tvorbě internetových stránek, výrazně snižují náklady na provoz prezentace.

Výhody

- šetří finanční náklady za provedení aktualizace webových stránek dodavatelem
- nezávislost na dodavateli webových stránek, změny které potřebuje provozovatel si snadno provede
- nejste závislí na své kanceláři, redakční systém můžete použít kdekoliv, kde je připojení k internetu

Nevýhody

- investice do redakčního systému zvyšuje cenu webových stránek

- při malém počtu úprav na stránkách, řádově několik za rok, je investice do redakčního systému neefektivní
- při vysokém počtu složitých úprav na stránkách je práce v redakčním systému časově neefektivní

Pro tuto práci sem se rozhodl implementovat vlastní redakční systém. Většina volně dostupných redakčních systémů je na tento druh informačního systému až příliš robustních.

3.1 Nároky kladené na systém

3.1.1 Povinný obsah

Obsah stanovený zákonem České republiky

Oficiální název obce

Důvod a způsob založení obce

Organizační struktura

Kontrolní údaje

Bankovní spojení

IČO, DIČ

Opravné prostředky

Formuláře

Nejdůležitější používané předpisy

Ceník služeb v oblasti poskytování informací

Výroční zprávy dle zákona č.106/1999sb

Seznam organizací působících na území správního celku

§26 zákona č.500/2004 stanovuje obci povinnost zřízovat úřední desku, tato musí být veřejně přístupná.

zřízení Elektronické podatelny

3.1.2 Nároky na funkčnost systému

Základem informačního systému je zobrazení a správa všech údajů povinných pro tento druh aplikace, které vyplývají ze zákonné úpravy platné pro Českou republiku, jako jsou elektronická podatelna, úřední deska, informace o obci a obecním úřadě. Tento obsah může být dále editován uživateli s příslušnými právy. Kromě toho pak zobrazení rubrik, dotvářených běžným registrovaným uživatelem. Diskusní fórum s možností vytvářet vlastní témata a vkládat příspěvky, fotogalerie s nabídkou uploadu fotografií pro registrovaného uživatele a mapa obce s vkládáním fotografií.

Cílem těchto dalších rubrik je oživení dění na webové prezentaci obce, vytvořením komunity pravidelných návštěvníků. Tím z informačního systému obce vytvořit prostor nejen pro uživatele hledající konkrétní informace o obci, ale i pro obyvatele obce, kteří budou systémem používat pro komunikaci mezi sebou.

3.1.3 Nároky uživatele

Dodržováním obecných podmínek pro web a dodržování pravidel pro "Bezbariérový web a přístupnost", až 20% uživatelů internetu je hendikepováno. Pokud bude web dobře čitelný a použitelný pro handicapované uživatele, mnohonásobně se tak zvýší jeho čitelnost a použitelnost i pro běžné uživatele. Dále bude plně použitelný pro jakéhokoli uživatele s jakýmkoli klientem respektujícím normu HTML.

Uživatelské rozhraní vyznačující se snadným ovládáním, jak pro anonymního uživatele, tak pro registrovaného uživatele systému provádějícího jeho správu.

Minimalizování nároků na zpracování, rychlá odezva na prováděné akce pro zvětšení komfortu uživatele.

3.1.4 Bezpečnost

Pro zajištění bezpečnosti uživatelů a systému je nutné dodržovat několik základních bezpečnostních zásad. Základem bezpečného webu je zákaz používání globálních proměnných, ten se provádí nastavením *register_globals* na hodnotu *off* v *php.in*.

Zabezpečení proti útoku typu SQL Injection. Při tomto typu útoku, je hodnota nad kterou bude prováděn SQL dotaz, pozměněna a tím je dosaženo nežádoucího výsledku. Útočník tak získá přístup k jinak pro něho nedostupným datům, nebo může dojít k poškození informací uložených v databázi. Z toho plyne nutnost kontrolovat veškerá vstupní data, odstranit z nich nežádoucí znaky (zpětná lomítka, uvozovky, bílé znaky, ostré závorky, ..) a nahradit je jejich HTML kódem.

V případě odcizení databáze je nutné zabránit útočníkovi ve zneužití důvěrných údajů, proto se do databáze tyto data neukládají jako čitelný text. Používá se hash funkce, která transformuje vstupní řetězec s libovolnou délkou na výsledný řetězec znaků s pevnou délkou, tzv. hash nebo také otisk. Pokud dojde k odcizení hashovaných dat, útočník není schopen v krátké době informaci rozluštit. Typický příklad užití je hashování uživatelských hesel.

Kapitola 4

Návrh řešení

Je nutné navrhnout potřebnou databázi a vytvořit hierarchický systém uživatelů, kterým budou podle jejich práv nabídnuty různé funkce informačního systému.

4.1 ER-diagram

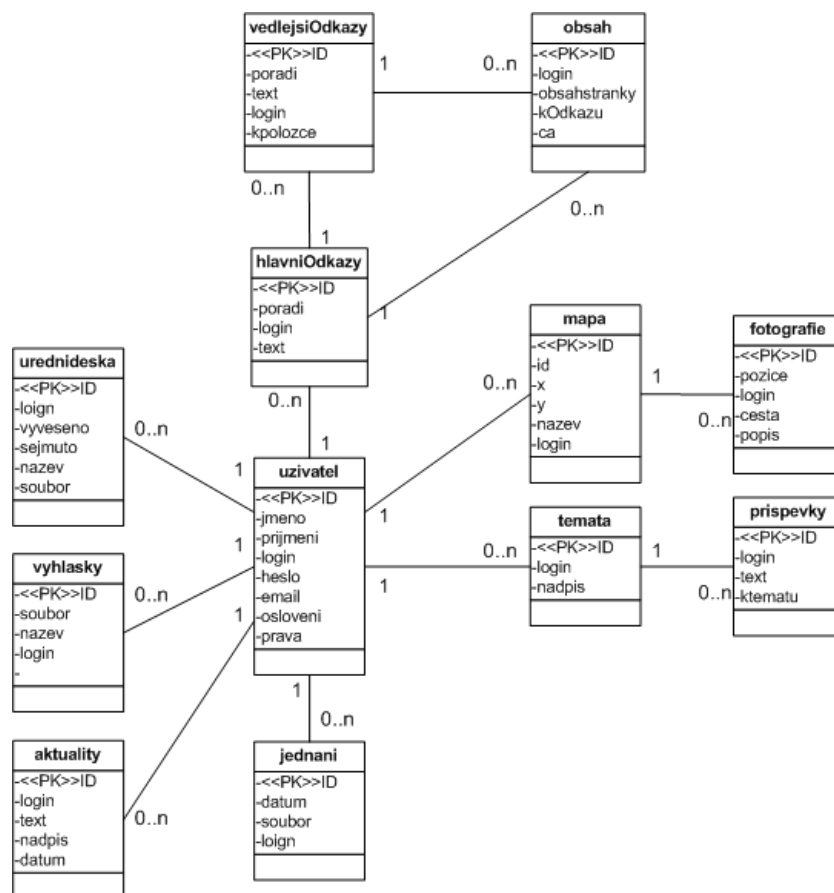
ER-diagram zobrazuje vnitřní strukturu databáze a vztahy v ní. Návrh tabulek je proveden tak, aby byly co nejefektivněji využity a nedocházelo ke zbytečným SQL dotazům. Celá databáze bude uložena v kódování UTF-8. viz [4.1](#)

4.2 USE case diagram

Use case diagram znázorňuje všechny uživatele systému a jejich možné interakce s ním. Systém rozlišuje čtyři skupiny uživatelů: anonymní uživatel, registrovaný, registrovaný uživatel s právy sekretáře a registrovaný uživatel s administrátorským oprávněním. Po vstupu do systému je uživateli přiřazeno jedinečné session ID, podle kterého je identifikovatelný.

Anonymní uživatel je běžný návštěvník systému, jehož práva jsou minimalizována na prohlížení informačního systému, používání podatelny a přispívání na fórum. Rubriky fotogalerie a mapa obce jsou pro něj přístupné pouze v režimu náhledu. Pro rozšíření svých možností musí uživatel provést přihlášení do systému svým jedinečným jménem a heslem, pokud nemá účet vytvořený, může tak učinit. Po přihlášení do systému získává práva registrovaného uživatele. viz [4.2](#)

Registrovanému uživateli jsou automaticky přidělena práva, která mu umožňují podílet se na vytváření obsahu některých částí informačního systému. Dostává možnost vytvářet nové fotogalerie a přispívat do mapy obce, tato činnost je kontrolována další skupinou

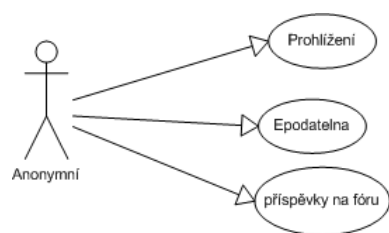


Obrázek 4.1: ER diagram

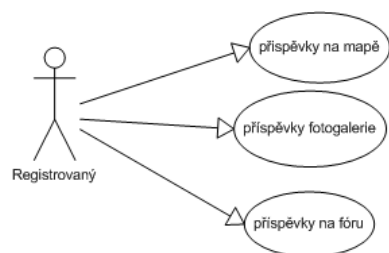
uživatelů na vhodnost obsahu a následně jsou změny ponechány, nebo zamítnuty. viz 4.3

Registrovaný uživatel s právem sekretáře je správce vlastního obsahu informačního systému. Jeho hlavní činností je vytváření a editace článků o obci, údržba úřední desky, vyvěšování vyhlášek a zápisů z jednání zastupitelstva, zobrazování výpisů o hospodaření v obci. Dále pak má možnost cenzurovat obsah vytvářený uživateli s nižším oprávněním. Účet sekretář vzniká z registrovaného uživatele zásahem administrátora, který provádí přidělení práv.viz 4.4

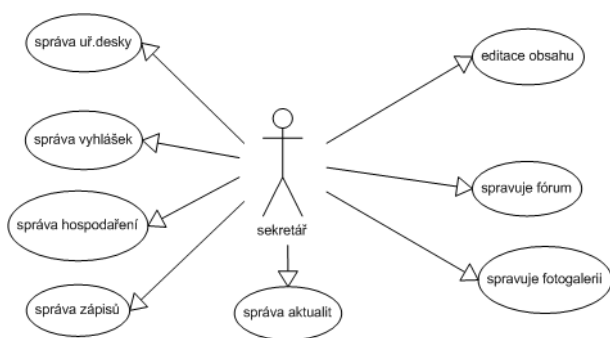
Registrovaný uživatel s právem administrátorským oprávněním spravuje samotnou strukturu webového rozhraní informačního systému. V jeho kompetenci je též spravování účtu jednotlivých uživatelů.viz 4.5



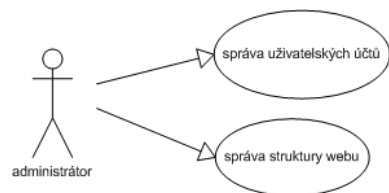
Obrázek 4.2: **Anonymní uživatel**



Obrázek 4.3: **Registrovaný uživatel**



Obrázek 4.4: **Uživatel s právem sekretáře**



Obrázek 4.5: **Adminitrátor**

Kapitola 5

Implementace

Začátek kapitoly je věnován popisům použitých softwarových prostředků, jejich vlastnostem a důvodům proč sem volil právě je. V druhé části se zabývám popisem jednotlivých částí informačního systému, jejich funkcí a strukturou. Při implementaci jsem využíval postupů z [3], [4] a [5]

5.1 Softwarové prostředky

5.1.1 HTML

HTML (Hypertext Markup Language) je jedním z jazyků sloužících k vytváření webových stránek. Jedná se o textové soubory, jejich části jsou formátovány pomocí HTML značek a tím vytvářejí element stránky. Součástí elementu mohou být vnořené elementy. Jednotlivé elementy obsahují atributy, které slouží k upřesnění jejich vlastností [6].

Rozdělení značek podle jejich funkce:

Strukturální značky - dávají dokumentu formu, dělení dokumentu na logické celky

Sémantické značky - popisují povahu příslušného elementu, tyto značky usnadňují automatizované zpracování a vyhledávání informací v dokumentu

Stylistické značky - ovlivňují vzhled elementu při vlastním zobrazení, funkci těchto značek převzalo kaskádové stylování

5.1.2 PHP

PHP je skriptovací programovací jazyk, určený především pro programování dynamických internetových stránek. Skripty jsou prováděny na straně serveru, k uživateli je přenášen až

výsledek jejich činnosti. Obsahuje rozsáhlé knihovny funkcí pro zpracování textu, grafiky, práci se soubory, přístup k většině databázových serverů, podporu celé řady internetových protokolů.[7]

5.1.3 Smarty

Smarty je šablonovací systém pro PHP, který umožňuje vkládat do HTML kódu další příkazy a oddělit tak aplikační logiku od prezentace dat. Největší uplatnění má tento systém při práci na větších projektech, kde je práce rozdělena mezi programátory a designéry [8].

Vlastnosti Smárty

Rychlý šablonovací systém.

Zkompilování aplikační logiky a prezentace dat se provádí pouze jednou, při dalším vstupu na stránky se použije již kompilovaný kód.

Opětovné kompilování se provede jen se šablonami, které byly změněny.

Snadné vytváření vlastních funkcí a modifikátorů proměnných a z toho vyplývající snadné rozšiřování šablonovacího jazyka.

Možnost použití cachování obsahu, kdy se negenerují PHP skripty, ale klasické HTML. Tímto se odlehčí zatížení serveru při zobrazování stránek.

5.1.4 CSS

CSS (Cascading Style Sheets) je jazyk určený pro správu způsobu zobrazení stránek napsaných v (X)HTML, XML. Hlavní myšlenkou je oddělení kódu vzhledu dokumentu od jeho struktury a obsahu. Nahrazuje funkci stylistických značek v HTML a to na jednom místě pro celý web, umožňující rychlé a snadné změny celého stylování. Stylový záznam je složen z posloupnosti pravidel, která určují stylování některého, nebo skupiny elementů. Element na který se dané pravidlo použije je určen selektorem, který je zapsán na začátku pravidla [9]. Výhody CSS:

Dobře strukturované dokumenty

Přehledný a jednoduchý kód

Menší datová velikost stránky

Rychlejší načítání stránek

5.1.5 JavaScript

JavaScript je multiplatformní, objektově orientovaný skriptovací jazyk, který se používá jako interpretovaný programovací jazyk pro WWW stránky, často vkládaný přímo do HTML kódu stránky. Jsou jím obvykle ovládány různé interaktivní prvky GUI nebo tvořeny animace a efekty obrázků. Tento systém má, stejně jako všechny systémy své výhody i nevýhody. Výhodou je mnohem menší zatěžování serveru, na kterém jsou stránky umístěny. Dále jsou to funkce a schopnosti, které lze provádět na straně klienta a které by byly na straně serveru neproveditelné. Největším omezením je nemožnost manipulovat se soubory a to jak na straně serveru, tak na straně klienta [10] .

5.1.6 MySQL

MySQL je multiplatformní databáze. Komunikace s ní probíhá pomocí jazyka SQL, podobně jako u ostatních SQL databází se jedná o dialekt tohoto jazyka s některými rozšířeními. Pro svou snadnou implementovatelnost, výkon a především díky tomu, že se jedná o volně šiřitelný software, má vysoký podíl na v současné době používaných databázích. Velmi oblíbená a často nasazovaná je kombinace MySQL, PHP a Apache jako základní software webového serveru. MySQL bylo od počátku optimalizováno především na rychlost, a to i za cenu některých zjednodušení.

5.2 Databáze

5.2.1 Tabulka uživatel

Tato tabulka slouží k uchování veškerých relevantních dat o uživateli. Hesla uživatelů jsou ukládána v hashované podobě.

5.2.2 Tabulka hlavní odkazy

Obsah této tabulky je tvořen informacemi o odkazech, které budou umístěny v panelu hlavního menu. Při instalaci webu je tato tabulka naplněna skriptem a je tak nahrána základní struktura webu.

5.2.3 Tabulka vedlejší odkazy

V tabulce je uložena struktura vedlejších menu. Stejně jako Tabulka hlavních odkazů je i tato naplněna základními daty při instalaci webu. Atribut *kpolozce* určuje ke kterému

odkazu hlavního menu bude záznam patřit. Atribut *odkaz* stanovuje adresu odkazu.

5.2.4 Tabulka obsah

Tabulka slouží pro uchovávání obsahu dynamicky vytvářených stránek. Obsahuje atributy *login*, *obsahstranky*, *cas*, *kodkazu*. *Login* slouží pro uchování odkazu na osobu, která provedla poslední změny, datum této změny je uložen v atributu *cas*. Atribut *kodkazu* zde slouží jako cizí klíč pro určení místa, kde bude stránka zobrazena. Vlastní obsah stránky je uložen v atributu *obsahstranky* v podobě HTML kódu.

5.2.5 Tabulka témata

Slouží pro uchování témat pro webové fórum. Atributy *login*, *nadpis* a *id* jako primární klíč.

5.2.6 Tabulka příspěvky

Uchovává vlastní příspěvky zobrazované na fóru. Atribut *idtematu* je cizí klíč, který slouží jako reference do tabulky témata.

5.2.7 Tabulka mapa

Tabulka obsahuje atributy *x* a *y*, pomocí kterých se určuje souřadnice na mapě. Této souřadnici je přiřazeno jméno, uložené v *nazev*, které slouží jako nadpis pro fotoalbum.

5.2.8 Tabulka foto

Používá se pro ukládání cesty k vlastní fotografii, která je v atributu *cesta*. Cizí klíč *pozice* slouží jako reference do tabulky mapa.

5.2.9 Další tabulky

Databáze dále obsahuje tabulky *hospodareni*, *vyhlasky*, *obsahobrazky* a *jednani*. Struktura těchto tabulek je velmi podobná, slouží pro ukládání cest k souborům, které jsou používány v informačních rubrikách systému a k dotváření obsahu systému.

5.3 Architektúra systému

Statické části systému jsou složeny z PHP skriptů pro načítání a ukládání dat uložených v databázi. Data jsou zobrazována přes příslušnou šablonu (např. *index.php* a *index.tpl*).

Tyto statické oddíly slouží pro implementaci stránek které pro svojí podstatou nemohou být tvořeny samotným uživatelem.

Část webu, u které je možnost modifikace je prováděna přes šablonu *duprava.tpl*, která obsahuje formuláře pro změny textového obsahu stránek. Tyto změny jsou zpracovány pomocí *duprava.php*.

Dotváření struktury webu je prováděno přes šablonu *strom.tpl*, provedené změny jsou kontrolovány a následně uloženy skriptem *strom.php*.

Skript *upload.php* zpracovává požadavky na nahrávání souborů na server a jejich zařazení do databáze. Veškeré mazání údajů z databáze je uskutečněno za použití *udrzba.php*.

5.4 Jednotlivé části systému

5.4.1 Obsah stránek

Textový obsah stránek je ukládán do tabulky *obsah* a to v podobě textu obsahujícího HTML tagy. Tento text je vkládán pomocí formulářů popsaných v šabloně *duprava.tpl*, která se zobrazuje pouze pro uživatele s právem sekretářů. Formulář je rozdělen na části pro psaní textu, vkládání obrázků na stránku a část pro formátování textu. Část pro editaci textu je tvořena pomocí tagu `TEXTAREA`, do které se načítá text z tabulky *obsah*. Formátování textu je uskutečněno pomocí HTML tagů, které jsou vkládány pomocí JavaScriptu *simple_tag*. Místo vložení tagu do textu je určeno pozicí kurzoru, nebo výběrem části textu, kdy je tato část vložena mezi tagy. Vkládání obrázků se provádí pomocí formuláře, umožňujícího nahrání obrázku na server a jeho převedení na požadovanou velikost. Následným výběrem obrázků ze seznamu obsaženého ve formuláři se na určenou pozici v `TEXTAREA` vloží tag *img* s atributy určujícími obrázek. Po potvrzení formuláře se jeho obsah odesílá na *duprava.php*, kde se nejprve pomocí *mysql_real_escape_string* úprava řetězce pro bezpečné použití v MySQL dotazech a až poté se řetězec uloží do tabulky *obsah*.

5.4.2 Struktura stránek

Struktura odkazů stránek je ukládána do tabulek *hlavniodkazy* a *vedlejsiodkazy*. Při instalaci stránek jsou tyto tabulky naplněny skriptem, který do nich uloží základní strukturu webu. Zásahy tohoto skriptu do databáze jsou vedeny jako úpravy provedené uživatelem *auto*, aby byly jeho akce jednoznačně identifikovatelné. Další úpravy struktury webu může provádět pouze uživatel s oprávněním administrátora a to přes formulář, který je dostupný v hlavním menu pod položkou *Mapa webu*. Zobrazení odkazů je pak prováděno pomocí částí šablon

hlavniMenu.tpl a *menu2.tpl*, které zpracovávají data posílaná z *obsah.php*.

5.4.3 Aktuality

Aktuality jsou zobrazovány na úvodní stránce celého systému pomocí skriptu *index.php* a šablony *index.tpl*, jestliže je zpráva příliš dlouhá je zobrazena pouze její část s možností rozkliknutí. Uživateli s právem sekretáře je zobrazen formulář na vkládání nových aktualit.

5.4.4 Úřední deska

Vývěsky úřední desky jsou uloženy v tabulce *uredniDeska*, pro jejich vkládání je nutné oprávnění sekretáře. Obsah desky je tvořen přes formulář v šabloně *uredniDeska.tpl*. Potřebná data pro charakteristiku příspěvku na desce jsou datum vyvěšení, předpokládaného sejmutí, nadpis a soubor obsahující plné znění příspěvku. Vyplněný formulář se odesílá na *uredniDeska.php*. Jestliže je datum dne, kdy uživatel přistupuje na desku větší než datum sejmutí příspěvku, bude tento příspěvek zobrazen jako neaktivní v archívu úřední desky. Sekretář má možnost předčasně sejmout, editovat a rušit již existující zprávy.

5.4.5 E-podatelna

E-podatelna je tvořena formulářem pro odesílání emailů na adresu podatelny obecního úřadu. Uživatel zde nastavuje typ zprávy, email na který se má odpovědět, text zprávy a má také možnost přiložení souboru ke zprávě.

5.4.6 Forum

Fórum je implementováno pomocí šablony *forum.tpl* a skriptu *forum.php*. Při vstupu na fórum se ve skriptu vybere obsah tabulky *temata*, ten se následně zobrazí přes šablonu jako seznam odkazů na skupiny příspěvků. Přechodem na určité téma se odešle identifikátor tématu v databázi (*\$_GET['poradi']*) ke skriptu *forum.php*, kde je nejprve zkontrolován funkcí *is_numeric* na správnost obsahu. Pokud je kontrola úspěšná poté jsou z tabulky *prispevky* vybrány požadované hodnoty a zobrazeny přes šablonu. Nové příspěvky jsou před vložením do databáze upraveny pomocí *mysql_real_escape_string* pro bezpečné použití v MySQL dotazech. Pokud je přihlášený uživatel s právem sekretáře dostává navíc možnost mazat příspěvky a celá témata. Požadavky na tuto údržbu jsou zasílány ke skriptu *duprava.php*, kde se nejprve ověří práva přihlášeného uživatele a následně se provede smazání z databáze.

5.4.7 Mapa obce

Je implementována pomocí JavaScriptu v *mapa.php*. Skript naplní hodnoty *x*, *y* a *nazev* z tabulky *mapa* do tří pomocných polí. Vlastní mapa využívá vlastností tabulky, které je přiřazen podkladový obrázek. Buňky tabulky pak obsahují nad významnými objekty průhledný obrázek, zarámovaný v HTML odkazu, u každého je pomocí handlerů událostí `onMouseOver` a `onMouseOut` vytvořena záměna průhledného obrázku na symbol indikující záznam v databázi. Hlavní částí mapy je funkce *VytvorMapu*, vstupními argumenty jsou rozměry pozadí, jméno souboru, který bude sloužit jako pozadí, a odkaz na pole obsahující informace vybrané z databáze. Kliknutím na odkaz na mapě se odesílá identifikátor tohoto bodu (`$_GET['id']`) ke skriptu *foto.php*, kde se nejprve pomocí funkce *is_numeric* ověří správnost předávané hodnoty. Podle této hodnoty se z tabulky *foto* vyberou příslušné fotografie.

5.4.8 Fotogalerie

Fotogalerie využívá stejně jako mapa obce tabulky *mapa* a *foto*, rozhodl jsem se pro tento krok z důvodu podobnosti dat, která používají tyto dvě rubriky. Zobrazení je prováděno skriptem *photovod.php* a šablonou *fotoUvod.tpl*. K nahrávání nových fotografií se používá skript *foto.php*. Použitím tabulky *mapa* vzniká potřeba odlišit údaje z mapy od fotogalerie, to je docíleno nastavením souřadnic *x* a *y* pro položky fotogalerie na hodnoty -1. Vlastní fotografie jsou zobrazovány pomocí JavaScriptové knihovny Lightbox2.

5.4.9 Uživatelé

Správa uživatelů je prováděna administrátorem, který jediný má na toto oprávnění. Je mu zobrazen list všech uživatelů s možnostmi na jejich mazání, nebo editaci jednotlivých položek.

5.4.10 Validita odesílaných formulářů

Validita formulářových polí je ověřena ještě před odesláním pomocí reakce na událost formuláře `onSubmit`, která volá JavaScript `Validate`. Tento script vrací v případě chyby v hodnotách formuláře alert okno s upozorněním na špatně zadané hodnoty. Při vypnutí vykonávání JavaScriptu uživatelem jsou formulářová data kontrolována až ve skriptu, který je zpracovává.

Kapitola 6

Návrh možných rozšíření

V případě zájmu je možné systém dále rozšířit o další funkční části. Jednalo by se hlavně o rozšíření zvyšující komfort uživatelů hledajících informace. Jako příklad bych uvedl možnost vytvoření RSS informačního kanálu, který by zasílal uživatelům upozornění na nové příspěvky v informačním systému. Systém je možné rozšířit o funkce exportu dat do XML nebo csv, pro jejich snadnou archivaci. Funkce by mohla sloužit pro uchovávání struktury webu s jejich obsahem.

Kapitola 7

Závěr

Cílem tohoto projektu bylo seznámit se s problematikou informačních systémů měst a obcí, zvolit vhodné implementační prostředky a vytvořit informační systém malé obce. Jedním z hlavních úkolů bylo vytvoření jednoduchého rozhraní pro spravování systému, tak aby uživatelé mohli tento systém používat bez větších znalostí programovacích technik. K editaci obsahu je vhodná znalost přibližné funkce HTML tagů, ale v dnešní době kdy je tento způsob formátování používán velice často (např. email, diskusní fóra), by to neměl být problém i pro méně zkušeného uživatele. Snadná změna obsahu je podpořena přehledným a přívětivým prostředím editačního formuláře.

Díky bakalářské práci jsem si prohloubil znalost PHP5, MySQL, HTML a seznámil jsem se s šablonovacím systémem SMARTY. Při počátcích projektu se systém SMARTY jevil jako nepřilíš přínosný pro vytváření kódu, jeho výhody jsem ocenil až v pozdějším stadiu řešení. Projekt který byl rozdělen na část obsahující PHP kód a na šablonu sloužící pro zobrazení, byl přehlednější a práce s ním intuitivnější.

Aplikace je dostupná na <http://www.vysocan.cz/is>

Literatura

- [1] *Qbizm technologies : Studie připravenosti měst a obcí na přímou komunikaci s občanem* [online]. 2006 [cit. 2008-01-01]. Dostupný z WWW: http://www.mvcr.cz/micr/images/dokumenty/studie_pripravenosti_nov2005.pdf.
- [2] *Český statistický úřad : Použití internetu ve vztahu k veřejné správě* [online]. 2007 , 26.4. 2007 [cit. 2008-01-01]. Dostupný z WWW: <http://www.czso.cz/csu/2006edicniplan.nsf/t/C7002ABAD2/>.
- [3] STEJSKAL, Jan. *Vytváříme WWW stránky pomocí HTML, CSS a JavaScriptu*,. [s.l.] : Computer Press, 2004. 256 s. ISBN 80-251-0167-3.
- [4] STANÍČEK, Petr. *CSS Kaskádové styly : Kompletní průvodce*. [s.l.] : Computer Press, 2003. 192 s. ISBN 80-7226-872-4.
- [5] GILMORE, W. Jason. *Velká kniha PHP 5 a MySQL*. RNDr. Jan Pokorný. [s.l.] : Zoner press, 2005. 712 s. ISBN 80-86815-20-X.
- [6] *Wikipedia : HyperText Markup Language* [online]. 2008 , 19.1.2008 [cit. 2008-01-20]. Dostupný z WWW: <http://cs.wikipedia.org/wiki/Html>.
- [7] *Wikipedie : PHP* [online]. 2008 , 7.1.2008 [cit. 2008-01-11]. Dostupný z WWW: <http://cs.wikipedia.org/wiki/Php>.
- [8] *Smarty : Chapter 1. What is Smarty?* [online]. 2007 , 30.10.2007 [cit. 2008-01-01]. Dostupný z WWW: <http://www.smarty.net/manual/en/what.is.smarty.php>.
- [9] *Wikipedia : Cascading Style Sheets* [online]. 2007 , 7.1.2007 [cit. 2008-01-01]. Dostupný z WWW: http://cs.wikipedia.org/wiki/Cascading_Style_Sheets.
- [10] *Wikipedia : JavaScript* [online]. 2007 , 29. 12. 2007 [cit. 2008-01-01]. Dostupný z WWW: <http://cs.wikipedia.org/wiki/JavaScript>.

Dodatek A

Příloha 1

Požadovaná konfigurace:

- Web server Apache v2.0 a vyšší
- MySQL 5.0
- PHP5

Postup instalace

- Zkopírujte soubory do adresáře nastaveného v konfiguračním souboru serveru
- Vytvořte databázi, přihlašovací jméno a heslo k ní nastavte v *configdb.php*
- Vytvořte a naplňte tabulky databáze provedením SQL dotazů uložených v *createdb.sql*