

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

INFORMAČNÝ SYSTÉM/INTRANET PRE MALÚ OBEC

BAKALÁRSKA PRÁCA
BACHELOR'S THESIS

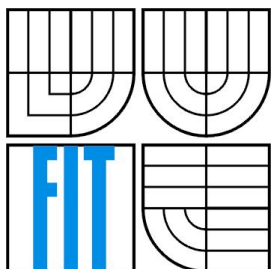
AUTOR PRÁCE
AUTHOR

LADISLAV HLAVIČKA

BRNO 2007



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

INFORMAČNÍ SYSTÉM/INTRANET PRO MALOU OBEC

INFORMATION SYSTEM/INTRANET FOR SMALL MUNICIPALITY

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

LADISLAV HLAVIČKA

VEDOUCÍ PRÁCE

SUPERVISOR

ING. LUKÁŠ STRYKA

BRNO 2007

Vysoké učení technické v Brně - Fakulta informačních technologií

Ústav informačních systémů

Akademický rok 2006/2007

Zadání bakalářské práce

Řešitel: **Hlavička Ladislav**

Obor: Informační technologie

Téma: **Informační systém/intranet pro malou obec**

Kategorie: Databáze

Pokyny:

1. Seznamte se s problematikou tvorby informačních systémů/intranetů.
2. Seznamte se s požadavky kladenými na internetovou aplikaci obce Dolnozemská, která umožní oprávněným osobám měnit obsah WWW stránek bez znalosti jazyka HTML.
3. Zvolte implementační prostředí a proveďte detailní návrh modulární webové aplikace pomocí jazyka UML.
4. Vytvořte funkční prototyp systému.
5. Ověřte funkčnost systému.
6. Zhodnoťte dosažené výsledky a na základě získaných zkušeností diskutujte možná vylepšení systému.

Literatura:

- DeLisle, M.: *PHPMy admin efektivní správa MySQL*. 1. vyd. Brno: Zonner Press, 2004.
- *Manuál PHP* [online]. URL: <<http://www.php.net/manual/cs/>>
- *MySQL Documentation* [online]. URL: <<http://dev.mysql.com/doc/>>
- KOSEK, Jiří. *Vše o WWW* [online]. URL: <<http://www.kosek.cz/>>

Při obhajobě semestrální části projektu je požadováno:

- Body 1 až 3.

Podrobné závazné pokyny pro vypracování bakalářské práce naleznete na adrese

<http://www.fit.vutbr.cz/info/szz/>

Technická zpráva bakalářské práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap (20 až 30% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním paměťovém médiu (disketa, CD-ROM), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Stryka Lukáš, Ing.**, UIFS FIT VUT

Datum zadání: 1. listopadu 2006

Datum odevzdání: 15. května 2007

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav informačních systémů
602 00 Brno, Božetěchova 2

doc. Ing. Jaroslav Zendulka, CSc.
vedoucí ústavu

LICENČNÍ SMLOUVA POSKYTOVANÁ K VÝKONU PRÁVA UŽÍT ŠKOLNÍ DÍLO

uzavřená mezi smluvními stranami

1. Pan

Jméno a příjmení: **Ladislav Hlavička**
Id studenta: 84374
Bytem: Pořná 43, 940 53 Nové Zámky
Narozen: 03. 09. 1985, Nové Zámky
(dále jen "autor")

a

2. Vysoké učení technické v Brně

Fakulta informačních technologií
se sídlem Božetěchova 2/1, 612 66 Brno, IČO 00216305
jejímž jménem jedná na základě písemného pověření děkanem fakulty:

.....
(dále jen "nabyvatel")

Článek 1

Specifikace školního díla

1. Předmětem této smlouvy je vysokoškolská kvalifikační práce (VŠKP):
bakalářská práce

Název VŠKP: Informační systém/intranet pro malou obec
Vedoucí/školitel VŠKP: Stryka Lukáš, Ing.
Ústav: Ústav informačních systémů
Datum obhajoby VŠKP:

VŠKP odevzdal autor nabyvateli v:

tištěné formě počet exemplářů: 1
elektronické formě počet exemplářů: 2 (1 ve skladu dokumentů, 1 na CD)

2. Autor prohlašuje, že vytvořil samostatnou vlastní tvůrčí činností dílo shora popsané a specifikované. Autor dále prohlašuje, že při zpracovávání díla se sám nedostal do rozporu s autorským zákonem a předpisy souvisejícími a že je dílo dílem původním.
3. Dílo je chráněno jako dílo dle autorského zákona v platném znění.
4. Autor potvrzuje, že listinná a elektronická verze díla je identická.

Článek 2 Udělení licenčního oprávnění

1. Autor touto smlouvou poskytuje nabyvateli oprávnění (licenci) k výkonu práva uvedené dílo nevýdělečně užít, archivovat a zpřístupnit ke studijním, výukovým a výzkumným účelům včetně pořizování výpisů, opisů a rozmnoženin.
2. Licence je poskytována celosvětově, pro celou dobu trvání autorských a majetkových práv k dílu.
3. Autor souhlasí se zveřejněním díla v databázi přístupné v mezinárodní síti:
 - ihned po uzavření této smlouvy
 - 1 rok po uzavření této smlouvy
 - 3 roky po uzavření této smlouvy
 - 5 let po uzavření této smlouvy
 - 10 let po uzavření této smlouvy(z důvodu utajení v něm obsažených informací)
4. Nevýdělečné zveřejňování díla nabyvatelem v souladu s ustanovením § 47b zákona č. 111/1998 Sb., v platném znění, nevyžaduje licenci a nabyvatel je k němu povinen a oprávněn ze zákona.

Článek 3 Závěrečná ustanovení

1. Smlouva je sepsána ve třech vyhotoveních s platností originálu, přičemž po jednom vyhotovení obdrží autor a nabyvatel, další vyhotovení je vloženo do VŠKP.
2. Vztahy mezi smluvními stranami vzniklé a neupravené touto smlouvou se řídí autorským zákonem, občanským zákoníkem, vysokoškolským zákonem, zákonem o archivnictví, v platném znění a popř. dalšími právními předpisy.
3. Licenční smlouva byla uzavřena na základě svobodné a pravé vůle smluvních stran, s plným porozuměním jejímu textu i důsledkům, nikoliv v tísní a za nápadně nevýhodných podmínek.
4. Licenční smlouva nabývá platnosti a účinnosti dnem jejího podpisu oběma smluvními stranami.

V Brně dne:

.....

Nabyvatel

.....

Muničan

Autor

Abstrakt

Cílem bakalářské práce je implementovat informační systém pro malou obec v prostředí internetu. S využitím získaných teoretických znalostí a praktických zkušeností jsem navrhl jednoduchý informační systém obce. Systém by měl poskytnout uživatelům prostředí, které poskytne jednoduché komunikační rozhraní se systémem, a oprávněným osobám dovolí měnit obsah WWW stránek bez nutnosti znalosti jazyka HTML. K vytvoření systému byly použity, v dnešní době běžné, technologie pro tvorbu dynamických internetových aplikací, jako jsou skriptovací jazyk PHP, značkovací jazyk XHTML a databázový systém MySQL.

Klíčová slova

Informační systém, portál, přístupnost, malá obec, databáze, UML, PHP, MySQL, JavaScript, XHTML, diagram případu užití, ER diagram

Abstract

The goal of the bachelor thesis is an implementation of the information system for a small municipality in the internet environment. I have designed the simple information system for small municipality with the theoretical knowledge and practical experience. The system should be able to provide an environment that offers a simple communication interface to ordinary users and allows the authorized persons to change the content of www site without a necessity of HTML language skills. Contemporary technologies, especially scripting language PHP, markup language XHTML and database system MySQL, were used for design of the system.

Keywords

Information system, portal, accessibility, small municipality, database, UML, PHP, MySQL, JavaScript, XHTML, use case diagram, ER diagram

Citace

Ladislav Hlavička: Informačný systém/intranet pre malú obec, bakalárska práca, Brno, FIT VUT v Brně, 2007

Informačný systém/intranet pre malú obec

Prehlásenie

Prehlasujem, že túto bakalársku prácu som vypracoval sám pod vedením Ing. Lukáša Stryku.

Uviedol som všetky literárne pramene a publikácie, z ktorých som čerpal.

.....
Ladislav Hlavička
10.5.2007

Pod'akovanie

Na tomto mieste by som chcel pod'akovať svojmu odbornému konzultantovi, pánovi Ing. Lukášovi Strykovi, za jeho hodnotné rady a pripomienky k vypracovaniu tejto bakalárskej práce.

© Ladislav Hlavička, 2007.

Táto práca vznikla ako školné dielo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práca je chránená autorským zákonom a jej použitie bez udeleného oprávnenia autora je nezákonné, s výnimkou zákonom definovaných prípadov..

Obsah

Obsah	1
1 Úvod.....	3
2 Informačné systémy	4
2.1 Pojem informačného systému	4
2.2 Vývoj informačného systému.....	5
2.2.1 Požiadavky na informačný systém.....	5
2.2.2 Analýza	5
2.2.3 Návrh	6
2.2.4 Implementácia.....	6
2.2.5 Testovanie systému.....	6
2.2.6 Nasadenie informačného systému a jeho údržba	6
2.3 Modely vývoja informačných systémov	7
2.3.1 Model vodopádu	7
2.3.2 Špirálový model.....	8
2.3.3 Evolučný model.....	9
3 Implementačné prostriedky.....	10
3.1 Jazyk UML.....	10
3.2 Značkovací jazyk XHTML	11
3.3 Kaskádové štýly – CSS	11
3.4 Skriptovací jazyk JavaScript.....	11
3.5 Skriptovací jazyk PHP	12
3.6 Relačný databázový systém MySQL	12
4 Analýza a návrh	13
4.1 Požiadavky na systém	13
4.2 Návrh aplikácie pomocou jazyka UML	15
4.2.1 Diagram prípadov užitia	15
4.2.2 ER diagram – diagram tried.....	16
5 Implementácia systému.....	20
5.1 Vytvorenie databázy.....	20
5.2 Uživatelské rozhranie – vzhľad systému	21
5.3 Registrácia nového užívateľa	22
5.4 Zabudnuté heslo	23
5.5 Prihlasovanie a odhlasovanie	24
5.6 Aktuality.....	25

5.7	Fotogaléria.....	26
5.8	Kalendár akcií	27
5.9	Dokumenty	29
5.10	Články	30
5.11	Služby a firmy	31
5.12	Úradná tabuľa.....	32
5.13	Elektronická podateľňa	33
5.14	Portál futbalistov	34
5.15	Diskusné fórum	35
5.16	Administrátorský profil	36
5.17	Ďalšie funkcie systému.....	37
6	Ďalší rozvoj systému.....	40
7	Záver	41
	Literatúra	42
	Zoznam príloh.....	43

1 Úvod

Žijeme v modernom svete, keď už základnými informačnými zdrojmi nie sú iba tlač a masovokomunikačné nástroje, ale postupne sa do popredia dostane internet. To, že jednotlivé firmy ponúkajú na svojich internetových stránkach svoje produkty, zákazníci nakupujú pomocou elektronického obchodovania, ľudia čítajú správy a rôzne články na webe, je už samozrejmou. Ale ľudí nezaujíma iba to, čo sa deje vo svete. Zaujíma ich aj každodenný život okolia, kde žijú. Preto už skoro každé mesto, každá obec má vlastnú domovskú stránku, kde návštevníci okrem základných dôležitých informácií sú oboznámené aj kultúrnymi, športovými podujatiami, ktoré sú pripravené pre občanov, a mnoho ďalšími zaujímavosťami. Pri prezeraní internetových stránok jednotlivých obcí sa často stretávame iba s jednoduchými domovskými stránkami, ktoré ponúkajú návštevníkom iba statické stránky, ktoré sú iba zriedkavo zmenené, a preto návštevnosť týchto stránok nie je až taká veľká. Práve preto sa do popredia postupne dostanú informačné systémy, informačné portály obcí, ktorých obsah na to oprávnené osoby pravidelne a jednoducho môžu zmeniť.

Cieľom tejto bakalárskej práce bolo vytvorenie takého informačného portálu, ktorý svojou mnohostrannosťou a aktuálnosťou upúta pozornosť návštevníkov, a aby vďaka týmto vlastnostiam návštevníci pravidelne prezerali stále aktualizovaný obsah portálu.

Druhá kapitola sa zaoberá popisom základných znalostí o informačných systémoch. Sú tu podrobne popísané jednotlivé fázy vývoja informačných systémov, ďalej uvedené niektoré vývojové modely, na základe ktorých môžeme postupovať pri vytváraní informačných systémov.

Tretia kapitola je venovaná implementačným nástrojom, ktoré som používal pri vytvorení celého systému. Nájdeme tu stručný popis jednotlivých implementačných prostriedkov - na dátovej vrstve bol použitý databázový systém MySQL, na aplikačnej vrstve technológia PHP, na klientskej vrstve jazyk XHTML spolu s kaskádovými štýlmi CSS.

Témou štvrtej kapitoly je samotná analýza a návrh celého systému. Na začiatku sú podrobne popísané požiadavky na informačný systém obce a hierarchia používateľov, ktorí budú mať prístup k systému. Pre každú skupinu používateľov sú popísané funkcie, ktoré by im mal systém poskytnúť. Ďalej nájdeme tu diagram prípadu použitia a ER diagram navrhovaného systému.

V piatej kapitole je popísaná implementácia jednotlivých funkcií systému. Na začiatku nájdeme popis vytvárania databázy a implementáciu užívateľského rozhrania. Potom nasleduje podrobný popis každej časti systému, ktoré boli vytvorené na základe vstupných požiadaviek. Pretože do systému má prístup až päť typov užívateľov, nájdeme tu podrobne rozpísané, aké možnosti majú jednotlivé skupiny užívateľov v daných častiach systému.

Ďalšie možnosti vylepšenia systému nájdeme v šiestej kapitole. Zhrnul som tu nápady, ako by sme mohli ešte rozšíriť funkcie, ktoré systém poskytuje.

2 Informačné systémy

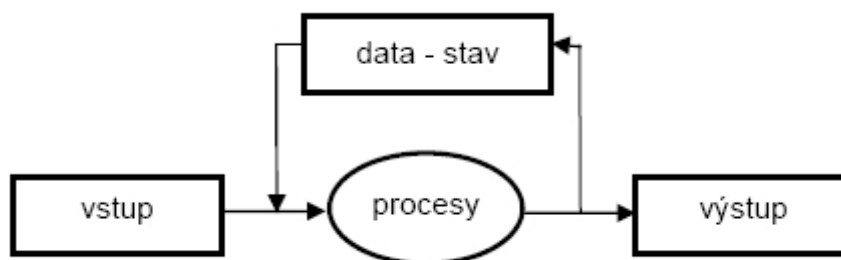
Kapitola obsahuje základné informácie o informačných systémoch, ďalej podrobný popis jednotlivých fáz vývoja informačných systémov. V ďalších podkapitolách sú uvedené jednotlivé modely vývoja, podľa ktorých môžeme postupovať pri vytváraní celého systému.

2.1 Pojem informačného systému

Informačný systém je systém na uchovávanie, znovu získavanie, spájanie a vyhodnocovanie informácií – teda súhrn prvkov, ktorý ako celok slúži na získavanie, uchovávanie a šírenie informácií. Rozvoj informačných systémov priniesol potrebu všeobecného prístupu k informáciám uložených v počítačových systémoch a to tak, aby manipulácia s nimi bola čo najjednoduchšia a zároveň funkcie, ktoré zabezpečuje boli dostatočne silné. (zdroj: literatúra [12])

Informačný systém sa skladá zo vstupnej časti, kde sa do systému vkladajú, a výstupnej časti, z ktorej sa získavajú informácie. Medzi vstupom a výstupom prebehne transformácia týchto dát. Transformačná časť typicky prevedie nad dátami rôzne algoritmy. Prebiehajú tu rôzne procesy. Typickou časťou informačného systému je spätná väzba, ktorá využíva uložený stav systému. Nie všetky výstupy procesov sú priamo závislé na okamžitých vstupoch, ale závisia na stave systému. (zdroj: literatúra [3])

Na obrázku 2.1 je znázornená schéma informačného systému.



Obrázok 2.1: Schéma informačného systému

(zdroj: literatúra [3] Prof. Ing. Tomáš Hruška, CSc – IIS Studijní opora)

Informačný systém môžeme rozdeliť na dve hlavné časti – na informačný zdroj a na používateľa, kde:

používateľ

- dožaduje sa informácií, posielajú požiadavky na zdroj informácií
- prijíma a spracúva odpovede od zdroja
- môže rozširovať informačný obsah zdroja (zapisuje nové informácie, obohacuje informačnú množinu zdroja) – závisí to od prístupových práv používateľa

informačný zdroj

- čaká na požiadavky od používateľov a spracúva ich (reaguje na ne)
- odosiela odpovede na požiadavky od používateľov – informácie z informačnej množiny zdroja

Informačný zdroj môže byť tvorený viacerými informačnými zdrojmi navzájom kooperujúcimi a javiacimi sa používateľovi ako jeden systém. Vtedy hovoríme o distribuovaných systémoch.

2.2 Vývoj informačného systému

Celý vývoj informačného systému môžeme zhrnúť do niekoľkých hlavných fáz. Sú to:

- požiadavky na informačný systém
- analýza
- návrh
- implementácia
- testovanie
- nasadenie informačného systému a jeho údržba

Podrobný popis týchto fáz nájdeme v ďalších podkapitolách.

2.2.1 Požiadavky na informačný systém

Tu spadá jednanie s klientom, vytýčenie toho, čo má a aj nemá informačný systém priniesť, ako bude celý systém fungovať. Ide vlastne o stanovenie požadovaných funkcií, ktoré má informačný systém poskytovať. Dôležité je stanoviť, že aký druh informácií bude v informačnom systéme spracúvaný, na ktoré ďalšie externé zdroje informácií sa bude pripájať a akým spôsobom sa jednotlivé pripojenia budú realizované. V tejto etape je potrebné určiť, aká bude bezpečnostná politika systému. Pod týmto pojmom rozumieme to, že aké prístupové práva budú mať používatelia. Ďalej stanovíme, akým spôsobom budú používatelia pristupovať k informačným zdrojom, rozsah možného počtu používateľov prístupujúcich naraz k informačným zdrojom. Nemôžeme zabudnúť ani na cenové relácie, v ktorých má byť informačný systém realizovaný (náklady na hardvér, softvér a vývoj informačného systému). (zdroj: literatúra [12])

2.2.2 Analýza

Ide o rozloženie problému na čiastkové algoritmizovateľné podproblémy, ktoré potom je možné jednoduchšie realizovať pomocou informačných prostriedkov (technológií), ktoré máme k dispozícii. Projektant má analyzovať všetky systémové požiadavky, ako napr. funkcia a schopnosti systému, organizačné a užívateľské požiadavky, požiadavky na bezpečnosť a ochranu, prevádzku a údržbu, apod. Fázy stanovení požiadaviek a analýzy dohromady môžeme označiť ako štádium expanzie,

pretože pri nich dochádza ku zhromaždení informácií potrebných pre vytvorenie aplikácie. Štádium expanzie končí rozpracovaním analytického konceptuálneho modelu, ktorý na logickej úrovni reprezentuje požadované zadanie a formálne popisuje riešený problém. (zdroj: literatúra [12])

2.2.3 Návrh

Projektant musí vytvoriť celkovú architektúru systému, jej procesnú, funkčnú a dátovú náplň. Úlohou tejto fázy vývoja informačného systému je teda navrhnuť vhodný dátový model – spôsob a pravidlá, akými budú jednotlivé informácie v systéme uchované, spracúvané a poskytované. Sem patrí návrh štruktúry databázy (entitné množiny, ich relačné vzťahy). Projektant má navrhnuť bezpečnostnú politiku systému, tzn. hierarchiu používateľov, že kto kam má prístup v rámci informačného systému. Výsledkom návrhu má byť model informačného systému so všetkými funkčnými celkami, ich vzájomná komunikácia, prepojenie na dátový model, a model komunikácie s používateľom systému.

2.2.4 Implementácia

Ide o proces tvorby jednotlivých funkčných celkov systému, ich vzájomné prepojenie a zabezpečenie správnej kooperácie na základe návrhov z analýzy informačného systému za pomoci zvolených prostriedkov realizácie – myslí sa vývoj komponent vo vhodnom vývojovom prostredí (tiež navrhnutom v analýze). Výsledkom by mala byť funkčná verzia informačného systému, realizujúca funkcie stanovené požiadavkami. (zdroj: literatúra [12])

2.2.5 Testovanie systému

Cieľom je podrobiť vytvorený informačný systém kritickému testovaniu jeho funkčnosti, najlepšie inou osobou, než realizátorom (programátorom) a zistiť chyby vzniknuté v priebehu vývoja samotného informačného systému v danom vývojovom prostredí. Testovanie je veľmi dôležitou časťou vývoja systému, pri testovaní sa dozvieme o tom, či systém funguje správne, či spĺňa všetky užívateľské požiadavky. Výstupom by mali byť informácie pre vývojára informačného systému, na základe ktorých by mal byť schopný objavené chyby odstrániť. (zdroj: literatúra [12])

2.2.6 Nasadenie informačného systému a jeho údržba

Ide o finálne nasadenie informačného systému u zadávateľa požiadavky. Je to inštalácia jednotlivých komponentov informačného systému a ich nastavenie pre vzájomnú kooperáciu.

Súčasťou ponuky informačného systému by mala byť aj jeho údržba – odstraňovanie chýb vzniknutých v priebehu prevádzky systému, inovácia komponent podľa najnovších trendov v oblasti IT a podľa požiadaviek objednávateľa (zadávateľa požiadavky na informačný systém).

2.3 Modely vývoja informačných systémov

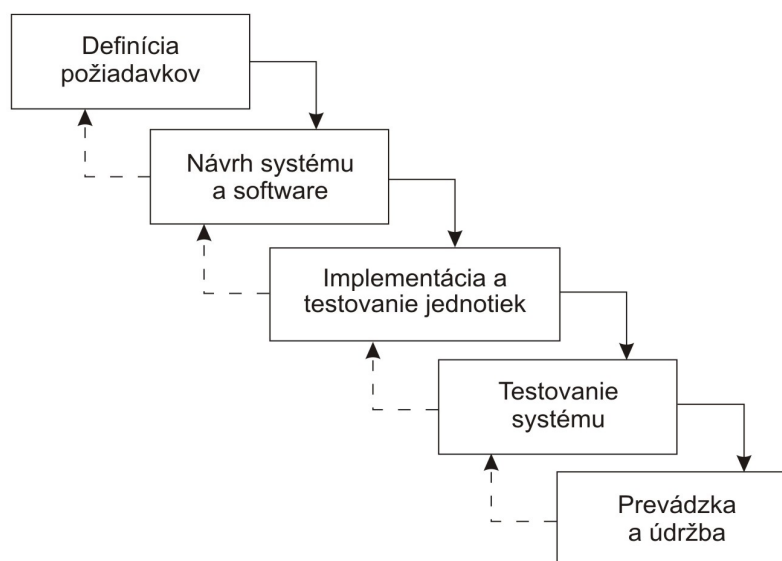
Pri vývoji informačných systémov každý vývojár, projektant má svoj vlastný štýl. Každému vyhovujú iné pravidlá, niekomu dokonca žiadne. Preto je ťažké hovoriť o pravom a univerzálnom spôsobe. Rôzne techniky majú svoje výhody, ktoré môžu byť v špecifických prípadoch nevýhodami a opačne.

Existuje proces tvorby softwaru, keď všetka energia vývoja sa sústreďí na programovanie. Neexistuje žiadna špecifikácia požiadaviek ani analýza a návrh. Táto forma procesu tvorby vôbec nie je výhodná ani efektívna, a vo väčšine prípadov výsledný systém ani nie je funkčný. Pri malých projektoch ale môže byť dokonalá.

Asi najčastejší prístup tvorby aplikácii je model „programuj a opravuj“. Jedná sa o jednoduchý proces vývoja systémov. Obvykle sa začína s určitou predstavou o probléme a tak sa nasadzujú základné pravidlá špecifikácie, programovanie a testovanie konkrétnej časti. Veľká výhoda tohto modelu spočíva v okamžitom možnom prezentovaní (nejakého) výsledku už v prvých štádiách vývoja. Nevýhodou je mnoho chýb vo výslednom projekte.

2.3.1 Model vodopádu

Jedná sa o „vrstevnatú“ tvorbu, kedy vývoj prebieha v určitých štádiách. Prechádza sa analýzou, potom inými procesmi, samotnou implementáciou, testovaním a výsledným produktom. Grafické znázornenie tohto modelu je na obrázku 2.2. Zásadným problémom je fakt, že nemôžeme v jednotlivých častiach vývoja vynechať nijakú časť, pretože by to na výsledný projekt mohlo pôsobiť katastrofálne. Tiež už nie je návratu, ak sa chcú udiat nejaké zmeny v špecifikácii a my sme už vo fáze vývoja. (zdroj: literatúra [2], [13])



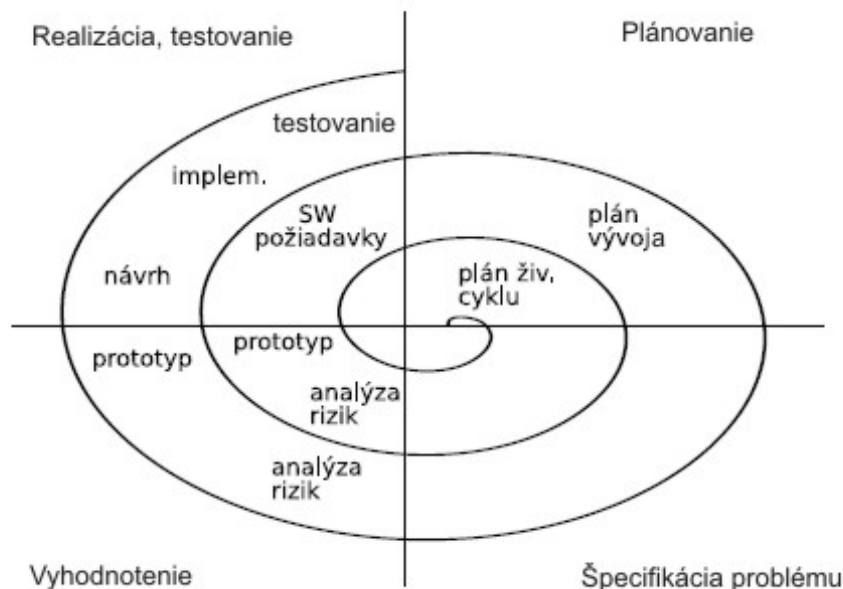
Obrázok 2.2: Model vodopádu

(zdroj: literatúra [2] Ing. Radek Kočí, Ph.D., Ing. Bohuslav Křena, Ph.D.: IUS studijní opora)

Zaujímavé je, že samotné programovanie sa kladie len ako jeden malinký dielik a najväčšiu časť zaberá analýza problému. Čo sa týka testovania, tento model je ideálny - existuje dokonale vypracovaný plán, dokumentácia, software je už „na stole“ - ale ako bolo už spomínané, v prípade nájdenia závažnej chyby už v analýze, znamená, že projekt pôjde do sveta chybný alebo sa začne vývoj od začiatku.

2.3.2 Špirálový model

V mnohých dokonale riadených projektoch je vodopádový model ideálnym riešením. Tam, kde nie je, sa začali vytvárať rôzne alternatívy a medzi nimi vznikla aj tzv. „špirála“. Tento model vznikol niekedy v polovici osemdesiatych rokov a ukázal sa, že je veľmi efektívny. Na začiatku sa vytvoria základné definície, pošlú sa zákazníkovi, ten pošle pripomienky a po ich spracovaní a vyhodnotení sa prechádza na ďalší stupeň vývoja. Model je založený na kombinácii prototypovania a analýzy rizík. Jednotlivé kroky pri vývoji systému sa vo špirále opakujú, ale keď je možné, na vyššom stupni zvládnutej problematiky (Obrázok 2.3). Model je výhodný iba pre skúsené tímy, zvláštne pri veľkých projektoch. Už v časných fázach modelu je pozornosť zameraná na použitie SW. Chyby a nevyhovujúce alternatívy sú vylúčené čo najskôr (vďaka „vyhodneniu“ niektorých prototypov a analyzovania rizík). (zdroj: literatúra [2], [13])



Obrázok 2.3: Špirálový model

(zdroj: literatúra [2] Ing. Radek Kočí, Ph.D., Ing. Bohuslav Křena, Ph.D.: IUS studijní opora)

2.3.3 Evolučný model

Jedná sa o akýsi hybridný model vývoja. V prvom štádiu sa vrstevnatou metódou zisťuje analýza celého projektu a až potom sa s drobnými obmenami prechádza na špirálový model. Teda najprv vytvoríme analýzu. Potom sa navrhujú časti systému - teda architektúra návrhu. Problém sa rozloží do jednotlivých malých „podproblémov“ (napríklad moduly, komponenty) a tie sa začnú vyvíjať špirálou, pozostávajúc z cyklu:

1. Podrobný návrh
2. Plán a rozpočet
3. Zostavenie
4. Integrácia
5. Testovanie
6. Nasadenie

Môžeme naraziť na problém aktuálnosti špecifikácie, keďže prebehla na začiatku. V prípade zmien si ale musíme uvedomiť, že nemeníme moloča, ale jednotlivé komponenty a ak sa na toto nezabudne pri analýze, z tohto problému sa stáva triviálna záležitosť. (zdroj: literatúra [13])

Medzi vývojové metódy by sme ešte mohli zaradiť metódu tunela. Jej princíp je veľmi jednoduchý. Na začiatku vývoja sa vstupuje do neznáma – do čierneho tunela. Vedúci projektu sa snaží operatívnymi zásahmi riadiť projekt a nájsť „svetielko“ na konci tunela. Úlohou je teda nájsť a identifikovať koniec tunela a projekt ukončiť. Pri tvorbe systém je rozdelený na časti, a každý vývojár, ktorý na projekte pracuje, prevedie všetky práce od analýzy po kódovanie.

Táto kapitola sa zaoberala základnými znalosťami o informačných systémoch. Bol tu vysvetlený pojem informačného systému, popísané časti, z ktorých sa skladá informačný systém. Ďalej čitateľ sa dozvedel, aké hlavné fázy obsahuje vývoj celého systému. Pri vývoji môžeme postupovať na základe spomínaných modelov.

3 Implementačné prostriedky

Keď si vezmeme architektúru informačného systému, ktorú môžeme rozdeliť na klientsku, aplikačnú a dátovú vrstvu, je samozrejmé, že implementácia každej vrstvy bude vyžadovať iné implementačné prostriedky. Pod implementáciou týchto vrstiev rozumieme napr. vytvorenie užívateľského rozhrania, cez ktorú bude používateľ komunikovať so systémom, vytvorenie aplikácií, ktoré zabezpečia bezchybné fungovanie celého systému, operácie nad dátami, prepojenie medzi užívateľským rozhraním a databázovým systémom, prístupy k dátam, apod. Aj pri vytvorení tejto bakalárskej práce boli použité viaceré implementačné technológie, ktoré sú nasledujúce:

- jazyk UML
- relačný databázový systém MySQL
- značkovací jazyk XHTML s použitím kaskádových štýlov CSS
- skriptovací jazyk PHP
- skriptovací jazyk JavaScript

Jazyk UML bol použitý pri návrhu systému, pomocou databázového systému MySQL som vytvoril celú databázu, ostatné spomínané implementačné prostriedky slúžili na vytvorenie aplikačných častí systému.

3.1 Jazyk UML

Pri analýze požiadaviek sa často ako nástroj pre komunikáciu medzi zadávateľom a riešiteľom využíva modelovanie. Vznikajú rôzne modely navrhovaného systému, ktoré môžu byť na tomto úrovni testované, overované, upravované. Kvalitný objektovo orientovaný návrh aplikácie je nevyhnutným predpokladom jeho implementácie.

V poslednej dobe UML (Unified Modeling Language) je najpoužívanejším prostriedkom pre objektovo orientovaný návrh. UML je jazykom s bohatou sémantikou a syntaxou, ktorý uľahčuje návrh a vizualizáciu rôznych typov aplikácií. UML je jazyk, ktorý umožňuje modelovať nielen jednoduché, ale aj zložité aplikácie. Model v UML sa skladá z rôznych diagramov, ktoré predstavujú pohľady na rôzne časti sémantického základu navrhovanej aplikácie. Jazyk UML rozoznáva päť základných pohľadov na systém, ktoré sú pohľad prípadu užití, logický pohľad, procesný pohľad, implementačný pohľad a pohľad nasadenia. Jazyk UML rozlišuje tri základné stavebné bloky:

- predmety – do predmetov sú zaradené samotné elementy modelu
- relácie – pomocou ktorých sú vzájomne prepojené jednotlivé predmety
- diagramy – ktoré zachytávajú rôzne aspekty modelovaného systému

Pohľady sú konkretizované v typoch diagramov, ako napr. diagram prípadu užitia, diagram tried, diagram objektov, diagram komponent, apod.

3.2 Značkovací jazyk XHTML

XHTML (Extensible HyperText Markup Language - Rozšíriteľný hypertextový značkovací jazyk) je rodina súčasných a budúcich typov dokumentov a modulov, ktoré rozširujú HTML 4. Rodina typu dokumentov XHTML je založená na XML. Prvým typom dokumentov rodiny XHTML je XHTML 1.0. Ide o preformulovanie HTML 4 ako aplikácia XML 1.0. Vývojári postupne prechádzajú na XHTML, pretože má určité výhody, ako napr.

- dokumenty XHTML sú zhodné s XML, a tak sú jednoducho zobraziteľné, upravovateľné a validovateľné štandardnými XML nástrojmi
- XHTML dokumenty môžu byť napísané tak, že fungujú takisto, alebo lepšie s existujúcimi užívateľskými agentmi s podporou HTML 4
- dokumenty XHTML môžu využívať aplikácie, ktoré pracujú s HTML objektovým modelom dokumentu alebo XML objektovým modelom dokumentu

Jazyk XHTML má tri definície DTD (document type definition), korešpondujúci s definíciami pre HTML 4.01: Strict (striktný), Transitional (prechodná) a Frameset (s podporou frames - rámce). Každá z nich definuje inú sadu XHTML tagov, udalostí atd. Element DOCTYPE nie je súčasťou XHTML dokumentu, nie je elementom XHTML a nemusí mať koncovú značku. Jazyk XHTML je v porovnaní s jazykom HTML oveľa prísnejší.

3.3 Kaskádové štýly – CSS

Kaskádové štýly (CSS - Cascading Style Sheets) vznikli v roku 1994, kedy bol podaný prvý návrh na normu. Kaskádové štýly slúžia na úpravu vzhľadu webových stránok, pričom sa využívajú najmä na formátovanie známych jazykov ako je HTML, XHTML, XML a podobne. CSS je jedinečný spôsob ako formátovať webové stránky, meniť ich vzhľad a zároveň jednoduchým spôsobom oddeliť formu od obsahu. Takže v prvom rade ide o oddelenie formy od obsahu, ale na ďalšej, nemenej dôležitej pozícii sú obrovské formátovacie schopnosti a možnosti. Štýly nám presne poukazujú na to, ako má jednotlivý element vyzerat'.

3.4 Skriptovací jazyk JavaScript

JavaScript je multiplatformný, objektovo orientovaný skriptovací jazyk. Jeho syntax patrí do rodiny jazykov C/C++/Java. Používa sa ako interpretovaný programovací jazyk pre WWW stránky, vkladáný priamo do HTML kódu stránky. JavaScript je klientsky skript. Program v JavaScriptu sa obvykle spustí až po stiahnutí WWW stránky z Internetu, na rozdiel od ostatných interpretovaných programovacích jazykov (napr. PHP a ASP), ktoré sa spúšťajú na strane serveru.

3.5 Skriptovací jazyk PHP

PHP je skriptovací programovací jazyk, určený predovšetkým pre programovanie dynamických internetových stránok. Najčastejšie sa začleňuje priamo do štruktúry jazyka HTML, XHTML, čo je veľmi výhodné pre tvorbu webových aplikácií. PHP je možné používať aj k tvorbe konzolových a desktopových aplikácií.

PHP je v súčasnosti veľmi rozšírená technológia, ktorá umožňuje jednoduché programovanie na strane servera. Stručne môžeme povedať, že skript napísaný v PHP je prevedený na serveri podľa zadaných kritérií a výsledok je odoslaný volajúcemu počítaču takým istým spôsobom, akým sa odosielaajú bežné statické stránky. Syntaxe jazyka kombinuje niekoľko programovacích jazykov (Perl, C, Pascal a Java). PHP je nezávislý na platforme, skripty fungujú bez úprav na mnohých rôznych operačných systémoch. Obsahuje rozsiahle knižnice funkcií pre spracovanie textu, grafiky, práci so súborami, prístup k väčšine databázových serverov (MySQL, ODBC, Oracle, PostgreSQL, MSSQL), podporu internetových protokolov (HTTP, SMTP, SNMP, FTP, IMAP, POP3, LDAP, ...)

PHP je programovací jazyk umožňujúci procedurálne i objektovo orientované programovanie. Znalosť objektovo orientovaného programovania (OOP) môže byť pri práci v PHP výhodou, nie je však nutnou podmienkou. PHP patrí medzi jazyky, kde napríklad nie je nutné dopredu definovať typ premenných, naviac každá premenná môže kedykoľvek zmeniť svoj typ.

3.6 Relačný databázový systém MySQL

MySQL (My Structured Query Language - systém pre riadenie databáze) je relačná databáza typu DBMS (database management system) a vychádza z deklaratívneho programovacieho jazyka SQL. Vďaka svojmu licencie (je Open Source – voľne šíriteľný) a rýchlosti je v poslednej dobe najobľúbenejším systémom. MySQL je malý, rýchly a jednoduchý databázový systém. Databáza MySQL má niektoré obmedzenia, ktoré obsahujú iné rozsiahle databázové systémy. Práve vďaka tomu dosahuje vynikajúcu rýchlosť.

Databáza MySQL je jeden z prvých veľmi rozšírených systémov. Práce s týmto systémom sa dajú využiť v C, C++, Java, Perl, PHP, Python, Tcl, Visual Basic alebo .NET. Najčastejšie sa používa vo spojení s jazykom PHP, ktorý umožňuje prístup k uloženým dátam.

Táto kapitola sa zaoberala implementačnými prostriedkami, ktoré boli použité pri vytváraní informačného systému obce. Pre každú vrstvu architektúry systému boli použité iné implementačné nástroje. Boli tu stručne popísané jednotlivé nástroje (jazyky), ktoré slúžili pre vytvorenie dátovej, aplikačnej a klientskej vrstvy. (zdroj: literatúra [14], [16])

4 Analýza a návrh

Aby výsledný, funkčný systém splnil všetky požiadavky zadávateľa, je potrebné podrobne analyzovať všetky požiadavky na tento systém. Zadanie tejto bakalárskej práce je vytvorenie informačného systému/intranetu pre malú obec. Pred návrhom celého systému som prezeral niekoľko domovských stránok rôznych obcí, aby som mal predstavu, čo sú tie hlavné funkcie, ktoré by mal poskytovať už hotový, nainplementovaný systém. V tejto kapitole podrobne popíšem analýzu vstupných požiadaviek a model navrhnutého systému.

4.1 Požiadavky na systém

Keď sa pokúsime porovnať niekoľko informačných systémov, môžeme hovoriť, že každý je vyvinutý na plnenie určitých funkcií, pričom nájdeme od jednoduchých systémov až veľmi zložité, podrobne prepracované systémy. Aj pri informačných systémoch obcí je to tak. Mojm cieľom bol navrhnuť a implementovať taký systém, ktorý splní všetky základné funkcie, ktoré má poskytovať tento druh informačných systémov, okrem toho má aj nejaké ďalšie, špecifické funkcie. Hlavnou podmienkou bolo to, aby na to oprávnené osoby jednoducho, bez znalosti HTML jazyka mohli zmeniť obsah jednotlivých stránok. Je to dôležité, lebo nemôže každý vedieť programovať v tomto jazyku. Práve preto je potrebné vytvorenie takého užívateľského rozhrania a takých funkcií, ktoré uľahčujú prácu užívateľa so systémom.

Jednotlivé požiadavky sú nasledujúce:

- bezpečnostná politika systému – vytvorenie hierarchie používateľov systému a určovanie, kto aké práva bude mať
- Hierarchia používateľov:
- neregistrovaní užívatelia
 - registrovaní užívatelia
 - zamestnanci obecného úradu
 - správca systému
- registrácia nového užívateľa
 - základné informácie, povinne zverejňované údaje
 - rubrika aktualít
 - úradná tabuľa
 - kalendár akcií
 - fotogaléria
 - dokumenty na stiahnutie

- články, publikácie
- zoznam služieb v obci, registrácia nových firiem
- elektronická podateľňa
- diskusné fórum
- jednoduchý portál futbalistov
- zabezpečiť čiastočnú správu jednotlivých častí systému zamestnancom obecného úradu, úplnú správu pre administrátora systému
- správa zamestnancov pre administrátora

Ako vidíme, do systému budú mať prístup až štyri typy používateľov – neregistrovaní užívatelia, registrovaní užívatelia, zamestnanci obecného úradu a správca systému. Práve preto je potrebné podrobne určiť, kto kam bude mať prístup v rámci informačného systému. Samozrejme táto hierarchia je zostavená podľa prístupových práv, takže neregistrovaný užívateľ má najnižšie práva, kým správca systému bude mať plnú moc nad správou celého systému.

Povolené funkcie jednotlivým užívateľom:

Neregistrovaný užívateľ môže:

- prezerat' povolené stránky systému
- zaregistrovat' sa
- pridať vlastnú firmu do zoznamu firiem (služieb)
- využívat' elektronickú podateľňu

Registrovaný užívateľ môže:

- všetko, čo je povolené neregistrovanému užívateľovi
- prihlásiť sa do systému, odhlásiť sa zo systému
- po prihlásení do systému prezrieť vlastný profil, zmeniť svoje prihlasovacie údaje
- pridať príspevky k jednotlivým akciám v kalendári akcií
- písať do diskusného fóra, pridať novú podskupinu (tému) do diskusných skupín
- pridať malý obrázok („avatar“) do vlastného profilu, ktorý sa zobrazí v diskusnom fóre pri vlastnom príspevku

Zamestnanec obecného úradu môže

- využívat' všetky funkcie, ktoré sú povolené aj registrovaným užívateľom
- **pridať:** novú aktualitu, novú kategóriu obrázkov, fotky do fotogalérie, novú akciu do kalendáre akcií, napísať článok do rubriky Články, novú kategóriu dokumentov, dokumenty do jednotlivých kategórií dokumentov
- upraviť alebo zmazať iba tie položky v jednotlivých rubrikách, ktoré sám pridal

Správca systému

- má pod kontrolou správu celého systému
- môže zmeniť, zmazať všetky položky pridané aj inými užívateľmi systému
- zaručuje správu zamestnancov – pridanie/zmazanie zamestnanca do/z databáze, editácia ich údajov

Okrem týchto typov užívateľov existuje ešte jeden typ, ktorého práva skoro vo všetkom zhodujú s právami registrovaného užívateľa. Je to typ správca futbalistov, ktorý navyše má pod kontrolou správu portálu futbalistov, tzn. môže pridať výsledky zápasov do tabuľky zápasov, pridať nový článok o futbale a samozrejme upraviť alebo zmazať tieto záznamy.

4.2 Návrh aplikácie pomocou jazyka UML

Ako som to už spomenul v kapitole 3.1, vytvorenie kvalitného modelu systému je veľmi dôležitou časťou celého návrhu. V dnešnej dobe pre objektovo orientovaný návrh je najpoužívanejší jazyk UML. Pri návrhu tohto systému som taktiež použil spomínaný jazyk. Pomocou UML som vytvoril dva typy diagramov, podľa ktorých bol potom celý systém implementovaný. Tie diagramy sú:

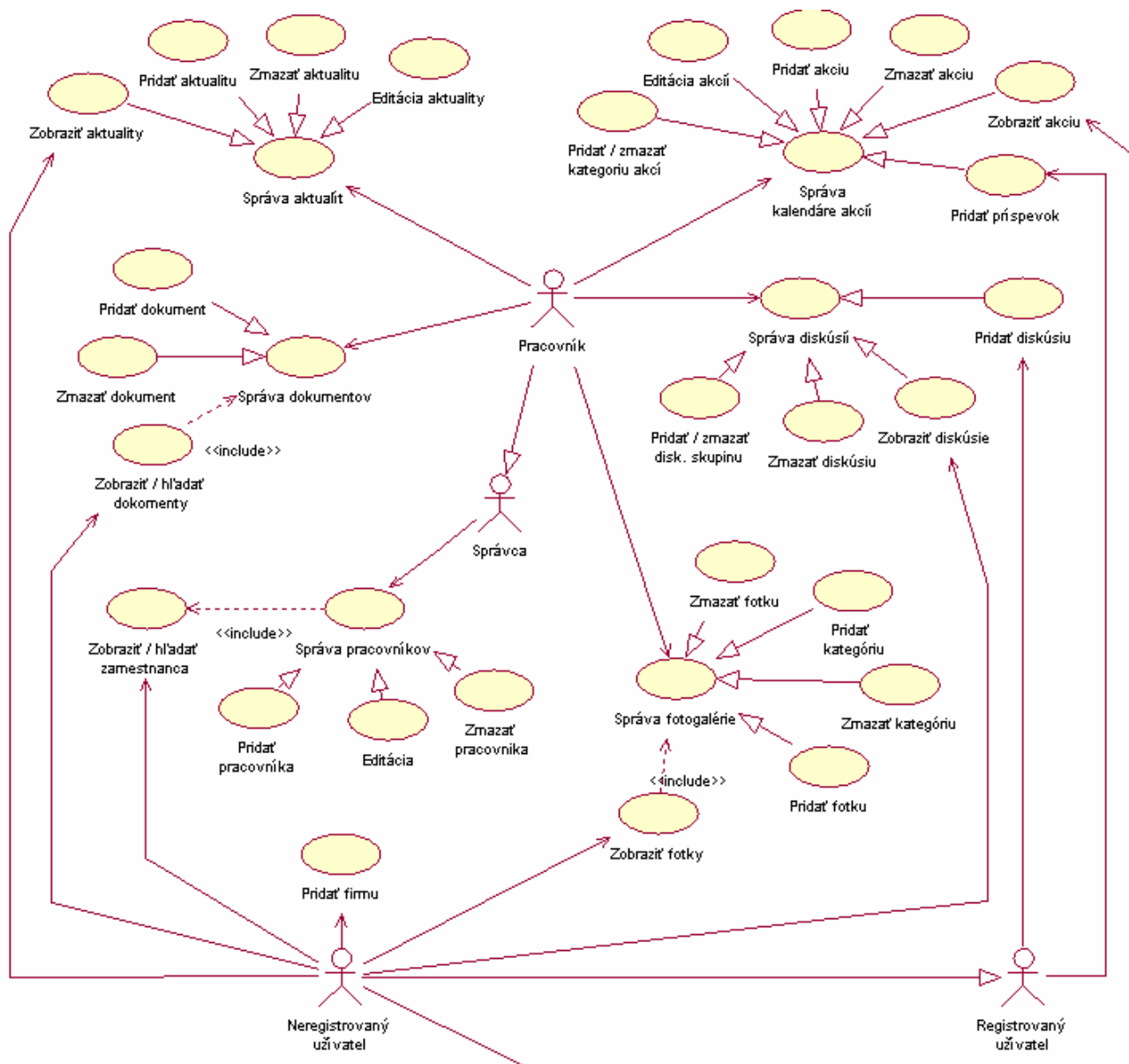
- diagram prípadov užitia – USE CASE diagram
- a diagram tried – ER diagram

Na základe týchto diagramov prebehlo vytvorenie databázy a implementácia jednotlivých funkcií systému.

4.2.1 Diagram prípadov užitia

Pri tvorbe informačných systémov hrá nezastupiteľnú úlohu USE CASE model. Diagramy prípadov užitia nám poskytujú rýchlu predstavu o jednotlivých funkciách systému. Diagramy užitia znázorňujú hranice navrhovaného/skúmaného systému, účastníkov analyzovaných prípadov užitia, analyzované prípady užitia a interakcie medzi aktérmi a prípadmi užitia. Samotné diagramy prípadov užitia ukazujú iba to, že aké akcie je možné v systéme vykonávať, a kto ich môže vykonávať.

Diagram prípadov užitia pre informačný systém obce Dolnozemska je znázornený na obrázku 4.1. Z tohto diagramu vidíme, aké aktéri sa vyskytujú v našom systéme, ďalej z diagramu jednoducho môžeme vyčítať, ktorý aktér ktorú funkciu systému môže používať. Vidíme teda hierarchiu, ktorá je vytvorená na základe prístupových práv. Napríklad kým neregistrovaný užívateľ môže iba prezerať jednotlivé časti systému, registrovaný užívateľ už má právo písať do diskusného fóra, alebo pridať príspevok k jednotlivým akciám. Samozrejme správca systému má najvyššie práva, a tak má prístup ku každej funkcii, ktorú systém poskytuje.



Obrázok 4.1: Diagram prípadov užívania

To, že ktorému aktérovi čo je vlastne povolené v rámci systému, je podrobne popísané v kapitole 4.1.

4.2.2 ER diagram – diagram tried

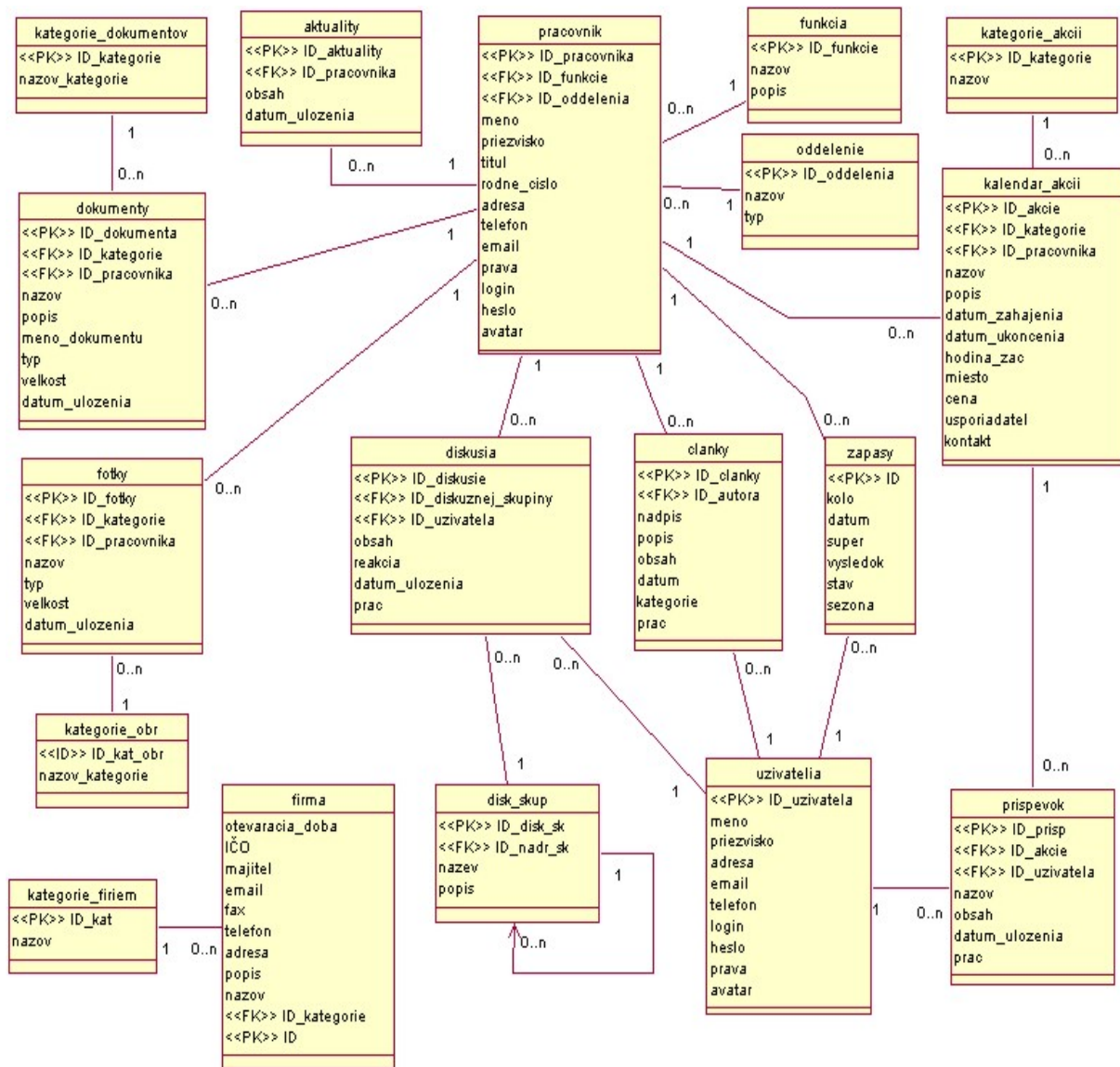
Najznámejšou a najčastejšie používanou modelovacou technikou pre návrh relačných databáz je entitne-vzťahové modelovanie, jeho výsledkom je entitne-vzťahový diagram (ER diagram alebo ERD). Pretože vytváraný model a odpovedajúci diagram slúži nielen ako podklad pre návrh databázy, ale takisto pre komunikáciu so zákazníkom, pre ktorého systém vytvárame, musí byť zrozumiteľný

a množstvo informácií, ktoré pre pochopenie nie je možné vyčítať z ER diagramu, by malo byť čo najmenšie.

Na obrázku 4.2 je uvedený ER diagram môjho informačného systému. Obsahuje 18 entít, ktoré sú nasledujúce:

- **pracovník** – reprezentuje tabuľku, v ktorej sú uložené všetky dôležité informácie o zamestnancoch. Primárnym kľúčom je ID, ktorý jednoznačne určuje daného pracovníka. Obsahuje cudzí kľúč ID_funkcie, v ktorom je uložený ID funkcie. Medzi entitnými množinami pracovník a funkcia je vzťah s kardinalitou 0..n : 1, čo znamená že každý pracovník musí mať nejakú funkciu, ďalej že tú istú funkciu môže plniť aj niekoľko pracovníkov, ale nemusí ani jeden. S entitnými množinami aktuality, dokumenty, fotky a kalendar_akcii je pracovník vo vzťahu s kardinalitou 1 : 0..n, čo znamená že napr. môže pridať viac aktualít, ale nemusí ani jednu.
- **funkcia** – táto entitná množina určuje názov a popis jednotlivých funkcií, ktoré môžu mať zamestnanci obecného úradu
- **oddelenie** – definuje, do ktorého oddelenia patrí zamestnanec obecného úradu
- **užívateľ** – primárnym kľúčom je ID a táto entitná množina obsahuje všetky potrebné atribúty o registrovaného užívateľa. S entitnými množinami príspevo a diskusia je vo vzťahu s kardinalitou 1 : 0..n.
- **kalendar_akcii** – reprezentuje tabuľku v relačnej databáze, do ktorého budú uložené údaje o jednotlivých akciách. Primárnym kľúčom je ID, a obsahuje dva cudzie kľúče – ID_kat a ID_prac, ktoré určujú, do ktorej kategórie patrí akcia a ktorý zamestnanec uložil do databázy.
- **katégorie_akcii** – obsahuje názvy jednotlivých kategórii akcií. ID je primárnym kľúčom.
- **príspevok** – ku každej akcie, ktorá je v kalendári akcií, registrovaný užívateľ môže pridať príspevok. Táto entitná množina obsahuje všetky údaje príspevku. Primárnym kľúčom je ID, ďalej obsahuje cudzie kľúče ID_akcie a ID_uziv, ktoré jednoznačne určujú, ku ktorej akcii patrí uložený príspevok, a kto je autorom. Nájdeme tu atribút prac, ktorý slúži na to, aby sme vedeli určiť, či príspevok pridal registrovaný užívateľ, alebo pracovník obecného úradu.
- **aktuality** – je to tabuľka v databáze, ktorá obsahuje atribúty jednotlivých aktualít, ako napr. obsah aktuality, dátum uloženia. Atribút ID jednoznačne určuje danú aktualitu, je to teda primárnym kľúčom, a ID_prac je cudzí kľúč, ktorá obsahuje ID pracovníka. S entitnou množinou pracovník je vo vzťahu s kardinalitou 0..n – 1, čo znamená, že každá aktualita musí mať autora, ale nie každý pracovník musí mať uloženú aktualitu.
- **dokumenty** – primárnym kľúčom tejto množiny je ID. Každý zamestnanec obecného úradu môže uložiť nejaký dokument, ktorý bude patriť do jednej kategórie z kategórií dokumentov – je to určené cudzím kľúčom ID_kat. Do databázy uložíme všetky potrebné informácie o jednotlivých dokumentoch.

- **kategorie_dok** – určuje kategóriu dokumentov. Primárnym kľúčom je ID. Do tej istej kategórie dokumentov môže patriť aj viac dokumentov, ale nemusí ani jeden. To vyjadruje vzťah s kardinalitou 1 - 0..n.
- **fotky** – informácie o uložených fotkách vo fotogalérii obsahuje táto tabuľka v databáze. Atribut ID jednoznačne určuje fotku, kým cudzie kľúče ID_prac a ID_kat určujú, kto uložil danú fotku, a do akej kategórie patrí. S obidvoma entitnými množinami (pracovník a kategorie_obr) je vo vzťahu s kardinalitou 0..n – 1.
- **kategorie_obr** – určuje ID (primárny kľúč) a názov galérií obrázkov
- **diskusia** – na to oprávnené osoby môžu písať do diskusného fóra, a pridané názory sú uložené do tejto tabuľky databáze. Obsahuje ID autora a ID diskusnej skupiny, ktoré sú cudzími kľúčmi. Jedným atribútom tejto entitnej množiny je reakcia, ktorá určuje, či je príspevok reakciou na iný názor, a keď áno, tak na ktorý. Ďalej je dôležitý aj atribút prac, podľa ktorého je možné zistiť, či autorom názoru je registrovaný užívateľ, alebo zamestnanec. Je to dôležité, pretože pri výpise názorov musíme vedieť, či meno autora máme hľadať v databáze pracovníkov, alebo užívateľov.
- **disk_skup** – určuje jednotlivé diskusné skupiny, do ktorých môžu písať užívatelia diskusného fóra. Ako vidíme z ER diagramu, táto entitná množina je vo vzťahu s kardinalitou 1 : 0..n sám so sebou. To znamená, že diskusná skupina môže mať nadradenú skupinu, ale nemusí. Takto môžeme vytvoriť hierarchiu diskusných skupín (sú teda rozlíšené hlavné skupiny a podskupiny)
- **clanky** – do rubriky „Články“ môžu pridať oprávnené osoby rôzne články, publikácie. Obsah a ďalšie potrebné údaje sú potom uložené do tejto tabuľky databázy. Jednoznačné identifikačné číslo je uložené v primárnom kľúče ID, v cudzom kľúče je uložený ID autora. Aj tu existuje atribút prac, ktorý slúži na to isté, ako pri príspevkoch.
- **firma** – obsahuje všetky povinné a nepovinné údaje o firmách/službách, ktoré môžu užívatelia systému zaregistrovať pomocou registračného formulára. Obsahuje cudzí kľúč ID_kat, ktorý určuje, do ktorej skupiny patrí daná položka.
- **kategorie_firem** – určuje ID a názov kategórií služieb
- **zápasy** – správca alebo správca futbalistov môže pridať výsledky zápasov do portálu futbalistov. Potrebné informácie sú potom uložené do tejto tabuľky databázy.



Obrázok 4.2: ER diagram systému

Táto kapitola sa týka analýzy a návrhu informačného systému. Na začiatku kapitoly sú podrobne rozpísané požiadavky na systém. Sú tu zhrnuté všetky funkcie, ktoré by mal systém poskytovať. Nájde tu aj hierarchiu používateľov, ktorí budú mať prístup do systému. Pre každý typ užívateľov je stanovené, ktoré funkcie majú právo využívať. Posledná časť tejto kapitoly opíše samotný návrh systému pomocou jazyka UML. Na základe navrhnutého Use case diagramu a ER diagramu bude potom prebiehať implementácia celého systému, ktorá je témou ďalšej kapitoly.

5 Implementácia systému

Táto kapitola obsahuje podrobný popis implementácie jednotlivých častí systému. Ďalej sú tu popísané jednotlivé funkcie, ktoré systém poskytuje užívateľom. Implementácia systému sa začala vytvorením celej databázy podľa návrhu, potom nasledovalo vytvorenie užívateľského rozhrania, ktoré zabezpečuje komunikáciu užívateľa so systémom, a nakoniec som implementoval všetky časti systému podľa požiadaviek.

5.1 Vytvorenie databázy

Databázu môžeme predstaviť ako súbor dát, ktoré slúžia pre popis reálneho sveta. Entita je prvok reálneho sveta, ktorý je popísaný svojimi vlastnosťami – tie môžeme považovať za atribúty. Dôležité sú ešte väzby medzi reláciami. Pri návrhu relačného modelu (ER diagram) už sme sa stretli s tými istými výrazmi. Je to samozrejmé, pretože pomocou tohto modelu sme navrhli štruktúru celej databázy. Tento konceptuálny model je už iba potrebné pretransformovať na tabuľky relačnej databázy.

Tabuľka nám popisuje nejakú entitu. Skladá sa zo stĺpcov, ktorým hovoríme atribúty. Databázový systém vyžaduje pri tvorbe databáze definíciu typu informácie, ktorú budeme do príslušného stĺpca ukladať. Existujú tri primárne kategórie - text, čísla, dátum a čas – z ktorých môžeme vyberať. Pre každý atribút som si vybral čo najvhodnejší typ, a nastavil som maximálnu dĺžku textových a číselných hodnôt. Vo väčšine prípadov, kde sa bude v jednotlivých záznamoch líšiť počet znakov, som použil typ VARCHAR, pretože hodnoty ukladané do stĺpcov typu VARCHAR sú vždy len tak dlhé, ako keby boli uložené samostatne. Tam, kde je potrebné uložiť viac znakov, ako 255, je potrebné používať typ TEXT. Pre primárne a cudzie kľúče – teda pre jednotlivé identifikačné čísla som si zvolil typ INT. Tam, kde je potrebné uložiť nielen dátum, ale aj čas, som použil typ DATETIME, inak stačí aj typ DATE.

Už pri návrhu tabuliek môžeme rozhodnúť o určitých obmedzeniach na jednotlivé položky. Hodnoty do našich vytvorených tabuliek budú spravidla vkladáť koncoví užívatelia. Na úrovni SQL serveru je možné zaistiť niektoré požiadavky, ktoré sú na jednotlivé položky kladené. Napr. môžeme požadovať, aby určité položky boli vždy zadané. Ďalej môžeme chcieť, aby nejaká položka mala jedinečnú hodnotu (väčšinou takou položkou bude primárny kľúč), alebo mala nejakú hodnotu defaultnú, v prípade, že ju užívateľ nezadal. Tieto obmedzenia nazývame integritnými obmedzeniami. Poznáme niekoľko typov integritných obmedzení, ako napr:

- **DEFAULT** <hodnota> – znamená, že sa použije uvedená hodnota, keď užívateľ nezadá žiadnu hodnotu
- **NOT NULL** - hodnota musí byť povinne zadaná
- **PRIMARY KEY** – primárny kľúč, jeho hodnota musí byť jedinečná
- **UNIQUE** – jedinečná hodnota

Väzby medzi tabuľkami sú vyriešené pomocou cudzích kľúčov (**FOREIGN KEY**). Pomocou klauzule **ON DELETE CASCADE** je zabezpečené kaskádne zmazanie záznamov s rovnakou hodnotou cudzieho kľúča.

Po určení všetkých potrebných integritných obmedzení potom pomocou príkazu **CREATE TABLE** boli vytvorené všetky tabuľky.

5.2 Užívateľské rozhranie – vzhľad systému

Komunikácia medzi užívateľom a systémom je zabezpečená pomocou užívateľského rozhrania. Dobre navrhované užívateľské rozhranie umožňuje, že sa užívateľ rýchlo a bez akýchkoľvek ťažkostí naučí pracovať s daným systémom. Určite sa už každý stretol s takými internetovými stránkami, v ktorých sa na prvý pohľad vôbec nevyznal. Keď je napríklad stránka veľmi preplnená, môže veľmi dlho trvať, kým nájdeme to, čo vlastne hľadáme. Preto je dôležité, aby stránka vyzerala esteticky, jednotlivé prvky boli usporiadané tak, aby užívateľ na prvý pohľad videl, ktoré veci kde nájde, čiže každý, kto využíva systém, čím lepšie sa zorientoval na stránke. Samozrejme aj mojím cieľom bolo vytvorenie takého užívateľského rozhrania.

Štruktúra celého systému sa skladá z viacerých stránok. Určité časti však sú zobrazené na všetkých stránkach (pod týmito časťami rozumieme napr. menu, obrázkov na pozadí v záhlaví, alebo rubriky aktualít a užitočných odkazov). Aby sme nemuseli do kódu každej stránky vložiť ten istý kód, štruktúru jednotlivých stránok som rozdelil na tri časti. Kód záhlavia a ľavej časti spolu s menu je umiestnený v súbore `head.html`, kód pravej a spodnej časti je uložený do súboru `body.html`. Takže html kód stránky sa začína v súbore `head.html`, a končí v `body.html`. Samotné telo stránky je potom umiestnený v jednotlivých php súboroch, do ktorých je potrebné pomocou funkcie `include()` zahrnúť spomínané dva súbory.

Samotný vzhľad systému je nasledovný: na ľavej časti sa nachádza menu, pomocou ktorého užívateľ sa môže jednoducho pohybovať medzi jednotlivými stránkami. V niektorých prípadoch po kliknutí na položku v menu sa zobrazí podmenu. Pod menu je umiestnený prihlasovací formulár. Na ľavej časti sú ešte odkazy na stránky registrácie nového užívateľa a na zabudnuté heslo. Po prihlásení užívateľa do systému prihlasovací formulár a spomínané odkazy už nie sú zobrazené, objaví sa tu formulár na odhlásenie. Pravá časť sa skladá z dvoch hlavných blokov – je tu umiestnený blok (rubrika) aktualít, kde vždy sú zobrazené tri najnovšie aktuality. Pod tým je blok obsahujúci užitočné odkazy na rôzne stránky. Stredná časť stránky je dynamická, jej obsah sa mení na základe

toho, ktorú položku sme vybrali z menu. Ešte by som pripomenul blok času a dátumu, ktorý je pod obrázkom v záhlaví, a ukazuje vždy presný dátum a čas, okrem toho je tu vypísané, kto má meniny. Na obrázku 5.1 vidíme celkový vzhľad

Po prihlásení zamestnanca alebo administrátora systému sa vzhľad užívateľského rozhrania výrazne nemení. Sú však nejaké odlišnosti, napr. menu obsahuje viac položiek, alebo na niektorých stránkach pravá časť s aktualitami a užitočnými odkazmi sa neobjaví.



Obrázok 5.1: Úvodná stránka systému

Náš systém poskytuje užívateľom diskusné fórum, ktoré však vyzerá trochu inak, ako predchádzajúce stránky. Pravá časť tu nie je zobrazená a obsahuje ešte jedno menu, ktoré je umiestnené v záhlaví stránky. Po prihlásení užívateľa v tomto menu sa objavujú iné položky, a je zobrazený aj meno prihláseného užívateľa.

5.3 Registrácia nového užívateľa

Náš systém je určený pre všetkých, ktorí chcú získať nejaké informácie o obci. Byť prihlásený do systému nie je podmienkou. Registráciou si však môžeme zriadiť užívateľský účet, do ktorého sa môžeme prihlasovať pomocou užívateľského mena a hesla. Získame tým možnosť využívať nové služby internetových stránok určené len registrovaným občanom. Po úspešnej registrácii užívateľ sa

stane registrovaným užívateľom, a po prihlásení môže pridať príspevky k jednotlivým akciám v kalendári akcií, alebo písať do diskusného fóra. Registrácia je veľmi jednoduchá, stačí iba vyplniť registračný formulár. Tento formulár nájdeme po kliknutí na odkaz Registrácia nového užívateľa, ktorý je umiestnený pod prihlasovacím formulárom, v ľavej časti stránky.

Registračný formulár je rozdelený na dve časti, takže užívateľ tu vidí dve tabuľky. V časti „Údaje“ užívateľ má zadať prihlasovacie meno a vlastné údaje, ktoré môžeme rozdeliť na povinné a nepovinné údaje. Povinné údaje sú označené hviezdíčkou. Do druhej časti je potrebné zadať heslo, s ktorým sa bude prihlásiť do systému, a to dvakrát, aby bolo potvrdené, že heslo zadal správne.

Po kliknutí na tlačidlo Registrovať systém skontroluje zadané údaje. Určité údaje musia mať preddefinovaný tvar, napr. PSČ povinne musí obsahovať 5 číslic, alebo emailová adresa musí byť vo správnom formáte. To všetko je kontrolované pomocou regulárnych výrazov. Veľmi dôležitá je kontrola prihlasovacieho mena. To musí byť v systéme unikátne, čo znamená, že nemôžu mať dvaja užívatelia rovnaké prihlasovacie meno, pretože by systém potom nevedel rozlíšiť, ktorý užívateľ je prihlásený. Okrem testovania unikátnosti je ešte potrebné overiť, či je prihlasovacie meno vo správnom formáte. Musí mať minimálne 4, maximálne 20 znakov, a môže obsahovať iba písmená bez diakritiky, číslice a podtržítka. Unikátnosť je potrebné overiť aj u emailovej adresy, pretože systém poskytuje službu Zabudnuté heslo, ktorá odosiela prihlasovacie meno a nové heslo na tú emailovú adresu užívateľa, ktorú zadal pri registrácii. Aj zadané heslá sú testované, takisto je zadané, aké znaky môže obsahovať, a musí sa zhodovať s heslom, ktoré máme zadať do kolónky Potvrdiť heslo. Keď nejaký údaj nie je správne zadaný, nad týmto údajom sa objaví chybové hlásenie.

Keď všetky údaje sú správne zadané, údaje užívateľa sú uložené do databázy, a pomocou prihlasovacieho mena a hesla sa už môže prihlásiť do systému.

5.4 Zabudnuté heslo

Aby sa užívateľ mohol prihlásiť do systému, musí správne zadať prihlasovacie meno a heslo. Môže sa však stať, že zabudne svoje heslo. Pomôckou preňho môže byť funkcia systému Zabudnuté heslo. Na mnohých internetových stránkach existuje táto funkcia, ktorá je veľmi užitočná. Vo väčšine prípadov pri registrácii je zadaná nejaká kontrolná otázka, na ktorú má užívateľ zadať odpoveď. Keď náhodou zabudne svoje heslo, práve táto kontrolná otázka bude mu zadaná, a po zadaní správnej odpovedi má možnosť zmeniť svoje heslo. Ja som túto funkciu vyriešil trochu jednoduchšie, ale z hľadiska funkčnosti takisto je to funkčný príklad.

Po kliknutí na odkaz Zabudli ste heslo? na stránke sa objaví jeden malý formulár, do ktorého užívateľ musí zadať iba jediný údaj – emailovú adresu, ktorú zadal pri registrácii. Problémom však môže byť, že sa ani na to nepamätá. V tomto prípade jedinou možnosťou dostať sa do systému je nová registrácia. V prípade, že sa na emailovú adresu pamätá, má to zadať do príslušnej kolónky. Po odoslaní formulára systém skontroluje zadanú emailovú adresu. Je tu testované nielen to, či je

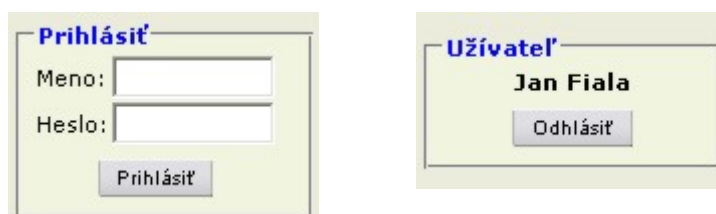
syntakticky správne zadané, ale aj to, či vôbec existuje registrovaný užívateľ s takou emailovou adresou. Keď dôjde k nejakej chybe, príčina chyby je vypísaná užívateľovi. Keď emailová adresa je správna, prihlasovacie údaje sú odoslané na túto adresu.

Vlastná funkcia odosielania údajov sa spočíva v tom, že je vyhľadané z databázy užívateľské meno toho užívateľa, ktorého emailová adresa sa zhoduje so zadanou adresou. Potom je vygenerovaný 8 miestny kód – nové heslo – ktorý sa skladá z čísiel a písmen anglickej abecedy. Prihlasovacie meno a heslo je potom odoslané na zadanú emailovú adresu a v databáze je obnovené heslo.

5.5 Prihlasovanie a odhlasovanie

Predtým, ako popíšem postup prihlasovania, pripomenul by som pár slov o reláciách (session), ktoré je potrebné použiť. Session umožňuje presnú identifikáciu užívateľa a pohyb na serveri. Používa sa na monitorovanie alebo overovanie užívateľa. Session pracuje na jednoduchom princípe. Každý užívateľ, ktorý vstúpi na stránku, dostane jedinečné číslo SID (Session ID). Nemusí obsahovať iba číslice, často sa v ňom vyskytujú aj písmená anglickej abecedy. Týmto číslom sa teda bude užívateľ identifikovať, tzn. vo všetkých odkazoch je potrebné predávať tento parameter. PHP implicitne predáva SID v premennom PHPSESSID. Keď užívateľ navštívi stránku, kde je použitý session, PHP automaticky otestuje existenciu relácie. Keď neexistuje, sú vytvorené všetky premenné a relácie. To je možné previesť pomocou funkcie session_start(). Premenné môžu byť uložené ako záznamy v globálne združenom poli \$HTTP_STATE_VARS alebo ako globálne premenné. Každá relácia má svoju životnosť. Relácia končí, keď užívateľ korektné opustí stránku (napr. sa odhlási), alebo vyprší jej časový limit. Je zrejmé, že na serveri musí bežať démon, ktorý kontroluje aktívne relácie a zmaže už neaktívne.

Ako som už spomínal, niektoré služby systému majú právo používať iba registrovaní užívatelia, alebo zamestnanci obecného úradu. Prihlásiť sa môžu pomocou prihlasovacieho formulára (obrázok 5.2), ktorý je umiestnený pod menu. Po zadaní prihlasovacích údajov a kliknutí na tlačidlo Prihlásiť systém skontroluje zadané údaje. Najprv podľa zadaných údajov hľadá v tabuľke uzivatel, a keď sa nenájde záznam, tak sa pokúsim nájsť záznam v tabuľke pracovník. Keď sa nenájde ani v jednej, zadané údaje nie sú správne, a do prihlasovacieho formulára je vypísané chybové hlásenie.



The image contains two side-by-side screenshots of a web application interface. The left screenshot shows a login form titled 'Prihlásiť' (Login). It has two input fields: 'Meno:' (Name) and 'Heslo:' (Password). Below the fields is a button labeled 'Prihlásiť'. The right screenshot shows a user profile titled 'Užívateľ' (User). The name 'Jan Fiala' is displayed, and below it is a button labeled 'Odhlásiť' (Logout).

Obrázok 5.2: Prihlasovací a odhlasovací formulár

Keď zadané údaje boli správne, prihlasovanie je úspešné a do superglobálneho poľa `$_SESSION` uložíme niektoré údaje o užívateľovi, ktoré budeme potrebovať pri ďalších operáciách. Sú to identifikačné číslo, meno a priezvisko a práva užívateľa. Podľa týchto údajov vždy budeme vedieť, či je prihlásený jednoduchý užívateľ, zamestnanec obecného úradu, či správca systému. Ako som to už spomínal, aby sme mohli používať užívateľské relácie – session – musí byť spustená funkcia `session_start()`. Túto funkciu volám v súbore `head.html`.

Pri odhlásení zo systému musíme zabezpečiť, aby všetky relácie boli odstránené. Môžeme to vykonať funkciou `unset()`, ale pomocou tejto funkcie môžeme odstrániť naraz iba jednu reláciu. Preto je užitočnejšie používať príkaz `$_SESSION = array();` ktorý vynuluje celé pole. Nakoniec vymažeme všetky dáta relácie príkazom `session_destroy()`.

Do systému je možné prihlásiť aj zo stránky diskusného fóra. Princíp prihlasovania je rovnaký s uvedeným princípom.

5.6 Aktuality

Celý systém sa skladá z viacerých stránok, preto je ťažké zistiť, keď niektorá z nich bola aktualizovaná. Aby sme nemuseli prezerat' všetky stránky, a hľadať, či je nejaká zmena alebo nie, užitočné môže byť pre užívateľov rubrika aktualít. Na túto stránku je možné dostať cez menu, po kliknutí na položku Aktuality. Ale na novinky nás upozorní aj malý blok aktualít na pravej časti stránky, do ktorej sú vypísané posledné tri aktuality. Pod nimi nájdeme odkaz na všetky aktuality. Do tejto rubriky môžu písať iba na to oprávnené osoby – zamestnanci obecného úradu a správca systému. Keď pridajú napr. nejaký dokument medzi stiahnuteľných súborov, alebo novú akciu do kalendára akcií, a o zmene napíšu krátku správu do aktualít, každý návštevník systému na prvý pohľad bude vidieť, že na ktorej stránke sa stala aktualizácia. Okrem tohto príkladu táto rubrika je určená aj na oznámenie kratších informácií, ktoré môžu byť pre občanov dôležité.

Aktuality sú na stránke radené podľa dátumu uloženia, od najnovších k najstarším. Užívateľ pomocou výberového poľa môže vybrať, koľko aktualít chce zobrazit' na stránke. Keď počet aktualít je väčší od počtu zobrazených na jednu stránku, pod zoznamom sa objavia odkazy na ďalšie, resp. predchádzajúce stránky.

Keď sa prihlási správca alebo zamestnanec obecného úradu, v menu sa objaví podmenu s položkami Zoznam aktualít a Pridať aktualitu. Títo užívatelia majú právo na pridanie nových aktualít. Keď chcú pridať novú aktualitu, objaví sa jeden formulár, do ktorého je potrebné zadať iba obsah aktuality. Keď niekto chce odoslať nevyplnený formulár, systém vypíše chybové hlásenie, inak je nová aktualita uložená do databázy.

Uložené aktuality je možné editovať alebo zmazať. Správca systému samozrejme môže upraviť alebo vymazať všetky položky, zamestnanec tieto operácie môže vykonať iba nad vlastnými aktualitami. Odkazy na editáciu a zmazanie sú zobrazené pri každej aktualite (v prípade zamestnanca

iba pri vlastných). Pri editácii sa zobrazí ten istý formulár, ktorý je pri pridaní, ale tu sa zobrazí aj pôvodný obsah aktuality, ktorú potom je možné zmeniť a uložiť. Keď chceme vymazať aktualitu, najprv sa objaví okienko, či chceme naozaj vymazať položku, a iba po kliknutí na Ok bude položka vymazaná.

5.7 Fotogaléria

Súčasťou skoro každej domovskej stránky rôznych obcí je fotogaléria. Môžeme v nej nájsť rôzne fotky o obci, o významných budovách alebo pamätihodnostiach. Keď sa dialo nejaké kultúrne alebo športové podujatie, takisto vo fotogalérii môžeme hľadať o týchto udalostiach obrázky. Aj súčasťou nášho informačného systému je fotogaléria.

Na stránku fotogalérie sa môže dostať užívateľ z hlavného menu. Celkový vzhľad tejto rubriky je nasledujúci: vo dvoch stĺpcoch sú vymenované názvy kategórií, ktoré sa nachádzajú vo fotogalérii. Okrem názvu je tu vypísaný aj počet obrázkov, ktoré sa nachádzajú v jednotlivých kategóriách. Kategórie sú radené podľa dátumu posledného aktualizovania. Pod názvami kategórií nájdeme takisto výpis kategórií, ale v inej podobe. Vidíme tu podrobnejšie informácie o jednotlivých galériách. Užívateľovi je zobrazený popis galérie, počet obrázkov v danej kategórii, ďalej celková veľkosť galérií v MB-och, a presný dátum a čas poslednej aktualizácie. Okrem toho je ešte zobrazená aj jeden obrázok z danej kategórie. Radenie je to isté, naposledy aktualizovaná galéria sa nachádza na prvom mieste. Po kliknutí na obrázok, alebo na názov galérie sa dostaneme do vybranej fotogalérie, kde sú zobrazené všetky fotky. Obrázky je potom možné zobraziť v pôvodnej veľkosti.

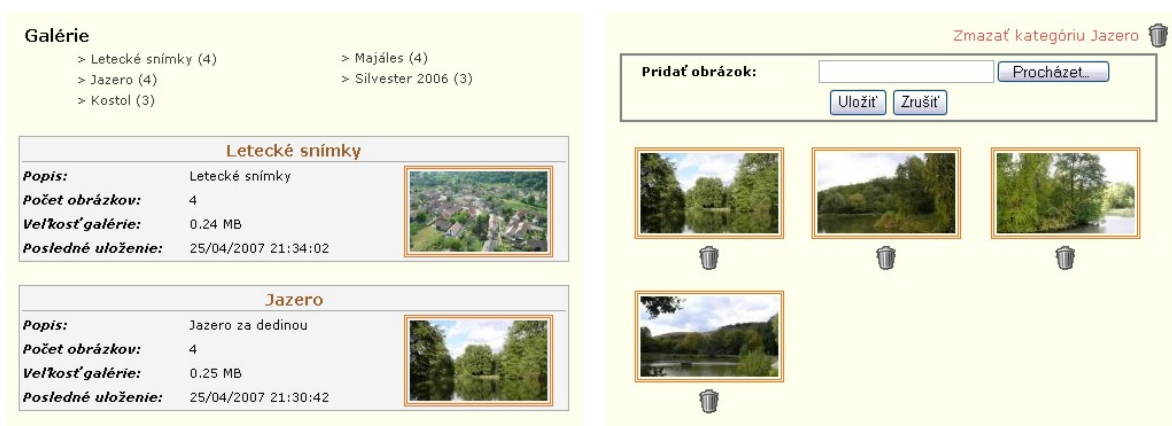
Právo na rozširovanie daných galérií majú iba zamestnanci obecného úradu a správca systému. Chcel som s tým zabezpečiť, aby na stránkach sa neobjavili také fotografie, ktoré tam nepatria. Keď je prihlásený administrátor alebo zamestnanec, nad kategóriami sa objaví odkaz na pridanie novej kategórií. Po kliknutí na odkaz sa zobrazí jeden formulár, kde je potrebné zadať názov kategórie. To je povinný údaj, keď bez názvu chce niekto pridať kategóriu, systém to nedovolí, a vypíše chybové hlásenie. Do formuláru je možné zadať aj popis galérie, ale to nie je povinné. Po úspešnom založení novej kategórie je možné priložiť obrázky.

Keď chceme pridať nový obrázok, pomocou vstupného poľa input typu file môžeme vybrať daný obrázok. Po odoslaní formuláru systém skontroluje vybraný súbor. Najprv je skontrolované, či zadaný súbor má správny typ. Sú povolené iba formáty obrázkov jpeg, bmp, png, gif. Keď chceme uložiť nejaký iný typ, systém to nedovolí, a vypíše chybové hlásenie. Obrázky sú uložené na serveri v adresári galerie, kde pre každú galériu je vytvorený samotný podadresár s názvom galX, kde X je ID kategórie. Keď pri vytvorení zložky dôjde k nejakej chybe, obrázok nebude uložený, a je vypísané chybové hlásenie. Keď všetko je v poriadku, do databázy pridáme nový záznam a získame ID tohto záznamu. Potrebujeme to na vytvorenie názvu pridanej fotky, pretože do príslušného podadresára

uložíme pod názvom `img_x.typ` (napr `img_16.jpg`). Obrázok je potom uložený na server, a do databázy uložíme názov a typ (príponu) fotky.

Hoci na stránke sú zobrazené fotky vo veľkosti maximálne 120 x 90px, keby sme pri zobrazení iba nastavili šírku a výšku obrázkov na tieto hodnoty, pri stiahnutí stránky by veľmi dlho trvalo, kým načítal všetky obrázky. Napriek tomu, že vidíme iba malý obrázok, pôvodná veľkosť obrázku ostane nezmenená (ktorá môže byť aj niekoľko MB-ov). Preto som napísal krátku funkciu, ktorá „prevzorkuje“ daný obrázok, teda z pôvodnej fotky vytvorí malý obrázok. To nahráme na server, a na stránke bude potom zobrazený tento malý obrázok. Názvy obrázkov majú tvar `imgX_1.pripona` (napr. `img26_1.jpg`).

Z galérie je možné odstrániť fotky po jednom, alebo môžeme vymazať celú galériu. Z databázy sú potom odstránené jednotlivé záznamy, ale predtým zo servera sú postupne vymazané všetky obrázky, ktoré patrili do danej kategórie. Správca má právo vymazať všetky obrázky, a každú galériu, zamestnanec obecného úradu môže odstrániť iba tie fotografie, ktoré sám uložil.



a) Fotogaléria s kategóriami b) vybraná galéria spolu s formulárom na pridanie nových obrázkov

Na obrázku 5.3 je zobrazený vzhľad fotogalérie, spolu s jednotlivými kategóriami, ďalej jedna samotná galéria, kde nájdeme formulár, pomocou ktorého je možné pridať nové obrázky. Kliknutím na horný odkaz môžeme vymazať celú galériu. Kliknutím na kôš môžeme odstrániť vybranú fotku.

5.8 Kalendár akcií

V kalendári akcií užívateľa systému môžu nájsť zoznam akcií, ktoré sú pripravené pre občanov, ale aj archív predošlých akcií. Jednotlivé akcie sú triedené podľa kategórií, a tak si každý môže vybrať iba tie kategórie, o ktoré má záujem. Návštevníci stránky dostanú podrobné informácie o jednotlivých pripravovaných akciách, a pre registrovaných užívateľov je pripravená aj rubrika príspevkov, do ktorého môžu napísať vlastné názory týkajúce sa jednotlivých akcií.

V menu si môžeme vybrať, či chceme zobraziť najbližšie akcie, alebo archív akcií. Potom sa na stránke zobrazí zoznam kategórií a zoznam jednotlivých akcií. Vedľa zoznamu kategórií je umiestnený jeden kalendár, aby každý si vedel pozrieť, presne v ktorých dňoch budú prebiehať jednotlivé akcie. Na začiatku sú vypísané všetky záznamy, po kliknutí na vybranú kategóriu sa zobrazia iba akcie patriace sa do danej kategórie. V zozname nájdeme iba pár informácií o akciách. Pre zobrazenie detailov je potrebné kliknúť na názov akcie, alebo na lupu, ktorá je zobrazená pri každej položke. Dostaneme sa tak na novú stránku, kde nájdeme všetky potrebné informácie. Tu sú zobrazené aj príspevky užívateľov. Na obrázku 5.4 vidíme detailné informácie vybranej akcie a jeden príspevok, ktorý bol pridaný k tejto akcie.

Koncert kapely Q7R

Kategória: hudba a tanec
Dátum: od 08/05/2007 do 08/05/2007
Hodina začiatku: 20:00
Miesto: Kultúrny dom
Opis: Vystúpenie kapely Q7R
Organizátor: KD Dolnozemska
Vstupné: 200 Sk

[Editovať](#)  [Zmazať](#) 

Príspevky Počet na stránku: 10 [Pridať príspevok](#) 

Názov: **Príspevok**
Dátum a čas: 29/04/07 22:18:11  Autor: **kubica**

Sem môžeme písať príspevky



Obrázok 5.4 Detaily akcie a príspevky

Správa akcií

Pridať novú akciu, alebo editovať či zmazať už existujúcu akciu má právo iba zamestnanec obecného úradu a správca systému. Pri pridaní nového záznamu sa zobrazí jeden formulár s niekoľkými položkami, sem treba zadať potrebné údaje. Niektoré údaje je nutné zadať, nie sú však všetky povinné. Z výberového poľa môžeme vybrať kategóriu, do ktorej bude patriť pridaná akcia. Keď sa v zozname nenachádza žiadaná kategória, máme možnosť pridať novú kategóriu. Dátum zahájenia a ukončenia môžeme nastaviť takisto pomocou výberových polí. Po zadaní potrebných údajov a odoslaní formulára systém skontroluje zadané údaje. V prípade chyby je vypísané chybové hlásenie. Obzvlášť dôležitá je kontrola zadaných dátumov. Z výberového poľa je možné vybrať dni od 1. do 31. Nie každý mesiac má ale 31 dní. Ďalej zvláštnym prípadom je február, ktorý v každom štvrtom

roku má 29 dní. Všetky tie prípady je potrebné testovať, ďalej zabezpečiť, aby dátum ukončenia danej akcie nebol skorší, než dátum zahájenia. Nový záznam je uložený do databázy iba v tom prípade, keď všetky zadané údaje sú správne.

Akcie nachádzajúce sa v databáze môžeme neskoršie editovať, alebo zmazať. Kliknutím na obrázok ceruzka môžeme zmeniť zadané údaje, formulár je ten istý, ako pri pridaní novej akcie. Zmazanie sa uskutoční kliknutím na kôš a potvrdení žiadosti o zmazaní.

Príspevky k akciám

Ku každej akcie je možné pridať príspevky. Túto funkciu však nemôžu využívať tí užívatelia, ktorí nie sú prihlásení. Oni môžu iba čítať pridané názory. Pomocou výberového poľa je možné nastaviť, koľko príspevkov chceme zobrazit' na jednu stránku. Keď počet príspevkov je väčší, než počet zobrazených na jednu stránku, pod zoznamom sa objavia odkazy na ďalšie stránky. Okrem obsahu príspevku je zobrazený názov príspevku (keď bol zadaný), dátum a čas uloženia a užívateľské meno autora. Aby sme vedeli rozlíšiť, či príspevok pridal registrovaný užívateľ, alebo zamestnanec obecného úradu, farby užívateľských mien sú odlišné. V prípade registrovaného užívateľa má čiernu farbu, inak červenú.

Pridanie nového príspevku je veľmi jednoduché. Stačí kliknúť na obrázok Pridať nový príspevok, a zadať obsah príspevku do zobrazeného formulára. Zadanie obsahu je povinné. Môžeme aj zadať názov príspevku, ale tu už nie je povinné. Po kliknutí na Pridať príspevok uložíme príspevok do databázy, a stránka je presmerovaná na výpis detailov tej akcie, ku ktorej sme písali názor.

Editáciu pridaných príspevkov neumožní systém. Správca systému však môže vymazať hociktorý príspevok.

5.9 Dokumenty

Na internetovej stránke obce určité dokumenty musia byť zverejnené a dostupné pre každého občana. Patria sem napríklad uznesenia zastupiteľstva obce, alebo dokumenty úradnej tabule. V našom informačnom systéme túto funkciu plní rubrika „Dokumenty“. Okrem spomínaných dokumentov občania tu nájdu rôzne formuláre, žiadosti v elektronickej podobe, ktoré takisto si môžu stiahnuť.

Na stránke tohto bloku sú najprv zobrazené jednotlivé kategórie dokumentov, radené podľa názvov. Vedľa názvov v zátvorkách je uvedený počet dokumentov nachádzajúcich sa v jednotlivých kategóriách. Po vybraní nejakej kategórie sa zobrazia všetky dokumenty, ktoré patria do danej kategórie. Vidíme tu názov dokumentu, krátky popis (čo vlastne obsahuje), dátum uloženia, ďalej formát (typ) a veľkosť dokumentov. Ikona dokumentu sa mení podľa formátu uloženého súboru, takže vždy vidíme, v akom formáte je daný dokument uložený. Po kliknutí na ikonu môžeme uložiť vybraný dokument. Zoznam dokumentov je radený podľa dátumu uloženia, naposledy uložený dokument je na začiatku zoznamu.

Správa dokumentov

Pridať dokumenty do jednotlivých kategórií môžu samozrejme iba zamestnanci a správca systému. Do formulára je povinné zadať názov dokumentu a vybrať dokument, ktorý chceme uložiť. Keď chceme podrobnejšie popísať, čo vlastne obsahuje daný dokument, môžeme to zadať do textového poľa „Popis“. Po odoslaní formulára systém skontroluje zadané údaje. Keď nemáme zadaný názov, je vypísané chybové hlásenie. Potom je kontrolovaný zadaný súbor. Táto kontrola sa skladá z viacerých častí. Najprv je testovaný typ súboru, pretože sú povolené iba vybrané formáty (doc, pdf, xls, rtf, zip). Priloženie iného súboru systém nedovolí. Ďalej je testovaná veľkosť a prípona súboru. Môže sa stať, že názov súboru obsahuje písmená s diakritikou. Keby sme chceli s takým názvom uložiť súbor, na strane servera sa v názve dokumentu objavili iné znaky, a tak by sme nemohli stiahnuť súbor, pretože meno dokumentu sa nezhodovalo s názvom uloženom v databáze. Tieto znaky je teda potrebné odstrániť. Na to používam jednoduchú funkciu, ktorá písmená s diakritikou vymení na písmená bez diakritiky. Nahrané dokumenty sú uložené do adresára „docs“, ďalej pre každú kategóriu je vytvorený podadresár. Názvom tejto zložky je ID danej kategórie. Keď všetky údaje sú správne, súbor je uložený na server a do databázy uložíme potrebné údaje.

Jednotlivé súbory je potom možné odstrániť. Správca systému môže vymazať všetky dokumenty, zamestnancom obecného úradu je povolené iba odstránenie vlastných súborov. Správca má možnosť vymazať celú kategóriu dokumentov. V tomto prípade všetky dokumenty, ktoré patria do vybranej kategórie, sú odstránené, a z databázy sú vymazané potrebné položky.

5.10 Články

V tejto rubrike nájdeme články a publikácie, ktoré napísali zamestnanci obecného úradu alebo správca systému. Témou týchto článkov môže byť napr. nejaké kultúrne podujatie, alebo oznam pre občanov, teda všetko, čo súvisí so životom obce. Rubrika aktualít slúži iba na oznámenie krátkych správ a noviniek, medzi články je možné pridať texty s rozsiahlym obsahom. Možným riešením mohlo byť vytvorenie novej kategórie v rubrike Dokumenty, do ktorej by bolo treba iba uložiť hotový textový dokument. Podľa mňa však oveľa praktickejšie je, keď užívateľ priamo na stránke môže prečítať daný článok a nemusí najprv stiahnuť dokument v elektronickej podobe.

Stránka článkov má veľmi jednoduchý výzor. Obsahuje zoznam pridaných článkov, v ktorom nájdeme názov a krátky popis článku, meno autora a dátum uloženia. Počet zobrazených položiek na stránku je nastavený na 10, ale užívateľ pomocou výberového poľa môže nastaviť aj iný počet. Po vybraní konkrétnej položky je zobrazený obsah článku.

Ako som to už spomenul, do tejto rubriky môžu písať iba zamestnanci a správca. Pridanie novej položky je veľmi jednoduché, pred užívateľom sa objaví formulár s textovým editorom (Obrázok 5.5). Aby formátovanie napísaného textu zvládol každý, použil som zjednodušenú verziu javascriptového vizuálneho editora Mozilla Rich Text Editor. Pomocou tohto editora môžeme

nastaviť rôzne štýly odstavcov, veľkosť, farbu písma, apod. Ďalej môžeme vytvoriť číslovaný či nečíslovaný zoznam, alebo zarovnať text do stredu. Čiže nájdeme tu všetky základné funkcie jednoduchého editora. Je to veľmi užitočná pomôcka pri písaní textu, pretože autor stále vidí, ako bude výsledok vyzerat' na stránke. Nemusíme teda vkladať do textu html kód, pomocou ktorého by bolo zabezpečené formátovanie danej časti textu.

The image shows a web interface for adding a new article. At the top is a button labeled "Pridať nový článok". Below it are two input fields: "Nadpis článku:" containing the text "test" and "Popis článku:". Underneath is a rich text editor toolbar with buttons for "Normal", "Font", "Size", "Bold", "Italic", "Underline", "Text Color", "Background Color", "Bulleted List", and "Numbered List". The editor area displays a preview of the text with various formatting applied, including bold, italic, underline, and a numbered list.

Obrázok 5.5 Formulár a editor na pridanie nohého článku

V prípade, keď vo formulári nemáme zadaný nadpis alebo popis článku, ktoré sú povinné údaje, systém nepovolí uložiť záznam do databázy a vypíše chybové hlásenie. Pre administrátora existuje možnosť vybrania kategórie, do ktorej bude daný článok patriť, pretože do portálu futbalistov takisto pomocou tohto formulára je možné pridať nový článok.

5.11 Služby a firmy

Táto rubrika ponúka návštevníkom zoznam tých firiem, ktoré ponúkajú svoje služby v obci Dolnozemska alebo v blízkom okolí. Do zoznamu si môže zaregistrovať každý svoju firmu, potrebné je iba vyplniť registračný formulár. Byť prihlásený do systému nie je podmienkou.

Registrácia novej služby/firmy

Aby údaje o firme užívateľa systému boli uložené do databázy a zobrazené na stránkach obce, stačí iba vyplniť jednoduchý registračný formulár. Odkaz na registráciu firmy nájdeme v podmenu hlavnej

menu položky „Služby a firmy“. Do registračného formulára je možné zadať všetky dôležité informácie týkajúce sa danej firmy. Údaje označené hviezdíčkou je potrebné vyplniť, sú to totiž povinné údaje. Kategóriu, do ktorej bude patriť registrovaná služba, je možné vybrať z výberového poľa. Prvou predstavou bolo, že pridať kategóriu služieb budú môcť aj jednoduchí užívatelia, ale v tomto prípade by sa mohlo stať, že boli pridané aj veľmi podobné typy. Práve preto podľa mňa rozumnejším riešením je, keď ponúkame bohatý výber kategórií, ktorý môže meniť iba administrátor. Z uvedeného zoznamu potom každý si môže vybrať tú najvhodnejšiu kategóriu.

Po vyplnení údajov a odoslani formulára systém skontroluje zadané údaje. Keď chýba nejaký povinný údaj, alebo je zadaný v nesprávnom tvare, systém vypíše chybové hlásenie. Jednotlivé údaje musia byť zadané v preddefinovanom tvare (napr. PSČ musí obsahovať 5 číslic, alebo emailová adresa musí obsahovať znak @). To všetko je testované pomocou regulárnych výrazov. Keď zadané údaje sú správne, sú uložené do databázy.

Ako som to už spomenul, pridať kategóriu má právo iba administrátor. Jediné, čo je potrebné zadať, je názov kategórie, a potom môžeme uložiť do databázy.

Zoznam služieb

Užívatelia pomocou menupoložky si môžu vybrať, či chcú prezrieť celý zoznam služieb, alebo iba konkrétnu kategóriu. Počet zobrazených služieb na jednu stránku je nastavený na 10, ale pomocou výberového poľa je možné nastaviť na iný počet. Takisto pomocou výberového poľa je možné vybrať konkrétnu kategóriu. Pod zoznamom nájdeme odkazy na ďalšie stránky. Pri každej firme sú vypísané všetky informácie, ktoré boli zadané pri registrácii.

Po prihlásení administrátora pod každou položkou nájdeme ikonku koša. Kliknutím na túto ikonku je možné vymazať z databázy danú firmu. Editáciu údajov už zaregistrovaných firiem systém neumožňuje.

5.12 Úradná tabuľa

Povinnou časťou informačného systému obce je úradná tabuľa. Sem majú uložiť zamestnanci obecného úradu tie dôležité dokumenty, ktoré majú byť zverejnené a dostupné pre každého občana. Táto sekcia je prepojená s rubrikou „Dokumenty“, tzn., že tu zobrazené dokumenty môžeme nájsť aj medzi dokumentmi v podkategórii Úradná tabuľa. Keď chceme pridať nový súbor, môžeme to urobiť takisto v rubrike Dokumenty pomocou formulára Pridať nový dokument. Postup pridania nového dokumentu je popísaný v kapitole 5.9.

Na stránke úradnej tabule nájdeme iba najnovšie dokumenty. Jednotlivé položky na tejto stránke sú triedené do dvoch skupín: aktuálne dokumenty a archív. V zozname pri každej položke vidíme jej názov, ďalej dátum uloženia a veľkosť súboru. Po kliknutí na názov vybraného dokumentu môžeme stiahnuť súbor.

Medzi aktuálne dokumenty som zaradil tie dokumenty, ktoré boli pridané v posledných 30 dňoch. V archíve nájdeme súbory, ktoré sú staré minimálne mesiac, maximálne 60 dní. Ostatné súbory, ako som to už spomenul, nájdeme v rubrike Dokumenty. Pri vytvorení databázového dotazu SELECT som použil databázovú funkciu DATE_ADD, ktorá umožňuje pričítať k dátumu určité obdobie (v našom prípade napr. 30 dní). Pomocou tejto funkcie jednoducho môžeme určiť, ktoré dokumenty do ktorej skupiny budú patriť.

5.13 Elektronická podateľňa

Elektronická podateľňa je určená na odosielanie elektronických správ na emailovú adresu obce priamo z webovej stránky. K emailu je možné priložiť aj súbor ako prílohu. Dovoľené formáty príloh sú *.doc, *.pdf, *.rtf, *.zip a *.xls. Odoslanie súboru v inom formáte nie je možné. Jednou emailovou správou je možné odoslať iba jednu prílohu. Maximálna veľkosť priloženého súboru je obmedzená na 1,5 MB, väčšie súbory nebudú odoslané.

Po kliknutí v hlavnom menu na položku Elektronická podateľňa sa pred užívateľom objaví formulár na odosielanie emailu. Tento formulár je znázornený na obrázku 5.6. Zadanie predmetu, obsahu a emailovej stránky odosielateľa je povinné. Keď chceme odoslať email s prílohou, je potrebné vybrať súbor, a tak odoslať email. Po odoslaní formulára nasleduje kontrola zadaných údajov. Najprv sú testované, či sú zadané všetky povinné údaje. Pri emailovej adrese odosielateľa pomocou regulárneho výrazu je kontrolované, či je zadaná v správnom tvare. V prípade chyby je vypísané chybové hlásenie. Keď nebol pridaný žiadny súbor, a všetky údaje sú správne zadané, nasleduje odosielanie emailu.

Nová správa

Predmet: *

Váš e-mail: *

Text: *

Pridať prílohu:

(max 1,5 MB)

* - tie údaje sú povinné

Obrázok 5.6: Elektronická podateľňa

V prípade, že posielame email s prílohou, sú potrebné aj ďalšie kontroly. Po odoslaní formulára takisto sú testované najprv zadané údaje. V ďalšom kroku je kontrolovaný samotný súbor. Pretože maximálna veľkosť je obmedzená na 1,5 MB, je potrebné určiť, či veľkosť vybraného súboru nepresahuje maximálnu dovolenú veľkosť. V prípade, že je väčší než 1,5 MB, užívateľovi je vypísané chybové hlásenie. Ďalej kontrolujeme formát súboru, povolené typy sú uvedené vyššie. Keď spomínané atribúty sú správne, na strane servera je vytvorený nový adresár „tmp“. Do tohto adresára budeme nahráť vybraný súbor. Na prenos súboru je použitá funkcia `move_uploaded_file`. Potom nasleduje práca so súborom. Otvoríme súbor na čítanie, do premennej skopírujeme celý obsah súboru a zatvoríme súbor. Ďalej už tento súbor potrebovať nebudeme, takže odstránime zo servera. Nasleduje zostrojenie emailovej správy. Keďže posielame aj prílohu, do tela správy je potrebné „zabudovať“ vybraný súbor. Po vytvorení tela správy pomocou funkcie `mail()` je odoslaný email.

5.14 Portál futbalistov

Požiadavkou bolo vytvorenie jednoduchého portálu, v ktorom je možné zverejniť výsledky zápasov futbalového klubu obce. Ďalej tento portál by mal poskytnúť aj možnosť pridania rôznych článkov a publikácií o jednotlivých zápasoch, či o aktivite futbalového klubu. Právo na správu tohto portálu má iba administrátor a správca futbalistov. Správca futbalistov je registrovaný užívateľ, ktorému je pridelená táto funkcia. „Povýšiť“ práva užívateľa na túto funkciu môže iba administrátor.

Keď si užívateľ systému vyberie rubriku Zápasy, objaví sa tabuľka, v ktorej sú zobrazené informácie o odohraných zápasoch (kolo, dátum, súper, výsledok). Aby každý videl, ktoré zápasy vyhrali, ktoré prehrali a na ktorých remizovali futbalisti obce, farebne sú odlišované jednotlivé výsledky. Návštevník stránky môže vybrať, výsledky ktorej sezóny chce zobrazit'.

Správca systému alebo správca futbalistov môže pridať novú položku do tabuľky zápasov. Do zobrazeného formulára je potrebné zadať, ktoré kolo bolo odohrané, dátum zápasu, meno súpera, stav výsledku a sezónu. Všetky údaje sú povinné. Zadanie dátumu je pomocou výberových polí. Stav výsledku znamená výhru, prehru alebo remízu tímu. Po zadaní údajov a odoslaní formulára systém skontroluje zadané údaje. V prípade, že niektorá položka nie je zadaná, alebo je nesprávna, systém to ohlási chybovým hlásením. Pri testovaní dátumu musíme dávať pozor na to, že nie každý mesiac má 31 dní, alebo február v prestupných rokoch má 29 dní, a nie iba 28. Keď všetky údaje sú správne, do databázy je pridaný nový záznam.

Pridané výsledky nie je možné editovať, je naimplementované iba zmazanie záznamov.

O jednotlivých zápasoch môžeme čítať v rubrike článkov. Je to prepojená s blokom „Články“, sú tu ale vypísané iba publikácie na tému futbal. Na pridanie nového textu slúži ten istý formulár, ktorý používame a bloku Články. Implementácia tohto formulára je podrobne popísaná v kapitole 5.10.

5.15 Diskusné fórum

Diskusné fórum poskytuje predovšetkým priestor pre diskutovanie, vyjadrovanie vlastného názoru, ale aj na poskytovanie rád, odporúčaní a pod. Hoci užívatelia môžu svoje názory vyjadrovať aj v kalendári akcií, ale tie súvisia iba s danou akciou. Tu však z viacerých diskusných skupín môže vybrať podľa toho, o akej téme chce práve diskutovať. Prezerat' príspevky jednotlivých diskusných skupín môže každý, pridať nový názor je ale možné iba po prihlásení. Vytvoriť novú registráciu pre fórum nie je potrebné. Do fóra je možné prihlásiť sa pomocou toho prihlasovacie mena a hesla, ktoré získal užívateľ pri registrácii do nášho systému.

Výzor diskusného fóra je trochu iný, ako na ostatných stránkach systému. Na ľavej strane pod hlavným menu sa nenachádza prihlasovací formulár, bloky aktualít a užitočných odkazov sú vynechané, aby šírka obsahu stránky mohla byť väčšia. Pod obrázkom v záhlaví sa nachádza ešte jedno menu, ktoré obsahuje odkazy napr. na prihlasovanie alebo registráciu. Po prihlásení obsah tohto menu sa trochu mení, vtedy vidíme položky, ktoré slúžia na zobrazenie vlastného profilu, alebo na odhlasovanie zo systému.

Vstupom na stránku diskusného fóra sú zobrazené názvy hlavných diskusných skupín, pod nimi sú vymenené jednotlivé podskupiny, ktoré patria do týchto hlavných skupín. V zozname podskupín nájdeme podrobnejšie informácie o daných diskusných skupinách. Vidíme tu názov podskupiny, ďalej krátky popis, počet tém, ktoré sa nachádzajú v skupine a informácie o tom, do ktorej témy bol naposledy pridaný príspevok, a čas pridania tohto príspevku. Po vybraní danej podskupiny v podobnej forme je vypísaný zoznam tém. K samotným príspevkom sa dostaneme po vybraní nejakej témy.



Obrázok 5.7: Príspevok v diskusnom fóre

Každý užívateľ môže pridať do vlastného profilu obrázok (tzv. avatar), ktorý je potom zobrazený pri každom príspevku, ktoré pridal. Okrem tohto obrázku v zozname príspevkov nájdeme pri každom príspevku užívateľské meno autora, celkový počet pridaných príspevkov autora názoru, presný čas a dátum pridania názoru a samotný text príspevku. Keď názor je reakciou na iný príspevok, vidíme aj obsah pôvodnej správy a meno jej autora. Na obrázku 5.7 vidíme príspevok

v diskusnom fóre, ktorý je reakciou na iný názor. Pridať nový príspevok môže iba prihlásený užívateľ. Po kliknutí na „nový názor“ sa objaví samotný formulár, do ktorého je potrebné zadať text príspevku. Fórum podporuje aj pridanie smejkov do textu. Použiteľné smejkovia takisto nájdeme v tomto formuláre. Pri vkladani smejka do obsahu názoru do textového poľa je vložený text v podobe `X`, kde X sa mení podľa toho, ktoré smejko sme vybrali. Do databázy je uložený text v tomto formáte. Aby pri zobrazení príspevku sme videli samotné obrázky – smejkovia, pri výpise jednoduchou funkciou znaky `X` sú prepísané na html kód, ktorý potom zobrazí dané obrázky. Uložiť prázdny príspevok systém nedovolí. Reakciu môžeme písať na hociktorý príspevok, v tomto prípade vo formulári vidíme aj pôvodný názor, na ktorý chceme reagovať.

V prípade, keby niektorý užívateľ napísal do fóra príspevok s takým obsahom, ktorý do fóra nepatrí, administrátor systému má právo editovať alebo vymazať tento názor. Ďalej správca môže založiť nové témy, podskupiny či hlavné diskusné skupiny. Pri pridani novej skupiny povinne musí zadať názov skupiny, zadanie popisu nie je nutnou podmienkou.

Každý užívateľ si môže pozrieť svoj profil, kde nájde vlastné registračné údaje, a formulár na pridanie obrázku (avatu). Nie je zadané, maximálne aký veľký obrázok môžeme nahráť. Pri uložení veľkosť obrázku je zmenšená tak, aby šírka mala maximálne 120 pixelov. Pomer šírky a výšky je zachovaný.

5.16 Administrátorsky profil

Funkciou administrátora je správa celého systému. V hierarchii užívateľov systému správca má najvyššie práva, a tak prístup ku každej funkcii, ktorú systém poskytuje. Jednotlivé časti systému teda nie sú viditeľné ani pre zamestnancov obecného úradu. Sem patrí napr. správa zamestnancov a správa užívateľov systému.

Správa zamestnancov obecného úradu

Na to, aby do systému mali prístup aj zamestnanci, takisto ako registrovaní užívatelia potrebujú užívateľský účet. Kým návštevníci internetovej stránky obce majú možnosť pomocou registračného formulára stať sa registrovanými užívatelmi, a tak získať vlastný užívateľský účet, zaregistrovať nového zamestnanca do databázy môže iba administrátor systému. Registračný formulár na pridanie nového zamestnanca je veľmi podobný, ako v prípade registrácie užívateľa, v údajoch o osobe sú však odlišnosti. Vyplnenie údajov, ktoré sú označené hviezdíčkou, je povinné. V prípade, že niektorý povinný údaj chýba, alebo je zadaný v nesprávnom tvare, systém to ohlási chybovým hlásením. Pomocou výberového poľa môžeme zadať funkciu zamestnanca na obecnom úrade, ďalej práva (či bude mať práva administrátora, alebo iba práva zamestnanca). Keď v zozname funkcií neexistuje funkcia, ktorú chceme nastaviť, je možné pridať do databázy novú funkciu. Pri registrácií je potrebné

zadat heslo, a potom potvrdiť toto heslo. Samozrejme toto heslo každý zamestnanec potom môže nastaviť na iné vo vlastnom profile.

Administrátorovi je umožnené zobrazit' zoznam zamestnancov. V tomto zozname sú vypísané údaje zamestnancov. Aby bolo jednoduchšie nájsť v zozname údaje konkrétneho pracovníka, systém umožňuje filtráciu podľa funkcie a hľadanie podľa zadaného reťazca. Zadaný reťazec je hľadaný v položkách meno, priezvisko, užívateľské meno a ulica. V zozname jednotlivé položky sú triedené podľa priezviska. Počet zobrazených položiek na stránku je možné nastaviť. Kliknutím na vybraného zamestnanca sa zobrazia všetky jeho údaje, ktoré potom je možné editovať a uložiť zmeny.

Správa užívateľov

Podobne, ako u správe zamestnancov, aj táto funkcia je prístupná iba pre administrátora systému. Zoznam registrovaných užívateľov je vypísaný podobným spôsobom ako v zozname zamestnancov. Sú tu vypísané všetky údaje o užívateľoch: priezvisko, meno, užívateľské meno, telefón, email a adresa. V zozname môžeme hľadať podľa zadaného výrazu. Zadaný výraz je potom hľadaný v položkách užívateľské meno, meno, priezvisko, ulica a mesto/obec. Položky v zozname sú radené podľa priezviska.

Pri návrhu som uvažoval nad tým, či mám administrátorovi povoliť alebo nie, aby mohol zmeniť údaje registrovaných užívateľov. Nakoniec som sa rozhodol tak, že údaje užívateľov správca nebude môcť zmeniť. Môže ale pridelit' právo správca futbalistov hociktorému užívateľovi. Z jednoduchého registrovaného užívateľa sa tak stane správca portálu futbalistov, ktorý bude mať právo pridať výsledky zápasov, či písať články do rubriky články, ale iba do kategórie futbal.

5.17 Ďalšie funkcie systému

V predchádzajúcich podkapitolách boli popísané všetky hlavné časti systému. Okrem tých sú však ešte implementované ďalšie funkcie a stránky, ktoré môže využívať a prezerať návštevník internetového portálu obce.

RSS

Na úvod by som popísal, čo je vlastne RSS a na čo sa používa. RSS je v podstate dialekt XML (eXtensible Markup Language) - teda RSS je rodina XML formátov, určených hlavne pre čítanie noviniek na webových stránkach. Technológia RSS umožňuje užívateľom internetu prihlásiť sa k odberu noviniek z webu, ktoré poskytuje RSS zdroj (RSS feed, RSS kanál, RSS channel). Tento zdroj sa väčšinou vyskytuje na stránkach, kde sa obsah mení a pridáva veľmi často (napríklad spravodajské servery).

RSS umožňuje publikovanie zoznamu odkazov spolu s ďalšími informáciami, ktoré bližšie popisujú daný odkaz. Keď autor webu použije RSS, návštevníci jeho stránok iste ocenia možnosť

získať informácie bez nutnosti jeho návštevy. Oproti iným spôsobom propagácie webu (napr. newslettery) nie je nutná registrácia návštevníka. Webu to v konečnom dôsledku prinesie zvýšenie návštevnosti, pretože sa ľudia budú viackrát vrátiť na stránku. Koncepcia RSS tak umožňuje udržiavať s návštevníkmi trvalý kontakt.

Každý web môže mať viac než jeden RSS kanál. Vedľa hlavného kanálu je vhodné publikovať ďalšie informácie, napríklad novinky na webe, oznámenia o nových produktoch, zoznam dokumentácií, súbory na stiahnutie apod. V poslednej dobe sú RSS kanály stále častejšie dostupné.

Na stránkach nášho informačného systému nájdeme taktiež také rubriky, ktoré sú často aktualizované. Sú to napríklad aktuality, alebo články. Na tie stránky som vytvoril RSS kanál. Ďalej používatelia diskusného fóra takisto môžu čítať pomocou RSS čítačky nové príspevky v jednotlivých diskusných skupinách.

Pri vytvorení RSS kanálu nesmieme zabudnúť na určité pravidlá. Značky (tagy) elementov RSS sú párové. Počiatočná značka, konečná značka a dáta medzi nimi tvorí element. Element môže obsahovať ďalšie vnorené elementy. Povinné značky, ktoré musíme zadať, sú:

- **<channel>** - informácie o danom kanálu. Element musí obsahovať elementy <title> (meno kanálu, max. 100 znakov), <link> (URL daného webu, max. 500 znakov), <description> (popis kanálu, max. 500 znakov), <language> (jazyk daného kanálu) a <image> (XML element obsahujúci ďalšie elementy)
- **<description>** - textový popis pre item, channel, image a textinput
- **<language>** - špecifikácia jazyka daného kanálu
- **<link>** - URL danej položky. Musí začínať http:// alebo ftp://
- **<title>** - textová identifikácia zdroje. Keď sa použije v <item>, je použitá ako meno odkazu. Pri použití u <image> je použitá ako alternatívny text obrázku. Pre <channel> je použitá ako titulok daného kanálu. U <textinput> slúži ako titulok textového poľa.

Okrem vymenovaných značiek existuje ešte veľa ďalších, ktoré však nie sú povinné. (zdroj: literatúra [14], [16])

Prístupnosť webu

Vstupnou požiadavkou bola, aby vytvorený informačný systém splnil podmienky prístupného webu. Pod pojmom prístupnosť chápeme taký stav, keď daná vec nekladie svojim užívateľom pri používaní žiadne prekážky. Pojem prístupnosti webových stránok je oproti obecnému pojmu zúžený na funkčnosť webových stránok. Prístupné stránky teda nestavia svojim užívateľom žiadne prekážky, ktoré by im znemožnili daný web efektívne používať.

Existujú pravidlá tvorby prístupného webu, podľa ktorých keď postupujeme pri tvorbe systému, môžeme zabezpečiť, že obsah webových stránok bude dostupný a čitateľný pre každého. Pri vytvorení systému som sa snažil jednotlivé pravidlá dodržať. Samozrejme týchto pravidiel je veľmi veľa (vid'. zdroj: literatúra [15]), preto tu spomeniem iba niekoľko.

Na stránkach systému som použil jasný a jednoduchý jazyk, ktorý zabezpečuje efektívnu komunikáciu. Nemôžeme predpokladať, že každý návštevník je odborníkom na problematiku, ktorá je na webových stránkach prezentovaná. Informácie na stránkach systému sú preto podávané čo najzrozumiteľnejším spôsobom. Na stránkach som nepoužíval žiadne odborné terminológie alebo cudzie slová, aby som zabezpečil, že každý rozumel napísanému textu.

Pri vytvorení užívateľského rozhrania som postupoval tak, aby užívateľ vždy jasne rozpoznal, ktoré informácie slúžia k navigácii medzi jednotlivými stránkami webu a ktoré patria k samotnému obsahu danej WWW stránky. Navigačné menu som preto združil do samostatného bloku. Toto menu je umiestnené na každej WWW stránke na stabilnom mieste, aby ho užívateľ nemusel hľadať. Okrem menu je na stránkach umiestnená aj navigácia, ktorá obsahuje odkaz na vyššiu úroveň v hierarchii webu a odkaz na úvodnú WWW stránku.

Mapa webu je dôležitým prvkom každého rozsiahleho webu. Preto som to aj ja naimplementoval, a nájdeme ju po kliknutí na odkaz *Mapa stránok*. Mapa webu je štruktúrovaný zoznam odkazov na všetky WWW stránky systému, je výborným pomocníkom pre orientáciu v rámci systému. Odkaz na mapu stránok máme k dispozícii na každej stránke v navigačnom menu.

Medzi pravidlá tvorby prístupného webu patrí, aby informácie zverejňované na základe zákona boli dostupné ako textový obsah webovej stránky. Povinne zverejňované informácie obsahuje aj náš systém, nájdeme ich v sekcii obecného úradu.

Na záver by som ešte pripomenul, že WWW stránky som vytvoril tak, aby boli blind friendly – tzn. aby užívatelia, ktorí majú zhoršenú zrak, vo všetkých bežne používaných internetových prehliadačoch mohli zväčšiť veľkosť písma. Preto som nepoužíval žiadne také predpisy definujúce veľkosť písma, ktoré by toto zväčšenie znemožňovali.

V tejto kapitole som rozoberal vlastnú implementáciu systému. Implementácia sa začala vytvorením databázy pomocou databázového systému MySQL. Potom nasledovala vytvorenie užívateľského rozhrania a implementácia všetkých funkcií systému, ktoré má poskytovať podľa stanovených požiadaviek. V každej podkapitole som opísal hlavnú funkciu danej časti systému, vzhľad užívateľského rozhrania a možnosti, ktoré poskytuje pre jednotlivé typy užívateľov. Jednotlivé časti systému sú demonštrované s pridanými obrázkami, aby čitateľ ľahšie vedel predstaviť, ako vlastne vyzerá dané užívateľské rozhranie.

6 Ďalší rozvoj systému

Implementácia informačného systému prebieha na základe analýzy a návrhu. V týchto fázach vývoja sú zhrnuté všetky požiadavky zadávateľa projektu. Napriek tomu, že vo väčšine prípadov vstupné požiadavky sú detailne popísané, pri testovaní alebo nasadení systému môžeme naraziť na určité nedostatky. Vtedy vidíme, čo nám poskytuje systém, a čo by ešte mohol. Vznikajú tak nové nápady na možné vylepšenia systému. Táto kapitola sa zaoberá možnými rozšíreniami nášho informačného systému.

Prvým možným rozšírením by mohlo byť rozdeliť stránku na určité bloky a umožniť jednoduché prepínanie medzi nimi. Rozumiem pod tým vytvorenie blokov ako napr. Občan, Turista, Futbal, Hasiči apod. Návštevník stránky tak jednoducho by mohol zobrazíť tie časti systému, ktoré ho zaujímajú. Pre občana samozrejme dôležité môžu byť oznamy obecného úradu, či všeobecné základné informácie. Preto v bloku Občan by mohli byť zobrazené časti systému týkajúce sa hlavne obecnému úradu. V bloku pre turistov by mohlo byť užitočné poskytovať informácie o ubytovaní v obci, fotogalériu okolia, či prehľad pozoruhodností. Podľa vybraného bloku sa tak trochu zmenil celkový vzhľad systému, dynamicky sa zmenil i menu. Na rôznych stránkach sa už môžeme stretávať so spomínanou funkciou, a podľa mňa je to užitočné riešenie.

Určite veľmi užitočným rozšírením by mohla byť implementácia vyhľadávania. Užívateľ systému tak jednoducho pomocou zadaného kľúčového slova by mohol nájsť záznamy, ktoré práve potrebuje, a nemusel prehľadať všetky stránky systému. Tento vyhľadávací systém by mal poskytovať rozšírené vyhľadávanie, kde po zadaní kľúčového slova sme mali možnosť zadať aj iné parametre, napr. rubriku, v ktorej chceme hľadať.

Rozšírenie portálu futbalistov s novými stránkami by určite potešil každého fanúšika futbalového klubu. Zatiaľ obsahuje iba zoznam výsledkov odohraných zápasov a články o futbale. Rozšíriť tento portál by sme mohli napr. pridaním zoznamu a údajov hráčov, či fotogalérie o jednotlivých zápasoch. Podobným spôsobom by sme mohli vytvoriť aj portál hasičov, kde by boli vymenované členovia hasičského zboru, história zboru, výsledky jednotlivých hasičských súťaží a ďalšie zaujímavosti.

Pridaním nových funkcií do administrátorskej časti by určite uľahčil prácu administrátora systému. Editácia určitých častí nie je možná ani pre administrátora, a preto umožnenie úprav týchto položiek by určite bolo výhodné. Ďalej by sme mohli implementovať možnosť hromadného zmazania jednotlivých položiek, aby správca nemusel položky zmazať po jednom.

Podobných možných vylepšení by sme určite našli ešte niekoľko. V budúcnosti možno vyššie uvedené príklady vylepšenia systému aj zrealizujem, alebo doplním ešte ďalšie funkcie systému, aby informačný systém obce Dolnozemska poskytoval čím viac možností pre užívateľov.

7 Záver

V dnešnej dobe, keď je internet hlavným predstaviteľom zdrojov informácií a je dostupný už skoro pre každého, je dobré, aby obce mali vlastný internetový portál a boli tak zviditeľnené vo svete modernej komunikácie medzi ľuďmi. Preto som sa rozhodol o vytvorenie funkčného informačného systému pre obec Dolnozemska. Prednosť tohto systému je dynamickosť a informovanosť. Občanom poskytuje informácie o plánovaných akciách a udalostiach, články o dianí v obci, informácie o obci, fotogalériu apod. Nedeliteľnou súčasťou takého informačného systému je poskytovanie informácií a oznámení obecného úradu, t.j. úradná tabuľa, záznamy o uzneseniach zo zasadnutí, e-podatelňa a mnoho ďalšieho. Pre občanov je pripravené diskusné fórum. Systém teda poskytuje veľa zaujímavých funkcií.

Cieľom tejto bakalárskej práce bolo vytvorenie funkčného prototypu informačného systému obce. Samotnú implementáciu predchádzala podrobný popis požiadaviek, ktoré by mal splniť hotový systém. Po stanovení všetkých požiadaviek nasledovala analýza a návrh celého systému. Tie časti tvorby informačných systémov hrajú nezastupiteľnú úlohu, a práve preto som sa snažil čím podrobnejšie analyzovať všetky požiadavky a navrhnúť kvalitný model systému. Pri tvorbe systému najviac práce bolo samozrejme s implementáciou, ktorú som vykonával na základe navrhnutého modelu systému. Výsledkom tvorby sa stal funkčný informačný systém, ktorý spĺňa všetky vstupné požiadavky.

Počas vývoja systém bol testovaný v rôznych internetových prehliadačoch, aby bola odstránená možná nekompatibilita, ktorá by mohla byť spôsobená odlišným zobrazením kódu v daných prehliadačoch. Všetky chyby, ktoré sa objavili počas testovania, boli úspešne odstránené. Systém bol testovaný aj z hľadiska funkčnosti na užívateľoch s rôznymi počítačovými znalosťami. Vďaka tomu, že pri implementácii som sa snažil vytvoriť užívateľské rozhranie a vlastne celý systém tak, aby bol prehľadný a zrozumiteľný, systém umožnil aj menej skúseným užívateľom pracovať s ním bez akýchkoľvek väčších problémov.

Pomocou tejto bakalárskej práce som si osvojil veľa nových poznatkov o procese tvorby informačného systému. Pri implementácii systému som použil rôzne implementačné nástroje, programovacie jazyky a tak som mal možnosť rozšíriť moje znalosti v týchto jazykoch. Oboznámil som sa s novými zaujímavými funkciami, ktoré ponúka jazyk PHP. Naučil som sa formátovanie prvkov webového rozhrania pomocou kaskádových štýlov a mal som možnosť získať nové skúsenosti so skriptovacím jazykom JavaScript.

Literatúra

- [1] Williams, H. E., Lane, D.: *PHP a MySQL – Vytváříme webové databázové aplikace*. Computer Press, 2002, 552 s. ISBN 8072267604
- [2] Ing. Radek Kočí, Ph.D., Ing. Bohuslav Křena, Ph.D.: *Úvod do softwarového inženýrství*. – studijní opora Brno 2006
- [3] Prof. Ing. Hruška, T. CSc.: *Pojem informačního systému*. – studijní opora, Brno 2006
- [4] Polák, J., Merunka, V., Carda, A.: *Umění systémového návrhu*. Grada, Praha 2003, ISBN 80-247-0424-02
- [5] Budd, A. Moll, C. Collison, S.: *CSS filtry, hacky a pokročilé postupy*. Brno: Zoner Press, 2007, ISBN 978-80-86815-54-1
- [6] Kosek, J.: *HTML, tvorba dokonalých www stránek*. Praha: Grada Publishing, 1998, ISBN 80-7169-608-0
- [7] Jan Stejskal: *Vytváříme WWW stránky pomocí HTML, CSS a JavaScriptu*. Brno: Computer Press, 2004, ISBN 80-251-0167-3
- [8] Kosek, J.: *PHP – Tvorba internetových aplikací*. Praha: Grada Publishing, 1999, ISBN 80-7196-373-1
- [9] The PHP Group: *PHP: Hypertext Preprocessor*. Dokument dostupný na URL: www.php.net, 03.05.2007
- [10] Janovský, D.: *O tvorbě internetových stránek*. Dokument dostupný na URL: <http://www.jakpsatweb.cz/>, 03.05.2007
- [11] MySQL AB: *MySQL Documentation*. Dokument dostupný na URL: <http://www.mysql.com/>, 03.05.2007
- [12] *Informačné systémy*. Dokument dostupný na URL: http://ki.fpv.ukf.sk/materialy_public/databazove_systemy/InformacneSystemy.doc, 03.05.2007
- [13] Bešenyi, L.: *Budovanie informačných systémov I. – Životný cyklus a inštalácia MySQL*. Dokument dostupný na URL: <http://www.inet.sk/clanok/3741/>, 03.05.2007
- [14] *Interval.cz webdesign a e-komerce denně*. Dokument dostupný na URL: <http://interval.cz/>, 03.05.2007
- [15] Špinar, D.: *Přístupnost: Web a weblog věnovaný přístupnosti webových stránek*. Dokument dostupný na URL: <http://pristupnost.nawebu.cz/>, 03.05.2007
- [16] *Wikipedie otevřená encyklopedie*. Dokument dostupný na URL: <http://cs.wikipedia.org/>, 03.05.2007

Zoznam príloh

Príloha 1. CD