

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

INFORMAČNÍ PORTÁL MĚSTYSE

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

ROMAN BÍLEK

BRNO 2009



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

INFORMAČNÍ PORTÁL MĚSTYSE

INFORMATION SYSTEM OF TOWN

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

ROMAN BÍLEK

VEDOUČÍ PRÁCE

SUPERVISOR

Ing. MARTIN STRAKA

BRNO 2009

Abstrakt

Práce pojednává o informačním portálu městyse Zubříč. V úvodu je popsán současný stav informačních portálů obcí a měst. Dále jsou zde vymezeny a také vysvětleny pojmy, které s touto problematikou souvisejí. Následuje část, popisující rozčlenění systému. V této části jsou vysvětleny jednotlivé role aktérů, včetně jejich oprávnění. Závěr práce popisuje způsob testování a možná rozšíření do budoucna.

Abstract

This working handle with about information system of the market–town Zubříč. The introduction describes the present state of the information systems of the villages and towns. Further there are determination as well as illustrate the notions, which are related with this problems. Next part describes the dismemberment of system. In this part are illustrated the individual role of participants, including their reason. The conclusion of the work describes the way of the testing and the other expansion into the future.

Klíčová slova

Informační portál, městys, databázový systém

Keywords

Information System, market–town, Data Base System

Citace

Roman Bílek: Informační portál městyse, bakalářská práce, Brno, FIT VUT v Brně, 2009

Informační portál městyse

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Martina Straky

.....
Roman Bílek
18.5.2009

Poděkování

Na tomto místě bych chtěl poděkovat vedoucímu mé bakalářské práce Ing. Martinu Strakovi za zájem, připomínky a čas, který věnoval mé práci.

© Roman Bílek, 2009.

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

Obsah

1 Úvod	3
2 Vymezení použitých pojmů	4
2.1 Městys	4
2.2 Informační systém	4
2.2.1 Informace	4
2.2.2 Systém	4
2.3 Databáze	5
2.3.1 Klíč	6
2.3.2 Integrita databáze	6
2.3.3 Systém řízení báze dat	6
2.3.4 Normalizace a normální formy	7
2.3.5 Vztahy mezi tabulkami	7
3 Vývoj informačních systémů	8
3.1 50. léta	8
3.2 60. léta	8
3.3 70. léta	8
3.4 80. léta	9
3.5 90. léta a současnost	10
3.5.1 Informační portály měst a obcí	10
4 Závazné zákony pro městys	12
4.1 Zákon o obcích	12
4.2 Zákon upravující počet zastupitelů	13
4.3 Zákon o informačních systémech veřejné správy	14
5 Neformální specifikace	15
6 Návrh systému	16
6.1 ER diagram	16
6.1.1 Občan a Typ role	17
6.1.2 Vyhlášky a nařízení	17
6.1.3 Úřední deska	17
6.1.4 Aktuality	17
6.1.5 Elektronická podatelna	18
6.1.6 Ankety a Kontrola ankety	18
6.1.7 Fotogalerie	18

6.1.8	Diskuze	18
6.1.9	Organizace v obci	18
6.1.10	Knihovna	19
6.1.11	SDH	19
6.1.12	Kino	19
6.1.13	Fotbalový klub	19
6.1.14	Kalendář akcí	19
6.1.15	Inzerce	19
6.1.16	Podniky	19
6.1.17	Výroční zpráva	20
6.1.18	Zastupitelstvo	20
6.1.19	Hlášení	20
6.1.20	Noviny městyse	20
6.1.21	Zápisy z jednání zastupitelstva	20
6.2	Rozdělení rolí v systému	22
6.2.1	Administrátor	22
6.2.2	Občan	22
6.2.3	Zastupitel	22
6.2.4	Starosta	23
6.2.5	Vedoucí knihovny	23
6.2.6	Vedoucí kina	23
6.2.7	Vedoucí fotbalu	23
6.2.8	Vedoucí SDH	24
6.3	Schéma databáze	24
7	Implementace	27
7.1	Apache	28
7.1.1	Historie Apache	28
7.2	PHP	28
7.2.1	Historie PHP	28
7.3	MySQL	29
7.4	CSS	29
7.4.1	Historie CSS	30
7.5	JavaScript	30
7.6	HTML	30
7.6.1	Historie HTML	30
7.6.2	Protokol HTTP	31
8	Testování	32
8.1	Testování při vývoji	32
8.2	Testování uživateli	33
8.2.1	Testování designu	33
8.2.2	Testování přehlednosti	35
9	Možná rozšíření do budoucna	36
10	Závěr	37
A	Seznam příloh	39

Kapitola 1

Úvod

Internet a jeho obsah bereme v dnešní době již jako samozřejmost. Díky jeho rozšířenélosti se také není čemu divit. Dnes už si málokdo uvědomí, že prapůvodní účel Internetu byl zcela jiný. Samotný vznik se datuje do 70. let minulého století. Cílem bylo vytvořit spolehlivou a hlavně stabilní síť, která by odolala i jadernému útoku. K docílení této stability a bezpečnosti slouží struktura směrovačů. Ta zajistí doručení paketů přes jiné funkční směrovače v případě poruchy. Postupem času se Internet rozšířil z vojenské a akademické půdy do běžného provozu.

Původní návrh Internetu nepočítal s tak razantním rozvojem. V počátcích jeho vývoje se počítalo s 32bitovým adresováním. Tyto adresy jsou typu IPv4. Adresový prostor, který měl být nevyčerpatelný se v dnešní době razantně zmenšuje. Proto se v nejbližší době plně začne využívat protokol IPv6. Ten má narozdíl od miliardy adres svého předchůdce stovky sextiliónů těchto adres.

Jedním z následků rozvoje Internetu je masivní rozvoj webových prezentací. Pro mnohé firmy je takováto prezentace na Internetu způsob jak na svoji komerční činnost poměrně levně upozornit. Z toho, co zde bylo napsáno se může zdát, že Internet slouží jen jako zdroj zábavy nebo komerce. Skutečnost naštěstí vypovídá o pravém opaku. Vedle zábavy tak nalezneme na Internetu mnohé užitečné informace. Ty jsou většinou na Internet umístovány právě přes informační portály.

Výstavba informačních portálů mne natolik nadchla, že jsem se rozhodl svou bakalářskou práci věnovat právě jim. Informační systém, či portál umožňuje jednoduchou správu jednotlivých webových prezentací. Je mnohem pohodlnější pro uživatele vložit na server data přes vytvořené uživatelské rozhraní než je pracně vepisovat do klasického HTML kódu.

Tato práce popisuje vývoj informačního portálu městyse. V jednotlivých kapitolách je popsán návrh, implementace a testování potřebné pro vznik funkčního systému. Důležitou složkou systému je rozdělení jednotlivých rolí. I tuto problematiku zde máme vyřešenou.

Kapitola 2

Vymezení použitých pojmů

2.1 Městys

Městys se někdy také nazývá městečkem [10]. Co do velikosti a významu stojí mezi obcí a městem. V minulosti se jednalo o místa, kde se pořádaly pravidelné týdenní dobytčí trhy. Tímto privilegiem se lišily od běžných vsí. Pro tyto okolní vesnice plnil městys roli spádového místa. Na rozdíl od města je zde silněji zastoupeno zemědělství.

Samotná městečka začala vznikat od 14. století. Statut městečka uděloval panovník. Později, ve 20. století ministerská rada. Od roku 1948 do roku 2006 se přestal udělovat tento status. Od roku 2006 uděluje status městysu předseda Poslanecké sněmovny Parlamentu ČR.

Pojem městys není znám pouze v České republice. Označují se jím také městečka na Slovensku, v Dánsku, Rusku či Francii.

2.2 Informační systém

Pojem **informační systém** se skládá ze dvou slov: **informační** a **systém**. Nejdříve objasním význam těchto slov [9].

2.2.1 Informace

Pojem informace lze chápat z různých hledisek jinak. Existuje filozofické, komunikační, kybernetické či matematické pojetí informace. V oblasti informačních systémů a počítačů vůbec se používá kybernetické pojetí informace. Informací nazýváme data, která mají svojí sémantiku. Sémantika dat, znamená, že data mají svůj význam a jsou interpretována uživatelem. Jako data označujeme reprezentaci skutečnosti, která je schopna přenosu, uchování, interpretace či zpracování.

2.2.2 Systém

Systém lze chápat jako množinu prvků a vazeb mezi nimi, které jsou účelově definovány na nějakém nosiči. Obecný systém se skládá ze vstupní a výstupní části. Do systému vstupují tyto zdroje. Než dojdou na výstup systému, jsou transformovány. Transformace mění zdroje a probíhají zde procesy. Typickým rysem, ne však nutným, obecného systému je zpětná vazba. Díky zpětné vazbě nejsou výstupy systému závislé pouze na vstupech, ale i na stavu systému. Stavem systému jsou tedy zdroje. U zdrojů se zabýváme jejich **perzistencí**

(přetrváním) a **konzistencí** (splněním jistých pravidel o možných kombinacích hodnot zdrojů ve stavu). Schéma takového obecného systému je na obrázku 2.1



Obrázek 2.1: Obecné schéma systému

Informační systém je otevřený systém, jehož nosič používá konceptuální zdroje, konkrétně informace. Z tohoto vyplývá, že nenakládá s hmotnými zdroji, ale naopak se zdroji nehmotnými. Jedná se o model reálného systému, kde stupeň abstrakce se určí podle požadavků na výsledný systém.

2.3 Databáze

Databáze je bez nadsázky přímým nástupcem papírové kartotéky [4]. Umožňuje třídit obsažená data podle jednotlivých atributů. Dále podporuje vkládání nových položek a jejich zařazení do tabulky. Data v tabulce jsou **perzistentní**. Znamená to, že musejí být dostupná i po skončení aplikace. Databáze začaly vznikat v 60. letech minulého století. V počátcích vzniku databází se jednalo spíše o objektově orientované. Až v 70. letech byl položen základ relačních databází. Tímto základem je článek od E.F. Codd, který vyšel v časopise *Communications of the ACM*. V této publikaci byl zaveden zcela nový pojem **perzistentní data**. Cílem bylo dosáhnout datové nezávislosti. Došlo k oddělení logické struktury od implementace.

Pro ucelení přehledu o druzích databází uvádím některé v následujícím výčtu:

1. **Hiearchická databáze** – omezuje logické uspořádání dat na stromovou strukturu
2. **Síťová databáze** – založena na síťovém modelu, data uspořádány jako uzly rovinného grafu
3. **Relační databáze** – založena na relačním modelu
4. **Objektová databáze** - využití principů objektového programování
5. **Objektově relační databáze** – existence pohledů, kombinace vlastí relační a objektové databáze

Název relační databáze nevychází z toho, že mezi tabulkami jsou relace. Na tento mylný fakt můžeme často narazit v literatuře. Relace v relačním modelu označují samotnou tabulku.

2.3.1 Klíč

Klíč je jednoznačný identifikátor záznamu, řádku tabulky. Klíčem může být jediný sloupec nebo kombinace více sloupců tak, aby byla zaručena jednoznačnost. Pole klíče musí obsahovat hodnotu. To znamená, že nesmí obsahovat hodnotu `NULL`. V praxi se často používají umělé klíče, což jsou číselné nebo písmenné identifikátory. Každý nový záznam tak dostává vyšší hodnotu identifikátoru než ten předchozí.

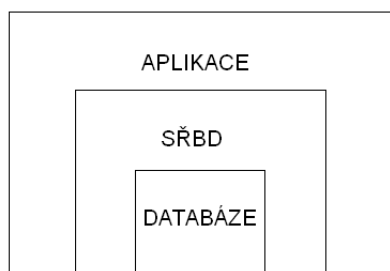
Dalším důležitým pojmem jsou nevlastní klíče. Slouží pro vyjádření vztahů a relací mezi databázovými tabulkami. Jedná se o pole nebo skupinu polí, která nám umožní identifikovat, které záznamy z různých tabulek spolu souvisí.

2.3.2 Integrita databáze

Integrita znamená, že uložená data jsou **konzistentní**. Lze vkládat pouze data, která splňují předem daná kritéria, jako je například datový typ slouce. Pro zachování integrity slouží mechanismus **integritních omezení**. Ten zabrání vzniku škody na datech při práci s databází. Díky tomu je možno docílit mazání dat, která ztratila v systému význam. Mezi entitní omezení patří doménové, referenční, nebo aktivní referenční omezení.

2.3.3 Systém řízení báze dat

Systémem řízení báze dat(dále SŘBD) rozumíme programovou vrstvu řešící operace nad databází. Názorně na obrázku 2.2. Hlavním cílem tohoto systému je odstínit aplikaci od samotných technických detailů(databáze). SŘBD zajišťuje vytvoření databáze, tabulky, vkládání nebo vyhledávání. Tyto systémy umožňují využití **triggerů**, provádějících předem definované činnosti nad databázovou tabulkou. Další funkcí SŘBD je správa klíčů. Ty mají své vlastní indexování, dodržují unikátnost ve sloupcích, ve kterých je primární nebo unikátní klíč. Pomocí indexování je implementováno také fulltextové vyhledávání. V neposlední řadě hlídá SŘBD integritu dat. Zamezí tak např. vložení duplicitního řádku, či vložení hodnoty `NULL` do sloupce, ve kterém být nemí. SŘBD společně s databází tvoří databázový systém. V dnešní době jsou na trhu např. následující systémy: Oracle, DB2, Firebird, Microsoft SQL Server, MySQL, SQLite, PostgreSQL.



Obrázek 2.2: Schéma databázového systému

2.3.4 Normalizace a normální formy

Jestliže máme relaci na složených či vícehodnotových doménách, která není ve skutečnosti relací ve smyslu relačního modelu dat označujeme ji jako **nenormalizovanou**. Proces převodu z **nenormalizované** na **normalizovanou** formu nazýváme **normalizací**. Normalizovaná tabulka obsahuje pouze skalární hodnoty. V tomto případě se jedná o **1. normální formu**. Těchto normálních forem je celkem 6. V této hierarchii platí, že pokud má relace splňovat *k-tou* normální formu, musí splňovat podmínky *k-1* normální formy a něco na víc. Poslední dvě (4. a 5.) normy definují vlastnosti z hlediska vícehodnotových závislostí na spojení.

Přehled normálních norem (dále jen NF):

- **0NF** – tabulka obsahuje alespoň jeden sloupec, který může obsahovat více druhů hodnot.
- **1NF** – všechny sloupce tabulky nelze dále dělit. Jeden sloupec neobsahuje více druhů.
- **2NF** – tabulka obsahuje pouze atributy, které jsou závislé na klíči.
- **3NF** – v tabulce neexistují žádné vztahy mezi neklíčovými atributy.
- **4NF** – sloupce tabulky popisují jeden fakt nebo jednu souvislost.
- **5NF** – tabulka by se přidáním libovolného nového sloupce rozpadla na více tabulek.

2.3.5 Vztahy mezi tabulkami

Data ve dvou nebo více tabulkách mohou spolu navzájem nějak souviset. Pro provázání těchto dat slouží relace. Rozlišujeme několik typů těchto relací:

1. **1:1** – pokud právě jednomu záznamu tabulky odpovídá právě jeden záznam jiné tabulky
2. **1:N** – nejpoužívanější typ, u kterého odpovídá jednomu záznamu tabulky více záznamů z jiné tabulky
3. **M:N** – nejméně vyskytující se typ vazby. Více záznamům jedné tabulky odpovídá více záznamů jiné tabulky. V praxi se nahrazuje tento typ přidáním další tabulky a vznikem vztahů 1:M a 1:N

Kapitola 3

Vývoj informačních systémů

V této kapitole si objasníme jak vznikaly informační systémy. Vývoj informačních systémů záležel nejen na vývoji hardwaru, ale i na vývoji programovacích jazyků [4]. Díky tomu se přístup k informačnímu systému strukturoval do jednotlivých modulů. Průřez vývoje je shrnut dle jednotlivých typických období.

3.1 50. léta

Období padesátých let minulého století se v oblasti informačních systémů označuje za tzv. *prehistorické období*. Typickým rysem je definice dat a definice procesů dohromady. Data tak byla součástí procesů z čehož vyplývá, že nemohla být sdílena více procesy. Veškerá data, algoritmy, uživatelské rozhraní a modely byly implementovány v jednom systému. Neexistovala ani metodologie vývoje software. Ten díky tomuto stavu vznikal živelně a tvůrci neměli klasifikovány všechny části těchto systémů. Vstupem programů byly děrné štítky, neexistovaly ani monitory či klávesnice.

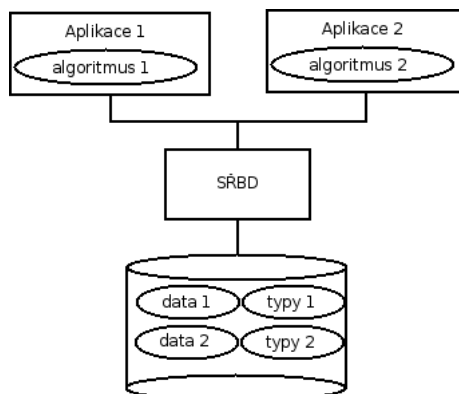
3.2 60. léta

V tomto období proběhlo v první řadě oddělení dat od procesů. Tento stav již umožnil pracovat procesům nad více množinami dat. Oddělení dat a procesů nebylo ještě v plné míře dokonáno. Popis dat stále zůstal fyzicky v programech. To stále neumožnilo sdílení dat více procesům. Výsledkem tohoto období bylo, že definice programu mohla pracovat s více databázemi a jedním systémem tak spravovat více sad údajů.

3.3 70. léta

Již od počátku 70. let se začal vyvíjet doslova revoluční přístup k databázi, který se používá do dnes. Jedná se o systém řízení báze dat (viz. 2.3.3). Data jsou uložena samostatně v databázi. Databáze je buď na jednom místě nebo je distribuovaná. Díky tomuto umístění je dosaženo nezávislosti definice procesů na datech (viz. obrázek 3.1). Data jsou strukturovaná a zavedením pojmu transakce se zvyšuje jejich konzistence. Jednou z hlavních výhod uložení dat v databázi je úplný přehled o datech. Dále pak snadný přístup k těmto datům pomocí speciálních jazyků. Toto vše vede i k atomičnosti transakcí, které umožňují zachovat integritu dat v databázových tabulkách. Data jsou sdílena, což umožňuje multiuživatelský přístup s případným omezením jednotlivých pohledů. V neposlední řadě je také

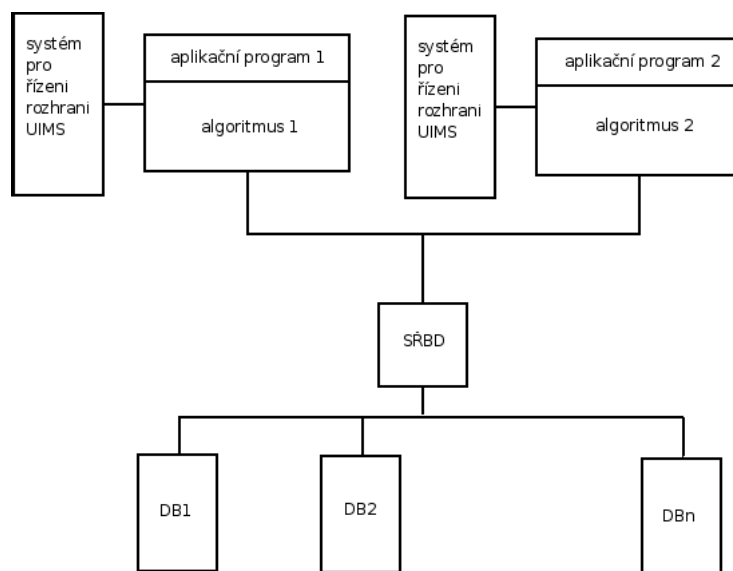
snadnější zabezpečit data v databázi oproti předcházejícím postupům. Taková zabezpečení se provádějí na základě oprávnění přístup k těmto tabulkám.



Obrázek 3.1: Systém řízení báze dat

3.4 80. léta

Nástupem 80. let se začalo čím dál více pohlížet na interakce s uživatelem. Aby tyto interakce byly pro uživatele co nejpohodlnější, začala se propracovávat čím dál více uživatelská rozhraní. Přístup k datům se oproti předešlým obdobím téměř nezměnil. Samotná definice uživatelského rozhraní není součástí definice procesů (viz. obrázek 3.2). Tato prostředí začala být standardizována. Díky tomu mohl uživatel přistupovat k aplikaci z různých variant rozhraní jako jsou displeje, tiskárny nebo datové kanály.



Obrázek 3.2: Interakce s uživatelem

3.5 90. léta a současnost

Pro 90. léta je typický nástup tzv. *workflow*. Díky tomuto přístupu může uživatel umístit definice procesů mimo aplikace. V těchto letech se vyvinuly speciální specifikační prostředky zejména pro deklarativní definici procesů. Tyto prostředky umožňují kromě definice vlastního procesu provádět i formální dokazování správnosti, proveditelnosti nebo testování samotných procesů.

Systémy od konce 90. let do současnosti již nemění stav systému. Hlavním předpokladem je kladení předem nspecifikovaných dotazů. Informační systémy jsou v dnešní době velmi rozšířeny. Na Internetu je možné najít mnoho odkazů na různé takovéto systémy. Využití našly i například v bankovním sektoru. Samotné elektronické bankovníctví je jeden propracovaný systém. Ale nejen zde. Dalším příkladem by mohly být systémy sportovních klubů, rádií, televízí a dalších masmédií.

3.5.1 Informační portály měst a obcí

Touto kapitolou bych chtěl samostatně rozvést aktuální stav elektronických systémů obcí, měst a městysů. Díky rozvoji Internetu a hlavně jeho technologií můžeme v dnešní době shlédnout nepřehledné množství takovýchto elektronických prezentací. Tyto prezentace samozřejmě musejí splňovat jistá kritéria. Mezi ně patří například povinně zveřejňované informace o městysu. Do těchto povinných informací patří:

- **Oficiální název** – plné znění názvu daného městyse
- **Důvod a způsob založení** – uvedení jakým způsobem a kdy byl městys založen. Toto založení je položeno na základě zákona č. 128/2000Sb.(viz. 4.1).
- **Organizační struktura** – zde nalezne návštěvník systému podrobné informace o uspořádání byrokracie v obci. Jedná se o informaci, kdo tvoří obecní zastupitelstvo, jednotlivé odbory a spolky.
- **Kontaktní spojení** – uvedení kontaktu, na kterém nalezneme budovu úřadu. Do těchto kontaktů patří například adresa, telefon, e-mail, elektronická podatelna a jiné.
- **Způsob případných plateb** – rozvedení informace pro případné plátce jistých poplatků. Uvedeno číslo účtu, na kterém probíhá zasílání poplatků, případně jiná alternativa plateb.
- **Identifikační údaj IČ** – toto číslo je osmiferné a udává identifikační číslo právnické osoby. V českém prostředí se zažilo i označení IČO, což znamená *identifikační číslo organizace*.
- **Identifikační údaj DIČ** – uvedení jednoznačného unikátního čísla daňového poplatníka. Každý takovýto daňový subjekt má své číslo DIČ. Číslo DIČ přiděluje příslušný finanční úřad.
- **Rozpočet** – každý občan má právo vidět rozpočet obce na další roky. Tento rozpočet musí být zveřejněný na úřední desce po schválení místním obecním zastupitelstvem.
- **Žádosti o informace** – učují obyvatelům, jakým způsobem mohou v místě bydliště podávat tyto žádosti. Ve většině případů se jedná o písemné. U každé této žádosti musí být provedena archivace na straně úřadu. Existuje i možnost ústní žádosti. V tomto případě se žádné archivace provádět nemusejí.

- **Příjem žádostí** – občan je v této části informován o stavu žádosti, ale hlavně o tom, jakým způsobem má žádosti podávat.
- **Opravné prostředky** – dle zákona č. 500/2004Sb. je občan informován jak se má úřad chovat v oblasti veřejné správy.
- **Formuláře** – tato sekce sděluje občanům kde si mohou vyzvednout formuláře potřebné pro komunikaci s úřadem.
- **Návody pro řešení nejdůležitějších životních situací** – každý dostane podrobnou informaci, pro danou životní situaci. Jedná se například o situaci, kdy občan neví jak vyřídit daná podání, kam se obrátit s potřebnými lhůtami. Také jsou zde obsaženy údaje o tom, jak oslovit příslušné zodpovědné orgány.
- **Předpisy** – pokud občan potřebuje nahlédnout do státem závazných předpisů, nalezne odkaz, kam s tímto požadavkem zajít.
- **Sazebník úhrad za poskytování informací** – je nutné, aby občan byl uvědoměn o veškerých poplatcích, které od něj bude úřad za zveřejněné informace požadovat.
- **Výroční zprávy** – povinností úřadu je minimálně jednou za rok zveřejnit výroční zprávu. Ve výroční zprávě nalezne občan hlavně údaje o hospodaření obce.
- **Organizace zřizované obcí** – téměř každá obec zaštiťuje svojí finanční podporou nějaké spolky nebo organizace. V této části je občan informován o těchto organizacích.

Při prozkoumávání internetových prezentací a elektronických systémů měst, obcí a městysů jsem došel k závěru, že současná situace na českém Internetu není příliš příznivá. Mezi hlavní nedostatky bych vyzdvihl hlavně přílišnou podobnost těchto stránek. Již běžným okem pozná uživatel, že stránky obce A a stránky obce B vytvářela stejná firma. Rozdíl mezi těmito stránkami je totiž často pouze v barvě prezentace. Pravdou je, že pokud se každý zaměří pouze na stránky svého města, tak tento jev až tolik nevadí.

Abych nezůstal pouze jen u kritiky současného stavu systémů tak je třeba i chválit. Při podrobném zkoumání systémů člověk narazí i na opravdu vyvedené prezentace. U těchto portálů je vhodně zakomponovaný typický prvek krajiny do vzhledu webového layoutu. I taková maličkost na mne zapůsobila tak, že jsem si v podvědomí tyto stránky uchoval v paměti. Samotná typická krajina či stavba ovšem není zařízenou výhrou, jak by se mohlo z předešlé věty zdát. V tomto ohledu na věc často rozhodují detaily a subjektivní pohled jedince na danou věc. Rozdílný názor bude mít jistě odborník na webové stránky než například běžný občan, který pouze stránky prohlíží.

Kapitola 4

Závazné zákony pro městys

Stejně jako každá fyzická či právnická osoba se i záležitosti, týkající se městyse musí řídit dle platných zákonů České republiky. Pro záležitosti městyse a jeho informačních portálů platí především 3 následující zákony, které okrajově popíšu níže. Při tvorbě tohoto systému jsem se i já, jako autor portálu musel těchto zákonů držet. Všechny tyto zákony vznikly v roce 2000. Jejich účinnost se vztahuje nejen na obce a města, ale i na městysy, které byly obnoveny v roce 2006 (viz. 2.1).

4.1 Zákon o obcích

Jedná se o zákon, který byl schválen 12. dubna 2000. Přesné označení tohoto zákona je *zákon č. 128/2000Sb.*[12]. Zákon definuje v první řadě co je vůbec obec. Podmínkou pro to, aby obec byla městem je nutnost mít alespoň 3000 obyvatel. Statut města uděluje předseda Poslanecké sněmovny. V případě sloučení více obcí, je-li alespoň jedna městem, tak po sloučení opět vzniká město. V opačném případě, při rozdělení města na menší správní celky, je městem ta část, která zahrnuje minimálně 3000 obyvatel. Speciálním typem města **statutární město**. Území těchto měst se může dělit na městské obvody s vlastní samosprávou.

Obce, města a městysy musejí mít svůj vlastní prapor a znak. Jestliže nenavrhnou vlastní motiv, je jim přidělen znak a prapor od předsedy Poslanecké sněmovny. Obec může navrhnout předsedovi Poslanecké sněmovny návrh na změnu znaku a praporu.

Obec má právo na svou samosprávu. Samospráva plní úkoly v souladu se zákonem a s potřebami dané obce. Určí-li vyšší moc výkon státní správy obci, nazývá se poté území obce **správním obvodem**. Povinnosti pro občany může obec ukládat v samostatné působnosti obecně závaznou vyhláškou. Obecně závazné vyhlášky a nařízení obce musejí být vyhlášeny. Vyhlášení se provede tak, že se umístí vyhláška na úřední desku dané obce. Právní předpis nabývá účinnosti 15. dnem po vyhlášení. V případě nutnosti je možno, aby právní předpis nabyl účinnosti dnem vyhlášení. Důvody, proč k tomuto došlo, musejí být ve vyhlášce uvedeny.

Dle paragrafu číslo 112 musí být úřední deska umístěna na místě veřejně přístupném. Umisťuje se zpravidla na budově, kde má obec sídlo úřadu. Údaje na úřední desce klasické a té elektronické musí být totožné.

Obec je spravována zastupitelstvem. Mezi další orgány obce patří rada obce, starosta a obecní úřad. Obce mohou zřizovat obecní policii. Jedná-li se o město či městys plní funkci zastupitelů zastupitelstvo města či městyse.

Zákon o obcích také určuje, kdo je občanem. Toto určení sděluje také práva a povinnosti občanů. Mezi práva patří například po dovršení věku 18–ti let právo volit, hlasovat v referendu obce, vyjadřovat se k rozpočtu obce nebo požadovat projednání určité záležitosti. Tato práva má i fyzická osoba, která pouze vlastní na území obce nemovitost.

Každá obec má svůj název. Pokud zažádá o změnu názvu tak musí podat žádost na Ministerstvo vnitra. Obec má právo rozhodovat o názvech svých částí, ulic a jiných veřejných prostranství. Po provedení takovéto změny musí obec změnu nahlásit na příslušný Katastrální úřad. Názvy ulic a městských částí se uvádí v českém jazyce. Jestliže na daném území žije alespoň 20% obyvatel jisté národnostní menšiny a tato skutečnost se prokázala při posledním sčítání lidu, mohou požádat tito obyvatelé o vícejazyčné názvy těchto částí v referendu.

Jednou z nejdůležitějších částí zákona o obcích jsou informace o veřejné vyhlášce. Pokud tato vyhláška porušuje zákony, pozastaví okresní úřad její výkon. Od podání tohoto návrhu na pozastavení plyne lhůta místnímu zastupitelstvu v rozsahu 3 měsíců. Pokud do konce této lhůty nezjedná nápravu, celou věcí se zabývá Ústavní soud. Po rozhodnutí soudu o zrušení vyhlášky je povinna obec vyvěsit toto oznámení do 15 dní na úřední desku.

4.2 Zákon upravující počet zastupitelů

V tomto paragrafu zákona se pojednává o stanovení počtu volených zastupitel dané obce [12]. Počet zastupitelů se určí dle tabulky 4.1. Počet zastupitelů, který má být zvolen se oznámí na úřední desce nejpozději do 2 dnů od tohoto ustanovení. Počet obyvatel pro toto ustanovení se bere vždy k 1. lednu daného roku, kdy jsou konané volby do místního zastupitelstva. Není-li v dané chvíli v obci zvoleno zastupitelstvo, určí počet členů na základě stejných podmínek okresní úřad. Je-li určen špatný počet zastupitelů, který nesouhlasí s tabulkou 4.1, určí okresní úřad počet zastupitelů na dolní hranici daného rozmezí.

Počet obyvatel	Počet členů zastupitelstva
do 500 obyvatel	5 až 9 členů
nad 500 do 3000 obyvatel	7 až 15 členů
nad 3000 do 10000 obyvatel	11 až 25 členů
nad 10000 do 50000 obyvatel	15 až 35 členů
nad 50000 do 150000 obyvatel	25 až 45 členů
nad 150000 obyvatel	35 až 55 členů

Tabulka 4.1: Počet členů zastupitelstva dle počtu obyvatel

Pro účely této práce jsem určil, že městyš Zubříč bude mít okolo 6000 obyvatel. Znamená to tedy, že v zastupitelstvu musí být 11 až 25 členů. Při jiném počtu zastupitelů program upozorní starostu, který tyto členy do systému zadává, že zadal špatný počet těchto obecních činitelů.

4.3 Zákon o informačních systémech veřejné správy

Pro účely informačních systémů veřejné správy byl dne 14. září přijat v platnost zákon č. 365/2000Sb.[12]. Tento zákon stanovuje práva a povinnosti, které souvisejí s vytvářením, užíváním, provozem a rozvojem informačních systémů pro veřejnou správu. Dle tohoto zákona je určen jako správce subjekt, který za daný systém zodpovídá. Na systém je možno získat atestaci, kterou provádějí atestační střediska. Atest se vydává maximálně na dobu 5 let. Systém se vyznačuje především dálkovým přístupem například s využitím Internetu.

Zákon se nevztahuje na informační systémy veřejné správy vedené zpravodajskými službami, Policií České republiky a Vězeňskou službou České republiky. Dále se nevztahuje na ministerstva financí, obrany, vnitra nebo na systémy Národního bezpečnostního odboru.

Kontrola dodržování pravidel informačních systémů je na straně příslušného ministerstva. Jestliže nastane situace, že ministerstvo zjistí nějaké nedostatky, vyzve orgán veřejné správy k nápravě tohoto nedostatku. Předmětem této výzvy je soupis nedostatků, které mají být v napraveny. Součástí tohoto upozornění je také lhůta, do kdy má být náprava provedena. Tato lhůta nesmí přesáhnout dobu 6 měsíců.

Jednou z nejdůležitějších pasáží tohoto zákona je oddíl ujednávací o zachování mlčenlivosti. Správce informačního systému má sice možnost přistupovat k citlivým datům, ale nikdy je nesmí použít ku svému či cizím prospěchu nebo škodě.

Zákon vymezuje i práva a povinnosti správce systému. Jedním z hlavních nařízení je, že správce může s daty nakládat pouze v rámci systému. Nemůže je tedy například kopírovat nebo dále jinak šířit. Další důležité nařízení nařizuje k tomu, aby se pořízená data uchovala na dobu nejméně tří let. Tato data lze ovšem zničit, jestliže je nelze přeposlat zpět či doručit nebo vznikla-li by z důsledku těchto dat škoda. Při porušení některých z uvedených práv hrozí sankce až do výše 1.000.000 Kč.

Kapitola 5

Neformální specifikace

Hlavním cílem projektu je vhodně navrhnout a poté naprogramovat informační systém městyse Zubříč. Městys Zubříč ve skutečnosti neexistuje. Aby nevznikly jakékoliv komplikace, tak jsem si jej vymyslel. Městys má přibližně 6000 obyvatel. Tento údaj je podstatný pro určení omezení na počet členů místního zastupitelstva.

Mezi stěžejní problémy, které je třeba vyřešit a vhodně naimplementovat je funkční úřední deska. Ta musí obsahovat mimo jiné datum vyvěšení a datum platnosti. Dále musí být vyřešena možnost pro občany vidět různé obecní vyhlášky a nařízení. Občan bude mít také možnost dávat na jevo své názory buď neveřejně pomocí elektronické podatelny a nebo veřejně v diskuzi. Další možností jak reagovat na položené otázky jsou vkládané ankety.

V navrhovaném městyse je také bohatý kulturně–společenský život. Funguje zde fotbalový klub, Sbor dobrovolných hasičů, knihovna nebo kino. Každá z těchto organizací má svou fotogalerii. Jednotliví vedoucí daných oddílů tyto fotogalerie budou mít možnost spravovat. Vedoucí fotbalového klubu a Sboru dobrovolných hasičů navíc spravují své sekce v diskuzi. Aby občan získal ucelený přehled, kdy se koná konkrétní akce, bude implementován kalendář akcí. Díky němu získá veřejnost zaručený přehled o kulturně–společenském dění.

Aby se docílilo růstu obchodu v okolí městyse bude v systému možnost vkládat místní firmy. Pro nefiremní objekty dále možnost vkládat inzeráty s poptávkou a nabídkou nejen práce.

V dnešní době, kdy je mnohem větší migrace obyvatel než dříve je nutno, aby mohl stránky navštívit i cizinec. Aby se tomuto docílilo je valná část systému trojjazyčná. Vedle rodné češtiny existuje anglická nebo německá verze. Nejen díky tomu se docílí funkčnosti systému pro široké spektrum obyvatel.

Kapitola 6

Návrh systému

Kapitola návrh systému pojednává o samotném postupu předpřípravy na programování informačního systému. Nejdříve bylo nutno si určit jednotlivé role. O rolích v systému pojednává kapitola 6.2. Veškeré údaje jsou uloženy v databázových tabulkách. Popis tabulek a celé databáze je popsán v kapitole 6.1. Pro samotné naprogramování databáze bylo nutno převést ER diagram na schéma databáze (viz.6.3). Při programování jsem musel návrh několikrát doupravovat. Některé položky byly navrženy zbytečně a neefektivně. Naopak při prvotním návrhu část položek chyběla. Tento jev nepovažuji za důsledek špatného návrhu jak by mohl někdo podotknout. Zkušenosti z praxe mne vedou k závěru, že se jedná o obvyklý jev. Vývoj tak postupuje iterativně. Vyrobí se část systému, konzultuje se se zákazníkem. Takto se dojde podle mého názoru k nejlepšímu výsledku a spokojenosti obou stran.

Pro účely návrhu existuje mnoho vývojových prostředí, které za programátora udělají notnou dávku práce. Tato prostředí jsem při návrhu a následném vývoji systému nepoužil. Psaním systému jsem se totiž chtěl i naučit nové věci a ne se jen dívat jak automat pracuje za mne. Díky této obtížnější a také zdlouhavější variantě vývoje jsem získal mnoho zkušeností, které budou do budoucího profesního života více než užitečné.

Při návrhu informačního systému jsem nejdříve musel prostudovat zákony (viz. kapitola 4). Samozřejmostí byla i inspirace z již existujících systémů na Internetu. Tímto zkoumáním jsem došel k mnohým nápadům, které jsem buď implementoval stejným způsobem, nebo jsem je vylepšil. Po prostudování těchto záležitostí začal vznikat samotný návrh systému.

6.1 ER diagram

V celém znění se nazývá **Entity Relationship Diagram**. Využívá se k modelování dat. Tato data potřebujeme v systému uchovat. Jedná se tedy o statická data. ER diagram patří do metody unifikovaného vývoje aplikací. Modeluje data na vyšší úrovni než jsou jen datové typy a hodnoty.

Pro ucelení této publikace považuji za nutné uvést následující pojmy související s ER diagramy.

- **Entita** – objekt reálného světa. Například konkrétní uživatel systému
- **Entitní množina** – je souhrn množin entit téhož typu. Například uživatelé

- **Vztah** – popis asociací mezi jednotlivými entitními množinami. Například uživatel vložil konkrétní příspěvek
- **Vztahová množina** – množina stejných vztahů. Například uživatelé vkládají příspěvky.

Dle obrázku 6.1 nyní popíšu jednotlivé entity systému.

6.1.1 Občan a Typ role

Tabulka **Občan** uchovává informace o jednotlivých občanech. Mezi tyto informace patří **jméno**, **příjmení**, **titul**, **přihlašovací jméno** a **heslo** pro bezpečné přihlášení. **Heslo** je uchováno v tabulce v zahashované podobě. Při psaní jsem využil 32 bitový MD5 hash. Díky tomuto zamezím případnému nežádoucímu přístupu ze strany útočníka. Sloupec **Přihlašovací jméno** je unikátní. Nemůže tedy nastat situace, že by se dva občané přihlašovali pod stejným jménem. Jednou z nejdůležitějších informací uchovávaných o občanech je **typ role**. Pomocí uvedené hodnoty se odkazují na tabulku **Typ role**, kde je uchována informace o rolích.

S tabulkou **Občan** úzce souvisí tabulka uvádějící rozdělení typu role. Na základě tohoto rozdělení je umožněno uživateli přistupovat do jednotlivých sekcí systému. Role v systému jsou následující: **starosta**, **administrátor**, **vedoucí SDH**, **vedoucí sociálního odboru**, **vedoucí stavebního odboru**, **vedoucí fotbalového klubu**, **běžný občan**, **vedoucí knihovny** a **vedoucí kina**. Zastupitelé obce jsou umístěni v samostatné tabulce.

6.1.2 Vyhlášky a nařízení

Úložným prostorem pro jednotlivé vyhlášky a nařízení je tabulka **Vyhlášky a nařízení**. Hlavní částí je sloupec **Cesta k datům**, kde je uložena cesta k adresáři s daty. Díky tomu, že data nejsou uložena přímo v tabulce jako **typ blob**, nedochází k tak velkému zatížení databáze při provozu. Neméně důležitá položka **Zobrazit** určuje zda-li se má daná vyhláška zobrazit. Sloupec **Typ sdělení** informuje o druhu zobrazované informace.

6.1.3 Úřední deska

Jednou ze stěžejních zobrazovaných informací je zcela jistě **úřední deska**. Vkládat nové údaje na úřední desku smí pouze starosta obce. Mezi zobrazované informace patří **Datum vyvěšení** a **Datum sejmutí**. První z těchto údajů se vkládá automaticky při vložení nové informace dle serverového času. **Datum sejmutí** informuje uživatele do kdy bude na úřední desce dokument k dispozici. Aplikace při vypršení tohoto data automaticky zneviditelní položku pro běžné občany. Jednou z dalších možností jak zabránit zobrazení informace je za pomoci položky **Zobrazit**. Díky tomuto může starosta kdykoliv danou položku zneviditelnit. Data opět nejsou fyzicky uložena v tabulce, ale pouze jako odkaz na umístění v adresáři na serveru.

6.1.4 Aktuality

Pro rychlé zobrazení aktuální informace je k dispozici v systému tabulka **Aktuality**, ve které jsou uloženy textové zprávy pro občany. Sekci aktuality spravuje starosta, který opět může konkrétní aktualitu zneviditelnit za pomoci sloupce **Zobrazit**.

6.1.5 Elektronická podatelna

Umožňuje občanům zadávat své podněty z pohodlí svého domova. Nahrazuje tedy klasickou podatelnu. Pro podání je zde umístěn formulář, kde je nutno zadat svou **e-mailovou adresu**, aby mohl starosta zodpovědět na daný mail. Běžný občan má k dispozici pouze již zmíněný formulář. Starosta po přihlášení vidí seřazené dle data přijaté podněty. Mezi hlavní položky těchto podání patří **E-mail**, **Datum podání**, **Předmět zprávy** a **Typ zprávy**.

6.1.6 Ankety a Kontrola ankety

Pro zjištění veřejného mínění jsou v systému zavedeny ankety. Ankety může zadávat **administrátor**. Při vyplnění ankety je nutno uvést jednotlivé možnosti a **název ankety**. Po dobu platnosti dané otázky vidí občané graficky znázorněno rozložení odpovědí v grafech. Zobrazené grafy využívají grafického prvku, který je dle procentuálního rozložení odpovědí dlouhý.

Úzce s tabulkou **ankety** souvisí tabulka **Kontrola ankety**. Jedná se o pomocnou tabulku, ve které je vždy uložena dvojice hodnot **IP adresa** a **číslo ankety**. Na základě těchto dvojic zamezím opětovnému hlasování uživateli ze stejné IP adresy. Další možnou variantou by bylo ukládat podobné údaje do **cookies**, ale ty se dají snadněji vypnout a tedy by byla tato ochrana snadno obejitelná.

6.1.7 Fotogalerie

Fotogalerie tvoří jednu velkou tabulku, která je díky sloupci **Sekce** rozdělena na jednotlivé části. V systému jsou zavedeny části pro **Obecní**, **hasičskou**, **fotbalovou fotogalerii**, dále pak pro **fotogalerii kina** a **knihovny**. Tyto jednotlivé sekce si spravují vedoucí jednotlivých organizací, obecní fotogalerii poté starosta. Fotogalerie je ošetřena aby šlo vkládat pouze grafické formáty dat. Opět, jako u všech tabulek, kde se zachází s binárními daty, je zde sloupec **cesta k datům**, který odkazuje na uložení fotografií v adresáři **fotogalerie**.

6.1.8 Diskuze

Díky diskuzi si mohou zaregistrovaní nebo i nezaregistrovaní uživatelé psát na libovolná témata. Diskuze je rozdělena do dílčích sekcí. Rozdělení je určeno pomocí **sekce diskuze**. Samostatné sekce si spravují jednotliví vedoucí. Při vložení příspěvku od nezaregistrovaného uživatele se v hlavičce příspěvku zobrazí **název host**. Pro účely zabezpečení proti útokům různých botů ze strany Internetu je nutno opsat kód vygenerovaný v obrázku nazývajícím se **captcha**. Po úspěšném vyplnění je příspěvek vložen do dané sekce s aktuálním časem a datem.

6.1.9 Organizace v obci

Obsahem tabulky je přehled organizací v obci. Tyto jednotlivé položky slouží jako **cizí klíče** na tabulky jednotlivých dílčích organizací. Přehled organizací je uveden v následujících kapitolách.

6.1.10 Knihovna

První z organizaí je knihovna. Sekce knihovny slouží jako on-line katalog zde uložených knih. Pro snadnější přehled jsou o knihách uloženy informace: **název knihy, autor, rok vydání a nakladatelství**. Díky této struktuře má uživatel umožněnu snadnější orientaci při hledání knihy.

6.1.11 SDH

Název sekce SDH znamená v plném znění sbor dobrovolných hasičů. Cílem tabulky je uchovat informace o členech hasičského sboru. Mezi ukládanými informacemi nalezneme **jméno hasiče, příjmení hasiče a jeho funkci v oddíle**. Každý návštěvník tak získá ucelený přehled o současném stavu sboru hasičů a jejich jednotlivých funkcích.

6.1.12 Kino

Sekci kino spravuje vedoucí kina. Ten má možnost vkládat do této tabulky údaje o vysílaných filmech. Obsah tabulky tedy tvoří on-line přehled vysílaných a plánovaných filmů v místním kině. Uživatel je informován o **názvu filmu, vstupném** a také o rozsahu vysílání a popisu filmu. Opět se nezobrazují již odvysílané filmy v textovém výpisu.

6.1.13 Fotbalový klub

Údaje v sekci fotbalový klub informují o současné soupisce klubu. U jednotlivých hráčů je i uveden **počet gólů**. Dalšími zobrazovanými informacemi je vedle **jména, příjmení, pozice a čísla dresu** také **datum narození**. Tyto zobrazované informace považuji za zcela dostatečné, aby si návštěvník sekce věnující se fotbalu udělal ucelený dojem o současném stavu klubu.

6.1.14 Kalendář akcí

Cílem tabulky **kalendář akcí** je občanovi sdělit v přehledném uspořádání přehled konaných a plánovaných kulturních akcí. Kompletní informaci o konané akci tvoří čtveřice: **Název akce, datum konání, organizátor a samotný popis akce**. Výstupem tabulky je graficky uspořádaný kalendář. V kalendáři jsou u dní, ve kterých se koná akce vyznačena souvislá políčka graficky a také odkazem na bližší informace o akci.

6.1.15 Inzerce

Pro účely vyvěšování inzerátů je v systému sekce **inzerce**. Rozdělení inzerce je na čtyři části: **prodám, koupím, práce nabídka a práce poptávka**. Platnost jednotlivých inzerátů je omezena samotným systémem na jeden měsíc. Tuto lhůtu jsem zvolil pro zřehlednění celkové inzerce. Pomocí sloupce **druh** rozpozná systém, do které sekce daný inzerát spadá a také jej v této sekci zobrazí. Pro zvýšení přehlednosti o autorech inzerátů mohou vkládat inzerci pouze zaregistrovaní uživatelé.

6.1.16 Podniky

Pro účely zvýšení konkurenceschopnosti místních podniků jsem připravil sekci **podniky**. Zde může uživatel po přihlášení vkládat firmy z místních částí městyse. Pro úplnost je

nutno o daném podniku uvést informace o názvu podniku, telefonu, e-mailu, adrese a popisu činnosti podniku.

6.1.17 Výroční zpráva

Dle zákona číslo 128 (více v kapitole 4.1) musí obec zveřejňovat výroční zprávu za daný rok. Aby systém splnil tyto požadavky je zde uložena tabulka s těmito výročními zprávami. Data nejsou fyzicky uložena v tabulce, ale je zde pouze odkaz na daná data ve sloupci cesta k datům. Dalším důležitým údajem je rok zprávy, který slouží k informaci o roku, ke kterému se vztahuje daná výroční zpráva.

6.1.18 Zastupitelstvo

Pro účely obecního zastupitelstva je určena tabulka s názvem Zastupitelstvo. Může obsahovat 11 až 25 položek. Pro zavedení systému je tedy nutno inicializovat tuto tabulku daty o konkrétních zastupitelích. Počet zastupitelů je daný dle počtu obyvatel (viz. tabulka 4.1). O jednotlivých zastupitelích je uchována informace o jméně, příjmení, funkci v obci, přihlašovací jméně a hesle. Pro zvýšení bezpečnosti je heslo zahasovanáno pomocí MD5 kryptografie. Mezi zastupiteli se určí dle sloupce funkce v obci jejich pracovní zařazení. Zda – li se jedná o běžné zastupitele, či vedoucí stavebního odboru nebo vedoucí sociálního odboru.

6.1.19 Hlášení

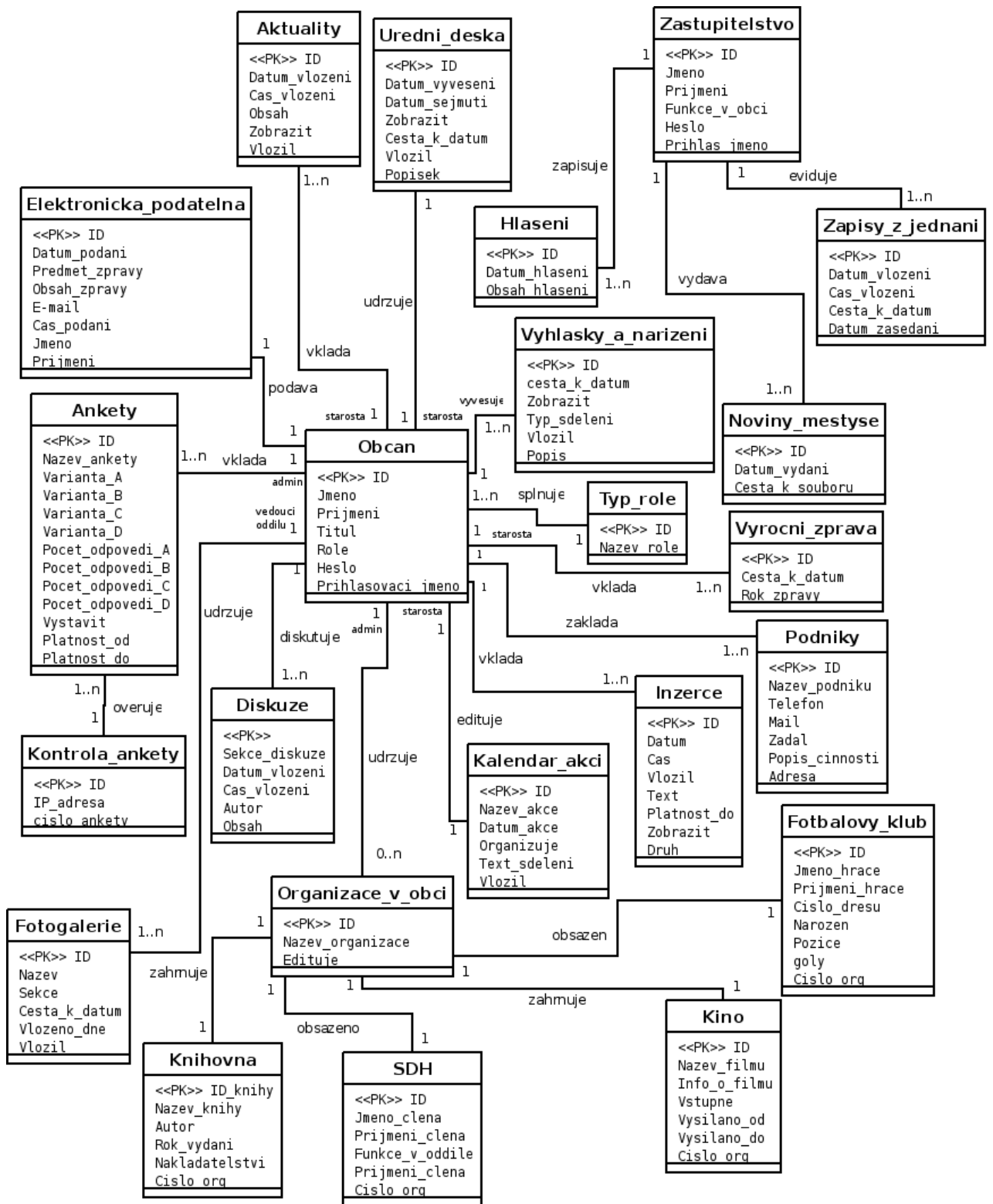
Může nastat situace, že proběhne hlášení rozhlasu městyse v době, kdy občan není přítomen. Aby se zamezilo neinformovanosti obyvatel, zavedl jsem v systému tabulku, ve které jsou uchovány přepisy těchto hlášení. Opět je použit pouze odkaz na data ve sloupci obsah hlášení. Pro úplnost tabulka obsahuje i sloupec datum hlášení, díky kterému je zřejmé, kdy k hlášení došlo.

6.1.20 Noviny městyse

Námi navrhovaný městys má i svoji redakci novin. Pro úplnost rozšíření těchto výtisků slouží i elektronická varianta výtisků. Tato data jsou uložena v tabulce s názvem noviny městyse. V tabulce je uchováno datum vydání a cesta k souboru. Tabulka opět spolupracuje s adresářovou strukturou na serveru.

6.1.21 Zápisy z jednání zastupitelstva

Pro účely uchování zápisů z jednání místního zastupitelstva je určena tato tabulka. Uchovává informaci o datu zasedání, datu a času vložení. Pro zabezpečení správného vložení těchto dat je uzpůsoben program tak, aby zamezil vložení dat o zasedání, které ještě nebylo. Vkládat jde tedy pouze zpětně. Data jsou fyzicky uložena v adresáři a v tabulce nalezneme odkaz na ně.



Obrázek 6.1: ER diagram navrhovaného systému

6.2 Rozdělení rolí v systému

System městysu umožňuje uživateli se přihlásit a provádět jako zaregistrovaný příslušné akce. Rozmezí jednotlivých uživatelských akcí je dle jednotlivých rolí. Samozřejmostí je, že **administrátor** bude mít větší rozsah pravomocí než běžný uživatel. Rozdělení jednotlivých rolí popisují následující podkapitoly. Kompletní zobrazení případů užití je na obrázku 6.2

6.2.1 Administrátor

Nejdůležitější rolí v hierarchii systému je zcela nepochybně **administrátor**. Po přihlášení může upravovat údaje o jednotlivých obyvatelech. Obyvatele hledá buď v tabulkovém přehledu nebo za využití funkce hledání na základě klíčů. Ze zaregistrovaných občanů může jako jediný zvolit nového starotu a jednotlivé vedoucí oddílů. Aplikace zabezpečuje, aby současně nebyli určeni dva a více starostů.

Jednou z dalších funkcí **administrátora** je vkládání anket. Díky anketám získají nejen představitelé úřadu přehled o veřejném mínění. Přehled mají i návštěvníci portálu ze strany občanů. Průběžné výsledky anket jsou zobrazovány v grafickém přehledu. Mínění mohou občané vyjadřovat také pomocí obecní diskuze. Jestliže nastane případ, že někdo zadá nevhodný příspěvek má **administrátor** možnost tuto skutečnost napravit. Příspěvek se tak nahradí hláškou: *Příspěvek porušil pravidla slušného vyjadřování*. Tímto opatřením se zamezí vkládání příspěvků porušující práva obyvatel a jiných nevhodných příspěvků.

Administrátor udržuje pořádek i v částech **Inzerce** a **podniky**. V obou těchto částech mohou zaregistrovaní občané realizovat své nejen podnikatelské záměry a vkládat reklamu. Předpokladem je, že bude někdo vkládat i nevhodné inzeráty nebo firmy. Tomuto nechtěnému jevu zamezí přihlášený **administrátor**.

6.2.2 Občan

Občané se v našem systému rozlišují na dvě velké skupiny. Buď je občanem zaregistrovaným a nebo nezaregistrovaným. Zaregistrovaný občan má mnohem více možností práce v systému. Naopak občan bez registrace může pouze vkládat příspěvky do jednotlivých diskuzí. Tyto příspěvky jsou označeny slovem **host**. Druhou aktivitou neregistrovaného občana je podávat náměty do **elektronické podatelny**. Poslední možností, kterou má i neregistrovaný občan je možnost hlasovat v anketách obce.

Při registraci a následném úspěšném přihlášení přibudou k již zmíněným funkcím ještě možnosti vložit inzeráty do sekce **Inzerce**. Takto zveřejněné inzeráty jsou přehledně označeny jménem inzerenta. Pokud jako funkcionalitu nepovažujeme prohlížení jednotlivých sekcí systému, tak poslední možností pro občana je vkládání firem. Samotné vkládání proběhne po vyplnění příslušných údajů do formuláře.

6.2.3 Zastupitel

Po zvolení občana do obecního zastupitelstva je tento občan přidán do této sekce. Zastupitelé mají v systému za úkol vkládat po každém zasedání jednotlivé zápisy. Takto zveřejněné zápisy si poté může kdokoli stáhnout do svého počítače a přečíst. Mezi další povinnosti zastupitelů patří vkládání elektronické podoby místních novin. Účelem této povinnosti je v co největší možné formě informovat širokou veřejnost o dění v obci. Posledním úkolem zastupitele v systému je vkládání textové formy hlášení obecního rozhlasu. Není totiž možnost zvolit čas, kdy by hlášení slyšeli všichni občané.

6.2.4 Starosta

Druhou nejdůležitější rolí v systému po administrátorovi je **starosta**. V systému může být pouze jeden. Ošetření pro změnu starosty tomu vždy zabrání. **Starosta** vkládá a maže fotografie v obecní fotogalerii. Jako jediný má také možnost vidět náměty od lidí v **elektronické podatelně**. Následně tak pomocí elektronické pošty může příslušné podněty řešit.

Z pohledu vykonání povinností starosty je zde možnost spravovat **úřední desku**. V ní je možnost buď přidat novou položku nebo již dříve zadané položky editovat. Mezi nejvýraznější možností editace je zneviditelnění příspěvku pro občany. Další úřednickou povinností starosty je určení a udržování zastupitelstva a jednotlivých odborů v obci. Aplikace je určena pro městy s cca 6000 obyvateli. S touto aglomerací musí mít městy 11 až 25 zastupitelů. Aplikace jinou možnost ani starostovi neumožní. Jednou z dalších neméně důležitých povinností starosty je vyvěšování **vyhlášky**. Zde jsou stejné možnosti editace jako u **úřední desky** či **výročních zpráv**. Aby byli občané informováni o aktuálních záležitostech co nejdříve vkládá starosta do **aktualit** krátké zprávy o aktuálním dění.

Starosta se v systému nezabývá pouze úřednickou činností. Informuje i občany o kulturním dění v obci. Veškeré vložené akce jsou následovně zobrazeny pro občany v přehledném **kalendáři akcí**. V případě změny data konání či jakékoliv jiné změny může **starosta** údaje o akcích libovolně měnit. Nastane-li situace, že je akce úplně zrušena má **starosta** možnost odstranit akci z kalendáře.

6.2.5 Vedoucí knihovny

Za jednotlivé sekce svých oddílů zodpovídají konkrétní vedoucí. **Vedoucí knihovny** spravuje elektronický katalog dostupných knih v místní knihovně. Mezi činnosti knihovníka tak spadá **vložení knihy** a **odstranění knihy** ze systému. Za použití těchto jednotlivých funkcí docílí knihovník stále aktuálního e-katalogu. Knihovna organizuje jednou za rok literární soutěž. Fotky z této soutěže vkládá **vedoucí knihovny** do systému.

6.2.6 Vedoucí kina

Dalším ze seznamu vedoucích je **vedoucí kina**. Po úspěšném přihlášení má tento vedoucí možnost spravovat nabídku s filmy, které nabízí místní kino. Nabízené filmy se po skončení vysílání automaticky již nezobrazují. Takto je docíleno aktuální nabídky. Při premiérách velkofilmů často zavítají tvůrci do našeho multikina. Fotky z těchto velkolepých akcí vkládá vedoucí do své části fotogalerie.

6.2.7 Vedoucí fotbalu

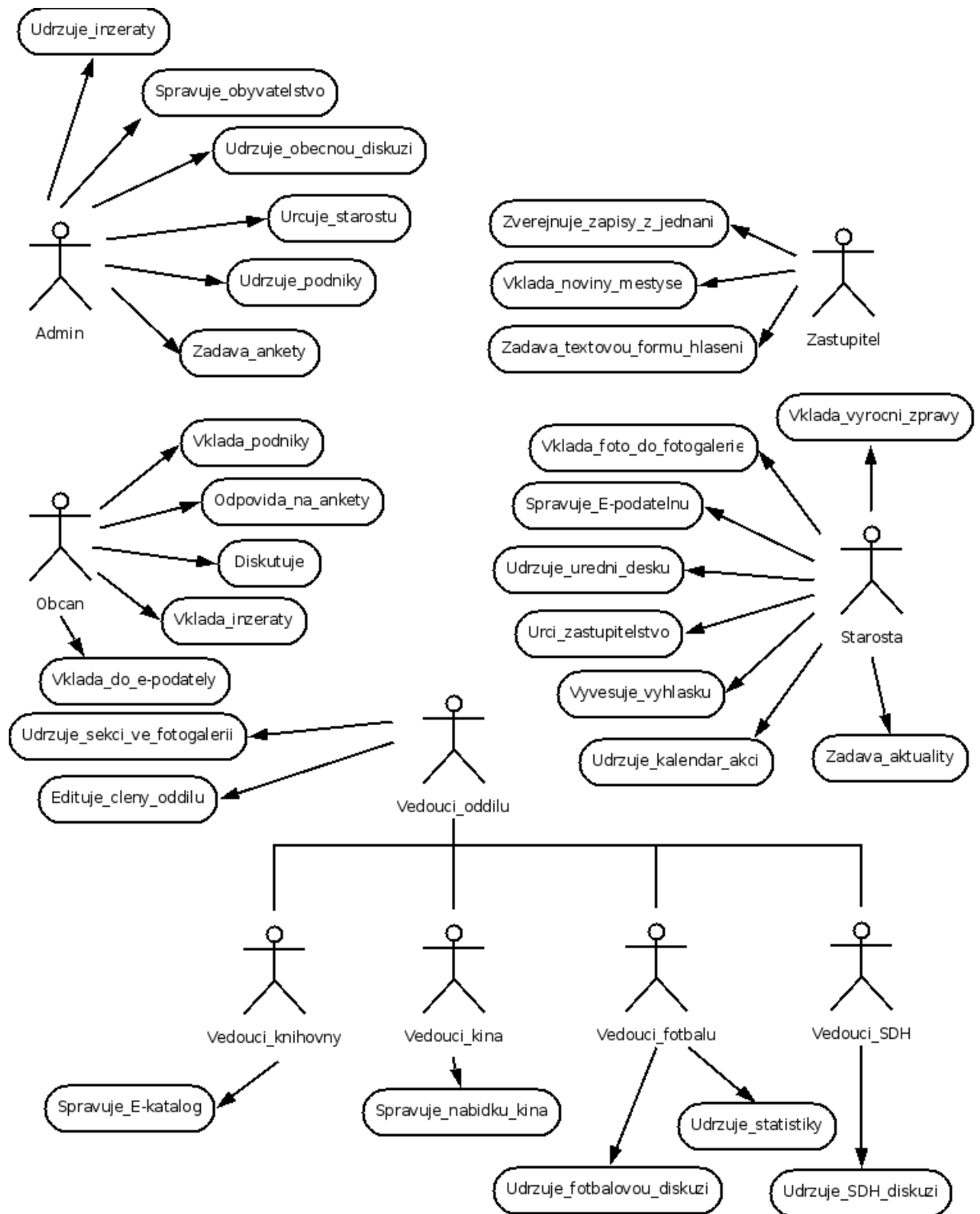
Místní fotbalový klub objíždí okolní turnaje. Proto má **vedoucí fotbalu** možnost vkládat fotografie ze zápasů do fotogalerie. Při jednotlivých vsítěných gólech edituje údaje hráčů. Po skončení sezóny má možnost vynulovat veškeré statistiky. Pro informování o zápasech a dění okolo klubu slouží fotbalová diskuze. Pokud nastane situace, že někdo poruší pravidla slušného diskutování má možnost **vedoucí fotbalu** takový příspěvek zrušit.

6.2.8 Vedoucí SDH

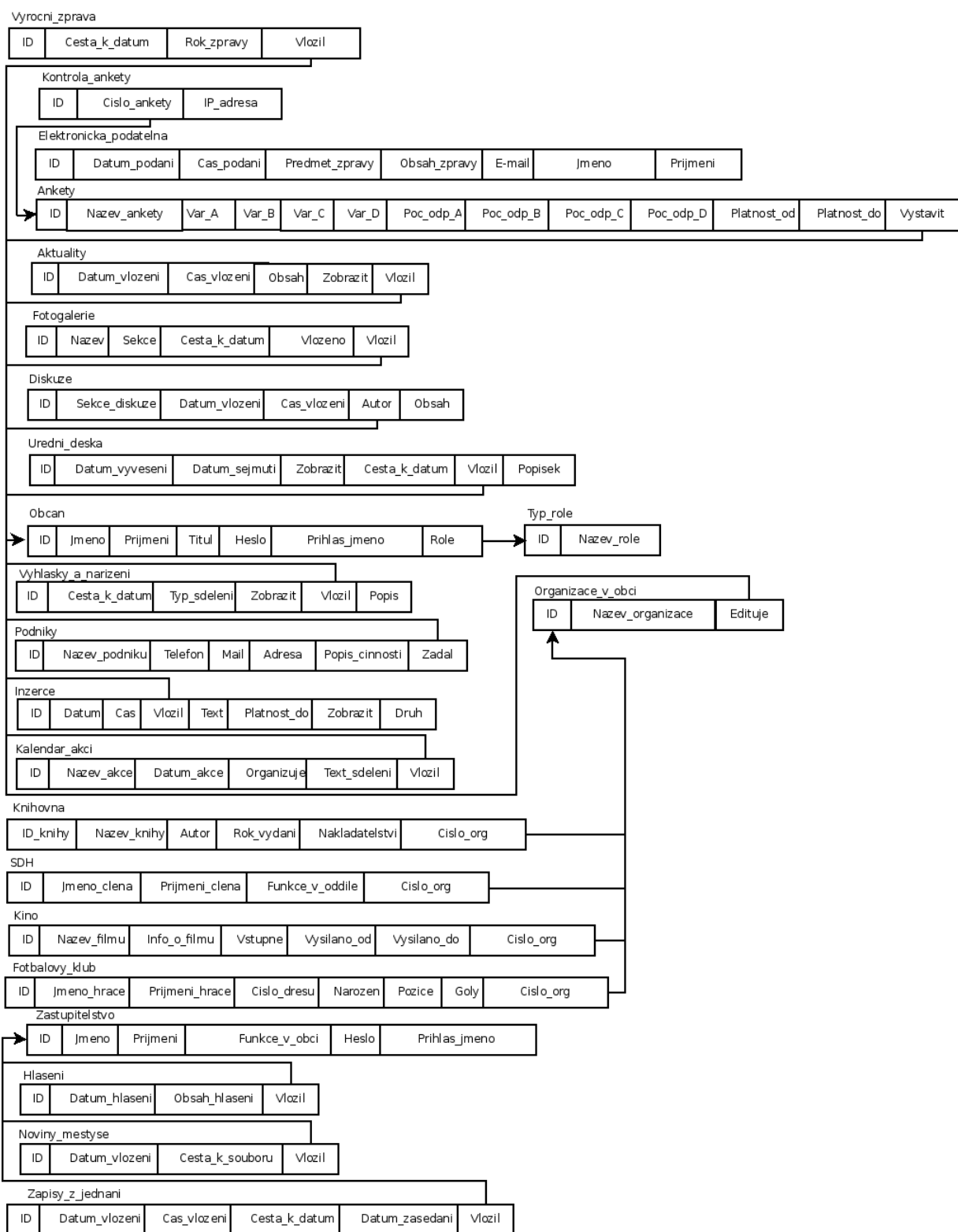
Místní sbor dobrovolných hasičů patří k nejméně aktivním organizacím v obci. Pro správu sekce SDH má oprávnění **vedoucí SDH**. Opět jako i jiní vedoucí vkládá fotografie do své sekce ve fotogalerii. Dále má na starosti správu hasičského klubu. Při vstoupení nového člena do organizace jej může přidat do systému. Naopak při odchodu členů je může odstraňovat ze systému.

6.3 Schéma databáze

Před samotným zahájením programování systému je nutno udělat ještě poslední krok v návrhu. Transformování ER diagramu na **schéma databáze**. Opět jsem zvolil cestu návrhu bez použití jakéhokoliv automatického nástroje, který by dělal práci za mne. Výsledkem návrhu je schéma na obrázku 6.3. Za použití tohoto schématu lze již implementovat celou databázovou strukturu.



Obrázek 6.2: Diagram případů užití



Obrázek 6.3: Schéma databáze

Kapitola 7

Implementace

Implementaci informačního portálu městyse bych rozdělil do několika dílčích částí. Převážná většina událostí se odehrává na straně serveru. Pro účely našeho programu je použit Apache server. Více informací je uvedeno v kapitole 7.1. Příkazy pro server jsou implementovány v jazyce PHP. Podrobnosti týkající se PHP jsou uvedeny v kapitole 7.2. Pro účely uchování dat a jednotlivých údajů je použita databáze MySQL (viz. 7.3). Na trhu existuje více databází, které by se daly použít. Jednalo by se například o PostgreSQL nebo o Oracle. Dynamické prvky systému jsou implementovány pomocí jazyka JavaScript (viz. 7.5). Aby bylo docíleno úspěšného zobrazení konvertuje prohlížeč kód PHP na kód HTML (viz. 7.6).

Pro dosažení maximální modularity a zároveň přehlednosti kódu je většina skriptů složena z dílčích částí. Tyto části jsou následně vkládány do všech skriptů. Mezi dílčí části patří skript pro vložení HTML hlavičky, položek levého menu, a přihlašovacího formuláře. Díky tomuto postupu se omezil rozsah dalších skriptů. Docílil jsem tak oddělení části obsahové od části designové.

Levý sloupec tvoří hlavní nabídku celého systému. Jsou zde uloženy hlavní kategorie jako například **Aktuality**, **Úřad**, **Obec** nebo **Společenské dění**. Pro účely vyšší přehlednosti jsou podkapitoly těchto sekcí schovány a zobrazí se po kliknutí na danou sekci. Tímto jsem zamezil příliš velkému a tudíž i nepřehlednému menu, které by zcela jistě odradilo nejednoho uživatele. Stav zobrazení kategorií je uchován v Cookies prohlížeče. Docílíme tak stavu, kdy zůstane pro uživatele uchován poslední stav zobrazení.

Celý systém je navržen ve třech jazykových mutacích. Prvotní načtení jazyka je určeno dle nastavení prohlížeče. Jazyky lze libovolně střídat kliknutím na příslušnou vložku symbolizující konkrétní jazyk. Samotný překlad je proveden do úrovně všech nadpisů první kategorie, chybových a informačních hlášek. Dále pak v převážné míře jsou přeloženy i nadpisy druhé kategorie. Informace o konkrétním používání jazyce je uchována v uživatelské *session*. Na základě tohoto sezení se použijí makra z dílčích jazykových skriptů (*cestina.php*, *deutsch.php* a *english.php*).

Uživatel má možnost se přihlásit do systému pomocí formuláře, který je umístěn v pravém sloupci aplikace. Při přihlášení musí určit jestli se jedná o zastupitele či nikoliv. Pro rozlišení zastupitelů je ve formuláři umístěn **checkbox**. Informace o přihlášeném uživateli jsou uloženy v *session*. Na základě ID uživatele a jeho role jsou umožněny přihlášenému jednotlivé funkce. Popis dílčích funkcí je uveden v kapitole 6.2.

Pokud se chce uživatel odhlásit ze systému, klikne na tlačítko **odhlášení**, umístěné na pozici přihlašovacího formuláře. Tímto se zničí současné sezení a informace o přihlášeném. Pro zajištění vyšší bezpečnosti je v systému i automatické odhlašování. Tato funkcio-

nalita funguje tak, že pokud nezjistí po dobu 15 minut od poslední činnosti žádnou aktivitu, automaticky odhlásí aktuálně přihlášeného. K rozlišení těchto intervalů se využívá Linuxového času, který porovnávám ve speciální funkci.

V systému se hodně pracuje s datem a časem. Linuxový čas má stejný formát jako typ `TIME` v `MySQL`. Proto nebylo třeba jej jakkoliv upravovat. Horší situace nastala v práci s datumy. Aplikace jako vstup od uživatele bere datum ve formátu `DD.MM.YYYY`. Do databáze se ukládá již jako `YYYY-MM-DD`. Pro konverze mezi těmito formáty času jsou implementovány funkce umístěné ve skriptu `overeni_data.php`. Bylo potřeba řešit i další funkce jako je porovnání dvou datumů nebo zjištění korektnosti zadaného data.

7.1 Apache

Modul `Apache` působí jako náš webový server. Jeho hlavní úlohou je nejen zpracování požadavků, které uživatelé odesílají prostřednictvím svých webových prohlížečů, ale také zobrazení výsledků připravených pomocí kódu umístěného ve vyžádaných souborech. `Apache` je velmi výkonný stroj a může splnit prakticky všechna přání, jaká může správce webu mít [5].

7.1.1 Historie Apache

Podnětem pro vytvoření serveru `Apache` byl odchod vývojáře Roba McCoola z National center for Supercomputing Applications, která vyvinula úspěšný webový server `NCSA`[8]. V únoru 1995 vznikla `Apache Group`, skupina vývojářů, která odstartovala vývoj nového serveru jako volně šiřitelné modifikace původního `NCSA`. K němu byly postupně přidány záplaty (`patches`), které zvyšovaly funkčnost a stabilitu serveru a daly tak i podnět k pojmenování nového serveru na `Apache` (z anglického `a patche - záplata`). První verze `Apache` vznikla 18. března 1995 a nesla označení `Apache 0.2`.

Vývoj `Apache` pokračoval i v následujících letech a přinesl mnohá vylepšení. Za zmínku stojí jistě podpora pro virtuální servery od verze `Apache 1.0`. Dále vylepšení vlastností v souboru `.htaccess` ve verzi `Apache 1.1`. V neposlední řadě podpora `HTTP/1.1` od verze `Apache 1.2`. Pro operační systém `Windows` vznikla podpora od verze `Apache 1.3`. Posledním výrazným pokrokem byla verze `Apache 2.0`, která obsahuje nástroje pro správu, i vzdálenou.

7.2 PHP

Převážná většina zdrojových kódů využívá jazyka `PHP`. Při vývoji systému jsem se rozhodl pro `PHP4`. Tímto krokem jsem chtěl docílit, aby nevznikl žádný problém s webhostingy, které novější `PHP5` nepodporují. Jedním z rozdílů verzí čtyři a pět je rozdílný způsob zápisu konstruktorů. Jak tedy z předchozích vět vyplývá, program je napsán funkcionálně. Skripty `PHP` se spouštějí výhradně na straně serveru.

7.2.1 Historie PHP

Počátky skriptovacího jazyka `PHP` se datují do roku 1994 [7]. V tomto roce jej začal vyvíjet pan `Rasmus Lerdorf`. Prvotní myšlenkou bylo vytvořit jednoduchý systém, který by počítal přístupy na své webové stránky. Původní `PHP` bylo napsáno v jazyce `PERL`. Brzy však byl systém přepsán do jazyka `C`, který méně zatěžuje server. Jednalo se o skupinu skriptů,

kteřá byla nazvána **Personal Home Page Tools**, zkráceně **PHP**. Zájem o tento nově vzniklý systém prudce stoupá. Proto autor doplňuje dokumentaci a elektronickou konferenci. Krátce na to autor začíná pracovat na druhé verzi systému. Ta je spojena s jeho dalším programem **Form interpreter**, zkráceně **FI**.

V následujících letech pracuje autor zcela sám na druhé verzi **PHP** označené jako **PHP/FI 2.0**. Druhá verze je na tolik oblíbená, že v roce 1997 ji má nainstalovanou přes 50.000 domén. Následně začne na vývoji spolupracovat více autorů, ale stále se projekt dá považovat za dílo jednoho muže.

Projektem **PHP** se začínají zabývat i další vývojáři- **Andi Gutmans** a **Zeev Suraski**, kteří celé **PHP** přepisují a více přizpůsobují pro e-komerci. Projekt **PHP/FI 2.0** je pozastaven a systém **PHP 3.0** je zrychlen a doplněn o další funkce. K projektu **PHP 3.0** se připojují desítky programátorů a vyvíjejí další rozšiřující moduly. Za zmínku jistě stojí, že od **PHP 3.0** je systém dostupný i pro **Windows**. Samotná zkratka je nyní překládána jako **Hypertext Preprocessor**. Na konci roku 1998 využívá **PHP** přibližně 10% všech **WWW** serverů.

Po oficiálním uvolnění **PHP 3.0** v červnu 1998 začínají programátoři pracovat na přepsání jádra. Vznikla tak podpora práce s databázemi. Výsledným produktem bylo zbrusu nové jádro **Zend**. Název jádra je složen z písmen křestních jmen autorů. Nové jádro **PHP 4.0** je uvolněno v roce 1999 a je desetinásobně rychlejší než předchozí **PHP 3.0**. Největšími přínosy nové verze byla podpora **HTTP sessions**, **bufferingu** nebo bezpečnějšího zpracování vstupů od uživatele.

V dnešní době využívá **PHP** přibližně 20% veškerých webových serverů. Pracuje se na vývoji systému **Zend II** a aplikaci prvků navržených pro **PHP 5.0**. Patří mezi ně například podpora **XML**, která bylo ze samotného **PHP** odsunuta do knihovny **libxml2**. Dále pak za zmínku stojí i podpora **SQLite**.

7.3 MySQL

K implementaci informačního systému jsem zvolil databázový systém **MySQL**. Ten umožňuje technologiím **PHP** a **Apache** spolupracovat na zpřístupnění a zobrazení dat ve formátu čitelném v internetových prohlížečích. **MySQL**, spolu s **PHP** a **Apache** tvoří tzv. **XAMPP**. Součástí modulu **XAMPP** je i nástroj na správu databází **PHPmyAdmin** [3]. Jedná se o nástroj sloužící ke snadnější správě tabulek **MySQL**.

MySQL se dělí na dva formáty uchování dat. Formáty se nazývají **MyISAM** a **InnoDB**. Nejpoužívanější je první zmiňovaný, který tvoří nástupce systému **ISAM**. Tento formát je použit i pro účely našeho informačního systému. Nejmarkantnějším rozdílem je, že formát **InnoDB** má zabudovanou podporu transakcí.

7.4 CSS

Pro účely samotného vzhledu je použit stylopis **CSS**. Název **CSS** je zkratkou pro **Cascading Style Sheets**. Vzhled systému využívá blokové elementy **DIV** a řádkové elementy **SPAN** ve zdrojovém textu **PHP/HTML**. Nejen díky těmto elementům mohl vzniknout samotný vzhled systému. Mnou navržené **CSS** využívá grafické prvky zpracované programem **Adobe Photoshop**. Využil jsem návrhy popsané v knize [2].

7.4.1 Historie CSS

Postupný vývoj HTML se začátkem 90. let začal ubírat špatným směrem. Již se vytrácela původní struktura dokumentu a vývojáři se soustředili více na vzhled. Postupem času začalo vývojářů a tvůrcům prohlížečů docházet, že tudy cesta nevede. Proto v roce 1996 vznikla první verze CSS [1]. Za otce dnešního CSS je považován Híkon Wium Lie, šéfvývojář Opery. Stylopis CSS podporoval Internet Explorer od verze 3.0. Jediný prohlížeč, skeptický k tomuto vývoji byl dlouho Netscape. Od verze Netscape 4.0 již CSS podporuje.

7.5 JavaScript

Jak již bylo zmíněno dříve pro zvýšení přehlednosti stránek jsem použil technologii JavaScript. Konkrétně na schovávání podsekcí menu umístěného v levém sloupci. Dalším prvek, na který jsem nasadil JavaScript je běžící aktuální čas. JavaScript běží na rozdíl od PHP na straně klienta. Tato skutečnost umožňuje uživateli stránek využít větší funkcionalitu a dynamičnost bez nutnosti poslání požadavku na server.

7.6 HTML

Samotné HTML využívám v celém projektu jen zřídka. Pomocí jazyka HTML je popsána pouze hlavička dokumentu s veškerými metadaty. Zbytek aplikace obstarává jazyk PHP. I přes tuto skutečnost je nutno se o jazyce HTML zmínit. Všechny mnou napsané skripty se totiž ve výsledku v prohlížeči přetransformují do podoby HTML, aby je mohly prohlížeče bez problémů zobrazit.

7.6.1 Historie HTML

První definici jazyka HTML vytvořil v roce 1991 Tim Bernes–Lee jako součást projektu WWW, který měl umožnit vědcům zabývajícím se fyzikou vysokých energií komunikaci a sdílení výsledků výzkumu po celém světě [6]. Ne náhodou proto celý projekt vznikl v CERNu. Tato verze HTML je známa pod označením HTML 0.9. Umožnila rozčlenit text do několika logických úrovní, použít několik druhů zvýraznění textu a zařadit do textu odkazy a obrázky.

Požadavky uživatelů na WWW vzrůstaly, a proto producenti různých prohlížečů obohacovali HTML o některé nové prvky. Aby tak byla zachována kompatibilita mezi různými verzemi, zavedl Bernes–Lee pod hlavičkou IETF (Internet Engineering Task Force) návrh standardu HTML 2.0. HTML 2.0 má dvě úrovně. První z nich rozšiřuje předchozí verzi HTML. Druhá úroveň navíc definuje práci s formuláři. Specifikaci HTML 2.0 nalezneme v RFC dokumentu číslo 1866.

Další rozšíření známé jako HTML+ zahrnují rozšířený o vytváření tabulek, matematických vzorců a lepší obtékání obrázků. Dave Raggett z laboratoří Hewlett–Packard HTML+ formalizoval a vytvořil jeho deklaraci DTD (Document Type Declaration) v jazyce SGML. Na jaře 1995 tak vznikl návrh standardu HTML 3.0. Na jaře 1996 již bylo jasné, že HTML 3.0 byl tak mohutným skokem vpřed, že se nenašel nikdo, kdo by implementoval prohlížeč s jeho podporou.

Členové konzorcia W3C se tedy shodli na vlastostech, o které rozšíří HTML 2.0, a vytvořili tak HTML 3.2. To však nezahrnuje vše z verze HTML 3.0. Dalším podstatným rozšířením bylo přidání podpory Java – appletů. Tato verze nese název Wilbur.

Na začátku července 1997 zveřejnilo konsorcium W3C návrh HTML 4.0, který vznikl úpravami předchozích verzí a vytvořením jednoho komplexního dokumentu popisujícího návrh standardu.

7.6.2 Protokol HTTP

Jazyk HTML, který je určen pro aplikaci WWW využívá aplikační protokol HTTP. Transportním protokolem je spolehlivý TCP [11]. Protokol HTTP pracuje na principu dotaz – odpověď. Jakákoli aktivita musí být vyvolána klientem. Komunikace se serverem probíhá ve většině případů přes port 80.

Součástí úplného dotazu/odpovědi musí být specifikovaná metoda, URI, verze a hlavičky. Server, podporující protokol 1.0 vrátí odpověď a spojení uzavře. Protokol 1.1 definuje tzv. **perzistentní spojení** a proto servery podporující verzi 1.1 spojení neuzavřou hned, ale čekají chvíli na další příkazy. Klient může pokračovat v dotazování na ostatní prvky HTML stránky a spojení sám ukončit.

Odpověď obsahuje verzi protokolu, kód odpovědi, textovou hlášku odpovědi, hlavičky, prázdný řádek jako oddělovač a většinou i tělo odpovědi. Kódy odpovědi se dělí do následujících skupin:

- **1xx** – informativní odpověď
- **2xx** – úspěch (např. nejčastější 200)
- **3xx** – přesměrování
- **4xx** – chyba na straně klienta (nejznámější 404)
- **5xx** – chyba na straně serveru

Jelikož je verze 1.1 perzistentní, musí být v hlavičce uvedena přesná délka těla odpovědi nebo alespoň typ kódování, aby klient věděl, kdy odpověď končí. Mezi nejznámější metody dotazu patří:

- **GET** – nejpoužívanější metoda. Slouží k vyzvednutí objektu ze serveru.
- **POST** – v těle této metody se dají na server přenést informace o uživateli. Používá se pro citlivější data.
- **HEAD** – chová se jako GET, ale nepřenáší se tělo.
- **PUT/DELETE** – vytvoří/smaže objekt na serveru
- **OPTIONS** – slouží ke zjištění o daném kontextu.
- **TRACE** – používá se ke sledování cesty celého dotazu
- **CONNECT** – slouží k tunelování HTTP protokolu

Kapitola 8

Testování

Testování je jednou z nejdůležitějších etap vývoje programu. Zabere nemalé množství času, které se počítá do vývoje. Z tohoto poznatku také plyne, že prodražuje výsledný produkt. Testování bych rozdělil do dvou dílčích částí. První částí je testování během samotného programování. Druhou částí je testování již hotového systému. Tuto část je nejlépe nechat na lidech, kteří by mohli s aplikací přijít do styku.

8.1 Testování při vývoji

Pro účely testování při vývoji poslouží tabulka 8.1. V té jsou uvedeny přihlašovací jména a hesla do aplikace. Aplikace využívá pro svůj provoz školní server. Naleznete ji na adrese <http://www.stud.fit.vutbr.cz/~xbilek01/bakalarka/>.

Přihlašovací jméno	Heslo	Role
admin	admin11	administrátor
Roomy	student	starosta
rada	rada11	vedoucí fotbalu
hasilsky	hasilsky11	vedoucí SDH
connery	connery11	vedoucí kina
knihovnik	knihovnik11	vedoucí knihovny
straka	straka11	zastupitel

Tabulka 8.1: Role v informačním systému

Při testování systému jsem se svým původním návrhem narazil na problém. Nevhodným zasíláním hlaviček přestala aplikace smysluplně pracovat. Tato skutečnost nastala následkem špatného návrhu. Abych docílil výsledné funkčnosti, musel jsem celou aplikaci znovu naprogramovat. Již nevyužívám pouze `index.php`, kde jsem původně přepínal mezi jednotlivými skripty. Nyní je využito celé spektrum jednotlivých podsriptů. Program tak získal na přehlednosti.

8.2 Testování uživatelí

Po docílení výsledné podoby programu jsem využil okruh přátel pro testování aplikace. Jednotlivce jsem volil dle dosaženého vzdělání a současné profese. Tímto jsem se snažil docílit širšího spektra pohledu. Již od počátku bylo jasné, že rozdílný pohled na různé části mají ženy a jiný pohled zase muži.

8.2.1 Testování designu

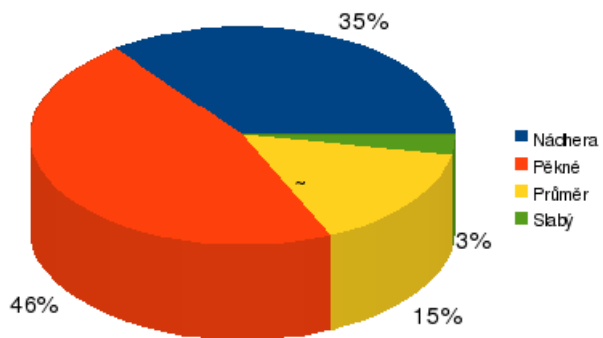
První verze systému měla zcela odlišný design než tato finální verze. Náhled první verze je na obrázku 8.1. Náhled finální verze poté na obrázku 8.2. K této razantní změně vzhledu mne dovedly názory jednotlivých respondentů. Většina se shodla, že prvotní verze nepůsobí jako portál obce. Výsledný pohled na systém z hlediska designu je znázorněn v grafu na obrázku 8.3.



Obrázek 8.1: Původní vzhled



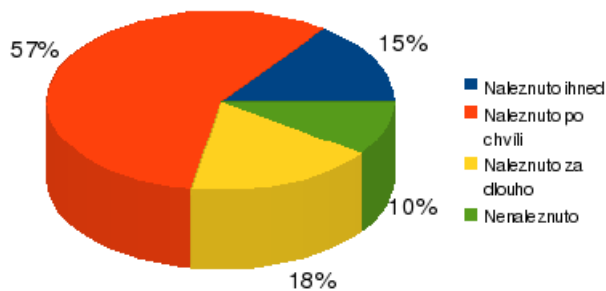
Obrázek 8.2: Výsledný vzhled



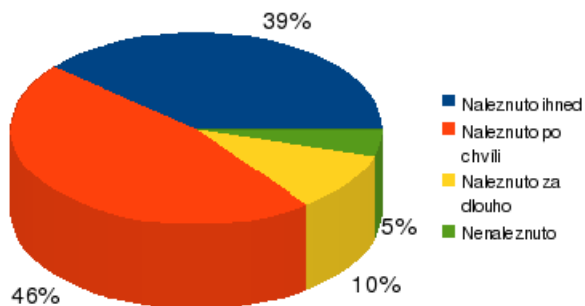
Obrázek 8.3: Oblíbenost výsledného vzhledu

8.2.2 Testování přehlednosti

Aby byl systém co nejvíce intuitivní na ovládání, bylo třeba provést další testování. Tentokrát jsem testovací osoby rozdělil do dvou skupin. V první skupině jsou lidé, kteří příliš na počítači nepracují. Ve druhé naopak lidé, kteří mají bohaté zkušenosti s obsluhou PC. Výsledné hodnoty znázorněné v grafu na obrázku 8.4 patří ke skupině lidí méně počítačově gramotných. Naopak na obrázku 8.5 jsou lidé, kteří na počítači pracují téměř denně.



Obrázek 8.4: Hledání méně zkušených



Obrázek 8.5: Hledání více zkušených

Výsledné hodnoty jasně napovídají, že lidé, kteří pracují denně na počítači se v našem systému orientují rychleji. Napomáhá jim k tomu nabytá zkušenost. Naopak u lidí, kteří nepřijdou tak často do styku s počítačem byly výsledky o něco horší. Po zvážení těchto hodnot jsem usoudil, že struktura programu je pro většinu lidí přijatelná.

Kapitola 9

Možná rozšíření do budoucna

System v současném stavu vyhovuje specifikaci a hlavně zadání. Nastolení tohoto stavu byl počáteční cíl. System proto považuji za úplný a funkční.

Pro možná rozšíření v budoucnu by bylo nutno nejprve přepracovat návrh a poté samotnou implementaci. Záleží na obsáhlosti rozšíření. Pro nasazení do praktického provozu by bylo nutno zabezpečit aplikaci na mnoha úrovních. Pokud by obsahovala data, která by se neměla dostat do nepravých rukou, tak by právě tato oblast byla první v rozvoji. Současná aplikace obsahuje data, která jsou následně nabízena uživateli. I přes tuto skutečnost je dbáno na zabezpečení. Další oblastí je uchování hesel uživatelů. Současná aplikace využívá hashe MD5, který by mohl být nahrazen například pomocí SHA1.

Rozšíření programu by nemuselo probíhat pouze v oblasti funkčnosti. V oblasti technologií bych do budoucna celou aplikaci přepracoval do objektového modelu. Již se bude předpokládat, že webhostingy tuto technologii zvládají. Dále by bylo vhodné využití například technologie **Flash**.

Program by šel přepracovat a rozšířit i na grafickém poli. Záleželo by pokaždé na požadavcích klienta a oficiálních barvách obce.

Kapitola 10

Závěr

Při řešení implementace **informačního portálu městyse** jsem se setkal s celou etapou vývoje software. Od počátečního návrhu, který směřoval z neformální specifikace až k té formální. Při tomto procesu jsem si prohloubil znalosti v oblasti modelování UML. Prvním formálním modelem je **ER diagram**. Ten modeluje jak budou data vypadat v klidu. Dalším modelem je **use case diagram**. Zde jsou naopak znázorněny veškeré akce, které budou jednotliví aktéři v systému vykonávat. Abych mohl naprogramovat samotnou databázi bylo nutno v dalším kroku provést transformaci **ER diagramu** na **schéma relační databáze**. Touto činností skončila etapa návrhu.

Následovala etapa samotného programování. Zde se ve finální fázi vývoje prokázala chyba v návrhu. Proto bylo nutné celou aplikaci přeprogramovat. Tato nemilá zkušenost mne utvrdila v důležitosti správného návrhu. Bez něj totiž vznikají podobné komplikace. Mezi další komplikace bych přiřadil například problém s kódováním na serveru `eva.fit.vutbr.cz`. Server mi totiž neustále přepisoval kódování uvedené v metadatech hlavičky **HTML dokumentu**. Problém jsem vyřešil nakopírováním souboru `.htaccess` na server. V tomto souboru jsem nadefinoval požadované kódování a znakovou sadu.

Na závěr bych tedy shrnul získané poznatky. Při vývoji aplikace jsem si prohloubil znalosti modelování systému. Dále pak při implementaci jsem se setkal s věcmi jako je například tzv. **captcha** (generovaný kód do obrázku pro účely zabezpečení proti robotům). V neposlední řadě také práci s adresáři na straně serveru. Veškerá data jsou ukládána pro urychlení aplikace do těchto adresářů.

Literatura

- [1] Cyroň, M.: *CSS kaskádové styly*. Grada, 2006, iISBN 80-247-1420-5.
- [2] Davis, L. D. . J.: *Photoshop WOW! Book*. Computer press, 2003, iISBN 80-7226-719-1.
- [3] DeLisle, M.: *phpMyAdmin efektivní správa MySQL*. Zoner press, 2004, iISBN 80-86815-09-9.
- [4] Jaroslav Zendulka, I. R.: *Databázové systémy - Studijní opora*. 2006.
- [5] Kolektiv autorů: *PHP5, MySQL, Apache - Vytváříme webové aplikace*. Computer press, 2006, iISBN 80-251-1073-7.
- [6] Kosek, J.: *HTML - tvorba dokonalých WWW stránek*. Grada, 1999, iISBN 80-7169-608-0.
- [7] Pokorný, M.: *PHP nejen pro začátečníky*. Computer media, 2005, iISBN 80-86686-38-8.
- [8] Pošmura, V.: *Apache - Příručka správce WWW serveru*. Computer press, 2002, iISBN 80-7226-696-9.
- [9] Tomáš Hruška, Z. K.: *Informační systémy - Pojem informačního systému, Data, Procesy, Transakce, Studijní opora*. 2007.
- [10] WWW: Městys. [online], [cit. 2. 5. 2009].
URL <http://cs.wikipedia.org/wiki/Městys>
- [11] WWW: Protokol HTTP pod lupou. [online], [cit. 29. 4. 2009].
URL <http://www.root.cz/clanky/protokol-http-1-1-pod-lupou>
- [12] WWW: Účetnictví, daně, právo - Sagit. [online], [cit. 29. 4. 2009].
URL <http://www.sagit.cz>

Dodatek A

Seznam příloh

- **Příloha 1** – DVD-ROM se zdrojovými kódy aplikace, včetně elektronické dokumentace a návodu na použití