

**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**  
BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA INFORMAČNÍCH TECHNOLOGIÍ**  
**ÚSTAV INFORMAČNÍCH SYSTÉMŮ**

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF INFORMATION SYSTEMS

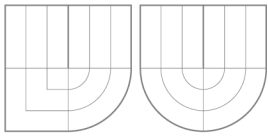
**INFORMAČNÍ SYSTÉM PRO SPRÁVU DOKUMENTŮ**  
**A FOTOGRAFIÍ**

**BAKALÁŘSKÁ PRÁCE**  
BACHELOR'S THESIS

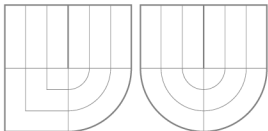
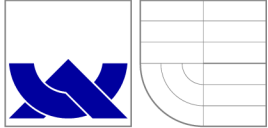
**AUTOR PRÁCE**  
AUTHOR

**LADISLAV VARGA**

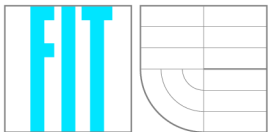
BRNO 2007



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 PRO SPRÁVU DOKUMENTŮ A FOTOGRAFIÍ

DOCUMENT AND PHOTOGRAPHY INFORMATION SYSTEM

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

LADISLAV VARGA

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. PETR CHMELAŘ

BRNO 2007

# Informační systém pro správu dokumentů a fotografií

## Document and Photography Information System

### Vedoucí:

[Chmelař Petr, Ing.](#), UIFS FIT VUT

### Oponent:

[Straka Martin, Ing.](#), UPSY FIT VUT

### Přihlášen:

Varga Ladislav

### Zadání:

1. Seznamte se s možnostmi ukládání a vyhledávání dokumentů v různých systémech. Definujte požadavky na informační systém, který bude umožňovat spravovat dokumenty a fotografie prostřednictvím databáze i souborového systému.
2. Analyzujte požadavky na tento systém. Pro analýzu využijte některý modelovací jazyk (např. UML).
3. Seznamte se s prostředím databázového systému MySQL a skriptovacími jazyky PHP a JavaScript.
4. Navrhněte informační systém pro vkládání a vyhledávání dokumentů s možností autentizace a autorizace uživatelů a implementujte systém jako modul redakčního systému.
5. Zhodnoťte dosažené výsledky a diskutujte možnosti dalšího pokračování projektu.

### Kategorie:

Databáze

### Literatura:

- KANISOVÁ, H., MÜLLER, M. *UML srozumitelně*. Brno, Computer Press, 2004.
- WELLING, L., THOMSONOVÁ, L. *PHP a MySQL - rozvoj webových aplikací*. Praha: SoftPress, 2003. 910 s.
- ŠKULTÉTY, R. *JavaScript, programujeme internetové aplikace*. Brno, Computer Press, 2004. 224 s.
- CHMELAR, P. *Multimediální databáze*. 2006.

## **Licenční smlouva**

Licenční smlouva je uložena v archívu Fakulty informačních technologií Vysokého učení technického v Brně.

## Abstrakt

Cílem bakalářské práce je navrhnout a implementovat informační systém pro správu dokumentů a fotografií. Pro implementaci byly použity následující technologie: PHP, MySQL, Javascript, HTML, CSS, Ajax.

## Klíčová slova

fotogalerie, dokument, sklad, PHP, MySQL, informační systém, Ajax

## Abstract

The Goal of this work is to project and implement document and photography information system. Information system was implemented by PHP, MySQL, Javascript, HTML, CSS, Ajax.

## Keywords

photogallery, document, store, PHP, MySQL, information system, Ajax

## Citace

Ladislav Varga: Informační systém pro správu dokumentů a fotografií, bakalářská práce, Brno, FIT VUT v Brně, 2007

# Informační systém pro správu dokumentů a fotografií

## Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Petra Chmelaře. Uvedl jsem všechny prameny ze kterých jsem čerpal.

.....  
Ladislav Varga  
15. května 2007

© Ladislav Varga, 2007.

*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

<b>1 Úvod</b>	<b>3</b>
<b>2 Analýza</b>	<b>4</b>
2.1 Analýza požadavků	4
2.2 Informační systém	5
2.3 Databáze	5
2.4 Dokument	6
2.5 Fotografie	8
2.6 Ukládání dokumentů	8
2.7 Použitelnost	8
2.8 Přístupnost webu	9
2.8.1 Pravidla přístupného webu	9
2.9 Optimalizace	11
2.9.1 Optimalizace pro vyhledávače (SEO)	12
<b>3 Návrh</b>	<b>13</b>
3.1 UML	13
3.2 Model systému - ER diagram	14
3.3 Tabulky databáze	16
3.4 Diagram případů použití	17
3.5 Návrh vzhledu	19
<b>4 Implementace</b>	<b>21</b>
4.1 Prostředí implementace	22
4.1.1 PHP	22
4.1.2 MySQL	23
4.1.3 HTML	24
4.1.4 CSS	25
4.1.5 JavaScript	25
4.1.6 Ajax	25
4.2 Thickbox	26
4.3 Vlastní implementace	26
4.4 Testování	26
4.5 Další vývoj	27
<b>5 Závěr</b>	<b>28</b>
<b>A Uživatelská příručka</b>	<b>32</b>





# Kapitola 1

## Úvod

S různými fotogaleriemi a sklady dokumentů se setkáváme po celý svůj život. Sklad dokumentů či fotogalerie nemusí být jen elektronický ale i prostěpapírová. My se budeme zabývat jen elektronickou/digitální formou. Na trhu existuje mnoho druhů fotogalerií. Jsou vytvořeny v různých programovacích jazycích. Pro ukládání se většinou používá nějaký databázový systém jako např. MySQL či MSSQL. Lze samozřejmě ukládat všechny údaje o fotkách do jednoho textového souboru, ale díky pracnosti tohoto řešení se od něho upouští. Pro publikování informačního systému se využívá nejčastěji PHP, ASP, Delphi, .net, C++ a mnoho jiných.

Nejčastěji existují v samostatné formě tzn. Jen informační systém pro ukládání fotografií spolu s publikací nejčastěji na WWW (World Wide Webu) v celosvětové síti internet. Mezi zástupce této kategorie patří např. Coppermine, Photogallery od Majordomo, Gallery. Většina těchto produktů je open source (pro volné využití) a k jejich zprovoznění stačí minimum znalostí.

Dále existují i zprostředkované fotogalerie, kdy potenciální uživatel nic neinstaluje a využívá již vytvořeného systému rovnou na webu. Český zástupce systému je např. <http://rajce.idnes.cz/>.

Další variantou jsou různé moduly speciálně vytvořené pro konkrétní informační nebo redakční systém. Mým úkolem bude vytvořit jeden takový informační systém pro ukládání nejenom fotek, ale i jiných dokumentů. Konkrétně vytvořím modul informačního systému.

Motivace pro vytvoření informačního systému pro správu fotografií a dokumentů je zpřístupnění fotek a dokumentů široké veřejnosti. Můj informační systém bude umožňovat jednoduchou a přehlednou správu všech dokumentů a fotek. Samozřejmě se inspiroji už udělanými systémy, které prošly dlouhým vývojem a hlavně nasazením na webu. Samozřejmostí bude ukládání nejen fotek a obrázků ale i ostatních dokumentů.

Jak jsme si již řekli informační systém bude vytvořen pro web tzn. bude to webový informační systém (WIS). Co je webový informační systém. Je to speciální informační systém uzpůsoben fungování na WWW (World Wide Web).

Dostáváme se k tomu co bude úkolem této práce. Nejprve si definuji požadavky na informační systém pro správu dokumentů a fotografií. Co od systému můžu očekávat, co od něj očekávají uživatelé. Analyzuji všechny důležité části webového informačního systému s důrazem na kladené požadavky. Podrobněji rozeberu pojmy jako je databáze, informační systém, dokument a další bez jejich znalosti se neobejdeme. Další kapitolou bude návrh celé aplikace s důrazem na návrh databáze přiblížení rolí všech uživatelů. Implementace bude zabírat nemalou část technické zprávy. Seznámíme se s mnoha prostředky, které pro implementaci využijeme a řekneme se něco o testování a optimalizaci systému. Budoucí rozšíření systému lze také zařadit do implementační části práce. Na závěr zhodnotím dosažené výsledky a můj přínos práce.

# Kapitola 2

## Analýza

Cílem práce je vytvoření informačního systému pro správu fotografií a dokumentů. Kapitola nás seznámí obecně s informačními systémy a s jejich používáním. Dále si řekneme něco o rozdělení dokumentů. Pro ukládání informací o dokumentech bude použita databáze. Informační systém bude nasazen na webu tzn. musí být přístupný a použitelný v majoritních prohlížečích a operačních systémech. Bez znalostí všech důležitých pojmů nelze pokračovat v návrhu a implementaci.

### 2.1 Analýza požadavků

Každý uživatel, který vstoupí na náš informační systém očekává různé služby. Návštěvníci si nejsou rovni. Budeme rozlišovat mezi obyčejným návštěvníkem, členem, editorem, redaktorem a administrátorem. Všichni patří minimálně do jedné z těchto skupin. Díky tomuto rozdělení lze získat požadavky a podle toho přizpůsobit svou práci a snažit se splnit všechny požadavky. Řekneme si i něco o získávání požadavků od lidí.

**Návštěvník** – Očekává 24-hodinový rychlý a přehledný přístup k informacím. Dále by chtěl komentovat články. S ohledem na fotogalerii by si prohlížel fotky ve všech formátech. Měl by mít možnost stáhnout si fotografie a dokumenty k sobě. Chtěl by také hodnotit fotografie a články.

**Člen** – Člen bude vlastnit uživatelský účet tzn. že se bude moci přihlašovat do informačního systému. Jediná výhoda oproti návštěvníkovi bude možnost číst i neveřejné články.

**Editor** – Může přidávat nové články do systému. Nově vytvořené články bude moci upravovat i mazat, ale nebude moci je zveřejňovat. V rámci modulu fotogalerie chce nahrávat nové fotky nebo dokumenty do fotogalerie.

**Redaktor** – Zastřešuje všechny editory. Chtěl by upravovat všechny články, mazat fotky z fotogalerie, starat se o kategorie fotogalerie. Mezi hlavní náplň bude patřit schvalování článků pro publikování.

**Administrátor** – Ten získá navíc právo starat se o všechny registrované osoby, přidělovat jim práva a role.

## 2.2 Informační systém

Informační systém je systém sběru, uchovávání, analýzy a prezentace dat určený pro poskytování informací mnoha uživatelům různých profesí. V nejobecnějším slova smyslu se informací chápe údaj o reálném prostředí, o jeho stavu a procesech v něm probíhajících. Informace snižuje nebo odstraňuje neurčitost systému (např. příjemce informace). Zvyšuje efektivitu při získávání informací. Informaci je možno také chápat jako data s nějakým přidaným významem (data + význam). Příkladem informačního systému může být kartotéka, telefonní seznam, kniha došlé pošty aneb účetnictví a systém pro správu fotografií a dokumentů. Systém nemusí být nutně automatizovaný pomocí počítačů a může být i v papírové podobě. V moderní době se setkáváme s informačními systémy stále častěji a jejich počet se bude neustále zvyšovat. Při návštěvě doktora si sestra vyhledá naši kartu v informačním systému ordinace. Lze říci že blízké budoucnosti nic nebude realizováno bez informačního systému nebo jeho nástupci. My budeme vytvářet webový informační systém (WIS) tzn. informační systém uzpůsobený provozem na webu. Část převzata z [11].

Obecně je chápeme jako systémy pro zpracování dat, které mají tyto cíle:

- strategické (plánování investic, vojenské plánování)
- taktické (vedení, kontrola rozpočtů...)
- operační (každodenní rutina)

Důležité jsou také úlohy IS:

- manažerské (EIS – Executive IS)
- taktické (DSS – Decision Support System)
- vedení (MIS – Management IS)
- expertní (KWS – Knowledge Work System)
- kancelářské (OIS – Office IS): Můj systém se nejvíce blíží této možnosti.
- operativní
  - TPS – transakční (banky...)
  - CRM – péče o zákazníka
  - RIS – rezervační systémy
  - CAM – konstrukční (CAD...)
  - GIS – geografické systémy

## 2.3 Databáze

Databáze je určitá uspořádaná množina informací (dat) uložená na paměťovém médiu. V širším smyslu jsou součástí databáze i softwarové prostředky, které umožňují manipulaci s uloženými daty a přístup k nim. Tento systém se v české odborné literatuře nazývá systém řízení báze dat (SRBD). Běžně se označením databáze – v závislosti na kontextu – myslí jak uložená data, tak i software (SRBD). V nepočítačovém světě je databáze i papírová kartotéka. Každá karta je prvek databáze s jednoznačnou identifikací např. podle rodného čísla. Převzata z [8]. V našem případě budeme

využívat relační model.

Existuje více způsobů ukládání dat a vztahů mezi nimi tím lze databáze rozdělit do více typů, respektive databázových modelů:

**Hiearchický model** – 2. polovina 60. let a dále. Záznamy jsou realizovány do stromů. Manipulace probíhá pomocí „navigačního programování“ po struktuře.

**Síťový databázový model** – norma z CODASYL z roku 1972, struktura dat byla realizována množinou záznamů a pojmenovanými vazbami (podobné ukazatelům). Vztahy realizovány fyzickými přímými adresami do adresových prostorů na discích. Pokud bylo potřeba přesunout nějaký záznam na jiné místo v databázi nebo do jiné databáze došlo ke změně hodnoty odkazu na něj.

**Relační model** – 1970 vznikla teorie a 1975 systém R (IBM). Data jsou uspořádána do tabulek, hodnoty v tabulce musí být z hlediska významu skalární. Manipulace s daty je velice jednoduchá: vždy se pracuje s řádky tabulky. Tento model je nejpoužívanější v současnosti a i já jej budu realizovat.

**Objektový model** – postupně se začíná prosazovat i objektový model databáze, kdy systém je chápán jako množina vzájemně se ovlivňujících objektů.

## 2.4 Dokument

Obsahuje informace. Mohou to být textové, obrazové či jiné informace. Z hlediska informatiky je to kontejner obsahující určitý druh informací. Z hlediska uložení je to soubor, kde jsou uloženy informace (uspořádané data). Dokumenty/soubory lze dělit podle různých kritérií.

Jednoznačně je určena přípona souboru a jeho velikost. Podle přípony souboru je identifikace ne vždy jednoznačná. Uživatel nám může nahrát soubor s příponou txt, ale bude obsahovat obrazová data. Uvedeme si zde jak rozdělení podle nejběžnějších skupin přípon tak i identifikaci pomocí MIME, které využívám i ve svém projektu.

### Přehled typů podle přípony:

**obrázek/fotografie** - různé fotky, malby, kresby schémata obvyklé přípony jsou .gif, .png, .jpg, .bmp, .tga, .tiff, .pcx

**video** - filmy, spoty, reklamy nejčastěji mají příponu .avi, .mov, .mpg, .wmw

**textové dokumenty** - přípona .txt, .pdf, .doc aj.

### Podle MIME typu:

MIME znamená Multipurpose Internet Mail Extensions a je to vlastně standartizovaný způsob obsahu zpráv v praxi hlavně elektronických dopisů a WWW dokumentů. I identifikaci využívám funkci, která zjišťuje MIME typ. Vše kolem MIME typů řídí IANA(Internet Assigned Number Authority) což je organizace která se lidově řečeno stará o Internet. Jsou zde definice internetových protokolů, MIME typů a mnoho dalších služeb. Vše je zaznamenáno přehledně v standartizovaných RFC dokumentech. Vše potřebné se lze dozvědět na <http://www.iana.org/>. Množství, které je zde zmíněno je jen zlomek MIME typů. Uvádím takové se kterými se lze setkat při běžném používání a které se budou ukládat do skladu dokumentů. Čerpal jsem z [4].

**Typ aplikace (application):** víceúčelové soubory

- *application/javascript*: JavaScriptové soubory
- *application/octet-stream*: Proud bytů. Nejčastěji to jsou spustitelné soubory. Z hlediska bezpečnosti by se pro přenos neměli používat, protože většinou obsahují škodlivý kód a jsou zdrojem virů a internetových červů. Nejlepší je zabalit a odeslat.
- *application/ogg*: multimediální proud informací, různá hudba nebo video
- *application/xhtml+xml*: XHTML dokumenty

**Typ zvuk(audio):** zvukové soubory/dokumenty

- *audio/mpeg*: mp3 nebo jiný mpeg formát audia
- *audio/x-wav*: zvuk ve formátu wav example

**Typ obraz(image):** soubory obsahující obrazové informace. Slouží k zobrazování fotografií

- *image/gif*: (Graphics Interchange Format) grafický formát pro rastrovou grafiku.
- *image/jpeg*: je standardní metoda ztrátové komprese používané pro ukládání počítačových obrázků ve fotorealistické kvalitě.
- *image/png*: Portable Network Graphics (přenosná síťová grafika) grafický formát určený pro bezztrátovou kompresi rastrové grafiky.
- *image/tiff*: (zkratka Tag Image File Format) je jeden z souborových formátů pro ukládání rastrové počítačové grafiky. Formát TIFF tvoří neoficiální standard pro ukládání snímků určených pro tisk.
- *image/vnd.microsoft.icon*: soubory s ikonami operačního systému Windows

**message**

**model multipart:** archívy či objekty obsahující více částí

- *multipart/mixed*: MIME E-mail
- *multipart/alternative*: MIME E-mail
- *multipart/related*: MIME E-mail a MIME HTML E-mail.

**text:** textové dokumenty a zdrojové kódy

- *text/css*: soubory kaskádových stylů
- *text/html*: HTML dokumenty
- *text/plain*: obyčejný text

**Typ video:** Video

- *video/mpeg*: video ve formátu MPEG-1 video se zvukem
- *video/mp4*: video ve formátu MP4
- *video/quicktime*: video ve formátu quicktime
- *video/x-ms-wmv*: formát videa od Microsoft Windows

## 2.5 Fotografie

Fotografie je proces získávání a uchovávání obrazu. Fotografie se řadí mezi zobrazovací technologie a už od svých počátků se těší velkému zájmu vědců i umělců. Vědci používají fotografii pro přesný a věrný záznam skutečnosti. Umělci se stejně tak zajímali o věrnost zobrazení fotografie, ovšem zkoumali také další nabízející se možnosti než je pouhé zachycování obrazu reálného světa. Část citována z [9]

Fotografii začala brzy využívat také armáda, policie a bezpečnostní složky. Dnes je fotografie nepřehlédnutelně zastoupena téměř ve všech médiích (noviny, internet, TV), využívá se hojně v marketingu a reklamě a jednoduché fotoaparáty v mobilních telefonech možná naznačují její další rozmach.

Je to speciální typ dokumentu, obsahující obrazové informace. V našem případě bude nabývat MIME typu image(obrázek). Nejčastěji se objevuje *image/jpeg* – fotky z digitálních fotoaparátů. Další je *image/gif*, malé fotky či kresby. Na fotkách můžeme zobrazit nejen 2D, ve formě vektorové (náčrtky, kresby, diagramy aj.) a rastrové grafiky (fotografie), ale i 3D což objekty reálného světa (koule, krychle, postava). Fotografii jako jedinému typu bude v informačním systému věnována zvláštní péče co se týká zpracování a zobrazování. Fotografie lze zmenšovat a vytvářet náhledy což bude pro prezentaci velice důležité.

## 2.6 Ukládání dokumentů

Dokumenty jak jsme si řekli je soubor informací/dat. Proto bychom měli zjistit jaké možnosti máme s ohledem na ukládání. Jelikož budeme využívat při naší práci webový server lze si vybrat z mnoha možností. Uvedu jen dvě, které by se daly realizovat. Budeme se zabývat digitálním ukládáním dokumentů.

**Na disk** Pro ukládání souborů se využívá většinou disk/server. Přístup k souborům je realizován buď přes FTP protokol nebo webové rozhraní. Výhoda tohoto způsobu je v rychlé přenositelnosti a jednoduché manipulaci. Programů pro přístup na FTP existuje desítky. Internetových prohlížečů je také bezpočet. Pokud je potřeba přenést všechny dokumenty prostě se přenese celý disk/server.

**Do databáze** Druhý způsob je ukládání souborů do databáze, většinou sloupec typu „medimblob“. Výkon databáze se mírně zhorší, ale hlavně budeme muset pro manipulaci se soubory napsat speciální skripty. Do databáze ukládat taková velká data je velice nevhodné. Velikost databáze je na většině serverů omezena na pár megabajtů, ale velikost serverů přesahuje stovky megabajtů.

## 2.7 Použitelnost

Jiným slovem usability. Použitelnost webových stránek určuje míru a efektivitu úsilí, nutného k nalezení a získání potřebných informací. Určuje jak snadno se na stránkách uživatelé orientují, jak rychle pochopí jejich uspořádání a ovládání a jaký uživatelský zážitek si z nich odnesou. Je lepší být výjimečný

Mnoho tvůrců webu se snaží návštěvníka nějak zaujmout, aby na ty jeho stránky, jen tak nezapomněl. Snaží se na něj zapůsobit speciálními vizuálními efekty, netradičním layoutem, či různými hříčkami a hádankami. Někdy je ale tato přílišná originalita na škodu věci.

Návštěvník se bude zcela jistě lépe cítit na webu, který sice není moc originální, ale na první pohled je jasné kde je navigace, na co se má kliknout a na co je web zaměřen. Převzato z [2].

## 2.8 Přístupnost webu

Přístupný web je web bez bariér, který plně respektuje své uživatele. Respektuje jejich trvalé či dočasné zdravotní nesnáze, technické vybavení, jejich znalosti a dovednosti. Web by měl být přístupný z většiny majoritních prohlížečů a v jakémkoli rozlišení. V současnosti většina uživatelů využívá minimálně rozlišení 800x600 nebo 1024x768. Web musí být přístupný nejen většině, ale všem kdo mají zájem navštívit naši prezentaci. V další kapitole si řekneme něco o pravidlech přístupného webu. Část převzata z [2].

Mezi skupiny, kteří se bez kvalitní použitelnosti nevrátí jsou:

**Uživatelé minoritních operačních systémů a prohlížečů** – Nejde jen o pár počítačových fanatiků, kteří vidí jen jedničky a nuly, ale jedná se i o profesionální programátory, IT odborníky a všechny, kterým vyhovuje něco jiného než většině.

**Uživatelé jiných zobrazovacích zařízení** – Žijeme ve třetím tisíciletí - v době rychlého rozšiřování moderních zařízení (osobních organizérů, chytrých telefonů, televizních set-boxů, konzolí a multi-mediálních systémů v automobilech, atd.).

**Majitele zastaralých počítačů** – Ačkoliv vy možná máte to nejnovější PC s 19" monitorem, stále ještě existuje tisíce malých kanceláří se zastaralými počítači, vybledlými malými monitory, kde je těžké rozeznat zelenou od hnědé.

**Dyslektiky, barvoslepé, slabozraké, apod.** – Tito navštěvníci mají problémy se špatným rozvržením barev, malou velikostí písma či špatně strukturovanými texty. Potřebují si velikost písma zvětšit. Neumožníte-li jim to, nebo zvětšení písma způsobí rozpad celého designu, těžko váš web budou používat.

**Uživatelé s trvalou či dočasnou nemožností používat horní končetiny** – Ať už vždy nebo jen dočasně s rukou v sádře, takoví lidé těžko využijí k ovládání stránek myš. Zbývá jen klasická nebo upravená klávesnice.

**Vyhledávače** – Ačkoliv si to většina lidí neuvědomuje, jsou roboti vyhledávačů těmi nejpostiženější uživateli webu. Nevidí obrázky, nerozeznají barvy, nepoužívají JavaScript, znají jen texty a odkazy. Vyhledávače prostě z podstaty své funkčnosti přístupné weby milují. Je až komické, jak se některé weby snaží o optimalizaci pro vyhledávače, šperkují meta tagy a přitom vinou nevhodně napsaného JavaScriptového menu nepustí vyhledávač dál než na titulní stranu.

### 2.8.1 Pravidla přístupného webu

Jsou to pravidla, kterými bychom se měli řídit při vytváření jakékoli webové prezentace. Pravidla mají doporučující charakter, mají být vodítkem pro zadavatele webových stránek a inspirací pro jejich správce. Best practice – Pravidla pro tvorbu přístupného webu připravila pracovní skupina složená ze zástupců Metodického centra informatiky Sjedenocené organizace nevidomých a slabozrakých (SONS), odborných konzultantů a pracovníků Ministerstva informatiky. Citováno z [1].

#### Obsah webových stránek je dostupný a čitelný

- Každý netextový prvek nesoucí významové sdělení má svou textovou alternativu.

- Informace sdělované prostřednictvím skriptů, objektů, appletů, kaskádových stylů, obrázků a jiných doplňků na straně uživatele jsou dostupné i bez kteréhokoli z těchto doplňků.
- Informace sdělované barvou jsou dostupné i bez barevného rozlišení.
- Barvy popředí a pozadí jsou dostatečně kontrastní. Na pozadí není vzorek, který snižuje čitelnost.
- Předpisy určující velikost písma nepoužívají absolutní jednotky.
- Předpisy určující typ písma obsahují obecnou rodinu písem.

#### **Práci s webovou stránkou řídí uživatel**

- Obsah WWW stránky se mění, jen když uživatel aktivuje nějaký prvek.
- Webová stránka bez přímého příkazu uživatele nemanipuluje uživatelským prostředím.
- Nová okna se otevírají jen v odůvodněných případech a uživatel je na to předem upozorněn.
- Na webové stránce nic neblinká rychleji než jednou za sekundu.
- Webová stránka nebrání uživateli posouvat obsahem rámců.
- Obsah ani kód webové stránky nepředpokládá ani nevyžaduje konkrétní způsob použití ani konkrétní výstupní či ovládací zařízení.

#### **Informace jsou srozumitelné a přehledné**

- Webové stránky sdělují informace jednoduchým jazykem a srozumitelnou formou.
- Úvodní webová stránka jasně popisuje smysl a účel webu. Název webu či jeho provozovatele je zřetelný.
- Webová stránka i jednotlivé prvky textového obsahu uvádějí své hlavní sdělení na svém začátku.
- Rozsáhlé obsahové bloky jsou rozděleny do menších, výstižně nadepsaných celků.
- Informace zveřejňované na základě zákona jsou dostupné jako textový obsah webové stránky.
- Na samostatné webové stránce je uveden kontakt na technického správce a prohlášení jasně vymezující míru přístupnosti webu a jeho částí. Na tuto webovou stránku odkazuje každá stránka webu.

#### **Ovládání webu je jasné a pochopitelné**

- Každá webová stránka má smysluplný název, vystihující její obsah.
- Navigační a obsahové informace jsou na webové stránce zřetelně odděleny.
- Navigace je srozumitelná a je konzistentní na všech webových stránkách.
- Každá webová stránka (kromě úvodní webové stránky) obsahuje odkaz na vyšší úroveň v hierarchii webu a odkaz na úvodní WWW stránku.
- Všechny webové stránky rozsáhlejšího webu obsahují odkaz na přehlednou mapu webu.



- Obsah ani kód webové stránky nepředpokládá, že uživatel již navštívil jinou stránku.
- Každý formulářový prvek má přiřazen výstižný nadpis.
- Každý rám má vhodné jméno či popis vyjadřující jeho smysl a funkčnost.

### **Odkazy jsou zřetelné a návodné**

- Označení každého odkazu výstižně popisuje jeho cíl i bez okolního kontextu.
- Stejně označené odkazy mají stejný cíl.
- Odkazy jsou odlišeny od ostatního textu, a to nikoli pouze barvou.
- Obrázková mapa na straně serveru je použita jen v případě, že nebylo možné pomocí dostupného geometrického tvaru definovat oblasti v obrázkové mapě. V ostatních případech je použita obrázková mapa na straně uživatele. Obrázková mapa na straně serveru je vždy doprovázena alternativními textovými odkazy.
- Uživatel je předem jasně upozorněn, když odkaz vede na obsah jiného typu, než je webová stránka. Takový odkaz je doplněn sdělením o typu a velikosti cílového souboru.

### **Kód je technicky způsobilý a strukturovaný**

- Kód webových stránek odpovídá nějaké zveřejněné finální specifikaci jazyka HTML či XHTML. Neobsahuje syntaktické chyby, které je správce webových stránek schopen odstranit.
- V metaznačkách je uvedena použitá znaková sada dokumentu.
- Prvky tvořící nadpisy a seznamy jsou korektně vyznačeny ve zdrojovém kódu. Prvky, které netvoří nadpisy či seznamy, naopak ve zdrojovém kódu takto vyznačeny nejsou.
- Pro popis vzhledu webové stránky jsou upřednostněny stylové předpisy.
- Je-li tabulka použita pro rozvržení obsahu webové stránky, neobsahuje záhlaví řádků ani sloupců. Všechny tabulky zobrazující tabulková data naopak záhlaví řádků a/nebo sloupců obsahují.
- Všechny tabulky dávají smysl čtené po řádcích zleva doprava.

## **2.9 Optimalizace**

Je proces výběru nejlepší varianty ze všech možností. V mé aplikaci to např. byl výběr MySQL systému pro ukládání informací o dokumentech. Jak si můžete přečíst výše MySQL je velice rychlý a optimalizovaný databázový systém. Dále jsem se snažil optimální uživatelské rozhraní. V době rozmachu počtu internetových stránek, většina lidí nepřistupuje na stránky přímo, ale přes specializované vyhledávače. Proto vznikla disciplína, zabývající se optimalizacemi webu pro lepší výsledky při vyhledávání.

### **2.9.1 Optimalizace pro vyhledávače (SEO)**

SEO je zkratka, která znamená Search Engine Optimization, tedy česky optimalizace pro vyhledávače. SEO jako takové je vlastně obecný postup, jak optimalizovat konkrétní webové stránky pro vyhledávače. SEO se zabývá optimalizací pro fulltextové vyhledávání, tzn. pro vyhledávání klíčového slova nebo slov. Optimalizací se v této souvislosti myslí vytvoření nebo upravení struktury webových stránek tak, aby vyhledávače zaindextovaly (uložili do své databáze) co nejvíce klíčových slov nebo frází, obsažených na těchto stránkách. Situace na internetu je velmi podobná reálnému konkurenčnímu prostředí. Čím více internet obsahuje webových prezentací, tím je náročnější přimět návštěvníka k návštěvě právě našeho webu. Je lhostejné zda se jedná o komerční web nebo neziskový server, naším cílem vždy je, když už vynaložíme energii, čas a prostředky ke vzniku nějakých stránek, aby na tyto stránky přicházeli návštěvníci, pokud možno v co největší míře.

# Kapitola 3

## Návrh

Navrhne vše co budu pro vytvoření kvalitního informačního systému pro správu fotografií a dokumentů potřebovat. S ohledem na požadavky uživatele vytvořím diagram případů použití na který využiji jazyk UML. Z analýzy nám vyplynulo, že nejvhodnější pro ukládání informací o dokumentech a fotografiích bude databáze. Proto si vytvoříme návrh databázových tabulek vycházející z navrhnutého ER diagramu. Pro návrh diagramů se využívá UML, protože nám poskytuje skvěle možnosti při navrhování nejen informačních systémů.

### 3.1 UML

UML, Unified Modeling Language je v softwarovém inženýrství grafický jazyk pro vizualizaci, specifikaci, navrhování a dokumentaci programových systémů. UML nabízí standardní způsob zápisu jak návrhů systému včetně konceptuálních prvků jako jsou business procesy a systémové funkce, tak konkrétních prvků jako jsou příkazy programovacího jazyka, databázová schémata a znovupoužitelné programové komponenty.

UML podporuje objektově orientovaný přístup k analýze, návrhu a popisu programových systémů. UML neobsahuje způsob, jak se má používat, ani neobsahuje metodiku(y), jak analyzovat, specifikovat či navrhovat programové systémy. Čerpal jsem z [14] a [6].

Pohledy jsou konkretizovány v následujících typech diagramů:

- **Diagram případů užití** – Zobrazení dynamické (funkční) struktury systému z pohledu uživatele. Je primárně určen k definici chování systému, aniž by odhaloval jeho vnitřní strukturu. Dále si ho lze přestavit jako soubor scénářů pro používání systému.
- **Diagram tříd (diagram objektů)** – Zobrazení statické struktury systému prostřednictvím tříd a vztahů mezi nimi.
- **Diagram komponent** – je to model komponent systému a jejich spolupráce
- **Diagram nasazení** – je model rozložení komponent při běhu systému
- **Sekvenční diagram** – model časové dynamiky uvnitř případů užití (Use Case) tzn. zobrazuje interakce ve vztahu k času. Každý objekt má časovou osu (čáru života). Lze s ním také vyjádřit chronologii zasílání zpráv.
- **Stavový diagram** – Popisuje stavy objektů a povolené přechody mezi těmito stavy

- Diagram aktivit – reprezentují objektově diagramy toků, lze je připojit k libovolnému modelovanému elementu. Obsahuje počátek a konec, stavy aktivity, přechody mezi stavy. Je to model průběhu jednotlivých Use Case (případů užití) a operací v třídách

## 3.2 Model systému - ER diagram

K vytvoření modelu systému využijeme ER diagram. Slouží k modelování dat aplikační domény a jejich vztahů „v klidu“. Slouží k datovému modelování. Má definovanou určitou grafickou notaci. ER diagram je na obrázku 3.1.

Základní pojmy datového modelu (ER modelu):

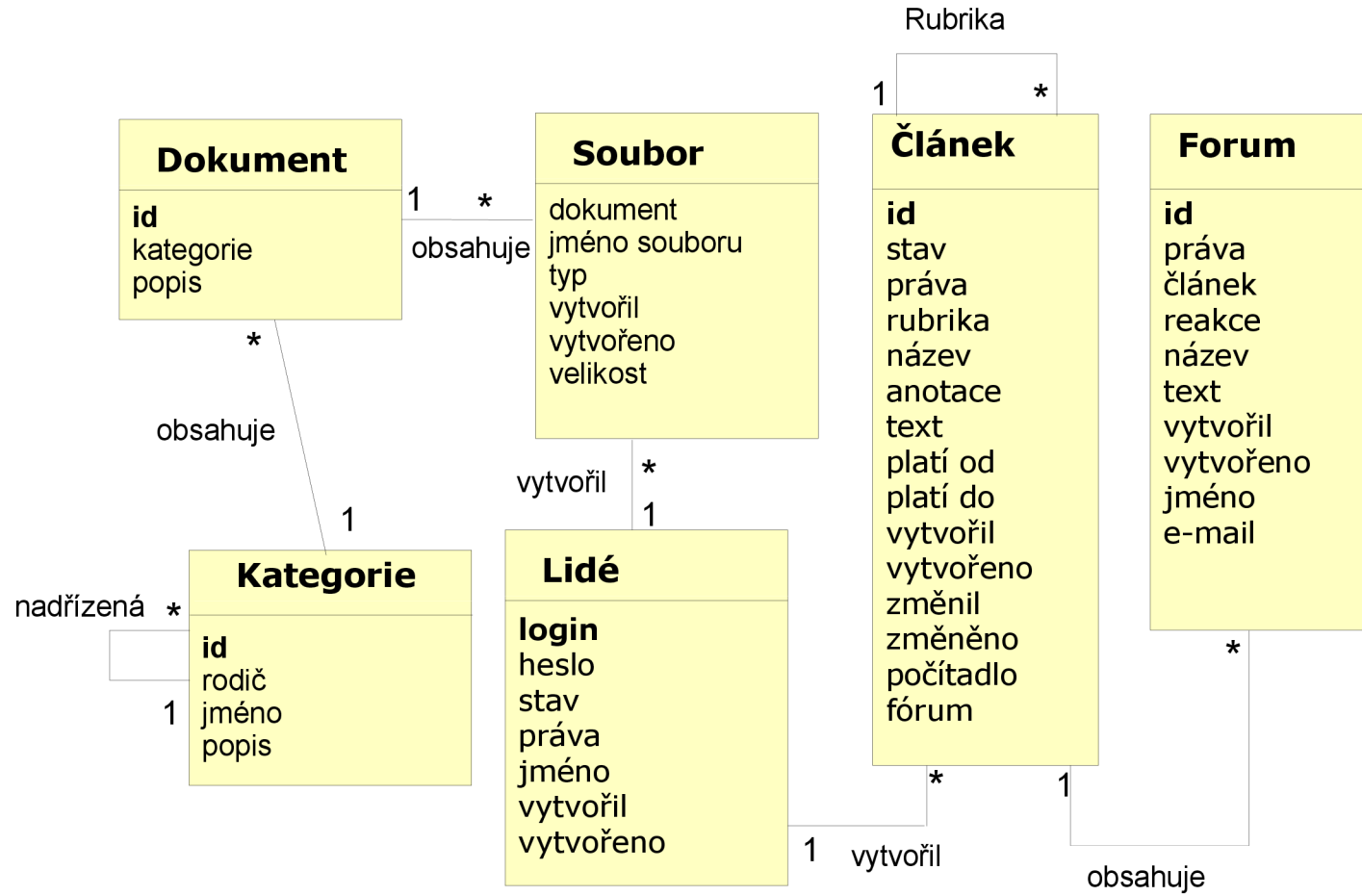
**Entita** – věc reálného světa (objekt) rozlišitelný od jiných objektů. Např. fotka domku s číslem 99

**Entitní množina** – množina entit stejného druhu či typu Např. dokument, uživatel, článek

**Atribut** – je vlastnost entity, která nás zajímá Např. dokument: jméno, popis

**Vztah** – asociace mezi entitami (objekty) Např. kategorie bydlení obsahuje fotky domů

Obrázek 3.1 : ER diagram



### 3.3 Tabulky databáze

Z návrhu ER diagramu vytvořím potřebné databázové tabulky pro můj databázový systém. U jednotlivých položek bude i datový typ. Podrobně uvedu jen tabulky<sup>1</sup> tykající se informačního systému pro správu dokumentů a fotografií což je tabulka Dokument, Kategorie a Soubor. Ostatní tabulky zmíním jen okrajově.

#### Dokument

**id:** číslo

*kategorie:* číslo

*jmeno:* text

*popis:* text

Primární klíč je **id**.

Cizí klíč je *kategorie* odkazující se na tabulku kategorie.

#### Kategorie

**id:** číslo

*id\_rodice:* číslo

*jmeno:* text

*popis:* text

Primární klíč je **id**.

Cizí klíč je *id\_rodice* odkazující se na tabulku kategorie.

#### Soubor

*id\_dokumentu:* číslo

*jmeno\_souboru:* text

*typ:* text

*login:* text

*datum\_pridani:* datum

*velikost:* číslo

Cizí klíč je *id\_dokumentu* odkazující se na tabulku kategorie.

Cizí klíč je *login* odkazující se na tabulku lide.

#### Lide

**login:** text

*heslo:* text

Primární klíč je **login**.

#### Clanek

**id:** číslo

*Stav:* text

*rubrika:* číslo

Primární klíč je **id**.

Cizí klíč je *rubrika* odkazující se na tabulku clanek.

<sup>1</sup>Podtržené slovo značí tabulku, tučná položka značí primární klíč a kurzívou značíme cizí klíč

Forum

**id:**číslo

*clanek:*číslo

reakce:text

Primární klíč je **id**.

Cizí klíč je *clanek* odkazující se na tabulku clanek.

### 3.4 Diagram případů použití

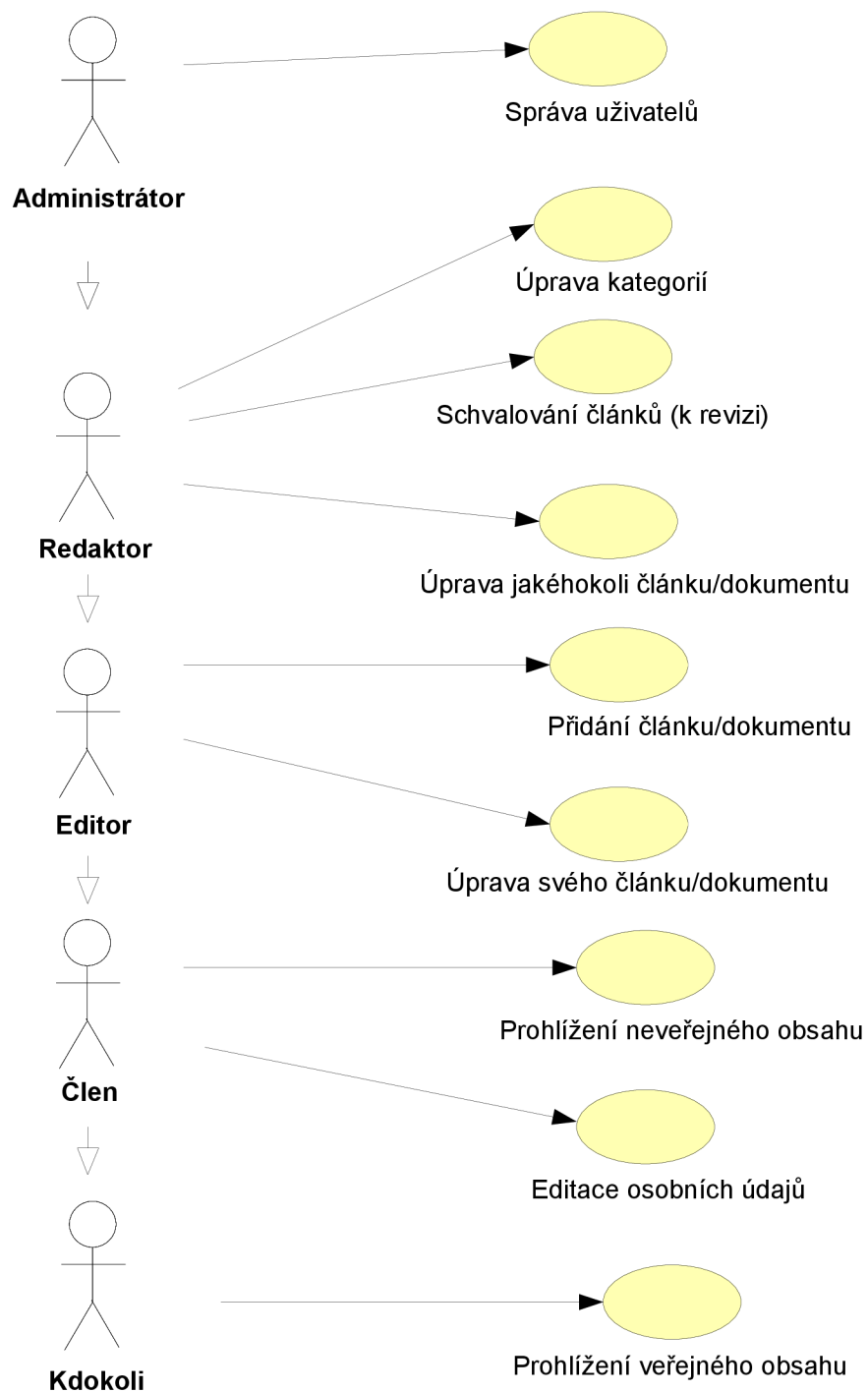
Uživatelé si nejsou rovnocenní. Následující diagram případů použití ukazuje (Use-case diagram) jaké činnosti lze vykonávat. Každému uživateli jsou přidělena práva. Na obrázku 3.2 lze vidět hierarchickou strukturu rolí uživatelů. Je zde využito dědičnosti tzn. ze výše postavený dědí práva na činnosti níže postaveného jedince. Díky následujícímu řešení lze systém velice rychle rozšířit.

**Role** – na obrázku značena panáčkem což je reprezentace uživatele

**Případ použití** – dokumentace události, na obrázku je vidět v podobě elipsy s popisem události

**Komunikace** – vazba mezi rolí a případem použití. Na obrázku je vyznačena pomocí plné šipky

**Hierarchy** – Role jsou v hierarchickém vztahu. Výše hierarchicky zobrazená role může vykonávat více činnosti a dědí možnosti níže hierarchicky zobrazené role. Na obrázku je hierarchie vyznačena prázdnými šipkami.

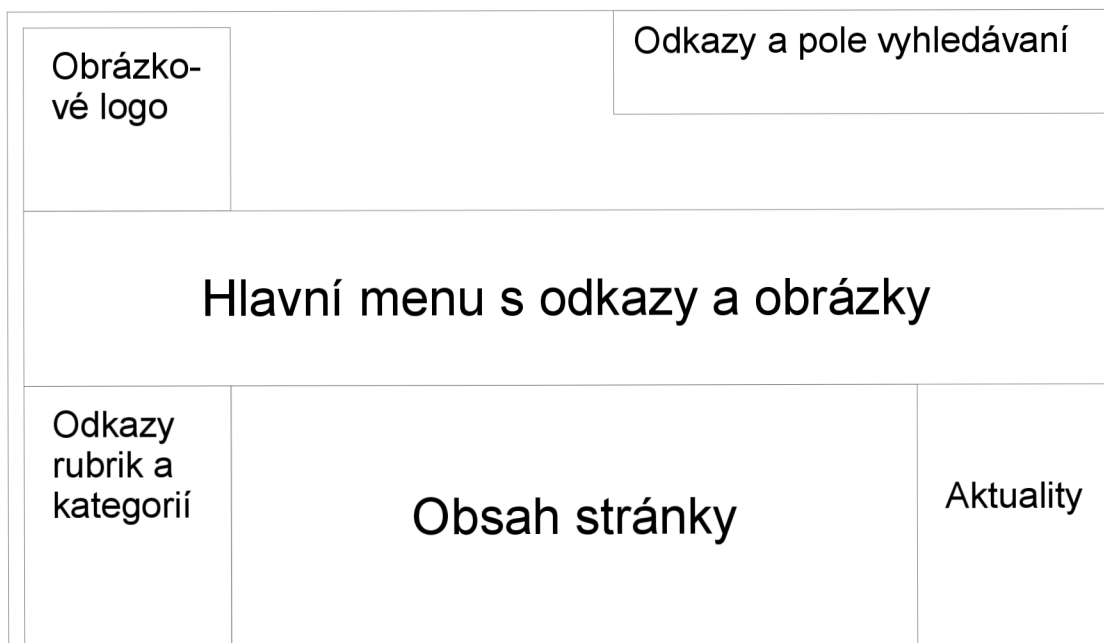


Obrázek 3.2: Diagram případů užití

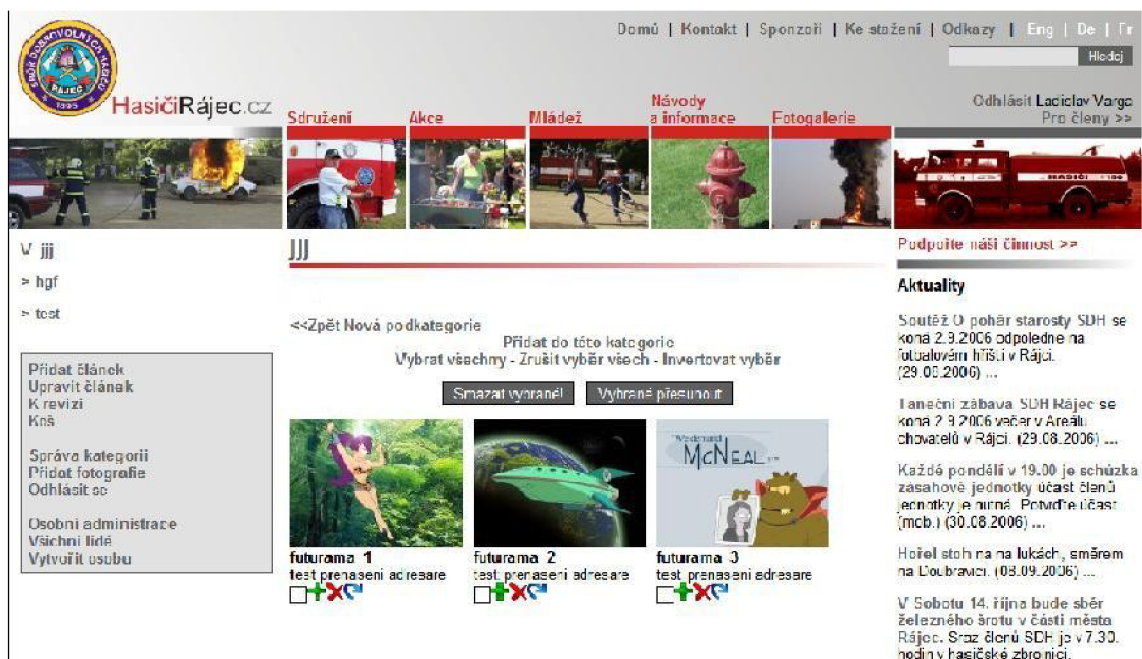


### 3.5 Návrh vzhledu

Na obrázku 3.3 vidíte přibližný vzhled s nejdůležitějšími oblastmi. Neexperimentuji, proto obsah zabírá největší oblast stránky. Důležitá je přehlednost a dostupnost všech ovládacích prvků. Na dalším obrázku 3.4 lze vidět výsledek již vytvořeného vzhledu.



Obrázek 3.3: Návrh vzhledu



Obrázek 3.4: Výsledek návrhu vzhledu (<http://www.hasicirajec.cz/foto/>)

## Kapitola 4

# Implementace

Na programování systému jsem si vybral jazyk PHP, který se mi zdál pro implementaci nejvhodnější. Existuje pro něj mnoho knihoven a skriptů. Jelikož je to nejvyužívanější jazyk pro publikaci dynamických stránek je zajištěna podpora ze strany ostatních uživatelů.

Při implementaci se snažím dbát na čistotu kódu a přehlednou strukturu souborů. Hojně implementuji funkce a snažím se o jednoduchost. Dodržuji nepsané zásady správného programování. Podobný systém by jistě šel napsat i v jiném jazyce např. ASP .net, Delphi, C#. Pro implementaci uložení dat jsem využil databázový systém MySQL.

Fotogalerie bude nasazena na webu tzn. musel jsem si vybrat nějaký publikační jazyk pro implementaci uživatelského rozhraní avzhledu. Volba byla jednoduchá. Využívám HTML ve spolupráci s CSS pro nastavení vzhledu. Uspořádání stránky je tvořeno tabulkovým layoutem. V budoucnu je možné přejít na CSS layout. Jestliže některé prohlížeče nebudou dodržovat standarty byl by převod časové nákladný.

Na práci s funkcemi, které se provádí na straně klienta používám JavaScript. Velice dobře se v něm dělají náhledy formulářů což využívám u přidávání článků. Dále ho využívám při vytváření funkcí pro označení všech fotografií a dokumentů, zvyšuje se tím efektivnost práce při spravování fotogalerie. Pokud je podpora JavaScriptu vypnuta funkčnost veřejně přístupného obsahu tím netrpí.

Web se neustále rozvíjí a vznikají nové technologie. Jednou z nich je technologie AJAX, která dokáže velice atraktivně práci se systémem. Pomocí ní lze vytvářet různé opravdu dynamické prvky stránky, jako je např. přenášení podoken, vysouvací menu, dynamická tlačítka atd. V mém případě využívám technologii AJAX prostřednictvím malé aplikace Thickbox, což je dynamické zobrazování velkého formátu fotografií.

Pro implementaci informačního systému pro správu dokumentů a fotografií jsem využil php s databázovým systémem Mysql. Jelikož programuji webový informační systém pro výstup bude použito klasické html spolu s css. Na některé funkce využiji JavaScript a jeden plugin z Ajaxu. Velký důraz je kladen na administrační část systému.

V kapitole vlastní implementace popíši vlastní přínos k vytvoření informačního systému pro správu fotografií a dokumentů. Po implementaci následuje testování, o kterém řeknu ty nejdůležitější informace. Na konci zmíním další vývoj systému, kde by měl být nasazen a co by se dalo ještě vylepšit.

## 4.1 Prostředí implementace

### 4.1.1 PHP

PHP (rekurzivní zkratka PHP: Hypertext Preprocessor, „PHP: Hypertextový preprocesor“, původně Personal Home Page) je skriptovací interpretovaný programovací jazyk, určený především pro programování a generování dynamických internetových stránek. Nejčastěji se začleňuje přímo do struktury jazyka HTML, XHTML či WML, což je velmi výhodné pro tvorbu webových aplikací. PHP lze ovšem také použít i k tvorbě konzolových a desktopových aplikací. Více se lze dozvědět na [13]. Podle údajů z dubna 2004 je PHP nasazen na více než 15 000 000 doménách a je to bezkonkurenčně nejčastěji používaný modul webového serveru Apache. Řekneme si něco proč programovat v PHP a proč ne.

#### Výhody:

- PHP podporuje širokou řadu souvisejících technologií, formátů a standardů
- je to otevřený projekt s rozsáhlou podporou komunity
- dají se najít kvanta již hotového kódu k okamžitému použití nebo funkční m PHP aplikace. Podstatná část z hotového kódu je šířena pod nějakou svobodnou licenci a dá se použít ve vlastních projektech.
- PHP si dobře rozumí s webovým serverem Apache (aby ne, vždyť je to sesterský projekt spravovaný Apache software foundation)
- PHP snadno komunikuje s databázemi, jako je MySQL, PostgreSQL a řada dalších
- PHP je multiplatformní a lze jej provozovat s většinou webových serverů a na většině dnes existujících operačních systémů
- PHP podporuje mnoho existujících poskytovatelů webhostingových služeb

#### Nevýhody:

- PHP je interpretovaný, ne kompilovaný jazyk
- kdokoli kdo má přímý přístup k serveru, může nahlédnout do vašich PHP skriptů
- protože je PHP aktivně vyvíjen, v budoucích verzích jazyka se mohou některé funkce změnit nebo se mohou chovat jinak než dosud což může vést až k nefunkčnosti aplikace
- je typově laxní tzn. číslo lze přiřadit řetězec a naopak což velice snižuje kontrolu kódu

Masově se začala využívat PHP verze 4, ve které je i můj informační systém. Čerpal jsem z [3].

#### PHP 4:

- zdokonalené zacházení s prostředky
- podpora objektově orientovaného programování
- domácí podpora zpracování sezení

- šifrování
- podpora ISAPI
- přirozená podpora COM/DCOM
- přirozená podpora javy
- knihovna regulárních výrazů kompatibilní s Perlem

PHP 5:

- nesmírně zdokonalená výbava objektově orientovaného programování
- zpracování výjimek ve stylu try/catch
- zdokonalené zpracování řetězců
- zdokonalená podpora XML a webových služeb
- domácí podpora SQLite

#### 4.1.2 MySQL

MySQL je databázový systém, vytvořený švédskou firmou MySQL AB. Jeho hlavními autory jsou Michael „Monty“ Widenius a David Axmark. Je považován za úspěšného průkopníka dvojího licencování – je k dispozici jak pod bezplatnou licenci GPL, tak pod komerční placenou licenci.

MySQL je multiplatformní databáze. Komunikace s ní probíhá – jak už název napovídá – pomocí jazyka SQL. Podobně jako u ostatních SQL databází se jedná o dialekt tohoto jazyka s některými rozšířeními. Čerpám z [3]

Od prvního veřejného vydání roce 1995 kladli vývojáři MySQL značný důraz na výkon a škálovatelnost softwaru. To vedlo k vysoce optimalizovanému produktu, který bohužel postrádal schopnosti důležité pro velké korporace: např. uložené procedury, triggerů a transakce. I přesto si získal obrovskou uživatelskou základnu. Seznámíme se tím proč je MySQL tak oblíbené a používané.

Přednosti MySQL databáze:

**Cachování dotazů** – jedna z nejlepších vlastností pro zvýšení výkonu aplikace. Když je schopnost zapnutá funguje velice jednoduše a efektivně. MySQL si uchovává dotazy SELECT spolu s jejich výsledky. Když se spouštějí další dotazy, MySQL je porovná s dotazy uloženými v cache. Pokud nalezne shodu, přeskočí nákladné získávání dat z databáze a prostě vytáhne výsledky z cache. Aby se zamezilo zastaralým výsledkům jsou zabudované mechanismy, které automaticky nahrazují stará data.

**Fulltextové indexování a vyhledávání** Od verze 3.23.23 he MySQL o fulltextové indexování a vyhledávání. Je to schopnost, která zvyšuje výkon při dolování dat ze sloupců obsahující texty.

**Replikace** Verze 3.23.15 přinesla do MySQL replikační schopnosti. Umožňuje zkopírování (duplikaci) databáze na jiný server. Využívá se hlavně při zálohování. Pokud vlastníme více strojů, můžeme rozprostřít zátěž na více počítačů.

**Multiplatformnost** Jsou k dispozici binární distribuce MySQL pro minimálně dvanáct platform: Dec OSF, FreeBSD, IBM AIX, HP-UX, Linux, Mac OS X, Novell Netware, OpenBSD, QNX, SGI Irix, Solaris a Microsoft Windows. Lze si stáhnout i zdrojový kód a zkompilovat si MySQL sami.

Ještě si něco řekneme o nejvýznamnějších verzích MySQL.

#### **MySQL 4:**

- přidání InnoDB do standardní distribuce
- skladování dotazů
- vložený (embedded)server MySQL
- poddotazy
- bezpečná připojení přes SSL (Secure Socket Layer)

**MySQL 5:** na mém systému využívám právě MySQL verzi 5

- kompletní podpora cizích klíčů
- uložené procedury
- trigger

### **4.1.3 HTML**

HTML je zkratka z anglického HyperText Markup Language, značkovací jazyk pro hypertext. Je jedním z jazyků pro vytváření stránek v systému World Wide Web, který umožňuje publikaci stránek na Internetu.

HTML je podmnožinou dříve vyvinutého rozsáhlého univerzálního značkovacího jazyka SGML (Standard Generalized Markup Language).

**SGML** je univerzální značkovací metajazyk, který umožňuje definovat značkovací jazyky jako své podmnožiny. Obecně lze chápat **značkovací jazyk** jako zdrojový text obsahující současně jak vlastní text tak i instrukce pro jeho zpracování. Vývoj HTML byl ovlivněn vývojem webových prohlížečů, které zpětně ovlivňovaly definici jazyka. Zvláštní podmnožinou SGML jazyka je i **XML** což je obecný značkovací jazyk. Umožňuje snadné vytváření konkrétních značkovacích jazyků pro různé účely a široké spektrum různých typů dat.

Pro každou verzi existuje definice pravidel DTD (Document Type Definition). Od verze 4.01 musí být odkaz na deklaraci DTD v dokumentu uveden pomocí klíčového slova DOCTYPE. DTD definuje pro určitou verzi, které elementy je možné používat a s jakými atributy. Vše potřebné lze nalézt na [10].

Dokument může mimo značkování obsahovat další prvky:

**Direktivy** – začínají znaky „<!“, jsou určeny pro zpracovatele dokumentu (prohlížeč)

**Komentáře** – pomocné texty pro programátora, nejsou součástí obsahu dokumentu a nezobrazují se (prohlížeč je ignoruje).

## Struktura dokumentu

Dokument v jazyku HTML má předepsanou strukturu:

**Deklarace DTD** – je povinná až ve verzi 4.01, je uvedena direktivou `<!DOCTYPE`.

**Kořenový element** – element `html` (značky `<html>` a `</html>`) reprezentuje celý dokument. Je nepovinný, ale je doporučeno ho používat.

**Hlavička elementu** - jsou to metadata, která se vztahují k celému dokumentu. Definují např. název dokumentu, jazyk, kódování, klíčová slova, popis, použitý styl zobrazení. Hlavička je uzavřena mezi značky `<head>` a `</head>`.

**Tělo dokumentu** - obsahuje vlastní text dokumentu. Vymezuje se značkami `<body>` a `</body>`

### 4.1.4 CSS

CSS je zkratka pro anglický název Cascading Style Sheets, česky tabulky kaskádových stylů. Je to jazyk pro popis způsobu zobrazení stránek napsaných v jazycích HTML, XHTML nebo XML.

Hlavním smyslem je umožnit návrhářům oddělit vzhled dokumentu od jeho struktury a obsahu. Původně to měl umožnit už jazyk HTML, ale v důsledku nedostatečných standardů a konkurenčního boje výrobců prohlížečů se vyvinul jinak. Starší verze HTML obsahují celou řadu elementů, které nepopisují obsah a strukturu dokumentu, ale i způsob jeho zobrazení. Z hlediska zpracování dokumentů a vyhledávání informací není takový vývoj žádoucí. Čerpáno z [5].

### 4.1.5 JavaScript

Podle [12] je JavaScript multiplatformní, objektově orientovaný skriptovací jazyk, jehož autorem je Brendan Eich z tehdejší společnosti Netscape.

Nyní se zpravidla používá jako interpretovaný programovací jazyk pro WWW stránky, vkládaný přímo do HTML kódu stránky. Jsou jím obvykle ovládány různé interaktivní prvky GUI (tlačítka, textová políčka) nebo tvořeny animace a efekty obrázků.

Program JavaScriptu se obvyklé použít až po stažení WWW stránky z Internetu (tzv. na straně klienta), narozdíl od jiných interpretovaných jazyků (PHP a ASP), které se použít na straně serveru ještě před stažením z internetu. Z toho vyplývají určitá bezpečnostní omezení: JavaScript nemůže pracovat se soubory, aby neohrozil soukromí uživatele.

### 4.1.6 Ajax

AJAX (Asynchronous JavaScript and XML) je obecné označení pro technologie vývoje interaktivních webových aplikací, které mění obsah svých stránek bez nutnosti jejich znovunačítání. Na rozdíl od klasických webových aplikací poskytují uživatelsky příjemnější prostředí, ale vyžadují použití moderních webových prohlížečů. Část převzata z [7].

Tyto aplikace jsou vyvíjeny s využitím technologií: HTML (nebo XHTML) a CSS pro prezentaci informací,

DOM a JavaScript pro zobrazování a dynamické změny prezentovaných informací,

XMLHttpRequest pro asynchronní výměnu dat s webovým serverem (typicky je užíván formát XML, ale je možné použít libovolný jiný formát včetně HTML, prostého textu, JSON či EBML).

## 4.2 Thickbox

Pro zobrazování fotografie ve velkém formátu využívám ajaxovou aplikaci Thickbox napsanou v Ajaxu. Je dostupný na stránce <http://jquery.com/demo/thickbox/>. Nevýhoda téhle aplikace je že se stává aktivní až po načtení kompletního obsahu stránky. Thickbox je jedna z mnoha aplikací, která otevírá velké formáty obrázků. Existuje např. Lightbox, Hoverbox a mnoho dalších. [7]

## 4.3 Vlastní implementace

Mám za úkol vytvořit informační systém pro vkládání a vyhledávání dokumentů. Systém jsem implementoval jako modul redakčního systému. Navrhl jsem strukturu tabulek databáze pro ukládání dokumentů a fotografií. Vytvořil jsem nástroje pro vytvoření nové fotogalerie tzn. systém zvládá vytváření nových kategorií a podkategorií. Celé kategorie lze libovolně přesouvat společně i s všemi dokumenty. Kategorii může změnit jméno a popis.

Dále jsem implementoval práci s dokumenty. Dokumenty lze přidávat dvěma způsoby. První způsob je nahraní (upload) pomocí webového rozhraní. Systém zjistí typ dokumentu. Jestliže je dokument jiného než obrazového typu uloží ho na server a do informace o něm do databáze. Pokud je dokument obrazového typu vytvoří dva náhledy. Jeden malý pro zobrazení ve fotogalerii a jeden větší pro publikování v člancích. Druhý způsob spočívá v nahraní adresáře, obsahující dokumenty pomocí FTP na server. Vytvořil jsem velmi jednoduchý způsob vkládání většího množství dokumentů. Po výběru adresáře pro vložení, můj modul zpracuje všechny soubory v adresáři a uloží všechny informace do databáze podobně jako v prvním způsobu.

Dokumenty lze mazat. Při nahraní dokumentu se stejným jménem se dotyčný dokument přepíše. Jeden nebo více dokumentů lze přesunout do jiné kategorie. Výběr všech dokumentů v kategorii lze realizovat jedním tlačítkem.

Dále jsem implementoval výpis všech kategorií a také výpis jedné kategorie.

Jestliže je dokument fyzicky smazán ze serveru, obyčejní uživatelé si ničeho nevšimnou. Redaktor vidí i smazané dokumenty a může rozhodnout jak s nimi naložit.

## 4.4 Testování

Každý program je potřeba otestovat. Většinou se využívá dvou typů testování. Můj informační systém pro správu dokumentů a fotografií také prošel krátkým testováním. Odhalil jsem některé závažné chyby, které jsem opravil ještě před publikováním. Velké firmy platí vynakládají mnoho prostředků na testování, aby po vyjití na trhu nedošlo ke ztrátám.

**Testovací scénář** nám přesně definuje kroky, které máme otestovat. Každý scénář otestuje různé oblasti programu např. menu, funkci uploadu, platnost odkazů aj. Vytvoření testovacích scénářů zabere mnoho času, ale pro velké systémy je to jediná možnost. Testuje se na různých platformách a střídají se webové prohlížeče.

**Ad hoc testování** se používá v kombinaci s testováním pomocí testovacích scénářů. Test inženýr náhodně testuje všechny možnosti programu. Snaží se testovat krajní hodnoty systému, dělá vše proto aby nasimuloval běžného uživatele. Některé fatální chyby lze odhalit ad hoc testováním.



## 4.5 Další vývoj

Rád bych zlepšil uživatelské rozhraní tzn. využil více prvků JavaScriptu nebo Ajaxu a zjednodušil tím manipulaci s dokumenty. Rád bych také umožnil vyhledávání v dokumentech podle jména popisu, typu a obsahu. Seřazování dokumentů podle určených kritérií by uživatelé s radostí uvítali. Stahování více dokumentů zkomprimovaných do zip archívu je další funkce kterou bych chtěl v nejbližší době realizovat. Další je také větší nastavitelnost informačního systému fotogalerie. Možnost nastavit velikost náhledů. Vytvořit jednoduchý editor fotografií, který by dokázal měnit základní vlastnosti fotek jako je jas, kontrast a rozměry.

## Kapitola 5

# Závěr

Cílem této práce bylo se seznámit s možností ukládání a spravování dokumentů a fotografií na různých systémech. Navrhnout takový systém a zanalyzovat různé možnosti implementace. Nakonec systém vytvořit na MySQL databázovém systému.

Věřím že všechny požadavky na mě kladené byli splněny. Vytvořil jsem systém pro zprávu fotek a dokumentů, který je součástí/modulem již vytvořeného redakčního systému. Umí pracovat s kategoriemi. Lze přidávat nové fotky jednotlivě tak i pomocí nahrání adresáře na FTP. Umí dokumenty vkládat do článků. Zvládá přesuny dokumentů i jejich mazání a modifikaci. Samozřejmě zobrazuje dokumenty a fotografie.

Do budoucna zbývá nasadit můj modul do ostrého provozu a i na další weby. Práce byla velice zajímavá a inspirující. Dozvěděl jsem se mnoho věcí o nových internetových technologiích, prohloubil jsem si znalosti programovacího jazyka PHP a databázového systému MySQL.

Práce na projektu mě bavila a doufám že budu moci systém dále rozvíjet a vylepšovat.

# Literatura

- [1] Ministerstvo informatiky České republiky – Přístupnost webu. [online], 2007, [cit. 14. 05. 2007].  
URL <<http://www.micr.cz/verejnost/pristupnost.htm>>
- [2] Tvorba kvalitních webových stránek - Tipy pro moderní web. [online], 2007, [cit. 6. 05. 2007].  
URL <<http://tvorba-www-stranek.kvalitne.cz/>>
- [3] Gilmore, W. J.: *Velká kniha PHP 5 a MySQL*. Zoner Press, 2005, 712 s., ISBN 80-86815-20-X.
- [4] IANA: The Internet Corporation for Assigned Names and Numbers – MIME media types. [online], 2001, poslední úprava: 6. 03. 2007, [cit. 14. 05. 2007].  
URL <<http://www.iana.org/assignments/media-types/>>
- [5] Janovský, D.: Jak psát web píše Yuhů. [online], 2007, [cit. 8. 05. 2007].  
URL <<http://www.jakpsatweb.cz/>>
- [6] Kučerová, H.: UML - Unified Modeling Language. [online], 2007, vyšší odborná škola informačních služeb, Praha 4, Pacovská 350/4, poslední úprava: 31. 03. 2007, [cit. 8. 05. 2007].  
URL <<http://info.sks.cz/users/ku/PRI/UML.HTM>>
- [7] Wikipedie: Ajax — Wikipedie: Otevřená encyklopedie. [online], 2007, poslední úprava: 6. 05. 2007, [cit. 8. 05. 2007].  
URL <<http://cs.wikipedia.org/w/index.php?oldid=1406062>>
- [8] Wikipedie: Databáze — Wikipedie: Otevřená encyklopedie. [online], 2007, [Online; navštíveno 6. 05. 2007].  
URL <<http://cs.wikipedia.org/w/index.php?oldid=1449480>>
- [9] Wikipedie: Fotografie — Wikipedie: Otevřená encyklopedie. [online], 2007, poslední úprava: 1. 05. 2007, [cit. 6. 05. 2007].  
URL <<http://cs.wikipedia.org/w/index.php?oldid=1450831>>
- [10] Wikipedie: HTML — Wikipedie: Otevřená encyklopedie. [online], 2007, poslední úprava: 8. 05. 2007, [cit. 8. 05. 2007].  
URL <<http://cs.wikipedia.org/w/index.php?oldid=1475285>>
- [11] Wikipedie: Informační systém — Wikipedie: Otevřená encyklopedie. [online], 2007, [Online; navštíveno 6. 05. 2007].  
URL <<http://cs.wikipedia.org/w/index.php?oldid=1440548>>

- [12] Wikipedie: Javascript — Wikipedie: Otevřená encyklopedie. [online], 2007, poslední úprava: 6. 05. 2007, [cit. 8. 05. 2007].  
URL <<http://cs.wikipedia.org/w/index.php?oldid=1467062>>
- [13] Wikipedie: PHP — Wikipedie: Otevřená encyklopedie. [online], 2007, poslední úprava: 2. 05. 2007, [cit. 6. 05. 2007].  
URL <<http://cs.wikipedia.org/w/index.php?oldid=1454986>>
- [14] Wikipedie: UML — Wikipedie: Otevřená encyklopedie. [online], 2007, poslední úprava: 8. 04. 2007, [cit. 8. 05. 2007].  
URL <<http://cs.wikipedia.org/w/index.php?oldid=1379277>>

Seznam příloh:

**A** – Uživatelská příručka (str. 31)

**C** – Programátorská příručka (str. 32)

**D** – přiložené CD (médium vloženo na konci technické zprávy)

## Dodatek A

# Uživatelská příručka

Něco si stručně řekneme o používání informačního systému pro správu dokumentů a fotografií. Ovládání je realizováno klikáním na odkazy v menu.

Prohlednutí fotogalerie:

1. V menu klikneme na odkaz fotogalerie.
2. Vidíme všechny kategorie i s jednou vzorovou fotografií.
3. Pokud si chceme prohlédnout fotky kategorie, klikneme na ní. Buď na vzorovou fotku nebo na textový odkaz kategorie.
4. Nyní již prohlížíme fotky a v případě zájmu lze fotografii otevřít ve velkém formátu kliknutím na ní. Nebo lze uložit na PC (kliknutím pravým a uložit odkaz jako...)

Přesunutí kategorie fotek:

1. Přihlásíme se do systému
2. V levém menu klikneme na odkaz „správa kategorií“
3. Klikneme na symbol tužky
4. otevře se nám nové okno, kde můžete nastavit do které kategorie editovaná kategorie náleží. Lze tu změnit i jméno a popis kategorie.
5. změníme potřebné údaje a klikneme na tlačítko uložit.

## Dodatek B

# Programátorská příručka

K nainstalování systému musíme mít webový server (Apache) spolu se zprovozněným PHP a MySQL. Pro domácí použití lze využít např. balík XAMMP, který vše potřebné již obsahuje. Většina bezplatných a všechny placené hostingy by měli PHP a MySQL podporovat.

1. nahrajeme soubory na server
2. zeditujeme soubor databaze.php v adresáři INC
3. nastavíme potřebné údaje, nejdůležitější je nastavení serveru, uživatelského jména a hesla pro přístup do databáze. Vše potřebné by vám měl sdělit provozovatel hostingů
4. Připojíme se do MySQL databáze, např. pomocí PHPAdmina a pustíme sql skript, který vytvoří potřebné tabulky pro provoz celého systému.