

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF INTELLIGENT SYSTEMS

INFORMAČNÍ SYSTÉM PRO TVORBU KATALOGU

BAKALÁŘSKÁ PRÁCE  
BACHELOR'S THESIS

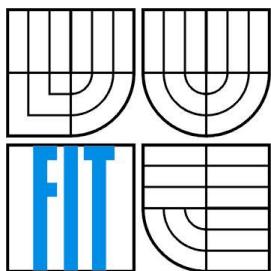
AUTOR PRÁCE  
AUTHOR

ROBERT ŠRAHOL

BRNO 2008



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF INTELLIGENT SYSTEMS

# INFORMAČNÍ SYSTÉM PRO TVORBU KATALOGU

INFORMATION SYSTEM FOR CATALOGUE CREATION

BAKALÁŘSKÁ PRÁCE  
BACHELOR'S THESIS

AUTOR PRÁCE  
AUTHOR

ROBERT ŠRAHOL

VEDOUČÍ PRÁCE  
SUPERVISOR

ING. PAVEL JURKA

BRNO 2008

## **Abstrakt**

Tento systém prezentuje obecný katalogový systém s možností nastavování různých atributů firmám. Při tvorbě systému byl kladen důraz především na optimalizaci pro vyhledávače a optimalizaci využití databáze a také na podporu více úrovní oprávnění zabezpečení. Pro implementaci byly zvolené technologie jazyka HTML pro zobrazení v internetových prohlížečích, CSS pro podporu šablon vzhledu klientské části stránek, skriptovací jazyk PHP za využití šablonovacího systému Smarty a databázový systém MySQL.

## **Klíčová slova**

Informační systém, databáze, katalog, HTML, CSS, PHP, MySQL, Smarty

## **Abstract**

This system presents universal catalogue system with option of various attributes setting for various companies. Optimization for search engines and optimization of database usage were the essential goals of this system and also multiple user security levels. HTML for displaying information in web-browsers, CSS for client part template-support of presentation, PHP scripting language using Smarty template system and MySQL database system has been used for system implementation.

## **Keywords**

Information system, database, catalogue, HTML, CSS, PHP, MySQL, Smarty

## **Citace**

Šrahol Robert: Informační systém pro tvorbu katalogu. Brno, 2008, bakalářská práce, FIT VUT v Brně.

# Informační systém pro tvorbu katalogu

## Prohlášení

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

.....  
Robert Šrahol  
23.1.2008

## Poděkování

Děkuji svému vedoucímu Ing. Pavlovi Jurkovi za pomoc s touto prací.

© Robert Šrahol, 2008.

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

# Obsah

Obsah .....	5
Úvod .....	7
1 Analýza systému .....	8
1.1 Požadavky na systém.....	8
1.1.1 Uživatelské požadavky .....	8
1.1.2 Požadavky na webhosting.....	8
1.2 Volba technologií .....	8
1.3 Návrh systému.....	9
1.3.1 Diagram užití (Use-case) .....	9
1.3.2 Konceptuální model (ER diagram) .....	10
2 Implementace .....	12
2.1 Dynamičnost webové aplikace.....	12
2.1.1 Dynamičnost vzhledu (stylů) .....	12
2.1.2 Dynamičnost rozložení ovládacích prvků stránek .....	12
2.2 Databázová vrstva .....	12
2.3 SEO Friendly URL.....	13
2.4 Bezpečnost .....	13
3 Části systému .....	14
3.1 Uživatelé.....	14
3.1.1 Hlavní administrátor .....	14
3.1.2 Ředitel firmy .....	15
3.1.3 Uživatel z firmy .....	15
3.2 Popis ovládání administrace.....	15
3.2.1 Správa šablon.....	16
3.2.2 Správa typů atributů.....	16
3.2.3 Správa firem.....	16
3.2.4 Správa uživatelů.....	16
3.2.5 Správa atributů .....	17
3.2.6 Správa položek.....	17
3.2.7 Správa menu.....	17
3.2.8 Odhlásit .....	17
3.3 Část pro návštěvníky .....	17
3.3.1 Filtrování.....	19
3.3.2 Vyhledávání .....	20

3.4	Databázový model.....	20
3.4.1	Tabulka firma.....	20
3.4.2	Tabulka menu.....	20
3.4.3	Tabulka stranky.....	20
3.4.4	Tabulka uzivatele .....	21
3.4.5	Tabulka sablona .....	21
3.4.6	Tabulka kraje .....	21
3.4.7	Tabulky polozka a priloha.....	21
3.4.8	Tabulka atribut.....	21
3.4.9	Tabulka typ atributu.....	21
3.4.10	Tabulka hodnotaatributu.....	22
3.4.11	Editace záznamů .....	22
3.5	Adresářová struktura .....	23
3.6	Nastavení práv na Unixových systémech.....	24
4	Výsledky a zhodnocení .....	25
5	Návrh na rozšíření.....	26
6	Závěr .....	27
	Literatura.....	28
	Seznam příloh .....	29
	Spustitelná verze.....	30
	Popis CSS šablony pro definici vzhledu.....	31

# Úvod

Na internetu se vyskytuje mnoho katalogových systémů, ale téměř všechny systémy jsou specializované pro určitý účel – např. to jsou katalogy automobilů, nemovitostí, realitních kanceláří a podobně. Většina katalogových systémů dnešní doby je zaměřená jen na určité účely, např. katalogy aut evidují položky jako spotřeba, typ řazení, palivo, počet míst k sezení, konstrukce vozu, barva a další položky relevantní k vozům, stejně jako počet pater, stav objektu, přítomnost výtahu, velikost, přítomnost garáže či balkónu a typ vlastnictví bytu pro byty nabízené realitními kancelářemi.

Cílem této práce bude eliminovat nutnost zaměřenosti jednotlivých informačních systémů, čehož bylo dosaženo podporou volitelného počtu atributů pro jednotlivé firmy, které mohou přidávat pouze oprávnění uživatelé. Tímto bude zajištěna univerzálnost představovaného informačního systému do míry, která by měla postačovat požadavkům všech v současné době využívaných tematicky zaměřených systémů.

Z tohoto důvodu jsem si zvolil téma univerzálního katalogového systému, založeného na principu definice vlastních atributů pro každou firmu v systému. V následujících kapitolách této práce budete seznámeni s požadavky na systém, důvody volby zvolených technologií, diagramem užití a vzájemného vztahu databázových entit (ER diagramem), informacemi o výhodách současně použité optimalizace pro vyhledávače, stejně jako optimalizace databáze včetně jejího znázornění v tabulkovém modelu.

# 1 Analýza systému

## 1.1 Požadavky na systém

Jako na každý softwarový produkt, i na tento informační systém jsou kladeny určité požadavky. Nejdůležitějšími požadavky byly především jednoduchost, rychlá navigace, ale zároveň také univerzálnost systému s podporou optimalizace pro vyhledávače.

Systém vychází ze systému správy obsahu, označovaného běžně CMS (z anglické zkratky Content Management System). Systém by měl pojímat tisíce záznamů o uživatelích všech firem definovaných v systému. Uživatelé by k systému měli přistupovat zvenčí a přes administraci spravovat svoje záznamy.

Systém by měl zobrazovat např. seznam automobilů a umožní tyto záznamy filtrovat podle vybraných kritérií, být schopen evidovat několik tisíc položek, které bude moci vkládat několik tisíc uživatelů. Tento adresář by mělo být možno filtrovat. Systém by měl kromě administračního rozhraní také umožňovat vkládat statický obsah. Dále by mělo být možné nahrávat na server obrázky a další formáty souborů (především soubory s příponou \*.pdf). Každý záznam by mělo být možno definovat N položkami, každá položka by měla mít N atributů. V systému by mělo být fulltextové vyhledávání.

Jako každý informační systém musí i tento systém splňovat i obecné požadavky zabezpečení aplikace a dat.

### 1.1.1 Uživatelské požadavky

Systém by měl mít kvalitní uživatelské rozhraní – jednoduché ovládání a příjemný vzhled.

### 1.1.2 Požadavky na webhosting

- Apache s podporou mod\_rewrite
- PHP 5.0 nebo vyšší
- MySQL 5.0 nebo vyšší

## 1.2 Volba technologií

Pro realizaci představovaného systému byl požadován skriptovací jazyk PHP ve své páté verzi kvůli své rozšířenosti tohoto skriptovacího jazyka, což je především z důvodu své multiplatformosti (PHP existuje pro systémy Windows i Unixové systémy). Jako permanentní úložiště dat jsem zvolil databázový systém MySQL ve své verzi 5.1. PHP ve verzi 5 bylo zvoleno především kvůli své vyšší



rychlosti oproti verzi 4 a lepší podpoře objektově orientovaných webových aplikací. Objektová orientace byla zvolena především kvůli čitelnosti zdrojového kódu a rychlejšímu ladění zdrojového kódu.

Databázový systém MySQL byl zvolen z důvodu své rozšířenosti využití pro webové aplikace. Pro zobrazování výsledného HTML kódu bylo použito šablonovacího systému Smarty, který sice využívá mnoho paměti, nicméně umožňuje jednoduchou editaci šablony zobrazení (např. administrativní části).

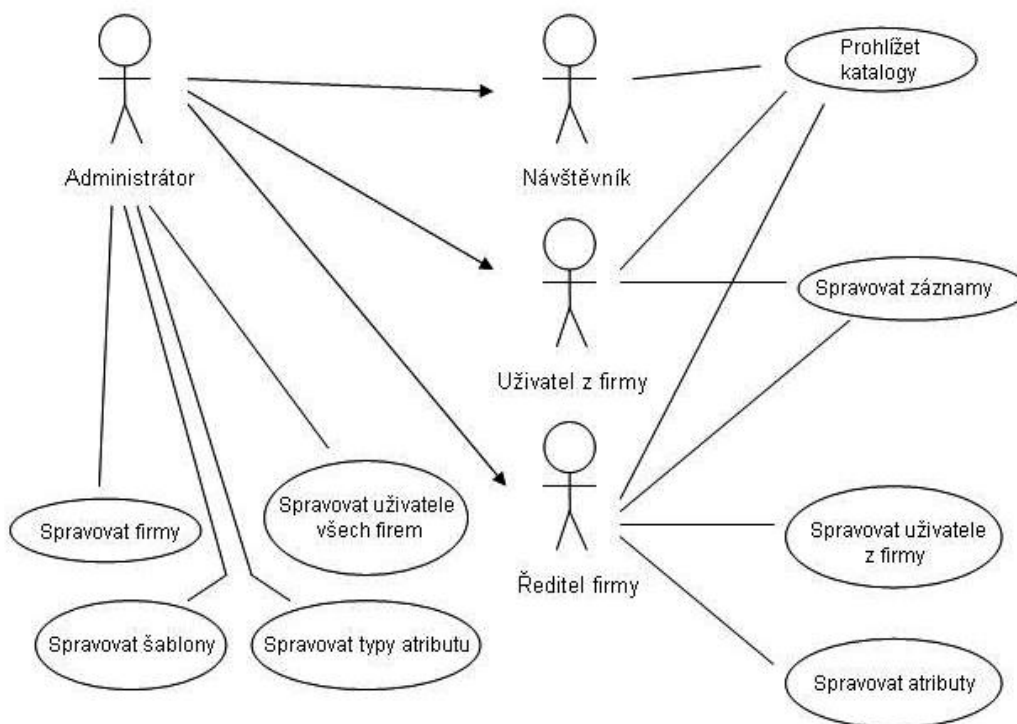
## 1.3 Návrh systému

### 1.3.1 Diagram užití (Use-case)

Diagram užití vymezuje hranice systému a zobrazuje vztah mezi systémem a okolím, do kterého patří uživatelé tohoto systému.

Systém definuje 4 typy aktorů (řazeno od nejvyššího po nejnižší):

- 1) **Hlavní administrátor** – má neomezená oprávnění, je mu umožněna kompletní správa informačního systému, musí být registrovaný v databázi a definovaný přihlašovacím jménem a heslem
- 2) **Ředitel firmy** – má oprávnění pouze k administraci firmy, do které patří, je mu umožněna správa uživatelů ze své firmy a atributů, které mohou uživatelé jeho firmy vyplňovat při přidávání položek
- 3) **Uživatel z firmy** – má pouze možnost spravovat záznamy a stránky ve své firmě, má též i možnost vyplňovat atributy, které ředitel firmy pro položky nadefinoval
- 4) **Návštěvník stránek** – jako jediný nevyžaduje tento typ uživatelského účtu registraci, může pouze prohlížet katalogy firem a nemá práva zasahovat nijak do systému, návštěvníkem stránek může být libovolná osoba, která navštívila danou adresu informačního systému

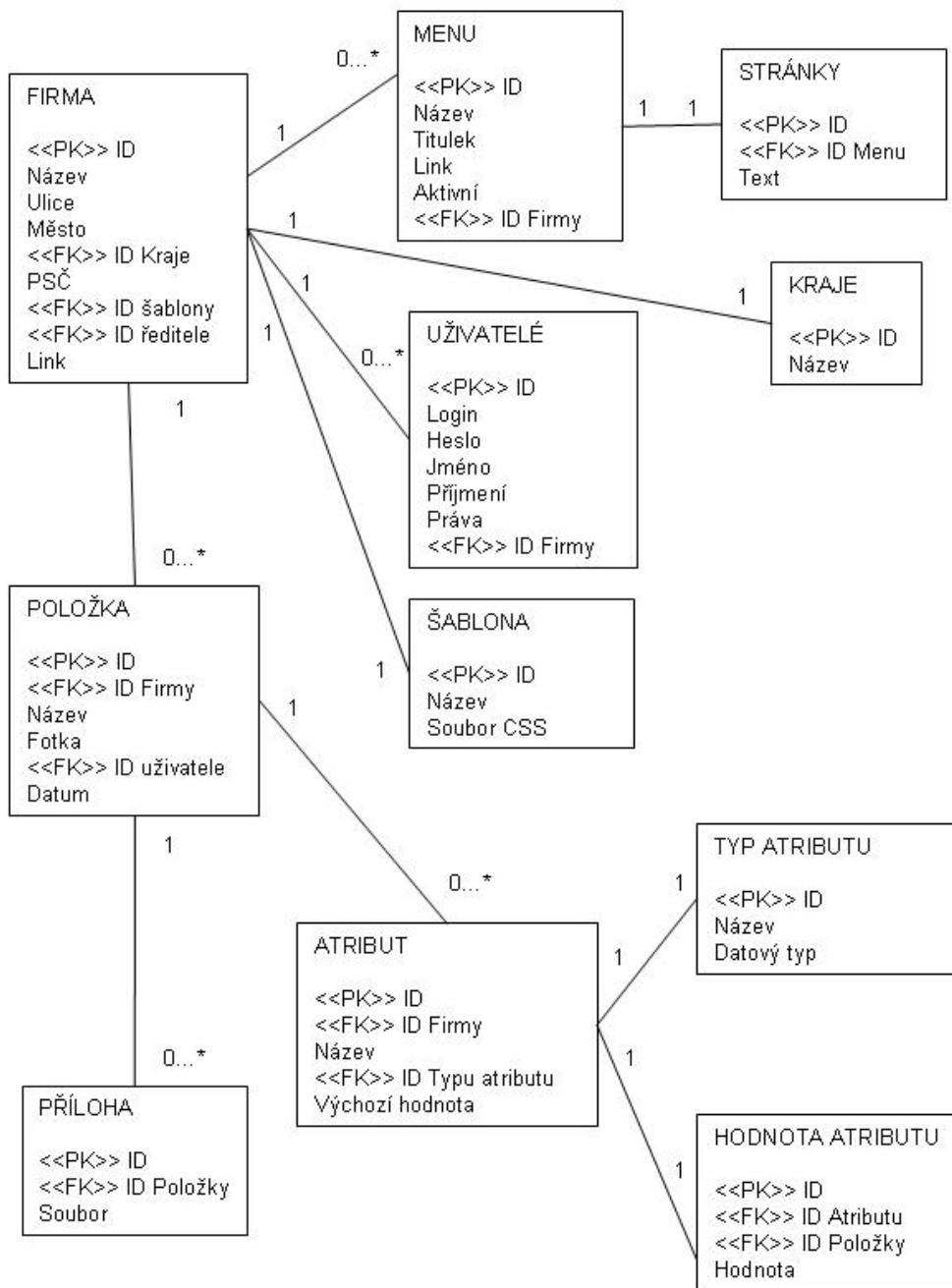


Obrázek 1: Use-case diagram

### 1.3.2 Konceptuální model (ER diagram)

Při navrhování informačního systému je třeba uvažovat požadavky na systém a analyzovat tyto požadavky tak, abychom mohli navrhnout a implementovat vyhovující konceptuální model, tzv. diagram vztahu mezi entitami, ER-diagram.

Jedná se o systém, který definuje, jaké vazby se budou v databázi nalézat, aby náš systém splňoval předpoklady dle požadavků vytvářeného informačního systému. Definice obsahuje primární klíče (v našem případě označené PK – z anglické zkratky Primary Key), stejně jako cizí klíče (označené zde FK – z anglického výrazu Foreign Key, tj. cizí klíč). Cizím klíčem definujeme identifikátor, který je napojen na jinou databázovou tabulku, z které jsou spojena data. Na jejich přítomnosti definice ve zmiňované tabulce závisí, jinak není systém dobře funkční.



Obrázek 2: ER Diagram

## 2 Implementace

Systém byl naprogramován v jazyce PHP ve verzi 5.0 za použití šablonovacího systému Smarty ve spojení s databázovým serverem MySQL 5.1 obsluhované databázovou vrstvou pro jazyk PHP. PHP verze 5.0 a vyšší bylo zvoleno z důvodu inteligentní správy objektů při využití objektového programování (OOP) a optimalizace výkonu webové aplikace. Dalším zmiňovaným požadavkem je běh na webovém serveru Apache s podporou `mod_rewrite` a povolenou definicí direktivy `RewriteEngine`. Toto nastavení nám umožňuje využívat možnosti SEO-friendly url zajišťující vyhledávačům (např. Google, Seznam a dalším) dobře čitelné adresy URL.

### 2.1 Dynamičnost webové aplikace

Samotná webová aplikace je do jisté míry dynamická, co se týče vzhledu a rozložení prvků na stránce. Toho je docíleno využitím beztabulkových layoutů pomocí CSS tříd zapouzdřených do tagů na stránce a definovaných pomocí identifikátorů a tříd kaskádových stylů (CSS).

#### 2.1.1 Dynamičnost vzhledu (stylů)

Vzhledem k nutnosti měnit vzhled stránek pro každou firmu, byla do představované webové aplikace implementována i možnost definice a užívání vlastních CSS šablon. Šablony pro jednotlivé firmy (dále jen firemní šablony) smí spravovat a jednotlivým firmám měnit pouze hlavní administrátor celého katalogového systému. Popis definice firemních šablon naleznete v příloze A.

#### 2.1.2 Dynamičnost rozložení ovládacích prvků stránek

Protože definice vzhledu pomocí kaskádových stylů (CSS) nestačí ve všech případech, byla webová aplikace naprogramována pomocí šablonovacího systému Smarty. Pro případnou změnu je tedy možné měnit i šablony pro Smarty, však by nemělo být nutno je měnit skoro vůbec, protože uživatelské šablony jsou definované pomocí tagů `div` bez využití tabulek (čímž je zajištěna dynamičnost vzhledu, popisovaná bodem výše). Rozložení ovládacích prvků stránek všech firem by se poté změnilo na základě jiných šablon systému Smarty.

## 2.2 Databázová vrstva

Pro napojení se k databázi byla využita ADOdb databázová vrstva [1] obsažená v souboru `db_mysql.inc`. Samotná třída optimalizuje a usnadňuje využití databázové platformy MySQL. Technicky se jedná o knihovnu zapouzdřující třídu `DB_Sql`, která zajišťuje spojení databáze MySQL s webovou aplikací. Tato třída byla využita pro vytvoření nové třídy `DB_Katalog` definované

v souboru `db_katalog.php` (klíčovým slovem `extends`), rozšiřující třídu o nastavení připojení k databázovému serveru a také získání identifikátoru posledního přidaného záznamu (což je využito, mimo jiné, v administraci).

## 2.3 SEO Friendly URL

SEO je anglická zkratka pro optimalizaci pro vyhledávače, plný název zní Search Engine Optimization a dle překladu se jedná o optimalizaci stránek pro vyhledávače. V praxi to znamená, že uživatel na první pohled nepozná skriptovací jazyk stránek, protože se stránky dle názvů tváří jako statické HTML. V představovaném systému jsou adresy stránek 3 typů:

- 1) <http://bp.srahol.cz> pro obecný seznam všech katalogů firem s možností vyhledávání ve všech firmách najednou při instalaci katalogového systému na webhosting o adrese [bp.srahol.cz](http://bp.srahol.cz) .
- 2) [http://bp.srahol.cz/1-nazev\\_firmy](http://bp.srahol.cz/1-nazev_firmy) pro katalog firmy `firma1`, první část (před pomlčkou) je id firmy a druhá část (za pomlčkou) je název firmy. Id firmy je zde z důvodu přesné identifikace firmy, i pokud je v systému více firem stejného názvu.
- 3) [http://bp.srahol.cz/1-nazev\\_firmy/1.html](http://bp.srahol.cz/1-nazev_firmy/1.html) pro položku s id 1 z firmy `firma1`.

Tím je zajištěna optimalizace pro vyhledávače, kterým se stránky tváří jako statické HTML a tedy nemají vyhledávače problém s cachováním těchto stránek (dynamické stránky běžně necachují).

## 2.4 Bezpečnost

Bezpečnost systému je řešena v části vyžadující přihlášení, tedy pro sekci administrace. Sekce administrace si ukládá přihlášení do SESSION proměnných prohlížeče, které jsou uloženy přímo na webovém serveru. Využitá `http session` při odhlášení uživatele zaniká a vzhledem k nevyužívání cookies pro ukládání těchto citlivých dat zde není nutnost se obávat jejich úniku.

Aby nemohl někdo k položkám přidat neplatné soubory, především nebezpečné skripty `php`, byla přípona `*.php` při jakémkoli přidávání záznamů zakázána. Fotografie musí být ve formátu `jpeg` (přípony `*.jpg` či případně `*.jpeg`) tedy zde také není možnost přidání zákeřného `php` skriptu pro získání přístupu k datům na serveru.

## 3 Části systému

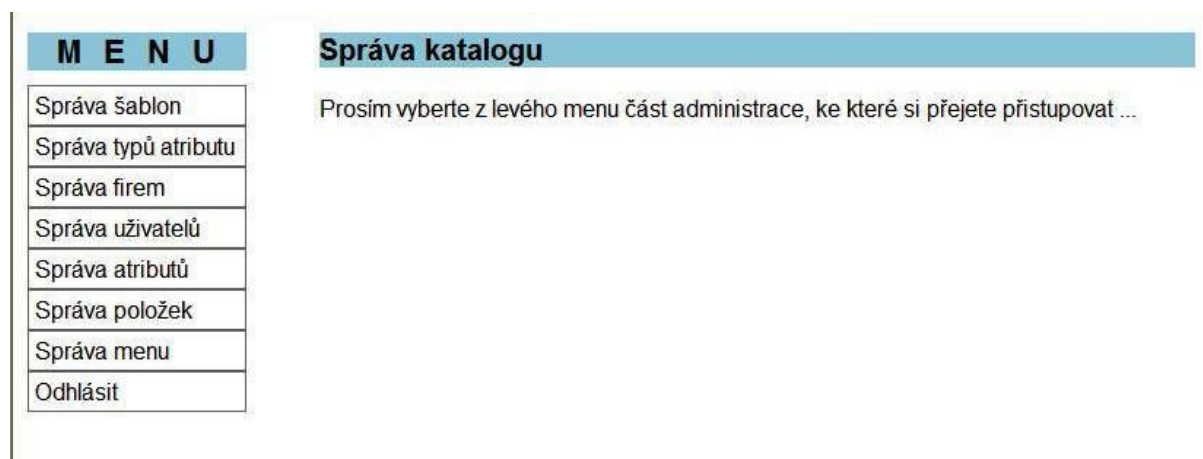
Systém se skládá ze 2 částí – části uživatelské a části administrativní. Pro přístup k uživatelské části webové aplikace není nutné přihlášení a prohlížení katalogů je možné pro všechny návštěvníky stránek. Při zadání obecné adresy instalace katalogového systému je zobrazena informace o systému v pravém panelu s možností vyhledávání v položkách, případně menu jednotlivých firem a se seznamem firem (obojí v levém panelu). Vyhledávat lze ve všech firmách najednou, a nebo je také možné si ze seznamu zvolit firmu, v níž se bude vyhledávat text daných položek nebo stránek menu. Výsledky vyhledávání budou zobrazeny v pravém panelu namísto informací o katalogovém systému, pokud bylo vyvoláno vyhledávání. Nenajdou-li se položky či menu obsahující zadaný text, je zobrazena informace, že žádné takové položky či menu v systému neexistují. Uživatel (návštěvník stránek) má právo na prohlížení katalogů v systému, nicméně pro správu je potřeba se přihlásit na administraci stránek přidáním adresy /admin za obecnou adresu katalogového systému.

### 3.1 Uživatelé

Pro přístup do administrace je nutné mít platné přihlašovací jméno (login) a heslo, jinak pokus o přihlášení do aplikace selže. Přihlásit se mohou uživatelé tří úrovní:

#### 3.1.1 Hlavní administrátor

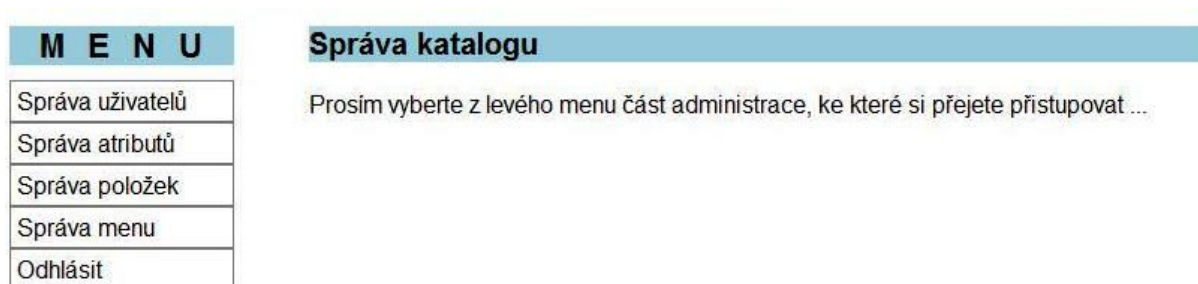
Hlavní administrátor má možnost kompletní správy katalogového systému, může spravovat šablony (nedefinované pomocí kaskádových stylů CSS), spravovat informace o firmách, jejichž katalogy systém obsahuje, definovat typy atributů, které mohou firmy využívat pro jednotlivé atributy, spravovat uživatele jednotlivých firem, stejně jako atributy, položky a stránky v menu daných firem.



Obrázek 3: Administrativní část – hlavní administrátor

### 3.1.2 Ředitel firmy

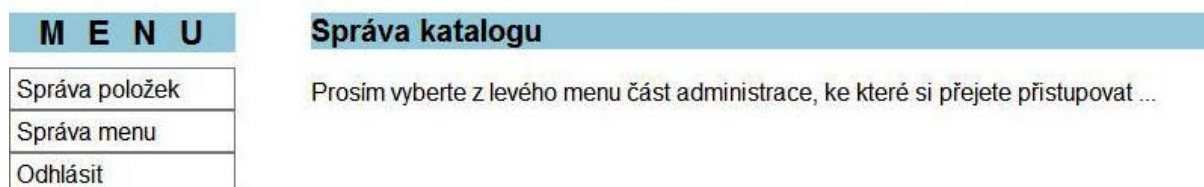
Ředitel firmy má možnost správy uživatelů ve své firmě (přidaný uživatel má vždy práva uživatele z firmy, ředitel nemá právo nastavit někoho jiného jako ředitele, to musí udělat hlavní administrátor), dále má právo spravovat atributy své firmy, např. přidat atribut Cena ke zboží či podobně. Typy atributu má právo zvolit jen ty, které nadefinoval hlavní administrátor systému. Dále má pochopitelně možnost správy položek a textových stránek své firmy.



Obrázek 4: Administrativní část – ředitel z firmy

### 3.1.3 Uživatel z firmy

Uživatel z firmy má nejnižší práva z celé administrace, může jen spravovat položky a textové stránky firmy, do níž spadá.



Obrázek 5: Administrativní část – uživatel z firmy

## 3.2 Popis ovládání administrace

Po přihlášení do administrace s právy hlavního administrátora můžete v levém menu vidět odkazy na správu šablon, typů atributů, firem, uživatelů, atributů, položek a menu.

### **3.2.1 Správa šablon**

Ve správě šablon je možné přidávat, editovat a mazat šablony kaskádových stylů CSS. Při přidávání jsou potřeba jen 2 činnosti – zadat název šablony a vybrat soubor CSS k nahrání na server. Nahrávaný soubor musí mít příponu CSS a musí pro validní zobrazení v uživatelské části obsahovat definice dle popisu v příloze A.

Požadovaná práva: Hlavní administrátor

### **3.2.2 Správa typů atributů**

Hlavní administrátor má také možnost spravovat jednotlivé typy atributů. Při zadávání nového typu atributu je možné jen zvolit název typu atributu k zobrazení a jeho datový typ. Datové typy jsou přednastavené a mohou být jen typu řetězce, textu, celého čísla, desetinného čísla, binární hodnoty (nabývající hodnot jen 0 nebo 1, tj. z programátorského hlediska typ BOOL či BOOLEAN) a data/času. Prakticky uživatel systému administrace nebo návštěvník stránek neuvidí rozdíl mezi jednotlivými datovými typy, nicméně typy jsou definované různé z důvodu úspory místa v databázi. Je tedy nutné, aby hlavní administrátor měl potřebné technické znalosti, pro srovnání uvádím níže i tabulku využití databáze (co se týče využitého místa) pro jednotlivé případy vstupů.

Požadovaná práva: Hlavní administrátor

### **3.2.3 Správa firem**

Hlavní administrátor má právo spravovat firmy využívající jeho katalogový systém, nacházející se pod jeho doménou. Pro definici nové firmy do systému je nutné zadat název firmy, sídlo firmy (ulice, město, kraj, psč), vyplnit přihlášení a jméno/příjmení ředitele firmy (to ředitel využije pro přístup do administrace své firmy) a také zvolit šablonu pro webovou prezentaci nově přidávané firmy. Šablony je možné přidávat pomocí odkazu Správa šablon, viz bod 4.1.1.

Požadovaná práva: Hlavní administrátor

### **3.2.4 Správa uživatelů**

Ve správě uživatelů je možné spravovat jednotlivé uživatele, v seznamu uživatelů lze vidět jméno a příjmení uživatele, stejně jako je v závorce uvedena úroveň oprávnění. Při přidávání nového uživatele hlavním administrátorem je zde i možnost zvolit firmu a práva nově přidávaného uživatele, včetně možnosti nastavit daného člověka jako dalšího hlavního administrátora. Pro přidání je nutné vyplnit login a heslo daného uživatele a také samozřejmě jeho/její jméno a příjmení.



Požadovaná práva: Hlavní administrátor, ředitel firmy

### **3.2.5 Správa atributů**

Ředitel firmy i hlavní administrátor mohou spravovat atributy. Atribut je definován názvem atributu, datovým typem z typu atributů a výchozí hodnotou, která se přednastaví při přidávání nových položek.

Požadovaná práva: Hlavní administrátor, ředitel firmy

### **3.2.6 Správa položek**

Umožňuje přidávat, editovat a mazat položky. Ke každé položce jsou automaticky načtené všechny atributy definované pro danou firmu s možností jejich vyplnění a editace. Každá položka je definována názvem, fotkou a neomezeným počtem příloh. U editace položky a v databázi je obsažena informace, kdo a kdy danou položku přidal do systému.

Požadovaná práva: Hlavní administrátor, ředitel firmy, uživatel z firmy

### **3.2.7 Správa menu**

Umožňuje spravovat stránky v menu firmy. Každá stránka je definovaná názvem (využitým pro text odkazu), titulkem (využitým pro pruh titulku v prohlížeči) a textem. Navíc je však i možné definovat, zda je odkaz aktivní či nikoli, tedy, zda se má zobrazovat v menu.

Požadovaná práva: Hlavní administrátor, ředitel firmy, uživatel z firmy

### **3.2.8 Odhlásit**

Umožňuje se odhlásit z administrativní části katalogového systému.

## **3.3 Část pro návštěvníky**

Při navštívení hlavní stránky katalogu se zobrazí obecná informace o katalogovém systému s možností vyhledávání ve všech firmách či zvolené firmě a také seznamem firem spadajících do tohoto katalogu.

Po rozkliknutí odkazu firmy se načtou stránky zvolené firmy za využití šablony definované pro vzhled stránek firmy. V tomto kroku je možné vyhledávat fulltextově v rámci firmy v jejích položkách a také si nechat zobrazovat textové stránky firmy (např. podmínky prohlížení). Také je zde možnost filtrování položek dle data přidání a definovaných atributů (včetně zadání rozmezí od – do, kde je to aplikovatelné).

System rozlišuje 3 základní části klientské části: hlavní stránka katalogu, stránky firem, stránky položek dané firmy. Pro ukázkou přikládám screenshoty všech částí:

## 1) Hlavní stránka katalogu

### Katalog firem

The screenshot shows a web interface for a company catalog. On the left, there is a search bar with the text 'Vyhledávání' and a dropdown menu set to '- ve všech firmách -'. Below the search bar is a 'Vyhledat' button. Underneath is a 'Firmy v katalogu' section with a link to 'Název firmy' and the text 'General Automobil'. The main content area is titled 'Katalog firem' and contains the following text: 'Toto je hlavní stránka obecného informačního systému pro tvorbu katalogů. Pro pokračování prosím vyberte firmu z levého menu.', 'Do sekce administrace prosím přejděte kliknutím zde...', and 'Pro výběr firmy pokračujte níže:'. Below this text is a link for 'Název firmy' and the text 'General Automobil'.

Obrázek 6: Hlavní stránka katalogu

## 2) Stránka firmy, využita byla výchozí šablona

### General Automobil

The screenshot shows a company page for 'General Automobil'. On the left, there is a search bar with the text 'Vyhledávání' and a 'Vyhledat' button. Below the search bar is a 'Menu' section with links to 'Hlavní stránka', 'Hlavní stránka General Automobil', and 'Filtrování'. The main content area is titled 'Katalog firmy' and contains two car listings. The first listing is for a red car with the name 'navez', price 'Cena: 200000', and date 'Datum přidání: 2007-11-11 16:49:54'. The second listing is for a red car with the name 'Opel', price 'Cena: 250000', and date 'Datum přidání: 2007-11-11 16:50:26'.

Obrázek 7: Stránka firmy

### 3) Stránka detailu položky, pro firmu s výchozí šablonou



Obrázek 8: Stránka detailu položky

#### 3.3.1 Filtrování

Katalogový systém umožňuje filtrování podle zvolených atributů včetně možnosti zadání rozsahu od-do pro číselný datový typ a typ data a času. Dotaz je spojován pomocí operací JOIN, který určuje jednotlivé informace, dle kterých se má filtrovat. Byl koncipován s ohledem na funkčnost a efektivitu filtrování. Filtrovat lze víceúrovňově podle jednotlivých atributů, navíc pro číselné a datové typy v rozsahu od-do.



Obrázek 9: Ukázka filtrování

### 3.3.2 Vyhledávání

System katalogu umožňuje též i fulltextové vyhledávání ve všech položkách a atributech. Vyhledávací dotaz na databázi je zkoncipovaný přesně pro vyhledávání ve všech attributech položek v systému, stejně tak i v názvech položek. Prohledávány jsou všechny typy atributů, tj. jak hodnoty typu *string*, text, tak i hodnoty typů *int*, *float* a *bool* či *datetime*. Vyhledávání je fulltextové, tedy je možné vyhledávat ve všech zmiňovaných položkách části textu a není nutné název či hodnotu atributu zadat přesně. U vyhledaných položek je zobrazena firma, z které je daná položka, pokud bylo vyhledáváno ve všech firmách nebo není zobrazena firma, bylo-li vyhledáváno jen v rámci jedné firmy.

## 3.4 Databázový model

Databáze tohoto systému se skládá z 11 tabulek, jedná se o tabulky *firma*, *menu*, *stranky*, *uzivatele*, *sablona*, *kraje*, *polozka*, *priloha*, *atribut*, *typatributu* a *hodnotaatributu*. Každá tabulka má primární klíč *Id*, který je jednoznačným autoinkrementálním identifikátorem.

### 3.4.1 Tabulka firma

Pro tabulku *firma* evidujeme název, ulici, město, *Id kraje*, *PSČ*, *Id šablony*, *Id ředitele* a *link*. *Link* je generován z názvu firmy pomocí osekání českých znaků pro *mod\_rewrite*, jelikož v URL není povoleno mít české znaky.

### 3.4.2 Tabulka menu

Pro tabulku *menu* *Id menu*, název, titulek, *link*, *aktivni*, *id\_firmy*. *Link* je generován z titulku a jedná se o hodnotu, která definuje adresu pro *mod\_rewrite*. Každá firma může mít více menu a tedy je zde sloupec *id\_firmy*. Menu může být deaktivováno pomocí nastavení hodnoty *aktivni* na 0.

### 3.4.3 Tabulka stranky

Tabulka *stranky* je definovaná pouze *id*, *id\_menu* (které odpovídá *id* z tabulky *menu*) a *textem*. Důvodem proč jsou tabulky *menu* a *stranky* oddělené je možnost pozdějšího rozšíření webové aplikace, např. o tabulku *skripty*, která bude definována pomocí *id*, *id\_menu* a souboru *skriptu*, který se bude inkudovat (namísto běžného vypsání textu, jako v případě tabulky *stranky*). Bylo by však nutné mírně editovat databázi a rozšířit tabulku *menu* o sloupec *typ*.

### 3.4.4 Tabulka uzivatele

Tabulka *uzivatele* definuje uživatele systému katalogu pomocí jejich id, loginu, hesla, jména, příjmení, přístupových práv (0 = hlavní administrátor, 1 = ředitel či 2 = uživatel z firmy) a identifikace firmy, z které je uživatel (či 0 pro žádnou firmu, jedná se o hlavního administrátora).

### 3.4.5 Tabulka sablona

Tabulka *sablona* určuje seznam CSS šablon dostupných pro využití jednotlivými firmami a je definována pouze id, názvem a souborem s šablonou css.

### 3.4.6 Tabulka kraje

Tabulka *kraje* je jen pomocnou tabulkou pro případné rozšíření počtu krajů, kde můžou firmy působit. Jinak by to nemuselo být definováno v databázi a mohlo to být definované přímo ve zdrojovém HTML kódu stránek (šablonách Smarty).

### 3.4.7 Tabulky polozka a priloha

Tabulka *polozka* obsahuje položky pro danou firmu, je definována primárním klíčem Id, identifikátorem firmy, kam spadá, názvem, fotkou, identifikátorem uživatele, který položku přidal a datem přidání položky. Každá položka může být definována libovolným počtem příloh (dokumenty DOC, PDF či obrázky atd.) pomocí tabulky *priloha*, která je definována id, id\_polozky, k níž se řadí a souborem s přílohou. Tyto soubory se ukládají do podadresáře *\_prilohy* na serveru.

### 3.4.8 Tabulka atribut

Tabulka *atribut* je definována id, id\_firmy (každá firma může mít jiné atributy), názvem, id\_typuatributu (viz dále) a výchozími hodnotami pro jednotlivé podporované typy atributu (vychozihodnota\_\*; \* = [string, text, int, float, bool, datetime]).

### 3.4.9 Tabulka typ atributu

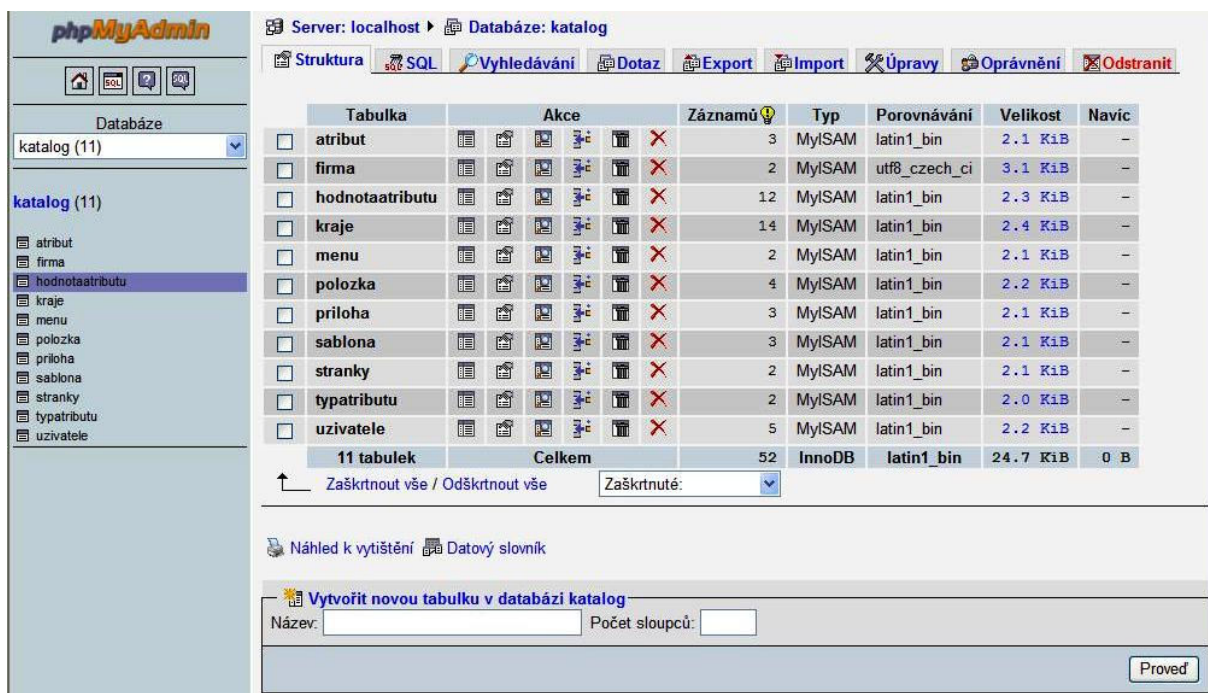
Tabulka *typatributu* udává typy atributu, které může ředitel firmy využít při definici atributů, obsahuje sloupce id, název a datovýtyp. Datový typ je číselná hodnota datového typu ze seznamu výše (*string, text, int, float, bool, datetime*).

### 3.4.10 Tabulka hodnotaatributu

Poslední tabulkou systému je *hodnotaatributu*, která určuje hodnotu daného atributu dle svého datového typu. 6 datových typů je definováno z důvodu optimalizace a úspory místa v databázi (viz testování na 10.000 záznamech výše).

### 3.4.11 Editace záznamů

Pro editaci záznamů přímo v MySQL databázovém systému je možné zvolit PHPMyAdmin, který je Open Source volně ke stažení na internetu, jedná se o velmi efektivní systém správy dat v databázi MySQL, nicméně pro import dat z příloženého SQL skriptu katalog.sql je možné též využít i konzolovou verzi MySQL klienta. Vzhledem ke své rozšířenosti a dostupnosti však na většině hostingů bývá právě databáze MySQL společně i s tímto nejrozšířenějším administrativním rozhraním, phpMyAdminem.



Obrázek 10: Ukázka phpMyAdmina

## 3.5 Adresářová struktura

Adresářová struktura je následující:

catalog / \_foto – obsahuje fotografie položek

- složka obsahuje ukázkové fotky

\_prilohy – obsahuje přílohy k jednotlivým položkám

- složka obsahuje ukázkové přílohy

admin/index.php – obsahuje skript pro administrativní část webové aplikace

classes – obsahuje nejdůležitější soubory, třídy pro zobrazování či správu dat

- AdminSmarty.php – obsahuje definici administrativního nastavení pro Smarty
- UserSmarty.php – obsahuje definici nastavení pro šablony Smarty pro uživatelskou (návštěvnickou) část stránek
- units\_admin.php – definuje inkudování skriptů pro administraci
- units\_user.php – definuje inkudování skriptů pro uživatelskou část
- db\_mysql.inc – definuje databázovou vrstvu
- db\_catalog.php – dědí databázovou vrstvu a definuje připojení k databázi
- atributy.php – definuje správu a zobrazování atributů
- firma.php - definuje správu a zobrazování firem
- login.php – obsahuje obsluhu přihlášení
- menu.php - definuje správu a zobrazování menu
- polozky.php - definuje správu a zobrazování položek
- priloha.php - definuje správu a zobrazování příloh
- sablony.php - definuje správu a zobrazování šablon
- uzivatel.php - definuje správu a zobrazování uživatelů

libs – obsahuje knihovny pro šablonovací systém Smarty, nezasahovat!

sablony – obsahuje CSS šablony pro využití v klientských stránkách jednotlivých firem. Pro přidávání šablon přes administraci hlavním administrátorem je nutné v Unixových systémech nastavit práva zápisu (běžně 777), elementy a identifikátory šablony budou popsány v další kapitole

templates – obsahuje šablony Smarty, běžně by nemělo být nutné nezasahovat. Pro změnu vzhladu by mělo postačit měnit pouze CSS šablony.

templates\_c – zkompileované šablony Smarty, nutná práva zápisu (777 v Unixových systémech) při přidání či změně Smarty šablony v templates

- .htaccess – přepisovací pravidla pro mod\_rewrite (SEO), nutné změnit relativní cestu v rámci serveru na cestu kořenového adresáře katalogového systému (kde se nachází index.php).

- admin.css – css (styly) pro administrativní část katalogového systému, běžně by nemělo být potřeba měnit, leda v případě potřeby změnit vzhled administrace

- default.css – výchozí šablona css stylů při neexistenci požadované šablony zvolené v administraci pro danou firmu

- index.php – hlavní jádro katalogového systému, obsahuje definice a inkluzi jednotlivých tříd a volá šablony Smarty, dle kterých dochází ke zobrazení klientské části prezentace či seznamu firem a obecných informacích o systému katalogu.

- katalog.sql – obsahuje dump MySQL databáze včetně ukázkových položek a ukázkových firem, výchozí přihlášení administrátora je login *admin* s heslem *admin* .

- normal.css – css šablona využitá při zobrazení hlavní stránky katalogového systému s obecnými informacemi o systému

## 3.6 Nastavení práv na Unixových systémech

Systém ukládá přílohy a fotky firem do podtržítkem začínajících adresářů. Jedná se o první adresáře v zobrazení při řazení podle jména, jsou to adresáře *\_foto* pro fotografie a *\_prilohy* pro přílohy. Těmto adresářům je nezbytně nutné na Unixových hostinzích nastavit všem práva pro zápis, jinak nedojde k zápisu souborů a položky v systému nebudou obsahovat přílohy či fotografie. Pro možnost přidávání šablon je též potřeba tato práva nastavit i pro adresář *sablony*.



## 4 Výsledky a zhodnocení

Z důvodu optimalizace využití databáze bylo využito celkem 6ti různých datových typů. Pro všechny typy bylo prováděno testování na 10.000 záznamech s hodnotami uvedenými v tabulce níže pro srovnání.

<b><u>Typ / vstup</u></b>	<i>String</i> (ahoj)	<i>Text (ahoj)</i>	<i>Int (1000)</i>	<i>Float</i> (1000.52)	<i>Bool (1)</i>	<i>Datetime</i> (23.11.2007 15:00:00)
<i>String</i>	196,3 kB	196,3 kB	94,38 kB	94,38 kB	196,3 kB	235,4 kB
<i>Text</i>	196,3 kB	196,3 kB	94,38 kB	94,38 kB	196,3 kB	274,4 kB
<i>Int</i>	-	-	69,4 kB	-	69,4 kB	-
<i>Float</i>	-	-	69,4 kB	69,4 kB	69,4 kB	-
<i>Bool</i>	-	-	-	-	69,4 kB	-
<i>Datetime</i>	-	-	-	-	-	88,9 kB

**Tabulka 1: Srovnání datových typů**

Z tabulky výše je zřejmé, že:

Při ukládání záznamů binární hodnoty (0/1) byla při uložení dat jako typ string či text velikost databáze o neuvěřitelných 282% větší oproti typu BOOL a do formátu datum/čas nebylo možné tuto hodnotu uložit.

Číslo 1000 nebylo z pochopitelných důvodů možné uložit do typu binární hodnoty (která smí nabývat jen hodnot 0 a 1, resp. TRUE a FALSE), ani do typu datum/čas. Nicméně uložení tohoto čísla do číselných datových typů bylo o 36% výhodnější než uložení do řetězcových typů.

Desetinné číslo 1000.52 nebylo možné uložit ani jako BOOL hodnotu, ani jako celé číslo (které se samo zaokrouhlilo, což není v pořádku) a ani jako typ datum/čas a velikost 10.000 záznamů tohoto čísla jako desetinného čísla (float) byla též o 36% menší než u řetězcových typů.

Datum 23.11.2007 v čase 15:00:00 bylo možné uložit jen do datových typů datum/čas a řetězcových typů, při čemž uložení do typu datum/čas bylo o 265% výhodnější než do typu běžného řetězce (varchar(255)) a o 308% výhodnější než uložení do typu text.

Pro větší počet znaků řetězce než je maximální definovaná hodnota (255 znaků) lze využít jen typ text, jelikož se do běžného řetězce, typu varchar(255), nevejde. Je tedy nutné, aby měl hlavní administrátor požadované technické znalosti ke správné definici názvů typů atributů (aby to i méně technicky zdatný ředitel pochopil)

## 5 Návrh na rozšíření

Informační systém byl navržen otevřeně, tedy je možnost jej stále rozšiřovat. Pro realizaci systému však nebylo tolik času, aby bylo možné všechny myšlenky realizovat, a tedy jsou uvedené zde jen jako návrh k rozšíření. Především by bylo dobré rozšíření přidáním možnosti cachování stránek.

Možnost cachování se hodí především, pokud máte mnoho klientů, kteří přistupují ke stejným stránkám, čímž se velmi vytěžuje samotný server Apache a server databázového systému MySQL. Z tohoto důvodu byla vymyšlena možnost cachování stránek, která umožňuje často zobrazované stránky uložit do vyrovnávací paměti, paměti cache. Jakmile uživatel přistupuje ke stránce, která již byla pár sekund či minut zpět zobrazena jinému klientovi, nemusí se dotazovat databázový server na ta samá data, ale může při prvním načtení těchto dat uložit na disk obsah dané stránky a dalším klientům pouze načíst tyto stránky, aniž by bylo nutné se dotazovat opět databáze na tato data. Snížíme tímto především režii mezi databázovým serverem a webovým serverem, a navíc databázový server nemá žádnou zátěž v tento okamžik, protože není vůbec dotazován a tedy nezpracovává vůbec žádné dotazy.

Cachování je, mimo jiné, také použito především na proxy serverech. Proxy server umožňuje jiným počítačům přístup do sítě, a proto bývá většinou ve společnostech velmi vytížen, tedy aby nemusel zpracovávat všechny dotazy klientů, má většinou povolené cachování stránek a jakmile si stránku vyžádá jeden uživatel napojený přes proxy server, je stránka po nějakou dobu uložena a dalším klientům je nabízena na proxy serveru lokálně uložená kopie.

Hlavní podstatou cachování je snížení využití operační paměti, stejně jako režie mezi přístupy k databázovým serverům a webservere, ale také i výrazné omezení zátěže databázového serveru z důvodu ukládání na disk webservere (či proxy serveru) a nabízení pouze kopie dané stránky. Z pochopitelných důvodů je cachování někdy nežádoucí, např. pro stránky, které jsou on-line statistikami, např. sledování aktuálního stavu počasí v reálném čase a podobně. Z tohoto důvodu je možné cachování explicitně zakázat pomocí zaslání speciálních http hlaviček do prohlížeče. Na tyto hlavičky reaguje případný proxy server tím, že načte stránky znovu a předá klientovi originální, právě načtené stránky, nikoli duplikát z disku. Nicméně pro případ klientské části katalogového systému s platností cache po dobu cca. 1 hodiny by tento problém neměl nastat.

U administrativní části je situace jiná, protože by se nejednalo o pasivní zobrazování dat, ale přímo modifikace dat, tedy je nutné načítat přesně aktuální data z databáze, která editujeme, stejně jako skutečně provádět aktualizaci dat a ne pouze zobrazit informaci, že data byla aktualizována, nicméně bez jejich aktualizace. [10]

## 6 Závěr

V této dokumentaci byl popsán návrh a implementace informačního systému pro tvorbu katalogu.

Cílem této bakalářské práce bylo vytvoření univerzálního informačního systému pro tvorbu a správu katalogu, eliminovat nutnost zaměřenosti jednotlivých informačních systémů, čehož bylo dosaženo podporou volitelného počtu atributů pro jednotlivé firmy, které mohou přidávat pouze oprávnění uživatelé. Tímto byla zajištěna univerzálnost představovaného informačního systému do míry, která by měla postačovat požadavkům všech v současné době využívaných tematicky zaměřených systémů.

Předkládaný systém umožní registrovaným uživatelům snadno vytvářet katalogy. Hlavní administrátor má přístup k administraci celého systému a může tak přidávat nové uživatele.

S ohledem na to, že na systému by ještě mohly být upraveny, případně rozšířeny jednotlivé funkce, které jsem uvedl výše, myslím si, že i tak v dostatečné míře odpovídá cíli a dá se použít jako univerzální katalogový systém.

Díky této bakalářské práci jsem si prohloubil znalosti z oblasti návrhu a implementace informačních systémů, z oblasti MySQL a programování PHP.

Závěrem si jako autor této aplikace si přeji, ať najde co největší uplatnění.

# Literatura

- [1] ADOdb Database Abstraction Library for PHP (and Python) [online]. 2004, [cit. 2007–10-1]. Dostupný z WWW: <<http://adodb.sourceforge.net>>.
- [2] Hruška, T., Křivka, Z.: Informační systémy (IIS, PIS) studijní opora, Pojem informačního systému, data, procesy, transakce. 2006.
- [3] Knihovna CSS -- Interval.cz [online]. 2007 , poslední aktualizace (není k dispozici) [cit. 2007–10-05]. Dostupný z WWW: <<http://css.interval.cz>>.
- [4] MySQL AB: MySQL 5.1 Reference Manual, 2007 [online], [cit. 2007-11-29]. Dostupný z WWW: <<http://dev.mysql.com/doc/refman/5.1/en/index.html>>.
- [5] PHP: Manuál PHP – Manual [online], poslední aktualizace 2007-03-24 [cit. 2007-11-29]. Dostupný z WWW: <<http://www.php.net/manual/cs>>.
- [6] Programujeme PHP profesionálně. 2. aktualizované vydání Brno : Computer Press, 2004. 656 s. ISBN 80-7226-310-2.
- [7] Jak psát web, návod na html stránky [online]. 2007, poslední aktualizace 7. listopadu 2007 [cit. 2008–01-05]. Dostupný z WWW: <<http://www.jakpsatweb.cz>>.
- [8] Smarty v PHP | Úvod [online]. 2007, poslední aktualizace 9. ledna 2006 [cit. 2007–12-08]. Dostupný z WWW: <<http://smarty.ronnieweb.net/index.php>>.
- [9] Serial PHP [online]. 2007, [cit. 2007–10-1]. Dostupný z WWW: <<http://www.linuxsoft.cz/php>>.
- [10] Cache [online], [cit. 2008–01-05]. Dostupný z WWW: <<http://en.wikipedia.org/wiki/Cache>>.

# Seznam příloh

Příloha 1. CD-R se zdrojovými texty a návodem k instalaci

Příloha 2. Spustitelná verze

Příloha 3. Popis CSS šablony pro definici vzhledu

# Spustitelná verze

System je možné vyzkoušet na adrese [www.bp.srahol.cz](http://www.bp.srahol.cz)

Administrační část na adrese [www.bp.srahol.cz/admin](http://www.bp.srahol.cz/admin)

Hlavní administrátor:

Login: admin

Heslo: admin

Ředitel z firmy:

Login: opel

Heslo: opel

Uživatel z firmy:

Login: user

Heslo: user

# Popis CSS šablony pro definici vzhledu

V systému je možné definovat vlastní css šablony, které můžete jako hlavní administrátor systému přidávat. K tomuto úkonu musíte mít minimálně znalosti o tvorbě CSS šablon (kaskádových stylů) a nyní si popíšeme jednotlivé elementy a identifikátory potřebné pro šablony.

Doporučuji nadefinovat i nepovinná nastavení stránka, jako např. nastavení stylů tagu <body> (obecné texty stránek) či odkazů, tedy tagu <a>.

Je nutné definovat identifikátor (objekt začínající křížkem v CSS) *celek*, který zajišťuje celkové rozložení stránek. Dle něj je možné nastavit pro zajímavost i maximální šířku stránek, např. pro optimalizaci na určité rozlišení apod. Dále je nutné definovat styl nadpisu v identifikátoru *nadpis*, což zajišťuje styl zobrazení nadpisu firmy.

Vzhledem k tomu, že byl využit beztabulkový layout, je nutné definovat levý a pravý sloupec s definovaným obtékáním (float), v pravém sloupci je ve výchozím nastavení menu včetně vyhledávání, případného menu textových stránek a menu položek a v pravém obsah stránek samotných. Obsah pravého sloupce závisí na zvolené stránce, kde se může objevovat buď samotný katalog nebo textová stránka s informacemi, např. podmínky prohlížení stránek (z mnoha stránek je zakázáno stahovat obrázky kvůli možnosti jejich zneužití).

Vyhledávání je možné nastylovat pomocí identifikátoru *vyhledavac* podřazeného identifikátoru levého sloupce *levysloupec*. Zde je možné např. zakázat zobrazování vyhledávače pomocí nastavení vlastnosti display na hodnotu none, či změnit šířku vyhledávacího pole. Nadpis samotného vyhledávání je možné změnit v elementu h3 podřazeném identifikátoru *vyhledavac*, vzhled vyhledávacího formuláře se řídí elementem form pro stejný identifikátor, vstupní pole textu elementem input a třídou *.submit* pro nastavení vstupního tlačítka vyhledávání. Stejně styly jsou využité i pro menu.

Pro zobrazování levého menu položek se používá css identifikátoru *polozky*, který pod sebou definuje pro nadpis menu položek styl nadpisu typu 3 (*h3*) a pro názvy položek se používá identifikátoru *nazvy-polozek*. Dále je nutné definovat zleva obtékající element s názvem *obsah*, též typu identifikátoru, pro definici vzhledu obsahové části. Zde je definován pro nadpisy styl nadpisu typu 2, tedy element *h2* a pro zobrazení jednotlivých položek css třída *.polozka* definující další, podřazené elementy, přesně nadpis třetího typu (úrovně), *h3*, a třídu pro obrázek v *.img*, třídu pro text v *.text*. Při zobrazení detailu položky se užívá stylu *.polozka\_detail* definující podřazené styly pro obrázek, *.img*, a text, třídu *.text*. Posledními elementy nutnými definovat jsou elementy pro možnost filtrování v identifikátoru *filtr*. Zde je definovaný název atributu v třídě *.nazev*, hodnota prvního řádu (mj. pro textové pole jediná využitá) v *.hodnota1*, hodnota druhého řádu (využitá jako druhá vstupní

položka pro rozmezí dat typu `datetime` či číselných typů) v `.hodnota2`, zalomení na další řádek v identifikátoru `.zalomeni` a poslední také třída `.submit` pro odesílání dat.

Při takovéto definici css šablony vzhledu by se vše mělo zobrazit v pořádku dle nadefinovaných css stylů. Pro ukázkou můžete shlédnout ukázkové šablony v adresáři sablony prezentovaného systému katalogu.