

**UNIVERZITA KOMENSKÉHO V BRATISLAVE**  
**FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY**

**Organizovanie literatúry**

BAKALÁRSKA PRÁCA

Študijný program: Informatika

Študijný odbor: 9.2.1. Informatika

Školiace pracovisko: Katedra Informatiky

Školiteľ: RNDr. Jana Katreniaková, PhD

**Bratislava 2010**

**Marek Lüley**

## **Pod'akovanie**

Chcem sa pod'akovať svojmu školiteľovi RNDr. Jane Katreniakovej, PhD. za odbornú a ochotnú pomoc pri vypracovaní tejto práce. V neposlednom rade ďakujem aj všetkým, ktorí mi vytvorili priestor a podporovali ma pri vypracovaní práce. Rodine, priateľom...

Ďakujem

## **Čestné prehlásenie**

Čestne vyhlasujem, že zadanie bakalárskej práce som vypracoval pod odborným vedením svojej školiteľky RNDr. Jany Katreniakovej, PhD. s použitím citovaných zdrojov.

V Bratislave dňa:

Marek Lüley

## **Abstrakt**

**Názov práce:** Organizácia literatúry

**Autor:** Marek Lüley

**Katedra:** Katedra informatiky

**Fakulta:** Fakulta matematiky, fyziky a informatiky

**Univerzita:** Univerzita Komenského v Bratislave

**Vedúci práce:** RNDr. Jana Katreniaková, PhD

**Rozsah:** 43

**Miesto, rok:** Bratislava, 2010

Táto bakalárska práca sa zaoberá jednoduchým organizovaním literatúry a pracou s .bib dokumentmi. Opisuje ich štruktúru a praktické využitie. Ďalej popisuje existujúce dostupne riešenia, porovnáva ich klady a zápory. V neposlednom rade ich porovnáva s mojím riešením.

Hlavným cieľom mojej práce je vytvorenie web rozhrania pre pracovanie s .bib dokumentmi nielen v rámci jednotlivca ale aj v rámci skupiny. V tejto oblasti sa jedna hlavne o úpravu a vytváranie bibliografických súborov, spolu s vyhľadávaním, a organizáciou príloh. Samozrejmosťou je možnosť zdieľania v rámci skupiny. A keďže sa jedna web rozhranie, prístup je limitovaný len internetovým pripojením. Práca ďalej popisuje môj spôsob riešenia daného problému spolu so všetkými výhodami a zabezpečením či už voči používateľom alebo útočníkom z vonku.

**Kľúčové slová:** BibTeX , Bibliografia , organizovanie literatúry , .bib

## Obsah

Úvod .....	6
1. Formulácia problému .....	7
1.1. Čo je to BibTeX .....	7
1.2. Čo je to .bib.....	7
2. Riešenie problému .....	11
2.1. Dostupné riešenia.....	11
2.2. Porovnanie dostupných riešení .....	11
2.2.1. CiteULike .....	11
2.2.2. I, Librarian .....	12
2.2.3. EndNote.....	13
2.3. Ako si vybrať ten správny manažér? .....	14
2.4. Nástroje riešenia problému .....	14
2.4.1. Použité prostriedky.....	15
2.5. Zvolené riešenie.....	15
2.5.1. Front-end .....	15
2.5.2. Back-end.....	17
3. Kód .....	18
3.1. CSS .....	18
3.2. XHTML .....	19
3.3. PHP .....	21
3.3.1. PHP 5 .....	21
3.4. DB .....	22
3.5. Funkčnosť kódu .....	22
4. Konečná implementácia kódu .....	33
5. Bezpečnosť .....	34
5.1. SQL Injection.....	35
5.2. XSS .....	37
6. SEO .....	38
7. Spustenie systému .....	41
8. Záver.....	42
Použitá literatúra.....	43

## Úvod

Už niekoľko rokov sa komunikácia medzi ľuďmi presúva na internet. Ide nielen o sociálne siete ale aj o obchodovanie, štúdium a takisto pracú. Aj na základe tohto fenoménu som sa rozhodol zvoliť ako riešenie problému pred ktorým som stal práve internet. Keďže sa v dnešnej dobe stále častejšie v mnohých oblastiach stretávame s takzvaným zahltením informáciami, tj. väčším množstvom dát, ako je človek schopný spracovať. Podobný, či skôr rovnaký problém sa často vyskytujú aj v literatúre, odborných a vedeckých článkoch. Je potrebné prejsť veľmi veľké množstvo literatúry a byť schopný sa k potrebným materiálom, ako aj k zaujímavým článkom vrátiť a odkazovať na nich. Na organizovanie literatúry v dnešnej dobe existuje veľké množstvo systémov a programov. Ale myslím, že je viac ako potrebné organizovať nielen samotne referencie ale aj články. Samozrejmosťou by malo byť vyhľadávanie v rámci skupiny.

Cieľom našej práce bolo vytvoriť funkčné prostredie pre takúto manipuláciu a organizovanie referencií a všetkých príloh s nimi spojených. Nasledovne je umožnené vyhľadávanie v tejto literatúre a manipulácia v rámci jednotlivca a skupiny.

Výsledkom je webové rozhranie umožňujúce kompletnú manipuláciu a organizáciu bibliografických dokumentov, samozrejme s adekvátnou úrovňou bezpečnosti .

Práca je rozdelená do šiestich kapitol. Prvá kapitola je úvodom, určuje cieľ našej práce. V druhej kapitole formulujeme samotný problém. Definujeme všetky základne pojmy potrebné pre porozumenie problematiky a rozoberáme štruktúru samotného bibliografického súboru. Tretia kapitola obsahuje stručný popis riešenia, popisuje a porovnáva dostupne riešenia. Hodnotí ich klady, zápory a porovnáva ich s naším riešením. Podkapitoly popisujú zvolené riešenie, použité prostriedky a nástroje. V štvrtej kapitole je popísaný samotný kód. Detailne je rozobraný každý aspekt. Od CSS cez HTML, PHP, až po samotnú realizáciu databázy. Rozobraný vývoj krok po kroku. Bližšie sú popísané problematiky ako napríklad funkčnosť CSS vo väčšine prehliadačov alebo podpora pre nevidiacich. Piata kapitola popisuje bezpečnostný aspekt problematiky. Nielen v PHP ale aj pri databázovom prístupe. Posledne časti sú zaver, zoznam použitej literatúry.

## 1. Formulácia problému

Ako bolo už v úvode načrtnuté, stále častejšie sa v dnešnej dobe stretávame s veľkým množstvom informácií. Bohužiaľ veľká časť z nich nie je užitočná. Veľmi podobný problém sa vyskytuje aj v literatúre. Bolo by vhodné po nájdení správnych materiálov sa k nim vrátiť a vedieť ich neskôr organizovať.

Problém pred ktorým sme stali bolo vytvoriť a vhodne implementovať riešenie pre prácu s takzvanými bibliografickými súbormi. Bibliografický súbor obsahuje všetky informácie o zdroji s ktorým sme pracovali pri písaní ľubovoľného dokumentu. Či už ide o citácie, alebo väčšie bloky textu. Najelegantnejším spôsobom práce s bibliografiou je použitie programu s názvom BibTeX.

### 1.1. Čo je to BibTeX

BibTeX bol veľmi široko používaný od predstavenia počítačovým vedcom Oren Patashnikom pred približne 20 rokmi. Ako už samotný názov naznačuje, bolo plánované priame použitie s LaTeXom. Ale jeho inštancie spolupracujú aj s programom MS Word pomocou jednoducho dostupných nástrojov.

BibTeX sa dá definovať ako reference management software. V preklade ide o softvér slúžiaci na pracovanie s referenciami. BibTeX zjednodušuje citácie zdrojových článkov a separuje ich do osobitného súboru s koncovkou .bib. Používaný file-format je text-based, čo znamená, že ide iba o holý text. (OpenOffice.org Wiki. 2010.)

### 1.2. Čo je to .bib

Ide o štandardný textový súbor obsahujúci citácie. Čo sa týka štruktúry samotného .bib súboru, jeho obsahom sú takzvané vstupy. Všetky vstupy obsahuje svoj vlastný subset premenných. Štandardné premenne sú napríklad:

- Address: adresa, najčastejšie ide o vydavateľovu adresu.
- Annote: anotácia, poznámky
- Author: autor, ide o meno autora, pri viacerých autoroch sa mena oddeľujú anglickou spojkou „and“

- Booktitle: názov knihy, používa sa ak ide o citáciu len niektorej pasáže z knihy
- Chapter: číslo kapitoly
- Edition: edícia knihy
- Editor: meno editora
- Institution: meno inštitúcie, ktorá bola zapojená do vydania, nemusí ísť o vydavateľstvo.
- Journal: magazín v ktorom bolo dielo publikované
- Month: mesiac publikácie
- Number: číslo vydania magazínu
- Organization: meno organizácie
- Pages: čísla strán z ktorých bolo citované
- Publisher: meno vydavateľstva
- Series: číslo vydania
- Title: názov práce
- url: www adresa na dokument
- volume: číslo vydania magazínu
- year: rok publikácie

Existujú aj ďalšie premenné. Tieto sú už ale menej používané:

- abstract: abstrakt práce
- ISBN: (The International Standard Book Number) medzinárodné číslo knihy
- Size: fyzické rozmery práce

Pre vstup existuje skupina premenných ktoré by mal a ktoré môže obsahovať.

- @Article  
Článok z časopisu alebo magazínu  
Povinné premenné: author, title, journal, year  
Voliteľné premenné: volume, number, pages, month, note, key
- @Book  
Kniha s explicitne uvedeným autorom  
Povinné premenné: author, title, Publisher, year  
Voliteľné premenné: volume, series, address, edition, month



- @Booklet  
Ide o pracú ktorá je vytlačená a zviazaná ale bez mena autora  
Povinné premenné: title  
Voliteľné premenné: address, month, year
- @Incollection  
Časť knihy majúca svoj vlastný nadpis  
Povinné premenné: author, title, year, booktitle  
Voliteľné premenné: edition, organization, publisher, address, month
- @Inproceedings  
Článok, zápisky z jednania, rokovania alebo prezentované články konferencii.  
Povinné premenné: author, title, year  
Voliteľné premenné: edition, series, pages, publisher, address
- @Manual  
Technická dokumentácia  
Povinné premenné: title  
Voliteľné premenné: author, organization, address, edition

Používanie týchto vstupov nie je odporúčané, ale môžu sa vyskytnúť v niektorých bibliografiách. Ide o @collection, veľmi podobne @proceedings. Ďalším je @pattent, ktorý ako názov naznačuje sa používa pri patente. Naopak najbežnejšie sú @article a @inproceedings. (Jacobsen, C. 2006)

Ku samotnému príkladu vstupu v bibliografickom súbore.

```
@article{DT88,
  author = {G. Di Battista and R. Tamassia},
  title = {{Algorithms for Plane Representations of Acyclic Digraphs}},
  journal = {Theoretical Computer Science},
  year = {1988},
  volume = {61},
  pages = {175--198},
}
```

Každý vstup začína symbolom „@“ . Nasleduje názov vstupu. V našom prípade ide o citáciu z článku. Po názve vstupu nasleduje telo. Je ohraničené symbolmi „{“ „,}“ . Za otváracou zátvorkou začína reťazec kľúčov daného vstupu. Kľúče sú oddelene „/“ .

Nasleduje čiarka ktorá označuje koniec kľúčov. Nasleduje premenná title, ktorá označuje názov článku z ktorého bolo citovane a opäť ako ukončenie čiarka. Rovnakým spôsobom sú riešené všetky premenne. Obsahy všetkých premenných sú buď v úvodzovkách alebo v zložených zátvorkách. Obe varianty sú ekvivalentne. Aj keď zlozene zátvorky sa používajú aj ja definovanie textu ktorý nesmie byť dotknutý pri formátovaní textu. Číže nepodlieha žiadnej zmene aj keď zvyšok textu sa upraví. Najlepším príkladom je používanie týchto zátvoriek pri špeciálnych symboloch. Ide o set znakov ktoré sú pri čítaní reprezentovane ako napríklad  $\pi$ ,  $\ddot{u}$ ,  $\phi$  a pod. pre lepšiu predstavu uvediem niekoľko príkladov.

ò	\' {o}	ó	\' {o}	ô	\~ {o}
ö	\" {o}	õ	\~ {o}	ō	\= {o}
ó	\. {o}	õ	\u {o}	õ	\v {o}
õ	\H {o}	ôö	\t {oo}	ø	\c {o}
ø	\d {o}	ø	\b {o}	œ	\oe
Œ	\OE	æ	\ae	Æ	\ae
ã	\aa	Å	\AA	ø	\o
Ø	\O	ł	\l	Ł	\L
ß	\ss	?'	?'	!'	!'

Ide o najpoužívanejšiu skupinu znakov. Ďalej existuje špeciálne znaky pre grécku abecedu, v ktorej je napríklad znak  $\Omega$  definovaný ako `\Omega`. Existujú ďalšie skupiny definujúce napríklad binárne operácie, vzťahy či názvy goniometrických funkcií.

## **2. Riešenie problému**

Bolo nutne vytvoriť užívateľský nenáročný a na používanie intuitívne prostredie, ktoré, keďže sa jednalo o web-based verziu, by pracovalo rovnako na všetkých bežne používaných prehliadačoch a operačných systémoch. Používanie muselo byť jednoduché a hlavne rýchle. Najväčšou výhodou riešenia mala byť možnosť pracovať on-line na bibliografii. Tým pádom už nie je nutne posielat' si bib súbory v prílohe mailom, alebo odkladať na server kde si ho kolegovia mohli stiahnuť. Pretože stále aj po úprave museli opäť uploadovať opravený súbor na server.

### **2.1. Dostupné riešenia**

V dnešnej dobe je dostupný veľký počet riešení pre prácu s bibliografickými súbormi. Spôsoby pracovania s bibliografiou sa líšia. Takisto sa líši aj druh programu. Môže ísť o web-based, software alebo centrally-hosted verziu. Odlíšnosti sú aj v licencií ktorú používajú. Či už ide o GPL alebo Proprietary licenciu. A zrejme najväčší argument pri rozhodovaní, či sú platene alebo nie a či podporujú Váš operačný systém.

### **2.2. Porovnanie dostupných riešení**

Samozrejme všetky dnes dostupné riešenia majú svoje výhody aj nevýhody. Ale jednu nevýhodu má drvivá väčšina spoločností. Je to neschopnosť spolupracovať na práci s kolegami ktorí sa fyzicky nenachádzajú na Vašom počítači. Ale ak si na moment odmyslíme túto nevýhodu, a fakt že veľká väčšina ľudí používa BibTeX, tak k favoritom v obľúbenosti by mohli patriť programy ako CiteULike, JabRef a EndNote. Z pomedzi menovaných má EndNote malú nevýhodu. A to cenu. Prvé dva sú totiž voľne dostupné. (Comparison of Free Bibliographic Managers. 2008.)

#### **2.2.1. CiteULike**

CiteULike je free on-line program na organizovanie publikácií. Je dostupný od roku 2004, kedy jeho tvorca bol v spojení s Univerzitou v Manchestri. Išlo o prvý web-based nástroj špeciálne pripravený pre potreby vedcov a študentov univerzity.

Ako aj ďalšie populárne nástroje umožňuje priamo zo stránky uložiť záložky spolu s metadátami cez prehliadač. Tieto uložené záložky je možné jednoducho zdieľať cez linky. Aktuálne je cez CiteULike uložených približne 4 milióny záložiek. Aj keď software CiteULike nie je open source, časť databázy je na verejnom servery.

CiteULike upravuje záložky pred samotným uložením do databázy, čo znamená, že zisťuje či už neexistuje pravé daná stránka u iného používateľa. Toto je veľmi dôležité pre vývoj CiteULike po stránke sociálneho vývoja, pretože umožňuje zistiť obľúbenosť stránok.

Jednou z pozoruhodných vlastností je systém rozpoznania či predchádzalo uloženiu stránky ako záložky čítanie., čo umožňuje ohodnotiť stránku a potenciálne ju odporučiť pre ďalších používateľov. Na základe tejto vlastnosti bolo zistené, že väčšina používateľov si stránky pred uložením precítala.

CiteULike je založená na princípe social-bookmarking-u a je zameraná na ochranu a vylepšovanie zdrojov pre vedecké účely. Rovnakým spôsobom ako je možné katalogizovať stránky je možné pracovať aj s obrázkami, čo umožňuje vedcom lepšiu organizáciu materiálov a uľahčuje prácu.

Nové príspevky sú automaticky zaradené ako verejne, čo znamená, že sú dostupné pre všetkých používateľov. Samotné vstupy sú ale už súkromné, tak isto ako všetky metadata. Používatelia ale môžu zdieľať všetky svoje záložky aj metadata z ďalšími používateľmi ak sa tak rozhodnú. Prídavnou vlastnosťou je možnosť vytvárania skupín a pridávania do skupín. Všetky vstupy ktoré si používateľ uloží si môže stiahnuť. Sú uložené v BibTeX-ovom formáte, ale na vyber u aj ďalšie.

Kód v pozadí funkčnosti CiteULike je kombináciou Lisp-u a Perl-u, dáta sú uložené pomocou PostgreSQL. Ako už bolo spomenuté, zdrojový kód nie je verejne dostupný, ale dáta sú uložené na verejnom servery. (CiteULike. 2010.)

### **2.2.2. I, Librarian**

I, Librarian je, ako aj predchádzajúci produkt, web-based systém pre organizovanie literatúry. Používa GPL licenciu, čo umožňuje prehliadať zdrojový kód. Väčšina funkčnosti je zameraná na sprostredkovanie prostredia pre prácu v PDF súboroch. Program nemá bohužiaľ schopnosť naformátovať referencie do klasickej podoby používanej vo väčšine prac.

Hlavnými vlastnosťami I, Librarian sú možnosť vybudovať vlastnú dokumentáciu vo formáte PDF, ako prehľadné hľadanie v PDF. Samozrejmosťou je vyhľadávanie a organizovanie nielen materiálov ale aj samotných príloh.

Aj keď je hneď v úvode avizovane, že I, Librarian sa zameriava výhradne na PDF súbory. V konkurencii sa nestratí pretože aj keď vie robiť len s jedným formátom, ide o veľmi používaný a rozšírený formát. Keďže ide opäť o web-based produkt, umožňuje prácu na všetkých operačných systémoch ako aj na väčšine prehliadačoch. Čo mu ale treba vytknúť je nie úplná funkčnosť pod operačným systémom Mac OS, ale keďže sa jedna len o maličkosti, je použiteľný aj tu. (I, Librarian. 2010.)

### **2.2.3. EndNote**

EndNote je oproti predchádzajúcim produktom komerčne zameraný. Umožňuje organizáciu bibliografie a referencií. Tvorcom je Thomson Reuters. Uložená bibliografia má podobu \*.enl, čo je odlišný formát od klasického \*.bib s ktorým pracujú ostatné programy. Všetky externé materiály k danej bibliografii sú uložené v rovnomernom súbore s koncovkou \*.data.

Existuje niekoľko riešení ako pridať v tomto programe referenciu. Môžeme si zvoliť manuálny spôsob, export, import alebo pripojenie z EndNote. Program sa prezentuje používateľovi ako okno s jednoduchým drop-down menu pre vyber typu referencie (napríklad @book, @article, @manual) a poliami ktoré môžu byť od všeobecných (author, title, year) po špecifické len pre danú referenciu. (ISBN, abstract).

EndNote spolupracuje s ďalšími databázami, čo umožňuje priamy export do EndNote databázy a šetrí používateľom čas pri vyhľadávaní v svojej práci. Ak používateľ korektne vyplní všetky povinné polia tak EndNote dokáže citáciu zobrazit v ľubovoľnom formáte, čo zjednodušuje prácu.

Možné komplikácie môžu nastať pri spájaní s programom Word 2010. Word nemôže byť spustený pokiaľ nie je EndNote vypnutý. Takisto od používateľa sa očakáva istý stupeň zručnosti pri práci s bibliografiou. A aj napriek tomu, že je tento program platený, nevie rozoznať a zareagovať na napríklad situáciu kedy do pola year je vložená hodnota 2008 a pola date hodnota 10.12.2008. výpis bude 10.12.2008 2008.

Zrejme najväčšou nevýhodou je absolútna neschopnosť vytvorenia bibliografického súboru s koncovkou .bib, ktorý je najviac používaný.

Napriek tomu, že každý program ma určité výhody, existuje skupina ľudí ktorý si pre pracú s bibliografiou zvolili zrejme tu najintuitívnejšiu možnosť. Textový editor. Aj keď si tento variant vyžaduje zročnosť a trpezlivosť, môže sa ukázať ako dobre a rýchle riešenie. Pre bližšie informácie od ďalších programoch je v prílohách uvedená prehľadná tabuľka. (EndNote. 2010.)

### **2.3. Ako si vybrať ten správny manažér?**

Keď hľadáme a hodnotíme dostupný softvér pre pracú s referenciami mali by sme sa zamerať na tieto základné vlastnosti.

- Hľadanie  
Umožňuje nám daný produkt vyhľadávanie v on-line databáze? Ak áno, tak v akom rozsahu?
- Zdieľanie  
Umožňuje na zdieľanie bibliografie medzi rôznymi počítačmi alebo používateľmi?
- Ukladanie  
Aké existujú možnosti pre uloženie nasej bibliografie? Môžeme si uložiť aj externe prílohy nielen samotne referencie?
- Čítanie  
Je možný import bibliografie? Môžeme si upraviť môj existujúci súbor, alebo použiť fulltextové vyhľadávanie?
- Zápis  
Môžeme vkladať citácie do prace? Môžeme vytvárať korektnú bibliografiu k prací?
- Použitie  
Dal by sa používať pri každodennom písaní a čítaní prací?
- Platforma  
Je pre naše použitie praktickejší program alebo internetová služba?

### **2.4. Nástroje riešenia problému**

Ako nástroje nášho riešenia sme zvolili pravé viac krát' spomínaný web-based prístup. Tato možnosť ponuka omnoho širšie využitie a zjednodušuje pracú v rámci skupiny.

Takisto splna drvivú väčšinu otázok uvedených v predchádzajúcej časti.

### **2.4.1. Použite prostriedky**

Použite nástroje zahrnovali IE Tester pre kontrolu funkčnosti CSS v starších verziách IE. ScreenReader pre uistenie sa, že stránka je naozaj prístupná a použiteľná aj pre nevidiacich. Firebug v prehliadači firefox pre kontroly a úpravy nekorektného kódu. W3C validátor ako nástroj kontroly korektnosti nielen CSS, ale aj html kódu. SEO Monitor pre testovanie kvality kódu podľa SEO štandardov.

## **2.5. Zvolené riešenie**

Keďže ide o web-based produkt, je nutne sa na pracú pozerat' z dvoch uhlov.

- Front-end ( časť zameraná na používateľa )
- Back-end ( časť určená pre administrátor )

### **2.5.1. Front-end**

Front-end je časť prace venovaná používateľovi. Drvivá väčšina obsahu je získavaná z databázy. Keďže sa jedna o používateľské prostredie bolo nutne vytvorit' příjemne grafické spracovanie.

Grafika by mala byt samoúčelná, ale predovšetkým funkčná. Podľa informácii zo servera Consumer WebWatch hodnotí viac ako 46 % používateľov funkčnosť a dôveryhodnosť podľa vzhľadu. Vzhľad webu je vôbec to prvé, čo môže používateľa k používaniu prilákať a naopak odlákať. Preto stále aj tu platí, že najdôležitejší je prvý dojem, keďže druhu šancu už nedostaneme.

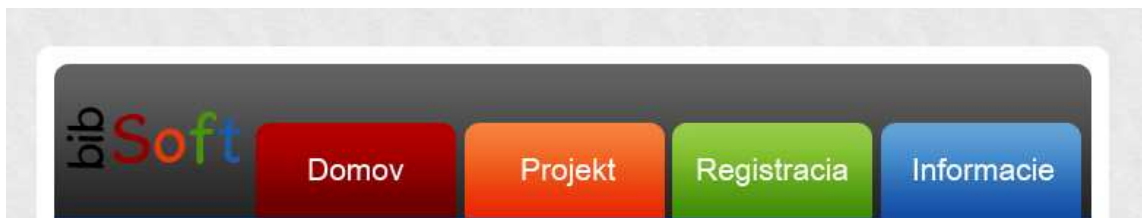
Môj grafický návrh je rozdelený na dve časti. Prvá časť je index, čiže prvotná stránka. Druhá časť pozostáva zo samotného obsahu stránky.

Štruktúra stránky:

Keďže sa jedna o prvú stránku ktorá sa používateľovi zobrazí hneď pri vstupe bolo potrebná zvolit' příjemný nie komplikovaný design. Podstatou stránky je istým spôsobom uvítať používateľa. Obsahuje logo, a veľmi intuitívne menu.



Názov v logu sa odvíja od faktu, že stránky pracuje so súbormi s koncovkou .bib. Farebnosť bolo zvolená na základe farieb menu.



Design stránky bol zvolený nasledovne pravé pre funkčnosť.

- Hlavička (Header)

- logo slúžiace ako odkaz na stránku indexu

- menu pre orientáciu na stránke

- Obsah (Content)

Keďže stránka neobsahuje po štruktúrnej stránke sidebar, telo tvory main div

- main div, časť obsahujúca samotný dátový obsah stránky.

Jeho obsah sa odlišuje na každej podstránke. Keď si podstránky ako informácie, kde je uložený manuál ku stránke spolu so špecifikáciou a registráciu ktorá obsahuje formulár pre nového používateľa odmyslime, obsahom tohto divu su :

- používateľov panel, ktorý obsahuje submenu pre navigáciu pri práci s bibliografiou, a možnosti úpravy údajov používateľa.

- jadro, ktorého obsahom sú najčastejšie tabuľky generovane z databázy obsahujúce informácie o bibliografii.

Okrem vyššie spomínaných sekcii ako sú registrácia a informácie sa obsah tohto div-u neprihlásených používateľom nikdy nezobrazí.

- Pata (Footer)

- Obsahuje copyright a takzvaný informačný riadok ktorý ma za úlohu informovať o stave pri práci s vašou bibliografiou. Napríklad, či mate otvorený súbor na úpravu alebo vytvárate nový súbor.



### **2.5.2. Back-end**

Back-end je časť stránky určená pre administrátora. Uľahčuje prácu s databázou a umožňuje kontrolu používateľov. Je sprístupnená len administrátorom čo znamená, že používateľ sa k týmto dátam nikdy nedostane. Ide o sprístupnenie druhej polovice stránky. Administračná zóna je ochránená heslom konkrétneho používateľa, ktorý je administrátorom. Implementuje niekoľko hierarchických systémov.

Ide o komplexný zoznam používateľov spolu s príslušenstvom do skupín Admin môže upravovať informácie používateľoch, pridávať im práva administrátora a samozrejme samotných používateľov vymazávať.

Ďalej ma k dispozícii zobrazený celý obsah databázy uložených bibliografií a príloh k nim.

### 3. Kód

Pre tvorbu web stránok sa už dlho používal štandardne HTML jazyk. Skratka znamená hypertext markup language. Ale nove trendy predstavili XHTML jazyk ktorý ma prísnejšiu syntax. Ide o aplikáciu XML. Naša práca je vytvorená pomocou XHTML 1.0 v kombinácii s CSS

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="sk" lang="sk">
```

#### 3.1. CSS

Css respektíve cascading style sheets sa používajú na oddelenie obsahovej stránky webu od grafickej stránky. Ak je design definovaný výlučne pomocou CSS, je veľmi jednoduché zmeniť kompletný design a usporiadanie stránky iba na základe zmenenia CSS súboru.

```
<link href="styles/style.css" rel="stylesheet" type="text/css" />
```

Kódovanie stránky je vo formáte UTF-8. Predchádza sa tým problémom s diakritikou pri práci s databázou aj je totožné kódovanie definované aj v nej.

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<meta http-equiv="Content-language" content="sk" />
```

Ako metódu práce s CSS som zvolil prístup pod názvom floating divs. Ide o metódu práce s css a html ktorá spočíva v separovaní jednotlivých častí do <div> elementov. Nasledovne sú <div> elementy veľmi vhodné pre nahradenie klasického tabuľkového riešenia designu. Element <div> je takzvaný blokový element, čiže rozdeľuje časti stránky na bloky ktorým už môžeme ďalej pridávať takmer ľubovoľné vlastnosti, čo zjednodušuje prácu s designom stránky.

Štruktúra stránky je, ako je hore zobrazené, rozdelená na <div> s id main. Ide o hlavný element. V ňou sa nachádza <div> header ktorý obsahuje logo slúžiace pre návrat na index, a menu pre navigáciu. Nasleduje <div> main\_content ktorého obsahom sú samotne dáta získané z databázy zobrazené v rôznych formátoch. Posledným je <div>

footer. Ako názov naznačuje, ide o patu stránky ktorá sa najčastejšie využíva pre copyright a drobné poznámky.

Napriek tomu, že metóda floating divs je veľmi rozšírená, každý prehliadač si reprezentuje hodnoty s malými odlišnosťami. Zrejme najväčším problémom pri vytváraní graficky pokročilej stránky je fakt, že sa v rôznych prehliadačoch môže design jemne meniť. Vytvárajú sa pixelove odskoky sa zarovnávanie je odlišné. Najlepším príkladom je zobrazenie stránok v prehliadači Internet Explorer 6. Zrejme najbežnejším riešením bolo používanie externého CSS súboru len pre prehliadač Internet Explorer 6. Toto riešenie nie je nesprávne ale nie je uznávané, pretože korektným postupom je možné docíliť grafičku zhodu aj v tomto prehliadači.

Medzi najbežnejšie problémy patria napríklad „box model“ problém alebo „peekaboo bug“. Ide o veľmi rozšírené problémy v tomto prehliadači.

Problém „box model“ je zapríčinený skutočnosťou, že Internet Explorer 6 nepodporuje min a max hodnoty css. Preto je nutne použiť napríklad hodnoty width a max-width. Najjednoduchším riešením je nastaviť pre zobrazovanie v tomto prehliadači rozmery o trochu väčšie. Najlepšie cez padding.

Takzvaný „peekaboo bug“ nastáva, keď používame float ktorý je obkolesený textom a spodný okraj divu je nižšie ako spodný okraj bloku textu. Správnym riešením je definovanie elementu <div> ktorý je väčší ako pôvodný a je mu nadradený. Definujeme mu width: 100% a pozíciu relatívnu voči textu.

Najlepším riešením problémov s prehliadačom Internet Explorer 6 je myslieť vždy o krok dopredu a keď poznáme aké chyby sa v ňom vyskytujú, tak sa snažiť im predchádzať. Správnym heslom pri implementácii designu by malo byť „tvoriť pre Firefox a Operu, optimalizovať pre Explorer“

### **3.2. XHTML**

XHTML alebo Extensible Hypertext Markup Language je, ako samotný názov napovedá, rozšírený hypertextový značkový jazyk. Je veľmi podobný HTML, ale s oveľa prísnejšou syntaxou. Hlavnou odlišnosťou je fakt, že HTML je aplikáciou SGML, ale XHTML je aplikáciou XML, obmedzenej časti SGML. V roku 2000 W3C odporučilo XHTML ako štandard. Medzi výhody XHTML patri napríklad fakt, že jeho dokumenty sú súčasne dokumentmi XML, čo umožňuje upravovanie a validovanie

nielen prehliadačmi. Čo sa týká rozdielov medzi HTML a XHTML najväčší rozdiel sa dá nájsť v porovnaní verzii XHTML 1.1, kde boli odstavené niektoré elementy.

Pomocou korektného formátu XHTML kódu je možné sprístupniť stránku aj pre zrakovo postihnutých. Takýto ľudia používajú špeciálne programy nazývané screen reader, ktoré čítajú informácie z obrazovky pre nevidiacich. Bohužiaľ veľká časť stránok, aj v dnešnej dobe, nespĺňa základy pre funkčnosť týchto programov. Existuje niekoľko pravidiel ktoré by mala stránka spĺňať aby ju bolo možné prehlásiť za blind friendly.

- Všetky grafické objekty musia mať alternatívny text, čiže element musí obsahovať časť alt  
``
- Takisto všetky informácie podávané skriptami, flashom alebo cez CSS musia byť prístupné bez rozšírení prehliadača. Ako príklad by som uviedol situáciu v ktorej je stránka vynikajúco prepracovaná v programe Flash, ale ako nevidiaci používateľ by som mal byť schopný citiť a prechádzať stránku aj bez nainštalovaného Flashu. Ako riešenie sa používa textové menu v pate stránky.
- Stránka by sa nemala neustále bez používateľovej interakcie meniť, pretože používané programy čítajú stránku ako celok a reagujú na každú jej zmenu.
- Odkazy by mali jasne pomenovať svoj cieľ. Medzi nesprávne odkazy patrí napríklad: *Informácie môžete nájsť* `<a href="info.php" >tu</a>`. Korektný kód by mal jasne pomenovávať cieľ odkazu: *Všetky informácie k projektu nájdete tu:* `<a href="info.php">Projektová špecifikácia</a>` Samozrejme aj nie je iná možnosť, tento problém sa dá vyrieši pomocou vloženia atribútu *title*.
- Všetky texty by mali byť dostupné v čiernej - bielej farbe. Niektorí farboslepí používatelia by inak neboli schopní rozoznať text. A samozrejme musí byť jasný kontrast medzi textom a pozadím.
- Samotný kód by mal byť validný, keďže niektoré citacie programy majú problémy s nevalidným kódom.
- Je nutnosťou správne používať nadpisy h1 a h2. Prednosť pri programovaní by mal mať element h1 ako napríklad element div ktorému nastavíme veľkosť písma.

- Pri odkazoch na súbory je nutne spresniť typ a veľkosť súboru.

```
<a href="bakalarka.zip" title="odkaz na bakalársku prácu (.zip
89kB)">Bakalárska práca</a>
```

Ak stránka splna tieto základne vlastnosti, tak by mala byť dostupná pre väčšinu nevidiacich a slabozrakých.

### 3.3. PHP

PHP patri medzi najpopulárnejšie open source skriptovacie jazyky. Umožňuje spracovanie a nasledovne generovane obsahu web stránky na strane servera. Server nasledovne odovzdáva stránku webovému prehliadaču. Najčastejšie sa používa pri práci s databázou. V tomto spojení je možné naprogramovať ľubovoľne komplikovanú webovú aplikáciu. Spoľahlivo pracuje spolu so všetkými rozšírenými webovými servermi.

K dátumu 4.3.2010 bola vydaná najnovšia verzia PHP5.3.2 ktorá opravuje veľké množstvo malých chyb z predchádzajúcej verzie.

#### 3.3.1. PHP 5

PHP5 je nový objektový model. Vďaka kompletnej zmene práce s objektmi umožňuje lepšiu funkcionality.

```
<?php
class vzorovy_class{
    //deklaracia premennej
    public $premenna = 'hodnota';
    //deklaracia metody
    Public function zobraz_premennu(){
        Echo $this -> premenna;
    }
}
?>
```

Objekty sú za ľubovoľných podmienok odovzdávané referenciou. Buď \$this->premenna alebo ak pracujeme v druhom súbore je nutne najprv deklarovať s ktorým

súborom pracujeme. \$subor = new vzorovy\_class (); a potom môžeme použiť \$subor->premenna;

Metoda deklarácie premennej cez kľúčové slovo var bolo nahradene pomocou public, private, protected. Triedam možno zakázať extendovanie a prepisovanie pomocou definície ako final.

Aj vďaka týmto výhodám sme si PHP5 zvolil za nástroj pre prácu s bibliografickými súborami.

### 3.4. DB

Pri práci s databázou bolo zvolené riešenie ktoré obsahovalo spoluprácu s databázou MySQL. Tato databáza je voľne šíriteľná. Ako administračný prvok bol zvolený nástroj phpMyAdmin. Tato kombinácia ponúka používateľovi možnosť komplexne pracovať s databázou a tabuľkami MySQL. Systémový administrátor poskytuje pre administrátora stránky nástroje pre manipuláciu a spravovanie všetkých tabuliek spolu s vlastnosťami a nastaveniami. Samozrejmosťou je export a import dát , vyhľadávanie a komplexná práca s tabuľkami. Je naprogramovaný v PHP.

### 3.5. Funkčnosť kódu

V tejto kapitole by sme sa chceli venovať priamemu vysvetleniu funkčnosti kódu v sekcii ktorá priamo pracuje s bibliografickými súborami. Tuto funkčnosť by som chcel prezentovať na jednoduchom príklade bibliografického súboru.

Súbor *literatura.bib*

```
@Book{picard/computing,  
  author = "Rosalind W. Picard",  
  title = "Affective Computing",  
  publisher = "The {MIT} Press",  
  year = "1997",  
  address = "Cambridge Massachusetts",  
  ISBN = "0-262-16170-2",  
}
```

```

@Article{DT88,
  author = {G. Di Battista and R. Tamassia},
  title = {{Algorithms for Plane Representations of Acyclic Digraphs}},
  journal = {Theoretical Computer Science},
  year = {1988},
  volume = {61},
  pages = {175--198},
}

@InProceedings{bobox2,
  title = {Using Methods of Parallel Semi-structured Data Processing for
  SemanticWeb 3rd International Conference on Advances in Semantic Processing
  {SEMAPRO}},
  author = {David Bedn{\a}rek and Ji{\v{r}}{\i} Dokulil and Jakub Yaghob
  and Filip Zavoral},
  year = {2009},
  publisher = {{IEEE} Computer Society Press},
  pages = {44--49},
  isbn = {978-0-7695-3833-4},
}

```

Tento bibliograficky súbor obsahuje tri vstupy. Prvá je citácia z knihy, nasleduje článok a posledná citácia je zo zápiskov.

Predtým ako odprezentujeme všetky funkcie program, považujem za nutnosť objasniť štruktúru databázy s ktorou pracujem.

Tabuľka	Typ	Zotriedenie	Veľkosť	Naviac
bc_bib	MyISAM	utf8_slovak_ci	2,5 KB	-
bc_pouzivatelia	MyISAM	utf8_slovak_ci	3,1 KB	128 Bajtov
bc_skupina	MyISAM	utf8_slovak_ci	2,2 KB	20 Bajtov
bc_vstupy	MyISAM	utf8_slovak_ci	3,7 KB	-

Databáza pozostáva zo 4 tabuliek.

- Bc\_pouzivatelia
  - Uid používateľove id
  - Prihlasmeno prihlasovacie menu používateľa
  - Heslo
  - Meno
  - Priezvisko
  - Mail
  - Admin hodnoty 1 a 0 označujú či je používateľ administrátorom
  - Skupina skupina do ktorej používateľ patri

Pole	Typ	Zotriedenie	Atribúty	Nulový	Predvolené	Extra
<b>uid</b>	smallint(6)			Nie		auto_increment
<b>prihlasmeno</b>	varchar(20)	utf8_slovak_ci		Nie		
<b>heslo</b>	varchar(50)	utf8_slovak_ci		Nie		
<b>meno</b>	varchar(30)	utf8_slovak_ci		Nie		
<b>priezvisko</b>	varchar(30)	utf8_slovak_ci		Nie		
<b>mail</b>	varchar(30)	utf8_slovak_ci		Nie		
<b>admin</b>	tinyint(2)			Nie		
<b>skupina</b>	tinyint(3)			Nie		

- Bc\_skupiny
  - Id id priradene skupine pri vytvorení
  - Zakladatel\_id id zakladateľa, umožňuje lokálnu administráciu
  - Nazov

Pole	Typ	Zotriedenie	Atribúty	Nulový	Predvolené	Extra
<b>id</b>	smallint(6)			Nie		auto_increment
<b>zakladatel_id</b>	smallint(6)			Nie		
<b>nazov</b>	varchar(50)	utf8_slovak_ci		Nie		

- Bc\_bib
  - Bib\_id
  - Author\_id id používateľa ktorý uložil bibliografiu
  - Skupina\_id id skupiny do ktorej používateľ patri
  - Nazov
  - Cesta cesta k súboru na server
  - Pocet počet vstupov
  - Vstupy id vstupov



Pole	Typ	Zotriedenie	Atribúty	Nulový	Predvolené	Extra
<b>bib_id</b>	smallint(6)			Nie		auto_increment
<b>author_id</b>	smallint(6)			Nie		
<b>skupina_id</b>	smallint(6)			Nie		
<b>nazov</b>	varchar(50)	utf8_slovak_ci		Nie		
<b>cesta</b>	varchar(50)	utf8_slovak_ci		Nie		
<b>pocet</b>	smallint(6)			Nie		
<b>vstupy</b>	text	utf8_slovak_ci		Nie		

- Bc\_vstupy tabuľka jednotlivých vstupov
  - Vstup\_id
  - Vstup\_typ definuje či sa jedna o article, book, manual alebo inproceeding
  - Kluce
  - Title
  - Author
  - Year
  - Publisher
  - Journal
  - Organization
  - ISBN
  - Priloha ak existuje príloha v tejto časti je k nej zapísaná cesta

Pole	Typ	Zotriedenie	Atribúty	Nulový	Predvolené	Extra
<b>vstup_id</b>	smallint(6)			Nie		auto_increment
<b>vstup_typ</b>	varchar(100)	utf8_bin		Nie		
<b>kluce</b>	varchar(100)	utf8_bin		Nie		
<b>title</b>	varchar(300)	utf8_bin		Nie		
<b>author</b>	varchar(100)	utf8_bin		Nie		
<b>year</b>	varchar(100)	utf8_bin		Nie		
<b>publisher</b>	varchar(100)	utf8_bin		Nie		
<b>journal</b>	varchar(100)	utf8_bin		Nie		
<b>organization</b>	varchar(100)	utf8_bin		Nie		
<b>ISBN</b>	varchar(100)	utf8_bin		Nie		
<b>priloha</b>	varchar(100)	utf8_bin		Nie		

Je nutné si uvedomiť, že v tejto tabuľke nebude v každom riadku vyplnený každý stĺpec. Tabuľka je navrhnutá tak by slúžila pre všetky typy vstupov zároveň. Preto pri uložení napríklad citácie z kníh budú v tabuľke uložené, okrem povinných vstup\_id, vstup\_typ, kluce, title, author, publisher, year, ISBN. Samozrejme ak sú všetky tieto premenne definované v samotnom bibliografickom dokumente.

Samotná adresárová štruktúra je veľmi jednoduchá. Ide o tri podpriechinky a samotný domovsky adresár.

- Files, tu sú uložené všetky uploadnute súbory
- Prilohy, tu sú uložené všetky prílohy k referenciám
- Temp, v tomto priečinku sa dočasné ukladajú bibliografické súbory ktoré sú otvorene na úpravu.

Súbory vytvorene používateľmi sa dočasné ukladajú v domovskom adresári.

Najzákladnejšou funkciou by mala byť vlastnosť prezerania si bibliografického súboru. Po načítaní súboru, pri ktorej prebieha kontrola či sa jedna skutočne o bibliografický súbor, prichádza k slovu časť kódu ktorá súbor spracuje do tabuľky.

```
while ($m>0) {
  $podjadro=trim(substr($jadro,0,$m));
  $jadro=substr($jadro,$m+1);
  $stred=strpos($podjadro,'=');
  if ($stred>0) {
    $obsah_bunky = substr(trim(substr($podjadro,$stred)),1);
    $this -> replace($obsah_bunky);
    echo '<tr><td style="border: 1px solid;">'.trim(substr($podjadro,0,$stred-1)).'</td>
        <td style="border: 1px solid;">'.$_SESSION['uprava'].'</td></tr>';
  }
  else {
    echo '<tr>
        <td style="border: 1px solid;">Kluce</td>
        <td style="border: 1px solid;">'.trim(str_replace("/"," ", "$podjadro)).'</td></tr>';
  }
}
```

Tejto funkcii predchádza rozdelenie celého súboru na sekcie začínajúce symbolom @. Keďže tento symbol je unikátny a v textovej podobe sa nachádza iba v spojení s názvom vstupu. Samozrejme sa môže nachádzať v nazve článku alebo dokumentu, ale vtedy je v tele sekcie.

Nasledovne sú spracovávané samotne sekcie. Tato časť pracuje nasledovne. Vďaka formátu vieme, že po nazve referencie, napríklad @Book a otvorení sekcie pomocou zlozenej zátvorky nasledujú kľúče v textovom formáte. Kľúče môžu byť oddelene buď čiarkou alebo symbol backslash. Nasleduje čiarka označujúca koniec kľúčov. Nasleduje niekoľko riadkov v ktorých sú definované všetky premenne.

```
author = "Rosalind W. Picard",
title = "Affective Computing",
publisher = "The {MIT} Press",
year = "1997",
address = "Cambridge Massachusetts",
ISBN = "0-262-16170-2",
```

Ich formát je totožný. Začínajú názvom premennej symbolom = a nasleduje obsah premennej. Obsah premennej môže byť ohraničený úvodzovkami alebo zloženými zátvorkami. Každý riadok je vyseparovaný a vložený do tabuľky pre prehľadnosť.

Výsledná tabuľka vyzerá nasledovne.

```
<h2>Book</h2>
<table><tr>
  <td>Kluc</td>
  <td>picard , computing</td></tr><tr>
  <td>author</td>
  <td>"Rosalind W. Picard"</td></tr><tr>
  <td>title</td>
  <td>"Affective Computing"</td></tr><tr>
  <td>publisher</td>
  <td>"The MIT Press"</td></tr><tr>
  <td>year</td>
  <td>"1997"</td></tr><tr>
  <td>address</td>
  <td>"Cambridge Massachusetts"</td></tr></tr>
```

```

<td>ISBN</td>
<td> "0-262-16170-2" </td>
</tr></table>

```

Pri finálnom výpise je nutne samozrejme skontrolovať hodnoty premenných na špeciálne znaky. Pre tento účel bola vytvorená funkcia ktorá všetky špeciálne znaky adekvátne nahradí.

Použite php funkcie sú:

- Trim() funkcia pre vymazanie whitespace znakov a niekoľko ďalších špeciálnych znakov, ako napríklad \t alebo \n
- Substr() slúži na oddelenie istého počtu znakov od daného bodu
- Strpos() ktorá sa používa na hľadanie daného znaku v reťazci

Okrem prezerania samotného bibliografického súboru môže používateľ tento súbor uložiť na server, či už za účelom uloženia dát, alebo s úmyslom úpravy. V tomto prípade je postup veľmi podobný riešeniu pri vypíšaní do tabuľky. Odlišuje sa ale tým, že hodnoty premenných sú uložené do *input*-ov.

```

<h2>Book</h2>
<input name=typ_vstup_1 size="50" value="Book" readonly="readonly" </input>
<p>Kluc</p>
<input name=kluce_vstup_1 size="50" value="picard , computing"></input>
<p>author</p>
<input name=author_vstup_1 size="50" value="Rosalind W. Picard"></input>
<p>title</p>
<input name=title_vstup_1 size="50" value="Affective Computing"></input>
<p>publisher</p>
<input name=publisher_vstup_1 size="50" value="The MIT Press"></input>
<p>year</p>
<input name=year_vstup_1 size="50" value="1997"></input>
<p>address</p>
<input name=address_vstup_1 size="50" value="Cambridge
Massachusetts"></input>
<p>ISBN</p>

```

```
<input name=ISBN_vstup_1 size="50" value="0-262-16170-2"></input>
```

Každý input ma pridelenú hodnotu name a value aby bolo možné pri vekom množstve dát rozoznať príslušnosť obsahov k sekciám. Po zobrazení výpisu je ešte možnosť pridať ku konkrétnej citácii prílohu.

```
<input type="file" name="input_vstup_1" id="input_vstup_1" accept="application/pdf"/> <input name="ulozit" type="submit" id="ulozit" value="Uloz" />
```

Akceptované sú prílohy vo formáte pdf ktoré sú nasledovne uložené na server v adresári `\prilohy`

Pri každej zobrazenej sekcii sa nachádza možnosť uloženia vo forme checkboxu ktorý je prednastavený na stav checked.

```
<input type="checkbox" name="ulozit_vstup_1" value="ulozit_vstup_1" checked = "yes" /> ulozit
```

Nasleduje možnosť uloženia sekcií, ktoré sú označené do databázy pomocou stlačenia tlačidla „uložiť“.

```
function post_upload_bib(){
    if (isset($_POST['ulozit'])){
        echo $_SESSION['pocet'];
        $this->upload_vstupy($_SESSION['pocet']);
        $bib_author_id = $_SESSION['uid'];
        $bib_skupina_id = $_SESSION['skupina'];
        $bib_nazov = $_SESSION['nazov'];
        $bib_cesta = $_SESSION['cesta'];
        $bib_pocet = $_SESSION['pocet'];
        $test = $_SESSION['id_vstupov'];
        echo 'toto je pocet vstupov'. $bib_pocet;
    }
}
```

Tato funkcia zavolá ďalšiu funkciu s názvom `upload_vstupy()` ktorá skontroluje o aký typ vstupu ide. V aktuálnej verzii sú rozoznávané štyri typy vstupov. Article, Manual, Book a InProceedings. Po rozlíšení prístupy k uploadovaniu sekcie do databázy na servery. Toto prebieha v dvoch krokoch. Prvým krokom je uploadovanie sekcie do

databázy. Druhým krokom je uploadovanie informácií i samotnom bibliografickom súbore do separátnej tabuľky.

V tabuľke vstupy sú teda uložené všetky obsahy vstupov. Pre každú sekciu je vytvorený separátny riadok. Ďalšie informácie ktoré sú uložené v každom riadku sú ID používateľa spolu s ID skupiny ku ktorej patri a takisto ID samotného vstupu ktoré je unikátne.

V tabuľke bib je vytvorený jeden riadok s unikátnym ID pre každý uploadnutý súbor v ktorom sú uložené ID všetkých aktuálne uložených sekcií spolu s cestou na súbor na servere.

Keďže už máme naplnené tabuľky, môžeme prejsť k demonštrácii ďalších funkcií. Ako bežný používateľ, by sme si chceli prezrieť už uložené referencie, či už nás alebo niekoho z našej skupiny. Mam k nim prístup hneď po prihlásení. Možnosti manipulácie sú široké. Môžeme si jednotlivé referencie prezerať. Môžeme ich upravovať, a samozrejme si môžeme stiahnuť celý bibliografický súbor, respektíve prílohu k danej referencii.

Funkciou ktorá by nemala chýbať pri žiadnom systéme ktorý obsahuje databázu s ktorou sa pracuje je vyhľadávanie. V prvom rade si treba vybrať, aký typ vstupu vyhľadávame. Samozrejme je možnosť zvoliť aj vyhľadávanie spomedzi všetkých typov vstupov. Nasledovne sa nám zobrazí možnosť hľadania podľa konkrétnej premennej. Tato možnosť sa generuje dynamicky podľa zvolenej kategórie v prvej časti. Napríklad po zvolení Article nám doplní pre vyhľadávanie najčastejšie typy premenných, ktoré sú kluce, title, author, journal a year. Nasledovne stačí vyplniť už len kľúčové slovo ktoré ma byť hľadané. Samotne vyhľadávanie používa takzvané FULLTEXTové vyhľadávanie. Tento systém podporuje vyhľadávanie časti textu v bunkách, nielen porovnávanie či sa obsah bunky zhoduje celým obsahom. Je nutné pred použitím tejto techniky upraviť tabuľku nasledovne.

```
mysql_query("ALTER TABLE bc_vstupy ADD FULLTEXT vyhladavanie ('kluce`,`title`,`author`,`year`,`publisher`,`journal`,`organization`,`ISBN`"));
```

Týmto bude pridaný do databázy index s názvom vyhladavanie ktorý umožňuje vyhľadávanie nad stĺpcami ktorých obsah je buď varchar alebo text Pre veľké dátové množiny je omnoho rýchlejšie nahrávanie do tabuliek, v ktorej nie sú vytvorené žiadne FULLTEXT indexy. Preto pridávame samotný index iba priamo pred vyhľadávaním a odstraňujeme ho hneď po ukončení vyhľadávania.

```
mysql_query("ALTER TABLE bc_vstupy DROP INDEX vyhľadavanie");
```

Nasledovne je možné používať vo vyhľadávaní funkcie MATCH() a AGAINST().

MATCH(zoznam stĺpcov) AGAINST (výraz)

MySQL hľadá výraz nad stĺpcami zoznam stĺpcov. Parameter výraz je ľubovoľný textový reťazec. Ak sú v parametri zoznam stĺpcov uvedené stĺpce tabuľky, nad ktorými nie je vytvorený fulltext index, MySQL hlási chybu.

MATCH(zoznam stĺpcov) AGAINST (výraz IN BOOLEAN MODE)

Pracuje skoro rovnako ako predchádzajúci príkaz, avšak vďaka modifikátoru IN BOOLEAN MODE presnejšie určujeme podmienky vyhľadávania. Príkaz je dostupný ale až od MySQL 4.0.1.

Modifikátor IN BOOLEAN MODE nám umožňuje zadávať kritéria vyhľadávania. Ako prvé je nutne pripomenúť, že ak je uvedený modifikátor IN BOOLEAN MODE, tak SELECT nám nevráti záznamy zoradené podľa score. To však môžeme dosiahnuť príkazom ORDER BY. Skôr ako si vysvetlíme operátory špecifikujúce kritérium vyhľadávania, je nutne pripomenúť, že ak používame modifikátor IN BOOLEAN MODE, nemusíme mať vytvorený FULLTEXT index, avšak vyhľadávanie je potom pomalšie.

(žiaden)

Ak nie je uvedený žiaden operátor (plus, mínus), tak výsledkom vyhľadávania sú záznamy, ktoré obsahujú aspoň jedno z uvedených slov.

+ (plus)

Ak operátorom + začína hľadaný výraz, výsledkom vyhľadávania sú všetky riadky, ktoré musia obsahovať tento výraz.

- (mínus)

Operátor - pred hľadaným výrazom značí, že výsledkom vyhľadávania budú tie riadky, v ktorých sa toto slovo nesmie vyskytovať.

< (menšie), > (väčšie)

Tieto dva operátory slúžia na zvýšenie (>) alebo zníženie (<) score. Najlepšie to pochopíte z príkladu uvedenom nižšie.

( ) (zátvorky)

Slúžia na zoskupovanie a vytvorenie zložitejších podmienok.

~ (tilda)

Je to operátor negácie. Všetky výrazy začínajúce týmto znakom budú mať najnižšie score, ale i tak to score nebude rovné nule. To znamená, že záznamy budú vrátené. Jednoducho, je to slabšia odroda operátora - (mínus).

\* (hviezdička)

Ako jediný zo všetkých operátorov je pridávaný na koniec hľadaného výrazu. Slúži ako náhrada ľubovoľných znakov.

" (úvodzovky)

Výsledkom vyhľadávania výrazu uzavretého dvojitémi úvodzovkami sú tie riadky, ktoré obsahujú presne takýto výraz.

Ako priamu implementáciu v kóde môžeme uviesť vyhľadávanie spomedzi referencii.

```
$sql = "SELECT * FROM bc_vstupy  
WHERE MATCH(`.addslashes($_POST['premenna']).`)  
AGAINST(`.addslashes ($_POST['retazec']).` IN BOOLEAN MODE)";
```

Tento dopyt vyhľadáva zadaný reťazec v stĺpci s prislúchajúcou premenou poslanou cez (*\$\_POST['premenna']*). Pomocou funkcie pridáme spätné lomítka pred znaky (,), ("), (\) čím z nich vytvára znaky bez vlastnosti.

A aký by to bol používateľský príjemný program keby tu nebola možnosť vytvoriť si vlastný bibliografický súbor. Pri tejto možnosti je vytvorený dočasný súbor. Tento súbor je otvorený neustále počas práce a kedykoľvek je možné pridať úplne nový vstup alebo po vyhľadaní v databáze pridať nájdený vstup. Samozrejme používateľ si môže kedykoľvek aktuálny súbor stiahnuť na svoj počítač.

Všetky vyššie spomenuté vlastnosti a možnosti sú k dispozícii pre používateľa aj keď si zvolí upravovanie nejakého súboru. Jediným rozdielom je umiestnenie súboru v zložke `\temp` z ktorej je nasledovne otvorený na úpravu. Samozrejme všetky tieto možnosti sú prístupné iba prihláseným používateľom.



## 4. Konečná implementácia kódu

Prvým krokom realizácie bolo programovanie databázy. Jej schéma bola už popísaná. Na základe vytvorenej databázy mohlo nasledovať programovanie časti pre administrátora. Výhodou programovanie administrátorskej časti je fakt, že veľká časť kódu bude využitá nasledovne v sekcii pre používateľa. V sekcii pre administrátora nám stačí zobrazit' všetky informácie v jednoduchých tabuľkách ktoré sú prehľadné a jednoducho použiteľné. Je nutne zabezpečiť aby administrátor mohol pracovať zo všetkými údajmi a zároveň ich upravovať bez toho aby musel vstupovať priamo do prostredia databázy. Ide o Tabuľku používateľov, bibliografie a skupín. Samotný prístup do prostredia pre administrátor je zabezpečený heslom.

Druhý krok je programovanie prostredia pre používateľa. Keďže už existuje administrátorské prostredie, máme uľahčenú prácu. Väčšina funkcií už existuje. Vyžaduje sa len grafické spracovanie ktoré uľahčí používateľovi prácu. Postup pri práci je nasledovný. Funkčné prostredie a grafické prostredie sú programované separátne. Následne sú po naprogramovaní a dokončení funkčného prostredia spojené. V grafickom editore bol navrhnutý design stránky rozstrihaný na časti ktoré boli opätovne v CSS spojené do celku. Samozrejmosťou dol'ohľad na funkčnosť v rôznych prehliadačoch.

## 5. Bezpečnosť

Pri internetovej stránke ktoré majú dynamicky obsah, alebo pracujú z databázou vždy existuje hrozba ohrozenia bezpečnosti. Môže ísť o cielený útok alebo o jednoduchú chybu používateľa, ktorá môže systému ublížiť. Základne hrozby sú SQL injection a XSS. Predtým ako sa budeme venovať cieleným útokom pomocou SQL injection a XSS, by sme sa mali zamyslieť nad bezpečnosťou na úrovni bežného používateľa. Je nutnosťou zabezpečiť aby v miestach kde ma byť číselná hodnota mohla byť zadaná pravé číselná hodnota. A naopak v textových poliach len text. Základňou funkciou pre kontrolu vstupov bola dlhu dobu funkcia `ereg()`. Ale podľa najnovších štandardov bola tato funkcia vylúčená a prehlásená za nedôveryhodnú. Ako náhrada bola zvolená funkcia `preg_match()`. Ma síce komplikovanejšiu syntax, ide ale o bezpečnejšiu a funkčnejšiu verziu kontroly obsahu reťazca. Ako vzor správneho použitia tejto funkcie pri kontrole vstupu by som použil funkciu pre kontrolu zadaného emailu pri registrácii. Keďže `Abc\@def@example.com`, `customer/department=shipping@example.com` a `!def!xyz%abc@example.com` sú všetko korektné emailové adresy, je nutné sa hlbšie zamyslieť nad kontrolou.

```
function reg_adresa($email){
    $isValid = true;
    $atIndex = strrpos($email, "@");
    if (is_bool($atIndex) && !$atIndex){$isValid = false;}
    else{
        $domain = substr($email, $atIndex+1);
        $local = substr($email, 0, $atIndex);
        $localLen = strlen($local);
        $domainLen = strlen($domain);
        if ($localLen < 1 || $localLen > 64){$isValid = false;}
        cast local predstavujuca prvu cast presiahla moznu velkost
        else if ($domainLen < 1 || $domainLen > 255){$isValid = false;}
        cast domeny presiahla maximalnu velkost
        else if ($local[0] == '.' || $local[$localLen-1] == '.'){$isValid = false;}
        lokalna cast zacina alebo konci s bodkou '.'
    }
}
```

```

else if (preg_match('/\|\.\|./', $local)){ $IsValid = false;}
v lokalnej cast sa nachadzaju dve bodky
else if (!preg_match('/^[A-Za-z0-9\|-.]+$', $domain)){ $IsValid = false;}
nevalidne znaky v sekcii domeny
else if (preg_match('/\|\.\|./', $domain)){ $IsValid = false;}
domena obsahuje dve podebe iduce bodky
else if (!preg_match('/^(\\|\\|/[A-Za-z0-9!#%&`_=$\'+?^{|~.-])+$/',
str_replace("\\|\\|", "", $local))){
pokial niesu specialne symboli komentovane su nevalidne
if (!preg_match('/^"(\\|\\|/[^\"])+"$',
str_replace("\\|\\|", "", $local))){ $IsValid = false;}
}

```

Tato funkcia zabezpečuje plnohodnotnú kontrolu vloženéj emailovej adresy.

Tato úroveň kontroly zabezpečuje alternatívu vloženia škodlivého kódu do textovej formy pomocou vytvorenia nového súboru ktorú by si nasledovne používateľ nevedome stiahol zo servera pod dojmom nájdenia bibliografie.

## 5.1. SQL Injection

Ďalšia vrstva ochrany je venovaná útokom pod názvom SQL injection. Ide o metódu ktorá využíva mierne nedostatky v kontrole vstupných dát k tomu aký hacker podstrčil vlastné dáta namiesto očakávaných informácii. Ďalším využitím je prevzatie kontroly nad databázou nad rámec možnosti, ktoré by mal ako návštevník alebo používateľ mať. Pri prevzatí kontroly nad databázou hovoríme o absolútnej kontrole, pri ktorej hacker môže databázu kopírovať, upravovať, mazať a čítať. Týmto spôsobom je možné sa dostať ku citlivým dátam, ako bolo preukázané niekoľko mesiacov dozadu pri hacknutí známeho slovenského portálu. Útočník získal kontrolu nad celou databázou a zverejnil niekoľko hesiel medzi ktorými bolo aj heslo administrátora. Preto by sme sa mali držať pravidla „nedôverovať žiadnym dátam od nikoho“. Častým dôvodom s SQL injection sa dostávame keď sa adekvátne nestaráme o dáta ktoré získavame z adresy URL. Keďže je tato metóda veľmi rozšírená, takisto je aj často napádaná. Napríklad ak posielame v URL adrese číselnú hodnotu a hacker ju prepíše na znak, skript nevie reagovať na

vzniknutú situáciu a výpise chybové hlásenie. Vo svete hackerov platí, že chybové hlásenie je rovnako užitočné pre hackera ako je užitočné pre programátora pri vývoji a používateľa pri práci. Preto je nutnosťou použiť niektoré z nasledujúcich opatrení.

- Pri zobrazovaní html kódu, v nevyhnutných prípadoch, ktorý sa nemá vykonať môžeme použiť funkciu `htmlspecialchars()`, ktorá mení znaky na neškodné HTML.
- Pomocou funkcie `escapeshellcmd()` potlačíme celý význam všetkých metaznakov. Zabránilo tým spúšťaniu ľubovoľných príkazov. Používa sa najčastejšie ak posielame funkciám ako `exec()` alebo `system()` dáta.
- Predtým ako ukladáme dáta do databázy je nutne použiť funkciu `addslashes()`. Tato funkcia potláča význam všetkých špeciálnych znakov. Predchádza sa tým veľkému množstvu problémov. Pri spätnom výpise z databázy použijeme funkciu `stripslashes()` ktorá pridane lomítka odstráni a údaje sú v pôvodnom stave. Jediným rozdielom je fakt, že v databáze nespôsobili žiadne problémy.
- Pri potrebe zobraziť v php časti kus HTML kódu ktorý sa môže vykonať a spôsobovať bezpečnostné riziko, je nutne použiť funkciu `strip_tags()`. Ide o funkciu na odstránenie všetkých tagov. Pri tejto funkcii existuje a nepovinný atribút ktorý povoľuje zapísať niektoré tagy. Vďaka tomu zabránilo prípadnému vykonaniu skriptu ktorý by sa mohol vykonať.
- Ďalším riešením potenciálneho útoku je použitie funkcie `magic_quotes_gpc` a `magic_quotes_runtime`. Prvá menovaná je funkcia takzvaných magických úvodzoviek. Tato funkcia všetkým údajom prichádzajúcim z GET, POST alebo COOKIES vkladá lomítka, čím ruší všetky špeciálne znaky. Pri používaní sa odporúča vytvoriť funkciu `addslashes()`, ktorá samostatne zisti či je funkcia `magic_quotes_gpc` zapnutá alebo vypnutá a podľa toho použije alebo nepoužije funkciu `addslashes()`. Druhá funkcia, `magic_quotes_runtime`, sa používa na ošetrovanie dát z databázy.
- Pri práci s číselnými hodnotami, ktoré napríklad posielame cez URL by sme ako prvé mali testovať hodnotu pomocou napríklad `intval()` alebo `is_numeric()`. Prvá funkcia slúži na kontrolu, či sa dá poslaná hodnota upraviť na číslo. Druhá kontroluje, či poslaná hodnota je priamo číslo.

## 5.2. XSS

Cross-site scripting pracuje na základe nájdenia slabého miesta vo webovom priestore do ktorého vloží škodlivý kód. Najjednoduchším príkladom je vyhľadávanie na stránke. Ak pri vyhľadávaní posielame informácie cez URL, útočník môže vložiť namiesto originálnej požiadavky poslať skript so škodlivým kódom. Napríklad namiesto

*/hladat.php?polozka = Article* použije útočník

*/hladat.php?polozka = <script src="zdrojovyyweb.sk/utok.js"></script>*

Vďaka tomu môže útočník vykonať ľubovoľný javascript ku ktorému vedie cesta. Bežne využitie v praxi je na prihlasovanie do účtov existujúcich používateľov.

## 6. SEO

V tejto časti by sme sa radi venovali prvku pod názvom SEO. Skratka znamená Search Engine Optimalization, čiže optimalizácia pre prehliadače. Pozícia nasej stránky vo vyhľadávacích stránkach a systémoch je veľmi dôležitá. Veľké percento používateľov nepozná často presný názov alebo adresu stránky ktorú hľadajú.

Najčastejšie hľadajú pomoc na vyhľadávacích stránkach ako sú napríklad google, yahoo alebo msn. Ak naša stránka nie je na prvej strane po vyhľadaní názvu alebo kľúčového slova šanca, že nás nový používateľ navštíví, sa radikálne znižuje. Metódy na vyhľadávanie s ktorými pracujú stránky ako google alebo yahoo sú vytvárané ich inžiniermi, a menia sa neuveriteľne rýchlo a často. Nikto z bežných tvorcov web stránok nepozná presne algoritmy pri hľadaní, ale pozná niekoľko pravidiel ktoré treba dodržiavať.

- Používanie kľúčových slov

Kľúčové slova sú frázy nachádzajúce sa v tele stránky, ktoré by mali byť nájdené pri vyhľadávaní. Napríklad kľúčové slova pre tenisový klub v Martine by mali byť: tenis, klub, šport, tenisklub, Martin. V našom prípade by bolo správne použiť kľúčové slová ako: bibliografia, referencie, SVN, literatúra a podobne. Tieto slova musia byť použité v tele stránky. Google má nástroje na vytvorenie efektívnych kľúčových slov nazvaných Google AdWords. Ide o voľne dostupnú aplikáciu.

- Názov domény

Názov domény je takisto veľmi dôležitý. Pred vytvorením web stránky by sme sa mali zamyslieť nad jej obsahom a veľmi dômyselné vybrať meno. Meno by malo čo najlepšie vystihovať stránku. Ak sa môžeme opäť vrátiť k vzoru tenisového klubu, ich stránky by mali mať mena ako: tenisklub.sk tenis-klub.sk. V našom prípade bol zvolený názov bibSoft, keďže sa jedná o prácu so súborami s koncovkou .bib. takisto vytvorenie domény a názvom ako napríklad: [www.tenisclub.com/club?id=423&cat=44](http://www.tenisclub.com/club?id=423&cat=44) nie je odporúčané a ani priateľské s vyhľadávačmi. Správnu metódou je písanie korektných adries ako napríklad: [www.tenisclub.com/profile/roger-federer](http://www.tenisclub.com/profile/roger-federer).

- Vytvorenie súboru .htaccess  
 V tomto súbore definujeme, presmerovanie alebo prepísanie domény. Základným použitím pre SEO je presmerovanie www stránok na non www stránky. Príkladom môže opäť byť modelová situácia. Ak napíšem do prehliadača www.tenisklub.sk prehliadač ma presmeruje na tenisklub.sk. Rovnako ak by náš projekt bol hostovaný napríklad na www.bibsoft.sk presmerovanie by bolo uskutočnené na bibsoft.sk. Google považuje tieto dve adresy za totožné. V inom prípade by ich považoval za duplicitné a znížil by hodnotenie. Takisto aj tu existujú nástroje ktoré kontrolujú náš prístup cez htaccess.
- Vytvorenie robots.txt  
 Tento textový súbor špecifikuje ktoré dáta a priečinky môžu byť prístupné pre vyhľadávacie programy a roboty. Najčastejšie upresňujú priečinky so systémovými alebo súkromnými dátami, fotkami a podobne. Na validáciu sú súboru robots.txt existujú opäť softvéry ktoré sú voľne dostupne.
- Vytvorenie sitemap  
 Ide o vytvorenie súboru, ktorý pomáha vyhľadávacím systémom navštíviť našu stránku. Niekedy chýbajú priame linky. Súbor sitemap obsahuje všetky stránky a podstránky. Je jednoduchšie prechádzať cez internetovú stránku ktorá má vytvorený sitemap. Existujú stránky ktoré efektívne vytvárajú sitemap.
- Názov stránky  
 Názov je situovaný v *< head >* elemente. Tento text by mal jasne špecifikovať celý obsah stránky v niekoľkých stručných slovách. Názvy stránok a podstránok by sa mali odlišovať. Názov sa nachádza aj na vrchu záložky internetového prehliadača.
- Meta tagy  
 Kľúčové slova: v tomto tagu definujeme všetky kľúčové slova. Tento text je zobrazený vyhľadávacím systémom, keď je zobrazený medzi nájdenými.  
 Napríklad:  

```
<meta name="description" content="Stranka o tenisovom klube v Martine ">
```

```
<meta name="description" content="Stranka pre pracu s bibliografiou a referenciami. ">
```

 Jazyk: Definícia jazyku použitého na nasej stránke.  
 Napríklad: 

```
<meta name="Content-language" content="sk" />
```

Autor: priestor pre spomenutie autora.

Například: `<meta name="author" content="Marek Luley" />`

Roboty: Definujeme či stránka je indexovaná alebo pozorovaná. Používa sa najčastejšie *index*, *noindex*, *follow*, *nofollow*

Například: `<meta name="robots" content="index,follow" />`

- HTML kód

HTML kód by mal byť absolútne validný. Mal by prejsť kontrolou validity W3C po všetkých stránkach. Validnosť CSS takisto veľmi pomáha. Samotne W3C ponuka na svojich stránkach kontrolu validnosti. (SEO chat. 2010.)

Toto je len niekoľko základných podmienok ktoré by mala SEO prístupná stránka spĺňať



## 7. Spustenie systému

Posledná fáza pri systémovej realizácii bol test funkčnosti. Po vytvorení administrátora sme zaregistrovali do systému niekoľko používateľov. Niektorí používatelia vytvorili skupiny a pridali do nich kolegov. Nasledoval test načítania súboru. Každý používateľ uložil niekoľko referencií a bibliografických súborov na sever. Ďalšia časť kontroly bola zameraná na vyhľadávanie. Vyhľadávanie v rámci skupiny a jednotlivca pracovalo správne. Posledným ktorom bolo vytvorenie vlastných bibliografických súborov a ich upravovanie.

Keďže bola databáza úspešne naplnená či už používateľmi alebo dátami, mohli sme prejsť na kontrol funkcií administratíva. Po zaregistrovaní administrátora boli všetky dáta prístupné a manipulácia s nimi fungovala. Či už išlo o prezeranie používateľov, alebo ich vymazávanie, alebo pracú s referenciami.

Po kontrole všetkých funkcií administrátora sme prešli na kontrolu voči chybám používateľov. Cielene sme zadávali do všetkých priestorov nesprávne dáta, či už text do číselných polí alebo naopak.

Po kontrole nepozorného používateľa sme prešli na kontrolu pri útoku hackera. Vyhladli sme si slabšie miesto kódu, v našom prípade išlo o posielanie ID hodnoty cez URL, a cielene sme sa snažili získať od databázy reakciu, ktorá by naznačovala slabé miesto. Po pridaní apostrofa pre kontrolu či prijímame len čísla a základnom pokuse obísť tuto kontrolu sme sa k žiadnym dátam nedostali. Takisto pokus o vloženie škodlivého JavaScript kódu zlyhalo.

Poslednou kontrolou bola kontrola validity a kvality SEO spracovania. Keďže stránka prešla kontrolou s dostatočným uspokojením, môže byť prehlásená za hotovú.

## 8. Záver

V úvode práce boli jasne stanovené ciele. Ich naplnenie bolo úspešné. Prvým krokom bola komplexná analýza problému, na základe ktorej bola navrhnutá databáza. Vďaka dobrým základom bolo programovanie veľmi úspešné, o čom sa dá veľmi jednoducho presvedčiť. Funkčné prevedenie programu nájdete na <http://www.st.fmph.uniba.sk/~luley1/BC/> . Stránka bude použitá na prezentačné účely. Preto sa môže stať, že v budúcnosti bude jej obsah zmenený alebo odstránený. Avšak ak by práce úspešne pokračovali, je možné, že nová verzia webu bude existovať na vlastnej doméne pre voľne používanie celej verejnosti.

Vďaka všeobecnému prístupu k programovaniu je možné funkcie kedykoľvek rozšíriť alebo upraviť bez väčšej námahy. V pláne je rozšírenie systému o OpenID a možnosť pracovať aj s informáciami ktoré boli uložené ako záložka v prehliadači. Nebude chýbať ani jednoduchý systém pre pracú v skupine ktorý bude informovať o pokrokoch v práci kolektívu.

V aktuálnej podobe je stránka určená pre používateľov, ktorý si pomocou nej môžu ukladať svoje bibliografické súbory a zdieľať ich s kolegami. Po uložení bibliografie je sprístupnená možnosť vyhľadávania. Vyhľadávanie môže vyhľadávať vo vlastnej bibliografii, alebo v bibliografii používateľov, ktorý sú v rovnakej skupine. Umožnené je aj vytváranie nových bibliografických súborov. Web je teda celkovo funkčný a prístupný.

## **Použitá literatúra**

Patashnik, O. 1988. BIBTEXing: [online]. Dostupné na internete: <http://www.ctan.org/get/biblio/bibtex/contrib/doc/btxdoc.pdf>

Open Directory - Computers: Software: Typesetting: TeX: BibTeX. 2009. [online]. Dostupné na internete: <http://www.dmoz.org/Computers/Software/Typesetting/TeX/BibTeX/>

Roberts, A. 2009. Getting to Grips with Latex - Bibliography Management - Latex Tutorials by Andrew Roberts. [online]. Dostupné na internete: <http://www.dmoz.org/Computers/Software/Typesetting/TeX/BibTeX/>

Katuščák, D. 2006. Chrestomatia k základom bibliografie. [online]. Dostupné na internete: <http://www.pulib.sk/kis/katuscak/BGCela.pdf>

Bibliographic/Software and Standards Information - OpenOffice.org Wiki. [online]. 2010. Dostupné na internete: [http://wiki.services.openoffice.org/wiki/Bibliographic\\_Software\\_and\\_Standards\\_Information](http://wiki.services.openoffice.org/wiki/Bibliographic_Software_and_Standards_Information)

Comparison of Free Bibliographic Managers. 2008. [online]. 2009 Dostupné na internete: <http://mahbub.wordpress.com/2007/03/04/comparison-of-free-bibliographic-managers/>

SEO chat - optimalizácia pre vyhľadávače. 2010. [online]. 2010. Dostupné na internete: <http://www.seochat.sk/>

Jacobsen, C. 1999. The BibTeX Format. [online]: 2006. Dostupné na internete: <http://www.fb10.uni-bremen.de/anglistik/langpro/bibliographies/jacobsen-bibtex.html>

Suchý, P. 2008. SEO.MONITORUJEM.SK. [online]: 2010 Dostupné na internete:

<http://seo.monitorujem.sk/>

CiteULike: Everyone's library 2010. [online]: 2010 Dostupné na internete:

<http://www.citeulike.org/>

I, Librarian - PDF manager for scientific articles. 2010. [online]: Dostupné na internete:

<http://www.bioinformatics.org/librarian/>

EndNote - Product Information 2010. . [online]: Dostupné na internete:

<http://www.endnote.com/eninfo.asp>