

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

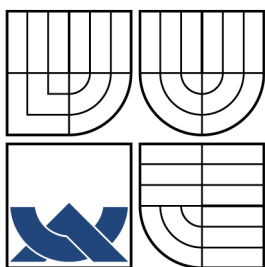
METODIKA SEO - TESTOVÁNÍ A ANALÝZA

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

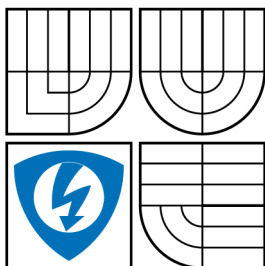
AUTOR PRÁCE
AUTHOR

BC. MARTIN KRUPA

BRNO 2010



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ



FACULTY OF ELECTRICAL ENGINEERING AND
COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

METODIKA SEO - TESTOVÁNÍ A ANALÝZA

METHODOLOGY SEO - TESTING AND ANALYSIS

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

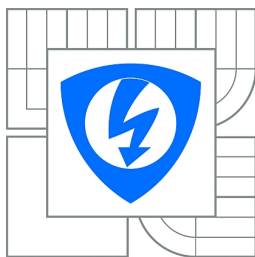
AUTOR PRÁCE
AUTHOR

BC. MARTIN KRUPA

VEDOUcí PRÁCE
SUPERVISOR

ING. PETRA LAMBERTOVÁ

BRNO 2010



**VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ**

**Fakulta elektrotechniky
a komunikačních technologií**

Ústav telekomunikací

Diplomová práce

magisterský navazující studijní obor
Telekomunikační a informační technika

Student: Bc. Martin Krupa

ID: 78621

Ročník: 2

Akademický rok: 2009/2010

NÁZEV TÉMATU:

Metodika SEO - testování a analýza

POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s metodikou SEO (Search Engine Optimization), popište její parametry a měřitelné hodnoty. Vytvořte portál, který by tyto parametry testoval na různých stránkách zadaných uživatelem. Uživatelé budou také zobrazeny návrhy na možné opravy chyb a dále porovnávána úspěšnost stránek před a po úpravách. Analyzujte vliv optimalizace SEO na návštěvnost stránek a jejich umístění ve vyhledávačích.

DOPORUČENÁ LITERATURA:

[1] KUBÍČEK, Michal. Velký průvodce SEO. [s.l.] : Computer Press, srpen 2008. 320 s. ISBN 978-80-251-2195-5.

[2] SMIČKA, Radim. Optimalizace pro vyhledávače - SEO. Dubany : Jaroslava Smičková, 2004. 120 s. ISBN 80-239-2961-5.

Termín zadání: 29.1.2010

Termín odevzdání: 26.5.2010

Vedoucí práce: Ing. Petra Lambertová

prof. Ing. Kamil Vrba, CSc.
Předseda oborové rady

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

SEO - Search Engine Optimization, alebo optimalizácia pre vyhľadávače, určuje spôsoby a metódy, ako vytvoriť web stránku tak, aby ju vyhľadávač zaradil do výsledkov vyhľadávania sám. Jeho veľkou výhodou je trvácnosť výsledkov a z toho vyplývajúca vysoká návratnosť vložených investícií. Práca najprv hovorí o základných a najdôležitejších parametroch SEO. Navrhne, ktoré parametre je najvhodnejšie testovať a najlepší spôsob ich testovania. Väčšina majiteľov webových stránok často ani nevie, že ich stránka je z hľadiska SEO absolútne neprístupná. Pomocou navrhnutého portálu si môžu svoje stránky rýchlo a jednoducho otestovať a zistiť, kde sú ich hlavné nedostatky. Po otestovaní parametra SEO portál určí spôsob opravy v prípade chyby. Dôležitou vlastnosťou portálu je jednoduchosť. Užívateľ zadá adresu jeho stránky a zobrazia sa mu prehľadne usporiadané výsledky.

KĹÚČOVÉ SLOVÁ

SEO, Optimalizácia pre vyhľadávače, parametre SEO, výsledky vyhľadávania, SEO portál, testovanie SEO

ABSTRACT

SEO - Search Engine Optimization, determine ways and methods to create a web page so that search engine can incorporate it into the search results. Its great advantage is the durability of results and it results to high return of an investment. This work talks about the basic and most important SEO parameters. It propose which parameters are most important to be tested and best way of testing. Most of publishers often do not know that their sites are in terms of SEO totally inaccessible. With portal they can make their sites quickly and easily tested and find out where are their main weaknesses. When parameter has been tested SEO Portal determines how to correct errors. An important feature of my portal is simplicity. After entering the address of the site they can see clearly arranged results.

KEYWORDS

SEO, Search engine optimization, SEO parameters, search engine results, SEO portal, SEO testing

KRUPA, M. *Metodika SEO - testování a analýza*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2010. 94 s. Vedoucí diplomové práce Ing. Petra Lambertová.

PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Metodika SEO - testování a analýza“ jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne

.....

(podpis autora)

PodĎakovanie

Ďakujem vedúcej mojej diplomovej práce Ing. Petre Lambertovej za účinnú metodickú, pedagogickú a odbornú pomoc a ďalšie cenné rady pri jej tvorbe a spracovaní.

V Brne dňa 15. mája 2010

.....
podpis autora

OBSAH

Úvod	13
1 Metodika SEO	15
1.1 SEM - Search Engine Marketing	15
1.1.1 PPC systémy a kontextové odkazy	15
1.1.2 SEO - Search Engine Optimization	15
1.1.3 Hlavné rozdiely medzi SEO a sponzorovanými odkazmi	16
1.1.4 Katalóg vs. fulltextový vyhľadávač	17
1.1.5 Princíp fungovania vyhľadávača	18
1.2 On Page SEO faktory	20
1.2.1 Titulok stránky	20
1.2.2 Hlavné nadpisy stránky	20
1.2.3 Meta tagy Keywords a Description	21
1.2.4 Valídnošť stránky	21
1.2.5 Prístupnosť stránky	22
1.2.6 Veľkosť stránky a doba načítavania	23
1.3 Off Page SEO faktory	23
1.3.1 Rank stránok	23
1.3.2 PageRank	24
1.3.3 S-Rank	25
1.3.4 Alexa rank	25
1.3.5 Spätné odkazy	25
1.4 Kľúčové slová na stránke - keywords	26
1.4.1 Kam umiestniť kľúčové slová	26
1.4.2 Long tail kľúčové slovo	27
1.5 Ostatné SEO faktory	27
1.5.1 Roboti a súbor robots.txt	27
1.5.2 Univerzálna mapa stránok - sitemap.xml	28
1.5.3 Hlavné kritériá ovplyvňujúce umiestnenie stránky na SERP	29
2 Návrh portálu na testovanie a analýzu SEO	30
2.1 Testované parametre stránok	30
2.1.1 Testované On Page SEO faktory	30
2.1.2 Testované Off Page SEO faktory	31
2.1.3 Ostatné testované SEO faktory	32
2.1.4 Testovanie kľúčových slov - keywords	32

3	Realizácia testovacích skriptov	33
3.1	On page SEO testovací skript	33
3.1.1	Vstupný formulár	33
3.1.2	Regulárne výrazy	33
3.1.3	Test vstupných parametrov	34
3.1.4	Test title tagu stránky	35
3.1.5	Test meta tagov stránky	36
3.1.6	Test hlavného nadpisu stránky	38
3.1.7	Test veľkosti zdrojového kódu stránky	38
3.1.8	Test štruktúrovania textu stránky	39
3.1.9	Test validnosti stránky	39
3.1.10	Test štýlovania stránky	40
3.1.11	Test alternatívnych textov obrázkov na stránke	40
3.1.12	Test použitia rámcov na stránke	41
3.1.13	Test použitia tabuliek na stránke	41
3.1.14	Test súboru sitemap.xml	42
3.1.15	Test súboru robots.txt	42
3.1.16	Zistenie času testu	42
3.2	Off page SEO testovací skript	43
3.2.1	Vstupný formulár	43
3.2.2	Google page rank	44
3.2.3	Seznam rank	44
3.2.4	Alexa rank	45
3.2.5	Alexa spätné odkazy	46
3.2.6	Google spätné odkazy	46
3.2.7	Google zaindexované stránky	47
3.2.8	Pozícia stránky na google	47
3.2.9	Pozícia stránky na seznam	49
3.2.10	Pozícia stránky na bing	50
3.3	Skript testujúci kľúčové slová	51
3.3.1	Vstupný formulár	51
3.3.2	Test frekvencie kľúčového slova	51
3.3.3	Test výskytu kľúčovej frázy v rôznych častiach stránky	56
4	Realizácia portálu na testovanie a analýzu SEO	59
4.1	Príprava databázy	60
4.1.1	Databázové tabuľky	60
4.2	Užívateľské transakcie	61
4.2.1	Registrácia užívateľov	62

4.2.2	Prihlasovanie užívateľov	64
4.2.3	Zmena aktuálne testovanej adresy webstránky	65
4.2.4	Úprava užívateľských účtov	67
4.2.5	Pripomenutie zabudnutého hesla	68
4.2.6	Odhlásenie užívateľa	68
4.3	Zloženie webovej stránky	69
4.3.1	Hlavička stránky - header.php	69
4.3.2	Obsah stránky	72
4.3.3	Päta stránky - inc/footer.php	72
4.4	Testovanie webových stránok neregistrovaného užívateľa	72
4.4.1	On page SEO test	72
4.4.2	Off page SEO test	74
4.4.3	Test kľúčových slov stránky	76
4.5	Testovanie webových stránok registrovaného užívateľa	77
4.5.1	On page SEO test	78
4.5.2	Off page SEO test	78
4.5.3	Test kľúčových slov stránky	79
4.6	Zobrazenie všetkých výsledkov testov konkrétnej stránky	79
4.6.1	Zobrazenie všetkých výsledkov on page SEO testov konkrétnej stránky	81
4.6.2	Zobrazenie všetkých výsledkov off page SEO testov konkrétnej stránky	82
4.6.3	Zobrazenie všetkých výsledkov testov kľúčových slov konkrétnej stránky	83
4.7	Odstraňovanie výsledkov testov	85
4.7.1	Odstraňovanie všetkých výsledkov testov konkrétnej stránky	85
4.7.2	Odstraňovanie špecifického výsledku testu konkrétnej stránky	86
4.8	Vývojový diagram portálu na testovanie SEO faktorov	87
5	Analýza vplyvu optimalizácie SEO na návštevnosť stránok a ich umiestenie vo vyhľadávačoch	89
6	Záver	91
	Literatúra	93
	Zoznam symbolov, veličín a skratiek	95

ZOZNAM OBRÁZKOV

1.1	Príklad rozdielu medzi platenými odkazmi a SEO vo vyhľadávači Google	16
1.2	Príklad katalógu z www.seznam.cz	18
1.3	Podiel vyhľadávačov v ČR na konci roka 2006. Zdroj www.toplist.cz	19
3.1	Vzhľad vstupného formulára na testovanie on page SEO faktorov	33
3.2	Vzhľad vstupného formulára na testovanie off page SEO faktorov	43
3.3	Popularita domény www.vutbr.cz zistená pomocou portálu www.alexa.com	45
3.4	Počet spätných odkazov na stránku www.sme.sk vo vyhľadávači google	46
3.5	Počet zaindexovaných stránok domény www.sme.sk vo vyhľadávači google	47
3.6	SERP vyhľadávača google po zadaní modelovej linky do internetového prehliadača	48
3.7	Vzhľad vstupného formulára na testy kľúčových slov zadanej stránky	51
3.8	Príklad výpisu prvkov poľa s počtom ich opakovaní v poli	55
4.1	Príklad SQL tabuľky <code>seo_users</code> z prostredia phpMyAdmin	61
4.2	Vzhľad registračného formulára	62
4.3	Vzhľad prihlasovacieho formulára	64
4.4	Vzhľad formulára na zmenu aktuálne testovanej webstránky	65
4.5	Grafické znázornenie zloženia stránky <code>index.php</code>	69
4.6	Ukážka časti stránky zobrazujúcej informácie o aktuálne prihlásenom užívateľovi	70
4.7	Vzhľad užívateľského menu neprihláseného užívateľa	71
4.8	Vzhľad užívateľského menu prihláseného užívateľa	71
4.9	Príklad výsledku On page SEO testu stránky www.feec.vutbr.cz	74
4.10	Príklad výsledku Off page SEO testu stránky www.feec.vutbr.cz so zadaným kľúčovým slovom <code>feec vutbr</code>	75
4.11	Príklad výsledku testu kľúčových slov stránky www.feec.vutbr.cz so zadaným kľúčovým slovom <code>vut</code>	77
4.12	Príklad tabuľky porovnávajúcej hodnotenia údajov on page SEO faktorov z hlavičky stránky www.feec.vutbr.cz	81
4.13	Príklad tabuľky porovnávajúcej hodnotenia údajov on page SEO faktorov z tela stránky www.feec.vutbr.cz	82
4.14	Príklad tabuľky porovnávajúcej hodnotenia údajov ostatných on page SEO faktorov stránky www.feec.vutbr.cz	82
4.15	Príklad tabuliek porovnávajúcich hodnotenia off page SEO faktorov pre stránku www.feec.vutbr.cz	83
4.16	Príklad tabuliek porovnávajúcich vstupné údaje užívateľa a výsledky hodnotenia výskytu kľúčového slova <code>vut</code> pre stránku www.feec.vutbr.cz	84

4.17	Príklad tabuliek porovnávajúcich výsledky hodnotenia výskytu kľúčového slova vut na špecifických miestach stránky www.feec.vutbr.cz	85
4.18	Vývojový diagram SEO portálu	88
5.1	Grafické zobrazenie vplyvu SEO na návštevnosť stránky a jej pozíciu na SERP	89

ZOZNAM TABULIEK

1.1	Približný percentuálny odhad dôležitosti faktorov ovplyvňujúcich pozície vo vyhľadávачoch	29
-----	---	----

ÚVOD

Internetové vyhľadávače sú jedným z najbežnejších spôsobov hľadania informácií na internete. Ľudí, ktorí denne požívajú vyhľadávač, sú milióny a pre firmy s ich internetovými stránkami predstavujú obrovské množstvo potencionálnych zákazníkov. Napríklad najznámejší svetový vyhľadávač Google má vo svojej databáze viac ako 20 miliárd stránok. Úlohou vyhľadávačov je doručiť relevantný obsah užívateľom, ktorí používajú vyhľadávače.

Zviditeľnením webovej prezentácie vo vyhľadávačoch sa zaoberá Search Engine marketing (SEM). SEM sa delí na dve základné časti, platenú a neplatenú. V platenej časti majiteľ webstránky platí priamo vyhľadávaču za zviditeľnenie svojej webstránky. Platená časť prináša viditeľné výsledky takmer okamžite.

Tento spôsob je ale finančne veľmi nákladný a preto sa čoraz viac majiteľov webstránok orientuje na neplatený spôsob a to je SEO - Search Engine Optimization alebo optimalizácia pre vyhľadávače. SEO určuje spôsoby a metódy ako vytvoriť webstránku tak, aby ju vyhľadávač zaradil do výsledkov vyhľadávania sám. Jeho veľkou výhodou je trvácnosť výsledkov a z toho vyplývajúca vysoká návratnosť vložených investícií. Prvé výsledky sa dostavujú niekedy až pár týždňov od uskutočnenia optimalizácie, jedná sa o zdĺhavý proces, ktorý je ale určite lacnejšou alternatívou ako platená reklama. Vyhľadávače neustále menia svoje algoritmy - hodnotiace programy, ktoré rozhodujú o tom, ktoré stránky budú na popredných miestach vo výsledkoch vyhľadávania pre dané hľadané výrazy. Preto je potrebné upravovať základné pravidlá SEO. Ak chceme mať neustále prehľad v tejto problematike je potrebné sa neustále vzdelávať.

Práca najprv popíše základné a najdôležitejšie parametre SEO. Ďalej bude navrhnuté, ktoré parametre je najdôležitejšie testovať, pretože nie všetky parametre a faktory ovplyvňujúce SEO sú testovateľné. Taktiež navrhne najlepší spôsob ich testovania. Po otestovaní parametra SEO portál navrhne najvhodnejší spôsob opravy v prípade chyby. Bude vytvorený prehľadný systém s návodmi, ktoré umožnia aj užívateľom s menšími znalosťami problematiky SEO jednoduchú a účinnú optimalizáciu stránok. Užívateľ portálu bude mať dve možnosti:

1. Jednoduché testovanie web stránok.
2. Testovanie web stránok spojené s registráciou.

Druhý spôsob je určený pre užívateľov, ktorí chcú vidieť, ako sa prípadná oprava chýb, ktoré našiel portál, prejaví vo výsledkoch vyhľadávania.

Väčšina majiteľov webových stránok často ani nevie, že ich stránka je z hľadiska SEO absolútne neprístupná. Pomocou navrhnutého portálu by si mohli svoje

stránky rýchlo a jednoducho otestovať a zistiť, kde sú ich hlavné nedostatky. Dôležitou vlastnosťou portálu by mala byť jednoduchosť. Užívateľ zadá adresu jeho stránky a zobrazia sa mu prehľadne usporiadané výsledky. Ďalšou vlastnosťou je prenositeľnosť. To znamená, že navrhnutý portál si prípadní užívatelia môžu upraviť podľa svojich predstáv a využívať ho na ich vlastnej doméne.

1 METODIKA SEO

1.1 SEM - Search Engine Marketing

V súčasnosti je rozvoj a používanie internetu veľmi výrazné a prináša so sebou nové formy marketingových "elektronických" techník. Jednou z najdôležitejších pre firmy je tzv. **Search Engine Marketing - SEM** je inak povedané marketing vo vyhľadávачoch. Podstatná vec pre **SEM** je ponúknuť zákazníkovi to, čo naozaj chce. Teda ak váš potencionálny zákazník hľadá cez internetový vyhľadávač napríklad autoservis a zadá do vyhľadávania slovo "*autoservis*", vyhľadávač by mu mal zobrazíť výsledok čo možno najpriaznivejší pre vás - teda odkaz priamo na stránky vašej firmy. **Search Engine Marketing - SEM** sa delí na dve základné časti:

1. Spolupráca s vyhľadávačmi pomocou **platených odkazov** - kontextové odkazy, PPC systémy.
2. Optimalizácia stránok tak, aby ich vyhľadávač sám umiestnil na čo možno najlepšiu pozíciu vo výsledkoch vyhľadávania - **SEO - Search Engine Optimization**.

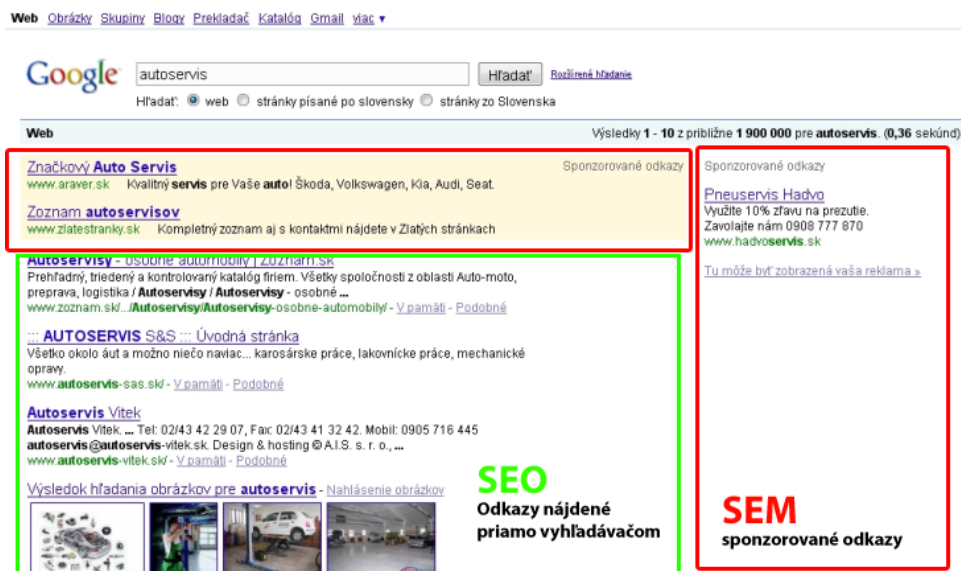
1.1.1 PPC systémy a kontextové odkazy

Ak chcete vaše stránky zviditeľniť ihneď a ste ochotní investovať do reklamy je možné využiť tzv. **PPC systémy**. Inzerent takto registruje svoju stránku vo vyhľadávači, ktorému potom platí nie za zverejnený odkaz, ale až za uskutočnený vstup zákazníka na svoju stránku. Reklamný odkaz môže byť textový alebo grafický. Môžete si nadefinovať ľubovoľný počet kľúčových slov, po zadaní ktorých do vyhľadávača sa vám zobrazí odkaz na vašu stránku v platenej sekcii výsledkov.

1.1.2 SEO - Search Engine Optimization

SEO - Search Engine Optimization alebo tiež **optimalizácia pre vyhľadávače** spočíva v tom, že sa snažíme vytvoriť, upraviť, doplniť a následne propagovať web stránky k lepším umiestneniam v neplatených výsledkoch hľadania vyhľadávačov (vo fulltextovom vyhľadávaní) na kľúčové slová, ktoré sú relevantné pre danú stránku. Tým sa zvyšuje návštevnosť stránok a zároveň konverzný pomer, tj. percento návštevníkov, ktorí uskutočnia na stránke to, čo je jej cieľom. Potreba vytvoriť webové stránky tak, aby boli dobre prístupné vyhľadávačom vznikla najmä kvôli nárastu ich počtu (vyhľadávač google má vo svojej databáze viac ako 20 miliárd stránok). Pre firmu je veľmi podstatné, aby sa vo výsledkoch vyhľadávania, teda

na anglicky nazvanej *SERP* - *search engine result page*, umiestnila na čo najlepších pozíciách. Takto totiž získava potencionálnych zákazníkov. Cieľom SEO je nájdiť informácií. Informácie vždy hľadajú ľudia a vyhľadávač je len nástroj k ich vyhľadaniu.



Obr. 1.1: Príklad rozdielu medzi platenými odkazmi a SEO vo vyhľadávači Google

SEO sa dá rozdeliť na tri základné časti:

1. On-page SEO - optimalizácia stránok ako takých,
2. Off-page SEO - optimalizácia faktorov mimo stránky,
3. Ostatné faktory SEO.

1.1.3 Hlavné rozdiely medzi SEO a sponzorovanými odkazmi

Reakčná doba, kým sa výsledky optimalizácie SEO objavia vo výsledkoch vyhľadávania, je dlhšia ako napríklad pri nasadení bannerovej reklamy. Z dlhodobého hľadiska je ale výhodnejšie a lacnejšie využiť SEO optimalizáciu. [1]

Výhody SEO:

- zákazník hľadá inzerenta a nie naopak,
- dlhodobý prísun návštevníkov,
- po úvodnej optimalizácii je možné udržiavať stály prísun zákazníkov za podstatne nižšie prostriedky.

Nevýhody SEO:

- výsledky sa nedostavia ihneď,
- vysoko konkurenčné prostredie je veľká nevýhoda,
- nevhodné na krátkodobé kampane.

Výhody sponzorovaných odkazov:

- rýchly nárast návštevnosti,
- flexibilita,
- viacero formátov.

Nevýhody sponzorovaných odkazov:

- drahšie ako SEO z dlhodobého hľadiska,
- oslovuje aj zákazníkov nemajúcich záujem o ponúkaný produkt,
- vysoko konkurenčné prostredie môže byť veľká nevýhoda.

1.1.4 Katalóg vs. fulltextový vyhľadávač

Katalóg je web rozdelený na kategórie, ktorý obsahuje odkazy na iné web stránky. Katalógový vyhľadávač hľadá výsledky podľa slov v titulku a v obecnom popise. Výsledok zobrazí, ak sa vami zadané slovo zhoduje s názvom kategórie. Rozdiel medzi katalógom a fulltextovým vyhľadávačom je teda jasný: Do katalógu sú odkazy na stránky vkladané ručne a jeho vyhľadávací nástroj zobrazí výsledky vyhľadávania len na stránkach zaregistrovaných v katalógu. V katalógoch sa pozícia zobrazenia na *SERP* - *search engine result page* dá zakúpiť.

Fulltextový vyhľadávač je web, ktorý aktívne prehľadáva web stránky na internete (pomocou robotov) a vytvára si vlastný index - databázu nájdených stránok. Na základe podnetov od návštevníkov potom ponúka výsledky zo svojho fulltextového indexu. Najviac návštevníkov web stránok na ne prichádza práve cez fulltextové

vyhľadávače. **Hlavný dôvod využívania katalógov** je ten že, veľký počet robotov prehľadáva odkazy v katalógoch ešte pred prehľadávaním internetu ako celku. Odkazy z katalógov ponúkne ako výsledok vyhľadávania na prvých miestach množstvo vyhľadávačov.



Obr. 1.2: Příklad katalógu z www.seznam.cz

1.1.5 Princíp fungovania vyhľadávača

Fulltextové vyhľadávače fungujú nezávisle ako napr. Google alebo v rámci katalógových vyhľadávačov. Fulltexty vytvárajú svoj index (databázu stránok, v ktorej hľadajú odpoveď hľadania) automaticky. Ich programy, zvané aj **boty (bot - skratka robot)**, automaticky prehľadávajú web, kopírujú prejdene stránky do svojho indexu a na základe prehľadania tejto databázy prezentujú výsledky. Tento proces sa nazýva **”prechádzanie”**, anglicky **”crawling”**, a riadi sa algoritmom, ktorý určuje aké stránky a v akom poradí sa budú prehliadať.

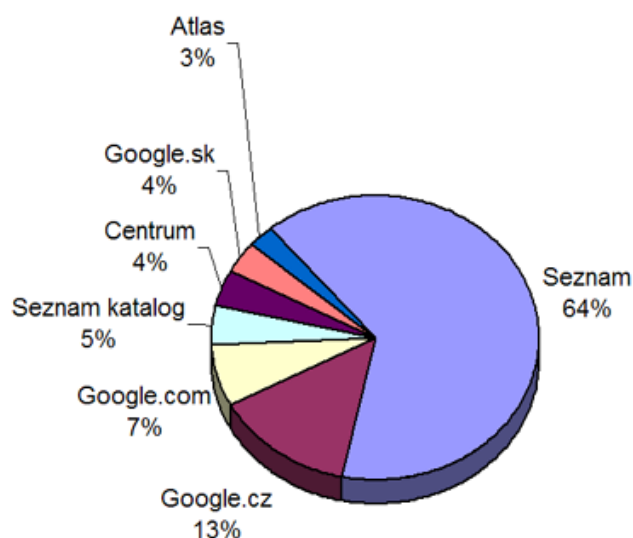
Robota zaujímajú najmä tieto údaje:

- hlavička dokumentu, metaznačky,
- presmerovanie,
- stavové kódy,
- informácie o zmenách,
- čas spracovávania stránky.

Roboty môžu mať obmedzené sťahovanie veľkých alebo dlho nedostupných súborov - dlhá odozva môže znamenať poruchu servera.

Výsledky vyhľadávania vo fulltextových vyhľadávačoch preto nikdy neodrážajú aktuálny stav stránky, ale taký, aký bol v čase posledného kopírovania webstránky do indexu.

Rôzne vyhľadávače aktualizujú svoj index v rôznych časových intervaloch. Niektoré slovenské a české fulltexty ako napr. Jyxo alebo Morfeo tvrdia, že aktualizujú svoj index každý deň príp. do 48 hodín. Google potrebuje na zaradenie nových informácií aj niekoľko týždňov. [9]



Obr. 1.3: Podiel vyhľadávačov v ČR na konci roka 2006. Zdroj www.toplist.cz

Z grafu vyplýva že pri optimalizácii webu je najdôležitejšie zamerať sa na vyhľadávače Google a Seznam.

1.2 On Page SEO faktory

Je známe, že vyhľadávače používajú On Page faktory (z anglického: on page = na stránke) k radeniu výsledkov. On Page optimalizácia teda zahŕňa všetky techniky, ktoré priamo súvisia s vylepšením týchto faktorov na stránkach. Na On Page optimalizáciu sa dá pozeráť z viacerých hľadísk: technické faktory, obsahové faktory a informačná architektúra. [10]

Technické faktory - Technické otázky riešia, ako nakódovať a naprogramovať web stránky, aby vyhľadávače mohli indexovať ich obsah a aby ho čo najlepšie pochopili (sémantika). Prakticky ide o vybrané state z pravidiel prístupnosti. Ak stránky zodpovedajú pravidlám prístupnosti, tieto faktory sú automaticky dodržané.

Obsahové faktory - V nich ide najmä o tvorbu kvalitného obsahu (copywriting) a rozmiestnenie zvolených kľúčových slov do prioritných miest na stránke. Prioritné miesta na stránke sú vymenované vyššie, v tomto prípade sa ale rieši ich element - textový obsah.

Informačná architektúra - Informačná architektúra v SEO je tá časť celkovej optimalizácie, ktorá rieši ako zahrnúť jednotlivé témy z rozboru kľúčových slov do webového projektu, čiže štruktúru a navigáciu. [10]

1.2.1 Titulok stránky

Má veľkú dôležitosť a býva umiestnený v párových značkách <title>. Mnohé vyhľadávače zobrazujú ako titulok vyhľadávaného spojenia práve slová z titulku stránky. **Značka title je z hľadiska vyhľadávania najdôležitejší on page SEO faktor.** Dobre zvolený titulok stránky korešponduje s obsahom stránky a s tým, čo ľudia hľadajú, môže stránke vo výsledkoch vyhľadávania veľmi pomôcť.

1.2.2 Hlavné nadpisy stránky

Hlavné nadpisy na stránke sa označujú ako **header tagy**. Celkovo sa môže použiť až **6 úrovní nadpisov**: H1, H2, H3, H4, H5, H6. Príklad zápisu header tagu H1: <h1>Hlavný nadpis stránky</h1>

Nadpis s najväčšou prioritou (väčšinou ten prvý) je H1, o jedno nižšou prioritou je H2, atď. Pretože pri písaní textov sa kľúčové slová prirodzene dávajú do nadpisov, aj vyhľadávače hodnotia slová v nadpisoch ako dôležitejšie než ostatný text. Dôležitosť nadpisov pre SEO je prakticky neporovnateľná s *title-ami*, čo však neznamená, že by sa nemali používať. Je tu aj iná výhoda, a to prístupnosť a použiteľnosť web stránok. [10]

Pre čo možno najlepšiu štruktúru html tagov na stránke, **je dobré, ak je na každej len jeden H1 tag**. H1 tag patrí na začiatok stránky, pokračuje sa nadpisom druhej úrovne H2 a potom H3 atď.

1.2.3 Meta tagy Keywords a Description

Meta Keywords

Meta keywords patrí do hlavičky html dokumentu a zapisuje sa takto:

```
<meta name="keywords"
  content="Sem sa píšú kľúčové slová oddelené čiarkou">
```

Tento tag je prežitok z doby, keď ešte vyhľadávače neboli fulltextové (neindexovali celý obsah stránok) a radili web stránky na základe meta dát. V dnešnej dobe je pre účely optimalizácie absolútne nepoužiteľný. Všetky moderné vyhľadávače tento tag ignorujú. Napriek tomu je dobré mať aj tento tag vyplnený. **Ideálna dĺžka meta keywords je do 10 slov.**

Meta Description

Patrí do hlavičky html dokumentu a zapisuje sa takto:

```
<meta name="description"
  content="Sem sa píše krátky text, ktorý popisuje stránku.">
```

Je to informácia o stránke, ktorú vyhľadávače niekedy používajú ako popis odkazu vo výsledkoch hľadania. **Ideálna dĺžka meta description je do 200 znakov.**

SEO experti tvrdia, že Google meta tagy už vôbec neberie do úvahy a že ich vypisovaním len strácate čas. Jedno je ale isté, pokiaľ sa v popise (Description) nachádza kľúčové slovo, ktoré pri vyhľadávaní zadáte, vo výsledkoch vyhľadávania sa zobrazí vami zadaný popis stránky, a tým pádom môžete dobrým popisom prilákať návštevníka práve na túto web stránku.

Tu takisto platí, že kľúčové slová použité v popise, by sa mali nachádzať v texte na danej stránke. Dĺžka popisu by mala byť do 200 znakov, aby sa celý zobrazil pri výsledkoch vyhľadávania vo vyhľadávačoch. Nemali by to byť len kľúčové slová oddelené čiarkou, ale súvislá veta alebo vety.

1.2.4 Valídnošť stránky

Súčasný webové prehliadače dokážu zobrazíť HTML kód aj s väčším množstvom chýb. Ak chýba ukončenie značky, snaží sa ju vhodne doplniť. Robot vyhľadávača

by to mal zvládnuť tiež, ale môže sa stať, že pri neukončenej značke môže zameniť text za HTML značku. Aby sa zabránilo takejto chybe, používa sa pre kontrolu validátor (validátor od W3C). **Takýto validátor bude implementovaný aj v navrhovanom portáli.** Ak sa aj zdá že web stránky sú napísané správne a bez chýb nie je to pravda. Nájsť takúto web stránku je v dnešnej dobe pomerne ťažké.

1.2.5 Prístupnosť stránky

Prístupná webová stránka je použiteľná pre každého užívateľa internetu bez ohľadu na jeho postihnutie, schopnosti, znalosti, skúsenosti alebo zobrazovacie možnosti. Veľmi veľká chyba ktorú by sme mohli urobiť je podcenenie počtu takýchto užívateľov. **Hendikepovaných užívateľov**, teda všetkých užívateľov, pri ktorých existuje rozumný dôvod, že s niektorými bariérami webu budú mať problémy, **môže byť až tretina z celkového počtu užívateľov**, ktorí by chceli webovú stránku používať.

Je veľmi podstatné uvedomiť si, že **robot vyhľadávača je v podstate takýto hendikovaný užívateľ**. Pretože nevidí grafiku, farby a iné veci, ktoré bežný užívateľ bežne vidí a využíva. Jediné čo si robot všima je text na stránke. Ak teda tento text skryjeme za obrázok, robot ho nevidí.

Najdôležitejšie pravidlá, na ktoré si dávať pozor pri tvorbe prístupných stránok:

- Všetky grafické prvky, ktoré slúžia na ovládanie stránky musia mať textovú alternatívu - teda správne vyplnený tzv. ALT text obrázka:
``
- Je lepšie nepoužívať tabuľky ak môže byť informácia podaná inou formou - nepoužívať vnorené tabuľky.
- Prvky tvoriace nadpisy by mali byť korektne vyznačené v zdrojovom kóde - na stránke by sa mal vyskytovať len jeden hlavný nadpis t.j. nadpis v párových značkách `<h1></h1>`.
- Ak sa dá nepoužívať na stránkach rámce. Ak to nie je možné mal by byť v značke `<frameset>` správne zvolený názov - parameter `name`
- Stránka by mala byť validna a nemala by obsahovať syntaktické chyby.

Existuje ešte veľké množstvo pravidiel pre tvorbu prístupného webu (WCAG 1.0 a WCAG 2.0). Tieto základné sú postačujúce.

1.2.6 Veľkosť stránky a doba načítavania

Častým dôvodom kvôli ktorému môžu stránky klesnúť na nižšie pozície vo výsledkoch vyhľadávania môže byť aj ich nedostupnosť alebo dlhšie načítavanie v čase ich prehliadania vyhľadávacím robotom. Roboty navštevujú internetové stránky v určitých intervaloch. Ak je teda v okamihu návštevy robota server nedostupný, stránka sa diskvalifikuje. Robot stránku síce znova navštíví ale to môže byť až za dlhšiu dobu. Dlhé načítanie stránok znepríjemňuje život aj návštevníkom, ktorý nie sú spokojní ak nedostanú odpoveď do 2 sekúnd. Ak nedostanú odpoveď do 8 sekúnd odchádzajú. [1]

Ako zabrániť pomalému načítaniu vašich stránok: [1]

- **Znížte dátový objem stránok** - optimalizujte grafiku, komprimujte stránky, dlhé texty rozdeľte na viacero stránok. Pokúste sa neprekročiť **veľkosť html kódu stránky 100kB**. Vyhľadávače preferujú menšie súbory - menej ako 40kB.
- Oddel'te od seba kaskádové štýly a vlastný html kód stránky. Nepoužívajte tzv. inline kaskádové štýly.
- Ak chcete vložiť do vašich stránok cudzí obsah, je potrebné presvedčiť sa, že odozva servera s týmto obsahom je dostatočne rýchla.
- Preferujte kvalitný a rýchly webhosting aj v prípade vyššej nákupnej ceny.

1.3 Off Page SEO faktory

Off Page optimalizácia teda optimalizácia faktorov mimo stránky spočíva zjednodušene **v budovaní spätných odkazov na vaše stránky**. Spätné odkazy sú všetky linky, ktoré na váš web odkazujú z iných stránok. Existuje viacero techník ako takéto odkazy získavať napríklad: výmena odkazov(link exchange), e-mailing, virálny marketing a iné.

Zistenie off page faktorov je veľmi dôležité v prípade ak je zisťované, či on page optimalizácia prebehla v poriadku a ako sa to prejavilo vo výsledkoch vyhľadávania. Taktiež je veľmi dôležité vedieť koľko spätných odkazov smeruje na naše stránky.

1.3.1 Rank stránok

Vyhľadávače určujú pozície stránok na *SERP* - *search engine result page* podľa vlastných hodnotiacich systémov tzv. **odkazových rankov**. Obsahujú rôzne hodnotenia

napr. podľa odkazovej popularity, podľa významnosti a iných. Výsledné hodnotenie stránky je teda súbor mnohých faktorov. Existuje množstvo rankov stránok, sú spomenuté len tie najpodstatnejšie pre český internet.

1.3.2 PageRank

PageRank (PR) je spôsob, akým Google meria dôležitosť stránky. Google predpokladá, že ak jedna stránka odkazuje na inú, hlasuje tým za jej dôležitosť. Takže, čím viac hlasov jedna stránka má, tým je dôležitejšia, má vyšší PageRank. Nejde však len o počet odkazov. Záleží aj na tom, aký PageRank a koľko odkazov má stránka, ktorá odkazuje.

Výpočet PageRank

Algoritmus PageRanku bol popísaný zakladateľmi Googlu – **Larry Page** (odtiaľ ten názov PageRank = Pageovo hodnotenie) a Sergey Brin, v ich práci: *The Anatomy of a Large-Scale Hypertextual Web Search Engine*. Pôvodný vzorec vyzerá takto:

$$PR(P_i) = (1 - d) + d \cdot \sum (PR(P_j) / C(P_j)) [14]$$

Novšia verzia je mierne upravená, ale v zásade sa na princípe PageRanku nič nemení:

$$PR(P_i) = (1 - d) / N + d \cdot \sum (PR(P_j) / C(P_j)) [14]$$

Ako je vidieť:

1. PageRank sa neurčuje pre celý web, ale pre každú stránku zvlášť.
2. PageRank stránky j sa neprenáša na stránku i celý, ale je rozdelený počtom odkazov na stránke j a ešte znížený o damping faktor d .
3. PageRank stránky je určený na základe PageRankov stránok odkazujúcich na počítanú stránku.

Pre výpočet PR jednej stránky je potrebné poznať všetky ostatné PR stránok, ktoré na ňu odkazujú. Preto sa výsledný PR počíta v niekoľkých kolách a prekvapivo, vypočítané hodnoty pomerne rýchlo po niekoľkých iteráciách konvergujú k reálnemu PR (odhaduje sa, že pre priradenie PageRanku celosvetovému webu je potrebných asi 100 iterácií).

Pokiaľ sa hovorí o PageRanku web stránky, je možné myslieť na dve rozdielne veci. Jednak môžeme hovoriť o reálnom PageRanku (real PR) alebo o Toolbar PageRanku (TBPR - ToolBar PageRank alebo GTPR - Google Toolbar PageRank).

Reálny PageRank je pravdepodobne dôležitý pre umiestnenie stránok vo výsledkoch hľadania. Lenže táto hodnota sa nedá nijak zistiť, Google túto informáciu nezverejňuje.[10]

Google však prevádza reálny PageRank na škálu od 0 do 10 = Toolbar PageRank. Reálny PageRank sa prevádza na toolbarový logaritmicky, čo teda znamená, že rozdiel medzi vyššími PR (PR8 vs. PR7) je väčší ako rozdiel medzi nižšími (PR3 vs. PR2). Hodnotu tohto tzv. toolbarového Pagerank-u je potrebné zistiť aj v navrhovanom portáli. [10]

1.3.3 S-Rank

S-rank stránky je veličina, ktorá by podobne ako PageRank mala vyjadrovať dôležitosť každej stránky na českom webe. Počíta sa najmä z odkazovej siete algoritmom, ktorý zohľadňuje odkazy, ktoré na stránku miera, ale aj to, kam odkazy zo stránky vedú.

Presný výpočet S-ranku nie je verejný. S-rank sa počíta váženou nelineárnou kombináciou rôznych veličín, v ktorých výrazne prevažujú off-page faktory. Výpočet hlavného zdroja ranku sa podobá známemu algoritmu **Hubs & Authorities**, ale je upravený tak, aby dával zmysel aj pre netematické množiny stránok.

Zistenie Sranku je veľmi podstatné a bude implementované aj v mojom portáli.

1.3.4 Alexa rank

Okrem Page Ranku a S-Ranku existuje aj iné hodnotenie úspešnosti stránky a volá sa **Alexa Rank**. Ten sa zvyšuje vždy po tom, ako vašu stránku navštívi niekto s nainštalovaným Alexa Rank toolbarom a ide o celkové poradie zo všetkých webov na internete. Ak sa uvažuje na zarábaní web stránkami, určite by sa nemalo zabúdať na tento faktor. **Čím je Alexarank menší, tým je web navštevovanejší.** [18]

Počet užívateľov, ktorí majú Alexa Toolbar a navštívili váš web sa ráta len za posledné tri mesiace a taktiež záleží aj na kvalitnom obsahu, keďže sa berie do úvahy aj počet stránok, ktoré si návštevník prezrie. Daný ToolBar používajú väčšinou iba webový vývojári, preto môže byť hodnotenie dosť skreslené.

Presná hodnotna alexa ranku sa dá zistiť na tejto stránke:
<http://www.alexa.com/siteinfo>

1.3.5 Spätné odkazy

Celá off-page optimalizácia spočíva v tom, ako získavať, čo najrelevantnejšie a najkvalitnejšie odkazy pre vaše stránky. V zásade sa dajú všetky techniky rozdeliť do

dvoch veľkých skupín: **platené a neplatené**, pričom platené neznamená iba peniazmi, ale napríklad vloženým časom alebo platbou odkazom (pri výmene odkazov platíte odkazom zo svojich stránok za iný odkaz). [10]

Zistenie počtu spätných odkazov bude po Page Ranku a S-ranku dôležitou funkciou portálu.

1.4 Kľúčové slová na stránke - keywords

Kľúčové slovo (z anglického **keyword**) je slovo alebo fráza, ktoré sa zadáva do vyhľadávača ako popis záujmu hľadania. Podľa kľúčového slova vie vyhľadávač nájsť informáciu, o ktorú máte záujem. Správne zvolené kľúčové slová sú základným predpokladom k dobrej SEO optimalizácii. Vybrať správne slovo, ktoré najlepšie vystihuje vašu stránku môže byť dosť problematické.

Je nevyhnutné **vybrať tie správne kľúčové slová, ktoré najlepšie popisujú vašu stránku a súčasne majú minimálnu konkurenciu**. Pre začínajúcu web stránku je priam sebevražda použiť slovo s vysokou konkurenciou, ktoré spôsobí, že na *SERP* - *search engine result page* bude táto stránka medzi poslednými nájdenými výsledkami. Dobré vybranými spojeniami možno doceliť popredné miesta na *SERP* - *search engine result page*

Približnú konkurenčnosť kľúčového slova, môžete zistiť priamo vo vyhľadávaní. Ak si dáte v Google hľadať akúkoľvek frázu, vpravo hore dostanete text: Výsledky 1 - 10 z približne 227 000 000 pre kľúčové slovo. (0,20 sekúnd). Číslo 227 000 000 je odhad počtu súperiach stránok. Nevypovedá to úplne o konkurenčnosti kľúčového slova, pretože vo výsledku môže byť len 100 stránok tak silných, že ich predbehnete ťažšie ako tých 227 000 000. [10]

Pre výber kľúčových slov sa dajú použiť rôzne techniky a nástroje. Najlepšou voľbou môže byť nástroj pre návrh kľúčových slov od firmy Google tzv. Google AdWords.

1.4.1 Kam umiestniť kľúčové slová

1. Do adresy URL - `www.klucove-slovo.sk` alebo `www.slovo.sk/klucove-slovo/`.
2. Do title tagu stránky - `<title>klucove slovo</title>`
3. Do meta značiek `<meta description="klucove slovo">` a `<meta keywords="klucove slovo">`
4. Do alt tagov obrázkov - ``

5. Do nadpisov stránky - `<h1>klucove slovo</h1>`

6. Do zvýrazneného textu stránky - `<p>klucove slovo</p>`

1.4.2 Long tail kľúčové slovo

Kľúčové slovo na dlhý chvost (po anglicky: long tail) je fráza väčšinou pozostávajúca z viacerých slov (3, 4, 5 a viac). Práve long tail frázy sú najväčším zdrojom návštevníkov (niekedy až 80-90 percent, celosvetovo). Výnimkou podľa najnovších výskumov je aj ČR, kde najviac užívateľov používa pre vyhľadávanie 2 slová.

Frekvencia kľúčových slov na stránke by mala byť 5 až 20 percent (všetky kľúčové slová/všetky slová na stránke). Portál bude obsahovať nástroj na testovanie frekvencie kľúčových slov na užívateľom zadanej web stránke.

1.5 Ostatné SEO faktory

Pre lepšiu prehľadnosť budú vo výslednom portále zaradené do sekcie **On page SEO faktory**.

1.5.1 Roboti a súbor robots.txt

Robot je program, ktorý dokáže prehľadávať naše stránky a indexovať ich obsah do vnútornej databázy vyhľadávača. Väčšina vyhľadávačov má svojich vlastných robotov. Najznámejším je asi Google bot, ktorý pripravuje dáta pre vyhľadávač Google. Dôležité je, aby stránka bola pre robot ľahko čitateľná. Základné údaje o stránke získava z tzv. meta tagov, ktoré obsahujú základné údaje ako názov stránky, krátky popis, jazykové verzie a pod.

Povoliť robotovi indexovanie vašej stránky je možné pomocou meta značky:

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

Súbor **robots.txt** sa umiestňuje zvyčajne do koreňovej zložky stránky. Pomocou neho dokážete zakázať robotom prístup k určitým súborom alebo adresárom, ktoré nemajú byť indexované vyhľadávačom. Niekedy napríklad nejde ani o robotov, ale skôr o ušetrenie prenesených dát hostingu. Roboti prenesú veľké množstvo dát. je potrebné uviesť si, koľko to musí byť, pokiaľ navštívi každú URL na stránke.

Robots.txt môže pomôcť zlepšiť pozíciu web stránok v internetových vyhľadávačoch. Je jedným z nástrojov SEO. Jeho nesprávne použitie však môže mať za následok stratu pozície vo výsledkoch vyhľadávania. **Robots.txt** musí mať ukončovanie riadkov typu UNIX (LF). Ako prvé sa do robots.txt píše text:

User-agent: .

User-agent: googlebot - definícia typu robota

Samozrejme robotov máme na svete obrovské množstvo, ale tých dôležitých je len niekoľko z nich. Ak chcete nastaviť robots.txt pre všetkých robotov, stačí ak zapíšete do **User-agent** nasledujúci parameter:

User-agent: *

Pod názov robota napíšeme na nový riadok **Disallow**. Tento zápis slúži na zákazanie/povolenie prístupu robotov na celé stránky, či len určité zložky a súbory.

User-agent: *

Disallow:

Overenie správne definovaného súboru **robots.txt** alebo metaznačky **robots** je veľmi dôležité a portál bude musieť tieto parametre testovať.

1.5.2 Univerzálna mapa stránok - sitemap.xml

Je to v podstate mapa web stránok vo formáte XML. Mapy XML sú pomerne ľahkým spôsobom, ako upozorniť vyhľadávacie roboty na web stránky. V podstate ide o súbor, v ktorom budú odkazy na vybrané stránky webu. Tento súbor by mal byť priebežne aktualizovaný. Taktiež je známe že súbor sitemap.xml má pevný formát a pevne definované tagy XML. Ukážka správne vytvoreného súboru **sitemap.xml**:

```
<?xml version="1.0" encoding="UTF-8"?>
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">
  <url>
    <loc>http://www.slovy.sk/</loc> ...
  </url> ...
</urlset>
```

Ako sa vyjadril Google sitemapa ako taká neovplyvňuje Page rank priamo:

"Sitemapa neovplyvňuje hodnotenie vašich stránok. Ale napomáha tomu, aby sme indexovali väčšiu časť vášho webu (tým, že nás informujete o URL adresách, o ktorých sme pred tým nevedeli, a/alebo tým, že nám pomôžete priradiť prioritu URL adresám v rámci vášho webu), môže spôsobiť zvýšenie viditeľnosti vašich stránok v našom indexe."

1.5.3 Hlavné kritériá ovplyvňujúce umiestnenie stránky na SERP

Kritérií, ktoré ovplyvňujú zobrazenie stránok vo výsledkoch vyhľadania je obrovské množstvo. Sú tu vypísané len tie najdôležitejšie. Z týchto kritérií sa bude vychádzať aj pri tvorbe portálu.

Faktor	Dôležitosť [%]
Title tag	20
Spätné odkazy	20
Text v body	15
Meno domény	10
Kľúčové slová	10
Tagy v hlavičke	5
Hustota kľúčových slov	5
Slová tučné alebo kurzívou	4
Názov súboru	4
Meta Description	3
Atribúty ALT	2
Meta keywords	2

Tab. 1.1: Približný percentuálny odhad dôležitosti faktorov ovplyvňujúcich pozície vo vyhľadávačoch

Faktory zlepšujúce umiestnenie na SERP - search engine result page:
Vzájomná poloha nájdených slov, umiestnenie nájdených slov, umiestnenie hľadaneého slova v title stránky, v metaznačke description a nadpisoch, váha stránky podľa vyhľadávača, hodnota odkazujúcich stránok, kľúčové slová na stránke a iné [1]

Faktory zhoršujúce umiestnenie na SERP - search engine result page:
Stránka bez title, málo textu na stránke, žiadne odchádzajúce linky, priveľa tučného textu, priveľa nadpisov, duplicitné stránky, stránky často neprístupné pre roboty, pomalé načítanie stránok, obsah veľmi podobný obsahu na iných stránkach, duplicitné značky Title a Description na stránkach v rámci web stránok a iné [1]

2 NÁVRH PORTÁLU NA TESTOVANIE A ANALÝZU SEO

Základné vlastnosti, ktoré musí portál obsahovať sú jasné zo zadania diplomovej práce. Portál musí byť schopný testovať základné faktory SEO. Užívateľ bude mať 2 možnosti, buď si otestuje On, Off page a ostatné SEO faktory samostatne, alebo sa bude môcť registrovať a následne otestovať tieto faktory. Výsledok testu bude:

1. Zobrazenie daného parametra SEO - napríklad atribút <title>
2. Zobrazenie návrhu opravy tohto parametra ak je oprava možná - napríklad skrátiť atribút <title> v prípade, že bude mať viac ako 70 znakov

V prípade, že užívateľ **bude na stránke registrovaný a prihlásený** jeho výsledky aj s návrhom riešenia sa uložia do databázy a užívateľ bude vyzvaný aby prípadné chyby opravil. Po opravení chýb bude možné testovať stránku znova a **porovnať si výsledky pred a po uskutočnení doporučených opráv.**

2.1 Testované parametre stránok

Ako som spomínal vyššie metodika SEO je rozdelená na tri základné časti a to:

- **On-page SEO** - optimalizácia stránok ako takých, testovanie existencie sitemap.xml a robots.txt
- **Off-page SEO** - optimalizácia faktorov mimo stránky
- **Testy kľúčových slov web stránky**

Z tohto rozdelenia sa bude vychádzať aj pri návrhu portálu a budú sa testovať zvlášť On page SEO faktory, Off page SEO faktory a kľúčové slová zadanej web stránky.

2.1.1 Testované On Page SEO faktory

Z vyššie uvedených teoretických podkladov boli vybrané najdôležitejšie On Page SEO faktory, ktoré by portál mal testovať. V prípade nepriaznivých výsledkov pre niektorý faktor by mal sám navrhnúť možnú opravu. **Zoznam testovaných On page SEO faktorov s informáciou či je možné navrhnúť jeho opravu:**

1. **Title tag** - Tag <title> je veľmi mocným nástrojom nielen pre SEO, môžete ním priamo ovplyvňovať pozíciu vašich stránok. - **ideálna dĺžka title tagu je 10 až 70 znakov.** - **Návrh možnosti opravy**

2. **Meta tagy** - Správne vyplnené meta tagy Keywords (do 10 slov) a Description (do 200 znakov). - **Návrh možnosti opravy**
3. **Header tagy (nadpisy)** - Kľúčové slová v nadpisoch majú väčšiu váhu ako obyčajný text. Bude dôležité zistiť koľko nadpisov a akého druhu je použitých na konkrétnej stránke. Doporučný počet hlavných nadpisov tj. nadpisov v tagu `<h1></h1>` je 1. - **Návrh možnosti opravy**
4. **Veľkosť zdrojového kódu** - veľkosť zdrojového kódu by nemala prekročiť 40 kB. - **Návrh možnosti opravy**
5. **Kaskádové štýly** - je vhodné použiť externý súbor s kaskádovými štýlmi a nepoužívať tzv. inline kaskádové štýly. - **Návrh možnosti opravy**
6. **Valídnosť stránky** - pri neukončenej značke môže robot zameniť text za HTML značku. - **Návrh možnosti opravy**
7. **Prístupnosť stránky** - test správne vyplnených ALT textov obázkov, nepožívanie vnorených tabuliek a ďalších pravidiel prístupnosti. - **Návrh možnosti opravy**
8. **Štruktúra webu** - Obsah je základom webových stránok. Text web stránky musí byť dobre štruktúrovaný do odstavcov (tagy `<p></p>`). - **Návrh možnosti opravy**

2.1.2 Testované Off Page SEO faktory

1. **Google Page Rank** - portál zistí tzv. toolbarový Page Rank.
2. **S-rank** - testovanie tzv. Seznam ranku.
3. **Alexa Rank** - hodnotenie nástroja Alexa.
4. **Spätné odkazy** - počet spätných odkazov na zadanú web stránku vo vyhľadávачi google.
5. **Alexa spätné odkazy** - počet spätných odkazov na zadanú web stránku vo vyhľadávачi google.
6. **Zaindexované stránky odkazy** - počet zaindexovaných stránok vo vyhľadávачi google.
7. **Pozícia vo vyhľadávačoch** - portál bude testovať pozíciu web stránok na *SERP* - *search engine result page* troch najpoužívanějších fulltextových vyhľadávачov `google.cz`, `seznam.cz` a `bing.com`.

2.1.3 Ostatné testované SEO faktory

Tieto faktory nie sú priamo On page, pretože sa nevyskytujú priamo v zdrojovom kóde stránky. Napriek tomu ich zaradím do tejto skupiny testov najmä kvôli prehľadnosti a ich veľkej súvislosti s týmito faktormi.

1. **Súbor robots.txt** - kontrola či sa na vašej stránke vyskytuje súbor **robots.txt** a správne vyplnená meta značka **robots**. - **Návrh možnosti opravy**
2. **Súbor sitemap.xml** - správne vyplnený podporuje indexáciu vašich stránok vo vyhľadávačoch. - **Návrh možnosti opravy**

2.1.4 Testovanie kľúčových slov - keywords

Užívateľ zadá svoje kľúčové slovo (kľúčovú frázu). Portál zistí celkový počet slov na stránke a počet výskytov zadanej kľúčovej frázy. Z týchto dvoch hodnôt vypočíta **frekvenciu výskytu kľúčovej frázy** na zadanej stránke. **Ako vyplýva z teoretickej časti ideálna frekvencia kľúčových slov na stránke by mala byť 5 až 20 percent.** Výhodou bude, že portál odstráni z výsledkov hľadania tzv. **Stop slová**, sú to slová, ktoré sú veľmi krátke - **1 až 3 písmená**. Väčšinou ide o spojky, častice a predložky (a, aj, na, pod atď.) Portál bude ďalej testovať výskyt kľúčového slova v rôznych dôležitých častiach stránky ako sú:

- Adresa URL
- Title tag stránky
- Meta značky keywords a description
- Alt texty obrázkov
- Nadpisy stránky - úrovně 1. až 3.
- Zvýraznený text stránky

3 REALIZÁCIA TESTOVACÍCH SKRIPTOV


V nasledujúcom texte sa budem snažiť jednoducho a prehľadne vysvetliť funkciu jednotlivých testovacích skriptov. Vytvoril som 3 testovacie skripty a to:

1. On page SEO testovací skript
2. Off page SEO testovací skript
3. Skript testující klíčové slova

3.1 On page SEO testovací skript

Tento skript sa nachádza v jedinom súbore nazvanom **seo-on.php** v adresári **scripts**. Stará sa o správne fungovanie všetkých on page SEO testov, ktoré boli navrhnuté v kapitole 2.1.1. Každý zistený výsledok je uložený do príslušnej premennej aby mohol byť následne zapísaný do databázy a zobrazený. Zahrnutie skriptu do celkovej koncepcie portálu je vysvetlené v kapitole 4.

3.1.1 Vstupný formulár



URL adresa stránky v tvare `www.google.com`

Vaša URL:

Zisti ON

Obr. 3.1: Vzhľad vstupného formulára na testovanie on page SEO faktorov

3.1.2 Regulárne výrazy

Je potrebné vysvetliť jeden pojem a to regulárne výrazy. V testoch sú využívané veľmi často. Regulárny výraz (skrátene aj regexp alebo regex) je textový reťazec, ktorý je maska alebo vzor na množinu reťazcov, na základe určitých syntaktických pravidiel. Regulárne výrazy sú používané v mnohých textových editoroch a utilitách na hľadanie a manipuláciu s textom podľa určitých vzorov. Mnoho programovacích jazykov podporuje regulárne výrazy na prácu s reťazcami. [15]

3.1.3 Test vstupných parametrov

Prvá a základná vec, ktorú musí skript obsahovať je **testovanie správnosti URL**. Každá URL musí byť zapísaná v tvare `http://www.priklad.com` pretože táto URL musí byť následne spracovaná. Pre užívateľov je ale výhodnejšie zadávať URL v tvare `www.priklad.com` teda bez definície protokola. Táto skutočnosť je ošetrená pomocou funkcie `preg_match()` a regulárneho výrazu:

```
$urltest=$_REQUEST['url'];
if (!preg_match("/^(https?:\\/+([\\w\\-]+\\.([\\w\\-]+)))/i", $urltest))
{ $url = "http://" . $urltest; }
else
{ $url = $urltest; }
```

Test existencie vstupnej URL

Tento test je uskutočnený pomocou php **knižnice CURL**. Táto knižnica bola vytvorená Danielom Stenbergom. Umožňuje spojenie a komunikáciu s mnohými rôznymi typmi serverov s rozličnými typmi protokolov. Aktuálne libcurl podporuje protokoly http, https, ftp, gopher, telnet, dict, file a ldap. Libcurl tiež podporuje HTTPS certifikáty, HTTP POST, HTTP PUT, FTP uploadovanie a mnoho iných. Základ CURL funkcií je, že sa inicializujete CURL sedenie pomocou `curl_init()`, môžu sa nastaviť všetky voľby pre prenos cez `curl_setopt()`, sedenie sa môže uskutočniť s `curl_exec()` a ukončiť pomocou `curl_close()`. [16]

```
$handle = curl_init($url);
curl_setopt($handle, CURLOPT_RETURNTRANSFER, TRUE);
$response = curl_exec($handle);
$httpCode = curl_getinfo($handle, CURLINFO_HTTP_CODE);
if($httpCode == 404 || $httpCode == 0) {
exit("<p>Stránka <strong><a href=\""$url\""$>$url</a></strong>
neexistuje. Vložte správnu URL.</p>"); }
curl_close($handle);
```

Do premennej `$httpCode` sa uloží stavový kód. V prípade, že stavový kód je rovný 0 t.j. stránka neexistuje resp. zadaný reťazec nie je adresa stránky alebo je rovný 404 t.j. stránka nebola nájdená uskutoční sa ukončenie programu pomocou funkcie `exit()` a vypíše sa text informujúci, že zadaná adresa stránky neexistuje.

3.1.4 Test title tagu stránky

Ako vieme z teórie SEO title tag (titulok stránky) je veľmi mocným nástrojom nielen pre SEO, môžete ním priamo ovplyvňovať pozíciu vašich stránok. Jeho ideálna dĺžka je 10 až 70 znakov. Je dôležité zistiť:

1. či sa na stránke nachádza `<title>` tag.
2. či má **správnú dĺžku t.j. 10 až 70 znakov**.

Zdrojový kód zadanej webovej stránky musí byť zapísaný ako reťazec do príslušnej premennej. Tento zápis sa dá uskutočniť viacerými spôsobmi. Sú využívané php funkcie `file()` a `implode()`. Prvá načíta celý obsah súboru (v našom prípade web stránky) do poľa, druhá združí elementy poľa do jedného reťazca nazvaného `$file`. Tento reťazec môže byť následne testovaný.

```
$file = file($url);  
$file = implode("", $file);
```

Na nájdenie tagu `<title>` je využitá php funkciu `preg_match()` s príslušným regulárnym výrazom.

```
preg_match("/<title>(.)</title>/i", $file, $m)
```

Regulárny výraz `/<title>(.)</title>/i` hovorí funkcii `preg_match()`, že má v reťazci `$file` hľadať výskyt html tagov `<title>` a `</title>` a text medzi nimi uložiť do poľa `$m`. V prípade, že titulok stránky bol nájdený zistí sa jeho dĺžka pomocou php funkcie `strlen()` a uloží sa do premennej `$titlesize`. Časť regulárneho výrazu `/i` hovorí, že regulárny výraz ignoruje veľkosť písma.

```
$titlesize = strlen($m[1]);
```

Dĺžka titulku je následne testovaná. V prípade, že je v poriadku t.j. jeho dĺžka je 10 až 70 znakov, do premennej `$titulok` je vložená správa o jeho správnosti. V prípade nesprávnej dĺžky je do tejto premennej vložená správa o chybe a taktiež je počítadlo chýb zväčšené o 1. **Počítadlo chýb je rovnakým spôsobom použité aj v ostatných testoch. V prípade zistenej chyby je hodnota premennej `$chyby` zvýšená o 1.** Ak neboli html tagy určujúce titulok stránky nájdené do premennej `$titulok` je vložená správa o neexistencii titulku stránky.

```
if ($titlesize > 70) {$titulok = '<span class=\"red\">  
Titulok stránky je vyplnený ale je dlhý.  
('. $titlesize. ' znakov).</span>'; $chyby = $chyby + 1; }
```

3.1.5 Test meta tagov stránky

Php skript, ktorý vypisuje a testuje meta tagy stránky je realizovaný ako funkcia. Má dva vstupné parametre a to \$url (adresa testovanej webstránky) a \$searchkey (kľúčové slovo resp. konkrétny meta tag).

```
function get_meta_data($url, $searchkey=' ')
```

Na začiatku sa do premennej \$data uložia všetky meta tagy danej stránky pomocou php funkcie get_meta_tags().

```
$data = get_meta_tags($url);
```

Ďalej sú len jednoducho vypisované pomocou php funkcie foreach(), v ktorej sa testuje či je použitá vstupná premenná, a ak je použitá vypisuje len obsah konkrétneho meta tagu definovaného podľa tejto premennej.

```
foreach($data as $key => $value) {  
    if($searchkey != ' ') {  
        if($key == $searchkey) {  
            $str = $value; }  
        }
```

Ak nie je použitá vstupná premenná v každom cykle sa vypíše druh meta tagu, počet slov meta tagu, počet znakov meta tagu a obsah samotného meta tagu.

```
    else { $pattern = '/ |,/i';  
    $array = preg_split($pattern, $value, -1, PREG_SPLIT_NO_EMPTY);  
    $str .= $key . ' (' . count($array) . ' slov | ' .  
    strlen($value) . ' znakov) ' . $value; }
```

V oboch prípadoch funkcia vracia premennú \$str.

```
    return $str;
```

Test meta keywords

Využíva sa vyššie definovanú funkciu get_meta_data(). S tým že ako vstupný parameter zadávam url adresu stránky a \$searchkey je rovný slovu **keywords** takže funkcia vráti len obsah tagu **meta keywords**.

```
$keywordy .= get_meta_data($url, "keywords");
```

V prípade zisťovania počtu meta keywords sa museli počítať len slová alebo slovné spojenia oddelené čiarkou, a preto bol použitý regulárny výraz, ktorý vloží do premennej \$array2 všetky slovné spojenia oddelené čiarkou a následne je zistený ich počet pomocou funkcie count().

```
$array2 = preg_split("/[,]/", $keywordy, -1, PREG_SPLIT_NO_EMPTY);  
$pocet = count($array2);
```

Z teoretickej časti vyplýva, že ideálny počet meta keywords je 10 (definované v premennej `$max`). Preto bol do skriptu zahrnutý testovací cyklus `if`, ktorý v prípade, že **počet meta keywords je väčší ako 10** vloží do premennej `$keywordy` chybovú hlášku.

```
if ($pocet > $max) {  
    $odst = $pocet - $max;  
    $keywordy = "<span class=\"red\">Príliš veľa keywordov.  
    ($pocet)</span>"; $chyby = $chyby + 1; }
```

Test meta description

Funguje podobne ako test meta keywords využíva vyššie definovanú funkciu `get_meta_data()` s tým, že vstupný parameter tejto funkcie bude **description**. Dĺžku tagu **meta description** zistím pomocou funkcie `strlen()`.

```
$desc .= get_meta_data($url, "description");  
$pocet2 = strlen($desc);
```

Doporučená **maximálna dĺžka meta description je 200 znakov** z toho vyplýva, že aj cyklus `if` testuje premennú `$pocet2` na túto dĺžku.

```
$max2 = 200;  
if ($pocet2 > $max2)  
{ $description = "<span class=\"red\">  
Príliš dlhý meta description. ($pocet2)</span>";  
$chyby = $chyby + 1; }
```

Test meta robots

Využíva vyššie definovanú funkciu `get_meta_data()` s tým, že vstupný parameter tejto funkcie bude **robots**.

```
$roboti = get_meta_data($url, "robots");
```

Správne vyplnený meta tag robots by mal vyzeráť takto:

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

Z tohto jasne vyplýva, že obsah premennej `$roboti` by mal byť **index,follow** poprípadne **index, follow**. V prípade, že platí táto podmienka tag meta robots je nastavený správne v opačnom prípade je do premennej `$robots` uložená správa o chybe.

```
if (($roboti == "index,follow") || ($roboti == "index, follow"))
    { $robots = "Správne nastavený tag meta robots."; }
```

3.1.6 Test hlavného nadpisu stránky

Z teórie SEO vyplýva, že je dôležité zistiť koľko nadpisov a akého druhu je použitých na konkrétnej stránke. **Doporučený počet hlavných nadpisov tj. nadpisov v tagu <h1></h1> je 1.** V premennej `$file` sa nachádza reťazec so zdrojovým kódom testovanej stránky. Pre nájdenie všetkých hlavných nadpisov je využitá php funkcia `preg_match_all()` s príslušným regulárnym výrazom, ktorá hľadá všetky výskyty tagov `<h1>` a ukladá ich postupne do poľa `$matches`.

```
preg_match_all('#<h([1])>(.*?)</h\1>#is', $html, $matches,
PREG_SET_ORDER);
```

Ukazateľ `PREG_SET_ORDER` zabezpečuje postupné ukladanie prvého nájdeného výsledku do `$matches[0]`, ďalšieho výsledku do `$matches[1]` atď. Týmto je zabezpečené, že do premennej `$matches[0]` je uložený prvý výskyt tagu `<h1>`. Je jasné, že ak nie je premenná `$matches[1]` prázdna na zadanej stránke sa hlavný nadpis vyskytuje viac ako jedenkrát. Obsah premennej testujem pomocou php funkcie `isset()`.

```
if(isset($matches[1]))
{ $nadpis = "<span class=\"red\">Hlavný nadpis stránky máte
definovaný viac krát.</span>"; $chyby = $chyby + 1; }
```

Ak je prázdna premenná `$matches[0]` znamená to, že **hlavný nadpis nie je definovaný vôbec**.

3.1.7 Test veľkosti zdrojového kódu stránky

Na zistenie veľkosti zdrojového kódu je využitých viacero funkcií jazyka PHP. V prvom rade funkciu `file_get_contents()`, ktorá do premennej `$content` vloží celý zdrojový kód zadanej stránky. Následne bude na serveri vytvorený súbor **test.html** pomocou funkcie `fopen()`. Do tohto súboru bude uložený zdrojový kód stránky pomocou funkcie `fwrite()` a súbor sa uzatvorí pomocou funkcie `fclose()`.

```
$content = file_get_contents($url);
$handle = fopen("test.html", "w+");
fwrite($handle, $content);
fclose($handle);
```

Teraz je možné zisťovať veľkosť súboru pomocou php funkcie `filesize()`. Veľkosť súboru v bytoch je vložená do premennej `$velkost`, ktorá môže byť testovaná. Z teórie SEO vyplýva, že **doporučená veľkosť zdrojového kódu stránky je 40 kB**.

```
$velkost = filesize('test.html');
$maxvelkost = 40960; //rovná sa 40kB v bytoch
if ($velkost > $maxvelkost)
{ $size = "<span class=\"red\">Znížte veľkosť stránky.
($velkost bytov)</span>"; }
```

3.1.8 Test štruktúrovania textu stránky

Obsah je základom webových stránok. Text web stránky musí byť preto dobre štruktúrovaný do odstavcov. Test štruktúrovania stránky vychádza z predpokladu, že **na stránke by sa mali vyskytovať najmenej 4 odstavce**. Vyhľadávanie odstavcov na stránke teda textu v html tagoch `<p></p>` funguje podobne ako v prípade vyhľadávania hlavných nadpisov stránky. Pre nájdenie všetkých odstavcov je využitá php funkcia `preg_match_all()` s príslušným regulárnym výrazom, ktorá hľadá všetky výskyty tagov `<p>` a postupne ich ukladá do poľa `$p`.

```
preg_match_all("/<p/i", $html, $p, PREG_SET_ORDER);
```

V prípade, že je nastavený aj štvrtý prvok poľa `$p` stránka je hodnotená ako správne štruktúrovaná v opačnom prípade je do premennej `$odstavec` vložená správa o chybe.

```
if(isset($p[4]))
{ $odstavec = "<span>Dostatočné použitie odstavcov.</span>"; }
```

3.1.9 Test validnosti stránky

Robot vyhľadávača by mal zvládnuť reprezenovať web stránku správne aj v prípade chýb v zdrojovom kóde stránky, ale môže sa stať, že pri neukončenej značke môže zameniť text za HTML značku. Aby sa zabránilo takejto chybe, používa sa pre kontrolu validátor (validátor od W3C). Do premennej `$pokus` bude uložený xml dokument obsahujúci výsledok testu zadanej adresy stránky:

```
$pokus = file_get_contents("http://validator.w3.org/check?uri="
. $url . "&output=soap12");
```

Číslo medzi `<m:errorcount>` a `</m:errorcount>` je počet chýb na stránke.


```

<m:errors>
    <m:errorcount>0</m:errorcount>
    <m:errorlist>
    </m:errorlist>
</m:errors>

```

Súbor xml je uložený pomocou **protokolu SOAP**. Je to protokol založený na pre-
dávaní xml dokumentov medzi klientom a serverom. Obsah premennej **\$pokus** je
testovaný pomocou php funkcie **preg_match()** a príslušného regulárneho výrazu.
Počet chýb je nahraný do poľa **\$valresult** na pozíciu [0].

```

$htmlerror = preg_match("/<m:errorcount>(.)<\/m:errorcount>/i",
$pokus, $valresult);

```

Jednoduchým cyklom **if** je výsledok testu vložený do premennej **\$valid**.

```

if($valresult[0] > 0) {
$valid = 'Stránka nie je XHTML valídna ('. $valresult[0]. ' chýb).
<a href="http://validator.w3.org/check?uri=' . $url . '">
Vaše výsledky</a>'; }

```

3.1.10 Test štýlovania stránky

Z dôvodu zníženia veľkosti stránky a jej rýchleho načítania je výhodné všetky html
elementy umiestniť do externého css súboru.

```

<link rel="stylesheet" href="main.css" type="text/css"
media="screen,projection" />

```

Je dôležité zistiť či sa v zdrojovom kóde stránky nenachádzajú časti kódu, ktoré by
mohli byť umiestnené v tomto externom súbore, tzv. **inline css štýly**. Každý takýto
inline štýl začína značkou **style**. Je teda jasné, že ak sa v zdrojovom kóde podobná
značka nachádza, je to považované za chybu z hľadiska SEO. Pomocou php funkcie
preg_match() a regulárneho výrazu **/<style/i** sú teda hľadané výskyty značky
style a v prípade jej existencie je do premennej **\$style** vložená správa o chybe.

```

if (preg_match("/<style|style=/i", $html, $style))
{ $style = "<span class=\"red\">Nájdené inline CSS štýly.</span>"; }

```

3.1.11 Test alternatívnych textov obrázkov na stránke

Z pravidiel prístupnosti webových stránok vieme že, všetky grafické prvky, **ktoré**
slúžia na ovládanie stránky musia mať textovú alternatívu - teda správne

vyplnený tzv. **ALT text**. Pomocou php funkcie `preg_match_all()` a regulárneho výrazu `/<img.+alt/i` sú vyhľadane všetky obrázky s vyplneným ALT textom. Pomocou rovnakej funkcie a tohto regulárneho výrazu `/<img/i` sú vyhľadane všetky obrázky na stránke.

```
preg_match_all("/<img.+alt/i", $html, $img1);  
preg_match_all("/<img/i", $html, $img2);
```

Rozdiel medzi počtom všetkých obrázkov na stránke a počtom obrázkov s vyplneným ALT textom nám dáva počet obrázkov bez alternatívneho textu.

```
$alty = count($img2) - count($img1);
```

Premenná `$alty` obsahuje počet obrázkov bez alternatívneho textu. V prípade, že je väčšia ako 0 bude do premennej `$style` vložená správa o chybe.

```
if ( $alty > 0) { $img = " Nie všetky obrázky majú správne definované  
alternatívne texty. ($alty)"; }
```

3.1.12 Test použitia rámcov na stránke

Z pravidiel prístupnosti webových stránok taktiež vyplýva, že **na modernej webovej stránke by sa nemali vyskytovať rámce**. Rámce sú definované pomocou html značiek `<iframe>` alebo `<frameset>`. Pomocou funkcie `preg_match()` a regulárneho výrazu `/iframe/i` sú hľadané výskyty značky `<iframe>` a v prípade jej existencie je do premennej `$iframe` vložená správa o chybe. Rovnaký postup platí aj pre značku `<frameset>`.

```
if (preg_match("/iframe|frameset/i", $html, $iframe))  
{ $iframe = "Nájdený rámec."; }
```

3.1.13 Test použitia tabuliek na stránke

Tabuľky by sa na modernom webe mali vyskytovať v minimálnom množstve. Preto bude upozornené na ich použitie a v návode bude ukázaný správny spôsob ich použitia. Každá tabuľka je definovaná pomocou html tagov `<table>` a `</table>`. Pomocou php funkcie `preg_match()` a regulárneho výrazu `</table/i` sú hľadané výskyty značky `<table>`. V prípade jej existencie je do premennej `$tables` vložené upozornenie o jej výskyte.

```
if (preg_match("</table/i", $html, $table))  
{ $tables = 'Nájdena tabuľka, pozor na jej správnu funkčnosť.'; }
```

3.1.14 Test súboru sitemap.xml

Mapy XML sú pomerne ľahkým spôsobom, ako upozorniť vyhľadávacie roboty na web stránky. V podstate ide o súbor, v ktorom sú odkazy na vybrané stránky vášho webu. Tento súbor by sa mal nachádzať v koreňovom adresári zadanej stránky. Hľadanie súboru sitemap.xml prebieha nasledovným spôsobom. Najprv pomocou funkcie `parse_url()` a nastaveného parametra `PHP_URL_HOST` je zistená presná url **adresa testovanej domény**. Je to dôležité najmä v prípadoch keď zadaná adresa nie je v tvare doménového mena napr. `www.kaplna.sk/kultura.php`.

```
$url2 = parse_url($url, PHP_URL_HOST);
```

Následne je vytvorená nová url v tvare: `http://www.kaplna.sk/sitemap.xml`:

```
$url4 = "http://" . $url2 . "/sitemap.xml";
```

Pomocou funkcie `file()` je vložený obsah `$url4` do premennej `$lines` a testuje sa či obsahuje dáta. V prípade, že neobsahuje dáta do premennej `$sitemap` bude vložená správa o chybe.

```
if ($lines !== false) { }  
else { $sitemap = "<span class=\"red\">Súbor sitemap.xml neexistuje.  
</span>"; }
```

3.1.15 Test súboru robots.txt

Robots.txt môže pomôcť zlepšiť pozíciu vašich stránok v internetových vyhľadávačoch. Jeho nesprávne použitie môže mať za následok stratu pozície vo výsledkoch vyhľadávania. **Umiestňuje sa do koreňového adresára stránky**. Hľadanie súboru **robots.txt** prebieha rovnakým spôsobom ako hľadanie súboru **sitemap.xml** rozdiel je len vo vytváraní url, na ktorej by sa mal nachádzať súbor **robots.txt**.

```
$url3 = "http://" . $url2 . "/robots.txt";
```

Správa o prípadnej chybe bude vložená do premennej `$robots2`.

3.1.16 Zistenie času testu

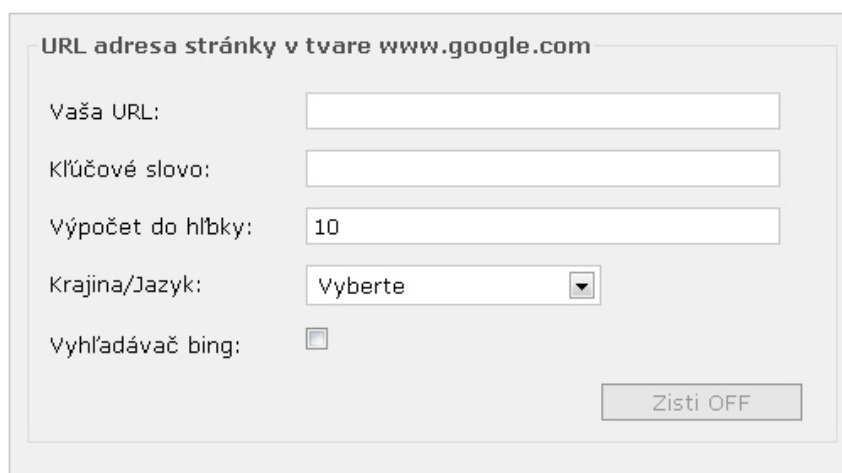
Čas testu je zisťovaný jednoducho pomocou php funkcie `getdate()`.

```
$akt_dat = getdate();
```

3.2 Off page SEO testovací skript

Tento skript sa nachádza v jedinom súbore nazvanom **seo-off.php** v adresári **scripts**. Stará sa o správne fungovanie všetkých off page SEO testov, ktoré boli navrhnuté v kapitole 2.1.2. Každý zistený výsledok je zapísaný do príslušnej premennej aby mohol byť následne zapísaný do databázy. Zahrnutie skriptu do celkovej koncepcie portálu je vysvetlené v kapitole 4. **Využíva sa rovnaký spôsob testovania vstupných parametrov a existencie url ako v prípade testovania on page SEO faktorov.**

3.2.1 Vstupný formulár



The image shows a web form titled "URL adresa stránky v tvare www.google.com". It contains five input fields: "Vaša URL:" (text input), "Kľúčové slovo:" (text input), "Výpočet do hĺbky:" (text input with "10" entered), "Krajina/Jazyk:" (dropdown menu with "Vyberte" selected), and "Vyhľadávač Bing:" (checkbox). A "Zisti OFF" button is located at the bottom right of the form.

Obr. 3.2: Vzhľad vstupného formulára na testovanie off page SEO faktorov

Funkcia CURL pre načítanie zdrojového kódu stránky

Umožňuje načítanie obsahu zadanej stránky do premennej. Navrhnutá funkcia využíva 3 funkcie CURL pre načítanie obsahu zadanej adresy:

```
curl_setopt($ch, CURLOPT_HEADER, 0);  
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);  
curl_setopt($ch, CURLOPT_URL, $url);  
$data = curl_exec($ch);
```

Funkcia vracia premennú `$data` obsahujúcu zdrojový kód danej stránky.

3.2.2 Google page rank

Na zistenie tzv. google toolbar page ranku bola navrhnutá funkcia `getpagerank()` so vstupným parametrom `$url`, teda zadanou url testovanej stránky. Najdôležitejšia časť funkcie je miesto, kde sa do tzv. **google toolbar linky** vkladá **hashová hodnota vstupnej adresy web stránky**.

```
$query="http://toolbarqueries.google.com/search?client=
navclient-auto&ch=".CheckHash(HashURL($url)).
"&features=Rank&q=info:". $url."&num=100&filter=0";
```

Nie je možné vložiť priamo hodnotu url, pretože google toolbar by túto vstupnú požiadavku vyhodnotil ako spam. Je preto potrebné vstupnú URL *hashovať* pomocou funkcií, ktoré boli navrhnuté priamo spoločnosťou google a takto vytvoriť tzv. **ochranné číslo**. Následne je pomocou funkcie `file_get_contents_curl()` uložený obsah dokumentu, ktorý vrátil google toolbar do premennej `$data`. Pomocou php funkcie `strpos()` je v tejto premennej vyhládné miesto, kde sa nachádza reťazec "Rank_". Na tejto pozícii sa nachádza aj hodnota hľadaného google toolbar page ranku.

```
$data=file_get_contents_curl($query);
$pos = strpos($data, "Rank_");
if($pos === false){} else {
$pagerank = substr($data, $pos + 9); return $pagerank; }
```

Ak sa v premennej `$data` hodnota toolbarového page ranku nenachádza funkcia vráti prázdnu premennú `$pagerank`.

3.2.3 Seznam rank

je využitý php **skript navrhnutý Romanom Matěnom**. Skript funguje na základe predávania informácií pomocou xml dokumentov medzi stránkou

`http://srank.seznam.cz` a klientom. Najdôležitejšie je pomocou php funkcie `xmlrpc_encode_request()` a `stream_context_create()` vytvoriť požiadavok **XML-RPC**, ktorý obsahuje zadanú adresu stránky a ostatné povinné údaje ako hlavička súboru xml atď. Tento súbor môže byť spracovaný stránkou `http://srank.seznam.cz`.

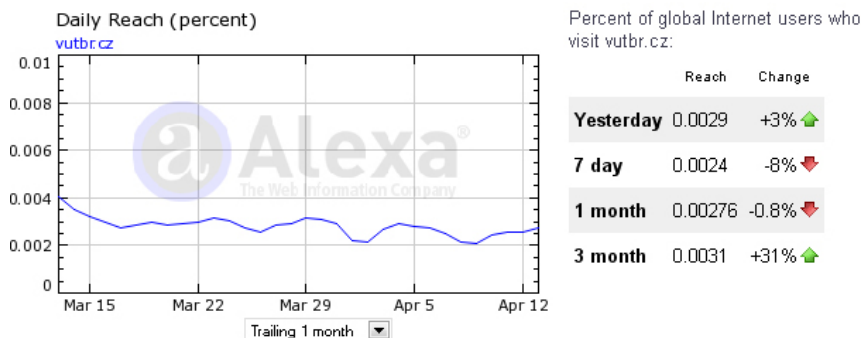
```
$params = array("0", htmlspecialchars($url), 0);
$request = xmlrpc_encode_request("getRank", $params);
```

Odpoveďou je pole obsahujúce hodnoty **status** a **rank**. Obsah vráteného poľa je pomocou php funkcie `file_get_contents()` uložený do premennej `$file` a dekodovaný pomocou funkcie `xmlrpc_decode()`. V prípade, že status sa rovná číslu 200

(to znamená OK) do premennej `$response` môže byť uložená hodnota rank delená číslom 2.55. Stáva sa, že doména `http://srank.seznam.cz` nefunguje. V tomto prípade funkcia nevráti hodnotu Sranku.[17]

3.2.4 Alexa rank

Z teórie SEO vieme, že **ALEXA rank nie je hodnoverná hodnota**. Napriek tomu sa zaoberám jeho zisťovaním najmä pre jeho **rastúcu dôležitosť**. [18]



Obr. 3.3: Popularita domény `www.vutbr.cz` zistená pomocou portálu `www.alexa.com`

Alexa portál umožňuje získať xml dokument priamo z jeho stránky a následne z neho parsovať údaje o zadanej stránke.

```
$alexaxml = "http://xml.alexa.com/data?cli=10&dat=nsa&url=".$url;
```

Stránka vráti xml dokument v tomto tvare:

```
<ALEXA VER="0.9" URL="kaplna.sk/" HOME="0" AID="">
<RLS PREFIX="http://" more="0">
</RLS><SD TITLE="A" FLAGS="" HOST="kaplna.sk">
<LINKSIN NUM="6"/>
<CHILD SRATING="0"/></SD><SD>
<POPULARITY URL="kaplna.sk/" TEXT="27773233"/>
<REACH RANK="27639051"/>
<RANK DELTA="+13780440"/></SD></ALEXA>
```

Pre spracovanie tohto xml dokumentu je veľmi výhodné použiť php funkciu `xml_parse_into_struct()`, ktorá parsuje xml do dvoch paralelných polí.

```
xml_parse_into_struct($xml_parser, $data, $vals, $index);
```

Následne sú vybrané jednotlivé hodnoty z polí a ukladané do príslušných premenných. Hodnota alexa ranku je zistená takto:

```
$alexarank = $vals[$index_popularity]['attributes']['TEXT'];
```

3.2.5 Alexa spätné odkazy

Počet spätných odkazov zistených alexou je zistený **rovnakým spôsobom ako alexa rank**. Z paralelných polí získaných z parsovania xml dokumentu zo stránky alexa je načítaná hodnota `$alexa_backlink`, ktorá predstavuje počet spätných odkazov.

```
$alexa_backlink = $vals[$index_linksin]['attributes']['NUM'];
```

3.2.6 Google spätné odkazy

Vyhľadávač google umožňuje vyhľadávať stránky pomocou rozličných parametrov. V odbornej literatúre sa môžeme stretnúť aj s pojmom **google hacking**. Základom je url adresa:

```
www.google.com/search?
```

Za otáznikom nasledujú parametre vyhľadávania:

```
www.google.com/search?premenna1=hodnota1&...
```

Pokročilý operátor(premenná) **link** slúži na hľadanie odkazov na stránku. Musí byť zadaný v tvare **operátor:hľadaný_termín**. Pri zisťovaní spätných odkazov nájdených googlom teda je zadaná ako premenná **link** a ako hodnota zadanú **adresu stránky**. Celá adresa potom vyzerá takto:

```
$url = 'http://www.google.com/search?&q=link:'. $uri. ';
```

Pomocou funkcie `file_get_contents_curl()` je do premennej `$v` uložený obsah vytvorenej adresy. Počet spätných odkazov je číslo, ktoré môžeme vidieť na vytvorenej adrese. V tomto prípade je to 2300 spätných odkazov.



Obr. 3.4: Počet spätných odkazov na stránku `www.sme.sk` vo vyhľadávači google

Počet spätných odkazov získam pomocou php funkcie `preg_match()` a príslušného regulárneho výrazu.

```
preg_match('/id=resultStats>(.) results\<nobr\>/', $v, $r);
```

3.2.7 Google zaindexované stránky

Počet zaindexovaných stránok je získaný podobne ako počet spätných odkazov. Na miesto pokročilého operátora **link** je ale využitý **operátor site**. Vytvorená adresa bude vyzeráť takto:

```
$url = 'http://www.google.com/search?hl=en  
&q=site:'. $uri. '&filter=0';
```

Počet zaindexovaných stránok je číslo na rovnakej pozícii ako v prípade počtu spätných odkazov.



Obr. 3.5: Počet zaindexovaných stránok domény www.sme.sk vo vyhľadávači google

Výsledný počet je vyhľadaný pomocou funkcie `preg_match()` a regulárneho výrazu. Funkcia vracia počet zaindexovaných stránok v premennej `$s[1]`.

3.2.8 Pozícia stránky na google

Na zistenie pozície zadanej stránky vo vyhľadávači google je použitý iný príklad tzv. google hackingu. **Google umožňuje zobrazíť ľubovoľný počet výsledkov na stránku.** Premenná **num** je nastavená na hodnotu **1**. Znamená to že google zobrazí na *SERP* - *search engine result page* 1 výsledok zo všetkých nájdených.

```
http://www.google.sk/search?q=feec&num=1
```

Je teda jasné, že postupným zvyšovaním premennej **num** o **1** je možné dosiahnuť postupné zvyšovanie počtu nájdených stránok zobrazených na 1 stranu maximálne však do hodnoty **100**. Ďalšia dôležitá premenná je **start**. Umožňuje určiť, od ktorej pozície má vyhľadávač zobrazovať výsledky vyhľadávania.

Modelový príklad:

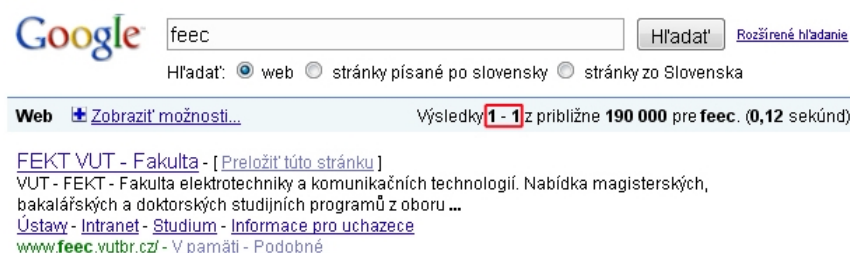
Je hľadaná pozíciu stránky **www.feec.vutbr.cz** na zadané kľúčové slovo hľadania napríklad **feec**.

Vstupné premenné zadané užívateľom vo vstupnom formulári:

- \$land - krajina hľadania, napríklad slovensko - **www.google.sk**,
- \$jazyk - jazyk hľadania, napríklad slovenčina - **&hl=sk**,
- \$keywords - kľúčové slovo hľadania - v našom prípade **feec**,
- \$url - hľadaná adresa - v našom prípade **www.feec.vutbr.cz**,
- \$do - hĺbka vyhľadávania je **maximálna pozícia, na ktorej sa môže vyskytnúť hľadaná stránka na SERP**.

Modelová linka zadaná do prehliadača bude vyzeráť takto:

`http://www.google.sk/search?q=feec&hl=sk&num=1&start=0`



Obr. 3.6: SERP vyhľadávača google po zadaní modelovej linky do internetového prehliadača

Vyhľadávač google bude hľadať na slovensku, výsledky v slovenčine, kľúčové slovo hľadania feec, začiatočná pozícia 0 a počet výsledkov na stranu 1.

Realizácia funkcie

Pomocou funkcie **while** sa bude postupne v každom jej cykle zvyšovať hodnota **start** o 1 až do hodnoty v premennej **\$do**, ktorú si môže nastaviť užívateľ vo vstupnom formulári ako tzv. **hĺbku hľadania**. Premenná **start** musí začínať na nulte pozícii. Hodnota premennej **num** bude stále 1. To znamená, že sa bude vždy zobrazovať jeden výsledok na SERP. V každom cykle je uložený obsah stránky s výsledkami vyhľadávania do premennej **\$page**. Testuje sa či sa v nej nachádza hľadaná stránka pomocou php funkcie **preg_match()** a regulárneho výrazu obsahujúceho kľúčové slovo, teda premennej **\$keywords**. V prípade, že sa hľadaná stránka na tejto pozícii nachádza funkcia vráti hodnotu pozície a skončí.

V prípade že sa hľadaná stránka nenájde, hodnota premennej **\$b** sa zväčší o jedna a začne nový cyklus v ktorom bude mať premenná **start hodnotu 1**. Znamená to, že SERP bude obsahovať jeden výsledok na prvej (google začína vždy na nulte pozícii) pozícii. Cyklus končí ak sa hľadaná stránka nájde na pozícii, ktorá je menšia alebo rovná zadanej hĺbke vyhľadávania. V každom cykle je potrebné **pozastaviť beh skriptu, aby google nevyhodnotil opakované požiadavky na svoj vyhľadávací mechanizmus ako tzv. spamové**. Je využitá funkcia `usleep()`, ktorá pozastaví beh cyklu na zadaný čas v mikrosekundách.

```
$b=0; while ($b <= $do) {  
$result = "http://www.google." . $land . "/search?q=" .  
$send . "&hl=" . $jazyk . "&num=1&start=" . $b;  
usleep(500000); $page = file_get_contents($result);  
if (preg_match("/\b$url\b/i", $page)) {  
$c = $b + 1; return $c; break; } $b++; } }
```

3.2.9 Pozícia stránky na seznam

Vyhľadávač **seznam.cz** umožňuje použitie veľmi podobných tzv. hackov. Z toho vyplýva, že zisťovanie pozície stránky vo výsledkoch vyhľadávania pre zadané kľúčové slovo bude takmer rovnaké ako v prípade vyhľadávača google.

Príklad modelovej linky zadanej do prehliadača:

```
http://search.seznam.cz/?q=feec&count=1&from=0
```

Vyhľadávač zobrazí výsledky vyhľadávania so začiatočnou pozíciou 0 a počtom výsledkov na stranu 1.

Realizácia funkcie

Samotná realizácia skriptu zisťujúceho pozíciu stránky vo výsledkoch vyhľadávania vyhľadávača **seznam.cz** je takmer totožná s funkciou, ktorá realizuje hľadanie pozície stránky na SERP vyhľadávača google. Jediný rozdiel je **nahradenie premennej start z googlu premennou from a nahradenie premennej num premennou count**. Všetky ostatné časti tejto funkcie sú rovnaké.

Vyhľadávací mechanizmus je taktiež citlivý na opakované požiadavky z jedného zdroja. Opakovaným testovaním bol zistený **minimálnemu čas medzi dvoma nasledujúcimi požiadavkami** na vyhľadávací mechanizmus rovný **0,5 sekundy**.

```
$i=0; while ($i <= $do) {  
$result2 = "http://search.seznam.cz/?q=" . $send . "&count=1&from=" . $i;
```

```

usleep(500000); $page2 = file_get_contents($result2);
if (preg_match("/\b$url\b/i", $page2))
{ $a = $i + 1; return $a; break; }

```

Funkcia vracia nájdenú pozíciu stránky na SERP vyhľadávača seznam v premennej \$a.

3.2.10 Pozícia stránky na bing

Testovanie pozície stránky na vo vyhľadávači **bing** sa mierne líši od predchádzajúcich dvoch. Hlavným dôvodom je, že **tento vyhľadávač nepodporuje niektoré tzv. hacky**. Preto využívam **Bing API na predávanie výsledkov vyhľadávania vo forme xml dokumentu**. Týmto spôsobom je možné presne nastaviť začiatočnú pozíciu vyhľadávania a aj počet výsledkov na jednu stranu. [20]

Príklad modelovej linky zadanej do prehliadača:

```

http://api.search.live.net/xml.aspx?Appid=
142B46B466DBF5F1530FE50E9CC7D20342733759&query=
feec&sources=web&web.count=1&web.offset=1

```

Popis parametrov modelovej linky:

- **Appid** - ID užívateľa používajúceho Bing API,
- **query** - kľúčové slovo alebo slová hľadania,
- **sources** - typ vyhľadávania,
- **web.count** - počet výsledkov na 1 stranu (jeden xml dokument),
- **web.offset** - začiatočná pozícia vyhľadávania.

Príklad xml dokumentu z vyhľadávača bing pre stránku www.feec.vutbr.cz:

```

<web:WebResult>
<web:Title>FEEC BUT - Fakulta</web:Title>
<web:Description>
BUT - FEEC - Faculty of Electrical Engineering and Communication ...
</web:Description>
<web:Url>
http://www.feec.vutbr.cz/fakulta/home.php.en?lang=en
</web:Url></web:WebResult>

```

Realizácia funkcie

Samotná realizácia skriptu zisťujúceho pozíciu stránky vo výsledkoch vyhľadávania vyhľadávača **bing.com** je vo svojej podstate totožná s predchádzajúcimi dvomi funkciami. Funkcia vracia výslednú pozíciu do premennej **\$f**.

3.3 Skript testujúci kľúčové slová

Tento skript sa nachádza v jedinom súbore nazvanom **seo-key.php** v adresári **scripts**. Stará sa o správne fungovanie všetkých testov kľúčových slov navrhnutých v kapitole 2.1.4. Každý zistený výsledok je zapísaný do príslušnej premennej aby mohol byť následne zapísaný do databázy. Zahrnutie skriptu do celkovej koncepcie portálu je vysvetlené v kapitole 4. **Využíva sa rovnaký spôsob testovania vstupných parametrov a existencie url ako v prípade testovania on page SEO faktorov.**

3.3.1 Vstupný formulár



The image shows a web form titled "URL adresa stránky v tvare www.google.com". It contains two input fields: "Vaša URL:" and "Kľúčové slovo(á):". Below the second field is a button labeled "Kľúčové slová".

Obr. 3.7: Vzhľad vstupného formulára na testy kľúčových slov zadanej stránky

3.3.2 Test frekvencie kľúčového slova

Zdrojový kód zadanej stránky je pomocou php funkcie `file_get_contents()` nahraný do premennej `$html`. Pred samotným zisťovaním frekvencie zadaného kľúčového slova je potrebné uskutočniť viacero operácií:

1. Získať text z meta tagov keywords a description,
2. Získať text z tela stránky a odstrániť všetky html tagy,
3. Spojiť všetok takto získaný text to jedného reťazca,
4. Z textu stránky odstrániť tzv. stop words,

5. Zaistiť správne kódovanie testovanej stránky.

Získať text z meta tagov keywords a description

je využitá php funkcia `get_meta_tags()`, ktorá obsah všetkých meta tagov zadanej stránky vráti do poľa `$meta_tags`.

```
$meta_tags = get_meta_tags($url);
```

Podstatné sú len meta keywords a description, preto sú z poľa získaných hodnôt vybrané jednoducho pomocou príkazu:

```
$tag_info = $meta_tags['description'] . " " . $meta_tags['keywords'];
```

Získať text z tela stránky a odstrániť všetky html tagy

Je jasné, že najčastejšie sa vyskytujúci text v zdrojovom kóde stránky sú html tagy. Tento text je nezaujímavý hľadiska kľúčových slov na stránke. Preto je dôležité odstrániť ho. Pre odstránenie html tagov existuje v jazyku php funkcia `strip_tags()`. Funkcia vracia zadaný reťazec bez html tagov.

```
$no_html = strip_tags($html);
```

Ďalej je podstatné odstrániť všetky prípadné **java skripty** na stránke. Bola navrhnutá funkcia `strip_tag_script()`. Funkcia hľadá prvý výskyt tagu `<script>` pomocou php funkcie `preg_match()` a regulárneho výrazu `/<script[>]*>/i`. Pomocou ďalšej php funkcie `strpos()` uloží pozíciu tohto tagu v danom reťazci do premennej `$pos1`. Rovnako je hľadaná pozícia tzv. uzatváracieho tagu `</script>` jeho pozícia je uložená do premennej `$pos2`.

```
preg_match("/<script[>]*>/i", $html, $matches);  
$str1 =& $matches[0];  
preg_match("/<\s*\/script>/i", $html, $matches);  
$str2 =& $matches[0];  
$pos1 = strpos($html, $str1);  
$pos2 = strpos($html, $str2);
```

Teraz je možné jednoducho pomocou funkcie `substr()` získať celý reťazec, ktorý začína na pozícii `$pos1` a končí na pozícii `$pos2`. Takto získaný reťazec predstavuje obsah java skriptu, ktorý sa vyskytuje na stránke a následne môže byť odstránený z textu stránky.

```
$first = substr($html, 0, $pos1);  
$html = $first . $second;
```

Kedže java skriptov sa na stránke môže nachádzať viacero je využitý cyklus `do{}` `while()`, a funkcia `if`, ktorá testuje či sa v texte vyskytuje nejaký java skript až kým nie sú všetky odstránené.

```
do { if ($pos1 != false && $pos2 != false) { ... } }  
while ($pos1 != false && $pos2 != false);
```

Spojiť všetok takto získaný text to jedného reťazca

Všetky vyššie získané reťazce sú spojené do jedného, ktorý predstavuje priamo text stránky bez všetkých tagov a skriptov.

```
$str3 = $tag_info." ".$str2;
```

Z textu stránky odstrániť tzv. stop words

Stop words (stop slová) sú slová ktoré roboty ignorujú z dôvodu nízkej relevantnosti. Patria sem napríklad predložky, častice, spojky atď. Tieto slová musia byť z testovaného textu stránky taktiež odstránené.

```
$str = clean(stopWords($str3));
```

Funkcia `clean()` odstráni z textu stránky všetky špeciálne znaky ako bodky, čiarky a podobne. Využíva na to php funkciu `str_replace()`, ktorá nahradí nájdený znak v reťazci za iný v našom prípade `' '` čiže prázdny symbol.

```
$text = str_replace('.', ' ', $text);
```

Funkcia `stopWords()` umožňuje nahradiť ľubovoľné stop slová zo súboru, porovnať ich so slovami v reťazci obsahujúcom text stránky a následne zmazať stop slová nájdené v tomto reťazci. Pomocou funkcie `file()` je uložený obsah textového súboru obsahujúceho stop slová do premennej. Je zistený ich počet a pomocou neho je nastavený cyklus `for()`. V každom cykle sú zmenené všetky písmená na malé pomocou funkcie `strtolower()`.

```
$common = file('kdawords.txt');  
$total = count($common);  
for ($x=0; $x<= $total; $x++)  
{ $common[$x] = trim(strtolower($common[$x])); }
```

Následne je reťazec obsahujúci text stránky zmenený na pole pomocou funkcie `explode()`. Všetky prvky poľa sú upravené rovnako pomocou funkcie `strtolower()`.

```

$_terms = explode(" ", $term);
foreach ($_terms as $line)
{ if (in_array(strtolower(trim($line)), $common)) { ... } }

```

Pomocou funkcie `in_array()` je prehľadávané pole s textom stránky `$line` a hľadajú sa všetky výskyty stop slov definovaných v poli `$common`. Ak bolo stop slovo nájdené je zistená jeho pozícia v poli, teda tzv. kľúč pomocou php funkcie `array_search()`. Následne je možné slovo na zistenej pozícii odstrániť pomocou php funkcie `unset()`.

```

$removeKey = array_search($line, $_terms);
unset($_terms[$removeKey]);

```

Funkcia vráti pole s textom stránky bez tzv. stop words v premennej `$clean_term`.

Zaistiť správne kódovanie testovanej stránky

Kódovanie testovanej stránky je zistené pomocou php funkcie `preg_match()` a regulárneho výrazu. Nastavenie kódovania stránky je vyhľadané v meta tagu `<meta http-equiv= ... >`.

```

preg_match( '@<meta\s+http-equiv="Content-Type"\s+content=
"([\w/]+)(;\s+charset=([^\s"]+))?\s+i', $aa, $matches );

```

Funkcia vracia nastavenie kódovania stránky v poli `$matches`. V prípade, že kódovanie je **utf-8** odstraňuje sa diakritika pomocou prevodnej tabuľky definovanej v súbore **tabulka.php** v adresári **scripts** a php funkcie `strtr()`. Diakritika zo zadaného kľúčového slova stránky v premennej `$keyword` je odstránená rovnako.

```

if ( $charset == 'utf-8' ) {
$keywordconv1 = strtr($keyword, $prevodna_tabulka); }

```

V prípade, že **kódovanie stránky** je **windows-1250** alebo **ISO-8859-2** je potrebný prevod z tohto kódovania do **utf-8** a následné odstránenie diakritiky.

```

if ( strtolower($charset) == 'windows-1250' ) {
$str0 = iconv('windows-1250', 'UTF-8', $str0);
$str0 = strtr($str0, $prevodna_tabulka); }

```

V premennej `$str0` sa nachádza text stránky bez diakritiky a v premennej `$keywordconv1` sa nachádza text kľúčového slova bez diakritiky. Takto nastavené premenné je možné následne spracovávať.

Výpočet frekvencie slov na stránke

Skript hodnotí použitie **jednoslovných, dvojslovných alebo trojslovných výrazov**.

Jednoslovné výrazy

Pomocou php funkcie `str_word_count()` uložíť hodnoty do poľa obsahujúceho všetky slová zo vstupného reťazca. Na zistenie počtu opakovaní jednotlivých slov v poli existuje v jazyku PHP verzie 5 jednoduchá funkcia `array_count_values()`. Táto funkcia vráti pole (do premennej `$word_count`), ktoré obsahuje na prvej pozícii každé slovo zo vstupného poľa a na druhej pozícii počet opakovaní tohto slova vo vstupnom poli. [19]

```
$words = str_word_count(strtolower($str),1);  
$word_count = array_count_values($words);
```

```
Array  
(  
    [kaplna] => 16  
    [obce] => 8  
    [obec] => 7  
    [hore] => 3  
    [marca] => 3  
    [domov] => 3  
    [stranku] => 2  
)
```

Výraz Počet opakovaní
 v poli

Obr. 3.8: Príklad výpisu prvkov poľa s počtom ich opakovaní v poli

Nasleduje postupné prechádzanie poľa `$word_count` funkciou `foreach()`. V každom cykle je vypočítaná percentuálna hodnota výskytu výrazu na stránke ako podiel počtu výskytov slova a počtu všetkých slov na stránke (`count($words)`).

```
foreach ($word_count as $key=>$val) {  
    $densitya1 = ($val/count($words))*100; ... }
```

Taktiež je potrebné v každom cykle testovať či je konkrétny prvok poľa zhodný s kľúčovým slovom zadaným užívateľom. Ak je momentálne prechádzaná položka poľa zhodná s kľúčovým slovom, ktoré zadal užívateľ, do premennej `$density2` je uložená percentuálna hodnota výskytu tohto slova a do premennej `$val2` je uložený počet opakovaní tohto slova v poli.

```
if ($key == $keywordconv) { $density2 = $densitya1; $val2 = $val;}
```


Dvojslovné výrazy

Testovanie výskytu dvojslovných výrazov prebieha rovnakým spôsobom ako v prípade jednoslovných výrazov. Pred samotným testovaním a hodnotením počtu opakovaní je potrebné dvojslovné výrazy vytvoriť. Najprv je zistené koľko slov je v vstupnom poli pomocou php funkcie `count()`, nastaví sa cyklus `for()` na počet opakovaní rovný počtu slov vo vstupnom poli a následne **je zložené slovo na pozícii [x] so slovom na pozícii [x+1]**. Takýmto spôsobom sú získané všetky dvojslovné výrazy na stránke vrátené do poľa `$phrase2`.

```
$total = count($words);  
for($x=0; $x<$total; $x++)  
{ $phrase2[$x] = $words[$x]." ".trim($words[$x+1]); }
```

Následné testovanie na výskyt zadanej kľúčovej frázy prebieha rovnakým spôsobom ako v prípade jednoslovných výrazov.

Trojslovné výrazy

Skladanie troch slov prebieha rovnakým spôsobom. Sú skladané zo slov vstupného poľa na pozícii `[x]`, `[x+1]` a `[x+2]`. Následné testovanie na výskyt zadanej kľúčovej frázy prebieha rovnakým spôsobom ako v prípade jednoslovných výrazov.

3.3.3 Test výskytu kľúčovej frázy v rôznych častiach stránky

Z teórie SEO vyplýva, že kľúčové slovo stránky by sa malo nachádzať na týchto dôležitých miestach testovanej webovej stránky:

- V adrese URL,
- V title tagu stránky,
- V meta značkách,
- V nadpisoch stránky,
- V alt tagoch obrázkov,
- V zvýraznenom texte stránky.

Hľadanie slova prebieha vždy pomocou php funkcie `preg_match()` a regulárneho výrazu.

Kľúčové slovo v adrese stránky

V premennej `$url` je uložená adresa testovanej stránky. Pomocou php funkcie `preg_match()` je zisťované či sa v nej nachádza kľúčové slovo uložené v premennej `$keyword2`. Ak adresa stránky obsahuje kľúčové slovo do premennej `$vurl` je uložená hodnota **áno**.

```
if(preg_match("/$keyword/i",$url,$murl)) {$vurl = 'Áno.';}
```

Kľúčové slovo v title tagu stránky

Titulok stránky je hľadaný pomocou regulárneho výrazu `/<title>(.)</title>/i` jeho obsah je vložený do poľa `$m` na pozíciu [1]. Následne sa testuje či obsahuje kľúčové slovo.

```
if(preg_match("<title>(.)</title>/i",$file,$m)) {  
if(preg_match("/$keyword2/i",$m[1],$mtitle)) {$vttitle = 'Áno.';}
```

Kľúčové slovo v meta značkách

Meta značky meta keywords a meta description sú preberané pomocou funkcie `get_meta_tags()`. Konkrétna značka je vyberaná v cykle `foreach()`. Výskyt kľúčového slova je hľadaný pomocou funkcie `preg_match()` a regulárneho výrazu `/$keyword2/i`. V premennej `$keyword2` sa nachádza kľúčové slovo zadané užívateľom.

Kľúčové slovo v nadpisoch stránky

Na webovej stránke sa môžu nadpisy, teda text v html tagoch `<h1>` až `<h6>` vyskytnúť viac krát. Je teda potrebné zistiť počet týchto výskytov a následne testovať každý tento výskyt na prítomnosť kľúčového slova. Výskyt konkrétneho html tagu resp. počet jeho výskytov na zadanej stránke je zisťovaný pomocou php funkcie `preg_match_all()`. Napríklad hlavný nadpis stránky teda text v html tagoch `<h1></h1>` je hľadaný pomocou tejto funkcie a regulárneho výrazu `/<h1>(.)</h1>/i`. Výsledok hľadania sa ukladá do poľa `$mh1`. V prípade, že sa tento nadpis vyskytuje na stránke viac krát ukladá sa vždy na novú pozíciu v tomto poli.

```
if(preg_match_all("<h1>(.)</h1>/i", $file, $mh1)) { ... }
```

Pomocou php funkcie `count()` je zisťovaný počet prvkov poľa a následne nastavený cyklus `for()`, ktorý takto otestuje všetky prvky poľa na výskyt kľúčového slova, resp. všetky nadpisy zadanej stránky.

```

$vh1 = count($mh1);
for($x=0; $x<$vh1; $x++)
{ if(preg_match("/$keyword2/i",$mh1[0][$x],$mh1a))
{ $vh1 = 'Áno.'; } }

```

Rovnakým spôsobom sú testované nadpisy úrovně 2 a 3.

Kľúčové slovo v alternatívnych textoch grafiky stránky

Rovnako ako nadpisy aj alternatívne texty sa môžu na stránke vyskytovať veľa krát. Alternatívny text obrázka je definovaný v značke ako parameter alt="". Výskyt resp. počet výskytov alt textov je zisťovaný pomocou php funkcie preg_match_all() a regulárneho výrazu /alt=(.+)\"/i. Podľa počtu prvkov poľa je nastavený cyklus for().

```

if(preg_match_all("/alt=(.+)\"/i", $file, $malt)) {
$valt = count($malt); for($x=0; $x<$valt; $x++)
{ if(preg_match("/$keyword2/i",$malt[0][$x],$malt1))
{ $valt = 'Áno.'; } }

```

Kľúčové slovo vo zvýraznenom texte stránky

Na zvýraznenie textu stránky sa používa najmä html tag , poprípade dnes už nevalídny tag . Je teda potrebné zaistiť vyhľadanie oboch tagov na stránke a nastaviť cyklus for() na počet opakovaní, ktorý je rovný súčtu výskytov tagu a výskytov tagu . Ďalší priebeh testu je zhodný s testom alternatívnych textov grafiky stránky.

```

if(preg_match_all("<strong>(.)<\/strong>/i", $file, $mstrong)
|| preg_match_all("<b>(.)<\/b>/i", $file, $mstrong2)) {
$strong = count($mstrong) + count($mstrong2); }

```

Každý výsledok testu je uložený do príslušnej premennej aby mohol byť následne zapísaný do databázy.

4 REALIZÁCIA PORTÁLU NA TESTOVANIE A ANALÝZU SEO

Celý portál je založený na programovacom jazyku PHP verzia 5.3.0 s výnimkou AJAX skriptov, ktoré testujú správnosť vyplnenia jednotlivých prvkov formulárov. Ďalej je využívaná databáza MySQL vo verzii 5.1.36 pre zápis všetkých výsledkov testov a ostatných dát. Funkčnú verziu navrhnutého portálu môžete vidieť na adrese <http://www.clinimed.sk/seo/>. **Stáva sa, že portál označí testovanú stránku ako neexistujúcu aj napriek tomu, že táto adresa existuje. Táto skutočnosť je ovplyvnená nižšou kvalitou hostingu, na ktorom je portál momentálne umiestnený a taktiež rýchlosťou pripojenia do internetu užívateľa portálu. Skripty ako také fungujú správne.**

Redakčný systém mnou navrhnutého portálu sa skladá z týchto základných častí:

1. **Príprava databázy** - kap. 4.1
2. **Užívateľské transakcie**: - kap. 4.2
 - registrácia užívateľa - kap. 4.2.1
 - prihlasovanie užívateľa - kap. 4.2.2
 - zmena testovanej adresy webstránky - kap. 4.2.3
 - úprava užívateľských účtov - kap. 4.2.4
 - pripomenutie zabudnutého hesla - kap. 4.2.5
 - odhlásenie užívateľa - kap. 4.2.6
3. **Zloženie portálovej webovej stránky** - kap. 4.3
4. **Testovanie webových stránok neregistrovaného užívateľa** - kap. 4.4
5. **Testovanie webových stránok registrovaného užívateľa** - kap. 4.5
6. **Zobrazenie všetkých výsledkov testov konkrétnej stránky** - kap. 4.6
7. **Odstraňovanie výsledkov testov** - kap. 4.7

Základnou vlastnosťou portálu je **oddelená možnosť testovania ľubovoľnej stránky a testovania stránky registrovaného užívateľa s možnosťou porovnania výsledkov predchádzajúcich testov.**

4.1 Príprava databázy

Prvou vecou, ktorú je potrebné spraviť je vytvoriť si databázu a nastaviť si tabuľky pre ukladanie jednotlivých dát z výsledkov testov a taktiež užívateľských dát ako sú meno, heslo, web stránky užívateľa a podobne.

Je veľmi výhodné vytvoriť si súbor s údajmi o používanej databáze. Tento súbor sa potom jednoducho využije všade tam, kde bude potrebné k datábazovým údajom pristupovať či už z dôvodu vkladania dát do databázy alebo ich výberu. Súbor s týmito údajmi bol nazvaný **config.php**.

4.1.1 Databázové tabuľky

Bolo navrhnutých **6 SQL tabuliek** pre ukladanie všetkých potrebných dát. Súbor, ktorý ich vytvorí bol nazvaný **seo-table.php** a je potrebné ho spustiť vždy pred "inštaláciou" portálu. **Popis SQL tabuliek:**

- **seo_users** - ukladanie dát užívateľov: meno, url adresa stránky, heslo atď.
- **seo_on** - ukladanie výsledkov on page testov registrovaného užívateľa
- **seo_off** - ukladanie výsledkov off page testov registrovaného užívateľa
- **seo_key** - ukladanie výsledkov testov kľúčových slov registrovaného užívateľa
- **seo_on2** - ukladanie výsledkov on page testov neregistrovaného užívateľa
- **seo_off2** - ukladanie výsledkov off page testov neregistrovaného užívateľa
- **seo_key2** - ukladanie výsledkov testov kľúčových slov neregistrovaného užívateľa

Najdôležitejšia položka tabuľky **seo_users** je **id**. Táto položka je generovaná automaticky. Každý nový užívateľ má generované nové a jedinečné id. Pomocou tohto ID je možné prepojiť tabuľku užívateľa a tabuľky výsledkov on page testov, off page testov a testov kľúčových slov. **Ukladanie výsledkov testov neregistrovaného užívateľa má význam v prípade budúceho rozširovania možností portálu.**

id	username	name	surname	email	url_user	url_user2	url_user3
1	martinko61	Martin	Martin	martinko61@gmail.com	www.kaplina.sk	www.igram.sk	www.sme.sk
2	test	test	test	test@test.sk	www.bahon.sk	www.igram.sk	www.sme.sk
3	test2	test2	test2	test2@test2.sk	www.bahon.sk	www.sme.sk	
5	test3	test3	test3	test3@test3.sk	www.bahon.sk	www.igram.sk	

Obr. 4.1: Príklad SQL tabuľky seo_users z prostredia phpMyAdmin

4.2 Uživatelské transakcie

Uživatelské transakcie sú všetky akcie, ktoré súvisia s nastavovaním užívateľského účtu. O všetky tieto akcie sa stará kód v súbore **user-trans.php**. Výpis užívateľských transakcií:

- registrácia užívateľov
- prihlasovanie užívateľov na stránku
- úprava užívateľského účtu
- zmena aktuálne testovanej URL
- pripomenutie zabudnutého hesla
- odhlásenie užívateľa

Skript v súbore **user-trans.php** využíva prepínač switch na zistenie akcie, ktorá má byť spracovaná. V prípade, že bola zvolená napríklad akcia **Vytvoriť účet** vykoná skript akciu definovanú medzi prepínačom a príkazom **break**;

```

if (isset($_REQUEST['akcia'])) {
    switch ($_REQUEST['akcia']) {
        case 'Vytvoriť účet':
            ...
            break;
    } }

```

4.2.1 Registrácia užívateľov

Registračný formulár obsahuje len základné údaje, ktoré sú po vyplnení nahrané do databázy. Formulár sa nachádza v súbore **user-reg.php**.



Vaše údaje

Užívateľské meno:

Meno:

Priezvisko:

Elektronická adresa:

Vaše webstránky

Webstránka č.1:

Webstránka č.2:

Webstránka č.3:

Heslá

Heslo:

Heslo (znova):

Vymazať údaje Vytvoriť účet

Obr. 4.2: Vzhľad registračného formulára

V prvom rade je potrebné skontrolovať správnosť vyplnenia jednotlivých prvkov formulára. Povinné položky sú užívateľské meno, emailová adresa, heslo, potvrdenie hesla a URL adresa užívateľa. Ďalej je potrebné aby sa obsah položiek heslo a potvrdenie hesla rovnali.

```
if ($_POST['username'] != ''  
    and $_POST['email'] != ''
```

```

and $_POST['password'] != ''
and $_POST['password2'] != ''
and $_POST['url_user'] != ''
and $_POST['password'] == $_POST['password2']) { ... }

```

Funkcia redirect - presmerovanie

V prípade, že niektorá z týchto položiek je vyplnená nesprávne je využitá funkciu **redirect(\$url)** definovanú v súbore **redirect.php**. Táto funkcia umožňuje presmerovanie na ľubovoľnú zadanú stránku a využíva na to php funkciu **header()** s parametrom **location**. Taktiež testuje či už boli odoslané hlavičky dokumentu. Premenná **\$url** je stránka, na ktorú má byť uskutočnené presmerovanie.

```

header('Location: http://'. $_SERVER['HTTP_HOST'] .
dirname($_SERVER['PHP_SELF']). '/' . $url);

```

Funkciu redirect(\$url) využívam na viacerých miestach.

Vkladanie dát do databázy

Po otestovaní vstupných parametrov je možné vložiť záznamy o novom užívateľovi do databázy. Je využitý jednoduchý SQL príkaz, kde sa do konkrétnych stĺpcov tabuľky s názvom **seo_users** vkladajú hodnoty z formulára.

```

"INSERT INTO seo_users (username, name, email, url_user,)" .
"VALUES ('".$_POST['username']. "', '".$_POST['name']. "'," .
" '".$_POST['email']. "', '".$_POST['url_user']. "')"
```

Zo vstupného formulára je jasné že užívateľ môže vložiť minimálne jednu a maximálne 3 testované webstránky. V prípade, že do položky formulára **url_user2** alebo **url_user3** nevloží žiadnu URL do SQL tabuľky bude vložená prázdna hodnota.

Heslo je do databázy ukladané v šifrovanej podobe. Využívam dnes už kompromitovanú php funkciu md5(), ktorá využíva 128 bitovú hashovú hodnotu. V prípade rozširovania portálu je potrebné zmeniť túto šifrovaciu metódu.

Po vložení záznamov do databázy je potrebné začať užívateľskú reláciu, inak povedané **prihlásiť užívateľa**. **Session** je premenná, ktorá je aktívna po celú dobu prihlásenia užívateľa. Vďaka tejto premennej je užívateľovi umožnený prístup k jeho záznamom a dátam.

Užívateľská relácia začína vždy príkazom **session_start()**. Následne je nastavených 5 druhov session, ktoré sú využívané pri zobrazovaní dát určených prihlásenému užívateľovi.


```

$_SESSION['id'] = mysql_insert_id($connect);
$_SESSION['username'] = $_POST['username'];
$_SESSION['url_user'] = $_POST['url_user'];
$_SESSION['url_user2'] = $_POST['url_user2'];
$_SESSION['url_user3'] = $_POST['url_user3'];

```

Týmto je užívateľ prihlásený a následne je presmerovaný na úvodnú stránku **index.php**.

4.2.2 Prihlasovanie užívateľov

Každý už registrovaný užívateľ sa môže prihlásiť cez tzv. prihlasovací formulár vyplnením emailovej adresy a hesla. Zadané údaje sú následne spracované pomocou skriptu v súbore **user-trans.php**.



Obr. 4.3: Vzhľad prihlasovacieho formulára

V prípade prihlasovania je hodnota uskutočnenej akcie **Prihlásiť**. V súbore **user-trans.php** je teda vykonaný kód medzi: `case 'Prihlásiť':` a `break;`. V prvom rade je potrebné overiť či boli vyplnené vstupné údaje.

```
if ($_POST['email'] != '' and $_POST['password'] != '') { ... }
```

V prípade, že boli zadané vstupné údaje nasleduje načítanie údajov o užívateľovi z databázy pomocou SQL príkazu **SELECT**. Hľadajú sa všetky záznamy z tabuľky **seo_users**, kde je emailová adresa užívateľa rovná emailovej adrese zadanej do prihlasovacieho formulára a zároveň heslo sa rovná heslu zadanému užívateľom.

```

"SELECT * " .
"FROM seo_users " .
"WHERE email='". $_POST['email'] . "' " .
"AND password='". $_POST['password'] . "'"

```

V prípade, že príkaz vráti výsledok s nula riadkami, znamená to že užívateľ nebol nájdený. Nasleduje presmerovanie na stránku informujúcu o nesprávne zadanych vstupných údajoch.

```
$num_rows = mysql_num_rows($result);  
if ($num_rows == 0) {redirect('index_error.php'); exit(); }
```

V prípade, že vstupné údaje boli zadane správne spustí sa užívateľská relácia príkazom `session_start()` a nastaví sa premenné relácie rovné údajom z databázy.

```
if ($row = mysql_fetch_array($result)) {  
    session_start();  
    $_SESSION['id'] = $row['id'];  
    $_SESSION['username'] = $row['username']; ... }
```

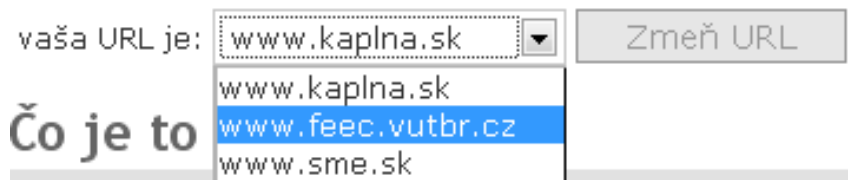
Nie každý užívateľ musí mať nastavené všetky 3 možné testované URL, preto je potrebné pred nastavením **SESSION** pre všetky 3 URL, overiť existenciu nastavovanej URL v databáze. Je využitá php funkcia `isset()`.

```
if(isset($row['url_user2']))  
{ $_SESSION['url_user2'] = $row['url_user2']; }
```

Po úspešnom nastavení všetkých **SESSIONS** resp. úspešnom prihlásení je užívateľ presmerovaný na domovskú stránku **index.php**.

4.2.3 Zmena aktuálne testovanej adresy webstránky

Pri registrácii má užívateľ možnosť zadať 3 adresy webových stránok, ktoré by chcel testovať. Po prihlásení užívateľa je ako `$_SESSION['url_user']` nastavená stránka, ktorú užívateľ zadal na pozícii 1 v registračnom formulári resp. adresa, ktorá sa nachádza v SQL tabuľke `seo_users` v stĺpci `url_user`.



Obr. 4.4: Vzhľad formulára na zmenu aktuálne testovanej webstránky

Spolu s premennou prostredia `$_SESSION['url_user']` sú nastavené ďalšie dve premenné `$_SESSION['url_user2']` a `$_SESSION['url_user3']`, ktoré obsahujú adresy ďalších dvoch stránok zadanych pri registrácii.

```
$_SESSION['url_user'] = $_POST['url_user'];
$_SESSION['url_user2'] = $_POST['url_user2'];
$_SESSION['url_user3'] = $_POST['url_user3'];
```

V prípade, že tieto premenné neboli zadane pri registrácii príslušná premenná neobsahuje žiadnu hodnotu.

Na zmenu aktuálne testovanej adresy stránky slúži formulár tvorený tzv. **select boxom** obsahujúcim adresy webstránok prihláseného užívateľa a tlačítkom potvrdzujúcim zmenu testovanej stránky. O spracovanie užívateľskej požiadavky na zmenu testovanej adresy sa stará kód v súbore **user-trans.php**. V prvom rade je potrebné načítať z databázy adresy webstránok prihláseného užívateľa. Načítanie uskutoční nasledujúci SQL príkaz:

```
SELECT id, username, url_user, url_user2, url_user3
FROM seo_users WHERE id="'. $_SESSION['id'] . '"
```

Následne do premennej \$urlcka uloží hodnotu adresy, ktorú užívateľ potvrdil pomocou formulára na zmenu aktuálne testovanej adresy westránky.

```
$urlcka = $_REQUEST['url'];
```

V prípade, že užívateľ potvrdil **adresu aktuálne sa nachádzajúcu na pozícii 1** nie je potrebné vykonať žiadnu akciu.

```
if ($urlcka == "url1") { }
```

V prípade, že užívateľ potvrdil **adresu aktuálne sa nachádzajúcu na pozícii 2** do premennej prostredia \$_SESSION['url_user'] bude uložená hodnota z SQL tabuľky seo_users zo stĺpca url_user2 a do premennej prostredia \$_SESSION['url_user2'] bude uložená hodnota z SQL tabuľky seo_users zo stĺpca url_user. Premenná prostredia \$_SESSION['url_user3'] ostáva nezmenená.

```
if ($urlcka == "url2") { ...
$_SESSION['url_user'] = $row['url_user2'];
$_SESSION['url_user2'] = $row['url_user'];
$_SESSION['url_user3'] = $row['url_user3']; }
```

V prípade, že užívateľ potvrdil **adresu aktuálne sa nachádzajúcu na pozícii 3** do premennej prostredia \$_SESSION['url_user'] bude uložená hodnota z SQL tabuľky seo_users zo stĺpca url_user3 a do premennej prostredia \$_SESSION['url_user3'] bude uložená hodnota z SQL tabuľky seo_users zo stĺpca url_user. Premenná prostredia \$_SESSION['url_user2'] ostáva nezmenená.

```

if ($urlcka == "url3") { ...
$_SESSION['url_user'] = $row['url_user3'];
$_SESSION['url_user3'] = $row['url_user'];
$_SESSION['url_user2'] = $row['url_user2']; }

```

Týmto je zabezpečená zmena aktuálne testovanej adresy webstránky. Po uskutočnenej zmene nasleduje presmerovanie na domovskú stránku **index.php**.

4.2.4 Úprava užívateľských účtov

Pre zmenu užívateľských údajov je využitý formulár v súbore **user-reg.php**. Skript obsahuje overenie či je už užívateľ prihlásený a teda je nastavená premenná `$_SESSION['id']`. V prípade, že užívateľ nie je prihlásený formulár funguje ako klasický registračný formulár.

```

if (isset($_SESSION['id'])) { ... }

```

Ak je užívateľ prihlásený skript načíta z SQL tabuľky **seo_users** všetky údaje tohto užívateľa príkazom **SELECT** a následne ich uloží do príslušných premenných.

```

SELECT * FROM seo_users WHERE id="'. $_SESSION['id'] . "' ...
$username = $row['username'];
$name = $row['name'];
$email = $row['email']; ...

```

Teraz je možné formulár naplniť údajmi z databázy:

```

<input type="text" class="field" name="name"
id="name" value="<?php echo htmlspecialchars($name); ?>" />

```

Po naplnení údajmi môže nasledovať úprava ľubovoľnej položky ukončená stlačením potvrdzovacieho tlačidla, ktorého hodnota je nastavená ako **Upraviť účet**. Nasleduje spracovanie zadaných údajov pomocou kódu v súbore **user-trans.php**. Skript v prvom rade overí správnosť vyplnenia jednotlivých údajov rovnako ako v prípade registrácie užívateľa. Nasleduje SQL príkaz **UPDATE**, ktorý zabezpečí zmenu užívateľských údajov na hodnoty nastavené vo formulári.

```

UPDATE seo_users SET email="'. $_POST['email'] . "',
name="'. $_POST['name'] . "' WHERE id='". $_SESSION['id']

```

Skript končí nastavením nových premenných prostredia a presmerovaním na stránku **index.php**.

4.2.5 Pripomenutie zabudnutého hesla

Formulár na pripomenutie zabudnutého hesla sa nachádza v súbore **forget.php**. Obsahuje jedinú položku a to emailovú adresu registrovaného užívateľa. Po vyplnení emailovej adresy a odoslaní formulára sa o jeho spracovanie stará kód v súbore **user-trans.php**. V prvom rade je potrebné overiť existenciu zadanej emailovej adresy. Pomocou SQL príkazu **SELECT** je načítané heslo z SQL tabuľky **seo_users**, kde stĺpec **email** obsahuje zadanú hodnotu emailovej adresy. Ak príkaz vráti premennú obsahujúcu 0 riadkov, zadaná emailová adresa sa v DB nenachádza a užívateľ je presmerovaný na stránku informujúcu o chybe **index_error3.php**.

```
SELECT password FROM seo_users WHERE email='".$$_POST['email']."'...
$num_rows = mysql_num_rows($vysledok);
if ($num_rows == 0) { redirect('index_error3.php'); exit(); }
```

Ak zadaná emailová adresa existuje, je na ňu zaslané zistené heslo a ostatné informácie pomocou php funkcie **mail()**.

```
$predmet = 'Pripomenutie hesla z SEO portálu.';
$headers .= "From: SEO portál\r\n" ;
$sprava .= "Vaše heslo je: " . $row['password'] .
mail($_POST['email'], $predmet, $sprava, $headers)
```

Po úspešnom odoslaní emailu je užívateľ presmerovaný na stránku s prihlasovacím formulárom **login.php**.

4.2.6 Odhlásenie užívateľa

Odhlasovanie užívateľa nasleduje okamžite po kliknutí na odkaz **user-trans.php?akcia=odhlasit**. Odhlasovanie ukutočňuje skript v súbore **user-trans.php**. Odhlásenie užívateľa znamená ukončiť užívateľskú reláciu. Sú využité nasledujúce príkazy:

```
session_start();
session_unset();
session_destroy();
```

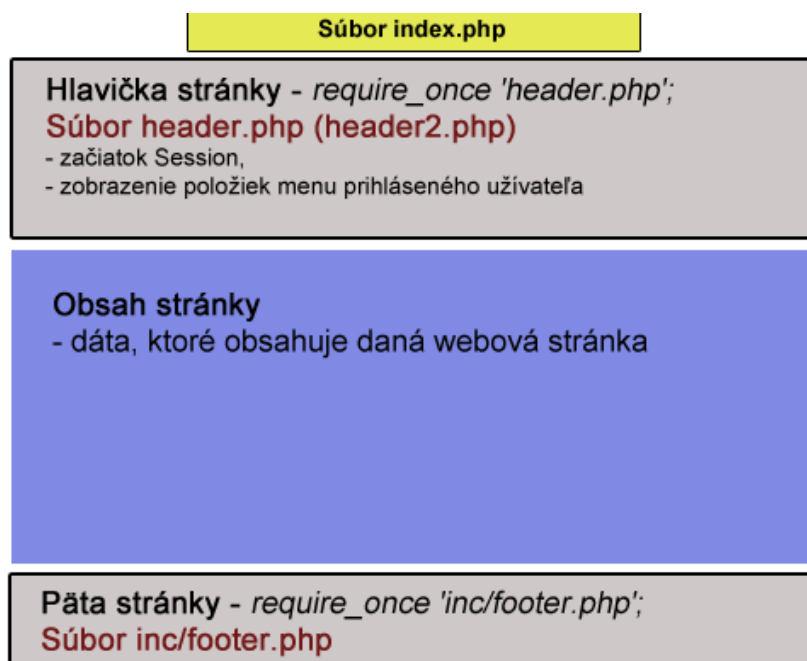
Po úspešnom odhlásení je užívateľ presmerovaný na domovskú stránku **index.php**.

4.3 Zloženie webovej stránky

Každá webová stránka na portále sa skladá z troch hlavných častí:

1. Hlavička stránky - súbor `header.php`
2. Obsah stránky
3. Päta stránky - súbor `inc/footer.php`

Jedinou výnimkou sú tzv. **transakčné stránky**, ktorých úlohou je vykonať určitú akciu napr. prihlásenie užívateľa alebo odstránenie záznamu z databázy. Ihneď po vykonaní požadovanej akcie nastáva presmerovanie na inú stránku portálu. Z tohto dôvodu táto stránka nemusí obsahovať hlavičku ani päť stránky.



Obr. 4.5: Grafické znázornenie zloženia stránky `index.php`

4.3.1 Hlavička stránky - `header.php`

Na prvom mieste obsahuje kód pre začiatok užívateľskej relácie v prípade, že je užívateľ prihlásený `<?php session_start(); ?>`. Ďalej obsahuje html kód hlavného menu. V prípade, že je užívateľ prihlásený zobrazí sa mu informácia o jeho prihlásení a taktiež menu odkazujúce ho na nové testy, výsledky jeho testov, úpravu užívateľského účtu a odhlásenie z portálu.

Užívateľské informácie

Užívateľské informácie a ponuka zmeny aktuálne testovanej URL sa zobrazujú v časti stránky medzi html tagmi: `<div class="header-breadcrumbs"> a </div>`. Overenie správnosti užívateľského prihlásenia spočíva v testovaní existencie premennej `$_SESSION['username']`.

```
if (isset($_SESSION['username'])) { ... }
```

V prípade, že táto premenná existuje užívateľovi je zobrazená informácia o jeho užívateľskom mene:

```
echo '<p>Ste prihlásený ako: ' . $_SESSION['username'];
```

a taktiež select formulár na zmenu aktuálne testovanej URL:

```
echo 'a vaša URL je: <select class="combo" name="url">';  
echo '<option value="url1">' . $_SESSION['url_user'] . '</option>  
</select>';  
echo '<input type="submit" name="akcia" class="button"  
value="Zmeň URL" />';
```



Obr. 4.6: Ukážka časti stránky zobrazujúcej informácie o aktuálne prihlásenom užívateľovi

Užívateľské menu

V prípade, že užívateľ nie je prihlásený, teda nie je nastavená premenná `$_SESSION['username']` je zobrazené menu s ponukou prihlásenia, odkazom na registračný formulár a prihlasovacím formulárom.

```
if (!isset($_SESSION['id'])) {  
echo '<dt><a href="login.php">Prihlásiť sa</a></dt>  
<dt><a href="user-reg.php">Registrácia</a></dt>';  
require_once 'inc/login-form.php'; }
```

Vaše testy:
Prihlásiť sa
Registrácia
Prihláste sa
Váš email:
<input type="text"/>
Vaše heslo:
<input type="password"/>
<input type="button" value="Prihlásiť"/>
Zabudli ste heslo?

Obr. 4.7: Vzhľad užívateľského menu neprihláseného užívateľa

Ak je užívateľ prihlásený teda je nastavená premenná `$_SESSION['username']` zobrazí sa mu menu s ponukou nových testov, výsledky jeho testov, úpravu užívateľského účtu a odhlásenie z portálu.

```
if (!isset($_SESSION['id'])) { ... } else {
echo '<dl class="nav3-grid">';
echo '<dt><a href="user-test.php?akcia=ontest">Nový test(On)</a></dt>';
echo ' </dl>'; }
```

Vaše testy:
Nový test (On)
Nový test (Off)
Nový test (Keywords)
Výsledky (On)
Výsledky (Off)
Výsledky (Keywords)
Užívateľské menu:
Upraviť účet
Návody
Odhlásiť sa

Obr. 4.8: Vzhľad užívateľského menu prihláseného užívateľa

4.3.2 Obsah stránky

Obsah stránky môže byť tvorený textom, formulárom alebo tabuľkou výsledkov vždy medzi html tagmi `<div class="main-content">` a `</div>`.

4.3.3 Päta stránky - inc/footer.php

Obsahuje údaje o stránke a jej tvorcovi a ukončovacie html tagy.

4.4 Testovanie webových stránok neregistrovaného užívateľa

Neregistrovaný užívateľ si môže vybrať z troch druhov testov a testovať ľubovoľnú stránku. Je potrebné vyplniť vstupný formulár obsahujúci adresu stránky a iné základné informácie a nastavenia.

4.4.1 On page SEO test

Vstupný formulár je definovaný v súbore **on.php**. Obsahuje html kód s formulárom s jedinou položkou URL adresa testovanej stránky. Po vyplnení formulára a jeho odoslaní sa o spracovanie a zobrazenie výsledkov stará súbor **result.php**. Tento súbor obsahuje v prvom rade php funkciu `include()` pomocou, ktorej som schopný vložiť do kódu stránky súbor s nastavením databázy **config.php**. Hlavičku stránky resp. súbor **header.php** vkladám pomocou php funkcie `require_once()`.

```
require_once 'header.php';
include ("config.php");
```

Súbor ďalej obsahuje prepínač `switch()`, pomocou ktorého zisťujem aká akcia bola vykonaná resp. aký test si užívateľ zvolil.

```
if (isset($_REQUEST['akcia'])) {
switch ($_REQUEST['akcia']) { ... } }
```

V prípade, že bola uskutočnená akcia Zisti ON realizuje sa časť skriptu medzi case 'Zisti ON': a `break`; . V prvom rade je vyžadovaný súbor, ktorý obsahuje skript s jednotlivými testami a to **scripts/seo-on.php**. Tento súbor bol podrobne opísaný v kapitole On page SEO testovací skript. **Stará sa o zisťovanie jednotlivých SEO faktorov a ukladá výsledky testov do príslušných premenných.**

```
require ("scripts/seo-on.php");
```

Všetky výsledky testov, resp. príslušné premenné sú zapísané do SQL tabuľky `seo_on2`. Je to výhodné v prípade budúceho rozširovania možností portálu. Napríklad ponúknuť užívateľom zobrazenie výsledkov všetkých uskutočnených testov. Taktiež je dobré mať prehľad o testovaných stránkach. Zápis výsledkov je uskutočnený príkazom `INSERT INTO`. Do odpovedajúceho stĺpca tabuľky vloží obsah premennej teda výsledok testu. Napríklad do tabuľky `seo_on2` do stĺpca **time** vloží obsah premennej `$cas`, ktorá obsahuje čas testu.

```
"INSERT INTO seo_on2 (time, url, chyby, titulok, ... )" .  
"values ('$cas', '$url', '$chyby', '$titulok', ... )" .
```

Po úspešnom vložení zistených výsledkov nasleduje ich načítanie a zobrazenie. Je využitý príkaz `SELECT`. Načíta sa posledný riadok z tabuľky `seo_on2`, kde adresa stránky resp. stĺpec **url** obsahuje adresu testovanej stránky.

```
SELECT * FROM seo_on2 WHERE url LIKE '%" . $nadpis1 . "%'
```

Zistené hodnoty postupne sú vkladané do nových premenných pomocou php funkcie `mysql_fetch_assoc()`.

```
while ($row = mysql_fetch_assoc($vysledky)) {  
    $time = $row["time"];  
    $url = $row["url"]; ... }
```

Takto zistené hodnoty je možné zapísať do tabuľky výsledkov a zobrazíť ako výsledok testov. Každý riadok tabuľky odkazuje taktiež na stránku s návodom. Ak je zistená chyba testu konkrétneho SEO faktora je tento riadok zobrazený červeným písmom.

```
echo '<tr><td>1</td><td>Titulok stránky:</td><td>'. $vysledky3;  
echo '</td><td><a href="navody.php?akcia=on#title">Návod</a>  
</td></tr>';
```

Test č.:	Názov testu:	Výsledok testu:	Návody
1	Titulok stránky:	Titulok stránky je vyplnený a je v poriadku. (18 znakov).	Návod
2	Nadpisy stránky:	Hlavný nadpis stránky máte definovaný správne.	Návod
3	Roboti:	Správne nastavený tag meta robots.	Návod
4	Meta keywords:	Príliš veľa keywordov. (25 slov)	Návod
5	Meta description:	Príliš dlhý meta description. (309 znakov)	Návod
6	Súbor robots.txt:	Súbor robots.txt existuje.	Návod
7	Súbor sitemap.xml:	Súbor sitemap.xml neexistuje.	Návod
8	Veľkosť stránky:	Veľkosť stránky je v poriadku. (11765 bytov)	Návod
9	Validnosť stránky:	Stránka nie je XHTML validna (12 chýb). Vaše výsledky	Návod
10	Iframe objekty:	Žiadne rámce.	Návod
11	Tabulky:	Žiadne tabulky.	Návod
12	Štrukturovanie textu:	Využívajte viac odstavcov..	Návod
13	Alt texty:	Správne definované alternatívne texty pre obrázky.	Návod
14	Inline CSS štýly:	Nájdené inline CSS štýly.	Návod

Ostatné informácie o stránke:

Počet chýb na stránke je **6 zo 14 testov**. Počet liniek na stránke je **32**. Čas testu: **13:12:8**
26.4.2010. Váš test prebehol v poriadku.

Obr. 4.9: Príklad výsledku On page SEO testu stránky www.feec.vutbr.cz

4.4.2 Off page SEO test

Vstupný formulár je definovaný v súbore **off.php**. Obsahuje html kód s formulárom a položkami:

- URL adresa testovanej stránky,
- Kľúčové slovo hľadania - slovo, pomocou ktorého je hľadaná zadaná stránka vo vyhľadávачi,
- Hĺbka vyhľadávania - počet výsledkov na SERP, v ktorých sa má hľadať zadaná stránka,
- Jazyk vyhľadávania,

- Checkbox na zahrnutie vyhľadávača bing do výsledkov vyhľadávania.

Po vyplnení formulára a jeho odoslaní sa o spracovanie a zobrazenie výsledkov stará php kód v súbore **result.php**. Pracuje podobne ako v prípade On page SEO testov. Off page SEO testy uskutočňuje skript v súbore **scripts/seo-off.php**, ktorý ukladá výsledky do príslušných premenných. Tie sú následne zapísané do SQL tabuľky seo_off2. Tento súbor bol podrobne opísaný v kapitole **Off page SEO testovací skript**.

```
include ("scripts/seo-off.php");
INSERT INTO seo_off2(url,pagerank,...) values('$url','$pagerank',...)
```

Zapísané výsledky sú následne načítané a zobrazené pomocou prehľadnej tabuľky obsahujúcej tiež odkazy na návody k jednotlivým testom.

```
SELECT * FROM seo_off2 WHERE url LIKE '%" . $nadpis1 . "%'"; ...
$vysledkyb = mysql_query( $select_off ) or die( mysql_error() );
while ($row2 = mysql_fetch_assoc($vysledkyb)) {
$url = $row2["url"]; ... }
```

Vaše kľúčové slovo(á): **feec vutbr**

Hodnotenie off page faktorov stránky www.feec.vutbr.cz

Test č.:	Názov testu:	Výsledok testu:	Návody
1.	Google page rank:	5	Návod
2.	Seznam rank:	70	Návod
3.	Alexa rank:	45322	Návod
4.	Alexa spätné odkazy:	143	Návod
5.	Google spätné odkazy:	278	Návod
6.	Google zaindexované stránky:	15,500	Návod
7.	Pozícia stránky na google:	1	Návod
8.	Pozícia stránky na seznam:	1	Návod
9.	Pozícia stránky na bing:	1	Návod

Obr. 4.10: Príklad výsledku Off page SEO testu stránky www.feec.vutbr.cz so zadávaným kľúčovým slovom feec vutbr

4.4.3 Test klíčových slov stránky

Vstupný formulár je definovaný v súbore **key.php**. Obsahuje html kód s formulárom a položkami:

1. URL adresa testovanej stránky.
2. Kľúčové slovo stránky.

Po vyplnení formulára a jeho odoslání sa o spracovanie a zobrazenie výsledkov stará súbor **result.php**. Pracuje podobne ako v prípade On a Off page SEO testov. Testy klíčových slov uskutočňuje skript v súbore **scripts/seo-key.php**, ktorý ukladá výsledky testov do príslušných premenných. Tie sú následne zapísané do SQL tabuľky **seo_key2**. Tento súbor bol podrobne opísaný v kapitole skript testujúci kľúčové slová.

```
include ("scripts/seo-key.php");  
$sql3 = "INSERT INTO seo_key2 ( time, slovo, ... )" .  
"values ( '$cas', '$keyword', ... )";
```

Zapísané výsledky sú následne načítané a zobrazené pomocou prehľadnej tabuľky a textu informujúcom o percentuálnom výskyte kľúčového výrazu na stránke a počte všetkých slov stránky.

Hodnotenie kľúčových slov stránky www.feec.vutbr.cz

- Čas testu 14:34:32 26.4.2010
- Vaše kľúčové slovo je **vut** a jeho výskyt na stránke je: **1.69** percent.
- Počet jeho výskytov je: **9**
- Počet všetkých slov na stránke je: **533**
- **Zvýš výskyt kľúčového slova na stránke.**
- **Podrobný návod k testu kľúčových slov vašej stránky**

Hodnotenie použitia kľúčového slova (vut) na vašej stránke www.feec.vutbr.cz

Test č.:	Testovaná pozícia:	Hodnotenie výskytu:
1.	V texte url:	Áno.
2.	V titulku stránky:	Áno.
3.	V meta keywords:	Áno.
4.	V meta description:	Áno.
5.	V nadpise úroveň 1:	Nie.
6.	V nadpise úroveň 2:	Nadpis úroveň 2. nevyplnený.
7.	V nadpise úroveň 3:	Nadpis úroveň 3. nevyplnený.
8.	V alt texte obrázkov:	Nie.
9.	V zvýraznenom texte:	Nie.

Obr. 4.11: Príklad výsledku testu kľúčových slov stránky www.feec.vutbr.cz so zadným kľúčovým slovom vut

4.5 Testovanie webových stránok registrovaného užívateľa

Všetky formuláre na zadanie vstupných údajov sa nachádzajú v súbore **user-test.php**. Súbor využíva prepínač **switch** pre zistenie testu vybraného užívateľom.

```
if (isset($_REQUEST['akcia'])) {  
    switch ($_REQUEST['akcia']) { ... }
```

O spracovanie údajov z formulárov sa stará kód v súbore **zapis-user.php**. Obsahuje kód zabezpečujúci zápis zistených výsledkov testov do databázy a ich následné zobrazenie.

4.5.1 On page SEO test

Test prebieha rovnako ako v prípade neregistrovaného užívateľa. Jediný rozdiel je v ukladaní výsledkov do databázy a ich následnom výbere. Súbor taktiež obsahuje overenie či je užívateľ naozaj prihlásený a teda je nastavená premenná `$_SESSION['id']`. Týmto spôsobom sa dá zabrániť neoprávnenému vstupu na stránku určenú výhradne registrovaným užívateľom.

```
if(isset($_SESSION['id'])) { ... }
else { exit('<p class="red">Nemate opravenie!</p>'); }
```

Ďalšou dôležitou vlastnosťou je predvyplnenie formulára url adresou stránky, ktorá je aktuálne nastavená ako `$_SESSION['url_user']`. Nemennosť takto nastavenej URL je zaručená vlastnosťou `readonly="readonly"`.

```
<input readonly="readonly" name="url"
value="<?php echo $_SESSION['url_user']; ?>" type="text" />
```

On page SEO testy uskutočňuje skript v súbore **scripts/seo-on.php**, ktorý ukladá výsledky do príslušných premenných. Tieto premenné môžu byť následne zapísane do SQL tabuľky s názvom `seo_on`. Tabuľka `seo_on` obsahuje dôležitý stĺpec `on_id`. Do tohto stĺpca je uložená hodnota, vyjadrujúca **id aktuálne prihláseného užívateľa**. ID užívateľa sa nachádza v premennej prostredia `$_SESSION['id']`.

```
require ("scripts/seo-on.php"); ...
INSERT INTO seo_on ( time, on_id, url, chyby)
values ('$cas', '". $_SESSION['id'] ."', '$url', '$chyby')
```

Nasleduje výber výsledkov z databázy a ich zobrazenie spolu s odkazmi na návody k testom. Nasledujúci kód predstavuje výber všetkých hodnôt z tabuľky `seo_on`, kde adresa stránky je podobná adrese stránky aktuálnej relácie a zároveň ID je rovné ID aktuálnej relácie.

```
SELECT * FROM seo_on WHERE url LIKE '%" . $_SESSION['url_user'] .
"%' AND on_id='" . $_SESSION['id'] . "' ORDER BY time
```

Výpis zistených výsledkov je rovnaký ako v prípade neregistrovaného užívateľa.

4.5.2 Off page SEO test

Postup je rovnaký ako v prípade On page SEO testu registrovaného užívateľa. Rozdiely sú len v získavaní výsledkov testov, o ktoré sa stará súbor **scripts/seo-off.php** a zápise do databázy - je využitá SQL tabuľka s názvom `seo_off`. Po zistení off

page SEO faktorov sú výsledky zapísané do tabuľky `seo_off` s `off_id` rovným ID aktuálne prihláseného užívateľa. Nasleduje výpis zistených výsledkov a odkazov k podrobným návodom. Zápis výsledkov do tabuľky `seo_off`:

```
require ("scripts/seo-off.php"); ...  
INSERT INTO seo_off ( off_id, time, time2, slovo )  
values ('". $_SESSION['id'] ."', '$cas', '$time', '$send')
```

Výber zistených výsledkov z SQL tabuľky `seo_off`:

```
SELECT * FROM seo_off WHERE url LIKE '%" . $_SESSION['url_user'] . "%'  
AND off_id='" . $_SESSION['id'] . "' ORDER BY time
```

4.5.3 Test kľúčových slov stránky

Postup je rovnaký ako v prípade On a Off page SEO testu registrovaného užívateľa. Rozdiely sú len v získavaní výsledkov testov, o ktoré sa stará súbor **scripts/seo-key.php** a zápise do databázy - je využitá SQL tabuľka s názvom `seo_key`. Po zistení výsledkov testov kľúčových slov sú výsledky zapísané do tabuľky `seo_key` s `key_id` rovným ID aktuálne prihláseného užívateľa. Nasleduje výpis zistených výsledkov a odkazu k podrobným návodom. Zápis výsledkov do tabuľky `seo_key`:

```
require ("scripts/seo-key.php"); ...  
INSERT INTO seo_key (key_id, time, time2, slovo )  
values ('". $_SESSION['id'] ."', '$cas', '$time', '$keyword' )
```

Výber zistených výsledkov z SQL tabuľky `seo_key`:

```
SELECT * FROM seo_key WHERE url LIKE '%" . $_SESSION['url_user']  
%' AND key_id='" . $_SESSION['id'] . "' ORDER BY time
```

4.6 Zobrazenie všetkých výsledkov testov konkrétnej stránky

O zobrazovanie všetkých výsledkov testov stránky, ktorá je aktuálne nastavená ako `$_SESSION['url_user']` sa stará kód v súbore **vyber.php**. Toto zobrazenie umožňuje užívateľom portálu porovnávať výsledky testov ich stránok a získať tak prehľad o uskutočnených zmenách. Na zobrazovanie výsledkov do prehľadných tabuliek využívam špeciálny mechanizmus jazyka php. Hlavička tabuľky je definovaná medzi značkami `<<<EOD` a `EOD`; . Kód hlavičky je následne uložený do premennej napr. `$zahlavie`.


```

$zahlavie =<<<EOD
<table>
    <tr>
<th>Čas testu:</th>
<th>Zmazať?</th>
<th>Titulok stránky:</th>
    </tr>
EOD;

```

Pre načítanie dát z tabuľky je využitá php funkcia `SELECT`, ktorá načíta dáta z tabuľky `seo_on`, `seo_off` alebo `seo_key` podľa požiadavky užívateľa pre url adresu rovnú `$_SESSION['url_user']` a `on_id` rovné `$_SESSION['id']`. Zároveň výsledky zoradí podľa času testu. Príklad načítania dát z tabuľky `seo_on`:

```

SELECT * FROM seo_on WHERE url LIKE '%" . $_SESSION['url_user'] .
"%'" AND on_id='" . $_SESSION['id'] . "'" ORDER BY time

```

Telo tabuľky postupne je napĺňané dátami z databázy vždy pre ID aktuálne prihláseného užívateľa a adresu aktuálne testovanej stránky. Napĺňanie tabuľky dátami uskutočňuje php funkcia `mysql_fetch_array()`, ktorá v každom cykle naplní jeden riadok tabuľky pokiaľ sa v databáze nachádzajú dáta. Príklad naplnenia tabuľky dátami z databázy:

```

while ( $row = mysql_fetch_array( $vysledky ) ) {
$vysledky4 = $row["nadpis"]; $vysledky5 = $row["robots"];
$data1 .=<<<EOD
    <tr><td>$vysledky4</td><td>$vysledky5</td>
</tr> EOD; }

```

Päta tabuľky obsahuje ukončovacie html tagy.

```

$pata1 ="</table>";

```

Teraz je možné hlavičku, telo a päta tabuľky spojiť do jednej premennej, a následne jej obsah vypísať do okna prehliadača. **Tento postup je využívaný pre zobrazenie výsledkov všetkých testov.**

```

$result1 =<<<RESULT1
    $zahlavie1
    $data1
    $pata1
RESULT1;
echo $result1;

```

4.6.1 Zobrazenie všetkých výsledkov on page SEO testov konkrétnej stránky

Výsledky on page SEO testov sú zobrazené do **3 tabuliek** najmä z dôvodu prehľadnosti. Jednotlivé testy sú odlišené podľa času ich uskutočnenia. Prvá tabuľka obsahuje hodnotenie údajov z hlavičky stránky vždy pre adresu stránky aktuálne nastavenú ako \$_SESSION['url_user'] a ID prihláseného užívateľa. Taktiež obsahuje odkaz na zmazanie konkrétneho výsledku.

Hodnotenie údajov z hlavičky stránky:

Čas testu:	Zmazať?	Titulok stránky:	Nadpisy stránky:	Roboti:	Meta keywords:	Meta description:
14:38:13 9.5.2010	[Zmazať]	Titulok stránky je vyplnený a je v poriadku. (18 znakov).	Hlavný nadpis stránky máte definovaný správne.	Správne nastavený tag meta robots.	Príliš veľa keywordov. (25 slov)	Príliš dlhý meta description. (309 znakov)
14:38:23 9.5.2010	[Zmazať]	Titulok stránky je vyplnený a je v poriadku. (18 znakov).	Hlavný nadpis stránky máte definovaný správne.	Správne nastavený tag meta robots.	Príliš veľa keywordov. (25 slov)	Príliš dlhý meta description. (309 znakov)

Obr. 4.12: Príklad tabuľky porovnávajúcej hodnotenia údajov on page SEO faktorov z hlavičky stránky www.feec.vutbr.cz

Ďalšia tabuľka obsahuje hodnotenie on page SEO faktorov z tela stránky. Posledná tabuľka obsahuje hodnotenie ostatných on page SEO faktorov ako sú hodnotenie existencie súborov robots.txt, sitemap.xml a hodnotenie veľkosti html kódu danej stránky.

Hodnotenie údajov z tela stránky:

Čas testu:	Validnosť stránky:	Iframe objekty:	Tabuľky:	Štruktúra textu:	Počet liniek:	Alt texty:	Inline CSS štýly:
14:38:13 9.5.2010	Stránka nie je XHTML validná (12 chýb). Vaše výsledky	Žiadne rámce.	Žiadne tabuľky.	Využívajte viac odstavcov.	34	Správne definované alternatívne texty pre obrázky.	Nájdene inline CSS štýly.
14:38:23 9.5.2010	Stránka nie je XHTML validná (12 chýb). Vaše výsledky	Žiadne rámce.	Žiadne tabuľky.	Využívajte viac odstavcov.	34	Správne definované alternatívne texty pre obrázky.	Nájdene inline CSS štýly.

Obr. 4.13: Príklad tabuľky porovnávajúcej hodnotenia údajov on page SEO faktorov z tela stránky www.feec.vutbr.cz

Hodnotenie ostatných údajov stránky:

Čas testu:	Súbor robots.txt:	Súbor sitemap.xml:	Veľkosť stránky:
14:38:13 9.5.2010	Súbor robots.txt existuje.	Súbor sitemap.xml neexistuje.	Veľkosť stránky je v poriadku. (14064 bytov)
14:38:23 9.5.2010	Súbor robots.txt existuje.	Súbor sitemap.xml neexistuje.	Veľkosť stránky je v poriadku. (14064 bytov)

Obr. 4.14: Príklad tabuľky porovnávajúcej hodnotenia údajov ostatných on page SEO faktorov stránky www.feec.vutbr.cz

4.6.2 Zobrazenie všetkých výsledkov off page SEO testov konkrétnej stránky

Výsledky testov sú zobrazené do **3 tabuliek**. Prvá tabuľka obsahuje údaje zadané užívateľom pri testovaní stránky tj. kľúčové slovo vyhľadávania, krajinu hľadania, jazyk hľadania a tiež odkaz na zmazanie konkrétneho výsledku. Druhá tabuľka obsahuje výsledky testov page rankov, spätných odkazov a zaindexovaných stránok. Tretia tabuľka obsahuje výsledky testov pozície zadanej stránky vo vyhľadávačoch google, seznam a bing. Jednotlivé testy sú odlíšené časom ich uskutočnenia.

Vaše zadané údaje:

Čas testu:	Zmazať?	Kľúčové slovo:	Jazyk/krajina hľadania:	Hĺbka vyhľadávania:
14:33:40 9.5.2010	[Zmazať]	feec+vutbr	CZ	10
14:34:0 9.5.2010	[Zmazať]	feec+vutbr	CZ	10

Page ranky, spätné odkazy a zaindexované stránky:

Čas testu:	Google page rank:	Seznam rank:	Alexa rank:	Alexa spätné odkazy:	Google spätné odkazy:	Google indexované stránky:
14:33:40 9.5.2010	5	70	46651	143	278	3,500
14:34:0 9.5.2010	5	70	46651	143	278	3,500

Pozícia stránky na google, seznam a bing:

Čas testu:	Pozícia stránky na google:	Pozícia stránky na seznam:	Pozícia stránky na bingm:
14:33:40 9.5.2010	1	1	1
14:34:0 9.5.2010	1	1	1

Obr. 4.15: Príklad tabuliek porovnávajúcich hodnotenia off page SEO faktorov pre stránku www.feec.vutbr.cz

4.6.3 Zobrazenie všetkých výsledkov testov kľúčových slov konkrétnej stránky

Výsledky testov zobrazujem do 4 **prehľadných tabuliek**. Prvá tabuľka obsahuje údaje zadané užívateľom pri testovaní stránky tj. kľúčové slovo stránky a tiež odkaz na zmazanie konkrétneho výsledku. Druhá tabuľka obsahuje hodnotenie výskytu kľúčových slov. Tretia tabuľka obsahuje hodnotenie výskytu kľúčového slova v hlavičke stránky a štvrtá tabuľka hodnotenie výskytu kľúčového slova v tele stránky. Jednotlivé testy sú odlíšené časom ich uskutočnenia.

Vaše zadané údaje:

Čas testu:	Zmazať?	Kľúčové slovo:
14:39:28 9.5.2010	[Zmazať]	vut
14:39:40 9.5.2010	[Zmazať]	vut

Hodnotenie kľúčových slov:

Čas testu:	Počet výskytov kľúčového slova:	Hustota kľúčového slova [%]:	Počet slov na stránke:	Výsledok testu:
14:39:28 9.5.2010	11	1.83	602	Chyba
14:39:40 9.5.2010	11	1.83	602	Chyba

Obr. 4.16: Príklad tabuliek porovnávajúcich vstupné údaje užívateľa a výsledky hodnotenia výskytu kľúčového slova vut pre stránku www.feec.vutbr.cz

Hodnotenie použitia kľúčového slova v hlavičke stránky:

Čas testu:	V texte url:	V titulku stránky:	V meta keywords:	V meta description:
14:39:28 9.5.2010	Áno.	Áno.	Áno.	Áno.
14:39:40 9.5.2010	Áno.	Áno.	Áno.	Áno.

Hodnotenie použitia kľúčového slova v tele stránky:

Čas testu:	V nadpise úroveň 1:	V nadpise úroveň 2:	V nadpise úroveň 3:	V alt texte obrázkov:	V zvýraznenom texte:
14:39:28 9.5.2010	Nie.	Nadpis úroveň 2, nevyplnený.	Nadpis úroveň 3, nevyplnený.	Nie.	Nie.
14:39:40 9.5.2010	Nie.	Nadpis úroveň 2, nevyplnený.	Nadpis úroveň 3, nevyplnený.	Nie.	Nie.

Obr. 4.17: Príklad tabuliek porovnávajúcich výsledky hodnotenia výskytu kľúčového slova vut na špecifických miestach stránky www.feec.vutbr.cz

4.7 Odstraňovanie výsledkov testov

V prípade, že prihlásený užívateľ chce odstrániť výsledky uskutočnených testov má dve možnosti:

1. odstrániť všetky výsledky testov konkrétnej stránky,
2. odstrániť špecifický výsledok testu konkrétnej stránky.

Užívateľ môže zobrazovať a teda aj odstraňovať záznamy pre stránku, ktorá je aktuálne nastavená ako `$_SESSION['url_user']`. V oboch prípadoch sa o odstraňovanie výsledkov stará kód v súbore **delete.php**.

4.7.1 Odstraňovanie všetkých výsledkov testov konkrétnej stránky

Kód v súbore **delete.php** obsahuje v prvom rade prepínač switch, ktorý zisťuje, ktoré hodnoty majú byť odstránené resp. z ktorej SQL tabuľky. V prípade, že bola zadaná požiadavka v tvare "zmazaton", teda užívateľ klikol na linku v tvare: `delete.php?akcia=zmazaton` skript vykoná akciu definovanú medzi case 'zmazaton': a break;. Je teda jasné, že je požadované zmazať všetky hodnoty výsledkov on page

SEO testov pre stránku aktuálne nastavenú ako `$_SESSION['url_user']` a ID užívateľa nastaveného v premennej `$_SESSION['id']`. Požadovanú operáciu vykoná SQL príkaz:

```
DELETE FROM seo_on WHERE url LIKE '%"$_SESSION['url_user']'.  
"% ' AND on_id='"$_SESSION['id']"."' ORDER BY time
```

Ihneď po odstránení hodnôt z databázy nasleduje presmerovanie na stránku zobrazujúcu všetky výsledky testov.

```
redirect('vyber.php?akcia=on');
```

Rovnakým spôsobom sú odstraňované aj ostatné výsledky či už v prípade off page SEO testov alebo v prípade testov kľúčových slov.

4.7.2 Odstraňovanie špecifického výsledku testu konkrétnej stránky

Ostraňovanie špecifického testu vychádza z jeho základnej vlastnosti a to z presného času, v ktorom bol test vykonaný. V prvom rade je potrebné uložiť tento čas do databázy v tvare, ktorý môže byť využitý ako parameter pre odstraňovanie záznamu, to znamená bez medzier. Odstránenie medzier z časového údaju je jednoduché a využíva funkciu `str_replace()`.

```
$time = str_replace(" ", "", "$cas");
```

Táto premenná môže byť následne uložená do SQL tabuľky a stĺpca s názvom `time2`. Odkaz na odstránenie konkrétneho riadka tabuľky sa skladá z definície potrebnej akcie `zmazat-selected-on` a parametra `time`, ktorý predstavuje čas uskutočneného testu bez medzier. V tomto prípade je aktuálny čas testu uložený v premennej `$vysledky2a`.

```
delete.php?akcia=zmazat-selected-on&time=$vysledky2a
```

O samotné odstraňovanie hodnôt sa opäť stará kód v súbore **delete.php**. V prípade, že bola zvolená akcia `zmazat-selected-on` s príslušným parametrom, ktorý predstavuje čas testu uskutoční sa príkaz **DELETE**, teda odstránenie riadka príslušnej SQL tabuľky v tomto prípade `seo_on`, v ktorom je url rovná adrese stránky aktuálne nastavenej ako `$_SESSION['url_user']` a `on_id` rovné ID užívateľa nastaveného v premennej `$_SESSION['id']`. Zároveň je odstránený len riadok, v ktorom je čas testu teda `time2` rovný času aktuálne odstraňovaného testu.

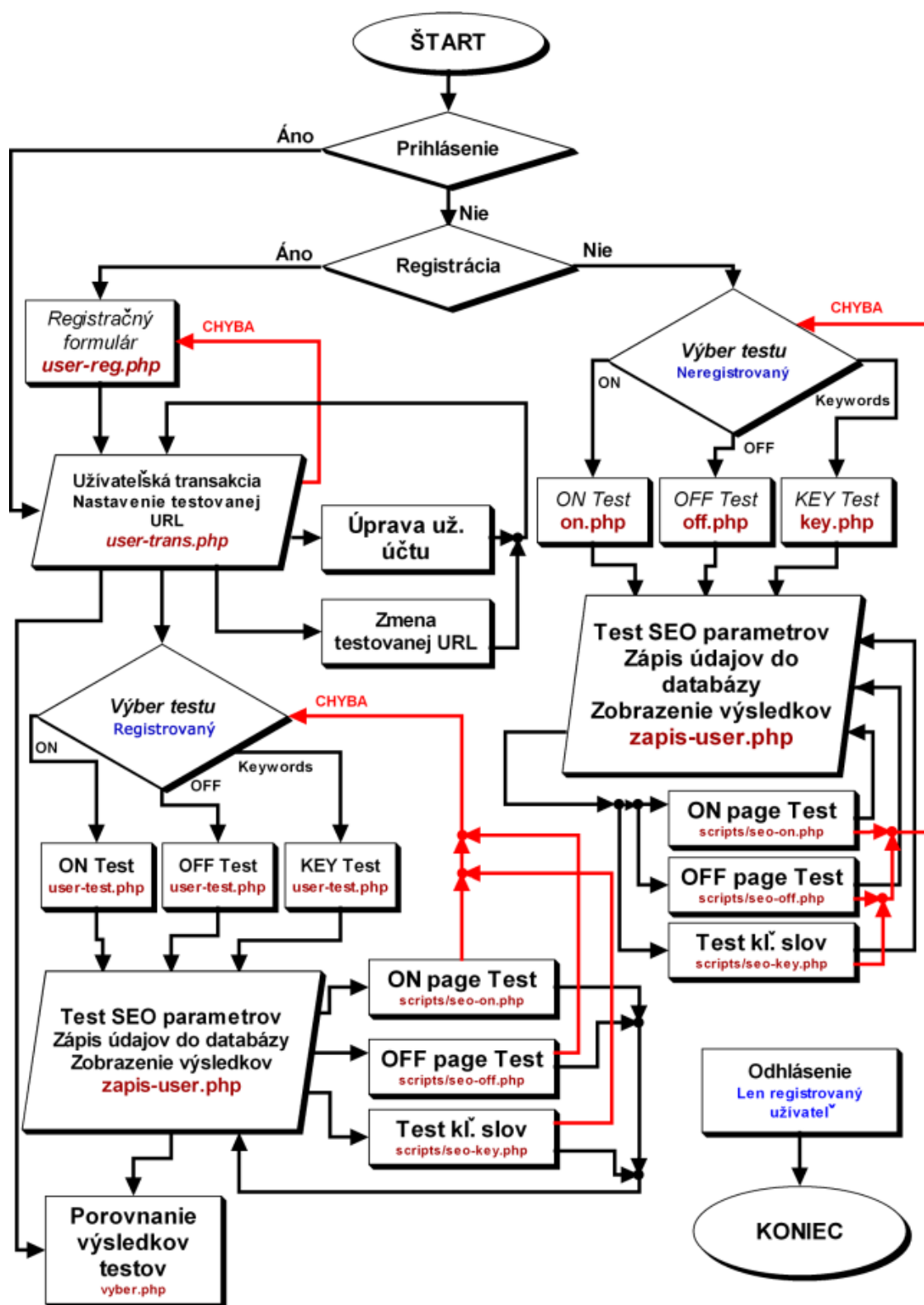
```
DELETE FROM seo_on WHERE url LIKE '%"$_SESSION['url_user']'.  
"% ' AND on_id='"$_SESSION['id']"."' AND time2='"$_GET['time']"."'
```

Týmto spôsobom je odstránený špecifický riadok tabuľky. Následne je užívateľ presmerovaný na stránku zobrazujúcu všetky výsledky testov. Rovnakým spôsobom sú odstraňované aj ostatné výsledky či už v prípade off page SEO testov alebo v prípade testov kľúčových slov.

4.8 Vývojový diagram portálu na testovanie SEO faktorov

Vývojový diagram zobrazuje možnosti užívateľa pri vstupe na stránku. Zobrazuje všetky možnosti testovania, registrácie, prihlásenia, úprav účtu a zobrazovania výsledkov jednotlivých testov. **Vývojový diagram nemá za úlohu presne popísať funkciu portálu ako programu, zobrazuje len interakciu jednotlivých súborov tvoriacich samotný portál.** Popisovanie portálu ako programu je pre objasnenie jeho funkcie zbytočné a bolo by veľmi neprehľadné. Z fungovania portálu je jasné, že užívateľ sa môže do bodu *výber testu neregistrovaného užívateľa* dostať prakticky z ľubovoľného miesta stránky. Z dôvodu väčšej prehľadnosti to vo vývojovom diagrame nie je zobrazené. Do bodu *výber testu registrovaného užívateľa* sa však užívateľ môže dostať len po prihlásení. Vývojový diagram taktiež z dôvodu väčšej prehľadnosti neobsahuje položky:

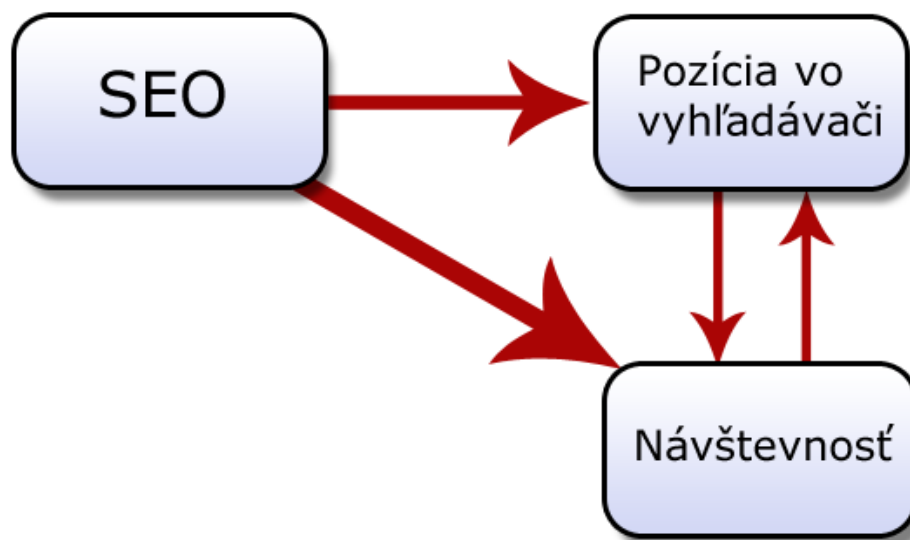
- Návody
- Pripomenutie hesla
- A iné menej podstatné časti portálu



Obr. 4.18: Vývojový diagram SEO portálu

5 ANALÝZA VPLYVU OPTIMALIZÁCIE SEO NA NÁVŠTEVNOSŤ STRÁNOK A ICH UMIES- TENIE VO VYHĽADÁVAČOCH

Popis vplyvu SEO na návštevnosť stránok a ich pozíciu vo vyhľadávači najlepšie popisuje nasledujúci obrázok.



Obr. 5.1: Grafické zobrazenie vplyvu SEO na návštevnosť stránky a jej pozíciu na SERP

Dôvod prečo optimalizácia pre vyhľadávače priamo vplýva na pozíciu stránky vo vyhľadávači je jasný. Robot vyhľadávača má tak povediac radšej stránku, ktorá je prispôbena tomu aby ju mohol jednoducho zaradiť do svojej databázy. Napríklad ak užívateľ optimalizoval veľkosť stránky na hodnoty, ktoré SEO doporučuje je veľká pravdepodobnosť, že sa stránka posunie na vyššie pozície vo výsledkoch vyhľadávania. Napriek tomu čas, ktorý uplnie od optimalizácie stránky po jej posun na *SERP* - *search engine result page* môže byť úplne náhodný. Závisí vždy od mechanizmu vyhľadávača. Môže to byť otázka hodín ale aj týždňov. Metodika SEO sa neustále vyvíja a prispôsobuje sa zmenám v algoritmoch vyhľadávačov.

Vplyv zmeny pozície stránky na *SERP* - *search engine result page* na návštevnosť danej stránky je očividný. Internetové vyhľadávače v dnešnej dobe používajú milióny ľudí, ktorý si medzi výsledkami vyhľadávania vždy vyberú stránky, ktoré sú na prvých pozíciách *SERP* - *search engine result page*, pretože plne dôverujú ich vyhľadávaciemu mechanizmu. Z vlastnej skúsenosti viem, že nárast počtu prístupov na optimalizované stránky môže byť až 200 percentný.

SEO môže vplývať na návštevnosť stránok aj priamo. Správne optimalizovaná stránka má veľký predpoklad zaujať väčšie množstvo užívateľov, ktorý si o nej môžu predať informácie aj priamo medzi sebou. Takýmto spôsobom môže návštevnosť stránky ovplyvniť aj jej pozíciu na SERP. Užívatelia zverejnia spätné odkazy na vašu stránku a vyhľadávač vám môže zvýšiť tzv. Page Rank. Tento prípad ale nastáva len vo výnimočných situáciách.

6 ZÁVER

V dnešnej dobe sú webové prezentácie firiem, organizácií a jednotlivcov bežnou a veľmi potrebnou záležitosťou. Až 90 percent ľudí hľadá informácie na internete pomocou internetových vyhľadávačov. Z dlhodobého hľadiska je SEO výhodnejší a najmä finančne lacnejší spôsob ako dostať webovú stránku na prvé pozície vo výsledkoch vyhľadávania. SEO je veľmi podstatná pre trvalé udržanie web stránok medzi prvými v týchto výsledkoch. Dá sa rozdeliť do troch základných častí - tzv. On, Off page SEO faktory a ostatné faktory ovplyvňujúce SEO. Portál testuje všetky tri základné časti a poskytne optimálne riešenie prípadných problémov.

Základom pre návrh portálu bola požadovaná jednoduchosť. Vyplýva z predpokladu, že prípadní užívatelia portálu nemusia byť profesionálni tvorcovia webstránok. Bola vyvinutá maximálna snaha o to, aby boli minimalizované prípadné chyby pri testovaní. Navrhnutý portál je taktiež plne validný a z veľkej časti spĺňa aj požiadavky SEO a prístupnosti na webe. Veľkou výhodou portálu je, že užívateľ môže pri registrácii uviesť až tri webstránky, ktoré chce testovať. Heslo je do databázy ukladané v šifrovanej podobe pomocou kryptografickej hašovacej funkcie md5. V prípade rozširovania portálu je ale potrebné zmeniť túto šifrovaciu metódu z dôvodu jej slabej bezpečnosti.

Základné problémy pri realizácii a programovaní portálu spočívali v neustále sa meniacich algoritmoch a výsledkoch vyhľadávania jednotlivých vyhľadávačov. Napríklad google prednedávnom kompletne zmenil vzhľad *SERP* - *search engine result page*, preto bolo potrebné upraviť viacero testovacích skriptov. Je jasné, že pre správnu funkčnosť portálu je žiadúce neustále sledovať najnovšie trendy problematiky SEO a následne im prispôbovať testovacie skripty.

Portál poskytuje veľa možností budúceho rozširovania. Napríklad vytvorenie tzv. administrátorského účtu, ktorý by mal za úlohu správu jednotlivých účtov. Taktiež je možné doplniť ďalšie testovateľné SEO faktory.

Optimalizácia pre vyhľadávače nemôže garantovať prvé miesto navždy. Pozíciu vo výsledkoch vyhľadávania určujú aj faktory, ktoré sa nedajú ovplyvniť. Predovšetkým je to zmena algoritmov usporiadania výsledkov vo vyhľadávačoch. Z výskumov však vyplýva, že nezáleží len na tom, či je web stránka na prvom alebo štvrtom mieste vo výsledkoch vyhľadávania, ale hlavne na kvalite jej spracovania.

Spoločnosť Trinity Expert Systems Limited uskutočnila kampaň, ktorej cieľom bolo porovnávanie návratnosti investícií SEO a PPC. Kľúčovým faktorom bolo vybratie najvhodnej služby pre propagáciu. Konečné výsledky neboli vôbec prekvapujúce. Ako účinnejšie sa ukázalo SEO s 352 percentnou návratnosťou investícií, zatiaľ čo PPC dosiahlo návratnosť iba 216.73 percent. Celkový záver je taký, že pre budúcu marketingovú stratégiu tejto spoločnosti by bola vhodná kombinácia oboch.

LITERATÚRA

- [1] Kubíček, M. *Velký průvodce SEO*. 2008, Zodpovedný redaktor Martin Domes. Prvé vydanie. Computer Press, 2008. 318s. ISBN 978-80-251-2195-5
- [2] Kolektiv autorů. *PHP5, MySQL, Apache - Vytváříme webové aplikace*. Datum vydání: květen 2006, Nakladatel: Computer Press 816 s. ISBN: 80-251-1073-7
- [3] Sirovich, Jamie. Darie, Cristian. *SEO v PHP - Programujeme profesionálně*. Datum vydání: květen 2008, Nakladatel: Computer Press 384 s. ISBN: 978-80-251-2083-5
- [4] Jeff Croft, Ian Lloyd, Dan Rubin. *Mistrovství v CSS - Pokročilé techniky pro webové designéry a vývojáře*. Datum vydání: červen 2007, Nakladatel: Computer Press 416 s. ISBN: 978-80-251-1705-7
- [5] SMIČKA, Radim. *Optimalizace pro vyhledávače - SEO*. Dubany : Jaroslava Smičková, 2004. 120 s. ISBN 80-239-2961-5
- [6] Online php manual *Online php manual* [online]. Dostupné z URL: <<http://www.php.net/manual/en/index.php>>.
- [7] Position Absolute *A jQuery inline form validation, because validation is a mess* [online]. Dostupné z URL: <<http://www.position-absolute.com/articles/jquery-form-validator-because-form-validation-is-a-mess/>>.
- [8] 1234.info *Webtemplates - Multiflex-3* [online]. Dostupné z URL: <<http://www.1234.info/webtemplates/multiflex3/>>.
- [9] Princíp fungovania vyhľadávačov *Ako fungujú vyhľadávače?* [online]. Dostupné z URL: <[Ako fungujú vyhľadávače?](#)>.
- [10] SEO chat - optimalizácia pre Slovensko *SEO chat - optimalizácia pre Slovensko* [online]. Dostupné z URL: <<http://www.seochat.sk/>>.
- [11] Lacko, Luboslav. *Optimalizace pro vyhledávače - SEO*. Datum vydání: říjen 2006, Nakladatel: Computer Press 304 s. ISBN: 80-251-1249-7
- [12] SEO-OPTIMALIZÁCIA PRE VYHLÁDÁVAČE *robots.txt* [online]. Dostupné z URL: <<http://www.seoprojekt.szm.com/robots.html>>.
- [13] Moskwa, Susan. Foucher, Trevor. *Sitemaps FAQs* [online]. Dostupné z URL: <<http://googlewebmastercentral.blogspot.com/2008/01/sitemaps-faqs.html>>.

- [14] Brin, Sergey. Page, Lawrence. *The Anatomy of a Large-Scale Hypertextual Web Search Engine* [online]. Dostupné z URL: <<http://infolab.stanford.edu/pub/papers/google.pdf>>.
- [15] Wikipédia. *Regulárny výraz* [online]. Dostupné z URL: <http://sk.wikipedia.org/wiki/Regulárny_výraz>.
- [16] FEM SPU v Nitre. *CURL, Client URL Library Functions* [online]. Dostupné z URL: <<http://www.fem.uniag.sk/manualy/php/mansk/ref.curl.html>>.
- [17] Webfaq - Autor: Roman Matěna. *Jak zjistiť S-Rank pomocí XML-RPC* [online]. Dostupné z URL: <<http://www.webfaq.cz/clanek/Jak-zjistit-S-Rank-pomoci-XML-RPC>>.
- [18] Blog o reklame - meu.sk. *Alexa Rank - čo to vlastne je?* [online]. Dostupné z URL: <<http://reklamnyporadca.meu.zoznam.sk/news/alexa-rank-co-to-vlastne-je/>>.
- [19] Stackoverflow. *What is Keyword Density and how to create a script in PHP??* [online]. Dostupné z URL: <<http://stackoverflow.com/questions/819166/what-is-keyword-density-and-how-to-create-a-script-in-php>>.
- [20] Bing.com. *Bing API Version 2.0* [online]. Dostupné z URL: <<http://www.bing.com/developers/s/API%20Basics.pdf>>.

ZOZNAM SYMBOLOV, VELIČÍN A SKRATIEK

SEM Marketing vo vyhľadávačoch – Search engine marketing

SEO Optimalizácia pre vyhľadávače – Search engine optimization

SERP Stránka výsledkov vyhľadávania – Search engine result page

PPC Platba za kliknutie – pay per clicke

W3C The World Wide Web Consortium

WWW World wide web

WCAG Web Content Accessibility Guidelines

CSS Cascading style sheet

HTML Hypertext markup language

XHTML Extended hypertext markup language

PHP Hypertext preprocessor

API Application programming interface

XML eXtensible Markup Language

SQL Structured Query Language