

Webová aplikace využívající funkce pro zpracování obrázků v databázi Oracle

A Web Application Using Functions for Image Processing in the
Oracle Database

Bc. Michal Matáš



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
akademický rok: 2011/2012

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: Bc. Michal MATIÁŠ
Osobní číslo: A10468
Studijní program: N 3902 Inženýrská informatika
Studijní obor: Počítačové a komunikační systémy

Téma práce: Webová aplikace využívající funkce pro zpracování obrázků v databázi Oracle

Zásady pro vypracování:

1. Nastudujte možnosti zpracování obrázků v multimediální databázi Oracle.
2. Nastudujte možnosti využití databáze Oracle ve webové aplikaci a vypracujte literární rešerši.
3. Vyberte vhodné řešení pro aplikaci, která bude využívat funkce pro zpracování obrázků a svůj výběr zdůvodněte.
4. Naprogramujte webovou aplikaci, která bude využívat zpracování obrázků v databázi Oracle.
5. Aplikaci zprovozněte na školním serveru.

Rozsah diplomové práce:

Rozsah příloh:

Forma zpracování diplomové práce: tištěná/elektronická

Seznam odborné literatury:

1. LACKO, Luboslav. ORACLE: Správa, programování a použití databázového systému. Vyd. 1. Brno: Computer Press, 2003, 464 s. ISBN 80-722-6699-3.
2. LECKY-THOMPSON, Ed a Steven D NOWICKI. PHP 6: programujeme profesionálně. Vyd. 1. Překlad Ondřej Gibl. Brno: Computer Press, 2010, 718 s. Programujeme profesionálně. ISBN 978-802-5131-275.
3. KOSEK, Jiří. PHP – tvorba interaktivních internetových aplikací: podrobný průvodce. Vyd. 1. Praha: Grada, 1999, 490 s. Průvodce (Grada). ISBN 80-716-9373-1.
4. ORACLE. Oracle? Multimedia Reference. 11g Release 1 (11.1). Redwood Shores, 2007. Dostupné z: http://docs.oracle.com/cd/B28359_01/appdev.111/b28414/toc.htm
5. URMAN, Scott, Ron HARDMAN a Michael MCLAUGHLIN. Oracle: programování v PL/SQL. Vyd. 1. Překlad Jiří Fadrný. Brno: Computer Press, 2007, 720 s. ISBN 978-802-5118-702.

Vedoucí diplomové práce:

Ing. Kateřina Ježková

Datum zadání diplomové práce:

24. února 2012

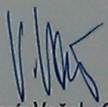
Termín odevzdání diplomové práce:

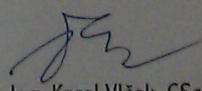
28. května 2012

Ve Zlíně dne 24. února 2012



L.S.


prof. Ing. Vladimír Vašek, CSc.
děkan


prof. Ing. Karel Vlček, CSc.
ředitel ústavu

ABSTRAKT

Cieľom tejto diplomovej práce bolo vytvoriť webovú aplikáciu na spracovávanie obrázkov pomocou databázy Oracle a tak demonštrovať jej pokročilé funkcie a prácu s multimediálnymi dátami. Navrhnutá aplikácia je vytvorená v skriptovacom jazyku PHP a obrázky sú spracovávané pomocou PL/SQL skriptov. Informácie sú získavané z obrazových metadát uložených priamo v databáze.

Kľúčová slova:

Oracle, PHP, PL/SQL, databáza, web, obrázok

ABSTRACT

The aim of this thesis was to create a Web application for image processing using the Oracle database and thus demonstrate its advanced features and multimedia data. The proposed application is developed in PHP scripting language and the images are processed using the PL / SQL scripts. Information is extracted from image metadata stored in the database directly.

Keywords:

Oracle, PHP, PL/SQL, database, web, image

Týmto by som rád poďakoval Ing. Kateřině Ježkové za cenné rady, pripomienky a všetok čas, ktorý mi venoval pri odbornom vedení tejto práce.

Prohlašuji, že

- beru na vědomí, že odevzdáním diplomové/bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že diplomová/bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové/bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji diplomovou/bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – diplomovou/bakalářskou práci nebo poskytnout licenci k jejímu využití jen s předchozím písemným souhlasem Univerzity Tomáše Bati ve Zlíně, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše);
- beru na vědomí, že pokud bylo k vypracování diplomové/bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky diplomové/bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem diplomové/bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na diplomové práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně

.....
podpis diplomanta

OBSAH

ÚVOD	9
I TEORETICKÁ ČÁST	10
1 ORACLE	11
1.1 VÝVOJ DATABÁZE	11
1.2 OBJEKTOVO RELAČNÁ TECHNOLOGIA.....	11
1.3 ORACLE INTERMEDIA	12
1.4 ARCHITEKTÚRA INTERMEDIA	12
1.5 MOŽNOSTI INTERMEDIA	14
2 SPRACOVANIE OBRAZOVÝCH DÁT.....	17
2.1 DIGITÁLNE OBRÁZKY	17
2.2 OBRÁZKOVÉ KOMPONENTY	17
2.3 METADÁTA V OBRÁZKOCH	18
2.4 EXTRAHOVANIE METADÁT.....	18
3 VYHĽADÁVANIE PODĽA OBSAHU	19
3.1 PREHĽAD A PRÍNOSY VYHĽADÁVANIA NA ZÁKLADE OBSAHU	19
3.2 PRINCÍP OBSAHOVÉHO POROVNÁVANIA	20
3.2.1 Farba	22
3.2.2 Textúra.....	24
3.2.3 Tvar.....	25
3.3 AKO VYHĽADÁVANIE FUNGUJE.....	25
3.3.1 Váha.....	26
3.3.2 Score	26
3.3.3 Využitie.....	27
II PRAKTICKÁ ČÁST.....	29
4 NÁVRH APLIKÁCIE.....	30
4.1 VÝVOJOVÉ PROSTRIEDKY	30
4.1.1 PHP	30
4.1.2 Oracle Call Interface	30
4.1.3 SQL a PL/SQL	31
4.1.4 HTML a CSS.....	32
4.1.5 Oracle Database.....	32
4.1.6 Apache	32
5 PRÍPRAVA DATABÁZE	34

5.1	VYTVORENIE SCHÉMY	34
5.2	VYTVORENIE TABULIEK.....	35
5.3	VYTVORENIE ZLOŽKY	36
5.4	PRIPOJENIE NA DATABÁZU.....	36
6	UŽÍVATEĽSKÁ APLIKÁCIA.....	37
6.1	REGISTRÁCIA	37
6.2	LOGIN.....	38
6.3	HOME STRÁNKA	39
6.4	UPLOAD.....	39
6.4.1	Naplnenie tabuľky.....	40
6.4.2	Nahratie obrázka do databázy	41
6.4.3	Generovanie obrázkového podpisu.....	42
6.4.4	Výpis vlastností.....	43
6.5	LIBRARY	43
6.6	EDIT	45
6.6.1	Rotate.....	45
6.6.2	Black/White	46
6.6.3	Scale.....	46
6.6.4	Gamma	47
6.6.5	Cut	47
6.6.6	Similar	47
6.6.7	Search.....	48
6.6.8	Ďalšie funkcie	50
	ZÁVĚR	52
	ZÁVĚR V ANGLIČTINĚ.....	53
	SEZNAM POUŽITÉ LITERATURY.....	54
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK.....	56
	SEZNAM OBRÁZKŮ.....	57
	SEZNAM TABULEK	58
	SEZNAM PŘÍLOH	59

ÚVOD

Len s ťažkosťou by sme v dnešnej dobe hľadali aplikáciu bez využitia databázy. Súčasný informatický vek stojí a padá na ukladaní, uchovávaní a vyhľadávaní v rôznych typoch databázových systémov. Databázy pôvodne slúžili na uchovávanie jednoduchých záznamov. Neskôr sa však ich pôsobnosť rozšírila natoľko, že v dnešnej dobe dokážu vyhľadávať v obrázkoch, videách a iných multimedialných súboroch.

Práca sa skladá z dvoch častí, prvá je teoretická a druhá praktická. V prvej kapitole teoretickej časti je popísaná štruktúra databázy a základný prehľad ako sa databáza Oracle vyvíjala. Ďalej je priestor venovaný metadátam – dáta, ktoré popisujú obrázky. Posledná časť je zameraná na vyhľadávanie na základe obsahu obrázku a s tým súvisiaci obrázkový podpis, čo je v dnešnej dobe veľmi populárna funkcia.

V úvode praktickej časti je popísaná komunikácia medzi PHP, databázou Oracle pomocou Oracle Call Interface a základné nastavenia, ktoré sú potrebné na funkčnosť skriptov. Posledná kapitola vysvetľuje jednotlivé funkcie webovej aplikácie a funkčnosť skriptov na upravovanie obrázkov.

I. TEORETICKÁ ČÁST

1 ORACLE

Oracle je jedna z najväčších spoločností, ktoré vyvíjajú relačné databázy, nástroje pre vývoj a správu databáz či customer relationship management (CRM) systémy. Zakladateľmi boli inžinieri zo Silicon Valley Larry Ellison, Bob Miner a Ed Oates.

1.1 Vývoj databáze

Prvá verzia databáze vyšla v roku 1979. Ďalšia verzia podporuje vykonávanie SQL dotazov a využíva ad-hoc prístup.

Piata verzia databázy jednou z prvých RDBMS databáz, ktoré fungujú v režime Client – Server. Veľkou výhodou je, že aplikácia beží na rôznych stolových počítačoch (klientoch) zároveň a umožňuje tak spoločný prístup k centrálnej databáze (server). Neskôr je pridaný „Hot Back – up“ – záloha databázy aj v stave online a používania.

V roku 1997 vzniká verzia 8, ktorá podporuje rozsiahle spravovanie a uchovávanie multimedialných dát. Oracle8i je prvá databáza na podporu webových technológií, ako je Java a Hypertext Transfer Protocol (HTTP) [11].

Najnovšou verziou databázy Oracle je verzia Oracle Database 11g Release 2.

1.2 Objektovo relačná technológia

Oracle Databáza je relačný databázový systém. Znamená to, že okrem svojej tradičnej úlohy (bezpečné a účinné riadenie relačných dát) poskytuje podporu pre definíciu objektov, vrátane údajov súvisiacich s objektami a metódy, ktoré je možno na nich vykonať. Objekt relačnej technológie zahŕňa aj podporu objektov BLOB, ktoré poskytujú základ pre prácu so zložitými objektami ako sú digitalizované audio, video a obraz.

1.3 Oracle InterMedia

Oracle intermedia je funkcia, ktorá umožňuje Oracle Database ukladať, spravovať a načítat' obrázky, audio, video alebo iné údaje v prostrediach integrovaným spôsobom s ďalšími informáciami. InterMedia Oracle Database rozširuje spoľahlivosť, dostupnosť a správu dát pre multimediálny obsah na internete, elektronickom obchode, a pre dátovo náročné aplikácie. Táto funkcia je ponechané na aplikačný softvér.

InterMedia poskytuje nasledovné funkcie:

- Ukladanie a vyhľadávanie
- Aplikácie metadát
- Podpora populárnych formátov (JPEG, PNG, MP3...)
- Prístup cez webové rozhranie
- Dotazovanie pomocou pripojených relačných dát
- Dotazovanie pomocou multimediálneho obsahu s možnosťou indexácie

Databáza vie s dátami pracovať, porovnávať ich avšak ich nedokáže zobrazit'. Narába s nimi pomocou jazyka PL/SQL (Procedural Language / Structured Query Language). Pomocou tohto jazyka sa dajú vytvárať rôzne procedúry, funkcie, programové balíky, triggerly alebo užívateľsky definovaný dátový typ [5].

V interMedia majú audio údaje vlastnosti objektu známeho ako ORDAudio, obrazové dáta ORDImage a video dáta ORDVideo. Heterogénne dáta prezentuje objekt typu ORDDoc. Všetky štyri typy využívajú ako zdroj informácií objekt relačného typu pod názvom ORDSrcSource.

1.4 Architektúra interMedia

InterMedia je jednotný a integrovaný doplnok, ktorý rozširuje databázu, ukladanie, správu a načítanie obrazových, audio a video dát a podporu webových aplikácií pre spracovanie multimediálnych dát. Architektúra definuje rámec, v ktorom je podporovaný multimediálny obsah, rovnako ako tradičné údaje. Tento obsah a údaje môžu byť ďalej bezpečne zdieľané medzi ďalšie viaceré aplikácie napísané v obľúbených programovacích jazykoch. Dajú sa jednoducho spravovať [4].

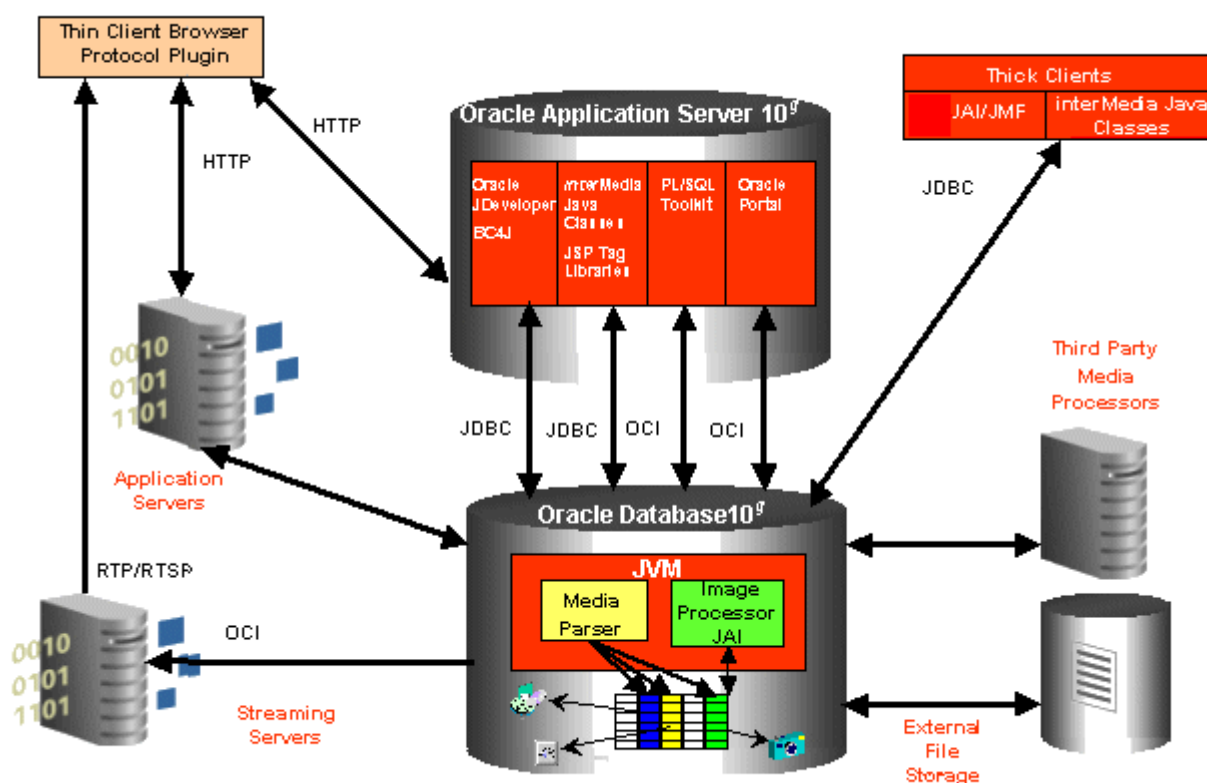
Obrázok 1 ukazuje architektúru interMedia z trojvrstvovej perspektívy:

- Databázová vrstva – Oracle Database
- Aplikačný server – Oracle Application Server
- Client vrstva – tenký a tlstý klient

V prvej vrstve, s pomocou interMedia, má databáza Oracle bohatú podporu typov dát, vrátane tradičných dát. Prostredníctvom vstavanej JVM je na strane servera podporovaný parser, rovnako ako obrazový procesor. Parser médií má objektovo orientované rozhranie, podporuje aplikáciu a parsovanie metadát a môže byť rozšírený o podporu ďalších formátov.

Obrazový procesor obsahuje JAI a zabezpečuje spracovanie obrazu pre operácie ako je vytváranie náhľadu, zmena formátu, indexovanie obrazov a vyhľadávanie. Pomocou interMedia metód je možný export a import súborov medzi databázou a operačným systémom (externé úložisko súborov). Taktiež podporuje špeciálne typy súborov zo servera, ako je napríklad streamovaný obsah z databázy. Pomocou plug-inov pre interMedia môže prenášať multimediálne dáta priamo z databázy pomocou RTSP (Real-Time Streaming Protocol) [4]. Okrem toho, procesory tretích strán ako napríklad rozpoznávanie reči, môžu vstupovať do databázy, spracovať médiá v nej uložené a vrátiť výsledky naspäť do databázy.

V druhej (strednej) vrstve, poskytuje Oracle Application Server prístup j interMedia prostredníctvom interMedia Java Classes, čo umožňuje prístup, manipuláciu a úpravu zvuku, obrazu a video dát pre všetky java aplikácie na všetkých vrstvách. Okrem toho Oracle intermedia triedy umožňuje odosielanie a načítanie multimediálnych dát uložených v databáze pomocou intermedia OrdAudio a OrdDoc a OrdImage a OrdVideo typy objektov pre servlety a JSP. Oracle Java interMedia triedy pre servlety a JSP majú priamy prístup k dátam uloženým v objektoch, BLOB alebo BFILE [4].



Obrázok 1 Architektúra Oracle interMedia [4]

Oracle interMedia je k dispozícii len na Oracle Technology Networks (OTN), ktorý zahŕňa nasledovné časti:

- interMedia plug-in pre servery RealNetworks Streaming
- interMedia Plug-in pre Microsoft Windows Media Services
- interMedia Custom Data Source a DataSink triedy JMF 2.0/2.1 (vyžaduje JMF 2.0 alebo vyšší)

1.5 Možnosti interMedia

Možnosti využitia Oracle interMedia sú naozaj široké. K hlavným schopnostiam patria ukladanie, vyhľadávanie, správa a manipulácia s dátami riadených databázou Oracle.

Multimediálne aplikácie majú spoločné a jedinečné požiadavky. Typy objektov interMedia majú spoločné požiadavky a môžu byť rozšírené o špecifické požiadavky podľa typu aplikácie. Multimediálne dáta sa dajú pomocou interMedia riadiť rovnako ľahko, ako štandardné atribúty dát.

InterMedia je prístupná rovnako relačným, tak aj objektovým rozhraním. Databázové aplikácie napísané v jazykoch Java, C++, alebo tradičnými programovacími jazykmi tretej generácie môžu komunikovať s interMedia pomocou modernej knižnice, PL/SQL, alebo Oracle Call Interface(OCI).

InterMedia podporuje ukladanie z populárnych formátov vrátane obrázkov, streamované audio a video formáty. Taktiež poskytuje prostriedky pre pridanie zvuku, obrazu a videa, alebo iných stĺpcov, objektov do existujúcich tabuliek. Táto databáza umožňuje rozšíriť existujúce databázy, alebo vybudovanie novej pre užívateľov pracujúcich s multimediálnymi dátami. Pomocou základných funkcií je možné vytvoriť špecializované multimediálne aplikácie.

InterMedia používa typy objektov podobne ako Java, alebo C++, pomocou ktorých popisuje multimediálne dáta. Tieto typy objektov sa nazývajú ORDAudio, ORDDoc, ORDImage a ORDVideo. Tieto typy objektov sa skladajú z atribútov, vrátane metadát a dátových médií a metód. Metadáta sú informácie o dátach, ako je dĺžka objektu, typ kompresie, alebo formátu. Metódy sú procedúry bez parametrov, ktoré môžu byť vykonané na objekt, napríklad getContent() a SetProperties() [1].

Metadáta sú uložené v databáze pod kontrolou interMedia. Či sú uložené priamo v databáze, alebo mimo nej interMedia spravuje metadáta pre všetky typy médií a môže automaticky extrahovať zo zvukového, obrazového, alebo video formátu. Tieto metadáta zahŕňajú:

- ukladanie informácií o zvuku, obraze, videu a iných heterogénnych prostrediach vrátane zdrojov typu: umiestnenie a názov zdroja a či sú dáta uložené v databáze, alebo mimo nej
- aktualizáciu časovej pečiatky pre multimédiá
- popis audio a video dát
- formát multimédií
- MIME typ
- Audio vlastnosti: typ kódovania, počet kanálov, vzorkovacia frekvencia, veľkosť súboru, typ kompresie a trvanie

- Obrazové vlastnosti: výška, šířka, délka, obsah obrazu, formát kompresie
- Video parametre: výška šířka rámu, rám, rozlíšenie, počet snímkov za sekundu, trvanie, počet farieb, prenosová rýchlosť

Okrem extrakcie metadát poskytuje súbor metód pracujúcich s grafikou. Pre obrazy sú k dispozícii metódy, ktoré vykonávajú: konverziu formátu, kompresiu, zmenšenie, zväčšenie, orezanie, kopírovanie, otáčanie, zrkadlenie a nastavenie jasu snímok.

Podporuje základnú sadu populárnych audio, obrazových a video formátov pre spracovanie multimediálnych dát, ktorá môže byť rozšírená. Napríklad pre podporu ďalších formátov, novú digitálnu kompresiu a dekompresiu (kodeky), zdroje dát a dokonca aj špecializované spracovanie dát – algoritmy pre spracovanie audia a videa. InterMedia je základným kameňom pre rôzne multimediálne aplikácie [4]. Niektoré príklady, z mnohých sú:

- Úložisko pre kontrolu digitálnych snímok
- Elektronické záznamy, vrátane snímok DICOM
- Call centrá
- Fyzická inventúra majetku
- On-line vzdelávania
- Realitný trh
- Archívy reklamných fotografií
- Zobrazovanie dokumentov
- Finančne spravodajstvo
- Audio a video webové obchody

2 SPRACOVANIE OBRAZOVÝCH DÁT

Oracle interMedia sa vo veľkej miere využíva na spracovávanie obrazových dát. Na tieto typy dát, má databáza Oracle k dispozícii ORDIImage objekt a špecializované ORDIImage objekty.

2.1 Digitálne obrázky

ORDIImage dokáže ukladať, hľadať a spravovať digitalizované obrázky v databáze. Podporuje dvojrozmerné, statické, digitalizované obrázky reálnych objektov a scén uložených v binárnej podobe. Obrázky môžu byť zhotovené skenerom, video zariadením, digitálnou kamerou, alebo video rekordérom napojeným na digitizér [4].

2.2 Obrázkové komponenty

Digitalizovaný obraz sa skladá z obrazových dát (digitalizované bity) a atribútov, ktoré popisujú a charakterizujú obrazové dáta. Aplikácie, ktoré spracúvajú obrazové dáta potrebujú niekedy špecifické informácie, ako je napríklad meno osoby zobrazenej na fotografií, popis obrázku, meno fotografa atď. Tieto informácie sa môžu uložiť ako popis v atribute, alebo priamo stĺpec v databázovej tabuľke.

Obrazové dáta (pixels) môžu mať rôznu hĺbku (bit/pixel) v závislosti na tom, ako bola fotka, obrázok zachytená. Preto môžu byť bity organizované rôznymi spôsobmi. Táto organizácia sa nazýva ako dátový formát. ORDIImage je schopný načítavať a ukladať obrazové dáta z akéhokoľvek dátového formátu.

ORDIImage dokáže spracovať a automaticky získať vlastnosti rôznych populárnych formátov. Avšak niektoré dátové typy nemajú takú širokú podporu pre spracovanie obrazu.

Úložný priestor pre digitalizované dáta môže byť veľký v porovnaní s atribútmi ako sú čísla, alebo text. Preto kompresné schémy umožňujú stlačiť obraz na menej bytov, čím sa znižuje pamäťové a sieťové zaťaženie. Bezstrátové kompresie dokážu veľkosť obrazu zmenšiť, ale po opätovnom rozbalení je výsledný obraz úplne rovnaký. Pri strátových kompresiách dochádza k zníženiu kvality obrazu, ale aj k podstatnej úspore miesta.

2.3 Metadáta v obrázkoch

Oracle 10g R2 dodáva pre interMedia funckiu Image metadata. Táto možnosť zlepšuje prácu s objektami typu ORDImage tým, že sa daná vlastnosť dá čítať, prepísať, vložiť, alebo odstrániť. Navyše sú tieto vlastnosti oddelené od obrazového súboru. Metadáta môžu byť uložené v databáze, indexované, vyhľadané a sú k dispozícii aplikácia, ktoré s nimi potrebujú pracovať, pomocou štandardných mechanizmov.

Ďalším veľkým využitím metadát je vo formáte DICOM (Digital Imaging and Communications in Medicine). Objekt ORDImage berie formát DICOM ako samostatný typ a dokáže prečítať podmnožinu údajov, ktoré sa prikladajú k pacientovi [4].

2.4 Extrahovanie metadát

InterMedia umožňuje extrahovať obsah a formát metadát z mediálnych zdrojov (audio a video súbory). Obsah metadát ukladá ako XML vo formáte CLOB. Akonáhle sa metadáty vytiahnu a uložia, môže sa vytvoriť index pre rýchle a efektívne vyhľadávanie pomocou týchto indexov.

3 VYHLÁDÁVANIE PODĽA OBSAHU

Následujúca časť vysvetľuje aké sú výhody obsahového vyhľadávania a na akom princípe funguje. Zahŕňa nasledujúce časti:

- Prehľad a prínosy vyhľadávania na základe obsahu
- Ako vyhľadávanie funguje, vrátane definícií a vysvetlení jednotlivých vizuálnych atribútov (farba, textúra, tvar, umiestnenie) a dôvod ich použitia v jednotlivých prípadoch
- Použitie vyhľadávania, vrátane porovnania, význam váh vizuálnych atribútov a určenie miery podobnosti medzi obrazmi
- Použitie indexovania k zlepšeniu vyhľadávania a zvýšeniu výkonu
- Príprava obrázka v maximálnemu využitíu

3.1 Prehľad a prínosy vyhľadávania na základe obsahu

Nenáročná technológia zachytávania a ukladania obrazu umožňuje vytvárať veľké kolekcie digitálnych snímok. Uchovávanie snímok v databáze prácu s nimi značne zjednodušuje a uľahčuje. Avšak čím rastie počet snímok rastie tým je aj ťažšie nájsť daný obrázok. Všeobecne sa k tomuto problému vyvinuli dva základné prístupy. Oba využívajú k identifikácii metadáta.

- Využívanie informácií, ktoré je možné ručne zadať, alebo sú zahrnuté v návrhu tabuľky ako sú názvy, kľúčové slová, identifikačné čísla atď.
- Pomocou automatickej extrakcie obrazu, príznakov a detekcie objektov priradiť obrazu obsah – to je hlavný atribut pre vyhľadávanie na základe obsahu

S Oracle interMedia sa dajú kombinovať oba prístupy pri navrhovaní tabuľky. Dajú sa používať štandardné textové stĺpce a nimi popísať semantický význam obrazu – napríklad, že automobil na obrázku má červenú farbu, alebo je jeho rok výroby 2007. Druhou možnosťou je opísať obrázok pomocou jeho podpisu. Táto možnosť je umožnená v Oracle pomocou metódy `ORDImageSignature` [1]. Obrázok sa dá identifikovať pomocou dotazov na základe obsahových vlastností obrazu – napríklad ako jeho farba a tvar zodpovedá konkrétnemu automobilu. Alternatívny spôsob kombinuje tieto dva spôsoby, kedy môže

návrhár do tabuľky zakomponovať špecializovaný dátový typ ORDImage a k nemu stĺpce s popisom, identifikačným číslom, dátumom a inými atribútmi.

Hlavným prínosom využitia vyhľadávania podľa obsahu je úspora času a úsilia na získanie potrebných informácií k danému obrazu. Vďaka častému updatu a pridávaniu snímok vo veľkých databázach nie je veľmi praktické manuálne zadávanie všetkých atributov, ktoré môžu byť potrebné pre všetky vykonávané dotazy a služby. Preto majú atributy obsahu väčšiu flexibilitu a praktické využitie. Poskytujú dotazovanie na atributy typu štruktúra, typ, alebo obrys, teda vlastnosti, ktoré by sa ťažko dali reprezentovať kľúčovými slovami [4].

Príklady databázových aplikácií, kde je vhodné využiť vyhľadávanie na základe obsahu:

- Ochranné známky, autorské práva a logá
- Umelecké galérie a múzeá
- Móda a módný dizajn
- Dizajn interiéru a dekorácie

Napríklad, webový katalóg pre maloobchodné firmy s oblečením umožňuje užívateľom vyhľadávať podľa tradičných kategórií, ako je štýl alebo cenové rozpätie. Tiež obrazové vlastnosti (napr. farba alebo textúra). Zákazník môže požiadať o košeľe v určitom cenovom rozpätí a s požadovanou farbou.

3.2 Princíp obsahového porovnávania

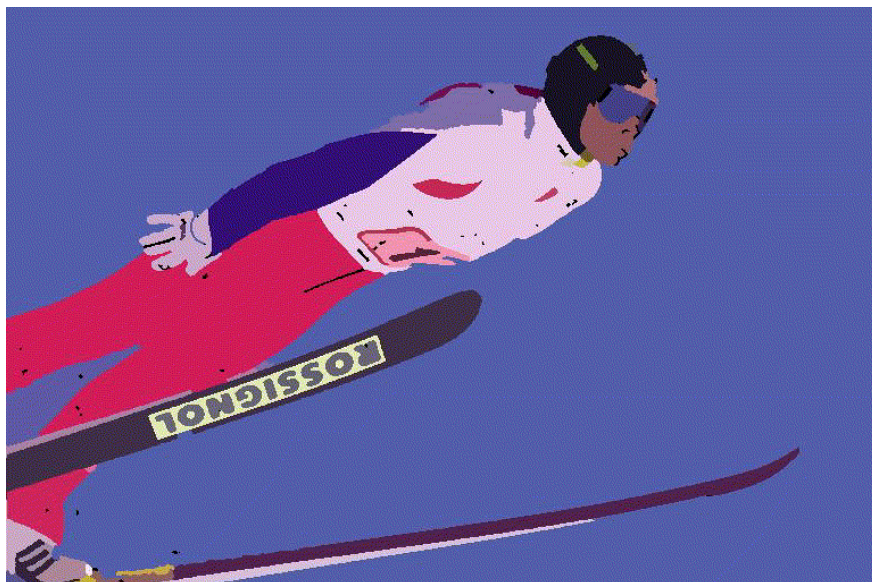
Vyhľadávací systém založený na obsahu vychádza zo spracovania informácie obsiahnutej v obrazových dátach a vytvára abstrakciu jeho obrazu z hľadiska vizuálnych atribútov. Všetky dotazy na tieto atribúty sa teda zaoberajú len jeho abstrakciou. Preto je každý obrázok vložený do databáze analyzovaný a jeho kompaktná reprezentácia jeho obsahu je uložená vo vektore, alebo v podpise.

Podpis pre obrázok 2 je získaný segmentáciou obrazu do regiónov na základe farby, ako je znázornené na obrázku 3. Každý segment má s ním spojené ostatné atribúty – farbu, textúru a tvar informácie. Podpis obsahuje informácie o segmente spolu s globálnou farbou,

textúrou a obrysom a reprezentuje tieto atributy pre daný obrázok. Obrázok 3 je rozdelený do 55 segmentov (plochy spojené podobnou farbou). Na pozadí je tiež tvar, ktorý sa skladá z malých tmavých plôch. Tieto malé plochy (zvyčajne rôznej farby) nepatria do žiadnych príslušných tvarov a sú brané ako tvar pozadia. Tvar pozadia je tiež braný do úvahy pri vyhl'adávaní.



Obrázok 2 Pôvodný obrázok [4]



Obrázok 3 Segmentovaný obrázok [4]

Obrázky sú porovnávané na základe farby, textúry a tvaru. Miesto týchto atribútov je reprezentované pozíciou. Lokácia nie je sama o sebe zmysluplným parametrom. Avšak v spojení s jedným z hlavných troch atribútov predstavuje hľadanie, kde sú oba atribúty rovnako dôležité.

Obrázkový podpis obsahuje nasledujúce vizuálne atribúty:

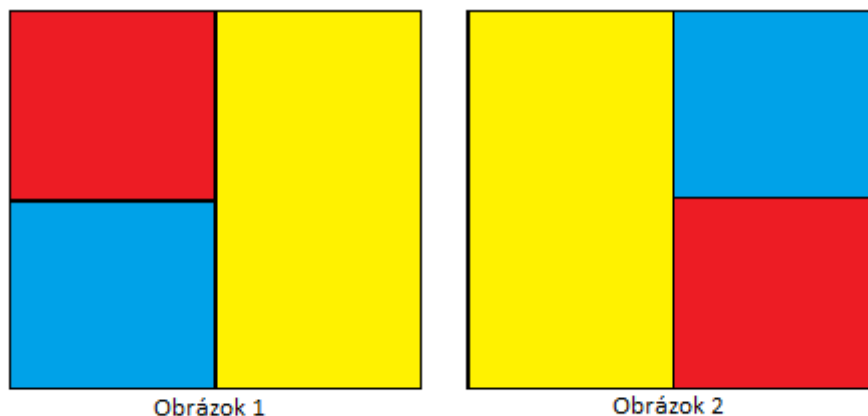
- Color – tento atribút predstavuje rozloženie farieb v rámci celého snímku. Obsahuje informácie o množstve každej zastúpenej farby.
- Texture – prezentuje vlastnosť obrázku ako je zrnitosť a jemnosť. Je citlivejší na funkcie s veľkou frekvenciou.
- Shape – zastáva tvary, ktoré sú v obrázku. Tak ako je reprezentovaná farba segmentačnou technikou, tak je tvar určený segmentom rovnakej farby.
- Location – predstavuje pozíciu komponentov tvaru, farby a textúry.

Všetky tieto vizuálne vlastnosti sú uložené v podpise, ktorého veľkosť sa pohybuje od 3000 do 4000 bytov. Pri veľkých databázach sa dá využiť indexovanie podpisov. Porovnanie môže prebehnúť medzi akýmikoľvek obrázkami vnútri, alebo aj mimo aktuálnej databázy, ktoré majú vygenerovaný obrázkový podpis.

Vyhľadávací proces vyžaduje, aby každý porovnávaný obraz mal vygenerovaný podpis pre porovnávací proces. Obrázky sú málokedy totožné, preto je pri vyhľadávaní rozhodujúca meracia funkcia pre vizuálne atribúty a zadané váhy. Score je relatívna vzdialenosť medzi dvoma porovnávanými obrazmi. Počet získaných bodov sa využíva na určenie stupňa podobnosti. Menšia vzdialenosť odráža väčšiu podobnosť.

3.2.1 Farba

Atribút farba odráža rozloženie farieb v rámci celého obrázka. Kombináciou farby umiestnenia dostaneme farebnú distribúciu na určitej danej ploche. Tento vzťah predstavuje obrázok 4.



Obrázok 4 Porovnanie na základe farby

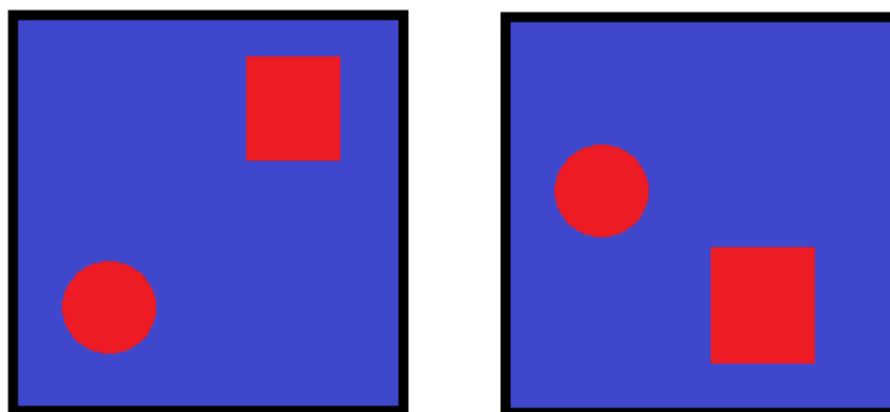
Obrázok 1 na ľavej a obrázok 2 na pravej strane a sú vyplnené farbami. V obrázku 1 je horná štvrtina červená (25%), dolná štvrtina modrá (25%) a pravá polovica je žltá (50%). Na ľavom obrázku je pravá štvrtina červená (25%), dolná štvrtina je modrá (25%) a ľavá polovica je žltá (50%).

Pri farebnom porovnávaní týchto dvoch obrázkov je výsledné skóre podstatne odlišné ako pri porovnávaní na základe farby a umiestnenia.

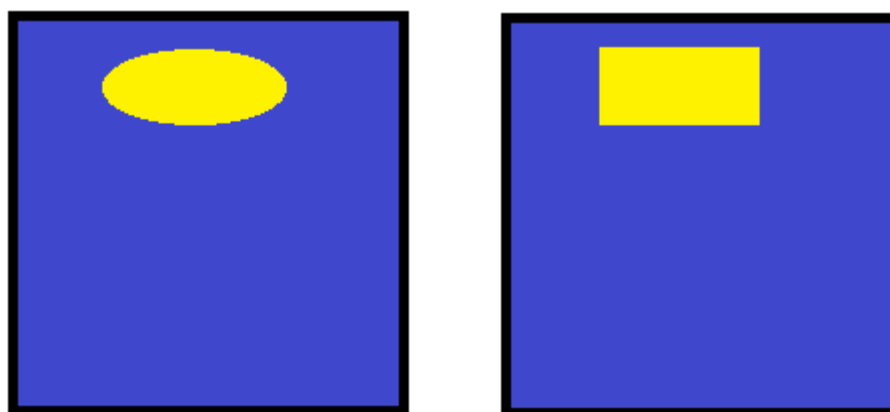
- Farba - obrázky sú totožné. Skóre je rovné nule, pretože každá z farieb je v oboch obrázkoch zastúpená v rovnakom pomere.
- Farba a umiestnenie - skóre s v tomto prípade bude rovnať 100 (žiadna zhoda). Parameter umiestnenia sleduje prekryvanie rovnakých farieb na danej ploche, čo sa v danom príklade nevyskytuje vôbec.

Ak je teda potrebné nájsť snímky na základe dominantnej farby, alebo farieb (napríklad nájsť autá červenej farby) je potrebné dať väčšiu váhu farbe. Naopak, ak hľadáme snímky s bežnými farbami v bežných miestach (napríklad dominantná zelená pri trávniku) je dôležité dať väčšiu váhu pre atribút umiestnenia.

Následujúci príklad znázorňuje obrázky veľmi podobné vo farbe. Druhý ukazuje obrázky podobné ako vo farbe tak aj v umiestnení.



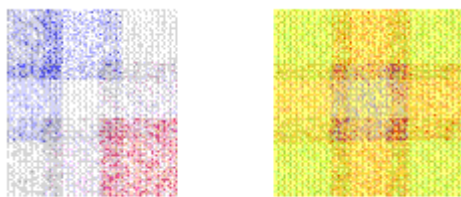
Obrázok 5 Obrázky podobné vo farbe



Obrázok 6 Obrázky podobné vo farbe a umiestnení

3.2.2 Textúra

Textúra popisuje štruktúru celého snímku. Táto atribúta je užitočná najmä pre obrazy plné textúr ako sú katalógy rôznych materiálov, tapiet, vzorkov atď. Takéto typy obrázkov je všeobecne ťažké roztriediť podľa kľúčových slov, pretože sa ťažko jednoslovne opisujú. Textúra môže byť efektívne použitá samostatne (bez farby), ale aj s farbou, pre niektoré špecifické prípady ako je drevo, alebo tkaniny. Následujúci obrázok ukazuje príklad obrázkov podobných v textúre.



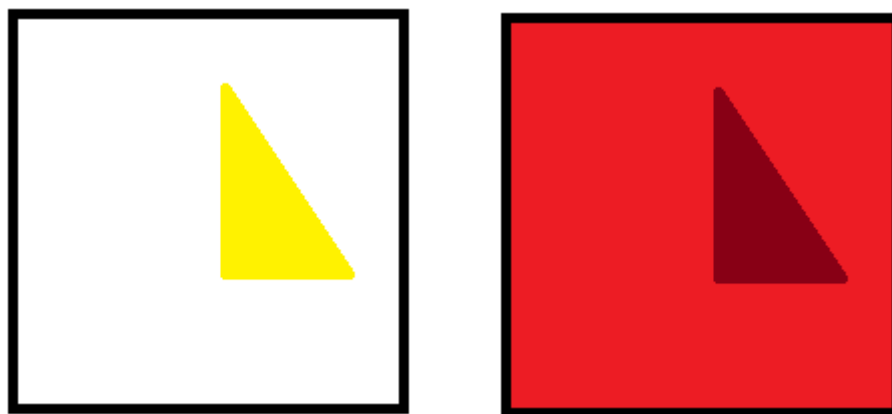
Obrázok 7 Obrázky s podobnou textúrou [4]

Ak sú špecifikované oba parametre (textúra aj umiestnenie) rovnako sa porovnáva umiestnenie tvarovaných oblastí v obraze.

3.2.3 Tvar

Atribút Shape interpretuje farby, ktoré sú zastúpené v obraze. Tvary sú v tomto prípade definované ako oblasti s rovnakou farbou. Užitočné je to najmä pre zachytenie horizontálnych línií v krajine, obdĺžnikových tvarov na budovách, alebo prírodné tvary ako napríklad stromy. Dotazy v tomto prípade fungujú najviac na jednoduchých tvaroch (kruhy, mnohoúhelníky, alebo diagonálne línie) a na dotaze na obraz kreslený ručne.

Následujúci obrázok ukazuje dva obrázky veľmi podobné v tvare.



Obrázok 8 Obrázky podobné v tvare

Ak sú definované parametre tvar a umiestnenie, berú sa do úvahy s rovnakou váhou.

3.3 Ako vyhľadávanie funguje

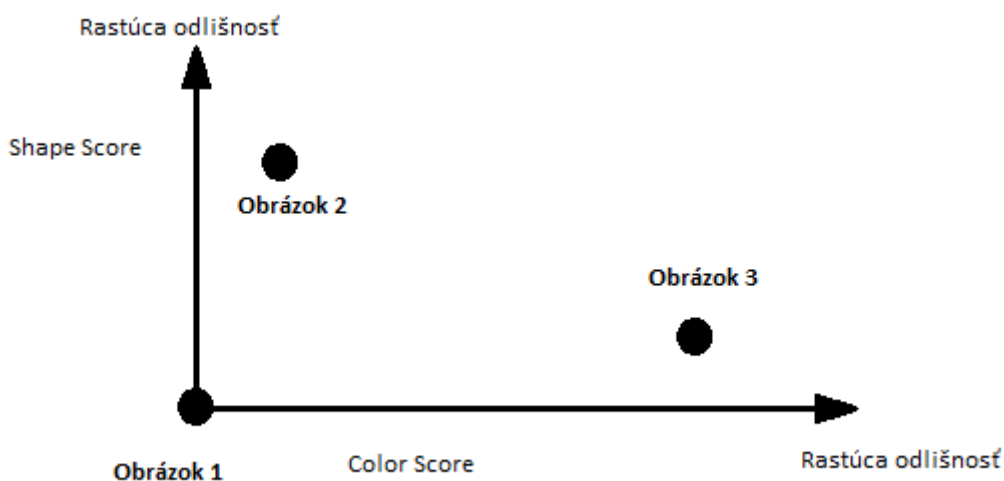
Ak vyhľadáваме podľa obsahu, je potrebné priradiť význam, váhy jednotlivým atributom. Podľa nich sa vypočíta podobnosť podľa našej potreby.

3.3.1 Váha

Každá hodnota váhy odráža ako citlivo sa má posudzovať daný atribut pre určenie podobnosti, alebo podobnosti medzi dvoma obrázkami. Napríklad ak pri hľadaní podobného snímku zadáme pre váhu farby hodnotu 0,0, prípadná zhoda obrázkov vo farbe nebude braná do úvahy. Na druhej strane ak je pre nás farba najdôležitejším atribútom a priradíme jej väčšiu váhu ako ostatným parametrom, budú výsledky výrazne ovplyvnené farbou. Rozsah váh sa môže zadávať v intervale od 0,0 do 1,0. Pri spracovaní, sú hodnoty spracované tak, aby celkový súčet bol 1. Váha aspoň jednej atribúty z farby, textúry alebo tvaru musí mať hodnotu vyššiu ako 0 [4].

3.3.2 Score

Podobnosť, pre každú danú atribútu je počítaná ako score - je to vlastne vzdialenosť medzi oboma obrázkami s ohľadom na daný atribút. Rozsah score sa pohybuje od 0.00 (žiadny rozdiel) po 100.0 (maximálny možný rozdiel). Príklad ako sa určuje score je na obrázku 9. Obrázok predstavuje tri obrázky s ohľadom na dva vizuálne atribúty - farbu a tvar vykreslené na osiach x a y.



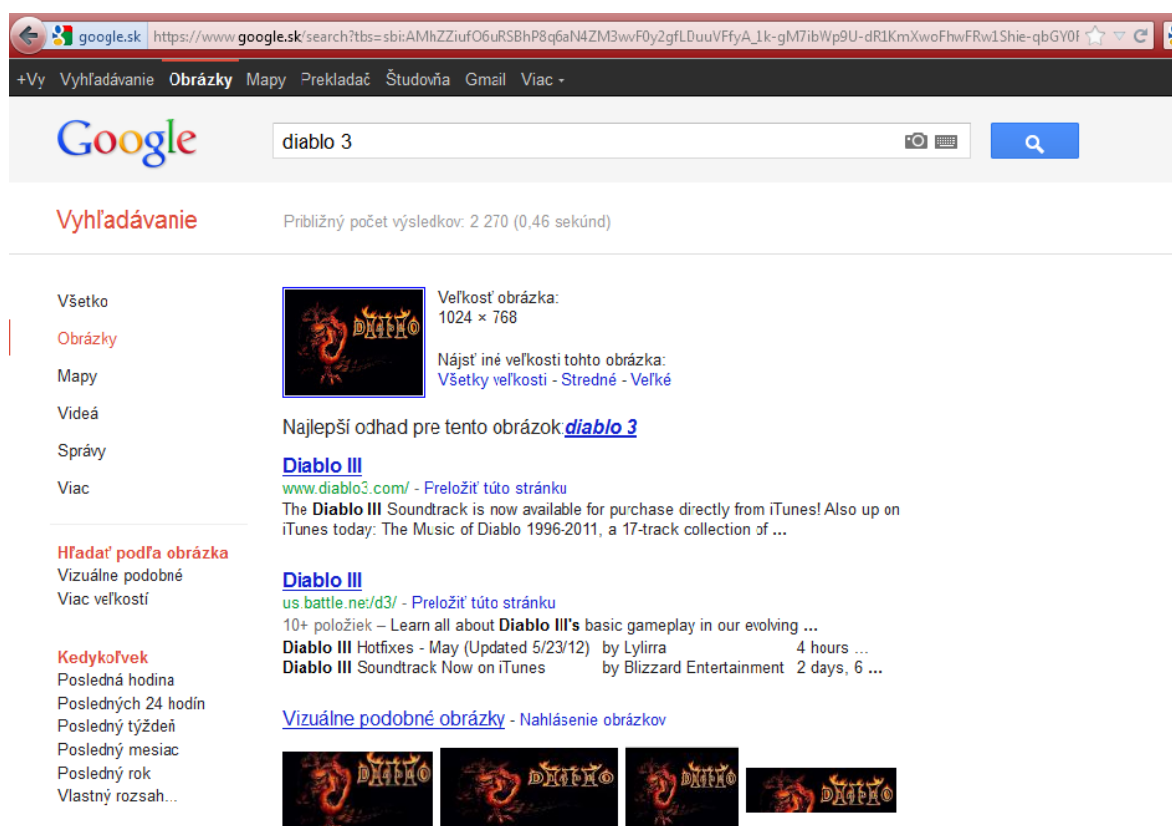
Obrázok 9 Znáozornenie výpočtu score

Na obrázku je znázornený obrázok 1 ako porovnávaný obrázok s obrázkami 2 a 3. Pokiaľ ide o atribút farby vynesenej na osi x, vzdialenosť medzi obrázkami 1 a 2 je relatívne malá (napr. 15), pričom vzdialenosť medzi 1 a 3 je oveľa väčšia (napr. 75). Ak by bola najväčšia váha na atribút farby, je pravdepodobné, že vo výsledku hľadania asi nebude ani

obrázok 2 ani obrázok 3. Ak však bude minimalizovaná váha farby a bude kladený väčší dôraz na tvar, potom sa bude obrázok 3 zhodovať s obrázkom 1 lepšie ako obrázok 2 s obrázkom 1.

3.3.3 Využitie

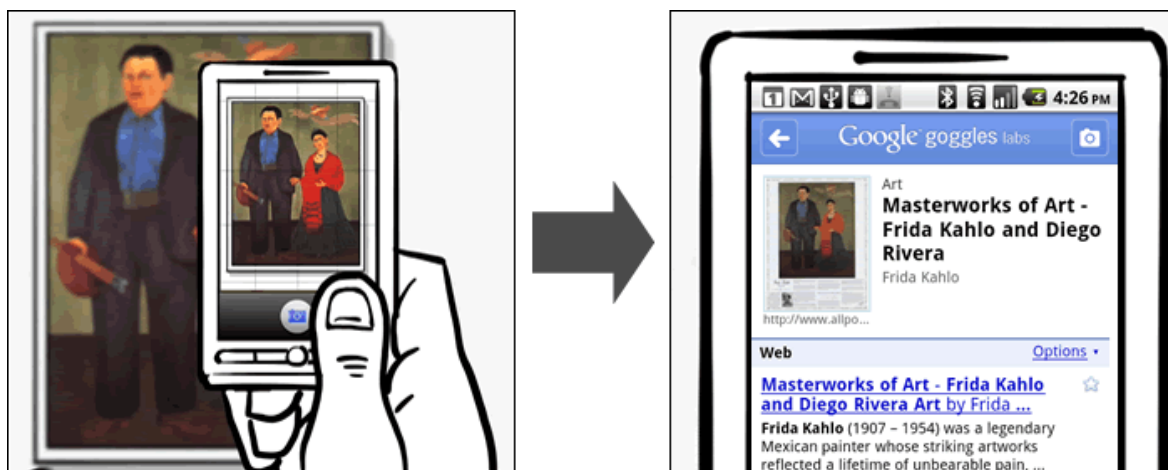
Vyhľadávanie podľa obsahu je v súčasnosti veľmi populárna záležitosť. Spoločnosť Google ponúka vyhľadávanie obrázkov na základe zdrojového obrázka. Vďaka jej veľkej databáze sú tieto vyhľadávania veľmi presné a využívané. Príklad užitia je uvedený nižšie.



Obrázok 10 Vyhľadanie v Google

Zdrojový obrázok, ktorý sa nahrával na porovnanie bol pomenovaný obrazok.jpeg. Dokonca je z veľkej časti čierny a aj cez tieto parametre dokázal vyhľadávací algoritmus správne vyhodnotiť obsah, vyčítať o aký titul ide a ponúknuť takmer totožné obrázky. Samozrejme Google disponuje obrovským množstvom dát, takže je jasné, že výsledky budú čo najpresnejšie.

Čo sa týka vyhľadávania podľa obsahu Google zašiel ešte ďalej. V máji roku 2010 vydal aplikáciu na mobilné telefóny pre OS Android a iOS, ktorá vyhodnocuje obsah fotografie odfotenej priamo fotoaparátom na mobilnom telefóne. Google Goggles dokáže z fotky určiť pred akou pamiatkou sa nachádzate, poprípade aké auto prešlo okolo vás. Táto aplikácia je vám dokonca schopná pomôcť s vylúštením sudoku, alebo preložiť cudzojazyčný text.



Obrázok 11 Google Goggles [13]

II. PRAKTICKÁ ČÁST

4 NÁVRH APLIKÁCIE

Pri návrhu webovej aplikácie na spracovávanie obrázkov sme museli v prvom rade brať do úvahy to, že databáza je schopná obrázky ukladať, porovnávať a editovať, ale nie je schopná ich zobraziť. Preto musia byť obrázky fyzicky uložené na serveri pre ich zobrazenie danému užívateľovi. Tak isto musí byť webové rozhranie prehľadné a funkčné v každom webovom prehliadači.

Ďalšou dôležitou časťou bolo navrhnuť databázu, do ktorej sa ukladá každý obrázok a edit obrázku. Užívateľ tak má prehľad o všetkých svojich nahratých obrázkoch, ktoré bude môcť editovať, vyhľadávať v nich. Do databáze sa tak isto ukladá aj každý záznam editu obrázku, ktorý sa využíva na zobrazovanie.

4.1 Vývojové prostriedky

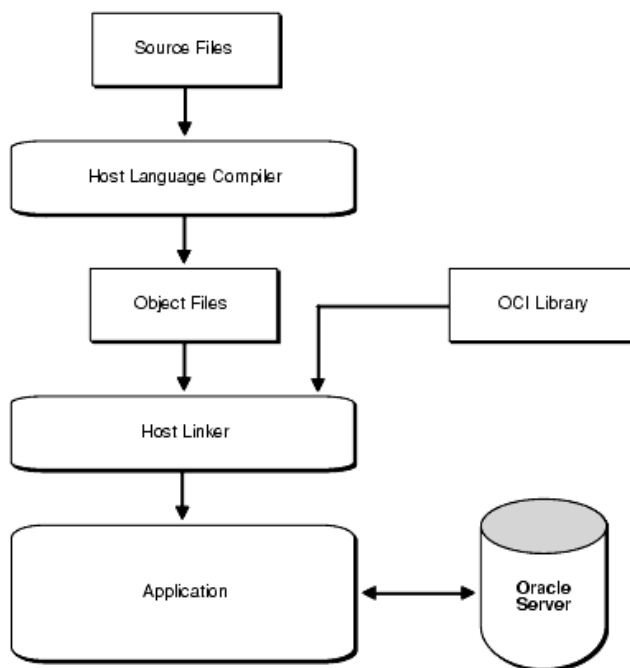
Pre tvorbu webovej aplikácie je potrebné vybrať si správny jazyk. V našom prípade to bol skriptovací jazyk PHP, kvôli jeho pomerne veľkej podpore v komunikácii s databázou Oracle prostredníctvom Oracle Call Interface. Ďalšie prostriedky na uľahčenie a sprehľadnenie práce boli napr. CSS, HTML, PL/SQL.

4.1.1 PHP

PHP (PHP: Hypertext Preprocessor) je široko používaný skriptovací jazyk, ktorý sa vo veľkej miere využíva na programovanie a vývoj webových aplikácií. Skripty sú vykonávané na strane serveru. PHP podporuje mnoho knižníc pre rôzne účely. Jednou z najhlavnejších je komunikácia s takmer každým databázovým systémom [12].

4.1.2 Oracle Call Interface

Call Oracle Interface (OCI) je rozhranie pre programovanie aplikácií (API), ktoré umožňuje vytvárať aplikácie, ktoré používajú natívne procedúry alebo volanie funkcie z jazykov tretej generácie pre prístup k databáze Oracle a kontrolovať všetky fázy realizácie SQL príkazu. OCI podporuje dátové typy, syntax a sémantiku C a C ++.



Obrázok 12 Vývoj OCI prístupu [8]

Prístup k databáze pomocou OCI má niekoľko výhod oproti iným prístupom. Medzi najvýraznejšie patria:

- Podpora dynamického SQL
- Dynamické bindovanie parametrov
- Vysoký stupeň kontroly nad spusteným programom
- Optimalizácia pre dotazy pomocou „prefetch buffer“
- Vylepšené pole pre manipuláciu s dátami (DML)

Pre komunikáciu PHP s Oracle databázou je dostupný modul OCI8, ktorý umožňuje svojimi funkciami prácu s databázou [8]. Preto je vo veľkom využívaný vo webových aplikáciach založených na Oracle.

4.1.3 SQL a PL/SQL

SQL je štruktúrovaný vyhľadávací jazyk, ktorý sa používa pri práci s databázovými systémami. Služi na výber, vkladanie, úpravu, mazanie a definíciu dát uložených v databáze. V základnej forme jazyka SQL však nedokážeme vykonávať niektoré zložitejšie operácie, ktoré sú pri práci s multimediálnymi dátami potrebné.

Preto viaceré firmy vyvinuli nadstavby pre jazyk SQL, ktoré dopĺňujú funkčnosť a komplexnosť. Napríklad firma Microsoft vyvinula jazyk T-SQL, firma IBM jazyk SQL PL a pod. V našom prípade sme využívali jazyk PL/SQL (Procedural Language/SQL) vyvinutý priamo firmou Oracle. Pridáva k jazyku SQL procedurálne prvky programovania. Umožňuje vytvárať procedúry, cykly, výnimky atď [5]. V tomto jazyku sú napísané všetky procedúry na úpravu, editovanie a porovnávanie obrázkov .

4.1.4 HTML a CSS

HTML (HyperText Markup Language) je značkovací jazyk, ktorý sa využíva na publikovanie dokumentov a textu na Internete. HTML elementy sú základom pri webových stránkach. Je písaný vo forme HTML prvkov, ktoré tvorené hranatými zátvorkami(napr. <div>). Niektoré sú párové (napr. <div></div>), iné sú samostatné, teda nepárové (napr.). Webový prehliadač HTML dokumenty číta a skladá ich do optických webových stránok. Do HTML stránok je možné vkladať skripty v mnohých jazykoch, ako je napríklad jazyku php, JavaScrip, Flash a pod.

Design webových stránok sa tvorí pomocou kaskádových štýlov (CSS). Definujú vzhľad, rozloženie textu, veľkosť písma atď... Kaskádové štýly nemusia byť výhradne používané len v dokumentoch typu HTML, ale aj dokumentoch typu XML, vrátane SVG a XUL. CSS slúži predovšetkým k tomu, aby sa oddelila programová a designová časť dokumentu. Umožňuje to väčšiu flexibilitu, kontrolu, zdieľanie jednotlivých parametrov a celkové sprehládnenie.

4.1.5 Oracle Database

Oracle Database je objektovo relačný databázový systém (RDBMS). Dáta ukladá logicky vo forme tabuľkových a fyzicky vo forme dátových súborov. Užívateľ si v databáze vytvára svoju schému, v ktorej si definuje svoje tabuľky, triggre, zložky atď. Prvá verzia Oracle Database bola verzia 5. V našom prípade sme mali k dispozícii nainštalovanú verziu 11g, z roku 2008.

4.1.6 Apache

Ako HTTP server aplikácia využíva Apache Server. Je to najpoužívanejší open-source HTTP server na Internete. Je multiplatformný a podporuje celú radu programovacích

jazykov vrátane PHP čo pre zadanú úlohu výhodné. Tak isto je jeho veľkou výhodou, že sa dá využívať ako localhost. V tomto prípade teda ide o kombináciu Windows, PHP, Oracle a Apache.

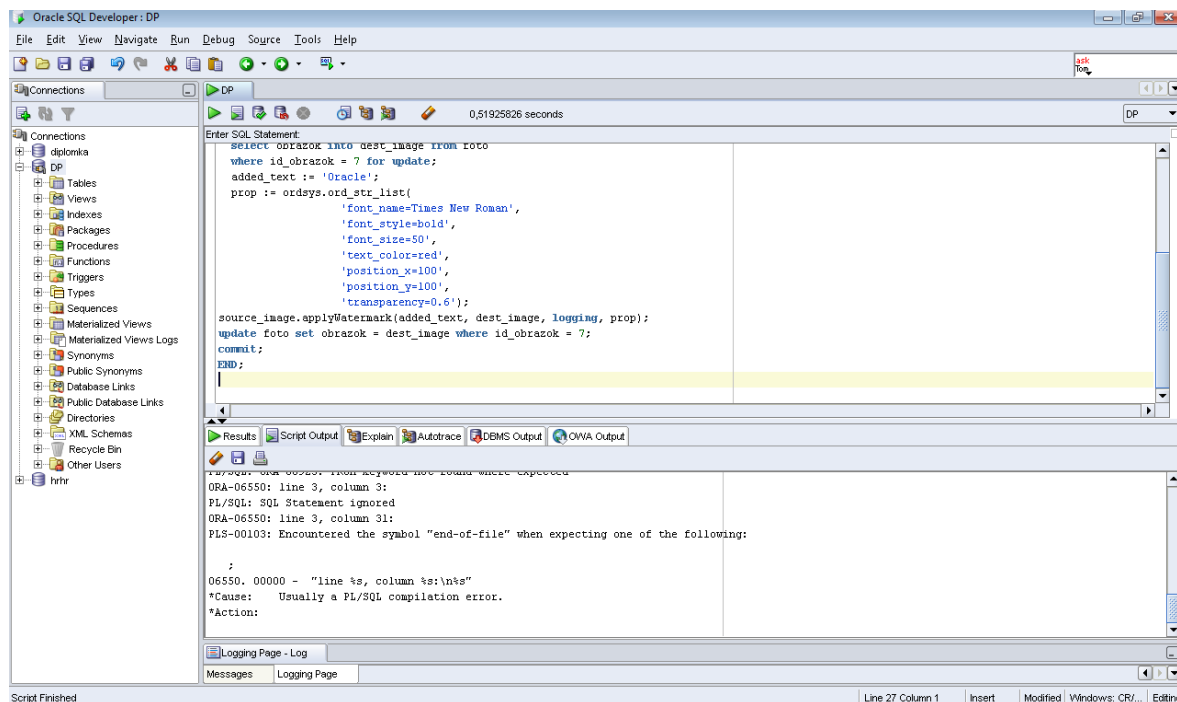
5 PRÍPRAVA DATABÁZE

Pred začiatkom samotného vývoja aplikácie si musíme pripraviť databázu na vkladanie záznamov a dát. Základom je mať vytvorenú vlastnú schému, v ktorej je možné si vytvoriť potrebné tabuľky a zložky.

5.1 Vytvorenie schémy

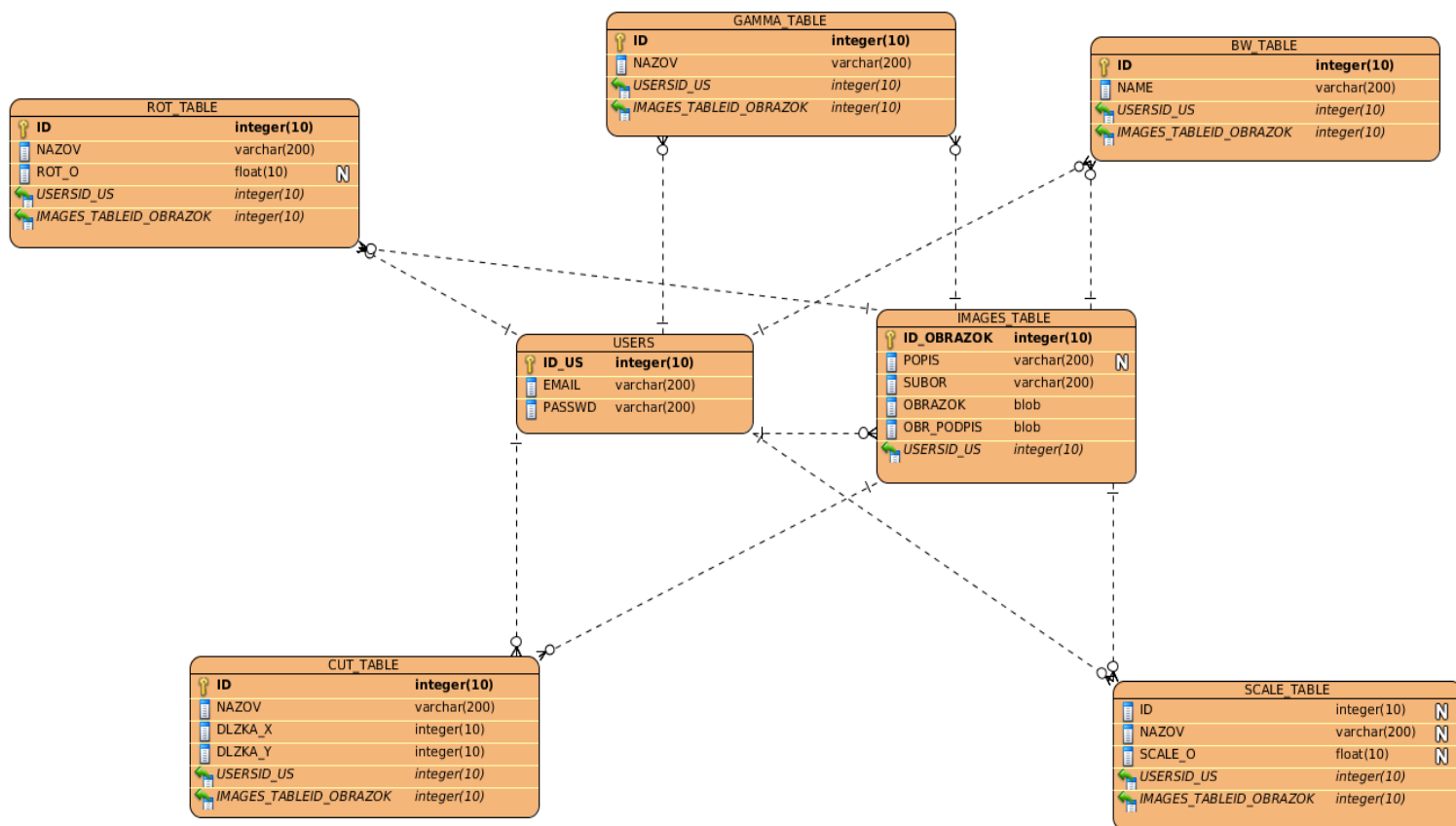
Každý užívateľ databázy potrebuje pre svoju prácu svoju vlastnú schému. Je to kolekcia databázových štruktúr ako sú pohľady, tabuľky, procedúry, indexy, zložky, sekvencie. Schéma nesie meno používateľa, ktorý ju riadi.

Schéma sa dá vytvoriť jednoducho pomocou webového rozhrania Enterprise manager. Po zadaní adresy do internetového vyhľadávača sa zobrazí prihlasovací dialóg, cez ktorý sa prihlásime a pomocou pár krokov si vytvoríme novú schému.



Obrázok 13 Pripojenie na schému

5.2 Vytvorenie tabuliek



Obrázok 14 Návrh tabuliek

Návrh tabuliek je jedna zo základných potrieb pre funkčnosť aplikácie. Musí sa prizeriť na to, že aj keď databáza dokáže pracovať s obrázkami, dokonca ich aj porovnávať podľa obsahu, nevie ich zobrazovať. Preto ak chceme aby mal užívateľ prehľad o svojich obrázkoch, musíme ich názvy a identifikačné čísla uchovávať.

Pre každú možnú zmenu obrázka je vytvorená tabuľka a po každom procese zmeny sa do tabuľky uloží ID užívateľa, ktorý túto zmenu vykonal a ID zdrojového obrázku, poprípade doplnujúce informácie. Potom je možné jednoznačne určiť, ktorý obrázok aký užívateľ vlastní a následne mu ho priradiť.

5.3 Vytvorenie zložky

Pre prácu s dátami ako sú obrázky, audio, alebo video súbory je potrebné mať vytvorenú v schéme zložku, ktorú využívame na import, export súborov. Metódy ako je napríklad `import()` využívajú ako parameter zdrojovú zložku. V podstate je to niečo ako zástupný znak umiestnenia zdrojového priečinku na serveri. Umiestnený môže byť kdekoľvek na disku. Vytvoríme ho pomocou príkazu:

```
CREATE OR REPLACE DIRECTORY IMAGES_MEDIADIR AS  
'C:\dev\public_html\dp\data';
```

Takýchto priečinkov môžeme mať viac, koľko potrebujeme pre svoju prácu. Je výhodné aby sa napríklad edity ukladali do svojich zložiek a mať tak dáta oddelené.

5.4 Pripojenie na databázu

K pripojenie na databázu pomocou OCI slúži funkcia `oci_connect` a v parametroch má meno, heslo, názov databáze. `OCI_connect` sa používa ako prvý parameter pri väčšine OCI funkcií. Aby nebolo meno a heslo spomínané pri každom úkone, vytvorili sme bezparametrovú funkciu `connect_oracle()`.

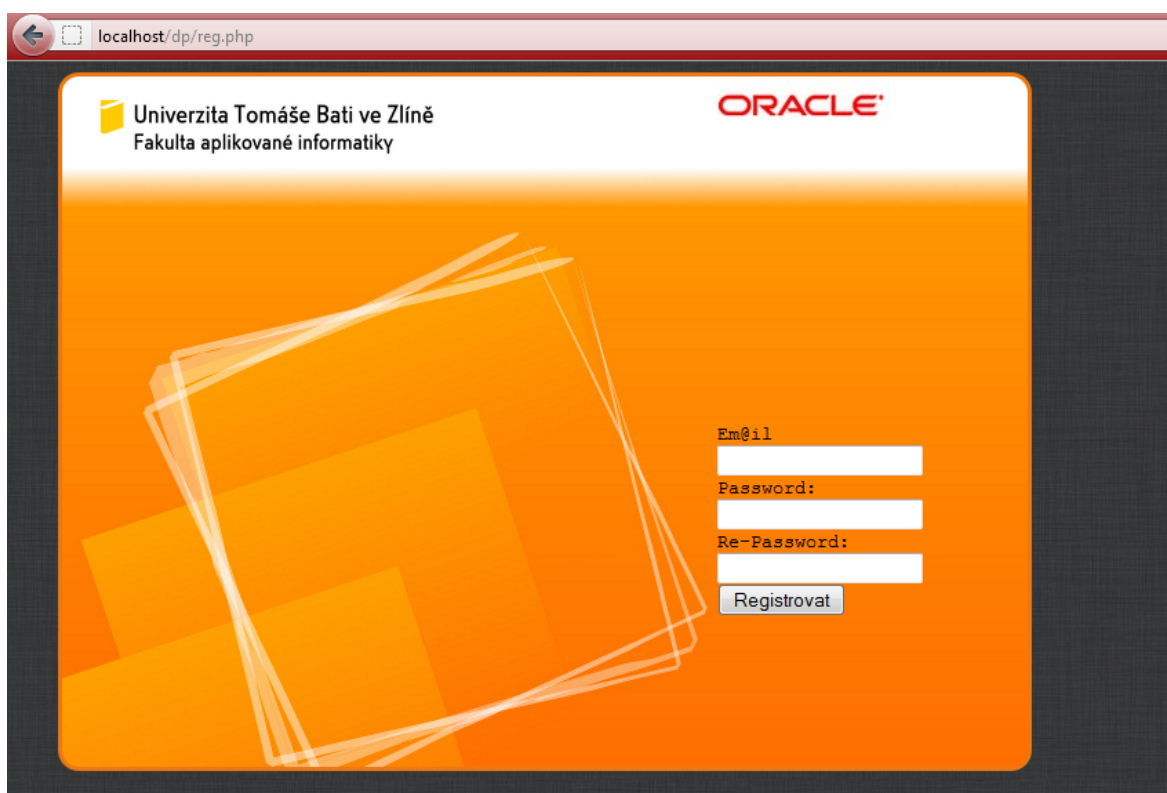
```
<?php  
function connect_oracle() {  
  
    $user = "user";  
    $pass = "pass";  
    $dbname = "name";  
    $charset = "set";  
  
    $conn = oci_connect($user, $pass, $dbname);}  
  
?>
```

6 UŽÍVATEĽSKÁ APLIKÁCIA

Výstupná webová aplikácia je naprogramovaná v jazykoch PHP, HTML. Skripty, ktoré spracúvajú obrázky a pracujú s metadátami sú napísané v jazyku PL/SQL. V predošlej kapitole sme jednotlivé jazyky v skratke charakterizovali. Všetky údaje s ktorými sa pracuje v aplikácií sú uchovávané a spracovávané v databáze Oracle.

6.1 Registrácia

Predtým ako sa užívateľ dostane dovnútra systému, musí sa zaregistrovať, aby mu systém povolil prístup a aby všetky jeho obrázky a editované obrázkov boli identifikované jeho prideleným identifikačným číslom. Pomocou neho má užívateľ zabezpečený jednoduchý prehľad o svojich obrázkoch.



The screenshot shows a web browser window with the address bar displaying 'localhost/dp/reg.php'. The page content is framed by a dark border. At the top left, there is a logo for 'Univerzita Tomáše Bati ve Zlíně, Fakulta aplikované informatiky'. At the top right, the 'ORACLE' logo is visible. The main area has an orange gradient background with a graphic of several overlapping, semi-transparent squares. On the right side, there is a registration form with three input fields labeled 'Email', 'Password:', and 'Re-Password:'. Below these fields is a button labeled 'Registrovat'.

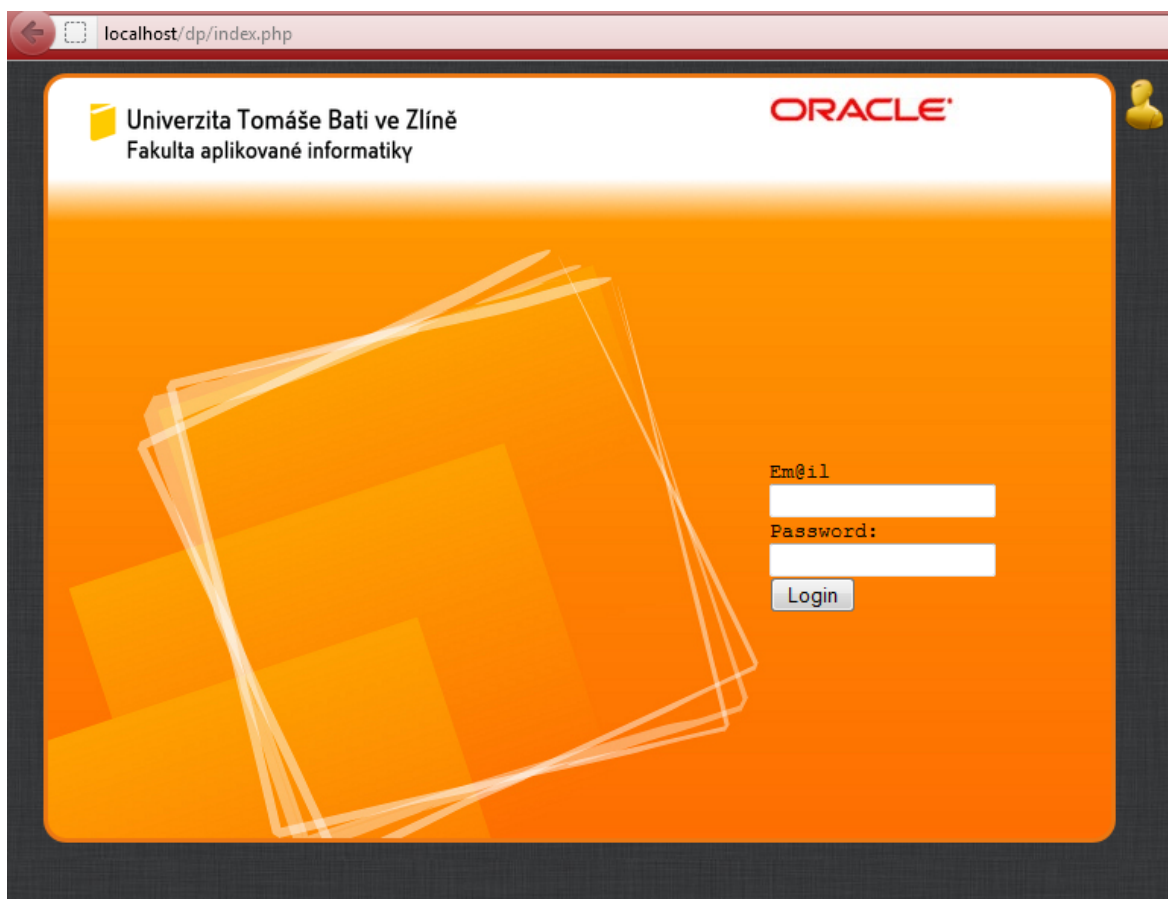
Obrázok 15 Registračná stránka

Užívateľ používa ako svoj nick/prezývku emailovú adresu, ktorá sa vpisuje do poľa „Em@il“. Následne stačí napísať heslo a druhý krát ho potvrdiť. Po odoslaní týchto údajov, je užívateľ zaregistrovaný a je mu umožnený prístup k aplikácií.

Po správnom vyplnení formulára sa vytvorí do tabuľky USERS nový záznam, ktorý obsahuje užívateľovu e-mailovú adresu a zahashované heslo. K týmto údajom sa pridá identifikačné číslo, ktoré jednoznačne identifikuje daného užívateľa. Pre každého nového užívateľa sa hodnota identifikačného čísla zvyšuje o 1.

6.2 Login

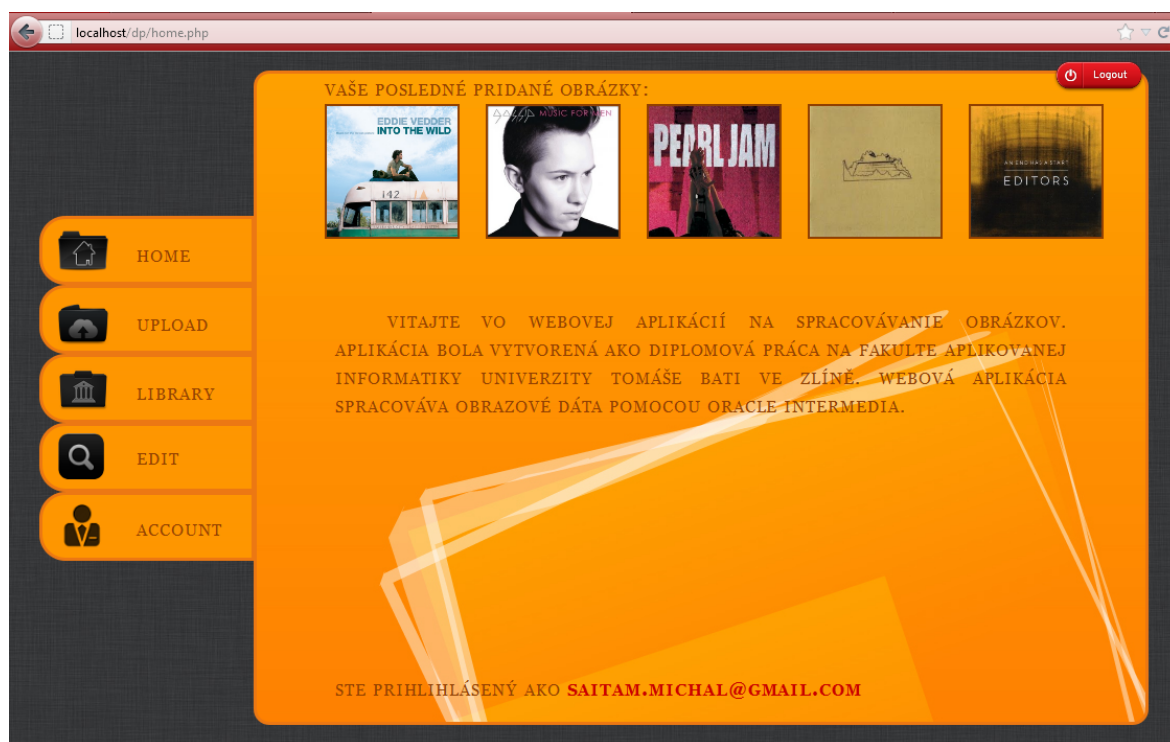
Po zaregistrovaní užívateľa je umožnený vstup do aplikácie. Ten je realizovaný pomocou prihlasovacieho formulára na stránke index.php. Po vložení mena a hesla sa tieto údaje odošlú a porovnajú s hodnotami v tabuľke USERS. Ak sa zhodujú, prístup je umožnený v inom prípade, je užívateľ upozornený o nesprávnom vyplnení prihlasovacieho mena, alebo hesla.



Obrázok 16 Prihlasovacie okno

6.3 Home stránka

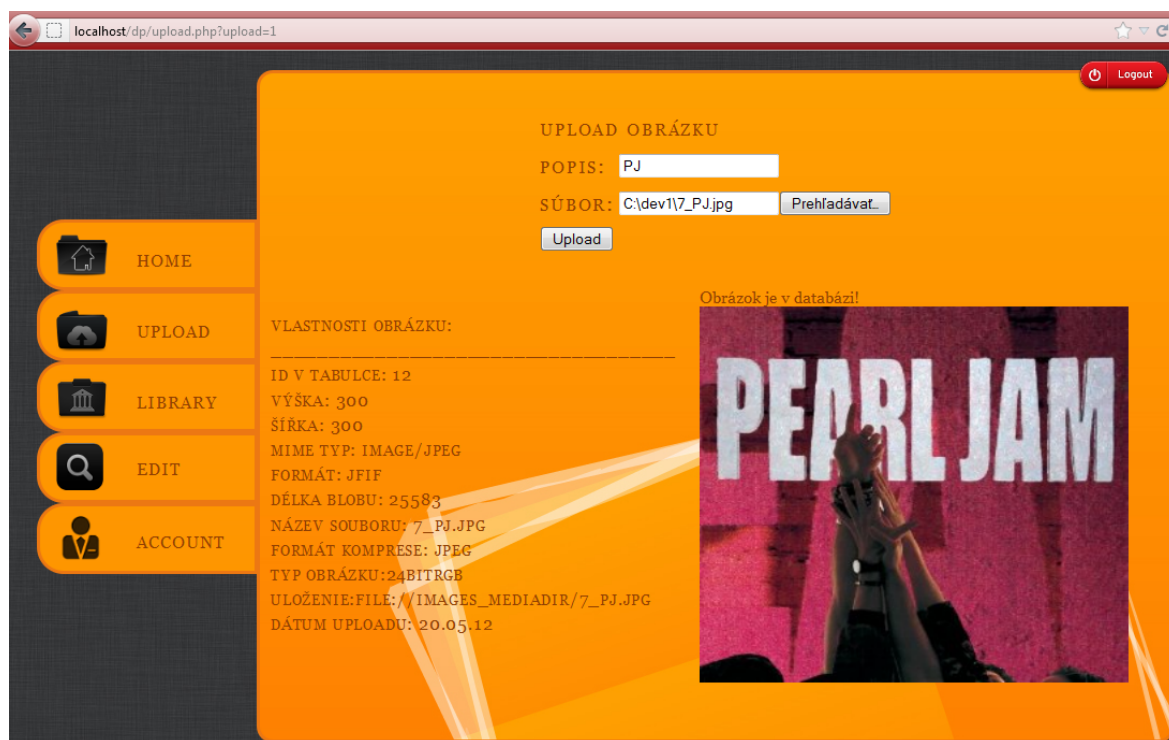
Po prihlásení sa dostaneme na „Home“ stránku získame prehľad o posledných pridaných obrázkoch užívateľom a názov konta, na ktorom sme prihlásený. Posledné pridané obrázky sa zobrazujú zľava od najnovšie pridaného. Na ľavej strane je umiestnené vertikálne menu, ktoré ponúka všetky možnosti aplikácie. V pravom hornom rohu je umiestnené tlačidlo na odhlásenie, ktoré užívateľa odhlási a presmeruje na stránku index.php.



Obrázok 17 Home stránka

6.4 Upload

Jednou z hlavných funkcií pre spracovanie obrázkov v databáze Oracle je načítanie obrázku do databáze, následné vygenerovanie obrázkového podpisu. Všetky funkcie, ktoré umožňuje interMedia totiž vychádzajú z podpisu a z obrázka uloženého v databáze. Sú v nich uložené všetky dôležité dáta a metadáta. Podľa nich dokáže interMedia hľadať podobné obrázky, vytiahnuť podrobné informácie o obrázku, prípadne porovnať dva obrázky medzi sebou. Popis uložených metadát v obrázku je priblížený v kapitole 2.3.



Obrázok 18 Upload obrázku

Pomocou tlačidla na prehľadávanie disku si užívateľ vyberie daný obrázok. Po jeho uploadnutí na server a nahratí do databáze sa vypíšu jeho vlastnosti.

6.4.1 Naplnenie tabuľky

Po nahratí obrázka na server, sa jeho základné údaje nahrávajú do tabuľky IMAGES. Pridelí sa mu jeho identifikačné číslo a číslo užívateľa, ktorým je nahratý. Tento proces sa vykonáva pomocou nižšie uvedeného skriptu. Zástupné znaky a bindovanie sa využívajú najmä v týchto prípadoch naplňovania tabuľky určitými dátami.

```
$ins_stmt=oci_parse(connect_oracle(),
"insert into images (id_obrazok, nazov, subor, OBRAZOK, OBRAZOK_PODPIS,
id_uz, edit)
values (:id_f, :nazov, :subor, ORDIImage.init('FILE',
'IMAGES_MEDIADIR',:subor), ORDIImageSignature.init(), :iduser,
ORDIImage.init())");
oci_bind_by_name($ins_stmt,':id_f', $idf);
oci_bind_by_name($ins_stmt,':nazov', $n_suboru);
oci_bind_by_name($ins_stmt,':subor', $u_subor);
oci_bind_by_name($ins_stmt,':iduser', $idUser);
oci_execute($ins_stmt);
oci_commit(connect_oracle());
oci_free_statement($ins_stmt);
oci_close(connect_oracle());
```

Po vykonaní skriptu je tabuľka naplnená údajmi ako je názov, popis, id obrázku, ale zároveň je aj pripravená na naplnenie samotným obrázkom, či vygenerovanie obrázkového podpisu.

6.4.2 Nahratie obrázka do databázy

Obrázok sa nahrá do databázy pomocou skriptu:

```
$ins_stmt_up=oci_parse(connect_oracle(),
"DECLARE
img ORDIMAGE;
ctx RAW(64) := NULL;

BEGIN

select obrazok into img from images where id_obrazok = :id_f for update;
img.setSource('file', 'IMAGES_MEDIADIR', :subor);
img.import(ctx);
update images set obrazok = img where id_obrazok = :id_f;
commit;
END;");
oci_bind_by_name($ins_stmt_up,':id_f', $idf);
oci_bind_by_name($ins_stmt_up,':subor', $u_subor);
oci_execute($ins_stmt_up);
oci_commit(connect_oracle());
oci_free_statement($ins_stmt_up);
oci_close(connect_oracle());
```

Funkcia `oci_parse` slúži v jazyku php na vykonanie daného PL/SQL skriptu. Má dva parametre. Prvým je pripojenie, v tomto prípade funkcia `connect_oracle()`, ktorá je nadefinovaná v zložke `/functions` na serveri a obsahuje prihlasovacie údaje potrebné k pripojeniu na databázu. Druhým parametrom je samotný PL/SQL skript. Oba parametre sa dajú zadať buď priamo, alebo pomocou premennej. Funkčnosť však ostáva rovnaká.

V prvej časti skriptu si nadefinujeme premenné, ktoré potrebujeme na vykonanie daného dotazu. V tomto prípade si nadefinujeme premennú `img` ako dátový typ `ORDIMAGE`. Druhou deklaráciou je `ctx RAW(64) := NULL` tá je použitá ako parameter pri metóde `import` a určí importovaciu metódu, ktorá vyčíta vlastnosti z BLOB-u.

Po príkaze `BEGIN` nasleduje samotný dotaz. Namiesto stĺpca `obrazok` použijeme našu nadefinovanú premennú `img`. Určíme pomocou podmienky `where`, ktorý obrázok sa uloží.

Tu používame zástupný znak `:id_f`, ktorý je nadefinovaný pomocou `oci_bind_by_name($ins_stmt_up,':id_f', $idf)`. Zo zápisu je zrejmé, že sa do skriptu namiesto zástupného znaku dosadzovať hodnota z premennej `$idf`, ktorá je

v našom prípade identifikačné číslo obrázka. Táto forma zápisu a samotné zastupovanie premennými je výhodná najmä pre aplikácie s automatickým ukladaním dát.

Ďalej musíme určiť, zdrojové miesto samotného obrázka. Na to slúži metóda `setSource()`. Zápis sa skladá z 3 parametrov. Prvý je typ zdroja, druhým je zložka, v ktorej sa súbor nachádza a posledným je názov. V podstate je to adresa obrázka. V ďalšom kroku obrázok už len nainportujeme do databázy. O to sa postará metóda `import(ctx);` s dopredu nadefinovanou importovacou metódou. Následne vykonáme update daného stĺpca.

V poslednom kroku je nutné vykonať ešte zvyšné metódy pre vykonanie a potvrdenie skriptu. Na to slúžia metódy `oci_execute($ins_stmt_up)` a `oci_commit(connect_oracle())`. Nakoniec uvoľníme vykonaný execute pomocou `oci_free_statement($ins_stmt_up)` a uzavrieme spojenie s databázou `oci_close(connect_oracle())`.

6.4.3 Generovanie obrázkového podpisu

Na generovanie obrázkového podpisu sa používa metóda pomenovaná príznačne `generateSignature()`. V podstate nahráme do pripraveného stĺpca podpis obrázku, podľa ktorého je Oracle schpný hľadať podobné obrázky. Ide o jednu zo základných vlastností, ktoré je nutné vykonať pre ďalšie využívanie Oracle interMedia.

```
$ins_stmt_sig=oci_parse(connect_oracle(),
"DECLARE
img ORDIMAGESignature;
obraz ORDImage;

BEGIN

select obrazok, obrazok_podpis into obraz, img from images where
id_obrazok = :id_f for update;
img.generateSignature(obraz);
update images set obrazok_podpis = img where id_obrazok = :id_f;
end;");
oci_bind_by_name($ins_stmt_sig,':id_f', $idf);
oci_execute($ins_stmt_sig);
oci_commit(connect_oracle());
oci_free_statement($ins_stmt_sig);
oci_close(connect_oracle());
```

Opäť sa využíva bindovanie podmienky, podľa ktorej je identifikovaný obrázok. Premenná `img` predstavuje stĺpec v tabuľke, kam sa podpis vygeneruje a uchová k ďalšiemu použitiu.

6.4.4 Výpis vlastností

Výpis vlastností obrázku je možné dvoma spôsobmi. Buď ich vyextrahovať do xml súboru a následne ich vypisovať, porovnávať, alebo editovať. Druhá možnosť je pomocou pripravených metód, ktoré obsahuje Oracle interMedia. Z metadát sa dá zistiť výška, šírka, dĺžka BLOB-u, zdroj atď.

Tak isto je možný výpis dvoma rôznymi spôsobmi. Prvým je pomocou obyčajného selectu, ktorý využijeme neskôr, alebo pomocou Server Outputu. V tomto konkrétnom prípade je výhodnejšie použiť Server Output, keďže je lepšie, ak sa nám vlastnosti nevypisujú do tabuľky, ale tak ako potrebujeme my.

```
$ins_stmt_meta=oci_parse(connect_oracle(),
"DECLARE
img ORDImage;
id_img integer;

BEGIN

SELECT id_obrazok, obrazok into id_img, img from images where id_obrazok
= :id_f;
DBMS_OUTPUT.PUT_LINE ('Vlastnosti obrázku:');
DBMS_OUTPUT.PUT_LINE ('-----');
DBMS_OUTPUT.PUT_LINE ('Id v tabuľke: ' || id_img);
DBMS_OUTPUT.PUT_LINE ('Výška: ' || img.getHeight());
DBMS_OUTPUT.PUT_LINE ('Šírka: ' || img.getWidth());
DBMS_OUTPUT.PUT_LINE ('MIME typ: ' || img.getMimeType());
DBMS_OUTPUT.PUT_LINE ('Formát: ' || img.getFileFormat());
DBMS_OUTPUT.PUT_LINE ('Dĺžka BLOBu: ' || TO_CHAR(img.getContentLength()));
DBMS_OUTPUT.PUT_LINE ('Název souboru: ' || img.getSourceName());
DBMS_OUTPUT.PUT_LINE ('Formát komprese: ' || img.getCompressionFormat());
DBMS_OUTPUT.PUT_LINE('Typ obrázku' || img.getContentFormat());
DBMS_OUTPUT.PUT_LINE('Umiestnenie: ' || img.getSource());
DBMS_OUTPUT.PUT_LINE('Dátum uploadu: ' || img.getUpdateTime());
END;");
oci_bind_by_name($ins_stmt_meta,':id_f', $idf);
```

Užívateľ si teda môže hneď po nahratí obrázku zistiť, ako s ním môže ďalej narábať, keďže niektoré operácie vykonávané pomocou interMedia sú limitované len na niektoré typy obrazových dát, resp. typy kompresíí. Tento krok je posledný, ktorý sa vykonáva pri nahrávaní obrázku na server a do databázy. Po vykonaní týchto operácií je obrázok pripravený na editovanie a je jednoznačne identifikovaný ako on samotný, tak aj jeho vlastník.

6.5 Library

Položka Library z hlavného menu slúži k prehľadu obrázkov a ich úprav. Každý užívateľ má pomocou nej zabezpečený prehľad o svojich obrázkoch a ich úpravách,

popřípadě tabulkový zoznam všetkých obrázkov aj s ich atribútmi. Pomocou jednoduchých selectov sa užívateľovi zobrazia buď len originály obrázkov, alebo len ich úpravy. Tretou variantou je zobrazenie všetkých obrázkov, ktoré sú vedené v databáze pod jeho identifikačným číslom.

Veľkou výhodou je, že za pomoci metadát a metód určených na ich vypisovanie je ľahké zistiť pomocou jednoduchého selectu napr. výšku, šírku, alebo názov súboru. Tým pádom nahrádza jeden stĺpec mnoho ďalších a šetrí tak prácu aj nároky na databázu.

ID	Názov Súboru	Výška	Šírka	Dátum Uploadu	Typ Obrázka	Typ Kompresie	Popis
1	1.png	400	400	22.05.12	32BITRGBA	DEFLATE	hte
2	abart.jpg	452	630	22.05.12	24BITRGB	JPEG	fiat
3	booklet.jpg	286	350	22.05.12	24BITRGB	JPEG	bkdt
4	ed_art.jpg	450	450	22.05.12	24BITRGB	JPEG	ed
5	coco.jpg	700	700	22.05.12	24BITRGB	JPEG	Parov
6	Diablo3-wp8.jpg	768	1024	22.05.12	24BITRGB	JPEG-PROGRESSIVE	d3
7	federer.jpg	310	324	22.05.12	24BITRGB	JPEG	rf
8	federer1.jpg	300	416	22.05.12	24BITRGB	JPEG	rf1
9	Ferrar.jpg	364	666	22.05.12	24BITRGB	JPEG	fer
10	ferrari.jpg	350	500	22.05.12	24BITRGB	JPEG	f2
11	Ferrari_288.jpg	285	400	22.05.12	24BITRGB	JPEG	old
12	FerrariShooting.jpg	332	500	22.05.12	24BITRGB	JPEG	frrr
13	FIFA_12.jpg	352	250	22.05.12	24BITRGB	JPEG	fifa
14	glory-1.jpg	422	700	22.05.12	24BITRGB	JPEG	bike
15	hte_desire.jpg	600	800	22.05.12	24BITRGB	JPEG-PROGRESSIVE	desire
16	HTC-EVO.jpg	300	300	22.05.12	24BITRGB	JPEG	evo
17	hte-google.jpg	600	600	22.05.12	24BITRGB	JPEG	google
18	htchd2v.png	383	588	22.05.12	24BITRGB	DEFLATE	hd2

Obrázok 19 Výpis zoznamu obrázkov

V tomto prípade je výpis realizovaný pomocou selectu a nie cez Server Output.

```
$librry = oci_parse(connect_oracle(),
'select id_obrazok, subor s,
i.obrazok.getHeight(), i.obrazok.getWidth(), i.obrazok.getUpdateTime(),
i.obrazok.getContentTypeFormat(), i.obrazok.getCompressionFormat(), nazov
from images i where id_uz = :id_u order by 1');
oci_bind_by_name($librry, ':id_u', $_SESSION["IDUSER"]);
oci_execute($librry);
```

Zoznam zobrazuje len obrázky práve prihláseného užívateľa. To je zabezpečené where podmienkou selectu.

6.6 Edit

V tejto položke si môže užívateľ rôzne editovať svoje obrázky. Tieto zmeny sú realizované pomocou PL/SQL skriptov, ktoré vykonáva databáza Oracle, resp. Oracle interMedia. Ponuka úprav je samozrejme väčšia ako tu zrealizovaných 5 operácií, avšak ostatné sú značne limitované typom kompresie, alebo typom obrázka. Preto sme sa rozhodli zrealizovať na ukážku 4 edity pre všetky formáty a 1 (monochrome efekt) pre obrázky s príponou png, alebo gif.

6.6.1 Rotate

Prvou možnosťou je rotovať obrázok o ľubovoľný počet stupňov. Do formulára na stránke stačí zadať ID obrázka a počet stupňov o koľko chceme daný obrázok orotovať. Obrázok sa točí v smere hodinových ručičiek. Po vykonaní daného skriptu sa výsledok objaví na stránke.

```
$ins_stmt_rotate=oci_parse(connect_oracle(),
"DECLARE
  idobr number;
  img_1 ORDSYS.ORDImage;
  img_2 ORDSYS.ORDImage;
  ctx Raw(64) := NULL;
  n_rot varchar2(100);
BEGIN
  SELECT obrazok, edit INTO img_1, img_2
  FROM images
  WHERE id_obrazok = :id_f FOR UPDATE;
  img_1.processCopy('rotate=$stupne', img_2);
  UPDATE images SET edit = img_2
  WHERE id_obrazok=:id_f;
  n_rot:='ROT$stupne$rot_nazov[SUBOR]';
  img_2.export(ctx, 'file', 'IMAGES_MEDIADIR', n_rot);
  COMMIT;
end;");
```

Na úpravu obrázkov sa môžu využívať dve metódy. V prvom prípade je možné využiť metódu process(). Táto metóda má jeden parameter a tým je žiadané zmeny na obrázku. Pri použití tejto metódy sa zdrojový obrázok mení a zmeny sa zapisujú priamo do neho. Tieto zmeny je možné vrátiť pomocou príkazu rollback.

Druhou používanou metódou je processCopy. Na rozdiel od predošlej metódy má dva parametre. Druhým je premenná, do ktorej sa zmeny zapisujú. Z toho vyplýva, že zdrojový obrázok ostáva nezmenený a vytvára sa nový obrázok so zmenami, ktorý sa nahráva na definované miesto na disku.

Výsledný názov exportovaného súboru je rovnaký ako zdrojového s určeným prefixom, ktorý obsahuje počet stupňov, o koľko sa obrázok otočil. To znamená, že ak má zdrojový obrázok názov jelen.jpg a orotujeme ho o 40 stupňov, výsledný obrázok má názov ROT40jelen.jpg.



Obrázok 20 Ukážka rotácie obrázku

6.6.2 Black/White

Táto možnosť úpravy robí z originálneho obrázku monochromatickú kópiu. Bohužiaľ jej funkčnosť je obmedzená len na určité prípony, tak ako veľa ostatných operácií. Skript na vytvorenie monochromatickej kópie je veľmi podobný ako v predchádzajúcom prípade. V metóde processCopy je vymenený prvý parameter, ktorý určuje úpravu. Zmena monochrome je funkčná bez ďalších parametrov, takže stačí zadať ID obrázka. Názov, ID pôvodného obrázka a ID užívateľa sú zapísané do ďalšej tabuľky.

6.6.3 Scale

Funkcia Scale umožňuje zmenšiť, alebo zväčšiť obrázok a to v dvoch variantách. Prvou je zmena veľkosti v percentách. Funkcia Scale reaguje na číselné typy float a to tak,

že v rozmedzí od 0 po 0.99 sa obrázok zmenší o požadovanú hodnotu. Ak zadáme číslo väčšie ako 1, potom sa obrázok zväčší.

Druhou možnosťou je pevná zmena veľkosti. To umožňuje funkcia `fixedScale`, ktorá má dva parametre a nimi sú veľkosť x a y , čiže šírka a výška obrázka. Tak isto ako v predchádzajúcom prípade môžu hodnoty presahovať pôvodné parametre obrázka, čím ale samozrejme trpí kvalita obrázku.

6.6.4 Gamma

Úprava gamy predstavuje zosvetlenie, alebo stmavenie jednej, dvoch alebo všetkých zložiek RGB v obrázku. Ak nám ide o zosvetlenie jednej, alebo viacero zložiek, hodnota sa musí pohybovať od 1.0 vyššie. Ak chceme dosiahnuť opačný efekt, hodnoty sa musia pohybovať v rozmedzí od 0 po 1, kde 0 znamená stlmenie farebnej zložky a 1 neznamená žiadnu zmenu.

Relatívne veľká odlišnosť sa týka zadávania parametrov do jednotlivých funkcií v skripte. Parametre zmien ako je napr. Zložky RGB sa nebindujú, ale zadávajú priamo ako premenná v jazyku PHP.

6.6.5 Cut

Posledná úprava umožňuje orez obrázka. Zadáva sa pomocou štyroch parametrov. Prvé dva určujú počiatok orezu. Ďalšie dva jeho šírku a výšku. Miernou nevýhodou u tejto metódy je fakt, že musíme poznať rozmery obrázka, čo môže byť pre veľkých rozlíšeníach problém. Tak isto ako všetky predošlé možné úpravy je aj Cut realizovaný pomocou metódy `processCopy`. Po každom vykonanom oreze sa teda výsledný obrázok nahrá na server a je zobrazený medzi editovanými obrázkami.

6.6.6 Similar

Následujúca funkcia už needituje obrázok, ale porovnáva dva zvolené obrázky na základe obsahu. Pre porovnania sa využíva spomínaný obrázkový podpis jedného a druhého obrázku. Mieru zhody v tomto prípade vyjadruje score, ktoré je vygenerované pomocou metódy `evaluateScore()`. Táto metóda má tri parametre prvými dvoma sú podpisy oboch obrázkov a tretím sú zadané váhy. Bližší význam parametrov je popísaný v kapitole

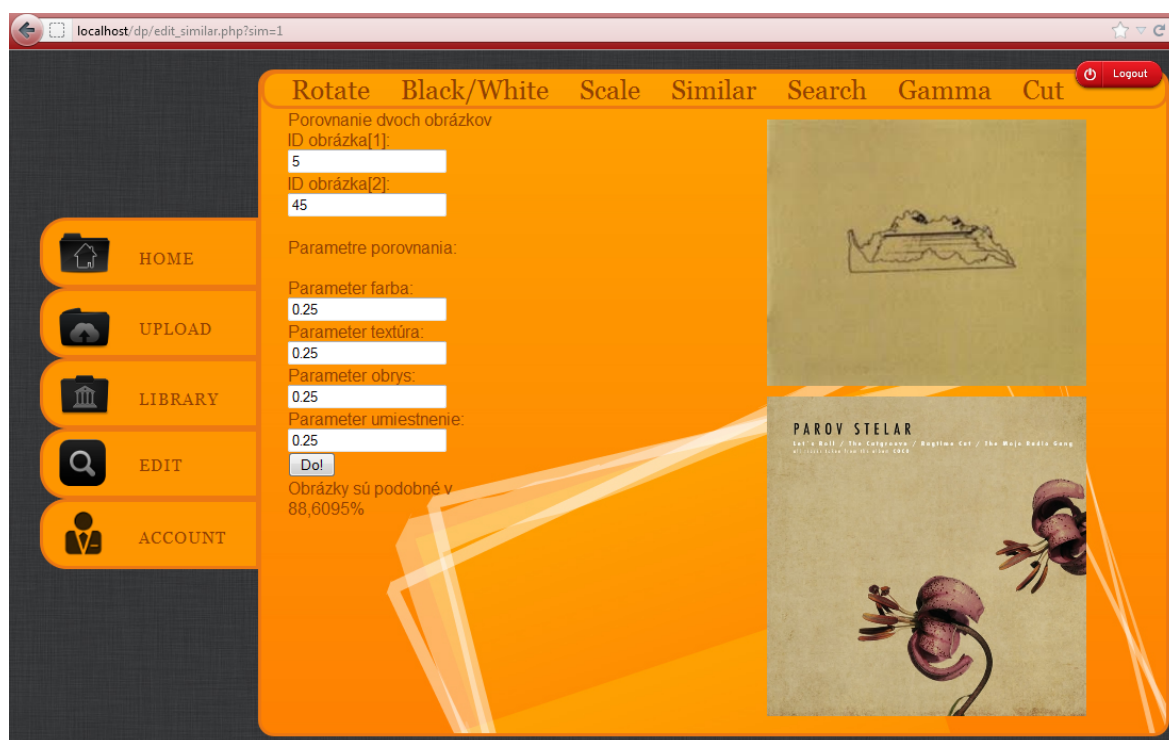
3.2. Score, ktoré je výstupom metódy určuje mieru nezhody, preto je výsledné score odpočítané od 100.

```
$ins_stmt_similar=oci_parse(connect_oracle(),
"declare
podpis1 OrdImageSignature;
podpis2 OrdImageSignature;
score float;

begin

select obrazok_podpis into podpis1 from images where id_obrazok=:id_o1;
select obrazok_podpis into podpis2 from images where id_obrazok=:id_o2;
score:=ORDImageSignature.evaluateScore(podpis1,podpis2,'color=$col,textur
e=$tex,shape=$shape,location=$loca');
score:=100-score;
dbms_output.put_line('Obrázky sú podobné v '||score||'%');
end;");
```

Po vykonaní skriptu sa nám objavia oba obrázky aj s výsledným score.



Obrázok 21 Výsledok porovnania

6.6.7 Search

Táto funkcia databázy slúži na vyhľadanie najpodobnejšieho obrázku z databázy podľa nami určených kritérií. Oracle dokáže vyhľadať najpodobnejšie obrázky podľa ich vygenerovanej signatúry aj keď obrázky samé o sebe nedokáže zobrazit'. Kritéria podľa

ktorých vyhladáme sú farba, textúra, obrys a umiestnenie. Algoritmus je založený na porovnávaní obrázku ako celku, takže nedokáže rozpoznať napr. Rovnaké tváre, alebo odtlačky prstov.

V tomto prípade používame na zrealizovanie metódu `IMGSimilar()`. Táto

```
$ins_stmt_por=oci_parse(connect_oracle(),
"SELECT id_obrazok,subor, ORDSYS.IMGScore(1) score
FROM images f,
(select obrazok_podpis from images where id_obrazok=:id_o1) porov
WHERE
ORDSYS.IMGSimilar(f.obrazok_podpis,porov.obrazok_podpis,
'color=$col,texture=$tex,shape=$shape,location=$loca', 60,1)=1
ORDER BY 3");
```

Výsledkom dotazu je score, ktoré určuje ako ďaleko sú vzdialené podobnosti obrázku.

Následujúci príklad ukáže výsledky dotazov na porovnávanie podľa obsahu. Obrázok s názvom logo je logo spoločnosti Apple. Ostatné obrázky sú obyčajné jablká, farba je napísaná v názve.

ID_OBRAZOK	NÁZOV	SCORE
4	Apple_purp_logo.jpg	0
6	Apple_blue_logo1.jpg	14,1429
24	Apple_green_logo.jpg	15,1975
34	Apple_black_logo.jpg	16,0669
27	Apple_green1_logo.jpg	16,0862
45	Red_apple.jpg	18,7211
5	Red_apple.png	18,9722

Tabuľka 1 Výsledok dotazu podobnosti

Výsledok nám ukazuje, že čím je výsledné score nižšie, tým sú si obrázky podobnejšie. V tomto prípade je porovnávaný obrázok s ID 4. Dotaz prehľadáva celú databázu, takže je jasné, že najpodobnejší je sám sebe. Dôraz je kladený na obrys a umiestnenie, preto sa ostatné výsledky líšia vo farbe a textúre. Naopak ak by sme pridali dôraz na farbu a textúru, výsledky sú nasledovné.

ID_OBRAZOK	NÁZOV	SCORE
4	Apple_purp_logo.jpg	0
3	Red_apple.jpg	9,1431
8	Red_apple2.jpg	9,1488
24	Apple_green_logo.jpg	13,3808
43	Apple_rainbow_logo.png	13,5903
18	Apple_green1_logo.png	18,1084
30	Apple_blue_logo.jpg	18,2816

Tabuľka 2 Výsledok podobnosti

Výsledky týchto dvoch dotazov nám jednoznačne ukazuje, ako celé porovnávanie funguje. Kým v prvom sú najpodobnejšie len logá, v druhom sú logá na nižších priečkach. Najnižšie score získali obrázky s obyčajnými jablkami, aj keď nemajú úplne rovnaký obrys, no textúra a farba je oveľa bližšia ako u obrázkov, ktoré sú totožné krivkami.

6.6.8 Ďalšie funkcie

Oracle investuje veľa času a prostriedkov na vývoj interMedia a každou novou verziou Oracle Database sa možnosti rozširujú. Už teraz je veľký výber funkcií na spracovávanie obrázkov, ktoré neobsahuje táto aplikácia ako je napr. zrkadlenie, zmena kontrastu, formátu, zaujímavé funkcie pre formát DICOM.

V najnovšej verzii Oracle Database 11g R2 sa objavila zaujímavá funkcia Watermarks. Pomocou tejto funkcie sa dá pridať do obrázku vodotlač s pomerne širokým nastavením.

```

DECLARE
    source_image ORDSYS.ORDImage;
    added_text varchar2(200);
    dest_image ORDSYS.ORDImage;
    prop ordsys.ord_str_list;
    logging VARCHAR2(2000);

BEGIN

    select obrazok into source_image from foto
    where id_obrazok = 6;
    select obrazok into dest_image from foto
    where id_obrazok = 7 for update;
    added_text := 'Oracle';
    prop := ordsys.ord_str_list(
        'font_name=Times New Roman',
        'font_style=bold',

```

```
        'font_size=50',  
        'text_color=red',  
        'position_x=100',  
        'position_y=100',  
        'transparency=0.6');  
source_image.applyWatermark(added_text, dest_image, logging, prop);  
update foto set obrazok = dest_image where id_obrazok = 7;  
commit;  
END;
```

Ak by bola verzia 11g R2 k dispozícií, určite by bola táto služba spracovaná v aplikácií, pretože možnosť pridania vodoznaku do obrázku bez potreby softwaru by bola určite vítaná. Databáza dokáže okrem formátovaného textu pridať ako vodoznak aj iný obrázok, čo sa určite hodí pri ochranných známkach a pod.

ZÁVĚR

Cieľom tejto práce bolo vytvoriť webovú aplikáciu na spracovávanie obrázkov pomocou databáze Oracle. Webová časť je vytvorená pomocou skriptovacieho jazyka PHP, ktorý má relatívne dobrú podporu pre komunikáciu s databázou Oracle pomocou Oracle Call Interface. Veľkou výhodou je možnosť bindovania parametrov jednotlivých metód, uľahčuje to prácu s dátami a tiež zvyšuje bezpečnosť a rýchlosť.

V rámci praktickej časti sme sa presvedčili o tom, že databáza Oracle má široké možnosti využitia, ktoré budú každou novou verziou určite ešte širšie. Rovnako veľmi príjemnou a využiteľnou funkciou je vyhľadávanie na základe obsahu obrázku. Použitý algoritmus na vyhodnocovanie pracuje spoľahlivo. Zadávanie váh jednotlivých parametrov je pomerne citlivé, preto chvíľu trvá, kým si na to užívateľ zvykne.

Menšou nevýhodou je, že databáza sama o sebe dokáže obrázky upravovať a porovnávať, no nedokáže ich zobraziť. Preto musia byť obrázky uložené na serveri, čo spôsobuje redundantnosť dát a tým aj väčšie požiadavky na kapacitu pevného disku. Naopak veľkou výhodou je práca s metadátami. Informácie v nich dopodrobna charakterizujú obrázok a tým dávajú oveľa väčšie možnosti na prácu s nimi. Metadáta nie je potrebné nikam ukladať, stačí ich vyexportovať pomocou metód na to určených a hneď sa s nimi dá pracovať ako s dátami uloženými v samostatných stĺpcoch tabuľky.

ZÁVĚR V ANGLIČTINĚ

The aim of this study was to develop a web application for image processing using an Oracle database. Web part is developed using PHP scripting language, which has relatively good support for communication with the Oracle database using the Oracle Call Interface. The big advantage is the possibility binding parameters of these methods, it makes it easier to work with data and also increases security and speed.

In the practical part, we are convinced that the Oracle database has a wide range of uses that each new version will definitely still wider. Also very nice and usable search function is based on image content. The algorithm used to evaluate work reliably. Entering weights of each parameter is quite sensitive, so takes a while to get used to the user.

Some small overhead is that the database itself can compare and edit images, but can not view them. Therefore, images must be stored on the server, resulting in data redundancy and the even greater demands on hard disk capacity. On the other hand is a great advantage to work with metadata. Information they describe in detail the image and give a much greater opportunity to work with them. Metadata can not be stored anywhere, just to export them using methods intended to be and now they can be handled as data stored in separate columns in the table.

SEZNAM POUŽITÉ LITERATURY

- [1] LACKO, Luboslav. *Oracle: správa, programování a použití databázového systému*. 2. dopl. vyd. Překlad Marek Kocan. Brno: Computer Press, 2007, 576 s. ISBN 978-80-251-1490-2.
- [2] LECKY-THOMPSON, Ed a Steven D NOWICKI. *PHP 6: programujeme profesionálně*. Vyd. 1. Překlad Ondřej Gibl. Brno: Computer Press, 2010, 718 s. Programujeme profesionálně. ISBN 978-80-251-3127-5.
- [3] KOSEK, Jiří. *PHP - tvorba interaktivních internetových aplikací: podrobný průvodce*. Vyd. 1. Praha: Grada, 1999, 490 s. Průvodce (Grada). ISBN 80-716-9373-1.
- [4] ORACLE. *Oracle Multimedia Reference. 11g Release 1 (11.1)*. Redwood Shores, 2007. Dostupné: http://docs.oracle.com/cd/B28359_01/appdev.111/b28414/toc.htm
- [5] URMAN, Scott, Ron HARDMAN a Michael MCLAUGHLIN. *Oracle: programování v PL/SQL*. Vyd. 1. Překlad Jiří Fadrný. Brno: Computer Press, 2007, 720 s. ISBN 978-802-5118-702.
- [6] DUNCKLEY, Lynne a Larry GUROS. *Oracle 10g: Developing Media Rich Application*. Oxford: Elsevier, 2007. 4. Dostupné z: <http://goo.gl/CkRj0>
- [7] GREENWALL, Rick a Jonathan STERN. *Oracle Essentials: Oracle Database 11g*. 4. Dostupné z: <http://goo.gl/F3PFv>
- [8] JONES, Christopher a Allison HOLLOWAY. *The Underground Php and Oracle Manual*. 1. vyd. Oracle, 2008.
- [9] PELSKI, Sue. *Oracle interMedia: User's Guide* [online]. 2005 [cit. 2012-05-12].
- [10] Databázový svět. [online]. Dostupné z: <http://www.dbsvet.cz/>
- [11] Oracle Time Line. [online].
Dostupné z: <http://oracle.com.edgesuite.net/timeline/oracle/>
- [12] PHP: Hypertext Preprocessor. [online]. Dostupné z: <http://php.net/>

[13] Google Goggles. [online].

Dostupné z: <http://www.google.com/mobile/goggles/#logo>

[14] W3Schools. [online]. Dostupné z: <http://www.w3schools.com/>

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

JAI	Java Advanced Imaging
JVM	Java Virtual Machine
RTSP	Real Time Streaming Protocol
JSP	Java Server Pages
JMF	Java Media Framework
PL/SQL	Procedural Language/Structured Query Language
SQL	Structured Query Language
OCI	Oracle Call Interface
BLOB	Binary Large Object
XML	eXtensible Markup Language
HTML	HyperText Markup Language
CSS	Cascading Style Sheets
SVG	Scalable Vector Graphics
XUL	XML User Interface Language
PHP	PHP: Hypertext Preprocessor
ID	Identifikačné číslo
RGB	Red Green Blue
R2	Release 2

SEZNAM OBRÁZKŮ

Obrázok 1 Architektúra Oracle interMedia [4]	14
Obrázok 2 Pôvodný obrázok [4]	21
Obrázok 3 Segmentovaný obrázok [4]	21
Obrázok 4 Porovnanie na základe farby	23
Obrázok 5 Obrázky podobné vo farbe	24
Obrázok 6 Obrázky podobné vo farbe a umiestnení	24
Obrázok 7 Obrázky s podobnou textúrou [4]	25
Obrázok 8 Obrázky podobné v tvare	25
Obrázok 9 Znázornenie výpočtu score	26
Obrázok 10 Vyhľadanie v Google	27
Obrázok 11 Google Goggles [13]	28
Obrázok 12 Vývoj OCI prístupu [8]	31
Obrázok 13 Pripojenie na schému	34
Obrázok 14 Návrh tabuliek	35
Obrázok 15 Registračná stránka	37
Obrázok 16 Prihlasovacie okno	38
Obrázok 17 Home stránka	39
Obrázok 18 Upload obrázku	40
Obrázok 19 Výpis zoznamu obrázkov	44
Obrázok 20 Ukážka rotácie obrázku	46
Obrázok 21 Výsledok porovnania	48

SEZNAM TABULEK

Tabuľka 1 Výsledok dotazu podobnosti.....	49
Tabuľka 2 Výsledok podobnosti	50

SEZNAM PŘÍLOH

Zdrojový kód webové aplikace a PL/SQL skripty na přibalenom CD.