



KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
PROGRAMŲ SISTEMŲ STUDIJŲ PROGRAMA

Martynas Juzaitis

VEIDŲ ATPAŽINIMO ĮRANKIO SUKŪRIMAS

Programavimo praktikos kursinis darbas

Vadovas
Dėst. V. Gurgždys

KAUNAS, 2014

TECHNINĖ UŽDUOTIS

1. Užduotis: Sukurti programinę įrangą, kuri nurodytose nuotraukose surastų žmogaus veidus. PĮ nurodytas nuotraukas turi apdoroti vienu kartu. Surasti veidai turi būti susieti su nuotraukomis, kuriose jie rasti. Veidams galima priskirti vardus. Turi būti realizuota nuotraukų paieška pagal nurodytą vardą.
2. Sistemos paskirtis ir tikslas. Supaprastinti žmogaus paiešką aibėje nuotraukų.
3. Apribojimai sistemai.
 - 3.1. PĮ turi veikti Windows operacinėje sistemoje.
 - 3.2. PĮ kuriama C# programavimo kalba.
 - 3.3. Duomenų prieigai naudojama Entity Framework arba NHibernate (*) technologijos.
 - 3.4. Užklausų rašymui naudojama Linq technologija.
 - 3.5. Duomenų saugojimui naudojama MS SQL duomenų bazė arba XML duomenų failai.

Turinys

1. Užduotis	2
2. Sistemos paskirtis ir tikslas.....	2
3. Apribojimai sistemai.....	2
Ivadas	4
Darbo problematika ir aktualumas	4
Darbo tikslas ir uždaviniai	4
Darbo struktūra	5
1. Užduoties Analizė	6
2. Sistemos architektūra	7
2.1. Funkciniai reikalavimai.....	7
2.2. Nefunkciniai reikalavimai.....	7
2.3. Panaudos atvejų diagrama.....	7
2.4. Paketų diagrama	8
2.5. Duomenų bazės diagrama	8
2.6. Klasių diagrama.....	10
2.7. Veiklos diagrama	11
2.8. Vartotojo sąsajos projektas.....	14
3. Sistemos realizacija ir testavimas	16
3.1. Vartotojo sąsajos realizacija	16
3.2. Sistemos realizacija.....	17
3.3. Sistemos testavimas	18
4. Dokumentacija naudotojui	20
4.1. Sistemos diegimo vadovas	20
4.2. Sistemos vartotojo vadovas.....	22
5. Išvados ir rezultatų apibendrinimas	23
5.1. Rezultatai	23
5.2. Išvados	23
6. Literatūra.....	24

IVADAS

Darbas priklauso Programų sistemų studijų programai, specializacija programų sistemos. Modulis P175B137Programavimo technologijų praktika.

Praktinis darbas atliekamas C-sharp programavimo kalba, kuri sukurta kompanijoje Microsoft kaip dalis .NET iniciatyvos. Tai programavimo kalba, kuri paprasta, moderni ir sparčiai populiarėjanti tarp programuotojų.

Kadangi darbo užduotis yra sukurti programinę įrangą, kuri turi veikti internetinėje erdvėje, pasitelkiama ASP.NET tinklapio struktūros technologija, kurios pagalba, kuriamos dinaminės internetinės svetainės. ASP.NET technologija yra suderinta su bet, kuria .NET programavimo kalba, nes ji yra Microsoft .NET platformos dalimi.

Darbo problematika ir aktualumas

Fotografija yra neatsiejama šiuolaikinės kultūros dalimi. Daugybėje nuotraukų yra užfiksuojamas mus supantis pasaulis. Jo dalimi yra žmonės. Siekiant paprastam vartotojui palengvinti darbą klasifikuojant nuotraukas yra kuriami įvairūs programiniai įrankiai.

Šiuo darbu siekiama sukurti tokį įrankį, kuris leistų paprastam interneto vartotojui sukelti turimas nuotraukas į savo aplanką ir suklasifikuoti jas pagal nuotraukose esančius žmones. Esant reikalui vartotojas galės surasti visas nuotraukas, kuriose yra užfiksuojamas konkretus asmuo. Tai palengvina nuotraukų paiešką, jeigu nuotraukų archyvas yra didelės apimties.

Praktinė darbo reikšmė apima kelis aspektus:

- 1) Programuotojas atlikdamas uždavinį susipažįsta C# objektinio programavimo kalba. Susipažįsta kaip C# kalba rašomas ASP.NET kodas ir kaip jis integruojamas į html bylų formatą, kurio pagrindu kuriami žiniatinklio formos puslapiai.
- 2) Programuotojas susipažįsta su Microsoft Visual Studio programine įranga ir jos funkcionalumu;
- 3) Programuotojas analizuoja, kaip į programą integruojami sudėtingi algoritmai, kurių pagalba apdorojama vaizdinė informacija tiek nuotraukose tiek ir video įrašuose.

Darbo tikslas ir uždaviniai

Pagrindinis darbo tikslas sukurti internetinį puslapį, kuriame vartotojas be didelių pastangų galėtų sukelti nuotraukas ir suklasifikuoti jas pagal jose esančius žmones, o konkrečiau jose matomus žmonių veidus. Atliekama darbą susiduriame su šiais uždaviniais:

- 1) Sukurti dinaminį interneto puslapį, kuris veiktų pagrindinėse naršyklėse;
- 2) pateikti vartotojui patogiai ir lengvai valdomą įrankį nuotraukų talpinimui į vartotojo aplanką;

- 3) be vartotojo įsikišimo surasti nuotraukose žmonių veidus ir leisti vartotojui suteikti jiems vardus. Vardai gali kartotis, nes vienas žmogus gali būti keliuose nuotraukose.
- 4) sukurti vartotojui galimybę peržiūrėti visas sukeltas nuotraukas ir jose esančius veidus;
- 5) suteikti vartotojui galimybę ieškoti nuotraukas pagal veidams suteiktus vardus, esant būtinybei leisti vartotojui koreguoti vardus ar net ištrinti nuotraukas su jose užfiksuotais veidais;

Darbo struktūra

Darbas susideda iš šių skyrių:

- 1) Užduoties analizė. Šioje dalyje analizuojama, kokį įrankį pageidauja gauti vartotojas. Kokie yra vartotojo pageidaujami funkciniai ir nefunkciniai reikalavimai įrankiui. Atliekama algoritmų, kurie yra būtinį įrankio kūrimui analizė, taip pat analizuojami algoritmo reikalavimai pradiniam duomenims, argumentams ir rezultatams. Pateikiami sprendimų metodai;
- 2) Sistemos architektūra. Aprašoma sprendimo projektavimo stadija, pateikiama detali darbo specifikacija (Panaudos atvejų, klasių ir sekų diagramos, apibrėžiama kuriamo sprendimo vizija (konceptija)). Analizuojami konkretūs parašyto kodo elementai, kurie suteikia kuriamam įrankiui funkcionalumą;
- 3) Sistemos realizacija ir testavimas: Pateikiama informacija susijusi su sukurtos įrangos realizacija, testavimu. Pateikiama sukurtos programos vizualiniai pavyzdžiai, kuriuose turi aiškiai matytis sistemos architektūros dalyje aprašytos funkcijos;
- 4) Dokumentacija naudotojui. Pateikiama informacija vartotojui, kaip įdiegti įrankį ir kaip naudotis jame sukurtomis funkcijomis;
- 5) Išvados ir rezultatų apibendrinimas. Pateikiama analizė, kaip pavyko realizuoti užduotis, kurias susiformavome užduočių dalyje;

1. UŽDUOTIES ANALIZĖ

Pagal užsakovo užsakymą reikia sukurti internetinį tinklą, kuriame turi būti galimybė talpinti nuotraukas. Turi būti sukurtas įrankis patalpintose nuotraukose padedantis surasti žmonių veidus.

Turi būti vartotojui sukurta galimybė identifikuotiems veidams suteikti vardus.

Nuotraukos turi būti talpinamos serveryje, vis informacija apie jas turi būti talpinama duomenų bazėse, kad esant reikalui vartotojas galėtų surasti konkrečią nuotrauką arba nuotraukas su identifikuotais veidais.

Turi būti galimybė vartotojui rasti pagal žmogaus vardą visas nuotraukas, kurios susijusios su tuo žmogumi.

Duomenims saugoti pasirenkama MSSQL server 2008 duomenų bazės, nes jos yra puikiai suderinamos su kitais Microsoft produktais naudojamais užduoties atlikimui. Tam kad būtų paprasčiau dirbti su duomenų bazėmis pasirenkame Fluent NHibernate duomenų valdymo karkasą.

Kadangi reikia, kad įrankis apdorotų vaizdinę medžiagą yra importuojama EmguCV biblioteka, kuri pritaikyti nuotraukų analizei. Nuotraukos analizei panaudojama CascadeClassifier klasė, kuri naudodama haarcascade algoritmus nespaltvotose nuotraukose identifikuoja visus potencialius objektus, kurie galėtų būti žmogaus veidu. Vėliau objektus susistemina ir pateikia realias kvadrato formos koordinates, kur yra rasti veidai.

EmguCV biblioteka ir jos funkcijos praplečia OpenCV (Open source computer vision) biblioteką, kurios pagalba yra atpažįstami objektai nuotraukose. OpenCV veikimas:

OpenCV veido paieška naudoja Viola-Jones metodą vaizdo medžiagoje nustatyti norimus objektus [5]. Nuotraukų apdorojimui naudojama nespaltvota vaizdo medžiaga.

Į tinklą įkeliamos nuotraukos paprastai būna spalvotos, todėl jos prieš analizę yra konvertuojamos į nespaltvotas. Taip pat nuotrauka yra sumažinama iki 400 px. Plotio, jeigu įkėlimo metu jos plotis buvo didesnis. Tokiu būdu didinamas veidų paieškos greitis.

Į serverį nuotrauka įkeliama prieš jos paruošimą veidų atpažinimui. Nuotraukoms sukuriamas unikalūs pavadinimas, panaudojus įkėlimo datą ir laiką, tam, kad serveryje nuotraukos nesidubliuotų. Tokiu būdu nuotraukai suteikiamas prasmingas vardas. Tuo tarpu į serverį keliamoms veidų nuotraukoms sukuriamas failo vardas pasinaudojus GUID kodo generatoriumi, nes šių nuotraukų failo pavadinimai taip pat turi būti unikalūs, tačiau jiems nereikalinga prasmė, nes kiekvienai veido nuotraukai suteikiamas vardas kuris gali kartotis jeigu priklauso tam pačiam savininkui.

Programavimas vykdomas C# programavimo kalba. Todėl internetinio tinklapio kūrimui pasitelkiama ASP.NET Razor programinė sintaksė.

2. SISTEMOS ARCHITEKTŪRA

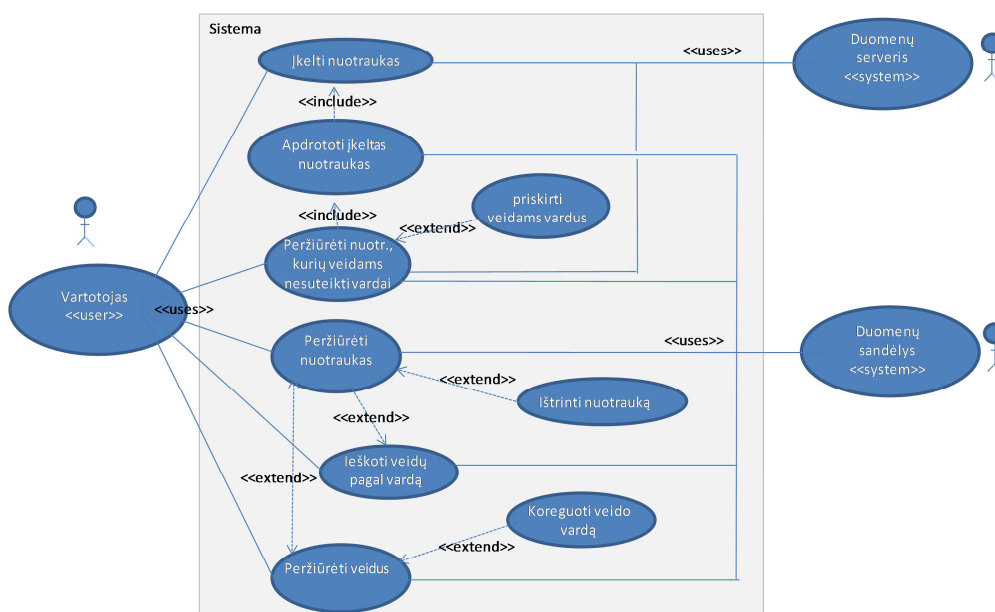
2.1. Funkciniai reikalavimai

- Nuotraukų aplanko įvedimas naudojant grafinę vartotojo sąsają.
- Veido vardo įvedimas naudojant grafinę vartotojo sąsają.
- Nuotraukų paieška naudojant grafinę vartotojo sąsają.
- Interaktyvi pagalbos sistema.

2.2. Nefunkciniai reikalavimai

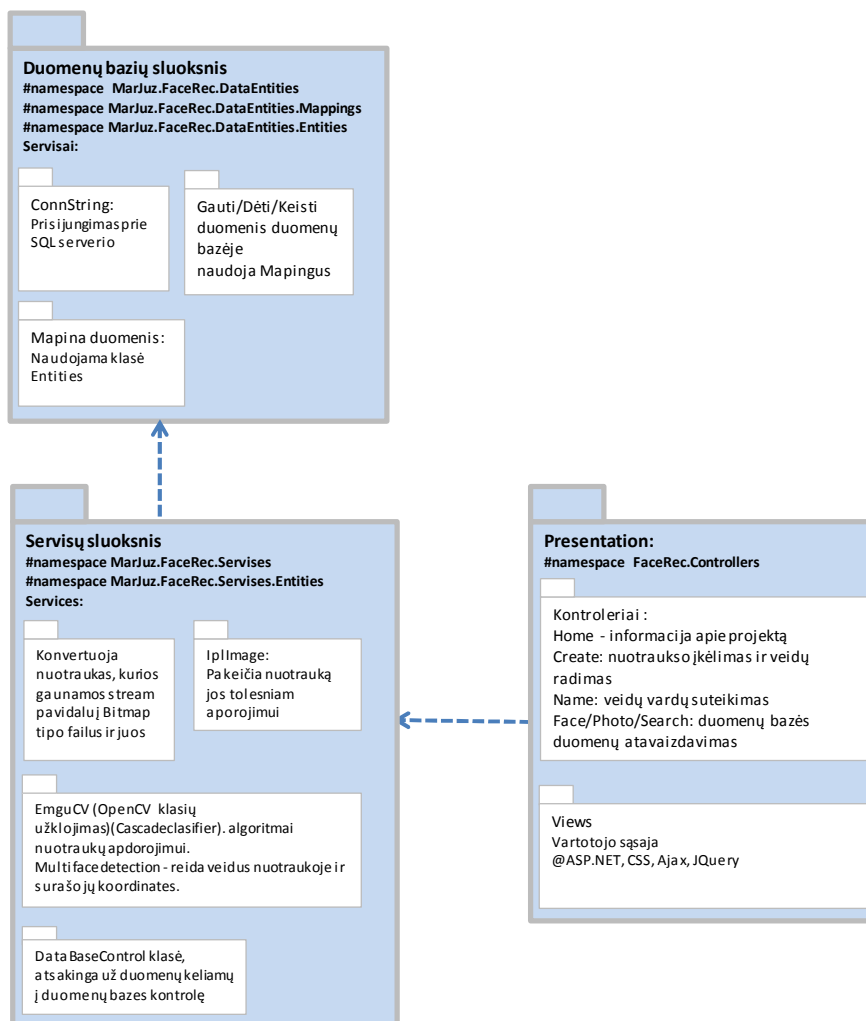
- 3.6. PĮ turi būti realizuota grafinė vartotojo sąsaja pradiniais duomenimis įvesti.
- 3.7. Vartotojo sąsaja lietuvių kalba.
- 3.8. Vartotojo sąsaja turi būti paprasta, intuityvi ir funkcionali.
- 3.9. Programos reakcijos laikas turi būti priimtinas vartotojui. Jeigu reikalingas ilgesnis laikas atsakui pateikti (sudėtingi skaičiavimai), apie tai turi būti pranešama vartotojui ir pagal galimybę prognozuojamas atsakymo pateikimo laikas.

2.3. Panaudos atvejų diagrama



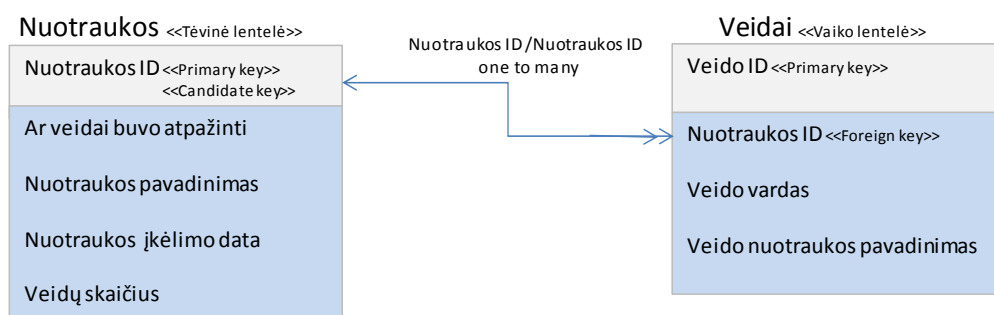
2.3.1. pav. Panaudos atvejų diagrama

2.4. Paketų diagrama

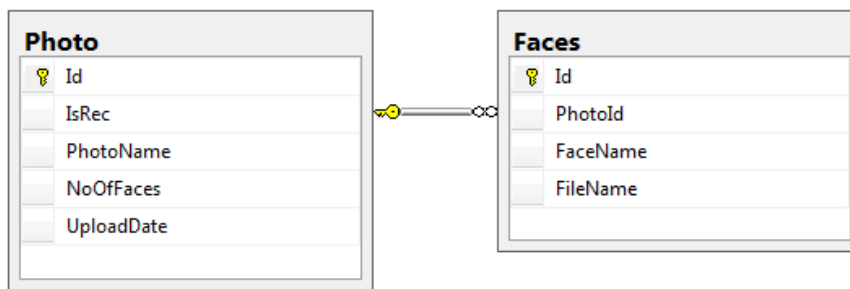


2.4.1. pav. Paketų diagrama

2.5. Duomenų bazės diagrama



2.5.1. pav. Detali duomenų bazės diagrama



2.5.2. pav. MSSQL Server suformuota duomenų bazės diagrama

Duomenų bazę sudaro dvi lentelės: Photo (Nuotraukos) ir Faces (Veidai). Nuotraukų lentelė yra tėvinė, o veidų lentelė vaiko. Tai yra tarp lentelių suformuotas ryšis vienas su daug. Viena nuotrauka gali turėti daug jai priskirtų veidų.

Nuotraukų lentelėje yra penki atributai:

- Id (Nuotraukos ID), kuris generuojamas automatiškai ir yra unikalus kiekvienai nuotraukai. Šis atributas yra ir pirminis lentelės raktas. Lentelės pirminis raktas (angl. primary key) yra stulpelis (ar kelių stulpelių derinys), kurį žinodami galime vienareikšmiškai pasakyti, iš kurios eilutės yra paimtas konkretus/unikalus atributų derinys [1];
- IsRec (Užregistruota), kuris gali turėti dvi reikšmes 0 arba 1. Sukūrus pirminį nuotraukos įrašą automatiškai suteikiama 0 reikšmė. Reikšmė pasikeičia į 1 tik tada, kai nuotraukai priskirtiems veidams suteikiami vardai;
- Nuotraukos pavadinimas yra sukuriamas nuotraukos įkėlimo metu ir jis nekinta. Vardas naudojamas nuotraukos radimui norint ją atvaizduoti, nes jo pagalba formuojama nuoroda į nuotrauką;
- NoOfFaces (Veidų skaičius). Nurodo kiek veidų yra priskirtų nuotraukai. Esminės naudos laukas neturi;
- UploadDate (įkėlimo data). Nurodo kada nuotrauka buvo įkelta. Gali turėti reikšmę jeigu būtų sukurtas filtras pagal kurį iškviečiamos nuotraukos įkeltos konkrečių laikotarpiu. Tai galėtų palengvinti nuotraukų paiešką.

Veidų lentelėje yra keturi atributai:

- Id (Veido identifikavimo numeris) kuris generuojamas automatiškai ir yra unikalus kiekvienam veidui. Šis atributas yra ir pirminis lentelės raktas.
- PhotoId (Susijusios nuotraukos identifikavimo numeris). Tai yra atributas, kurio pagalba galima nustatyti su kuriuo atributu iš tėvinės lentelės yra susijęs konkretus/unikalus atributų derinys. Šis laukelis yra ir svetimasis raktas. Stulpelis ar stulpelių derinys, kuri yra kitos lentelės pirminis raktas, vadinamas svetimuoju raktu (angl. foreign key)[1];

- **PhotoName** (nuotraukos vardas). Atributas, kuriami įrašomas veidui priskirto vardo reikšmė. Šios reikšmės pagalba nuotraukas, kuriose yra konkretus asmuo sugrupuoti į vieną grupę ir esant reikalui pakelti per paiešką;
- **FileName** (Failo vardas). Šiam atributui reikšmės suteikiamos automatiškai, kiekviena reikšmė yra unikali, nes naudojamais funkcija Guid. Šiame attribute pateikti įrašai naudojami veidų paveikslukų adresams formuoti, kad būtų galima juos atvaizduoti vartotojo sąsajoje.

2.6. Klasių diagrama

Kuriant įrankį panaudotas trylika klasių 2.6.1 pav. Penkios klasės duomenų bazės sluoksnyje, kuris atsakingas už informacijos saugojimą ir judėjimą duomenų bazėse. Dvi klasės servisų sluoksnyje, kuris atsakingas už visus atliekamus skaičiavimus, darbą su nuotraukomis ir darbą su informacijos paėmimu ir išsiuntimu į duomenų bazių sluoksnį. Šešios klasės panaudotos ir prezentacijos sluoksnyje (kontroleriai).

Duomenų bazės sluoksnis:

Dvi klasės atsakingos už atributų naudojamų duomenų bazės lentelėse sukūrimą.

Dvi klasės atsakingos už duomenų bazės lentelėse kuriamų reikšmių sukūrimą (Mapping). Šiose klasėse sukuriama lentelių viena nuo kitos priklausomybių ryšiai. Vienas su daug.

Pagrindinė viena klasė duomenų bazės sluoksnyje atsakinga už ryšio su duomenų bazės aptarnaujančiu serveriu užmezgimą, duomenų į ir iš duomenų bazės lentelių kontrolę. Šioje klasėje parašytos visos SQL užklauskos LINQ sintakse. Ši klasė yra Generic tipo, kas leidžia naudojant tas pačias funkcijas dirbti su skirtingais objektais.

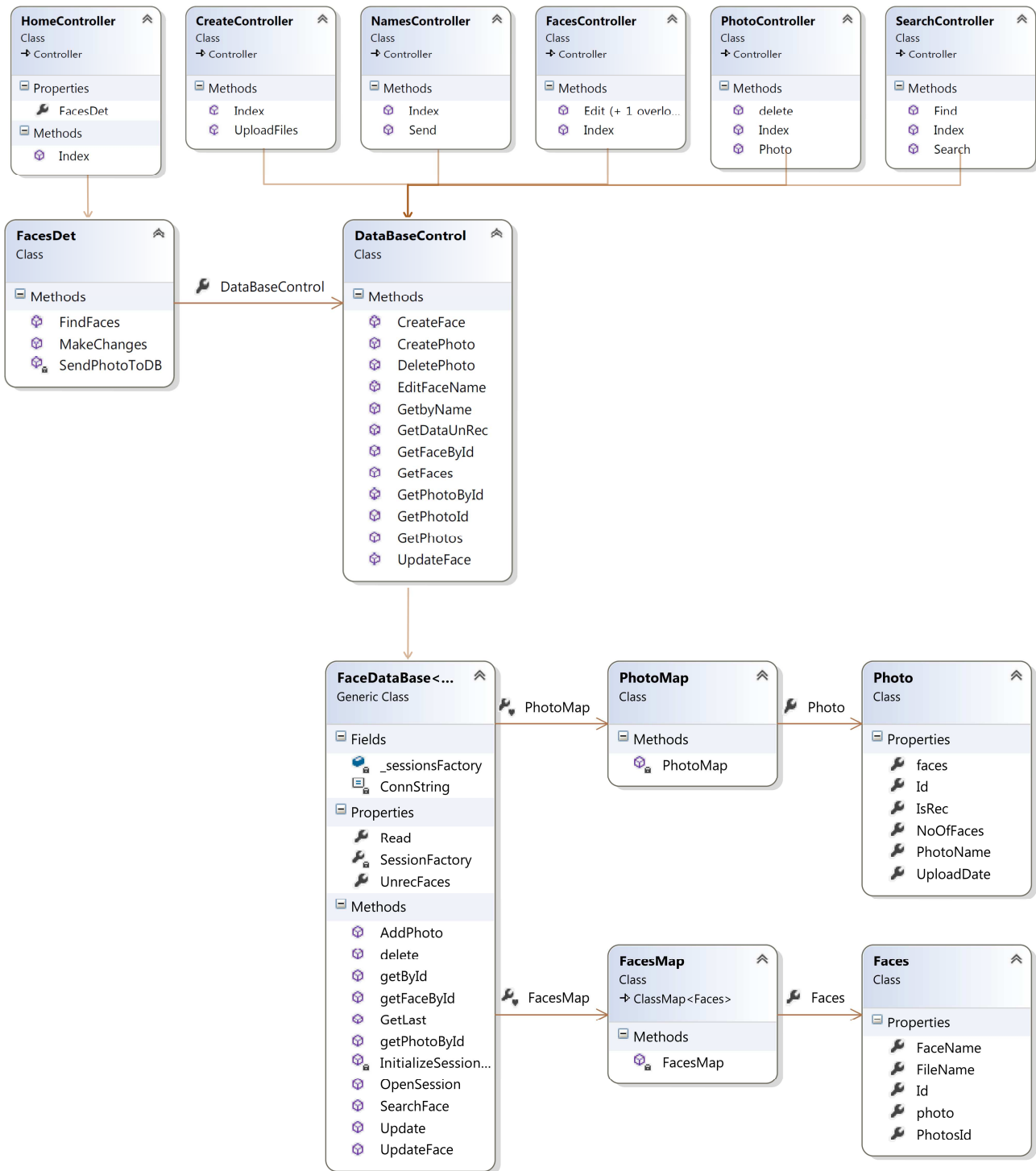
Servisų sluoksnis:

Servisų sluoksnį sudaro dvi klasės. Vienoje vykdomi visi veiksmai su nuotraukomis: nuotraukų įrašymas į serverį, nuotraukų konvertavimas į tinkamus formatus. Algoritmų nuotraukoms apdoroti įkėlimą ir panaudojimą.

Kita klasė naudojama duomenų bazės sluoksnio valdymui. Čia formuojami objektai, kurie siunčiami į duomenų bazes, čia jie ir sukeliama iš duomenų bazių. Tai tarsi tarpinė stotelė tarp duomenų bazės ir vartotojo sąsajos.

Prezentacijos sluoksnis:

Šiame sluoksnyje esančios klasės tai kontroleriai, kurie valdo duomenų judėjimą tarp servisų klasės ir vartotojo sąsajos.



2.6.1. pav. UML Klasių diagrama

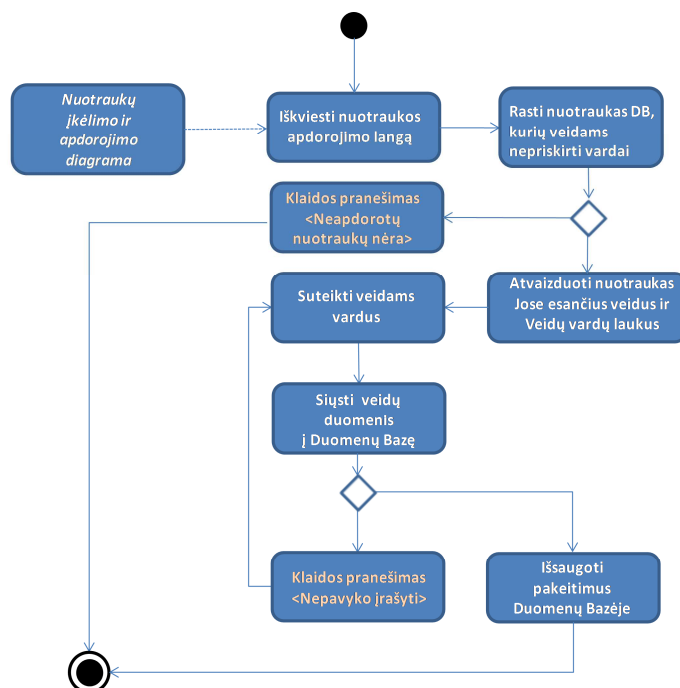
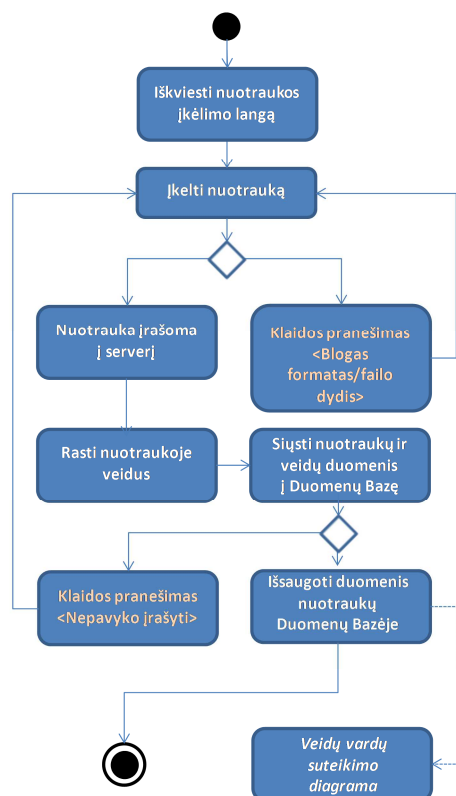
2.7. Veiklos diagrama

Pateikiamos penkios veiklos diagramos. Veiklos diagramos sudarytos priklausomai nuo funkcijų, kurias gali atlikti vartotojas atsidaręs internetinį puslapį.

- Vartotojas gali įkelti nuotrauką į nurodytą aplanką. Nuotraukos įkėlimo metu nuotrauka/os yra apdorojama/os ir jeigu ji/jos atitinka keliamus reikalavimus randami veidai ir visa reikiama

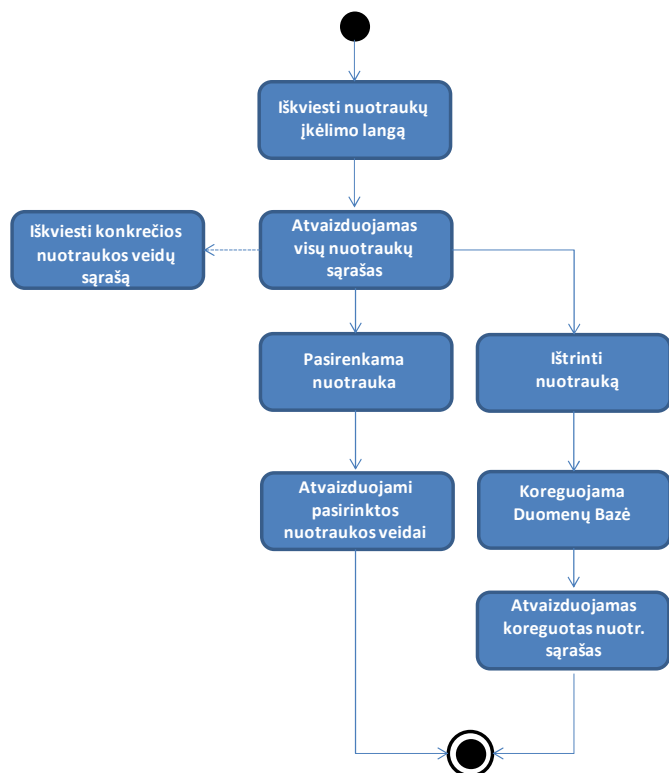
informacija sukeliami į duomenų bazes, kad vėliau būtų galima duomenis atvaizduoti, koreguoti ar ištrinti. 2.7.1. pav

- Antra veiklos diagrama susijusi su rastų veidų vardų suteikimu. Jeigu nuotraukos nėra įkeltos į nurodytą aplanką veidų vardų suteikti nebus galimybės. Todėl ši veiklos diagrama yra priklausoma nuo ankstesniosios. Atsidarius veidų vardų langą vartotojas gali nuotraukose rastiems veidams suteikti vardus ir pakoreguoti duomenų bazę. Vėliau šiame lange koreguoti veidų vardų nebebus galima. 2.7.2 pav

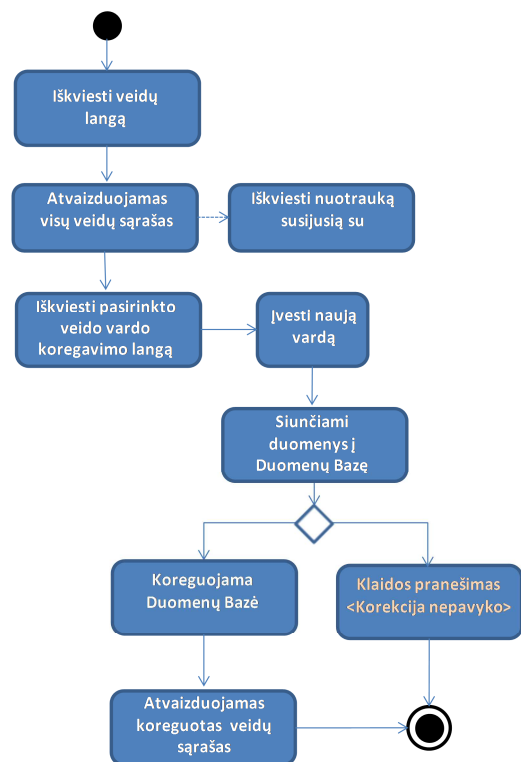


2.7.1 pav. Nuotraukos įkėlimo ir apdorojimo veiklos diagrama 2.7.2. pav. Vardų suteikimo veidams veiklos diagrama

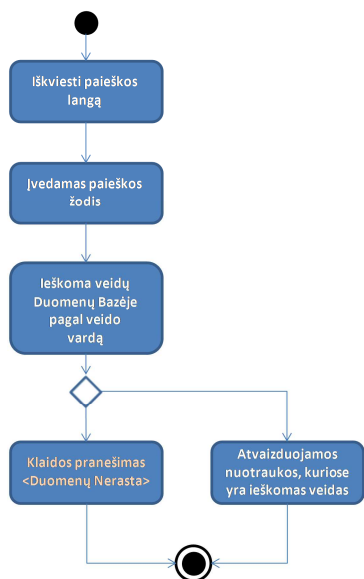
- Trečia veiklos diagrama vaizduoja veiksmus, kurie atliekami pasirinkus nuorodą į nuotraukų sąrašą. Nuotraukas galima peržiūrėti ir jas esant reikalui ištrinti. Kartu su nuotraukomis pašalinami ir su nuotrauka susiję veidai. Iš nuotraukų sąrašo galima iškviešti šiai nuotraukai priskirtų veidų sąrašą. 2.7.3. pav.
- Ketvirta veiklos diagrama vaizduoja veiksmus, kurie atliekami pasirinkus nuorodą į veidų sąrašą. Visus veidus galima peržiūrėti ir esant reikalui galima koreguoti veidų vardus. Iš veidų sąrašo galima iškviešti su veidu susijusią nuotrauką 2.7.4. pav. ;
- Penkta veiklos diagrama vaizduoja paieškos pagal veidus veikimą. Įvedus į paieškos lauką norimą veido vardą randamos visos nuotraukos susijusios su nurodytu vardu. 2.7.5. pav..



2.7.3. pav. Nuotraukų sąrašo veiklos diagrama



2.7.4. pav. Veidų sąrašo veiklos diagrama



2.7.5. pav. Paieškos veiklos diagrama

2.8. Vartotojo sąsajos projektas

Veidų atpažinimo programa							
Header							
Menu	Pradžia	Įkelti nuotrauką	Veidų vardai	Nuotraukos	Veidai	Pagalba	
						Paieškos laukas	Ieškoti
Pateikiama užsakovo užduoties lapas. Užduoties lapas išsaugomas PDF formatu ir atvaizduojamas pasinaudojus google plėtiniu.							
Praktikos darbas 2014							
Footer							

2.8.1. pav. Pradinio lango eskizas

Veidų atpažinimo programa							
Header							
Menu	Pradžia	Įkelti nuotrauką	Veidų vardai	Nuotraukos	Veidai	Pagalba	
						Paieškos laukas	Ieškoti
Drag and drop laukas, kuris realizuojamas JQuery ir Ajax skriptų pagalba							
Pateikiami įkeltų nuotraukų pavadinimai							
Praktikos darbas 2014							
Footer							

2.8.2.pav. Įkelti nuotraukas lango eskizas

Veidų atpažinimo programa							
Header							
Menu	Pradžia	Įkelti nuotrauką	Veidų vardai	Nuotraukos	Veidai	Pagalba	
						Paieškos laukas	Ieškoti
Nuotrauka	Veidas1	<input type="text"/>					
	Veidas2	<input type="text"/> <input type="button" value="Įkelti"/>					
	Veidas3	<input type="text"/>					
Pateikiami įkeltų nuotraukų pavadinimai							
Praktikos darbas 2014							
Footer							

2.8.3. pav. Veidų vardų suteikimo lango eskizas

Veidų atpažinimo programa							
Header							
Meniu	Pradžia	Įkelti nuotrauką	Veidų vardai	Nuotraukos	Veidai	Pagalba	
						Pateškos laukas	Įeškoti
Nuotraukos pav.	Nuotrauka			Veidų sk. <link Face>			
Nuotraukos pav.	Nuotrauka			Veidų sk. <link Face>			
Nuotraukos pav.	Nuotrauka			Veidų sk. <link Face>			
Praktikos darbas 2014							
Footer							

2.8.4. pav. Nuotraukų sąrašo lango eskizas

Veidų atpažinimo programa						
Header						
Meniu	Pradžią	Įkelti nuotrauką	Veidų vardai	Nuotraukos	Veidai	Pagalba
						Paieškos laukas
						Įeškoti
<div><div>Veidas</div><div>Vardas <link Photo></div><div>Veidas</div><div>Vardas <link Photo></div><div>Veidas</div><div>Vardas <link Photo></div></div>						
<div>Praktikos darbas 2014</div> <div>Footer</div>						

2.8.5. pav. Veidų sąrašo lango eskizas

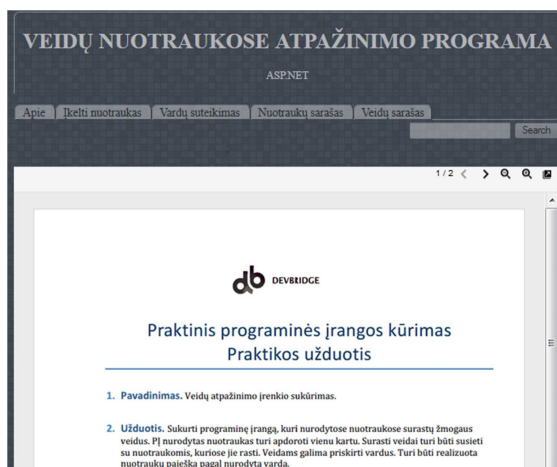
3. SISTEMOS REALIZACIJA IR TESTAVIMAS

Sistemos realizavimas ir testavimas atliktas. Internetinis tinklapis buvo užkrautas lokaliame IIS serveryje MS Windows 7 operacinėje sistemoje.

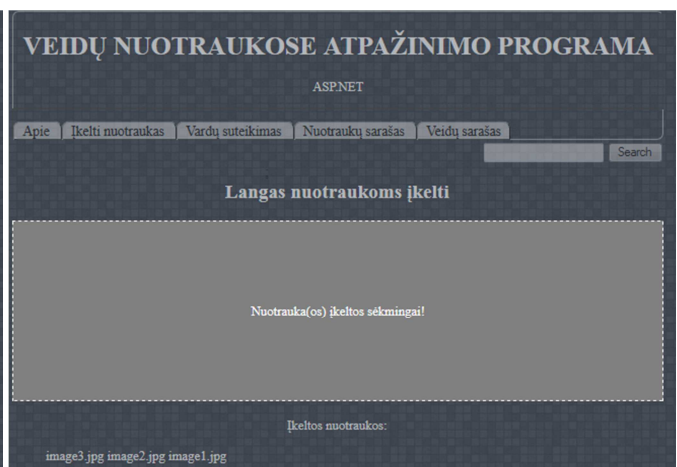
Veidų atpažinimo nuotraukose įrankis buvo sėkmingai išbandytas. Nuotraukose buvo atpažinti įkelti veidai ir sėkmingai sukelti į duomenų bazės lenteles. Vėliau duomenys buvo sėkmingai iš duomenų bazės paimti ir panaudojus gautus rezultatus atvaizduoti tinklapyje. Sudarytas testavimo planas buvo sėkmingai įvykdytas.

Pavyko nustatyti programos stipriąsias ir silpnąsias puses.

3.1. Vartotojo sąsajos realizacija



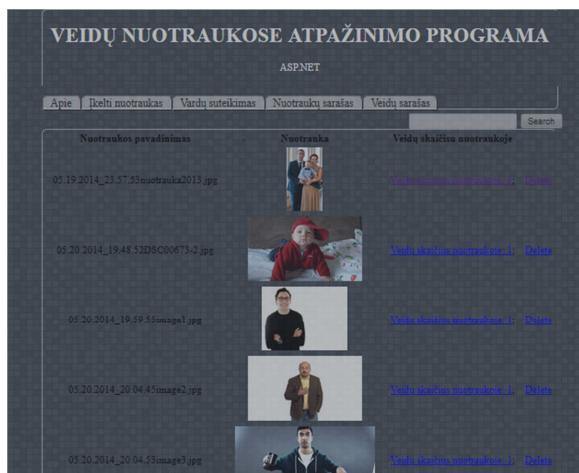
3.1.1 pav. Pradinis langas



3.1.2. pav. Nuotraukų įkėlimo langas



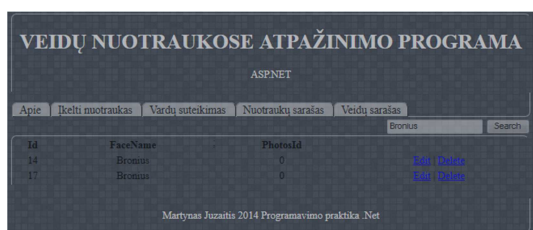
3.1.3-3.1.4. pav. Vardų suteikimo langai



3.1.5. pav. Nuotraukų peržiūros langas



3.1.6. pav. Veidų peržiūros langas



3.1.7. pav. paieškos rezultatų langas

Penkiuose paveiksluose pateiktas visas kelias, kuris buvo realizuojamas testuojant sukurta įrankį. Pradžioje buvo įkeltos trys nuotraukos, vėliau jose rastiems veidams suteikti vardai. Galiausiai peržiūrėtas nuotraukų ir veidų sąrašai.

3.2. Sistemos realizacija

Programos veikimui būtinos bibliotekos:

- duomenų bazės valdymui panaudotas atvirojo kodo karkasas Fluent NHibernate;
- SQL užklausų rašymui panaudotas LINQ (Language Integrated Query), kuris yra Microsoft .NET Framework komponentas.
- EmguCV platforma yra OpenCV vaizdo apdorojimo bibliotekos plėtinys parašytas C# programavimo kalba. OpenCV biblioteka yra sukurta C++ kalba.
- System.Drawing klasių biblioteka suteikia prieigą prie GDI+ bazinių grafikos funkcijų;
- Drag & drop failų įkėlimui panaudotas JQuery.filedrop skriptas;
- Vartotojo sąsajai kurti panaudotas ASP.NET Razor programinė sintaksė;

Sistemos stipriosios pusės:

- Įrankiu paprasta naudotis;
- Patogi failų įkėlimo sistema, nors ir nėra realizuotas atskiras failų įkėlimo mygtukas per explorerį;
- Visi veiksmai su informacijos apdorojimu atliekami pastebimai greitai;
- Yra paieškos pagal veidus galimybė, kuri padeda rasti norimas nuotraukas greičiau, esant ženkliam nuotraukų kiekiui;
- Nuotraukose rastus veidus galima suvesti ne iš karto ir esant būtinybei galima vardus lengvai koreguoti

Sistemos silpnosios pusės:

- Nėra realizuota atskirų vartotojų prisijungimo sistema;
- Įkėlus nuotrauką matosi tik jų pavadinimai, o pačią nuotrauką galima išvysti tik kitame lange;
- Vienu pelės paspaudimu galima į duomenų bazę suvesti tik vienos nuotraukos veidų vardus;
- Jeigu nuotraukoje žmogus stovi ženkliai pasukęs galvą jo veido rasti nepavyks;
- Įrankis kartais klaidingai interpretuoja žmogaus veidą jeigu nuotraukoje yra detalių, kurios primena žmogaus akis. Pavyzdžiui tamsūs karoliai ant kaklo klaidina paieškos sistemą.

3.3. Sistemos testavimas

Planas:

- Patikrinti ar veikia visos nuorodos;
- Patikrinti ar galima įkelti vienu metu kelias nuotraukas.
- Patikrinama ar pakeičiami vardai juos suvedus patvirtinus „Veidų vardai“ lange;
- Patikrinama ar veikia paieška;

Testavimo duomenų rinkiniai:

- Vienu kėlimu įkeliamos trys nuotraukos ir patikrinama ar visi duomenys sėkmingai buvo sukelti į duomenų bazę. Rezultatas pateiktas Vartotojų sąsajos realizacijoje. Parenkamos nuotraukos, kurių dydis mažesnis kaip 1 megabaitas, nes toks nustatytas įkeliamų nuotraukų dydžio limitas. (Vykiant veidų paiešką nuotraukos dar kartą mažinamos todėl nėra tikslo įkelti didelės apimties nuotraukų

saugant vietą serveryje);



3.3.1. pav. nuotraukų pavyzdžiai.

- Kiekvienai nuotraukai vardai suteikiami atskirai. Jeigu nuotraukoje daugiau kaip vienas veidas, tai vienu kartu visiems jiems ir suteikiami vardai.
- Atlikus paiešką, pagal raktinį žodį Bronius gauti du rezultatai, nes Broniaus nuotrauka buvo sukelta du kartus.

Testavimas atliktas Windows 7 operacinės sistemos aplinkoje. Naršyklė, kurioje peržiūrima vartotojo sąsaja Mozilla Firefox 28.0. Duomenims duomenų bazėje saugoti panaudotas www.amazon.com duomenų sandėlis RDS. Duomenų bazių valdymui panaudota MSSQL server 2008 duomenų bazių valdymo sistema.

4. DOKUMENTACIJA NAUDOTOJUI

4.1. Sistemos diegimo vadovas

4.1.1. Programinio paketo parsisiuntimas.

Norinti įsidiegti programą reikalingas programos failų paketas. Programos failų paketą galima atsisiųsti iš GitHub pagal nuorodą <https://github.com/marjuz/Veidai>. Parsisiuntus failų paketą jį reikia išskleisti laisvai pasirinktame kataloge;

4.1.2. ISS serverio paruošimas tinklapio talpinimui.

Kiekviename kompiuteryje su Windows operacine sistema IIS yra instaliuotas mums reikės jį aktyvuoti. Nagrinėsime veiksmus Windows 7 operacinėje sistemoje.

1 žingsnis. IIS serverio aktyvavimas. ISS aktyvuojama „Windows features“ lange, kuris paleidžiamas per Start->Search paieškos lauke įvedus Windows features, pasirenkama nuoroda „Turn Windows features on or off“. Visas kelias Start->Control panel->Programs-> Turn Windows features on or off. [3]

2 žingsnis. Windows features lange aktyvuojame Internet Information Services (IIS) . Išskleidus ISS aktyvuojame „FTP Server“ ir jame esančius „FTP extensibility“ ir „FTP Service“.

Aktyvuojame „Web Management Tools“, jame esančius „IIS 6 Management Tools“ (visas funkcijas jame), „IIS Management Console“, „ISS Management Scripts and Tools“ ir „IIS Management Service“.

Aktyvuojame „World Wide Web Services“, jame esančius „Application Development Features“ kartu su jame esančiais „NET Extensibilities“, „ASP .NET“, „ISAPI Extensions“, „ASAPI Filters“ ir „Server-Side Includes“. Aktyvuojame „Common HTTP Features“, „Health and Dieagnostics“, „Performance features“, „Security“ ir visus laukus juose.

Aktyvuojame „Internet Information Services Hostable Web Core“.

Spaudžiame OK. ISS bus aktyvuotas. [3]

3 žingsnis. Paruošime ISS serverį tinklapio talpinimui. Einame Start->Search ir į paiešką įvedame „Inetmgr“. Atidarome IIS Manager. [4]

4 žingsnis. ISS Maneger programoje Connections dalyje sukursime vietą tinklapio talpinimui. Išskleidus esanti serverį spaudžiame dešiniu pelės klavišu ant „Sites“ ir pasirenkame „Add Web Site“.[4]

5 žingsnis. Atsidariusiame „Add Web site“ lange, esančiame lauke „Site name“ nurodome savo tinklapio pavadinimą, lauke „Physical path:“ nurodome katalogą, kuriame bus talpinamas tinklapis. „Bindings“ sekcijoje pakeičiame „Port“ į norimą reikšmę pvz. 8080, kuri pagal nutylėjimą yra 80.[4]

6 žingsnis. „Connections“ sekcijoje pasirenkame „Application pools“ nuorodą ir atsivėrusiame lange surandame mūsų sukurtą tinklapį. Spaudžiame dešinio pelės klavišu ant jo ir pasirenkame „Advanced Settings“. Sekcijoje „General“ pakeičiame „.NET framework versijon“ reikšmę į „v4.0“. Spaudžiame OK. [4];

7 žingsnis. „Connections“ sekcijoje pasirenkame mūsų sukurtą tinklapį ir iš kairėje atsivėrusio lango pasirenkame „Directory browsing“ ir aktyvuojame jį. [4]

IIS serverio konfigūracija baigta.

4.1.3. Puslapio sukėlimas į IIS serverį.

1. žingsnis. Išskeisti parsisiųstą programinį paketą ir paleisti paleidžiamąjį failą MarJuz.FaceDetection.sln.

2 žingsnis. Esant poreikiui atnaujinti Fluent Nhibernate biblioteką duomenų bazių sluoksnyje „Data“. Patikriname ar References dalyje Fluent Nhibernate nėra su šauktuku, jeigu yra turime iš naujo įkelti jį (pasinaudojant NuGet paketus). Atnaujinti EmguCV biblioteką servisų sluoksnyje „Business“. Patikriname ar References dalyje Emgu.CV, Emgu. CV.UI ir Emgu.util nėra su šauktuku, jeigu yra turime iš naujo įkelti juos (Reikiamus failus rasime paketo kataloge „EMGU“).

3 žingsnis. Duomenų bazės sluoksnyje esančioje klasėje FaceDataBase.cs sukonfigūruoti MSSQL serverio prisijungimo duomenis ConnString. Reikia nurodyti serverį, vartotojo vardą, slaptažodį ir kitus duomenis reikalingus prisijungimui prie serverio. Pagal nutylėjimą sukurta prieiga prie www.amazon.com sukurtos RDS duomenų bazės.

4 žingsnis. Puslapio talpinimas į IIS serveryje sukurtą katalogą tinklapio talpinimui (4.1.2. dalyje).

Atidaroma prezentacijos klasė ir įvykdoma funkcija iš meniu Build->Publish FaceDet. Nurodyti IIS serveryje sukurtą katalogą tinklapio talpinimui (4.1.2. dalyje).

„Profile“ sekcijoje „Select or import a publish profile“ išskleidžiamo sąrašą ir pasirenkame <New profile...>. Atsidariusiame lange nurodome norimą profilio pavadinimą.

„Connections“ sekcijoje pasirenkame „Publish method:“ „File system“, o „Target location“ nurodome katalogą, kuriame talpinsime tinklapį (sukurtą 4.1.2. dalyje).

Spaudžiame mygtuką „Publish“

5 žingsnis. Tinklapis paleidžiamas per adresą <http://localhost/8080>. Jeigu tinklapis nepasileidžia, gali būti problema, kad nėra aktyvuotas IIS serveris.

6 žingsnis IIS serverio aktyvavimas. Nueiname į katalogą C:\Windows\Microsoft.NET\Framework64\ ir susirandame kokai yra .NET Framework v.4.0 versija pvz. v4.0.30319. Tada Windows run aplikacijoje (Start->Search įvedame Run) paleidžiama aktyvavimo komanda:

```
%windir%\Microsoft.NET\Framework64\v4.0.30319\aspnet_regiis.exe -i.
```

Šiuos veiksmus reikia atlikti per kompiuterio vartotoją turintį administratoriaus teises.

4.1.4. Reikalavimai programinei įrangai:

- MS Visual Studio 2012;

- MSSQL Server 2008;
- IIS serveris;
- Atnaujinta interneto tinklapių naršyklė;

4.2. Sistemos vartotojo vadovas

Paleidus tinklapį matome, kad viršutinėje jo dalyje yra valdymo meniu.

Apie meniu punktas. Pateikta užduotis pdf formatu.

Įkelti nuotraukas meniu punktas. Vartotojas gali įkelti nuotraukas jas temdamas į tam skirtą kvadratą. Vienu metu gali būti užkrautos ne daugiau kaip 5 nuotraukos, kurių dydis neviršija 1 megabaito.

Vardų suteikimas meniu punktas. Vartotojui pateikiamos viso sukulto nuotraukos, kurių veidams nebuvo priskirti vardai. Vartotojas vienu metu gali priskirti vardus vienai nuotraukai. Paspaudus mygtuką „Įkelti“ nuotraukos veidų vardai įkeliami į duomenų bazę.

Nuotraukų sąrašas meniu punktas. Pateikiamos visos į serverį užkrautos nuotraukos. Čia galimi du veiksmai pirmas paspaudus ant nuorodos, su nurodytu veidų skaičiumi pamatyti visus nuotraukoje esančius veidus. Arba pašalinti nuotraukas iš serverio.

Veidų sąrašas meniu punktas. Vartotojui pateikiamos visos veidų nuotraukos sus vardais. Vartotojas gali keisti vardų reikšmes paspaudęs nuorodą „Koreguoti vardą“. Taip pat vartotojas gali iškviesti nuotrauką, kuri susijusi su konkrečiu veidu. Paspaudęs ant veido paveikslėlio.

Paieškos laukas yra aktyvus visuose languose. Paieškos lauke įvedus vardą ir paspaudus ieškoti, vartotojui pateikiami visi įrašai apie nuotraukas susiję su tokiu vardu.

5. IŠVADOS IR REZULTATŲ APIBENDRINIMAS

5.1. Rezultatai

Darbo metu pavyko susipažinti su C# programavimo kalbos pradmenimis. Įsigilinus pavyko suprasti NHibernate karkaso veikimo principus ir pagilinti žinias apie SQL užklausų rašymą.

Atliekant pavyko sukurti įrankį, kuris duotose nuotraukose suranda žmonių veidus, o panaudojus papildomas bibliotekas pavyko nuotraukų duomenis apdoroti, sukelti į duomenų bazę ir pasinaudojus duomenų bazėmis vėliau rezultatus atvaizduoti interneto naršyklėse.

5.2. Išvados

- 1) Dinaminis interneto tinklapis veikia visose pagrindinėse naršyklėse;
- 2) Interneto tinklapiu naudotis patogiu, nes vartotojui nereikia atlikti sudėtingų veiksmų, kad būtų pasiektas rezultatas. Įrankio reakcijos laikas turėtų būti priimtinas, nes nėra dirbama su didelės apimties duomenimis. Nuotraukas sėkmingai pavyksta importuoti į serverį o vėliau jas išsikleti, peržiūrai;
- 3) Už vartotoją nuotraukoje esančių veidų ieško kompiuteris pasinaudodamas OpenCV bibliotekomis, kurias C# kalboje praplečia EmguCV bibliotekos.
- 4) Nuotraukų peržiūrai sukurti du įrankiai. Vienas pateikia visas nuotraukas, o kitas visus veidus;
- 5) Vartotojas gali rasti nuotraukas pasinaudodamas paieškos lauku, kuris interneto tinklapyje yra nuolat matomas. Nuotraukose atpažintų veidų vardus vartotojas gali nesunkiai koreguoti;

6. LITERATŪRA

1. Internetinis tinklapis <http://lt.wikipedia.org/>;
2. Internetinis tinklapis <http://www.emgu.com/>;
3. Internetinis tinklapis <https://www.youtube.com/watch?v=dZAbdmPrU4g>;
4. Internetinis tinklapis <http://www.brainbrushups.com/2013/05/how-to-deploy-or-publish-aspnet-website.html>;
5. Internetinis tinklapis http://www.cognotics.com/opencv/servo_2007_series/part_2/sidebar.html;
6. Internetinis tinklapis <http://www.codesizzle.com/face-detection/>;
7. Internetinis tinklapis <http://www.codeproject.com/Articles/239849/Multiple-face-detection-and-recognition-in-real-ti>;
8. Internetinis tinklapis <http://stackoverflow.com/questions/2472897/fluent-nhibernate-many-to-one-mapping>;
9. Internetinis tinklapis <http://www.asp.net/mvc/tutorials/mvc-5/introduction/adding-search>;
10. Internetinis tinklapis <http://www.essentialobjects.com/Products/EOWeb/Menu.aspx?gclid=CPe22MDXub0CFerpwgod56sANw>;
11. Internetinis tinklapis <http://www.w3schools.com/css/DEFAULT.asp>.