



**KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
INFORMACIJOS SISTEMŲ KATEDRA**

Arvidas Naujokas

Studijų centro akademinė informacijos sistema

Magistro darbas

Darbo vadovas
prof. S. Gudas

Kaunas
2006

TURINYS

1. ĮVADAS.....	4
2. ANALITINĖ DALIS.....	6
2.1. Duomenų bazių valdymo programos.....	7
2.2. Veiklos analizės modeliavimas.....	10
2.2.1. Informacijos sistemų kūrimo modeliai.....	11
2.2.2. Duomenų organizavimo modeliai.....	12
2.3. Reikalavimai sistemai.....	13
2.4. Reikalavimai programinei įrangai.....	13
2.5. Aparatiniai ir sisteminiai resursai bei naudojamos technologijos.....	14
2.6. Pradiniai duomenys ir rezultatų išvedimas.....	14
2.7. Galimos modifikacijos ateityje.....	15
2.8. Testavimas.....	15
2.9. Programinės įrangos paskirtis ir teikiamos paslaugos.....	15
3. PROJEKTINĖ DALIS.....	17
3.1. Reikalavimų specifikacija.....	17
3.1.1. Funkciniai reikalavimai.....	18
3.1.2. Nefunkciniai reikalavimai.....	20
3.2. Duomenų struktūra.....	24
3.3. Sistemos architektūra.....	26
3.3.1. Architektūros tikslai ir apribojimai.....	26
3.3.2. Panaudojimo atvejai.....	27
3.3.3. Sistemos statinis vaizdas.....	29
3.3.4. Sistemos dinaminis vaizdas.....	30
3.3.5. Sistemos išdėstymas.....	33
3.3.6. Duomenų vaizdas.....	34
3.4. Detali architektūra.....	35
3.4.1. Lentelės.....	35
3.4.2. Įvedimo formos.....	38
3.4.3. Užklausos.....	40
3.4.4. Ataskaitos.....	45
3.4.5. Vartotojo sąsaja.....	46
4. VARTOTOJO DOKUMENTACIJA.....	48
4.1. Sistemos funkcinis aprašymas.....	48
4.2. Sistemos vadovas.....	49
4.2.1. Duomenų peržiūra.....	50
4.2.2. Duomenų įvedimas ir koregavimas.....	50
4.2.3. Studijų, sesijos ir ataskaitų langai.....	52
4.3. Sistemos instaliavimas.....	53
4.3.1. Kliento – serverio architektūra.....	55
5. KOKYBĖS ĮVERTINIMAS.....	55
6. IŠVADOS.....	56
7. LITERATŪRA.....	57
8. TERMINŲ IR SANTRUMPŲ ŽODYNAS.....	58
9. SUMMARY.....	59

1. ĮVADAS

Informacija šiuolaikiniame pasaulyje tapo labai svarbiu veiklos ištekliu. Informacijai tvarkyti organizacijose kuriamos informacijos sistemos, kurios surenka, apdoroja, saugo ir paskirsto informaciją organizacijos nariams. Informacijos sistema veikia tam tikroje organizacinėje ir technologinėje aplinkoje.

Informacijos sistemų inžinerija kaip mokslo ir praktikos sritis susiformavo XX amžiaus paskutiniajame ketvirtyje. Tai sąlygojo vis tobulėjanti kompiuterinė technika: atsiradusi grafinė vartotojo sąsaja, kompiuterių tinklai, vis greitėjantys procesoriai ir vis talpesnė išorinė atmintis. Informacijos sistemų svarba ir reikšmė ypač išaugo pradėjus formuotis informacinei visuomenei. Dabar bet kokia funkcionuojanti organizacija sunkiai įsivaizduojama be kompiuterinės informacijos sistemos. Svarbiausi informacijos sistemų ir duomenų bazių plėtros veiksniai informacijos technologijų pasaulyje – vis augantis techninės ir programinės įrangos galingumas ir patikimumas, jos kainų mažėjimas.

Svarbiausias informacijos sistemos elementas – duomenų bazė. Pasaulyje gyvuoja daugybė priemonių duomenų bazėms kurti. Tai Microsoft produktai MS Access, Microsoft SQL, MySQL, PostgreSQL, Oracle ir kt. Šiame darbe pabandysiu apžvelgti šių priemonių privalumus bei trūkumus.

Šio magistrinio darbo tikslas – nedidelės mokymo įstaigos (studijų centro) akademinė informacijos sistema. Šioje srityje nėra specializuotų programinės įrangos sprendimų. Yra didelės apimties informacijos sistemų, kurios nėra visiškai tinkamos duotu atveju. Tiriamajame darbe bandysiu analizuoti egzistuojančias sistemas, jas vertinti. Apžvelgus esamus programinės įrangos sprendimus, yra suprojektuota ir realizuota specifikacijoje nusakytus poreikius ir reikalavimus atitinkanti informacinė sistema. Ši programinė įranga yra pritaikyta konkrečios įstaigos veiklai.

Antrojoje (analitinėje) šio darbo dalyje bandysiu apžvelgti ir palyginti pasaulyje naudojamas duomenų bazių kūrimo priemonės, o taip pat Lietuvos aukštosiose mokyklose sukurtas informacijos sistemas. Taip pat šioje dalyje pateikiama informacijos sistemų kūrimo metodų apžvalga.

Trečiojoje (projektinėje) dalyje pateikiama reikalavimų duotai informacijos sistemai specifikacija, duomenų struktūros vaizdai, panaudojimo atvejai, sistemos architektūra.

Ketvirtoji darbo dalis – vartotojo dokumentacija. Sistemos funkciniam aprašyme (4.1. skyrius) yra trumpai apžvelgiamos sistemos galimybės ir paskirtis. Sistemos vadove (4.2. skyrius) aprašytos visos sistemos funkcijos ir galimybės bei jų naudojimas. Sistemos instaliavimo (4.3.) skyrius skirtas sistemos administratoriui.

Darbo pabaigoje pateikiamos išvados ir literatūros sąrašas.

Analizės stadijoje atlikti tokie projektavimo uždaviniai:

- Atlikta esamos programinės įrangos analizė, nustatyti jos privalumai ir trūkumai.
- Išanalizuoti vartotojo reikalavimai bei nustatyti funkciniai reikalavimai.
- Sudarytas programinės įrangos architektūrinis modelis.
- Nustatytos galimos duomenų struktūros.
- Sudarytas projekto atlikimo grafikas.
- Nustatytos projekto rizikos ir jų išvengimo priemonės.
- Sukurta vartotojo sąsaja.
- Sudarytas sistemos testavimo planas.
- Parašyta vartotojo dokumentacija.

Darbe naudojau tokią programinę įrangą:

Operacinės sistemos:

- MS Windows 2000;
- MS Windows XP Professional.

Programinė įranga:

- MS Office XP Pro;
- MS Visual Basic 6.0;
- MS Visio 2000;
- Internet Explorer 6.0 .

Sistemai kurti naudojau MS Access programinės įrangos paketą. Access programinė įranga licencijuojama kartu su MS Office programų paketu, todėl sistemos kaštams programinės įrangos kaina neturės jokios įtakos, kas biudžetinei įstaigai yra svarbu.

2. ANALITINĖ DALIS

Daugelyje šalies aukštųjų mokyklų yra sukurtos informacijos sistemos. Tokios sistemos, kiek žinau, veikia Kauno technologijos universitete, Vilniaus dailės akademijoje, Klaipėdos universitete ir kitur. Lietuvos universitetuose veikiančios informacijos sistemos yra per daug sudėtingos ir tokiai mažai įstaigai netinka. Pavyzdžiui, KTU akademinėje informacijos sistemoje naudojamas trijų lygių vartotojų prisijungimas: studentams, darbuotojams ir fakultetų raštinėms. Studento prisijungimo vardas yra studento bilieto numeris, o slaptažodis – asmens kodas. Studentui pateikiama informacija: kiekvienų mokslo metų studijų planas, kur matyti semestras, modulio kodas ir pavadinimas, kreditų skaičius, dėstomoji kalba, forma, atestuotas ar ne, įvertinimas. Kitame lange pateikiamas trumpas modulio aprašymas. Yra galimybė planą atspausdinti. Aišku, gerai būtų turėti tokią galingą informacijos sistemą, bet sukurti tokią sistemą - ne vieno žmogaus darbas.

Lietuvos kolegijose padėtis kiek kitokia. Vieningos informacijos sistemos kolegijose dar nėra sukurta. Kiekviena įstaiga bando kažką daryti savo jėgomis. Pavyzdžiui, Alytaus kolegijoje sukurta studentų priėmimo programa, kurią norėčiau šiek tiek plačiau pristatyti. Tai yra informacijos sistemos kūrimo pirmas etapas, nes priimtų studentų duomenis Alytaus kolegija importuoja į studentų duomenų bazę. Priėmimo programa sukurta Delphi kalba ir naudoja Paradox duomenų bazę. Studentų duomenų bazėje saugomos visos studentų kortelės ir formuojami žiniaraščiai. Taip pat Alytaus kolegija daro studentų pažangumo apskaitos programą, kurią planuoja paleisti kitais mokslo metais. Be to, jie dar turi studentų mokėjimo už mokslą apskaitos programą, kurią naudoja buhalterija. Žodžiu, šie keli moduliai yra kolegijos informacijos sistemos kūrimo pradžia.

Kėdainių Jonušo Radvilos studijų centro padėtis kiek kitokia. Kėdainių Jonušo Radvilos studijų centras yra ne savarankiška įstaiga, o Kauno kolegijos padalinys, kuris rengia užsienio kalbų mokytojus ir socialinius pedagogus Lietuvos mokykloms. Studentai studijuoja tris užsienio kalbas: anglų, vokiečių ir prancūzų, o taip pat socialinės pedagogikos specialybę. Šiuo metu studijų centre mokosi apie 350 studentų, dirba apie 40 dėstytojų.

Mano darbo tikslas yra sukurti studijų centro akademinę informacijos sistemą, kuria galėtų naudotis įstaigos administracija, dėstytojai ir studentai. Ši sistema labiau reikalinga administracijos poreikiams tenkinti. Kadangi studijų centras yra Kauno kolegijos padalinys ir įstaigos veikla nėra savarankiška, o visos ūkinės ir buhalterinės funkcijos ir studentų priėmimas vykdomas centralizuotai, todėl projektuojama informacijos sistema apima tik akademinę įstaigos veiklos pusę. Studentų priėmimo programą dėl nurodytų priežasčių kurti

taip pat neverta. Informacijos sistemos duomenų bazėje kaupiami duomenys apie visas studijų programas, dėstomas disciplinas, studentus ir dėstytojus.

Darbe stengiausi kompiuterizuoti studijų centro akademinę veiklą. Visų pirma tai yra mokymo procesas. Duomenų bazė nuolat papildoma naujais įrašais, duomenis pagal tam tikrus kriterijus galima atrinkti bei rūšiuoti. Pasenę bei nereikalingi duomenys šalinami. Duomenų kaupimą bei visas kitas funkcijas atlieka katedrų sekretorės.

Taigi pirmoji šios informacijos sistemos dalis - duomenys apie visus studentus, pradedant nuo stojamojo balo. Jei studentas pašalinamas ar pats meta studijas, jo duomenys perkeliama į kitą lentelę. Baigus studijas duomenų bazėje registruojami visi absolventams išduodamų diplomų numeriai.

Kita informacijos sistemos dalis – studijų duomenys. Duomenų bazėje kaupiami visų studentų egzaminų įvertinimai, klasifikuojama pagal semestrus.

Trečioji informacijos sistemos dalis – visi dėstomi privalomieji ir pasirenkamieji dalykai (moduliai). Duomenų bazėje matomas visų disciplinų valandų ir kreditų skaičius.

Dar viena informacijos sistemos dalis – duomenys apie dėstytojus: bazėje kaupiami duomenys apie jų atestaciją, išsilavinimą, dėstomus dalykus (modulius). Visos šios trys dalys siejamos tarpusavyje: galima analizuoti tiek studentų pažangumą, tiek atitinkamo modulio įvertinimus.

Pagrindinės studijų centro akademinės informacijos sistemos veiklos funkcijos ir uždaviniai:

- apskaitos funkcija – kasmet registruojami visi naujai įstoję studentai, duomenų bazė papildoma naujais įrašais;
- kontrolės funkcija – duomenų bazėje fiksuojami egzaminų įvertinimai, matomi visi skolininkai ir nepažangūs studentai, taip pat kaupiami duomenys ir apie dėstytojų atestaciją;
- informacijos kaupimo funkcija – kaupiami duomenys apie studijų centro studentus bei dėstytojus, baigusių (ir nebaigusių) studijas studentų duomenis galima saugoti neribotą laiką, duomenys nuolat papildomi, atrenkami pagal tam tikrus kriterijus, gaminamos atitinkamos ataskaitos, kurias galima peržiūrėti ir/arba spausdinti, duomenų bazės įrašai taip pat peržiūrimi arba spausdinami.

2.1. Duomenų bazių valdymo programos

Kuriamai programinei įrangai galima naudoti vieną ar kitą technologiją. Reikia įvertinti esamų sistemų parametrus ir galimybes. Toliau peržvelgiamos kelios duomenų bazių

valdymo programos. Toks palyginimas leidžia pasirinkti geriausiai poreikius atitinkantį sprendimą. Atsižvelgiama į privalumus, trūkumus, užsakovo reikalavimus.

- ***Microsoft Access***

Gamintojas – kompanija Microsoft. Tai duomenų bazių valdymo programa, esanti Microsoft Office programų pakete. Naudojant Access galima sukurti nemažos apimties, lengvai susiejamas duomenų bases. Access 2003 pateikia didelį įrankių rinkinį, kurių pakanka profesionaliems kūrėjams bei lengva išmokti jais naudotis naujiems vartotojams. Access 2003 palaiko daugelį duomenų formatų, įskaitant XML, OLE, ODBC. Įmanomas duomenų importas ir eksportas.

Pagrindiniai privalumai:

- a) nesudėtingas darbas;
- b) lengva integracija;
- c) kaina neturi įtakos turint Office licenciją.

Pagrindiniai trūkumai:

- a) galima naudoti tik Microsoft Windows platformoje;
- b) sunku integruoti su ne Microsoft korporacijos produktais.

- ***Microsoft SQL***

Gamintojas – kompanija Microsoft. Tai duomenų bazė, skirta įvairiems vartotojų poreikiams. Paprastai yra naudojama kai reikia saugoti didelius informacijos kiekius, pasižymi stabilumu ir sparta.

Pagrindiniai privalumai:

- a) didelės galimybės;
- b) stabilumas, saugumas ;
- c) galimybė dirbti per keliose darbo vietose.

Pagrindiniai trūkumai:

- a) per didelė mažiems projektams;
- b) labai didelė kaina.

- ***MySQL***

Gamintojas – kompanija MySQL AB. Tai duomenų bazė, skirta įvairiems vartotojams. Ji naudojama mažuose ir vidutinės apimties projektuose.

Pagrindiniai privalumai:

- a) didelės galimybės;
- b) stabilumas, saugumas;
- c) nemokamas;

Pagrindiniai trūkumai:

- a) palaikomos ne visos standartinės duomenų bazių funkcijos;
- b) reikia gerų programavimo įgūdžių MySQL serverio valdymui.

▪ ***PostgreSQL***

Kuria ir palaiko grupė žmonių Kanadoje. Duomenų bazė, kuri gali būti taikoma visais atvejais. Ji turi visas reikalingas funkcijas, yra greita, stabili, tinka Linux ir Windows platformoms.

Pagrindiniai privalumai:

- a) didelės galimybės;
- b) stabilumas, saugumas;
- c) nemokama.

Pagrindiniai trūkumai:

- a) reikalingi programavimo įgūdžiai.

▪ ***Oracle***

Gamintojas – Oracle korporacija. Reliacinių duomenų bazių lyderis. Turi visas reikalingas funkcijas, į ją orientuojasi dauguma kitų reliacinių duomenų bazių kūrėjų.

Pagrindiniai privalumai:

- a) palaikomos visos reikalingos funkcijos;
- b) didelės galimybės;
- c) greitai apdoroja didelius duomenų kiekius.

Pagrindiniai trūkumai:

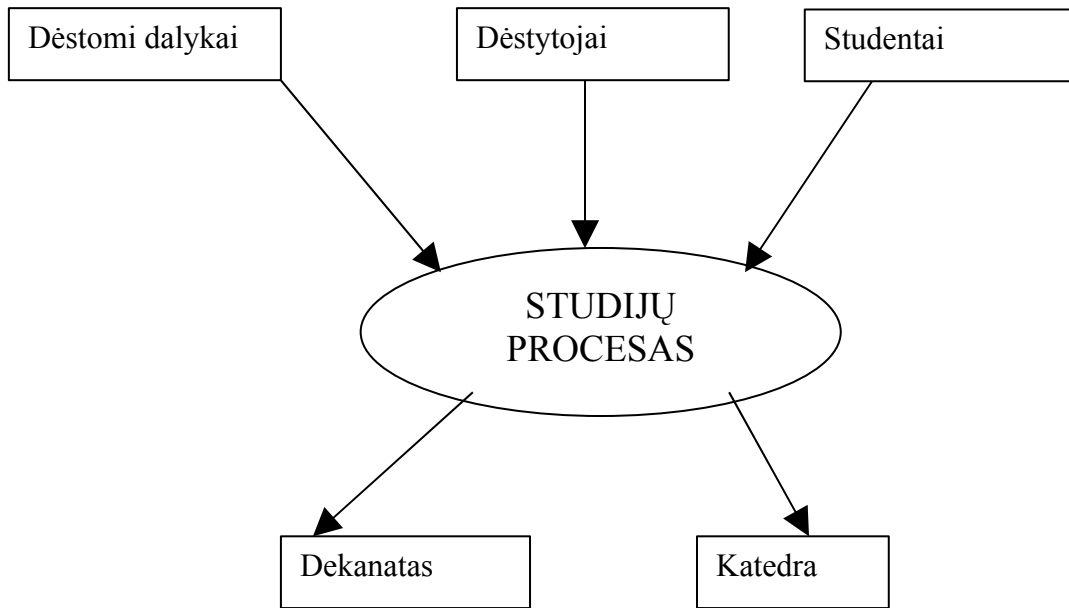
- a) per didelė nedideliems projektams;
- b) didelė kaina.

Projektuojant IS pasirinkta MS Access duomenų bazių valdymo programa:

- a)** tai neblogai žinoma programa, dirbti su ja nesudėtinga;
- b)** Access įeina į MS Office programinės įrangos paketą, todėl turint Office licenciją nebereikės rūpintis kitos programos legalizavimu ir mokėti nemažus pinigus;
- c)** Access galimybių sumanytam projektui pilnai pakanka;
- d)** darbui su kitomis programomis trūksta programavimo įgūdžių.

2.2. Veiklos analizės modeliavimas

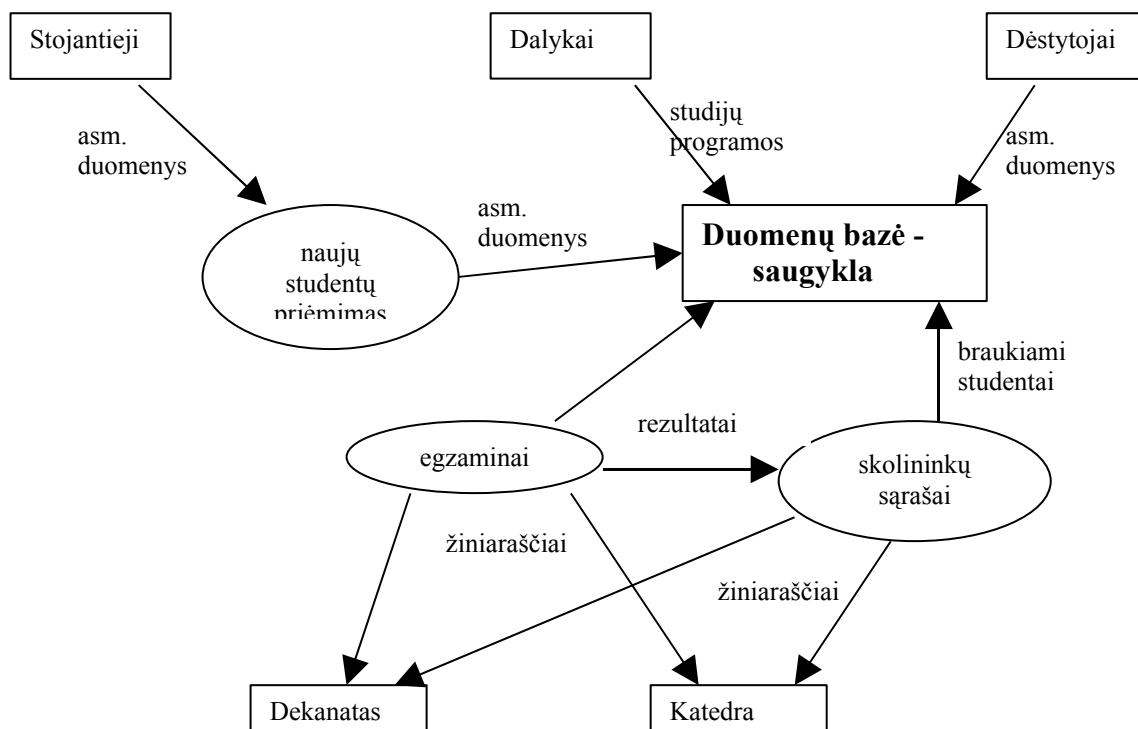
IS inžinerijos pirmasis žingsnis – sudaryti kompiuterizuojamos srities veiklos modelį. Veiklos modelio paskirtis – aprašyti kompiuterizuojamą veiklos sritį, jos svarbiausius procesus, duomenų transformacijas. Vienas populiariausių struktūrinės analizės metodų yra **C.Gane** ir **T.Sarson** sukurtas grafiniais modeliais – duomenų srautų diagramomis- (**DFD** – **Data Flow Diagrams**) pagrįstas sistemų analizės būdas.



1 pav. Aukščiausiojo lygmens duomenų srautų diagrama „Studijų procesas“.

Grafiniai metodai suteikia galimybę vartotojams ir projektuotojams gauti bendrą ir aiškų sistemos vaizdą. Metodologija apima sistemos kūrimą „iš viršaus žemyn“. Pirmiausia sukuriama aukščiausiojo lygmens DFD, kur visa analizuojama organizacija (t.y. visa veiklos sritis) vaizduojama kaip vienintelis, modeliuojamos išorinės organizacijos sąveikos su aplinka procesas.

Duomenų srautai yra vienas iš modelio elementų, kuris parodo, kurie duomenys naudojami procesuose. Duomenų srautų šaltiniai ir vartotojai yra veiklos dalyviai arba duomenų saugyklos. Kitas, pagrindinis, diagramos elementas yra procesas, kuris modeliuoja duomenų apdorojimo operacijas. Pateiksiu tokį veiklą aprašančios duomenų srautų diagramos pavyzdį:



2 pav. Studijų centro akademinės veiklos duomenų srautų diagrama (nulinio lygio).

2.2.1. Informacijos sistemų kūrimo modeliai

„IS inžinerija – tai IS kūrimo teoriniai pagrindai, apimantys modeliavimo metodologijas, analizės ir projektavimo metodus, darbų tvarką ir priemones“ (Garšva; Gudas; Sekliuckis; 2003: 30). IS galima kurti naudojant skirtingas metodologijas, pasirinkus konkretų metodą ir priemones. Žinomi keli klasikiniai IS kūrimo modeliai:

1. **Tradicinis arba „krioklio“** metodas – realizuoja funkcinę dekompoziciją „iš viršaus žemyn“.
2. **Objektinis arba „fontano“** metodas realizuoja objektiškai orientuotą požiūrį „iš apačios į viršų“.
3. **Iteracinis, spiralės tipo** – realizuoja evoliucinį IS kūrimą.

1.Tradicinis IS kūrimo metodas. Trys apibendrinti IS inžinerijos etapai (analizė, projektavimas, realizavimas skaidomi į smulkesnius. Pagrindiniai etapai būtų tokie:

- 1.Vartotojo reikalavimų analizė;
- 2.Vartotojo reikalavimų specifikavimas;
- 3.Reikalavimų programinei įrangai specifikavimas;
- 4.Sistemos architektūros projektavimas;
- 5.Detalus IS projektavimas;
- 6.IS realizavimas (taikomųjų programų kodo kūrimas);
- 7.Programų testavimas: atskirų dalių testavimas;
- 8.Sistemos testavimas: visos IS testavimas;

9.IS diegimas organizacijoje;

10.IS funkcionavimo palaikymas: priežiūra.

Šio metodo svarbiausi trūkumai yra tokie: neatsižvelgiama į evoliucinius kompiuterizuojamos sistemos pakitimus. Toks projektavimo metodas netinka siekiant sukurti programinę įrangą, kurią galima būtų panaudoti pakartotinai – vienai organizacijai sukurtą IS įdiegti kitoje organizacijoje.

2. **Objektinis „fontano“ tipo modelis.** Kaip ir „krioklio“ tipo modelyje čia paeiliui vykdomi IS kūrimo etapai, tačiau iš kiekvieno etapo nuolat grįžtama į ankstesnius etapus. Tai reiškia nuolatinį analitinių ir projektinių sprendimų koregavimą, iteratyvų IS kūrimą.

3. **Spiralės tipo modelis.** Šio modelio esmė – keturių programinės įrangos kūrimo fazių (planavimo ir analizės, projektavimo, realizavimo ir sistemos įvertinimo) kartojimas.

Šiame darbe informacijos sistemos kūrimui naudojau tradicinį metodą – kūrimas vykdomas nuosekliai etapais, pradedant reikalavimų specifikavimu. „Fontano“ tipo modelis reikštų nuolatinį taisymą ir grįžimą atgal, dėl to padidėtų darbo laiko sąnaudos ir projektas nebūtų užbaigtas laiku. Spiralės tipo modelis, be abejo, geras, tačiau jį naudojant reikia geros „susigrojusios“ komandos.

2.2.2. Duomenų organizavimo modeliai

Pagrindiniai duomenų organizavimo modeliai apibūdinami priklausomai nuo to, koku būdu DB duomenys jungiami į visumą. Skiriami tokie DB modeliai: hierarchinis, tinklinis, reliacinis, objektinis.

1. **Hierarchinis modelis.** Jame išskiriami skirtingi duomenų įrašų hierarchijos lygiai. Vienas įrašas yra pagrindinis, kiti – jam pavaldūs. Pavaldus įrašas gali būti siejamas su žemesnio lygio įrašu, šis gali būti siejamas su dar žemesnio lygio įrašu ir t.t.

Tokio modelio svarbiausi trūkumai yra šie:

- priesti prie bet kokio pavaldaus įrašo galima tik per pagrindinį įrašą;
- negali būti tiesiogiai realizuojami abipusiai sudėtingi ryšiai „daug su daug“;
- sudėtingos duomenų įvedimo ir šalinimo operacijos.

2. **Tinklinis modelis.** Čia leidžiama didesnė ryšių tarp įrašų įvairovė. Įrašas gali būti susietas su bet kuriuo kitu įrašu nepriklausomai nuo jo lygio. Tinklinio modelio DB pagrindinis sandaros elementas yra rinkinys, kuris sujungia kelių tipų įrašus. Taigi rinkinys yra susijusių įrašų aibė. Pagrindinis šio modelio trūkumas – sudėtingumas: reikia gerai išmanyti loginę DB struktūrą.

3. **Objektiniuose modeliuose** taikomos objektinės technologijos. Tokiuose modeliuose naudojami objektai, kuriems būdingi ne tik jų savybes aprašantys atributai, bet ir jiems taikomų veiksmų rinkiniai. Objektinio modelio sandaros elementai artimi objektinio programavimo sąvokoms.

4. **Reliacinis modelis** – tai tarpusavyje susietų lentelių rinkinys. Lentelę sudarančios eilutės vadinamos įrašais, o stulpeliai vadinami laukais. Laukams suteikiami vardai. Visi lentelės įrašai yra vienodos struktūros: yra tiek pat vienos rūšies laukų, skirtinguose laukuose gali būti įvairių tipų duomenys. Lentelėje negali būti tuščių, taip pat ir identiškų įrašų, nors atskiri laukai gali būti tušti arba jų reikšmės gali pasikartoti. Kiekvienai lentelei suteikiamas vardas, ir ji saugojama kompiuterio atmintyje kaip atskiras objektas.

Lentelei nustatomi raktai – laukai ar laukų grupės, kurių reikšmės yra nepasikartojančios. Negali būti eilučių su vienodomis rakto reikšmėmis. Toks laukas vadinamas raktiniu lauku.

Lentelės tarpusavyje susiejamos. Ryšiai tarp lentelių padaromi tarp tą pačią prasmę skirtingose lentelėse turinčių laukų. Taip susietų lentelių visuma apibūdinama kaip reliacinis DB modelis.

Šiame darbe aprašoma informacijos sistemos duomenų bazė yra reliacinė.

2.3. Reikalavimai sistemai

- Duomenys įvedami rankiniu būdu, naudojantis specialiomis formomis.
- Duomenys išvedami ekrano formose ir į spausdintuvą.
- Sistema turi naudoti duomenų bazę informacijos saugojimui.
- Sistema turi būti saugi.
- Sistemos vartotojai skirstomi keliais nevienodais lygiais.
- Vartotojo sąsają ir vartotojų prisijungimą tvarko sistemos administratorius.
- Sistema turi naudoti mažai vidinio įstaigos tinklo resursų.
- Sistemoje turi būti realizuota kliento- serverio architektūra, t.y. galimybė dirbti keliems vartotojams.
- Vartotojo sąsaja turi būti aiški ir patogi.

Vienas esminių vartotojo pageidavimų – produkto patikimumas. Informacijos sistema turi veikti be sutrikimų viename ar kitame lygmenyje.

Kitas svarbus reikalavimas – sistemos saugumas. Informacijos sistema neturėtų būti pažeista dėl vartotojo kaltės, sakysim, neteisingai įvedamų duomenų. Tam tikslui visur, kur tik įmanoma, naudojama duomenų tikrinimo funkcija (validation). Nereikšmingos klaidos (neteisingai įvesta pavardė ar telefono numeris) nesukels sistemos sutrikimo, o tokias klaidas

aptikus visada įmanoma ištaisyti. Kur kas svarbiau yra apsaugoti pačią sistemą nuo nesankcionuoto naudojimo ar duomenų sunaikinimo. Tam tikslui startiniame informacijos sistemos lange būtina vartotojo identifikacija - sistema reikalauja nurodyti, kuris vartotojas jungiasi prie sistemos: raštinės sekretorė, dėstytojas ar administratorius. Informacija apie vartotojų teises ir slaptažodžius įtraukiama į specialų failą, kuris prieinamas tik sistemos administratoriui.

2.4. Reikalavimai programinei įrangai

- programa turi veikti be sutrikimų;
- vartotojo klaidos neturi sukelti programos kritinės būsenos, t.y. sistema dėl vartotojo padarytų klaidų neturi nustoti veikti;
- programą lengva valdyti;
- sukurtos vaizdžios ir aiškiai suprantamos įvedimo formos;
- sukurtos logiškos užklauskos, tenkinančios vartotojo poreikius;
- pagal vartotojo pageidavimus formuojamos ataskaitos leidžia atspausdinti reikiamus duomenis;
- išeiti iš programos nesudėtinga;
- nesunku atlikti techninius veiksmus.

2.5. Aparatiniai ir sisteminiai resursai bei naudojamos technologijos

Programinės įrangos kūrimui ir testavimui naudojami asmeniniai IBM PC. Realizuojant projektą naudojamos operacinės sistemos: MS Windows 2000, MS Windows XP Pro. Programinė įranga :

- MS Office XP Pro;
- MS Visual Basic 6.0;
- MS Visio 2000;
- Internet Explorer 6.0 .

Programinei įrangai kurti naudojama duomenų bazių kūrimo programa Microsoft Access bei MS Windows XP terpė. Dokumentacija rašoma MS Word XP.

Informacijos sistema gyvuos neribotą laiką, kol bus sukurta geresnė sistema. Be abejo, laikui bėgant iškils naujų reikalavimų, todėl esamą informacijos sistemą reikės modifikuoti, pakeisti, priderinti.

2.6. Pradiniai duomenys ir rezultatų išvedimas

Pradiniai duomenys – visa tai, kas yra sukaupta duomenų bazėje rengiant akademinę informacijos sistemą. Tai duomenys apie studijų centro dėstytojus (vardas, pavardė, asmens kodas, adresas, telefonas, kvalifikacinė kategorija,), duomenys apie studijų centre dėstomus

dalykus (pavadinimas, kreditų skaičius), duomenys apie studentus (vardas, pavardė, studento bilieta numeris, akademinė grupė, stojamasis balas, studijų pradžios metai, studijų pabaigos metai, studijuojami dalykai, dalykų įvertinimai). Iš visų šių duomenų per atitinkamas užklausas ir ataskaitas gaunami rezultatai: greitai galima sužinoti bet kurio studento ar dėstytojo asmeninius duomenis, paskaičiuoti studentų grupės pažangumą, suformuoti skolininkų sąrašą, sužinoti nepažangių ar išbraukiamų studentų pavardes ir kt.

Pradiniai duomenys įvedami naudojant atitinkamas duomenų bazės įvedimo formas, kur vartotojui tereikia užpildyti reikiamus laukus. Čia galima pritaikyti duomenų įvedimo kontrolę, pvz. lauke „asmens kodas“ galima įvesti tik reikiamą skaitmenų kiekį, ir pirmasis jų turi būti „3“ arba „4“, klaidos atveju sistema išduoda pranešimą.

Rezultatai pateikiami vartotojui per atitinkamų užklausų langus. Vartotojas pasirenka norimą užklausą ir, įvedęs papildomą reikalaujamą informaciją (pvz. akademinę grupę), užklausos lange gauna norimą informaciją:

Vartotojo sąsaja taip pat buvo kuriama naudojant daugiausia standartines MS Access priemones. Sąsaja nėra orientuota informacinių technologijų profesionalui, orientavausi daugiau į patyrusį vartotoją, kuris turi pakankamai darbo kompiuteriu įgūdžių ir kuriam nereikia aiškinti, kaip pele spragtelėti mygtuką. Vartotojo sąsajos įvedimo formose naudojami komandinių mygtukų užrašai „Naujas įrašas“, „Baigti darbą“ lengvai suprantami ir nepatyrusiam vartotojui. Siekiant užtikrinti įvedamų duomenų korektiškumą visur kur įmanoma naudojamas duomenų tikrinimas (validation), vartotojo klaidos atveju programa pateikia trumpą pranešimą apie klaidą, pvz. „Neteisingi duomenys!“, „Įveskite 6 simbolius!“ ir t. t.

2.7. Galimos modifikacijos ateityje

Kartu su informacijos sistemos duomenų baze pateikiama vartotojo dokumentacija, todėl ateityje galima būtų įgyvendinti kai kuriuos pakeitimus:

- vartotojo sąsajos pakeitimas;
- pasikeitus vartotojo reikalavimams pakeisti duomenų bazės struktūrą (pridėti papildomai lentelių, užklausų, ataskaitų);
- gali būti taisomos vartotojo pastebėtos klaidos;
- padaryti sistemą prieinamą studentams. Kadangi tokiu atveju ženkliai padidės vartotojų skaičius, reikėtų sistemą perdirbti iš pagrindų: suprogramuoti MySQL serverį ir ten įdėti sistemos duomenis. Padaryti priėjimą internete.

2.8. Testavimas

Testavimas skirtas patikrinti, ar programa veikia teisingai bei ar joje nėra klaidų.

Sistema kuriama ir testuojama tradiciniu būdu: kūrimo etapai vykdomi paeiliui, kiekvienas elementas testuojamas atskirai, užbaigus kūrimo procesą.

Vartotojo sąsaja testuojama praktiniu būdu, pasodinus prie duomenų bazės eilinių vartotoją. Buvo žiūrima, kaip vartotojas sugeba dirbti su duomenų baze, atlikti įvairius veiksmus, ar jam suprantami visi pateikiami pranešimai, ar tinkamai pateikiama pagalba. Galutinis duomenų bazės testavimas ir priėmimas numatomas paleidus ją SQL serveryje ir prisijungiant keletui vartotojų. Tai bus galutinis, integracinis testavimo etapas. Siekiama užtikrinti, kad programa dirbtų bet kurioje Windows terpėje (OS).

2.9. Programinės įrangos paskirtis ir teikiamos paslaugos

Akademinė informacijos sistema skirta naudotis kolegijos administracijai ir dėstytojams. IS pasiekiami iš bet kurio studijų centro tinkle esančio kompiuterio.

Akademinė informacijos sistema teikia tokias paslaugas: matomi visi dėstomi dalykai (moduliai), suskirstyti pagal katedras, dalyko kreditų skaičius, visų studentų sesijos egzaminų galutiniai įvertinimai. Studijų centro administracijai pateikiama visa informacija apie studentus ir dėstytojus, ir su šia informacija galima atlikti kai kuriuos veiksmus: paskaičiuoti studentų grupės ar kurso pažangumo vidurkį, matyti skolininkų sąrašą, matyti šalinamus studentus ir t.t.

Akademinė informacijos sistema skirta Kėdainių Jonušo Radvilos studijų centrui, tačiau sukurtos duomenų bazės rėmus galima pritaikyti bet kurios Lietuvos kolegijos poreikiams, padarius atitinkamus pakeitimus ir suvedus savo įstaigos duomenis. Duomenų bazės instaliuoti nereikia, ji puikiai veiks bet kuriame kompiuteryje, turinčiame įdiegtą MS Access paketą. Adaptavimas konkrečiai aplinkai nebūtų labai sudėtingas: reikėtų pagal įstaigos poreikius peržiūrėti duomenų bazės struktūrą, lentelių tarpusavio ryšius. Iš kitos pusės, pritaikyti svetimą produktą gal ir nėra taip lengva, gal paprasčiau sukurti savo produktą pagal savo poreikius.

Vartotojui dirbant su duomenų baze ir atliekant įvairius veiksmus programa pateikia pranešimus netinkamo veiksmo ar klaidos atveju. Tai trumpi įspėjamojo ar rekomenduojamo pobūdžio pranešimai, iš kurių aiškiai suprantama klaida ir kaip elgtis toliau.

Kuriamos informacijos sistemos prototipo nėra. Produkto modelis taipogi nebus kuriamas, nes tam nėra laiko ir nesiruošiu šios sistemos platinti. Numatoma sistemą patalpinti į SQL serverį ir paleisti naudojimui Kauno kolegijos Kėdainių Jonušo Radvilos studijų centro vietiniame kompiuterių tinkle.

Vartotojui ypatingų reikalavimų nekeliama, tačiau vartotojas privalėtų:

- mokėti naudotis kompiuteriu;
- turėti darbo su Windows operacine sistema įgūdžių.

Eksploatavimo reikalavimai:

- 468DX/66 MHz arba greitesnis mikroprocesorius (rekomenduojamas ne mažesnis kaip 700 MHz dažnis);
- VGA 800 x 600 arba didesnės skiriamosios gebos ir ne mažiau 16 bitų spalvinės raiškos darbinis ekrano režimas palaikomas MS Windows;
- Windows 98SE, Windows 2000 arba Windows XP operacinė sistema;
- minimalūs reikalavimai 64 MB RAM operacinei sistemai MS Windows 98SE, 128 MB – Windows 2000 ar Windows XP;
- atitinkamos talpos kompiuterio kietasis diskas.

3. PROJEKTINĖ DALIS

3.1. Reikalavimų specifikacija

Sistemos apribojimai

Nustatyti tokie apribojimai sistemos realizacijai:

- Sistema turi kaupti ir leisti peržiūrėti sukauptus duomenis;
- sistema privalo gebėti atlikti skaičiavimus, juos apjungti ir pavaizduoti konsoliduota forma;
- sistemos funkcijos turi būti prieinamos visiems vartotojams;
- sistema turi turėti galimybę būti praplečiama ar adaptuojama ateityje, jei pasikeičia rinkos poreikiai ar atsiranda nauji reikalavimai;
- sistemos vartotojams prieinamos funkcijos turi atitikti įstaigos nuostatas.

Diegimo aplinka

Sistema veiks bet kurioje Windows operacinėje sistemoje (9X/ 2000/ XP).

Bendradarbiaujančios sistemos

Sistemos pagrindas yra MS Access, todėl jos palaikymui reikalingas MS Office programinis paketas.

Komerciniai specializuoti programų paketai

Į kuriamą sistemą nenumatoma įtraukti jokių papildomų specializuotų programų paketų. Planuojama, jog sistema pilnai funkcionuos naudojant MS ACCESS programinį paketą.

Numatoma darbo vietos aplinka

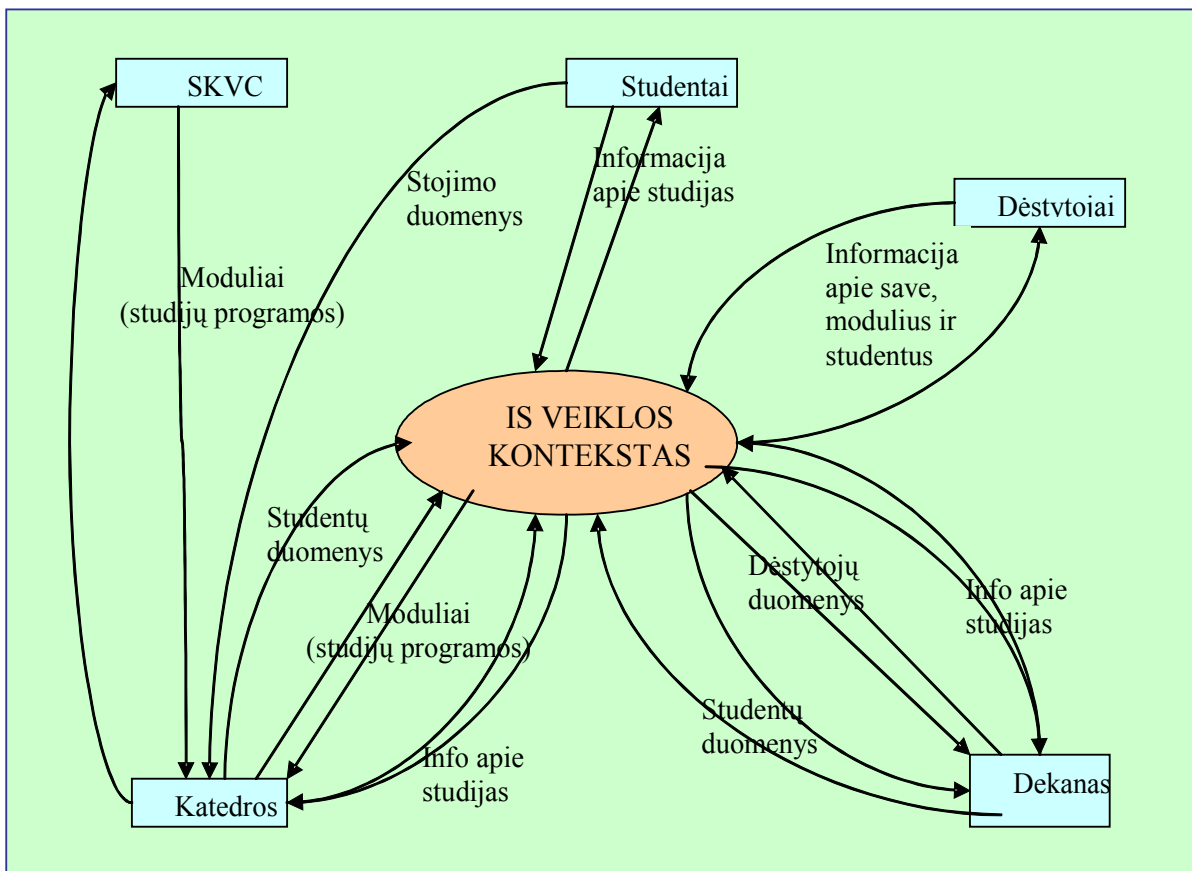
Darbo vietos visur yra darbo kabinetai. Kiekvienoje darbo vietoje yra spausdintuvas. Sistema reikalauja autorizuoto vartotojo prisijungimo. Kiekvienas vartotojas prisijungdamas privalo įvesti prisijungimo slaptažodį. Slaptažodžius tvarko ir periodiškai keičia sistemos administratorius.

Vartotojo sąsaja

Vartotojo sąsajai išreikšti reikalavimai tokie: ji turi būti nesudėtinga, neįmantri, lengvai valdoma. Vartotojo neturi šokiruoti nematytos naujovės. Tai bus standartinis Windows aplinkos langas su mygtukais.

3.1.1. Funkciniai reikalavimai

Veiklos kontekstas (konteksto diagrama)



3 pav. Konteksto diagrama.

Iš šios schemos matyti visas IS veiklos kontekstas. Katedra siunčia studijų programas (modulius) į studijų kokybės vertinimo centrą (SKVC). Patvirtinus studijų programą, ji įtraukiama į IS duomenų bazę. Taip pat į DB įtraukiami įstojusių studentų duomenys. Į DB įtraukiami studijų duomenys: tai sesijos rezultatai. Taip pat į DB įvedami dėstytojų duomenys. Toliau visais šiais duomenimis laisvai gali disponuoti dekanatas ir katedros. Duomenys gali būti koreguojami ir papildomi. Nereikalingi duomenys iš duomenų bazės pašalinami.

Reikalavimai duomenims

- R1.** Sistema turi formuoti vartotojo užklausas tenkinančias ataskaitas.
- R2.** Sistema turi leisti vartotojui įvesti ir atnaujinti duomenis.
- R3.** Sistemoje turi būti kaupiami visų studijų centrą baigusiu studentų duomenys.
- R4.** Sistemoje turi būti kaupiami pašalintų studentų duomenys.
- R5.** Sistemoje turi būti galimybė fiksuoti egzaminų įvertinimus.

R6. Sistema privalo būti lanksti ir lengvai modifikuojama, pritaikoma naujiems poreikiams.

Reikalavimas R1.

Aprašymas: Sistema turi formuoti vartotojo užklausas tenkinančias ataskaitas.

Pagrindimas: Reikalinga, kad sistemos naudotojas galėtų gauti jo užduočių vykdymui reikalingus duomenis konsoliduota forma.

Tinkamumo kriterijus: Turi būti pateikiamos įvairios ataskaitos apie studentų pažangumą, modulius (valandų ir kreditų skaičius), dėstytojų kvalifikaciją. Jei įvyko sistemos klaida, į ekraną išvedamas atitinkamas pranešimas.

Užsakovo patenkinimas: 5 **Užsakovo nepatenkinimas:** 4

Reikalavimas R2.

Aprašymas: Sistema turi leisti vartotojui įvesti ir atnaujinti duomenis.

Pagrindimas: Reikalinga, siekiant užtikrinti informacijos teisingumą ir naujumą.

Tinkamumo kriterijus: Darbuotojui leidžiama įvesti/keisti studentų, modulių, dėstytojų duomenis, egzaminų rezultatus ir kitus duomenis.

Užsakovo patenkinimas: 4 **Užsakovo nepatenkinimas:** 4

Reikalavimas R3.

Aprašymas: Sistemoje turi būti kaupiami studijų centrą baigusiu studentų duomenys.

Pagrindimas: Reikalinga siekiant sukurti ilgalaikę duomenų bazę ir norint turėti pilną informaciją apie baigusius studentus.

Tinkamumo kriterijus: Į duomenų bazę įvedami išduotų baigimo diplomų numeriai, pagal šį požymį baigusieji studentai atrenkami, jų duomenys perkeliama į atskirą lentelę.

Užsakovo patenkinimas: 4 **Užsakovo nepatenkinimas:** 4

Reikalavimas R4.

Aprašymas: Sistemoje turi būti kaupiami pašalintų studentų duomenys.

Pagrindimas: Reikalinga siekiant sukurti ilgalaikę duomenų bazę ir norint turėti pilną informaciją apie pašalintus studentus.

Tinkamumo kriterijus: Užpildomas atskiras laukas „Šalinimo data“. Duomenys atrenkami pagal šį požymį. Šiuos duomenis bet kada turi būti įmanoma peržiūrėti.

Užsakovo patenkinimas: 4 **Užsakovo nepatenkinimas:** 4

Reikalavimas R5.

Aprašymas: Sistemoje turi būti galimybė fiksuoti egzaminų įvertinimus.

Pagrindimas: Reikalinga siekiant sukurti ilgalaikę duomenų bazę ir norint turėti pilną informaciją apie studentus ir jų mokymosi rezultatus.

Tinkamumo kriterijus: Darbuotojui leidžiama įvesti/keisti egzaminų rezultatus. Tiems duomenims sukuriama atskira lentelė ir įvedimo forma.

Užsakovo patenkinimas: 5 **Užsakovo nepatenkinimas:** 5

Reikalavimas R6.

Aprašymas: Sistema privalo būti lanksti ir lengvai modifikuojama, pritaikoma naujiems poreikiams.

Pagrindimas: Reikalinga, kad būtų galima informacijos sistemą pritaikyti prie pasikeitusios organizacijos ar mokymo struktūros.

Tinkamumo kriterijus: Turi būti įmanoma kurti naujas lenteles, įvedimo formas bei užklausas ir ataskaitas, ir visi šie nauji produktai turi lengvai įsirašyti į sistemą.

Užsakovo patenkinimas: 5 **Užsakovo nepatenkinimas:** 3

3.1.2. Nefunkciniai reikalavimai

Reikalavimai sistemos išvaizdai:

- sistemos išvaizda neturėtų būti šokiruojanti ar pritrenkianti: reikalinga paprasta vartotojo sąsaja kasdieniam darbui;
- vartotojo sąsajos valdymo elementai turėtų būti standartiniai įprasti mygtukai;
- vartotojo sąsaja privalo būti lietuviška.

Reikalavimai panaudojamumui

R7. Sistema turi būti paprasta naudotis.

R8. Sistemoje turi būti galimybė taisyti padarytas klaidas.

R9. Sistema privalo turėti išplėtimo galimybę.

Reikalavimas R7.

Aprašymas: Sistema turi būti paprasta naudotis.

Pagrindimas: Sistema naudotis turi būti paprasta, kad darbuotojai nevengtų ja naudotis.

Tinkamumo kriterijus: Žinantis sistemos funkcines galimybes vartotojas sugebėtų naudotis sistema be papildomo apmokymo.

Užsakovo patenkinimas: 5 **Užsakovo nepatenkinimas:** 3

Reikalavimas R8.

Aprašymas: Sistemoje turi būti galimybė taisyti padarytas klaidas.

Pagrindimas: Reikalinga, kadangi sistemos ir vartotojo tikslas yra kaupti tik teisingus duomenis .

Tinkamumo kriterijus: Vartotojo sąsaja turi galimybę pereiti į duomenų redagavimo režimą. Duomenys redagavimui pateikiami lentelės forma. Kai kuriuos duomenis įmanoma redaguoti tiesiog įvedimo formoje.

Užsakovo patenkinimas: 5 **Užsakovo nepatenkinimas:** 4

Reikalavimas R9.

Aprašymas: Sistema privalo turėti išplėtimo galimybę.

Pagrindimas: Užsakovui gali prireikti integruoti į sistemą naujus komponentus arba patobulinti esamus.

Tinkamumo kriterijus: Naujo programinio komponento integravimas į sistemą neturi sukelti didelių problemų.

Užsakovo patenkinimas: 5 **Užsakovo nepatenkinimas:** 4

Reikalavimai vykdymo charakteristikoms

R10. Operacijos turi būti atliekamos pakankamai sparčiai.

R11. Sistema turi būti kuo labiau apsaugota nuo vartotojo klaidų.

Reikalavimas R10.

Aprašymas: Operacijos turi būti atliekamos pakankamai sparčiai.

Pagrindimas: Operacijos turi būti vykdomos taip, kad vartotojui nekiltų įtarimas, jog sistema „užstrigo“

Tinkamumo kriterijus: Operacijų vykdymo greitis turi būti proporcingas apdorojamų duomenų kiekiui.

Užsakovo patenkinimas: 3 **Užsakovo nepatenkinimas:** 3

Reikalavimas R11.

Aprašymas: Sistema turi būti kuo labiau apsaugota nuo vartotojo klaidų – naudojamas įvedamų duomenų tikrinimas.

Pagrindimas: Sistemos korektiškumas: neapkrauti vartotojo papildomais reikalavimais; pateikti klaidos atveju išpėjamuosius pranešimus.

Tinkamumo kriterijus: Sistema funkcionuoja be klaidų arba klaidų skaičius yra minimalus.

Užsakovo patenkinimas: 5 **Užsakovo nepatenkinimas:** 4

Reikalavimai veikimo sąlygoms

Kadangi sistema veiks normalioje aplinkoje, kur nėra jokių ypatingų išorinių trukdžių, vartotojų darbo sąlygos geros tiek žiemą, tiek vasarą, todėl jokių ypatingų reikalavimų sistemos veikimo sąlygoms nebuvo užfiksuota.

Reikalavimai sistemos priežiūrai

- sistema turi būti realizuota taip, kad jos veiklai užtikrinti nereikėtų nuolatinio programuotojo įsikišimo;
- kartu su sistema privalo būti pateikiama jos dokumentacija, kad pasikeitus personalui nesutriktų sistemos veikla.

Reikalavimas R12.

Aprašymas: Dokumentavimo sistema – pateikti vartotojo vadovą.

Pagrindimas: Pateikti tokią dokumentaciją, kuri leistų ateityje atlikti palaikymo darbus organizacijai savo jėgomis.

Tinkamumo kriterijus: Savalaikė dokumentacija.

Užsakovo patenkinimas: 5 **Užsakovo nepatenkinimas:** 4

Reikalavimai saugumui:

R13. Konfidencialumas – sistemoje esantys duomenys apsaugoti nuo neteisėto naudojimo.

R14. Registruotiems sistemos vartotojams prieinami ne visi duomenys, o tik tie, kuriais jiems leidžiama naudotis.

Reikalavimas R13.

Aprašymas: Apsauga nuo neteisėto prisijungimo.

Pagrindimas: Sistema gali naudotis tik autorizuoti vartotojai .

Tinkamumo kriterijus: Sukurta prisijungimo prie sistemos slaptažodžių sistema

Užsakovo patenkinimas: 5 **Užsakovo nepatenkinimas:** 5

Reikalavimas R14.

Aprašymas: Duomenų konfidencialumas – sistemoje esantys duomenys apsaugoti nuo neteisėto naudojimo.

Pagrindimas: Vartotojai skirstomi keliais lygiais ir gali naudotis tik savo lygmens duomenimis.

Tinkamumo kriterijus: Sukurta prisijungimo prie sistemos slaptažodžių sistema.

Užsakovo patenkinimas: 5 **Užsakovo nepatenkinimas:** 5

Kultūriniai – politiniai reikalavimai

Reikalavimas R15.

Aprašymas: Sistemoje nėra įžeidžiančių terminų ir simbolių.

Pagrindimas: Sistema gali naudotis įvairių tikėjimų ar įsitikinimų žmonės, todėl joje privalo būti naudojamos tik visoms vartotojų kategorijoms priimtinos notacijos.

Tinkamumo kriterijus: Produktas atitinka etikos normas.

Užsakovo patenkinimas: 3 **Užsakovo nepatenkinimas:** 3

Teisiniai reikalavimai

Reikalavimas R16.

Aprašymas: Sistema nepažeidžia Lietuvos Respublikos įstatymų.

Pagrindimas: Visi sistemos atliekami veiksmai yra teisėti, neprieštarauja LR įstatymams.

Tinkamumo kriterijus: Sistema apsaugota nuo neteisėto kitų fizinių ar juridinių asmenų naudojimo.

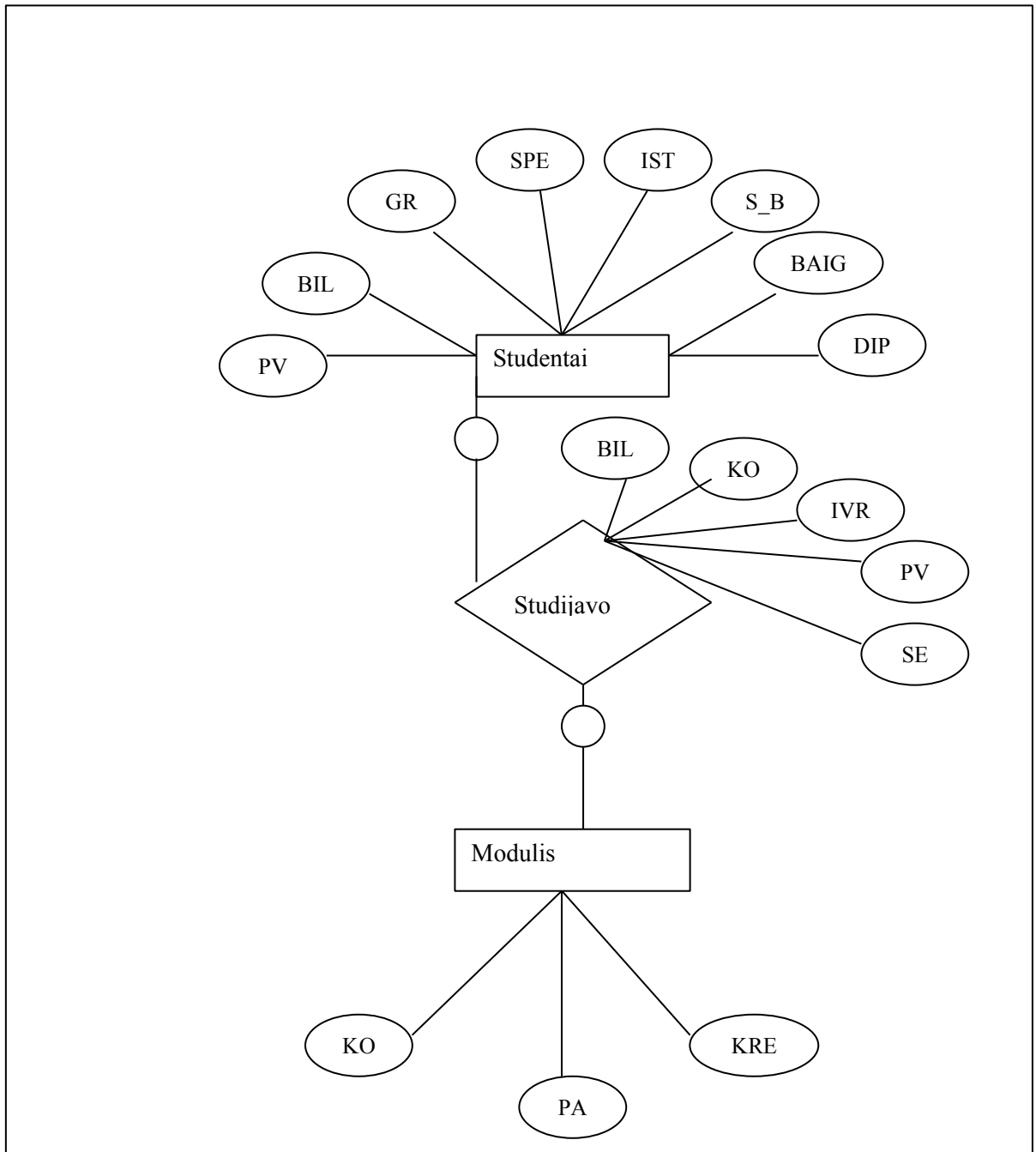
Užsakovo patenkinimas: 4 **Užsakovo nepatenkinimas:** 5

Perspektyviniai reikalavimai

Netolimoje ateityje reikia pertvarkyti sistemą taip, kad prie jos galima būtų prisijungti naudojant interneto naršyklę. Tam reikia suprojektuoti MySQL serverį ir atitinkamai sugrupuoti vartotojus. Sėkmingai realizavus projektą galima pagalvoti apie sistemos pateikimą rinkai, todėl projektuojant vertėtų atsižvelgti į sistemos lankstumą: galimybę modifikuoti, pritaikyti naujoms sąlygoms ir reikalavimams.

3.2. Duomenų struktūra

Kiekvienos DB projektavimo procedūra remiasi kiekvieno vartotojo sampratos realizavimu. Tokiu būdu suprojektuotas DB modelis orientuotas į loginį reliacinį modelį. Bendresnio pavidalo konceptualiam DB modeliui sudaryti ir smulkiau jį aprašyti dažnai naudojamos **ER (Entity – Relation)** (lietuviškai esybių – ryšių) diagramos.



4 pav. DB ER diagrama.

Aukščiau esančioje diagramoje panaudotos santrumpos reiškia lentelių laukų pavadinimus:

PAVARDĖ_VARDAS (PVR)
 BILieto_NUMERIS (BIL)
 AKADEMINĖ_GRPĖ (GRU)
 STUDIJŲ_PROGRAMA (SPE)
 ĮSTOJIMO_METAI (IST)
 STOJAMASIS_BALAS (S_B)
 BAIGIMO_METAI (BAIG)
 DIPLOMO_NUMERIS (DIP)
 BILieto_NUMERIS (BIL)
 DISCIPLINOS_KODAS (KOD)
 ĮVERTINIMAS (IVR)
 PAVARDĖ_VARDAS (PVR)
 SEMESTRAS (SEM)
 DISCIPLINOS_KODAS (KOD)
 PAVADINIMAS (PAV)
 KREDITŲ_SKAİČIUS (KRED)

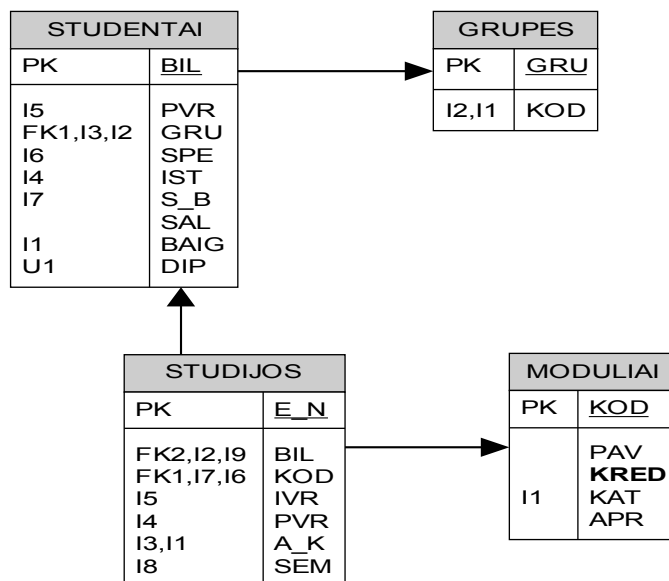
Duomenų bazės ER diagramą galima pakeisti loginiu reliaciniu modeliu. Tuomet turime tokias lenteles:

Studentai (PVR, BIL, GRU, SPE, IST, S_B, BAIG, DIP)

Studijos (BIL, KOD, IVR, PVR, SEM)

Moduliai (KOD, PAV, KRED)

Grupės (GRU, KOD)



5 pav. DB reliacinė schema.

3.3. Sistemos architektūra

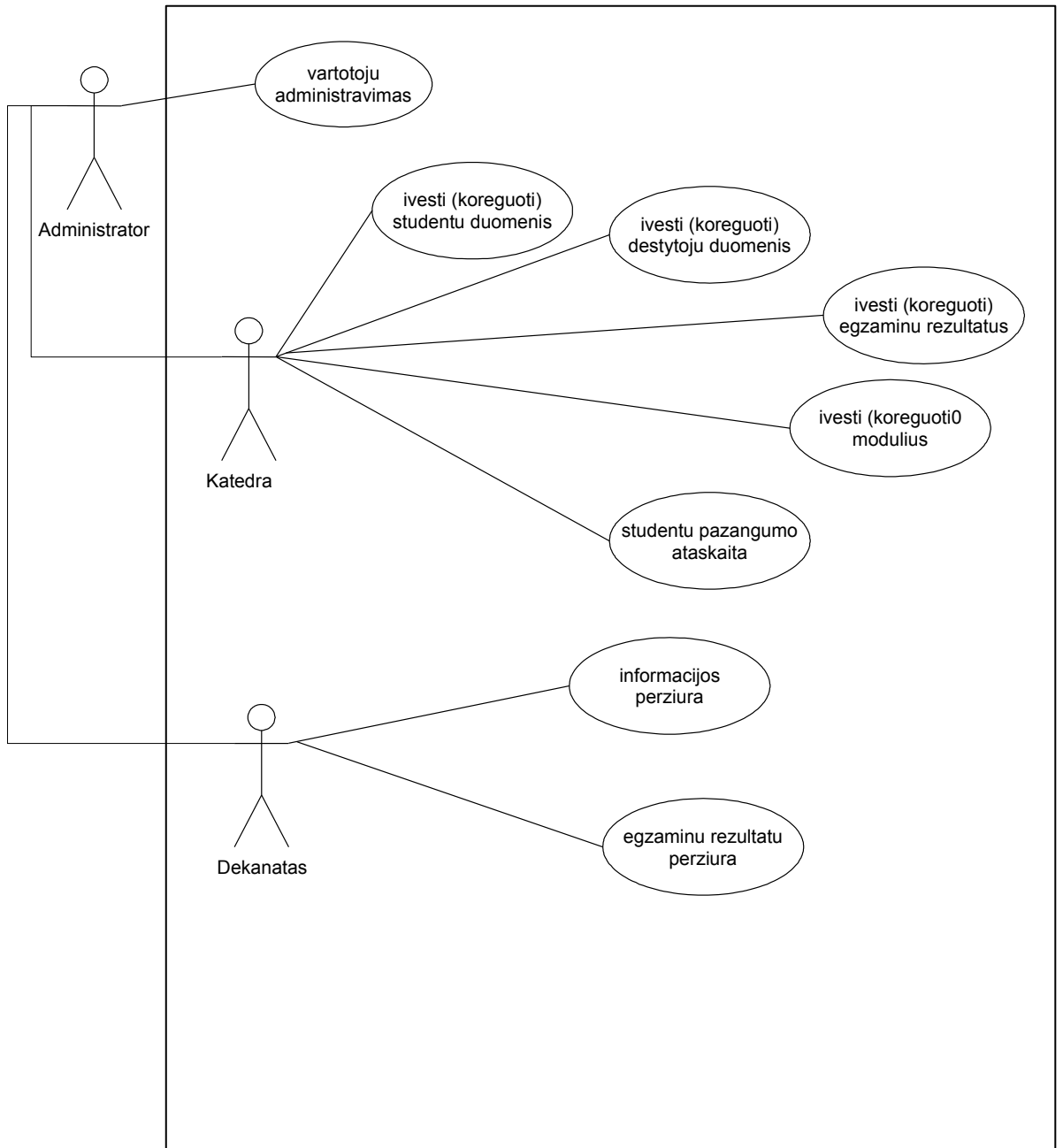
Sistemos architektūra šiame dokumente pateikiama keliais vaizdais: panaudojimo atvejų (PA), procesų, išdėstymo ir realizavimo. Visi tie vaizdai pateikiami diagramose su paaiškinimais kartu naudojant UML. Sistemos architektūra pateikiama remiantis RUP rekomendacijomis. Sistemos architektūros specifikacija pateikiama:

- panaudojimo atvejų vaizdu (panaudojimo atvejų diagrama);
- loginis programos vaizdas (klasių diagrama, skaidymas į paketus);
- procesų vaizdas (būsenų kaitų diagramos, sekų diagramos);
- išdėstymo vaizdas (išdėstymo diagrama).

3.3.1. Architektūros tikslai ir apribojimai

- Kuriama programinė įranga yra nekomercinė, ji bus pateikiama studijų centre bendram naudojimui kaip atviro kodo programinė įranga;
- sistemos architektūra turi būti suprojektuota taip, kad ją būtų galima lengvai praplėsti ar prijungti naujus komponentus;
- sistema kuriama mokymo tikslu, todėl jos architektūra turi būti lengvai suprantama kitiems;
- sistema neturi leisti prisijungti neautorizuotiems vartotojams;
- sistema realizuojama kliento-serverio modeliu. Viena kompiuteryje įdiegiamas SQL serveris, ten įdedama duomenų bazė ir tuomet prie DB galės prisijungti keli vartotojai;
- sistema turi užtikrinti kaupiamų duomenų saugumą. Sistemos vartotojui prieinami duomenys priklauso nuo jo pareigų įstaigoje ir suteikiamų teisių kaip duomenų bazės vartotojui;
- Projektavimo įrankiai: MS Word, MS Visio.

3.3.2. Panaudojimo atvejai



6 pav. Panaudojimo atvejų diagrama.

Panaudojimo atvejų aprašymas:

PAI: Vartotojų administravimas.

Vartotojas: Sistemos administratorius.

Aprašymas: Registruojami, šalinami, koreguojami sistemos vartotojai.

Panaudojimo atvejo scenarijus: Įvedama vartotojo vardas, pavardė, slaptažodis, prisijungimo vardas.

Prieš sąlyga: Vartotojas dar nėra registruotas sistemoje.

Sužadinimo sąlyga: Atsiranda naujas dar neregistruotas vartotojas.

Po sąlyga: Sistemoje užregistruojamas naujas vartotojas.

PA2: Įvesti (koreguoti) studentų duomenis.

Vartotojas: Katedra.

Aprašymas: Įvedami naujų studentų duomenys, koreguojami (reikalui esant) esamų studentų duomenys.

Panaudojimo atvejo scenarijus: Vartotojas prisijungia prie sistemos, iškviečia reikalingą įvedimo formą ir suveda naujų studentų asmeninius duomenis arba koreguoja esamus duomenis.

Prieš sąlyga: Studentai dar neregistruoti sistemoje (arba pasikeitė duomenys).

Sužadinimo sąlyga: Įstojo nauji studentai.

Po sąlyga: Sistemoje registruojami nauji studentai arba pakeičiami esami duomenys.

PA3: Įvesti (koreguoti) dėstytojų duomenis.

Vartotojas: Katedra.

Aprašymas: Įvedami naujų dėstytojų duomenys, koreguojami (reikalui esant) esamų dėstytojų duomenys.

Panaudojimo atvejo scenarijus: Vartotojas prisijungia prie sistemos, iškviečia reikalingą įvedimo formą ir suveda naujus asmeninius dėstytojų duomenis arba koreguoja esamus.

Prieš sąlyga: Dėstytojai dar neregistruoti sistemoje (arba pasikeitė duomenys).

Sužadinimo sąlyga: Atėjo dirbti naujas dėstytojas.

Po sąlyga: Sistemoje registruojami nauji dėstytojai arba pakeičiami esami duomenys.

PA4: Įvesti (koreguoti) modulius.

Vartotojas: Katedra.

Aprašymas: Įvedami nauji moduliai, pasikeitus mokymo planui koreguojami modulių duomenys.

Panaudojimo atvejo scenarijus: Vartotojas prisijungia prie sistemos, iškviečia reikalingą įvedimo formą ir suveda naujus modulius arba koreguoja esamus.

Prieš sąlyga: Pasikeitė studijų centro mokymo planai.

Sužadinimo sąlyga: Atsirado naujos dar neregistruotos studijų programos.

Po sąlyga: Sistemoje registruojami nauji studentai arba pakeičiami esami duomenys.

PA5: Įvesti egzaminų rezultatus.

Vartotojas: Katedra.

Aprašymas: Įvedami įvykusių egzaminų rezultatai.

Panaudojimo atvejo scenarijus: Vartotojas prisijungia prie sistemos, iškviečia reikalingą įvedimo formą ir suveda įvykusių egzaminų rezultatus.

Prieš sąlyga: Semestro rezultatai dar neregistruoti sistemoje.

Sužadinimo sąlyga: Vyksta egzaminų sesija.

Po sąlyga: Sistemoje užregistruojami naujos sesijos egzaminų rezultatai.

PA6: Suformuoti studentų pažangumo ataskaitas.

Vartotojas: Katedra.

Aprašymas: Naudojant sistemos galimybes pagal atitinkamas užklausas formuojamos studentų grupių pažangumo ataskaitos.

Panaudojimo atvejo scenarijus: Vartotojas prisijungia prie sistemos, iškviečia reikalingą užklausą ir suformuoja norimos grupės ataskaitą.

Prieš sąlyga PA5.

Sužadinimo sąlyga: Baigėsi semestras, dekanatui reikia pateikti pažangumo duomenis.

Po sąlyga: Sistema pateikia nurodytą ataskaitą.

PA7: Peržiūrėti informaciją.

Vartotojas: Dekanatas.

Aprašymas: Dekanatas peržiūri savo duomenų bazės duomenis.

Panaudojimo atvejo scenarijus: Vartotojas prisijungia prie sistemos ir išsikviečia reikalingus duomenis.

Prieš sąlyga Dekanatas yra registruotas kaip vartotojas, o reikalingi duomenys egzistuoja duomenų bazėje.

Sužadinimo sąlyga: Dekanatas nori peržiūrėti tam tikrus duomenis.

Po sąlyga: Vartotojui pateikiama reikalinga informacija.

PA8: Peržiūrėti egzaminų rezultatus.

Vartotojas: Dekanatas.

Aprašymas: Dekanatas peržiūri egzaminų sesijos rezultatus.

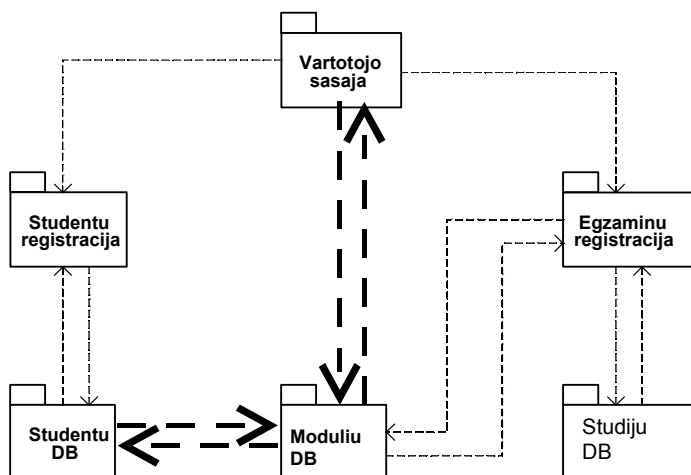
Panaudojimo atvejo scenarijus: Vartotojas prisijungia prie sistemos ir išsikviečia reikalingus duomenis.

Prieš sąlyga: Dekanatas yra registruotas kaip vartotojas, o egzaminų duomenys jau yra duomenų bazėje.

Sužadinimo sąlyga: Vartotojas nori patikrinti egzaminų sesijos duomenis.

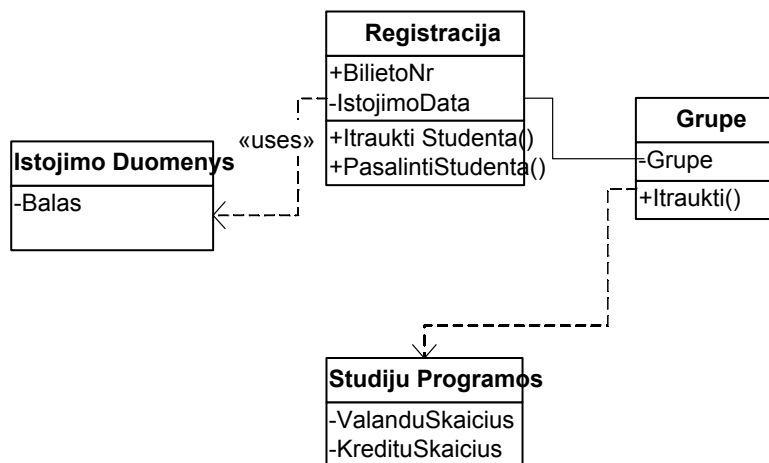
Po sąlyga: Vartotojui pateikiama reikalinga informacija.

3.3.3. Sistemos statinis vaizdas

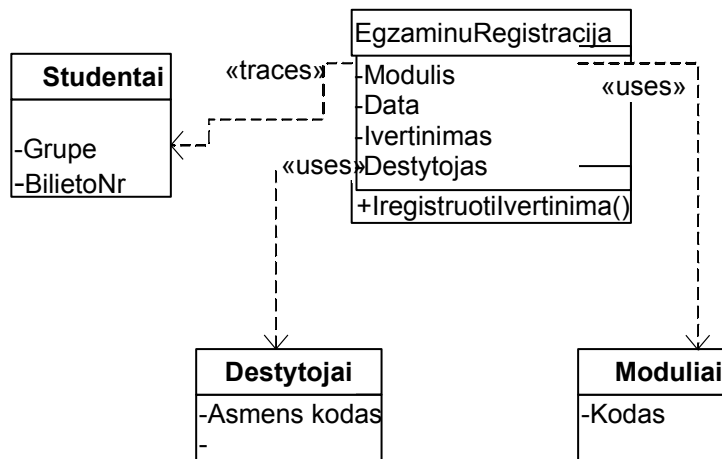


7 pav. Sistemos statinis vaizdas.

Paketų detalizavimas



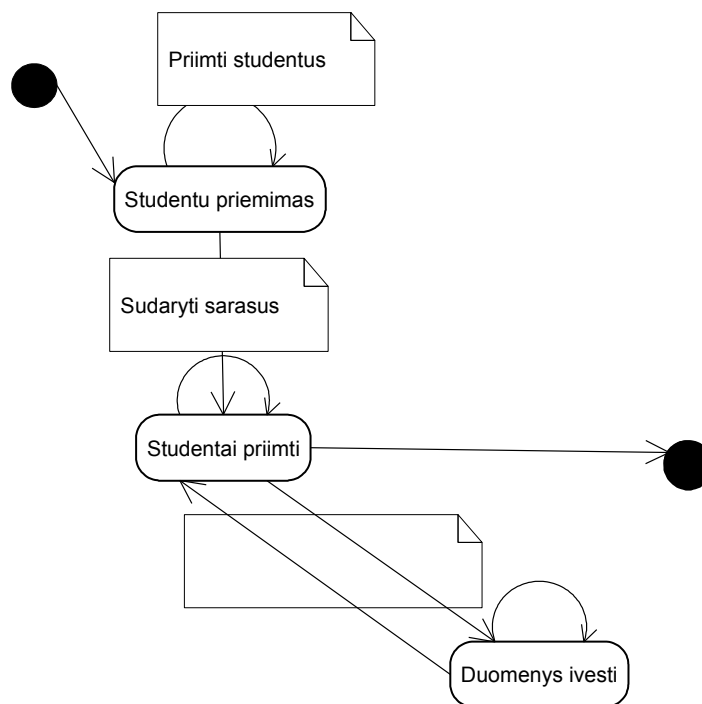
8 pav. Studentų įvedimo paketas.



9 pav. Sesijos rezultatų įvedimo paketas.

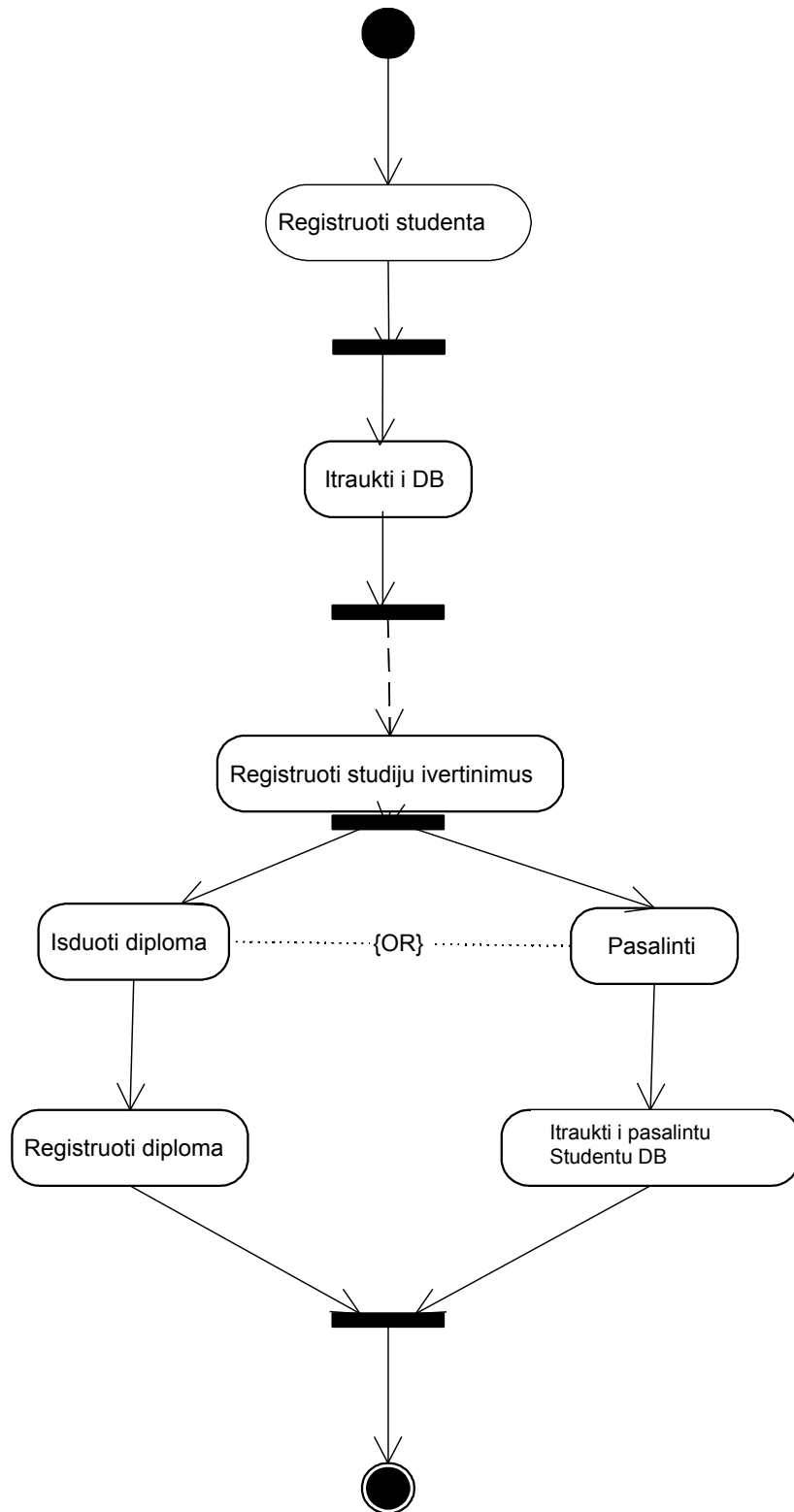
3.3.4. Sistemos dinaminis vaizdas

Šį vaizdą iliustruoja būsenų ir veiklos diagramos. Būsenų diagramos parodo visas būsenas, kuriose gali pabuvoti objektas. Šia diagrama stengiausi pavaizduoti informacijos sistemos studentų registravimo (įvedimo) būseną.



10 pav. Studentų registravimo būsenų diagrama.

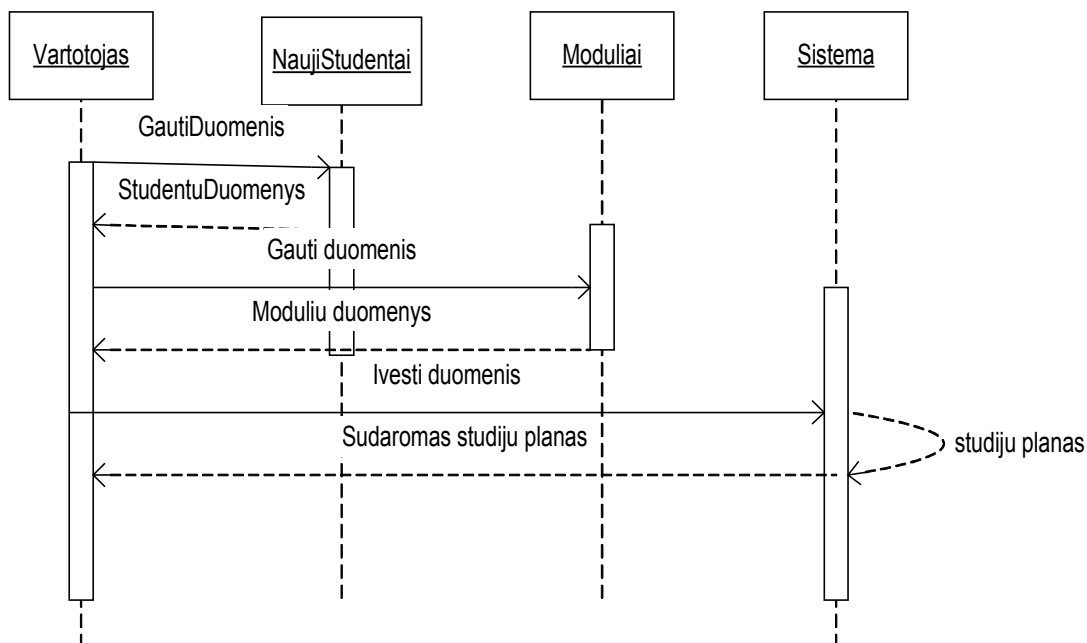
Veiklos diagrama



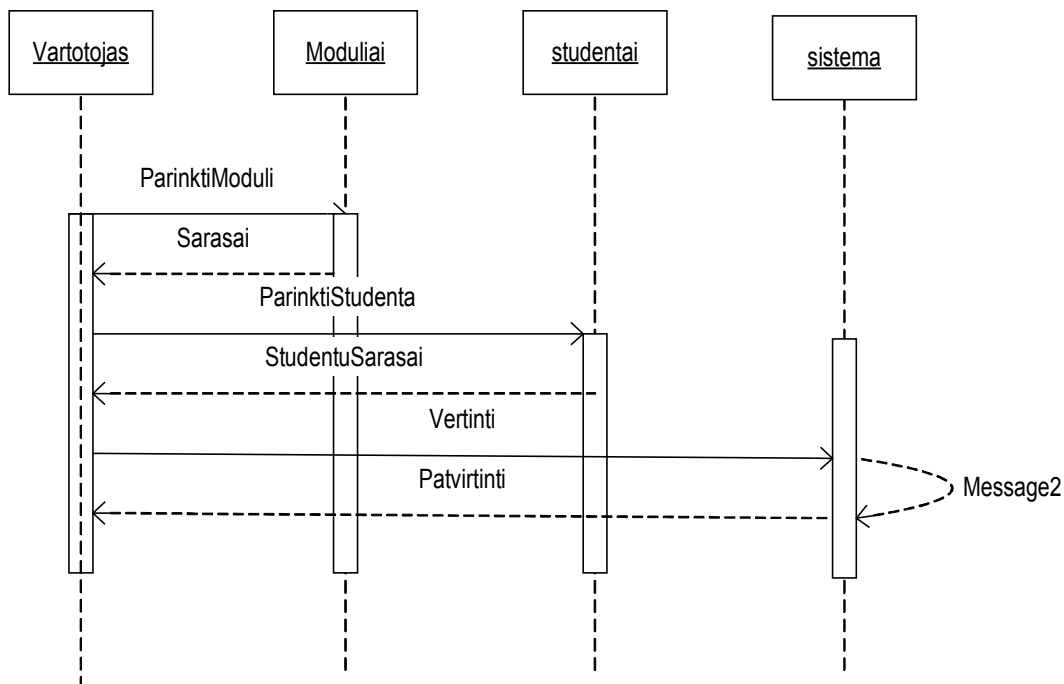
11 pav. Sistemos veiklos diagrama.

Sekų diagramos

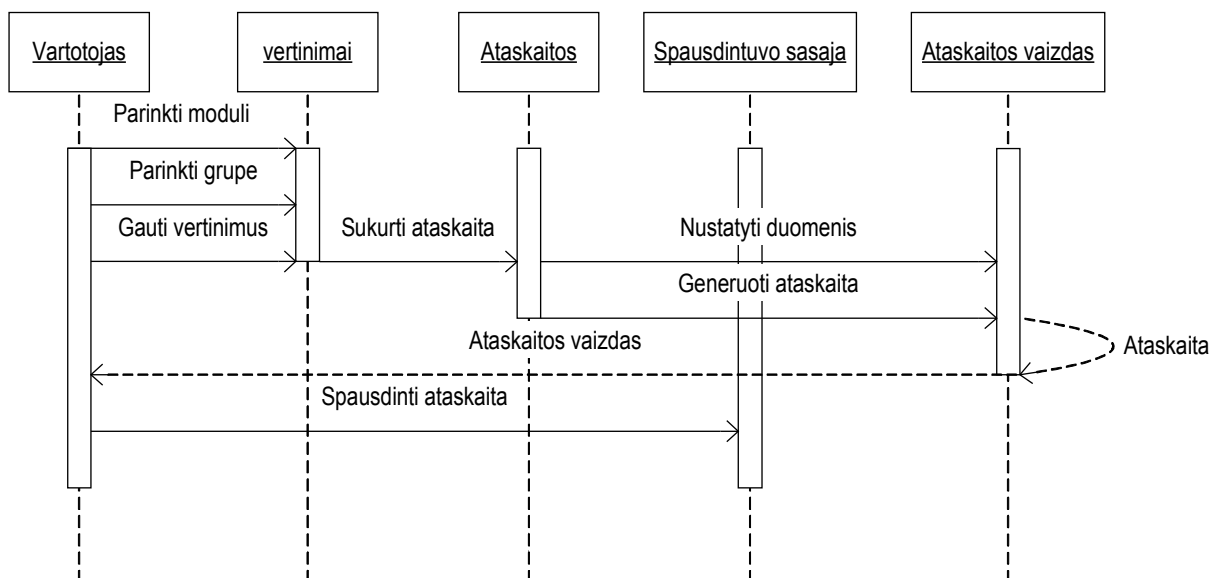
Pagal panaudojimo atvejus pateikiamos atitinkamos sekų diagramos. Kadangi panaudojimo atvejai mažai kuo skiriasi, čia parodau tik tris svarbiausias diagramas.



12 pav. Studentų duomenų įvedimo sekų diagrama.



13 pav. Egzaminų rezultatų įvedimo sekų diagrama.



14 pav. Pažangumo ataskaitos sudarymo sekų diagrama.

3.3.5. Sistemos išdėstymas

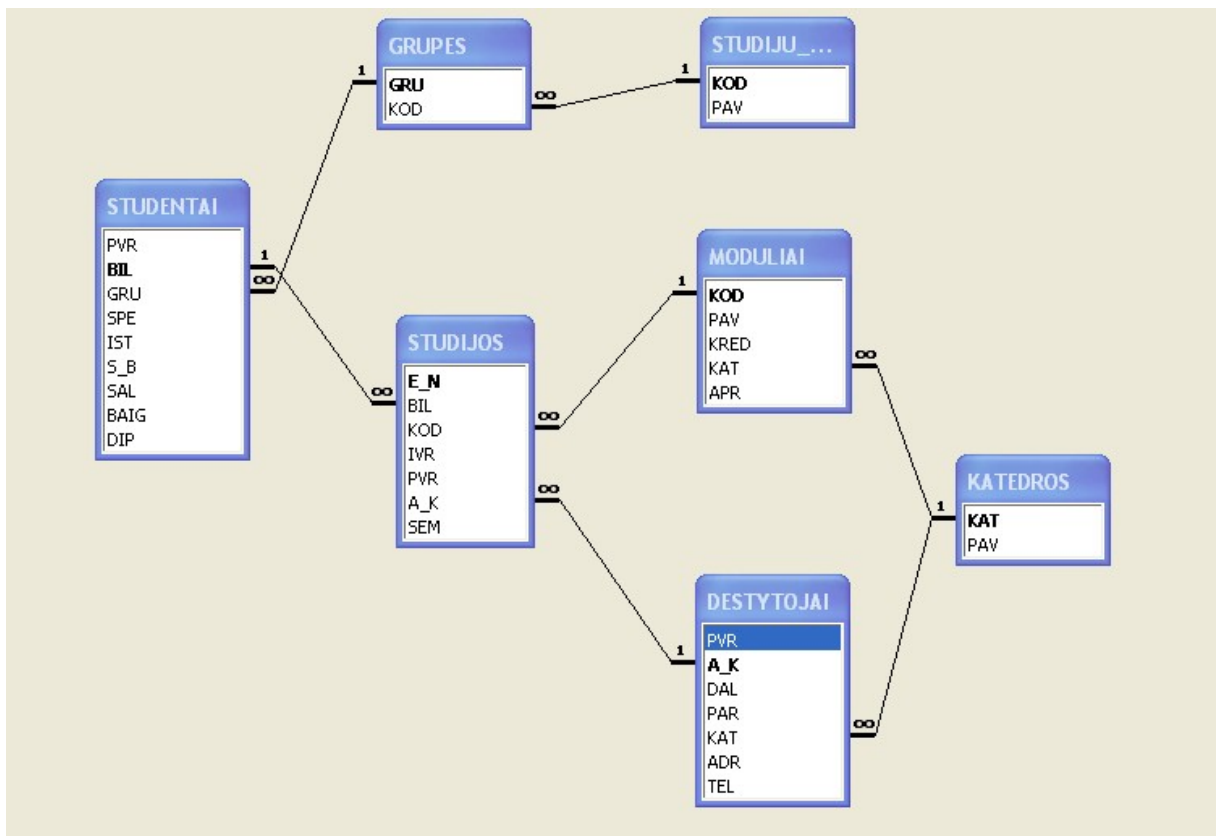
Informacijos sistema įdiegiama studijų centro vietiniame kompiuterių tinkle. Įdiegiamas SQL serveris, po to į serverį „įdedama“ duomenų bazė, ir sistema tampa prieinama keliems vartotojams priklausomai nuo jiems suteiktų teisių. Informacinės sistemos diegimui reikia užtikrinti darbo vietos tinkle ir serverio reikalavimus:

Serveris – ne blogesni nei šie parametrai: 2,6 GHz procesoriaus taktinis dažnis; 512 MB operatyviosios atminties; įdiegta SQL Server programinė įranga.

Vartotojo darbo vieta tinkle – ne blogesni nei šie parametrai: 500 MHz procesoriaus taktinis dažnis; 128 MB operatyviosios atminties; 10 GB kietasis diskas; 100 Mbit tinklo adapteris; įdiegta Windows 9x/Me/Nt/2000/XP operacinė sistema. Monitoriaus rezoliucija turėtų būti 800 x 600, 16 mln. spalvinis režimas ir ne mažesnis nei 85Hz atnaujinimo dažnis.

3.3.6. Duomenų vaizdas

Kadangi pagrindinis elementas MS Access duomenų bazėje yra lentelės, todėl čia pateikiu lentelių ryšių schemą. Pagrindinės lentelės yra trys: **Studentai**, **Studijos** ir **Moduliai**.



15 pav. Duomenų bazės lentelės ir ryšiai.

Lentelėms nustatomi raktiniai laukai ir ryšiai tarp jų. Lentelės **Studentai** raktinis laukas yra **BIL** (Studento bilieto numeris), kuris yra indeksuotas ir neleidžiamos pasikartojančios reikšmės. Šis raktinis laukas siejamas su lentelės **Studijos** analogišku lauku, kur įrašai gali kartotis (studentas laikys ne vieną egzaminą). Lentelės **Moduliai** raktinis indeksuotas laukas yra **KOD**, kur irgi neleidžiamos pasikartojančios reikšmės. Šis laukas siejamas su lentelės **Studijos** analogišku lauku, kuriame įrašai gali kartotis (egzaminą laikys ne vienas studentas), ryšys vienas su daug.

3.4. Detali architektūra

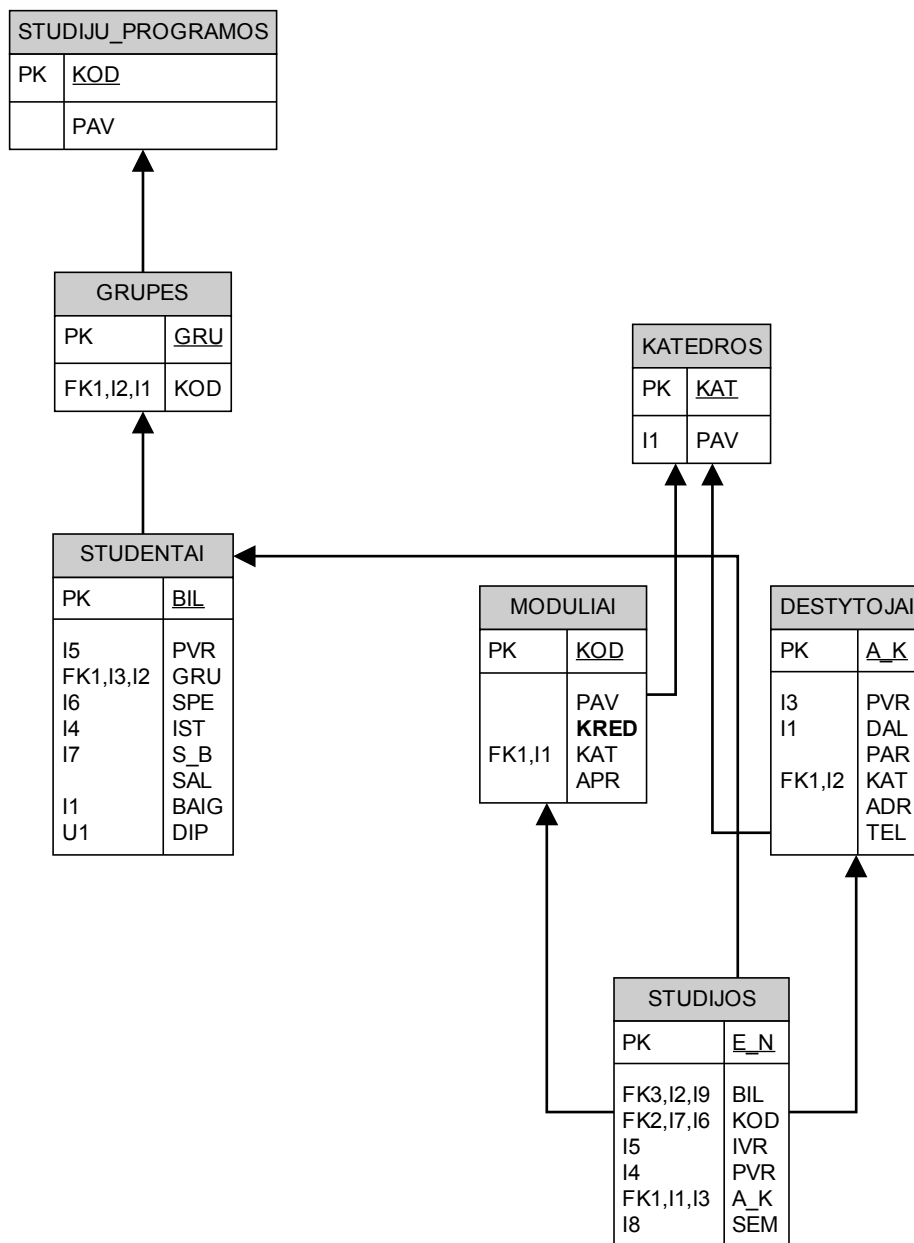
3.4.1. Lentelės

Duomenų bazė yra pagrindinis kuriamos informacijos sistemos elementas, kuris tiesiogiai sąveikauja su visais kitais kuriamos sistemos komponentais: užklausomis, formomis ir ataskaitomis.

Duomenų bazės pagrindas yra lentelės. Komponento paskirtis - kaupti sistemos naudojamus duomenis ir juos saugoti duomenų bazėje. Duomenys turi būti pasiekiami iš bet kurios vietiniame kompiuterių tinkle esančios darbo vietos.

Duomenų bazė saugoma SQL serveryje, esančiame studijų centro kompiuterių tinkle. Duomenys pasiekiami tik registruotiems vartotojams. Norint prisijungti prie duomenų bazės būtina turėti prisijungimo vardą bei slaptažodį.

Struktūra



16 pav. Duomenų bazės struktūra.

Lentelės

Pavadinimas *Grupės.*

Aprašymas Akademiinių grupių pavadinimai. Laukai: GRU- akademinės grupės pavadinimas, tekstinis, įrašai indeksuojami; KOD – studijų programos kodas, tekstinis, indeksuotas. Raktinis laukas – GRU.

Sąsaja Prieinama operatoriui duomenų papildymui ir koregavimui - duomenų peržiūros lange mygtukas „Grupės“ atidaro lentelę.

Resursai SQL serveris.

Sąveikavimas Laukas GRU siejamas su atitinkamu lentelės „Studentai“ lauku, laukas KOD – su lentelės „Studijų programos“ lauku KOD.

Pavadinimas	<i>Studentai.</i>
Aprašymas	Studentų duomenys. Laukai: PVR – pavardė vardas, tekstinis; BIL – studento bilieto numeris, tekstinis, indeksuojamas, pakartojimai negalimi; GRU – akademinė grupė, tekstinis; <i>SPE</i> - specialybė, tekstinis; IST – studijų pradžios metai, skaičius; S_B – stojamasis balas, skaičius; SAL – šalinimo data; BAIG – studijų baigimo metai, skaičius; DIP – diplomo numeris, tekstinis. Raktinis laukas BIL.
Sąsaja	Prieinama operatoriui duomenų papildymui ir koregavimui - duomenų peržiūros lange mygtukas „Studentai“ atidaro lentelę.
Resursai	SQL serveris.
Sąveikavimas	<i>Studentai</i> sąveikauja su lentelėmis <i>Grupės</i> ir <i>Studijos</i> . Laukas GRU siejamas su atitinkamu lentelės <i>Grupės</i> lauku, laukas BIL siejamas su lentelės <i>Studijos</i> analogišku lauku.
Pavadinimas	<i>Dėstytojai.</i>
Aprašymas	Dėstytojų duomenys. Laukai: PVR – tekstinis; A_K – tekstinis, indeksuojamas, pakartojimai negalimi; DAL – dėstomas dalykas, tekstinis; PAR – pareigos, tekstinis; KAT – katedra, tekstinis; ADR – adresas, tekstinis; TEL – telefonas, tekstinis. Raktinis laukas A_K.
Sąsaja	Prieinama operatoriui duomenų papildymui ir koregavimui - duomenų peržiūros lange mygtukas „Dėstytojai“ atidaro lentelę.
Resursai	SQL serveris.
Sąveikavimas	<i>Dėstytojai</i> sąveikauja su lentelėmis <i>Katedros</i> ir <i>Studijos</i> . Laukas KAT siejamas su lauku PAV lentelėje <i>Katedros</i> . Su lentele <i>Studijos</i> sieja laukas A_K.
Pavadinimas	<i>Studijos.</i>
Aprašymas	Egzaminų rezultatai. Laukai: E_N – įrašo eilės numeris, automatinė numeracija, raktinis laukas; BIL – studento bilieto numeris, tekstinis; indeksuojamas, pakartojimai leidžiami; KOD – disciplinos kodas, tekstinis; IVR – įvertinimas, skaičius; A_K – egzaminatoriaus asmens kodas, tekstinis; SEM – semestras, tekstinis.
Sąsaja	Prieinama operatoriui duomenų papildymui ir koregavimui - duomenų peržiūros lange mygtukas „Studijos“ atidaro lentelę.
Resursai	SQL serveris.

Sąveikavimas	<i>Studijos</i> sąveikauja su lentelėmis <i>Studentai</i> ir <i>Dėstytojai</i> ir <i>Moduliai</i> . Laukas BIL siejamas su atitinkamu lauku lentelėje <i>Studentai</i> . Su lentele <i>Dėstytojai</i> siejama per lauką A_K. Su lentele <i>Moduliai</i> siejama per lauką KOD, kur šis laukas yra raktinis ir indeksuotas.
Pavadinimas	<i>Moduliai</i> .
Aprašymas	Dėstomų disciplinų (modulių) įrašai. Laukai: KOD – disciplinos kodas, tekstinis, indeksuojamas, pakartojimai neleidžiami, PAV– pavadinimas, tekstinis, KRED – kreditų skaičius, skaitmeninis; KAT – katedra, tekstinis; APR – aprašymas, memo tipo laukas. Raktinis laukas KOD.
Sąsaja	Prieinama operatoriui duomenų papildymui ir koregavimui - duomenų peržiūros lange mygtukas „Moduliai“ atidaro lentelę.
Resursai	SQL serveris.
Sąveikavimas	<i>Dalykai</i> sąveikauja su lentele <i>Studijos</i> . Laukas <i>KOD</i> siejamas su analogišku lauku lentelėje <i>Studijos</i> .
Pavadinimas	<i>Katedros</i> .
Aprašymas	Katedrų įrašai. Laukai: KAT – katedra, tekstinis; PAV – pavadinimas, tekstinis. Raktinis laukas KAT.
Sąsaja	Prieinama operatoriui duomenų papildymui ir koregavimui - duomenų peržiūros lange mygtukas „Katedros“ atidaro lentelę.
Resursai	SQL serveris.
Sąveikavimas	<i>Katedros</i> sąveikauja su lentelėmis <i>Dėstytojai</i> ir <i>Moduliai</i> . Laukas KAT siejamas su analogiškais laukais.
Pavadinimas	<i>Studijų programos</i> .
Aprašymas	Teikiamų studijų programų įrašai. Laukai: KOD – studijų programos kodas, tekstinis; PAV – pavadinimas, tekstinis. Raktinis laukas KOD.
Sąsaja	Prieinama operatoriui duomenų papildymui ir koregavimui - duomenų peržiūros lange mygtukas „Studijų programos“ atidaro lentelę.
Resursai	SQL serveris.
Sąveikavimas	<i>Studijų programos</i> sąveikauja su lentelėmis <i>Grupės</i> . Laukas KOD siejamas su analogišku lauku.

3.4.2. Įvedimo formos

Formų paskirtis – pagelbėti vartotojui peržiūrėti, surasti, įvesti bei koreguoti duomenis. Formos naudoja duomenis, saugomus lentelėse, arba gaunamus vykdant įvairias užklausas. Šiame darbe visoms lentelėms yra sudarytos įvairios duomenų įvedimo formos.

Pavadinimas *Studentai.*

Klasifikacija Sudėtinė forma.

Aprašymas Forma skirta duomenų įvedimui į lentelę „Studentai“. Naudojami visi lentelės laukai. Peržiūros lange galima pasirinkti iš sąrašo norimus duomenis koregavimui.

Duomenų tikrinimas Nėra.

Sąsaja Prieinama operatoriui duomenų papildymui ir koregavimui - duomenų koregavimo lange mygtukas „Studentai“ atidaro formą.

Sąveikavimas Įrašai išsaugomi lentelėje „Studentai“. Peržiūros lange pateikiamos studentų pavardės ir bilietų numeriai iš lentelės „Studentai“, kitame lange pateikiamas specialybių sąrašas iš lentelės „Studijų programos“.

Apribojimai Nėra.

Pavadinimas *Dėstytojai.*

Klasifikacija Sudėtinė forma.

Aprašymas Forma skirta duomenų įvedimui į lentelę „Dėstytojai“. Naudojami visi lentelės laukai. Peržiūros lange galima pasirinkti iš sąrašo norimus duomenis koregavimui.

Duomenų tikrinimas Laukui A_K (asmens kodas) galioja tikrinimo taisyklė:
 $Len([A_K])=11 \text{ And } (Left([A_K];1)="3" \text{ Or } Left([A_K];1)="4")$. Įspėjimo tekstas: *Patikrinkite asmens kodą.*

Sąsaja Prieinama operatoriui duomenų papildymui ir koregavimui - duomenų koregavimo lange mygtukas „Dėstytojai“ atidaro formą.

Sąveikavimas Įrašai išsaugomi lentelėje „Dėstytojai“. Peržiūros lange pateikiamos dėstytojų pavardės.

Apribojimai Nėra.

Pavadinimas *Studijos.*

Klasifikacija Sudėtinė forma.

Aprašymas Forma skirta duomenų įvedimui į lentelę „Studijos“. Naudojami visi lentelės laukai. Peržiūros languose iš sąrašo parenkamos dėstytojų pavardės ir

disciplinų pavadinimai. Nurodžius akademinę grupę pateikiamas grupės studentų sąrašas.

Duomenų tikrinimas Įvertinimo laukui taikoma tikrinimo taisyklė: skaičius turi būti nuo 0 iki 10. Klaidos atveju pateikiamas įspėjimo tekstas „Negalimas įvertinimas“.

Sąsaja Pasiekama operatoriui duomenų papildymui - duomenų koregavimo lange mygtukas „Studijos“ atidaro formą.

Sąveikavimas Įrašai išsaugomi lentelėje „Studijos“. Pasirinkimo languose pateikiamas studentų sąrašas (lentelė „Studentai“), disciplinų sąrašas (lentelė „Moduliai“), dėstytojų sąrašas (lentelė „Dėstytojai“).

Apribojimai Formoje uždraustas esamų duomenų koregavimas: *Data entry – Yes*.

Pavadinimas *Moduliai*.

Klasifikacija Sudėtinė forma.

Aprašymas Forma skirta duomenų įvedimui į lentelę „Moduliai“. Naudojami visi lentelės laukai. Peržiūros lange matomas modulių sąrašas.

Duomenų tikrinimas Nėra.

Sąsaja Pasiekama operatoriui duomenų papildymui - duomenų įvedimo lange mygtukas „Moduliai“ atidaro formą.

Sąveikavimas Įrašai išsaugomi lentelėje „Moduliai“. Pateikiamas disciplinų sąrašas iš lentelės „Moduliai“.

Apribojimai Formoje uždraustas esamų duomenų koregavimas, galimas tik naujų duomenų įvedimas.

Pavadinimas *Grupės*.

Klasifikacija Sudėtinė forma.

Aprašymas Forma skirta duomenų įvedimui į lentelę „Grupės“. Naudojami visi lentelės laukai. Peržiūros lange matomas studijų programų sąrašas. Pažymėjus įrašą programos kodas įvedamas į reikalingą lauką.

Duomenų tikrinimas Nėra.

Sąsaja Pasiekama operatoriui duomenų papildymui - duomenų įvedimo lange mygtukas „Grupės“ atidaro formą.

Sąveikavimas Įrašai išsaugomi lentelėje „Grupės“. Pateikiamas studijų programų sąrašas iš lentelės „Studijų programos“.

Apribojimai Nėra.

Pavadinimas	<i>Studijų programos.</i>
Klasifikacija	Forma.
Aprašymas	Forma skirta duomenų įvedimui į lentelę „Studijų programos“. Naudojami visi lentelės laukai. Matomi visi esantys įrašai.
Duomenų tikrinimas	Nėra.
Sąsaja	Pasiekama operatoriui duomenų papildymui - duomenų įvedimo lange mygtukas „Studijų programos“ atidaro formą.
Sąveikavimas	Forma sąveikauja su to paties pavadinimo lentele.
Apribojimai	Formoje neleidžiama koreguoti ar šalinti esamus įrašus, galimas tik naujų duomenų įvedimas.

3.4.3. Užklauso

Naudodamas užklauso vartotojas gali iš lentelės ar kelių lentelių išrinkti reikalingus duomenis, atlikti su tais duomenimis įvairius skaičiavimus ir gauti pageidaujamos sandaros duomenų lentelę.

Pavadinimas	<i>Grupės studentai.</i>
Klasifikacija	Atrenkamoji dialoginė užklausa.
Aprašymas	atrenka iš studentų sąrašo pasirinktos grupės studentus ir pateikia sąrašą lentelės forma.
Sąsaja	Pasiekama operatoriui duomenų peržiūrai - lange „Studijos“ mygtukas „Grupės studentai“ atidaro gautų duomenų lentelę.
Sandara	Naudojami lentelės „Studentai“ laukai PVR , BIL ir GRU .
Skaičiavimai	Neatliekami.

Pavadinimas	<i>Sesijos dešimtukai.</i>
Klasifikacija	Atrenkamoji užklausa.
Aprašymas	atrenka iš sesijos rezultatų lentelės dešimties balų įvertinimus ir pateikia sąrašą lentelės forma.
Sąsaja	Pasiekama operatoriui duomenų peržiūrai - lange „Sesija“ mygtukas „Sesijos dešimtukai“ atidaro gautų duomenų lentelę.
Sandara	Naudojami lentelės „Studentai“ laukai PVR ir GRU ., lentelės „Studijos“ laukai IVR ir SEM įvedant atrankos kriterijų.
Skaičiavimai	Neatliekami.

Pavadinimas	<i>Skolininkai.</i>
Klasifikacija	Atrenkamoji užklausa.

Aprašymas	atrenka skolininkus: nelaikiusius egzamino arba gavusius nepatenkinamą pažymį studentus.
Sąsaja	Pasiekiami operatoriui duomenų peržiūrai - lange „Sesija“ mygtukas „Skolininkai“ atidaro gautų duomenų lentelę.
Sandara	Naudojami lentelės „Studentai“ laukai PVR ir GRU. , lentelės „Studijos“ laukai BIL , IVR , PVR , SEM KOD , lentelės „Moduliai“ laukas PAV . Įvedamas atrankos kriterijus.
Skaičiavimai	Neatliekami
Pavadinimas	<i>Edukologijos moduliai.</i>
Klasifikacija	Atrenkamoji užklausa.
Aprašymas	atrenka edukologijos katedros kuruojamas disciplinas.
Sąsaja	Pasiekiami operatoriui duomenų peržiūrai - lange „Studijos“ mygtukas „Edukologijos moduliai“ atidaro gautų duomenų lentelę.
Sandara	Naudojami lentelės „Moduliai“ laukai KOD ir PAV įvedant atrankos kriterijų.
Skaičiavimai	Neatliekami.
Pavadinimas	<i>Filologijos moduliai.</i>
Klasifikacija	Atrenkamoji užklausa.
Aprašymas	atrenka filologijos katedros kuruojamas disciplinas.
Sąsaja	Pasiekiami operatoriui duomenų peržiūrai - lange „Studijos“ mygtukas „Filologijos moduliai“ atidaro gautų duomenų lentelę.
Sandara	Naudojami lentelės „Moduliai“ laukai KOD ir PAV įvedant atrankos kriterijų.
Skaičiavimai	Neatliekami.
Pavadinimas	<i>Pirmakursiai.</i>
Klasifikacija	Atrenkamoji užklausa.
Aprašymas	atrenka pirmakursius studentus.
Sąsaja	Pasiekiami operatoriui duomenų peržiūrai - lange „Studijos“ mygtukas „Pirmakursiai“ atidaro gautų duomenų lentelę.
Sandara	Naudojami lentelės „Studentai“ laukai PVR , GRU , S_B ir IST įvedant atrankos kriterijų.
Skaičiavimai	Neatliekami.

Pavadinimas *Studento rezultatai.*

Klasifikacija Dialoginė atrenkamoji užklausa.

Aprašymas pagal nurodytą bilieto numerį atrenka studento sesijos rezultatus.

Sąsaja Pasiekama operatoriui duomenų peržiūrai - lange „Sesija“ mygtukas „Studento rezultatai“ atidaro dialogo langą.

Sandara Naudojami lentelės „Studentai“ laukai **PVR, BIL**, lentelės „Studijos“ laukai **KOD, IVR, PVR, SEM** nurodant atrankos kriterijų.

Skaičiavimai Neatliekami.

Pavadinimas *Sesijos pirmūnai.*

Klasifikacija Atrenkamoji - skaičiuojamoji užklausa.

Aprašymas atrenka tik dešimtukais besimokančius studentus.

Sąsaja Pasiekama operatoriui duomenų peržiūrai - lange „Sesija“ mygtukas „Sesijos pirmūnai“ atidaro užklauskos rezultatų langą.

Sandara Naudojama pagalbinė užklausa Pag_UA8, kuri iš sesijos rezultatų atrenka tik 10 balų įvertinimus. Ji sudaroma iš lentelių „Studijos“ laukų **BIL, SEM, IVR** ir lentelės „Studentai“ laukų **PVR** ir **GRU**. Užklausa UA8 iš pagalbinės užklauskos rezultatų atrenka tik tuos studentus, kurie visus egzaminus išlaikė dešimtukais.

Skaičiavimai Skaičiavimo lauke naudojama funkcija Count.

Pavadinimas *Diplomantai.*

Klasifikacija Atrenkamoji užklausa.

Aprašymas Atrenka baigusius studijas studentus .

Sąsaja Pasiekama operatoriui duomenų peržiūrai - lange „Studijos“ mygtukas „Diplomantai“ atidaro užklauskos rezultatų langą.

Sandara Naudojami lentelės „Studentai“ laukai **PVR, GRU, BIL** ir **DIP**.

Skaičiavimai Neatliekami.

Pavadinimas *Disciplinų vidurkis.*

Klasifikacija Skaičiuojamoji užklausa.

Aprašymas Iš studijų lentelės paskaičiuoja semestro kiekvienos disciplinos vidurkį.

Sąsaja Pasiekama operatoriui duomenų peržiūrai - lange „Sesija“ mygtukas „Disciplinų vidurkis“ atidaro užklauskos rezultatų langą.

Sandara	Naudojami lentelės „Moduliai“ laukai PAV ir lentelės „Studijos“ laukai IVR, PVR ir SEM .
Skaičiavimai	Skaičiavimo lauke naudojama funkcija Avg.
Pavadinimas	<i>Edukologijos vidurkiai.</i>
Klasifikacija	Atrenkamoji skaičiuojamoji užklausa.
Aprašymas	paskaičiuoja edukologijos katedros kiekvienos disciplinos vidurkį.
Sąsaja	Pasiekama operatoriui duomenų peržiūrai - lange „Sesija“ mygtukas „Edukologijos vidurkiai“ atidaro užklauskos rezultatų langą.
Sandara	Naudojami lentelės „Moduliai“ laukai KOD ir PAV ir lentelės „Studijos“ laukai IVR, PVR ir SEM .
Skaičiavimai	Skaičiavimo lauke naudojama funkcija Avg.
Pavadinimas	<i>Filologijos vidurkiai.</i>
Klasifikacija	Atrenkamoji skaičiuojamoji užklausa.
Aprašymas	paskaičiuoja filologijos katedros kiekvienos disciplinos vidurkį.
Sąsaja	Pasiekama operatoriui duomenų peržiūrai - lange „Sesija“ mygtukas „Filologijos vidurkiai“ atidaro užklauskos rezultatų langą.
Sandara	Naudojami lentelės „Moduliai“ laukai KOD ir PAV ir lentelės „Studijos“ laukai IVR, PVR ir SEM .
Skaičiavimai	Skaičiavimo lauke naudojama funkcija Avg.
Pavadinimas	<i>Studentų skaičius.</i>
Klasifikacija	Atrenkamoji skaičiuojamoji užklausa.
Aprašymas	paskaičiuoja kiek yra studentų kiekvienoje akademinėje grupėje.
Sąsaja	Pasiekama operatoriui duomenų peržiūrai - lange „Studijos“ mygtukas „Studentų skaičius“ atidaro užklauskos rezultatų langą.
Sandara	Naudojami lentelės „Grupės“ laukai GRU ir lentelės „Studentai“ laukai BIL .
Skaičiavimai	Skaičiavimo lauke naudojama funkcija Count.
Pavadinimas	<i>Grupės vidurkis.</i>
Klasifikacija	Dialoginė skaičiuojamoji užklausa.
Aprašymas	paskaičiuoja nurodytos grupės studentų pažangumo vidurkį.

Sąsaja	Pasiekama operatoriui duomenų peržiūrai - lange „Sesija“ mygtukas „Grupės vidurkis“ atidaro užklauso rezultatų langą.
Sandara	Naudojami lentelės „Studentai“ laukai PVR, GRU, BIL ir lentelės „Studijos“ laukai KOD, IVR, SEM .
Skaičiavimai	Skaičiavimo laukuose naudojamos funkcijos Avg ir Count.
Pavadinimas	<i>Sesijos pažymiai.</i>
Klasifikacija	Kryžminė skaičiuojamoji užklausa.
Aprašymas	paskaičiuoja kiek kokių kiekvienos disciplinos pažymių parašyta per sesiją.
Sąsaja	Pasiekama operatoriui duomenų peržiūrai - lange „Sesija“ mygtukas „Sesijos pažymiai“ atidaro užklauso rezultatų langą.
Sandara	Naudojama pagalbinė užklausa Pag_US6, kuri sudaroma iš lentelės „Moduliai“ laukų KOD ir PAV ir lentelės „Studijos“ laukų BIL ir IVR . Užklausa US6 grupuoja duomenis: eilutė – modulis; stulpelis – įvertinimas; Skaičiuojami užpildyti laukai.
Skaičiavimai	Skaičiavimo lauke naudojama funkcija Count.
Pavadinimas	<i>Atrinkti braukiamus.</i>
Klasifikacija	Užklausa padaro duomenų lentelę.
Aprašymas	atrenka studentus, turinčius 2 ir daugiau skolų; duomenys išsaugomi atskiroje lentelėje.
Sąsaja	Pasiekama operatoriui duomenų peržiūrai - lange „Sesija“ mygtukas „Atrinkti braukiamus“ atidaro užklauso rezultatų langą.
Sandara	Naudojami atrenkamosios užklauso UA3 duomenys: laukai PVR, GRU, BIL ir IVR .
Skaičiavimai	Skaičiavimo lauke naudojama funkcija Count nurodant atrankos kriterijų ≥ 2 .
Pavadinimas	<i>Šalinti skolininkus.</i>
Klasifikacija	Šalinimo užklausa .
Aprašymas	Pašalina užklauso UL1 atrinktus įrašus.
Sąsaja	Operatorius atlieka veiksmą lange „Studijos“ mygtuko „Šalinti skolininkus“ paspaudimu.

Sandara	Naudojamos lentelės „Studentai“ ir laikina lentelė „Braukiami“. Iš duomenų bazės pašalinami tie įrašai, kurie sutampa su esančiais lentelėje „Braukiami“.
Skaičiavimai	Neatliekami.

3.4.4. Ataskaitos

Ataskaitos yra specialus DB objektas, padedantis parengti ir vaizdžius tinkamus analizei dokumentus. Ataskaitų duomenų šaltiniai gali būti lentelės ar užklausų rezultatai. Duomenis ataskaitose galima įvairiai rikiuoti bei grupuoti.

Pavadinimas	<i>A1.</i>
Aprašymas	Pateikiami sugrupuoti duomenys apie skolas turinčius studentus. Duomenys rikiuojami pagal akademinę grupę.
Sąsaja	Operatorius atlieka veiksmą „Ataskaitų“ lange mygtuko „Skolininkai“ paspaudimu. Ekrane pateikiamas ataskaitos vaizdas, ataskaitą galima spausdinti naudojant standartines Windows priemones.
Sandara	Naudojami užklausos UA3 duomenys, laukai GRU, PVR, IVR, PAV.
Skaičiavimai	Neatliekami.

Pavadinimas	<i>A2.</i>
Aprašymas	Pateikiami naujai įstojusių studentų duomenys. Duomenys rikiuojami pagal akademinę grupę.
Sąsaja	Operatorius atlieka veiksmą „Ataskaitų“ lange mygtuko „Pirmakursiai“ paspaudimu. Ekrane pateikiamas ataskaitos vaizdas, ataskaitą galima spausdinti naudojant standartines Windows priemones.
Sandara	Naudojami užklausos UA6 duomenys, laukai GRU, PVR, S_B.
Skaičiavimai	Neatliekami.

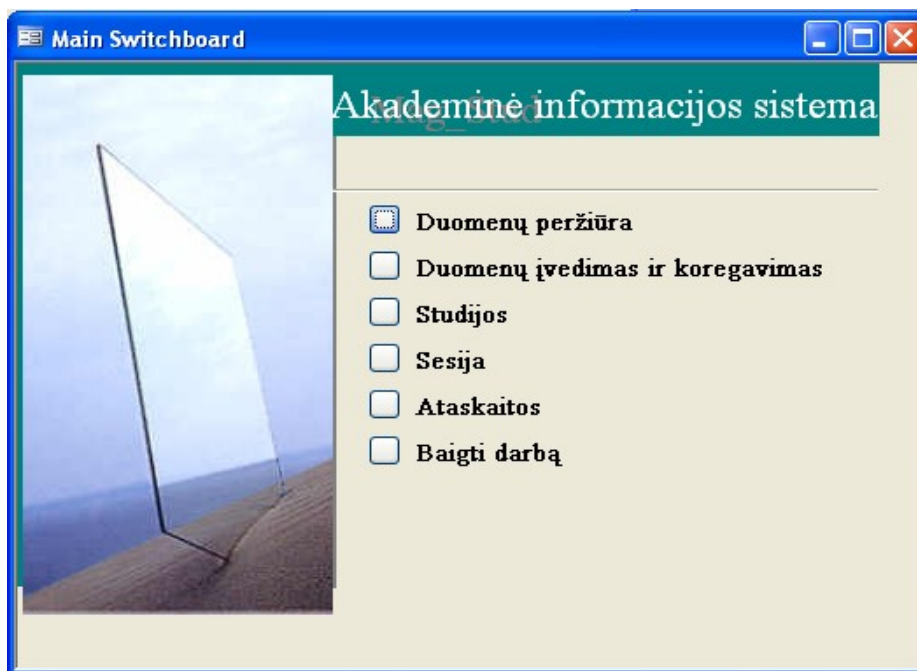
Pavadinimas	<i>A3.</i>
Klasifikacija	Ataskaita.
Aprašymas	Pateikiami visų studentų sesijos rezultatų vidurkiai. Duomenys rikiuojami pagal akademinę grupę.
Sąsaja	Operatorius atlieka veiksmą „Ataskaitų“ lange „Vidurkio“ mygtuko paspaudimu. Ekrane pateikiamas ataskaitos vaizdas, ataskaitą galima spausdinti naudojant standartines Windows priemones.

Sandara Naudojami kryžminės užklauskos **Vidurkiai** duomenys, laukai **GRU, PVR, PAV, IVR**.

Skaičiavimai Neatliekami.

3.4.5. Vartotojo sąsaja

Vartotojo sąsaja realizuojama sudėtine mygtukų forma, kuri valdoma makrokomandomis. Pagrindiniame šios formos lange matome puslapių sąrašą. Kiekvienas puslapis atidaromas atitinkamo mygtuko paspaudimu.



17 pav. Vartotojo sąsajos pagrindinis langas.

Kiekviename puslapyje vėl atidaroma mygtukų forma. Mygtuko paspaudimu atliekamas vienas ar kitas veiksmas: atveriamas duomenų lentelės arba įvedimo formos langas, kur galima atlikti įvairius veiksmus naudojantis standartinėmis MS Access priemonėmis. Taip pat mygtuko paspaudimu galima vykdyti įvairias užklauskas. Išėiti iš programos galima naudojantis sugrįžimo mygtuku, kuris yra kiekviename pagrindinės formos puslapyje, arba naudojantis standartiniais Windows sistemos mygtukais, kurie nėra nuo vartotojo paslėpti.

4. VARTOTOJO DOKUMENTACIJA

4.1. Sistemos funkcinis aprašymas

Kadangi studijų centras nėra savarankiška įstaiga ir visos ūkinės bei buhalterinės funkcijos, o taip pat studentų priėmimas vykdomas centralizuotai, todėl projektuojama informacijos sistema apima tik akademinę įstaigos veiklos pusę.

Akademinė informacijos sistema skirta studijų centro (fakulteto) administracijai ir dėstytojams naudotis mokslo ir studijų tikslais. Informacijos sistemoje pateikiama tiek administracijai, tiek ir dėstytojams reikalinga informacija, kuri pasiekama iš bet kurio įstaigos tinkle esančio kompiuterio.

Akademinė informacijos sistema teikia tokias paslaugas: kaupiama informacija apie esamas studijų programas, studijuojamas disciplinas, dėstytojus ir studentus. Studijų programos identifikuojamos pagal skaitmeninį kodą, suteikiamą SKVC. Studentai identifikuojami pagal studento pažymėjimo numerį, dėstytojai pagal asmens kodą. Disciplinos (moduliai) identifikuojami pagal suteikiamą kodą, (pvz. EBLM001) kur pirmasis simbolis reiškia katedrą (E – edukologijos, F - filologijos), tolimesni du simboliai nurodo modulio paskirtį (BL – bendrojo lavinimo, KV – kvalifikacinis, PA – pasirenkamas), toliau yra modulio numeris (M001). Moduliai (disciplinos) irgi klasifikuojami: bendrojo lavinimo disciplinos turi skaičius nuo 1 iki 100, kvalifikaciniai dalykai žymimi skaičiais nuo 101 iki 200, pasirenkamieji nuo 201 iki 300, ir praktikos nuo 301 iki 400. Tokia sistema leidžia taikyti įvairius atrankos kriterijus ir filtrus, norint vienaip ar kitaip pateikti informaciją.

Didžiausią informacijos sistemos dalį užima studentų duomenys. Įvedant naujai įstojusius studentus nurodomas stojamasis balas. Pagal šiuos duomenis galima palyginti įstojusių į vieną ar kitą specialybę studentų lygį, taip pat palyginti skirtingų metų stojamuosius balus. Informacijos sistemos duomenų bazėje kaupiami duomenys apie visus studentų egzaminų ir įskaitų įvertinimus, nurodant modulio (disciplinos) kodą, egzaminuojančio dėstytojo pavardę ir semestrą. Iš šių duomenų galima nustatyti kiekvienos studentų akademinės grupės semestro pažangumo vidurkį, nustatyti pirmūnus arba tik gerais pažymiais besimokančius studentus, nustatyti skolininkus (nelaikiusius ar neišlaikiusius egzamino), jei studentas yra išbraukiamas ar meta studijas, yra numatyta galimybė pašalinti skolininkus iš duomenų bazės. Baigusiems studijas studentams įvedamas išduodamo diplomo numeris, tada galima baigusius studentus atrinkti ir jų duomenis saugoti atskiroje lentelėje, o iš aktyvios duomenų bazės pašalinti. Taip pat galima paskaičiuoti kiekvieno modulio (disciplinos) įvertinimų vidurkį, paskaičiuoti kiek ir kokių įvertinimų parašė kiekvienas dėstytojas per sesiją arba kiek ir kokių kiekvienos disciplinos įvertinimų buvo parašyta per

sesiją. Visi šie duomenys pateikiami ekrane skaitymui arba juos galima spausdinti, o taip pat yra galimybė suformuoti ataskaitą skaitymui arba spausdinimui.

Duomenų bazei sudaryti reikalinga suvesti tokius duomenis:

Studijų programos – kodas ir pavadinimas.

Grupės – akademinė grupė ir studijų programos kodas.

Moduliai – kodas, pavadinimas, kreditų skaičius, aprašymas.

Dėstytojai – pavardė vardas, asmens kodas, atestuota sritis, pareigos, katedra, adresas ir telefonas.

Studentai – pavardė vardas, studento pažymėjimo numeris, akademinė grupė, studijuojama specialybė, įstojimo metai, stojamasis balas, baigimo metai, diplomo numeris.

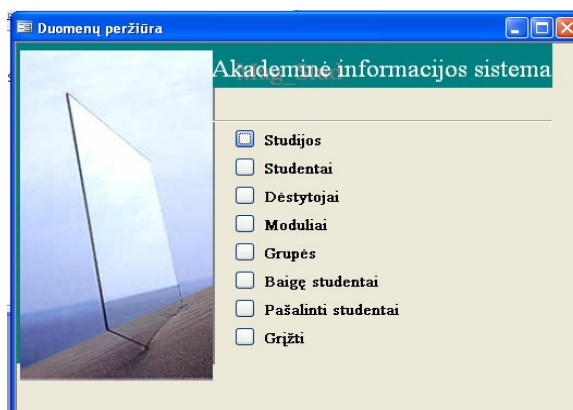
Studijos - studento pažymėjimo numeris, disciplinos kodas, įvertinimas, egzaminatorius, semestras.

Duomenų įvedimui yra sukurtos atitinkamos formos, kas žymiai palengvina operatoriaus darbą.

Akademinė informacijos sistema kuriama Kauno kolegijos Kėdainių Jonušo Radvilos studijų centrui ir derinantis prie jo poreikių, tačiau sukurtos duomenų bazės rėmus galima pritaikyti bet kurios Lietuvos kolegijos fakultetui, padarius atitinkamus pakeitimus ir suvedus savo įstaigos duomenis. Duomenų bazės instaliuoti nereikia, ji puikiai veiks bet kuriame kompiuteryje, turinčiame įdiegtą MS Access paketą. Adaptavimas konkrečiai aplinkai nebūtų labai sudėtingas: reikėtų pagal įstaigos poreikius peržiūrėti duomenų bazės struktūrą, lentelių tarpusavio ryšius. Iš kitos pusės, pritaikyti svetimą produktą ne visuomet paprasta, gal paprastesnis sprendimas būtų sukurti savo produktą pagal savus poreikius.





4.2. Sistemos vadovas

Vartotojo sąsaja yra sudėtinė mygtukų forma, valdoma makrokomandomis. Startiniame sistemos lange vartotojas turi veiksmų pasirinkimą: Kiekvienas mygtukas atidaro atskirą formos puslapį, kur atliekami vienokie ar kitokie veiksmai.

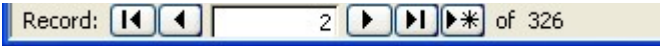


18 pav. Vartotojo sąsajos duomenų peržiūros langas.



4.2.1. Duomenų peržiūra


Duomenų peržiūros lange vartotojui pateikiamas visų lentelių pasirinkimas: galima peržiūrėti ir koreguoti duomenis, filtruoti, rikiuoti, šalinti įrašus. Šiems veiksams atlikti naudojamos standartinės MS Access priemonės. Duomenų filtras nustatomas naudojant meniu komandas **Records** → **Filter**. Filtras nuimamas (pašalinamas) atlikus meniu komandas **Records** → **Remove Filter/Sort**. Duomenų rikiavimas (rūšiavimas) atliekamas įrankių juostoje esančių mygtukų  arba  paspaudimu. Įrašus rikiuoti galima bet kuriame lauke (stulpelyje). Norint rikiuoti įrašus, reikia aktyvuoti norimą lauką (spragtelėti tame lauke pelytę), ir tada spausti norimą mygtuką. Pirmasis surikiuos įrašus didėjimo tvarka, o antrasis – mažėjimo. Norint pašalinti įrašą reikia spausti pelytę patį pirmąjį kvadratinį laukelį, kuriame yra juodas trikampis kairėje duomenų lentelės pusėje  + **Medelytė Jurgita**. Tuomet šis įrašas išrenkamas ir jį galima pašalinti spaudžiant klavišą „Delete“ arba įrankių juostoje esantį mygtuką . Velkant pelytę žemyn išpaudus klavišą galima pažymėti kelis įrašus.

Kiekvieno duomenų lango apačioje yra išdėstyti standartiniai MS Access mygtukai,

 kuriais naudojantis galima keliauti po sąrašą žemyn aukštyn. Paspaudę juodą trikampėlį persikelsime per vieną įrašą pirmyn arba atgal, o paspaudę trikampį su brūkšneliu persikelsime per visą lango puslapį. Skaičius 2 laukelyje per vidurį rodo įrašo eilės numerį, o skaičius dešiniau rodo, kiek yra įrašų iš viso. Paspaudus trikampį su žvaigždute persikelsime į lentelės apačią, kur bus atidaryta tuščia eilutė naujam įrašui.

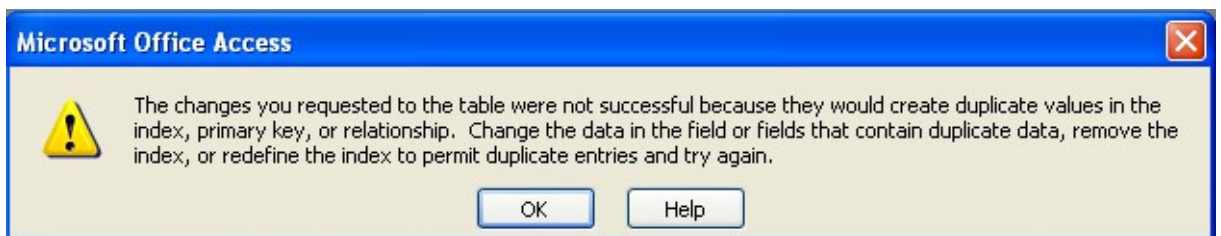
4.2.2. Duomenų įvedimas ir koregavimas

Šiame lange vartotojui pateikiamos visos įvedimo formos. Beveik visos formos leidžia koreguoti duomenis tiesiog sąrašė: tai formos „Dėstytojai“, „Studentai“, „Studijų programos“, „Grupės“. Formoje esančiame išsiskleidžiančiame sąrašė pelytę pasirenkame norimą įrašą, tuomet visas įrašo turinys pateikiamas formos laukuose. Bet kurią lauką galima koreguoti, tokiu atveju pataisytas įrašas bus įrašomas į lentelę. Norint įvesti naują įrašą reikia spausti mygtuką su žvaigždute formos apačioje , tuomet beveik visi formos laukai bus tušti, o kai kuriuose bus matoma įvesta numatytoji reikšmė (default value), pvz. Studentų įvedimo formoje laukuose „Įstojimo metai“ ir „Baigimo metai“ matysime atitinkamai „2005“ ir „2010“. Esamas reikšmes galima redaguoti, jei jos netinkamos. Mygtuko  paspaudimas uždaro formą.

Forma „Studijos“ neleidžia koreguoti ar taisyti įrašų. Ši forma atidaroma tokiu režimu, kai leidžiama tik įvesti naujus įrašus, o esamų įrašų peržiūra draudžiama - formos savybei „Data Entry“ (duomenų įvedimas) nustatyta reikšmė „taip“. Paleidus šią formą, pirmiausia reikia spragtelėti šį lauką **Parinkite studentą** , tuomet atsidaro dialogo langas su užrašu „Įveskite akademinę grupę“. Įvedus grupės numerį atsidaro grupės studentų sąrašas, kuriame reikia nurodyti studento pavardę. Analogišku būdu įvedame reikalingus duomenis ir kituose disciplinų ir dėstytojų pasirinkimo laukuose. Lauke įvertinimas įrašome pažymį nuo 0 iki 10. Klaidos atveju sistema išduoda pranešimą „Negalimas įvertinimas“. Įvertinimo lauką užpildyti būtina, priešingu atveju sistema neleis užbaigti darbo.

Duomenų tikrinimo taisyklė dar yra taikoma lentelėje „Dėstytojai“ įvedant asmens kodą. Pirmasis simbolis turi būti skaitmuo 3 arba 4, ir įrašyti galima 11 skaitmenų. Klaidos atveju sistema išduoda pranešimą „Patikrinkite asmens kodą“.

Vartotojo klaidos įmanomos ir tikėtinos įvedant duomenis į taip vadinamus raktinius indeksuotus laukus, kur neleidžiamos pasikartojančios reikšmės: tai studento pažymėjimo numeris, dėstytojo asmens kodas, modulio (disciplinos) kodas, studijų programos kodas, akademinės grupės kodas. Kiekvienu atveju vartotojui įvedus pasikartojančią reikšmę, sistema išduoda tokį pranešimą:



Tai reiškia, kad lygiai toks pat įrašas jau yra įvestas, todėl reikia arba keisti įrašą, arba lentelės projektavimo režime pakeisti indeksavimo nuostatas, kad pasikartojančios reikšmės būtų leidžiamos, arba visai nuimti indeksavimą. Tokius nustatymus gali atlikti tik sistemos administratorius, todėl vartotojui belieka tik pirmasis variantas. Kol tai nebus padaryta, sistema neleis užbaigti darbo ar pereiti prie kitos užduoties.

Bet kokių atveju padaręs klaidą vartotojas visuomet turi galimybę pereiti į duomenų peržiūros langą, atitinkamoje lentelėje surasti reikalingą įrašą ir jį pataisyti arba ištrinti. Vartotojui neprieinamas tik lentelių, formų ir užklausų projektavimo režimas – tuos dalykus tvarko sistemos administratorius.

4.2.3. Studijų, sesijos ir ataskaitų langai

Visi šių langų mygtukai valdomi makrokomandomis. Kiekviename lange yra mygtukas „Grįžti“, kuris sugrąžina į pagrindinį langą. Lange „Studijos“ vykdomos makrokomandos :

1. Grupės studentų sąrašas – vykdoma dialoginė atrenkamoji užklausa įvedant reikalingus parametrus (grupės pavadinimą). Duomenys pateikiami ekrane lentelės forma.
2. Edukologijos katedros moduliai – vykdoma atrenkamoji užklausa, duomenys pateikiami ekrane lentelės forma.
3. Filologijos katedros moduliai – vykdoma atrenkamoji užklausa, duomenys pateikiami ekrane lentelės forma.
4. Pirmakursiai - vykdoma atrenkamoji užklausa, duomenys pateikiami ekrane lentelės forma.
5. Studentų skaičius grupėse – vykdoma skaičiuojamoji atrenkamoji užklausa, duomenys pateikiami ekrane lentelės forma.
6. Diplomantai – vykdoma atrenkamoji užklausa, sukuriama nauja lentelė.
7. Braukiami studentai - vykdoma atrenkamoji skaičiuojamoji užklausa, duomenys įrašomi į naują laikiną lentelę.

Lange „Sesija“ vykdomos šios makrokomandos:

1. Dešimtukai – vykdoma atrenkamoji užklausa, duomenys pateikiami ekrane lentelės forma.
2. Skolininkai - vykdoma atrenkamoji užklausa, duomenys pateikiami ekrane lentelės forma.
3. Studento rezultatai – vykdoma atrenkamoji užklausa įvedant reikalingus parametrus (studento pažymėjimo numerį), duomenys pateikiami ekrane lentelės forma.
4. Pirmūnai – vykdoma atrenkamoji užklausa, duomenys pateikiami ekrane lentelės forma.
5. Dalykų vidurkis - vykdoma skaičiuojamoji užklausa, duomenys pateikiami ekrane lentelės forma.
6. Grupės vidurkis - vykdoma skaičiuojamoji užklausa, duomenys pateikiami ekrane lentelės forma.
7. Sesijos įvertinimai - vykdoma atrenkamoji užklausa, duomenys pateikiami ekrane lentelės forma.

Ataskaitų lange vykdomos makrokomandos:

1. Skolos – ekrane pateikiama sumaketuota ataskaita, ją galima spausdinti naudojant standartines MS Access priemones.
2. Pirmakursiai - ekrane pateikiama sumaketuota ataskaita.
3. Sesijos vidurkiai - ekrane pateikiama sumaketuota ataskaita.

Kiekvienu atveju vartotojo veiksmai neturi sukelti sistemos kritinės būsenos, nes makrokomandos tiksliai apibrėžia, kokią užklausą ar ataskaitą vykdo mygtuko paspaudimas. Bet koku atveju, atsiradus kažkokiems nenumatytiems nukrypimams, sistema išduoda išpėjimąjį pranešimą, pavyzdžiui, tokį:

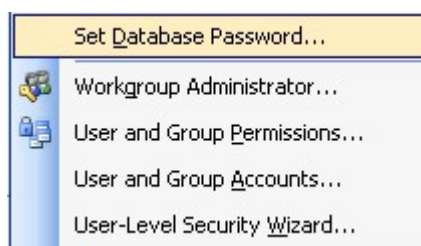


Komanda nevykdoma. Tokiu atveju reikia kreiptis į sistemos administratorių.

4.3. Sistemos instaliavimas

Sistemos instaliuoti nereikia, ji veiks tiesiog įkėlus visą DB į kompiuterį ir paleidus MS Access programą.

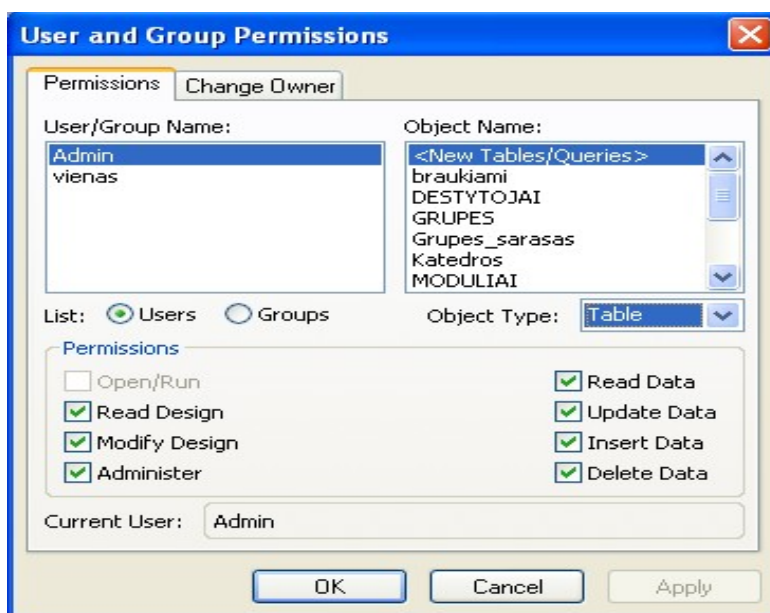
Sukurtai duomenų bazei reikia įvesti įvairių vartotojų lygių apsaugos sistemą. Vartotojams suteikiami identifikavimo vardai ir slaptažodžiai. Vartotojų grupei (ar grupėms) nustatomi apribojimai. Visa informacija apie slaptažodžius ir apribojimus užrašoma į specialų failą, kuris prieinamas tik administratoriams. Priemonės įvairių lygių apsaugos sistemai nustatyti atveriamos per meniu komandą **Tools**→**Security**.



Šiame meniu komanda „Set Database Password“ naudojama duomenų bazės atidarymo slaptažodžiui nustatyti. Kad galima būtų tą padaryti, duomenų bazė turi būti paleista (atidaryta) išskirtiniu būdu.

Eilutė „Workgroup Administrator“ leidžia prisijungti prie vartotojų grupės (įsirašyti į darbo grupės informacijos failą) arba sukurti naują informacijos failą naujiems vartotojams.

Komanda „User and Group Permissions“ leidžia atskiriems vartotojams ar vartotojų grupėms nustatyti jų teises, t. y. kas jiems leidžiama ir ko neleidžiama daryti.



19 pav. Vartotojų teisių nustatymo langas.

Kairiajame langelyje parinkus vartotojo ar grupės vardą dešiniajame lange „Object Name“ galima parinkti atskirai kiekvieną DB elementą arba padaryti nustatymus, galiojančius visai DB (laukelyje „Object Type“ išsiskleidžiančiame sąraše galima pasirinkti „Database“). Leidimai atlikti vienokius ar kitokius veiksmus vartotojui ar grupei nustatomi sudedant varneles atitinkamuose laukeliuose lango apatinėje dalyje.

Komanda „User and Group Accounts“ žemiau esančiame paveikslėlyje leidžia sukurti naujus vartotojus ar jų grupes. Kiekvienam vartotojui ar grupei suteikiamas vardas ir identifikacijos kodas (tarsi slaptažodis). Šiame lange sukurtus vartotojus galima suskirstyti į grupes.



20 pav. Vartotojų grupių nustatymo langas.

4.3.1. Kliento – serverio architektūra

Kaip perkelti MS Access DB į SQL serverį:

Norint jau sukurtos DB duomenis perkelti į SQL serverį, reikia sukurti tuščią duomenų bazę SQL serveryje.

Prisijungti iš MS Access aplinkos prie SQL serverio, tai atliekama meniu punkte **File** → **Connection**.

MS Access DB esantys duomenys perkeliama į SQL Serveryje sukurtą tuščią duomenų bazę: meniu punktas **File** → **Get external data** → **Import** nurodyti sukurtą Access duomenų bazę. Gausime DB importo dialogo langą.

Pažymėti visus perkeliamus objektus: lenteles, formas, ataskaitas ir t.t. Paspaudus OK, visi pažymėti objektai sukeliama į naują vietą.

Užklausų perkėlimas. Užklausa (**Query**) yra Access aplinkoje, tuo tarpu MS SQL serveryje užklausų nėra, užklausa čia atitinka vaizdai **View**. Todėl užklausų automatiškai į SQL serverį perkelti neįmanoma. Taigi kiekvienai Access užklausiai reikia naujajame projekte sukurti analogišką objektą **View**. Patogiausia užklausa tiesiog nukopijuoti (iš **Query** SQL kodo peržiūros režimo lango) ir perkelti į naujojo projekto **View**. Gautą vaizdą (**View**) išsaugoti tuo pačiu vardu, kaip buvo vadinama užklausa.

Atlikus šiuos veiksmus, duomenų bazė jau yra MS SQL serveryje. Taip gaunama kliento-serverio architektūros IS, kuri integruoja du programų paketus MS Access ir SQL Server. Tokia IS leidžia vienu metu keliems vartotojams, dirbantiems MS Access aplinkoje, naudoti duomenis iš vienos DB, esančios SQL serveryje.

5. KOKYBĖS ĮVERTINIMAS

Informacijos sistemos kokybė bus vertinama sistemos testavimo metu įdiegiant kliento – serverio architektūrą bei vertinant vartotojų atsiliepimus (anketos vartotojams).

6. IŠVADOS

1. Analizuodamas vartotojo poreikius nustaciau informacijos sistemai keliamus reikalavimus bei uždavinius.
2. Palyginus duomenų bazių valdymo sistemas pasirinktas tinkamiausias variantas mažai organizacijai - MS Access: žinomas produktas, nesunku dirbti, licencijuota programa.
3. Suprojektavau ir sukūriau reliacinę duomenų bazę: ją sudaro 7 pagrindinės lentelės, 7 įvedimo formos, 25 įvairios užklausos, o taip pat ataskaitos ir makrokomandos. Vartotojo sąsaja – sudėtinė mygtukų forma.
4. Įdiegiant sistemą numatoma realizuoti kliento – serverio architektūrinį sprendimą: panaudoti MS SQL serverį, kad vienu metu dirbti su sistema galėtų keli vartotojai.

7. LITERATŪRA

1. Adomavičius J., Sekliuckis V. Ir kt. Informatika (I dalis): Vadovėlis. Kaunas: Technologija, 2001. -227 p.
2. Baronas R. Duomenų bazių sistemos. Vilnius: TEV, 2002. -126 p.
3. Boguslauskas V., Paradauskas B., Šakys V. Automatizuotos įmonių valdymo sistemos. Vilnius: Mokslas, 1983. -132 p.
4. Butkienė R. Informacijos sistemų projektavimas Oracle Designer/2000 priemonėmis. Kaunas: Technologija, 1998. -165 p.
5. Butleris R. Informacijos sistemų konstravimas FoxPro priemonėmis: Vadovėlis. Kaunas: Technologija, 1997. -199 p.
6. Duomenų bazės. Informacinių technologijų institutas, Vilnius: Žara, 2001. -120 p.
7. Gudas S. Organizacijų veiklos modeliavimas: Vadovėlis.- Kaunas: Technologija, 2001. -134 p.
8. Gudas S. Veiklos analizė ir informacinių poreikių specifikavimas. Kaunas: Naujasis lankas, 2002. – 93 p.
9. Gudas S., Garšva G., Sekliuckis V. Duomenų bazės: Mokymo knyga. Kaunas: Naujasis lankas, 2001. -96p.
10. Microsoft Access duomenų bazės aprašymas (2005), Internetė: <http://www.microsoft.com/lietuva/officexp/programs/access/>
11. Microsoft SQL Server/2005 serverio aprašymas (2005), Internetė: <http://www.microsoft.com/sql/evaluation/overview/>
12. MySQL duomenų bazės aprašymas (2005), Internetė: <http://www.mysql.com>
13. PostgreSQL duomenų bazės aprašymas (2005), Internetė: <http://www.postgresql.org>
14. Sekliuckis V., Gudas S., Garšva G. Informacijos sistemos ir duomenų bazės: Vadovėlis. Kaunas: Technologija, 2003. -338 p.
15. Simanauskas L. Informacinės sistemos: Vadovėlis. Vilnius: VU leidykla, 2000. -290 p.

8. TERMINŲ IR SANTRUMPŲ ŽODYNAS

DB – duomenų bazė (kūrinių, duomenų arba kitokios medžiagos susistemintas ar metodiškai sutvarkytas rinkinys, kuriuo galima individualiai naudotis elektroniniu ar kitu būdu, išskyrus kompiuterių programas, naudojamas tokių duomenų bazėms sukurti ar valdyti).

DBVS – duomenų bazių valdymo sistema.

GUI – grafinė varotojo sąsaja (Graphic User Interface) (menu scenarijus, langai).

HTTP – kompiuterinio tinklo protokolas duomenims perduoti (Hyper Text Transfer Protocol).

PA - panaudojimo atvejai.

RUP - apibendrintas projektavimo metodas (Rational Unified Process).

Specifikacija - sistemos funkcionalumo aprašymas formaliais metodais.

UML – unifikuota modeliavimo kalba (Unified Modeling Language).

DFD – duomenų srautų diagrama (Data Flow Diagram).

IS – informacijos sistema.

SQL – struktūrizuota užklausų kalba (Structured Query Language).

9. SUMMARY

In modern informational society the accumulation and processing of the data is becoming more and more important. Nowadays hardly any organization can function without an information system. The user's requirements for an information system are being discussed while working as well as the basis of its design.

The basis of information system is the database. In this paper the database management systems are briefly discussed. The majority of them are very powerful and small institutions simply are not able to acquire them. The types of the database are also analysed in the paper. Nowadays the relational database is the most popular. The paper is concentrated on the MS Access database, which is dedicated for a small educational institution. The DB and its architecture is being analysed using UML. The possibility to implement the client - server architecture while using MS SQL is also discussed in the paper.