

KAUNO TECHNOLOGIJOS UNIVERSITETAS  
INFORMATIKOS FAKULTETAS  
INFORMACIJOS SISTEMŲ KATEDRA

Tomas Zemblys

**Paskirstytų duomenų bazių  
replikavimo komponentas**

Magistro darbas

Darbo vadovas

doc. dr. B. Paradauskas

Kaunas  
2005

# Turinys

ĮVADAS .....	4
1. VARTOTOJO POREIKIŲ ANALIZĖ .....	6
1.1. Pragmatiškai motyvuotų kilpų panaudojimas .....	6
1.2. UML panaudojimas .....	8
1.3. Organizacijos veiklos analizė panaudojant UML .....	8
1.3.1. Organizacijos veikla .....	8
1.3.2. Veiklos tikslų modelis .....	8
1.3.3. Veiklos panaudojimo atvejų modelis .....	9
1.3.4. Veiklos objektų modelis .....	10
1.3.5. Organizacinės struktūros modelis .....	11
1.3.6. Procesų veiklos diagrama .....	11
1.3.7. Vartotojų bei jų poreikių analizė .....	12
1.4. Organizacijos veiklos vaizdavimas komunikacinėmis kilpomis .....	12
1.4.1. Komunikacinės priklausomybės tarp aktorių .....	12
1.4.2. Pragmatinės priklausomybės .....	13
1.5. Vartotojo poreikių analizės dalies išvados .....	13
2. SISTEMOS ARCHITEKTŪRINIAI SPRENDIMAI .....	15
2.1. Galimi architektūriniai sprendimai .....	15
2.2. XML srautų panaudojimo aktualumas .....	16
2.3. XML technologijos duomenų bazių valdymo sistemose .....	18
2.3.1. Oracle duomenų bazė ir XML panaudojimo galimybės .....	18
2.3.2. MS SQL Server duomenų bazė ir XML panaudojimo galimybės .....	19
2.3.3. Oracle ir MS SQL Server teikiamų XML savybių palyginimas .....	20
2.4. Replikavimo priemonė MS SQL serveryje .....	22
2.5. Sistemos architektūrinių sprendimų dalies išvados .....	23
3. XML PANAUDOJIMAS DUOMENŲ BAZIŲ SISTEMŲ REPLIKAVIME .....	25
3.1. Techninė užduotis .....	25
3.2. Kuriamos sistemos darbo proceso etapai .....	27
3.3. Replikavimo srauto išskyrimas .....	27
3.4. Projektavimo metodų, priemonių parinkimas .....	28
3.5. Kompiuterizuojamos sistemos varianto parinkimas .....	29
3.5.1. Sudėties ir architektūros pasirinkimas .....	29
3.5.2. Programavimo ir projektavimo įrankių pasirinkimas .....	29

3.5.3. Dokumentacijos ruošimo įrankių pasirinkimas .....	29
3.6. Informacinės sistemos komponentai.....	30
3.6.1. Komponentų diagrama.....	30
3.6.2. Paskirstymo diagrama.....	30
3.7. Sistemos administratoriaus sekų ir elgsenos diagramos.....	31
3.7.1. Duomenų persiuntimo į universitetinę DB sekų diagrama.....	31
3.7.2. Sistemos elgsenos diagrama, kai vartotojas administratorius.....	32
3.8. Testavimo modelis bei duomenys, kontrolinis pavyzdys .....	33
3.9. Sistemos administratoriaus – sistemos vartotojo naudojimo instrukcija .....	35
3.10. Uždavinio techninis išpildymas .....	40
3.11. Eksperimentinis tyrimas .....	40
3.11.1. Sukurtos sistemos kokybės tikrinimas .....	40
3.11.2. Tolimesnio sistemos tobulinimo, plėtojimo galimybės .....	40
3.12. Sukurto valdiklio intelektualizavimo galimybės .....	41
IŠVADOS .....	42
LITERATŪRA .....	44
TERMINŲ IR SANTRUMPŲ ŽODYNAS .....	46
SANTRAUKA ANGLŲ KALBA .....	47
PRIEDAI.....	48
Priedas Nr. 1 IS apskaitos paketų integracija .....	48
Priedas Nr. 2 Sistemos naudojamos duomenų bazės .....	55
Priedas Nr. 3 Uždavinio techninis išpildymas .....	57

## ĮVADAS

Šiuo metu dauguma organizacijų naudoja reliacines duomenų bazes (RDB). Šios duomenų bazės gali būti skirtingų gamintojų (Oracle, MS SQL Server ir kt.), gali veikti skirtingose operacinėse sistemose, naudoti skirtingą programinę įrangą. Tačiau organizacijoms norint tarpusavyje bendrauti iškyla poreikis keistis duomenimis. Esant bet kokiems nesuderinamumams tarp organizacijose naudojamų duomenų struktūrų, apsikeitimas informacija tampa ne tik sudėtingas, bet kartais ir neįmanomas. Taip pat yra ir kai ta pati organizacija naudoja kelias skirtingų gamintojų duomenų bazes ir nori apsikeisti informacija tarp jos naudojamų duomenų bazių. Norint integruoti skirtingas duomenų bazes, visų pirma reikia naudoti bendrą, integruojamų duomenų bazių struktūras apjungiantį standartą. Šis standartas turi būti universalus, turėtų būti pritaikomas skirtingoms duomenų bazėms neatsižvelgiant į jų struktūros specifiką. Vienas populiariausių tokių standartų yra XML.

Duomenų bazių integracijos uždavinio aktualumą sąlygoja informacijos sistemos reinžinerija. Informacijos sistemos reinžinerija vykdoma, kai:

- atsiranda naujos efektyvios informacinės technologijos;
- atsiranda nauji sistemai keliami funkciniai reikalavimai;
- reikia dokumentuoti naudojamą sistemą.

Darbo pirmas tikslas – išanalizuoti ir palyginti du funkcinį reikalavimų pateikimo būdus:

- panaudojant UML kalbą;
- panaudojant binarines komunikacines kilpas.

Binarinių komunikacinių kilpų metodas plėtojamas Informacijos sistemų katedroje (ISK) ir jo pagrindinis privalumas yra tas, kad galima griežtai atskirti funkcinis reikalavimus nuo nefunkcinių. Binarinių komunikacinių kilpų metodas aktualus, kai veikianti informacijos sistema yra perprojektuojama nedidelėmis dalimis ir po pakartotinės inžinerijos lieka panaudotos palikuoninės duomenų bazės.

Darbo antras tikslas – išskirti paskirstytų duomenų bazių replikavimo sustruktūrintus srautus. Darbe buvo naudojamos Informacijos sistemų katedroje ir Kauno technologijos universitete įdiegtos duomenų bazės (Informacijos sistemų katedroje – MS SQL Server, o Kauno technologijos universitete – Oracle). Darbe sprendžiamas uždavinys, kaip taisyklingai suformuoti XML srautus tarp paskirstytų duomenų bazių, kad nereikėtų įvedinėti vienodos informacijos į naudojamą reliacines duomenų bazes. Paskirstytų duomenų bazių projektavimui buvo naudojamos akademinės mokomosios duomenų bazės.

Eksperimentiniam nagrinėjimui buvo apibrėžtas padalinio uždavinys „Dėstytojų individualių planų sudarymas“, kurio rezultatai turi būti perkelti į universitetinę duomenų bazę (Oracle DB).

Programinės realizacijos dalyje buvo sukurtas valdiklis, kurio pagalba administratorius gali suformuoti XML failą iš MS SQL Server ir nuskaičius šį failą surašyti duomenis į Oracle duomenų bazę. Padalinio vadovas Internetu gali peržiūrėti ir atlikti individualių dėstytojų planų tvarkymą. Dėstytojas gali sudaryti ir peržiūrėti savo individualų planą. Sistema prieinama per Internetą naudojant Interneto naršyklę. Sistemos realizavimui naudojama VB .NET programavimo kalba. Valdiklio projektavimui panaudotos CASE priemonės „Rational Rose 2000 Enterprise Edition“ ir „Microsoft Visio Professional 2002“.

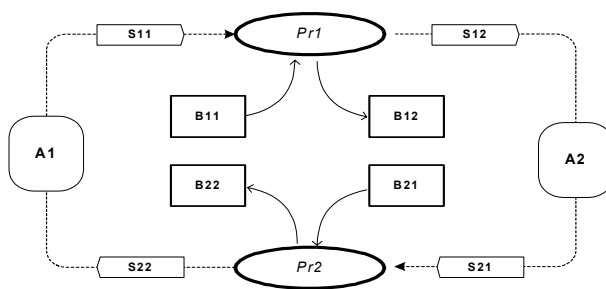
Darbo pabaigoje pateikiamos šio darbo išvados, taip pat nurodoma naudota literatūra, terminų žodynas, sutrumpinimų sąrašas bei priedai. Priedų skyriuje yra pateikiamas devintojoje magistrantų ir doktorantų konferencijoje pristatytas straipsnis „IS apskaitos paketų integracija“.

# 1. VARTOTOJO POREIKIŲ ANALIZĖ

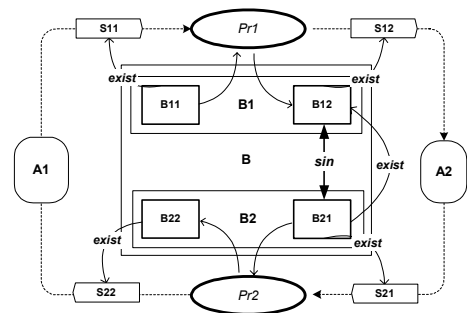
## 1.1. Pragmatiškai motyvuotų kilpų panaudojimas

Organizacijoje vykstančiai veiklai atvaizduoti vienas iš galimų būdų yra komunikacinių kilpų sudarymas. Veikla atvaizduojama komunikacinėmis binarinėmis kilpomis (angl. *Communicative Action Loop*), kurios yra susietos tarpusavyje semantinėmis ir pragmatinėmis priklausomybėmis. Kiekviena CAL susideda iš dviejų aktorių užsakovo A1 ir vykdytojo A2, srautų S11 - S22 ir dviejų procesų – užsakovo veiksmų Pr1 ir vykdytojo veiksmų Pr2. kompiuterizuotos informacijos sistemos dinaminė situacijų kaita užduodama dviem perėjimais iš būsenos B11 ir B21 į būsenas B12 ir B22.

Paprasta komunikacinė kilpa yra pavaizduota 1 pav., o išplėstinė komunikacinė kilpa - pavaizduota 2 pav. [8]:



1 pav. Paprasta komunikacinė kilpa



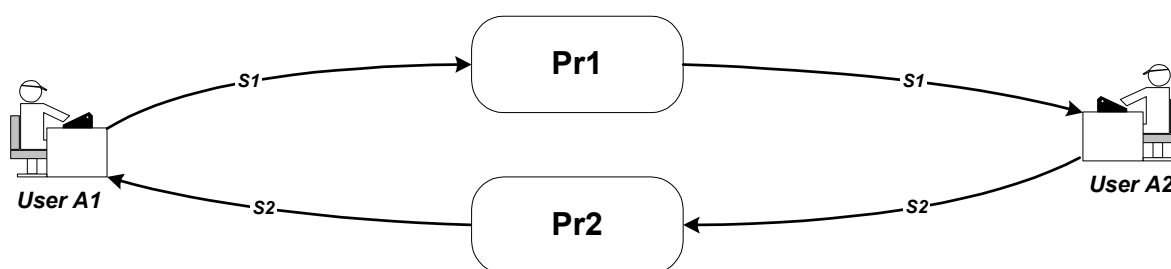
2 pav. Išplėstinė komunikacinė kilpa

Jeigu kompiuterizuotos informacinės sistemos projektuojamos komunikacinių kilpų analizės metodu, tuomet galima užtikrinti tam tikrus semantinio integralumo kriterijus:

- Glaudumo (angl. *cohesion*) kriterijus nustato reikalavimus komunikacinių srautų valdomų būsenų atvaizdavimui duomenų bazėje. Ryšiai tarp būsenų ir komunikacinių kilpų struktūros palaikomi atvaizduojant būsenų pokyčius į duomenų bazės egzempliorių atributų aibes;
- Pilnumo (angl. *completeness*) kriterijus reiškia, kad nepilnumą galima identifikuoti iš probleminių būsenų, kurios rodo poreikius tobulinti kompiuterizuotą informacinę sistemą;
- Darnos (angl. *coherence*) kriterijus reiškia, kad galinės būsenos turi būti suderintos su aktorių tikslais;
- Gyvybingumo (angl. *viability*) kriterijus yra išreiškiamas pragmatinėmis priklausomybėmis:

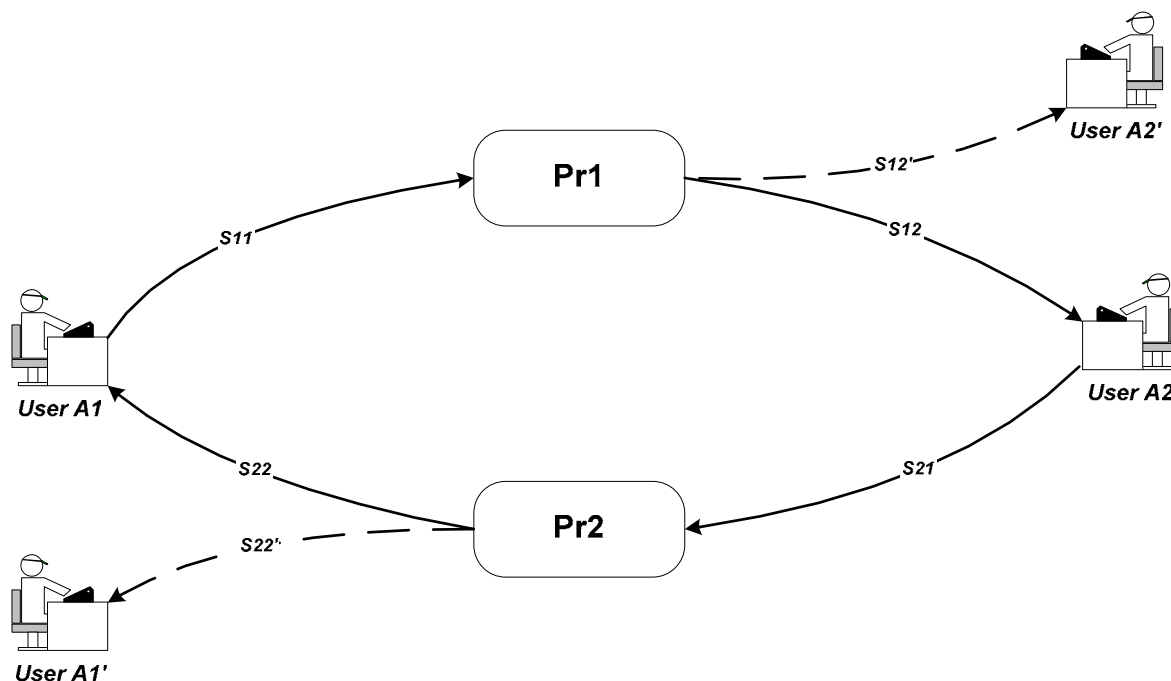
Komunikacinė kilpa yra gyvybinga, jei  $A1 \xrightarrow{p} B11, A2 \xrightarrow{o} B22, A2 \xrightarrow{p} B21, A1 \xrightarrow{g} B12$ , tuomet  $B11 \xrightarrow{-} B12, B22 \xrightarrow{+} B12, B22 \xrightarrow{-} B11, B12 \xrightarrow{-} B21$ ; čia simbolis "  $\xrightarrow{\quad}$  " žymi pragmatines priklausomybes:  $\xrightarrow{g}$  - reiškia tikslą,  $\xrightarrow{p}$  - problema,  $\xrightarrow{o}$  - proga,  $\xrightarrow{+}$  - teigiamos įtakos priklausomybę, o  $\xrightarrow{-}$  - neigiamos įtakos priklausomybę [9].

Vartotojų siuntinėjamos informacijos srautams ar duomenų bazėse esančios informacijos apsisikeitimui modeliuoti ir kuriamos sistemos funkciniam reikalavimams užduoti galima remtis taip pat komunikacinių kilpų formavimo požiūriu, kuris yra naudojamas ir Informacijos sistemų katedroje.



3 pav. Binarinė vartotojų bendradarbiavimo kilpa

Binarinę vartotojų bendradarbiavimo kilpą galima dekomponuoti į bendradarbiavimo kilpą, kai tarp vartotojų yra siuntinėjami asinchroniniai pranešimai.



4 pav. Vartotojų bendradarbiavimo kilpa, kai siunčiami asinchroniniai pranešimai

## 1.2. UML panaudojimas

Taikant UML organizacijos veiklos analizei galima pilnai aprašyti kuriamos sistemos probleminę sritį, reikalavimus bei realizacijos procesą. Pagrindinis UML požymis yra universalumas. Taip pat galima aprašyti visus informacinės sistemos gyvavimo ciklo etapus, tačiau UML nėra numatyta priemonių pragmatinėms priklausomybėms modeliuoti.

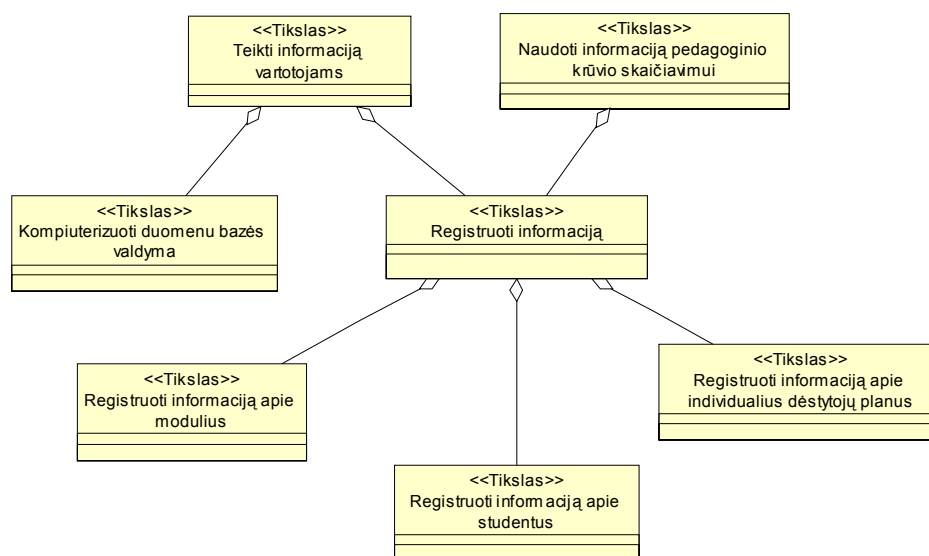
## 1.3. Organizacijos veiklos analizė panaudojant UML

### 1.3.1. Organizacijos veikla

Darbe aprašoma organizacija – KTU Informacijos sistemų katedra. Informacijos sistemų katedroje vyksta įvairūs procesai (pvz.: individualių dėstytojų planų sudarymas). Padalinio vadovas įvedinėja vienodą informaciją į dvi duomenų bazines. Padalinyje vykstančių procesų kompiuterizavimas leidžia padalinio vadovui ir dėstytojams efektyviau organizuoti savo veiklas.

### 1.3.2. Veiklos tikslų modelis

Organizacijos tikslai yra atvaizduoti tikslų modelyje. Organizacijos tikslus apima visos veiklos, kuriomis siekiama įgyvendinti organizacijos pagrindinius uždavinius.

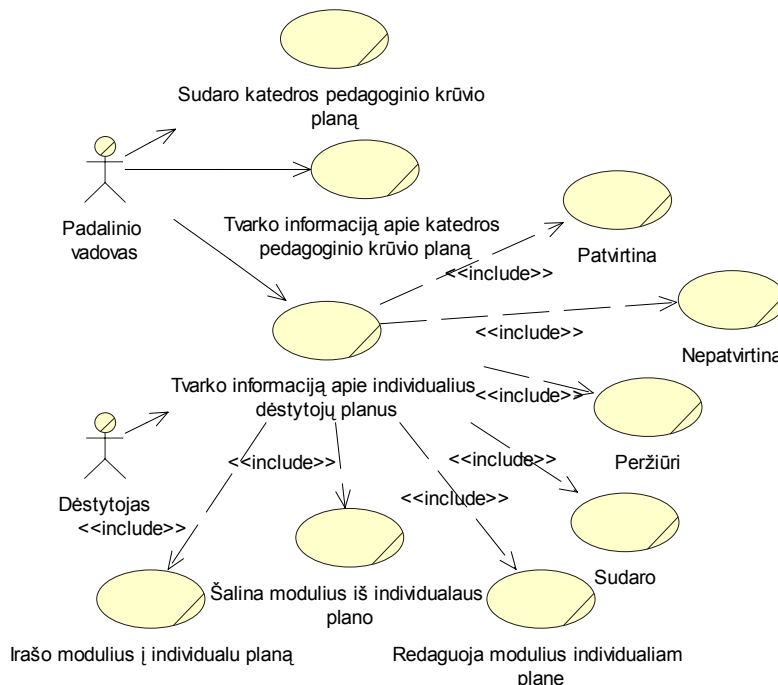


5 pav. Veiklos tikslų modelis



### 1.3.3. Veiklos panaudojimo atvejų modelis

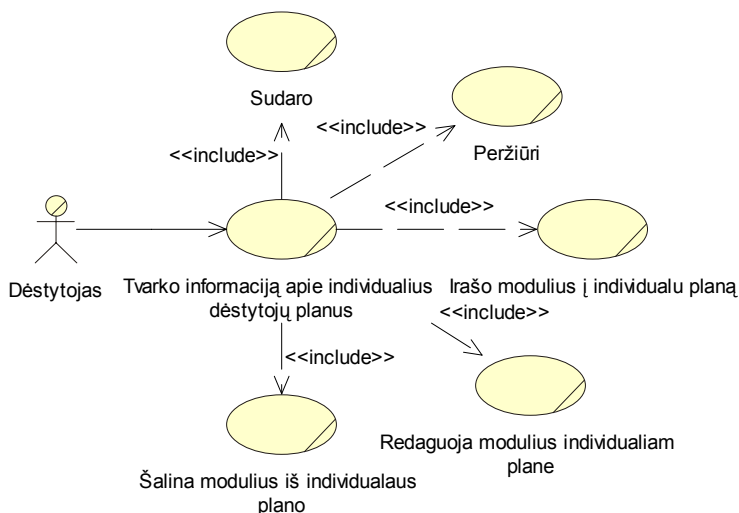
Iš veiklos panaudojimo atvejų diagramos matome, kokius darbo procesus atlieka kiekvienas organizacijos veikėjas.



6 pav. Veiklos panaudojimo atvejų modelis

#### 1.3.3.1. Dėstytojo veiklos panaudojimo atvejo modelis

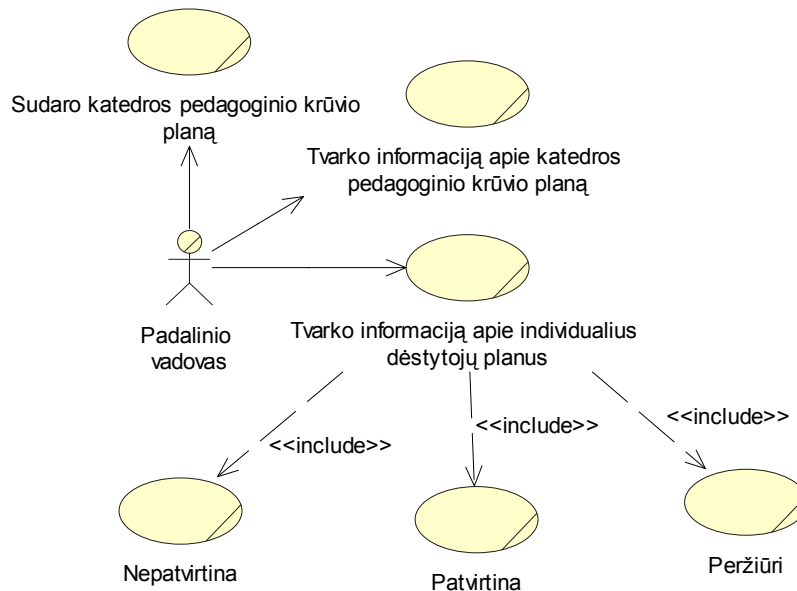
Iš veiklos panaudojimo atvejų diagramos, matome, jog dėstytojas atlieka tokius organizacijos darbo procesus: sudaro savo individualų planą ir peržiūri savo individualų planą.



7 pav. Dėstytojo veiklos panaudojimo atvejo modelis

### 1.3.3.2. Padalinio vadovo veiklos panaudojimo atvejo modelis

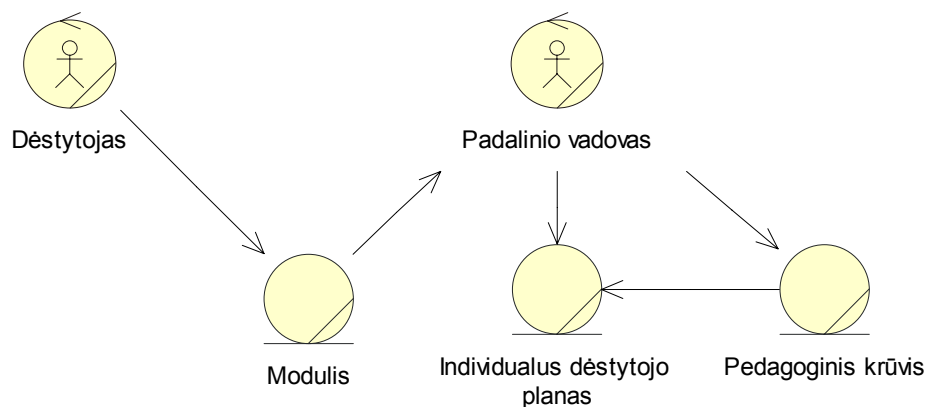
Iš veiklos panaudojimo atvejų diagramos, matome, jog padalinio vadovas atlieka tokius organizacijos darbo procesus: peržiūri individualų dėstytojo planą, redaguoja individualius dėstytojų planus, sudaro katedros pedagoginio krūvio planą ir redaguoja katedros pedagoginio krūvio planą.



8 pav. Padalinio vadovo veiklos panaudojimo atvejo modelis

### 1.3.4. Veiklos objektų modelis

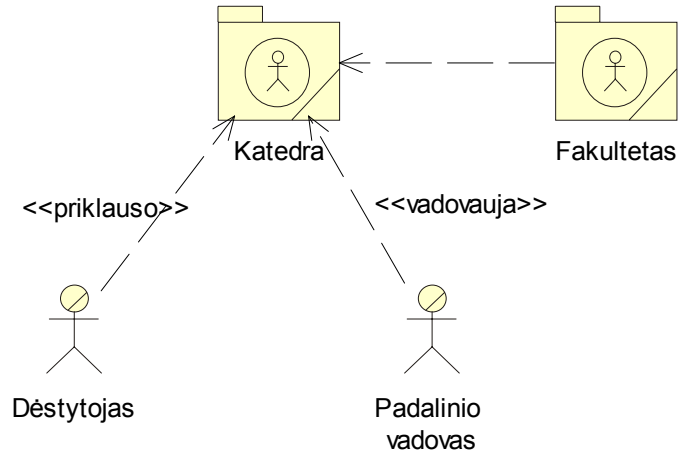
Pagrindiniai veiklos objektai yra: dėstytojas ir padalinio vadovas. Dėstytojas gali iš savo dėstomų modulių sudaryti savo individualų planą ir atlikti informacijos apie individualų planą tvarkymą. Padalinio vadovas gali peržiūrėti ir patvirtinti visų dėstytojų individualius planus ir sudaryti katedros pedagoginį krūvį.



9 pav. Veiklos objektų modelis

### 1.3.5. Organizacinės struktūros modelis

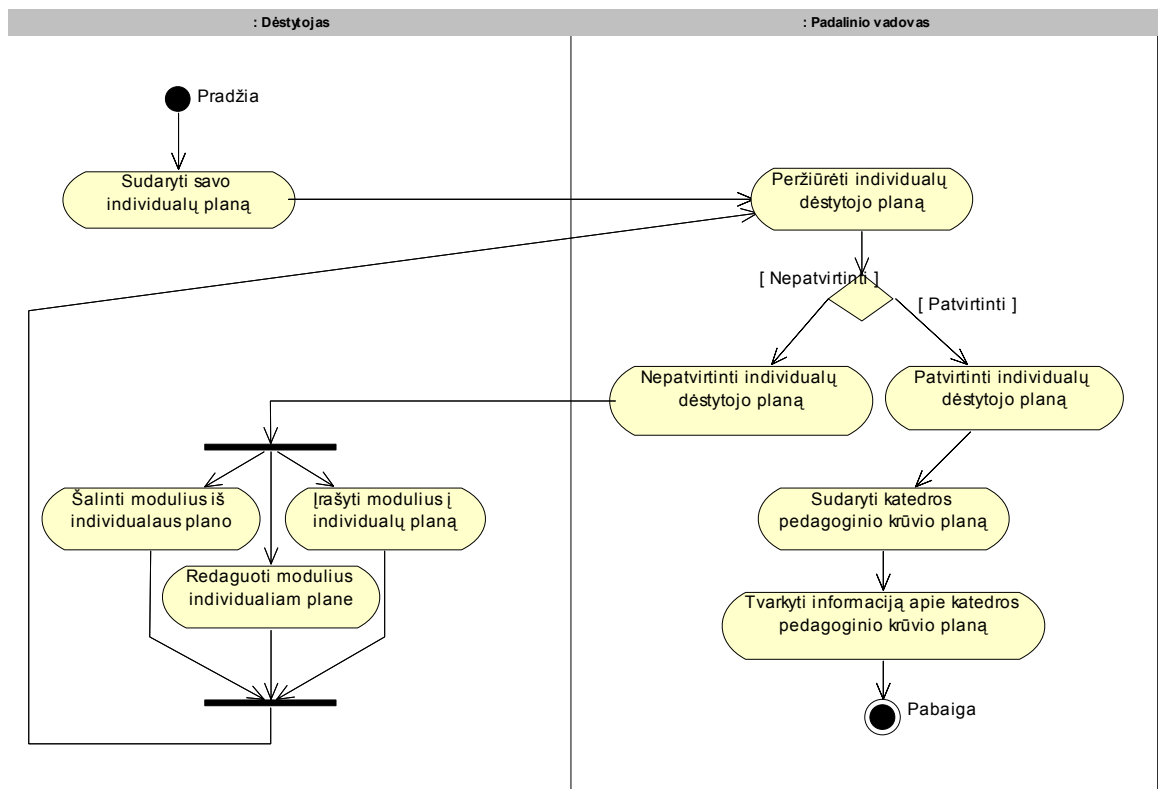
Organizacinės struktūros modelis atvaizduotas diagramoje pateiktoje žemiau.



10 pav. Organizacinės struktūros modelis

### 1.3.6. Procesų veiklos diagrama

Iš procesų veiklos diagramos matyti kaip ir kokie procesai vyksta padalinyje.



11 pav. Procesų veiklos diagrama

### 1.3.7. Vartotojų bei jų poreikių analizė

Sukurta sistema naudosis trys vartotojai: padalinio vadovas, dėstytojas ir administratorius. Visi vartotojai turės būti registruoti sistemos vartotojai.

Naudodamasis sistema administratorius turi galėti:

- nuskaityti duomenis į XML failą iš MS SQL Server DB;
- nuskaityti duomenis į XML failą iš Oracle DB;
- peržiūrėti XML failus ir schemas;
- atstatyti duomenis iš XML failo į MS SQL Server DB;
- nusiusti duomenis iš XML failo į Oracle DB.

Naudodamasis sistema padalinio vadovas turi galėti:

- peržiūrėti individualius dėstytojų planus;
- patvirtinti arba nepatvirtinti pasirinkto dėstytojo individualų planą;
- atsispausdinti pasirinkto dėstytojo sudarytą individualų planą.

Naudodamasis sistema dėstytojas turi galėti:

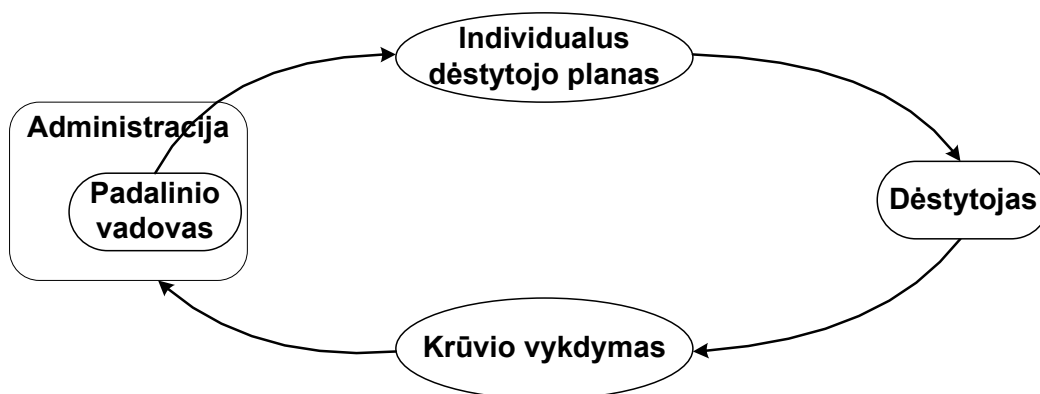
- sudaryti savo individualų planą;
- atsispausdinti savo individualų planą;
- priskirti naujus, redaguoti ir šalinti jau esamus modulius iš savo individualaus

plano.

## 1.4. Organizacijos veiklos vaizdavimas komunikacinėmis kilpomis

Organizacijoje vykstančius procesus galima atvaizduoti komunikacinėmis kilpomis.

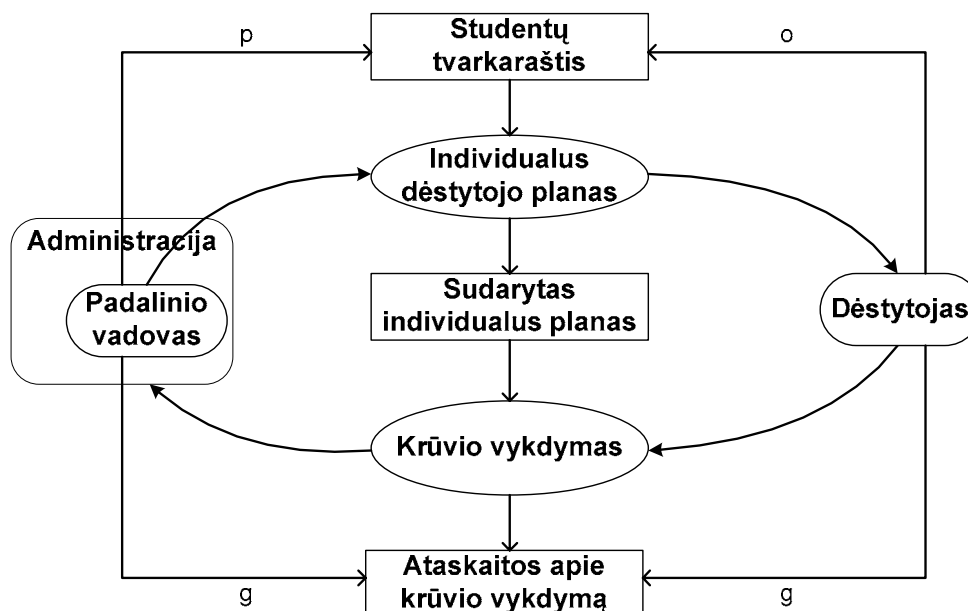
### 1.4.1. Komunikacinės priklausomybės tarp aktorių



12 pav. Komunikacinės priklausomybės tarp aktorių

Komunikacinių priklausomybių tarp aktorių modelyje galime išskirti panaudojimo atvejus (individualus dėstytojo planas ir krūvio vykdymas).

#### 1.4.2. Pragmatinės priklausomybės



13 pav. Pragmatinės priklausomybės

Pragmatinių priklausomybių modelyje galime matyti kiekvieno aktoriaus tikslus, galimybes ir problemas.

- g – tikslas (en. *goal*);
- o – galimybė (en. *opportunity*);
- p – problema (en. *problem*).

Dėstytojo ir padalinio vadovo tikslas – pateikti ataskaitas apie padalinyje sudaryto krūvio vykdymą. Dėstytojo galimybė – sudarytas studentų tvarkaraštis, kuris yra padalinio vadovo problema.

#### 1.5. Vartotojo poreikių analizės dalies išvados

1. Palygintos dvi metodikos (UML ir komunikacinių kilpų), skirtos organizacijos veiklai modeliuoti.
2. Pagrindinės UML modeliavimo savybės:
  - universalumas;
  - apibrėžiami visi informacinės sistemos kūrimo proceso etapai;
  - nėra numatyta priemonių pragmatinėms priklausomybėms modeliuoti.

3. Pagrindinės komunikacinių kilpų modeliavimo savybės:
  - vartotojo poreikiai pragmatiškai motyvuoti;
  - verslo funkcijų detali analizė atskiriama nuo architektūrinių sprendimų;
  - leidžia segmentuoti organizacijos veiklą savarankiškais veiklos vienetais.
4. Pateikta nagrinėjamos organizacijos veikla UML ir komunikacinių kilpų požiūriu.
5. Pateikta komunikacinė kilpa Individualus dėstytojo planas / Krūvio vykdymas (13 pav.) iliustruoja sudarytos komunikacinės kilpos gyvybingumą.

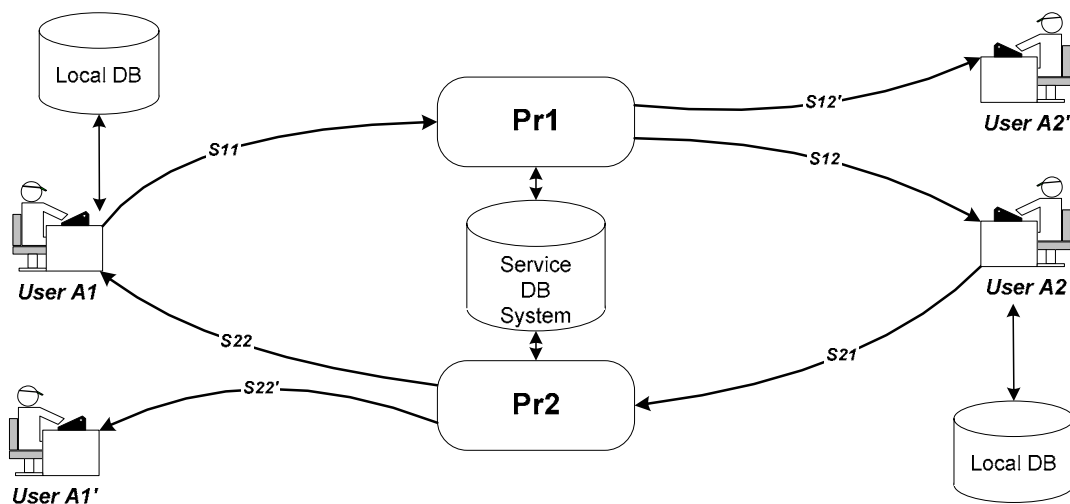
## 2. SISTEMOS ARCHITEKTŪRINIAI SPRENDIMAI

### 2.1. Galimi architektūriniai sprendimai

Sprendžiant paskirstytų duomenų bazių apjungimo ir duomenų persiuntimo tarp skirtingų duomenų bazių uždavinį galimi keli būdai. Architektūrinuose sprendimuose tarp naudojamų duomenų bazių yra vaizduojami tarpininkai, kurie gali būti tokių tipų:

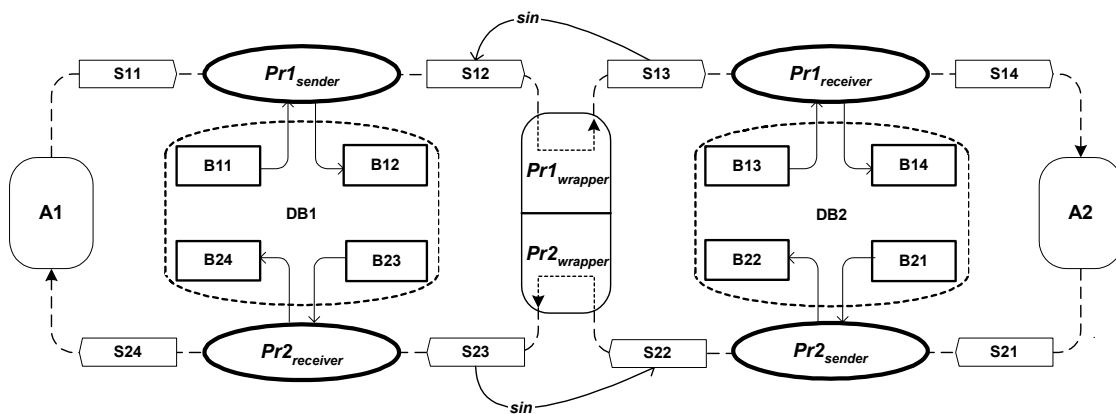
- valdikliai (ODBC);
- apvaskalai (wrapper);
- replikatoriai (replication);
- mediatoriai;

Vartotojai siuntinėdami duomenis gali naudoti serviso duomenų bazių sistema, kuri gali būti paskirstyta. Kiekvienas vartotojas gali turėti individualias duomenų bases (pvz.: padalinyje padalinio vadovas naudojami FoxPro duomenų baze).



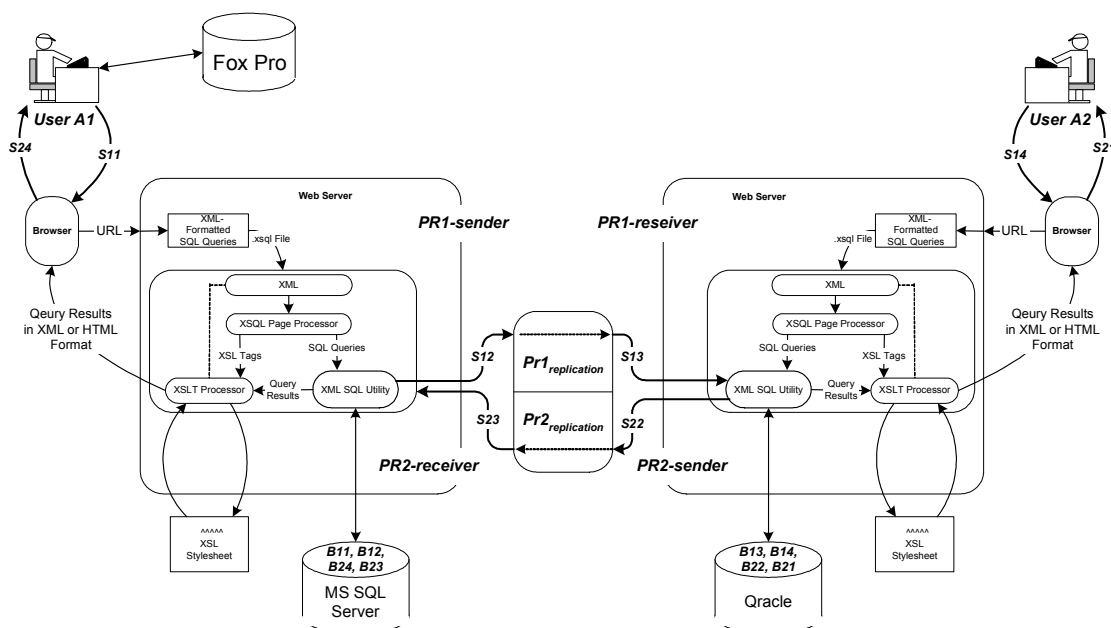
14 pav. Komunikacinė kilpa, kurioje panaudota serviso duomenų bazių sistema

Naudojant paskirstytų duomenų bazių apvaskalus duomenų srautus galima atvaizduoti komunikacine kilpa pateikta 15 pav.



15 pav. Komunikacinė kilpa, kurioje procesai dekomponuoti ir naudojamas apvalkalas

Vienas iš galimų duomenų bazių tarpininkų yra replikatorius. Kai vartotojai dirba su MS SQL Server ir Oracle duomenų bazėmis (vartotojai gali turėti savo individualias bazines, pvz.: FoxPro) ir yra naudojamas XML komunikacinė kilpa galime atvaizduoti taip kaip pateikta 16 pav.



16 pav. Komunikacinė kilpa, kai naudojamos MS SQL Server ir Oracle duomenų bazės ir XML. Vartotojai gali turėti individualias duomenų bazines (pvz.: FoxPro)

## 2.2. XML srautų panaudojimo aktualumas

XML (eXtensible Markup Language) buvo sukurta 1996 metais. Ją sukūrė XML darbo grupė, padedama W3C (World Wide Web Consortium) konsorciumo.

XML aprašo duomenų objektų klasę, vadinamą XML dokumentu ir iš dalies aprašo šiuos dokumentus apdorojančių kompiuterinių programų elgesį. XML yra SGML (Standart



Generalized Markup Language) [ISO 8879] apribota forma. Savo struktūra XML dokumentai atitinka SGML dokumentus.

XML dokumentai yra sudaryti iš elementų, vadinamų esybėmis, kuriose yra išnagrinėti arba neišnagrinėti duomenys. Išnagrinėti duomenys yra sudaryti iš simbolių. Vieni simbolių junginiai sudaro duomenis, kiti junginiai sudaro žymes. Žymėmis yra koduojama dokumento sandara ir loginė struktūra. XML pateikia sandaros ir loginės struktūros apribojimų pritaikymo mechanizmą [1].

XML yra naudojama kaip žymių kalbų sudarymo šablonas:

- XML naudoja neribojamą žymių skaičių, t.y. skirtingai nei HTML nėra jokių apribojimų jų kiekiui – vartotojas pats gali susikurti žymes atsižvelgdamas į informacijos pobūdį;
- Kiekviena XML kalba yra skirta savo taikomajai sričiai, tačiau kalbos pasižymi bendromis savybėmis;
- Dokumentams apdoroti yra skirti bendri įrankiai.

Galima būtų išskirti šiuos pagrindinius XML kalbos paskirties bruožus:

- Atskirti sintaksę nuo semantikos. Tai yra reikalinga todėl, kad yra kuriamas bendras informacijos struktūrizavimo mechanizmas (naršyklės vaizduojamos semantikos yra apibrėžiamos stilių lentelėmis);
- Leisti kurti žymes bet kokiai įsivaizduojamai taikomųjų programų sričiai;
- Palaikyti internacionalizaciją ir nepriklausomybę nuo platformų;
- Ateityje vaizduoti struktūrizuotą informaciją, kuri apimtų ir duomenų bazines.

#### **XML paskirtis ir privalumai:**

- XML skirta keitimuisi duomenimis;
- XML skirta keistis finansine informacija, t.y. yra naudojama daugumoje B2B taikomųjų programų;
- XML gali būti naudojama duomenų paskirstymui;
- XML gali būti skirta duomenų saugojimui;
- XML gali padaryti duomenis labiau prieinamus.

Remiantis aukščiau išvardintais XML požymiais galima daryti išvadą, jog XML kalba labiausiai tinka duomenų keitimuisi tarp heterogeninių duomenų bazių. Tačiau šis procesas nėra toks paprastas, nes reikia atsižvelgti į įvairių sistemų funkcinius reikalavimus. Reikia sukurti bendrą šabloną, kuriuo remiantis būtų įmanoma atlikti keitimąsi duomenimis tarp įvairių sistemų. Naudojant XML šiuo šablonu patogiau naudoti XML schemas [4].

Visos didžiosios programinės įrangos kūrimo kompanijos (Microsoft, Oracle, IBM ir pan.) įsisavino XML ir pradėjo taikyti tai naujose programų versijose. Taigi duomenys tarp skirtingų programų gamintojų pradėjo „keliauti“ visiems suprantu formatu.

### 2.3. XML technologijos duomenų bazių valdymo sistemose

Duomenų bazės palaikančios XML kalbą turi įdiegtas specifines sąsajas duomenų mainams tarp XML dokumentų ir DB.

*Lentelė Nr. 1*

*XML palaikančios duomenų bazės [14]*

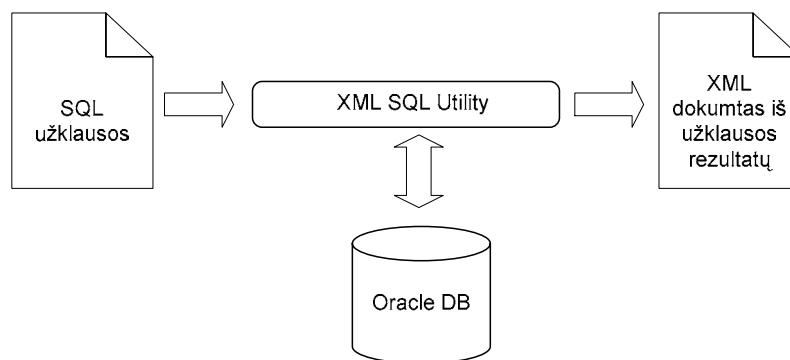
<b>Product</b>	<b>Developer</b>	<b>License</b>	<b>DB Type</b>
<u>Access 2002</u>	Microsoft	Commercial	Relational
<u>Cache</u>	InterSystems Corp.	Commercial	Post-relational
<u>DB2</u>	IBM	Commercial	Relational
<u>eXtremeDB</u>	McObject	Commercial	Navigational
<u>FileMaker</u>	FileMaker	Commercial	FileMaker
<u>FoxPro</u>	Microsoft	Commercial	Relational
<u>Informix</u>	IBM	Commercial	Relational
<u>Matisse</u>	Matisse Software	Commercial	Object-oriented
<u>Objectivity/DB</u>	Objectivity	Commercial	Object-oriented
<u>OpenInsight</u>	Revelation Software	Commercial	Multi-valued
<u>Oracle 8i, 9i</u>	Oracle	Commercial	Relational
<u>SQL Server 2000</u>	Microsoft	Commercial	Relational
<u>Sybase ASE 12.5</u>	Sybase	Commercial	Relational
<u>UniData</u>	IBM	Commercial	Nested relational
<u>UniVerse</u>	IBM	Commercial	Nested relational
<u>Versant enJin</u>	Versant Corp.	Commercial	Object-oriented

Kadangi šiame darbe bus dirbama su MS SQL Server ir Oracle duomenų bazėmis toliau apžvelgsime tik šiose duomenų bazėse XML panaudojimo galimybes.

#### 2.3.1. Oracle duomenų bazė ir XML panaudojimo galimybės

Oracle turi XML parserius Java, C, C++ ir PL/SQL kalboms, todėl programuojant šiomis kalbomis XML panaudojimas yra lengvesnis.

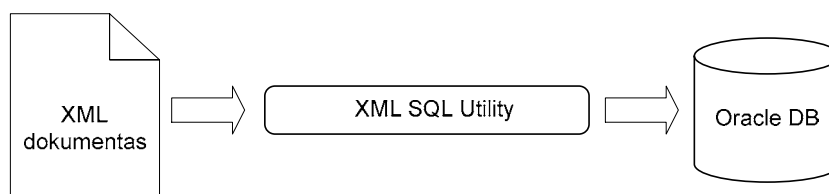
Naudojant XML SQL Utility galima vykdyti SQL užklausas ir gauti rezultatus XML dokumentą iš suformuotų užklausų rezultatų.



17 pav. XML dokumento formavimas naudojant XML SQL Utility

Gauto XML dokumento struktūra atitinka Oracle DB struktūrą, kuri buvo nurodyta formuojant SQL užklausa. Duomenų bazės stulpelių pavadinimai atitinka XML elementų grupių pavadinimus, o reikšmės yra priskiriamos vidiniams elementams.

Taip pat naudojantis XML SQL Utility galima ir surašyti XML duomenis į Oracle duomenų bazę.



18 pav. XML dokumento duomenų surašymas į Oracle DB

Saugant duomenis iš XML dokumento į Oracle duomenų bazę kartu išsaugoma ir dokumento struktūra. XML dokumento elementų pavadinimai atitinka duomenų bazės lentelių ir lentelių stulpelių pavadinimus, o reikšmės yra įrašomos į atitinkamus lenteles laukus [13].

### 2.3.2. MS SQL Server duomenų bazė ir XML panaudojimo galimybės

MS SQL Server duomenų bazės duomenis į XML galima perduoti trimis būdais:

- Baigiant SELECT sakinį FOR XML;
- XPath queries, kurios komentuoja XML schemas;
- OpenXML funkcija procedūrose.

OpenXML funkcija dažniausiai naudojama atvaizduoti XML dokumento duomenims į reliacinės bazės lenteles [12].

MS SQL Server yra galimos tokios XML savybės:

- HTTP pasiekiamumas (rašant SQL sakinį naršyklės adresų eilutėje);

- Formuoti XML galima SELECT sakinio pagalba (FOR XML);
- XML views (nurodoma kaip reliaciniai duomenys turi būti atvaizduoti XML);
- XPath queries (nurodoma kaip duomenys turi būti paimti iš DB);
- OpenXML (nurodo, kaip duomenys turi būti atvaizduoti XML ir suteikia galimybę XML failo duomenis atvaizduoti duomenų bazėje lentelėse);
- OLE DB ir ADO pasiekiamumas (naudojama sąsajai su reliacinėmis DB formuojant XML iš DB ir surašant XML duomenis į DB).

### 2.3.3. Oracle ir MS SQL Server teikiamų XML savybių palyginimas

Lentelė Nr. 2

Oracle ir MS SQL Server teikiamos XML savybės

Savybės	SQL Server 2000	Oracle
<p><b>Deklaratyvus žymėjimas laisvai susietų biznio sistemų (ši savybė suteikia dvikryptį žymėjimą tarp XML ir reliacinių duomenų vaizdų).</b></p> <p>SQL Server 2000 turi integruotas XML savybes, kurios leidžia priėjimą prie sistemos su minimaliu programavimu.</p> <p>Oracle neturi integruotų XML specifinių savybių.</p>	<p>XML Views</p> <p>XPath support</p> <p>Transact-SQL FOR XML išplėtimai</p> <p>Transact-SQL OPENXML išplėtimai</p>	<p>Neturi specifinių savybių. Reikia rašyti papildomas programas.</p>
<p><b>Vietiniai XML išplėtimai skirti SQL ir leidžiantys vartotojui vykdyti aplikaciją greitai ir be papildomo programavimo.</b></p> <p>SQL Server 2000 teikia išplėtimus SQL kalbai, kurie gali būti panaudoti gražinti XML duomenis iš standartinių SQL užklausų.</p> <p>Oracle teikia serverio pagalbinius įrankius, kuriuos vykdant neišplečiama SQL ir reikalaujama papildomo programavimo gražinant XML duomenis.</p>	<p>Transact-SQL FOR XML išplėtimai</p> <p>Transact-SQL OPENXML išplėtimai</p>	<p>Neturi specifinių savybių. Reikia rašyti papildomas programas naudojant XSQL Utility.</p>

<p><b>Įvairūs XML prieigos metodai, kurie palaiko lankstumą, leidžiantį programuotojams pasiekti XML duomenis per tinklo prisijungimus.</b></p> <p>SQL Server 2000 ir Oracle palaiko įvairius metodus leidžiančius pasiekti XML duomenis, esančius duomenų bazėse.</p>	<p>HTTP prieiga</p> <p>OLE DB/ADO prieiga</p> <p>JDBC</p>	<p>HTTP prieiga</p> <p>OLE DB/ADO prieiga</p> <p>JDBC</p>
<p><b>XML šablonai</b></p> <p>SQL Server ir Oracle palaiko išsaugotus XML užklausių šablonus serveryje su laisvai pasirenkamais parametrais.</p>	<p>URL/HTTP prieiga</p> <p>XML šablonai</p>	<p>URL/HTTP prieiga</p> <p>XSQL šablonai</p>
<p><b>Saugi Web aplikacijų prieiga prie XML duomenų</b></p> <p>SQL Server 2000 palaiko saugumą, kuris gali būti valdomas individualios lentelės lygyje.</p> <p>Oracle palaiko apribotą saugumą tik duomenų bazės vartotojo, bet ne individualios lentelės lygio.</p>	<p>Duomenų bazės vartotojo lygmens saugumas</p> <p>Duomenų bazės objekto lygmens saugumas</p>	<p>Duomenų bazės vartotojo lygmens saugumas</p>
<p><b>Galimybės detalesnei paieškai sudėtingiems XML dokumentams.</b></p> <p>SQL Server 2000 ir Oracle teikia duomenų bazės palaikomą sudėtingų tekstinių dokumentų paiešką apimančią ir XML.</p>	<p>Pilna teksto paieška</p>	<p>Tarpinė teksto paieška su segmentų savybėmis.</p>
<p><b>XML parseriai (skirti programuotojams, kurie siekia didesnės galios ir lankstumo, kai kuriamos XML aplikacijos).</b></p>	<p>MSXML COM-priėjimo parserį palaiko daugelis kalbų</p>	<p>Oracle parserius palaiko daugelis kalbų</p>

Microsoft SQL Server 2000 siūlo XML savybes daugiausia skirtas Web ir duomenų bazių programuotojams. Web programuotojams nereikia mokintis duomenų bazių programavimo subtilybių norint pasinaudoti SQL Server 2000 XML privalumais, nes SQL Server teikia standartines XML konstrukcijas. Taip pat, duomenų bazių programuotojams

nereikia išmanyti XML programavimo subtilybių, nes jie gali naudotis FOR XML ar OpenXML siekdami manipuluoti XML duomenimis.

Oracle labiau skirtas naudoti patyrusiems programuotojams. Norint pasiekti sprendžiamų uždavinių rezultata tenka naudotis keliomis priemonėmis arba kurti specifinius įrankius [11].

Kadangi šiame darbe yra dirbama su XML ir viena iš duomenų bazių yra Oracle, todėl ir bus kuriamas papildomas įrankis su VB .NET, siekiant integruoti MS SQL Server ir Oracle duomenų bazes.

## 2.4. Replikavimo priemonė MS SQL serveryje

MS SQL serverio replikavimas leidžia persiųsti duomenis iš vieno serverio į kitą serverį, net jei kitas serveris yra nutolęs. Yra daug programinių paketų, kurie leidžia persiųsti duomenis, tačiau pačio SQL serverio teikiama tokia galimybė, kai duomenis persiuntinėjami iš vieno SQL serverio į kitą SQL serverį, yra labiau patikima.

### Replikavimo galimybė

Pagrindinės savybės, kurias reikia žinoti norint pasinaudoti SQL serverio teikiamu replikavimo įrankiu:

- sujungiamumas (en. *connectivity*) (ar yra galimybė pasiekti kitą serverį);
- savarankiškumas (en. *autonomy*) (kiek serveriai bus nepriklausomi vienas nuo kito);
- duomenys ir schema (en. *data and schema needs*) (kokie duomenys bus siunčiami);
- vėlavimo laikas (en. *latency*) (kiek dažnai reikia atnaujinti duomenis);
- lyderio nurodymas (en. *determine the ladder*) (kaip duomenys bus atnaujinami jei tuo pačiu metu du vartotojai atnaujina duomenis. Ar hierarchiškai ar bus suteikiama kuriam nors vartotojui pirmenybė).

### Replikavimo architektūra

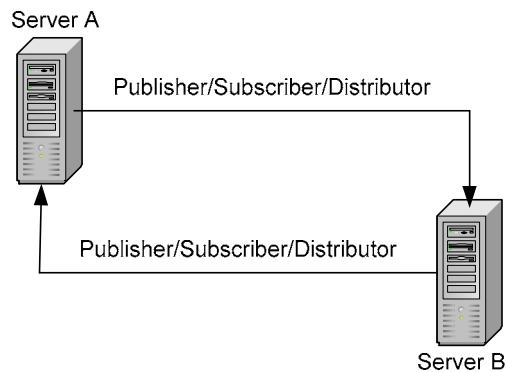
Replikavimo architektūra yra paprasta. Aibė elementų, kuriuos replikuojame, yra vadinama publikacija (en. *publication*). Galimi replikavimo būdai:

- visos lentelės (en. *entire table*) (replikuojama visa lentelė);
- vertikalus padalijimo (en. *vertical partition*) (pasirenkami lentelės stulpeliai, kuriuos norime replikuoti);
- horizontalus padalijimo (en. *horizontal partition*) (leidžia replikuoti tik išrinktus duomenis);

- išsaugotos procedūros rezultatų (en. *stored procedure results*) (replikuojama išsaugota procedūra);
- vaizdų (en. *views*) (replikuojamas vaizdų turinys).

Galimi SQL serverio replikavimo tipai:

- snapshot (leidžiamas didžiausias savarankiškumas);
- merge (sudėtingos logikos bet turi galimybę atstatyti klaidas);
- transactional (siūlomi momentiniai atnaujinimai, neleidžiamas savarankiškumas) [3].



19 pav. Dvikryptis replikavimo modelis

Šiame darbe dirbama su XML ir MS SQL Server ir Oracle duomenų bazėmis, o MS SQL siūloma replikavimo priemonė nenaudoja XML ir ji geriausiai tinkama ir labiausiai patikima, kai duomenys persiuntinėjami iš vieno SQL serverio į kitą SQL serverį, todėl reikia ieškoti papildomų sprendimų integruojant MS SQL Server ir Oracle duomenų bazines ir naudojant XML.

## 2.5. Sistemos architektūrinių sprendimų dalies išvados

1. Pateikti galimi architektūriniai sprendimai uždaviniui spręsti. Vienas iš galimų sprendimų norimam rezultatui pasiekti panaudoti XML.
2. Apžvelgta XML kalbos specifika ir išvardinti privalumai:
  - XML skirta keitimuisi duomenimis;
  - XML skirta keistis finansine informacija, t.y. yra naudojama daugumoje B2B taikomųjų programų;
  - XML gali būti naudojama duomenų paskirstymui;
  - XML gali būti skirta duomenų saugojimui;
  - XML gali padaryti duomenis labiau prieinamus.

3. Apžvelgtos XML technologijos duomenų bazių valdymo sistemose ir pateiktas Oracle ir MS SQL Server teikiamų XML savybių palyginimas. Microsoft SQL Server 2000 siūlo XML savybes daugiausia skirtas Web ir duomenų bazių programuotojams ir teikia standartines XML konstrukcijas. Oracle labiau skirtas naudoti patyrusiems programuotojams. Norint pasiekti sprendžiamų uždavinių rezultatą tenka naudotis keliomis priemonėmis arba kurti specifinius įrankius.
4. Apžvelgta MS SQL Server siūloma replikavimo priemonė.



### **3. XML PANAUDOJIMAS DUOMENŲ BAZIŲ SISTEMŲ REPLIKAVIME**

#### **3.1. Techninė užduotis**

Tvirtinu

Vedėjas

Suderinta

Vadovas Bronius Paradauskas

Data 2004-06-15

#### **TECHNINĖ UŽDUOTIS**

##### **1. TEMA:**

Paskirstytų duomenų bazių replikavimo komponentas

##### **2. ANALITINIS IR TIRIAMASIS DARBAS:**

2.1. UML panaudojimo organizacijos veiklai modeliuoti analizė;

2.2. Komunikacinių kilpų panaudojimo organizacijos veiklai modeliuoti analizė;

2.3. Organizacijoje vykstančių procesų funkcijų analizė.

**3. SUPROJEKTUOTI, REALIZUOTI IR PARUOŠTI, VARTOJIMUI REIKIAMAS INDIVIDUALIŲ DĖSTYTOJŲ PLANŲ SUDARYMUI FUNKCIJAS, TAIP PAT SUFORMUOTI XML SRAUTĄ IŠ MS SQL SERVER DB Į ORACLE DB IR ATLIKTI ŠIO SRAUTO ANALIZĘ:**

3.1. Individualių dėstytojų planų įvedimas;

3.2. Individualių dėstytojų planų duomenų redagavimas;

3.3. Individualių dėstytojų planų pašalinimas;

3.4. MS SQL Server DB esančių duomenų nuskaitymas į XML failą;

3.5. XML faile esančių duomenų surašymas į Oracle DB.

##### **4. PARUOŠTI SISTEMOS NAUDOJIMO DOKUMENTUS:**

4.1. Sistemos vartotojo (sistemos administratoriaus) vadovą;

4.2. Interneto vartotojo vadovą.

##### **5. REIKALAVIMAI PROJEKTAVIMUI, PROGRAMINEI IR TECHNINEI ĮRANGAI**

5.1. Realizuoti sąsają su vartotoju per IE;

5.2. Duomenų bazės ir vartotojo sąsajai naudoti VB programavimo kalbą;

5.3. Iš MS SQL Server DB į Oracle DB suformuoti XML srautą;

5.4. Sistema turi funkcionuoti Windows 9x/XP ir NT/2000 aplinkose;

5.5. Reikalavimai techninei įrangai:

- Kompiuteris su Pentium procesoriumi;
- Operacinė sistema:
  - ✓ Microsoft Windows 9x/XP, NT/2000;
- Atmintis:
  - ✓ 64 MB RAM – minimali atmintis;
- Spalvotas monitorius;
- Pelė;
- Klaviatūra;
- Modemas ar tinklo plokštė (priėjimui prie Interneto).

6. REIKALAVIMAI DARBO PRISTATYMUĮ:

- 6.1. Pateikti darbo aprašą pagal pateiktą magistro darbo struktūrą;
- 6.2. Pateikti kompaktinį diską su sukurtos programos išeities kodu, vykdomaisiais ir/arba instaliavimo failais, kontroliniu pavyzdžiu, naudojimo instrukcija, magistro darbo tekstu ir pristatymo skaidrėmis;
- 6.3. Darbo gynimui pateikti pristatymą, paruoštą MS PowerPoint formatu;
- 6.4. Gynimo metu pademonstruoti individualių dėstytojų planų tvarkymą ir XML srauto sudarymą.

IFM-9/4 gr. studentas T. Zemblys

### 3.2. Kuriamos sistemos darbo proceso etapai

Darbe naudojamą duomenų bazių integracijos ir XML srauto sudarymo tarp MS SQL Server DB ir Oracle DB procesą galime suskirstyti į tokius etapus:



20 pav. Duomenų bazių integracijos etapai

#### 1 etapas:

Pagal užklausą iš MS SQL Server duomenų bazės yra išrenkami reikalingi duomenys.

#### 2 etapas:

Gauti duomenys iš MS SQL Server DB konvertuojami į XML formatą.

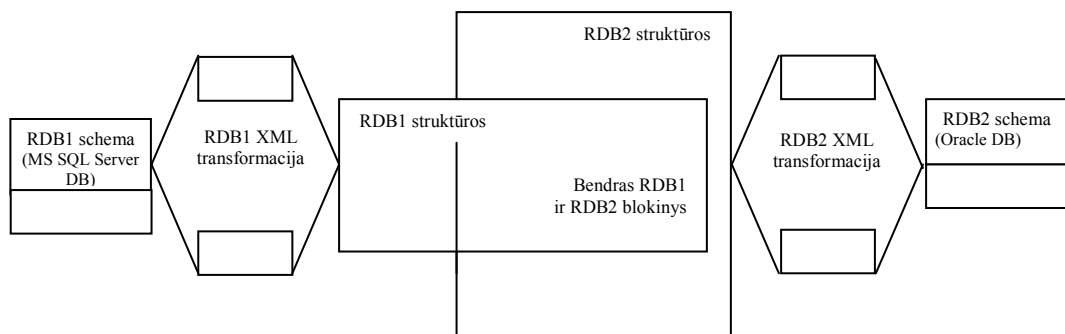
#### 4 etapas:

Gauto XML dokumento duomenys yra talpinami į Oracle DB atitinkamas lenteles.

Visi duomenų bazių integracijos etapai pasižymi svarbia integracijos savybe: „gavėjui“ ir „siuntėjui“ nėra svarbi adresato duomenų bazių struktūra, tačiau tokia integracija padeda išsaugoti duomenų konfidencialumą, t.y. vartotojai keičiasi tik tais duomenimis, kurie yra reikalingi.

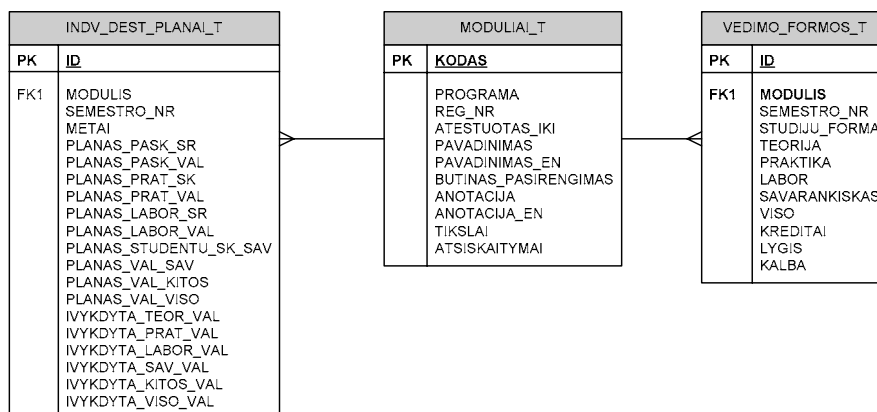
### 3.3. Replikavimo srauto išskyrimas

Šiame darbe naudojamos dvi reliacinės duomenų bazės (MS SQL Server ir Oracle) ir jos turi keistis duomenimis ir tiems duomenims suvienodinti yra naudojamas bendras blokiny. Bendro blokinio funkciją šiame darbe atliks XML.



21 pav. DB integracija panaudojant XML

Iš paveikslu matyti, kad turint dvi reliacines duomenų bazes reikia sukurti bendrą apsikeitimui duomenimis reikalingą blokinį. 22 pav. pateiktas duomenų bazių bendras blokinys:



22 pav. Duomenų bazių bendras blokinys

Naudojamų duomenų bazių schemas pateiktos priede (žr. 9.2 skyrelis). Bendro blokinio informacijai apsikeisti bus naudojamas XML.

### 3.4. Projektavimo metodų, priemonių parinkimas

Kuriama sistema turi suteikti galimybę pasiekti duomenis per Internetą naudojantis standartinėmis priemonėmis (Interneto naršykle). Kuriamos sistemos viena iš dalių – duomenų bazė ir kita dalis – skirta tos duomenų bazės dalies pasiekimui per Internetą. Duomenų bazė sukurta su MS SQL Server 2000. Internetinės sistemos dalies realizavimui galime naudoti įvairias programavimo kalbas (C#, C++, VB, Java).

Lentelė Nr. 3

ADO.NET (Microsoft) ir JDO (Sun Microsystems) produktų lyginamoji analizė

Pasirinkimo kriterijai	ADO.NET	JDO
Duomenų modelis	Reliacinis	Objektinis
Duomenų apibrėžimo kalba (Data Definition Language)	XML	Java ir XML
Užklausų kalba	SQL	JDOQL
Duomenų manipuliavimo kalba	SQL ir (C# arba C++ arba Visual Basic)	Java
Paveldėjimo ir polimorfizmo realizacija	Ne	Taip
Unikalus identifikatorius	Pirminis raktas	Pirminis raktas arba duomenų šaltinio identifikatorius
Transakcijos	Taip	Taip
Konkurentiškumas	Taip	Taip
Objektinio duomenų modelio palaikymas užklausose	Ne	Taip

Atlikus ADO.NET (Microsoft) ir JDO (Sun Microsystems) produktų lyginamąją analizę pasirinktas ADO.NET produktas, kadangi ADO.NET viskas yra saugoma vienoje XML byloje, kurią apdoroja speciali taikomoji programa. JDO technologijos trūkumas yra du šaltiniai, kadangi kuriant Java klases sąsajai su duomenų šaltiniu naudojamas Java išeities tekstas ir XML byloje užrašyti metaduomenys [5]. Kuriamos sistemos internetinės dalies realizavimui pasirinkta Visual Basic (VB) programavimo kalba.

### **3.5. Kompiuterizuojamos sistemos varianto parinkimas**

#### **3.5.1. Sudėties ir architektūros pasirinkimas**

Informacinei sistemai realizuoti pasirinkta XML kalba ir Visual Basic.NET programavimo kalbos. Programa jungiasi prie duomenų bazės esančios Microsoft SQL Server 2000. Duomenys yra saugomi ir Microsoft SQL Server 2000 ir Oracle Server 8.1 duomenų bazėse. Tarp šių duomenų bazių formuojami duomenų srutai duomenų bazėse esančiai informacijai apsikeisti. Duomenų srutai formuojami XML formate. Sąsaja su vartotoju realizuota naudojant ASPX (*Active Server Pages*) puslapius, duomenų bazės ir ASPX puslapių sąsajai naudojami VB.NET programavimo kalba.

#### **3.5.2. Programavimo ir projektavimo įrankių pasirinkimas**

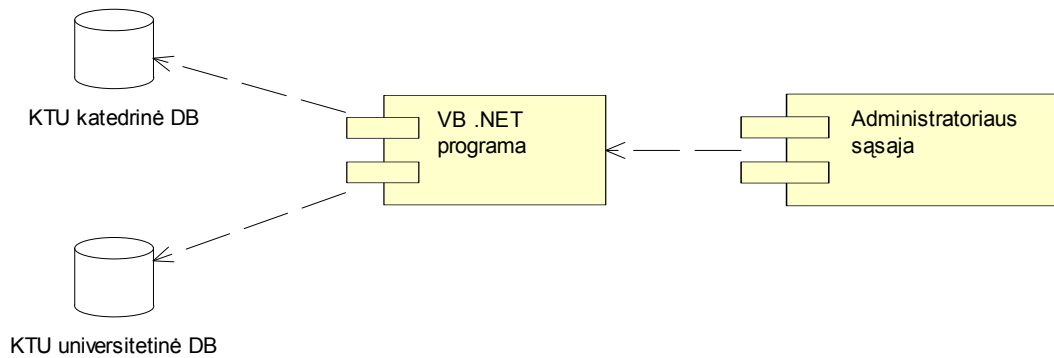
Sistemos kūrimo įrankiais pasirinkti paketai Microsoft Visual Studio.NET 2003, Microsoft SQL Server 2000 ir Oracle Server 8.1. Paketas Microsoft Visual Studio.NET 2003 naudojamas programos kūrimui ir redagavimui. Paketai Microsoft SQL Server 2000 ir Oracle Server 8.1 naudojami atitinkamų duomenų bazių palaikymui, priėjimui prie duomenų realizuoti ir prisijungimui prie duomenų bazių.

#### **3.5.3. Dokumentacijos ruošimo įrankių pasirinkimas**

Dokumentacijos ruošimo bei projektavimo įrankiais pasirinkti „Rational Rose 2000 Enterprise Edition“ ir „Microsoft Visio Professional 2002“ paketai. Dokumentacijai rašyti buvo pasirinktas Microsoft Word 2000 paketas.

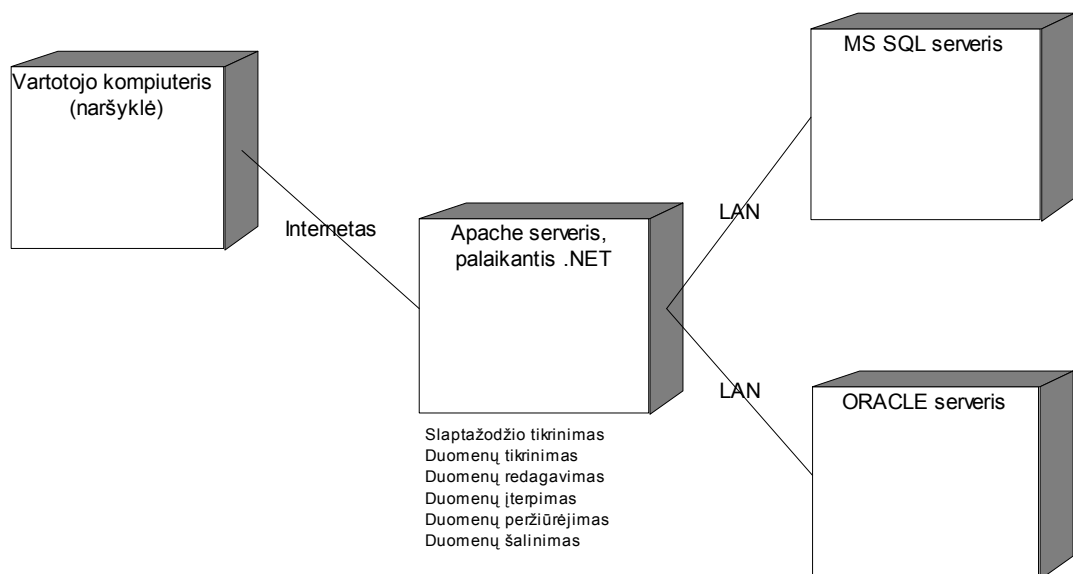
## 3.6. Informacinės sistemos komponentai

### 3.6.1. Komponentų diagrama



23 pav. Komponentų diagrama

### 3.6.2. Paskirstymo diagrama

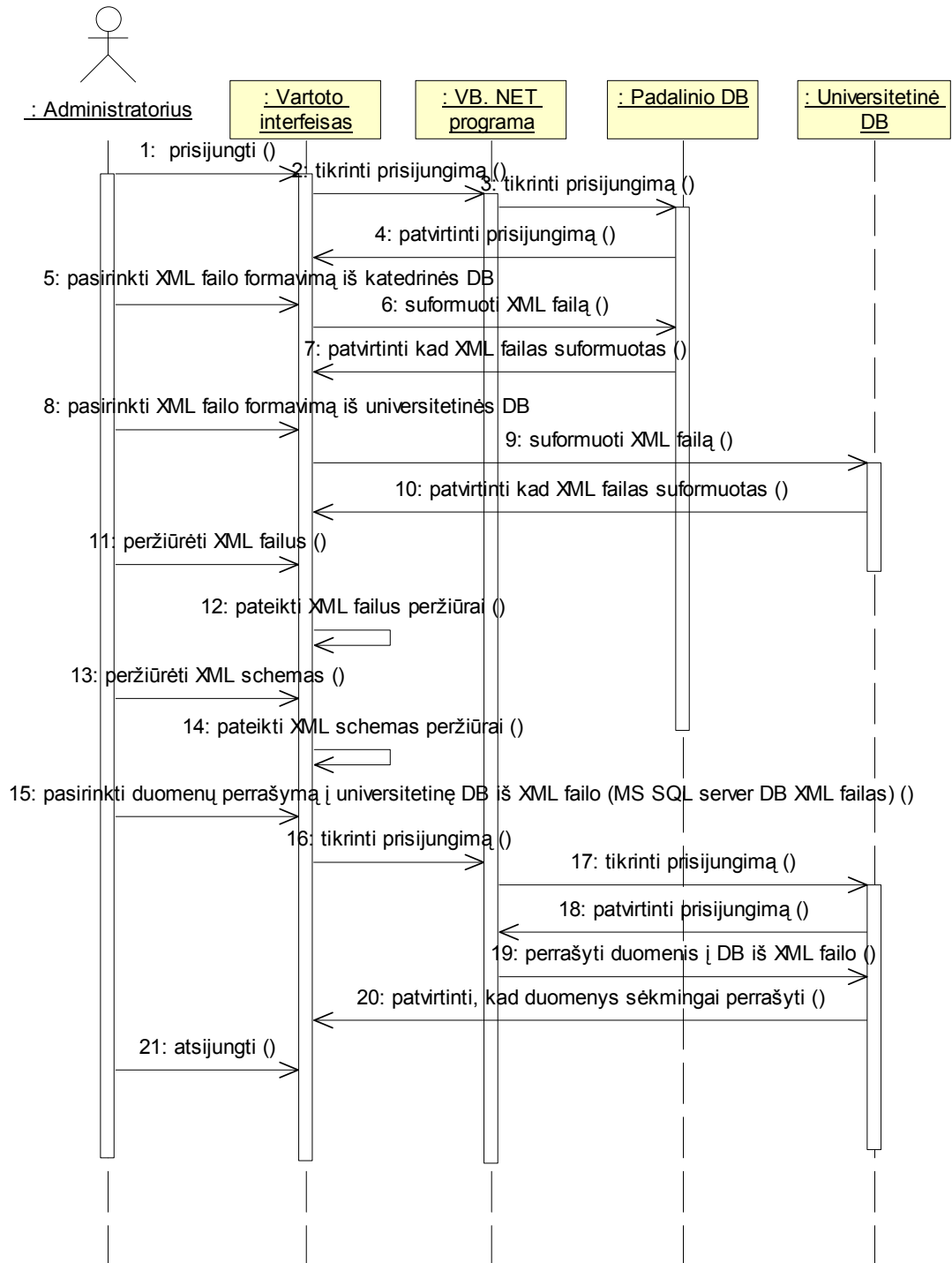


24 pav. Paskirstymo diagrama

Vartotojas gali prisijungti prie sistemos iš savo kompiuterio paleidęs Interneto naršyklę. VB .NET valdo jungimąsi prie MS SQL serverio, kuriame saugoma informacija gali būti peržiūrima, trinama, pildoma, redaguojama. VB .NET valdo ir jungimąsi prie Oracle serverio, į kurį persiunčiami duomenys iš MS SQL serverio, naudojant XML.

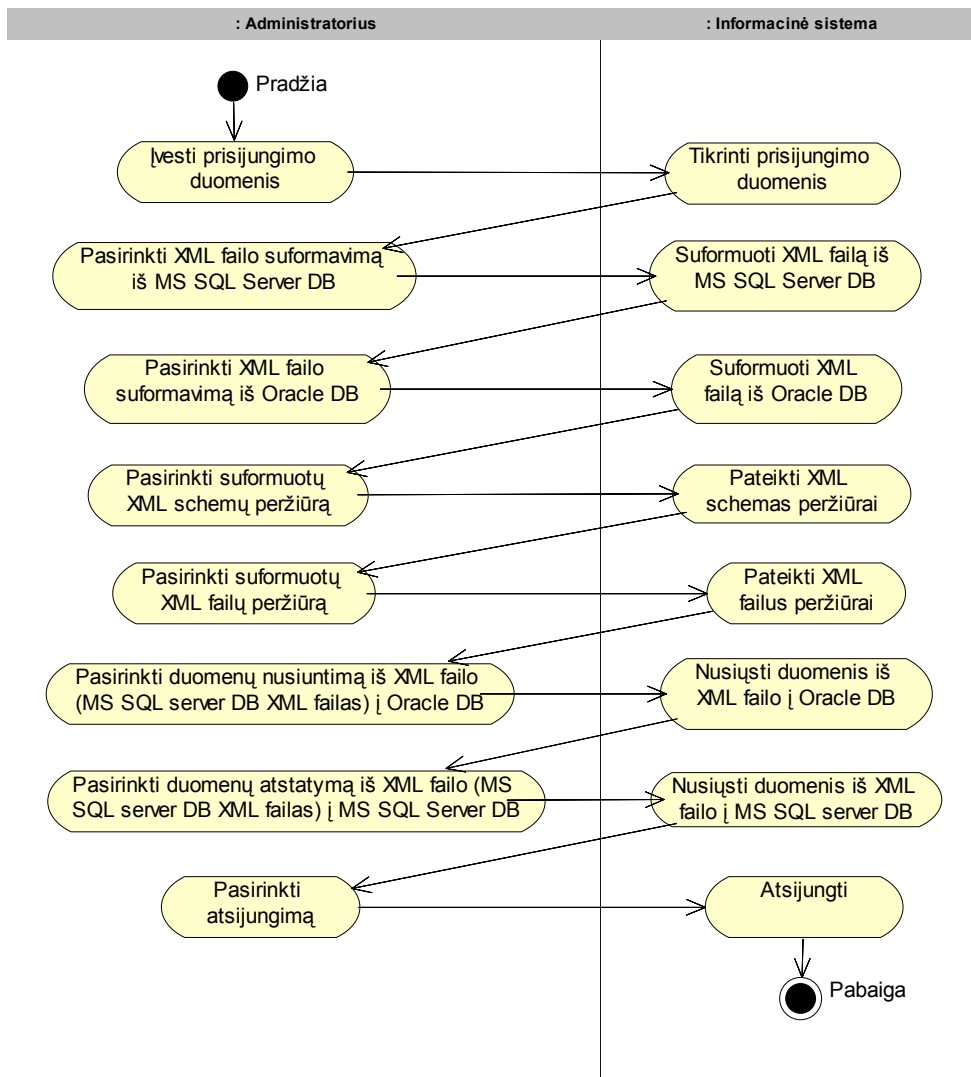
### 3.7. Sistemos administratoriaus sekų ir elgsenos diagramos

#### 3.7.1. Duomenų persiuntimo į universitetinę DB sekų diagrama



25 pav. Duomenų persiuntimo į universitetinę DB sekų diagrama

### 3.7.2. Sistemos elgsenos diagrama, kai vartotojas administratorius



26 pav. Sistemos elgsenos diagrama, kai vartotojas administratorius

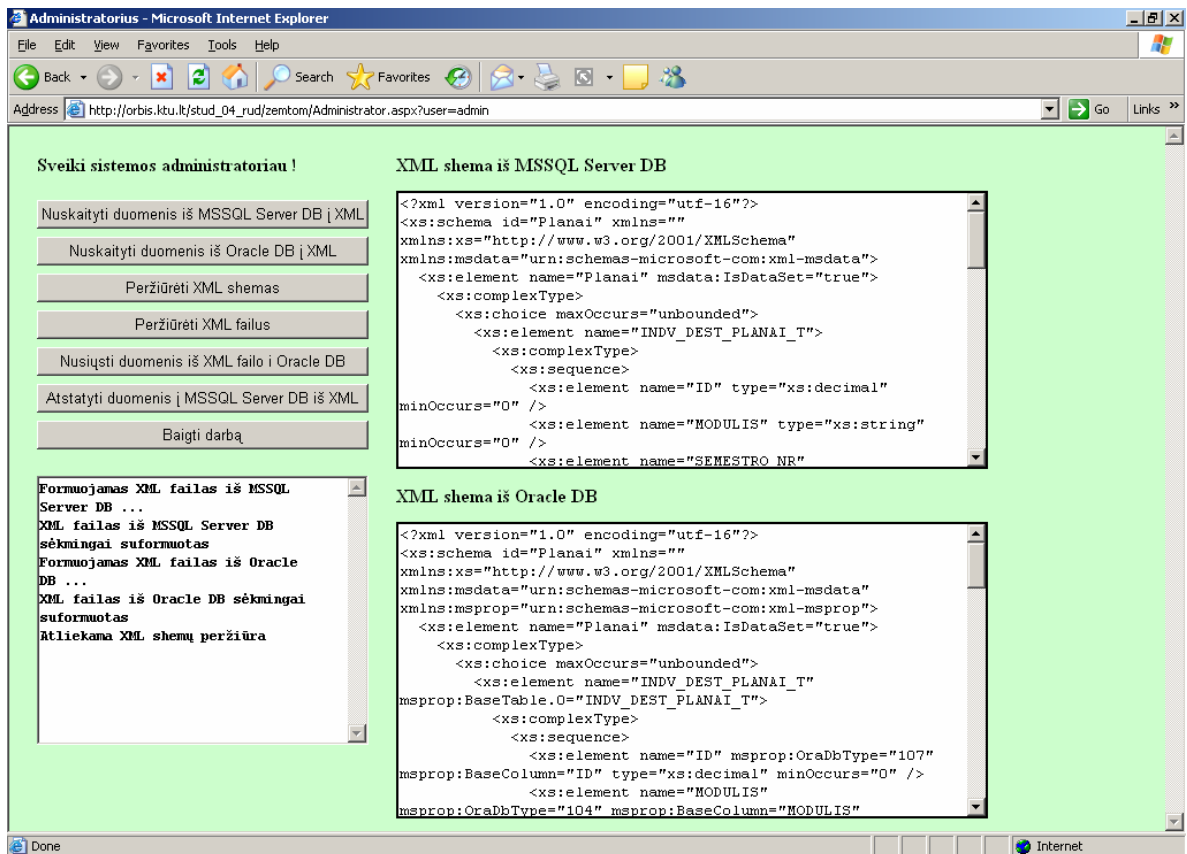


### 3.8. Testavimo modelis bei duomenys, kontrolinis pavyzdys

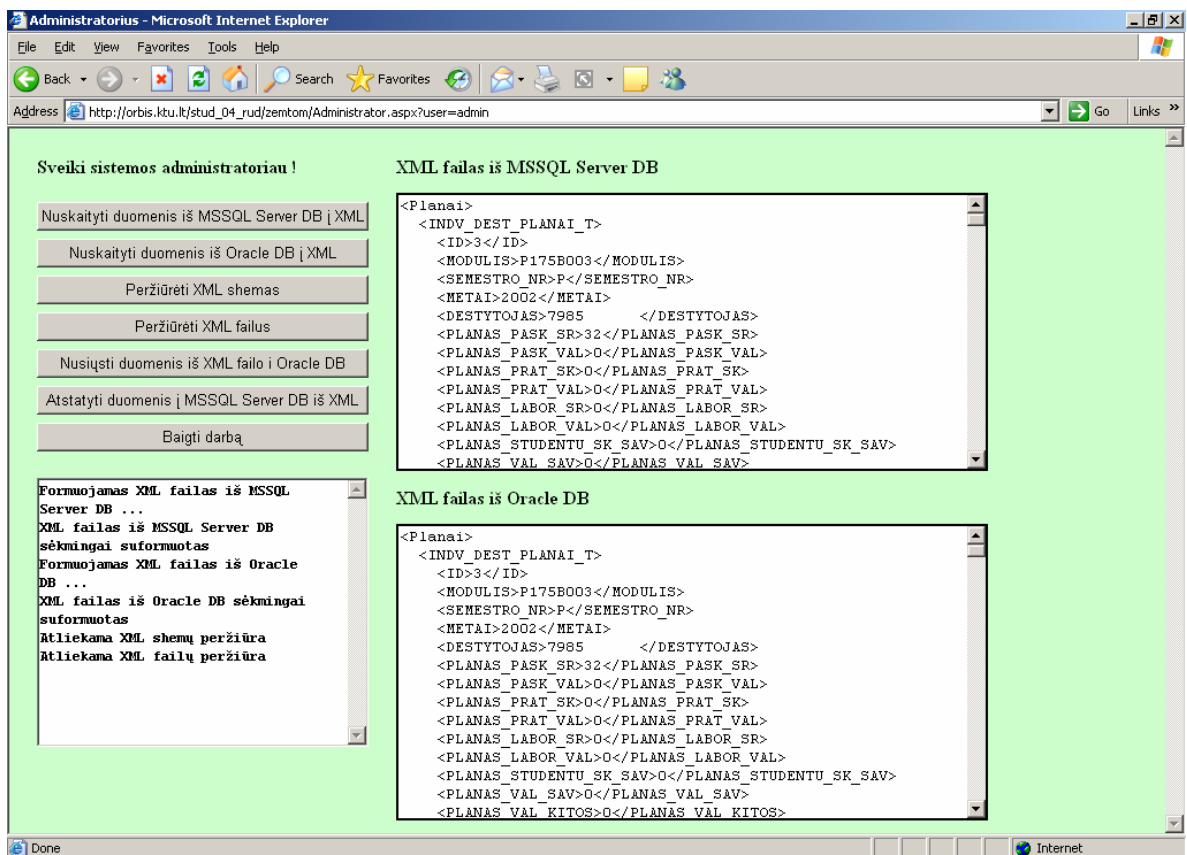
Testavimo metu buvo tikrinama, kaip funkcionuoja sukurta sistema, ar ji teisingai atlieka reikalavimuose specifikuotas funkcijas, kurios turi patenkinti vartotojo poreikius:

- Ar teisingai vykdomas prisijungimas prie sistemos, ar sistema atrenka koks vartotojas prisijungė ir pagal tai suteikia jam atitinkamas teises.
- Ar teisingai nuskaitomi iš duomenų bazės ir išvedami individualūs dėstytojų planai.
- Ar leidžiama priskirti naują informaciją apie modulį individualiam pasirinkto dėstytojo planui ir ar priskiriant naują modulį tikrinamas įvedamos informacijos korektiškumas.
- Ar leidžiama ištrinti pasirinktą informaciją apie modulį iš individualaus pasirinkto dėstytojo plano.
- Ar leidžiama koreguoti pasirinktą informaciją apie modulį ir ar įrašant pakoreguotą informaciją į DB tikrinamas įrašomos informacijos korektiškumas.
- Ar priskyrus arba pašalinus informaciją apie modulį iš individualaus pasirinkto dėstytojo plano ar pakoregavus informaciją apie modulį atnaujinama informacija apie individualius dėstytojų planus.
- Ar suformuojamas XML failas ir XML schema iš MS SQL Server DB.
- Ar suformuojamas XML failas ir XML schema iš Oracle DB.
- Ar galima peržiūrėti XML failus suformuotus iš MS SQL Server DB ir Oracle DB.
- Ar galima peržiūrėti XML schemas suformuotas iš MS SQL Server DB ir Oracle DB.
- Ar galima nusiųsti duomenis iš XML failo (XML failas iš MS SQL Server DB) į Oracle DB.
- Ar galima atstatyti duomenis iš XML failo (XML failas iš MS SQL Server DB) į MS SQL Server DB.
- Ar visuose languose yra numatyta galimybė grįžti atgal ar baigti darbą.
- Ar pateikiami pranešimai apie atliktus veiksmus.

Sistema buvo testuojama su duomenimis, kurie yra įrašyti į MS SQL Server DB ir į Oracle DB, kurios yra patalpintos serveryje. Šie duomenys nėra realūs, tačiau atitinka realių DB struktūras ir galimus įrašus.



27 pav. Suformuotų XML schemų iš MS SQL Server DB ir Oracle DB peržiūros kontrolinis pavyzdys



28 pav. Suformuotų XML failų iš MS SQL Server DB ir Oracle DB peržiūros kontrolinis pavyzdys

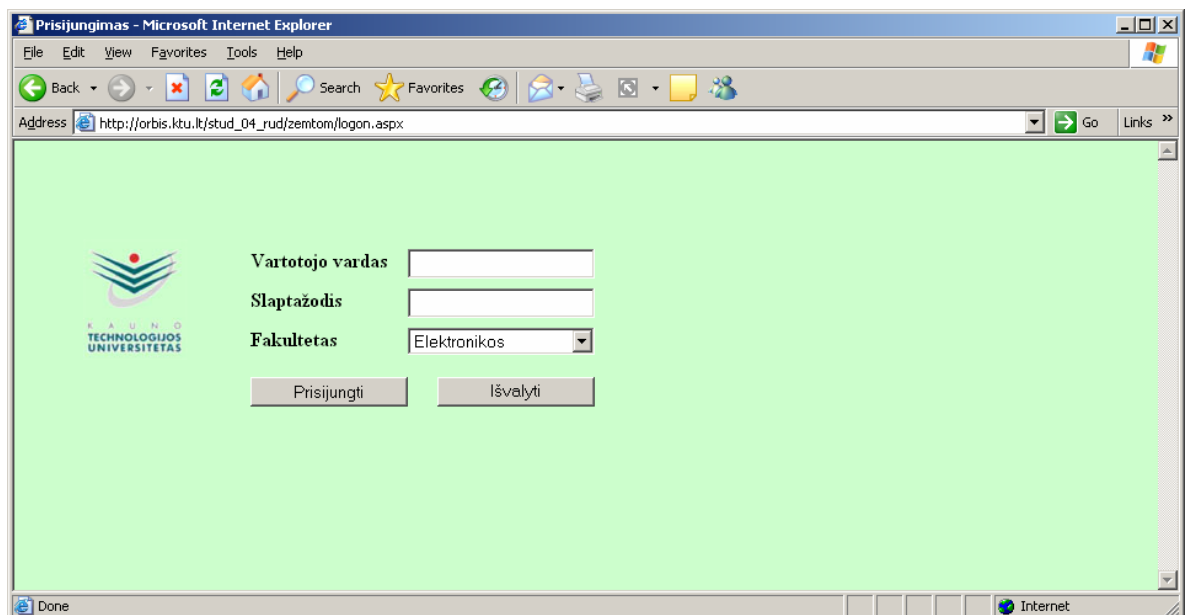
### 3.9. Sistemos administratoriaus – sistemos vartotojo naudojimo instrukcija

Šiame skyriuje aprašysime leistinas sistemos administratoriui funkcijas, jų atlikimo tvarką ir patarimus.

#### ▪ Prisijungimas

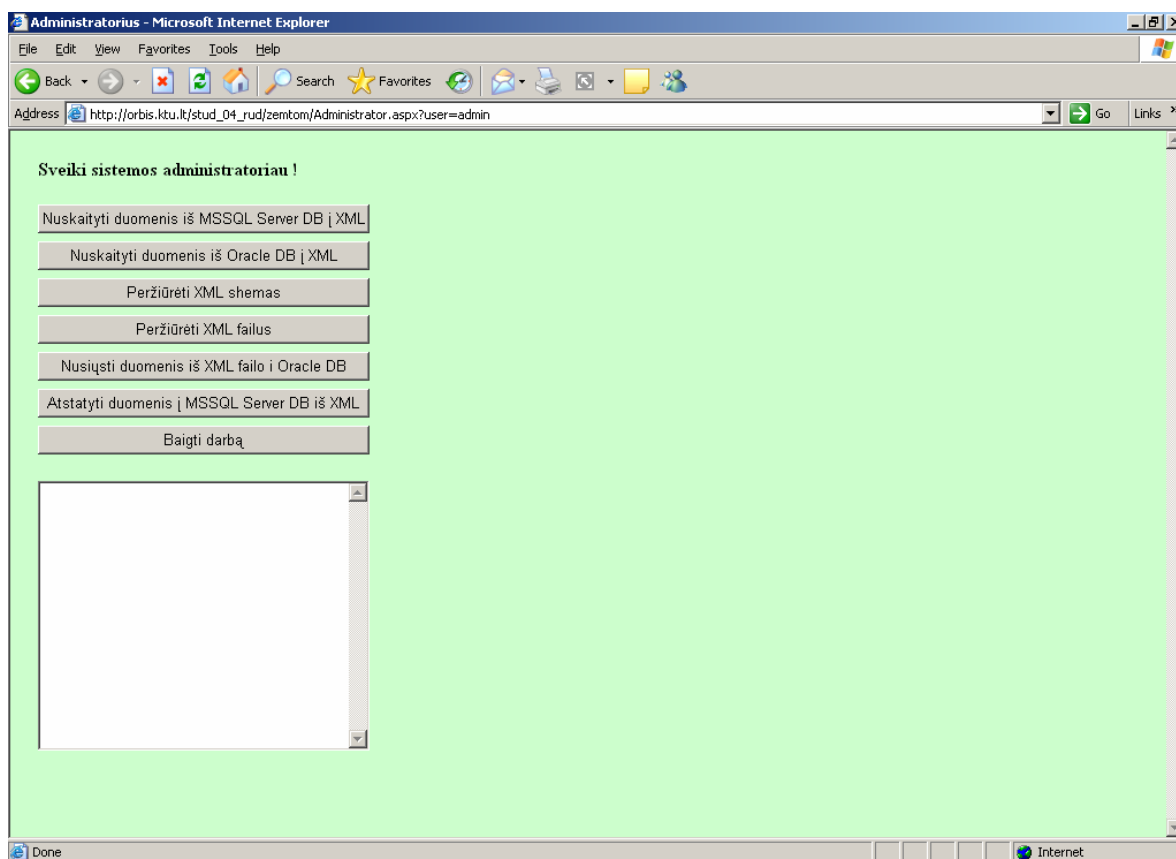
Prisijungimas atliekamas naudojantis viena iš reikalavimuose išvardintų naršyklių įvedant adresą, kuriuo yra patalpinta programa, adreso įvedimo lauke. Programa patalpinta serveryje orbis.ktu.lt adresu [http://orbis.ktu.lt/stud\\_04\\_rud/zemtom/logon.aspx](http://orbis.ktu.lt/stud_04_rud/zemtom/logon.aspx). Gali tekti palaukti kurį laiką kol bus užkrautas titulinis sistemos puslapis.

#### ▪ Prisijungimo langas



29 pav. Prisijungimo prie sistemos langas

Administratorius prie sistemos prisijungia įveddamas prisijungimo vardą ir slaptažodį, kurie yra sukurti administratoriui atskirai. Įvedęs duomenis paspaudžia mygtuką „Prisijungti“. Jei įvedant prisijungimo vardą ar slaptažodį nebuvo padaryta klaidų – sistema prisijungs, jei ne sistema praneš apie klaidą. Prisijungus administratoriui, jam bus užkrautas pagrindinis sistemos langas. 29 pav. Parodytas administratoriaus pasirinkimo langas.

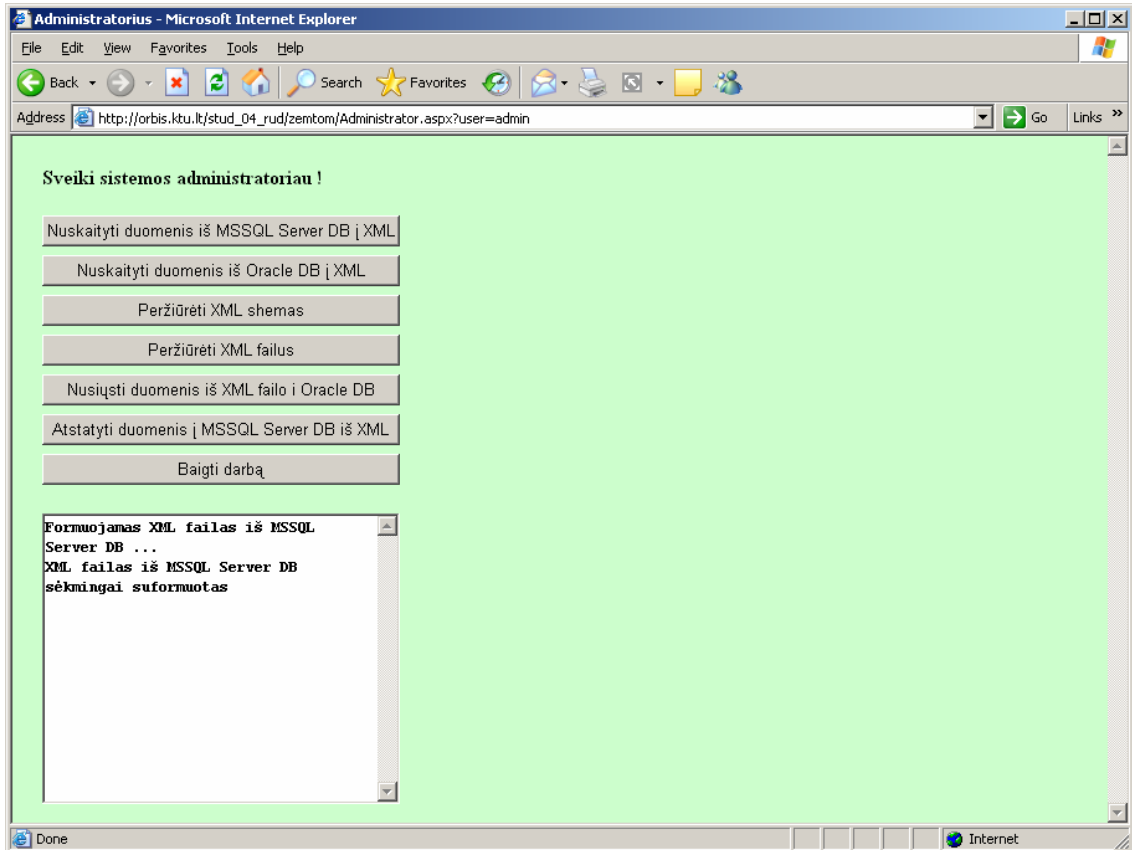


30 pav. Prisijungimo prie sistemos langas

Administratorius gali atlikti veiksmus naudodamasis pateiktomis nuorodomis - spustelėjęs pele norimą nuorodą atlieka norimą veiksmą. Administratorius gali pasirinkti tokius variantus:

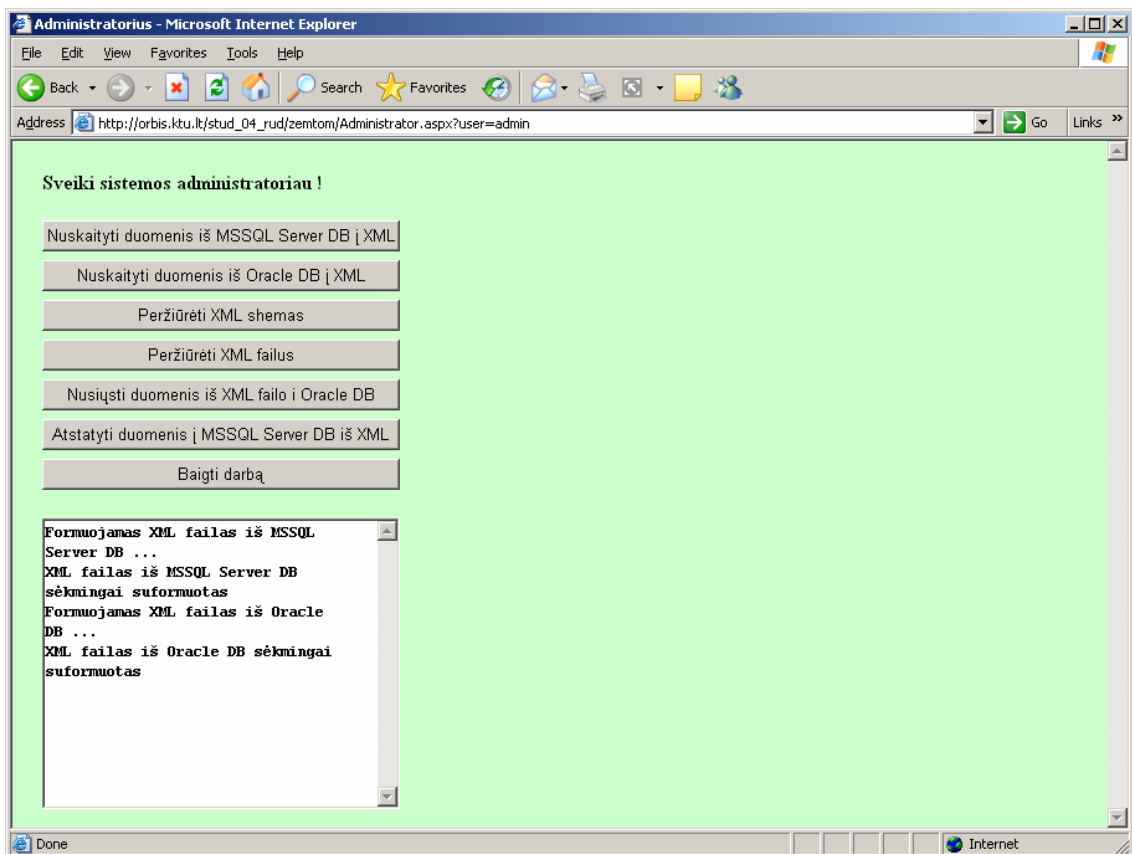
1. Nuskaityti duomenis iš MSSQL Server DB į XML – suformuoti XML failą ir XML schemą iš MS SQL Server DB esančių duomenų;
2. Nuskaityti duomenis iš Oracle DB į XML – suformuoti XML failą ir XML schemą iš Oracle DB esančių duomenų;
3. Peržiūrėti XML shemas – atlikti suformuotų XML schemų peržiūrą;
4. Peržiūrėti XML failus – atlikti suformuotų XML failų peržiūrą;
5. Nusiųsti duomenis iš XML failo į Oracle DB – surašyti duomenis iš XML failo (iš MS SQL Server DB suformuotas XML failas) į Oracle esančią DB;
6. Atstatyti duomenis į MSSQL Server DB iš XML – surašyti duomenis iš XML failo (iš MS SQL Server DB suformuotas XML failas) į MS SQL Server esančią DB;
7. Baigti darbą – darbo su sistema pabaiga.

- Veiksmas „Nuskaityti duomenis iš MSSQL Server DB į XML“



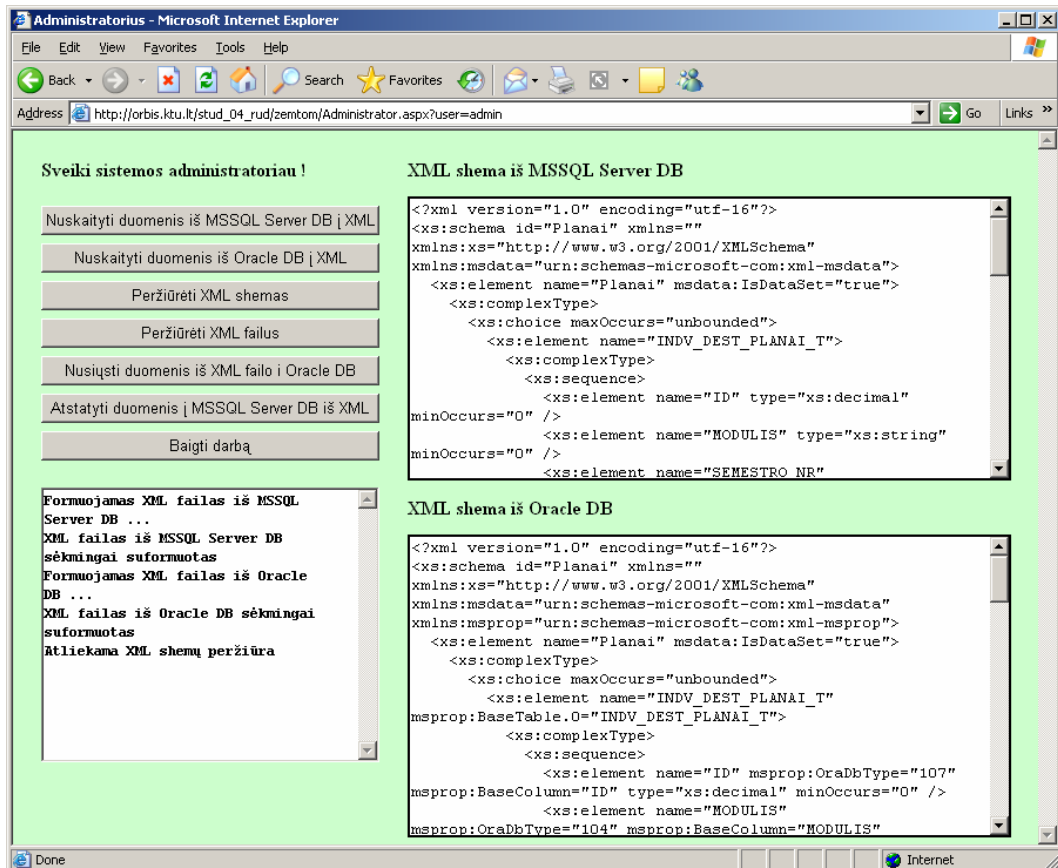
31 pav. Veiksmo „Nuskaityti duomenis iš MSSQL Server DB į XML“ langas

- Veiksmas „Nuskaityti duomenis iš Oracle DB į XML“



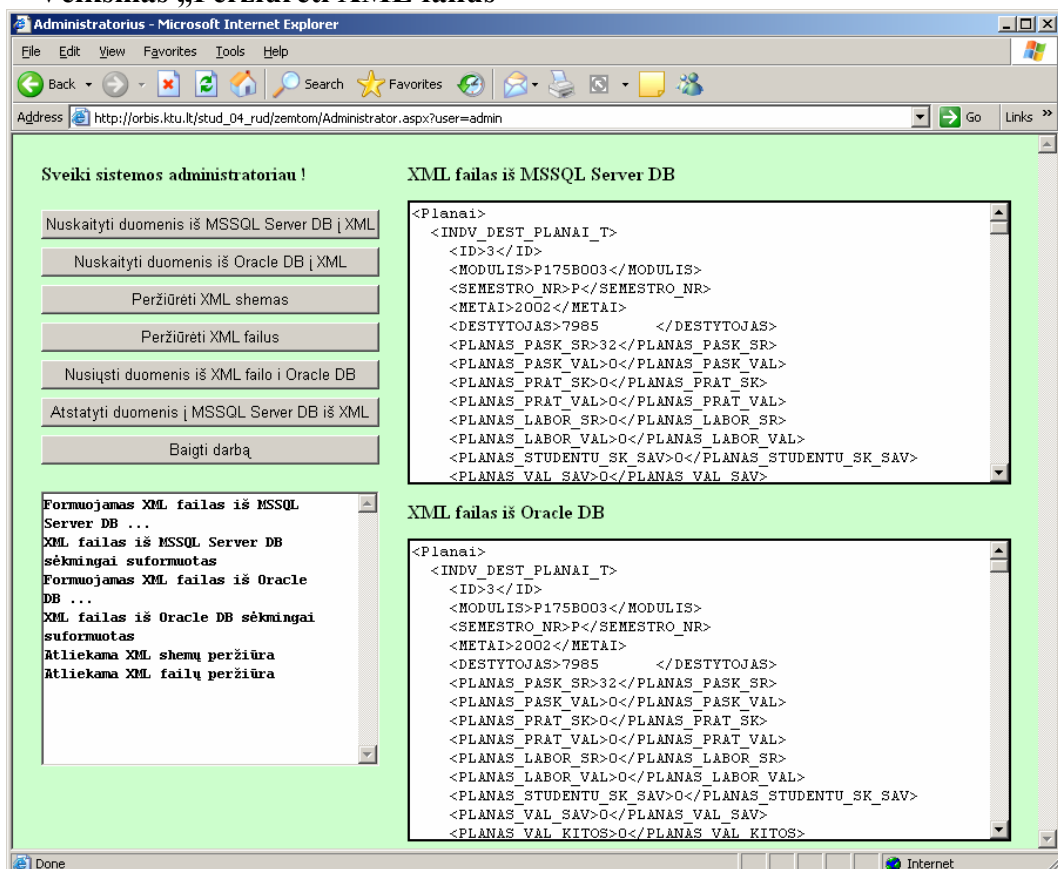
32 pav. Veiksmo „Nuskaityti duomenis iš Oracle DB į XML“ langas

- Veiksmas „Peržiūrėti XML schemas“



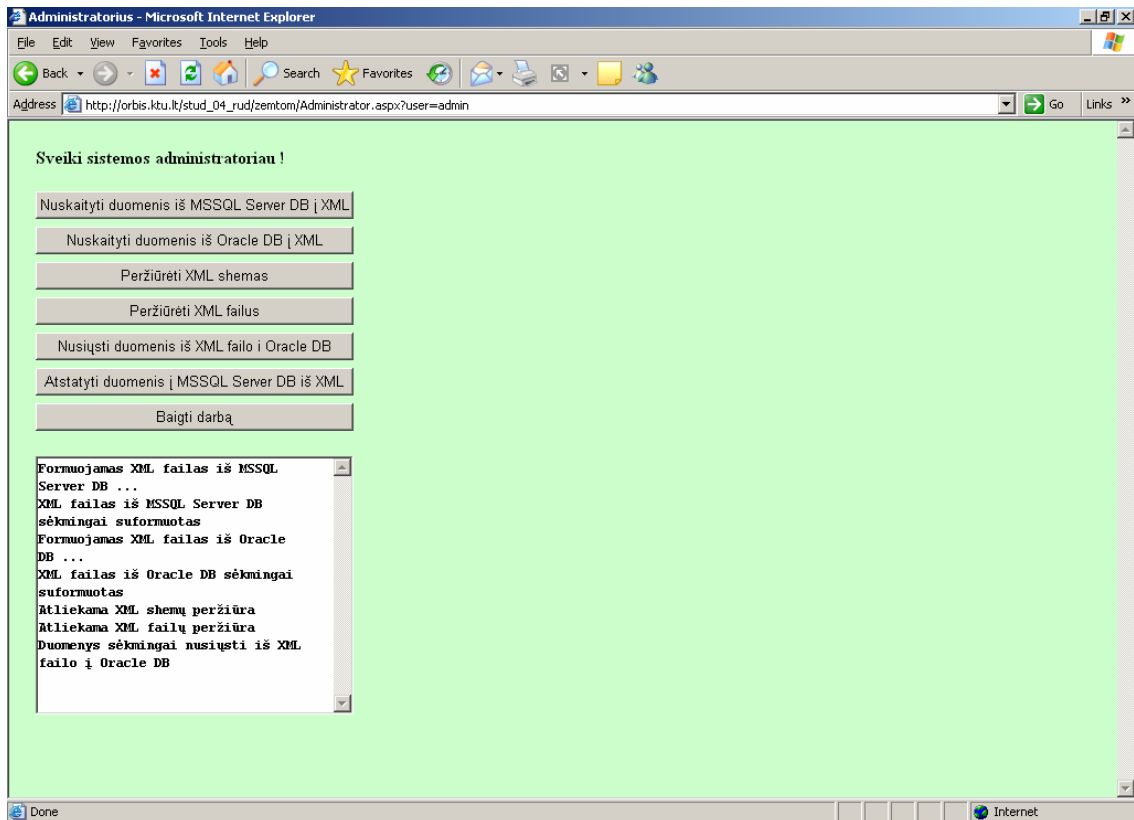
33 pav. Veiksmo „Peržiūrėti XML schemas“ langas

- Veiksmas „Peržiūrėti XML failus“



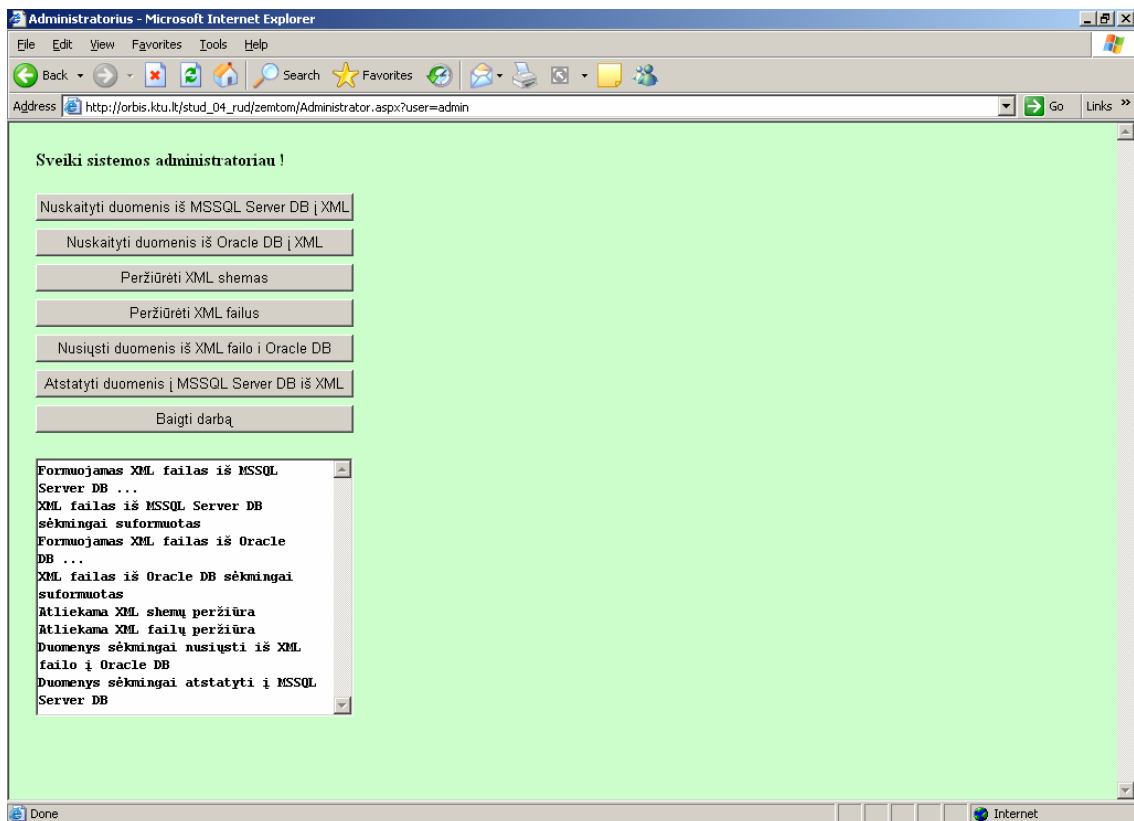
34 pav. Veiksmo „Peržiūrėti XML failus“ langas

▪ Veiksmas „Nusiųsti duomenis iš XML failo i Oracle DB“



35 pav. Veiksmo „Nusiųsti duomenis iš XML failo i Oracle DB“ langas

▪ Veiksmas „Atstatyti duomenis į MSSQL Server DB iš XML“



36 pav. Veiksmo „Atstatyti duomenis į MSSQL Server DB iš XML“ langas

### **3.10. Uždavinio techninis išpildymas**

Visa reikalinga medžiaga sprendžiamo uždavinio techniniam išpildymui pateikta priede (žr. Priedas Nr. 3 Uždavinio techninis išpildymas). Šiame skyriuje pateiktas kuriamos sistemos kokybės kriterijus, visų sistemos vartotojų panaudojimo atvejų diagrama, panaudojimo atvejų specifikacijos, dalykinės srities klasių diagrama ir vartotojo interfeiso modelis. Pateikiamos sistemos panaudojimo atvejų diagramos, statinės struktūros ir elgsenos modeliai. Taip pat pateikti informacinei sistemai keliami nefunkciniai reikalavimai, reikalavimai sistemos funkcionalumui ir reikalavimai sukurtos sistemos palaikymui. Galima rasti ir sistemos naudojimo instrukciją skirtą padalinio vadovui ir dėstytojui.

### **3.11. Eksperimentinis tyrimas**

#### **3.11.1. Sukurtos sistemos kokybės tikrinimas**

Pagrindinis kokybės kriterijus – veikianti programinė įranga. Pagrindinė sukurtos programinės įrangos funkcija - visos duomenų bazės ar jos dalies nuskaitymas iš duomenų bazės, esančios MS SQL serveryje, ir konvertavimas į XML kalbą, duomenų išsaugotų XML faile surašymas į Oracle duomenų bazę, bei pateikimas duomenų bazėje (MS SQL Server DB) esančių duomenų vartotojui.

Sukurta programinė įranga atitinka jai keliamus funkcinis ir nefunkcinius reikalavimus bei vartotojo poreikius.

Sukurtoje programoje galima įvesti naujus duomenis, redaguoti ir šalinti juos. Galima suformuoti XML failą iš MS SQL Server DB esančių duomenų ir surašyti duomenis iš XML failo į Oracle DB. Duomenų įvedimo, redagavimo ir šalinimo metu yra atliekama duomenų kontrolė.

Apžvelgus į sukurtos programinės įrangos funkcionalumą, galime teigti, kad sukurta informacinė sistema tenkina apibrėžtus kokybės kriterijus.

#### **3.11.2. Tolimesnio sistemos tobulinimo, plėtojimo galimybės**

Sukurta programinė įranga yra paruošta naudojimui ir atitinka jai keliamus vartotojo reikalavimus. Sistemą galima tobulinti ir plėtoti praplečiant jos vartotojų reikalavimus (pvz.: pateikti modulį pasirinkusių studentų sąrašus ir t.t.).



Sukurta sistema atlieka tik MS SQL Server duomenų bazės ir Oracle duomenų bazės duomenų integravimą, todėl sistemą galima tobulinti šioje srityje, pritaikant jos pateiktą funkcionalumą kitokioms duomenų bazėms ar universaliam pritaikymui, kai vartotojas pats pasirenka ir nurodo, kokias reliacines duomenų bazes jis nori naudoti.

### **3.12. Sukurto valdiklio intelektualizavimo galimybės**

Tolimesnė darbo vystymo kryptis – intelektualizuoti sukurto administratoriaus (duomenų paskirstymo) valdiklio darbą:

- nustatyti komunikacinės kilpos formalius parametrus (vartotojų identifikatorius, srautų identifikatorius, verslo transakcijų laiko apribojimus ir kitus). Siekiant nustatyti komunikacinės kilpos formalius parametrus reikia sukurti šios kilpos metaduomenų modelį. Šis metaduomenų modelis apibrėžia, kokios esybės (informacijos objektai) turi būti apibrėžtos reikalavimų specifikacijoje, ir kokios priklausomybės sieja šias esybes su kitomis esybėmis ir sąsajomis [7].
- sudėtingų naujų funkcinių uždavinių atveju reikia išplėtoti sudėtinių verslo transakcijų formavimo iš pavienių kilpų metodą, panaudojant dinamines *exist* ir *coexist* priklausomybes [10].

## IŠVADOS

- Vartotojo poreikių analizės dalyje palygintos dvi metodikos (UML ir komunikacinių kilpų) skirtos organizacijos veiklai modeliuoti.
- Pateikta nagrinėjamos organizacijos veikla UML ir komunikacinių kilpų požiūriu. Pateiktoje komunikacinėje kilpoje Individualus dėstytojo planas / Krūvio vykdymas (13 pav.) iliustruojamas sudarytos komunikacinės kilpos gyvybingumas.
- Binarinių komunikacinių kilpų metodas plėtojamas Informacijos sistemų katedroje (ISK) ir jo pagrindinis privalumas yra tas, kad galima griežtai atskirti funkcinis reikalavimus nuo nefunkcinių. Binarinių komunikacinių kilpų metodas aktualus, kai veikianti informacijos sistema yra perprojektuojama nedidelėmis dalimis ir po pakartotinės inžinerijos lieka panaudotos palikuoninės duomenų bazės.
- Pateikta komunikacinių kilpų srautų stereotipinis modelis, kuriuo numatomi duomenų mainai tarp dviejų vartotojų tipų. Išanalizuotas komunikacinės kilpos replikavimo architektūrinis sprendimas (RAS), kai:
  - vartotojai susiejami per paskirstytas replikuojančias duomenų bazes;
  - duomenų mainai tarp vartotojų atliekami, naudojant XML srautus.
- RAS privalumai:
  - leidžia panaudoti palikuonines duomenų bazes;
  - leidžia spręsti palikuoninius uždavinius;
  - skirtingi vartotojų tipai gali dirbti nehomogeninėse aplinkose.
- Apžvelgtos XML technologijos duomenų bazių valdymo sistemose ir pateiktas Oracle ir MS SQL Server 2000 teikiamų XML savybių palyginimas. Microsoft SQL Server 2000 siūlo XML savybes daugiausia skirtas Web ir duomenų bazių programuotojams ir teikia standartines XML konstrukcijas. Oracle labiau skirtas naudoti patyrusiems programuotojams. Norint pasiekti sprendžiamų uždavinių rezultatą tenka naudotis keliomis priemonėmis arba kurti specifinius įrankius.
- Valdiklio programinei įrangai realizuoti pasirinktas Microsoft Visual Studio .NET 2003 programavimo paketas (Visual Basic (VB) programavimo kalba), nes tai suteikti galimybę pasiekti duomenis per Internetą naudojantis standartinėmis priemonėmis (Interneto naršykle) ir galimas XML panaudojimas.

- Praktiškai pritaikyta ir realizuota valdiklio programinė įranga, kurios pagalba galima:
  - Suformuoti XML failą iš MS SQL Server DB;
  - Nuskaityti duomenis į XML failą iš Oracle DB;
  - Surašyti duomenis iš XML failo į Oracle DB.
- Sukurta programinė įranga leidžia atlikti MS SQL Server ir Oracle duomenų bazėse esančių duomenų apjungimą, panaudojant XML.
- Programinė įranga realizuota atsižvelgiant į KTU Informacijos sistemos katedros veiklą.
- Tolimesnė darbo vystymo kryptis – intelektualizuoti sukurto administratoriaus (duomenų paskirstymo) valdiklio darbą:
  - nustatyti komunikacinės kilpos formalius parametrus (vartotojų identifikatorius, srautų identifikatorius, verslo transakcijų laiko apribojimus ir t.t.).
  - sudėtingų naujų funkcinių uždavinių atveju reikia išplėtoti sudėtinių verslo transakcijų formavimo iš pavienių kilpų metodą, panaudojant dinamines *exist* ir *coexist* priklausomybes.

## LITERATŪRA

- [1] Extensible Markup Language (XML) 1.0 (Third Edition). Prieiga per internetą:  
<http://www.w3.org/TR/REC-xml> [žr. 2003-12-05]
- [2] Gustas R. Towards a communication based approach for enterprise modelling integration / 2-nd International conference on Information Reuse and Integration : proceedings of the ISCA, November 1-3, 2000. - Hawaii, USA : ISCA, 2000. - ISBN 1-880843-36-6. - p. 1-5.
- [3] Knight B. SQL Server 2000 for Experienced DBAs. 2003.
- [4] Konovalovas T., Paradauskas B. Nehomogeninių duomenų bazių integracija naudojant XML formatus // Informacinės technologijos ir mokslų integracija - 2003 = Information technologies and integration of science - 2003 : aštuntoji magistrantų ir doktorantų konferencija : konferencijos pranešimų medžiaga, [Kaunas, 2003 gegužės 23 d.] / Vilniaus universitetas. Kauno humanitarinis fakultetas ; [atsakingasis redaktorius V. Sakalauskas]. - Kaunas : Technologija, 2003. - ISBN 9955-09-427-3. - p. 45-48.
- [5] Kontrimas V. Java sąsaja su reliacinėmis duomenų bazėmis // Informacinės technologijos 2003 : konferencijos pranešimų medžiaga, Kaunas 2003 sausio 28-29 d. / Kauno technologijos universitetas. - Kaunas : Technologija, 2003. - ISBN 9955-09-335-8. - p. III-31 - III-34.
- [6] Lankelis V., Švažinskas M., Zemblys T., Paradauskas B. IS apskaitos paketų integracija // Informacinė visuomenė ir universitetinės studijos : 9-oji tarpuniversitetinė magistrantų ir doktorantų konferencija [2004 m. balandžio 15 d., Kaunas] : pranešimų medžiaga / Vytauto Didžiojo universitetas, Kauno technologijos universitetas, Vilniaus universitetas ; [atsakingas redaktorius R. Krikštolaitis; redakcinė kolegija: V. Kaminskas...[et al.]. - Kaunas : VDU, 2004. - p. 87-93.
- [7] Nemuraitė L., Paradauskas B. From Use Cases to Well Structured Conceptual Schemas // Information Systems Development: Advances in Theory, Practice and Education. (priimtas spausdinti).
- [8] Nemuraitė L., Paradauskas B., Salelionis L. Extended communicative action loop for integration of new functional requirements // Informacinės technologijos ir valdymas = Information technology and control. ISSN 1392-124X. - Kaunas : Technologija. - 2002, nr. 2(23), p. 18-26.

- [9] Paradauskas B., Nemuraitė L. Verslo transakcijų modeliavimas / Duomenų bazės ir semantiniai modeliai. Kaunas : Technologija, 2002. - ISBN 9955-09-436-2. - p. 227-243.
- [10] Paradauskas B., Šidlauskas K. Duomenų dinaminių priklausomybių naudojimas modeliuojant verslo transakcijas // Informacijos mokslai. Vilnius : Vilniaus universitetas. (priimtas spausdinti).
- [11] SQL Server 2000 XML Overview. Prieiga per internetą:  
<http://www.microsoft.com/technet/prodtechnol/sql/2000/evaluate/xmlsql.msp> [žr. 2004-12-04]
- [12] SQL Server 2000. Prieiga per internetą:  
<http://www.rpbouret.com/xml/ProdsXMLEnabled.htm#sqlserver> [žr. 2004-12-04]
- [13] Using XML in Oracle Database Applications. Prieiga per internetą:  
[http://www.oracle.com/technology/tech/xml/info/htdocs/otnwp/about\\_oracle\\_xml\\_products.htm](http://www.oracle.com/technology/tech/xml/info/htdocs/otnwp/about_oracle_xml_products.htm) [žr. 2004-12-04]
- [14] XML-Enabled Databases. Prieiga per internetą:  
<http://www.rpbouret.com/xml/XMLDatabaseProds.htm#xmlenabled> [žr. 2004-12-03]

## TERMINŲ IR SANTRUMPŲ ŽODYNAS

**XML (eXtensible Markup Language)** – išplėstinė žymių kalba.

**XML dokumentas** – XML kalba aprašytas dokumentas.

**XML schema** – XML kalba parašytas dokumentas, skirtas XML dokumentams apibrėžti.

**SQL (Structured Query Language)** – struktūrizuota užklausų kalba. SQL yra skirta reliacinių duomenų bazių duomenų apdorojimui. SQL komandos gali būti naudojamos interaktyviam darbui su duomenų baze arba gali būti įdiegta į taikomąsias programas kaip duomenų bazės sąsaja. SQL palaikančios duomenų bazės: DB2, SQL/DS, Oracle, Sybase, SQLbase, INFORMIX ir CA-OpenIngres (Ingres) ir kt.

**ODBC (Open DataBase Connectivity)** – Microsoft duomenų bazių programavimo sąsaja, kuri leidžia Windows taikomosioms programoms naudotis tinklo duomenų bazėmis.

**DB (DataBase)** – Duomenų bazė – yra duomenų bazės valdymo sistemos valdomų susijusių failų rinkinys. DB gali turėti bet kokius duomenų tipus: tekstą, garsą, vaizdą, animaciją.

**UML (Unified Modeling Language)** – modeliavimo kalba, naudojama objektiškai orientuotame projektavime.

# SANTRAUKA ANGLŲ KALBA

## Replication component of distributed databases

### *Summary*

Many organizations use relational databases (RDB) at this time. These databases can be different producers (Oracle, MS SQL Server and etc.), operate different operating systems and use different software. But the main problem if organizations want collaborate they must exchange data between them. Sometimes if they use different data structures exchange data between them is very hard or impossible. If they want integrate different databases they must use general standard. This standard must be adjustable for different databases. One of the general standards is XML.

The main purpose of this project was to analyse two functional requirements supply methods (UML and Communication Action Loops), possible architecture decisions and separate replication streams of distributed databases. One of the possible architecture decisions is replication using XML. Considering to this was created the appropriate software tool. Using this software, administrator can manipulate data and exchange data between two databases (MS SQL Server and Oracle) using XML. We have same data in two databases and we need write same date in two different databases and in this project used XML to solve this problem.

This system has three types of users – administrator, head of apartment and teacher. They have different access rights.

- Administrator - ability to write data from MS SQL Server or Oracle database to XML and write data from XML to MS SQL Server or Oracle database. He can see XML schemas and files structure.
- Head of apartment – ability to edit (insert / edit / delete) information about his individual plane and confirm apartment teachers individual planes;
- Teacher – ability to insert / edit / delete his individual plane data in the database;

The modern CASE tools “Rational Rose 2000 Enterprise Edition” and “Microsoft Visio Professional 2002” were used to design, create and analyze necessary diagrams. Programming tool Microsoft Visual Studio .NET 2003 (Visual Basic programming language) was used to create information system.

# PRIEDAI

## Priedas Nr. 1 IS apskaitos paketų integracija

### IS APSKAITOS PAKETŲ INTEGRACIJA

Vilius Lankelis, Marius Švažinskas, Tomas Zemblys

Kauno Technologijos Universitetas, Informacijos sistemų katedra

Studentų g. 50-308, LT-3031 Kaunas

(Vadovas: doc. Bronius Paradauskas)

*Pranešime yra nagrinėjami trys pagrindiniai klausimai :*

1. Mažų įmonių finansinių paketų funkcinių galimybių palyginimas;
2. Gamybos valdymo paketų funkcinės galimybės;
3. Paketų integracija per bendrus duomenis.

Pirmu klausimu apibūdinti šiuolaikiniai kompiuterizuotos apskaitos paketai, kurių pagrindą sudaro integruotos sistemos, veikiančios realaus laiko režimu. Jas naudojant, duomenys centralizuotai kaupiami ir paskirstomi tarp įvairių posistemų. Taip pat palygintos tarpusavyje atskirų finansinių paketų, pritaikytų mažoms įmonėms, funkcinės galimybės.

Antru klausimu yra nagrinėjamos gamybos valdymo paketų funkcinės galimybės, remiantis MFG\Pro paketu. Šis paketas palygintas su kitais didesniais apskaitos paketais, turinčiais gamybos modulius.

Pranešimo pabaigoje pateikiama pasirinktų pavyzdinių Visual FoxPro 6.0 ir MFG\Pro paketų integracija per bendrus duomenis. Kadangi paketų naudojamos duomenų bazės skiriasi savo duomenų struktūra ir yra nutolusios viena nuo kitos, todėl pateikiamas sprendimas kaip išspręsti duomenų perdavimą tarp šių duomenų bazių.

## 1 Įvadas

Įmonių, tiek didelių, tiek mažų, vadybininkams rinkos sąlygomis tenka nuolat analizuoti vykdomos veiklos pelningumą, ieškoti būdų gamybai ar prekybai plėsti. Ieškant teisingą sprendimą reikia išsamiai išnagrinėti daug informacijos, o pagrindinis jos šaltinis yra kompiuterizuota apskaita.

Šiuolaikinių kompiuterizuotos apskaitos paketų pagrindą sudaro integruotos sistemos, kurias naudojant duomenys centralizuotai kaupiami ir paskirstomi tarp įvairių posistemų, tokių kaip: darbo užmokesčio apskaičiavimas, įmonės skolų ir skolų įmonei apskaita, didžiosios knygos vedimas, produkcijos savikainos apskaičiavimas ir kt. Taikoma realaus laiko sistema – tai sistema, kuri automatiškai atnaujina visos sistemos duomenis, įvykus pasikeitimams vienoje iš posistemų. Turi būti numatyta galimybė valdytojui kreiptis į kompiuterį ir gauti jį dominančią informaciją tuo metu, kai jam pačiam to reikia, kitaip tariant, dirbti dialogo režimu, laiku gaunant reikiamą apskaitinę informaciją. Taigi, kompiuterizuojant apskaitą, reikia įvertinti tai, kad kompiuterinė apskaitos sistema turi būti priderinta prie egzistuojančios verslo organizavimo sistemos ir sudarytų harmoningą visumą su organizacinėmis ir valdymo struktūromis [3].

## 2 Mažų įmonių finansinių paketų funkcinių galimybių palyginimas

Mažų įmonių finansinių paketų lyginamajai analizei buvo paimti Lietuvoje naudojami kompiuterizuotos apskaitos paketai: „PRAGMA“, „KONTO“, „RODA“, „CENTAS“, „VIOLA“, „DB-Apskaita“, „Pagautė-mini“.

Su programa „PRAGMA“ įmonėje galima vesti: atsargų apskaitą, buhalterinę apskaitą, personalo, darbo užmokesčio, ilgalaikio turto apskaitas, be to, gamybos ir kuro apskaitas. „Pragmos“ atsargų apskaita nereikalauja papildomo apskaitininko darbo: ji atliekama pirkimo ir pardavimo dokumentų įvedimo metu. Įvedus į duomenų bazę pirkimo dokumento duomenis, suskaičiuojama kiekvienos prekės įsigijimo savikaina, įvertinant prekės kainą ir papildomas įsigijimo išlaidas (transportavimo, maitinimo, akcizo ir kt.), t.y. prekė pajamuojama į sandėlį. Įvedus pardavimo dokumentą, automatiškai suskaičiuojama parduotų prekių savikaina (FIFO, konkrečių kainų metodu) ir prekės nurašomos iš sandėlio [8]. Programa „MikroPragma“ skirta naudoti mažmeninės prekybos įmonių kompiuterizuotose pardavėjo darbo vietose. „MikroPragma“ galima greitai registruoti pardavimus ir spausdinti sąskaitas bei važtaraščius.

Programoje „KONTO“ galima pasirinkti kompiuterizuotos apskaitos organizavimo būdus: apskaita gali būti vedama kaupimo arba pinigų principu; aprašomi apskaitos registrai bei apskaitos žurnalai, nustatomi jų tarpusavio ryšiai; atsargų savikainos apskaitai gali būti taikomas FIFO arba LIFO metodai; pateikiama išsami kliento atsiskaitymo kortelė bei lengvai generuojamos skirtingos pirminių dokumentų spausdinimo formos [6]. MiniBonus - tai specialus pasiūlymas „KONTO“ naudotojams. Ši žmoniškųjų resursų valdymo sistema skirta nedidelėms, bet įvairios veiklos įmonėms. Tai visa būtina ir naudinga informacija, priemonės ir metodai įmonės personalui valdyti, planuoti, analizuoti. Remiantis tarptautiniais standartais, Lietuvoje sukurta sistema BONUS yra universali darbo laiko apskaitos, atlyginimo skaičiavimo ir personalo valdymo priemonė.



“**RODA**” - pilnas buhalterinės apskaitos iki balanso programų paketas bet kokio profilio įmonei. Šis programinis paketas susideda iš 4 modulių: Atsiskaitymai su tiekėjais / pirkėjais ir atsargų (prekių ir medžiagų) apskaita; turto nusidėvėjimo ir likutinės vertės skaičiavimas; atlyginimai ir socialinis draudimas; buhalterija [9]. “RODA” leidžia taisyti uždarytų praeitų apskaitos periodų duomenis ir iš naujo uždaryti periodus. Šios galimybės neturi daugelis kitų programų. RODA leidžia vesti NERIBOTO ĮMONIŲ SKAIČIAUS apskaitą. Visas spausdinamas ataskaitas galima eksportuoti į EXCEL bylas tolimesniam apdorojimui ar ryšiui su kitomis programomis.

Materialinių vertybių ir buhalterinės apskaitos programoje “**CENTAS**” yra pirkimai - pardavimai., atsiskaitymų apskaita; Sandėlių apskaita, neribotas sandėlių skaičius, vidiniai judėjimai; Paslaugos, suteiktų paslaugų registravimas; Sąskaitų planas, operacijų žurnalas, korespondencijų ataskaitos, Didžioji knyga, balansas; Darbo užmokesčio apskaita; Ilgalaikio turto apskaita; Kasos ir banko apskaita ir dar daug kitų sprendžiamų klausimų [4].

**DB-Apskaita** yra įmonės veiklos dokumentų registravimo ir ataskaitų formavimo sistema. Joje galima įvesti ir saugoti biudžetinių, valstybinių, akcinių bendrovių bei individualių įmonių finansinės ūkinės veiklos duomenis, juos apdoroti, sisteminti pagal finansų apskaitoje priimtas taisykles [5].

DB-Apskaita sudaryta iš atskirų tarpusavyje suderintų modulių. Šie moduliai integruoti į vieną bendrą komplektą ir leidžia gauti apibendrintus rezultatus. Bet kuriuo metu Jūs galite matyti įmonės finansinę būklę, skolas, išsiskolinimus pagal įvestus pirminius buhalterinius dokumentus. Visus būtinus pirminius dokumentus ir ataskaitas galima peržiūrėti, spausdinti arba eksportuoti į MS WORD, MS EXCEL ir kt. Materialinių vertybių apskaita gali būti vedama kiekvienam apskaitos objektui atskiromis svartinėmis, fiksuotomis kainomis, FIFO, LIFO metodais. Įvedant pirminius dokumentus, dvejybinius įrašus galima paskirstyti į pajamų bei sąnaudų sąskaitas pagal padalinius ar objektus. Įvedus duomenis, pasitikrinimui galima daryti ataskaitas bet kuriam mėnesiui ar dienai.

Programa “**VIOLA**” skirta įmonės integruotai buhalterinei apskaitai. Darbas su programa maksimaliai supaprastintas – užtenka įvesti operaciją į žurnalą, o visą likusį darbą padaro programa. Bet kurią jau įregistruotą klaidingą operaciją galima anuliuoti ir ištaisyti [10]. Programa veda kiekinę-suminę prekių, žaliavų, inventoriaus apyvartos bei likučių apskaitą. Vartotojas savo įmonės atsargas gali sugrupuoti pagal tris nepriklausomus požymius: grupė, klasifikavimas I, klasifikavimas II. Atsargų likučiai ir jų judėjimas gali būti vedami kiekvienam įmonės padaliniui atskirai. Operacijos su atsargomis registruojamos atitinkamuose pirkimų, pardavimų, perkėlimų ir nurašymo žurnaluose. Atsargų apyvartos, pardavimo, likučių ataskaitos gali būti filtruojamos bei grupuojamos pagal padalinius, atsargų grupes, verslo vienetus, darbo projektus.

“**PAGAUTĖ-mini**” apskaitos sistema mažoms ir vidutinėms įmonėms. Programų paketą sudaro atskiri moduliai: “Gamyba” - žaliavų sandėliavimas ir gamybos apskaita; “Balansas”- ūkinė operacijų apskaita; “Sandėlis” - atsargų (prekių) apskaita; “Atlyginimas” - darbo užmokesčio skaičiavimas; “Turtas” - ilgalaikio turto nusidėvėjimo paskaičiavimas ir kt. Kiekvienas šių modulių gali dirbti visiškai atskirai arba gali būti komplektuojamas su kitais moduliais pagal vartotojo poreikius [7].

Atliekant šių programinių paketų lyginamąją analizę buvo pastebėti kai kurie trūkumai ar skirtumai:

- Ne visų programinių paketų darbe numatytas darbinės DB išskėlimas į archyvinę DB (Konto, DB-Apskaita, Pagautė mini).
- “Violoje” nėra numatyta kompiuterizuota darbo užmokesčio apskaita.
- Duomenų atstatymas numatytas tik “Pragmoje”, “Cente” ir “Pagautė mini”.
- Gaminių pajamavimas į sandėlį vykdomas irgi ne visose sistemose.
- Greitas atsargų likučių nustatymas nevykdomas Rodos programoje.
- Nuolaidų taikymas nevykdomas “Cento” ir DB-Apskaitos programose.
- Pirkėjų išankstiniai apmokėjimai už prekes nerodomi “Rodoje”, “Violoje” ir “Pagautėje mini”.
- Pirkėjų užsakytų prekių rezervavimas nefiksuojamas “Rodoje”, DB-Apskaitoje ir “Pagautėje mini”.
- Prekių gražinimų apskaita vedama tik “Pragmoje”, “Rodoje” ir Pagautėje mini.
- Sandėlio apyvarta neskaičiuojama “Rodoje”.
- Savikainos nustatymas nevykdomas “Rodoje” ir DB-Apskaitoje.

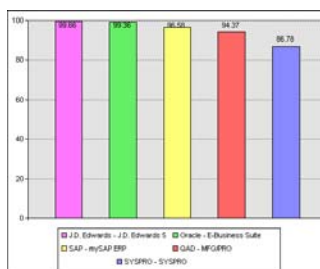
Be anksčiau paminėtų apskaitos programinių paketų, gan plačiai naudojamas ir korporacijos QAD, Inc, vienos iš pirmaujančių pasaulinio lygio sprendimų tiekėjų gamybos, planavimo ir valdymo srityje, produktas **MFG/PRO**. Tai integruota, automatizuota gamybinės - ūkinės veiklos valdymo informacinė sistema, kurioje yra pilnai integruotas modulinis ERP (įmonės išteklių planavimo) programinis paketas. Šio paketo funkcines galimybes apžvelgsime kitame skyriuje.

### 3 Gamybos valdymo paketų funkcinės galimybės

Kompiuterinė verslo valdymo sistema (VVS) yra neatsiejama šiuolaikinių įmonių valdymo priemonė. Sėkmingai pasirinkta ir įdiegta VVS gali pagreitinti verslo procesus ir suteikti pranašumą prieš konkurentus, ir priešingai – sistema, neatitinkanti įmonės poreikių, gali stabdyti jos veiklą ir net privesti iki bankroto. Šiuo metu didžioji dalis Lietuvos įmonių naudoja vietinių programuotojų sukurtas programas, skirtas daugiausia nedidelių įmonių apskaitai kompiuterizuoti [2]. Bendros su užsieniu įmonės dažniausiai pasirenka verslo partnerių rekomenduojamas sistemas tokias kaip: *MFG/PRO*, *J. D. Edwards*, *SYSPRO*, *SAP*, *Oracle* ar kitas sistemas. Stengdamiesi įsitvirtinti naujose rinkose, didžiųjų VVS pardavėjai dažnai taiko įvairias nuolaidas. Lietuvoje tokios nuolaidos vidutiniškai siekia 20-40%. Taigi ieškantiems naujos sistemos yra iš ko rinktis, tereikia tik nuspręsti, kuri sistema geriausiai atitinka įmonės poreikius.

Pasirenkant paketą reikia įvertinti kainą ir atsižvelgti ne vien į tai, kiek daug įvairių funkcijų turi programa, bet ir į tai, kiek užsakovui tų savybių reikės. Kaina ir funkcinės galimybės - tai ne paskutiniai faktoriai lemiantys pasirinkimą. Užsakovas turi sugebėti formuluoti funkcinis reikalavimus, suprasti paketų funkcinės galimybes, mokėti įvertinti jų atitikimą keliamus reikalavimus, įvertinti savo specialistų, kurie dirbs su tuo paketu, gebėjimus. Čia užsakovui gali tekti samdyti ekspertus, kurie sugeba įvertinti paketo funkcinės galimybes ir nustatyti nefunkcinius reikalavimus.

Šiame pranešimo medžiagos skyrelyje palygintos žinomų didelių paketų, turinčių gamybos modulius, funkcinės galimybės. 1 paveikslėlyje pateikta diagrama [15], kuri iliustruoja gamybos valdymo procesų kompiuterizavimo laipsnį.



1 pav. Gamybos valdymo procesų kompiuterizavimo laipsnis

Turimos ir prieinamos informacijos ribose įvairių gamybos valdymo paketų funkciniai uždaviniai buvo sugrupuoti ir palyginimo tikslu patalpinti 1 lentelėje. Išskirtos uždavinių grupės orientuotos į maisto pramonės įmones.

1 lentelė. Maisto pramonės gamybos valdymo uždavinių grupavimas ir grupių palyginimas

	SAP	J.D Edwards	Oracle	MFG	SYSPRO
Automatinis komponentų performulavimas remiantis rinkinio bendra kaina	Nepalaikoma	Modifikuojama (Kuriama)	Planuojama įgyvendinti	Pritaikyta individualiam vartotojui	Nepalaikoma
Galiojimo datos suskaičiavimas ir suteikimas pagaminti produktų partijai	Nepalaikoma	Palaikoma	Palaikoma	Palaikoma	Modifikuojama (Kuriama)
Draudimas vartoti produkto sudedamąsias dalis, kurių galiojimo laikas yra pasibaigęs	Nepalaikoma	Palaikoma	Palaikoma	Palaikoma	Modifikuojama (Kuriama)
Medžiagų klasifikavimas norint išvengti brokuotų medžiagų panaudojimo ar pardavimo	Palaikoma	Palaikoma	Palaikoma	Palaikoma	Nepalaikoma
Kaštų apskaičiavimas gaminimui naudojant ir grįžtamas atliekas	Palaikoma	Pritaikyta individualiam vartotojui	Palaikoma	Palaikoma	Nepalaikoma
Nurodymas maks. ir min. procentas atliekų, kurias galima perdirbti	Palaikoma	Modifikuojama (Kuriama)	Planuojama įgyvendinti	Modifikuojama (Kuriama)	Nepalaikoma
Gaminio struktūros ataskaitos gavimas įvedus komponentą ar gaminį	Palaikoma	Palaikoma	Palaikoma	Palaikoma	Palaikoma

Kaip matyti iš diagramų, paketai savo gamybos procesų kompiuterizavimo lygiu yra gana panašūs. Kadangi MFG\Pro produktas pagal savo kainą ir kokybės santykį yra priimtinausias, bei vis daugiau populiarumo sulaukia rytų Europos šalyse, pateiksime trumpą QAD organizacijos ir jos produkto charakteristiką. QAD produktu MFG\Pro naudojasi automobilių, maisto, gėrimų, elektronikos, medicinos ir pramoninių prekių gamintojų atstovai. Jau dabar šis produktas yra naudojamas daugiau kaip 80 šalių ir

parduota apie 5400 įmonių licenzijų. MFG/Pro paketų yra sukurta net 26 kalbomis, tarp kurių, yra ir lietuvių kalba [11].

Atlikus analizę, kaip pagrindinį MFG/Pro sistemos privalumą, lyginant su kitomis kompiuterizuotomis apskaitos ir valdymo programomis, reikia įvardinti jos vientisumą ir galimybę valdyti atsargas, pradedant jų poreikio planavimu, pereinant gamybos stadiją ir baigiant pagaminto produkto išleidimu pirkėjui. MFG/PRO teikia galimybę fiksuoti atsargų pirkimą, gauti detalią informaciją apie jų sandėliavimą, išsigijimo ir galiojimo datas, suteikti atsargoms įvairias būsenas, tuo ribojant jų judėjimą įmonės viduje (pvz. apsaugant išbrokuotą partiją nuo galimo pardavimo ir pan.), sekti produkcijos gamybą, valdant receptūras ir technologines instrukcijas, ir analizuoti medžiagų sunaudojimo skirtumus, prognozuoti gaminio savikainą, valdyti informaciją apie pagamintą produkciją, jos realizacijos terminus, registruoti toje pačioje apskaitos sistemoje pirkėjų užsakymus ir atlikti produkcijos paskirstymą kiekvienam klientui pagal jo pageidaujamas datas ir kiekius, išrašyti sąskaitas už atkrovimus ir registruoti pinigines įplaukas [21]. Taigi MFG/PRO apima visas pagrindines įmonės veiklos sritis ir suteikia galimybę priiminėti valdymo sprendimus remiantis vienoje apskaitos sistemoje sukaupta informacija. Tačiau MFG/PRO paketas, kaip ir daugelis kitų, neapima specifinių, kiekvienai individualiai įmonei būdingų funkcijų, pavyzdžiui, alternatyvių nelinejinių technologinių maršrutų optimizavimo, vienietinio darbo užmokesčio skaičiavimo uždavinių. Be to, keičiantis verslo įstatyminei bazei ir vystant kooperacinius ryšius programinius modulius tenka pakeisti ir sukurti naujus arba spręsti pagrindinių ir specialiosios paskirties paketų integracijos uždavinius.

SAP paketo kūrėjai turbūt yra pagrindiniai QAD kampanijos konkurentai. Šioje programoje taip pat yra finansų, gamybos, marketingo, darbo užmokesčio apskaitos moduliai. Taip pat yra galimybė fiksuoti atsargų pirkimą, gauti informaciją apie jų sandėliavimą. Tačiau šio paketo kaina dažnai nėra nurodoma, kadangi ji labai svyruoja priklausomai nuo kiekvieno kliento poreikių, bei nuo to kaip sunku ar lengva SAP paketo savybes pritaikyti organizacijos tikslams. Šį pritaikymą dažnai atlieka programą parduodančios įmonės, dėl ko pirkėjui vėliau gali kilti problemų, keičiant reikalavimus. Dėl to paketas praktiškai yra skirtas didelėms arba bent jau vidutinio dydžio organizacijoms [11, 13].

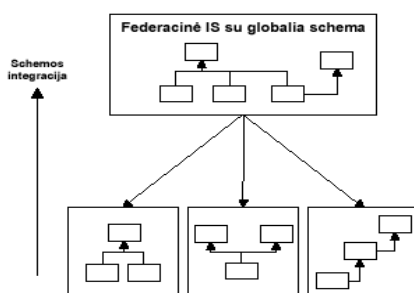
SYSPRO verslo valdymo sistema yra išsigijusios daugiau kaip 6000 įvairiausių kampanijų pasaulyje, daugiau kaip 50 šalių. Programa turi daugiau kaip 40 modulių, pradedant nuo apskaitos, pirkimų, pardavimų, atsargų valdymo, planavimo ir kt. SYSPRO leidžia vartotojui klasifikuoti pagamintą produkciją, peržiūrėti turimus likučius sistemoje, apskaičiuoti gaminio savikainą, valdyti receptūras. Yra ir dar daugiau panašių savybių, kurios kartojasi visose gamybos modulius turinčiose sistemose [11, 12].

MFG/Pro paketas turi daug funkcinių galimybių, tačiau daugelis informacinių sistemų naudoja skirtingas duomenų bazes, todėl norint apsieikti informacija tarp IS tenka atlikti naudojamų paketų integraciją. Todėl ketvirtame skyriuje apžvelgsime kaip galima atlikti paketų integraciją per bendrus duomenis ir pateiksime mūsų pasirinktų pavyzdinių FoxPro ir MFG/Pro paketų integravimo per bendrus duomenis pavyzdį.

#### 4 Paketų integracija per bendrus duomenis

Daugelis informacinių sistemų naudoja skirtingas duomenų bazes ir dažniausiai tos duomenų bazės yra heterogeninės (saugo skirtingų tipų duomenis, duomenis pateikia skirtingai, naudoja skirtingą programinę įrangą valdymui, patalpina skirtingose kompiuterinėse platformose), todėl tokioms duomenų bazėms reikalingas vieningas priėjimas prie duomenų, išsaugant duomenų bazių struktūrą. Atliekant skirtingų paketų integraciją per bendrus duomenis galime remtis šiuo požiūriu, laikydami kad šios duomenų bazės yra heterogeninės. Egzistuoja du heterogeninių duomenų bazių integravimo būdai: globali schema ir federacinė schema. Globalios schemas atveju kiekviena schema, atitinkanti lokalių duomenų bazę yra prijungiama prie vienintelės bendros integruotos schemas. Federacinės schemas atveju kiekviena lokali duomenų bazė pateikia tą dalį savo schemas, kurią ji nori dalintis su kitomis duomenų bazėmis [1].

Duomenų schemų integracija yra šių abiejų integravimo būdų pagrindinė ir neatsiejama dalis. Tai procesas, kurio metu gaunama viena ar daugiau schemų iš egzistuojančių duomenų bazių schemų. Šios schemas aprašo integruojamų duomenų bazių semantiką ir naudojamos kaip pradiniai duomenys integravimo procese.



2 pav. Federacinės IS su globalia schema schemas integracija

Federacinė informacijos sistema yra sistema, integruojanti palikuoninius duomenų šaltinius ir taikomąsias programas. Federacinės duomenų bazės yra vienas iš federacinių informacijos sistemų tipų. Pagrindiniai kriterijai, išskiriantys federacines duomenų bazines iš kitų federacinių informacijos sistemų yra globalios schemos egzistavimas federacijos lygyje ir duomenų šaltinio lygio sudėtyje tik duomenų bazės. Federacinėse duomenų bazėse egzistuoja visų tipų heterogeniškumas (sintaksinis, duomenų modelio ir loginis), išskyrus užklausų apribojimų skirtumus, vykdymo autonomiškumas, duomenų šaltinių vietas, schemų ir dalinai kalbų užslėpimas nuo vartotojo, stipri federacija, kolekcijos ir lydinio tipo semantinė integracija, virtuali ar dalinai virtuali integracija.

Kiekviena duomenų bazė turi savo lokalią schemą. Šios lokalios schemos yra transformuojamos į kanoninį duomenų modelį, kuris yra pasirinktas kaip bendras federacijos duomenų modelis [22]. Transformacijos rezultatas yra schema. Dažniausiai vartotojui yra reikalinga tik dalis duomenų arba duomenų bazė, kuri gali teikti tik tam tikrus duomenis, todėl eksporto schema yra šių „derybų“ rezultatas, komponentinės schemos projekcija, skirta federacijos lygiui [20]. Federacinės schemos apima vieną ar kelias eksporto schemas, kiekviena federacinė schema yra skirta tam tikrai vartotojų klasei. Išorinės schemos pagalba yra atrenkama informacija, skirta konkrečioms vartotojams, nedidelėms jų grupėms ar taikomosioms programoms. Vartotojų naudojamas duomenų modelis gali skirtis nuo kanoninio federacinės duomenų bazės duomenų modelio, tokiu atveju atliekamos atitinkamos transformacijos.

Kai vartotojo duomenų modelis skiriasi nuo kanoninio duomenų modelio, federacijos komponentai yra federacinės duomenų bazės, egzistuoja kelios semantikos federacinėje schemoje, yra naudojama išplėsta federacinių duomenų bazių architektūra [18]. Joje yra trys papildomos schemos – derybų schema, vartotojo išorinė schema ir taikomųjų atvejų schema. Derybų schema, sudaroma iš lokalios schemos, leidžia kaip komponentą naudoti kitą federaciją, naudojant tik dalį jos duomenų resursų. Vartotojo išorinė schema yra išorinė schema, transformuota į vartotojo duomenų modelį, besiskiriantį nuo kanoninio duomenų modelio. Taikomųjų atvejų schema apima vieną federacinės schemos semantiką, skirtą vienai ar kelioms vartotojų grupėms [22].

Atliekant paketų integraciją sistemoms reikia užsiduoti funkcinis reikalavimus, todėl vienas iš būdų tą padaryti, tai funkcinis reikalavimus atvaizduoti komunikacinėmis kilpomis [19].

Tačiau pagrindinė problema, kuri atsiranda tarp skirtingas duomenų struktūras naudojančių duomenų bazių ar programinių paketų, kurie naudojami šiomis duomenų bazėmis yra duomenų perdavimas. Perduodant duomenis iš vienos sistemos į kitą kyla įvairių problemų, todėl problemų sprendimas yra universalus duomenų struktūrų aprašymas. Kadangi paketai paprastai programuojami skirtingose kalbose ir operacinėse sistemose, tai universalus būdas duomenims apsaugoti yra XML technologijų panaudojimas. XML naudojimas nepaprastai sparčiai plinta ir vis dažniau ši kalba naudojama kaip tarpinis kodas, kuris yra labiausiai išvystytas standartas universaliam duomenų aprašymui. Todėl toliau plačiau apžvelgsime XML technologijos panaudojimą perduodant duomenis tarp skirtingas duomenų struktūras naudojančių sistemų.

XML (Extensible Markup Language) buvo sukurta 1996 metais. Ją sukūrė XML darbo grupė, padedama W3C (World Wide Web Consortium) konsorciūmo. Tačiau tai nėra visiškai nauja technologija – ji pagrįsta SGML (*Standard Generalized Markup language*), kuri tapo ISO standartu 1986 metais [16]. SGML plačiai naudojama dideliems dokumentacijos archyvams aprašyti, saugoti, publikuoti. XML projektuotojai daugelį idėjų perėmė iš SGML. Naudoti SGML informacijai per internetą perduoti – neefektyvu ir nenaudinga, nes ji nėra pritaikyta internetui. Buvo nuspręsta SGML susiaurinti ir padaryti labiau tinkamą internetui. Naujasis standartas pavadintas *eXtensible Markup Language* – Praplečiama Žymėjimo Kalba, arba tiesiog – XML.

XML yra metakalba, naudojama apibrėžti specifiniams taikymams skirtas žymėjimo (*markup*) kalbas. XML – tai rinkinys taisyklių, nurodymų ir susitarimų apie tai, kaip pateikti duomenis tekstiniame faile struktūrizuotu pavidalu. Tai, kad duomenys pateikiami tekstiniu, o ne binariniu pavidalu, leidžia programuotojui ir net galiniam vartotojui pažvelgti į XML dokumento turinį nenaudojant programos, kuria tas dokumentas buvo sukurtas. Vis tik dažniausiai XML duomenų vartotojas ir naudotojas yra kompiuterinė programa. HTML standartas apibrėžia ką reiškia kiekviena žymė bei atributas, ir kaip juos atvaizduoti naršyklės lange, kai tuo tarpu XML naudojamas tik informacijai sužymėti. Duomenų interpretaciją ir atvaizdavimo būdus galima pasirinkti laisvai. XML dokumentai gali būti naudojami pačiose įvairiausiose srityse, tokiose kaip e-komercija, komunikacijai tarp verslo partnerių ar organizacijos viduje.

Pagrindinė XML paskirtis ir privalumai:

- XML skirta keitimuisi duomenimis;
- XML skirta keistis finansine informacija, t.y. yra naudojama daugumoje B2B programų;
- XML gali būti naudojama duomenų paskirstymui;
- XML gali būti skirta duomenų saugojimui;
- XML gali padaryti duomenis labiau prieinamus [17].

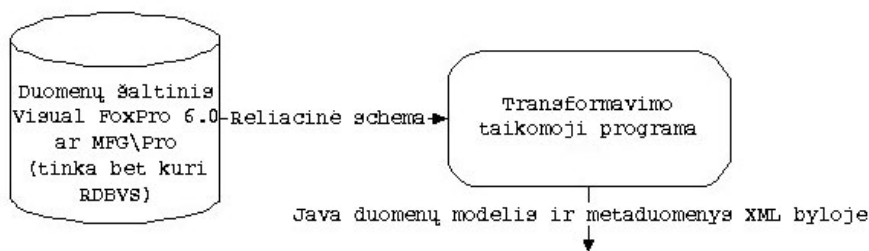
Remiantis aukščiau išvardintais XML požymiais galima daryti išvadą, jog XML kalba labiausiai tinka duomenų keitimuisi tarp skirtingas duomenų struktūras naudojančių sistemų. Naudojant XML kalbą galima rinktis ir technologiją, pvz.: JDO (Sun Microsystems) ar ADO.NET (Microsoft) [14].

**2 lentelė. ADO.NET (Microsoft) ir JDO (Sun Microsystems) technologijų lyginamoji analizė**

	ADO.NET	JDO
Duomenų modelis	Reliacinis	Objektinis
Duomenų apibrėžimo kalba ( <i>Data Definition Language</i> )	XML	Java ir XML
Užklausų kalba	SQL	JDOQL
Duomenų manipuliavimo kalba	SQL ir (C++ arba Visual Basic)	Java
Paveldėjimo ir polimorfizmo realizacija	Ne	Taip
Unikalus identifikatorius	Pirminis raktas	Pirminis raktas arba duomenų šaltinio identifikatorius
Transakcijos	Taip	Taip
Konkurentiškumas	Taip	Taip
Objektinio duomenų modelio palaikymas užklausose	Ne	Taip

ADO.NET technologijoje viskas yra saugoma vienoje XML byloje, kurią apdoroja speciali taikomoji programa. JDO technologijoje yra du šaltiniai, kadangi kuriant Java klases sąsajai su duomenų šaltiniu naudojamas Java išeities tekstas ir XML byloje užrašyti metaduomenys [23].

Iš mūsų pasirinktų pavyzdinių paketų (Visual FoxPro 6.0 ir MFG\Pro) XML bylų sudarymą naudojant JDO technologiją galėtume atvaizduoti taip:



**3 pav. Reliacinės schemas transformavimas į JAVA duomenų modelį ir XML bylą**

Kaip matome Visual FoxPro 6.0 ir MFG\Pro DBVS transformavus jų reliacines schemas rezultatas yra Java klasės ir metaduomenys, saugomi XML formatu. Mūsų pasirinkti pavyzdiniai paketai (Visual FoxPro 6.0 ir MFG\Pro) naudoja skirtingas duomenų struktūras ir iš jų duomenų bazių galima suformuoti XML bylas, tai duomenų perdavimas tarp šių paketų galimas būtent per XML bylas, kuriose ir yra aprašyta universali duomenų struktūra. XML bylų pagalba ir atliekama šių paketų integracija. Taigi šiuo metu vienas iš pagrindinių būdų atliekant duomenų bazių ar paketų integraciją, kurie naudojami šiomis duomenų bazėmis, yra naudoti XML, kadangi XML labiausiai išvystytas standartas universaliam duomenų aprašymui ir labiausiai atitinka sistemos duomenų bazių duomenis.

## 5 Apibendrinimas

IS programinių paketų vientisam panaudojimui buvo išnagrinėtos dvi – finansinės apskaitos ir serijinės gamybos valdymo uždavinių klasės. Palygintos įvairių paketų, suprojektuotų naudojant skirtingas platformas, funkcinės galimybės ir parodyta, kaip pagrindinio gamybos valdymo MFG/Pro modulio funkcinės galimybės galima praplėsti specializuotų, mažoms įmonėms skirtų, finansinės apskaitos FoxPro paketų funkcinėmis galimybėmis. Paketų vientisam funkcionalumui užtikrinti buvo pasiūlyta naudoti XML, kuris yra labiausiai išvystytas standartas universaliam duomenų aprašymui ir labiausiai atitinka sistemos duomenų bazių duomenis, nes daugelis šiuolaikinių IS naudoja skirtingas duomenų bazes ir norint perduoti duomenis tarp tokių IS reikia atlikti duomenų bazių integraciją.

## Literatūros sąrašas

- [1] A.Elmagarmid, M.Rusinkiewicz, A.Sheth. Management of heterogeneous and autonomous database systems, 1999.

- [2] **Č.Ratkevičius.** Devynis kartus pagalvok, dešimtą... pirk. Kokią kompiuterinę verslo valdymo sistemą įsigyti. *Vadovo pasaulis*, 2003.
- [3] **G.Černius, G.Kalčinskas.** Finansinė ir valdymo apskaita. *Aušra*, 1999.
- [4] Interneto prieiga: "Centas". <http://www.edlonta.lt/centas.htm> 2004-03-18.
- [5] Interneto prieiga: "DB Apskaita". <http://www.dbtopas.lt/apskaita/> 2004-03-15.
- [6] Interneto prieiga: "KONTO" <http://www.labbis.lt/konto> 2004-03-15.
- [7] Interneto prieiga: "Pagautė-mini" <http://www.infotema.lt/index1.html> 2004-03-15.
- [8] Interneto prieiga: "Pragma". <http://www.proringas.lt/moduliai.html> 2004-03-15.
- [9] Interneto prieiga: "Roda". <http://www.taide.lt/~akf/apskaita.html> 2004-03-15.
- [10] Interneto prieiga: "Viola". <http://www.is.lt/delta> 2004-03-15.
- [11] Interneto prieiga: [http://www.homercomputer.com.au/homer\\_software\\_guide/PP/](http://www.homercomputer.com.au/homer_software_guide/PP/) 2004-02-10.
- [12] Interneto prieiga: <http://www.manufacturingtalk.com/news/ifo/> 2004-02-10.
- [13] Interneto prieiga: <http://www.sap.com/> 2004-02-10.
- [14] Interneto prieiga: [http://www.service-architecture.com/database/articles/comparison\\_of\\_dbms\\_standards.html](http://www.service-architecture.com/database/articles/comparison_of_dbms_standards.html) 2004-03-17.
- [15] Interneto prieiga: <http://www.technologyevaluation.com> 2004-01-10.
- [16] Interneto prieiga: <http://www.w3.org/TR/2004/REC-xml-20040204/#ISO8879> 2004-03-14.
- [17] Interneto prieiga: <http://www.w3.org/TR/REC-xml> 2004-03-14.
- [18] **J.Samos, F.Saltor, J.Sistac, A.Bardes.** Database architecture for data warehousing: an evolutionary approach, <http://citeseer.ist.psu.edu/samos98database.html> 2004-03-14.
- [19] **L.Nemuraitė, B.Paradauskas, L.Selelionis.** Extended communicative action loop for integration of new functional requirements. *Informacinės technologijos ir valdymas, Technologija*, 2002, Nr. 2 (23), p. 20-23.
- [20] **M.Strassler, M.Schonhoff.** Integrating engineering databases: how does the application domain influence the FDBMS architecture, <http://citeseer.ist.psu.edu/177061.html> 2004-03-14.
- [21] Qad inc. "MFG/Pro User guide introduction" Volume 1.
- [22] **V.Kontrimas.** Federacinės duomenų bazės. *Informacinės technologijos 2004: konferencijos pranešimų medžiaga, Technologija*, 2004, p. 465, 468-469.
- [23] **V.Kontrimas.** Java sąsaja su reliacinėmis duomenų bazėmis. *Informacinės technologijos 2003: konferencijos pranešimų medžiaga, Technologija*, 2003, p. III-31 - III-34.

## IS accounting packages integration

In this report are three matters under investigation:

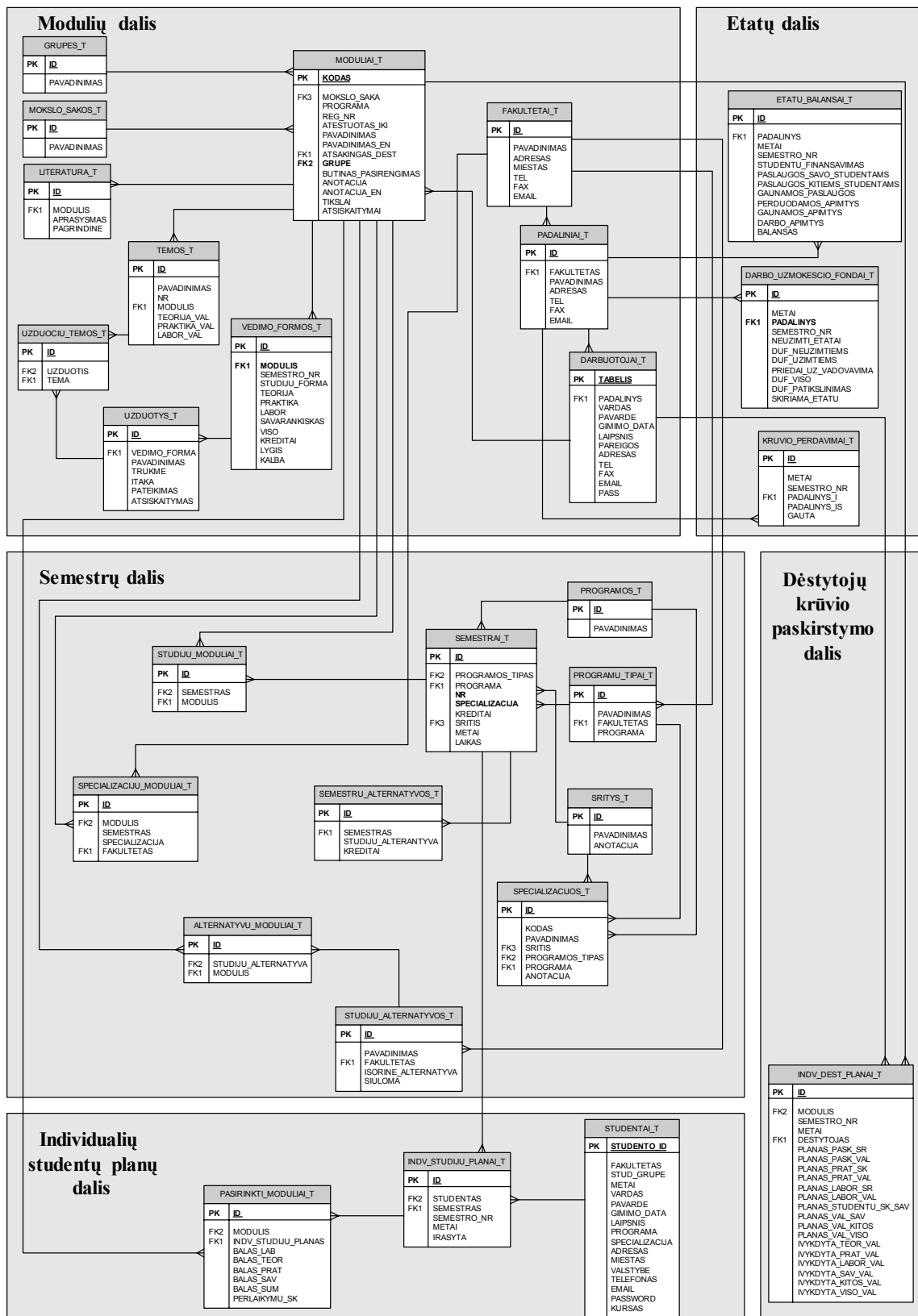
1. The functional possibilities comparison of small companies financial packages;
2. The functional possibilities of production possession packages;
3. The packages integration through general data.

Modern computerized accounting packages, which based on integrated systems working on the real-time mode, are defined on the first part of article. Using them, data are centralized, collecting and distributing between various subsystems. Also here is compared the functionality of separate financial systems, which are adjusted to small companies.

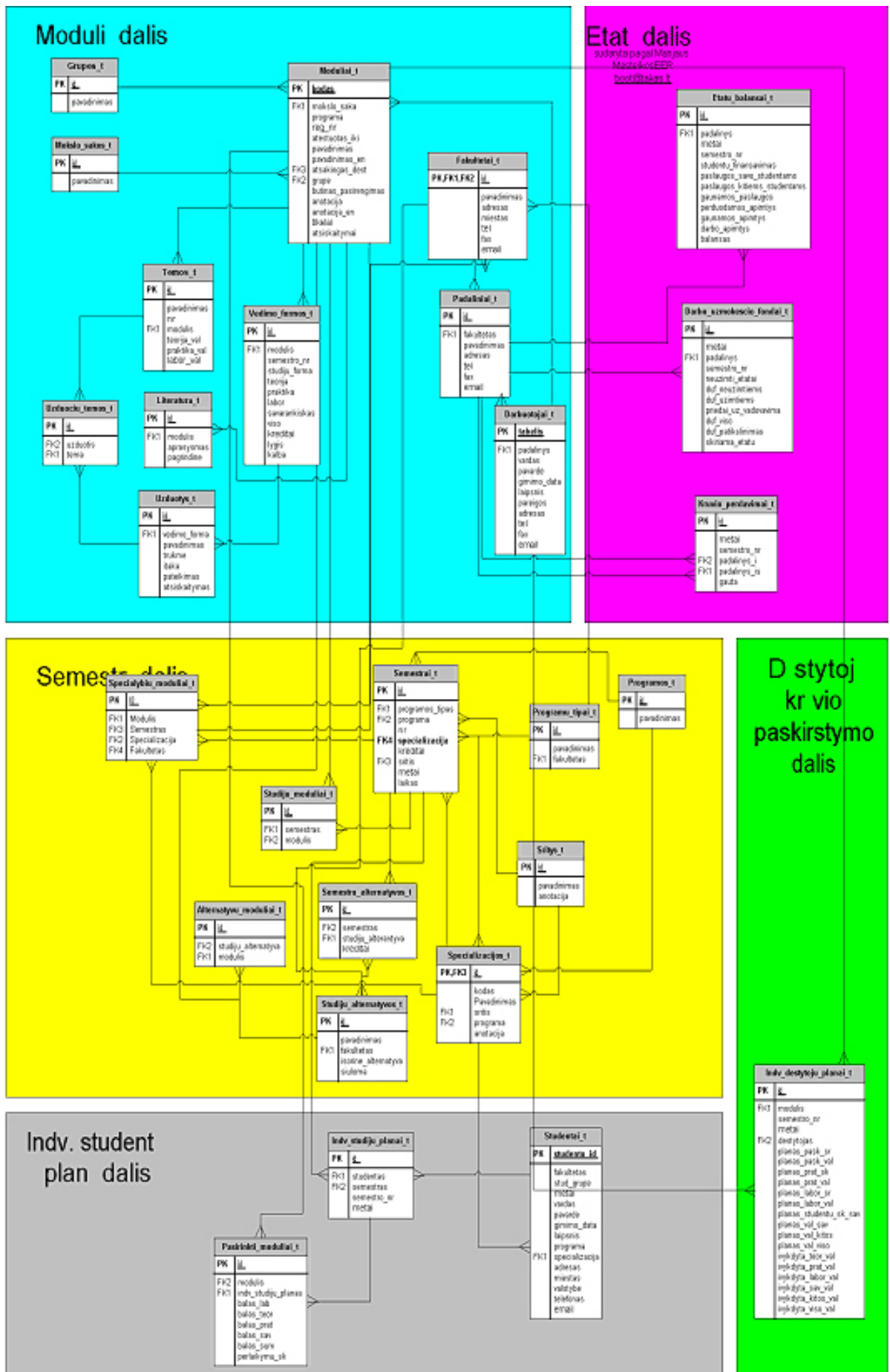
Functionality of the systems, which has manufacturing modules (sustaining MFG\Pro system) is analyzing on the next chapter. This system is compared with other big financial systems, which have manufacturing modules.

Packages, which were programmed with Visual FoxPro 6.0 and MFG\Pro package, integration through general data are represented at the end of article. Packages are using databases, which are different data structure and are on a long distance, consequently is representing solution, how to resolve data transmission between following databases.

## Priedas Nr. 2 Sistemos naudojamos duomenų bazės



37 pav. MS SQL Server duomenų bazės modelis



38 pav. Oracle duomenų bazės modelis



## Priedas Nr. 3 Uždavinio techninis išpildymas

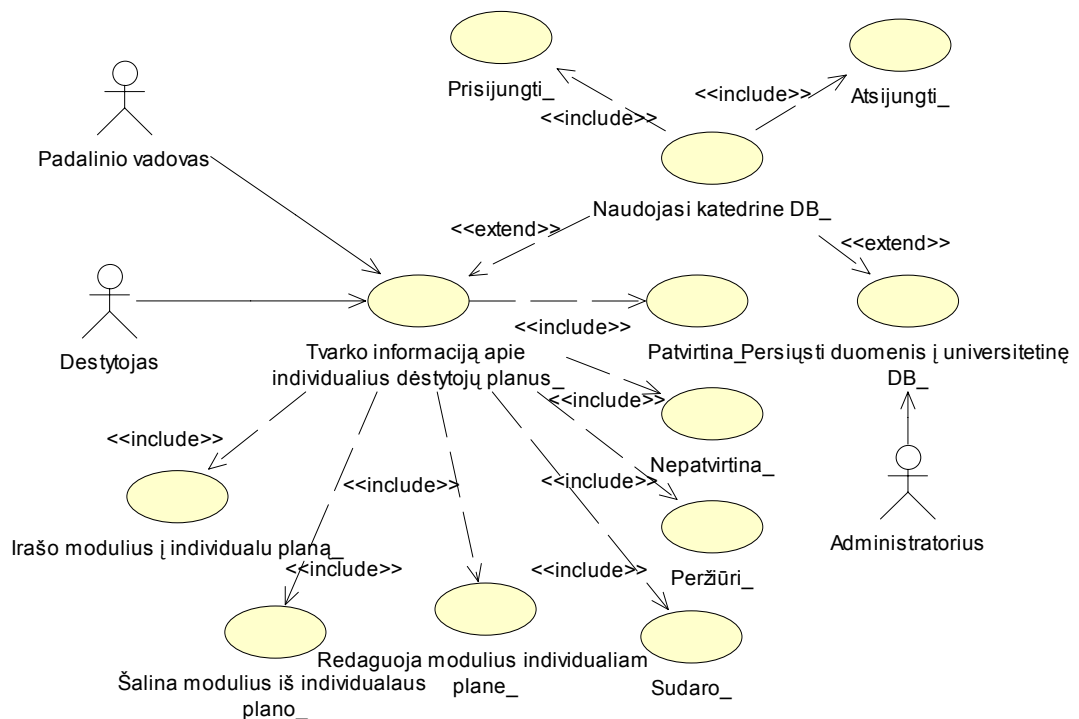
### 1. Kuriamos sistemos kokybės kriterijus

Pagrindinis kuriamos sistemos kokybės kriterijus – veikianti programinė įranga. Pagrindinė sukurtos programinės įrangos funkcija bus visos duomenų bazės ar jos dalies nuskaitymas iš duomenų bazės, esančios MS SQL serveryje, ir konvertavimas į XML kalbą, duomenų išsaugotų XML faile surašymas į Oracle duomenų bazę, bei pateikimas duomenų bazėje esančių duomenų vartotojui naudojant sukurtą programinę įrangą.

Sukurta programinė įranga turi būti patikrinta ar atitinka jai keliamus tokius reikalavimus:

- ar atitinka jai keliamus funkcinis reikalavimus;
- ar atitinka reikalavimus vartotojo sąsajai;
- ar yra atliekama klaidų kontrolė;
- ar yra integruojama ir kitose aplinkose.

### 2. Visų sistemos vartotojų panaudojimo atvejų diagrama



39 pav. Sistemos vartotojų panaudojimo atvejų diagrama

Iš sistemos vartotojų panaudojimo atvejų diagramos matyti, kad ir padalinio vadovas ir dėstytojas gali tvarkyti informaciją apie individualius dėstytojų planus. Padalinio vadovas prisijungęs prie sistemos gali peržiūrėti dėstytojų individualius planus ir juos patvirtinti arba nepatvirtinti. Dėstytojas prisijungęs prie sistemos gali sudaryti savo individualų planą, peržiūrėti jį ir jei padalinio vadovas nepatvirtina individualus plano gali jį redaguoti įrašydamas naujus modulius arba šalindamas juos iš savo individualus plano. Sistemos administratorius prisijungęs prie sistemos gali suformuoti XML failą iš padalinio duomenų bazės (MS SQL Server DB), peržiūrėti jį ir persiųsti šiame XML faile esančius duomenis į universitetinę DB (Oracle DB) (apima panaudojimo atvejį Persiųsti duomenis į universitetinę DB). Visi sistemos vartotojai naudojami padalinio duomenų baze prisijungdami ir atsijungdami nuo jos.

### 3. Specifikacijos kiekvienam panaudojimo atvejui

Lentelė Nr. 4

*Panaudojimo atvejis - Tvarko informaciją apie individualius dėstytojų planus -> Sudaro\_*

<b>Panaudojimo atvejis</b>	Tvarko informaciją apie individualius dėstytojų planus -> Sudaro_
<b>Aktorius</b>	Dėstytojas
<b>Sistema</b>	Individualaus dėstytojo plano sudarymo informacinė sistema
<b>Prieš sąlyga</b>	Dėstytojas turi būti registruotas duomenų bazėje.
<b>Pagrindinis įvykių srautas</b>	<b>Sistemos reakcija ir sprendimai</b>
1. Suvedami duomenys apie individualų dėstytojo planą į formą; 2. Paspaudžiamas patvirtinimo mygtukas.	2.1. Sistema tikrina, ar užpildyti būtinieji laukai; 2.2. Sistema įveda individualaus dėstytojo plano duomenis į duomenų bazę; 2.3. Sistema praneša apie sėkmingą individualaus dėstytojo plano duomenų įvedimą.
<b>Po sąlyga</b>	Individualus dėstytojo planas sudarytas. Padalinio vadovas gali peržiūrėti sudarytą planą ir atlikti informacijos apie planą tvarkymą, t.y. patvirtinti arba nepatvirtinti individualų dėstytojo planą.
<b>Alternatyvos (nesėkmės atvejai)</b>	2.1.a Neužpildyti visi būtinieji laukai: Sistema parodo pranešimą apie neužpildytus būtinuosius laukus ir grįžta į 1 žingsnį; 2.1.b Laukai užpildyti neteisingai: Sistema parodo pranešimą apie užpildytus neteisingai laukus ir grįžta į 1 žingsnį.
<b>Vykdomo variantai</b>	
<b>Veiklos taisyklės</b>	Vartotojas turi įvesti būtinuosius duomenis ir būtinai teisingai įvesti reikalingus duomenis.
<b>Specialūs reikalavimai (nefunkciniai)</b>	Duomenys turi saugiai pasiekti duomenų bazę.

Panaudojimo atvejis - Tvarko informaciją apie individualius dėstytojų planus -&gt; Peržiūri\_

<b>Panaudojimo atvejis</b>	Tvarko informaciją apie individualius dėstytojų planus -> Peržiūri
<b>Aktorius</b>	Dėstytojas
<b>Sistema</b>	Individualaus dėstytojo plano sudarymo informacinė sistema
<b>Prieš sąlyga</b>	Dėstytojas turi būti registruotas duomenų bazėje ir turi būti sudaryti jo individualūs planas.
<b>Pagrindinis įvykių srautas</b>	<b>Sistemos reakcija ir sprendimai</b>
1. Paspaudžiamas prisijungimo prie sistemos mygtukas.	1.1. Sistema išveda pasirinkto dėstytojo individualų planą.
<b>Po sąlyga</b>	Peržiūrimas individualus dėstytojo planas, iš kurio tiesiai galima pereiti į plano redagavimo langą, kuriame galima įvedinėti naujus modulius ar redaguoti jau esamus individualiajame dėstytojo plane modulius.
<b>Alternatyvos (nesėkmės atvejai)</b>	1.1. Dėstytojo individualus planas nesudarytas: Sistema parodo pranešimą apie nesudaryta individualų dėstytojo planą.
<b>Vykdyimo variantai</b>	
<b>Veiklos taisyklės</b>	
<b>Specialūs reikalavimai (nefunkciniai)</b>	

Panaudojimo atvejis - Tvarko informaciją apie individualius dėstytojų planus -&gt; Peržiūri\_

<b>Panaudojimo atvejis</b>	Tvarko informaciją apie individualius dėstytojų planus -> Peržiūri
<b>Aktorius</b>	Padalinio vadovas
<b>Sistema</b>	Individualaus dėstytojo plano sudarymo informacinė sistema
<b>Prieš sąlyga</b>	Padalinio vadovas turi būti registruotas duomenų bazėje ir turi būti sudaryti dėstytojų individualūs planai.
<b>Pagrindinis įvykių srautas</b>	<b>Sistemos reakcija ir sprendimai</b>
1. Pasirenkamas dėstytojas, kurio individualų planą norime peržiūrėti.	1.1. Sistema išveda pasirinkto dėstytojo individualų planą.
<b>Po sąlyga</b>	Peržiūrimas individualus dėstytojo planas, kuri galima patvirtinti arba nepatvirtinti. Patvirtinimo arba nepatvirtinimo atveju pasirinktam dėstytojui išsiunčiamas pranešimas apie jo individualaus plano patvirtinimą arba nepatvirtinimą.
<b>Alternatyvos (nesėkmės atvejai)</b>	1.1. Dėstytojo individualus planas nesudarytas: Sistema parodo pranešimą apie nesudaryta individualų dėstytojo planą.
<b>Vykdyimo variantai</b>	Dėstytoją, kurio individualų planą norima peržiūrėti, galima pasirinkti išsiskleidžiančiame sąrašė.
<b>Veiklos taisyklės</b>	
<b>Specialūs reikalavimai</b>	

*Panaudojimo atvejis - Tvarko informaciją apie individualius dėstytojų planus -> Įrašo modulius į individualų planą\_*

<b>Panaudojimo atvejis</b>	Tvarko informaciją apie individualius dėstytojų planus -> Įrašo modulius į individualų planą_
<b>Aktorius</b>	Dėstytojas
<b>Sistema</b>	Individualaus dėstytojo plano sudarymo informacinė sistema
<b>Prieš sąlyga</b>	Dėstytojas turi būti registruotas duomenų bazėje ir turi būti sudaryti jo individualūs planas.
<b>Pagrindinis įvykių srautas</b>	<b>Sistemos reakcija ir sprendimai</b>
1. Įrašomi duomenys formoje; 2. Paspaudžiamas patvirtinimo mygtukas.	2.1. Sistema tikrina, ar užpildyti būtinieji laukai; 2.2. Sistema įrašo individualaus dėstytojo plano duomenis duomenų bazėje; 2.3. Sistema praneša apie sėkmingą individualaus dėstytojo plano duomenų įrašymą.
<b>Po sąlyga</b>	Duomenys įrašyti į duomenų bazę.
<b>Alternatyvos (nesėkmės atvejai)</b>	2.1.a Neužpildyti visi būtinieji laukai: Sistema parodo pranešimą apie neužpildytus būtinuosius laukus ir grįžta į 1 žingsnį; 2.1.b Laukai užpildyti neteisingai: Sistema parodo pranešimą apie užpildytus neteisingai laukus ir grįžta į 1 žingsnį.
<b>Vykdomo variantai</b>	
<b>Veiklos taisyklės</b>	Vartotojas turi įvesti būtinuosius duomenis ir būtinai teisingai įvesti reikalingus duomenis
<b>Specialūs reikalavimai (nefunkciniai)</b>	Redaguoti duomenys turi saugiai pasiekti duomenų bazę, o ištrinti duomenys turi būti pašalinti iš duomenų bazės

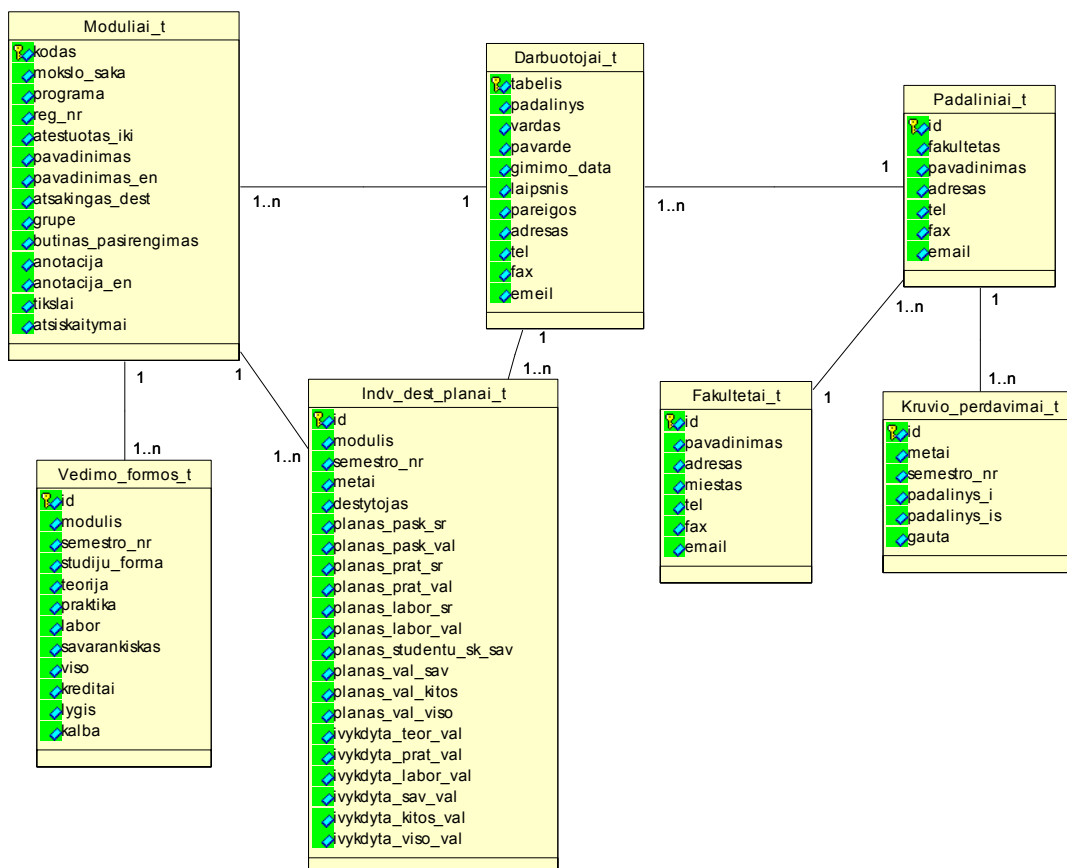
**Tvarko informaciją apie individualius dėstytojų planus -> Ištrina modulius iš individualaus plano\_** – analogiška Tvarko informaciją apie individualius dėstytojų planus -> Įrašo modulius į individualų planą\_ (žr. lentelė 6), tik šiuo atveju iš pateikto dėstytojo individualus plano pašalinamas pasirinktas modulis.

**Tvarko informaciją apie individualius dėstytojų planus -> Redaguoja modulius individualiam plane\_** – analogiška Tvarko informaciją apie individualius dėstytojų planus -> Įrašo modulius į individualų planą\_ (žr. lentelė 6), tik šiuo atveju pateikimas pasirinktas modulis redagavimo formoje.

Panaudojimo atvejis - Persiųsti duomenis į universitetinę DB\_

<b>Panaudojimo atvejis</b>	Persiųsti duomenis į universitetinę DB_
<b>Aktorius</b>	Administratorius
<b>Sistema</b>	Individualaus dėstytojo plano sudarymo informacinė sistema
<b>Prieš sąlyga</b>	Administratoriui turi būti suteiktas vartotojo vardas ir slaptažodis prisijungimui prie informacinės sistemos.
<b>Pagrindinis įvykių srautas</b>	<b>Sistemos reakcija ir sprendimai</b>
<p>1. Suformuojamas XML failas ir schema iš padalinio DB (MS SQL Server DB);</p> <p>2. Suformuojamas XML failas ir schema iš universitetinės DB (Oracle DB);</p> <p>3. Peržiūrimi suformuoti XML failai ir schemas;</p> <p>4. Persiunčiami XML faile (MS SQL Server DB XML failas) esantys duomenys į universitetinę DB (Oracle DB);</p> <p>5. Atstatomi XML faile (MS SQL Server DB XML failas) esantys duomenys į MS SQL Server DB.</p>	<p>1.1. Sistema tikrina, ar galima prisijungti prie padalinio DB;</p> <p>1.2. Sistema suformuoja XML failą ir schema;</p> <p>2.1. Sistema tikrina, ar galima prisijungti prie universitetinės DB;</p> <p>2.2. Sistema suformuoja XML failą ir schema;</p> <p>3.1. Sistema tikrina, ar suformuoti XML failai ir schemas;</p> <p>3.2. Sistema pateikia XML failus ir schemas peržiūrai;</p> <p>4.1. Sistema tikrina, ar galima prisijungti prie universitetinės DB;</p> <p>4.2. Sistema persiunčia ir surašo XML faile (MS SQL Server DB XML failas) esančius duomenis į universitetinę DB;</p> <p>5.1. Sistema tikrina, ar galima prisijungti prie universitetinės DB;</p> <p>5.2. Sistema surašo XML faile (MS SQL Server DB XML failas) esančius duomenis į padalinio DB;</p>
<b>Po sąlyga</b>	XML faile esantys duomenys persiųsti ir surašyti į universitetinę DB.
<b>Alternatyvos (nesėkmės atvejai)</b>	<p>1.1. Nepavyko prisijungti prie padalinio DB: Sistema parodo pranešimą, kad nepavyko prisijungti prie DB;</p> <p>2.1. Nepavyko prisijungti prie universitetinės DB: Sistema parodo pranešimą, kad nepavyko prisijungti prie DB;</p> <p>3.1. Nepavyko suformuoti XML failų ir schemų: Sistema parodo pranešimą, kad nesuformuoti XML failai ir schemas;</p> <p>4.1. Nepavyko prisijungti prie universitetinės DB: Sistema parodo pranešimą, kad nepavyko prisijungti prie DB;</p> <p>5.1. Nepavyko prisijungti prie padalinio DB: Sistema parodo pranešimą, kad nepavyko prisijungti prie DB;</p>
<b>Vykdyimo variantai</b>	
<b>Veiklos taisyklės</b>	
<b>Specialūs reikalavimai (nefunkciniai)</b>	<p>Duomenys iš XML failo (MS SQL Server DB XML failas) turi saugiai pasiekti universitetinę duomenų bazę.</p> <p>Duomenys iš XML failo (MS SQL Server DB XML failas) turi būti saugiai atstatyti į padalinio duomenų bazę.</p>

#### 4. Dalykinės srities klasių diagrama



40 pav. Dalykinės srities klasių diagrama

#### Fakultetai\_t - informacija apie KTU fakultetus

Pavadinimas	Tipas	Dydis	Reikšmė
id	decimal	5	fakulteto kodas
pavadinimas	nchar	20	fakulteto pavadinimas
adresas	nchar	50	fakulteto adresas
miestas	nchar	20	miestą, kuriame yra fakultetas
tel	nchar	15	fakulteto telefono numerį
fax	nchar	15	fakulteto fakso numeris
email	nchar	40	fakulteto elektroninio pašto adresas

#### Padaliniai\_t - aprašo padalinius fakultete, pvz.: kompiuterių katedra

Pavadinimas	Tipas	Dydis	Reikšmė
id	decimal	5	padalinio kodas
pavadinimas	nchar	30	padalinio pavadinimas
fakultetas	decimal	5	nurodo, kokiam fakultetui priklauso padalinys
adresas	nchar	50	padalinio adresas
tel	nchar	15	padalinio telefono numerį
fax	nchar	15	padalinio fakso numeris
email	nchar	40	padalinio elektroninio pašto adresas

**Darbuotojai\_t** – informacija apie KTU darbuotojus (pvz.: dėstytojus)

<b>Pavadinimas</b>	<b>Tipas</b>	<b>Dydis</b>	<b>Reikšmė</b>
tabelis	nchar	10	darbuotojo id
padalinys	decimal	9	padalinys, kuriame darbuotojas dirba
vardas	nchar	20	darbuotojo vardas
pavarde	nchar	30	darbuotojo pavardė
gimimo_data	datetime	8	darbuotojo gimimo data
laipsnis	nvarchar	50	žodžiu parašytas darbuotojo laipsnis
pareigos	nvarchar	50	darbuotojo pareigos. pvz, vedėjas
adresas	nvarchar	50	darbuotojo adresas
tel	nchar	15	darbuotojo telefono numeris
fax	nchar	15	darbuotojo fakso numeris
email	nvarchar	50	darbuotojo elektroninio pašto adresas
pass	nvarchar	30	darbuotojo slaptažodis prisijungti prie DB

**Moduliai\_t** – informacija apie dėstomus modulius, pvz.: Informacinių sistemų modulis

<b>Pavadinimas</b>	<b>Tipas</b>	<b>Dydis</b>	<b>Reikšmė</b>
kodas	nchar	8	modulio kodas
mokslo_saka	nchar	4	modulio mokslo šaka
programa	nchar	1	modulio programos numeris
reg_nr	nchar	3	modulio registracijos numeris
atestuotas_iki	datetime	8	data iki kada modulis atestuotas
pavadinimas	nvarchar	60	modulio pavadinimas
pavadinimas_en	nvarchar	60	modulio pavadinimas anglų kalba
atsakingas_dest	nchar	10	už modulį atsakingo dėstytojo kodas
grupe	nchar	1	modulio grupė
butinas_pasirengimas	nvarchar	120	moduliui būtinas pasirengimas
anotacija	nvarchar	480	informaciją apie modulį (anotaciją)
anotacija_en	nvarchar	480	informaciją apie modulį anglų kalba
tikslai	nvarchar	420	informacija apie modulio tikslus
atsiskaitymai	nvarchar	240	informacija apie atsiskaitymus už modulį

**Vedimo formos\_t** – informacija apie modulio vedimo formą

<b>Pavadinimas</b>	<b>Tipas</b>	<b>Dydis</b>	<b>Reikšmė</b>
id	decimal	17	eilės numeris
modulis	nchar	8	modulio kodas
semestro_nr	nchar	1	semestro numeris (pvz.: R (rudens))
studiju_forma	nchar	1	studijų forma, kuriai skirtas šis modulis (pvz.: D (dieninė))
teorija	decimal	17	teorijai skirtų valandų skaičius
praktika	decimal	17	praktikai skirtų valandų skaičius
labor	decimal	17	laboratoriniams darbams skirtų valandų skaičius
savarankiskas	decimal	17	savarankiškam darbui skirtų valandų skaičius
viso	decimal	17	bendras valandų skaičius, skirtas moduliui
kreditai	decimal	13	modulio kreditų skaičius
lygis	nchar	1	lygis, kuriam skirtas šis modulis (pvz.: B (bakalauras))
kalba	nchar	1	kalba, kuria dėstomas modulis (pvz.: L (lietuvių kalba))

**Kruvio\_perdavimai\_t** - darbuotojų krūvio perdavimai

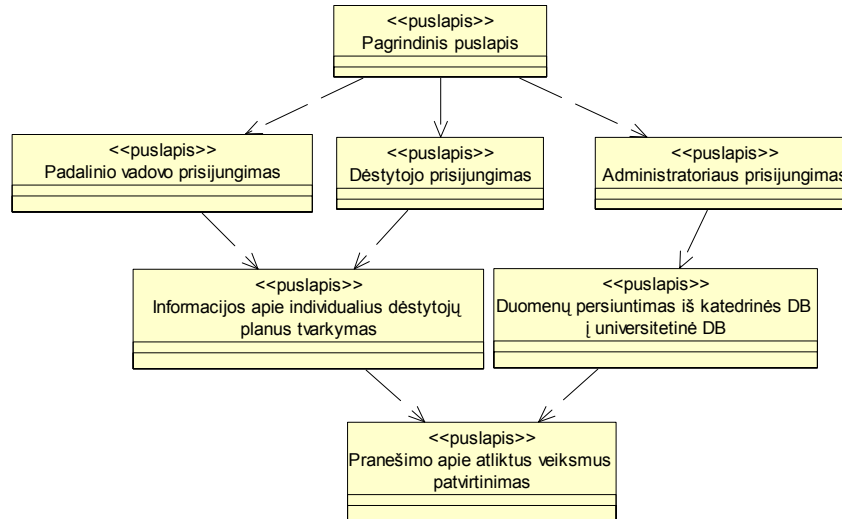
<b>Pavadinimas</b>	<b>Tipas</b>	<b>Dydis</b>	<b>Reikšmė</b>
id	decimal	9	eilės numeris
metai	decimal	17	metai, kuriais perduodamas krūvis
semestro_nr	nchar	1	semestro numeris (pvz.: R (rudens))
padalinys_i	decimal	9	padalinys, kuriam perduodamas krūvis
padalinys_is	decimal	9	padalinys, iš kurio perduodamas krūvis
gauta	numeric	13	nurodoma ar krūvis gautas

**Indv\_dest\_planai\_t** – informacija apie darbuotojų individualius planus

<b>Pavadinimas</b>	<b>Tipas</b>	<b>Dydis</b>	<b>Reikšmė</b>
id	decimal	9	eilės numeris
modulis	nchar	8	modulio kodas
semestro_nr	nchar	1	semestro numeris (pvz.: R (rudens))
metai	decimal	17	metai, kuriais dėstomas modulis
dėstytojas	nchar	10	dėstytojo kodas
planas_pask_sr	decimal	17	planuojamų paskaitų srautas
planas_pask_val	decimal	17	planuojamų paskaitų valandų skaičius
planas_prat_sr	decimal	17	planuojamų pratybų srautas
planas_prat_val	decimal	17	planuojamų pratybų valandų skaičius
planas_labor_sr	decimal	17	planuojamų laboratorinių srautas
planas_labor_val	decimal	17	planuojamų laboratorinių valandų skaičius
planas_studentu_sk_sav	decimal	17	planuojamų studentų skaičius per savaitę
planas_val_sav	decimal	17	planuojamų valandų skaičius per savaitę
planas_val_kitos	decimal	17	planuojamų valandų skaičius kitiems darbams
planas_val_viso	decimal	17	bendras planuojamų valandų skaičius
ivykdyta_pask_val	decimal	17	įvykdytų paskaitų valandų skaičius
ivykdyta_prat_val	decimal	17	įvykdytų pratybų valandų skaičius
ivykdyta_labor_val	decimal	17	įvykdytų laboratorinių valandų skaičius
ivykdyta_sav_val	decimal	17	įvykdytų valandų skaičius per savaitę
ivykdyta_kitos_val	decimal	17	įvykdytų valandų skaičius kitiems darbams
ivykdyta_viso_val	decimal	17	bendras įvykdytų valandų skaičius



## 5. Vartotojų interfeiso modelis



41 pav. Vartotojo interfeiso modelis

## 6. Reikalavimai sistemos funkcionalumui

### 6.1. Reikalavimai veikimo sąlygoms

Sukurta programa bus patalpinta serveryje. Serveris turi palaikyti ASP.NET. Vartotojas norėdamas naudotis programa turi naudojantis viena iš populiarių Interneto naršyklių (Internet Explorer 5.0, Netscape Communicator 4.0 ar Opera). Interneto puslapis geriausiai matomas 1024 x 768 vaizduoklio skiriamąja geba.

### 6.2. Reikalavimai sistemos patikimumui, patogumui, priežiūrai ir perkeliamumui

Sukurta sistema tolerantiška klaidoms. Įvykus klaidai sistema pateikia informaciją lietuvių kalba apie įvykusią klaidą.

Sukurta programa patogi jos vartotojams, nes informacija puslapiuose išdėstyta medžio principu. Kiekvieno puslapio įrankiai išdėstyti meniu principu.

Ypatingų reikalavimų sistemos priežiūrai nėra. Programą lengva perkelti iš vieno serverio į kitą nukopijuojant programos failus. Serveris turi palaikyti ASP.NET

### 6.3. Reikalavimai saugumui

Programoje bus numatyta programos apsauga. Programa galės naudotis tik registruoti programos vartotojai.

#### ***6.4. Reikalavimai vykdymo charakteristikoms***

Programoje turi būti numatytas naujų duomenų, susijusių su dėstytojų individualiais planais, įvedimas, duomenų bazėje esančių duomenų redagavimas, šalinimas.

Duomenų įvedimo ir redagavimo metu turi būti vykdoma duomenų kontrolė.

Programoje turi būti numatytas XML failų sukūrimas iš MS SQL Server DB ir Oracle DB ir išsaugojimas, peržiūra, duomenų nusiuntimas iš MS SQL Server DB į Oracle DB ir duomenų atstatymas į MS SQL Server DB.

Persiunčiant duomenis iš vienos duomenų bazės į kitą ir atstatant duomenis turi būti užtikrinta, kad duomenys saugiai pasieks duomenų bazę.

### **7. Informacijos sistemai keliami nefunkciniai reikalavimai ir apribojimai**

#### ***7.1. Reikalavimai sistemos išvaizdai***

Programos išvaizdai specifinių reikalavimų nėra ir programa bus daroma atsižvelgiant į programuotojo pasirinktas spalvas.

#### ***7.2. Reikalavimai panaudojamumui***

Sukurta programa turi atitikti techninėje užduotyje aprašytus reikalavimus ir turi būti patogi savo galimybėmis bei vartotojo sąsaja programos vartotojų naudojimui.

#### ***7.3. Reikalavimai vykdymo charakteristikoms***

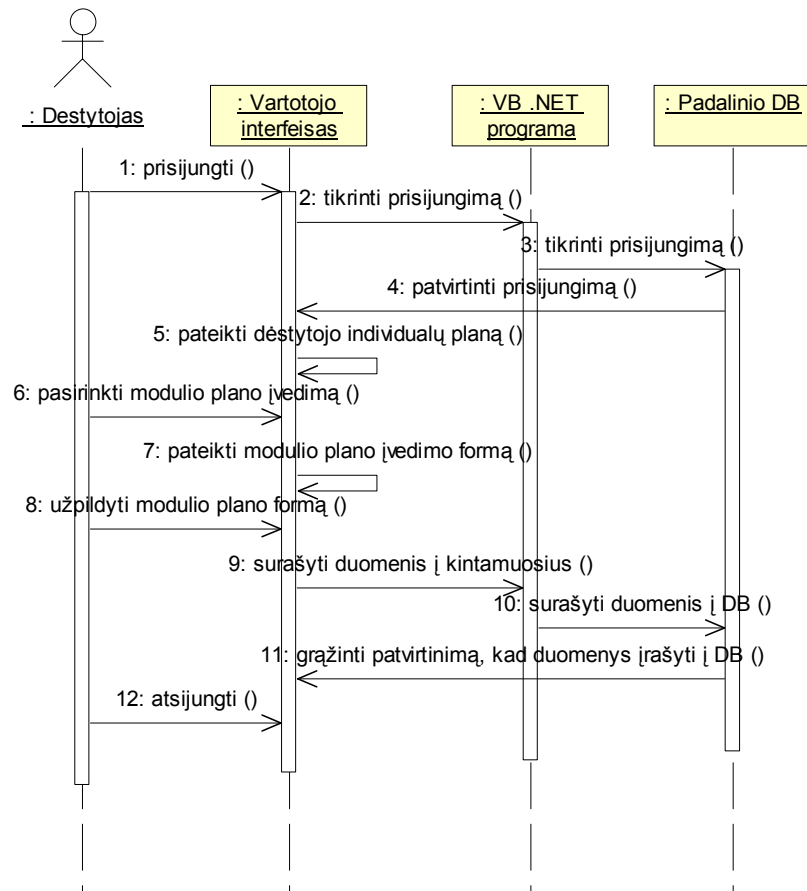
Programoje turi leisti sudarinėti individualius dėstytojų planus ir turi leisti persiųsti duomenis iš vienos duomenų bazės į kitą panaudojant XML ir užtikrinant duomenų kontrolę.

#### ***7.4. Reikalavimai patogumui, priežiūrai ir perkeliamumui***

Sukurta programa turi būti patogi jos vartotojams. Ypatingi reikalavimai programos priežiūrai nėra keliami. Programą turi būti lengva perkelti iš vieno serverio į kitą. Serveris turi palaikyti ASP.NET

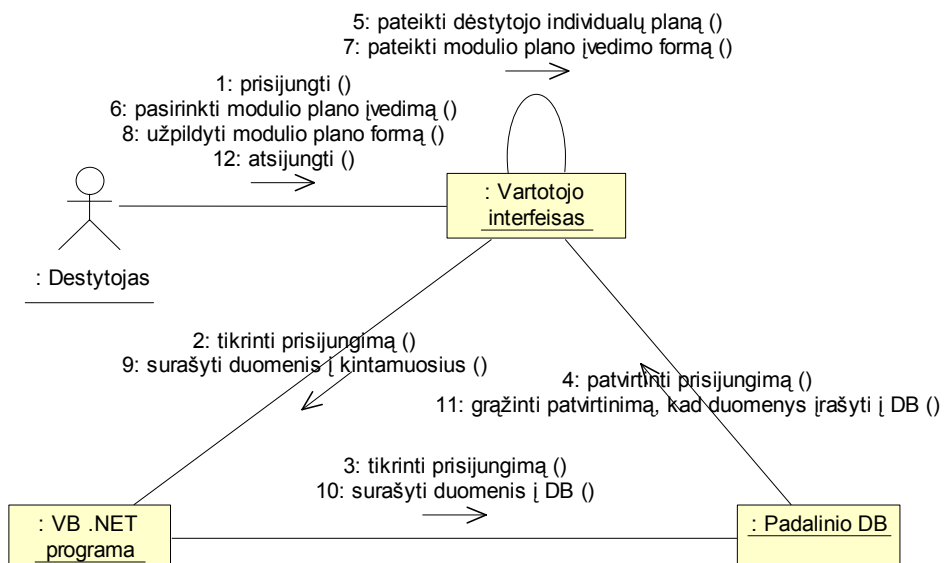
## 8. Sistemos panaudojimo atvejų ir sekų (arba bendradarbiavimo) diagramos

### 8.1 Individualaus dėstytojo plano sudarymo sekų diagrama



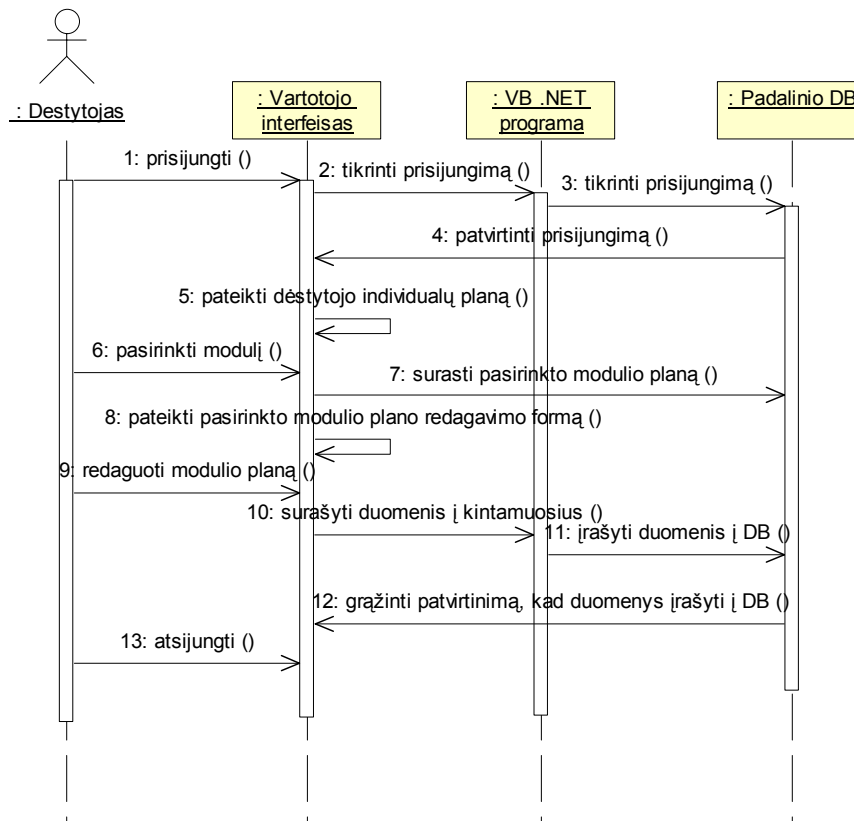
42 pav. Individualaus dėstytojo plano sudarymo sekų diagrama

### 8.2 Individualaus dėstytojo plano sudarymo bendradarbiavimo diagrama



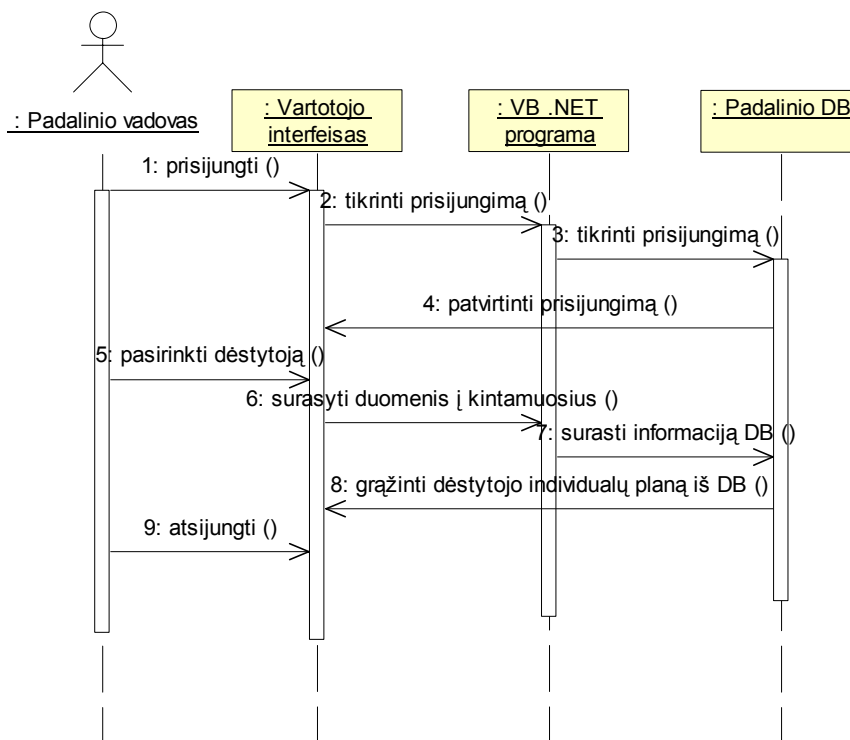
43 pav. Individualaus dėstytojo plano sudarymo bendradarbiavimo diagrama

### 8.3 Dėstytojo plano redagavimo sekų diagrama



44 pav. Dėstytojo plano redagavimo sekų diagrama

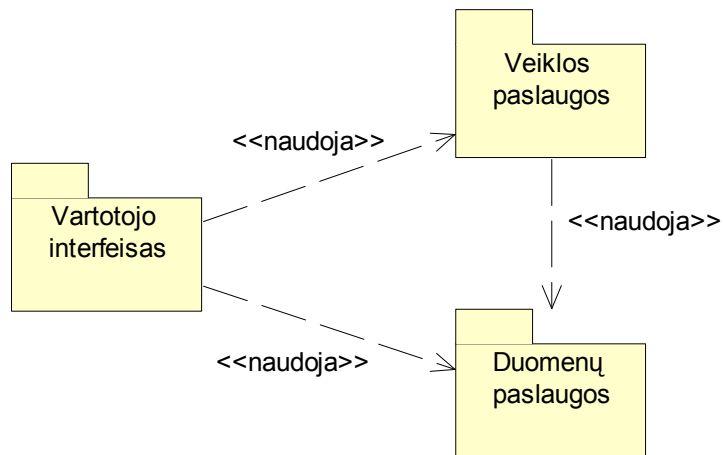
### 8.4 Individualaus dėstytojo plano peržiūros sekų diagrama



45 pav. Individualaus dėstytojo plano peržiūros sekų diagrama

Analogiškos sekų diagrama gaunama individualaus dėstytojo plano peržiūrai, kai peržiūri dėstytojas. Dėstytojui peržiūrint individualų dėstytojo planą nereikia pasirinkti dėstytojo, nes sistema pateikia jo individualų planą.

## 9. Sistemos architektūra - statinės struktūros modelis



46 pav. Sistemos architektūra - statinės struktūros modelis

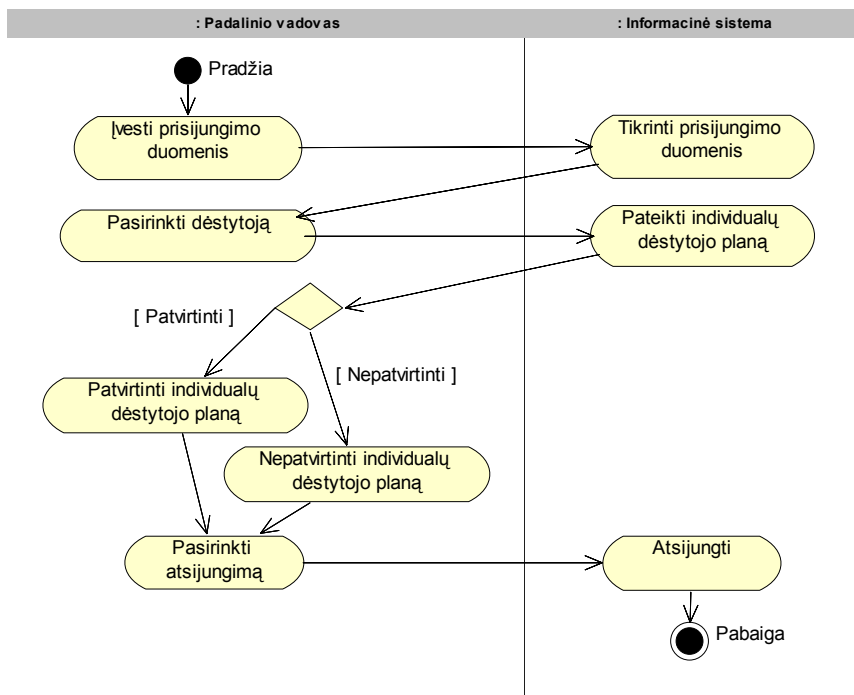
Vartotojo interfeisų posistemį sudaro – padalinio vadovo, dėstytojo ir administratoriaus interfeisai.

Veiklos paslaugų posistemį sudaro visos paslaugos, kurias teikia informacinė sistema (individualių dėstytojų planų peržiūra, sudarymas, redagavimas ir šalinimas; duomenų persiuntimas naudojant XML iš MS SQL Server į Oracle DB).

Duomenų paslaugų posistemį sudaro – MS SQL Server ir Oracle duomenų bazės, kuriomis naudojasi informacinė sistema. MS SQL Server (padalinio DB) saugoma informacija apie individualius dėstytojų planus, kuriuos galima nuskaityti, įrašyti ir saugoti, o Oracle (universitetinė DB) naudojama duomenų persiuntimui ir saugojimui iš MS SQL Server DB.

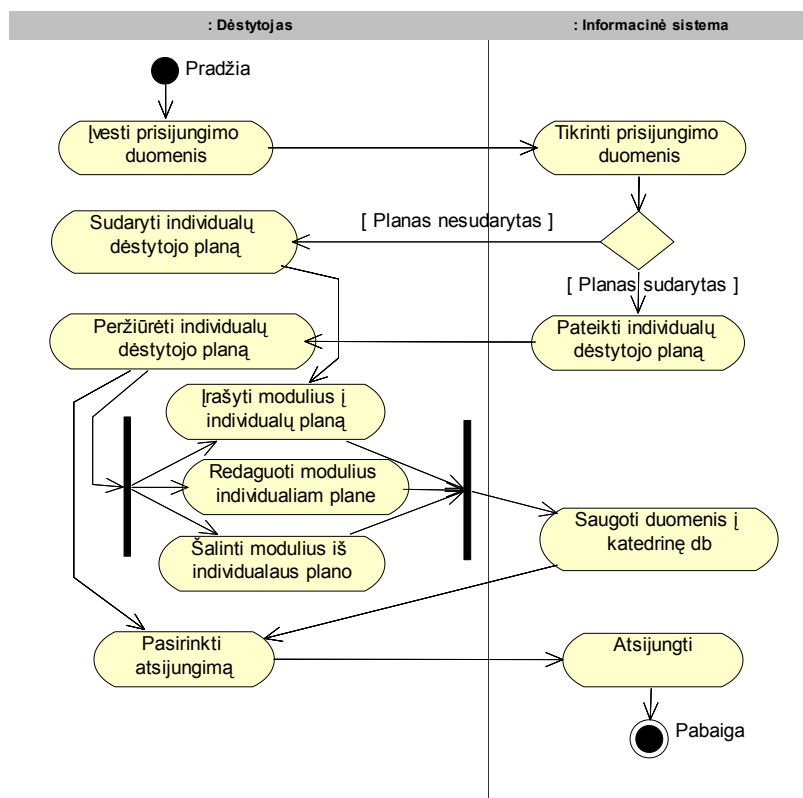
## 10. Sistemos elgsenos modelis

### 10.1. Sistemos elgsenos diagrama, kai vartotojas padalinio vadovas



47 pav. Sistemos elgsenos diagrama, kai vartotojas padalinio vadovas

### 10.2. Sistemos veiklos diagrama, kai vartotojas dėstytojas



48 pav. Sistemos veiklos diagrama, kai vartotojas dėstytojas

## 11. Reikalavimai sistemos funkcionavimo palaikymui

Techninės įrangos reikalavimai:

- Pentium su 64 MB RAM
- Priemonės, palaikančios ryšį su Internetu (modemas, tinklo plokštė)
- VGA monitorius
- Pelė
- Klaviatūra

Programinės įrangos reikalavimai:

- Microsoft Windows 9x/XP, NT/2000
- Interneto naršyklė:
  - ✓ Internet Explorer 5.0
  - ✓ Netscape Communicator 4.0
  - ✓ Opera

Organizacinės aplinkos reikalavimai:

- MS SQL Server
- Oracle
- VB .NET
- ODBC

## 12. Sistemos naudojimo instrukcija

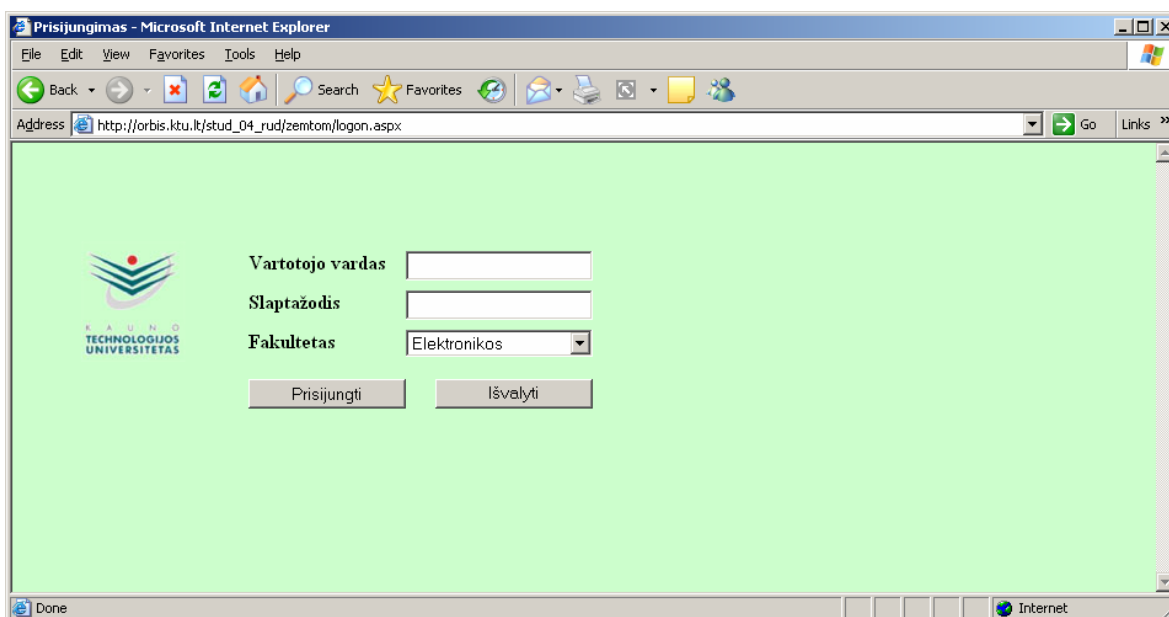
### 12.1. Dėstytojo – sistemos vartotojo naudojimo instrukcija

Šiame skyriuje aprašysime leistinas dėstytojų funkcijas, jų atlikimo tvarką ir patarimus.

#### ▪ **Prisijungimas**

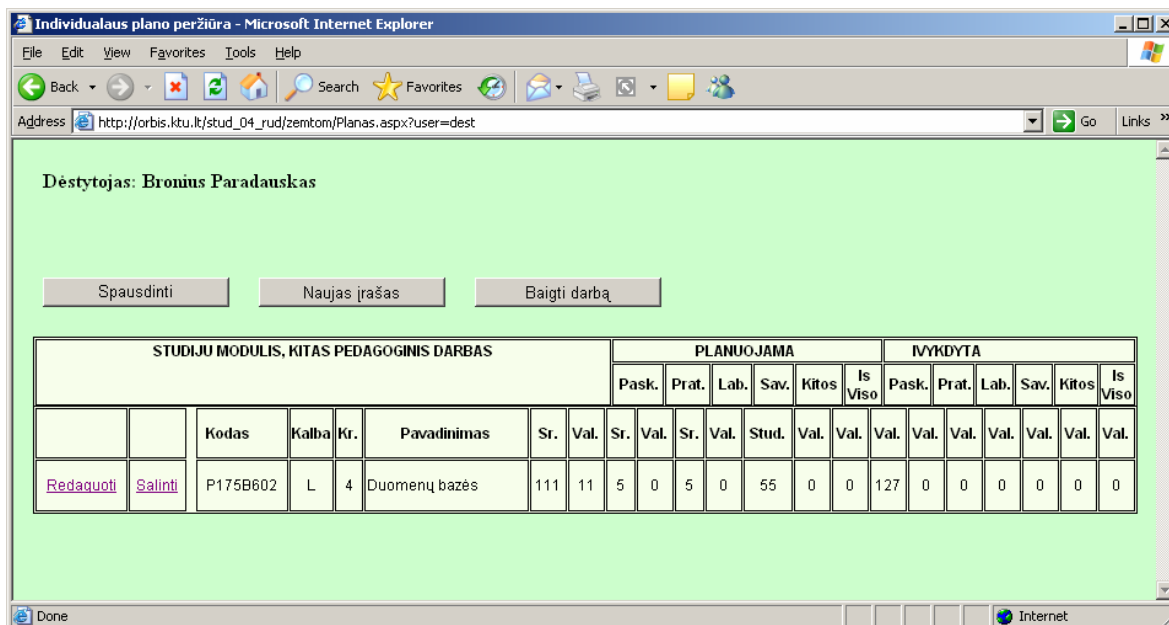
Prisijungimas atliekamas naudojantis viena iš reikalavimuose išvardintų naršyklių įvedant adresą, kuriuo yra patalpinta programa, adreso įvedimo lauke. Programa patalpinta serveryje orbis.ktu.lt adresu [http://orbis.ktu.lt/stud\\_04\\_rud/zemtom/logon.aspx](http://orbis.ktu.lt/stud_04_rud/zemtom/logon.aspx). Gali tekti palaukti kurį laiką kol bus užkrautas titulinis sistemos puslapis.

## ▪ Prisijungimo langas



49 pav. Prisijungimo prie sistemos langas

Dėstytojas prie sistemos prisijungia įveddamas prisijungimo vardą, kuris yra dėstytojo tabelio numeris, slaptažodį ir pasirinkdamas fakultetą, kuriame jis dirba. Įvedęs savo duomenis paspaudžia mygtuką „Prisijungti“. Jei įvedant dėstytojo tabelio numerį ar slaptažodį ar pasirenkant fakultetą nebuvo padaryta klaidų – sistema prisijungs, jei ne sistema praneš apie klaidą. Prisijungus dėstytojui, jam bus užkrautas pagrindinis sistemos langas. 49 pav. Parodytas dėstytojo pasirinkimo langas.



50 pav. Pagrindinis dėstytojo langas



Dėstytojas gali vaikščioti per puslapius naudodamasis pateiktomis nuorodomis - spustelėjęs pele norimą nuorodą nueina į atitinkamą puslapį. Dėstytojas gali pasirinkti tokius variantus:

1. Spausdinti – atsispausdinti savo individualų planą;
2. Naujas įrašas – priskirti naują informaciją apie pasirinktą modulį savo individualiam planui;
3. Redaguoti – redaguoti informaciją apie pasirinktą modulį iš savo individualaus plano;
4. Šalinti – šalinti pasirinkto modulio informaciją iš savo individualaus plano;
5. Baigti darbą – darbo su sistema pabaiga.

▪ **Puslapis „Naujas įrašas“**

Į šį puslapį patenkama pagrindiniame puslapyje nuspaudus mygtuką „Naujas įrašas“.

Individualaus plano redagavimas - Microsoft Internet Explorer

Address [http://orbis.ktu.lt/stud\\_04\\_rud/zemtom/Modulis\\_red.aspx?k=1&user=dest](http://orbis.ktu.lt/stud_04_rud/zemtom/Modulis_red.aspx?k=1&user=dest)

Kodas: H100B309 | Dalykinis bendravimas anglų kalba | [rašyti]

Kalba: L | [Grįžti]

Kreditai: 4 | Semestro\_nr: R

Pavadinimas: Dalykinis bendravimas anglų kalba | Metai: 2004

Paskaitų planas   Pratybų planas   Laboratorinių planas   Savaitinis planas

sr	0	sk	0	sk	0	studentų	0	Kitos val	0
val	0	val	0	val	0	val	0	Viso val	0
Įvykdyta paskaitų val	0	Įvykdyta laboratorinių val	0	Įvykdyta kitos val	0				
Įvykdyta pratybų val	0	Įvykdyta savaitinių val	0	Įvykdyta viso val	0				

51 pav. Puslapis „Naujas įrašas“

- **Puslapis „Redaguoti“**

Į šį puslapį patenkama pagrindiniame puslapyje nuspaudus mygtuką „Redaguoti“.

The screenshot shows a web browser window titled "Individualaus plano redagavimas - Microsoft Internet Explorer". The address bar contains the URL: [http://orbis.ktu.lt/stud\\_04\\_rud/zemtom/modulis\\_red.aspx?kodas=P175B602&k=2](http://orbis.ktu.lt/stud_04_rud/zemtom/modulis_red.aspx?kodas=P175B602&k=2). The form has a light green background and contains the following fields and buttons:

Kodas	<input type="text" value="P175B602"/>	<input type="button" value="Irašyti"/>
Kalba	<input type="text" value="L"/>	<input type="button" value="Grįžti"/>
Kreditai	<input type="text" value="4"/>	Semestro_nr <input type="text" value="R"/>
Pavadinimas	<input type="text" value="Duomenų bazės"/>	Metai <input type="text" value="2004"/>

Below these are four tabs: "Paskaitų planas", "Pratybų planas", "Laboratorinių planas", and "Savaitinis planas". Under each tab are input fields for "sr" (start), "sk" (end), "val" (hours), and "studentų" (students). There are also fields for "Kitos val" (other hours) and "Viso val" (total hours). At the bottom, there are fields for "Įvykdyta" (executed) for each category, all currently set to 0.

52 pav. Puslapis „Redaguoti“

- **Puslapis „Šalinti“**

Į šį puslapį patenkama pagrindiniame puslapyje nuspaudus mygtuką „Šalinti“.

The screenshot shows a web browser window titled "Ištrintas įrašas - Microsoft Internet Explorer". The address bar contains the URL: [http://orbis.ktu.lt/stud\\_04\\_rud/zemtom/irasas\\_pan.aspx?kodas=P170B605&trinti=2](http://orbis.ktu.lt/stud_04_rud/zemtom/irasas_pan.aspx?kodas=P170B605&trinti=2). The main content area has a light green background and displays the message "Įrašas sėkmingai pašalintas!" in red text. Below the message is a single button labeled "Grįžti".

53 pav. Puslapis „Šalinti“

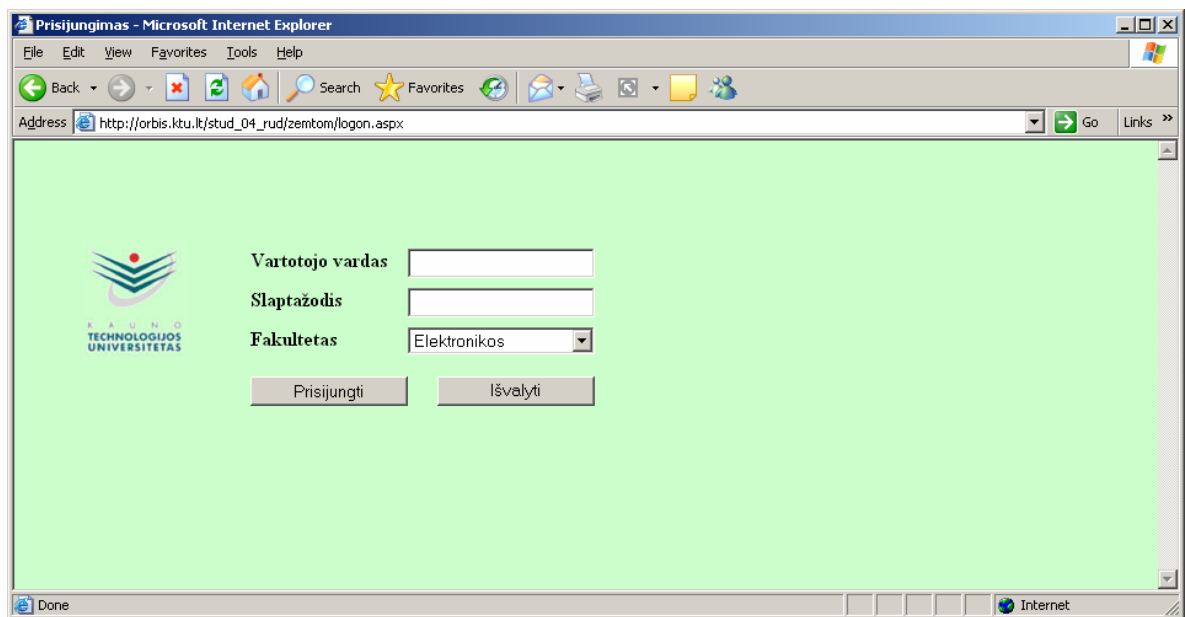
## 12.2. Padalinio vadovas – sistemos vartotojo naudojimo instrukcija

Šiame skyriuje aprašysime leistinas padalinio vadovui funkcijas, jų atlikimo tvarką ir patarimus.

### ▪ Prisijungimas

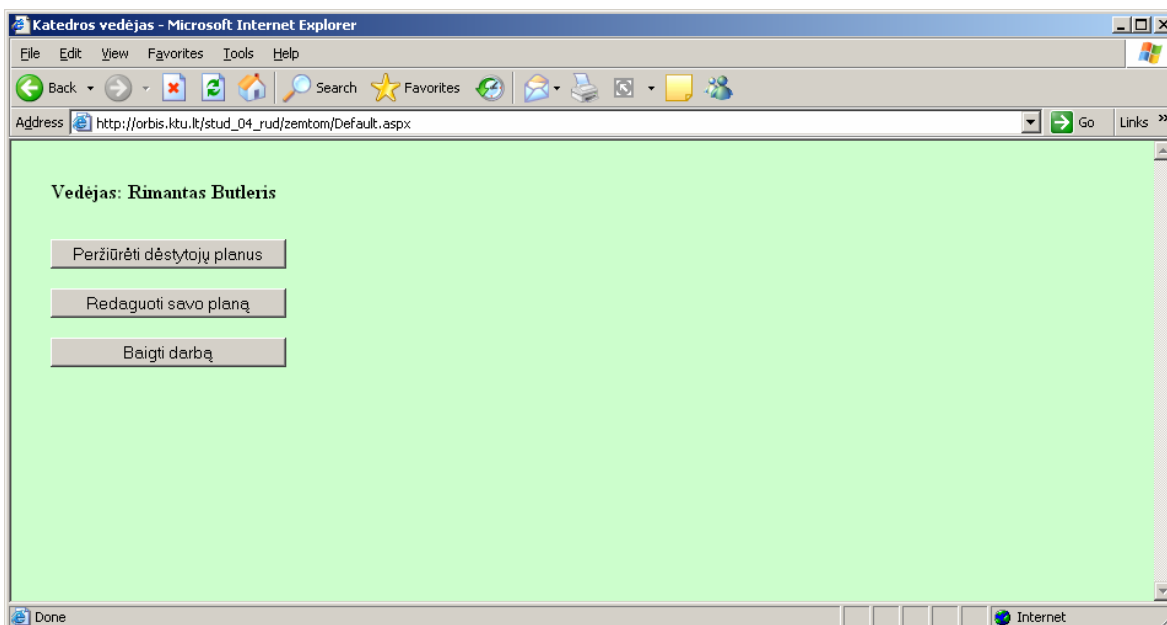
Prisijungimas atliekamas naudojantis viena iš reikalavimuose išvardintų naršyklių įvedant adresą, kuriuo yra patalpinta programa, adreso įvedimo lauke. Programa patalpinta serveryje orbis.ktu.lt adresu [http://orbis.ktu.lt/stud\\_04\\_rud/zetom/logon.aspx](http://orbis.ktu.lt/stud_04_rud/zetom/logon.aspx). Gali tekti palaukti kurį laiką kol bus užkrautas titulinis sistemos puslapis.

### ▪ Prisijungimo langas



54 pav. Prisijungimo prie sistemos langas

Padalinio vadovas prie sistemos prisijungia įveddamas prisijungimo vardą, kuris yra padalinio vadovo tabelio numeris, slaptažodį ir pasirinkdamas fakultetą, kuriame jis dirba. Įvedęs savo duomenis paspaudžia mygtuką „Prisijungti“. Jei įvedant dėstytojo tabelio numerį ar slaptažodį ar pasirenkant fakultetą nebuvo padaryta klaidų – sistema prisijungs, jei ne sistema praneš apie klaidą. Prisijungus padalinio vadovui, jam bus užkrautas pagrindinis sistemos langas. 54 pav. Parodytas padalinio vadovo pasirinkimo langas.

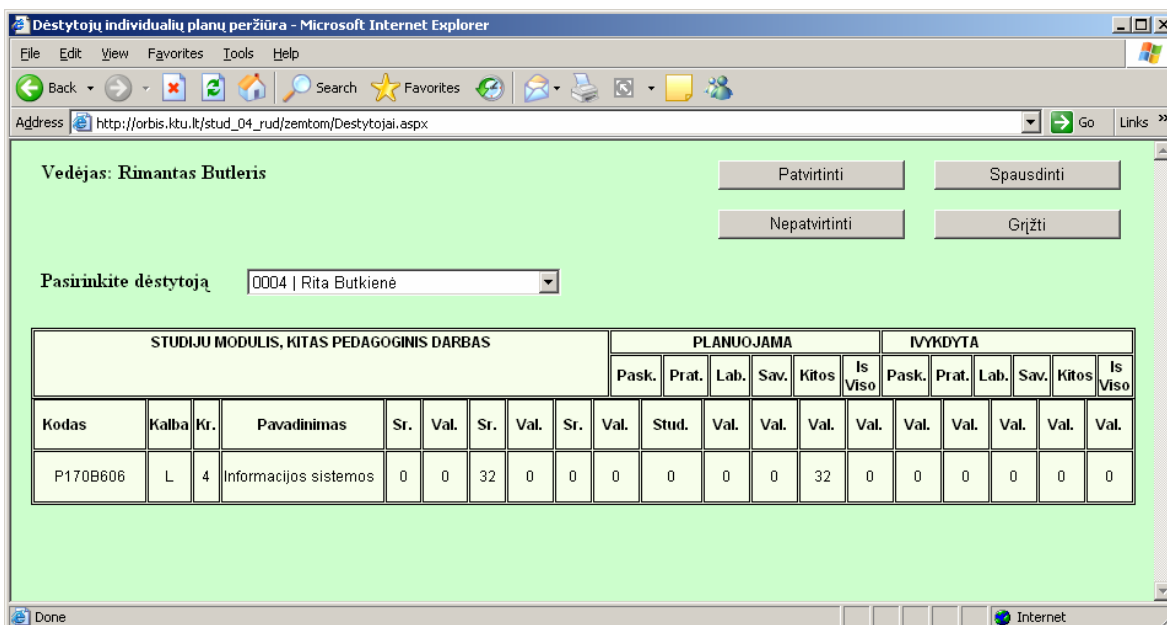


55 pav. Pagrindinis padalinio vadovo langas

Padalinio vadovas gali vaikščioti per puslapius naudodamasis pateiktomis nuorodomis - spustelėjęs pele norimą nuorodą nueina į atitinkamą puslapį. Dėstytojas gali pasirinkti tokius variantus:

1. Peržiūrėti dėstytojų planus – peržiūrėti padalinio dėstytojų individualius planus;
2. Redaguoti savo planą – redaguoti informaciją apie pasirinktą modulį iš savo individualaus plano;
3. Baigti darbą – darbo su sistema pabaiga.

▪ **Puslapis „Peržiūrėti dėstytojų planus“**

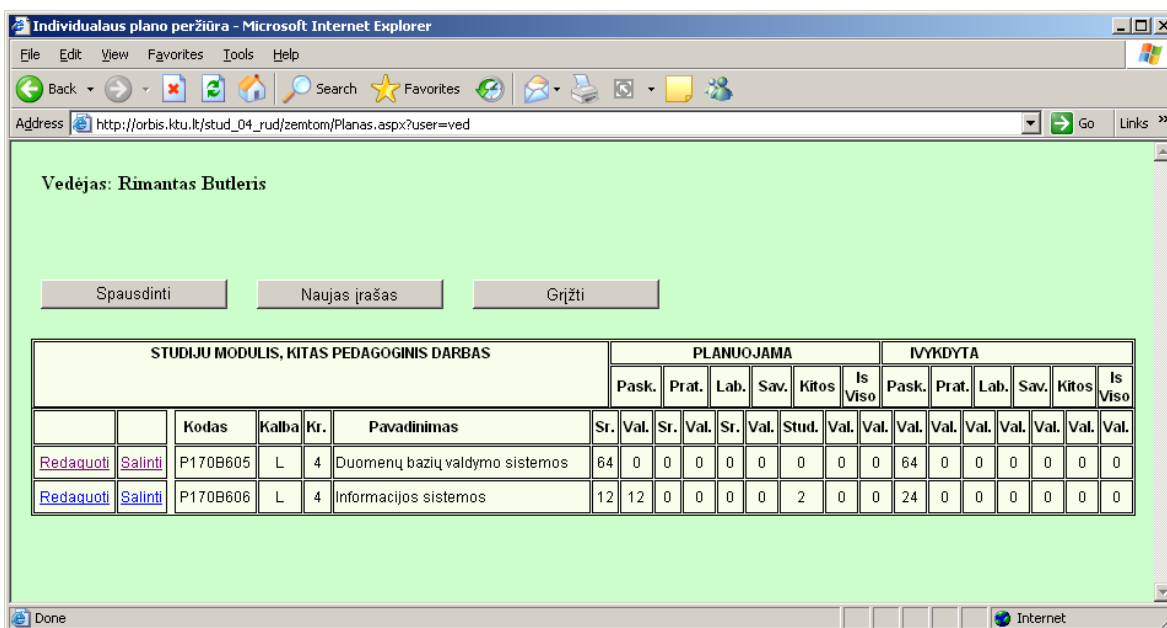


56 pav. Puslapis „Peržiūrėti dėstytojų planus“

Padalinio vadovas šiame puslapyje gali pasirinkti tokius variantus:

1. Patvirtinti – nusiųsti pranešimą į dėstytojo elektroninį paštą apie jo individualaus plano patvirtinimą;
2. Nepatvirtinti – nusiųsti pranešimą į dėstytojo elektroninį paštą apie jo individualaus plano nepatvirtinimą;
3. Spausdinti – atsispausdinti norimo dėstytojo individualų planą;
4. Grįžti – grįžti į pagrindinį padalinio vadovo langą.

▪ **Puslapis „Redaguoti savo planą“**



57 pav. Puslapis „Redaguoti savo planą“

Padalinio vadovas šiame puslapyje gali pasirinkti tokius variantus:

1. Spausdinti – atsispausdinti savo individualų planą;
2. Naujas įrašas – priskirti naują informaciją apie pasirinktą modulį savo individualiam planui;
3. Redaguoti – redaguoti informaciją apie pasirinktą modulį iš savo individualaus plano;
4. Šalinti – šalinti pasirinkto modulio informaciją iš savo individualaus plano;
5. Grįžti – grįžti į pagrindinį padalinio vadovo langą.

### **13. Informacijos sistemos diegimo priemonių planas**

Informacinė sistema yra įdiegiama tokiu būdu:

1. VB .NET programos projektą reikia perkopijuoti į sistemai skirtą Windows NT serverio direktoriją. Serveris turi palaikyti ASP.NET;
2. Turi būti įdiegtas MS SQL serveris;
3. Turi būti įdiegtas Oracle serveris.