

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
KOMPIUTERIŲ KATEDRA

Ramūnas Norvaišas

**DUOMENŲ APIE PILIEČIŲ KREDITUS
INTEGRAVIMO SISTEMA**

Magistro darbas

Darbo vadovas: doc. Valentinas Kiauleikis

Kaunas, 2008

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
KOMPIUTERIŲ KATEDRA

DUOMENŲ APIE PILIEČIŲ KREDITUS
INTEGRAVIMO SISTEMA

Magistro darbas

Vadovas:

doc. Valentinas Kiauleikis
2008-01-09

Recenzentė:

prof. Lina Nemuraitė
2008-01-10

Atliko:

IFM-1/4 gr. stud.
Ramūnas Norvaišas
2008-01-09

Kaunas, 2008

Turinys

ĮVADAS	5
1. VARTOTOJŲ POREIKIŲ ANALIZĖ.....	7
1.1. Tyrimo sritis, objektas ir problema.....	7
1.2. Modelio sudarymo metodo ir priemonių parinkimas.....	8
1.2.1. Vartotojo reikalavimų specifikavimas naudojant UML.....	9
1.2.2. UML programinio paketo reikalavimai.....	10
1.3. Organizacijos veiklos analizė	11
1.3.1. Analizės tikslas	11
1.3.2. Veiklos sąveikų modelis.....	11
1.3.3. Veiklos panaudojimo atvejų modelis	14
1.3.4. Veiklos tikslų modelis	15
1.3.5. Veiklos objektų modelis	16
1.3.6. Organizacinės struktūros modelis	17
1.3.7. Veiklos procesų modelis	17
1.3.8. Vartotojų bei jų poreikių analizė (reikalavimai sistemos funkcionalumui)	18
1.3.9. Informacijos sistemai keliami nefunkciniai reikalavimai ir apribojimai	19
1.4. Projekto tikslas ir jo pagrindimas, kokybės kriterijų apibrėžimas	19
1.5. Architektūros ir galimų įgyvendinimo priemonių variantų analizė	20
1.5.1. Architektūros analizė.....	20
1.5.2. Technologijų ir architektūros palyginimas.....	21
1.5.3. Programavimo įrankių analizė.....	22
1.6. Esamų sprendimų duomenų integravimo problemai spręsti analizė.....	23
1.6.1. Standartiniai duomenų bazių įrankiai duomenų importavimui ir eksportavimui	23
1.6.2. Kompanijos IBM įrankis duomenų transformavimui.....	24
1.6.3. Duomenų sandėlis.....	25
1.7. XML ir EDI technologijos duomenų integravimui.....	28
1.7.1. EDI formatas	28
1.7.2. XML formatas	29
1.7.3. XML dokumentų nagrinėjimas.....	30
1.7.4. Korektiškumo tikrinimo priemonės.....	31
1.7.5. XML dokumentų transformavimas.....	33
1.7.6. XML ir EDI formatų palyginimas.....	34
1.8. Panašių veikiančių sistemų lyginamoji analizė.....	35
1.9. Analizės išvados	37
2. DUOMENŲ APIE PILIEČIŲ KREDITUS INTEGRAVIMO SISTEMOS MODELIS	38
2.1. Reikalavimų modelis	38
2.1.1. Administravimo modulio panaudojimo atvejų diagrama ir jos specifikacijos.....	38
2.1.2. Duomenų integravimo modulio panaudojimo atvejų diagrama ir jos specifikacijos	40
2.1.3. Internetinės prieigos modulio panaudojimo atvejų diagrama ir jos specifikacijos	44
2.1.4. Dalykinės srities klasių diagrama	46
2.2. Vartotojo sąsajos modelis	47
2.2.1. Administravimo modulio vartotojo sąsajos modelis	47
2.2.2. Duomenų integravimo modulio vartotojo sąsajos modelis.....	48
2.2.3. Internetinio modulio vartotojo sąsajos modelis.....	48
2.3. Sistemos projektas	49
2.3.1. Projekto tikslas	49
2.3.2. Sistemos architektūra	49
2.3.3. Sistemos veiklos diagramos su pagrindinių objektų šranta	52
2.3.4. Duomenų bazės modelis	58
2.3.5. Testavimo modelis bei testavimo atvejai	59
2.3.6. Reikalavimai sistemos funkcionavimo palaikymui.....	59
2.4. Sistemos projektavimo išvados.....	60
3. SISTEMOS REALIZACIJA BEI EKSPERIMENTINIS TYRIMAS	61
3.1. Realizacijos modelis	61

3.2.	Duomenis apie piliečių kreditus integruojančios sistemos testavimas	62
3.2.1.	<i>Administratoriaus sąsajos testavimas</i>	62
3.2.2.	<i>Internetinės prieigos vartotojo sąsajos testavimas</i>	68
3.3.	Sukurtos sistemos kokybės tyrimas	73
3.4.	Tolimesnio sistemos tobulinimo, plėtojimo galimybės	74
IŠVADOS IR REZULTATAI		76
LITERATŪROS SĄRAŠAS		77
TERMINŲ IR SANTRUMPŲ ŽODYNAS		79
SANTRAUKA ANGLŲ KALBA		80
PRIEDAI		81
1 priedas.	Uždavinio techninis išpildymas	81
1.	<i>Specifikacijos panaudojimo atvejams</i>	81
2.	<i>Sistemos panaudojimo atvejų sekų diagramos</i>	84
3.	<i>Duomenų bazės lentelių aprašai</i>	90

ĮVADAS

Šiuolaikiniam ekonomikos vystimosi ir technikos pažangos etape, kai daugelis rinkos sričių Lietuvoje yra užpildytos, įmonės, norinčios optimaliai ir kokybiškai valdyti padėtį rinkoje, priverstos žvalgytis alternatyvių pajamų šaltinių perkeldamos savo teikiamas paslaugas į elektroninį verslą. Taigi nei trupučio nestebina tai, kad verslo subjektų strateginiuose planuose integracija su kitomis biznio sistemomis ir elektroninis verslas užima vis svarbesnę, o gal net svarbiausią vietą [1].

Lietuvoje tik prieš porą metų buvo pradėta teikti e-bankininkystės paslauga. Šiandien jau galima atlikti visas būtinausias sąskaitų tikrinimo, apmokėjimo, pinigų pervedimo, valiutos konvertavimo operacijas. Tačiau vis dar išlieka nemažas bankinių paslaugų paketas, kuris nėra perkeltas į elektroninę paslaugų sistemą.

Paskolos suteikimas internetu yra viena iš sudėtingiausių bankinių operacijų, kai paskolos davėjas turi įvertinti paskolos gavėjo kreditavimo riziką, mokumo ir patikimumo kriterijus [21]. Norint kokybiškai, greitai ir patikimai įvertinti šią riziką bei optimizuoti paskolų išdavimo procesą, būtina sukurti internetinį kredito istoriją saugantį banką, kuris padėtų atskiriems verslo subjektams integruoti savo kaupiamus duomenis į bendrą visumą, juos išanalizuoti pasitelkus ekspertinius - analitinius įrankius, bei rizikos nustatymo metodus. Įsteigus Kredito banką fiziniai asmenys paskolas gaus greičiau ir paprasčiau: kadangi informacija bus laikoma skaitmeniniu pavidalu ir lengvai pasiekama, kreditorius kliento mokumą galės įvertinti akimirksniu [10]. Todėl kuriant tokią sistemą susiduriama su viena aktualiausių informacinių sistemų problema – duomenų naudojimas ir pasidalinimas tarp skirtingų vartotojų, skirtingų įmonių ir skirtingų duomenų saugyklų.

Skirtingų informacinių sistemų naudojimas ir galimas korektiškas bendravimas tarp jų tampa vis aktualesnis. Organizacijų duomenų resursai gali būti daugybėje skirtingų formų: reliacinių duomenų bazių lentelėse, įvairiuose dokumentuose, objektinėse duomenų bazėse, elektroniniuose duomenų apsikeitimo pranešimuose ir daugelyje kitų [14]. Todėl organizacijų kompiuterinių sistemų integracija (EAI) turi išspręsti skirtingų sistemų naudojamą bendrais verslo procesais ir duomenimis. Informacinės sistemos integravimą (EAI) galima vykdyti keliais būdais: duomenų lygio, uždavinio lygio, metodo lygio ir vartotojo sąsajos lygio integracija [3].

Šiame darbe bus nagrinėjama tik duomenų lygio integracija, kadangi ji yra pigiausias ir paprasčiausias būdas įmonių verslo informacijos apsikeitimui tarp skirtingų sistemų. Taip pat šis būdas buvo pasirinktas, nes jam yra sukurta daug įvairių priemonių ir technologijų, norint greitai ir kokybiškai atvaizduoti duomenų šaltinio atvaizdą integruojančioje sistemoje. Integruojant sistemas duomenų integracijos lygyje užtenka žinoti: kur yra duomenys, surinkti informaciją

apie juos, ir taikant veiklos principus nustatyti kurie duomenų šrautai reikalingi realizacijai, o kurie nereikalingi. Naudojant šį būdą, nebūtina nagrinėti palikuoninių sistemų programinės logikos ar naudojamų duomenų bazės technologijų ir struktūros, o esant jų žymiems pakitimams užtenka atlikti minimalius pakeitimus integracijos taške [3].

Darbo tikslas - išnagrinėti ir pritaikyti duomenų integracijos metodus ir technologijas duomenų lygio informacinių sistemų integracijai bei realizuoti duomenų apie piliečių kreditus integruojančią sistemą.

Darbo aktualumas ir praktinė nauda:

- Piliečių kredito istoriją kaupianti ir analizuojanti sistema greitai ir kokybiškai suteiktų kreditoriams jų klientų finansinę riziką įvertinančią informaciją. IS praktinė nauda: geresnis rizikos valdymas - nuostolių prevencija; didesni pardavimai mažesniais kaštais - pelnas; greitas ir kokybiškas aptarnavimas - kliento pasitenkinimas.
- Praktinę vertę turi darbe pateiktas duomenų integravimo metodas, kurį galima naudoti kuriant kitas duomenis integruojančias sistemas.

Darbo struktūra

Darbo apimtis – darbą sudaro trys skyriai. Darbo apimtis yra 94 puslapiai, įskaitant 73 paveikslus ir 25 pavadinimų literatūros sąrašą.

1-ame skyriuje atlikta būsimos organizacijos veiklos analizė, apibrėžti kokybės kriterijai pagal kuriuos bus vertinama sukurta sistema, palyginta keletas skirtingų pasaulyje naudojamų duomenų integravimo sprendimų. Išnagrinėti sistemos modeliavimo įrankiai ir metodai, jų skirtumai, privalumai ir trūkumai. Palyginti XML ir EDI formatai, šių formatų transformavimo, nuskaitymo, duomenų kokybės tikrinimo įrankiai ir jų privalumai.

2-ame skyriuje pateiktas RUP proceso pagalba aprašytas duomenų apie piliečių kreditus integruojančios sistemos modelis. Apibrėžta projektuojamos sistemos techninė užduotis.

3-ame skyriuje apžvelgta sukurto duomenis integruojančios sistemos realizacija – pagrindiniai sistemos ypatumai, pagrindinių scenarijų realizacija bei eksperimentinis tyrimas, kaip sistema tenkina vartotojo reikalavimus bei apsibrėžtus kokybės kriterijus. Taip pat aprašytos tolimesnės sistemos plėtojimo galimybės.

Paskutiniuose šio darbo skyriuose pateikiama: darbo išvados, literatūros sąrašas, sutrumpinimų žodynas bei darbo santrauka anglų kalba.

1. VARTOTOJŲ POREIKIŲ ANALIZĖ

1.1. Tyrimo sritis, objektas ir problema

Darbo tyrimo sritis – duomenų apie piliečių kreditus integravimo sistema, kurios pagalba būtų surenkami ir integruojami gauti duomenys iš įvairių šaltinių apie piliečių finansinius įsipareigojimus įvairioms institucijoms.

Kuriama duomenų apie piliečių kreditus integravimo sistema pasižymės tokiomis savybėmis:

- Sistema bus suprojektuota taip, kad ateityje ją galima bus integruoti į kitas verslo valdymo sistemas;
- Sistema bus pajėgi pažangiausiais metodais įvertinti surinktus iš skirtingų šaltinių duomenis, juos išanalizuoti ir rezultatus atvaizduoti ataskaitose.
- Kad tiksliau įvertinti kliento riziką, į sistemą bus galima integruoti ir apskaitos, mokesčių mokėjimo duomenis renkamus įvairiausių institucijų ir paslaugas teikiančių įmonių.
- Sistema leis ateityje į ją įdiegti įvairias naujas paslaugas.
- Duomenų apsikeitimui su įvairiomis institucijomis sistema naudos populiariausias šių dienų technologijas.
- Vartotojo sąsaja bus patogi ne tik su sistema dirbantiems darbuotojams, bet ir klientams, kurie greitai ir kokybiškai galės susirasti juos dominančią informaciją.
- Bus įdiegtos B2B (angl. Business to business) verslas verslui idėjos, kurios klientams leis sutaupyti ne tik laiko, bet ir pinigines sąnaudas, sumažins popierizmą.

Gali kilti klausimas ar tokia sistema gali būti perspektyvi, ar jos atnešama nauda ateityje bus pelninga? Taip, specialistai teigia, kad per ateinančius 4 m. labiausiai vystysis interneto elektroninės B2B sistemos. Per šį laikotarpį tokios elektroninės sistemos turėtų duoti daugiausia pelno, kurio didžiausią dalį sudarys gauti mokesčiai už elektroninių sandorių vykdymą, verslo partnerystės tarpininkavimą ir papildomas paslaugas, kaip pvz.: elektroninės konsultacijos, paskolų išdavimo ir pan.

Tyrimo objektas - duomenų apie piliečių kreditus integravimas.

Šiuo metu Lietuvoje nėra bendros duomenų bazės, kurioje būtų kaupiami iš įvairių bankų ir kredito suteikimo įmonių surinkti duomenys apie piliečių turimus ar turėtus įsiskolinimus praeityje. Nėra bendros sistemos, kuri iš turimų duomenų galėtų analizuoti informaciją apie piliečio patikimumą ir jo finansavimo riziką. Kiekvienas bankas ar kredito suteikimo įstaiga duomenis kaupia tik savo sistemoje. Todėl norint gauti kreditą reikalinga daug bereikalingo

popierizmo. Viso to būtų galima išvengti, surinkus visus duomenis ir suintegravus juos į bendrą sistemą, kuri turėtų galimybę iš turimų duomenų daryti išvadas, analizuoti ir įvertinti kredito vartotojo mokumą, riziką, bei pasitikėjimą klientu. Norint sukurti tokią sistemą reikia atsakyti į daug įvairių klausimų. Visų pirma, tai konfidencialios informacijos apsauga, nes kiekvieno piliečio duomenys yra konfidenciali informacija, kurios bankai ir kredito suteikimo įmonės negali atskleisti tretiesiems asmenims. Kaupti duomenis apie LR piliečius yra draudžiama įstatymais. Todėl norint dalintis tokiais duomenimis, reikėtų gauti piliečio raštišką sutikimą.

Pagrindinės problemos tokios sistemos sukūrimui būtų:

- Duomenų surinkimas ir integravimas:
 - priėjimas prie tokių duomenų (saugumo reikalavimai);
 - tokių duomenų transportavimas į integruojančią sistemą;
 - duomenų iš skirtingų šaltinių suderinimas;
 - duomenų formatas – struktūra – semantika.
- Duomenų apdorojimas pagal tam tikrus kriterijus (turimų duomenų analizė).
- Duomenų pateikimas vartotojams.
- Teisinio egzistavimo LR problemos.
- Reikalingų duomenų trūkumas.

1.2. Modelio sudarymo metodo ir priemonių parinkimas

Sistemos projektavimo metodų bei priemonių pasirinkimas šiuo metu yra gana platus. Galima išskirti dvi sistemos projektavimo priemonių (CASE - angl. Computer Aided Software Engineering) rūšis – struktūrinio ir objektinio projektavimo priemonės. Kai kurias priemones galima priskirti abejoms rūšims. Objektinis projektavimas atsirado daug vėliau už struktūrinį, tačiau paskutiniu metu būtent jis dažniau naudojamas. Gerai žinomos šios objektinės analizės projektavimo priemonės – UML CASE įrankiai: Rational Rose, Magic Draw, Visio Enterprise, ProVision WorkBench, Enterprise Architect ir kiti. Dauguma jų palaiko ir senesnes struktūrinio projektavimo diagramas.

CASE priemonės leidžia atsiriboti nuo painaus programinio kodo, pirmuosius sistemos kūrimo darbus atliekant konceptuali lygmeniu, kuriame sistemos architektūros specifikacija ir projektavimo rezultatai yra daug vaizdesni, lengviau suprantami ir modifikuojami. Tai iš dalies priartina IS kūrimą prie vartotojo lygio. CASE priemonės naudojamos visame sistemos kūrimo cikle. Šiomis priemonėmis organizuojamas ir valdomas programinės įrangos kūrimas, o tai ypač naudinga dideliems projektams, kadangi padeda susisteminti IS kūrimą ir griežčiau jį kontroliuoti.

1.2.1. Vartotojo reikalavimų specifikuojimas naudojant UML

Programinės įrangos projektavimui atlikti ir dokumentacijai sukurti naudosime UML (unifikuota modeliavimo kalba). UML – tai standartinių metodų visuma skirta specifikuoti, vizualiai atvaizduoti, konstruoti bei dokumentuoti pasirinktą sistemą.

UML pasirinkimą programos modeliavimui lėmė tai, jog UML atstovauja rinkiniui, šiuo metu vienam iš geriausių, inžinerinių praktinių sprendimų, skirtų modeliuoti kompleksines sistemas. UML yra labai svarbi objektiškai orientuotų programų projektavimo dalis. Tam, kad aprašyti projektuojamos sistemos dizainą, UML naudoja grafinę notaciją. UML panaudojimas padeda projektavimo grupėms bendradarbiauti, kuriant atskiras projekto dalis. UML pateikia ir sistemos dokumentaciją, reikalingą ne tik kūrimo procese, bet ir atliekant sistemos priežiūrą pasibaigus visiems kūrimo etapams. Turint išsamų sistemos aprašymą ateityje bus galima lengviau atlikti sistemos atnaujinimą ar sistemos duomenų išgavimą.

Objektinių modeliavimo (OM) kalbų atsiradimas siejamas su 1970 metais, kuomet daugelis OM pradininkų eksperimentavo su skirtingais objektinio programavimo analizės ir dizaino metodais. Nuo 1989 iki 1994 OM kalbų skaičius pakito nuo 10 iki daugiau nei 50. Dažniausiai vartotojai nelikdavo patenkinti viena ar kita modeliavimo kalba. Pradedant nuo 1990 metų, atsiranda naujų metodų savyje jungiančių keletą ar keliolika jau ankščiau sukurtų OM. Taip susiformavo keletas populiariesnių metodų [11].

Po 1996 tapo aišku, jog UML kūrimo kryptis pasiteisina. Buvo įkurtas UML partnerių konsorciumas. Padedant ir kitoms organizacijoms buvo suvienytos jėgos kuriant UML 1.0 specifikaciją. Šiai dienai jau egzistuoja UML 2.0 specifikacija.

Kiekviena UML diagrama suformuota taip, kad leistų kūrėjams ir vartotojams peržiūrėti programinę sistemą skirtinga perspektyva ir skirtingu abstrakcijos lygiu.

UML diagramos paprastai braižomos grafiniu redaktoriumi turi:

- panaudojimo atvejų diagramas;
- klasių diagramas;
- bendradarbiavimo diagramas;
- būsenų diagramas;
- veiklos diagramas;
- komponentų diagramas;
- įrengimų diagramas.

1.2.2. UML programinio paketo reikalavimai

Kuriant sistemą yra labai svarbu ne tik projektavimo etapuose naudojamos abstrakcijos, bet ir vykdant sistemos priežiūros, atnaujinimo bei patobulinimo darbus. Pageidautina turėti tą pačią projektavimo priemonę. Todėl projektuojant sistemą racionaliau pasirinkti platesnių galimybių projektavimo paketus, kad ateityje vykdant sistemos atnaujinimą ar papildymą būtų galima tikėtis daugiau naujovių iš projektavimo įrankių paketų.

Šiuolaikiniams UML programiniams modeliavimo įrankiams keliami reikalavimai:

- pagrindinių standartinių diagramų generavimas;
- diagramų eksportas, spausdinimas;
- kodo generavimas;
- daugiau nei vienos programavimo kalbos palaikymas;
- pilna dokumentacija;
- atvirkštinės inžinerijos galimybė.

1 lentelėje pateiktas kelių UML specifikacijas palaikančių modeliavimo priemonių palyginimas.

1 lentelė. UML modeliavimo priemonių palyginimas

UML modeliavimo programa	Gamintojas	Internetinis adresas	UML diagramų palaikymas						Platforma	Programavimo kalba	Kodo generavimas	Atvirkštinė inžinerija	Paketo kaina
			Use Case	Klasės diagramos	Sekų diagramos	Būsenų diagramos	Veiklos diagramos	Diagramų savybės					
Rational Rose Enterprise	IBM	www.rational.com/rose	Y	Y	Y	Y	Y	Y	Win	C++, ADA, VB, JAVA, Corba	Y	Y	2442\$
Enterprise Architect Professional	Sparx systems	www.sparxsystems.com	Y	Y	Y	Y	Y	Y	Win	C++, Delfi, VB, JAVA	Y	Y	149\$
MagicDraw UML Professional	MagicDraw	www.magicdraw.com	Y	Y	Y	Y	Y	Y	Java VM	Java, C++, IDL	Y	Y	899\$
UML Studio	PragSoft	www.pragsoft.com	Y	Y	Y	Y	Y	Y	Win	Java, C++, IDL	Y	Y	500\$

Sistemos modeliui kurti buvo pasirinktas RUP (*angl. Rational Unified Process*) sistemos projektavimo procesas bei jį realizuoti leidžianti priemonė – programinis paketas Magic Draw (objektinio projektavimo įrankis), kuris:

- palaiko UML metodologiją;
- paprastas naudoti;
- puikiai pritaikytas išsamiai aprašyti sistemos probleminei sričiai;
- yra nemokama versija skirta naudoti studentams.

1.3. Organizacijos veiklos analizė

1.3.1. Analizės tikslas

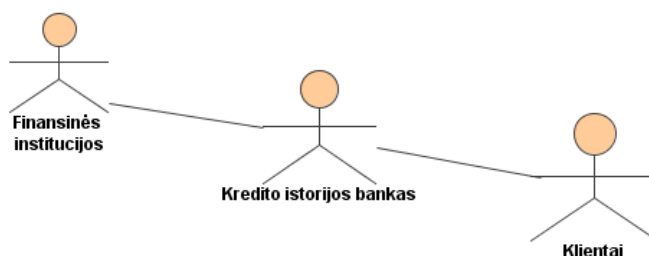
Organizacijos veiklos analizės tikslas – išsiaiškinti kuriamos informacinės sistemos veiklos sritį bei sprendžiamą problematiką, apibrėžti pagrindinius veiklos objektus, naudotojų tipus bei bendradarbiavimą tarp atskirų naudotojų tipų. Pagal suformuluotus reikalavimus vėliau bus modeliuojama sistema bei projektuojami sistemos komponentai.

1.3.2. Veiklos sąveikų modelis

Veiklos analizę tikslinga pradėti nuo veiklos sąveikų modelio sudarymo. Šis procesas susideda iš tokių žingsnių:

- išorinių veikėjų identifikavimo,
- pagrindinių veiklų identifikavimo,
- santykių tarp veiklų ir išorinių veikėjų specifikuojimo,
- veiklos sąveikų modelio nubraižymo ir aprašymo.

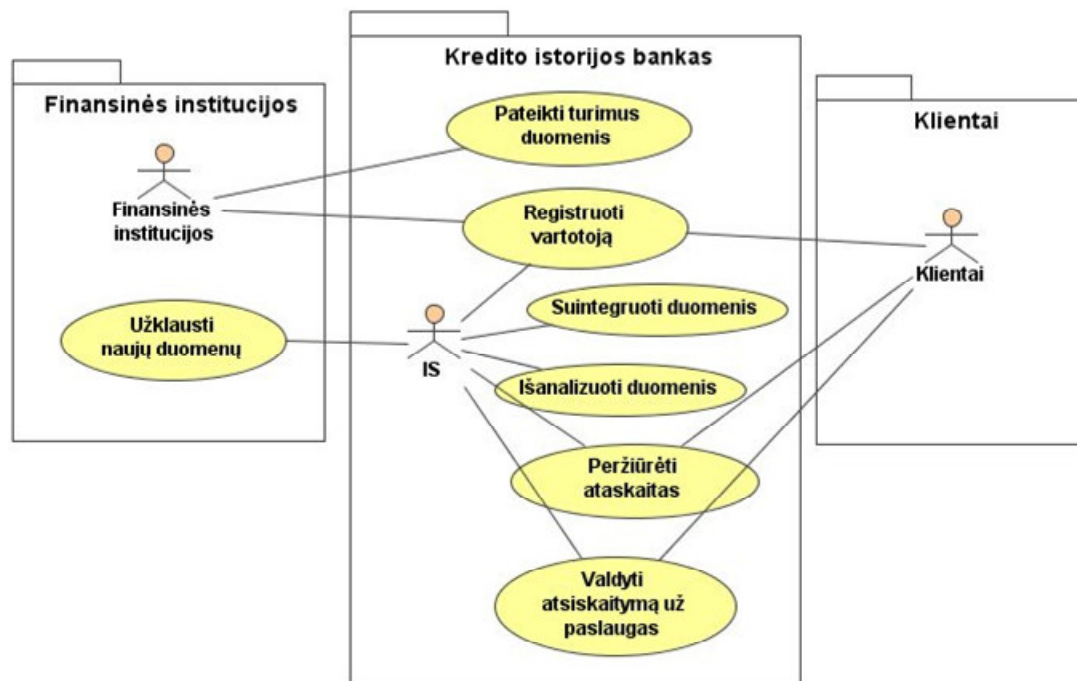
Veiklos sąveikos modelis atspindi ne kompiuterinės sistemos panaudojimo atvejus, bet analizuojamoje organizacijoje vykdomus veiklos procesus. Sudarome tris veiklos panaudojimo atvejų diagramas: aukščiausio lygio veiklos kontekstinę diagramą (žr. 1 pav.), kuri vaizduoja veiklos ryšius su išoriniais aktoariais, detalesnę kontekstinę diagramą (žr. 2 pav.), vaizduojančią organizacijos veiklos procesus, kuriuose dalyvauja išoriniai aktoriai – asmenys, organizacijos, kitos sistemos, bei detalizuotą veiklos kontekstinę diagramą (žr. 3 pav.), kuri atspindi ir vidinę organizacijos vykdomą veiklą.



1 pav. Aukščiausio lygio veiklos kontekstinė diagrama (veiklos sąveikų modelis)

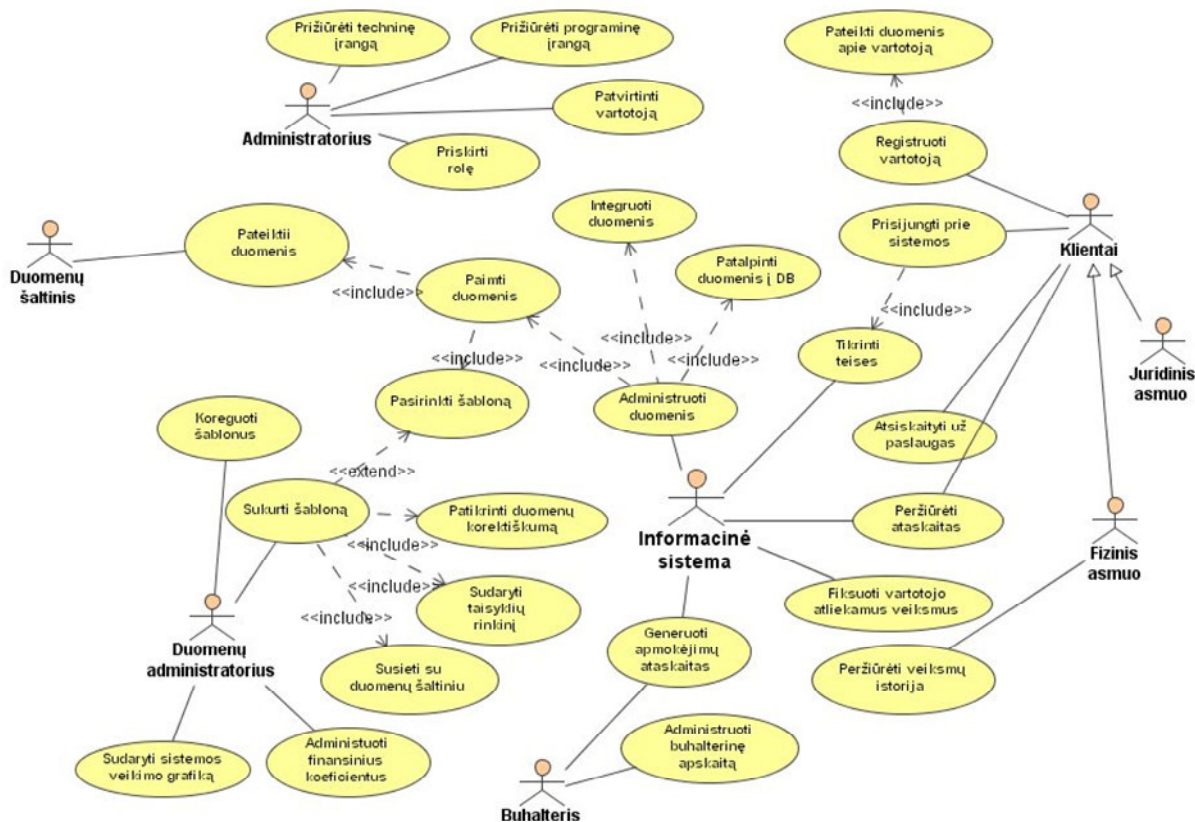
Aukščiausio lygio veiklos kontekstinėje diagramoje identifikuojami tiriamos veiklos aktoriai – finansinės institucijos, kredito istorijos bankas ir klientai. Finansinės institucijos - Lietuvos Respublikos įmonė arba įstaiga arba užsienio valstybės įmonės padalinys, veikiantis Lietuvoje ir besiverčiantis bent vienos finansinės paslaugos teikimu[16]. Šiame darbe bankai, kreditą teikiančios įmonės, Sodra, komunalines paslaugas teikiančios įmonės bus

vadinamos bendru vardu – finansų institucijomis arba duomenų tiekėjais. Iš aukščiausio lygio kontekstinės diagramos matome, kad mūsų kuriama sistema yra kaip tarpininkavimo sistema tarp finansines operacijas atliekančių subjektų ir paskolas teikiančių institucijų.



2 pav. Detalesnė veiklos kontekstinė diagrama

Detalesnėje veiklos kontekstinėje diagramoje pateiktos veiklos aktorių bendrai vykdomos veiklos. Visa grandinė prasideda nuo to, kad pirmiausia išorinis aktorius inicijuoja veiklą „Užregistruoti vartotoją“. Tokiu būdu jis kaip duomenų šaltinis užregistruojamas sistemoje. Prasideda nuolatinis duomenų apsisikeitimas iš įvairių išorinių duomenų šaltinių. Sistema pati valdo visas vidines grandis t.y. duomenų surinkimą, integraciją, duomenų patalpinimą į DB. Bet tuo ši grandinė nesibaigia, nes kitoje grandinės pusėje stovi „Klientai“. Jie visų pirmiausia turi užsiregistruoti sistemoje, o tik tada gali naudotis sistemos teikiamomis paslaugomis. Kadangi paslaugos klientams yra mokamos, tai sistemoje bus piniginių atsiskaitymų modulis, kuris padės klientams greitai ir patogiai atsiskaityti internetu ar bankiniu pavedimu už suteiktas paslaugas..



3 pav. Detalizuota veiklos kontekstinė diagrama

Detalizuotoje veiklos kontekstinėje diagramoje įvardinami vidiniai veiklos darbuotojai bei pagrindinės jų inicijuojamos veiklos. Administratorius dalyvauja tik keliuose organizacijos veiklose, kurios yra labai svarbiose būsimos sistemos įdiegimo ir eksploatacijos fazėse. Nuolatinė techninės ir programinės įrangos priežiūra garantuoja, kad sistema bet kuriuo metu bus prieinama jos naudotojams. Administratorius taip pat atsakingas už fizinių ir juridinių asmenų patvirtinimą patikrinus jų pateiktų duomenų teisingumą (reikalinga tam, kad būtų registruojami tik realiai egzistuojantys subjektai).

Detalizuotoje veiklos kontekstinėje diagramoje taip pat atsiranda tokie veikėjai kaip duomenų administratorius ir buhalterija. Duomenų administratorius valdo visus su duomenimis susijusius procesus, pradedant nuo naujų duomenų formato ir sintaksės peržiūros, duomenų teisingumo analizavimo, duomenų integracijos taisyklių sukūrimo iki naujų finansinių informacijos įvertinimų/koeficientų sudarymo ir koregavimo operacijų. Buhalterija, kaip ir kiekvienoje įmonėje valdo ir prižiūri apskaitos ir atsiskaitymų ūkį. Šiame darbe ši veikla yra šalutinė ir detaliau ji nebus nagrinėjama.

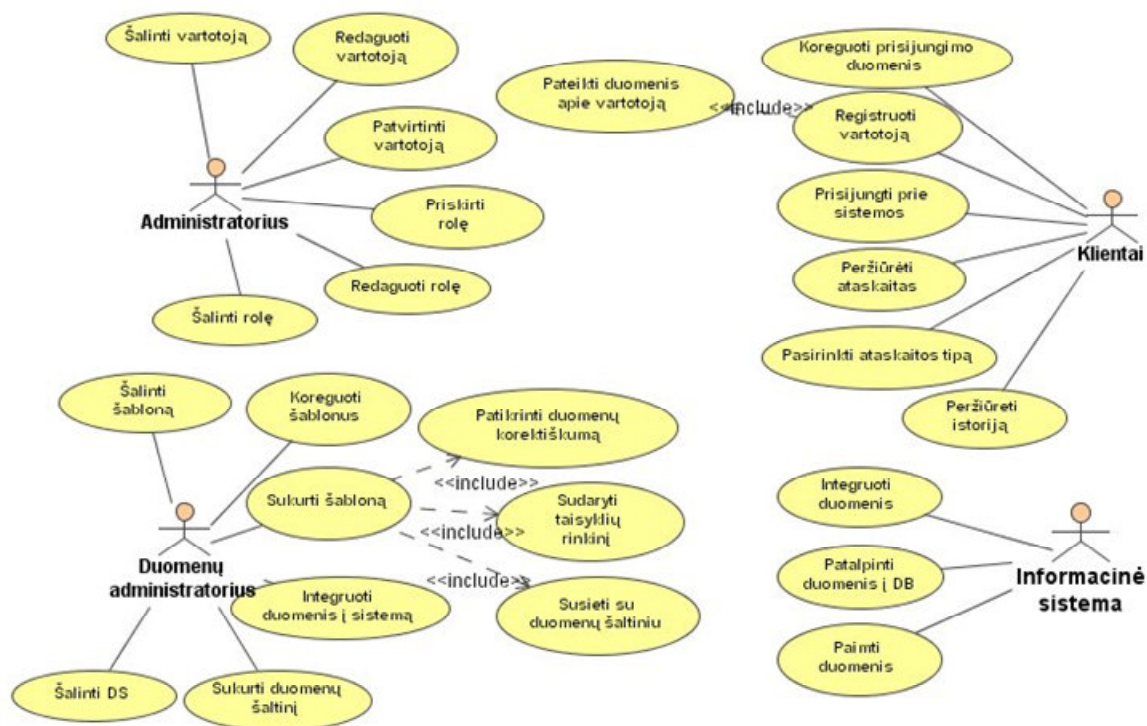
Pateiktoje diagramoje matomi ir analizuojami sistemos išoriniai vartotojai t.y. fiziniai ir juridiniai asmenys. Kiekviena klientų klasė priklausys atskiroms klientų grupėms. Tokiu būdu jie

galės matyti tik jiems naudingą informaciją, daryti tik jiems skirtus veiksmus: juridinis asmuo galės gauti informaciją susijusią su fizinių asmenų kreditavimo rizika, o fizinis asmuo galės gauti informaciją apie tai, kas domėjosi jo kredito istorijos informacija.

Finansų institucijos kaupia duomenis apie savo klientus, jų atliekamas finansines operacijas savo vidinėse informacijos saugyklose. Todėl kuriamos informacinės sistemos tikslas yra, suderinti visą procesą nuo duomenų gavybos, integracijos į bendrą sistemą valdymo, iki užklausų priėmimo, turimų duomenų analizės pažangiausiai analizavimo metodais atlikimo ir ataskaitų generavimo. Galutinis produktas – informacijos apie piliečius surinkimas iš skirtingų (įdiegus papildomą funkcionalumą ir iš heterogeninių) šaltinių surinkimas, kreditingumo rizikos nustatymas ir pateikimas galutiniam vartotojui.

1.3.3. Veiklos panaudojimo atvejų modelis

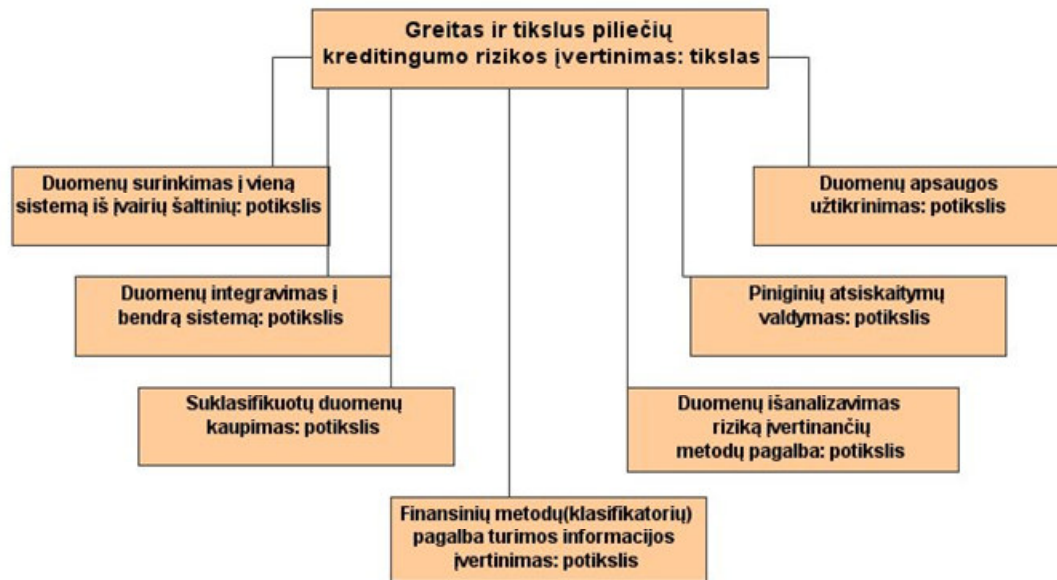
Veiklos panaudojimo atvejų modelyje (žr. 4 pav.) išskirtos veiklos iš veiklos sąveikų modelio (žr. 3 pav.), kurios bus kompiuterizuojamos.



4 pav. Veiklos panaudojimo atvejų modelis

1.3.4. Veiklos tikslų modelis

Veiklos tikslų modelyje (žr. 5 pav.) pateikiamas tiriamos veiklos aukštesnio lygio tikslas ir šiam tikslui pasiekti tarnaujantys žemesnio lygio potiksliai, kuriuos padės įgyvendinti kuriama sistema.

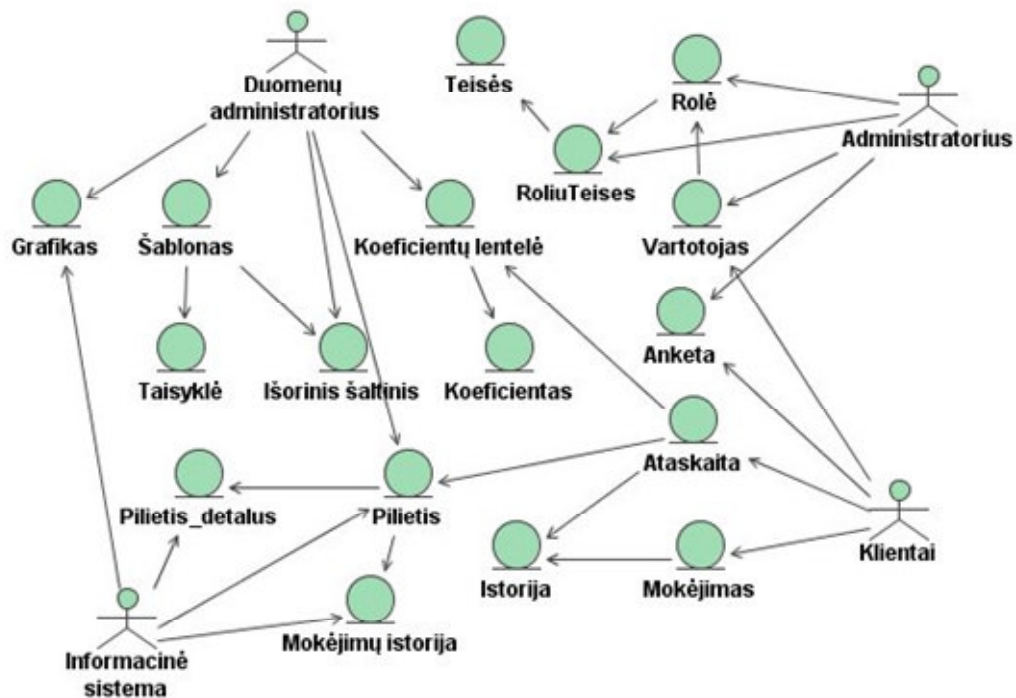


5 pav. Veiklos tikslų modelis

Pagrindinis nagrinėjamos sistemos veiklos tikslas – greitas ir tikslus piliečių kreditingumo rizikos įvertinimas. Šiai dienai kiekviena kreditus ir paskolas teikianti finansinė institucija tiesiogiai bendrauja su galutiniu vartotoju. Todėl sugaištama labai daug laiko. Norint įvertinti vartotojo mokumą, reikalinga daug papildomo popierizmo, žmogiškųjų resursų ir t.t. O didėjant teikiamų kreditų portfeliui ir esant Lietuvoje dideliame žmonių migravimui finansinėms institucijoms atsiranda vis didesnė rizika pinigus investuojant į paskolų išdavimą. Kita akivaizdi nauda tokios tarpininkavimo sistemos atsiradimui, tai padidėtų konkurencija tarp paskolas išduodančių finansinių institucijų, to pasekoje patikimi klientai paskolas gautų geresnėmis sąlygomis. Todėl, norint įgyvendinti pagrindinį kuriamos sistemos tikslą, reikia įvertinti ir įgyvendinti to tikslo potikslis:

- duomenų surinkimas į vieną sistemą iš įvairių šaltinių;
- duomenų integravimas į bendrą sistemą;
- suklasifikuotų duomenų kaupimas;
- finansinių metodų (klasifikatorių) pagalba turimos informacijos įvertinimas;
- duomenų apsaugos užtikrinimas;
- piniginių atsiskaitymų valdymas;
- duomenų išanalizavimas riziką įvertinančių metodų pagalba.

1.3.5. Veiklos objektų modelis

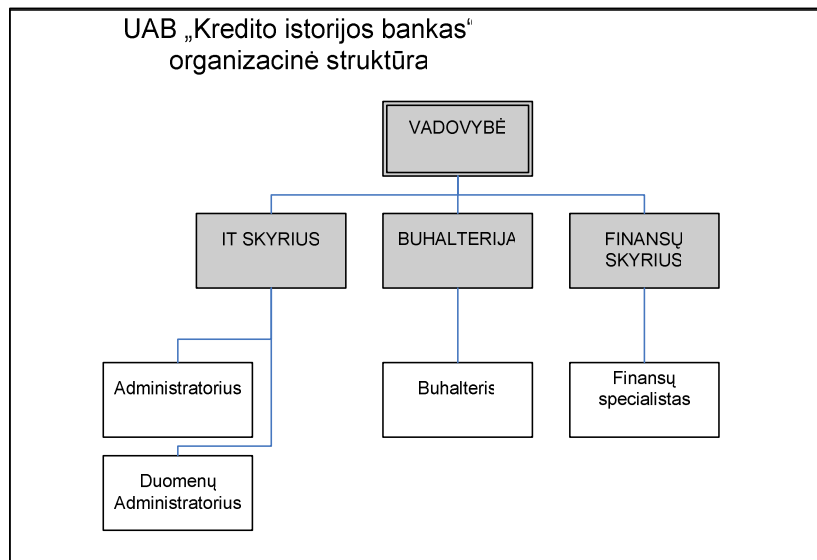


6 pav. Veiklos objektų modelis

Veiklos objektų modelyje (žr. 6 pav.) pavaizduoti su kiekviena veiklos esybe susiję veiklos darbuotojai ir veiklos aktoriai. Administratorius suteikia darbuotojams ir klientams prisijungimus prie sistemos bei patvirtina naujų vartotojų registraciją, prieš tai patikrinęs apie tokio kliento egzistavimą. Klientai prisijungę prie sistemos gali redaguoti savo duomenis, vykdyti paiešką ir generuoti ataskaitas, kurias gali atsispausdinti arba išsisaugoti savo kataloge. Klientai taip pat vykdo atsiskaitymo procesą, nes už kiekvieną suteiktą paslaugą reikia apmokėti pagal nustatytas kainas. Informacinė sistema pati kaupia duomenis apie piliečius, juos nuolat papildydama jau turimą informaciją DB t.y. sukuria naujus mokėjimo istorijos įrašus, redaguoja piliečio informaciją ir t.t. Duomenų administratoriaus funkcijos yra kurti naujus informacijos integravimo šablonus ir juos redaguoti arba šalinti. Duomenų administratorius atlieka sukūrimo, redagavimo ir naikinimo funkcijas objektui „pilietis“ ir su juo susijusiems objektams. Duomenų administratorius sukuria redaguoja ir šalina išorinių duomenų šaltinių informaciją. Jis sudaro ir sistemos veikimo grafiką, pagal kurį sistema automatiškai bendrauja su išoriniais duomenų šaltiniais.

1.3.6. Organizacinės struktūros modelis

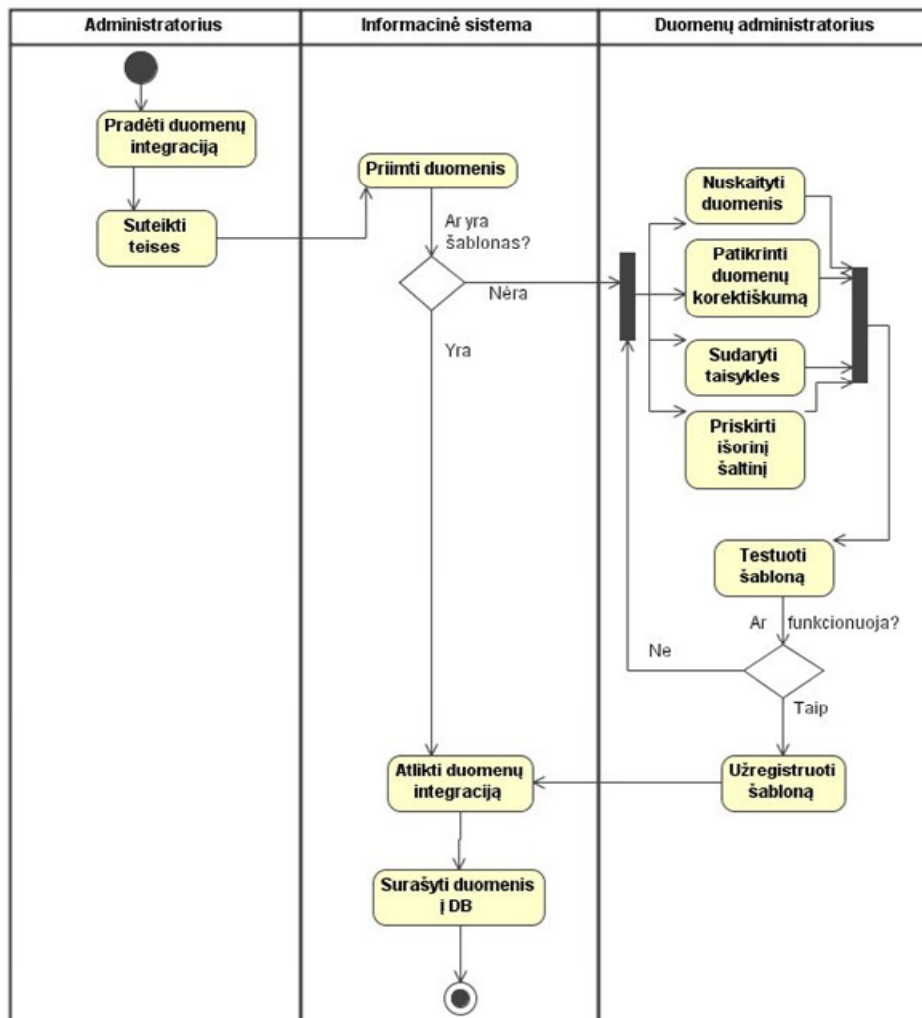
Organizacinės struktūros modelyje (žr. 7 pav.) nurodomi veiklos darbuotojai bei jų priklausomybės padaliniams. Organizacijos struktūra yra standartinė ir plačiai paplitusi tarp Lietuvos įmonių.



7 pav. Organizacinės struktūros modelis

1.3.7. Veiklos procesų modelis

8 paveiksle vaizduojamas veiklos proceso modelis, aprašantis informacinės sistemos duomenų integracijos procesą iš skirtingų duomenų šaltinių į bendrą sistemoje naudojamą duomenų formatą. Integruojančios sistemos veiklos darbuotojas administratorius inicijuoja pirminę integraciją ir suteikia visas reikiamas teises, kad sistema galėtų priimti duomenis. Gavus duomenis, sistema tikrina ar tiems duomenims yra sukurtas šablonas, pagal kurį ji galėtų apdoroti tuos duomenis ir juos suintegruoti į bendrą duomenų formatą. Jei šablono nėra, sistema pateikia užklausimą duomenų administratoriui, kad jis sukurtų naują šabloną gautiems duomenims. Savo ruožtu duomenų administratorius nuskaito gautus duomenis į administravimo programą, patikrina jų korektiškumą, priskiria išorinį duomenų šaltinį ir sudaro transformacijos taisykles, pagal kurias sistema sekantį kartą galės duomenis suintegruoti į DB. Padaręs šabloną, duomenų administratorius jį pratestuoja ir jei šablonas neveikia, jis yra vėl iš naujo koreguojamas. Veikiantis šablonas yra užregistruojamas DB, kad sekantį kartą, be administratoriaus įsikišimo, sistema galėtų automatiškai integruoti duomenis.



8 pav. Veiklos procesų modelis

1.3.8. Vartotojų bei jų poreikių analizė (reikalavimai sistemos funkcionalumui)

Pagrindiniai sistemai keliami funkciniai reikalavimai:

- sistema turi palaikyti tokių tipų vartotojus:
 - klientai (juridiniai asmenys) – įvairios įmonės;
 - klientai (fiziniai asmenys) – Lietuvos respublikos piliečiai;
 - duomenų administratorius – duomenis tvarkantis ir koordinuojantis vartotojas;
 - administratorius, palaikantis sistemos veikimą bei suteikiantis prisijungimo teises prie sistemos visų trijų tipų naudotojams;
- klientų registravimosi prie sistemos valdymas;
- klientų duomenų registro palaikymas ir atnaujinimas;
- piliečių kredito istorijos duomenų atnaujinimas ir paieškos valdymas;
- išorinių duomenų integracijos valdymas.

1.3.9. Informacijos sistemai keliami nefunkciniai reikalavimai ir apribojimai

Pagrindiniai sistemai keliami nefunkciniai reikalavimai:

- programinė įranga turi užtikrinti, kad bendradarbiavimas ir duomenys sistemoje būtų saugūs ir neprieinami pašaliniam naudotojams;
- sistema turi būti sukurta naudojant šias programines priemones: ASP.NET, Microsoft SQL SERVER 2005;
- reikalavimai naujos IS duomenų bazės tarnybinei stočiai: Operacinė sistema Microsoft Windows 2003 Server, įdiegtas ir funkcionuojantis TCP/IP tinklo protokolas, Microsoft SQL SERVER 2005, IIS;
- sistemos kūrimui reikalingas VisualStudio.NET programinis paketas.

1.4. Projekto tikslas ir jo pagrindimas, kokybės kriterijų apibrėžimas

Projektas kuriamas norint užtikrinti efektyvesnį verslo subjektų (juridinių ir fizinių asmenų) tarpusavio bendradarbiavimą. Norima suderinti, struktūrizuoti, optimizuoti bei integruoti finansines operacijas atliekančių ir finansines paslaugas teikiančių įmonių operatyvinius duomenis. Juos išanalizavus pateikti finansavimo riziką įvertinančias ataskaitas ir pastabas.

Projekto tikslas yra sukurti tarpininkavimo sistemą, kuri surinktų ir suintegruotų finansinių institucijų duomenis į bendrą vieningą formatą; esant klientų užklausimui – padarytų finansinę analizę; ataskaitas pateiktų galutiniam informacijos vartotojui.

Pagrindinis sukurtos sistemos kokybės įvertinimo aspektas yra jos visų numatytų funkcinių reikalavimų tenkinimas. Tačiau, kad sistema būtų tikrai kokybiška, neužtenka vertinti tik pilną funkcionalumo realizavimą – būtina atsižvelgti ir į nefunkcinių reikalavimų įgyvendinimą, todėl būtina apibrėžti tokios sistemos kokybės vertinimo kriterijus:

- *Patogumas* – sistemos naudotojo sąsaja:
 - patogi naudoti, intuityviai suprantama ir patraukli vartotojui;
 - kuo mažiau laukų, kuriuos reikia pildyti ranka;
 - sistemos įspėjimai ar pranešimai apie klaidas ekrane turi būti aiškūs bei suprantami;
 - naudojami dalykinės srities terminai;
 - sutrumpinimų panaudojimas, jei to reikia;
- *Saugumas*. Turi būti įvertinti trys saugumo aspektai: konfidencialumas - sistemoje esantys duomenys turi būti apsaugoti nuo neteisėtos prieigos: tik sistemoje registruoti naudotojai gali atlikti tam tikras funkcijas pagal jų rolėms suteiktas teises; naujų

naudotojų registravimas ir patvirtinimas negali būti prieinamas visiems. Registracijos duomenys bus patikrinami siekiant apsisaugoti nuo pašalinių naudotojų. Vientisumas – sistemos duomenys vienareikšmiškai atitinka šaltinio perduotus (iš jo gautus) duomenis, kartu užtikrinant jų panaudojimo teisėtumą. Pasiekiamumas - duomenų bazė ir pati sistema turi būti pasiekiami bet kuriuo momentu.

- *Sąveikos su kitomis sistemomis* – sistema turi turėti galimybę eksportuoti duomenis.
- *Efektyvumas* – turi būti užtikrinta sistemos greitimeika ir optimalus išteklių naudojimas.
- *Patikimumas* – sistema turi būti prieinama per internetą bet kuriuo paros metu, ir turėti bent jau minimalų galimų klaidų įvertinimo mechanizmą.
- *Lankstumas* – būsima sistema turi būti suprojektuota taip, kad ateityje būtų galima lengvai ir paprastai pridėti papildomus paslaugų modulius. Turi būti sukurtas universalus vartotojų modelis, kad neapsiriboti tik konkrečia vartotojų sritimi.

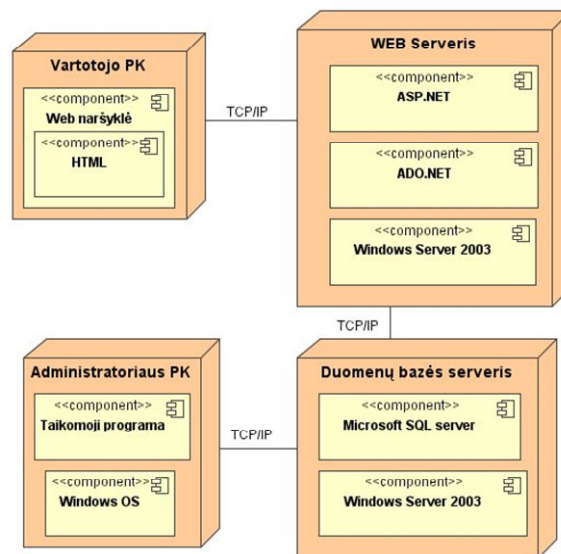
1.5. Architektūros ir galimų įgyvendinimo priemonių variantų analizė

Šioje dalyje apžvelgiami architektūriniai ir programavimo priemonių sprendimai, atliekamas jų palyginimas ir pasirenkamas tinkamiausias variantas mūsų realizuojamai informacinei sistemai.

1.5.1. Architektūros analizė

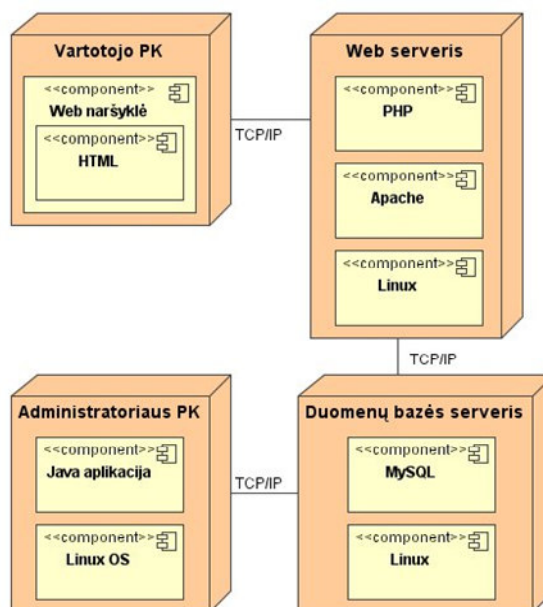
Realizuojant informacinę sistemą, numatomi du galimi architektūriniai sprendimai. Pirmas paremtas technologijomis veikiančiomis „Windows“ OS pagrindu, antras „Linux“ pagrindu.

Naudojant firmos „Microsoft“ technologijas ir „Windows“ operacinę sistemą, architektūrinis sprendimas pavaizduotas 9 paveiksle:



9 pav. Pirmojo galimo architektūrinio sprendimo UML realizacijos diagrama

Antrasis architektūrinis sprendimas pavaizduotas 10 paveiksle – tai nemokamos atvirojo kodo technologijos ir „Linux“ operacinė sistema.



10 pav. Antrojo galimo architektūrinio sprendimo UML realizacijos diagrama

1.5.2. Technologijų ir architektūros palyginimas

Architektūriniuose sprendimuose naudojamų technologijų palyginimas pateiktas 2 lentelėje:

2 lentelė. Architektūriniuose sprendimuose naudojamų technologijų palyginimas

Pasirinkimo kriterijai	Architektūrinis sprendimas 1 (OS „Windows“ pagrindu)	Architektūrinis sprendimas 2 (OS „Linux“ pagrindu)
DBVS valdymas	Lengvas	Sudėtingas
DBVS transakcijų palaikymas	Palaiko	Nepalaiko
WEB serverio galimybė veikti kitose OS	Ribota	Neribota
DBVS serverio galimybė veikti kitose OS	Ribota	Neribota
Integruotas ataskaitų generavimo įrankis	Yra	Nėra
Programinės įrangos klaidų ištaisymo greitis	Greitai	Lėtai
Internetinės vartotojo sąsajos kūrimo sudėtingumas	Nesudėtingas	Sudėtingas
Kaina	Mokama	Nemokama

Atlikus technologijų palyginimą bus naudojamos firmos „Microsoft“ technologijos ir „Windows“ operacinė sistema. Nepaisant to, kad šios technologijos yra mokomos, toks architektūros sprendimas ir technologijų pasirinkimas pasirinktas, nes:

- reikalinga DBVS palaikanti transakcijos;
- lengva ir patogiu ateityje pridėti papildomus programinius modulius;
- geras visų produktų suderinamumas ir atnaujinimas;
- reikalingas patogus įrankis ataskaitų generavimui;
- greitas ir patogus internetinės sąsajos kūrimas;
- aukštas DBVS palaikomas saugumas.

1.5.3. Programavimo įrankių analizė

Modeliuojama sistema turi suteikti galimybę pasiekti duomenis per internetą naudojantis standartinėmis priemonėmis (internetu naršykle). Kuriama sistema susideda iš trijų dalių: duomenų bazės, internetinės sąsajos (leidžiančios vartotojui iš nutolusio PK naudotis ir manipuluoti duomenų bazėje esančiais duomenimis) ir administravimo programos. Vartotojo sąsajų realizavimui galime naudoti įvairias programavimo kalbas (*C#, C++, Java, VB, PHP, ASP.NET*). 3 lentelėje palyginsime keletą galimų programavimo įrankių pasirinkimo variantų.

3 lentelė. ASP.NET ir PHP produktų lyginamoji analizė

Pasirinkimo kriterijai	PHP	ASP.NET
Programavimo kalbos	Viena kalba	Daugelio kalbų palaikymas: Java, C#, Visual Basic ir kt.
Objektinis programavimas	Nėra pilnai išvystytas	Yra pilnai išvystytas
Saugumas	Aukštas	Aukštas
Greitis	Didelis	Didelis
Atskirti grafinės sąsajos ir funkcijų kodai	Nėra	Yra
Programavimo klaidų ieškojimo sistema (angl. „debug“)	Nėra	Yra
Kompiliavimas	Nėra	Yra
Pagalbos (angl. Help) būvimas	Pakankama, reikalaujanti laiko sąnaudų	Gera, greitai prieinama
Platforma	Bet kokia	Tik Win32 (būtina IIS)
Licenzija	Nereikalinga	Reikalinga (IIS, VisualStudio.NET paketui)

Atlikus ASP.NET (Microsoft) ir PHP (The PHP Group) produktų lyginamąją analizę pasirinktas ASP.NET produktas, kadangi ASP.NET technologija yra labiau išvystyta – pilnai realizuotas objektinis programavimas, efektyvi klaidų ieškojimo sistema ir kompiliavimas (galimas sistemos vykdymas žingsnis po žingsnio), kelių lygių saugumo užtikrinimas. Kuriamos sistemos vartotojo sąsajų realizavimui pasirinkta Visual Basic (VB) programavimo kalba.

1.6. Esamų sprendimų duomenų integravimo problemai spręsti analizė

1.6.1. Standartiniai duomenų bazių įrankiai duomenų importavimui ir eksportavimui

Šiai dienai yra sukurta daug įvairių būdų ir metodų egzistuojančiai duomenų transformavimo problemai išspręsti t.y. efektyviam automatizuotam duomenų konvertavimui į ir iš reliacinių duomenų bazių. Šiai problemai išspręsti dažniausiai yra naudojama XML standarto struktūra ir joje esančių žymių naudojimas aprašant duomenis, ar struktūrizuojant dokumento duomenų struktūras. Duomenų bazių kūrėjai, tokie kaip IBM, Microsoft, Oracle ir Sybase yra išvystę programinius įrankius padedančius konvertuoti XML dokumentus į reliacines lenteles. Štai keletas siūlomų sprendimų [9]:

- Oracle XML SQL įrankis modeliuoja XML dokumentų elementus kaip lizdo tipo lentelių kolekciją. O pridėtas elementas yra modeliuojamas kaip objekto duomenų tipas. „SQL į XML“ konvertavimas sudaro XML dokumentą naudojant „vienas su vienu“ ryšį tarp lentelių, nurodant objekto duomenų tipą ir lizdo tipo elementą. „XML į SQL“ reikalauja arba duomenų modelio redagavimo (konvertuojant jį iš reliacinio į objektiškai orientuotą) arba pertvarkymo originalaus XML dokumento.
- IBM DB2 XML plėtinys leidžia talpinti XML dokumentus kaip „Blob“ tipo objektus arba kaip suskaidytus į lenteles. Naujausia transformacija, žinoma kaip XML kolekcija, apibrėžta XML 1.0 sintaksės.
- Microsoft kompanija duomenų konvertavimo problemą sprendžia praplėsdami SQL galimybes ir pateikdami OPENXML eilučių seką.
- Sybase Adaptive Server įveda „ResultSetXml“ Java klasę kaip abiejų krypčių XML konvertavimo pagrindą.

XML dokumentų konvertavimas natūraliai yra simetrinis. XML talpinimas neleidžia modifikuoti daugiau kaip vieną lentelę. XML išgavimas transformuoja SQL užklausos rezultatus į plokščios struktūros dokumentą (angl. - flat file).

Duomenų bazių kūrėjų palyginimas pateikiamas 4 lentelėje.

4 lentelė. Duomenų bazių kūrėjų palyginimas[9]

Kūrėjas	Konvertavimo taisyklės	Viena lentelė / daug lentelių	Konvertavimo priemonės	Simetrinis konvertavimas
Oracle	Besąlygiškai; sudarant objektų – reliacinių duomenų modelį	Daug	Nustatytos Java klasės	Simetrinis, jei XML dokumentas ir objektų – reliacinis modelis derinasi
IBM	Duomenų	Daug	Suprojektuotos	Simetrinis

Kūrėjas	Konvertavimo taisyklės	Viena lentelė / daug lentelių	Konvertavimo priemonės	Simetrinis konvertavimas
	priėjimo apibrėžimo failas		talpinimo procedūros	
Microsoft	SQL išplėtimas; eilučių kolekcijos funkcijos	Daug išgaunant XML Viena talpinant XML	Naudojant SQL konstruktorių FOR XML ir eilučių kolekciją OPENXML	Asimetrinis
Sybase	DTD	Viena; užklausa gali apimti kelias lenteles	Naudojant Java klases	Simetrinis

Bendros savybės:

- Visos priemonės importuoja ar eksportuoja duomenis į ar iš duomenų bazės, tačiau neturi galimybių tai atlikti pagal tam tikrą logiką, esant poreikiui transformavimo metu atlikti tam tikrus veiksmus ir pagal tai spręsti kokius duomenis naudoti;
- Šios priemonės netinka esant poreikiui sujungti kelių skirtingų technologijų duomenų bazių duomenis;
- Nėra bendrų taisyklių konvertuojant duomenų bazes į XML; specialus žymėjimas reikalingas, jei XML dokumentas naudoja naują sintaksę;
- XML transformavimas į duomenų bazę reikalauja duomenų paruošimo, pvz., skaičių/datos performavimo atitinkamiems reikalavimams.

1.6.2. Kompanijos IBM įrankis duomenų transformavimui

„XML Data Mediator“ – tai įrankis abipusiam duomenų konvertavimui tarp struktūrinių duomenų formatų, tokių kaip reliacinės duomenų bazės ir XML formato dokumentų. Šis įrankis, anksčiau vadintas XML integratoriumi (angl. „XML Integrator“), turi taisykles duomenų atvaizdavimui tarp XML ir reliacinių duomenų bazių. Jo pagalba paprasčiausiai programiniu kodu yra aprašomas XML dokumentų ir reliacinių duomenų konstrukcijų ryšys. „XML Data Mediator“ gali būti naudojamas kaip atskiras produktas arba gali būti integruotas kaip biblioteka į kitus programinius paketus [8].

„XML Data Mediator“ pagalba galima lengviau atvaizduoti XML ir kitų duomenų struktūrų kūrimą, leidžiant deklaratyvias specifikacijas ryšių tarp XML struktūrų (tokių kaip elementai ar atributai) nurodymui ir reliacinių schemų elementų (tokių kaip eilutės ar stulpeliai) aprašymui. Kai toks atvaizdavimas yra specifikuotas schemas lygyje, „XML Data Mediator“ gali automatiškai konvertuoti duomenis į ir iš XML per „Data to XML“ ir „XML to Data“ komponentus. „Data to XML“ komponentai gali surinkti duomenis iš vienos ar kelių duomenų

saugyklų ir sutvarkyti juos į nuoseklų XML dokumentą, aprašytą specifikuotoje schemeje. „XML to Data“ komponentai išrenka reikalingas apdorojamo XML dokumento dalis ir patalpina juos į vieną ar į kelias duomenų saugyklas pagal duomenų keitimo komandas (įterpti, ištrinti, pakeisti ar kitas).

„XML Data Mediator“ taip pat yra suprojektuotas priimti įeinantį duomenų srautą kaip šabloną transformacijų atlikimui. Priklausomai nuo įeinamo srauto ar transformavimo krypties, iš duomenų saugyklos (DS) į XML ar iš XML į DS, IBM kompanijos įrankis sukuria atitinkamą medžio struktūrą transformacijų vykdymui. Medžio struktūra charakterizuoja ir DS vykdomas procedūras, ir XML struktūros informaciją, kuri yra arba XML hierarchija, kuri bus sukurta vykdant duomenų ištraukimą iš DS, arba bus atitinkamos DS hierarchijos atitikmuo. Vykdamas duomenų paieškos procesą DS, šis įrankis sukuria atvaizdavimą, vadinamą „žymių medžiu“ – struktūrą, kuri aprašo hierarchiją XML elementų. Atvirkštiniame procese sukuriama medis, kuris aprašo atitinkamas XPath išraiškas, naudojamas reikalingų elementų išrinkimui iš skaitomo XML dokumento ir duomenų talpinimo procedūras korektiškam duomenų surašymui į DS [8].

1.6.3. Duomenų sandėlis

Duomenų sandėlio apibrėžimas

Duomenų sandėlį pirmą kartą apibrėžė Bill Inmon 1990 metais. Apibrėžimas skambėjo taip: „Duomenų sandėlis yra orientuota į objektą, integruota, laikui bėgant besipildančių, bet nekintamų duomenų sąranka, kuri yra skirta palengvinti valdymo sprendimų priėmimą“ (Bill Inmon, 1990: 8).

Duomenų sandėlio pagrindinės atpažinimo savybės:

- **Orientacija į objektą**

Pagrindinis duomenų sandėlio tikslas yra saugoti informaciją, kuri reikalinga strategiškai svarbių įmonės sprendimų priėmimui. Savaimė suprantama, kad kiekvienoje kompanijoje renkami duomenys apie kažkokį tai objektą, o iš čia ir išplaukia, kad informacija renkama būtent apie tuos dominančius objektus, o ne apie kokias nors programas ar procesus [5].

- **Integralumas**

Nors duomenys duomenų sandėlyje išmėtyti po skirtingas lenteles, duomenų bazes ar net serverius, tačiau duomenys yra nuosekliai apjungti bendrų kintamųjų reikšmėmis, įvardinant tiek formalų tiek ir fizinį duomenų aprašymą [5].

- **Nekintamumas**

Duomenys būdami duomenų sandėlyje atspindi operacinės duomenų bazės būseną tam tikru laiko momentu ir tas vaizdas nebegali būti keičiamas nuo to momento, kai jis čia pakraunamas iš operatyvių duomenų sistemos. Kadangi šis atvaizdas atspindi operacinius

duomenis duotu laiko momentu taigi savaime aišku, kad duomenų sandėlis saugo ir tiksliai kintamųjų reikšmes buvusias tuo momentu [5]. Egzistuoja tik dvi operacijos – laiku paremtas duomenų pakrovimas ir priėjimas prie pakrautųjų duomenų. Operatyvių duomenų reikšmės keičiasi bėgant laikui. Laiku paremtas duomenų iš operatyvių sistemų archyvavimas duomenų sandėlyje ir suteikia ypatingai didelę vertę duomenims [13].

Duomenų sandėliai pasižymi tuo, kad automatiškai užpildo duomenų sandėlius ir duomenų muges gaunama informacija ir paprastina duomenų sandėlių kūrimo, vykdymo ir palaikymo procesus. Juose yra integruoti įrankiai duomenų išgavimui, transformavimui, užkrovimui, darbų planavimui, sandėlio valdymui ir kontrolei, vartotojų užklausoms ir ataskaitoms.

Pagrindiniai tokių duomenų sandėlių privalumai:

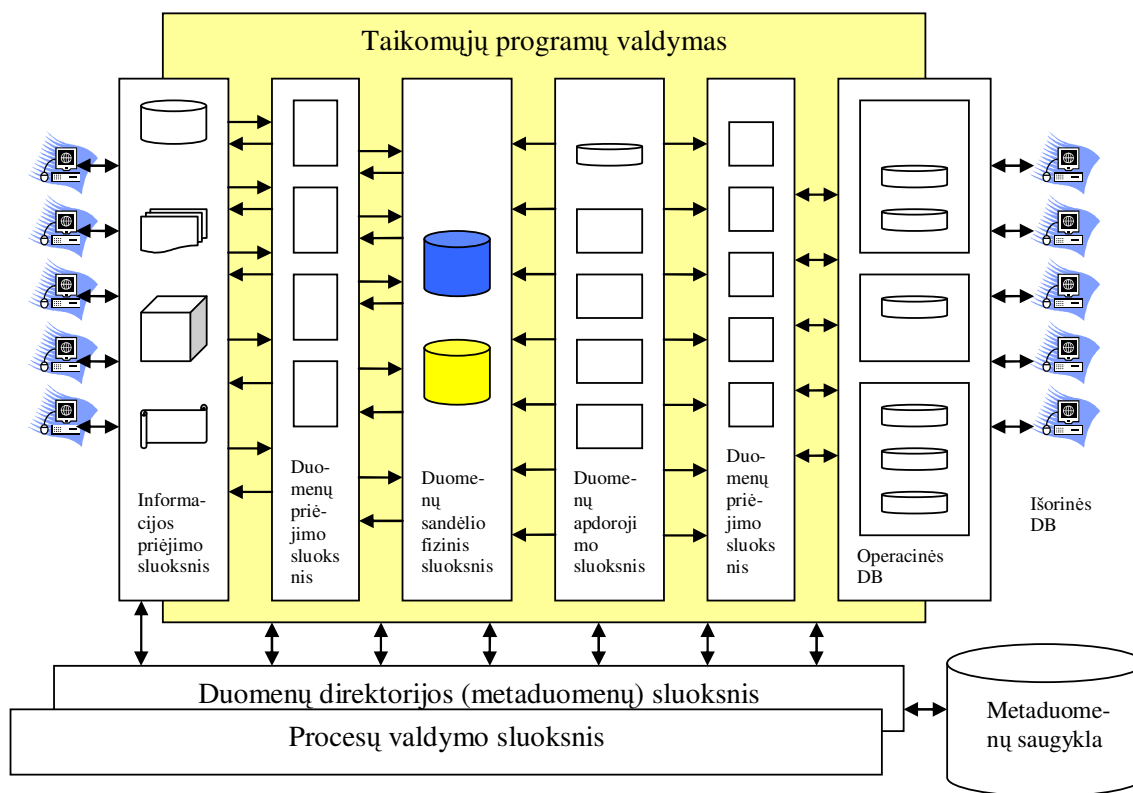
- Suteikia galimybę verslo klientams turėti pačią tiksliausią informaciją, todėl jie gali priiminėti teisingus sprendimus turėdami reikiamus duomenis reikiamu laiku ir reikiamoje vietoje.
- Suteikia administratoriui vieną kontrolės tašką valdyti, palaikyti ir suprasti centralizuotą ir paskirstytą duomenų sandėlio procesą.

Duomenų sandėlio architektūra

Duomenų sandėlio architektūros pagalba galima atvaizduoti bendras duomenų struktūras, jų komunikavimą, apdorojimą ir pateikimą [17]. Architektūra sudaryta iš keleto tarpusavyje apjungtų dalių:

- operacinių duomenų bazių / išorinio duomenų bazių sluoksnio;
- informacijos priėjimo sluoksnio;
- duomenų prieigos sluoksnis;
- duomenų direktorijos (Metaduomenų) sluoksnio;
- procesų valdymo sluoksnio;
- taikomųjų programų pranešimų mainų sluoksnio;
- duomenų sandėlio (fizinis) sluoksnio;
- duomenų apdorojimo sluoksnio [19].

Sekančiame paveiksle (žr. 11 pav.) pavaizduota detali duomenų sandėlio architektūra.



11 pav. Duomenų sandėlio architektūra

Informacijos priėjimo sluoksnis

Duomenų priėjimo sluoksnis duomenų sandėlio architektūroje, tai vieta su kuria tiesiogiai bendrauja galutinis vartotojas. Būtent šioje vietoje apjungiami įrankiai, kuriuos paprastai naudojame kiekvieną dieną, pavyzdžiui, Excel, Lotus 1-2-3, Focus, Access, SAS ir pan. Šis sluoksnis taip pat atspindi tiek programinę, tiek aparatūrinę įrangą, kuri yra susijusi su ataskaitų atvaizdavimu ir spausdinimu, dinaminėmis elektroninėmis lentelėmis, grafikais ir schemomis skirtomis analizei ir prezentacijoms [17].

Duomenų prieigos sluoksnis

Duomenų prieigos sluoksnis duomenų sandėlio architektūroje yra glaudžiai susijęs su informacijos priėjimo sluoksnio galimybe gauti duomenis iš išorinių duomenų bazių. Duomenų prieigos sluoksnis apjungia ne tik skirtingas DBVS ir failų sistemas, esančias toje pačioje aparatūrinėje dalyje, bet ir gamintojus bei tinklų protokolus [13].

Duomenų direktorijos (Metaduomenų) sluoksnis

Norint sukurti universalią prieigą prie duomenų, neišvengiamai tenka sugalvoti direktoriją arba duomenų saugyklą (angl. repository), kurioje bus saugomi meta-duomenys. Meta-duomenys – tai duomenys apie duomenis esančius organizacijoje. Taigi, pilnai funkcionuojančiam duomenų sandėliui tiesiog būtina operuoti daugybe meta-duomenų [18].

Procesų valdymo sluoksnis

Procesų valdymo sluoksnis apima įvairių užduočių planavimą, kurios turi būti atliktos siekiant sukurti ir palaikyti duomenų sandėlį, bei meta-duomenų saugyklą.

Taikomųjų programų pranešimų mainų sluoksnis

Taikomųjų programų pranešimų mainų sluoksnis turi susidoroti su informacijos transportavimu organizacijos kompiuteriniuose tinkluose. Šis sluoksnis dar vadinamas „middleware“ [17].

Duomenų apdorojimo sluoksnis

Paskutinis duomenų sandėlio architektūros komponentas tai duomenų apdorojimo sluoksnis. Jis dar vadinamas perrašymo arba replikacijų valdikliu, nors iš tikrųjų apjungia visus procesus, kurie reikalingi atrinkti, redaguoti, apibendrinti, suvienyti ir pakrauti duomenis ir informaciją iš operatyvių ir/arba išorinių duomenų bazių į duomenų sandėlį [20].

1.7. XML ir EDI technologijos duomenų integravimui

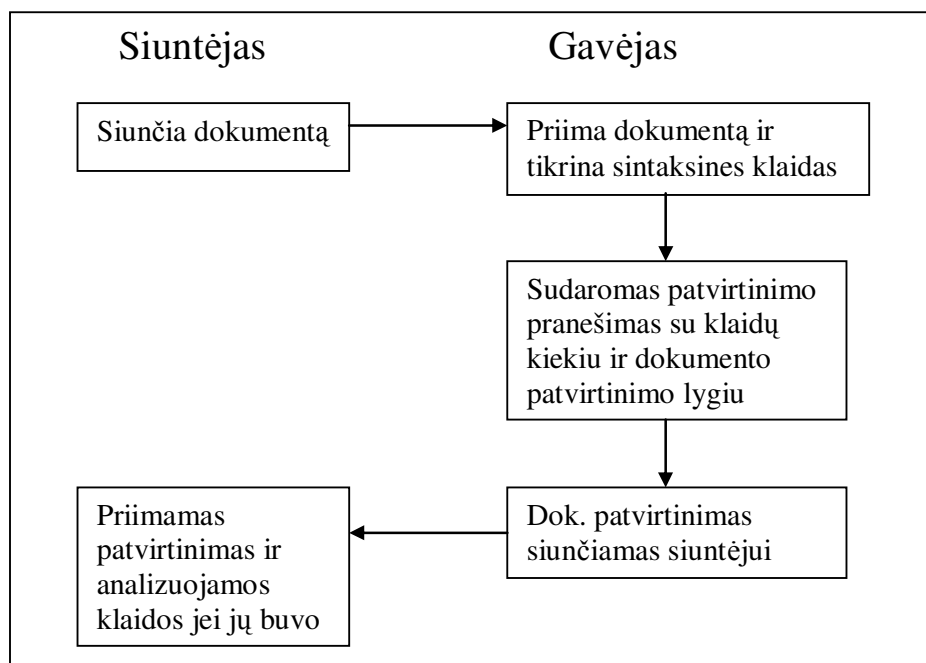
1.7.1. EDI formatas

EDI (angl. Electronic Data Interchange) – tai standartizuotas biznio duomenų apsiskeitimo formatas. Šio formato standartas buvo sukurtas Data Interchange Standards Association ir pavadintas ANSI X12. Šis standartas daugiausia naudojamas JAV, bet tarptautiniu mastu jis buvo sujungtas su kitu EDIFACT standartu.

EDI dokumentas yra formuojamas visai kitaip negu XML. EDI žinutės susideda iš eilutės duomenų elementų, kurie apibūdina tam tikrus faktus (pvz. kaina, pavadinimas ir t.t.). Šie visi duomenų elementai yra atskiriami tam tikromis žymėmis vadinamomis riba. Visa ši eilute apie tam tikrą įrašą yra vadinama duomenų segmentu. Vienas arba daugiau tokių duomenų segmentų yra talpinami į rėmelį, kuris turi savo antraštę ir treilerį. Šis rėmelis sudaro vieną transakciją, kuri yra EDI duomenų formato veikimo – perdavimo vienetas [21].

Daugumas kompanijų pasirinko EDI formatą, nes šiuo formatu labai supaprastėja dokumentų perdavimas, nes kiekvienas dokumentas yra paverčiamas transakcijų rinkiniu ir galima laisvai konvertuoti tokius duomenis į kitus formatus XML, ebXML ir panašiai.

Sekančiame paveiksle (žr. 12 pav.) pavaizduotas EDI formatu siunčiamų duomenų apdorojimo procesas.



12 pav. Siunčiamos informacijos EDI formatu apdorojimo procesas

1.7.2. XML formatas

XML (angl. „eXtensible Markup Language“) yra „žymėjimo“ kalba dokumentams, kuriuose saugoma struktūrizuota tekstinė informacija. W3C patvirtino XML standartą 1998 metais. Iš esmės tai — seniai pramonėje vartojamos kalbos SGML poaibis, specialiai pritaikytas naudojimui internete. Todėl kiekvienas XML dokumentas yra ir SGML dokumentas, bet ne atvirkščiai [15].

XML dokumentas gali prasidėti antrašte, po jos gali eiti vykdymo instrukcijos, dokumento tipo ir intarpų apibrėžimai. Po šio pradžios bloko seka esminis dokumento turinys — elementai. XML dokumentas gali būti gerai suformuotas („well formed“) ir teisingas („valid“). Gerai suformuotas dokumentas, tai XML sintaksę atitinkantis dokumentas. Jis turi turėti vieną (ir tik vieną) aukščiausio lygio (šakninį) elementą, jo elementai turi būti teisingai sukomponuoti. Teisingas dokumentas, tai savo tipo apibrėžimą atitinkantis dokumentas (žinoma, jis turi būti ir gerai suformuotas).

Komentarai gali būti bet kur dokumente, o CDATA blokai — ten, kur gali būti elementai. Visi specialūs XML simboliai, esantys tekste, o ne žymėjime (ne žymėse, atributuose, CDATA blokuose ir t. t.), turi būti pakeisti intarpais. Radusi klaidą, XML dokumentą apdorojanti programa turi pranešti apie klaidą ir nutraukti apdorojimą — skirtingai nuo HTML, kur klaidas buvo leidžiama ignoruoti. XML dokumento struktūra kartais skirstoma į „orientuotą į duomenis“ ir „orientuotą į dokumentą“.

XML privalumai: tai paprastas, išplečiamas, nuo konkrečios OS ar programavimo kalbos nepriklausomas, tarptautiniam naudojimui pritaikytas, atviras ir nemokamas standartas. XML (ir tam tikru dokumento tipo apibrėžimu) daugeliu atveju galima patogiai pakeisti atgyvenusius tekstinius ar binarinius formatus (protokolus). XML duomenimis galima lengvai manipuluoti programose, naudoti juos duomenų bazėje ar net vietoj jos. Vienas didžiausių XML privalumų — jį galima transformuoti. Tiems patiems duomenims pritaikius skirtingas transformacijas, galima gauti keletą (galbūt visiškai skirtingų) pavidalų rezultatus. Pvz., tam tikros apibrėžtos struktūros XML failuose saugomi WWW serverio „logai“ gali būti viena XSLT transformacija pervedi į visai kitokios struktūros XHTML dokumentą internetui, kita — į SVG grafiką, trečia — į WML dokumentą, skirtą mobiliesiems telefonams. [15]

1.7.3.XML dokumentų nagrinėjimas

Norint patikrinti XML dokumento korektišką sandarą ar jį panaudoti, dokumentas prieš tai turi būti apdorotas. Tai atlieka programos, vadinamos XML nagrinėtojai (angl. XML parser). Jos sugeba nuskaityti ir (dažniausiai) patikrinti XML dokumento duomenų korektiškumą. Nagrinėtojas leidžia programai naudotis XML dokumente esančiais duomenimis. Žinomiausi ir patys pagrindiniai yra du nagrinėtojų tipai [6]:

- medžiais pagrįsti nagrinėtojai, kurie XML dokumentą programoje paverčia į vidinę atminties struktūrą (DOM);
- įvykiais pagrįsti nagrinėtojai, kurie skaitydami XML dokumentą generuoja įvykius ir praneša apie tai programai (SAX).

Pasirinkimas kurį nagrinėtoją naudoti, lemia kuriama programa. Dažniau yra pasirenkamas DOM tipo, kadangi jo programavimo modelis yra paprastesnis, nei SAX įvykių apdorojimas. Tačiau, kad ir koks neįprastas būtų SAX programavimo modelis, jis gali būti truputį efektyvesnis. SAX programos gali būti žymiai spartesnės už savo DOM variantus, ir beveik visada naudoja žymiai mažiau atminties. Ypač patogiu naudoti SAX dirbant su dokumentais srautiniu režimu.

Šie programos bruožai nurodo, kad programoje geriau būtų naudoti SAX nagrinėtoją:

- dokumentai netelpa į atmintį;
- dokumentą galima apdoroti trumpais gabaliukais. Nereikia užsikrauti viso dokumento, kad galėtume pradėti su juo dirbti;
- apdorojant dokumentą, užtenka tik žinių apie prieš tai buvusius elementus, nereikia žiūrėti į priekį;
- apdorojimas gali būti išskaidytas į kelias nuoseklias operacijas[6].

Kiti bruožai kurie nurodo, kad programa tikriausiai turėtų naudoti vieną iš DOM palaikančių nagrinėtojų:

- programa vienu metu turi kreiptis į atskiras dokumento dalis ar net kelis dokumentus vienu metu;
- vidinės duomenų struktūros yra panašaus sudėtingumo kaip ir pats dokumentas;
- programa turi modifikuoti dokumentą pakartotinai;
- jei dokumentas saugomas ilgą laiką ir su juo vis atliekamos operacijos, o ne vieną kartą apdorojamas ir pamiršamas[6].

Atsižvelgiant į aptartus bruožus, šiame darbe racionaliausias pasirinkimas būtų - DOM metodas. Toliau pateikiama skirtingų DOM modelių palyginimo lentelė (žr. 5 lentelė).

5 lentelė. Skirtingų DOM modelių palyginimas[6,24]

Savybė	W3C DOM	dom4j	JDOM
Pagrįstas Java interfeisais	Taip	Taip	Ne
Palaiko Java kolekcijas	Ne	Taip	Taip
Gali naudoti bet kokį SAX nagrinėtoją ar XMLFilter	Dažniausiai	Taip	Taip
Konvertuoja iš ir į DOM medžius	Taip	Taip	Taip
Pats įgyvendina DOM sąsajas	Taip	Taip	Ne
Integruotas XPath palaikymas	Ne	Taip	Taip
Įvykiais pagrįsto nagrinėjimo palaikymas	Galimas	Taip	Ne
Gali dirbti su XML srautais	Ne	Taip	Ne
XML Schema duomenų tipų palaikymas	Ne	Taip	Ne
Didelių XML dokumentų palaikymas	Ne	Taip	Ne

JDOM yra populiariausias dokumento modelis, nes jo programavimo sąsaja labai patogi. Nors DOM suteikia daug galimybių dirbti orientuojantis į dokumento struktūrą, tačiau jis turi trūkumų pateikdamas objektiškai orientuotą supaprastinimą. Programuotojams, kurie dirba su daugiau į duomenis orientuotomis struktūromis, kitaip nei knygomis, straipsniais ar kitais pilnaverčiais dokumentais, žymiai patogiau naudotis JDOM ar dom4j, nes labiau atitinka jų poreikius. Kadangi darbe bus naudojami struktūrizuoti duomenų šaltiniai, todėl geriausias modelio pasirinkimas – JDOM.

1.7.4. Korektiškumo tikrinimo priemonės

XML dokumentas yra tekstas, kuris aprašo hierarchinę duomenų struktūrą. XML standartas aprašo sintaksę, kaip turi būti sudarytas dokumentas. Atitinkantis XML sintaksę dokumentas

vadinamas taisyklingu. Susidūrę su netaisyklingais dokumentais, įrankiai turėtų pranešti apie klaidas, nes tai jau nebėra XML dokumentas. Tačiau norint žinoti, ar dokumentas yra prasmingas, neužtenka vien tik taisyklingumo tikrinimo. Teisingi duomenys turi atitikti tam tikrą loginę struktūrą, kurią aprašo schemų kalbos. Būtent schemų kalbomis aprašomos naujos XML kalbos. Bet kuris dokumento korektiškumą tikrinantis nagrinėtojas gali užtikrinti, kad nagrinėjamas dokumentas atitinka nustatytas kalbos taisykles. Kadangi, kartu su XML standartu pateikta schemų kalba DTD turėjo mažai galimybių, įvairios organizacijos sukūrė daug įvairių schemų kalbų [12].

Seniausia schemų kalba yra dokumento tipo apibrėžimas (angl. DTD – Document Type Definition), kuris sudarytas iš taisyklių apie dokumento struktūrą ir turinį. DTD išvardina visus galimus elementus, jų pasirodymo tvarką ir atributus.

Nors dokumento tipo apibrėžimo metodas puikiai tiko pirmiems bandymams, jo sukūrimo momentu, bet kuriant sudėtingas sistemas iš karto paaiškėjo jo trūkumai. Viena pagrindinių DTD problemų – tai nėra korektiškas XML dokumentas. Todėl darbu su juo nebetinka standartiniai XML nagrinėtojai.

Dar daugiau problemų kelia tai, kad DTD turi tik ribotas galimybes dokumento struktūros ir turinio aprašymui. Juo neįmanoma aprašyti sudėtingų dokumento elementų ryšių, be to, vienam elementui aprašytos taisyklės negali būti panaudotos kito elemento aprašymui.

Šias problemas išsprendžia kitas standartas – XML Schema. Šie dokumentai yra taisyklingi XML dokumentai, jie palaiko vardų sritis bei turi priemones elementams priskirti tipus, taip leisdamas tikrinti dokumentus ne tik pagal elementų struktūrą, bet ir pagal jų turinį. Viena, labiausiai išsiskirianti iš kitų schemų kalbų XML Schema savybė, yra tipų paveldimumas.

6 lentelė. DTD ir XML schema palyginimas[4]

Savybė	DTD	XML schema
Naudojama XML sintaksė	Ne	Taip
Turi duomenų tipų savyje	10	37
Duomenų tipai gali būti naujai aprašyti pagal vartotojo poreikius	Ne	Taip
Atributo reikšmė pagal nutylėjimą	Yra	Yra
Elemento reikšmė pagal nutylėjimą	Nėra	Dalinai
Paveldimumas	Nėra	Yra
Unikalumas atributams	Galimas	Galimas
Unikalumas ne atributams	Nėra	Galimas
Atributų, elementų raktai	Nėra	Yra
Paties schemos dokumento tikrinimas	Negalimas	Galimas

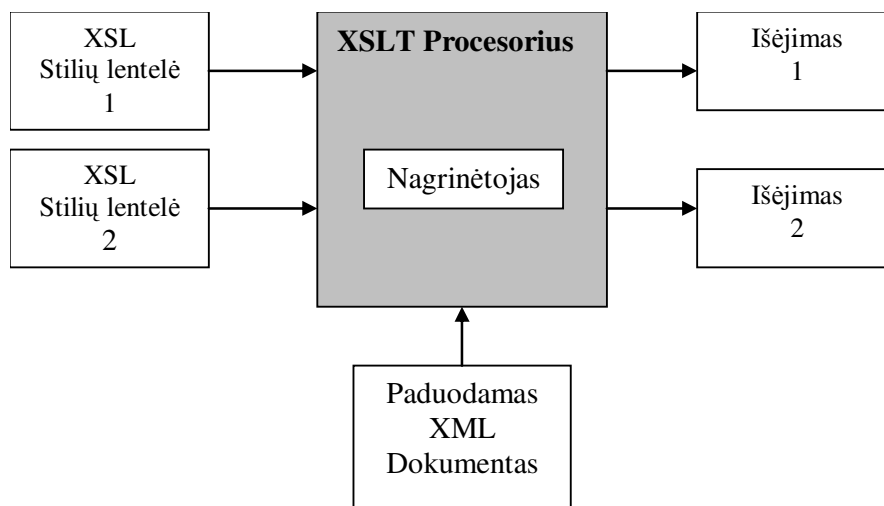
Po truputį XML Schema tampa pagrindine schemų kalba, kadangi ji yra oficialus standartas ir leidžia aprašyti žymiai daugiau informacijos apie XML dokumentą. Ja remiasi ir kiti, naujesni XML šeimos standartai: XQuery, XPath, XSLT. Atlikus šių dviejų XML dokumentų schemų savybių ir galimybių palyginimą, galima teigti, kad patogiau naudoti XML Schemas.

1.7.5. XML dokumentų transformavimas

Vienas pagrindinių XML technologijos privalumų – tai galimybė konvertuoti XML duomenis iš vienos formos į kitą, naudojantis sukurtomis priemonėmis. Technologija, kuri padeda tai atlikti, vadinasi XSLT (angl. eXtensible Stylesheet Language for Transformations) – išplečiama stilių kalba transformacijoms.

XSL stilių lentelė – tai transformavimo instrukcijų rinkinys, kuriomis išeitinis XML dokumentas verčiamas kitu dokumentu (nebūtinai XML). Norint atlikti XSLT transformacijas reikalingas XSLT procesorius, kuris transformuoja išeities XML dokumento duomenis, susiedamas juos su XSL stilių lentelės šablonais. Bet XSLT procesoriai yra lėti, todėl kartais reikia naudoti papildomas priemones norint sugeneruoti dokumentus realiu laiku [2].

XSLT kalba yra aukšto lygio programavimo kalba, kurios pagrindai paimti iš funkcinio programavimo teorijos. Ji žymiai galingesnė nei panašios paskirties standartai CSS ar CSS2, todėl vidutiniam programuotojui ją pradėti naudoti yra pakankamai sunku. Lygiai tokia pati situacija buvo ir su SQL jos atsiradimo metu, tai tik dar kartą įrodo, kad XSLT nepaprastai galinga technologija [2].



13 pav. XSLT procesoriaus darbo schema [2]

1.7.6. XML ir EDI formatų palyginimas

7 lentelėje pateiktas detalus XML ir EDI duomenų formatų palyginimas.

7 lentelė. Pagrindinių XML ir EDI formatų palyginimas

Savybės, kriterijai	XML	EDI
Standartai	Jų yra daug ir įvairių, priklausomai nuo situacijos pasirenkamas atitinkamas standartas	Pagrindinis standartas EDI yra ANSI X12 bet į jį yra įtrauktas ir EDIFACT aprašai
Galimybė transformuoti duomenis į kitus formatus	Yra	Yra
Struktūrų sudėtingumas ir suprantamumas	Dažnai XML dokumentai apsunkina duomenų apdorojimą	Duomenų struktūros labai paprastos, lengva naudoti kuriant programinę įrangą
Užkodavimo galimybės	Priklausomai nuo struktūros sudėtingumo, ne visi kodavimo algoritmai tinka	Paprasta užkoduoti
Dokumento klaidų tikrinimo galimybė	Yra priklausomai nuo XML dokumento tipo parenkamas dokumento tikrinimo tipas	Yra
Universalumas	Nepriklauso nuo sistemos platformos tipo ar OS, galima naudoti bet kur	Nepriklauso nuo sistemos platformos tipo ar OS, labiau skirtas biznio dokumentų formavimui
Populiarumas, kur labiausiai naudojami	Labai populiarus formatas Lietuvoje ir pasaulyje	Labiau paplitęs JAV tarp didžiųjų korporacijų

Abu duomenų apsikeitimo formatai yra plačiai naudojami visame pasaulyje. Jiems yra sukurta nemažai transformavimo įrankių iš vieno formato į kitą, tačiau XML formatas yra labiau priimtinas nes:

- pasirinkta sistemos programavimo kalba Visual Basic turi visus reikalingus įrankius sistemos realizavimui;
- darbe bus naudojami ne pramoninio tipo duomenys, o finansiniai duomenys;
- turi griežtas taisykles dokumentų struktūrai, jos sudarymui ir transformavimui;
- populiariesnis tarp Lietuvos vartotojų.

1.8. Panašių veikiančių sistemų lyginamoji analizė

Pasaulyje jau egzistuoja panašaus tipo tarpininkavimo sistemos, kurios atlieka tos šalies piliečių kreditavimo rizikos įvertinimą. 8 lentelėje pateiktas šių sistemų palyginimas.

8 lentelė. Panašių sistemų palyginimas

	Sistemos			
	Equifax	Experian	Infobankas	Kuriama sistema
Internetinė prieiga	www.equifax.com	www.experian.com	www.infobankas.lt	-
Integracija su klientų sistemomis	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Suderinamumas su XML standartu	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Galimybė jungtis prie sistemos per WEB	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Suderinamumas su kitų kompanijų taikomosiomis programomis ir DB	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Automatinis vartotojų informavimas apie juos dominančius pasiūlymus	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Grafinė išanalizuotų duomenų pateikimo informacija	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Galimybė apjungti ir komunalines paslaugas teikiančių įmonių informaciją į bendrą sistemą	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Analitinių duomenų pateikimas	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Interaktyvi pagalba	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Paslaugos orientuotos tik juridiniams asmenims	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Paslaugos orientuotos tik fiziniams asmenims	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Galimybė atsiskaityti už paslaugas internetu	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Aptarsime kiekvieną 8 lentelėje pateiktą nagrinėjamą sistemą:

Equifax

Tai JAV veikianti tarpininkavimo sistema, kuri buvo įkurta praeitame dešimtmetyje ir šiandien turi savo veikiančius filialus net 14-oje Pietų ir Šiaurės Amerikos šalių. Įmonei įsikūrus teikiamų paslaugų paketas buvo minimalus, bet šiandien jų paslaugų ratas prasiplėtė, pradedant nuo informacijos gavimo iki rizikos įvertinimo ir kredito išdavimo. Jų devizas skamba taip: “Mes transformuojam duomenis į informaciją”[7]. Pasitelkę naujausius technologinius sprendimus jie surenka ir išanalizuoja labai didelius informacijos kiekius. Jų teigimu šiai dienai jie išanalizuoja 6 milijonus baitų per 6 sekundes. O tai yra milžiniški informacijos kiekiai. Jų internetinis puslapis pateikia ne tik grafines duomenų analizės ataskaitas, bet ir pateikia pasiūlymus fiziniams asmenims, kaip ir kokiais būdais, jie gali pagerinti savo kredito istoriją. Tokiomis papildomomis paslaugomis jie praplečia klientų ratą ir gali sakyti, kad yra vieni iš šios srities rinkos lyderių.

Experian

Experian – tai visame pasaulyje žinoma įmonė kaip informacijos paslaugų tiekėja. Įmonės 20 metų sukaupta patirtis konsultuojant finansiniais klausimais, jai leidžia užimti lyderio pozicijas. Experian teikia paslaugas ne tik per elektronines B2B sistemas, bet ir B2C, B2B2C, B2G (angl. Business to Government) sistemas[23]. Ši kompanija pakeitė daugelį pasaulinių informacijos rinkimo ir analizavimo tendencijų. Experian sistemos integracijos lygis su kitomis pasaulinėmis internetinėmis organizacijomis yra išvystytas aukščiausiam lygmenyje. Klientų duomenų analizė yra panaudojama, net perkant internetu. Experian sistemos pagrindu yra sukurti tokie internetiniai e-bankai gigantai kaip www.paypal.com, www.e-bay.com ir kt.

Infobankas

Bendrovė teikia visas su skolų išieškojimu ir administravimu susijusias paslaugas, taip pat siūlo profesionalias teises paslaugas. Tarp bendrovės klientų - bankai, draudimo, investicinės, statybų ir kitos bendrovės. Pagrindinis Infobanko akcentas yra juridiniai asmenys, jų finansinės informacijos įvertinimas. Jų naudojamas komunikavimo įrankis yra internetinio puslapio pagalba registruojantis serveryje. Naršyklės pagalba galima gauti visus norimus duomenis. Bet šita sistema turi ir nemažai trūkumų. Vienas iš jų yra, kad sistema turi žemą integracijos lygį t.y. ji neturi galimybės būti integruota į kitų klientų informacines sistemas, tokiu būdu nepraplečiant teikiamų paslaugų ir naujų vartotojų rato. Jų sukurta ir naudojama CRM valdymo sistema veikia puikiai ir šiai dienai tenkina jų užsibrėžtų tikslų poreikius.

1.9. Analizės išvados

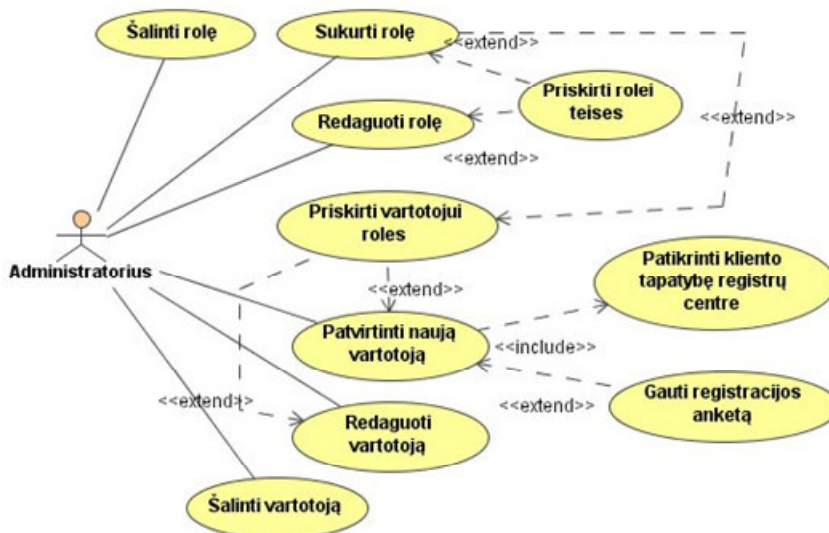
1. Išanalizuota kuriamos sistemos problematika, apibrėžta veiklos sritis, veiklos objektai, naudotojų tipai, pagrindiniai funkciniai bei nefunkciniai sistemai keliami reikalavimai.
2. Išanalizavus sistemos projektavimo metodus ir priemones, nuspręsta pasirinkti sistemos modeliui sudaryti RUP sistemos projektavimo procesą ir jį realizuojančią objektinio projektavimo priemonę Magic Draw 9.5.
3. Atlikus architektūros lyginamąją analizę, pasirinktos „Microsoft“ technologijos ir „Windows“ operacinė sistema, nes kuriamam prototipui reikalinga DBVS palaikanti transakcijos, patogus įrankis ataskaitų generavimui, bei greitai realizuojama internetinė vartotojo sąsaja.
4. Atlikta panašių pasaulyje ir Lietuvoje egzistuojančių sistemų palyginamoji analizė ir nustatyta, kad Lietuvoje tokio tipo sistemų nėra, bet jos yra būtinos, nes gali greitai ir tiksliai įvertinti klientų mokumą, bei jų finansavimo riziką.
5. Buvo išanalizuoti EDI ir XML formatai ir pasirinktas XML pagrindu sudarytų duomenų šaltinių integravimas į kuriamą informacinę sistemą, nes XML turi griežtas dokumentų kūrimo taisykles, turi daug sukurtų įrankių duomenų transformacijai, duomenų kokybės analizei, išrinkimui ir nagrinėjimui.
6. Atlikus XML standartų ir priemonių analizę nustatyta, kad nėra vienos priemonės duomenų surinkimui, nuskaitymui, transformavimui ir kokybės tikrinimui.
7. Atlikus duomenis apie piliečių kreditus integruojančios informacinės sistemos analizę, nustatyti rezultato kokybės kriterijai, kurie bus įvertinti darbo pabaigoje.

2. DUOMENŲ APIE PILIEČIŲ KREDITUS INTEGRAVIMO SISTEMOS MODELIS

2.1. Reikalavimų modelis

Šiame skyriuje bus specifikuotos skirtingų modulių (administravimo, integravimo ir internetinio) panaudojimo atvejų diagramos, bus aprašomos pagrindinių panaudojimo atvejų specifikacijos, pateikiama reikalavimo modelio klasių diagrama su ryšiais tarp klasių.

2.1.1. Administravimo modulio panaudojimo atvejų diagrama ir jos specifikacijos



14 pav. Vartotojų valdymo panaudojimo atvejų diagrama

Panaudojimo atvejams iš panaudojimo atvejų diagramos (žr. 14 pav.) pateikiamos specifikacijos (9 – 10 lentelės).

Sistemos administravimo modulio pagrindinis aktorius yra „Administratorius“. Jo pagrindinės funkcijos yra vartotojų informacijos valdymas, jų užregistravimas sistemoje, rolių ir teisių suteikimas. Aktorius „Administratorius“ taip pat atlieka ir vartotojų, rolių, teisių redagavimo funkcijas.

9 lentelė. Panaudojimo atvejo „Patvirtinti naują vartotoją“ specifikacija

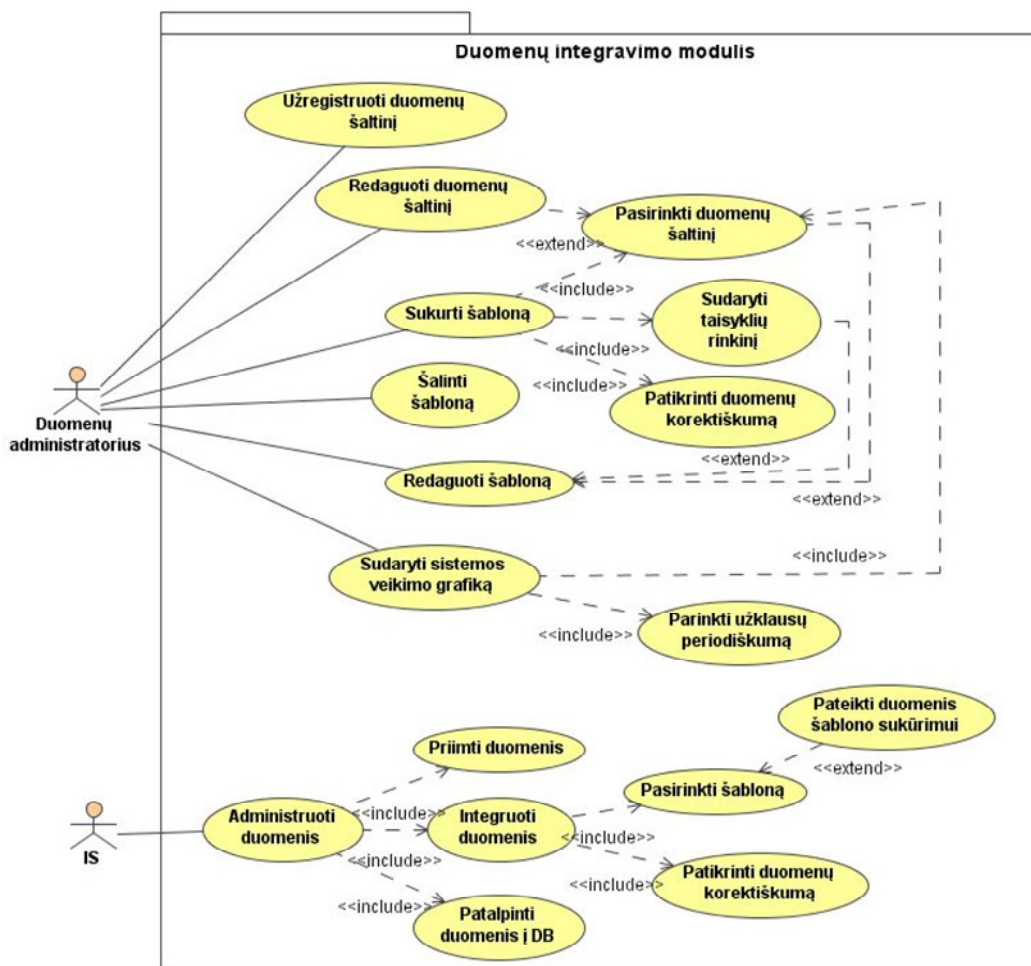
Panaudojimo atvejis	„Patvirtinti naują vartotoją“
Aktorius	Administratorius
Sistema	Duomenų apie piliečių kreditus integravimo sistema
Prieš sąlyga	Administratorius turi turėti prisijungimo prie sistemos teisę ir žinoti prisijungimo slaptažodį Vartotojas turi būti užregistravęs duomenis internetinėje anketoje arba ne.
Pagrindinis įvykių srautas	Sistemos reakcija ir sprendimai

<p>1. Įvedamas prisijungimo slaptažodis</p> <p>2. Iš administratoriaus valdymo meniu pasirenkamas punktas „vartotojai“.</p> <p>3. Pasirenkamas naujai užsiregistravęs vartotojas ir mygtuko „Tikrinti“ paspaudimu patikrinama tokio vartotojo egzistavimas registru centre/Sodroje.</p> <p>4. Mygtuko „Sukurti“ paspaudimu patvirtinamas vartotojo sukūrimas</p>	<p>1. Sistema tikrina, ar teisingai įvestas slaptažodis;</p> <p>2. Sistema išveda informaciją apie vartotojus;</p> <p>3. Paspaudus mygtuką „Tikrinti“ sistema jungiasi prie registru centro ar prie Sodros duomenų bazių(priklausomai koks fizinis ar juridinis asmuo registruojamas) ir patikrina ar toks subjektas egzistuoja ir patikrina pateiktus duomenis su esančiais išorinėse DB. Jeigu pateikta informacija teisinga sistema leidžia patvirtinti naujo vartotojo sukūrimą. Jeigu informaciją neatitinką faktų, sistema nepatvirtina naujo vartotojo tapatybės ir neaktyvuoja „Sukurti“ mygtuko, tuo pačiu pateikiamas sisteminis klaidos pranešimas administratoriui;</p> <p>4. Sistema praneša apie sėkmingai atliktą duomenų redagavimą;</p>
Po sąlyga	Pakeitimai saugomi duomenų bazėje, sistemos administratorius gali peržiūrėti pakitimus
Alternatyvos (nesėkmės atvejai)	Blogai įvedamas slaptažodis – išvedamas pranešimas apie klaidą ir grįžtama į pirmą žingsnį
Vykdyto variantai	
Veiklos taisyklės	Vartotojas turi pateikti teisingą slaptažodį
Specialūs reikalavimai (nefunkciniai)	Paredaguoti duomenys turi saugiai pasiekti duomenų bazę

10 lentelė. Panaudojimo atvejo „Sukurti Rolę“ specifikacija

Panaudojimo atvejis	„Sukurti rolę“
Aktorius	Administratorius
Sistema	Duomenų apie piliečių kreditus integravimo sistema
Prieš sąlyga	Administratorius turi turėti prisijungimo prie sistemos teisę ir žinoti prisijungimo slaptažodį.
Pagrindinis įvykių srautas	Sistemos reakcija ir sprendimai
<p>1. Įvedamas prisijungimo slaptažodis</p> <p>2. Iš administratoriaus valdymo meniu pasirenkamas punktas „Rolės“.</p> <p>3. Spaudžiamas mygtukas „Sukurti naują rolę“.</p> <p>4. Suvedama reikalinga informacija ir paspaudžiamas mygtukas „Sukurti“.</p>	<p>1. Sistema tikrina, ar teisingai įvestas slaptažodis;</p> <p>2. Sistema išveda sąrašą egzistuojančių rolių;</p> <p>3. Sistema atidaro naujos rolės sukūrimo langą;</p> <p>4. Suvedus informaciją sistema praneša apie sėkmingai atliktą duomenų įvedimą arba nesėkmės atveju apie klaidos pranešimą;</p>
Po sąlyga	Naujai įvesti pakeitimai saugomi duomenų bazėje, sistemos administratorius gali peržiūrėti pakitimus
Alternatyvos (nesėkmės atvejai)	Blogai įvedamas slaptažodis – išvedamas pranešimas apie klaidą ir grįžtama į pirmą žingsnį
Vykdyto variantai	
Veiklos taisyklės	Vartotojas turi pateikti teisingą slaptažodį
Specialūs reikalavimai (nefunkciniai)	Paredaguoti duomenys turi saugiai pasiekti duomenų bazę

2.1.2. Duomenų integravimo modulio panaudojimo atvejų diagrama ir jos specifikacijos



15 pav. Duomenų integravimo valdymo ir objektų kūrimo panaudojimo atvejų diagrama

Panaudojimo atvejams iš panaudojimo atvejų diagramos (žr. 15 pav.) pateikiamos specifikacijos (11 – 13 lentelės).

Duomenų administravimo modulis sudarytas iš dviejų dalių: duomenų prieigos ir veikimo nustatymo, bei sistemos automatinės duomenų integravimo dalies. Aktorius „Duomenų administratorius“ organizuoja ir valdo sistemos suderinimo darbus, kad sistema pagal sudarytą grafiką galėtų vykdyti automatinį duomenų integravimo procesą. T.y. šiam tikslui turi būti sukurti duomenų šaltiniai, sudarytas sistemos veikimo grafikas ir sukurti šablonai duomenų integracijai, pagal kuriuos sistema automatiškai galės iš gautų dokumentų atrinkti ir surašyti tik sistemai reikiamus duomenis. Šablono sudarymo metu yra priskiriamas duomenų šaltinis, sudaromos veiklos taisyklės. Visą tai reikalinga, nes kiekviena įmonė ar veikianti palikuoninė DB turi tik jai savitą sukurtą duomenų formatą ir struktūrą, kurių integracijos į bendrą sistemą metu, reikia įvertinti duomenų formatą, struktūrą ir sintaksę. Visos šios operacijos ir bus

atliekamos sistemos duomenų integravimo modulyje, kurio viena dalis automatizuota, o kita būtina neautomatizuota, kad išvengti klaidingos informacijos patekimą į integruojančią sistemą.

11 lentelė. Panaudojimo atvejo „Užregistruoti duomenų šaltinį“ specifikacija

Panaudojimo atvejis	„Užregistruoti duomenų šaltinį“
Aktorius	Duomenų administratorius
Sistema	Duomenų apie piliečių kreditus integravimo sistema
Prieš sąlyga	Duomenų administratorius turi turėti prisijungimo prie sistemos teisę ir žinoti prisijungimo slaptažodį.
Pagrindinis įvykių srautas	Sistemos reakcija ir sprendimai
1. Įvedamas prisijungimo slaptažodis 2. Iš duomenų administratoriaus valdymo meniu pasirenkamas punktas „Duomenų šaltiniai“. 3. Spaudžiamas mygtukas „Sukurti naują duomenų šaltinį“. 4. Suvedama reikalinga informacija ir paspaudžiamas mygtukas „Sukurti“.	1. Sistema tikrina, ar teisingai įvestas slaptažodis; 2. Sistema išveda sąrašą egzistuojančių duomenų šaltinių; 3. Sistema atidaro naujo duomenų šaltinio sukūrimo langą; 4. a) Suvedus naujo duomenų šaltinio pavadinimą, nuorodą, vartotoją, slaptažodį ir paspaudus mygtuką „sukurti“ sistema patikrina prisijungimą prie nutolusio duomenų šaltinio. b) Testuojamas prisijungimas prie naujos sistemos; c) Jei ryšys pavyksta sistema praneša apie sėkmingai atliktą duomenų įvedimą; d) nesėkmės atveju sistema pateikia klaidos pranešimą ir klaidos tipą ir grįžtama į 4.a punktą pakoreguoti informaciją.
Po sąlyga	Naujai įvesti pakeitimai saugomi duomenų bazėje, duomenų administratorius gali peržiūrėti pakitimus
Alternatyvos (nesėkmės atvejai)	Blogai įvedamas slaptažodis – išvedamas pranešimas apie klaidą ir grįžtama į pirmą žingsnį
Vykdyto variantai	
Veiklos taisyklės	Vartotojas turi pateikti teisingą slaptažodį
Specialūs reikalavimai (nefunkciniai)	Paredaguoti duomenys turi saugiai pasiekti duomenų bazę

12 lentelė. Panaudojimo atvejo „Sukurti šabloną“ specifikacija

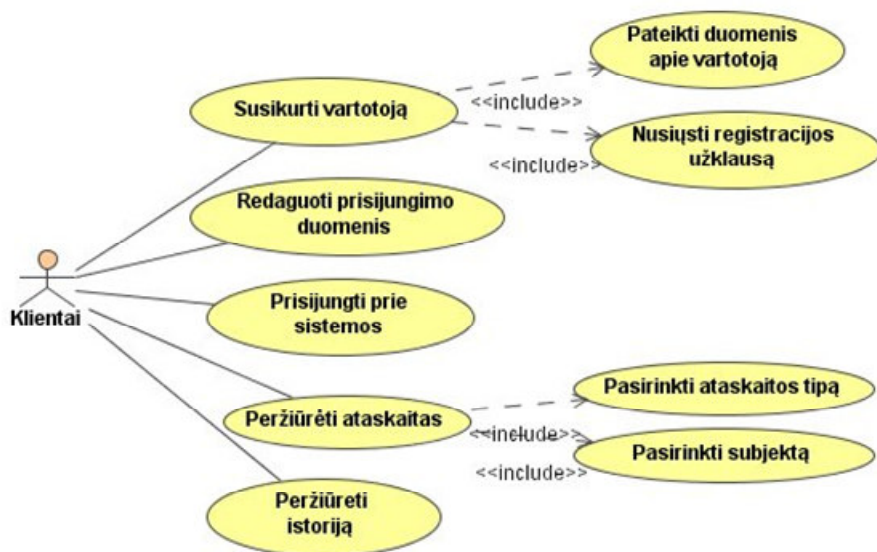
Panaudojimo atvejis	„Sukurti šabloną“
Aktorius	Duomenų administratorius
Sistema	Duomenų apie piliečių kreditus integravimo sistema
Prieš sąlyga	Duomenų administratorius turi turėti prisijungimo prie sistemos teisę ir žinoti prisijungimo slaptažodį. Šablono būseną „Naujas“;
Pagrindinis įvykių srautas	Sistemos reakcija ir sprendimai

<ol style="list-style-type: none"> 1. Įvedamas prisijungimo slaptažodis. 2. Iš duomenų administratoriaus valdymo meniu pasirenkamas punktas „Šablonai“. 3. Spaudžiamas mygtukas „Sukurti naują šabloną“. 4. Pasirenkamas duomenų šaltinis iš esančio sąrašo. Arba įvedamas naujas duomenų šaltinis. 5. Patikrinami ar toliau iš XML failo pateikti duomenys yra korektiški ir tinkami naudoti. 6. Kiekvienai XML failo žymei sudaromas taisyklių rinkinys iš pateikto sąrašo. 7. Spaudžiamas mygtukas „Sukurti“ 8. Spaudžiamas mygtukas „Testuoti“ 9. Spaudžiamas mygtukas „Patvirtinti šabloną“ 	<ol style="list-style-type: none"> 1. Sistema tikrina, ar teisingai įvestas vartotojo slaptažodis; 2. Sistema išveda sąrašą egzistuojančių šablonų; 3. Sistema atidaro naujo šablono sukūrimo langą, kuriame paeiliui yra išdėstoma tokia informacija: <ol style="list-style-type: none"> a) suformuojama duomenų šaltinio pasirinkimo forma ir nuoroda į naujo įrašo sukūrimą, b) suformuojama taisyklių rinkinio nustatymo forma (sistema nuskaito iš pasirinkto XML failo duomenis ir struktūriškai išdėstomi laukai pagal gautas XML dokumento žymes); 4.a) Pasirinkus duomenų šaltinį sistema patikrina to šaltinio egzistavimą ir, kad XML failas, kuriam bus kuriamos integravimo taisyklės yra iš to paties šaltinio. b) pasirinkus punktą „Sukurti naują duomenų šaltinį“ visos operacijos yra atliekamos kaip panaudojimo atveju „Užregistruoti duomenų šaltinį“ pradedant 3 punktu. c) įvedus naują duomenų šaltinį pereinama į 4.a punktą. 5. Paspaudus mygtuką „Sukurti“ sistema patikrina ar visi laukai korektiškai užpildyti. <ol style="list-style-type: none"> a) Klaidos atveju nurodoma kuris laukas blogai užpildytas ir pereinama į 4.a punktą. b) Aktyvuojamas mygtukas „Testuoti“. Šablono būseną pakeičiama į „Testuojamas“. 6. Paspaudus mygtuką „Testuoti“ sistemą pagal sudarytas laukų integravimo į DB taisykles surašo informaciją į DB. <ol style="list-style-type: none"> a) jeigu informacijos įrašymas pavyko išmetamas šablono testavimo patvirtinimas ir aktyvuojamas mygtukas „Patvirtinti“ b) įrašymo klaidos atveju išmetamas sisteminis klaidos pranešimas ir pereinama į 4.a punktą duomenų ar taisyklių koregavimui. 7. Paspaudus mygtuką „patvirtinti“ sistema pateikia sisteminį informacijos patvirtinimo pranešimą. Šablono būseną pakeičia į „Patvirtintas“.
Po sąlyga	Naujai įvesti pakeitimai saugomi duomenų bazėje, duomenų administratorius gali peržiūrėti pakitimus
Alternatyvos (nesėkmės atvejai)	Blogai įvedamas slaptažodis – išvedamas pranešimas apie klaidą ir grįžtama į pirmą žingsnį
Vykdymo variantai	
Veiklos taisyklės	Vartotojas turi pateikti teisingą slaptažodį
Specialūs reikalavimai (nefunkciniai)	Integravimo taisyklės turi būti sudarytos logiškai pagal duomenis atitinkančias XML žymes ir DB lentelės laukus.

13 lentelė. Panaudojimo atvejo „Sudaryti sistemos veikimo grafiką“ specifikacija

Panaudojimo atvejis	„Sudaryti sistemos veikimo grafiką“
Aktorius	Duomenų administratorius
Sistema	Duomenų apie piliečių kreditus integravimo sistema
Prieš sąlyga	Duomenų administratorius turi turėti prisijungimo prie sistemos teisę ir žinoti prisijungimo slaptažodį.
Pagrindinis įvykių srautas	Sistemos reakcija ir sprendimai
1. Įvedamas prisijungimo slaptažodis. 2. Iš duomenų administratoriaus valdymo meniu pasirenkamas punktas „Sistemos grafikas“. 3. Spaudžiamas mygtukas „Sukurti naują įrašą“. 4. Suvedama reikalinga informacija kada, koku periodiškumu sistemai jungtis prie išorinės sistemos, kad pasiimti duomenis ir pasirenkamas duomenų šaltinis. Suvedus informaciją paspaudžiamas mygtukas „Įvesti“.	1. Sistema tikrina, ar teisingai įvestas slaptažodis; 2. Sistema išveda sąrašą įrašų, pagal kuriuos sistema nustatytu laiku jungiasi prie išorinių duomenų šaltinių pasiimti informacijai; 3. Sistema atidaro naujo įrašo sukūrimo langą; 4. a) Paspaudus mygtuką „Įvesti“ sistema patikrina ar įvesti jungimosi duomenys korektiški ir jungimosi periodiškumas nesikerta su jau egzistuojančiais įrašais grafike. Jei operacija įvykdyta sėkmingai, pateikiamas sisteminis patvirtinimo pranešimas. b) jei netinka prisijungimo laikas, sistema paprašo pakoreguoti sistemos jungimosi periodiškumą ar laikotarpį ir grįžtama į 3 punktą.
Po sąlyga	Naujai įvesti pakeitimai saugomi duomenų bazėje, duomenų administratorius gali peržiūrėti pakitimus. Sistema dirba pagal naujai sudarytą duomenų apsikeitimo grafiką.
Alternatyvos (nesėkmės atvejai)	Blogai įvedamas slaptažodis – išvedamas pranešimas apie klaidą ir grįžtama į pirmą žingsnį. Blogai sudarytas jungimosi grafikas, gali pakenkti sistemos optimaliam sistemos darbui ir sistemos našumui.
Vykdyimo variantai	
Veiklos taisyklės	Vartotojas turi pateikti teisingą slaptažodį
Specialūs reikalavimai (nefunkciniai)	Paredaguoti duomenys turi saugiai pasiekti duomenų bazę. Periodiškumas ir laikotarpis turi būti parinkti logiškai abiejų sistemų (išorinės ir vidinės) apkrovimo atžvilgiu.

2.1.3. Internetinės prieigos modulio panaudojimo atvejų diagrama ir jos specifikacijos



16 pav. Internetinio modulio „Klientai“ vartotojo panaudojimo atvejų diagrama

Panaudojimo atvejams iš panaudojimo atvejų diagramos (žr. 16 pav.) pateikiamos specifikacijos (14 – 15 lentelės).

Internetinės prieigos modulio paskirtis, suteikti klientams galimybę prisijungti prie sistemos ir naudotis sistemos resursais. Pagrindinis aktorius yra „Klientai“. „Klientai“ gali neprisijungę prie sistemos užregistruoti anketą naujo vartotojo gavimui, o sistemos administratoriui patvirtinus jų tapatybę yra sukuriamas vartotojas. Prisijungę „Klientai“ turi savo duomenų redagavimo teises, ir įvairių ataskaitų kūrimo teises. Ataskaitos kūrimo metu, „Klientai“ atlieka duomenų paieškos ir ataskaitos tipo pasirinkimo funkcijas.

14 lentelė. Panaudojimo atvejo „Susikurti vartotoją“ specifikacija

Panaudojimo atvejis	„Susikurti vartotoją“
Aktorius	Klientai
Sistema	Duomenų apie piliečių kreditus integravimo sistema
Prieš sąlyga	Prie sistemos neprisijungusiam vartotojui, sistema turi būti suteikusi „anonymous“ vartotojo teises.
Pagrindinis įvykių srautas	Sistemos reakcija ir sprendimai
1. Internetiniame tinklapyje pasirenkamas meniu punktas „Registracija“. 2. Pasirenkama fizinio ar juridinio asmens registracijos forma. 3. Suvedami duomenys į visus laukus ir paspaudžiamas mygtukas „Registruoti“. 4. Jeigu nebenorima registruoti ar norima perrašyti duomenis	1. Sistema tikrina, ar neregistruotas vartotojas turi anonymous vartotojo teises ir pagal tai suformuoja jam skirtus peržiūrėti meniu ir internetinių puslapių šablonus; 2. Priklausomai nuo pasirinkimo, sistema pateikia detalią registracijos formą, su nurodytais privalomais užpildyti laukais; 3. a) Paspaudus mygtuką registruoti sistema patikrina ar suvesti visi privalo laukai ir ar

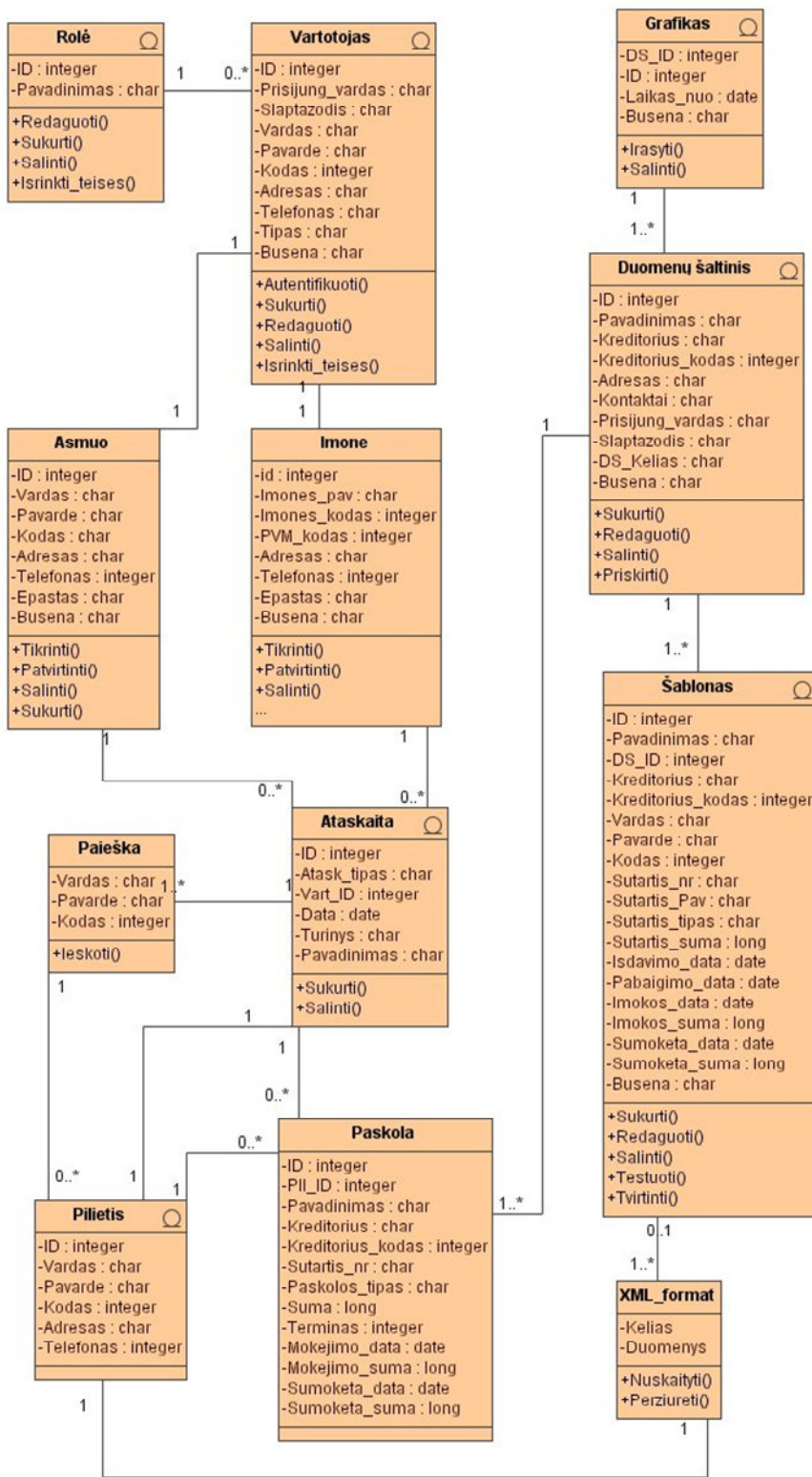
spaudžiamas mygtukas „Atšaukti“.	duomenys pateikti korektiški, jei neišmetamas sisteminis pranešimas ir grįžtama į 2 punktą; b) jei pateikti duomenys teisingi, sistema užregistruoja ir tolesniam patvirtinimui nusiunčia anketą administratoriui. 4. Paspaudus mygtuką „Atšaukti“ iš registracijos formos ištrinami visi į anketą suvesti duomenys ir sistema laukia tolimesnių vartotojo veiksmų.
Po sąlyga	Sukuriama naujo vartotojo anketa, kuri yra nusiunčiama sistemos administratoriui, sistemos administratorius gali peržiūrėti pakitimus.
Alternatyvos (nesėkmės atvejai)	-
Vykdyimo variantai	
Veiklos taisyklės	Vartotojas turi pateikti korektiškus duomenis, kurie bus patikrinti sistemos administratoriaus.
Specialūs reikalavimai (nefunkciniai)	Paredaguoti duomenys turi saugiai pasiekti duomenų bazę.

15 lentelė. Panaudojimo atvejo „Peržiūrėti ataskaitas“ specifikacija

Panaudojimo atvejis	„Peržiūrėti ataskaitas“
Aktorius	Klientai
Sistema	Duomenų apie piliečių kreditus integravimo sistema
Prieš sąlyga	Klientas turi turėti prisijungimo prie sistemos teisę ir žinoti prisijungimo slaptažodį.
Pagrindinis įvykių srautas	Sistemos reakcija ir sprendimai
1. Įvedamas prisijungimo slaptažodis. 2. Internetiniame tinklapyje pasirenkamas meniu punktas „Ataskaitos“. 3. Paspaudus mygtuką „Ieškoti“ per paieškos formą pasirenkamas fizinio asmens subjektas. 4. Pasirenkamas paieškos subjektas. 5. Pasirenkamas ataskaitos tipas.	1. Sistema tikrina, ar teisingai įvestas slaptažodis; 2. Sistema pateikia vartotojui ataskaitos generavimo langą, kuris susideda iš subjektų paieškos formos ir ataskaitos tipo pasirinkimo; 3. a) Paspaudus mygtuką „Ieškoti“ sistema pagal pateiktus duomenis patikrina ar toks subjektas egzistuoja, o jei egzistuoja jo duomenis pateikia į ekraną; b) jei pasirinkti paieškos kriterijai neteisingi, sistema pateikia sisteminį klaidos pranešimą, kad toks subjektas nerastas ir grįžtama į 2 punktą. 4. Sistema pateikia ataskaitų pasirinkimo variantus. 5. Sistema į vartotojo sąsają pateikia pasirinktos ataskaitos detalius duomenis.
Po sąlyga	Peržiūrai pateikiama sugeneruota ataskaita.
Alternatyvos (nesėkmės atvejai)	-
Vykdyimo variantai	
Veiklos taisyklės	Vartotojas turi pasirinkti korektiškus duomenis
Specialūs reikalavimai (nefunkciniai)	Ataskaitos formavimas neturi užtrukti ilgą laiko tarpą (ne ilgiau kaip 1 – 2 min.)

2.1.4. Dalykinės srities klasių diagrama

Klasių diagrama vaizduoja sistemos klases ir ryšius tarp jų. Klasių diagrama sudaryta iš tokių klasių (žr. 17 pav.) :



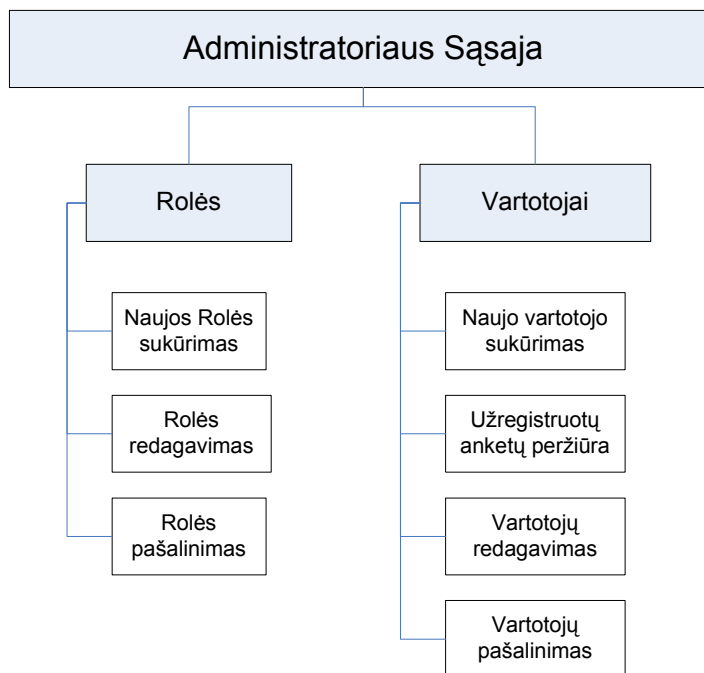
17 pav. Dalykinės srities klasių diagrama

- Vartotojas – informacija apie sistemos vartotojus;
- Rolė – informacija apie sistemos roles;
- Įmonė – informacija apie užsiregistravusią įmonė-klientą;
- Asmuo – informacija apie užsiregistravusį fizinį asmenį-klientą;
- Ataskaita – informacija apie sugeneruotas ataskaitas su bendra informacija apie konkretų pilietį;
- Paieška – informacija naudojama duomenų paieškai;
- Piliėtis – informacija apie pilietį;
- Paskola – piliečių paimtų paskolų informacija;
- XML_format – gaunama informacija iš išorinių šaltinių;
- Šablonas – tai struktūrinė informacija, kuria remiantis informacinė sistema integruoja gautus duomenis į bendrą sistemą;
- Duomenų šaltinis – informacija apie išorinius duomenų šaltinius;
- Grafikas – informacija apie sistemos veikimo grafiką.

2.2. Vartotojo sąsajos modelis

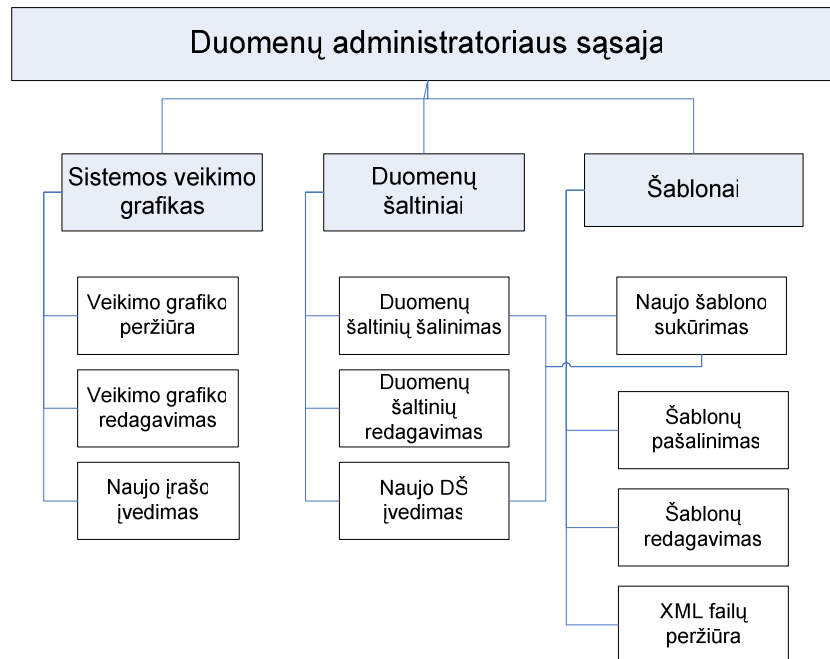
Šiame skyrelyje bus sudaryti pagrindinių sistemos modulių vartotojo sąsajos modeliai.

2.2.1. Administravimo modulio vartotojo sąsajos modelis



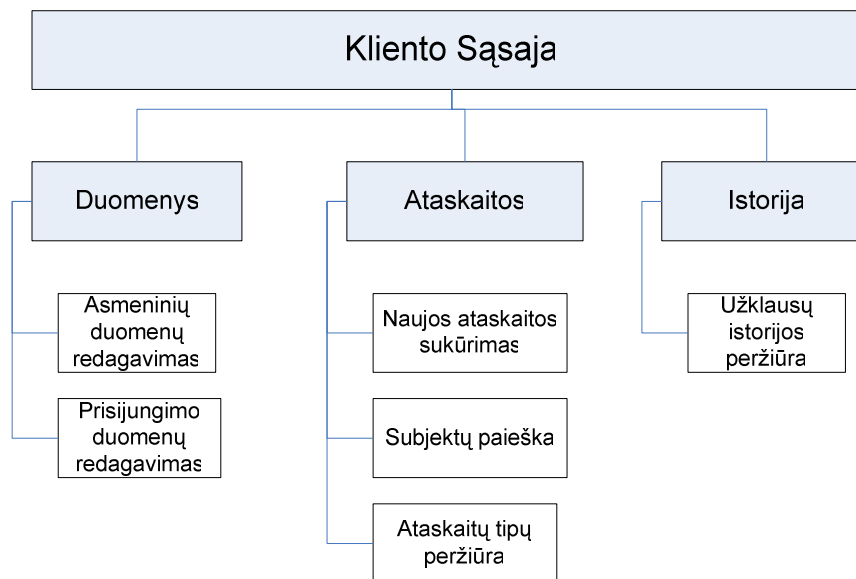
18 pav. Administravimo sąsajos modelis

2.2.2. Duomenų integravimo modulio vartotojo sąsajos modelis



19 pav. Duomenų administratoriaus sąsajos modelis

2.2.3. Internetinio modulio vartotojo sąsajos modelis



20 pav. Kliento sąsajos modelis

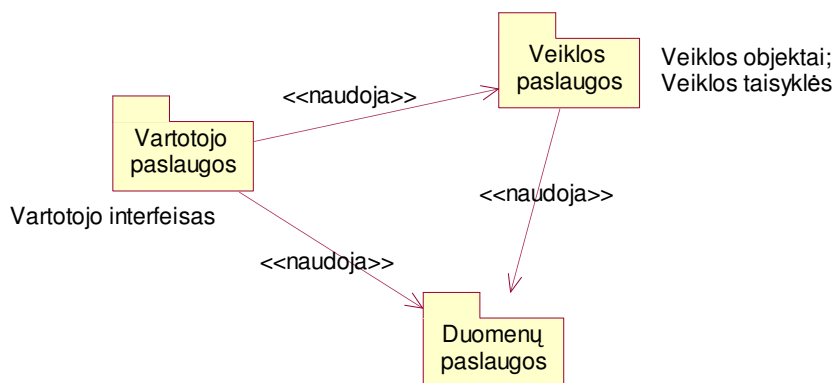
2.3. Sistemos projektas

2.3.1. Projekto tikslas

Projekto tikslas – sukurti sistemos administravimo sąsają ir internetinę prieigą turintį duomenų apie piliečių kreditus integravimo sistemos modelį, kuris tenkintų visus prieš tai aprašytus keliamus reikalavimus.

2.3.2. Sistemos architektūra

Pasirinkta trijų sluoksnių architektūra sudaryta iš atvaizdavimo, veiklos logikos ir duomenų sluoksnių (žr. 21 pav.).



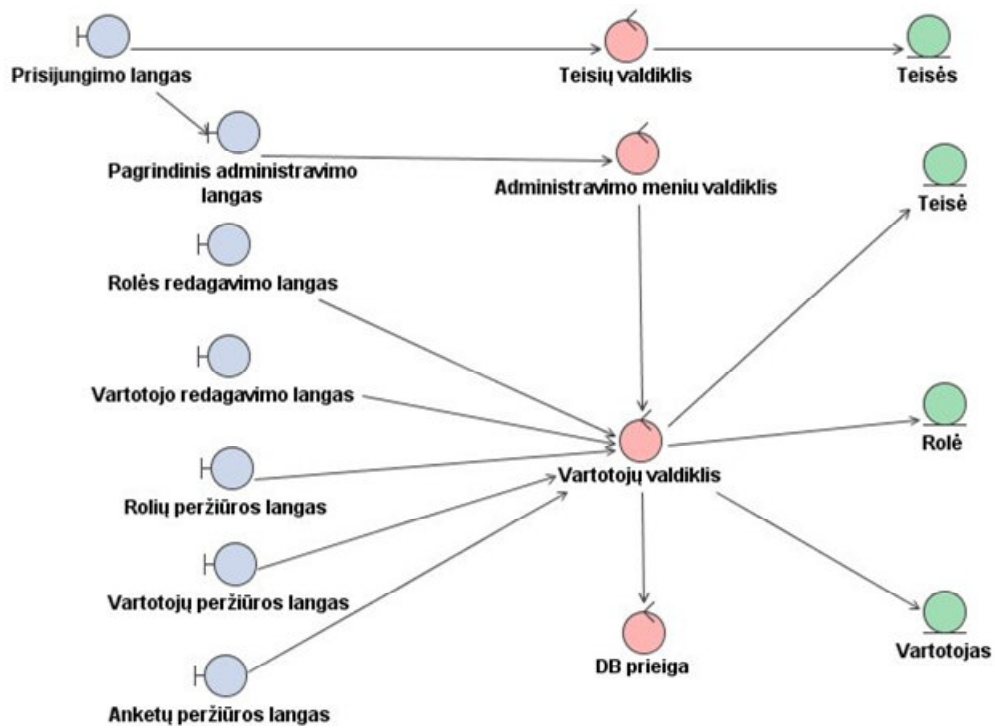
21 pav. Sistemos architektūra

Atvaizdavimo sluoksnis, atsakingas už vartotojo paslaugas, realizuoja vartotojo sąsają, kad vartotojas galėtų pasirinkti norimas atlikti funkcijas ir matytų pasirinktą informaciją.

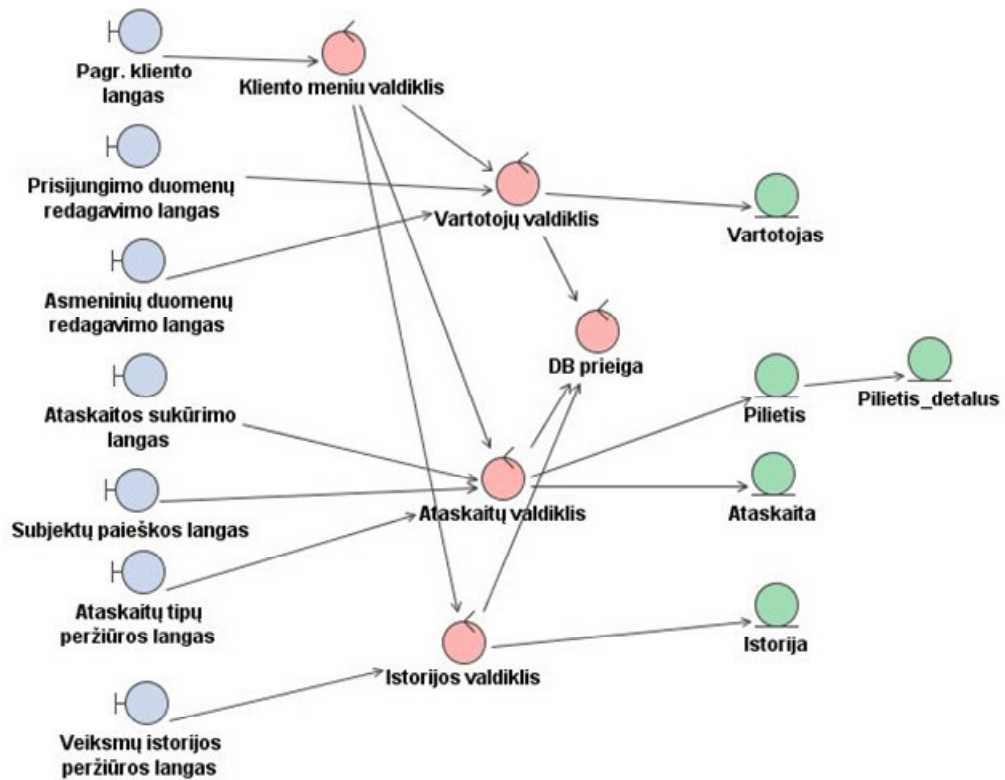
Veiklos logikos sluoksnis, atsakingas už veiklos paslaugas, užtikrina sistemos funkcionalumą, t.y. vartotojo sistemoje pasirinktų funkcijų atlikimą. Visi veiklos valdikliai atlikdami veiksmus su duomenimis kreipiasi į prisijungimo prie duomenų bazės valdiklį „DB prieiga“.

Duomenų sluoksnis atsakingas už duomenų paslaugas – saugo sistemai reikalingus duomenis. Duomenų įterpimas, atnaujinimas, šalinimas ir nuskaitymas iš duomenų bazės realizuojamas tik per vieną veiklos logikos elementą (valdiklį „DB prieiga“) taip užtikrinant programinį tvarkingumą.

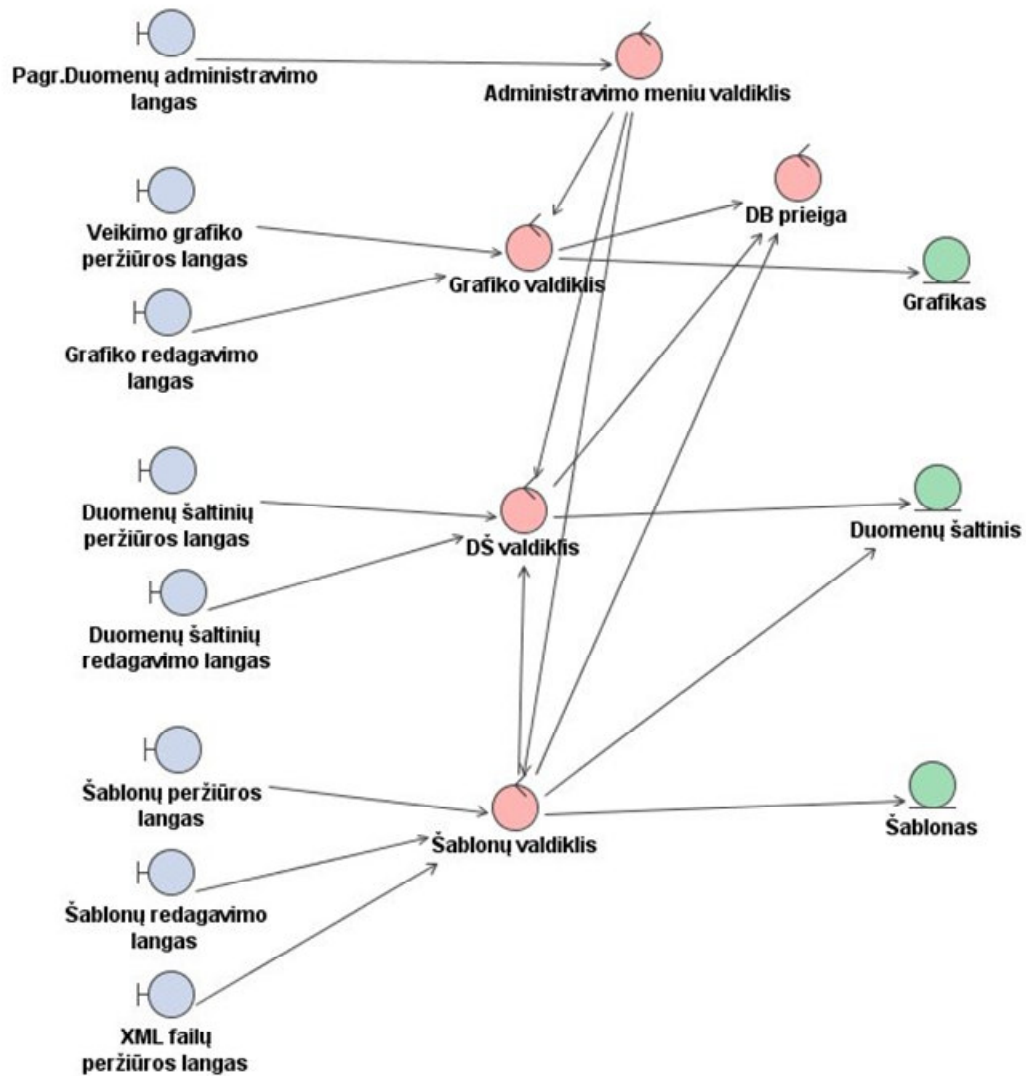
Sekančiuose paveiksluose (žr. 22 pav. – 24 pav.) pateikiamos komponentų klasių diagramos, parodančios vartotojo sąsajos, duomenų bei veiklos paslaugų elementus bei ryšius tarp jų.



22 pav. Administravimo posistemio vartotojų valdymo modelis – vaizdas – valdiklis diagrama



23 pav. Internetinio posistemio vartotojų „Klientai“ valdymo modelis – vaizdas – valdiklis diagrama



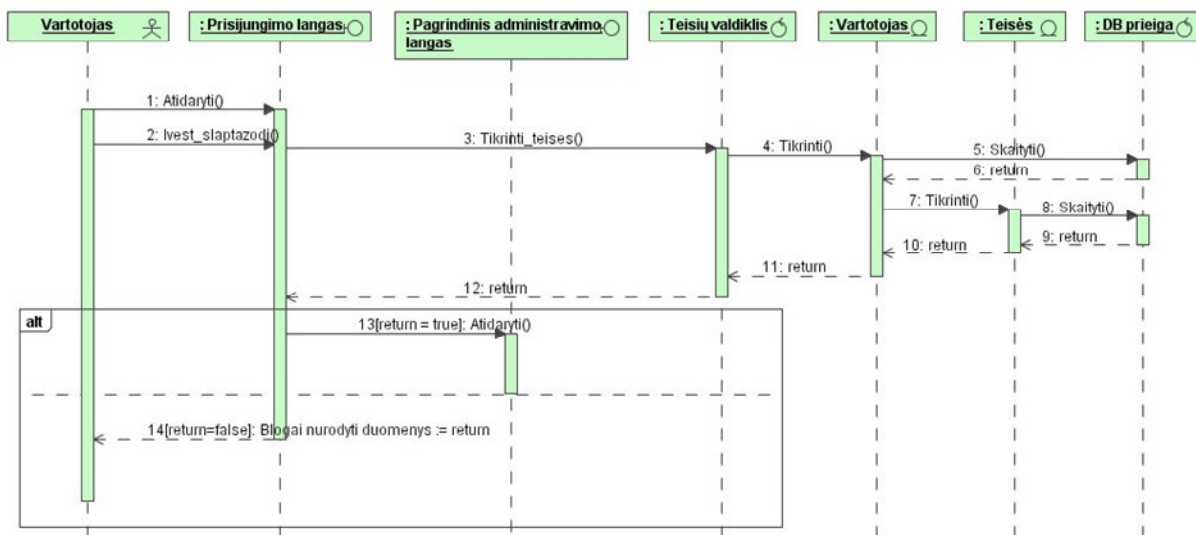
24 pav. Duomenų administravimo posistemo vartotojų valdymo modelis – vaizdas – valdiklis diagrama

2.3.3. Sistemos veiklos diagramos su pagrindinių objektų srautais

Veiklos diagramos su objektų srautais leidžia susieti veiksmus su objektų būsenų pasikeitimais. Todėl pateikiamos tik pagrindinių panaudojimo atvejų sekų, veiklos ir būsenų diagramos, kurios reikalingos parodyti darbo idėją ir rezultatą.

Sistemoje atliekama daug panašių pasikartojančių įvykių, tokių kaip prisijungimas prie sistemos. Todėl tikslinga nagrinėti šį procesą tik vieną kartą, o vėliau naudoti tik nuorodą (reference) į jį.

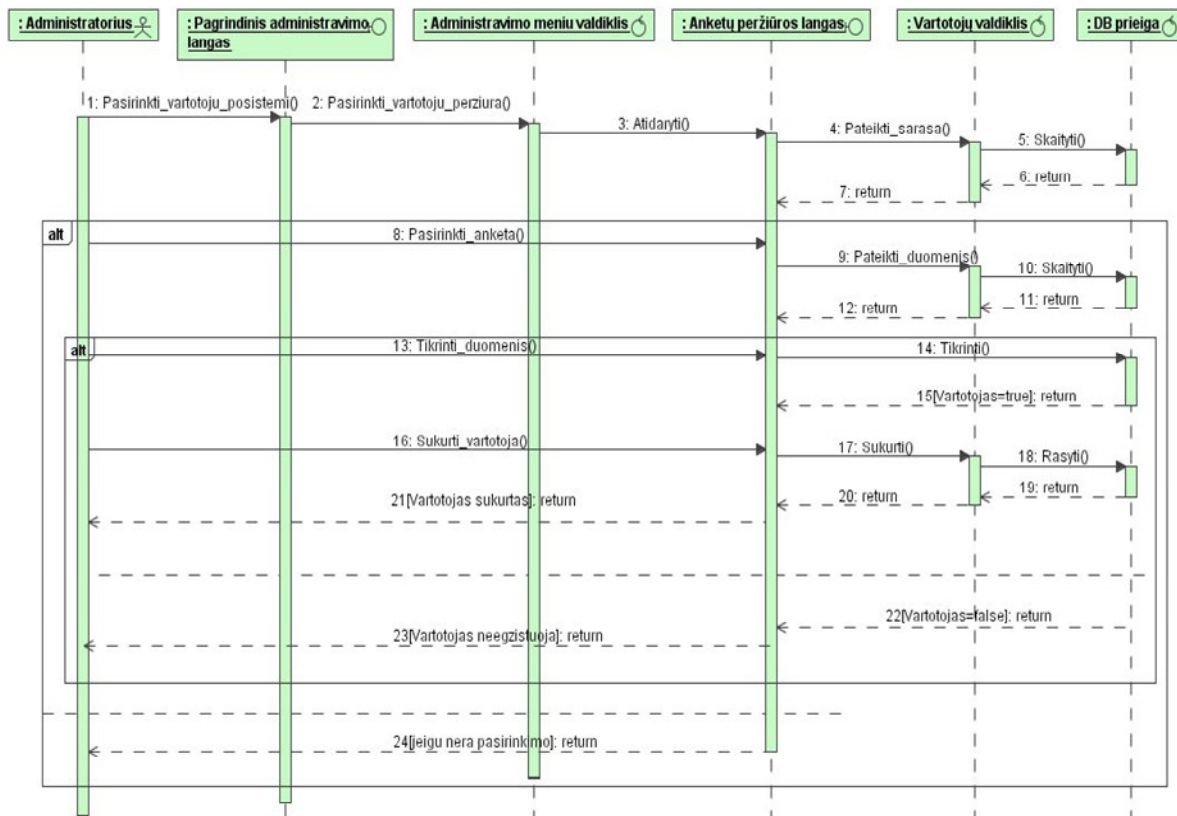
25 paveiksle pateikiama prisijungimo prie sistemos sekų diagrama.



25 pav. Prisijungimo prie sistemos sekų diagrama

Vartotojas inicijuoja pagrindinio prisijungimo lango atidarymą. Atidarius langą, paprašoma suvesti vartotojo vardą ir slaptažodį. Vartotojui įvedus duomenis, sistema tikrina vartotojo prisijungimo duomenis ir teises. Jeigu prisijungimo duomenys buvo nurodyti teisingi, sistema automatiškai atidaro tik tam vartotojui skirtą pagrindinį langą ir vartotojo meniu. Jeigu buvo įvesti blogi duomenys, tada sistema pateikia sisteminį klaidos pranešimą ir vartotojo paprašoma patikslinti prisijungimo duomenis. Ši sekų diagrama yra analogiška visų tipų vartotojams, tik priklausomai nuo vartotojo tipo, atidaroma tik jam skirta vartotojo sąsaja.

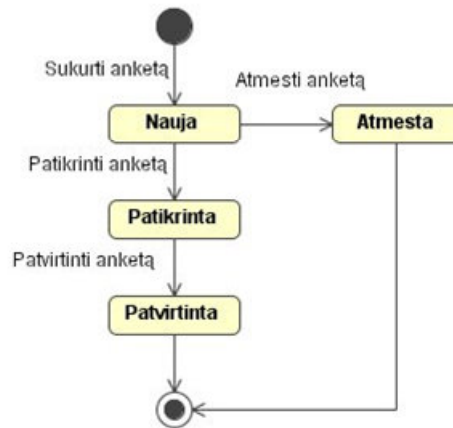
Sekanti sekų diagrama (žr. 26 pav.) vaizduoja panaudojimo atvejo „Patvirtinti naują vartotoją“ sistemos komponentų sąveiką.



26 pav. Panaudojimo atvejo „Patvirtinti naują vartotoją“ sekų diagrama

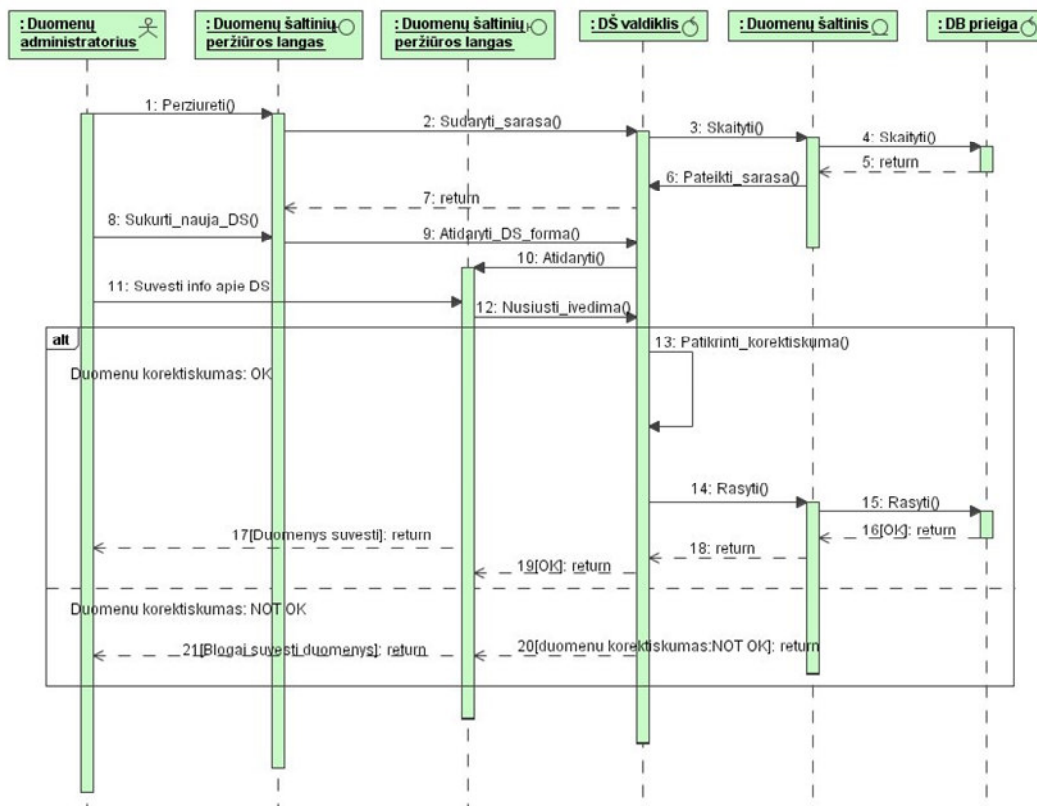
Pagrindinis aktorius „Administratorius“ pasirinkęs iš vartotojų meniu „Naujų anketų peržiūra“ atsidaro naujų anketų registravimo sąrašo langą. Sistema automatiškai iš duomenų bazės nuskaityti duomenis apie nepatvirtintas naujai užregistruotas anketas, ir pateikia sąrašo forma. Jeigu naujų anketų nėra, sistema pateikia sisteminių pranešimą apie jų nebuvimą. Pateiktame sąraše administratorius gali išsirinkti anketą ir pažymėjus ją peržiūrėti registracijos duomenis. Prieš užregistruojant naują vartotoją sistemos administratorius turi patikrinti anketos duomenis mygtuko paspaudimu, o sistema LR registro duomenų bazėje automatiškai patikrina duomenis apie subjekto egzistavimą ir sėkmės atveju leidžia patvirtinti naujo vartotojo sukūrimą.

27 paveiksle pavaizduota naujo vartotojo kūrimo būsenų diagrama. Naujas sistemos vartotojas turi užsiregistruoti sistemoje prieš pradėdamas naudotis sistema. Jis sukuria vartotojo anketą, kurios būseną „Nauja“. Sistemos administratoriui patikrinus kliento anketą, jos būseną pasikeičia į „Patikrinta“. Ši tarpinė būseną tarp „Nauja“ ir „Patvirtinta“ reikalinga saugumo spragų ir klaidų išvengimo minimizavimui ar visiškam panaikinimui. Jei duomenys buvo pateikti neteisingi, sistemos administratorius turi galimybę atmesti anketą, tuomet būseną pasikeičia į „Atmesta“.

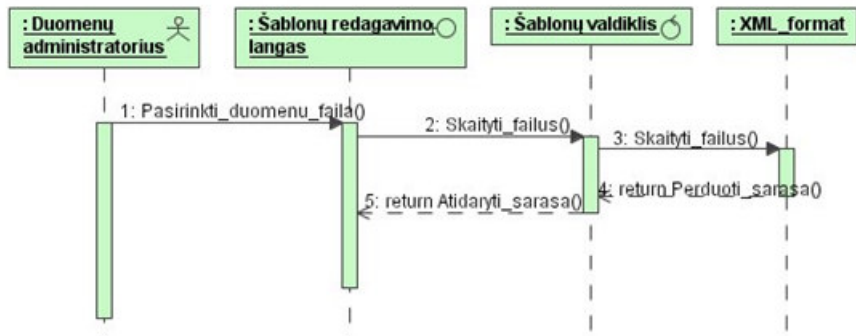


27 pav. Vartotojų anketos būsenų diagrama

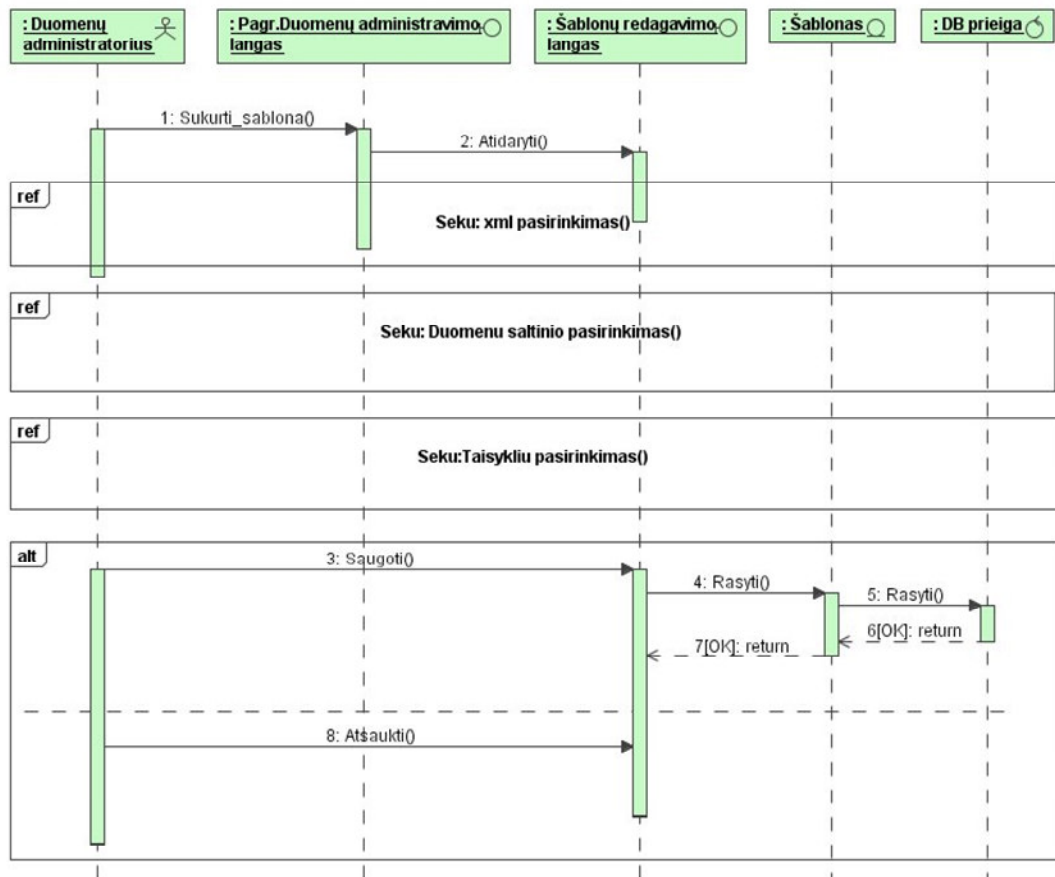
Panaudojimo atvejui „Sukurti šabloną“ išnagrinėti bus naudojama sudėtinė sekų diagrama (žr. 30 pav.), kurią sudarys nuorodos į kitų panaudojimo atvejų sekų diagramas (žr. 28 pav. ir 29 pav.)



28 pav. Panaudojimo atvejo „Užregistruoti duomenų šaltinį“ sekų diagrama

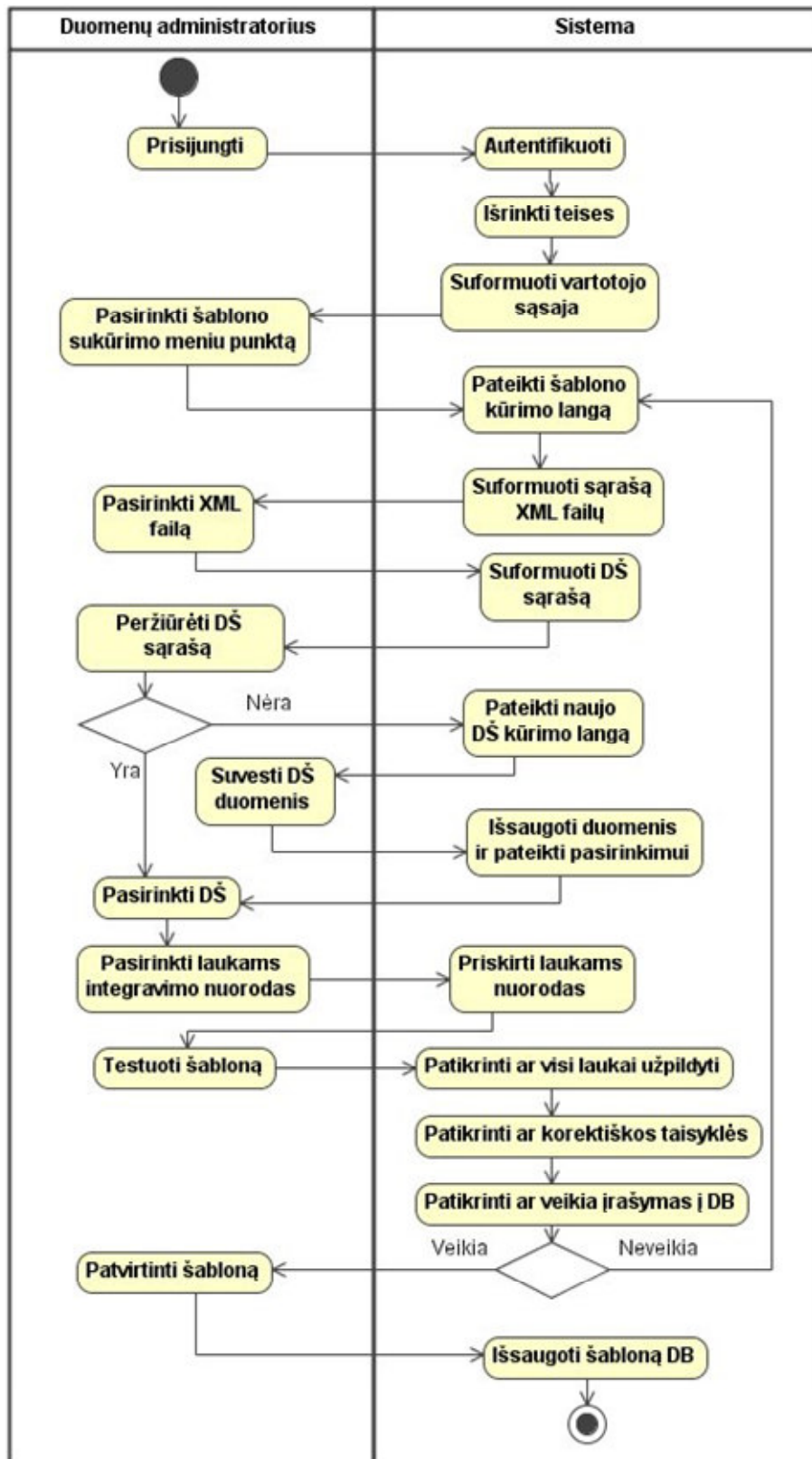


29 pav. XML pasirinkimo sekų diagrama



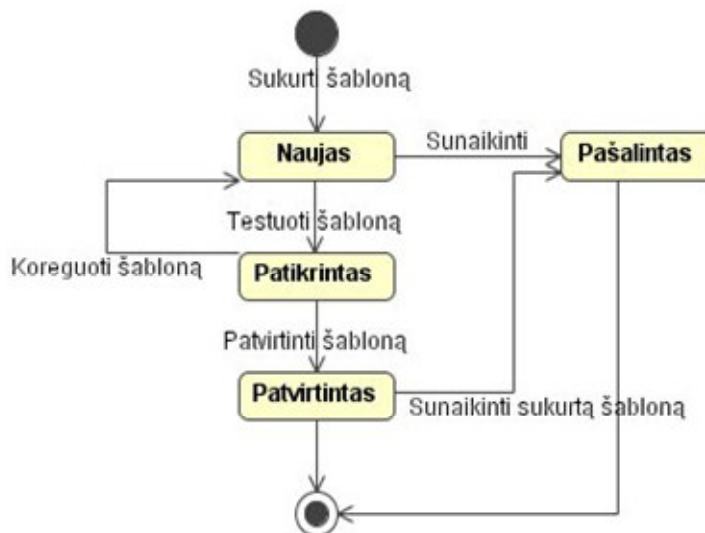
30 pav. Panaudojimo atvejo „Sukurti šabloną“ sekų diagrama

Pagrindinės, duomenų integravimo modulyje, šablono sudarymo funkcijos „sukurti šabloną“ veiklos diagrama pateikta 31 paveiksle.



31 pav. Veiklos diagrama panaudojimo atvejui „Sukurti šabloną“

Šablono būsenų kaita pateikta 32 paveiksle šablono būsenų diagramoje.

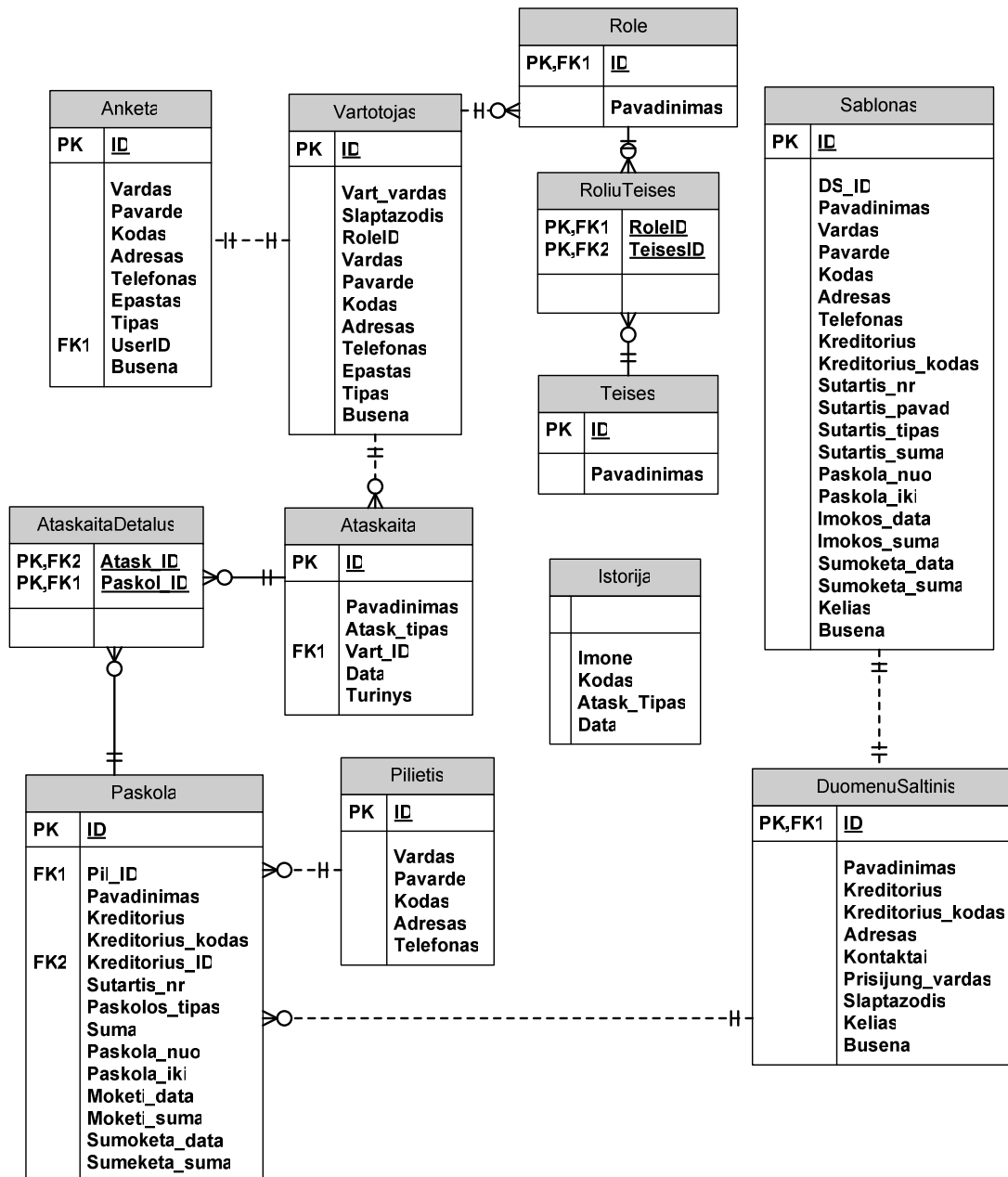


32 pav. Šablonų būsenų diagrama

Prisijungęs prie sistemos, šablono kūrimą inicijuoja duomenų administratorius, pasirinkdamas iš administravimo meniu punktą „Šablonų redagavimo langas“. Sistema atidaro šablonų redagavimo langą su nustatyta šablono būsena „Naujas“. Kurdamas naują šablona, duomenų administratorius turi prisilaikyti veiksmų eiliškumo, kuris pateiktas 31 paveiksle. Taigi pirmiausia iš sistemos neatpažintų failų sąrašo pasirenkamas duomenų failas, paskui priskiriamas duomenų šaltinis ir tik po to kiekvienam laukui yra nustatomos integravimo taisyklės, t.y. kokius duomenis iš gauto duomenų failo naudoti ir kur tuos duomenis talpinti. Atlikus visus nustatymo veiksmus, duomenų administratorius inicijuoja šablono testavimą, kurio metu sistema patikrina ar sudarytas šablonas veikia ir ar nieko netrūksta. Jei testavimo procesas pavykęs, šablono būsena pakeičiama į „Patikrintas“, jeigu nepavyko, būsena yra nekeičiama ir šablonas yra grąžinamas duomenų administratoriui duomenų koregavimui.. Koreguoti duomenis galima ir būsenoje „Patikrintas“, nes tik patvirtinus šablono išsaugojimą, sistema išsaugo šablono duomenis duomenų bazėje. Sukurtą ar nepatvirtintą šablona galima ištrinti bet kuriuo momentu, tuomet jo būsena pasikeičia į „pašalintas“ ir yra traktuojamas kaip neaktyvus (negaliojantis) šablonas.

2.3.4. Duomenų bazės modelis

Žemiau esančiame paveiksle (žr. 33 pav.) Microsoft Visio paketo pagalba sudaryta duomenų bazės modelio schema.



33 pav. Duomenų bazės modelis

Sudarytos duomenų bazės modelio diagramos laukų aprašymai yra pateikti Prieduose (žr. Priedas Nr.1).

2.3.5. Testavimo modelis bei testavimo atvejai

Žemiau esančioje 16 lentelėje aprašytas naudotojo sąsajos testavimo modelis.

16 lentelė. Scenarijus (sistemos funkcionalumo lygmenyje) pakartojamas kiekvienam langui

#	Objektas	Testavimo sąlygos, atvejis	Laukiami rezultatai
1.	Menui	Patikrinti pavadinimą, gramatiką, veikimą	Menui pavadinimas turi būti kuo aiškesnis ir trumpesnis
2.	Laukai	Patikrinti spalvas, dydžius, pavadinimus, gramatiką, stilių	Privalomi laukai yra pažymimi spalva. Nėra gramatinių klaidų. Korektiški laukų dydžiai (atsižvelgiant į maksimalų įvedamų simbolių ilgį)
3.	Mygtukai	Patikrinti pavadinimus, spalvas, dydžius, gramatiką, išdėstymą	Mygtuko pavadinimas – kuo trumpesnis.
4.	Duomenų užkrovimas	Patikrinti, ar užkraunami reikalingi duomenys	Pasirinkus norimą meniu punktą užkraunami reikalingo lango duomenys; pasirinkus konkretų įrašą užkraunami būtent to įrašo duomenys.
5.	Vientisumas	Patikrinti, ar visi langai, laukai, mygtukai ir kiti sistemos objektai realizuoti pagal vienodas taisykles, t.y. vienodą spalvą, tą patį veiksmą atliekančių mygtukų pavadinimų vienodumas ir pan., langų pavadinimų taisyklingumas ir vienodumas	Visoje sistemoje langų elementai išdėstyti laikantis tos pačios tvarkos, vienodos spalvos, tą patį veiksmą atliekančių mygtukų pavadinimai vienodi.

2.3.6. Reikalavimai sistemos funkcionavimo palaikymui

Sistema suprojektuota taip, kad funkcionuotų be sistemos projektuotojų palaikymo ir reikalautų minimalaus sistemos administratoriaus įsikišimo bei pastovaus sistemos priežiūros atlikimo iš duomenų administratoriaus pusės. Sistemoje užtikrintas saugumas, griežtai apibrėžtos kiekvieno sistemos vartotojo funkcijos ir jiems galimi atlikti veiksmai bei realizuotas galimų klaidų apdorojimas. Šios priemonės užtikrins sistemos funkcionavimą bet kuriuo metu. Pagrindiniai reikalavimai sistemos funkcionalumui yra susiję su technine įranga – taikomųjų programų ir duomenų bazių serveriais ir jų galimybe teikti paslaugas.

Serveriui

Programinė įranga: ASP.NET WEB serveris.

MSQL 2005 duomenų bazė valdymo sistema

Vartotojui (klientui)

Kompiuteris su Interneto naršykle (rekomenduojama IE 5 arba naujesnė)

2.4. Sistemos projektavimo išvados

Remiantis iškeltais reikalavimais, suprojektuota:

- a) loginė architektūra;
- b) klasių diagramos;
- c) sistemos elgsenos modelis, pateiktas sekų diagramomis;
- d) reliacinės duomenų bazės schema;
- e) sistemos komponentų modeliai;
- f) sistemos testavimo modelis.

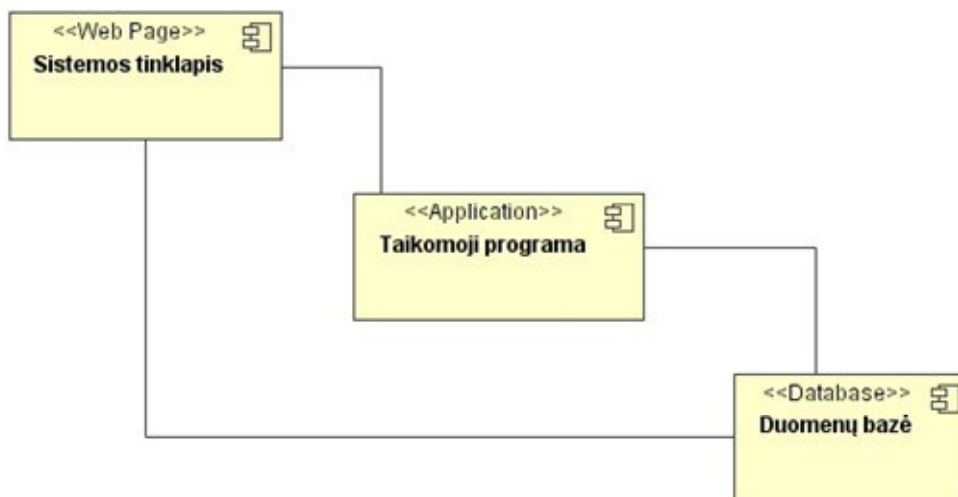
Remiantis projektavimo rezultatais bus sugeneruotas sistemos programinio kodo karkasas ir duomenų bazės SQL kodas.

3. SISTEMOS REALIZACIJA BEI EKSPERIMENTINIS TYRIMAS

Ekperimentinio tyrimo dalyje yra aprašytas duomenis integruojančios sistemos diegimas bei pateiktas vartotojo sąsajos testavimas - vadovas. Pateiktas kokybės kriterijų tyrimas ir išvados.

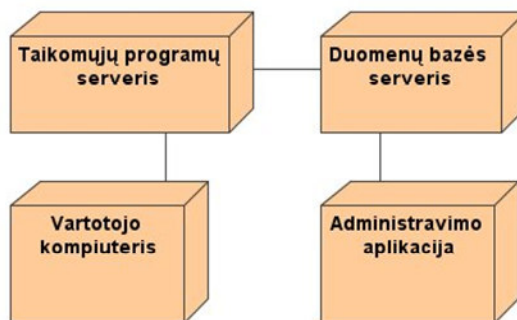
3.1. Realizacijos modelis

Komponentų diagrama (34 pav.) rodo sistemos programinės įrangos architektūrą – fizinių sistemos vaizdą.



34 pav. Sistemos komponentų diagrama

Įdiegimo diagramoje (35 pav.) atvaizduojami sistemos techninės įrangos architektūra - procesoriai, kurių pagalba sistema veikia.



35 pav. Sistemos įdiegimo diagrama

3.2. Duomenis apie piliečių kreditus integruojančios sistemos testavimas

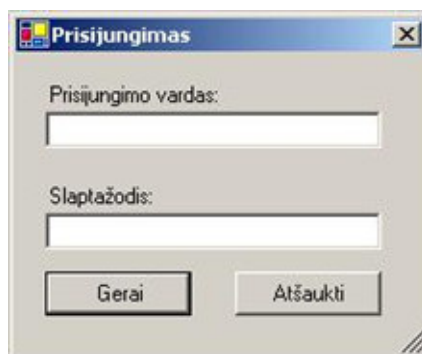
Kadangi duomenis apie piliečių kreditus integravimo sistemos pilna realizacija susideda iš daug modulių: internetinės prieigos, sistemos administravimo, duomenų integravimo, ekspertinės sistemos, apmokėjimų valdymo, saugumo valdymo modulių, tai buvo pasirinkta realizuoti ir iširti tik duomenų integravimo procesą realizuojantį modulį ir su juo susijusius administravimo ir internetinės prieigos modulius.

Sistemos vartotojo sąsajos realizaciją galima suskirstyti į dvi pagrindines dalis: internetinę sąsają ir aplikaciją – vartotojo taikomąją programą. Sekančiuose skyreliuose bus pateiktas vartotojo vadovas testavimo metodu t.y. kaip naudotis sistema, kokie yra langai ir kokia jų paskirtis.

3.2.1. Administratoriaus sąsajos testavimas

Prisijungimas prie sistemos

Administratorius prie sistemos prisijungti gali tik per programą, kuri turi būti įdiegta į kompiuterį. Paleidus programą atsidaro prisijungimo langas (36 pav.), kuriame reikia įvesti prisijungimo duomenis: vartotojo vardą ir slaptažodį.



36 pav. Administratoriaus programos prisijungimo langas

Jei kuris nors iš reikalaujamų duomenų įvedamas blogai, arba nustatoma, kad vartotojas neturi tinkamų sistemos administravimo teisių, tada išvedamas pranešimas apie klaidą (žr. 37 pav.).



37 pav. Sisteminis klaidos pranešimas

Teisingai suvedus prisijungimo duomenis atsidaro pagrindinis sistemos langas su kontekstiniu meniu (žr. 38 pav.).

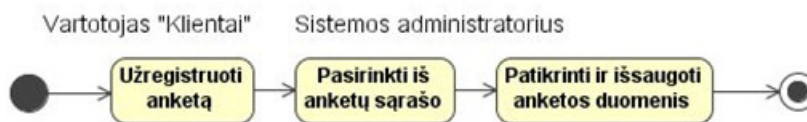


38 pav. Pagrindinis sistemos administravimo langas

Vartotojų administravimas

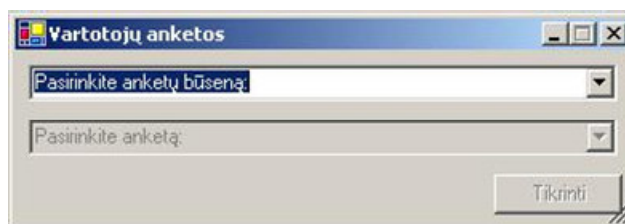
Sistemos administratorius turi teises sukurti, patikrinti ir pašalinti naujus vartotojus ir sistemos vartotojų roles. Tam tikslui yra skirti meniu punktai „Vartotojai“ ir „Rolės“.

Naujo vartotojo sukūrimo veiksmų seka yra tokia (žr. 39 pav.):




39 pav. Naujo vartotojo sukūrimo veiksmų seka

Vartotojas „Klientai“ per internetinę sąsają turi sukurti ir užregistruoti „naujo vartotojo registravimo anketą“. O sistemos administratorius prisijungęs prie sistemos ir pasirinkęs meniu punktą „Vartotojai“ ir „Anketų peržiūra“ atsidariusiame anketų peržiūros lange gali pasirinkti naujų anketų peržiūrą (žr. 40 pav.). Visų pirma, pasirenkama anketos būseną („Nauja“ – jeigu tai naujai užregistruota anketa; „Atmesta“ – jeigu anketa dėl kažkokių priežasčių buvo atmesta; „Patikrinta“ – anketa kuri jau yra praėjusi duomenų patikrinimo etapą ir laukia patvirtinimo), nes pagal anketos būseną yra filtruojamos ir atrenkamos anketos, kurių sąrašas pateikiamas sekančiame pasirinkimo lauke.



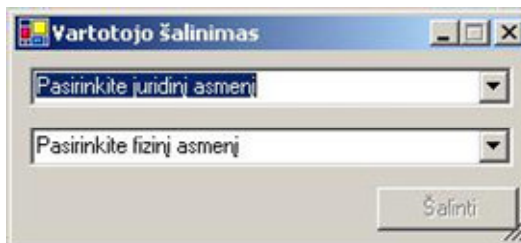
40 pav. Naujų anketų peržiūros langas

Naujo vartotojo sukūrimo langas pateiktas 41 paveiksle.



41 pav. Naujo vartotojo sukūrimo langas

Vartotojų šalinimo langas pateiktas 42 paveiksle.



42 pav. Vartotojo šalinimo langas

Naujos rolės sukūrimo langas pateiktas 43 paveiksle. Rolės kūrimo langą sudaro: pavadinimo laukas, sistemos vartotojų teisių sąrašas ir kuriamos rolės priskirtų teisių sąrašo laukas.

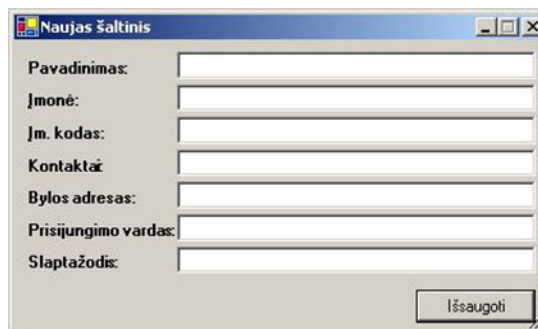


43 pav. Naujos rolės sukūrimo langas

Vartotojų ir rolių redagavimo langai yra identiški jų sukūrimo langams. Tačiau atsidarius vartotojų ar rolių redagavimo langus juose duodama pasirinkti iš sąrašo kuriuos objektus norima redaguoti, o pasirinkus juos sistemos logika nuskaito duomenis ir užpildo atitinkamus laukus atitinkama informacija. Mygtuko „Išsaugoti“ paspaudimu patvirtinama redagavimo pabaiga ir duomenys išsaugomi duomenų bazėje.

Duomenų šaltinių administravimas

Duomenų šaltinius turi teisę suvesti tik duomenų administratorius. Naujo duomenų šaltinio sukūrimo langas pateiktas 44 paveiksle.



44 pav. Naujo duomenų šaltinio sukūrimo langas

Duomenų šaltinių šalinimo langas pateiktas 45 paveiksle. Norint pašalinti duomenų šaltinį reikia iš pateikto sąrašo išsirinkti norimą pašalinti duomenų šaltinį ir paspausti mygtuką „Šalinti“.



45 pav. Duomenų šaltinio šalinimo langas

Sistemos integravimo paleidimas(rankiniu būdu)

Kadangi šiame darbe kuriamas tik sistemos prototipas, tai sistema neturi pilno jai reikalingo funkcionalumo. Todėl sistemos duomenų integracijos procesas yra neautomatizuotas, o atliekamas rankiniu būdu. Pasirinkus meniu punktą „Kreditai“, atsidaro integravimo langas, kuriame galima pasirinkti šaltinius iš kurių bus integruojama informacija.

Duomenų importavimo langas pavaizduotas 46 paveiksle.



46 pav. Duomenų integravimo langas

Pirmame lauke galima pasirinkti kokius šaltinius importuosime, sekantis elementas yra importavimo statuso eilutė, kuri rodo operacijos atlikimo statusą. O apatiniame lauke rodomas rezultatas: ar operacijos pavyko, ar ne.

Šablonų sukūrimas ir redagavimas, sunaikinimas

Tik duomenų administratoriaus teises turintis vartotojas gali administruoti sistemos integravimo šablonus. Iš meniu punkto „Šablonai“ pasirenkamas punktas „Naujas“ ir atidaromas naujo šablono sukūrimo langas (žr. 47 pav.)

47 pav. Naujo šablono sukūrimo langas

Naujo šablono kūrimo lange yra išdėstyti tokie elementai:

- Šablono pavadinimo laukas, kuris užpildomas ranka;
- Bylos katalogo ir bylos pasirinkimo laukai, kurie yra pasirenkami mygtuko „Pasirinkti“ pagalba;
- Duomenų šaltinio pasirinkimo laukas, kuriame yra suformuotas egzistuojančių duomenų šaltinių sąrašas, o jeigu sąrašė nėra reikiamo šaltinio, mygtuko „Naujas šaltinis“ pagalba iškviečiamas naujo šaltinio sukūrimo langas;
- Laukų – XML žymių suderinimo laukai. Kairėje lango pusėje yra surašyti reikalingi – būtini laukai, kuriems reikia nustatyti integravimo taisykles pasirenkant iš XML failo arba pagal žymes, arba pagal duomenis.

Nustatant integravimo taisykles būtina atidžiai peržiūrėti kuri žymė kokią informaciją aprašo ir ką reiškia, kad išvengti netikslios informacijos patekimo į sistemos duomenų bazę.

O šablonų redagavimo langas (žr. 48 pav.) yra analogiškas naujų šablonų kūrimo langui, tik, kad šitam lange pirmiausia reikia pasirinkti, kuri šabloną redaguosime iš sąrašo ir sistemos logika automatiškai pateiks jau sudaryto šablono visą turinį: integravimo taisykles, šaltinį ir t.t.

48 pav. Šablono redagavimo langas

Šablonų sunaikinimo langas pavaizduotas 49 paveiksle:

49 pav. Šablono sunaikinimo langas

Paspaudus mygtuką “Šalinti” sistema pakeičia šablonų būseną į “Pašalintas” ir šabloną neberodo tarp aktyvių sistemos duomenų apdorojimo šablonų.

3.2.2. Internetinės prieigos vartotojo sąsajos testavimas

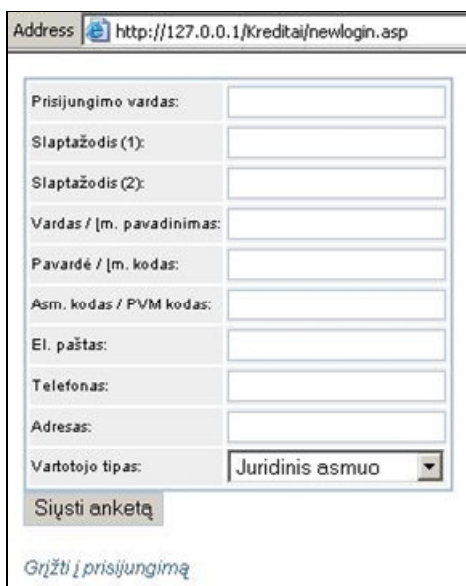
Internetinės vartotojo sąsajos paskirtis yra leisti vartotojui prisijungti prie sistemos per interneto naršyklę ir atlikti tik jam leidžiamus veiksmus : peržiūrėti istoriją, generuoti ataskaitas, atlikti paiešką. Klientai yra suskirstyti į dvi rūšis: juridiniai ir fiziniai asmenys.

Sekančiame paveiksle (žr. 50 pav.) pateiktas pradinis prisijungimo langas, kuriame, kad prisijungti prie sistemos reikia suvesti vartotojo prisijungimo duomenis.



50 pav. Internetinės vartotojo sąsajos prisijungimo langas

Jei vartotojas yra neužsiregistravęs sistemoje ir neturi prisijungimo duomenų, jis gali užsiregistruoti sistemoje paspaudęs mygtuką „Naujas prisijungimas“. Paspaudus mygtuką, sistema atidaro naujo vartotojo registravimo anketą (žr. 51 pav.).



51 pav. Naujo vartotojo registravimo anketos langas

Suvedus duomenis ir paspaudus mygtuką „Siųsti anketą“ anketa yra nusiunčiama sistemos administratoriui, kad jis patikrintų anketos duomenis ir aktyvuotų naują vartotoją.

Prisijungus vartotojui prie sistemos, sistema pagal jam suteiktas teises pateikia, vartotojo meniu (žr. 52 pav.).



52 pav. Pagrindinis vartotojo langas

Vartotojo sąsajos langas sudarytas iš dviejų dalių, kairės pusės, kuriame pateikiamas vartotojo meniu ir pagrindinio lango, kuriame pateikiama informacija ir pasirinkimo objektai.

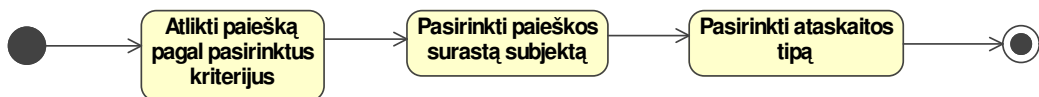
Pasirinkus punktą „Jūsų informacija“, sistema pateikia vartotojo informacijos redagavimo langą (žr. 53 pav.), kuriame vartotojas gali atlikti koregavimo veiksmus. O paspaudus mygtuką „Išsaugoti“ sistema išsaugo korektiškus pakeitimus. Jei pakeitimai nekorektiški, sistema apie pateiktus nekorektiškus duomenis praneša ir paprašo pakartotinai pakoreguoti įvestą informaciją.



53 pav. Vartotojo asmeninės informacijos redagavimo langas

Pasirinkus meniu punktą „Ataskaitos“ priklausomai nuo vartotojui suteiktų teisių sistema atidaro ataskaitų tipų pasirinkimo langą.

Juridinio asmens ataskaitų pasirinkimo veiksmų eiga (žr. 54 pav.):



54 pav. Vartotojo veiksmų sekos langas

Pasirinkus meniu punktą „Ataskaitos“, sistema vartotojui pateikia paieškos langą (žr. 55 pav.), kuriame galima atlikti piliečių paiešką sistemoje pagal jų vardą, pavardę ar asmens kodą. Būtina sąlyga įvesti mažiausiai 3 simbolius į paieškos kriterijų laukus.

55 pav. Duomenų paieškos langas

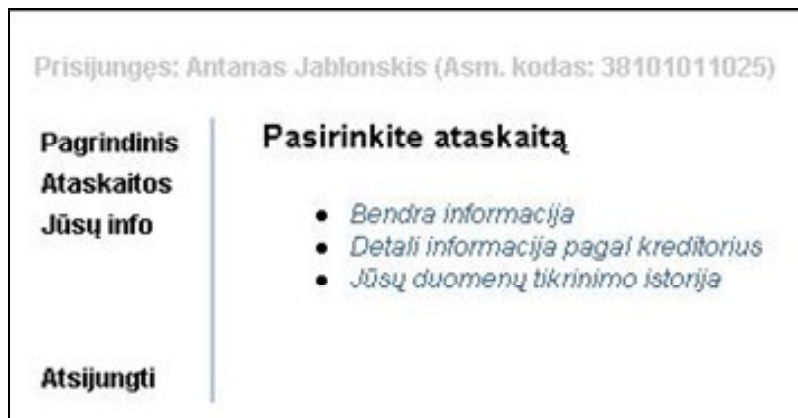
Paspaudus mygtuką „Ieškoti“, sistema pagal nurodytus kriterijus atlieka paiešką sistemos duomenų bazėje ir pateikia rezultatus (žr. 56 pav.).

56 pav. Paieškos rezultatų langas

Pasirinkus vieną iš pateiktų paieškos rezultatų, sistema atidaro ataskaitų tipų pasirinkimo langą (žr. 57 pav.), kuriame iš pateikto sąrašo galima pasirinkti norimą peržiūrėti ataskaitą ir sistema ją atvaizduos ekrane.

57 pav. Ataskaitų pasirinkimo langas

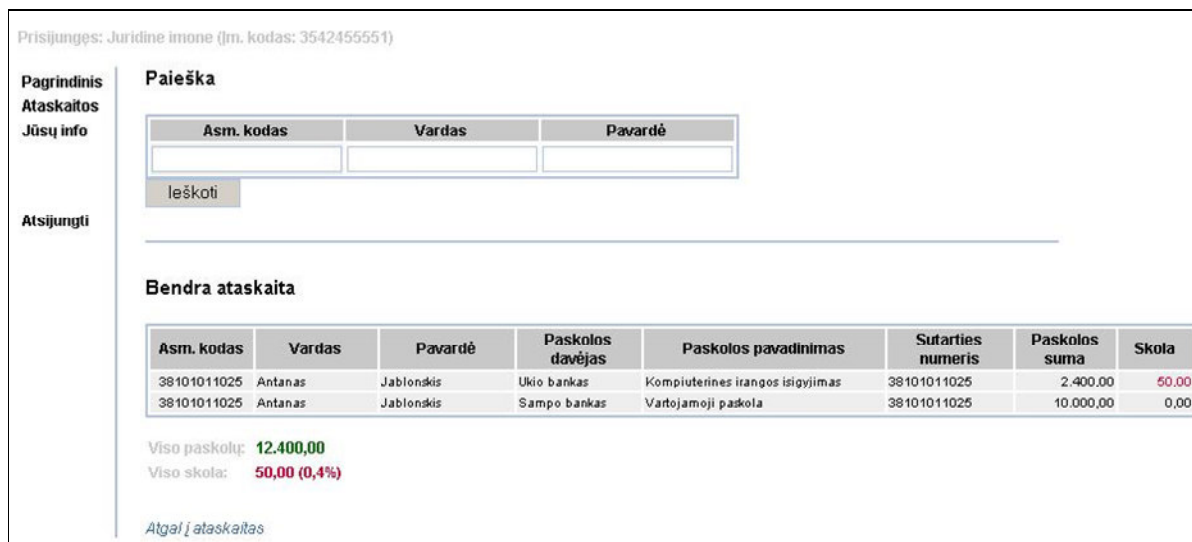
Fizinio asmens ataskaitų pasirinkimo langas pavaizduotas 58 paveiksle. Fizinis asmuo neturi teisių naudoti paieškos, todėl jam paieškos langas yra nerodomas, o pasirinkus meniu punktą „Ataskaitos“ iš karto pateikiamas ataskaitų tipų pasirinkimo langas.



58 pav. Fizinio asmens ataskaitų pasirinkimo langas

Pasirinkus meniu punktą „Atsijungti“ vartotojas yra atjungiamas nuo sistemos.

Sistemos testavimui buvo sukurtos 4 tipų ataskaitos: „Bendra informacija“, „Detali informacija pagal kreditorius“, „Jūsų duomenų tikrinimo istorija“, „Mokumo išvada“. Kiekvienas vartotojas gali peržiūrėti tik jam skirtas ataskaitas. Sugeneruotų ataskaitų pavyzdžiai pateikti sekančiuose paveiksluose (žr. 59 pav.- 62 pav.):



59 pav. Ataskaita: „Bendra informacija“

Prisijungęs: Juridine imone (Im. kodas: 3542455551)

Pagrindinis Ataskaitos Jūsų info

Paieška

Asm. kodas	Vardas	Pavardė
<input type="text"/>	<input type="text"/>	<input type="text"/>

leškoti

Atsijungti

Detali ataskaita pagal kreditorius

Kreditorius:

Pasirinktas: **Antanas Jablonskis (38101011025)**

Sutarties numeris	Paskolos pavadinimas	Paskolos tipas	Paskolos pradžia	Paskolos pabaiga	Paskolos suma	Mokėjimo data	Mokėjimo suma	Imokos data	Imokos suma
38101011025	Kompiuterines irangos isigyjimas	Lizingas	2007.11.01	2008.11.01	2.400,00	2007.12.01	200,00	2007.11.28	200,00
38101011025	Kompiuterines irangos isigyjimas	Lizingas	2007.11.01	2008.11.01	2.400,00	2008.01.01	200,00	2007.12.29	150,00

Viso įmokų: **2**
 Pavėluotų įmokų: **0 (0%)**
 Visų įmokų suma: **400,00**
 Nepilnų įmokų suma: **50,00 (12,50%)**

[Atgal į ataskaitas](#)

60 pav. Ataskaita: „Detali informacija pagal kreditorius“

Prisijungęs: Juridine imone (Im. kodas: 3542455551)

Pagrindinis Ataskaitos Jūsų info

Paieška

Asm. kodas	Vardas	Pavardė
<input type="text"/>	<input type="text"/>	<input type="text"/>

leškoti

Atsijungti

Mokumo išvada

Antanas Jablonskis (38101011025)

Klientas turi skolų.

Patikimumas: Nepatikimas.

[Atgal į ataskaitas](#)

61 pav. Ataskaita: „Mokumo išvada“

Prisijungęs: Antanas Jablonskis (Asm. kodas: 38101011025)

Pagrindinis Ataskaitos Jūsų info

Jūsų duomenų tikrinimo istorija

Data	Ataskaita	Imonė
2008.01.06 21:24:15	Detali informacija pagal kreditorius	UAB "kasta"
2008.01.06 21:24:10	Detali informacija pagal kreditorius	UAB "kasta"
2008.01.06 20:31:41	Bendra infomacija	Juridine imone

Atsijungti

[Atgal į ataskaitas](#)

62 pav. Ataskaita: „Jūsų duomenų tikrinimo istorija“

3.3. Sukurtos sistemos kokybės tyrimas

Šiame darbe realizuotos duomenų apie piliečių kreditus integravimo sistemos dalys: sistemos vartotojų administravimo ir duomenų administravimo posistemiai, internetinės prieigos posistemis. Todėl vertinama ne visos sistemos, o tik realizuotų posistemų kokybė.

17 lentelėje pateikti pagrindiniai sistemos veikimui, duomenims bei funkcijoms išskelti reikalavimai ir pagrįstas sistemos atitikimas jiems.

17 lentelė. Sistemos atitikimo reikalavimams įvertinimas

Reikalavimas	Atitikimo reikalavimui pagrindimas
Reikalavimai saugumui	Reikalavimas tenkinamas, nes tenkinami nustatyti reikalavimai slaptažodžio saugojimui, vartotojų teisių nustatymui ir tikrinimui, prieigos prie sistemos automatizavimui.
Reikalavimai suprantamumui	Reikalavimas tenkinamas, nes pagal atliktus kriterijų vertinimus, vartotojo sąsaja buvo įvertinta kaip paprasta ir lengvai suprantama
Reikalavimai sistemos patikimumui	Reikalavimas tenkinamas, nes sistema atlieka visus jai skirtus uždavinius.
Sistema turi būti lengvai plečiama	Reikalavimas tenkinamas, nes į sistemą galima lengvai įdiegti papildomus modulius ir komponentus.
Reikalavimai efektyvumui	Reikalavimas tenkinamas, nes sistema atliekant veiksmus, pastoviai tikrina atminties būklę ir naikina nenaudojamus objektus ir procesus.
Reikalavimai sistemos patikimumui	Reikalavimas tenkinamas, nes sistemos palaikymui yra naudojama naujausia techninė įranga ir galimybė naudotis ja bet kuriuo metu.
Reikalavimai sąveikai su kitomis sistemomis	Reikalavimai dalinai įgyvendinti, nes nėra realios galimybės išbandyti sistemos darbo su kitomis sistemomis (bankais, registrų centru ir t.t.)

Iš sistemos atitikimo reikalavimams įvertinimų matome, kad sistema atitinka visus jai pateiktus reikalavimus.

Sistemos analizės dalyje buvo nagrinėti metodai duomenų integravimui, bet visi šie metodai netenkina šios sistemos, todėl buvo sukurtas tik šiai sistemai skirtas duomenų integravimo metodas naudojant šablonus ir integravimo taisykles. Jo kokybiškam veikimui

įvertinti naudojamas „juodos dėžės“ metodas. Duomenų integravimo įvertinimas pateiktas 18 lentelėje.

18 lentelė. Sistemos atitikimo reikalavimams įvertinimas

Pateikti duomenys	Teorinis rezultatas	Gautas rezultatas
<data>	Vytautas	Vytautas
<name>Vytautas	Kurklius	Kurklius
<surname>Kurklius	35209011145	35209011145
<personid>35209011145	Tundros g. 10, Alytus	Tundros g. 10, Alytus
<address>Tundros g. 10, Alytus	861132132	861132132
<phone>861132132	Union kredito unija	Union kredito unija
<bank>Union kredito unija	Kompiuterinės	Kompiuterinės
<loaname>Kompiuterinės	įrangos įsig.	įrangos įsig.
<type>Lizingas	Lizingas	Lizingas
<amount>1000.00	1000.00	1000.00
<loanfrom>2007.08.20	2007.08.20	2007.08.20
<loantill>2007.12.20	2007.12.20	2007.12.20
<frequency>12	2007.09.20	2007.09.20
<validity>>true	250.00	250.00
<payments>	2007.09.18	2007.09.18
<datetill>2007.09.20	250.00	250.00
<datetopay>250		
<paymentdate>2007.09.28		
<amountpaid>250		
</payments>		
</data>		

Duomenų integravimo posistemio įvertinimo metu nustatyta, kad sistemos posistemis duomenis parenka tiksliai pagal aprašytas integravimo taisykles. Todėl jis yra tinkamas naudoti finansinių duomenų integravimui.

3.4. Tolimesnio sistemos tobulinimo, plėtojimo galimybės

Sukurti duomenų apie piliečių kreditus integravimo sistemos modelis ir realizacija yra prototipiniai. Norint realiai naudoti šią sistemą reikėtų jos funkcionalumą patobulinti ir išplėsti. Galimos tokios išplėtimo galimybės:

- *Automatinis vartotojų registracijos duomenų patikrinimas.* Šiuo metu sistemoje fizinių ar juridinių asmenų užregistruotus duomenis bei tokių asmenų egzistavimą tikrina sistemos administratorius. Jis rankiniu būdu patikrina apie tokių duomenų korektiškumą ir sukuria naujus vartotojus, bei prisijungimą prie sistemos. Šį procesą galima atlikti automatiškai. Tam tikslui, reikėtų įdiegti bendradarbiavimą su Fizinių ir Juridinių asmenų registrų duomenų bazėmis.
- *Ekspertinės sistemos naudojimas duomenų analizei.* Sistemai funkcionuoti pagal paskirtį t.y. teikti finansinę riziką įvertinančią informaciją, būtina sukurti ekspertinę sistemą analizuojančią duomenis apie piliečius, kuri iš daugybės kriterijų ir skirtingų duomenų galėtų tiksliai įvertinti finansinės rizikos laipsnį.
- *Atsiskaitymų valdymo modulis finansinėms operacijoms atlikti.* Tam, kad sistema taptų funkcionali ir pelninga, sistemos teikiamos paslaugos (duomenų analizę, ataskaitų peržiūrą) turi būti mokamos. Tam tikslui, reikia sukurti internetinio atsiskaitymo ir automatinių apmokėjimų sekimo ir valdymo modulį.
- *Duomenų integracija iš įvairių sistemų,* kurios padėtų tiksliau įvertinti piliečių mokėjimo įpročius ir riziką. Taip pat duomenų integravimo modulis pritaikytas, kad į sistemą būtų galima greitai ir lengvai integruoti duomenis iš komunalines paslaugas ir kitas paslaugas teikiančių įmonių DB. Tokiu būdu praplečiamos sistemos rizikos įvertinimo galimybės.
- *Įvairių ataskaitų generavimo galimybė.* Sistemos funkcionalumui praplėsti, galima sukurti daugiau įvairių tipų ataskaitų, jų pateikimo formų.
- *Duomenų banko rezultatų integravimo į vartotojų biznio sistemas.* Praplėtus sistemos funkcionalumą, galima sukurti modulį, kuris į įvairių draudimo kompanijų, internetinių parduotuvių, e-bankų informacines sistemas integruotų sistemos išanalizuotą finansinę informaciją, taip sutaupant daug laiko ir pinigų savo klientams analizuojant jų klientų mokumo riziką.

IŠVADOS IR REZULTATAI

1. Išanalizavus Lietuvoje ir pasaulyje sukurtų panašių tarpininkavimo sistemų privalumus ir trūkumus, nustatyta, kad tokios universalios Lietuvos rinkai pritaikytos, piliečių finansinių duomenų surinkimo iš skirtingų šaltinių ir finansavimo rizikos nustatymo, sistemos nėra. Įvertinus tokių sistemų teikiamą naudą kitose valstybėse, galima teigti, kad Lietuvai tokia sistema reikalinga.
2. Išanalizavus esamus technologinius sprendimus, naudojamus duomenų lygio informacinių sistemų integracijai atlikti, nustatyta, kad universalių įrankių šių sistemų duomenims integruoti nėra.
3. Išanalizavus EDI ir XML technologijų galimybes duomenų integracijai atlikti, darbe pasirinktas XML formatas, nes tai labiausiai tarp Lietuvos įmonių paplitęs duomenų apsikeitimo formatas, kuriam yra sukurta daug įvairių duomenis apdorojančių įrankių.
4. UML diagramų pagalba, atlikta organizacijos veiklos analizė ir sudaryta vartotojo reikalavimų bei poreikių specifikacija.
5. Atsižvelgiant į užsiduotus reikalavimus, suprojektuota:
 - a) sistemos loginė architektūra;
 - b) sistemos elgsenos modelis, pateiktas sekų diagramomis;
 - c) reliacinės duomenų bazės schema;
 - d) sistemos komponentų ir diegimo modeliai;
 - e) sistemos testavimo modelis.
6. Pagal sudarytą duomenų integravimo sistemos modelį, buvo sukurta prototipinė sistema, kuri yra realizuota VB.NET ir SQL programavimo kalbomis, panaudojant Microsoft SQL 2005 duomenų bazės valdymo sistemą.
7. Duomenų integracijos problemai sistemoje spręsti naudojamas sukurtas XML dokumentų integravimo metodas sudarytas iš šablonų ir integravimo taisyklių.
8. Naudojant administratoriaus ir paprasto vartotojo sąsajas, sėkmingai atliktas sistemos testavimas ir nustatyta, kad sistema tenkina visus užsiduotus funkcinis ir nefunkcinis reikalavimus.
9. Sukurta sistema optimizuoja klientų mokumą ir finansinę riziką vertinančių institucijų darbą, tačiau ji nenaudinga, nepatikimiems ir nemokiems jų klientams.

LITERATŪROS SĄRAŠAS

1. Chuck Phillips and Mary Meeker. B2B roadmap – Report. Iš Morgan Stanley Dean Witter. [interaktyvus]. 2000, balandis [žiūrėta 2006-10-07]. Prieiga per internetą: <http://www.msdw.com/institutional/eEnterpriseSoftware/194.html>
2. David S. Linthicum. What XSLT Brings to the Application Integration Party. EAI Journal, August 2001, p. 12 – 15.
3. David S. Linthicum. Enterprise Application Integration [interaktyvus]. O'Reilly 1999 Lapkritis [žiūrėta 2006-10-20]. Prieiga per internetą: <http://safari.oreilly.com/0201615835>
4. Dongwon Lee, Wesley W. Chu. Comparative Analysis of Six XML Schema Languages [interaktyvus]. 2000, birželis [žiūrėta 2006-09-10]. Prieiga per Internetą: <http://www.cobase.cs.ucla.edu/tech-docs/dongwon/ucla-200008.html>
5. Don Awalt, Brian Lawton. Data Warehousing: Back to Basics. Iš Penton Media Inc. padalinio Windows IT Pro [interaktyvus]. 2000, sausis [žiūrėta 2006-04-08]. Prieiga per internetą: <http://www.winnetmag.com/SQLServer/Issues/IssueID/367/367.html>
6. Elliotte Rusty Harold. Processing XML with Java: A Guide to SAX, DOM, JDOM, JAXP and TrAX. 2003 ISBN: 0-201-77186-1.
7. Equifax. Įmonės Equifax internetinis puslapis [interaktyvus]. 2006 Kovas. [žiūrėta 2007-11-08]. Prieiga per internetą: http://www.equifax.com/ecm/docs/2006_annual_report.pdf
8. IBM alpha Works. XML Data Mediator [interaktyvus]. 2002 rugsėjis. [žiūrėta 2006-09-10]. Prieiga per Internetą: <https://secure.alphaworks.ibm.com/tech/XI>
9. Igor Dayen. Storing XML in Relational Databases [interaktyvus]. 2001 birželis. [žiūrėta 2006-12-01]. Prieiga per Internetą: <http://www.xml.com/pub/a/2001/06/20/databases.html>
10. Infobankas. Įsteigus Kredito biurą sąžiningi piliečiai ir įmonės lengviau gaus paskolas. [interaktyvus] 2006 lapkritis. [žiūrėta 2007-10-07]. Prieiga per internetą: <http://infobankas.lt/?PageID=623&NewsID=1197>
11. John Lepton. UML istorija [interaktyvus]. 2000, spalio [žiūrėta 2006 10 27]. Prieiga per internetą: <http://cgi.omg.org/news/pr97/umlprimer.html>.
12. JP Morgenthal. XML for Data Integration. EAI Journal, October 2001, p. 13 – 16
13. Jon Peterson. Data Warehousing (Data Warehouse). Iš Information Builders. [interaktyvus]. 2001, birželis [žiūrėta 2006-10-02]. Prieiga per internetą: http://www.informationbuilders.com/products/webfocus/cm_factsheet.html

14. Joshua Fox, Ph. D. Active Information Models for Data Transformation. EAI Journal, May 2003, pages 26 – 30.
15. Kas yra XML? [interaktyvus]. 2003 [žiūrėta 2006-01-15]. Prieiga per internetą: <http://www.xml.lt/#document>
16. Lietuvos laisvosios rinkos institutas. Pastabos ir pasiūlymai dėl Finansų institucijų įstatymo projekto (IXP-514) [interaktyvus]. 2001, gegužė [žiūrėta 2007-10-08]. Prieiga per internetą: <http://www.lrinka.lt/Tyrimai/Imones/Fininstitutijos.phtml>
17. Michael F. Jennings. Enterprise Architecture View: The Complete Meta Data Repository. Iš DM Review. [interaktyvus]. 2004, birželis [žiūrėta 2006-10-08]. Prieiga per internetą: http://www.dmreview.com/article_sub.cfm?articleId=1000941
18. Michael Gorman. Comprehensive Metadata Management. The Data Administration Newsletter. [interaktyvus]. 2003, vasaris [žiūrėta 2006-09-28]. Prieiga per internetą: <http://www.tdan.com/i025fe03.htm>
19. Nexus Consulting Group. Transforming Data Into Information. Iš ITtoolbox Data Warehouse.[interaktyvus].2002, kovas[žiūrėta 2006-09-28]. Prieiga per internetą: <http://datawarehouse.ittoolbox.com/browse.asp?c=DWPeerPublishing&r=%2Fpub%2FAG022702%2Epdf>
20. Rajan Jindal and Abhishek Acharya. Federated Data Warehouse Architecture. Iš ITtoolbox Data Warehouse. [interaktyvus]. 2004, gegužė [žiūrėta 2006-10-08]. Prieiga per internetą: <http://datawarehouse.ittoolbox.com/browse.asp?c=DWPeerPublishing&r=http%3A%2F%2Fhosteddocs%2Eittoolbox%2Ecom%2FFederated+data+Warehouse+Architecture%2Epdf>
21. Roger Clarke. Electronic Data Interchange (EDI): An Introduction. [interaktyvus] 1998, Gruodis [žiūrėta 2006-10-08]. Prieiga per internetą: <http://www.anu.edu.au/people/Roger.Clarke/EC/EDIIntro.html>
22. Wikipedia. Loan [interaktyvus]. 2007, spalio [žiūrėta 2007-10-08]. Prieiga per internetą: <http://en.wikipedia.org/wiki/Loan>
23. World spy. Angliškų frazių bei akronimų aiškinamasis žodynas [interaktyvus]. 2000 gegužė [žiūrėta 2007-10-12]. Prieiga per Internetą: <http://www.wordspy.com/words/B2B2C.asp>
24. W3C[®]. W3C Technical Reports and Publications [interaktyvus] [žiūrėta 2006-10-08]. Prieiga per Internetą: <http://www.w3.org/TR/>

Darbo formatavimui naudota literatūra:

25. Viktorija Baršauskienė, Irena Mačerinskienė. Studijų darbų parengimo tvarka. KTU 2003, Kaunas. ISBN 9955-09-064-2

TERMINŲ IR SANTRUMPŲ ŽODYNAS

19 lentelė. Terminai ir santrumpos

Sutrumpinimas	Paiškinimas
DB	(<i>angl. data base</i>) Duomenų bazė.
IS	(<i>angl. Information System</i>) Informacijos sistema.
UML	(<i>angl. Unified Modeling Language</i>) Unifikuota modeliavimo kalba
RUP	(<i>angl. Rational Unified Process</i>) Sistemų projektavimo procesas
ASP.NET	(<i>angl. Active Server Pages</i>) Dinaminių ir interaktyvių internetinių puslapių kūrimo technologija
VB.NET	(<i>angl. Visual Basic</i>) Visual Basic .NET programavimo kalba
PHP	(<i>angl. Hypertext Preprocessor</i>) HTML paremta programavimo kalba
HTML	(<i>angl. Hyper Text Markup Language</i>) internetinių puslapių programavimo kalba
DBVS	Duomenų bazių valdymo sistema
SQL	(<i>angl. Structured Query Language</i>) Struktūrizuota užklausų kalba.
CASE	(<i>angl. Computer-Aided System Engineering</i>) Automatizuotas projektavimas
EAI	(<i>angl. Enterprise Application Integration</i>) Organizacijų kompiuterinių sistemų integracija
B2B	(<i>angl. Business-to-Busines</i>) Verslas-verslui integravimas
B2C	(<i>angl. Business-to-Customer</i>) Verslas-klientui integravimas
B2B2C	(<i>angl. Business-to-Busines-to-Customer</i>) Verslas-verslui-klientui integravimas
DOM	(<i>angl. Document Object Model</i>) Dokumento objektinis modelis
EDI	(<i>angl. Electronic Data Interchange</i>) Elektroninis apsikeitimas duomenimis
DTD	(<i>angl. Document type definition</i>) Dokumento tipų aprašymas
CSS	(<i>angl. Cascading Style Sheets</i>) Stilių lentelės
XML	(<i>angl. eXtensible Markup Language</i>) Išplėstinė žymių kalba
XSLT	(<i>angl. eXtensible Stylesheet Language for Transformations</i>) Išplečiama stilių kalba transformacijoms
XPath	Kalba skirta identifikuoti XML dokumento dalis
XQuery	Duomenų paieškos XML dokumentuose užklausų kalba

SANTRAUKA ANGLŲ KALBA

Citizen Credits Data Integration System

Summary

This graduation paper researches the principles of integrating data from many different sources into one electronic collaboration system. It overlooks what technologies and methods may be used for data level Enterprise Application Integration.

In this paper was discovered and analyzed the main problems (data format, structure and semantics) of integrating different data resources. To solve those problems the method of using special patterns and integration rules was represented. This method was implemented in the specific business sphere – finance sector. For this purpose was designed and realized information system - Citizen Credits Data Integration System. This system gathers coded XML documents with citizen data about their credits and payments from different financial institutions, analyses and transforms XML documents into the main data format used in the system DB.

System project was built using CASE technologies. User requirements and demands represented using UML notation. Also it was overlooked the possibility to expand the systems functions by adding web interface for system clients, information analyzing knowledge module, payment for services module, the possibility to integrate public utility organizations clients payments data to give more accurate results on clients financial riskiness.

PRIEDAI

1 priedas. Uždavinio techninis išpildymas

1. Specifikacijos panaudojimo atvejams

Kiekvienam panaudojimo atvejui pateikiamos specifikacijos, kurios nebuvo pateiktos darbe (žr. 20-26 lentelės).

20 lentelė. Panaudojimo atvejo „Sukurti rolę“ specifikacija

Panaudojimo atvejis	„Sukurti Rolę“
Aktorius	Administratorius
Sistema	Duomenų apie piliečių kreditus integravimo sistema
Prieš sąlyga	Administratorius turi turėti prisijungimo prie sistemos teisę ir žinoti prisijungimo slaptažodį.
Pagrindinis įvykių srautas	Sistemos reakcija ir sprendimai
1. Įvedamas prisijungimo slaptažodis 2. Pasirenkamas meniu punktas „Rolės“ 3. Pasirenkamas meniu punktas „Nauja“. 4. Pavadinimo lauke įvedamas rolės pavadinimas ir pasirenkamos sistemos teisės kuriamai rolei. 5. Spaudžiamas mygtukas „Išsaugoti“.	1. Sistema tikrina, ar teisingai įvestas slaptažodis; 2. Sistema atidaro „Rolės“ kontekstinį meniu. 3. Sistema pateikia vartotojui naujos rolės sukūrimo langą, kuris susideda iš pavadinimo lauko ir teisių pasirinkimo laukų; 5. Paspaudus mygtuką „Išsaugoti“ sistema išsaugo duomenų bazėje naujos sukurtos rolės duomenis. Uždaromas rolės sukūrimo langas.
Po sąlyga	Naujai įvesti pakeitimai saugomi duomenų bazėje
Alternatyvos (nesėkmės atvejai)	-
Vykdyto variantai	
Veiklos taisyklės	Vartotojas turi pasirinkti korektiškus duomenis
Specialūs reikalavimai (nefunkciniai)	

21 lentelė. Panaudojimo atvejo „Redaguoti rolę“ specifikacija

Panaudojimo atvejis	„Redaguoti Rolę“
Aktorius	Administratorius
Sistema	Duomenų apie piliečių kreditus integravimo sistema
Prieš sąlyga	Administratorius turi turėti prisijungimo prie sistemos teisę ir žinoti prisijungimo slaptažodį.
Pagrindinis įvykių srautas	Sistemos reakcija ir sprendimai
1. Įvedamas prisijungimo slaptažodis 2. Pasirenkamas meniu punktas „Rolės“ 3. Pasirenkamas meniu punktas „Redaguoti“. 4. Pasirenkama rolė, kurios duomenis norima redaguoti. Ir atliekami koregavimo veiksmai. 5. Spaudžiamas mygtukas „Išsaugoti“.	1. Sistema tikrina, ar teisingai įvestas slaptažodis; 2. Sistema atidaro „Rolės“ kontekstinį meniu. 3. Sistema pateikia vartotojui rolės redagavimo langą, kuriame reikia pasirinkti kurią rolę norima redaguoti; 4. Pasirinkus rolę sistema nuskaito tos rolės duomenis iš duomenų bazės ir pateikia į rolės redagavimo langą. 5. Paspaudus mygtuką „Išsaugoti“ sistema išsaugo duomenų bazėje rolės duomenų pakeitimus. Uždaromas rolės redagavimo langas.

Po sąlyga	Naujai įvesti pakeitimai saugomi duomenų bazėje
Alternatyvos (nesėkmės atvejai)	-
Vykdyto variantai	
Veiklos taisyklės	Vartotojas turi pasirinkti korektiškus duomenis
Specialūs reikalavimai (nefunkciniai)	

22 lentelė. Panaudojimo atvejo „Redaguoti vartotoją“ specifikacija

Panaudojimo atvejis	„Redaguoti Vartotoją“
Aktorius	Administratorius
Sistema	Duomenų apie piliečių kreditus integravimo sistema
Prieš sąlyga	Administratorius turi turėti prisijungimo prie sistemos teisę ir žinoti prisijungimo slaptažodį.
Pagrindinis įvykių srautas	Sistemos reakcija ir sprendimai
1. Įvedamas prisijungimo slaptažodis 2. Pasirenkamas meniu punktas „Vartotojai“ 3. Pasirenkamas meniu punktas „Redaguoti“. 4. Pasirenkama vartotojas, kurio duomenis norima redaguoti. Ir atliekami koregavimo veiksmai. 5. Spaudžiamas mygtukas „Išsaugoti“.	1. Sistema tikrina, ar teisingai įvestas slaptažodis; 2. Sistema atidaro „Vartotojai“ kontekstinį meniu. 3. Sistema pateikia vartotojui vartotojo redagavimo langą, kuriame reikia pasirinkti kurį vartotoją norima redaguoti; 4. Pasirinkus vartotoją sistema nuskaito to vartotojo duomenis iš duomenų bazės ir pateikia į vartotojo redagavimo langą. 5. Paspaudus mygtuką „Išsaugoti“ sistema išsaugo duomenų bazėje vartotojo duomenų pakeitimus. Uždaromas vartotojo redagavimo langas.
Po sąlyga	Naujai įvesti pakeitimai saugomi duomenų bazėje
Alternatyvos (nesėkmės atvejai)	-
Vykdyto variantai	
Veiklos taisyklės	Vartotojas turi pasirinkti korektiškus duomenis
Specialūs reikalavimai (nefunkciniai)	

23 lentelė. Panaudojimo atvejo „Redaguoti duomenų šaltinį“ specifikacija

Panaudojimo atvejis	„Redaguoti duomenų šaltinį“
Aktorius	Duomenų administratorius
Sistema	Duomenų apie piliečių kreditus integravimo sistema
Prieš sąlyga	Administratorius turi turėti prisijungimo prie sistemos teisę ir žinoti prisijungimo slaptažodį.
Pagrindinis įvykių srautas	Sistemos reakcija ir sprendimai
1. Įvedamas prisijungimo slaptažodis 2. Pasirenkamas meniu punktas „Šaltiniai“ 3. Pasirenkamas meniu punktas „Redaguoti“. 4. Pasirenkama šaltinis, kurio duomenis norima redaguoti. Ir atliekami koregavimo veiksmai. 5. Spaudžiamas mygtukas „Išsaugoti“.	1. Sistema tikrina, ar teisingai įvestas slaptažodis; 2. Sistema atidaro „Šaltiniai“ kontekstinį meniu. 3. Sistema pateikia vartotojui duomenų šaltinio redagavimo langą, kuriame reikia pasirinkti kurį duomenų šaltinį norima redaguoti; 4. Pasirinkus šaltinį sistema nuskaito to duomenų šaltinio duomenis iš duomenų bazės ir pateikia į duomenų šaltinio redagavimo langą. 5. Paspaudus mygtuką „Išsaugoti“ sistema išsaugo duomenų bazėje redaguojamo duomenų šaltinio pakeitimus. Uždaromas duomenų šaltinio redagavimo langas.

Po sąlyga	Naujai įvesti pakeitimai saugomi duomenų bazėje
Alternatyvos (nesėkmės atvejai)	-
Vykdyto variantai	
Veiklos taisyklės	Vartotojas turi pasirinkti korektiškus duomenis
Specialūs reikalavimai (nefunkciniai)	

24 lentelė. Panaudojimo atvejo „Redaguoti šabloną“ specifikacija

Panaudojimo atvejis	“Redaguoti šabloną”
Aktorius	Duomenų administratorius
Sistema	Duomenų apie piliečių kreditus integravimo sistema
Prieš sąlyga	Administratorius turi turėti prisijungimo prie sistemos teisę ir žinoti prisijungimo slaptažodį.
Pagrindinis įvykių srautas	Sistemos reakcija ir sprendimai
1. Įvedamas prisijungimo slaptažodis	1. Sistema tikrina, ar teisingai įvestas slaptažodis;
2. Pasirenkamas meniu punktas „Šablonas“	2. Sistema atidaro „Šablonai“ kontekstinį meniu.
3. Pasirenkamas meniu punktas „Redaguoti“.	3. Sistema pateikia vartotojui šablono redagavimo langą, kuriame reikia pasirinkti kurį šabloną norima redaguoti;
4. Pasirenkamas šablonas, kurio duomenis norima redaguoti. Ir atliekami koregavimo veiksmai.	4. Pasirinkus šabloną sistema nuskaito to šablono duomenis iš duomenų bazės ir pateikia į šablono redagavimo langą.
5. Spaudžiamas mygtukas „Testuoti“.	5. Paspaudus mygtuką „Testuoti“ sistema testuoja pakoreguotą duomenų šabloną ir jeigu testavimas OK, tada sistema leidžia išsaugoti pakoreguotus duomenis.
6. Spaudžiamas mygtukas „Išsaugoti“.	6. Paspaudus mygtuką „Išsaugoti“ sistema išsaugo duomenų bazėje redaguojamo šablono pakeitimus. Uždaromas šablono redagavimo langas.
Po sąlyga	Naujai įvesti pakeitimai saugomi duomenų bazėje
Alternatyvos (nesėkmės atvejai)	-
Vykdyto variantai	
Veiklos taisyklės	Vartotojas turi pasirinkti korektiškus duomenis
Specialūs reikalavimai (nefunkciniai)	

25 lentelė. Panaudojimo atvejo „Redaguoti prisijungimo duomenis“ specifikacija

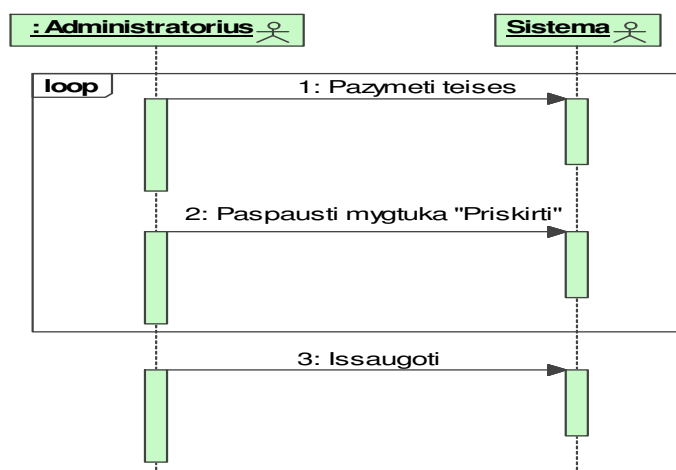
Panaudojimo atvejis	“Redaguoti prisijungimo duomenis”
Aktorius	Klientai
Sistema	Duomenų apie piliečių kreditus integravimo sistema
Prieš sąlyga	Klientas turi turėti prisijungimo prie sistemos teisę ir žinoti prisijungimo slaptažodį.
Pagrindinis įvykių srautas	Sistemos reakcija ir sprendimai
1. Įvedamas prisijungimo slaptažodis	1. Sistema tikrina, ar teisingai įvestas slaptažodis;
2. Pasirenkamas meniu punktas „Jūsų info“.	2. Sistema pateikia vartotojui prisijungimo duomenų redagavimo langą, ir nuskaito iš DB kliento duomenis, kuriuos surašo į laukus redagavimo procesui atlikti;
3. Atliekami redagavimo veiksmai ir spaudžiamas mygtukas „Išsaugoti“.	3. Paspaudus mygtuką „Išsaugoti“ sistema išsaugo duomenų bazėje redaguojamų prisijungimo duomenų pakeitimus.
Po sąlyga	Naujai įvesti pakeitimai saugomi duomenų bazėje
Alternatyvos (nesėkmės atvejai)	-

Vykdymo variantai	
Veiklos taisyklės	Vartotojas turi pasirinkti korektiškus duomenis
Specialūs reikalavimai (nefunkciniai)	

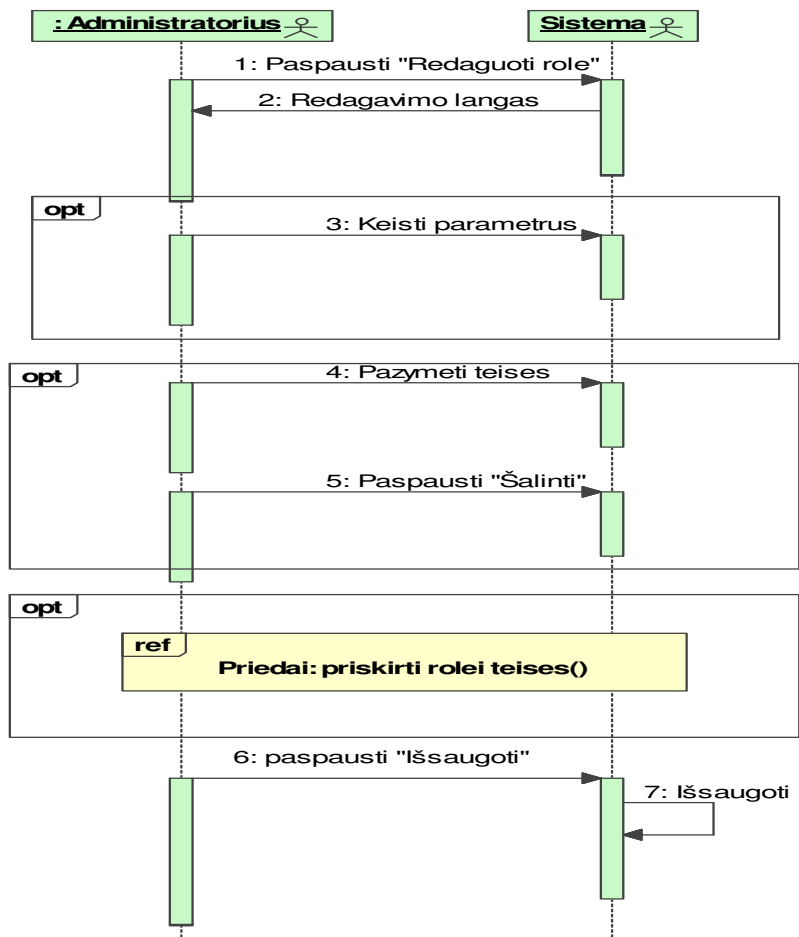
26 lentelė. Panaudojimo atvejo „Prisijungti prie sistemos“ specifikacija

Panaudojimo atvejis	„Prisijungti prie sistemos“
Aktorius	Klientai
Sistema	Duomenų apie piliečių kreditus integravimo sistema
Prieš sąlyga	Klientas turi turėti prisijungimo prie sistemos teisę ir žinoti prisijungimo slaptažodį.
Pagrindinis įvykių srautas	Sistemos reakcija ir sprendimai
1. Atidaromas pradinis vartotojo sąsajos langas su prisijungimo prie sistemos langu. 2. Įvedamas prisijungimo slaptažodis 3. Spaudžiamas mygtukas „Prisijungti“.	1. Sistema atidaro prisijungimo langą. 3. Sistema tikrina, ar teisingai įvestas vartotojas ir slaptažodis; a) jeigu suvesti duomenys yra teisingi, sistema nuskaito autentifikuoto vartotojo teises ir atitinkamai pagal vartotojo turimas teises pateikia Kliento vartotojo sąsajos meniu ir langus. b) jeigu įvestas blogas slaptažodis ar prisijungimo vardas, sistema pateikia sisteminių pranešimą apie blogai įvestus duomenis ir prašo pakartotinai įvesti juos iš naujo. Toliau atliekami veiksmai nuo 1 punkto.
Po sąlyga	Vartotojas prisijungia prie sistemos
Alternatyvos (nesėkmės atvejai)	-
Vykdymo variantai	
Veiklos taisyklės	Vartotojas turi įvesti korektiškus duomenis
Specialūs reikalavimai (nefunkciniai)	

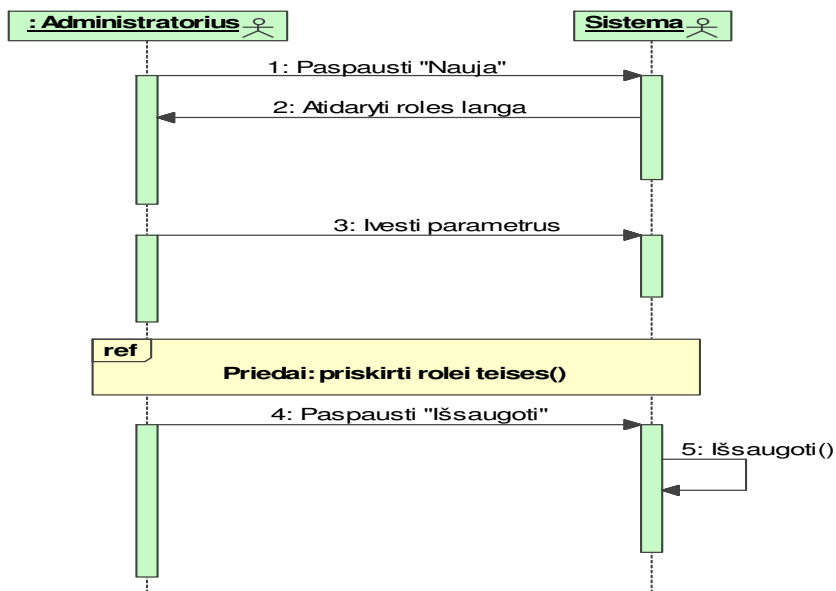
2. Sistemos panaudojimo atvejų sekų diagramos



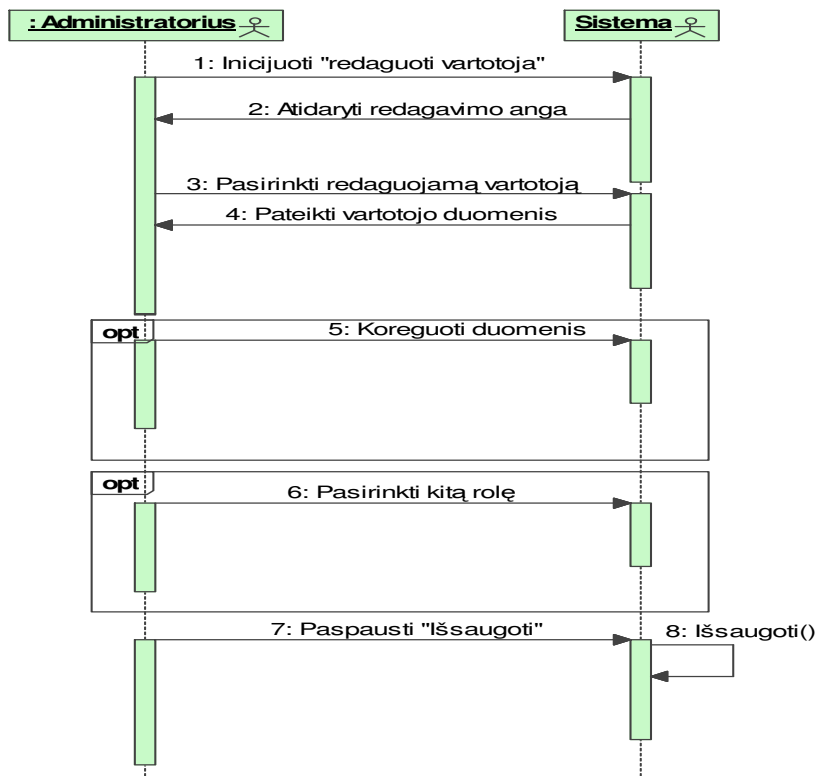
63 pav. „Priskirti rolei teises“ sekų diagrama



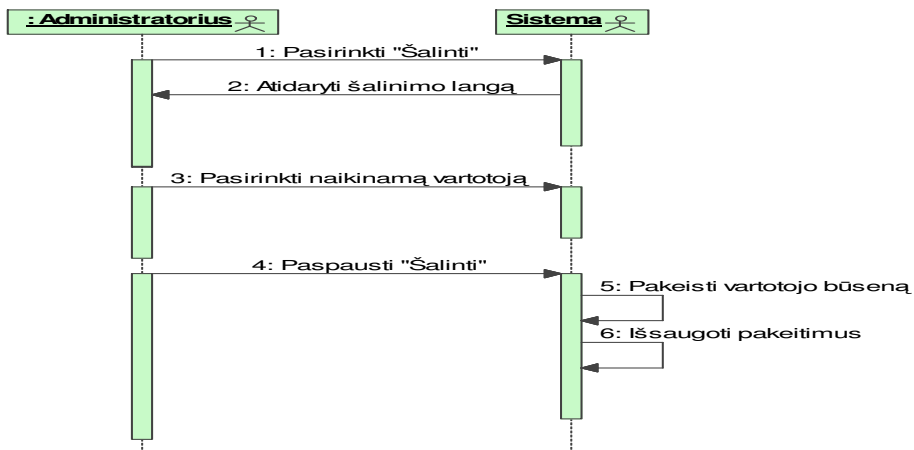
64 pav. „Redaguoti rolę“ sekų diagrama



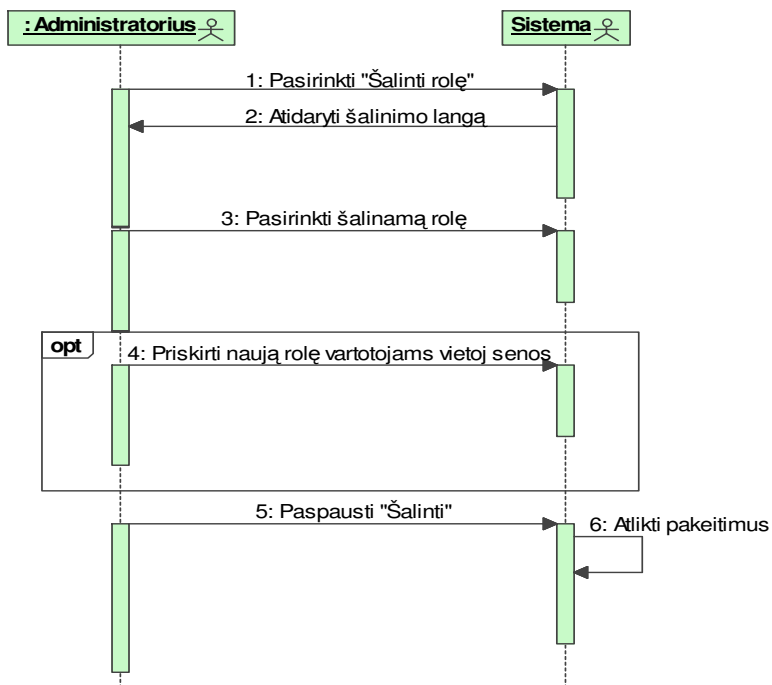
65 pav. „Sukurti rolę“ sekų diagrama



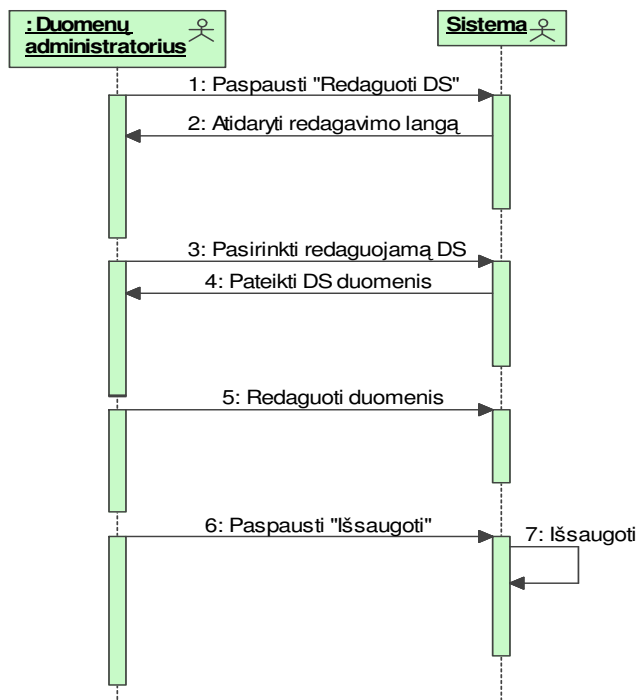
66 pav. „Redaguoti vartotoją“ sekų diagrama



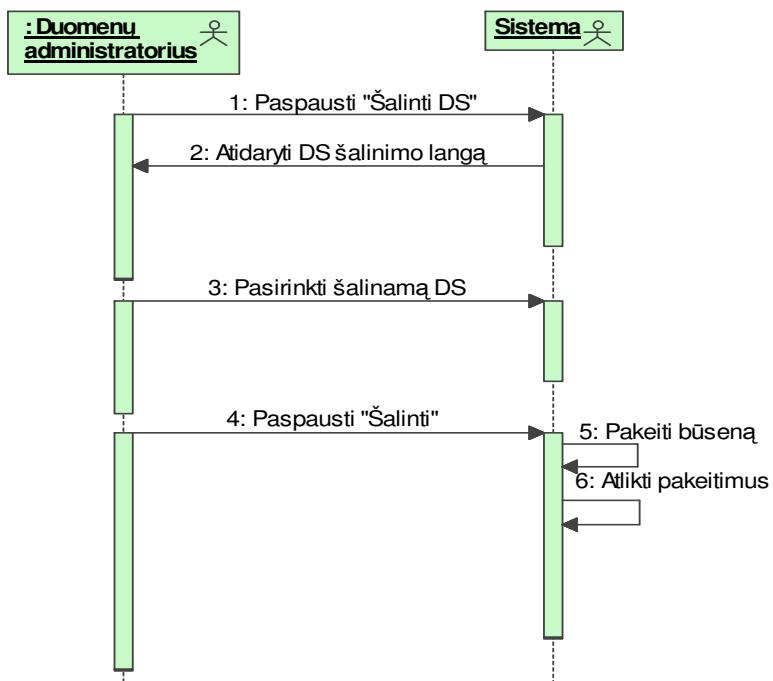
67 pav. „Šalinti vartotoją“ sekų diagrama



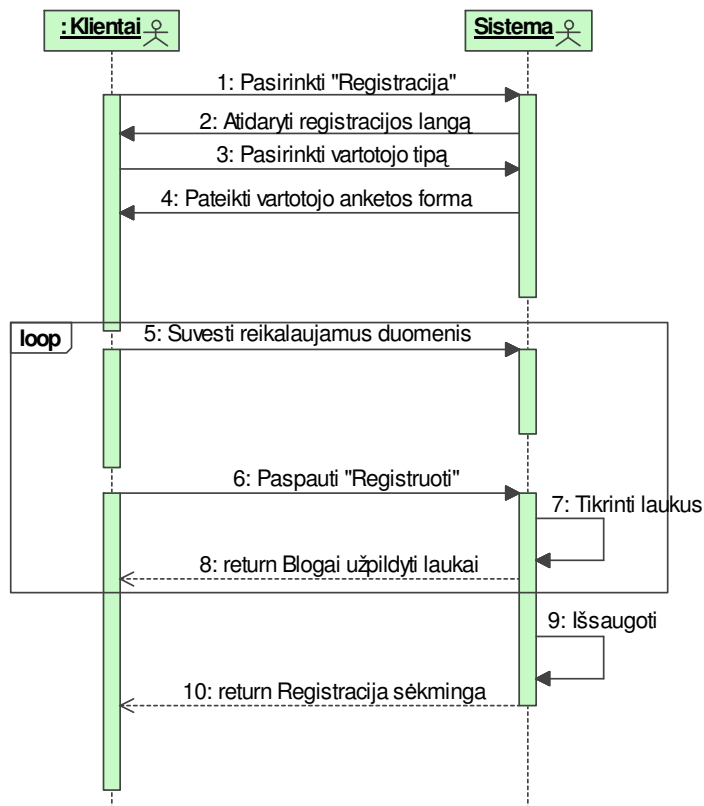
68 pav. „Šalinti rolę“ sekų diagrama



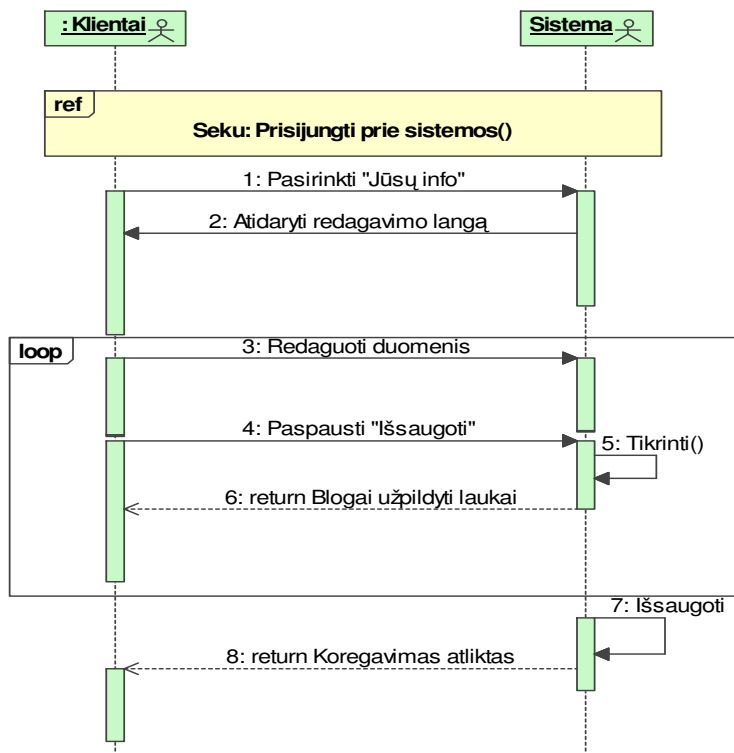
69 pav. „Redaguoti duomenų šaltinį“ sekų diagrama



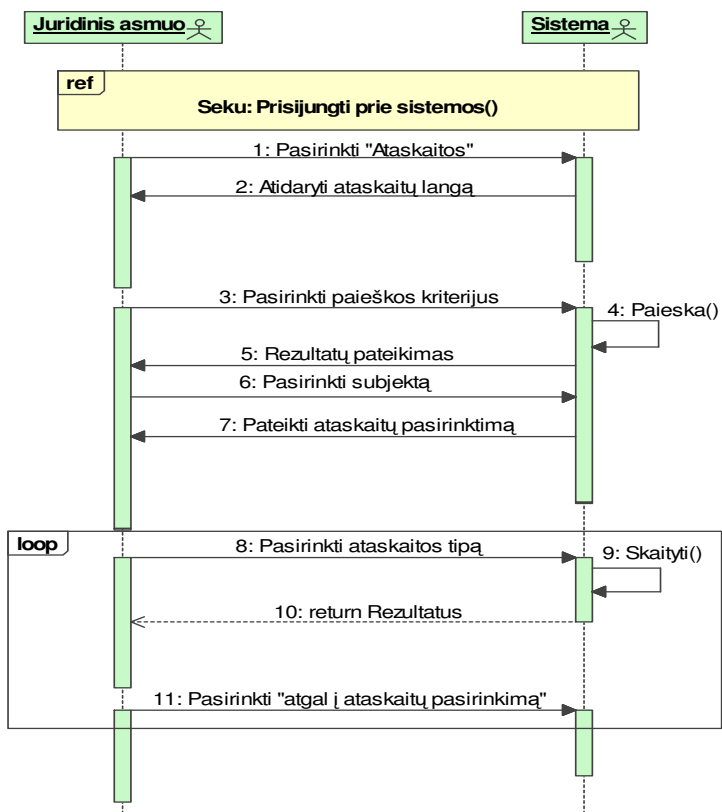
70 pav. „Šalinti duomenų šaltinį“ sekų diagrama



71 pav. „Susikurti vartotoją“ sekų diagrama



72 pav. „Redaguoti prisijungimo duomenis“ sekų diagrama



73 pav. „Peržiūrėti ataskaitas“ sekų diagrama

3. Duomenų bazės lentelių aprašai

Žemiau pateiktose lentelėse (žr. 27-37 lentelės) pateikta duomenų bazės lentelių struktūrų aprašymai.

27 lentelė. „Vartotojas“ lentelės struktūra

Lentelė: Vartotojas		
Lauko pavadinimas	Tipas	Aprašymas
ID	Bigint	Vartotojo identifikacinis numeris (PK)
Vart_vardas	Varchar(15)	Vartotojo prisijungimo prie sistemos vardas
Slaptažodis	Varchar(15)	Vartotojo slaptažodis
RoleID	Bigint	Rolės identifikacinis numeris
Vardas	Varchar(20)	Vartotojo vardas
Pavarde	Varchar(20)	Vartotojo pavardė
Kodas	Varchar(10)	Vartotojo asmens kodas
Adresas	Varchar(20)	Vartotojo adresas
Telefonas	Varchar(20)	Vartotojo telefonas
Epastas	Varchar(50)	Vartotojo elektroninis paštas
Tipas	Char(1)	Vartotojo tipas : fizinis ar juridinis asmuo
Busena	Char(1)	Vartotojo būseną (naujas, patvirtintas, pasalintas)

28 lentelė. „Anketa“ lentelės struktūra

Lentelė: Anketa		
Lauko pavadinimas	Tipas	Aprašymas
ID	Bigint	Anketos identifikacinis numeris (PK)
UserID	Bigint	Vartotojo identifikacinis numeris (FK)
Vardas	Varchar(20)	Anketos vartotojo vardas
Pavarde	Varchar(20)	Anketos vartotojo pavardė
Kodas	Varchar(10)	Anketos vartotojo asmens kodas
Adresas	Varchar(20)	Anketos vartotojo adresas
Telefonas	Varchar(20)	Anketos vartotojo telefonas
Epastas	Varchar(50)	Anketos vartotojo elektroninis paštas
Tipas	Char(1)	Anketos vartotojo tipas : fizinis ar juridinis asmuo
Busena	Char(1)	Anketos būseną (nauja, patvirtinta, pasalinta)

29 lentelė. „Role“ lentelės struktūra

Lentelė: Role		
Lauko pavadinimas	Tipas	Aprašymas
ID	Bigint	Rolės identifikacinis numeris (PK)
Pavadinimas	Varchar(20)	Rolės pavadinimas

30 lentelė. „RoliuTeises“ lentelės struktūra

Lentelė: RoliuTeises		
Lauko pavadinimas	Tipas	Aprašymas
RoleID	Bigint	Rolės identifikacinis numeris (FK)
TeisesID	Bigint	Teisės identifikacinis numeris (FK)

31 lentelė. „Teises“ lentelės struktūra

Lentelė: Teises		
Lauko pavadinimas	Tipas	Aprašymas
ID	Bigint	Teisės identifikacinis numeris (PK)
Pavadinimas	Varchar(20)	Teisės pavadinimas

32 lentelė. „Ataskaita“ lentelės struktūra

Lentelė: Ataskaita		
Lauko pavadinimas	Tipas	Aprašymas
ID	Bigint	Ataskaitos identifikacinis numeris (PK)
Vart_ID	Bigint	Vartotojo identifikacinis numeris (FK)
Pavadinimas	Varchar(20)	Ataskaitos pavadinimas
Atask_tipas	char(1)	Anketos tipas
Data	Date	Ataskaitos sukūrimo data
Turinys	Varchar(300)	Ataskaitos turinys

33 lentelė. „AtaskaitaDetalus“ lentelės struktūra

Lentelė: AtaskaitaDetalus		
Lauko pavadinimas	Tipas	Aprašymas
Atask_ID	Bigint	Ataskaitos identifikacinis numeris (FK)
Paskol_ID	Bigint	Paskolos identifikacinis numeris (FK)

34 lentelė. „Paskola“ lentelės struktūra

Lentelė: Paskola		
Lauko pavadinimas	Tipas	Aprašymas
ID	Bigint	Paskolos identifikacinis numeris (PK)
Pil_ID	Bigint	Piliečio identifikacinis numeris (FK)
Pavadinimas	Varchar(20)	Paskolos pavadinimas
Kreditorius	Varchar(20)	Kreditoriaus pavadinimas
Kreditorius_kodas	Varchar(10)	Kreditoriaus įmonės kodas
Kreditorius_ID	Bigint	Kreditoriaus identifikacinis numeris (FK)
Sutartis_nr	Varchar(50)	Paskolos sutarties numeris
Paskolos_tipas	Varchar(50)	Paskolos tipas
Suma	numeric(18, 2)	Bendra paskolos suma
Paskola_nuo	Varchar(50)	Paskotos termino pradžia (išdavimo data)
Paskola_iki	Varchar(50)	Paskolos termino pabaiga (paskolos užbaigimo data)
Moketi_data	Varchar(50)	Vienkartinės įmokos numatoma sumokėjimo data
Moketi_suma	numeric(18, 2)	Vienkartinės įmokos numatoma sumokėjimo suma
Sumoketa_data	Varchar(50)	Sumokėta vienkartinės įmokos data
Sumoketa_suma	numeric(18, 2)	Sumokėta vienkartinės įmokos suma

35 lentelė. „Piliėtis“ lentelės struktūra

Lentelė: Piliėtis		
Lauko pavadinimas	Tipas	Aprašymas
ID	Bigint	Piliečio identifikacinis numeris (PK)
Vardas	Varchar(20)	Piliečio vardas
Pavarde	Varchar(20)	Piliečio pavardė
Kodas	Varchar(10)	Piliečio asmens kodas
Adresas	Varchar(20)	Piliečio adresas
Telefonas	Varchar(20)	Piliečio telefonas

36 lentelė. „DuomenuSaltinis“ lentelės struktūra

Lentelė: DuomenuSaltinis		
Lauko pavadinimas	Tipas	Aprašymas
ID	Bigint	Duomenų Šaltinio identifikacinis numeris (PK)

Lentelė: Duomenų Šaltinis		
Lauko pavadinimas	Tipas	Aprašymas
Pavadinimas	Varchar(20)	Duomenų Šaltinio pavadinimas
Kreditorius	Varchar(20)	Kreditoriaus pavadinimas
Kreditorius_kodas	Varchar(10)	Kreditoriaus įmonės kodas
Adresas	Varchar(50)	Kreditoriaus adresas
Kontaktai	Varchar(150)	Kreditoriaus kontaktiniai duomenys
Prisijung_vardas	Varchar(20)	Prisijungimo prie duomenų šaltinio vartotojo vardas
Slaptažodis	Varchar(20)	Prisijungimo prie duomenų šaltinio slaptažodis
Kelias	Varchar(50)	Prisijungimo prie duomenų šaltinio nuoroda
Busena	Char(1)	Duomenų Šaltinio būseną

37 lentelė. „Šablonas“ lentelės struktūra

Lentelė: Šablonas		
Lauko pavadinimas	Tipas	Aprašymas
ID	Bigint	Paskolos identifikacinis numeris (PK)
DS_ID	Bigint	Duomenų šaltinio identifikacinis numeris (FK)
Pavadinimas	Varchar(20)	Šablono pavadinimas
Vardas	Varchar(20)	Piliečio vardas
Pavarde	Varchar(20)	Piliečio pavardė
Kodas	Varchar(10)	Piliečio asmens kodas
Adresas	Varchar(20)	Piliečio adresas
Kreditorius	Varchar(20)	Kreditoriaus pavadinimas
Kreditorius_kodas	Varchar(10)	Kreditoriaus įmonės kodas
Sutartis_nr	Varchar(50)	Paskolos sutarties numeris
Sutartis_pavad	Varchar(50)	Sutarties pavadinimas
Sutartis_tipas	Varchar(50)	Paskolos tipas
Sutartis_Suma	numeric(18, 2)	Bendra paskolos suma
Paskola_nuo	Varchar(50)	Paskotos termino pradžia (išdavimo data)
Paskola_iki	Varchar(50)	Paskolos termino pabaiga (paskolos užbaigimo data)
Imokos_data	Varchar(50)	Vienkartinės įmokos numatoma sumokėjimo data
Imokos_suma	numeric(18, 2)	Vienkartinės įmokos numatoma sumokėjimo suma

Lentelė: Šablonas

Lauko pavadinimas	Tipas	Aprašymas
Sumoketa_data	Varchar(50)	Sumokėta vienkartinės įmokos data
Sumoketa_suma	numeric(18, 2)	Sumokėta vienkartinės įmokos suma
Kelias	Varchar(50)	Prisijungimo prie duomenų šaltinio nuoroda
Busena	Char(1)	Šablono būseną (naujas, pratestuotas, patvirtintas, panaikintas)