

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
PROGRAMŲ INŽINERIJOS KATEDRA

Viktoras Babelis, Robert Kel

**Elektroninių mokėjimų universitetinėje
aplinkoje programinės įrangos projektavimo tyrimas**

Magistro darbas

Darbo vadovas

doc. dr. E. Karčiauskas

Kaunas, 2008

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
PROGRAMŲ INŽINERIJOS KATEDRA

Viktoras Babelis, Robert Kel

**Elektroninių mokėjimų universitetinėje
aplinkoje programinės įrangos projektavimo tyrimas**

Magistro darbas

Recenzentas

2008-05-26

doc. dr. A. Venčkauskas

Vadovas

2008-05-26

doc. dr. E. Karčiauskas

Atliko

2008-05-26

IFM-2/2 gr. stud.,
Viktoras Babelis
Robert Kel

Kaunas, 2008

Elektroninių mokėjimų universitetinėje aplinkoje programinės įrangos projektavimo tyrimas

SANTRAUKA

Magistrinio darbo tikslas yra ištirti ir sukurti elektroninių mokėjimų universitetinėje aplinkoje programinę įrangą. Darbo metu buvo atlikta projektavimo ir technologinių sprendimų analizė. Iškelti penki pagrindiniai sistemos realizavimo probleminiai uždaviniai ir pateikti jų galimi sprendimai. Suprojektuotos ir sukurtos programinės įrangos architektūra remiasi trisluoksnio projektavimo principais. Atliekant sistemos kokybės tyrimą ji buvo įvertinta kaip vidutiniška išskiriant tik gana prastai įvertinta sistemos palaikomumo kriterijų. O lyginant sistemos teikiamą naudingumą su alternatyviais sprendimais ji buvo įvertinta kaip naudingiausia.

The software for digital cash in the university environment: the research of designing

SUMMARY

The objective of this master project was to research and to develop the software for digital cash in the university environment. During the work execution, the analysis of design and technology solutions was performed. Further, five basic goals of system realization were formulated, the potential solutions for which were presented in the thesis. The architecture of the designed and the developed software is based on the principle of three layer design. During the quality analysis of the system, it was evaluated as an average system except that its maintainability rate was rather low. Its comparison with the alternative systems disclosed its superiority over them from the viewpoint of its expediency.

TURINYS

1. ĮVADAS	21
1.1. Dokumento paskirtis	21
1.2. Santrauka	23
2. ANALITINĖ DALIS	25
2.1. Įvadas	25
2.2. Tikslas	26
2.3. Motyvacija	27
2.4. Egzistuojantys sprendimai	29
2.4.1. Harvard's Electronic Shop and Pay	29
2.4.2. Financial Management Service.....	30
2.4.3. E-payment.....	32
2.4.4. Stanford ePay	33
2.4.5. SDePay.....	34
2.4.6. Enroll & Pay	36
2.4.7. Temple universiteto TUpay sistema	37
2.4.8. John Hopkins universiteto elektroninio apmokėjimo sistema E-Billing	39
2.4.9. Oregono universiteto elektroninio apmokėjimo sistema QuikPAY	40
2.5. Elektroninio apmokėjimo sistemos samprata	41
2.5.1. Elektroninio apmokėjimo sistemos veikimo principas.....	41
2.5.2. Transakcijos	41
2.5.3. Apmokėjimo schemas.....	42
2.5.3.1. Apmokėjimo schemų reikalingumo samprata	42
2.5.3.2. Mokėjimas kortele	43
2.5.3.3. SET	43
2.5.3.4. Skaitmeninis-kodinis apmokėjimas	43
2.5.3.5. Millicent.....	44
2.5.4. Elektroninio apmokėjimo formos	45
2.5.4.1. Elektroninio apmokėjimo formų tipai.....	45
2.5.4.2. Secure presentation	45
2.5.4.3. Customer registration.....	45
2.5.4.4. Credit-debit instruments	46
2.5.4.5. Electronic currency(valiuta)	46

2.5.4.6.	Server scrip	47
2.5.4.7.	Direct transfer	47
2.6.	Reikalavimai elektroninio apmokėjimo sistemoms	48
2.6.1.	Reikalavimai	48
2.6.2.	Reikalavimų apžvalga	49
2.6.2.1.	Privatumas	49
2.6.2.2.	Atomiškumas	49
2.6.2.3.	Bendradarbiavimas	49
2.6.2.4.	Pajėgumai	50
2.6.2.5.	Saugumas	50
2.6.2.6.	Patikimumas	51
2.6.2.7.	Patogumas	51
2.6.2.8.	Automatiškumas	51
2.6.2.9.	Anonimiškumas	52
2.7.	Galimos grėsmės	53
2.8.	Elektroninio apmokėjimo saugumui taikomos technologijos	54
2.8.1.	Saugumo technologijų reikalingumo samprata	54
2.8.2.	Duomenų saugumo ypatybės	54
2.8.2.1.	Slaptumas	54
2.8.2.2.	Vientisumas	55
2.8.2.3.	Tinkamumas	56
2.8.3.	Saugumo ir apmokėjimo protokolai	56
2.8.3.1.	Saugumo protokolų samprata	56
2.8.3.2.	SSL	56
2.8.3.3.	TLS	59
2.8.3.4.	Kerberos	59
2.8.4.	Prieigos kontrolė	60
2.8.4.1.	Prieigos kontrolės prasmė	60
2.8.4.2.	Prieigos kontrolių apžvalga	60
2.8.4.3.	RBAC modelis	63
2.8.4.4.	LDAC (arba MAC) modeliai	65
2.8.4.5.	RBAC dėl LBAC konfigūravimas	67
2.8.4.5.1.	Pagrindinis tinklelis	67
2.8.4.5.2.	Pagrindinė konstrukcija	68
2.8.4.5.3.	LBAC variantai	70

2.8.4.6.	Galimų RBAC konfigūracijų išplėtimas.....	72
2.8.4.7.	DAC modeliai	75
2.8.4.8.	RBAC dėl DAC konfigūravimas	77
2.8.4.8.1.	Bendri aspektai	77
2.8.4.8.2.	Griežtas DAC.....	79
2.8.4.8.3.	Liberalus DAC	80
2.8.4.8.4.	DAC su nuosavybės keitimu.....	80
2.8.4.8.5.	Daugialypė nuosavybė	81
2.8.4.8.6.	Suteiktos-priklausomybės panaikinimas.....	81
2.8.4.9.	Apibendrinimas.....	82
2.9.	Elektroninių apmokėjimų paslaugų teikimo saugumą užtikrinančios kompanijos ir technologijos-priemonės	85
2.9.1.	Saugumo technologijų-priemonių reikalingumo samprata.....	85
2.9.2.	Svarbiausios elektroninių apmokėjimo paslaugų teikimo saugumą užtikrinančios kompanijos ir technologijos-priemonės	85
2.9.2.1.	Baltimore „UniCERT“.....	85
2.9.2.2.	Brokat „Twister“	85
2.9.2.3.	VeriSign „VeriSign“	86
2.10.	Apžvelgtų sistemų kokybės ir funkcionalumo apibendrinimas	87
2.11.	Kokybės vertinimas	90
2.12.	Išvados	94
3.	PROJEKTAVIMO TYRIMO DALIS	96
3.1.	Įvadas	96
3.2.	Apžvalga	97
3.2.1.	Sistemos paskirtis	97
3.2.1.1.	Sistemos kūrimo pagrindimas.....	97
3.2.1.2.	Sistemos tikslai	99
3.2.1.3.	Sistemos kontekstas	99
3.2.2.	Svarbūs faktai ir prielaidos	100
3.2.2.1.	Svarbūs faktai	100
3.2.2.2.	Prielaidos	100
3.2.3.	Koncepcija	101
3.2.4.	Vartotojai	105
3.2.4.1.	Vartotojai	105

3.2.4.2.	Vartotojo prioritetai	107
3.2.4.3.	Vartotojo charakteristikos.....	107
3.2.4.4.	Vartotojo problemas	108
3.2.4.5.	Vartotojo tikslai	109
3.3.	Architektūros pateikimas	110
3.3.1.	Architektūros pateikimas	110
3.3.2.	Architektūros tikslai ir apribojimai.....	111
3.4.	Priimti techniniai sprendimai.....	113
3.4.1.	Projekto įgyvendinimo planai ir kokybės vertinimas	113
3.4.1.1.	Įgyvendinimas.....	113
3.4.1.2.	Reikalavimų inžinerijos metodika	115
3.4.1.3.	Projektavimo metodika	116
3.4.1.4.	Išplečiamumas	116
3.4.2.	Projekto įgyvendinimo etapai	117
3.5.	Numatoma eksploatacijos aplinka	119
3.6.	Sprendimų įtaka sistemos architektūrai	120
3.7.	Projekto technologiniai apribojimai.....	121
3.7.1.	Apribojimai sprendimui.....	121
3.7.2.	Diegimo aplinka.....	122
3.7.3.	Bendradarbiaujančios sistemos.....	124
3.7.4.	Komerciniai specializuoti programų paketai	124
3.8.	Veiklos kontekstas	126
3.9.	Veiklos padalinimas.....	128
3.10.	Panaudojimo atvejai.....	129
3.11.	Funkciniai reikalavimai	131
3.12.	Reikalavimai duomenims	134
3.13.	Nefunkciniai reikalavimai.....	135
3.13.1.	Reikalavimai sistemos išvaizdai	135
3.13.2.	Reikalavimai panaudojamumui	135
3.13.3.	Reikalavimai vykdymo charakteristikoms.....	136
3.13.4.	Reikalavimai veikimo sąlygoms	136
3.13.5.	Reikalavimai sistemos priežiūrai	136
3.13.6.	Reikalavimai saugumui.....	137
3.14.	Architektūros specifikacija	138
3.14.1.	Sistemos statinis vaizdas.....	138

3.14.1.1.	Paketai.....	138
3.14.1.2.	Paketų detalizavimas.....	139
3.14.1.2.1.	„Buhalteris“ paketo detalizavimas.....	139
3.14.1.2.2.	„Pardavėjas/Teikėjas“ paketo detalizavimas	141
3.14.1.2.3.	„Klientas-vartotojas“ paketo detalizavimas.....	142
3.14.1.2.4.	„Duomenų bazės serveris“ paketo detalizavimas	143
3.14.2.	Sistemos saugumo užtikrinimas.....	144
3.14.3.	Išdėstymo vaizdas	153
3.14.3.1.	Išdėstymo vaizdas	153
3.15.	Duomenų modelis	158
3.16.	Projekto išeiga.....	161
3.16.1.	Atviri klausimai	161
3.16.2.	Egzistuojantys sprendimai	161
3.16.2.1.	Pagamintos sistemos, kurios gali būti nupirktos.....	161
3.16.2.2.	Pagaminti komponentai, kurie gali būti panaudoti	164
3.16.2.3.	Galimas pakartotinas panaudojimas	167
3.16.3.	Naujos problemos	170
3.16.3.1.	Problemos diegimo aplinkai	170
3.16.3.2.	Įtaka jau instaliuotoms sistemoms	171
3.16.3.3.	Neigiamas vartotojų nusiteikimas.....	172
3.16.3.4.	Kliudantys diegimo aplinkos apribojimai.....	173
3.16.3.5.	Galimos naujos sistemos sukeltos problemos.....	174
3.16.4.	Sistemos vystymo etapai.....	174
3.16.5.	Pritaikymas	175
3.16.5.1.	Reikalavimai esamų duomenų perkėlimui.....	175
3.16.5.2.	Reikalingas duomenų transformavimas perkeliant į naują sistemą	176
3.16.6.	Rizikos	177
3.16.6.1.	Galimos sistemos kūrimo rizikos.....	177
3.16.6.2.	Atsitiktinumų (rizikų) valdymo planas	178
3.16.7.	Perspektyviniai reikalavimai.....	179
4.	EKSPERIMENTINĖ TIRIAMOJI DALIS.....	180
4.1.	Įvadas	180
4.2.	Programų sistemos kokybės vertinimo charakteristikos.....	181
4.3.	Sistemos kokybės tyrimas.....	184

4.4.	Sistemos naudingumo tyrimas	190
4.5.	Tolimesnės sistemos tobulinimo, plėtojimo galimybės	200
4.6.	Apibendrinimas.....	201
5.	IŠVADOS.....	203
6.	LITERATŪRA.....	206
7.	TERMINŲ IR SANTRUMPŲ ŽODYNAS.....	213
8.	PRIEDAI.....	216
8.1.	PRIEDAS 1: Apklausa.....	216
8.2.	PRIEDAS 2: „Elektroninių mokėjimų universitetinėje programinė įranga“ duomenų bazės schema.....	220
8.3.	PRIEDAS 3: Sistemos dinaminis vaizdas	221
8.3.1.	Sistemos dinaminio vaizdo diagramos	221
8.3.1.1.	Veiklos diagramos	221
8.3.1.1.1.	Buhalterio veiklos diagramos	221
8.3.1.1.1.1.	„B1. Identifikavimas“ veiklos diagrama.....	222
8.3.1.1.1.2.	„B2. Tvarkyti vartotojus“ veiklos diagrama	223
8.3.1.1.1.3.	„B2.1. Įtraukti naujus“ veiklos diagrama.....	224
8.3.1.1.1.4.	„B2.2. Koreguoti vartotojus“ veiklos diagrama.....	225
8.3.1.1.1.5.	„B2.3. Šalinti vartotojus“ veiklos diagrama	226
8.3.1.1.1.6.	„B3. Sąskaitų tvarkymas“ veiklos diagrama.....	227
8.3.1.1.1.7.	„B3.1. Sudaryti naujas sąskaitas“ veiklos diagrama.....	228
8.3.1.1.1.8.	„B3.2. Koreguoti sąskaitas“ veiklos diagrama	229
8.3.1.1.1.9.	„B3.3. Patvirtinti sąskaitas“ veiklos diagrama.....	230
8.3.1.1.1.10.	„B3.4. Pateikti sąskaitas“ veiklos diagrama	231
8.3.1.1.1.11.	„B3.5. Pašalinti sąskaitas“ veiklos diagrama.....	232
8.3.1.1.1.12.	„B4. Pinigų priėmimas“ veiklos diagrama	233
8.3.1.1.1.13.	„B5. Piniginiai pervedimai“ veiklos diagrama	234
8.3.1.1.1.14.	„B6. Sąskaitų-ataskaitų spausdinimas“ veiklos diagrama	235
8.3.1.1.2.	Pardavėjo/Teikėjo veiklos diagramos	236
8.3.1.1.2.1.	„PT1. Identifikavimas“ veiklos diagrama.....	236
8.3.1.1.2.2.	„PT2. Priimti apmokėjimą“ veiklos diagrama	237
8.3.1.1.2.3.	„PT2.1. Pateikti pardavėjo duomenis“ veiklos diagrama	238
8.3.1.1.2.4.	„PT2.2. Pateikti kliento duomenis“ veiklos diagrama	239
8.3.1.1.2.5.	„PT2.3. Pateikti sumą“ veiklos diagrama	240

8.3.1.1.2.6.	„PT3. Sąskaitos balanso peržiūra“ veiklos diagrama	241
8.3.1.1.2.7.	„PT4. Sąskaitos tvarkymas“ veiklos diagrama	242
8.3.1.1.2.8.	„PT4.1. Peržiūrėti sąskaitas“ veiklos diagrama	243
8.3.1.1.2.9.	„PT4.2. Apmokėti sąskaitas“ veiklos diagrama.....	244
8.3.1.1.2.10.	„PT5. Sąskaitų spausdinimas“ veiklos diagrama.....	245
8.3.1.1.3.	Kliento-vartotojo veiklos diagramos	246
8.3.1.1.3.1.	„KV1. Identifikavimas“ veiklos diagrama.....	246
8.3.1.1.3.2.	„KV2. Sąskaitos balanso peržiūra“ veiklos diagrama	247
8.3.1.1.3.3.	„KV3. Sąskaitos tvarkymas“ veiklos diagrama.....	248
8.3.1.1.3.4.	„KV3.1. Peržiūrėti sąskaitas“ veiklos diagrama	249
8.3.1.1.3.5.	„KV3.2. Apmokėti sąskaitas“ veiklos diagrama	250
8.3.1.1.3.6.	„KV4. Sąskaitų spausdinimas“ veiklos diagrama.....	251

LENTELIŲ SĄRAŠAS

Lentelė 2.10-1. Apžvelgtų sistemų funkcionalumo apibendrinimas	88
Lentelė 2.11-1. Apžvelgtų sistemų kokybės vertinimas	92
Lentelė 3.2-1. Vartotojas buhalteris.....	105
Lentelė 3.2-2. Vartotojas administratorius	105
Lentelė 3.2-3. Vartotojas pardavėjas/teikėjas.....	106
Lentelė 3.2-4. Vartotojas klientas-vartotojas.....	106
Lentelė 3.2-5. Vartotojo prioritetai.....	107
Lentelė 3.3-1. Architektūros pateikimo vaizdai ir modeliai.....	110
Lentelė 3.4-1. Projekto įgyvendinimo etapai.....	117
Lentelė 3.7-1. Apribojimai sprendimui.....	121
Lentelė 3.9-1. Veiklos įvykių sąrašas.....	128
Lentelė 3.15-1. Duomenų bazės modelio esybės.....	160
Lentelė 3.16-1. Sistemos vystymo etapai.....	175
Lentelė 3.16-2. Reikalavimai esamų duomenų perkėlimui.....	175
Lentelė 3.16-3. Sistemos kūrimo rizikos.....	177
Lentelė 3.16-4. Sistemos kūrimo atsitiktinumų (rizikų) valdymo planas.....	178
Lentelė 4.2-1. ISO 9126 modelio kriterijai ir subkriterijai.....	181
Lentelė 4.3-1. „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos kokybės vertinimas pagal ISO 9216 standartą.....	184
Lentelė 4.4-1. Egzistuojančių sistemų naudingumo palyginimas su realizuota sistema („Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“).	191
Lentelė 4.4-2. Galimybių skaičiaus ir svorinio naudingumo įverčio priklausomybė.....	197
Lentelė 8.1-1. Sistemos kokybės vertinimas pagal ISO 9216 standartą anketa.....	216

PAVEIKSLIUKŲ SĄRAŠAS

Paveikslas 2.4-1. Temple University Payment System Statement [].....	38
Paveikslas 2.8-1. SSL rankos paspaudimas su dvipuse autentifikacija ir sertifikatais. []	58
Paveikslas 2.8-2. RBAC96 modelis.....	63
Paveikslas 2.8-3. Dalinai sutvarkytas tinklelis	66
Paveikslas 2.8-4. Hierarchijos vaidmuo tinkleliui	67
Paveikslas 2.8-5. Kintama vaidmens hierarchija griežtai *- ypatybei.....	70
Paveikslas 2.8-6. Kintama vaidmens hierarchija Griežtom *-ypatybėm.....	72
Paveikslas 2.8-7. Vaidmens hierarchija ir jos vartotojų priskyrimai.....	74
Paveikslas 2.8-8. (a) Vaidmenų administravimas sujungtas su objektu; (b) Administracinio vaidmens hierarchija.....	78
Paveikslas 2.8-9. Skaityti_O vaidmenys susiję su TĖVAS_O nariais.	82
Paveikslas 2.8-10. Modelių apribojimai.	83
Paveikslas 2.10-1. Apžvelgtų sistemų funkcionalumo apibendrinimas, pagal funkcijų kiekį.	89
Paveikslas 2.10-2. Apžvelgtų sistemų funkcionalumo įvertinimas.....	89
Paveikslas 2.11-1. Apžvelgtų sistemų kokybės įvertinimas.....	93
Paveikslas 3.2-1. „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos išdėstymo vaizdo koncepcinė diagrama.....	104
Paveikslas 3.8-1. Veiklos konteksto diagrama.	127
Paveikslas 3.10-1. Panaudojimo atvejų diagrama.	130
Paveikslas 3.12-1. Pradinis duomenų modelis.....	134
Paveikslas 3.14-1. Sistemos išskaidymas į paketus aukščiausiam lygyje diagrama.....	138
Paveikslas 3.14-2. Paketo <i>Buhalteris</i> klasių diagrama.	140
Paveikslas 3.14-3. Paketo <i>Pardavėjas/Teikėjas</i> klasių diagrama.	141
Paveikslas 3.14-4. Paketo <i>Klientas-vartotojas</i> klasių diagrama.	142
Paveikslas 3.14-5. Paketo <i>Duomenų bazės serveris</i> klasių diagrama.....	143
Paveikslas 3.14-6. Prieigos kontrolės duomenų schema.	144
Paveikslas 3.14-7. Vartotojų, vaidmenų ir teisių hierarchinis pasiskirstymas.....	145
Paveikslas 3.14-8. Vartotojo prieigos kontrolė.....	146
Paveikslas 3.14-9. Vartotojo identifikacinė kontrolė.....	147
Paveikslas 3.14-10. Žingsniai po autorizavimo.....	148
Paveikslas 3.14-11. Autorizavimas pagal Vaidmenų Grupes.....	149
Paveikslas 3.14-12. Prieigos kontrolė sistemos taikymas internetiniams puslapiams.....	150
Paveikslas 3.14-13. Vartotojo slaptažodžio pakeitimas.....	151

Paveikslas 3.14-14. Vartotojo slaptažodžio atstatymas.	152
Paveikslas 3.14-15. „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos išdėstymo vaizdo koncepcinė diagrama.	156
Paveikslas 3.14-16. „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos išdėstymo vaizdo diagrama.	157
Paveikslas 3.15-1. Duomenų bazės modelis.	159
Paveikslas 4.1-1. Eksperimentinio tyrimo aspektai.	180
Paveikslas 4.2-1. ISO 9126 modelis.	181
Paveikslas 4.4-1. Sistemų galimybių palyginimas.	194
Paveikslas 4.4-2. Sistemų naudingumo svorinių įverčių palyginimas.	195
Paveikslas 4.4-3. Sistemų naudingumo ir realizuotų galimybių procentinis palyginimas.	196
Paveikslas 4.4-4. Realizuotų galimybių ir svorinių naudingumo įverčių grafikas.	198
Paveikslas 4.4-5. Realizuotų galimybių ir svorinių naudingumo įverčių įtaka bendram sistemos naudingumui.	198
Paveikslas 8.4-1. Duomenų bazės schema.	220
Paveikslas 8.5-1. <i>B1. Identifikavimas</i> veiklos diagrama.	222
Paveikslas 8.5-2. <i>B2. Tvarkyti vartotojus</i> veiklos diagrama.	223
Paveikslas 8.5-3. <i>B2.1. Įtraukti naujus vartotojus</i> veiklos diagrama.	224
Paveikslas 8.5-4. <i>B2.2. Koreguoti vartotojus</i> veiklos diagrama.	225
Paveikslas 8.5-5. <i>B2.3. Šalinti vartotojus</i> veiklos diagrama.	226
Paveikslas 8.5-6. <i>B3. Sąskaitų tvarkymas</i> veiklos diagrama.	227
Paveikslas 8.5-7. <i>B3.1. Sudaryti naujas sąskaitas</i> veiklos diagrama.	228
Paveikslas 8.5-8. <i>B3.2. Koreguoti sąskaitas</i> veiklos diagrama.	229
Paveikslas 8.5-9. <i>B3.3. Patvirtinti sąskaitas</i> veiklos diagrama.	230
Paveikslas 8.5-10. <i>B3.4. Pateikti sąskaitas</i> veiklos diagrama.	231
Paveikslas 8.5-11. <i>B3.5. Pašalinti sąskaitas</i> veiklos diagrama.	232
Paveikslas 8.5-12. <i>B4. Pinigų priėmimas</i> veiklos diagrama.	233
Paveikslas 8.5-13. <i>B5. Piniginiai pervedimai</i> veiklos diagrama.	234
Paveikslas 8.5-14. <i>B6. Sąskaitų-ataskaitų spausdinimas</i> veiklos diagrama.	235
Paveikslas 8.5-15. <i>PT1. Identifikavimas</i> veiklos diagrama.	236
Paveikslas 8.5-16. <i>PT2. Priimti apmokėjimą</i> veiklos diagrama.	237
Paveikslas 8.5-17. <i>PT2.1. Pateikti pardavėjo duomenis</i> veiklos diagrama.	238
Paveikslas 8.5-18. <i>PT2.2. Pateikti kliento duomenis</i> veiklos diagrama.	239
Paveikslas 8.5-19. <i>PT2.3. Pateikti sumą</i> veiklos diagrama.	240
Paveikslas 8.5-20. <i>PT3. Sąskaitos balanso peržiūra</i> veiklos diagrama.	241

Paveikslas 8.5-21. <i>PT4. Sąskaitos tvarkymas</i> veiklos diagrama.....	242
Paveikslas 8.5-22. <i>PT4.1. Peržiūrėti sąskaitas</i> veiklos diagrama.	243
Paveikslas 8.5-23. <i>PT4.2. Apmokėti sąskaitas</i> veiklos diagrama.	244
Paveikslas 8.5-24. <i>PT5. Sąskaitų spausdinimas</i> veiklos diagrama.....	245
Paveikslas 8.5-25. <i>KV1. Identifikavimas</i> veiklos diagrama.	246
Paveikslas 8.5-26. <i>KV2. Sąskaitos balanso peržiūra</i> veiklos diagrama.	247
Paveikslas 8.5-27. <i>KV3. Sąskaitos tvarkymas</i> veiklos diagrama.	248
Paveikslas 8.5-28. <i>KV3.1. Peržiūrėti sąskaitas</i> veiklos diagrama.....	249
Paveikslas 8.5-29. <i>KV3.2. Apmokėti sąskaitas</i> veiklos diagrama.....	250
Paveikslas 8.5-30. <i>KV4. Sąskaitų spausdinimas</i> veiklos diagrama.	251

DOKUMENTO SKYRIŲ AUTORYSTĖ

Pateikima informacija apie šio dokumento skyrių autorystę.

Skyrių autorius Viktoras Babelis

1. ĮVADAS, 21	2.8.2.1. Slaptingumas, 54
1.1. Dokumento paskirtis, 21	2.8.2.3. Tinkamumas, 56
1.2. Santrauka, 23	2.8.3.1. Saugumo protokolų samprata, 56
2. ANALITINĖ DALIS, 25	2.8.3.2. SSL, 56
2.1. Įvadas, 25	2.8.3.3. TLS, 59
2.2. Tikslas, 26	2.8.3.4. Kerberos, 59
2.3. Motyvacija, 27	2.8.4.1. Prieigos kontrolės prasmė, 60
2.4.1. Harvard's Electronic Shop and Pay, 29	2.8.4.2. Prieigos kontrolių apžvalga, 60
2.4.2. Financial Management Service, 30	2.8.4.3. RBAC modelis, 63
2.4.3. E-payment, 32	2.8.4.5.1. Pagrindinis tinklelis, 67
2.4.4. Stanford ePay, 33	2.8.4.5.3. LBAC variantai, 70
2.4.5. SDePay, 34	2.8.4.7. DAC modeliai, 75
2.4.6. Enroll & Pay, 36	2.8.4.8. RBAC dėl DAC konfigūravimas, 77
2.5.1. Elektroninio apmokėjimo sistemos veikimo principas, 41	2.8.4.8.1. Bendri aspektai, 77
2.5.2. Transakcijos, 41	2.8.4.8.3. Liberalus DAC, 80
2.5.3.1. Apmokėjimo schemų reikalingumo samprata, 42	2.8.4.8.5. Daugialypė nuosavybė, 81
2.5.3.2. Mokėjimas kortele, 43	2.9.1. Saugumo technologijų-priemonių reikalingumo samprata, 85
2.5.3.3. SET, 43	2.9.2.1. Baltimore „UniCERT“, 85
2.5.3.4. Skaitmeninis-kodinis apmokėjimas, 43	2.9.2.2. Brokat „Twister“, 85
2.5.3.5. Millicent, 44	2.9.2.3. VeriSign „VeriSign“, 86
2.6.1. Reikalavimai, 48	2.10. Apžvelgtų sistemų kokybės ir funkcionalumo apibendrinimas, 87
2.6.2.1. Privatumas, 49	3. PROJEKTAVIMO TYRIMO DALIS, 96
2.6.2.2. Atomiškumas, 49	3.1. Įvadas, 96
2.6.2.3. Bendradarbiavimas, 49	3.2.1.1. Sistemos kūrimo pagrindimas, 97
2.6.2.4. Pajėgumai, 50	3.2.1.2. Sistemos tikslai, 99
2.6.2.5. Saugumas, 50	3.2.2.1. Svarbūs faktai, 100
2.6.2.6. Patikimumas, 51	3.2.2.2. Prielaidos, 100
2.6.2.7. Patogumas, 51	3.2.3. Konceptija, 101
2.6.2.8. Automatiškumas, 51	3.2.4.1. Vartotojai, 105
2.6.2.9. Anonimiškumas, 52	3.2.4.2. Vartotojo prioritetai, 107
2.7. Galimos grėsmės, 53	3.3.1. Architektūros pateikimas, 110
2.8.1. Saugumo technologijų reikalingumo samprata, 54	3.3.2. Architektūros tikslai ir apribojimai, 111
	3.4.1.1. Įgyvendinimas, 113

3.4.1.2. Reikalavimų inžinerijos metodika, 115	8. PRIEDAI, 216
3.4.1.3. Projektavimo metodika, 116	8.1. PRIEDAS 1: Apklausa, 216
3.4.1.4. Išplečiamumas, 116	8.2. PRIEDAS 2: „Elektroninių mokėjimų universitetinėje programinė įranga“ duomenų bazės schema, 220
3.4.2. Projekto įgyvendinimo etapai, 117	8.3.1.1.1.1. „B1. Identifikavimas“ veiklos diagrama, 222
3.8. Veiklos kontekstas, 126	8.3.1.1.1.2. „B2. Tvarkyti vartotojus“ veiklos diagrama, 223
3.9. Veiklos padalinimas, 128	8.3.1.1.1.3. „B2.1. Įtraukti naujus“ veiklos diagrama, 224
3.10. Panaudojimo atvejai, 129	8.3.1.1.1.4. „B2.2. Koreguoti vartotojus“ veiklos diagrama, 225
3.12. Reikalavimai duomenims, 134	8.3.1.1.1.5. „B2.3. Šalinti vartotojus“ veiklos diagrama, 226
3.13.1. Reikalavimai sistemos išvaizdai, 135	8.3.1.1.1.6. „B3. Sąskaitų tvarkymas“ veiklos diagrama, 227
3.13.2. Reikalavimai panaudojamumui, 135	8.3.1.1.1.7. „B3.1. Sudaryti naujas sąskaitas“ veiklos diagrama, 228
3.13.3. Reikalavimai vykdymo charakteristikoms, 136	8.3.1.1.1.8. „B3.2. Koreguoti sąskaitas“ veiklos diagrama, 229
3.13.6. Reikalavimai saugumui, 137	8.3.1.1.1.9. „B3.3. Patvirtinti sąskaitas“ veiklos diagrama, 230
3.14.1.1. Paketai, 138	8.3.1.1.1.10. „B3.4. Pateikti sąskaitas“ veiklos diagrama, 231
3.14.1.2.1. „Buhalteris“ paketo detalizavimas, 139	8.3.1.1.1.11. „B3.5. Pašalinti sąskaitas“ veiklos diagrama, 232
3.14.1.2.4. „Duomenų bazės serveris“ paketo detalizavimas, 143	8.3.1.1.1.12. „B4. Pinigų priėmimas“ veiklos diagrama, 233
3.14.3.1. Išdėstymo vaizdas, 153	8.3.1.1.1.13. „B5. Piniginiai pervedimai“ veiklos diagrama, 234
3.16.2.1. Pagamintos sistemos, kurios gali būti nupirktos, 161	8.3.1.1.1.14. „B6. Sąskaitų-ataskaitų spausdinimas“ veiklos diagrama, 235
3.16.2.2. Pagaminti komponentai, kurie gali būti panaudoti, 164	
3.16.3.1. Problemos diegimo aplinkai, 170	
3.16.3.2. Įtaka jau instaliuotoms sistemoms, 171	
3.16.4. Sistemos vystymo etapai, 174	
3.16.6.1. Galimos sistemos kūrimo rizikos, 177	
3.16.6.2. Atsitiktinumų (rizikų) valdymo planas, 178	
4. EKSPERIMENTINĖ TIRIAMOJI DALIS, 180	
4.1. Įvadas, 180	
4.2. Programų sistemos kokybės vertinimo charakteristikos, 181	
4.4. Sistemos naudingumo tyrimas, 190	
4.5. Tolimesnės sistemos tobulinimo, plėtojimo galimybės, 200	

Skyrių autorius Robert Kel

- 2. ANALITINĖ DALIS, 25
 - 2.4.7. Temple universiteto TUpay sistema, 37
 - 2.4.8. John Hopkins universiteto elektroninio apmokėjimo sistema E-Billing, 39
 - 2.4.9. Oregono universiteto elektroninio apmokėjimo sistema QuikPAY, 40
 - 2.5.4.1. Elektroninio apmokėjimo formų tipai, 45
 - 2.5.4.2. Secure presentation, 45
 - 2.5.4.3. Customer registration, 45
 - 2.5.4.4. Credit-debit instruments, 46
 - 2.5.4.5. Electronic currency(valiuta), 46
 - 2.5.4.6. Server scrip, 47
 - 2.5.4.7. Direct transfer, 47
 - 2.8.2.2. Vientisumas, 55
 - 2.8.4.4. LDAC (arba MAC) modeliai, 65
 - 2.8.4.5.2. Pagrindinė konstrukcija, 68
 - 2.8.4.6. Galimų RBAC konfigūracijų išplėtimas, 72
 - 2.8.4.8.2. Griežtas DAC, 79
 - 2.8.4.8.4. DAC su nuosavybės keitimu, 80
 - 2.8.4.8.6. Suteiktos-priklausomybės panaikinimas, 81
 - 2.11. Kokybės vertinimas, 90
 - 2.12. Išvados, 94
- 3. PROJEKTAVIMO TYRIMO DALIS, 96
 - 3.2.1.3. Sistemos kontekstas, 99
 - 3.2.4.3. Vartotojo charakteristikos, 107
 - 3.2.4.4. Vartotojo problemos, 108
 - 3.2.4.5. Vartotojo tikslai, 109
 - 3.5. Numatoma eksploatacijos aplinka, 119
 - 3.6. Sprendimų įtaka sistemos architektūrai, 120
 - 3.7.1. Apribojimai sprendimui, 121
 - 3.7.2. Diegimo aplinka, 122
 - 3.7.3. Bendradarbiaujančios sistemos, 124
 - 3.7.4. Komerciniai specializuoti programų paketai, 124
 - 3.13.4. Reikalavimai veikimo sąlygoms, 136
 - 3.13.5. Reikalavimai sistemos priežiūrai, 136
 - 3.14.1.2.2. „Pardavėjas/Teikėjas“ paketo detalizavimas, 141
 - 3.14.1.2.3. „Klientas-vartotojas“ paketo detalizavimas, 142
 - 3.14.2. Sistemos saugumo užtikrinimas, 144
 - 3.16.1. Atviri klausimai, 161
 - 3.16.2.3. Galimas pakartotinas panaudojimas, 167
 - 3.16.3.3. Neigiamas vartotojų nusiteikimas, 172
 - 3.16.3.4. Kliudantys diegimo aplinkos apribojimai, 173
 - 3.16.3.5. Galimos naujos sistemos sukeltos problemos, 174
 - 3.16.5.1. Reikalavimai esamų duomenų perkėlimui, 175
 - 3.16.5.2. Reikalingas duomenų transformavimas perkeliant į naują sistemą, 176
 - 3.16.7. Perspektyviniai reikalavimai, 179
- 4. EKSPERIMENTINĖ TIRIAMOJI DALIS, 180
 - 4.3. Sistemos kokybės tyrimas, 184
 - 8.3.1.1.2.1. „PT1. Identifikavimas“ veiklos diagrama, 236
 - 8.3.1.1.2.2. „PT2. Priimti apmokėjimą“ veiklos diagrama, 237
 - 8.3.1.1.2.3. „PT2.1. Pateikti pardavėjo duomenis“ veiklos diagrama, 238
 - 8.3.1.1.2.4. „PT2.2. Pateikti kliento duomenis“ veiklos diagrama, 239
 - 8.3.1.1.2.5. „PT2.3. Pateikti sumą“ veiklos diagrama, 240
 - 8.3.1.1.2.6. „PT3. Sąskaitos balanso peržiūra“ veiklos diagrama, 241
 - 8.3.1.1.2.7. „PT4. Sąskaitos tvarkymas“ veiklos diagrama, 242
 - 8.3.1.1.2.8. „PT4.1. Peržiūrėti sąskaitas“ veiklos diagrama, 243
 - 8.3.1.1.2.9. „PT4.2. Apmokėti sąskaitas“ veiklos diagrama, 244
 - 8.3.1.1.2.10. „PT5. Sąskaitų spausdinimas“ veiklos diagrama, 245
 - 8.3.1.1.3.1. „KV1. Identifikavimas“ veiklos diagrama, 246
 - 8.3.1.1.3.2. „KV2. Sąskaitos balanso peržiūra“ veiklos diagrama, 247

8.3.1.1.3.3. „KV3. Sąskaitos tvarkymas“ veiklos
diagrama, 248

8.3.1.1.3.4. „KV3.1. Peržiūrėti sąskaitas“ veiklos
diagrama, 249

8.3.1.1.3.5. „KV3.2. Apmokėti sąskaitas“ veiklos
diagrama, 250

8.3.1.1.3.6. „KV4. Sąskaitų spausdinimas“ veiklos
diagrama, 251

Skyrių bendraautoriai: Viktoras Babelis ir Robert Kel

2. ANALITINĖ DALIS, 25

2.8.4.9. Apibendrinimas, 82

3. PROJEKTAVIMO TYRIMO DALIS, 96

3.11. Funkciniai reikalavimai, 131

3.14.2. Sistemos saugumo užtikrinimas, 144

4. EKSPERIMENTINĖ TIRIAMOJI DALIS, 180

4.6. Apibendrinimas, 201

5. IŠVADOS, 203

6. LITERATŪRA, 206

7. TERMINŲ IR SANTRUMPŲ ŽODYNAS, 213

1. ĮVADAS

1.1. Dokumento paskirtis

Dokumentas skirtas pateikti išsamų „Elektroninių mokėjimų universitetinėje aplinkoje programinės įrangos“ projektavimo tyrimą įvairiais sistemos vaizdais, naudojant skirtingus architektūrinius vaizdus, kurie parodo skirtingus sistemos architektūros aspektus. Šio dokumento tikslas yra surinkti ir pateikti svarbius architektūrinius sprendimus, kurie buvo atlikti projektuojant ir tyrinėjant sistemą.

Magistro baigiamojo darbo dokumento struktūra:

Santrauka (SUMMARY – angl.) – lietuvių ir užsienio (anglų) kalbomis glaustai apibūdinamas darbo turinys, mokslinė problema ir išvados.

Įvadas – aptariamos teorinės darbo prielaidos bei metodologija, apibrėžiamas tiriamasis objektas, aptariama problema liečianti teorinę literatūrą, tolesnės analizės tvarka.

Analitinė dalis - skirta projektuojamos sistemos ir problemų sprendimo būdų analizei. Pristatomas projektas.

Suformuluojamas uždavinys, projektavimo metu iškilusios projektavimo ar programavimo inžinerijos problemos. Pateikiama galimų sprendimų apžvalga, priimto ir realizuoto sprendimo pagrindimas bei įvertinimas.

Projektavimo tyrimo dalis - magistratūros studijų metu sukurtos programinės įrangos projektavimo analizė, joje tyrinėjamos programų sistemos projektavimo galimybės, pagrindžiamos ir realizuojamos siūlomos galimybės. (Šiame skyriuje taip pat naudojami magistratūros studijų metu sukurtos programinės įrangos techninės-projektinės dokumentacijos esminiai aspektai.)

Eksperimentinė tiriamoji dalis - magistratūros studijų metu sukurtos programinės įrangos eksperimentinis kokybės ir naudingumo tyrimas. Joje tyrinėjama programų sistemos kokybės ir naudingumo tobulinimo galimybės, pagrindžiami realizuoti ir siūlomi patobulinimai.

Išvados - glaustai išdėstoma nuomonė apie svarbiausius tezėse paliestus klausimus: projektavimo metu priimti sprendimai, atliktų tyrimų esmė ir gauti rezultatai.

Literatūra - nurodoma panaudota literatūra.

Terminų ir santrumpų žodynas – pateikiami darbo tekste vartojamos santrumpos ir specialūs terminai, reikalaujantys paaiškinimo.

Priedai – pateikiama tyrimams reikalinga medžiaga.

1.2. Santrauka

Finansiniai apmokėjimai yra vienas iš natūraliausių žmonių atsiskaitymo už įvairias prekes ir paslaugas būdų. Tai liečia ir informacines technologijas, kurios atveria plačias ir neribotas galimybes, kurias galima realiai pritaikyti finansiniams apmokėjimams. Nors idėja panaudoti informacines technologijas elektroniniams apmokėjimams nėra nauja, tačiau yra sukurta labai nedaug programinės įrangos priemonių, kurios leistu elektroninius apmokėjimus realizuoti universitetinėje aplinkoje. Kitaip tariant informacinių technologijų panaudojimas elektroninių apmokėjimų srityje yra vis dar daug žadanti sritis su didelėmis galimybėmis ir perspektyvomis.

Realizuotas tyrimas leido išsamiau ištirti naujausias elektroninių apmokėjimų technologijas, jų teikiamas galimybes, funkcionalumą, realizaciją ir pritaikymą realiame gyvenime, bei universiteto aplinkoje.

Kombinuojant informacines technologijas su tradiciniu apmokėjimu (grynais), bus pasiektas naujas apmokėjimo būdas, t.y. elektroninis apmokėjimas („elektroniniais pinigais“). To pasėkoje bus pasiektas didesnis darbo atliekant apmokėjimus bei tvarkant su apmokėjimais susijusią informaciją, efektyvumas ir našumas, nei naudojant tradicinį apmokėjimo būdą.

Elektroniniai apmokėjimai yra ypač aktualūs didelėms organizacijoms tokioms kaip prekybinės organizacijos ar universitetai. Universitetai aptarnauja tūkstančius žmonių, yra atliekama dešimtys tūkstančių įvairiausių atsiskaitymų, kuriems realizuoti ir aptarnauti sugaištama begalės brangaus laiko. O elektroninis apmokėjimas yra gal būt vienintelė priemonė, kurią ypač efektyviai galima panaudoti apmokėjimų infrastruktūroje, taupant darbuotojų darbo resursus, bet kokioje organizacijoje, o ypač universitete.

Projektavimo tyrimo metu buvo realizuotas elektroninių mokėjimų universitetinėje aplinkoje programinės įrangos prototipas, su elektroninių apmokėjimų ir finansinės informacijos tvarkymo funkcijomis.

Dėl elektroninių technologijų sudėtingumo, keliamų didelių reikalavimų patikimumui ir saugumui ir itin aukštos kainos, dar nėra sukurta pilnai tinkamų priemonių leidžiančių pilnai organizuoti ir eksploatuoti elektroninius mokėjimus universitetinėje aplinkoje. Tačiau daugelis pasaulio ir Lietuvos universitetų bei privačių organizacijų dirba šia linkme. Todėl panaudojant iki šios sukauptas žinias ir patirtį apie elektroninius mokėjimus bei egzistuojančias elektroninių mokėjimų technologijas, šiame projekte bus sukurtas

„Elektroninių mokėjimų universitetinėje aplinkoje programinės įrangos“ prototipas, kuris leis realizuoti elektroninius mokėjimus, bei su jais susijusios informacijos tvarkymą.

Tikslas

Ištirti naujausias elektroninių apmokėjimų technologijas, jų teikiamas galimybes, funkcionalumą, realizaciją ir pritaikymą realiame gyvenime, bei universiteto aplinkoje, šių technologijų specifikaciją ir problemas, bei atlikti realizuotos sistemos kokybės ir naudingumo tyrimą.

Problemos

Elektroninių mokėjimų universitetinėje aplinkoje programinės įrangos funkcionavimo principai, panaudojimo ir realizacijos galimybės - elektroniniams mokėjimams universitetinėje aplinkoje.

Problemos išplaukiančios iš pagrindinės problemos:

- Elektroniniai mokėjimai universitetinėje aplinkoje
- Sąskaitų ir apmokėjimų valdymas
- Centralizuotas finansinės informacijos valdymas
- Vartotojų kontrolė finansinės informacijos aplinkoje
- Elektroninių mokėjimų sistemos integracija universitetinėje aplinkoje

Uždaviniai

Darbo uždaviniai yra tiesiogiai išvedami iš šio darbo pagrindinio tikslo:

- I. Atlikti projektavimo ir probleminių uždavinių, bei technologinių sprendimų analizę.
- II. Remiantis atliktos analizės išvadomis realizuoti „elektroninių mokėjimų universitetinėje aplinkoje programinę įrangą“ sistemą.
- III. Atlikti sukurtos sistemos kokybės ir naudingumo tyrimą.

2. ANALITINĖ DALIS

2.1. Įvadas

Informacinės technologijos nuolat vystomos ir plėtojamos. Vis atsiranda naujų pritaikymo sričių ir būdų. Taip pat nuolat yra plėtojamos ir tobulinamos programinės įrangos kūrimo priemonės, kurios leidžia sukurti tai ką išsivaizduojame ir ko trokštame.

Tobulėjant informacinėms ir programinėms technologijoms neatsiejama gyvenimo dalimi tapo WWW pasaulis, be kurio šiandien daugeliui yra sunku išsivaizduoti sėkmingą ir turiningą darbą. Iš pradžių WWW buvo tik informacijos šaltinis, bet plėtojantis technologijoms atsirado ir nauji pritaikymo būdai, vienas iš pritaikymo būdų yra elektroninis apmokėjimas, kuris sparčiai braunasi į mūsų gyvenimą.

Elektroninis apmokėjimas išsprendžia begales problemų susijusių su prekių, paslaugų įsigijimu ir apmokėjimu. Nebereikia vargintis dėl elektroninių atsiskaitymų, kurie anksčiau sukeldavo daug problemų (pvz. gaišamas brangus laikas stovint eilėje, kai yra norima apmokėti už kažkokias tai paslaugas).

Elektroninės paslaugos sparčiai veržiasi į mūsų gyvenimą, tad šias paslaugas būtų galima pritaikyti ir mokslo pasaulyje bei gyvenime, t.y. sukurti sistemą, kuri leistų atlikti įvairias finansines operacijas universitetinėje aplinkoje.

Technologijos nuolat tobulėja, kai kurie pasaulio koledžai ir universitetai, kartu evoliucionuoja su naujomis technologijomis ir jų atveriamomis galimybėmis. Koledžai ir universitetai pasinaudoję naujausių technologijų teikiamomis galimybėmis, sukuria savo elektroninio apmokėjimo sistemas. Sukurtos sistemos leidžia elektroniniu būdu (pasinaudojant kreditinėmis kortelėmis, ar kitokiomis apmokėjimo sistemomis), studentams pervesti stipendijas į jų sąskaitas, apmokėti už mokslą, apmokėti už universiteto teikiamas įvairias paslaugas, ar paslaugas teikiamas universitetų partnerių. Šios sistemos labiausiai palengvina studentų, universitetų darbuotojų ir pan. veiklą.

Apmokėjimo universitetinėje aplinkoje sistemos yra naudojamos mokslo įstaigose: Harvard University, Indiana University, Trent University, Stanford University, South Dakota State University, University of Kansas, Temple University, John Hopkins University, Oregono university, ir daugelyje kitų. Tokios sistemos susilaukė didelio studentų, darbuotojų ir pan. dėmesio, bei teigiamų atsiliepimų[1].

2.2. Tikslas

Darbo tikslas yra ištirti elektroninių mokėjimų universitetinėje aplinkoje programinę įrangą, bei sukurti jos prototipą. Sukurtas apmokėjimų programinės įrangos prototipas galės būti panaudotas atlikti įvairioms finansinėms operacijoms universitetinėje aplinkoje (t.y. tyrinėti jo panaudojimo galimybes), kaip pavyzdį būtų galima paminėti, tokias tyrinėjamos/kuriamos programinės įrangos galimybes: priimti iš studentų ar darbuotojų apmokėjimus už: kopijavimo, spausdinimo, maitinimo, apsipirkinėjimą knygyne ir kt. paslaugas. Sukurta programinė įranga leis pasinaudojant studento ar darbuotojo pažymėjimu atlikti apmokėjimą už paslaugas. Peržiūrėti savo sąskaitos likutį, atlikti pervedimus elektroninio apmokėjimo sistemos teikiamomis galimybėmis.

Naudojant elektroninio apmokėjimo programinę įrangą, finansinio atsiskaitymo operacijos bus atliekamos kur kas greičiau, o be to vartotojas bet kada galės gauti išsamią savo finansinę informaciją.

2.3. Motyvacija

Tiriamąo darbo tikslas yra iširti bei sukurti elektroninių mokėjimų universitetinėje aplinkoje programinės įrangos prototipą. Sukurta apmokėjimų programinė įranga galės būti panaudota atlikti įvairioms finansinėms operacijoms universitetinėje aplinkoje, kaip pavyzdį būtų galima paminėti tokias kuriamos programinės įrangos galimybes: priimti iš studentų ar darbuotojų apmokėjimus už: kopijavimo, spausdinimo, maitinimo, apsipirkinėjimo knygyne ir pan. paslaugas.

Naudojant elektroninio apmokėjimo programinę įrangą, finansinio atsiskaitymo operacijos bus atliekamos kur kas greičiau nei tradiciniu būdu. Kadangi atsiskaitymai atliekami grynais mokėtojas turi suskaičiuoti reikiamą pinigų sumą, o po to pardavėjas vėl ją perskaičiuoti ir dar jei reikia suskaičiuoti grąžą, tai užtrunka labai ilgai ir gaištama bereikalingai daug laiko. Dar blogesnis atvejis, kai apmokėjimui už paslaugas ir pan. išduodamas kvitas, kurį reikia kažkur apmokėti, o po to vėl grąžinti, čia yra apkraunami ir klientai ir darbuotojai, nes mokėtojas sugaišta daug laiko, kol apmoka, o darbuotojams reikia suregistruoti, kas ir kada sumokėjo ar nesumokėjo. Tai labai nepatogu, todėl tiriama ir kuriama sistema tai leis atlikti vos per kelias sekundes, o ne per kelias valandas ir daugiau, kaip tai yra dabar. Taip pat vartotojas, bet kada galės gauti išsamią finansinę informaciją, kurios dabar negali gauti, nes paprasčiausiai niekas individualiai nerenka duomenų, kiek kur ir koks asmuo išleido, nebent pats asmuo renka sau tokią informaciją. Be to sistema, bet kuriuo laiku rodys realią universiteto aplinkoje vykstančių finansinių operacijų apimtį (ten kur ji bus įdiegta), nes dabar visa finansinių operacijų informacija gaunama dienos ar mėnesio pabaigoje, po to kai atliekami „dideli“ ir labai varginantys skaičiavimai. Sukurta programinė įranga leis pasinaudojant studento ar darbuotojo pažymėjimu atlikti apmokėjimą už paslaugas (tiesiog pinigai bus pervedami iš vienos sąskaitos į kita), taip pat bus kur kas saugiau nes nereikės turėti grynųjų pinigų, kuriuos gali kas nors pavogti. Taip pat bus galima peržiūrėti savo sąskaitos likutį, atlikti pervedimus elektroninio apmokėjimo sistemos teikiamomis galimybėmis.

Tačiau žvelgiant į elektroninių mokėjimų universitetinėje aplinkoje sistemos galimybes kyla klausimas. Kodėl negalima būtų tam panaudoti, gerai žinomų: SEB Vilniaus bankas, HansaBankas, DnB Nord bankas, Snoras bankas, ir kitų, bankų teikiamų elektroninių paslaugų. Kiekvienas bankas naudoja savo individualią finansinių operacijų sistemą, todėl yra labai sunku suderinti visus bankus. Be to finansinės transakcijos tarp pardavėjo ir banko trunka labai ilgai. Atsiskaitymų informacija gaunama tik nustatytais terminais. Galiausiai,

bankai apmokestina visas savo teikiamas paslaugas, todėl iš karto kyla problema, kas padengs aptarnavimo išlaidas universitetas, ar mokėtojas. Todėl mažavertės paslaugos, su pridėtinu banko aptarnavimo paslaugų įkainiu, tampa vis didesnės (pvz. kopijuojam ar spausdiname 1 lapą, kaina būtų kopijavimas ar spausdinimas 0,10 Lt (tokia kaina buvo 2008 balandį Kauno technologijos universitete) plius 0,80 Lt banko paslaugų aptarnavimo mokestis (tokia vidutinė kaina buvo 2008 balandį Lietuvos bankuose), viso gaunasi 0,90 Lt, todėl paslauga šiuo atveju pabrangsta net 9 kartus). Naudojant elektroninių mokėjimų universitetinėje aplinkoje sistemą: finansinės transakcijos būtų atliekamos labai greitai, visa informacija būtų gaunama realiu laiku, nekiltų suderinamumo su bankais problemos, be to nebūtų naudojamas paslaugos teikimo mokestis, to pasekoje net ir mažavertės paslaugos neužbrangtu.

Naudojant elektroninių apmokėjimų universitetinėje aplinkoje programinę įrangą, finansinio atsiskaitymo operacijos bus atliekamos kur kas greičiau nei tradiciniu būdu, o be to bus pateikiama visa išsami finansinių operacijų informacija.

2.4. Egzistuojantys sprendimai

Pasaulyje yra nemažai koledžų ir universitetų, kurie naudodami naujausias technologijas yra sukūrę ir įsidiegę bei naudoja apmokėjimo universitetinėje aplinkoje sistemas. Kurios teikia paslaugas nuo galimybės susimokėti už mokslą iki apsipirkinėjimų elektroninėse parduotuvėse.

2.4.1. Harvard's Electronic Shop and Pay

Electronics Shop and Pay[2] (ESP) sistema Harvard universitete yra plačiai naudojama, kaip žiniatinklio pagrindinis įrankis užsisakinėjant prekes bei paslaugas, bei jas apmokant, t.y. atsiskaitant už prekes ir paslaugas su atitinkamais pardavėjais.

Sistema leidžia iš vienos vietos pasiekti daugelio pardavėjų svetaines, įskaitant pagrindines paslaugas, su mokslu susijusias paslaugas, WRM Scientific[3] (t.y. elektroninė parduotuvė prekiaujanti pramonei, vyriausybei, mokslo įstaigoms, mokslininkams, elektronika, farmacijos įstaigoms ir pan. reikalinga laboratorine įranga ir priemonėmis). Visa informacija yra patalpinta specialiame pardavėjų sąrašė, kuris kitaip gali būti vadinamas Harvard-specifiniu katalogu, kuriame prekių kainos yra pateikiamos su ženklia nuolaida, nei tiesiogiai pas pardavėjus.

Visi ESP sistemos pardavėjai turi specialų profilį, kuris atitinka Oracle navigacinį langą ir jų profilis yra vadinamas HRVD^ESP^Shopping. Paspaudus šį profilį bus galima patekti į pardavėjo sistemą (t.y. pardavėjo svetainę sujungtą su ESP sistema). Pardavinėti gali visi, kurių elektroninės parduotuvės tenkina griežtus Harvard universiteto nustatytus reikalavimus. Sistemoje yra pateikiama išsami informacija apie pardavėjus ir jų prekes bei paslaugas. Parduodamas prekes ir paslaugas galima iš karto užsisakyti ir atlikti apmokėjimus. Taip pat pateikiama išsami informacija apie kliento sąskaitos balansą, ir atsiskaitymus su pardavėjais.

Į ESP sistemą buvo įtraukta galimybė atsiskaityti už mokslo paslaugas, kas susilaukė itin didelio dėmesio ir pasisekimo. Tad dabar didžioji dalis atsiskaitymų už mokslą transakcijų paslaugas eina per ESP sistemą.

Aukštoji mokykla tikisi, kad visi vartotojai tinkamai įvertins tinkamai ESP sistemą, naudosis jos teikiamomis paslaugomis ir apsipirkinės šioje sistemoje, ir to pasėkoje bus pritraukta dar daugiau pardavėjų[4].

Electronics Shop and Pay sistemos galimybės:

- Peržiūrėti sąskaitas
- Apmokėti įvairias sąskaitas
- Peržiūrėti partnerių siūlomas prekes
- Apsipirkti partnerių svetainėse
- Sąskaitos balanso kontrolė
- Automatinis sąskaitų apmokėjimas
- Sąskaitų istorija
- Automatiniai pranešimai elektroniniu paštu apie naujas sąskaitas
- Automatiniai elektroninių apmokėjimų patvirtinimai elektroniniu paštu
- Veikla 24 valandas per parą ir 7 dienas per savaitę

2.4.2. Financial Management Service

Financial Management Service[5] (FMS) sistema yra plačiai naudojama Indiana universitete. Tai yra gerai žinoma sistema tarp universitetinių apmokėjimo sistemų, ir gal būt tobuliausia.

FSM sistema veikia žiniatinklio pagrindu, todėl visos sistemos teikiamos paslaugos yra prieinamos per naršyklę. Esminis šios sistemos bruožas yra tas, kad sistema teikia daug paslaugų sistemos vartotojams, todėl yra sukurti specialūs servisai skirti aptarnauti vartotojus.

Sistemos servisai aptarnaujantys vartotojus:

- *Sąskaitų valdymas* – tai servisas konstruojantis visas vartotojų sąskaitas, bei užtikrinantis vartotojo teisėtų veiksmų galimumą.
- *Sąskaitų apmokėjimo galimumas* – tai servisas kontroliuojantis sąskaitų balansą, bei mokumo galimybes, kaip kad atsiskaitymą kortelėmis, čekiais ir pan..
- *Sąskaitų informacijos gaunamumas* – tai servisas kontroliuojantis visą gaunamą informaciją susijusią su vartotojo sąskaitomis, kaip kad: gautos sąskaitos kurias reikia apmokėti, apmokėjimų ataskaitos, pervedimų ataskaitos, įvairios informacijos išgavimo galimybės ir pan.
- *Pagalbinė sąskaityba* – tai servisas leidžiantis atlikti įvairius pagalbinius veiksmus su sąskaitomis, kaip kad sąskaitų sąskaityba, sąskaitų grupavimas, prioritetų skirstymas ir pan.
- *Pagrindinės informacijos valdymo sistema* – tai servisas valdantis visą sąskaitų informaciją ir sąskaitų veiksmų galimumą, susijusį su universitetu.

- *Apmokėjimo kontrolė* – tai servisas kontroliuojantis apmokėjimo operacijas.
- *Sutarčių ir subsidijų administravimas* – tai servisas kontroliuojantis sutarčių sudarymą, bei subsidijų administravimą.
- *Algalapiai* – tai servisas tvarkantis visą informaciją susijusią su algalapiais ir algomis.

FMS turi papildomus komponentus, kurie užtikrina sklandų sistemos darbą.

Sistemos komponentai užtikrinantys sklandų sistemos darbą:

Financial Informatikos System (FIS) – komponentas atsakingas už finansiniu operacijas vykstančias sistemoje.

Šis komponentas susideda iš dviejų komponentų:

- *Transakcijų valdymas* – tai komponentas kontroliuojantis visas sistemoje vykstančias transakcijas, dokumentų pateikimą vartotojui, pagrinde t.y. sąskaitos, jis apjungia vienodus ar panašius dokumentus į vieną visumą, taip supaprastinamas vartotojo darbas.
- *Sprendimų valdymas* – komponentas kontroliuoja sistemos „sprendimus“, t.y.: analizė, atvaizdavimas, spausdinimas, grupavimas į eiles, į lygius, ataskaitų darymas, santraukų darymas, skirstymas pagal datą, transakcijų resursų paskirstymas ir pan.

Atsižvelgiant į Indiana Universiteto Financial Management Service (FMS) sistemą ir jos finansines galimybes, kurių šioje sistemoje yra iš ties labai daug, galima drąsiai teigti, kad tai yra viena tobuliausių apmokėjimo sistemų universitetinėje aplinkoje[6].

Financial Management Service sistemos galimybės:

- Peržiūrėti sąskaitas
- Apmokėti įvairias sąskaitas
- Automatiškai tvarkoma sąskaitų informaciją
- Atliekami automatiniai apmokėjimai
- Sąskaitos balanso kontrolė
- Sąskaitų istorija
- Automatiniai pranešimai elektroniniu paštu apie naujas sąskaitas
- Automatiniai elektroninių apmokėjimų patvirtinimai elektroniniu paštu
- Veikla 24 valandas per parą ir 7 dienas per savaitę

2.4.3. E-payment

E-payment[7] yra apmokėjimo sistema taikoma Trent universitete. Šios universiteto studentai gali susimokėti sąskaitas už mokslą būdami namuose ar kokioje nors kitokioje buveinėje naudodamiesi internetu.

Studentai gali sumokėti mokesčius naudodamiesi, bet kokia populiaria atsiskaitymo sistema, nes visos populiarios finansines paslaugas teikiančios sistemos yra integruotos Trent universiteto IT skyriaus į E-payment sistemą.

Taip pat galima laisvai atsiskaityti naudojantis Visa, MASTERCARD aptarnaujamas American Express aptarnavimo centru.

Ši paslauga yra ypač tinkama studentams ir jų tėvams, kadangi visa reikiama informacija yra pateikiama laiku, be to ypač patogiu ir lengvu apmokėti sąskaitas. Apmokėjimo paslauga teikiama per žiniatinklį yra ypač patogi Trent universitetui tvarkant finansinę informaciją, todėl yra aišku, kurie studentai susimokėjo už mokslą, o kurie ne. Tai ženkliai sumažina studentų laukimo laiką, todėl yra pilnai aišku, kuriuos studentus reikia išbraukti iš studentų sąrašų, ir nebelaukti.

Teigiama, kad E-payment sistema padeda eliminuoti ilgą laiko tarpą, kada studentas sugalvoja grįžti į mokslus, pasibaigus semestru.

E-payment sistemos privalumas yra tas, kad studentai gauna iš kart sąskaitas, kai jos tik išrašomos, kurias turi apmokėti. Iš kart jas gali peržiūrėti ekrane, ar atsispausdinti,. Ir jai turi pakankamai pinigų savo sąskaitoje, gali jas apmokėti (t.y. patvirtinti apmokėjimo transakciją).

Apmokėjimas yra įvykdomas per 24 ar 72 valandas, t.y. kur kas greičiau, nei alternatyviais apmokėjimo būdais. Visa žiniatinklio puslapiams perduodama finansinė (sąskaitų, kortelių, sąskaitų ir pan..) informacija yra koduojama

Kol kas studentai naudodamiesi Trent universiteto E-payment sistema gali apmokėti sąskaitas už mokslą, tačiau ateityje, studentai naudodamiesi šia sistema, galės apmokėti ir sąskaitas už kitokias paslaugas[8].

E-payment sistemos galimybės:

- Peržiūrėti sąskaitas
- Apmokėti universitetines sąskaitas už mokslą
- Sąskaitos balanso kontrolė
- Sąskaitų istorija

- Apmokėjimo įvykdymas per 24 ar 72 valandas
- Veikla darbo dienomis ir darbo laiku

2.4.4. Stanford ePay

Stanford ePay[9] yra Stanford universiteto elektroninio sąskaitų išrašymo ir elektroninio apmokėjimo sistema. Sistema teikia galimybę studentams ar autorizuotiems mokėtojams pamatyti savo universitetines sąskaitas ir jas apmokėtais. Mokėtojai apmokėti sąskaitas gali 24 valandas per parą, t.y. bet kuriuo paros metu.

Universitetinę sąskaitą sudaro: mokestis už mokslą, nuoma, maitinimas, sveikatos draudimas, medicininė priežiūra, gyvenamosios vietos mokesčiai, pašto paslaugos, kabelinės televizijos paslaugos. Tačiau negalima susimokėti baudos už negražintas laiku į biblioteką ar pamestas knygas, parkingo paslaugas, įvairių renginių bilietus, knygas.

Autorizuoti studentai gali į Stanford ePay sistemoje esančia savo sąskaitą įtraukti kitus narius, atlikti apmokėjimams, kad studentams būtų patogiau. Kiti individualūs nariai įtraukti į studento sąskaitą yra vadinami „Autorizuotais mokėtojais“. Studentas gali nurodyti vieną ar daugiau autorizuotų mokėtojų. Pavyzdžiui autorizuotais mokėtojais galėtų būti: studentų tėvai, rėmėjai, šeimos nariai. Autorizuoti mokėtojai studento sąskaitoje turi itin paprastas funkcijas, kaip kad: pamatyti apmokėjimų ir sąskaitų istoriją, tačiau apmokėjimus gali matyti tik tuos, kuriuos yra padarę patys studento sąskaitoje, ir negali matyti kitų autorizuotų mokėtojų padarytų apmokėjimų.

Studentai naudodami Stanford ePay sistemą gali atlikti:

- Apmokėti savo universitetines sąskaitas internete.
- Autorizuoti mokėtojai ar kiti apmokėti studentų universitetines sąskaitas internete.
- Automatizuoti periodinius apmokėjimus.
- Peržiūrėti sąskaitų istoriją.
- Atlikti depozitus, susijusius Stanford'o universiteto debito paslaugomis.

Apmokėjimo metodą sudaro:

- eCheck – elektroniniai pervedimai iš JAV finansinių paslaugų teikimo ir kontrolės institucijų.

- MASTERCARD, American Express, ir Discover kreditinės kortelės – tai sąskaitų apmokėjimas naudojant kreditines korteles. Tačiau studentai naudodami kreditines korteles, negali atlikti depozitinių apmokėjimų susijusių su Stanford'o universiteto debito paslaugomis.

Stanford ePy sistema siuntinėja elektroniniu paštu pranešimus studentams apie jiems atėjusias universitetines sąskaitas, taip pat apie tai (apie studentams atėjusias universitetines sąskaitas) siuntinėjami pranešimai elektroniniu paštu ir studentų autorizuotiems mokėtojams. Stanfords ePy sistemoje yra pateikiama išsami informacija apie sąskaitas ir jų istoriją., t.y. pateikiama išsami informacija, netik apie nesenas sąskaitas, bet ir apie senas sąskaitas[9].

Stanford ePay sistemos galimybės:

- Peržiūrėti sąskaitas
- Apmokėti įvairias sąskaitas susijusias su universiteto teikiamomis paslaugomis.
- Automatiškai tvarkoma sąskaitų informacija
- Atliekami automatiniai apmokėjimai
- Sąskaitos balanso kontrolė
- Išsami visų rūšių sąskaitų istorija
- Automatiniai pranešimai elektroniniu paštu apie naujas sąskaitas
- Automatiniai elektroninių apmokėjimų patvirtinimai elektroniniu paštu
- Veikla 24 valandas per parą ir 7 dienas per savaitę

2.4.5. SDePay

SDePay[10] yra elektroninių sąskaitų pateikimo ir elektroninio apmokėjimo sistema. Ši sistema aptarnauja visus studentus ir jų sąskaitas ir sąskaitų apmokėjimus, kurie mokosi South Dakota State universitete ir South Dakota Public universitete.

Studentams visos Universitetinės sąskaitos yra pateikiamos tik SDePay sistemoje, o elektroniniu paštu gauna tik pranešimą, apie atėjusias sąskaitas. Sąskaitos gali būti: mokestis už mokslą, maitinimą, gyvenamąją vietą, honorarai ir t.t..

Sistemoje dalyvauja ne tik studentai, bet ir studentų svečiai, t.y. autorizuoti mokėtojai. Studentų autorizuotais mokėtojais gali tapti individai, kuriuos autorizuoja studentai, tokias autorizuotais mokėtojais gali tapti: tėvai, rėmėjai ir t.t. Svečiai gali peržiūrėti sąskaitas, jei reikia jas atsispausdinti, apmokėti, peržiūrėti sąskaitų ir apmokėjimų istoriją. Tačiau svečias

apmokėjimo istorijoje gali peržiūrėti tik tuos apmokėjimus, kuriuos yra pats padaręs. Studento svečias šioje sistemoje turi mažiau teisių ir galimybių, nei studentas.

SDePAY sistema veikia 24 valandas per parą ir 7 dienas per savaitę. Sąskaitų informacija ir apmokėjimų informacija į sistema patenka iš kart. Apmokėjimai įvykdomi iš kart. Apmokėjimo patvirtinimai siunčiami iš kart į studentų ir/ar jų svečių elektroninius paštus. Sąskaitų ir apmokėjimų informacija yra privati ir griežtai saugoma.

Studentai ir jų rėmėjai gali apmokėti sąskaitas pasinaudodami JAV ir pasaulyje populiariais apmokėjimo būdais:

- eCheck – elektroniniai pervedimai iš finansinių paslaugų teikimo ir kontrolės institucijų.
- MASTERCARD, American Express, ir Discover kreditinės kortelės ir debitinės kortelės – tai sąskaitų apmokėjimas naudojant kreditines korteles. Tačiau studentai ir jų rėmėjai negali apmokėti sąskaitų naudodamiesi VISA kortelėmis.
- Paprasti asmeniniai čekiai.

Studentams ir jų svečiams naudotis SDePay sistema yra itin paprasta, studentams tereikia prisiregistruoti prie sistemos ir priregistruoti savo svečius (svečiams visa reikiam informacija yra išsiunčiam automatiškai nurodytu elektroninio pašto adresu). Patogumo dėlei į elektroninį paštą atėjęs pranešimas turi specialią nuorodą, kurios pagalbą galima itin lengvai prisijungti prie sistemos (negaištamas brangus laikas suvedinėjant prisijungimo informaciją ir ieškant sąskaitos), ir atlikti apmokėjimą tinkamu būdu.

SDePay sistema turi iš tiek daug galimybių, skirtų peržiūrėti ir apmokėti sąskaitas, be to šia sistema itin paprasta naudotis. Todėl nenuostabu, kad ši sistema susilaukė itin didelio dėmesio iš vartotojų, ir dabar paprasto – popierinės sąskaitos nebespausdinamo ir nebesiuntinėjamos studentams ir jų rėmėjams[11].

SDePay Sistemos galimybės:

- Peržiūrėti sąskaitas
- Apmokėti įvairias sąskaitas
- Automatiškai tvarkoma sąskaitų informaciją
- Sąskaitos balanso kontrolė
- Sąskaitų istorija
- Automatiniai pranešimai elektroniniu paštu apie naujas sąskaitas
- Automatiniai elektroninių apmokėjimų patvirtinimai elektroniniu paštu
- Greitai įvykdomi apmokėjimai
- Svečiai-mokėtojai

- Tiesioginis prisijungimas prie sistemos iš vartotojo elektroninio pašto
- Veikla 24 valandas per parą ir 7 dienas per savaitę

2.4.6. Enroll & Pay

Enroll & Pay[12] yra Kansas universiteto elektroninių sąskaitų pateikimo ir elektroninio apmokėjimo sistema.

Studentai naudojantys Enroll & Pay sistema gali peržiūrėti jiems pateiktas sąskaitas ir apmokėti tiesiogiai sistemoje. Sąskaitos gali būti: mokestis už mokslą, gyvenamąją vietą, nuomą, baudos už laiku negražintas knygas ir pan.). Sąskaitas galima apmokėti kreditinėmis kortelėmis, čekiais, ar populiariomis elektroninio apmokėjimo paslaugas teikiančiomis sistemomis.

Sistemoje studentai taip pat gali registruoti autorizuotus mokėtojus. Studentų autorizuotais mokėtojais gali būti : tėvai, rėmėjai, šeimos nariai ir pan.

Studentų sąskaitų informacija yra atnaujinama iš kart, kai tik pasirodo informaciją apie naujas sąskaitas, ir iš kart siunčiamas pernešimas studentui ir jo autorizuotiems mokėtojams apie naują sąskaitą. Sąskaitos sistemoje pateikiamos detalizuotai ir suskirstytos į atitinkamas grupes. Taip pat atlikus apmokėjimą, kuris sistemoje atliekamas iš kart, kai yra patvirtinamas, yra siunčiamas pranešimas į studento ir jo autorizuotų mokėtojų elektroninius paštus. Sistema įsimeina mokėtojų finansinių sąskaitų duomenis, todėl kitą kartą norint apmokėti sąskaitas, nebereikia, suvedinėti, kortelių, čekių, apmokėjimų sistemų duomenų, tereikia, tik nurodyti reikiamą sumą, ir sąskaitą už kurią mokama.

Enroll & Pay sistema pateikia detalę sąskaitų ir apmokėjimų istoriją (sistemoje saugoma visa kada nors buvusių sąskaitų ir apmokėjimų informacija). Sąskaitų istoriją pilnai gali peržiūrėti studentai ir jų autorizuoti mokėtojai, o mokėjimų pilną istoriją tik studentai, autorizuoti mokėtojai gali peržiūrėti tik apmokėjimų istoriją, susijusią su jų pačių atliktais apmokėjimais.

Enroll & Pay sistema, visą esančią informaciją, ir transakcijas koduoja SSL protokolu. Tokiu būdu yra užtikrinamas sistemos vartotojų privatumas ir finansinės informacijos konfidencialumas.

Kansas Universiteto Enroll & Pay sistema kol kas gali pateikti tik universitetinis sąskaitas, ir priimti už jas apmokėjimus, bet planuojama, kad ateityje, studentai turės galimybę apmokėti ir kitokio pobūdžio sąskaitas, užsisakyti įvairias paslaugas[13].

Enroll & Pay sistemos galimybės:

- Peržiūrėti sąskaitas
- Apmokėti įvairias universitetines sąskaitas
- Sąskaitos balanso kontrolė
- Automatiniai pranešimai elektroniniu paštu apie naujas sąskaitas
- Automatiniai elektroninių apmokėjimų patvirtinimai elektroniniu paštu
- Autorizuoti mokėtojai
- Detali, bet kada buvusių sąskaitų istorija
- Automatinis sąskaitų apmokėjimo formų pildymas
- Veikla darbo dienomis ir darbo laiku

2.4.7. Temple universiteto TUpay sistema

Kas tai yra

Tupay[14] yra oficiali sąskaitų išrašymo ir apmokėjimų sistema aktyviems-įtrauktiems į sąrašą studentams Temple universitete. Sąskaitų tvirtinimai gali būti atliekami tik realiu laiku. Studentas gauna e-laišką pranešimą, nes tai oficiali Temple universiteto sistema vartotojus informuoja kad jų e-sąskaitos yra nagrinėjamos. Prie TUpay sistemos galima prieiti adresu[15] arba prisijungti prie OWLnet tiesiogiai[16]. Galima mokėti realiu laiku elektroniniais pinigais be tikrinimo arba išsaugojimų arba kreditine kortele.[14]

Kaip tai veikia

Žmogus gauna e-laišką kai jo e-sąskaita įsigalioja ir būna paruošta apmokėjimams TUpay sistemoje. Laiškas nusiunčiamas į universiteto teikiama el. paštą priklausanti tam tikram vartotojui. Jeigu yra dar kitų mokėtojų TUpay sistemoje, tokių kaip tėvai, globėjas, laiduotojas; ir jie yra išvardinti el. pašto adrese TUpay sistemoje, jie gali taip pat gauti e-laišką perspėjanti kad e-sąskaita galioja[14].


Kaip įvykdyti apmokėjimą

Sąskaitos apmokamos per TUpay: apmokėjimas priimamas e-čekiu arba kreditinė kortelė;

E-čekis yra greitas ir saugus metodas elektroniniu būdu apmokėti e-sąskaitą[14]

E-sąskaitos privalumai

- Studentų slaptumas – sąskaita galima internetiniame puslapyje ir ten nėra išvardinta studentų pažymių ir kt.
- Automatinis pranešimas į el. paštą kai sąskaita keičiasi
- Patogus priėjimas – galiojantis 24 valandas per parą ir 7 dienas per savaitę.
- Informacija saugi ir konfidenciali – tik vartotojas ir jo patvirtinti asmenys gali matyti jo e-sąskaitą.
- Priėjimas prie 16 mėnesių sąskaitybės istorijos[14].



STATEMENT OF ACCOUNT
as of 07/23/2006 Victor E. Owl

Easy Payment Plan Due:
The Easy Payment Plan Due reflects the minimum amount due. If you pay at least the "Easy Payment Plan Amount Due", you will be enrolled automatically in Temple's Easy Payment Plan, which consists of 3 payments per semester (Fall and Spring only). You will be assessed an \$11 fee per installment. The Easy Payment Plan payment represents 33% of your current balance plus any prior balance owed. For the first bill of each semester, you MUST pay the Easy Payment Plan Due amount in order to avoid having your classes cancelled. If the Easy Payment Plan Due amount is \$0, you MUST confirm your registration. You can confirm your registration through the Financial Services Tab in OWL.net.

TUID	Residency Status	Total Amount Due	Easy Payment Plan Due	Due Date
999999999	Pennsylvania Resident	3,749.16	1,260.72	08/10/2006

Account Activity

Transaction Date	Term	Transaction	Amt.	Sub total
		Account Balance as of Last Statement (date)	74.00	
06/16/2006		Check	(74.00)	
07/08/2006	Fall 2006	Tuition	4570.00	
07/08/2006	Fall 2006	Computer / Tech Fee	100.00	
07/08/2006	Fall 2006	General Activity Fee	35.00	
07/08/2006	Fall 2006	Stud. Facilities Fee	25.00	
07/08/2006	Fall 2006	Student Health Fee	60.00	
07/08/2006	Fall 2006	Recreation Serv Fee	30.00	
07/08/2006	Fall 2006	Health Insurance Int	398.16	
07/12/2006	Fall 2006	Residence Hall Chrg	2,529.00	
07/12/2006	Fall 2006	Meal Plan	977.00	
07/12/2006	Fall 2006	Diamond Dollars	150.00	
07/18/2006	Fall 2006	PHEAA Grant	(875.00)	7,999.16
		Anticipated Aid		
	Fall 2006	Pell Grant	(500.00)	
	Fall 2006	Federal Stafford Loan	(2,750.00)	
	Fall 2006	Federal S.E.O.G. Grant	(1,000.00)	(4,250.00)
		Total Amount Due		3,749.16

Questions: E-mail bursar@temple.edu or contact Student Financial Services at 215-204-2244.

Paveikslas 2.4-1. Temple University Payment System Statement [14].

2.4.8. John Hopkins universiteto elektroninio apmokėjimo sistema E-Billing

Kas tai yra

John Hopkins universitetas pristato e-sąskaitų apmokėjimo sistemą. Elektroninė sąskaita yra oficiali priemonė generuoti sąskaitas už mokslą visiems Peabody studentams. Tai reiškia kad popierinės sąskaitos nėra siunčiamos paštu užsiregistravusiems studentams, nebent specialiai reikia[17].

Kaip tai veikia

Kai studentai prisijungia prie sistemos pirmą kartą, naujos elektroninės sąskaitos sistema automatiškai siunčia e-laišką pranešimą į studentų universiteto elektroninį paštą, taip ir kai nauja sąskaita yra paruošta prisijungimui. Studentam reikia jų ID ir slaptažodžio tam kad prisijungti prie sąskaitų vaizdavimo ir apmokėjimo sistemos. Autorizuoti mokėtojai naudoja prisijungimo vardą ir slaptažodį. Kiekvienas prisiregistravęs studentas yra automatiškai įtraukiamas į elektroninį sąskaitos atvaizdavimą[17].

Ką galima padaryti prisijungus

- pamatyti studentų sąskaitas realiam laike
- apmokėti studentų sąskaitas realiu laiku su kreditinę kortele(MasterCard, Visa)
- parsisiųsti ir atspausdinti savo sąskaitos kopiją
- pamatyti sąskaitybos ir apmokėjimo istoriją
- persiųsti e-laiško kopija tėvams, globėjams
- priskirti asmenys kurie gali atlikti pamokėjimus už tavo sąskaita[17].

E-sąskaitos privalumai

- patogiu ir paprastu
- asmeniškai ir saugu
- nėra laukimų norint padaryti apmokėjimą
- priminimai ir patvirtinimai siunčiami į el. paštą
- elektroninių sąskaitų ir apmokėjimų istorija iki 12 mėnesių
- priėjimas prie vartotojo srities bet kur per internetą[17].

2.4.9. Oregono universiteto elektroninio apmokėjimo sistema QuikPAY

Kas tai yra

Oregonos universitetas naudoja QuickPay[18], elektroninę sistemą kad visi studentai galėtų atsispausdinti jų sąskaitas, atlikti apmokėjimus realiu laiku, nustatyti trčius asmenys kaip autorizuotus mokėtojus, kurie gali daryti tą patį. Per QuickPay sistemą, elektroninis laiškas siunčiamas kiekvienam studentui į oficialu universiteto elektroninį paštą[18].

Kaip tai veikia

- Elektroninio laiško pranešimas, kad studento sąskaitos ataskaita yra paruošta peržiūrai/apmokėjimui, yra išsiunčiamas į oficialų studento elektroninio pašto adresą, ir autorizuotų mokėtojų elektroninius paštus taip pat[18].
- Studentas prisijungia prie Duckweb naudodamas ID numerį ir asmeninį priėjimo kodą. Iš Duckweb meniu galima saugiai patekti į QuickPAY sistemos puslapį. Autorizuoti mokėtojai gali prieiti prie sistemos tiesiog paspaudę nuorodą, kuri yra atsiųsta jiems į paštą[18].
- Prisijungęs studentas mato “Message Board” QuickPAY sistemos puslapyje. Svarbi universiteto informacija matoma toje skelbimų lentoje ir pastoviai atnaujinama[18].
- Pasirinkus “View accounts”, tada “Current Billing Statatement” galima matyti einama studentų sąskaita[18].
- Galima atlikti apmokėjimą, atsispausdinti sąskaitos ataskaitą, ir galimybe matyti 16 mėnesių ataskaitas[18].

2.5. Elektroninio apmokėjimo sistemos samprata

2.5.1. Elektroninio apmokėjimo sistemos veikimo principas

Elektroninio apmokėjimo procedūroje dalyvauja, pirkėjas (vartotojas), paslaugos teikėjas, bankas. Visi sistemos dalyviai turi būti registruoti sistemoje. Taip pat sistemoje dalyvauja operacijų koordinatorius, kuris koordinuoja visas vykdomas operacijas, tarp paprasto tinklo ir apsaugoto banko tinklo. Galima teigti, kad tokiu būdu yra užtikrinamas ryšys tarp tradicinės apmokėjimo infrastruktūros ir elektroninio apmokėjimo infrastruktūros. Viename tinklo gale yra pirkėjas ir paslaugos teikėjas, ir jie abu turi sąskaitas banke, ir jie abu prisijungia prie apsaugoto banko tinklo. Paslaugos teikėjo bankas pateikia visą reikalingas priemones apmokėjimui įvykdyti. Kai pardavėjas patvirtina apmokėjimo pareikalavimą bankas įregistruoja šį reikalavimą.

Peikėjas gauna pareikalavimą apmokėti sąskaitą, tuomet jis pateikia visą reikiamą finansinę informaciją ir patvirtina apmokėjimą. Prieš įvykdant pervedimą yra nustatomas autorizuotas kelias tarp pirkėjo, pardavėjo ir banko, visa tai atlieka operacijų koordinatorius. Jei viskas gerai, tai pinigai iš pirkėjo sąskaitos pervedami į paslaugos teikėjo sąskaitą. Jei viskas gerai tai visiems dalyviams išsiuntinėjami patvirtinimo pranešimai, jei kas nors negerai, tai padedama kartoti viskas nuo atitinkamo etapo[19].

2.5.2. Transakcijos

Piniginiams pervedimams (apmokėjimams) elektroninio apmokėjimo sistemose yra naudojamos transakcijos.

Transakcija - veikslių su duomenų baze seka, kuri turi pavykti arba nepavykti . Jei transakcija nepavyko todėl, jog jos metu kompiuterinė sistema nustojo dirbusi, naujai paleidus sistemą joje neturi likti neužbaigtos transakcijos pėdsakų. Pavedant pinigus, t.y. mokant už paslaugas, pervedama suma nuskaičiuojama nuo vienos sąskaitos ir pridedama prie kitos. Normaliam darbui būtina, jog, nepriklausomai nuo, bet kokių aplinkybių turi būti atlikti arba abu šie veiksmai, arba nė vienas[20].

Teisingai dirbančioje duomenų bazių sistemoje transakcijai būdingos šios savybės[20]:

- *Atomiškumas* - transakcijos operacijos pavyksta arba visos, arba nė viena.

- *Stabilumas* - tiek prieš transakciją, tiek ir po jos sistema yra normalioje darbo būsenoje.
- *Izoliacija* - tarpiniai transakcijos operacijos rezultatai yra nematomi. Matomas galutinis jos rezultatas, kuris iškart pakeičia pradinį duomenų bazės turinį.
- *Ilgalaikiškumas* - jei vartotojo programa gavo informaciją, jog transakcija pavyko, transakcijos rezultatai savaime išnykti nebegali[20].

Transakcija universitetinėje apmokėjimo aplinkoje veikia, tarp dviejų transakcijos dalyvių: pirkėjo (paslaugos vartotojo) ir pardavėjo (universiteto institucijos), ir dar naudojamos papildomos transakcijos tarp banko (universiteto finansinės institucijos).

Kad sėkmingai įvyktų transakcija turi būti įvykdomi keturi žingsniai[21]:

1. *Autentifikacija* – kiekvienas dalyvis turi identifikuotis sistemoje, kad būtų patvirtinta autorizacija, ir galėtų prasidėti transakcija.
2. *Sandoris* – šiame žingsnyje pirkėjas perveda pinigus pardavėjui, šiame žinginyje taip pat naudojamas autentifiškumo tikrinimas. Pabaigoje pirkėjas gauna patvirtinimą iš pardavėjo apie pervedimą.
3. *Patvirtinimas* – abu dalyviai vienas kitam siunčia patvirtinimus, kad apmokėjimas yra įvykdytas arba ne.
4. *Užbaigimas* – abu dalyviai išseina iš perdavimų kanalo, ir transakcija pasibaigia. Problemų atveju visa informacija yra išsaugoma ir pradedama viskas kartoti[21].

2.5.3. Apmokėjimo schemas

2.5.3.1. Apmokėjimo schemų reikalingumo samprata

Pasaulyje egzistuoja daug elektroninio apmokėjimo sistemų, kurios teikia įvairias galimybes. Galimybės yra pačios įvairiausios, tačiau visos vienaip ar kitai yra susijusios su saugumo reikalavimais. Saugumas yra pats svarbiausias apmokėjimo sistemų uždavinys. Tačiau apibendrintai galime teigti, kad visos sistemos veikia pagal, tam tikras apmokėjimą schemas. Apmokėjimo schema apjungia visus elektroninių apmokėjimo sistemų reikalavimus ir sukuria tam tikrą posistemę, kuri suvaidina svarbiausią vaidmenį vykdant apmokėjimus ir užtikrinant saugumą. Populiariausios apmokėjimo schemas yra: mokėjimas kortele, SET,

Skaitmeninis-kodinis apmokėjimas, Millicent ir kitos, čia aptarsime tik pačias svarbiausias[22].

2.5.3.2. Mokėjimas kortele

Tai yra kartelinė transakcija, kortelės savininkas, šiuo atveju paslaugos gavėjas siunčia kortelės informaciją paslaugos teikėjui, o šis savo ruožtu gali patikrinti pirkėjo autentiškumą ir mokumą. Visa tai paslaugos teikėjas gali patikrinti susisiekdamas su banku.

Kada vartotojas siunčia informaciją internetiniu tinklu, visa informacija yra koduojama, paprastai tam yra naudojamas SSL protokolas. Tačiau čia paslaugos gavėjas negali būti tinkamai apsaugotas nuo paslaugos teikėjo nesąžiningumo. Siekiant apsaugoti pirkėją, nuo pardavėjo nesąžiningumo, gali būti atmetas pirkėjo autorizavimas, t.y. naudojamas anonimiškumas, kai yra naudojama tik tiksliai reikalinga finansinė informacija (suasmeninta informacija nenaudojama). Tokiu atveju mokėjimo kortelės tampa prasčiausia pinigine[22].

2.5.3.3. SET

SET (saugi elektroninė transakcija) standartas dažniausiai naudojamas Visa ir MASTERCARD mokėjimo kortelėms. SET naudoja skaitmeninį sertifikatą, kuri identifikuoja prekybos narius, tam panaudojama viešo rakto kriptografija, siekiant apsaugoti transakciją.

Kada vykdomas prekybos sandoris, pirkėjas ir pardavėjas apsikeičia skaitmeniniais sertifikatais, tokiu būdu jie autorizuoja ir tokiu būdu yra patvirtinama, kad jie yra partneriai. Po to jie turi identifikuotis vienas kitam. Toliau pirkėjas siunčia savo užkoduotą apmokėjimo informaciją pardavėjui. Pardavėjas siunčia informaciją autorizuoti sistemai (bankui), kuri patikrina ar pirkėjo sąskaitoje yra pakankamai pinigų ir transakcija yra autorizuota. Jei viskas gerai, įvykdomas pervedimas. Ir siunčiama apie tai informacija pardavėjui, o jis apie tai informuoja pirkėją[23].

2.5.3.4. Skaitmeninis-kodinis apmokėjimas

Skaitmeninis-kodinis apmokėjimas yra nieko daugiau, kaip tik skaitmeninių simbolių rinkinys. Šia skaitmeniniai simboliai (kitai vadinami žetonais) yra gaunami iš banko. Bankas patikrina kiekvieną žetoną prieš jį išsiusdamas į kliento kompiuterį. Kada bankas gauna

virtotojo žetoną, jis jį pašalina iš virtotojo žetonų sąrašo. Kad nebūtų žetonų dublikatų, bankas kiekvienam žetonui suteikia serijinį numerį. Visi žetonų serijiniai numeriai yra saugomi banko duomenų bazėje. Kai virtotojas panaudoja žetoną, bankas jį įregistruoja, kaip panaudotą. Kai virtotojas panaudoja žetoną, bankas pastebi, kad jis yra jau panaudotas (iš žetono serijinio numerio), ir apie tai informuoja sandorio dalyvius. Kad būtų apsaugoti žetonai sistema turi dirbti realių laiku.

Skaitmenini-kodinis apmokėjimas panaudojama taip. Virtotojas norėdamas pasinaudoti skaitmeniniu-kodiniu apmokėjimas, iš banko turi atsisiusti ir įsidięti specialų žetonų generatorių, kuri jam generuos žetonus. Žetonas siunčiamas bankui užkoduotas slaptu raktu, bankas nežino kokį žetoną, virtotojas panaudojo, tačiau jis jį gali patikrinti, gautą informaciją išsikoduodamas viešu raktu. Tokiu būdu bankas nustato žetono autentiškumą.

Kai virtotojas apmoka sąskaitas, jo programa išsiunčia žetoną paslaugos teikėjui (pardavėjui), paslaugos teikėjas (pardavėjas) prideda įmokos informaciją ir viską išsiunčia bankui. Bankas patikrina žetoną, ir jai viskas gerai įvykdo pinigų pervedimą, ir informuoja virtotoją, ir paslaugos teikėją (pardavėją).

Skaitmeninio-kodinio apmokėjimo technologiją rekomenduoti naudoti mikro apmokėjimams[24].

2.5.3.5. Millicent

Millicent yra mikro apmokėjimo sistema išvystyta Digital Equipment Corporation. Millicent kodams naudoja skriptus, ir kiekvienas pardavėjas turi tik jį identifikuojantį skriptą. Čia dalyvauja papildomas pardavėjas – brokeris, kuris parduota pardavėjui pastarąjį identifikuojantį skriptą. Kada pirkėjas ateina pas pardavėją apsipirkti, jis susitinka su pardavėjo skriptu, kurio autentiškumą gali patikrinti naudodamas brokerio skriptą. Pirkėjas tai gali atlikti naudodamas specialius saugius protokolus. Finansinis atsiskaitymas vykdomas naudojant pardavėjo skriptą.

Kadangi millicent naudojamas mažiems apmokėjimas, tad jam nereikalingas stiprus šifravimas. Millicent naudoja specialią žymę saugumui užtikrinti. Žymė yra pranešimo dalis, kuri apsaugo pirkėjo informaciją. Pardavėjas patikrina apmokėjimą naudodamas žymę ir skaitmeninį numerį[25].

2.5.4. Elektroninio apmokėjimo formos

2.5.4.1. Elektroninio apmokėjimo formų tipai

- Secure presentation
- Vartotojų registracija
- Credit-debit instrumentai
- Elektronine valiuta
- Server scrip(kvitas)
- Duomenų perdavimas²⁶

2.5.4.2. Secure presentation

- Naudoja tradicinius kreditinės kortelės numerius
 - saugus kaip telefonu
 - galima didžiulė vartotojų bazė
 - mažas infrastruktūros poreikis
- Pavyzdžiai – produktai grindžiami:
 - SSL (Secure Socket Layer)
 - SHTTP(Secure Hypertext Transfer Protocol)
- Privalumai
 - nereikia pirkėjo parašo
 - pirklio teisėtumas
 - autorizacija realiu laiku
 - transakcijų kaina^[26]

2.5.4.3. Customer registration

- Vartotojai registruojas ir gauna slaptažodžius, raktus, ir naujus vartotojų identifikatorius
 - transakcijos visiškai per finansinio serviso teikėją kuris įeina į egzistuojančias finansinės sistemas

- apsaugo išorinių atsiskaitymų informaciją
- Pavyzdžiai:
 - First Virtual
 - CyberCash
 - SET
- Privalumai:
 - specifinių sistemos reikalavimų saugumas
 - autorizacija realiu laiku
 - transakcijų kaina[26]

2.5.4.4. Credit-debit instruments

- Finansinio serviso tiekėjas prižiūri pirkėjų atsiskaitymus
 - apmokėjimo įrankiai autorizuoja pervedimą
 - būdai: kreditas kaip kreditinė kortelė, debetas kaip sąskaita
 - reikalauja naujos infrastruktūros[26]
- Pavyzdžiai:
 - USC's NetCheque
 - FST C Electronic Chec Project
 - CMU's NetBill[26]
- Privalumai:
 - Specifinių sistemos įrankių saugumas
 - Agregacija ir ryšys finansinei sistemai
 - Sąskaitos informacijos trukmė[26]

2.5.4.5. Electronic currency(valiuta)

- Vartotojai perka valiuta ir valiutų serverių. Valiuta yra pateikta pirkliai, kuris įjungia ją valiutos serverį
 - anonimiškumo galimybė
 - autonominių operacijų galimybė
- Pavyzdžiai:

- Mondex
- Digicash
- NetCash
- Privalumai:
 - valiutos palaikymas
 - techninis pasipriešinimas nuo klastotojų
 - anonimiškumo lygis
 - tiesioginis prieš autonominį[26]

2.5.4.6. Server scrip

- Apmokėjimo įrankis naudojamas su individualiu pirkliu
 - scrip(kvitas) verifikavimas yra lokali svarstoma problema
 - reikalauja rinkos ir kitų apmokėjimo formų įgalinti pirklio pirkimo raštui
- Pavyzdžiai:
 - Milicent
 - Payword
- Privalumai:
 - pirkinių surinkimas gerina atlikimą
 - tiesa, turi valdyti daug valiutos rūšių[26]

2.5.4.7. Direct transfer

- Pirkėjas inicijuoja lėšų pavedimą į pirklio sąskaitą
 - gali paveikti įrankius siųsti išoriškai
- Pavyzdžiai:
 - dauguma on-line apmokėjimų mechanizmų
- Privalumai:
 - apmokėjimo parinkimas pirkėjui arba operacijai
 - sąskaitos valdymas panašus į credit-debit modelį[26]

2.6. Reikalavimai elektroninio apmokėjimo sistemoms

Trumpai aptarsime reikalavimus elektroninio apmokėjimo sistemoms, tai yra aptarsime reikalavimus, kurie yra neatsiejami, tarp klientų ir paslaugų teikėjų ir jų sistemų.

2.6.1. Reikalavimai

Sėkmė ar nesėkmė internetinės apmokėjimo sistemos pagrinde priklauso ne nuo naudojamų technologijų ir jų galimybių, bet nuo jas naudojančių vartotojų požiūrio ir supratimo. Vartotojo pripažinimas labai priklauso nuo daugelio priežasčių, bet ypač nuo rėmėjų, reklamos pozicijos, vartotojo pomėgių. Tad kada yra diskutuojamas apie internetines apmokėjimo sistemas, reikia galvoti netik apie internetinėms apmokėjimo sistemos naudojamoms technologijoms, bet ir vartotojo požiūrį į atitinkamus faktus ir pan.. Todėl kad elektroninio Apmokėjimo sistema patenkins visus lūkesčius ir jos vartotojai ją pamėgtų, reikia nusistatyti sistemos reikalavimus, ir pasistengti, kad ji juos visus įvykdytų[27].

Trumpai aptarsime reikalavimus elektroninio apmokėjimo sistemoms, tai yra aptarsime reikalavimus, kurie yra neatsiejami, tarp klientų ir paslaugų teikėjų ir jų sistemų.

Elektroninės apmokėjimo sistemos pagrindiniai reikalavimai galėtų būti[28,29,30,31,32]:

- *Privatumas* – turėtų būti užtikrintas klientų informacijos saugumas ir konfidencialumas[29].
- *Atomiškumas* – visos transakcijos turi būti užbaigto arba nepradėtos[29].
- *Bendradarbiavimas* – elektroninio apmokėjimo sistemos turi bendradarbiauti tarpusavyje[30].
- *Pajėgumas* – elektroninio apmokėjimo sistema turi turėti pakankamai resursų, kad galėtų tinkamai aptarnauti klientus[29].
- *Saugumas* – visos atliekamos transakcijos turi būti saugios, ir elektroniniai pinigai turi būti, kiek įmanoma apsaugoti[29].
- *Patikimumas* – elektroninio apmokėjimo sistemos turi veikti patikimai ir be jokių sutrikimų[31].
- *Patogumas* – elektroninio apmokėjimo sistema turi būti draugiška vartotojui, lengvai suprantama ir vartojama[29].

- *Automatiškumas* – elektroninio apmokėjimo sistema, kai kurias funkcijas atlieka automatiškai[31]
- *Anonimiškumas* – asmeninės informacijos pateikimo blokavimas[32].

Toliau šiuo punktus aptarsime plačiau.

2.6.2. Reikalavimų apžvalga

2.6.2.1. Privatumas

Privatumas suteikia identifikaciją, privatumą ir asmeninės informacijos individualumą, naudojant įvairius internetinių apmokėjimo sistemų metodus, kurie negali būti viešinami. Tačiau kai kurie elektroninio apmokėjimo metodai palieka tam tikrą pėdsaką individualios apmokėjimo informacijos. Pavyzdys galėtų būti kai yra naudojamos atsiskaitymui debitinės kortelės, yra įmanoma rasti dalį su apsipirkimu susijusios informacijos registre pas paslaugos teikėją ar banko duomenų bazėje. Tad kai kurios elektroninio apmokėjimo sistemos nevisai užtikrina privatumą. Kai kuriose sistemose, anonimiškumas gali apriboti tam tikrų detalių išgavimą, ir apsaugoti vartotojo itin svarbią informaciją. Tam yra priimti tam tikri įstatymai, kurie garantuoja vartotojų informacijos privatumą ir informacijos apsaugą finansinėse institucijose[29].

2.6.2.2. Atomiškumas

Atomiškumas garantuoja, kad vartotojo visos transakcijos bus tinkami užbaigtos, ir nebus padėtos į kokią nors vietą, dėl kokių nors priežasčių. Jei kokia nors transakcija elektroninio apmokėjimo sistemoje nulūžta, tai ji turi būti atkurta iš prieš tai buvusio būsenos. Ši funkcija turėtų būti panaši į transakcijų duomenų bazę iš kurios būtų galima atkurti, esant reikalui, reikiamą transakciją[29].

2.6.2.3. Bendradarbiavimas

Elektroninio apmokėjimo technologijos yra labai skirtingos, ir skirtingi vartotojai mėgsta skirtingas apmokėjimo sistemas. Skirtingos elektroninio apmokėjimo sistemos dirba

skirtingai, naudoja skirtingus darbo metodus, tačiau jos turi bendradarbiauti tarpusavyje. Jei sistemos yra suderinamos, tai jos turi turėti tam tikrą dalį per kurią turėtų būti galimybė jas suderinti. Gyvenimiška, situacija, kai yra galimybė suskirstyti sistemas į atitinkamas dalis, ir jas suderinti, kad bendradarbiautų. Bendradarbiavimas gali būti pasiekimas nustatant duomenų perdavimui atitinkamus protokolus ir infrastruktūrą, bei nustatant reikiamus standartus. Bendradarbiaujančio sistemos yra geriau vertinamos ir geriau pritaikomos, nei pavienės apmokėjimo sistemos. Kadangi technologijos greitai kaičiasi, o užtikrinti skirtingą elektroninio apmokėjimo sistemų bendradarbiavimą nėra taip lengva[30].

2.6.2.4. Pajėgumai

Elektroninio apmokėjimo sistemos iš vartotojų susilaukia vis didesnio ir didesnio pripažinimo, o kad sistemos galėtų patenkinti visus vartotojų lūkesčius jų pajėgumai turi nuolat didėti. Apmokėjimo sistemos turi priimti visus norinčius vartotojus ir juos tinkami aptarnauti. Kad būtų užtikrintas tinkamas paslaugų teikimo lygis, elektroninio apmokėjimo sistemos turi turėti pakankamus resursus, ypač reikiamą centrinių serverių kiekį. Centriniai serveriai ypač reikalingi vykdant apmokėjimo transakcijų procesą ir tikrinant apmokėjimo transakcijas. Kadangi užklausų kiekis į centrinius serverius nuolat auga, todėl ypač yra limituotas elektroninio apmokėjimo sistemų pajėgumas[29].

2.6.2.5. Saugumas

Saugumas yra pagrindinis rūpestis elektroninio apmokėjimo sistemų, t.y. kritinis elementas, kuris nulemia elektroninėse apmokėjimo sistemose vienokį ar kitokį elektroninio apmokėjimo metodų panaudojimo galimumą. Internetas yra atviras tinklas, todėl elektroninio apmokėjimo sistemos turi būti tinkamai apsaugotos, ir be jokios saugumo rizikos teikti tinkamai visas paslaugas vartotojams. Kada vartotojas apmoka sąskaitas elektroniniu būdu, jis turi būti įsitykinęs, kad jo piniginės transakcijos yra „sveikos“ ir saugios. Visos elektroninio apmokėjimo organizacijos, bankai, finansinės institucijos, jei nori išlaikyti klientus, ir iš jų susilaukti tinkamo įvertinimo, turi tinkamai apsaugoti visą klientų informaciją, kiek tai yra įmanoma[29].

2.6.2.6. Patikimumas

Bet kokia verslo, ar elektroninių apmokėjimo sistemų veiklą turi būti patikima ir efektyvi. Bet kokia elektroninio apmokėjimo sistema gal turėti trūkumų, priklausančių ar nepriklausančių nuo naudojamos technologijos, todėl visi pastebėti trūkumai turi būti tinkamai ištestuoti ir pašalinti, kad sistema būtų kuo patikimesnė. Yra daug priežasčių, kodėl sistema yra nepatikima vartotojų atžvilgiu. Kai kuriais atvejais tai saugumo grėsmės, dėl atitinkamų pritaikymų ir klaidingų situacijų[29].

2.6.2.7. Patogumas

Patogumas yra svarbiausia charakteristika interaktyvių produktų, tokių kaip elektroninio apmokėjimo sistemos. Elektroninio apmokėjimo sistemos turi būti draugiškos vartotojui, ir lengvai suprantamos. Bet kokios elektroninio apmokėjimo sistemos su komplikuotomis procedūromis, kompleksiniu apmokėjimo procesu, su veiksniais, kurie gali sukelti komplikotas situacijas, niekada negaus vartotojų pripažinimo. Sudėtingas elektroninės apmokėjimo sistemos naudojimas gali sunaikinti pačią sistemą. Kad elektroninio apmokėjimo sistema taptų draugiška vartotojui ir patogi naudojimui, ji turi būti sukurta taip, kad vartotojui, atliekant apmokėjimą, reikėtų atlikti kuo mažiau autorizacijos ir informacijos įvedimo veiksmų[29].

2.6.2.8. Automatiškumas

Šiandien informacijos kiekiai greitai auga, pas vartotojus elektroninio apmokėjimo sistemose kaupiasi sąskaitos, kurias sudėtinga aprėpti. Be to visas sąskaitas reikia apmokėti laiku, niekur nuo to pasislėpsi. Apmokėjimo informacijos įvedimas mokant už paslaugas yra labai varginantis ir daug laiko atimantis procesas, todėl elektroninio apmokėjimo sistemoje turėtų būti užtikrintas automatizuotas ar automatinis, reikiamos informacijos surinkimas, bei apmokėjimas[31].

2.6.2.9. Anonimiškumas

Anonimiškumas yra didelis rūpesnis apmokėjimo sistemos, o dar didesnis sistemų vartotojams. Anonimiškumas sukelia begales diskusijų, vieni teigia, kad jį reikia taikyti, kiti ne. Anonimiškumas yra susijęs su vartotojo duomenų slaptumu. Elektroninio apmokėjimo sistemos visada turi duomenis ne tik apie klientų sąskaitas, bet ir apie pačius klientus. O tai jau nebėra visiškais kliento konfidencialumas. Taip pat problema yra susijusi ir su elektroniniu atsiskaitymu, nes kai yra vykdomas atsiskaitymas pas paslaugos teikėją (pardavėją) iš kurio vartotojas įsigyja prekes/paslaugas patenka ne tik apmokėjimo informacija, bet ir konfidencialūs vartotojo (pirkėjo) duomenys, o tai yra negerai (paslaugos teikėjas (pardavėjas) gali panaudoti įgytą informaciją netinkamai). Todėl siūloma mikro apmokėjimuose naudoti anonimiškumą, kai atsiskaitant pas paslaugos teikėją patenka tik atsiskaitymui reikalingi duomenys[32].

2.7. Galimos grėsmės

Visas galimas grėsmes laisvai galime suskirstyti į dvi grupes, išorines ir vidines:

- *Išorinės* - kitaip vadinamos šnipinėjimų, kai yra šnipinėjami išoriniai komunikacijų tinklai. Šnipinėjimas gali būti vykdomas iš bet kurio komunikacijų tinklo taško. Ši grėsmė yra itin pavojinga ir dažniausiai taikoma. Nes kompiuterinis tinklas yra didelis, ir pilnai apsaugoti viso tinklo, kad joks šnipas negalėtų į jį patekti yra neįmanoma.
- *Vidinės* – tai yra ataka nukreipta į tam tikrą sistemos vidinę dalį, ir tai yra bandymas panaudoti naudingą informaciją, bei sistemos teikiama galimybe siekiant užvaldyti svetimą finansinę informaciją. Ši grėsmės rūšis gali būti pritaikyta, bet kokiai elektroninio apmokėjimo sistemai, kurioje yra kokia nors konfidenciali kliento informacija[33].

2.8. Elektroninio apmokėjimo saugumui taikomos technologijos

2.8.1. Saugumo technologijų reikalingumo samprata

Elektroninio apmokėjimo technologijos, laibiau nei kitos egzistuojančios technologijos turi teikti kokybiškas ir itin saugias paslaugas vartotojams. Kad vartotojo duomenys būtų saugūs ir pašaliečiams neprieinami yra naudojami apsaugos įvairūs apsaugos mechanizmai, kaip kad kriptografija, saugumą užtikrinantys protokolai. Kriptografija yra pati populiariausia apsaugos priemonių, ir visų apsaugos technologijų pagrindas. Kriptografija yra itin svarbi elektroninio apmokėjimo sistemos, todėl yra ypač daug dėl to diskutuojama[29]. Toliau apžvelgsime pačias svarbiausias kriptografijos technologijas naudojamas elektroninio apmokėjimo sistemose.

2.8.2. Duomenų saugumo ypatybės

Bet kokiai sistemai ar duomenims, kuriems taikomi saugumo principai, jie turi atitikti svarbias ypatybes, būtent slaptumą, vientisumą ir tinkamumą. Šitos ypatybės gali atrodyti konfliktuojančios, bet be visų trijų ypatybių, duomenys, kurie būtų apsaugoti, prarastų savo vertę bei naudingumą.[34]

2.8.2.1. Slaptumas

Slaptumas apsaugant duomenis reikalauja kad nebūtų laivo priėjimo prie saugomų duomenų. Tai reiškia, kad informacija turi būti slapta ir turi būti apsaugota nuo visų kas neturi priėjimo prie jos teisių. Didžiausi sunkimai ir trūkumai yra pastebimi iš interneto vartotojų perspektyvos, kada yra naudojamos teisės, ir jomis remiantis yra siunčiama informacija iš vienos vietos į kitą, ypač tai akivaizdu kai informacija yra saugoma kažkokiam kompiuteryje, ar kompiuterių grupėje, yra nėra tiksliai žinoma kokie vartotojai ją kontroliuoja. Tai reiškia jog atsižvelgiant į kažkokiais tai teises visa informacija gali būti atšaukta, kad ir įprastos kažkokios tai naudojamos sistemos, taip pat neretai yra įmanoma atšaukti teisią į duomenis, naudojant transakcijas per internetą.

Pavyzdys galėtų būti vartotojo naudojama mokamoji, kreditinė ir pan. kortelės detalūs duomenys, kurie yra paprastai saugomi tinkle (ang. online) paslaugos teikėja, po pirmo pasinaudojimo. Kol vartotojas įvykdys ar vykdys transakciją(-as) internetu, jo duomenys bus saugomi banke, ar kažkur kitus (jai kortelę aptarnauja ne bankas, o tarpininkas). Vykdamas transakcijas internetu kiekvienas paslaugos teikėjas turi priėjimą prie mokamosios ar kreditinės ir pan. kortelės duomenų (atsižvelgiant į kontrolės reikalavimus). Jeigu iškyla poreikis pašalinti kažkokias tai informacijos detales, yra susisiekiama su paslaugos teikėju, ir jis neturėdamas kitos išeities privalo jas pašalinti iš sistemos.

Pastebėtina tai, kad slaptumą gali garantuoti tik tinkamai apsaugotais informacijos taškais, kuriame apsaugojimui panaudojamos reikiamos informacijos slaptumo užtikrinimui teisės. Aišku gali būti kitų taškų kuriuose yra garantuojamas informacijos slaptumas, tačiau nėra garantuota ar bus efektyvus slaptumas. Kiekviena papildoma vieta, kuri turi teises į privačią informaciją, yra didelė rizika, kad privati informacija tyčia ar netyčia gali būti atskleista, iš esmės duomenų saugumas yra gan silpnas įvairiose sistemose, kuriose yra saugomi duomenys[34].

2.8.2.2. Vientisumas

Dar viena saugių duomenų savybę yra vientisumas. Tai reiškia kad kol galimas teisėtas priėjimas prie informacijos, ji turi būti tiksli. Dar kartą internetas pateikia didelę kliūtį šiai savybei, būtent tą kartą vartotojas nusiuntę savo informaciją kitai šaliai, ir ta kita šalis išsaugoja informaciją, informacijos vientisumas yra prarastas. Bet kokie tolimesni šitos informacijos atnaujinimai nepritaikomi kopijoms saugomoms įvairiose vietose internete, nebent vartotojas laiko kiekvienos organizacijos kuri saugo kopija įrašus, ir perspėja juos atskirai. Netgi kai vartotojas neturi metodo užtikrinančio informacijos atnaujinimo įvykdymą. Pristatymo adresas įsigyjant knygą internetu yra geras šitos problemos pavyzdys. Turėtu pirkėjas pakeisti adresą po knygų užsakymo, bet prieš pristatymą, šito adreso kopija pardavėjui nebus teisinga nebent pardavėjas yra perspėjamas apie tokius pasikeitimus. Net jeigu siuntinys dar nepristatytas, pristatymo etiketė (kita informacijos kopija) galėtu jau būti atspausdinta, rezultatas gautus neteisingas pristatymas.

Informacijos vientisumas susidaro kaip paslaptis kai tik priėjimas prie tokios informacijos yra susijęs su kita šalimi (informacijos gavėją) internete[34].

2.8.2.3. Tinkamumas

Tinkamumas yra reikalingas, kad informacija būtų pasiekama per prieigą, jei tokią prieigą suteikta. Kontekste komercijos internete tai yra dažnai ne problema, kadangi, bet kokia kompanija, suteikianti prieigą prie informacijos apskritai, turi lokaliai saugomą kopiją, ir iki tol, kol jų sistemos yra pasiekiamos, informacija siunčiama jiems, turi būti taip pat pasiekama. Tačiau tai reiškia, kad informacijos tinkamumas yra taip pat geras, kaip sistema organizacijos, kurioje tai saugoma — nėra jokio kelio informacijos savininkui, kad pagerintų tinkamumą per tradicines priemones tokias kaip lygiagretūs serveriai ar saugant tai naudojant aukšto lygio tinkamumo serverius.

Informacijos naudojamos internete tinkamumas, yra apskritai labai geras, bet kadangi nėra kontroliuojama vartotojo prieiga prie jos, todėl nėra jokio būdo vartotojui patikrinti, bet kokiame duotame taške, jei kažkokia organizacija duotų prieigą prie informacijos, tuomet gali iš tikrųjų gauti prieigą prie tos informacijos[34].

2.8.3. Saugumo ir apmokėjimo protokolai

2.8.3.1. Saugumo protokolų samprata

Protokolas - taisyklių visuma, apibrėžianti duomenų mainų būdą tarp dviejų kompiuterinių sistemų[35]. Saugumo protokolas užtikrina saugų dviejų kompiuterinių sistemų bendradarbiavimą.

Elektroninio apmokėjimo sistemų komunikavimas tarp skirtingų dalių yra labai svarbus ir turi būti tinkami apsaugotas. Klientai ir serveriai turi tinkamai nusistatyti saugius kelius internete, kad galėtų įvykdyti saugiai informacijos perdavimus. Čia aptarsime svarbiausius saugumą užtikrinančius protokolus[36].

2.8.3.2. SSL

Security Socket Layer [SSL] - tai yra kriptografinis internetinis saugumo protokolas, kuri naudojamas užtikrinti saugumą tarp komunikuojančių taškų šifruojant, t.y. tam tikras komunikuojančių sokerių mechanizmas. Šis mechanizmas buvo išvystytas Netscape

Corporation in 1994. Pagrindinė priežastis šio protokolo kūrimui buvo siekis užtikrinti saugumą internete.

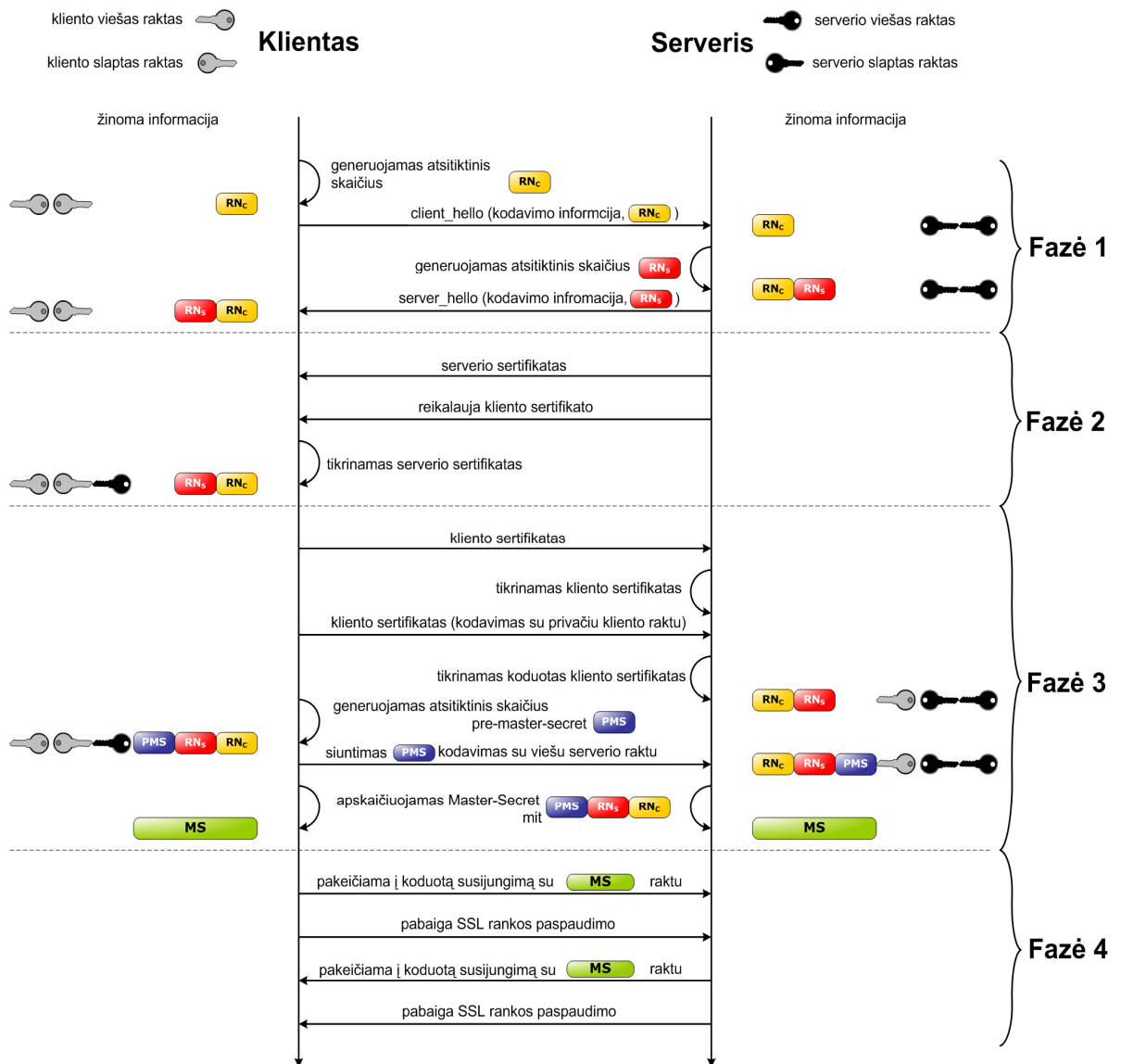
SSL protokolas veikia įdėdamas saugumą užtikrinantį protokolą tarp HTTP ir TCP kanalo galų, t.y. sukuria saugų kanalą. Elektroninio apmokėjimo technologijose autentifikacinė transakcijų dalis yra labai svarbi.

SSL protokolas leidžia klientui ir serveriui komunikuoti tarpusavyje, ir kaisti saugiai privačia informacija. SSL protokolas užtikrina autentiškumą ir privatumą naudojant kriptografiją. Paprastai serveris visą laiką būna autentifikuotas, jei neautentifikuotas klientas bando prisijungti, tai jį serveris ignoruoja, jai prisijungia autentifikuotas klientas, tai serveris su juo pradeda bendrauti. Tokiu būdu abi pusės žino tiksliai kas su kuo bendrauja. Kadangi abi pusės žino su kuo vyksta bendravimas, tad tokiu būdu ir yra užtikrinama autentifikacija. Kliento autentifikumui patvirtinti yra naudojama viešojo rakto infrastruktūra.

SSL protokolo pagrindiniai žingsniai:

1. Perdavimo algoritmo palaikymo užtikrinimas
2. Viešojo rakto kriptografija – vyksta apsikeitimas viešaisiais raktais ir sertifikatais – įvykdoma autentifikacija.
3. Simetrinė kriptografija – koduojami persiunčiami duomenys[36].

SSL protokolo funkcionavimo pagrindiniai žingsniai pateikti paveiksle (Paveikslas 2.8-1.).



Paveikslas 2.8-1. SSL rankos paspaudimas su dvipuse autentifikacija ir sertifikatais. [37]

Secure Socket Layer yra trumpiausias kelias norint užtikrinti kanalo saugumą tarp vartotojo ir internetinio portalo. SSL turi integruotą apsaugą ir gali apsaugoti, bet kokią informaciją siunčiamą kanalu.

SSL naudoja sertifikatus pateiktus internetinių raktų tiekėjų, kurie priklauso atitinkamoms organizacijoms. Viešojo rakto kodavimas yra naudojamas autentifikacijai originaliam sraute, prieš portalo sertifikavimą sertifikavimų centre, ir sesijos raktų pakeitimui. Pseudo atsitiktinių skaičių generavimas ir šifravimas turi funkcijas, kurios naudojamos sesijos raktų generavimui, ir simetrinio kodavimo kodavimas yra naudojamas daugelyje komunikacijų. Pranešimo autentifikacijos kodas yra naudojamas, siekiant aptikti padirbinėjimus, bet kokiam pranešime.

SSL yra vienas iš pagrindinių protokolų, kurį galima pritaikyti, bet kokiame internetiniame portale, be to daug įvairių sričių specialistų jį siūlo taikyti. Reiktų žinoti tai, kad SSL informaciją apsaugo tik perduodamą informaciją[Error! Bookmark not defined.]. Slapta informacija esanti internetiniame portale nėra apsaugota SSL, ir tai nėra mechanizmas galintis ją ten apsaugoti[38].

2.8.3.3. TLS

TLS - Transport Layer Security [TLS] - tai yra kriptografinis internetinis saugumo protokolas, - saugumą užtikrinantis soketas, jis yra labai panašus į SSL, Šis protokolas buvo sukurtas kaip priemonė užtikrinti saugumą tarp privačių komunikacijų. Šis protokolas leidžia kliento/serverio aplikacijos bendrauti tarpusavyje, visiškai apsaugotu keliu, nuo pasiklausymų ar įsibrovimų. Protokolo saugumą užtikrina kriptografija ir bendravimo linijos vientisumas, bei protokolo efektyvumas. Protokolas skirstomas į du lygius.

Protokolo skirstymo lygiai:

- TLS įrašų protokolas
- TLS bendradarbiavimo užmezgimo protokolas

TLS įrašų protokolas užtikrina privatų kliento ir serverio bendradarbiavimą. Saugumui užtikrinti yra naudojama privačių raktų kriptografija ir hash funkcijos

TLS bendradarbiavimo užmezgimo protokolas užtikrina komunikacijos autentifikaciją, tarp kliento ir serverio. Protokolas patogus tuo, kad serveris ir klientas susitaria, dėl šifravimo algoritmo ir šifravimo raktų, prieš duomenų siuntimą[36].

2.8.3.4. Kerberos

Kerberos yra tinklo autentifikavimo protokolas iš vystytas MIT's projekto Athena. Kerberos palaiko centralizuota autentifikacijos mechanizmą apopotam ir autentifikuotam tinklui ir nepatikimiems tinklams. Jis lengvai leidžia serveriui autentifikuoti klientui ir klientui serverį[39]. Šio protokolo svarbiausi elementai yra:

- *Saugumas* – tinklo infrastruktūra neleidžia pateikti pakankamai informacijos kompromisinėms sistemoms. Todėl pašaliečiai negali gauti reikiamos informacijos.
- *Patikimumas* – sistema yra labai patikima ir paskirstoma.

- *Permatomumas* – vartotojai turi atlikti tik vieną itin paprastą autentifikaciją. Visos kitos autentifikacijos yra atliekamos automatiškai ir nepastebimai.
- *Pajėgumas* – sistema gali aptarnauti didelį kiekį vartotojų ir serverių, bendrai t.y. distribucija, ir moduliacija.

Kerberos pagrinde dirba su centrine vartotojų autentifikacija. Sistema duoda vartotojui autentifikacijos bilietuką, kuriuo naudodamasis jis gali gauti kitus individualius bilietukus. Pirmiausias vartotojas autentifikuojasi autentifikacijos serveryje Ticket Granting Ticket [TGT]. TGT yra užšifruojami ir juo iššifruoti gali tik vartotojas (gavėjas). Pas vartotoja yra Ticket Granting Service [TGS]. Siunčiama informacija yra patvirtinama slaptažodžiu, ir tinkle apsaugant nuo pažeidimų, yra apauginama saugumo lygiais. Kiekvienai paslaugai, kurią vartoja vartotojas, gauna atitinkamą bilietuką iš TGS. Taigi vartotojas gali naudoti šiuos autentifikacinius bilietukus, ir bet kada pasitikrinti ar naudojamas servisas yra tikras[40].

2.8.4. Prieigos kontrolė

2.8.4.1. Prieigos kontrolės prasmė

Prieigos kontrolė tradiciškai apėmė privalomą prieigos kontrolę (arba tinkleliu pagrįstą prieigos kontrolę) ir savarankišką prieigos kontrolę. Vėliau, vaidmenimis pagrįsta prieigos kontrolė buvo įvesta, greta reikalavimų, nes jos mechanizmai yra gana bendri, kad imituotų tradicinius metodus. Toliau apžvelgiami įvairios sisteminės konstrukcijas bendroje tradicinių prieigos kontrolės paradigmu formoje, naudojant vaidmenimis pagrįstą prieigos kontrolę (ang. role-based access control) (RBAC) Sandhu ir kt. modelius, paprastai vadinamus RBAC96. Matome, kad visos RBAC96 modelio ypatybės yra reikalingos privalomam prieigos kontrolės imitavimui, tiktai vienas administracinis vaidmuo turi būti priimtas, savarankiškam prieigos kontrolės imitavimui, sudėtingas administracinių vaidmenų kompleksas yra taip pat reikalingas[41]. Prieigos kontrolė yra reikalinga siekiant apsaugoti duomenis ar informaciją, nuo nesankcionuoto pasinaudojimo.

2.8.4.2. Prieigos kontrolių apžvalga

Vaidmenimis pagrįsta prieigos kontrolė (ang. role-based access control) (RBAC) yra sulaukusi didelio dėmesio kaip daug žadanti alternatyvą tradicinėms, savarankiškomis ir

privalomoms prieigos kontrolėms. RBAC teisės yra sujungtos su vaidmenimis, ir vartotojus padaryti nariais atitinkamų vaidmenų, tokiu būdu išigyjančių vaidmenų teises. Tai labai supaprastina teisių valdymą. Vaidmenys gali būti sukurti įvairioms darbo funkcijoms organizacijoje, ir vartotojams priskiriami vaidmenys, pagrįsti jų pareigomis ir kvalifikacijomis. Vartotojams lengvai gali pereiti nuo vieno vaidmens prie kito. Vaidmenims gali būti suteiktos naujos teisės, kuomet naujos taikomosios programos, ir sistemos yra įtrauktos, ir teisės gali būti panaikinti nuo vaidmenų kai yra būtina.

Svarbi RBAC savybė yra, kad savyje turi neutralius poliusus. RBAC yra priemonė tam, kad aiškus poliusus pritaikytų greičiau, negu įgyvendintų tam tikrą saugumo politiką (tokią kaip viena kryptinio informacijos srauto tinklelis). Poliai primetami tam tikroje sistemoje, yra grynas rezultatas tikslios konfigūracijos ir sąveika įvairių RBAC komponentų tiesiogiai nuo sistemos savininko. Be to, prieigos kontrolės poliai gali augančiai plėtotis per sistemos gyvavimo ciklą, ir didelėse sistemose taip iš tikrųjų yra. Gebėjimas modifikuoti polisą yra reikalingas, kad būtų galima patenkinti besikeičiančius organizacijos poreikius, tai yra svarbi RBAC galimybė.

Tradiciniai prieigos kontrolės modeliai apima privalomą prieigos kontrolę (ang. mandatory access control) (MAC), kuri mes vadinsime tinkleliais pagrįsta prieigos kontrole (ang. lattice-based access control) (LBAC) [Denning; Sandhu], ir savarankiška prieigos kontrolė (ang. discretionary access control) (DAC) [Lampson; Sandhu ir Samarati]. Nuo RBAC pradžios keli autoriai yra aptarę santykį tarp RBAC ir šitų tradicinių modelių [Sandhu; Sandhu ir Munawer; Munawer; Nyanhama ir Osborn]. Tvirtinimas, kad RBAC yra bendresnis negu visi šitie tradiciniai modeliai būdavo dažnai daromas. Šio aprašo tikslas yra parodyti, kaip RBAC gali būti sukonfigūruotas, kaip apima šituos tradicinius modelius.

Klasikiniai LBAC modeliai yra specialiai konstruoti taip, kad įtrauktų polisą viena kryptinio informacijos srauto tinklelyje. Šis viena kryptinis informacijos srautas gali būti taikomas dėl konfidencialumo, vientisumo, konfidencialumo ir vientisumo kartu, ar sukaupimo polisams tokiems kaip Kinų Siena [Sandhu]. Yra vis dėlto didelis panašumo tarp saugumo žymės ir vaidmens sąvokų. Ypač, kai dalyvauja tas pats vartotojas, pavyzdžiui, Paslapčiai apsaugoti, gali būti naudojamos skirtingo registracijos į sistemą Slaptuose ir Neįslaptintuose lygmenyse. Tam tikra prasme vartotojas nustato, koks vaidmuo (Slaptas ar Neįslaptintas) turi būti aktyvuotas tam tikroje sesijoje.

Tačiau kyla klausimas ar tikrai LBAC gali būti imituotas, naudojant RBAC. Jei RBAC yra neutralus poliusas ir turi tinkamą bendrumą, tai turi iš tikrųjų sugebėti padaryti taip, ypač kadangi supratimas vaidmens ir registracijos vardo sesijos lygmenyje yra taip panašūs. Šis klausimas yra teoriškai reikšmingas todėl, kad teigiamas atsakymas nustatytų, kad LBAC yra

tik vienas atvejis RBAC, tuo būdu siedamas du skirtingus prieigos kontrolės modelius, kurie buvo išvystyti su skirtinga motyvacija. Teigiamas atsakymas yra taip pat beveik prasmingas, todėl, kad jis duoda suprasti, kad tas pats Patikimo Skaičiavimo Pagrindas gali būti formuotas taip, kad primestų RBAC apskritai ir LBAC ypač. Tuo atkreipiamas didelis dėmesys į daugia lygmeninio saugumo gynėjų ta technologiją, kuri tenkina didesnės komercinės rinkos poreikius būti pritaikomai LBAC. Klasikinis metodas šio noro įvykdymo turėtų teigti, kad LBAC turi tam tikras panaudojimo galimybes komerciniame sektoriuje. Tačiau lygi šiol šis argumentas nebuvo baisiai produktyvus. Antra vertus RBAC yra specialiai motyvuotas komercinio sektoriaus poreikiams. Jo pritaikymas į LBAC galėtų būti labiau produktyvus, panaudojant dvigubo naudojimo technologijų metodu.

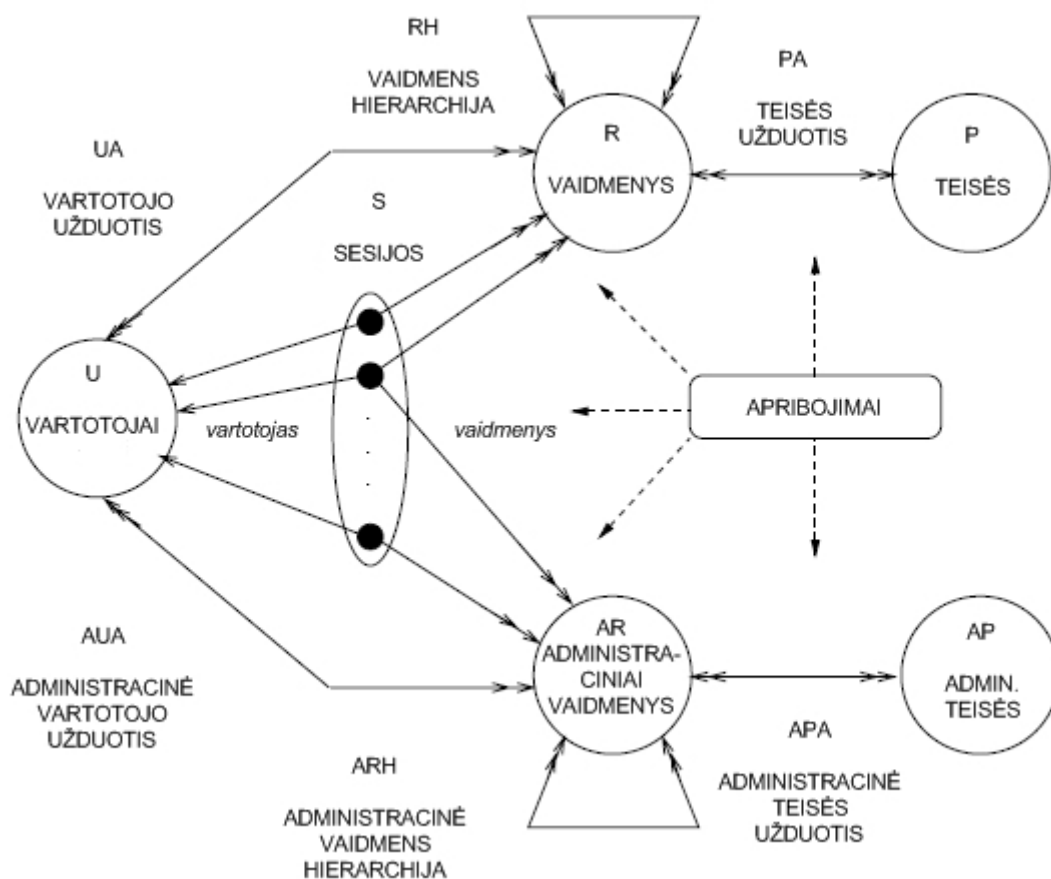
Toliau bus nagrinėjama ir įrodinėjama, kad keli LBAC variantai gali būti lengvai pritaikyti RBAC, formuojant kelis RBAC komponentus. (Turi būti pažymėtina, kad RBAC tikrai trukdys atviriems informacijos srautams. Tai yra aišku, bet kokiam prieigos kontrolės modeliui, įskaitant ir LBAC. Informacijos srautas prieš vien kryptinį reikalavimą tinklelyje su vadinamųjų slaptų kanalų pagalba esančių už prieigos kontrolės srities. Nei LBAC, nei RBAC neatkreipia dėmesio į slapto kanalo problemą tiesiogiai. Technika, panaudota, kad aptarnautų slaptus kanalus LBAC, taip pat ji gali būti panaudota tuo pačiu tikslu RBAC.) Panaudoti konstravimai rodo, kad vaidmens hierarchijų ir apribojimų koncepcijos yra kritiškos gaunamiems rezultatams.

Vaidmens hierarchijos ir apribojimų pakeitimai priveda prie skirtingų LBAC variantų.

Savarankiška prieigos kontrolė (ang. discretionary access control) (DAC) buvo panaudota ankstyvosiose komercinėse programose, ypač operacinėse sistemose ir reliacinėse duomenų bazių sistemose. Pagrindinė idėja DAC yra tokia, kad objekto savininkas, kuris yra paprastai jo kūrėjas, turi savarankišką valdžią, lyginant su tuo kas dar gali gauti prieigą prie to objekto. Kitaip tariant DAC apima savininkais pagrįstą prieigos teisių administravimą. Atsižvelgiant į LBAC, nereikia nagrinėti sudėtingo prieigos teisių administravimo, nes DAC, administraciniai vaidmenys vytyti Sandhu ir kt. yra svarbesni. Kadangi kiekvienas objektas galėjo potencialiai būti valdomas unikalios savininko, administracinių vaidmenų skaičius gali būti ganėtinai didelis. Tačiau, toliau bus parodyta, kad vaidmenų administravimo patogumai RBAC96 modelyje yra tinkami kuriant sudėtingas administravimo sistemas[41].

2.8.4.3. RBAC modelis

Bendras RBAC modelis apima administratoriaus vaidmenis, kurie buvo aprašyti Sandhu (1996 m.). Jų santrauka pateikta paveiksle (Paveikslas 2.8-2.). Modelio pagrindas yra trys aibės: vartotojų (U), vaidmenų (R) ir teisių (P). Intuityviai vartotojas yra žmogus, arba automatinis agentas, vaidmuo yra darbinė funkcija arba darbo pavadinimas organizacijoje su kažkokia asocijuojama semantika susijusia su valdžia ir atsakomybės suteiktos nariui vaidmens, ir teisės suteikiančios tam tikru būdu prieigą prie vieno ar daugiau objektų sistemoje.



Paveikslas 2.8-2. RBAC96 modelis.

Vartotojo užduotis (UA) ir teisės užduotis (PA) (pavaizduoti paveiksle Paveikslas 2.8-2.) yra susietos „daug su daug“ tipo ryšiais (pažymėti rodyklėmis su dvigubomis strėlėmis). Vartotojas gali būti nariu daugelio vaidmenų, ir vaidmuo gali turėti daug vartotojų. Taip pat, taip pat vaidmuo gali turėti daug teisių, ir ta pati teisė gali būti priskirta daugeliui vaidmenų. Yra dalinė vaidmenų hierarchija RH , tai galima užrašyti taip \geq , kur $x \geq y$, reiškiančio kad

vaidmuo x paveldi teises priskirta vaidmeniui y . vaidmenų hierarchija gali būt pristatoma kaip asiklinis tiesioginis grafas, ir tiesioginiai ryšiai vaidmenų hierarchijoje yra nuorodinės *kraštinės*. Paveldėjimas pagal vaidmenų hierarchiją yra tranzityvinis, daugialypis paveldėjimas yra leistinas daliniuose nurodymuose.

Paveikslas 2.8-2. parodo aibę sesijų S . kiekviena sesija sieja vartotoją su galbūt daugeliu vaidmenų. Intuityviai vartotojas nustato sesiją, per kurią vartotojas aktyvuoja kažkokį tai vaidmenų poaibį, priklausomai nuo to koks jis ar ji yra narys (tiesioginis ar netiesioginis atsižvelgiant į vaidmenų hierarchiją). Dvigubo rodyklių strėlės nuo sesijos iki R rodo, kad daugialypiai vaidmenys gali būti tuo pačiu laiku aktyvuoti. Taisės pasiekiamos vartotojui yra sąjunga yra sąjunga teisių nuo visų vaidmenų, aktyvuotų toje sesijoje. Kiekviena sesija yra asocijuojama su vienu vartotoju, pažymėta vienos strėlės rodykle nuo sesijos iki U . Ši asociacija lieka pastovi per visą sesijos gyvavimą. Vartotojas gali turėti daugialypes sesijas atidarytas tuo pačiu laiku, kiekvieną skirtingam lange, kompiuterizuotoje darbo vietos ekrane, pavyzdžiui. Kiekviena sesija gali turėti skirtingą aktyvių vaidmenų kombinaciją. Sesijos sąvoka nustato lygybę su tradiciniu supratimu *subjekto* prieigos kontrolėje. Subjektas (ar sesija) yra prieigos kontrolės vienetas, ir vartotojas gali turėti daugialypius subjektus (ar sesijas) su skirtingomis teisėmis, aktyviomis tuo pačiu metu.

Paveikslas 2.8-2. apatinė dalis rodo administracinius vaidmenis ir teise. RBAC96 atskiria vaidmenis ir teises nuo administracinių vaidmenų ir teisių atitinkamai, kur pastarieji yra panaudojami, kad valdytų ankstesnį. Administracinių vaidmenų ir teisių administravimas yra kontroliuojama svarbiausio saugumo pareigūno ar iš dalies yra deleguojamas iš dalies į administracinius vaidmenis. Administraciniai aspektai RBAC96 detaliau yra apibūdintas nagrinėjant DAC. LBAC aprašyme yra kalbama apie tai, kad vienas saugumo pareigūnas yra vienintelis, kuris gali formuoti įvairius RBAC96 komponentus.

Galiausiai, Paveikslas 2.8-2. rodo *apribojimų* kolekciją. Apribojimai gali prisitaikyti prie, bet kurių iš ankstesnių komponentų. Apribojimo pavyzdys yra bendrai susieti vaidmenys, tokie kaip supirkimų skyriaus vadovas ir mokėjimų vadybininkas, kur tam pačia vartotojui neleidžia būti abiejų vaidmenų nariu[41].

Sekantys apibrėžimai formalizuoja anksčiau paminėtus faktus.

1 apibrėžimas. RBAC96 modelis turi sekančius komponentus:

- U , vartotojų aibė
 R ir AR , išskiria vaidmenų (reguliarių) ir administracinių vaidmenų aibę
 P ir AP , išskiria teisių (reguliarių) ir administracinių teisių aibę
 S , sesijų aibė
- $PA \subseteq P \times R$, taisės daug su daug ryšiu susieta su vaidmens užduotimis

$APA \subseteq AP \times AR$, taisybės daug su daug ryšiu susieta su administracinio vaidmens užduotimis

- $UA \subseteq U \times R$, vartotojas daug su daug ryšiu susieta su vaidmens užduotimis
 $AUA \subseteq U \times AR$, vartotojas daug su daug ryšiu susieta su administracinio vaidmens užduotimis
- $RH \subseteq R \times R$, iš dalies sutvarkyta vaidmenų hierarchija
 $ARH \subseteq AR \times AR$, iš dalies sutvarkyta administracinių vaidmenų hierarchija (abi hierarchijos yra parašytos, kaip \geq intarpo ženklas)
- $var\ totojas : S \rightarrow U$, funkcija pažyminti kiekvieną sesiją S_i žemėlapyje vienam vartotojui $var\ totojas(S_i)$ (konstanta visai sesijos gyvavimo trukmei),
 $vaidmenys : S \rightarrow 2^{R \cup AR}$, pažyminti kiekvieną sesiją S_i žemėlapyje ir vaidmenų ir administracinių vaidmenų aibę
 $vaidmenys(S_i) \subseteq \{r \mid (\exists r' \geq r)[(var\ totojas(S_i), r') \in UA \cup AUA]\}$ (kurios gali pasikeisti laike)
sesija S_i turi teises $\bigcup_{r \in vaidmenys(S_i)} \{p \mid (\exists r'' \leq r)[(p, r'') \in PA \cup APA]\}$
- yra kolekcija apribojimų, kurios, įvairių komponentų vertės, išvardintos anksčiau, yra leistinos ar uždraustos[41].

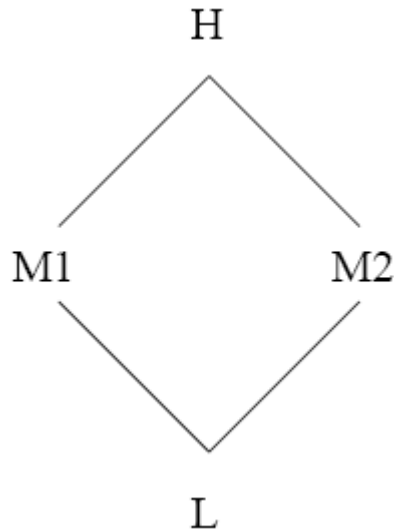
2.8.4.4. LDAC (arba MAC) modeliai

Tinkleliu pagrįsta priėjimo kontrolė susijusi su vieno kryptinės informacijos srauto spaudimu ir tinklelio saugumo žymeklius. Paprastai tai taikoma kaip priedas prie klasikinių savarankiškų priėjimo kontrolių, tačiau toliau aptarsime tik MAC komponentą. Priklausomai nuo tinklelio prigimties vienkryptis informacijos srautas sustiprintas LBAC gali būti pritaikytas slaptumui, integralumui, kartu slaptumui ir integralumui[42]. Tai pat yra LBAC pakitimų, kur vienkryptės informacijos srautas yra dalinai laisvas kad pasiekti selektyvu informacijos kritimą arba taikomųjų programų integralumui[43].

Privaloma priėjimo kontrolės veikla išreikšta saugumo žymeklių ribose prijungtuose prie veiksnio arba objekto. Žymeklis objekte yra vadinamas *saugumo klasifikavimas*, kol žymeklis pas vartotoją - *saugumo leidimas*. Yra svarbu suprasti kad *slaptas* vartotojas gali paleisti tokią pačią programą, kaip tekstinis redaktorius, kaip *slaptas* veiksnys arba kaip neklasifikuotas veiksnys. Nors abu veiksniai paleidžia tokią pačią programą to paties

vartotojo naudai, jie išgauna skirtingas privilegijas priklausančias prie jų saugumo žymeklių. Žinoma kad saugumo žymekliai veiksmuose arba objektuose, vieną kartą nustatyti, negali būti pakeisti. Šita prielaida, kad saugumo žymekliai negali keistis žinoma kaip *ramybė*. Tinklelio struktūros saugumo žymeklių forma pateikta žemiau:

Apibrėžimas 2. (Saugumo tinklelis) Yra saugumo žymeklių baigtinis tinklelis \mathcal{S} su dalinai sutvarkytu dominavimo ryšiu \geq ir mažiausiu viršutinės ribos operatoriumi.



Paveikslas 2.8-3. Dalinai sutvarkytas tinklelis

Saugumo tinklelio pavyzdys parodytas paveiksle Paveikslas 2.8-3. Informacijai tinklelyje yra leista tekėti tik į viršų. Šitame pavyzdyje, H ir L atitinkamai pažymi aukštą ir žemą, ir $M1$ ir $M2$ yra du nesuderinami žymekliai pereinamieji H ir L . Tai yra tipiškas konfidencialumo tinklelis kur informacija gali tekėti iš žemo į aukštą bet ne atvirkščiai.

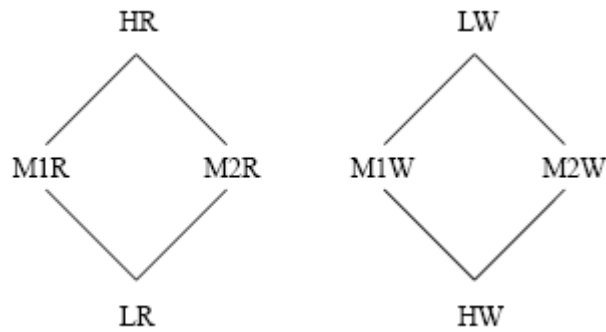
Specifinės privalomos priėjimo taisyklės paprastai specifikuotos tinkleliui šitaip; kur λ pažymi saugumo žymeklį nurodyto veiksnio arba objekto[41].

Apibrėžimas 3. (Paprasta saugumo ypatybė) veiksnys s gali rašyti objektą o tik tada jeigu $\lambda(s) \geq \lambda(o)$.

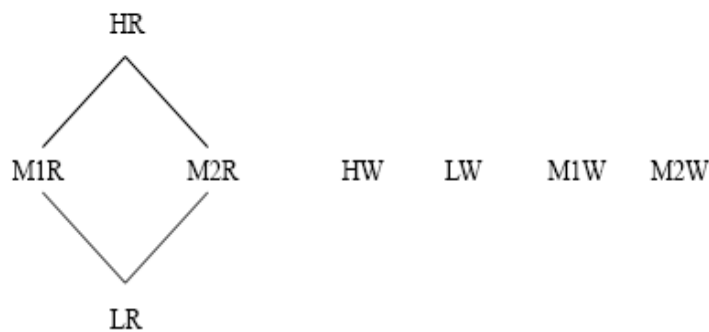
*Apibrėžimas 4. (Negriežta *- ypatybė)* veiksnys s gali rašyti objektą o tik tada jeigu $\lambda(s) \leq \lambda(o)$.

*- ypatybė kaip ne atsiliepiamas arba rašyti-lygu.

LBAC variantai, paprasto-saugumo ypatybė paprastai paliekama nepakeista kaip daroma visuose pavyzdžiuose. Variantai *- ypatybės LBAC pagal kuri vienakryptis informacijos srautas dalinai yra laisvas kad pasiektu pasirenkamą informacijos kritimą arba taikomųjų programų integralumui bus apgalvota vėliau[44].



(a) negriežta *- ypatybė



(b) griežta *- ypatybė

Paveikslas 2.8-4. Hierarchijos vaidmuo tinkleliui

2.8.4.5. RBAC dėl LBAC konfigūravimas

Toliau pateikiama, kaip skirtingi LBAC pakeitimai gali būti imituoti RBAC96. Pasirodo, kad galima pasiekti metodiškai keičiant vaidmens hierarchiją ir apibrėžiant tinkamus apribojimus. Tuo manoma, kad vaidmens hierarchijos ir apribojimai gali būti naudojami centrinių polisų RBAC96 apibūdinimui[41].

2.8.4.5.1. Pagrindinis tinklelis

Pirmiausias apžvelgiamas pavyzdžio tinklelis paveiksle Paveikslas 2.8-3. su liberaliom *-Ypatybėm. Subjektai su žymekliais esantys viršuje tinklelio turi daugiau valdžios atžvilgiu skaitymo operacijų, bet turi mažiau valdžios atžvilgiu rašymo operacijų. Tokiu būdu, šis tinklelis turi dvejoją charakterį. Vaidmens hierarchijos subjektai (sesijos) su aukštesniais vaidmenimis hierarchijoje, visada turi daugiau galių, negu tie kurie turi žemesnius vaidmenimis hierarchijoje. Kad pritaikytume dvejoją charakterį tinkleliu LBAC, panaudojamos dvi dvejopas hierarchijas RBAC96, viena skaitymui ir vienas rašymui. Šios dvi

vaidmens hierarchijas tinkleliui pateiktam paveiksle Paveikslas 2.8-3. parodytos paveiksle Paveikslas 2.8-4.(a). Kiekvienas tinklelio žymeklis x yra modeliuojamas, kaip du vaidmenys xR ir xW skaitymui ir rašymui žymeklyje x atitinkamai. Sąryšis tarp keturių skaitymo vaidmenų ir keturių rašymo, parodytas kairėje ir dešinėje pusėse paveiksle Paveikslas 2.8-4.(a), atitinkamai. Dvilypumas tarp kairiųjų ir dešiniųjų grotelių yra akivaizdus iš diagramų.

Kad užbaigti konstravimą, reikia primesti tinkamus apribojimus, kad atspindėtų žymeklyje apie subjektą LBAC'e. Kiekvienas vartotojas LBAC turi unikalų saugumo leidimą. Taip primetamas reikalavimas, kad kiekvieną vartotoją RBAC96 būtų paskirtas tiksliai dviem vaidmenims xR ir LW . LBAC vartotojas gali registruotis, bet kokiam žymeklyje, kurį įtakoje vartotojo leidimas. Šis reikalavimas yra sukauptas RBAC96 reikalavimo, kad kiekviena sesija turėtų tiksliai du derančius yR ir yW . Sąlyga, kad $x \geq y$, kuris reiškia, kad vartotojo leidimas valdo žymeklį, bet kokios registracijos sesijos, nustatytam vartotojui, nėra aiškiai tiesiogiai todėl, kad tai tiesiogiai paskiriama RBAC96 konstrukcijos. Pažymėkite, kad laisvi narystės ryšiai LW , kiekvienas vartotojas gali aktyvuoti, bet kokį rašymo vaidmenį. Tačiau, rašymo vaidmens aktyvavimas tam tikroje sesijoje, turi atitikti sesijos perskaitytą vaidmenį. Tokiu būdu, ir vaidmens hierarchija, ir RBAC96 apribojimai yra eksploatuoti šioje konstrukcijoje.

LBAC yra primestos laike skaitymo ir rašymo operacijos. RBAC96 tai reiškia, kad mūsų leidimai skaitymui, ir rašymui ant individualių objektų, užrašomas atitinkamai (o, r) ir (o, w) . LBAC objektas turi vieną jautrų žymeklį, asocijuotą su juo. Tai išreiškiama RBAC96 reikalavimu, kad kiekviena teisių pora (o, r) ir (o, w) būti paskiriama tiksliai vienai xR ir xW atitinkamai vaidmenų porai. Skiriant leidimus (o, r) ir (o, w) į vaidmenis xR ir xW , atitinkamai, netiesiogiai nustatomi jautrumas žymeklio objekto o į x [41].

2.8.4.5.2. Pagrindinė konstrukcija

Duotoji \mathcal{S} su saugumo žymekliais $\{L_1 \dots L_n\}$, ir dalinis nustatymas \geq_{LBAC} , sistemos RBAC96 ekvivalentas yra pateiktas:

*Konstrukcija 1. (Laisva *- ypatybė)*

- $R = \{L_1R \dots L_nR, L_1W \dots L_nW\}$
- RH kuris susideda iš dviejų išskirstytų vaidmenų hierarchijų. Pirmo vaidmens hierarchija susideda iš „skaityti“ vaidmenų $\{L_1R \dots L_nR\}$ ir turi tokį patį dalinį nustatymą kaip \geq_{LBAC} ; antras - susideda iš „rašyti“ vaidmens $\{L_1W \dots L_nW\}$ ir turi dalinį nustatymą kuris yra priešingas \geq_{LBAC} .

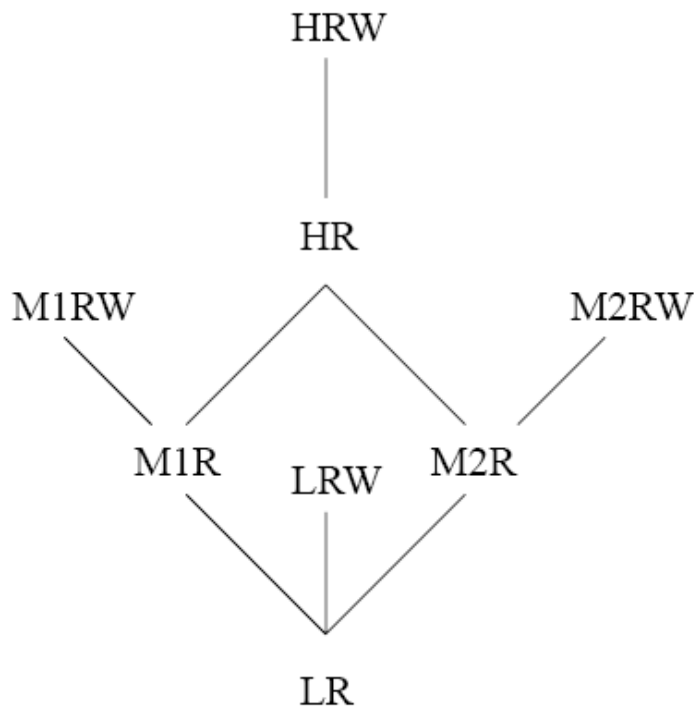
- $P = \{(o, r), (o, w) \mid o \text{ yra sistemos objektas}\}$
- *UA apribojimas*: Kiekvienas vartotojas priskirtas tiksliai dviem vaidmenims xR ir LW , kur x yra žymeklis priskirtas vartotojui LW yra rašymo vaidmuo priklausantis žemiausiam saugumo lygiui pagal \geq_{LBAC}
- *Apribojimas sesijoms*: Kiekviena turi tiksliai du vaidmenys yR ir yW
- *PA apribojimas*:
 - (o, r) yra paskirtas xR jeigu (o, w) paskirtas xW
 - (o, r) yra paskirtas tiksliai vienam vaidmeniui xR tokiam kad x yra o žymeklis

Teorema 1. *RBAC96 sistema apibrėžta Konstrukcijoje 1 patenkina Paprasto Saugumo Ypatybę ir Laisvą *- Ypatybę*

Irodymas. (a) Paprasto Saugumo ypatybę: Veiksniai LBAC terminologijoje atitinka RBAC96 sesijas. Kad veiksnys s skaitytų o (o, r) privalo būti leidimuose priskirtuose vaidmeniui, tiesiogiai arba netiesiogiai, kuris yra tarp vaidmenų įmanomų sesijai s , kuris atitinka tiksliai vienam vartotojui u . Kad u būtų įtrauktas į šitą sesiją, vaidmuo turi būti UA būsenoje. Tarkime $\lambda(u) = z$ ir $\lambda(s) = y$. Remiantis PA apribojimais, pateiktais *Konstrukcijoje 1*, (o, r) yra priskirtas tiesiogiai vienam vaidmeniui xR , kur $x = \lambda(o)$, ir remiantis RH konstrukcija, yra paveldėta yR vaidmenų tokių kaip $y \geq_{LBAC} x$. Kad s galėtų skaityti o , turi būti vienas iš šitų yR toje sesijoje. Pagal vaidmenų apibrėžimą iš *Apibrėžimo 1*, bet kuris vaidmuo jaunesnis už zR gali būti sesijoje dėl u . Kitaip sakant, sesija dėl u gali įtraukti vienas skaitymo vaidmuo yR toks kaip $z \geq_{LBAC} y$. Taigi, apibrėžta RBAC96 sistema leidžia veiksmui s skaityti objektą o jeigu $\lambda(u) \geq_{LBAC} \lambda(s)$ ir $\lambda(s) \geq_{LBAC} \lambda(o)$, kurie yra būtent Paprasto Saugumo Ypatybė.

(b) Laisva *- Ypatybė: Kiekvienas vartotojas, u priskirtas UA į xR , kur x yra vartotojo leidimas. Remiantis LBAC, vartotojas gali skaityti duomenys klasifikuotus x lygyje arba lygiuose su dominuojančių x . Tai taip pat reiškia kad vartotojas gali pradėti sesiją lygyje kuriame dominuoja x . Taigi, jeigu vartotojas sako lygyje x , norima paleisti sesiją lygyje y , tokią kaip $x \geq_{LBAC} y$, apribojimai *Konstrukcijoje 1* leidžia sesijai turėti du aktyvius vaidmenys yR ir yW . Kadangi kiekvienas vartotojas priskirtas LW , yra įmanoma kiekvienam vartotojui turėti sesiją su yW kaip vienam iš tų vaidmenų. Dviejų vaidmenų hierarchijų struktūra reiškia kad jeigu vaidmuo yW yra įmanoma vartotojui sesijoje, vartotojas gali rašyti objektus kuriems leidimas (o, w) yra yW . Pagal vaidmenų hierarchijų konstrukcija, sesija gali rašyti į lygi y . LBAC ribose, veiksnys, s atitinka sesijai, ir sesijos viduje rašymas gali būti atliekamas

jeigu (o, w) yra leista vaidmens, kuris pagal konstrukciją yra tada kai $\lambda(o) \geq_{LBAC} \lambda(s)$. Tai yra Laisva *- Ypatybė[41].



Paveikslas 2.8-5. Kintama vaidmens hierarchija griežtai *- ypatybei.

2.8.4.5.3. LBAC variantai

LBAC variantai gali būti pritaikyti, modifikuojant įvairiai pagrindinę konstrukciją. Ypač, griežtos *-Ypatybės išsaugo hierarchiją skaitymo vaidmenyse, bet traktuojami rašymo vaidmenys nepalyginamai su vienas, kitu kaip parodytas paveiksle (Paveikslas 2.8-4.(b)) pagrindinių tinklelių pavyzdžiui.

*Konstrukcija 2. (Griežtos *-Įpatybės),* Identiškas Konstrukcijai 1 išskyrus *RH* turi dalinį nurodymą tarp skaitymo vaidmenų, identiškų LBAC daliniam nurodymui, ir neturi jokiems santykiams tarp rašymo vaidmenų.

*TEOREMA 2. RBAC96 sistemą apibrėžia Konstrukcija 2, atitinkanti Paprastas Saugumo Ypatybes ir Griežtas *-Ypatybes.*

Teoremos įrodymas, ir iš to išeinantys rezultatai praleidžiami.

Toliau nagrinėjama versiją LBAC, kurios subjektui duoda daugiau valdžios negu leidžia paprastas saugumas ir *-ypatybės [Bell]. Pagrindinė idėja yra leidžia subjektams pažeisti *-ypatybes kontroliuojamu būdu. Tai pasiekama, sujungiant saugumo žymių poras λ_r porą, ir λ_w

su kiekvienu subjektu (objektai vis dar turi vieną saugumo žymę). Paprastos saugumo ypatybės yra pritaikytas atžvilgiu λ_r ir liberalių *-ypatybių atžvilgiu λ_w . Bell LBAC modelyje, reikalaujama, kad λ_r turi dominuoti λ_w atžvilgiu. Su šiuo apribojimu, subjektas gali skaityti ir parašyti diapazone žymių tarp λ_r ir λ_w , kuris vadinamas *patikimumo diapazonu*. Jei λ_r ir λ_w yra lygūs, modelis mažinamas iki paprasto LBAC modelio su patikimumo diapazonu turinčiu vieną žymeklį.

Ankstesniuose skyriuose yra apibrėžta glaustai RBAC konstrukcijos. Du žymekliai λ_r ir λ_w susirašinėja tiesiogiai su dviem vaidmenimis xR ir yW , kurių mes įvedėme anksčiau. Dominavimas reikalingas tarp λ_r ir λ_w , yra nereikšmingai perdirbtas kaip dominavimo apribojimas tarp x ir y . Tai priveda prie sekančių konstravimų:

*Konstrukcija 3. (Liberalios *-Ypatybės su Patikimumo Diapazonu)* Identiška Konstrukcijai 1 išskyrus:

- UA apribojimas: Kiekvienas vartotojas paskiriamas tiksliai dviem vaidmenims xR ir yW , taip kad $x \geq y$ originaliame tinklelyje.
- Sesijų apribojimas: Kiekviena sesija turi tiksliai du vaidmenis xR ir yW , taip kad $x \geq y$ originaliame tinklelyje.

LEE ir Schockley teigė, kad Clark-Wilson vientisumo modelis [Clark ir Wilson] gali būti palaikomas, naudodamas LBAC. Jų modeliai yra panašūs į anksčiau minėtą išskyrus tai, kad joks dominavimo ryšys nėra reikalingas tarp x ir y . Tokiu būdu, rašymo diapazonas gali būti visiškai nesusietas su subjekto skaitymo. Tai lengvai išreiškiama RBAC96, kaip pateikta toliau.

*Konstrukcija 4. (Liberalios *-Ypatybės su Nepriklausomu Rašymo, Diapazonu)* Identiškas Konstrukcijai 3 išskyrus $x \geq y$, kuris nėra reikalingas apribojime UA ir sesijų apribojime.

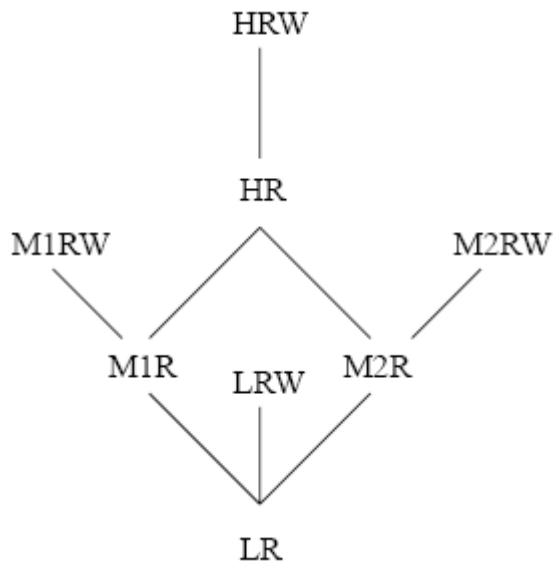
Anksčiau minėta variantas turi panaudoti griežtas *-ypatybes, kaip kad pateikta sekančiai.

*Konstrukcija 5. (Griežtos *-Ypatybės su Pažymėtu Rašymu)* Identiškas Konstrukcijai 2 išskyrus:

- UA apribojimas: Kiekvienas vartotojas paskiriamas tiksliai dviem vaidmenims xR ir yW .
- Sesijų apribojimas: Kiekviena sesija turi tiksliai du vaidmenis xR ir yW .

Konstrukcija 5 gali būti, taip pat tiesiogiai gauta iš Konstrukcijos 4, reikalaujant griežtų *-ypatybių vietoj liberalių *-ypatybių. Konstrukcija 5 gali pritaikyti Clark-Wilson transformacijos procedūras, kaip apibrėžia bendrais bruožais Lee ir Schockley. (Lee ir

Schockley iš tikrųjų naudoja liberalias *-ypatybes savo konstravimuose, bet jų tinkleliai yra tokie, kad konstrukcijos yra tiesiogiai išreikštos išreiškiant griežtas *-ypatybes.)[41]



Paveikslas 2.8-6. Kintama vaidmens hierarchija Griežtom *-ypatybėm.

2.8.4.6. Galimų RBAC konfigūracijų išplėtimas

Šiame skyriuje mes patikrinsime pasirenkamas RBAC sistemas, kurioms nereikia laikytis *konstrukcijų* ir vistiek patenkina LBAC ypatybes. Tam kad padaryti tai, tarkime kad visi vartotojai ir objektai turi saugumo žymeklius, ir įtraukti leidimai yra skaitymas ir rašymas.

Prieš tai kalbėta buvo apie visų konstrukcijų sukurtas vaidmenų hierarchijas su išskiriamais skaitymo ir rašymo vaidmenimis. Tai nėra griežtai privaloma; vaidmens hierarchija paveiksle (Paveikslas 2.8-6.). Galėtų būti konstrukcija griežtai *-ypatybei su sekančiais pakeitimais:

- UA apribojimas: Kiekvienas vartotojas yra priskirtas visiems vaidmenims, xRW toks kad vartotojo dominavimo leidimas yra saugumo žymeklis x .
- Apribojimas sesijoms: Kiekviena sesija turi tiksliai vienas vaidmuo: yRW
- PA apribojimas:
 - (o, r) yra paskirtas xR jeigu (o, w) paskirtas xRW
 - (o, r) yra paskirtas tiksliai vienam vaidmeniui xR

Vis dėlto, vaidmenų hierarchijų struktūra, kuri suteikia planą veiksmingoms LBAC konfigūracijoms yra stipriai apribotas, kaip pavyzdžiui, vaidmuo su leidimais abiems skaitymui ir rašymui aukšto duomenų objekto ir žemo objekto negali būti priskirtas aukštam vartotojui kad tai leistu rašyti. Taip pat negali priskirta žemam vartotojui kad tai leistu jam skaityti. Jeigu vaidmuo turi tik skaitymo leidimą kai kuriems objektams klasifikuotiems M1, kitiems objektams klasifikuotiems M2, veiksnys esantis H galētu būti priskirtas šitam vaidmeniui.

Kol skaitymo operacija pasiekia, veiksnys gali turėti vaidmenį r jo sesijose jeigu veiksnio žymeklis dominuoja visuose o lygiuose, taip kad (o, r) yra šitame vaidmenyje. Nuo tada kai viršutine riba yra nustatyta saugumo tinkleliui, šis gali būti visą laiką nustatytas. Panašiai, rašymo operacijoms, jeigu didžiausia iš žemiausių ribų yra nustatyta saugumo lygiams, tada laisva *- ypatybė yra patenkinama sesijoje, jeigu veiksnio saugumo lygis dominuoja didžiausioje iš žemiausių ribų su visomis o tokioms kad (o, w) yra šitame vaidmenyje. Jeigu tokia didžiausia iš žemiausių ribų neegzistuoja, toks vaidmuo neturētu būti nei viename vartotojo UA.

Sekančiuose dviejuose apibrėžimuose pristatomas paėmimas maksimalaus objekto skaitymo lygio vaidmenyje, ir minimalaus rašymo lygio, jeigu egzistuoja.

Apibrėžimas 6. R -lygis r vaidmenyje yra **mažiausia viršutinė riba** objekto saugumo lygių kuriems (o, r) yra r leidimuose.

Apibrėžimas 7. W -lygis r vaidmenyje yra **didžiausia apatinė riba** objekto saugumo lygių kuriems (o, w) yra r leidimuose, jeigu toks žemiausias lygis egzistuoja. Jeigu neegzistuoja, tada w -lygis nėra apibrėžtas.

Sekanti teorema išplaukia iš šitų apibrėžimų.

Teorema 3. RBAC96 konfigūracija patenkina paprasta saugumo ypatybę ir laisva *- ypatybę jeigu visi iš sekančių yra:

- Apribojimas Vartotojams: $(\forall u \in U)$
- Apribojimas Leidimams:

$$P = \{(o, r), (o, w) \mid o \text{ yra sistemos objektas}\}$$

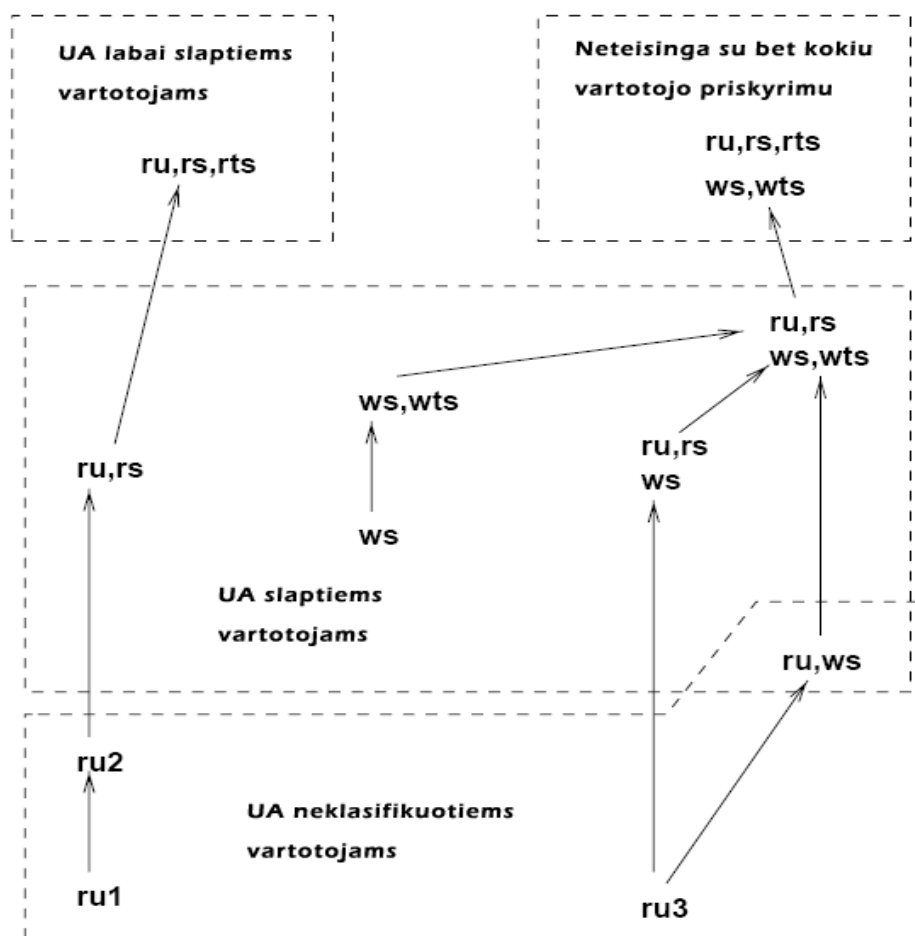
$$(\forall o \in P)$$
- Apribojimas UA:

$$((\forall r \in UA) [w\text{-lygis}(r) \text{ yra nustatytas}])$$

$$((\forall (u, r) \text{ priklauso UA}) [\lambda(u) \geq r\text{-lygis}(r)])$$

$$((\forall (u, r) \text{ priklauso UA}) [\lambda(u) \geq w\text{-lygis}(r)])$$
- Apribojimas Sesijoms: $((\forall s \in \text{sesijos}) [\lambda(s) \leq \lambda(u)])$

Pavyzdys parodo įmanomą vaidmens hierarchiją duota paveiksle (Paveikslas 2.8-7.), kur pabrauktas saugumo tinklelis turi žymeklių {neklasifikuotas, slaptas, labai slaptas} ir vaidmenys yra parodomi, pvz.: (*ru*, *rs*) reikšmės leidimų vaidmenyje įskaitant skaitymą kai kurių neklasifikuotų ir kelis slaptus objektus. Vaidmenys pažymėti *ru1* ir *ru3* apačioje turi skaitymo priėjimą prie atskirų objektų pažymėtų neklasifikuoti; *ru2* paveldi leidimus iš *ru1* ir turi papildomą skaitymo priėjimą prie objektų neklasifikuotame lygyje. Vaidmuisi pažymėtas (*ru*, *ws*) turi leidimų skaityti kai kuriuos neklasifikuotus objektus ir rašyti kai kuriuos slaptus objektus. Šis vaidmuo galėtų būti priskirtas UA neklasifikuotiems vartotojams arba slaptiems vartotojams[41].



Paveikslas 2.8-7. Vaidmens hierarchija ir jos vartotojų priskyrimai.

Iškėlimas vaidmens į viršų vaidmenų hierarchijoje, pažymėta (*ru*, *rs*, *rts*, *ws*, *wts*). Šis vaidmuo negali būti priskirtas nei vienam vartotojui nepažeidžiant Paprasto Saugumo Ypatybės arba Laisvos *-Ypatybės. Pastaba, jeigu vaidmuo yra ištrintas iš vaidmenų hierarchijos, mes turime vaidmenų hierarchijos pavyzdį, kuri patenkina Paprasto Saugumo Ypatybę ir Laisvą *- Ypatybę.

RBAC96 konfigūracija patenkina griežta *- ypatybę jeigu visos sąlygos išpildomos, keičiant apribojimus sesijoms į:

- Apribojimas Sesijoms: $(\forall s \in \text{sesijos}) [\lambda(s) \leq \lambda(u)]$ [41].

2.8.4.7. DAC modeliai

Nagrinėjami DAC polisai, kurie buvo aptarti ankstesniuose skyriuose. Pagrindinė DAC idėja yra tokia, kad objekto savininkas, kuris yra paprastai jo kūrėjas, turintis savarankišką valdžią lyginant su to, kas dar gali gauti prieigą prie to objekto. Kitaip tariant pagrindinis DAC polisas yra savininku pagrįsta prieigos teisių administracija. Yra daug variantų DAC polisų, ypač dėl to, kaip savininko savarankiška valdžia gali būti deleguota kitiems vartotojams ir kuomet prieiga yra panaikinta. Tai buvo pripažinta nuo ankstyviausių formuluočių DAC [Lampson; Graham ir Denning].

Nagrinėjamas metodas turi identifikuoti pagrindinius pakitimus DAC ir parodyti jų konstravimą RBAC96. Konstravimai yra tokie, kad bus akivaizdu, kaip jie gali būti išplėsti, kaip bus pasielgta su kitais susijusiais DAC pakitimais. Tai yra intuityvus, bet gerai įsteigtas, pateisinimas už tvirtinimą, kad DAC gali būti imituotas RBAC (Formalus įrodymas reikalingas, kad formalus apibrėžimas DAC, apima visus jo variantus, ir konstrukcija turėtų traktuoti visus RBAC96. Šis metodas buvo pateiktas Munawer).

DAC polisais laikomos visos dalys toliau išvardytų savybių.

- Objekto kūrėjas tampa jo savininku.
- Yra tikrai vienas objekto savininkas. Kai kuriais atvejais nuosavybė lieka pastovi su originaliu kūrėju, tuo tarpu kitais atvejais ji gali būti perduota kitam vartotojui. (Ši prielaida nėra kritiška mūsų konstravimams. Bus akivaizdu, kaip daugialypiai savininkai gali būti traktuojami.)
- Sunaikinti objektą gali tik tai jo savininkas.

Remiantis šia mintimi, yra apibrėžiami sekantys DAC variantai, į prieigos suteikimo galimybes.

1. **Griežtas DAC** reikalauja, kad savininkas būtų vienintelis, kuris turi savarankišką valdžią suteikti prieigai prie objekto ir kad nuosavybė negali būti perduodama. Pavyzdžiui, manykime, kad Alisa sukūrė objektą (Alisa yra objekto savininkė), ir suteikia skaitymo prieigą Bobui. Griežtas DAC reikalauja, kad Bobas negalėtų skleisti prieigos prie objekto kitam vartotojui. (Žinoma, Bobas gali nukopijuoti Alisos objekto turinį į objektą, kurį jis valdo, ir paskui skleisti

prieigą prie kopijos. Tai yra, todėl kad DAC negali pristesti informacijos srauto kontrolės, ypač dėl Trojos Arklių.)

2. **Liberalus DAC** leidžia savininkui deleguoti savarankišką valdžią tam, kad suteiktą prieigą prie objekto kitiems vartotojams. Toliau apibrėžiame kiti liberalaus DAC variantai.

A. **Viena lygmens suteikimas:** Savininkas gali deleguoti suteikdamas valdžią kitiems vartotojams, bet jie negali toliau deleguoti šios valdžios. Taigi Alisa būdama objekto O savininkė gali suteikti prieigą Bobui, kuris gali suteikti prieigą Čarlsui. Bet Bobas negali suteikti Čarlsui valdžios, kad toliau galėtų suteikti prieigą Dorotei.

B. **Dviejų lygmenų suteikimas:** Į vieno lygmens prieigą savininkas gali leisti kai kuriems vartotojams toliau deleguoti suteikiant valdžią kitiems vartotojams. Tokiu būdu, Alisa gali dabar įgalioti Bobą dviejų lygmenų suteikimams, tokiu būdu Bobas gali suteikti prieigą Čarlsui, su valdžia toliau suteikti prieigą Dorotei. Tačiau, Bobas negali suteikti dviejų lygmenų suteikiamos valdžios Čarlsui. (Galima svarstyti n-lygmenų suteikimą, bet bus akivaizdu, kaip padaryti tai nuo dviejų lygmenų konstrukcijos.)

C. **Daugialygmėnis suteikimas:** Šiuo atveju valdžia deleguojama valdžiai suteikiant ir duodant suprasti, kad ši valdžia gali būti jo deleguota. Tokiu būdu Alisa gali įgalioti Bobą, kuris toliau gali įgalioti Čarlsą, kuris gali toliau įgalioti Dorotę, ir taip toliau - neapibrėžtai.

3. **DAC su nuosavybės pakeitimu:** Šis variantas leidžia vartotojui perleisti nuosavybę objekto kitam vartotojui. Jis gali būti kombinuojama su griežtu ar liberaliu DAC, visuose anksčiau minėtuose variantuose.

Panaikinimui, pateikiami du tolimesni atvejai.

1. **Nepriklausomo suteikimo panaikinimas:** Panaikinimas yra nepriklausomas nuo suteikimo. Tokiu būdu Bobui gali būti suteikta prieigą Alisos, bet ji panaikinta Čarlso.
2. **Priklausomo suteikimo panaikinimas:** Panaikinimas yra smarkiai surištas su suteikimu. Tokiu būdu, jei Bobas gauna prieigą iš Alisos, prieiga gali būti panaikinta tik Alisos.

Šiuose konstravimuose, pradžioje priimamas nepriklausomo suteikimo panaikinimą ir paskui numanomas, kaip gali būti imituotas priklausomo suteikimo panaikinimas. Apskritai, taip pat manoma, kad bet kas turintis valdžia suteikti taip pat turi valdžią panaikinti. Toks sujungimas dažnai taikomas praktiškai. Kur atitinkamai, galima atskirti juos naudojamam imitavime, kadangi jiems atstovauja skirtingos teisės (leidimai).

Šita DAC polisai tikrai neišvengiami visais atvejais. Greičiau jie yra atstovaujanti polisų, kurie parodys, kaip kiti variantai gali būti taip pat traktuojami[41].

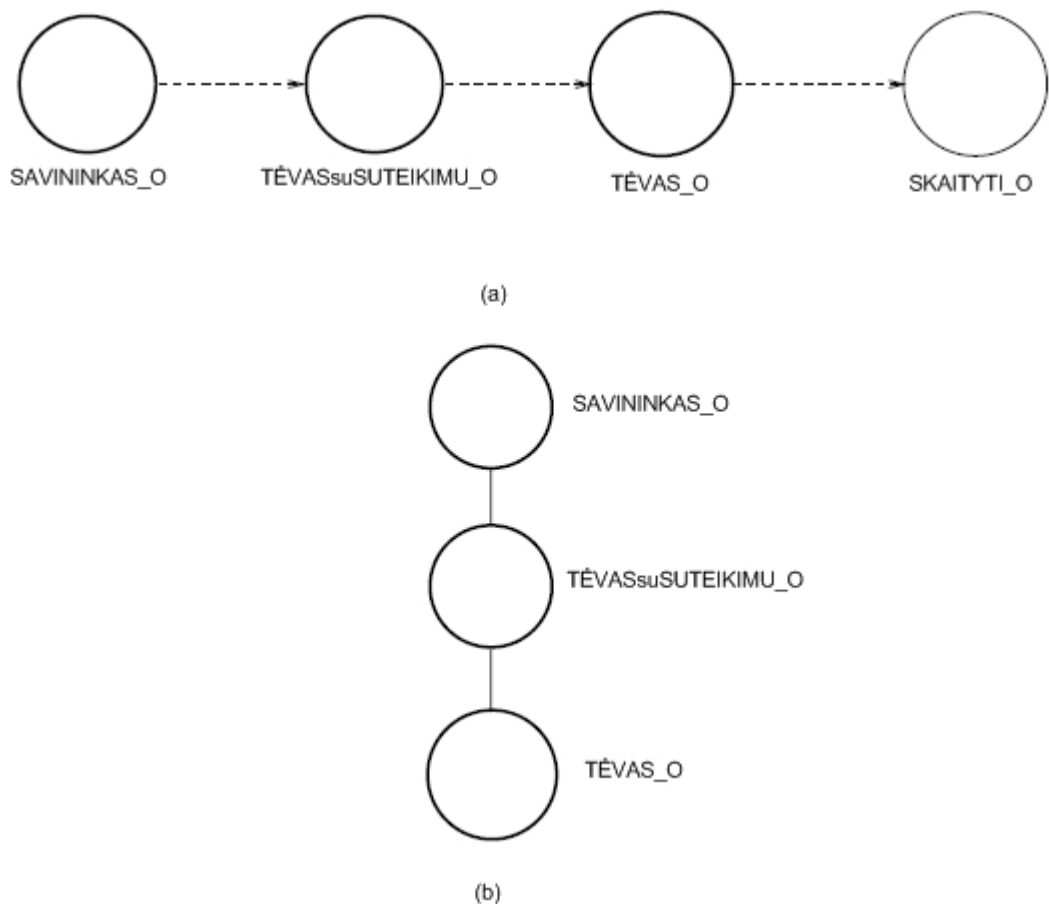
2.8.4.8.RBAC dėl DAC konfigūravimas

Apibrėžiami RBAC96 pakitimus, pakanka tam panaudoti DAC su viena operacija, šiuo atveju bus naudojama skaitymo operacija. Panašios konstrukcijos yra ir kitom operacijom tokiom, kaip kad rašymo, vykdymo ir papildymo, yra tai pateikti galima nesunkiai. Daug sudėtingesnės operacijos tokios kaip kopijavimas gali būti apibrėžiamos kaip skaitymas originalaus objekto ir rašymas (ir galbūt sukūrimas) kopijos. Gali būti naudinga asocijuoti kai, kurias numatytas (pagal nutylėjimą) teises su kopijavimu. Pavyzdžiui, kopijavimas gali prasidėti nuo prieigos, susietos su originaliu objektu, arba gali prasidėti nuo kažkokių tai kitų numatymų (įvykių, elementų ir pan.). Specifiniai polisai čia gali būti imituojami, plečiant naudojamas konstrukcijas. Prieš specifinių DAC variantų nagrinėjimą, iš pradžių apibūdinami bendri konstravimo aspektai[41].

2.8.4.8.1. Bendri aspektai

Pagrindinė įdėjo konstravime yra imituoti centrinius savininku polisus DAC naudojimas vaidmenis, kurie yra sujungti su kiekvienu objektu.

Sukurti Objektą. Kiekvienam objektui O, kuris yra sukurtas sistemoje, yra reikalaujama sukurti tris administracinius vaidmenis ir vieną reguliarų vaidmenį, kaip parodyta toliau.



Paveikslas 2.8-8. (a) Vaidmenų administravimas sujungtas su objektu; (b) Administracinio vaidmens hierarchija.

— Trys administraciniai vaidmenys AR: SAVININKAS_O, TĒVAS_O ir TĒVASsuSUTEIKIMU_O

— Vienas reguliarus vaidmuo R: SKAITYTI_O

Vaidmuo SAVININKAS_O turi privilegijas pridėti ir pašalinti vartotojus iš vaidmens TĒVASsuSUTEIKIMU_O, kuris savo ruožtu turi privilegijas pridėti ir pašalinti vartotojus iš vaidmens TĒVAS_O. Ryšys tarp šių vaidmenų parodytas Paveikslas 2.8-8. Paveikslas 2.8-8.(a) administraciniai vaidmenys parodyti su paryškintais apskritimų lankais negu reguliarūs vaidmenys. Paveikslas 2.8-8.(a) punktyrinės dešininės rodyklės rodo, kad vaidmuo kairėje pusėje turi savyje administracines teises, valdančias vaidmenį dešinėje. Paveikslas 2.8-8.(b) rodo administracinio vaidmens hierarchiją, su viršesniu vaidmeniu virš jo neatidėliotinai žemesnio, prijungto linija. Pavyzdžiui vaidmuo SAVININKAS_O turi administracinę valdžią vaidmeniui TĒVASsuSUTEIKIMU_O kaip parodyta Paveikslas 2.8-8.(a). Be to, dėl pavaldėjimo per vaidmens hierarchiją Paveikslas 2.8-8.(b) SAVININKAS_O taip pat turi administracinę valdžią prieš TĒVAS_O ir SKAITYTI_O.

Be to, reikalingas vienalaikis sukūrimas kitų aštuonių teisių greta sukūrimo kiekvieno objekto O.

- galiSkaityti_O: įgalioja skaitymo operaciją objektui O. Jis yra priskiriama vaidmeniui SKAITYTI_O.
- sunaikintiObjektą_O: įgalioja objekto pašalinimą. Jis yra paskiriamas vaidmeniui SAVININKAS_O.
- pridėtiSkaitytiVartotojas_O, pašalintiSkaitytiVartotojas_O: atitinkamai įgalioja operacijas pridėti vartotojus prie vaidmens SKAITYTI_O ir pašalinti juos nuo šio vaidmens. Jie paskiriami vaidmeniui TĖVAS_O.
- pridėtiTėvas_O, pašalintiTėvas_O: atitinkamai įgalioja operacijas pridėti vartotojus prie vaidmens TĖVAS_O ir pašalinti juos nuo šio vaidmens. Jie paskiriami vaidmeniui TĖVASsuSUTEIKIMU_O.
- pridėtiTėvasSuSuteikimu_O, pašalintiTėvasSuSuteikimu_O: atitinkamai įgalioja operacijas pridėti vartotojus prie vaidmens TĖVAS_O ir pašalinti juos nuo šio vaidmens. Jie paskiriami vaidmeniui SAVININKAS_O.

Šio teisės yra paskiriamos nurodytiems vaidmenims, kai objektas yra sukurtas, ir nuo to laiko jos negali būti pašalintos nuo šitų vaidmenų ar paskirtos kitiems vaidmenims.

Sunaikinti Objektą. Sunaikinimas objekto O reikalauja pašalinti keturis vaidmenis būtent SAVININKAS_O, TĖVAS_O, TĖVASsuSUTEIKIMU_O ir SKAITYTI_O ir aštuonias teises (sunaikinamos kartu su objektu). Tai gali padaryti tiktai savininkas, remdamasis sunaikintiObjektą_O teisės vykdymu[41].

2.8.4.8.2. Griežtas DAC

Griežtame DAC tik savininkas gali suteikti skaitymo priėjimą kitiems vartotojams. Autorius yra objekto savininkas. Narystės privalumas TĖVAS_O ir TĖVASsuSUTEIKIMU_O savininkas gali keisti vaidmens SKAITYTI_O priskyrimus. Naryste iš trijų administracinių vaidmenų negali keistis, taigi tik savininkas turi tokią teisę. Tokia veikla sustiprinta skiriant pagrindinį apribojimą iš 1 į SAVININKAS_O ir 0 į TĖVAS_O ir TĖVASsuSUTEIKIMU_O.

Šita veikla galētu būti imituota naudojant tik du vaidmenys SAVININKAS ir SKAITYTI, ir duodant pridėtiSkaitytiVartotojas_O ir trintiSkaitytiVartotojas_O leidimus tiesiogiai į SAVININKAS_O sukuriant iš O [41].

2.8.4.8.3. Liberalus DAC

Tris variantai liberalaus DAC, apibūdinti 2.8.4.7. skyriuje, apibudinami plačiau.

Vieno lygmens suteikimas. Vieno lygmens suteikimo DAC polise gali būti imituotas, pašalinant kardinalius apribojimus griežto DAC narystėje TĖVAS_O. Savininkas gali paskirti vartotojus TĖVAS_O vaidmeniui, kurie pagal eiliškumą gali paskirti vartotojus SKAITYTI_O vaidmeniui. Bet kardinalus apribojimas 0 ant TĖVASsuSUTEIKIMU_O lieka.

Dviejų lygmenų suteikimas. Dviejų lygmenų suteikimo DAC polise kardinalus TĖVASsuSUTEIKIMU_O apribojimas yra taip pat pašalintas. Dabar savininkas gali paskirti vartotojus TĖVASsuSUTEIKIMU_O, kurie toliau gali paskirti vartotojus TĖVAS_O. Pažymėtina, kad TĖVASsuSUTEIKIMU_O nariai gali taip pat paskirti vartotojus tiesiogiai SKAITYTI_O, tokiu būdu jie turi diskretiškumą šiuo atžvilgiu. Panašiai savininkas gali paskirti vartotojus TĖVASsuSUTEIKIMU_O, TĖVAS_O ar SKAITYTI_O atitinkamai. (N-lygmenų suteikimas gali būti panašiai imituotas, turint N vaidmenų, TĖVASsuSUTEIKIMU_O^{N-1}, TĖVASsuSUTEIKIMU_O^{N-2}, ..., TĖVASsuSUTEIKIMU_O, TĖVAS_O.)

Daugioalygmeninis suteikimas. Kad suteiktume prieigą per du lygmenis, mes įgaliojame vaidmenį TĖVASsuSUTEIKIMU_O priskirti vartotojus TĖVASsuSUTEIKIMU_O. Tai pasiekama, skiriant pridėtiTėvasSuSuteikimu_O teisę vaidmeniui TĖVASsuSUTEIKIMU_O, kai objektas O yra sukurtas. Kaip per bendrą sujungimo polisą suteikiama ir panaikinama valdžia, yra taip pat skiriamas pašalintiTėvasSuSuteikimu_O teisę vaidmeniui TĖVASsuSUTEIKIMU_O, kada O yra sukurtas. Ši sujungimo polisas yra be abejonės neprotingam kontekste nepriklausomų nuo suteikimo panaikina, tokiu būdu pašalintiTėvasSuSuteikimu_O teisė galėjo būti išsaugota tikrai su SAVININKAS_O vaidmeniu jei taip pageidaujamas. Kadangi priklausomas suteikimas panaikina sujungimą daug protingiau[41].

2.8.4.8.4. DAC su nuosavybės keitimu

Nuosavybės keitimas gali būti atliktas gana lengvai tinkamai apibrėžiant SAVININKAS_O nario administracinę valdžią. Atgaminti šitą nuosavybės keitimą tokiam kontekste reiškia nuosavybės perdavimą iš vieno vartotojo kitam. Vadinasi, SAVININKAS_O vaidmuo reikalauja leidimų kurie įgalina perdavimą ir gali būti priskirtas tik šitame

vaidmenyje. SAVININKAS _O narys gali priskirti kitą vartotoją į SAVININKAS _O bet savo nuosavybės dalies praradimo sąskaita[41].

2.8.4.8.5. Daugialypė nuosavybė

Daugialypė nuosavybė gali taip pat būti pritaikyta, pašalinant kardinalius apribojimus narystės SAVININKAS_O vaidmenyje. Kadangi visi SAVININKAS_O nariai turi identišką valdžią, įskaitant gebėjimą panaikinti kitus savininkus, tai būtų galima su nepriklausomu suteikimo panaikinimu, kad atskirtų originalų savininką. Iš esmės, galima turėti priklausomą suteikimą panaikinant nuosavybę[41].

2.8.4.8.6. Suteiktos-priklausomybės panaikinimas

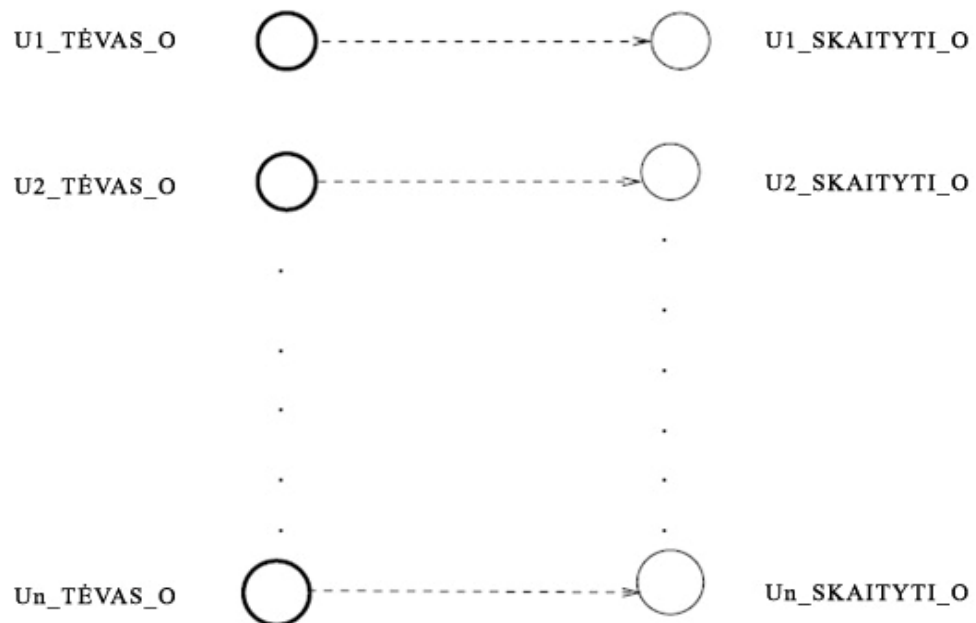
Taigi, laikoma suteiktos-priklausomybės panaikinimu kur panaikinimas priklauso nuo suteikėjo. Dabar apgalvosime kaip imituoti suteiktos-priklausomybės panaikinimą su RBAC96. Šio atveju tik tas vartotojas kuris turi priėjimą prie kito vartotojo, gali panaikinti priėjimą.

Ypatingai apgalvokime vieno lygio suteiktą DAC valdymą imitavusi ankščiau leidžiant TĖVAS_O vaidmens nariams priskyrimai vartotojus SKAITYTI_O vaidmeniui. Kad imituoti suteiktos-priklausomybės panaikinimą su šito vieno lygio valdymu, reikalingas skirtingas administravimo vaidmuo U_TĖVAS_O ir skirtingas pastovus vaidmuo U_SKAITYTI_O kiekvienam vartotojui U autorizuoto savininko daryti vieno-lygio suteikimą. Šitie vaidmenys yra automatiškai sukuriami kai savininkas autorizuoja vartotoją U. Tai pat reikalingi du nauji sukurti administraciniai leidimai.

- pridėtiU_SkaitytiVartotojas_O, trintiU_SkaitytiVartotojas_O: atitinkamai autorizuoja vartotojų įdėjimus į vaidmenį U_SKAITYTI_O ir ištrina juos iš šito vaidmens. Jie yra priskirti vaidmeniui U_TĖVAS_O.

U_i_TĖVAS_O valdo U_i_SKAITYTI_O vaidmens narystės priskyrimus paveikslas Paveikslas 2.8-9. vartotojui U_i. U_i_PARENT_O turi narystės kardinalumą apribota vienetu. Dar daugiau, šita narystė negali būti pakeista. Vartotojas U bus vienintelis suteikiantis ir panaikinantis vartotojus iš U_SKAITYTI_O. U_SKAITYTI_O vaidmuo sau priskiręs leidimus gali U_SKAITYTI_O kūrimo momentu. Mes galime leisti savininkui panaikinti vartotojus iš U_SKAITYTI_O vaidmens keičiant U_TĖVAS_O jaunesnį į

SAVININKAS_O administracinių vaidmenų hierarchijoje. Išplėtimai į sudėtinės narystės taip pat įmanomi[41].

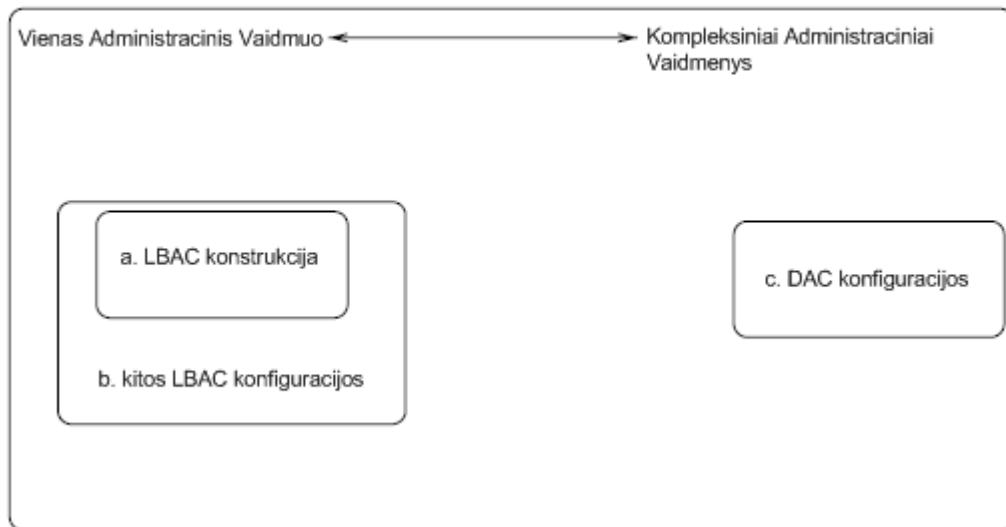


Paveikslas 2.8-9. Skaityti_O vaidmenys susiję su TĖVAS_O nariais.

2.8.4.9. Apibendrinimas

Aptartos LBAC ir DAC modelių standartinės formos gali būti imituotos kartu su RBAC96 ir su sisteminėmis konstrukcijomis. Visi komponentai RBAC96 modelio parodyti Paveikslas 2.8-2., yra reikalingi atliekant imitavimą. Vartotojai ir taisyės yra būtinos, kad išreikštų, bet kokį prieigos kontrolės modelį. Vaidmens Hierarchija yra svarbi LBAC imitavime. Administracinė Vaidmens Hierarchija yra būtina DAC polisams, kuomet administracinis vartotojas susiejamas su vaidmeniu, kuris atitinka tam tikrą užduotį[41].

RBAC modeliai



Paveikslas 2.8-10. Modelių apribojimai.

Galima pastebėti, kad teisės, kuriuos suteikiamos vartotojams DAC sistemoje gali duoti pasirinktinai turtingą vaidmens hierarchiją, kaip buvo pažymėtas pakeitime reliacinės duomenų bazės teisių į vaidmens grafus Osborn ir kt.. Apribojimai vaidina svarbų vaidmenį visose konstrukcijoje. Svarbu pažymėti, kad LBAC imitavimas apima vieną administracinį vaidmenį, tuo tarpu DAC imitavimui reikalingas didelis kiekis administracinių vaidmenų, kurie yra dinamiškai kuriami ir naikinami[41].

Galima pateikti kelias išvadas naudojantis Venn diagrama (Paveikslas 2.8-10.). Plotas esantis paveikslo kairėje parodo kad šitame RBAC96 konfigūracijų poaibyje nėra administravimo vaidmenų būtinybės, išskyrus vieną fiktyvų vaidmenį. Dešinėje, RBAC96 administracinė dalis yra pilnai panaudota. Dalis (a) pateikia įmanomų RBAC96 konfigūracijų poaibį, kuris sudėtas iš konstrukcijų 1 ir 2. Plotas (b) parodo kad yra ir kitokių konfigūracijų nesudėtu iš šitų dviejų konstrukcijų kuriuos patenkina LBAC savybės. Dalis (c) pateikia iš pagrindų RBAC96 konfigūracijas sudėtas iš skirtingų konstrukcijų. Pastaba, šitos paskutinės konstrukcijos papuola į regioną, kur RBAC96 modelio administraciniai vaidmenys yra pilnai panaudoti[41].

Ateities darbas turētu būti sufokusuotas ant to kas vyksta su RBAC96 modeliais neįtrauktais į plotus konstruotus šiame straipsnyje. Modeliai decentralizuotam vaidmenų valdymui, kurie patenka tarp šitų kraštutinybių pasiūlytų Sandhu[45]. Šitie modeliai leidžiami dideliems administracinių vaidmenų skaičiams, bet tas skaičius tikėtina bus mažesnis negu objektų skaičius sistemoje[41].

Pabaigai, apžvelgėme skirtingas sistemingas konstrukcijas kaip imituoti ir sustiprinti tradicinį LBAC ir DAC priėjimo valdymo modelius esančius RBAC96 sudėtyje[41].

2.9. Elektroninių apmokėjimų paslaugų teikimo saugumą užtikrinančios kompanijos ir technologijos-priemonės

2.9.1. Saugumo technologijų-priemonių reikalingumo samprata

Interneto pagalba teikiamos paslaugos turi begales privalumu, tačiau jos susiduria su įvairiomis saugumo problemomis. Saugumui užtikrinti yra nemažai įvairiausių kriptografijos algoritmų. Be to vien tik su programuotas kriptografijos vienas neapsaugos sistemos, todėl dar prie visa to reikia sukurti galinga sistema kuri rūpintųsi tuo saugumus. Be to turi tenkinti begales reikalavimų ir standartų. Todėl visa tai programuoti yra sudėtinga ir netikslinga. Todėl pasaulyje yra nemažai kompanijų siūlančių apsaugos paslaugas internetinėms svetainėms[46]. Toliau paminėsime tik pačias populiariausias sistemas

2.9.2. Svarbiausios elektroninių apmokėjimo paslaugų teikimo saugumą užtikrinančios kompanijos ir technologijos-priemonės

2.9.2.1. Baltimore „UniCERT“

Baltimore tai kompanija kuri specializuojasi vystydama ir gamindama apsaugos sistemas ir teikdama apsaugos paslaugas, įvairaus pobūdžio verslo svetainėms, esančioms internete ar intranete. Pagrindinis Baltimore kompanijos produktas yra „UniCERT" sistema, skirta apsaugoti tinklapiams, ir aparatiniai įrangai bei kurti sertifikatus[47].

2.9.2.2. Brokat „Twister“

Brokat kompanija veikia daugiau kaip 16 pasaulio šalių. Kompanija gamina programinę įrangą skirtą elektroninio verslo sektoriui, ir užima svarbia vietą elektroninių bankinių sistemų nišoje. Pagrindinis kompanijos produktas „Twister“ kuris turi visas reikiamas priemones, leidžiančias sukurti ir apsaugoti modernias internetinio verslo sistema[48].

2.9.2.3. VeriSign „VeriSign“

VeriSign tai kompanija, kuri specializuojasi gamindama svetainių apsaugos produktus, ir užtikrindama pastarųjų apsaugą. Šios kompanijos produktai yra skirti apsaugoti internetiniu portalus, ir jie specializuojasi į vieno pilno portalo apsaugą, nepriklausomai nuo naudojamų tinklų tipo, tai gali būti: internetas, intranetas ir t.t.

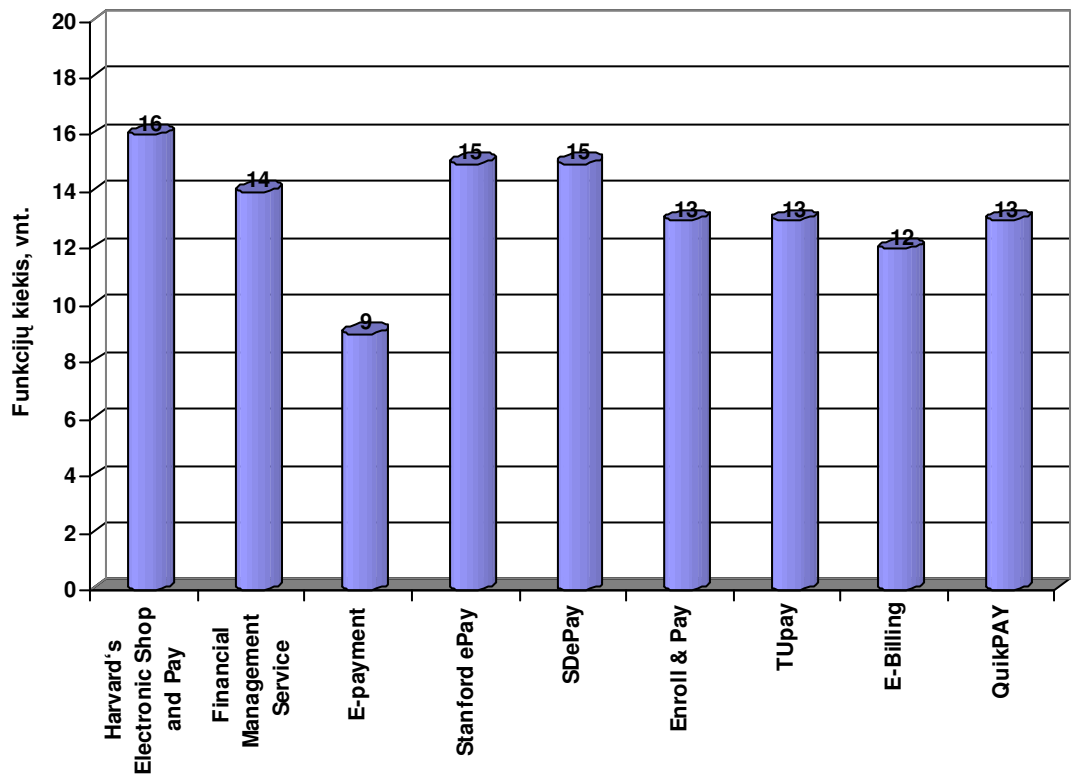
VeriSign kompanija prekiauja skaitmeniniais sertifikatai kurie yra populiarūs visame pasaulyje, taip pat ir Lietuvoje. VeriSign klientams siūlo skaitmeninius 128-bit SSL sertifikatus ir skaitmeninius 40-bit sertifikatus[49].

2.10. Apžvelgtų sistemų kokybės ir funkcionalumo apibendrinimas

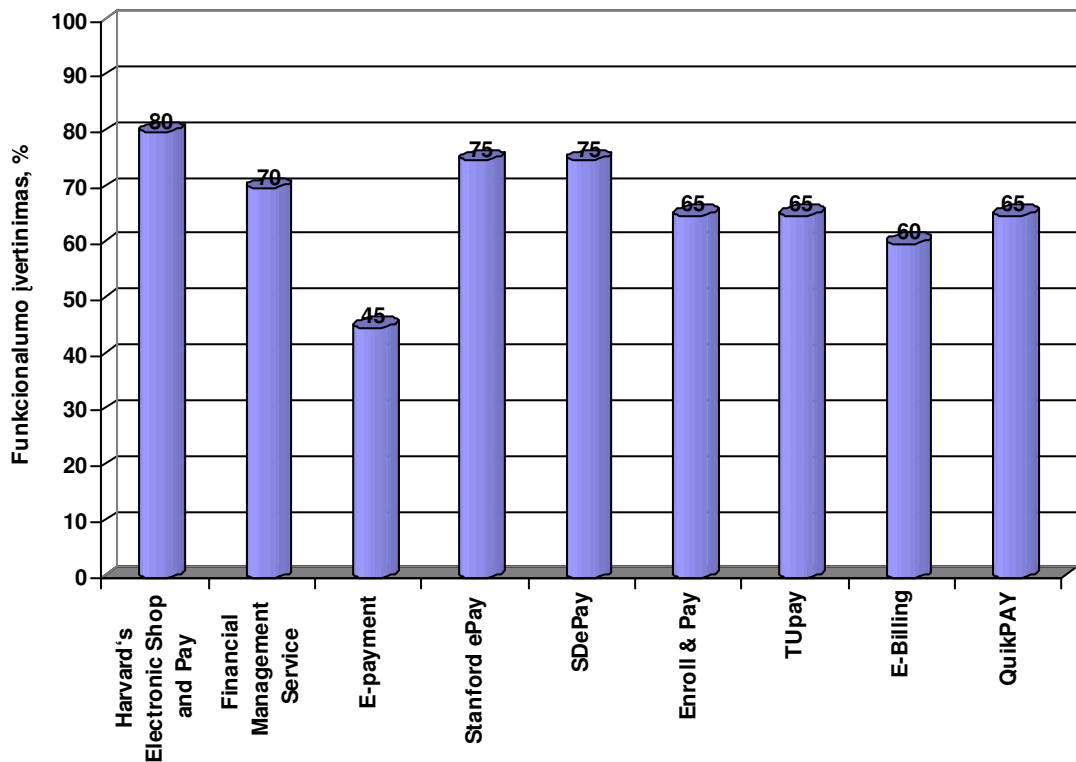
Elektroninių mokėjimų universitetinėje aplinkoje sistemų apžvalgos ir analizės metu buvo apžvelgti tokie egzistuojantys sprendimai kaip: Harvard's Electronic Shop and Pay, Financial Management Service, E-payment, Stanford ePay, SDePay, Enroll & Pay, TUpay sistema, John Hopkins universiteto elektroninio apmokėjimo sistema E-Billing, Oregono universiteto elektroninio apmokėjimo sistema QuikPAY. Iš visų nagrinėtų sistemų galima išvardinti tokias visom sistemų kokybes bei giminingas funkcijas, kurios yra pateiktos žemiau esančioje lentelėje Lentelė 2.10-1. Apžvelgtų sistemų funkcionalumo apibendrinimas“.

Lentelė 2.10-1. Apžvelgtų sistemų funkcionalumo apibendrinimas

Eil. Nr.	Atliekamo funkcijos	Sistemos								
		Harvard's Electronic Shop and Pay	Financial Management Service	E-payment	Stanford ePay	SDePay	Enroll & Pay	TUpay	E-Billing	QuikPAY
		<i>Harvard University</i>	<i>Indiana University</i>	<i>Trent University</i>	<i>Stanford University</i>	<i>South Dakota State University</i>	<i>University of Kansas</i>	<i>University of Temple</i>	<i>Johns Hopkins University</i>	<i>University of Oregon</i>
1.	Vartotojo autentifikacija	+	+	+	+	+	+	+	+	+
2.	Informacijos konfidencialumas	+	+	+	+	+	+	+	+	+
3.	Peržiūrėti sąskaitas	+	+	+	+	+	+	+	+	+
4.	Apmokėti įvairias sąskaitas	+	+			+		+	+	+
5.	Apmokėti įvairias sąskaitas susijusias su universiteto teikiamomis paslaugomis.	+	+		+	+	+	+	+	+
6.	Apmokėti universitetines sąskaitas už mokslą	+	+	+	+	+	+	+	+	+
7.	Automatinis sąskaitų apmokėjimas	+	+		+					
8.	Automatiškai tvarkoma sąskaitų informacija	+	+		+	+				
9.	Automatinis sąskaitų apmokėjimo formų pildymas						+			
10.	Sąskaitos balanso kontrolė	+	+	+	+	+	+	+	+	+
11.	Sąskaitų istorija	+	+	+	+	+	+	+	+	+
12.	Išsami visų rūšių sąskaitų istorija			+	+		+	+		
13.	Automatiniai pranešimai elektroniniu paštu apie naujas sąskaitas	+	+		+	+	+	+	+	+
14.	Automatiniai elektroninių apmokėjimų patvirtinimai elektroniniu paštu	+	+		+	+	+	+	+	+
15.	Veikla 24 valandas per parą ir 7 dienas per savaitę	+	+		+	+				
16.	Veikla darbo dienomis ir darbo laiku	+	+	+	+	+	+	+	+	+
17.	Autorizuoti mokėtojai			+	+	+	+	+	+	+
18.	Tiesioginis prisijungimas prie sistemos iš vartotojo elektroninio pašto					+				+
19.	Peržiūrėti partnerių siūlomas prekes/paslaugas	+								
20.	Apsipirkti partnerių svetainėse(arba lokaliai)/pirkti prekes ar paslaugas	+								
Funkcijų kiekis:		16	14	9	15	15	13	13	12	13
Funkcionalumo įvertinimas:		80%	70%	45%	75%	75%	65%	65%	60%	65%



Paveikslas 2.10-1. Apžvelgtų sistemų funkcionalumo apibendrinimas, pagal funkcijų kiekį.



Paveikslas 2.10-2. Apžvelgtų sistemų funkcionalumo įvertinimas.

2.11. Kokybės vertinimas

Elektroninių mokėjimų universitetinėje aplinkoje sistemoms svarbus toks kaip kokybės vertinimo parametras, nes pagal tai dauguma sistemų vartotojų pasirenka sistemas.

Apžvelgus elektroninių mokėjimų universitetinėje aplinkoje sistemas (buvo apžvelgtos tokios sistemos, kaip kad: Harvard's Electronic Shop and Pay, Financial Management Service, E-payment, Stanford ePay, SDePay, Enroll & Pay, TUpay sistema, John Hopkins universiteto elektroninio apmokėjimo sistema E-Billing, Oregono universiteto elektroninio apmokėjimo sistema QuikPAY). Apžvelgus anksčiau paminėtas sistemas, buvo išskirtos kiekvienos sistemos, kokybės įvertinimas. Kiekviena sistema gavo tam tikrą balą.

Sistemos buvo vertinamos pagal tokius punktus:

1. Saugumas
2. Išplėčiamumas
3. Panaudojamumas
4. Vartotojo interfeisas
5. Funkcionalumas
6. Stabilumas
7. Greitaveika

Saugumas – sistemos galimybė apsisaugoti nuo įsilaužėlių, nesankcionuotų priėjimų prie jos resursų ir operuojamų duomenų.

Išplėčiamumas – vertinama kiek sistema galima išplėsti ir pritaikyti lankstesniam naudojimui

Panaudojamumas – parodo kiek sistema yra panaudojama, kiek kokios operacijos dažnai naudojamos, gal kurios net nereikalingos

Vartotojo interfeisas – parodo kiek sistema lengvai suprantama vartotojui, ant kiek yra interaktyvus meniu

Funkcionalumas – rodo kiek gerai funkcionuoja sistema ir atskiri jos moduliai

Stabilumas – kiek sistema yra stabili įvairiems nenumatytiems atvejams

Greitaveika – sistemos veikimo sparta

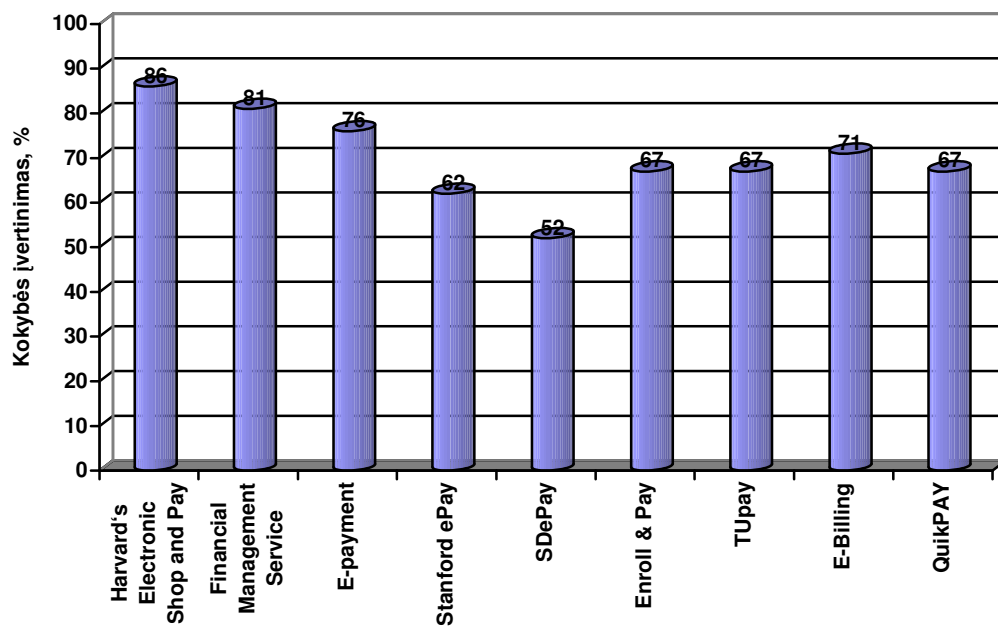
1 – sistema veikia blogai, dar reikia darbo kad ištaisyti klaidas

2 – sistema veikia neblogai, beveik nepasitaiko klaidų

3 – sistema veikia gerai, klaidų pranešimų nėra, sklandus veikimas

Lentelė 2.11-1. Apžvelgtų sistemų kokybės vertinimas

Eil. Nr.	Kokybės vertinimo požymis	Sistemos								
		Harvard's Electronic Shop and Pay	Financial Management Service	E-payment	Stanford ePay	SDePay	Enroll & Pay	TUpay	E-Billing	QuikPAY
		<i>Harvard University</i>	<i>Indiana University</i>	<i>Trent University</i>	<i>Stanford University</i>	<i>South Dakota State University</i>	<i>University of Kansas</i>	<i>University of Temple</i>	<i>Johns Hopkins University</i>	<i>University of Oregon</i>
1.	Saugumas	3	3	2	2	1	2	3	2	3
2.	Išplečiamumas	2	3	1	1	1	2	3	2	2
3.	Panaudojamumas	3	2	2	1	2	2	2	3	1
4.	Greitaveika	3	3	3	2	1	3	1	1	2
5.	Funkcionalumas	2	2	2	3	2	2	2	3	3
6.	Stabilumas	3	2	3	2	3	2	1	2	2
7.	Vartotojo interfeisas	2	2	3	2	1	1	2	2	1
Kokybės įvertinimas:		86%	81%	76%	62%	52%	67%	67%	71%	67%



Paveikslas 2.11-1. Apžvelgtų sistemų kokybės įvertinimas.

2.12. Išvados

Analizės metu buvo apžvelgti tokie egzistuojantys sprendimai kaip: Harvard's Electronic Shop and Pay, Financial Management Service, E-payment, Stanford ePay, SDePay, Enroll & Pay, Temple universiteto TUpay sistema, John Hopkins universiteto elektroninio apmokėjimo sistema E-Billing, Oregono universiteto elektroninio apmokėjimo sistema QuikPAY. Iš visų nagrinėtų sistemų galima išvardinti tokias visom giminingas funkcijas:

- Peržiūrėti sąskaitas
- Apmokėti įvairias sąskaitas
- Automatiškai tvarkoma sąskaitų informacija
- Atliekami automatiniai apmokėjimai
- Sąskaitos balanso kontrolė
- Sąskaitų istorija
- Automatiniai pranešimai elektroniniu paštu apie naujas sąskaitas
- Automatiniai elektroninių apmokėjimų patvirtinimai elektroniniu paštu
- Veikla 24 valandas per parą ir 7 dienas per savaitę

Elektroninio apmokėjimo sistemos yra labai populiaros užsienio universitetuose. Dėl tokių teikiamų privalumų:

- Patogu ir paprasta
- Asmeniškai ir saugu
- Nėra laukimų norint padaryti apmokėjimą
- Priminimai ir patvirtinimai siunčiami į el. pašta
- Elektroninių sąskaitų ir apmokėjimų istorija iki 12 mėnesių
- Priėjimas prie vartotojo srities, bet kur per internetą

Elektroninės apmokėjimo sistemos pagrindiniai reikalavimai galėtų būti:

- Privatumas
- Atomiškumas
- Bendradarbiavimas
- Pajėgumas
- Saugumas
- Patikimumas
- Patogumas

- Automatiškumas

Saugumas išvardintuose sistemose yra itin svarbus, todėl jam užtikrinti yra naudojami tokios priemonės kaip: simetrinis, asimetrinis šifravimas, skaitmeninis parašas, nepastebimas parašas. Taip pat yra naudojami saugumą užtikrinantys protokolai: SSL, TLS, Kerberos. Duomenų ir informacijos apsaugai sustiprinti naudojama prieigos kontrolė.

Kadangi saugumo sistemas yra pakankamai sudėtinga sukurti, be to jos turi tenkinti daugybę standartų, būti suderinamos su bet kokia vartotojo naudojama programine įranga. Tokios elektroninių apmokėjimo paslaugų teikimo saugumą užtikrinančios kompanijos ir jų produktai yra: Baltimore „UniCERT“, Brokat „Twister“, VeriSign „VeriSign“.

Apžvelgus visas paminėtas sistemas, nagrinėjome jų funkcionalumo, saugumo, patikimumo aspektus. Įsitikinome kad kai kurios vietos jų nėra visiškai išbaigtos, todėl mes kurdami panašią sistemą bandysime ankščiau paminėtus reikalavimus pateikti pilnai išpildytus.

3. PROJEKTAVIMO TYRIMO DALIS

3.1. Įvadas

Projektavimo tyrime nagrinėjamos „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos projektavimo technologinės detalės. Konkretizuojami priimti sprendimai, pateikiamos architektūros ir modulių diagramos, aptariami priimtų sprendimų rezultatai. Pirmuosiuose skyriuose trumpai aptariami priimti techniniai sprendimai. Vėliau apžvelgiamas bendras sistemos architektūros vaizdas, tuo pačiu pateikiamos esminės sistemos architektūrą paaiškinančios schemas. Taip pat pateikiamas sistemos duomenų modelis ir aptariama projekto išėiga.

3.2. Apžvalga

3.2.1. Sistemos paskirtis

3.2.1.1. Sistemos kūrimo pagrindimas

Finansiniai apmokėjimai yra vienas iš natūraliausių ir seniausių žmonių atsiskaitymo už įvairias prekes ir paslaugas būdų. Tai liečia ir informacines technologijas, kurios atveria plačias ir neribotas galimybes, kurias galima realiai pritaikyti finansiniams apmokėjimams. Nors idėja panaudoti informacines technologijas elektroniniams apmokėjimams nėra nauja, tačiau yra sukurta labai nedaug programinės įrangos priemonių, kurios leistų elektroninius apmokėjimus realizuoti universitetinėje aplinkoje. Kitaip tariant informacinių technologijų panaudojimas elektroninių apmokėjimų srityje yra vis dar daug žadanti sritis su itin didelėmis galimybėmis ir perspektyvomis, kurios bus išnaudotos dar labai greitai.

Realizuojamo tyrimo siekis yra sukurti „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemą. Sukurta apmokėjimų programinė įranga galės būti panaudota atlikti įvairioms finansinėms operacijoms universitetinėje aplinkoje, kaip pavyzdį būtų galima paminėti tokias kuriamos programinės įrangos galimybes: priimti iš studentų ar darbuotojų apmokėjimus už: kopijavimo, spausdinimo, maitinimo, apsipirkinėjimą knygyne ir pan. paslaugas.

Naudojant elektroninio apmokėjimo programinę įrangą, finansinio atsiskaitymo operacijos bus atliekamos kur kas greičiau nei tradiciniu būdu, kuomet atsiskaitymai atliekami grynais, kai mokėtojas turi suskaičiuoti reikiamą pinigų sumą, o po to pardavėjas vėl ją paskaičiuoti ir dar jei reikia suskaičiuoti grąžą, tai užtrunka labai ilgai ir gaištama bereikalingai daug laiko. Dar blogesnis atvejis, kai apmokėjimui už paslaugas ir pan. išduodamas kvitas, kurį reikia kažkur apmokėti, o po to vėl grąžinti, čia yra apkraunami ir klientai ir darbuotojai, nes mokėtojas sugaišta daug laiko, kol apmoka, o darbuotojams reikia suregistruoti, kas ir kada sumokėjo ar nesumokėjo. Tai labai nepatogu, todėl kuriama sistema apmokėjimą leis atlikti vos per kelias sekundes, o ne per kelias valandas ir daugiau, kaip tai yra dabar. O, be to, vartotojas, bet kada galės gauti išsamią savo finansinę informaciją, kurios dabar negali gauti, nes paprasčiausiai niekas individualiai neranka duomenų, kiek kur ir koks asmuo išleido, nebent pats asmuo renka tokią savo informaciją. Be to, sistema, bet kuriuo

laikų rodys realią universiteto aplinkoje vykstančių finansinių operacijų apimtį (ten kur ji bus įdiegta), ne taip kaip dabar, kai visa finansinių operacijų informacija gaunama dienos ar mėnesio pabaigoje, po to kai atliekami „dideli“ ir labai varginantys skaičiavimai. Sukurta programinė įranga leis pasinaudojant studento ar darbuotojo pažymėjimu atlikti apmokėjimą už paslaugas (tiesiog pinigai bus pervedami iš vienos sąskaitos į kitą), be to bus kur kas saugiau, nes nereikės turėti grynųjų pinigų, kuriuos gali kas nors pavogti. Taip pat bus galima peržiūrėti savo sąskaitos likutį, atlikti pervedimus „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos teikiamomis galimybėmis.

Tačiau žvelgiant į „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos galimybes kyla klausimas. Kodėl negalima būtų tam panaudoti, gerai žinomų: SEB Vilniaus bankas, Hansabankas, DnB Nord bankas, Snoras bankas, ir kitų bankų teikiamų elektroninių paslaugų. Kiekvienas bankas naudoja savo individualią finansinių operacijų sistemą, todėl yra labai sunku suderinti visus bankus. Be to, finansinės transakcijos tarp pardavėjo ir banko trunka labai ilgai. Atsiskaitymų informacija gaunama ne iškart, tik nustatytais terminais. Taip pat bankai apmokestina visas savo teikiamas paslaugas, todėl iškart kyla problema, kas padengs aptarnavimo išlaidas universitetas, ar mokėtojas. Todėl mažavertės paslaugos, prie jų pridėjus banko aptarnavimo paslaugų kainą, tampa labai brangiomis (pvz. kopijuojame ar spausdiname 1 lapą, kaina būtų kopijavimas ar spausdinimas 0,10 Lt (tokia kaina buvo 2008 metų balandį Kauno technologijos universitete) plius 0,80 Lt banko paslaugų aptarnavimo mokestis (tokia vidutinė kaina buvo 2008 metų balandį Lietuvos bankuose), viso gaunasi 0,90 Lt, todėl paslauga šiuo atveju pabrangsta net 9 kartus). Todėl naudojant „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos programinę įrangą: finansinės transakcijos būtų atliekamos labai sparčiai, visa informacija būtų gaunama realiu laiku, nekiltų suderinamumo su bankais problemos, be to nebūtų naudojamas paslaugos teikimo mokestis, to pasėkoje net ir itin mažavertės paslaugos netaptų labai brangiomis.

Naudojant „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos programinę įrangą, finansinio atsiskaitymo operacijos bus atliekamos kur kas greičiau nei tradiciniu būdu, o, be to, bus pateikiama visa išsami finansinių operacijų informacija.

3.2.1.2. Sistemos tikslai

Tiriamąo projektavimo tikslas iširti naujausias elektroninių mokėjimu technologijas, jų teikiamas galimybes, funkcionalumą, realizaciją ir pritaikymą realiame gyvenime, bei universiteto aplinkoje, šių technologijų specifikaciją ir problemas. Šio projekto metu bus sukurta „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos prototipas. Sukurta elektroninių mokėjimų programinė įranga galės būti panaudota atlikti įvairioms finansinėms operacijos universitetinėje aplinkoje, kaip pavyzdį būtų galima paminėti, tokias kuriamos programinės įranga galimybes: priimti iš studentų ar darbuotojų apmokėjimus už: kopijavimo, spausdinimo, maitinimo, apsipirkinėjimą knygyne ir pan. paslaugas. Sukurta programinė įranga leis pasinaudojant studento ar darbuotojo pažymėjimu atlikti apmokėjimą už paslaugas. Peržiūrėti savo sąskaitos likutį, atlikti pervedimus elektroninio mokėjimo sistemos teikiamomis galimybėmis. Taigi šiame darbe bus koncentruojamasi daugiau ties elektroninių apmokėjimų atsiskaitymo galimybėmis bei finansinės informacijos tvarkymo funkcijomis ir galimybėmis, t.y. bus koncentruojamasi ties kuriamos sistemos branduoliu, nei ties kitomis kuriamos sistemos teikiamomis paslaugomis.

Sukurta programinė įranga galės būti panaudota ir kituose panašiuose projektuose, kuriose bus poreikis realizuoti programinę įrangą, susijusią su elektroniniais mokėjimais. Panaudojant sukurta programinę įrangą (sistemą), bus padidintas elektroninių mokėjimų universitetinėje aplinkoje efektyvumas ir našumas.

Naudojant „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos programinę įrangą, finansinio atsiskaitymo operacijos bus atliekamos kur kas greičiau, o, be to, vartotojai (universiteto studentai, personalas, bei kiti susiję asmenys), bet kada galės gauti išsamią savo finansinę informaciją.

3.2.1.3. Sistemos kontekstas

Sukurta apmokėjimų programinė įranga galės būti panaudota atlikti įvairioms finansinėms operacijos universitetinėje aplinkoje, kaip pavyzdį būtų galima paminėti, tokias kuriamos programinės įranga galimybes: priimti iš studentų ar darbuotojų apmokėjimus už: kopijavimo, spausdinimo, maitinimo, apsipirkinėjimo knygyne ir pan. paslaugas. Tačiau šita sistema, žvelgiant į „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos galimybes, panaudoti, gerai žinomų: SEB Vilniaus bankas, HansaBankas, DnB Nord bankas, Snoras bankas, ir kitų, bankų teikiamas elektroninės paslaugas.

Apskritai į produkto sudėtį įeina tokios dalys kaip: elektroninių mokėjimų serveris, universiteto elektroninių mokėjimų tinklas, duomenų bazė, internetas ir interneto naršyklės, pardavėjas/tiekėjas, buhalteris, administratorius. Kiekvienas jų veikia kaip darni sistema bendradarbiaudami tarpusavyje. Vartotojai santykiauja per interneto naršyklę su duomenų bazėje saugoma informacija. Jie yra per interneto naršyklę pasiekiantys, bei tvarkantys, valdantys savo sąskaitas iš bet kurios pasaulio vietos. Pardavėjas kitaip sakant paslaugos tiekėjas, tiesiogiai santykiauja su klientu, nes priima elektroninius apmokėjimus. Administratorius santykiauja su universiteto elektroninių mokėjimų tinklu ir duomenų baze; tvarkydamas kompiuterinį tinklą ir duomenų bazėje esančius įrašus. Buhalteris pagrinde santykiauja su vartotojo sąskaitomis ir elektroninių mokėjimų serveriu.

3.2.2. Svarbūs faktai ir prielaidos

3.2.2.1. Svarbūs faktai

Pateikiami svarbūs išoriniai faktoriai, įtakoiantys kuriamą „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemą, tačiau nepriklausantys iš anksto nustatomiems apribojimams, kurie įtakoja reikalavimų specifikaciją. Šiame punkte charakterizuojami kiti išoriniai veiksniai, veiklos (pasaulio mastu), galintys įtakoti sistemą.

Svarbūs faktai:

- Kuriamas projekto „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ programinė įranga bus pateikta, kaip uždaro kodo programinė įranga, todėl ja naudotis galės tik asmenys turintys atitinkamas teises bei leidimus.
- Bus užtikrintas elektroninių mokėjimų sistemoje dalyvaujančių asmenų, asmeninės ir finansinės informacijos konfidencialumas ir saugumas.

3.2.2.2. Prielaidos

Pateikiamas „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos prielaidų sąrašas. Jos liečia įvairius sistemos kūrimo aspektus. Prielaidos, tam tikra prasme, yra faktų priešingybė, nes gali būti neteisingos, tačiau gali ir pasiteisinti.

Prielaidos:

- Absoliuti dauguma programos funkcijų bus realizuota nesinaudojant kitų šalių sukurtais komponentais.
- Kliento-vartotojo identifikacijai gali būti panaudoti, kitų šalių sukurti komponentai.
- Bendravimas su Kauno Technologijos Universiteto akademinė informacinė sistema, reiks naudoti duomenų ištraukimą iš informacinės sistemos duomenų saugyklos.
- Sprendimui pateikti gali tekti modifikuoti Kauno Technologijos Universiteto finansinio atsiskaitymo infrastruktūrą.
- Sprendimui pateikti klientui gali tekti įsigyti specializuotą programinę ir techninę įrangą klientų-vartotojų identifikacijai.
- Gali pasikeisti esami ar atsirasti nauji įstatymai, kurių pasikeitimas ar atsiradimas įtakoja finansinius atsiskaitymus ir finansinių atsiskaitymų programinės įrangos veiklą.
- Sėkmingai atlikus „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektą, ir sėkmingai pritaikius sukurtą elektroninių mokėjimų universitetinėje aplinkoje programinę įrangą Kauno Technologijos Universitete, tikimasi toliau plėsti elektroninių mokėjimų universitetinė aplinkoje programinės įrangos pritaikymą įvairiose Kauno Technologijos Universiteto srityse, o taip pat produktą pritaikyti pardavimui.

3.2.3. Konceptija

Sukurtą „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projekto sistema, bus panaudota atlikti įvairioms finansinėms operacijoms universitetinėje aplinkoje, kaip pavyzdį būtų galima paminėti, tokias kuriamos programinės įrangos galimybes: priimti iš studentų ar darbuotojų apmokėjimus už: kopijavimo, spausdinimo, maitinimo, apsipirkinėjimą knygyne ir pan. paslaugas. Sukurta programinė įranga leis pasinaudojant studento ar darbuotojo pažymėjimu atlikti apmokėjimą už paslaugas. Peržiūrėti savo sąskaitos likutį, atlikti pervedimus „Elektroninių mokėjimų universitetinėje aplinkoje programinės įrangos“ teikiamomis galimybėmis.

Toliau pateikiama aprašymas, kaip visa tai turėtų atrodyti ir funkcionuoti.

Klientas-vartotojas norėdamas susimokėti už prekes ar suteiktas paslaugas pas pardavėją, ar paslaugų teikėją (pardavėją/teikėją), kai pardavėjas/ teikėjas suformuoja sąskaitą

ir paprašo kliento-vartotojo apmokėti, klientas-vartotojas pateikia mokėjimo kortelę (t.y. studento ar darbuotojo pažymėjimą, kuris atitinka mokamąją kortelę). Pardavėjas/teikėjas parbraukia kortele per mokėjimo kortelių nuskaitymo įrenginį, toliau sistema prie kliento-vartotojo informacijos gautos iš mokomosios kortelės prideda pardavėjo/teikėjo bei pastarojo sąskaitos informaciją ir toliau visa informacija per universitetinį mokėjimu (universiteto vidinį) tinklą išsiunčiama į elektroninių mokėjimų serverį, kuris patikrina ar klientas-vartotojas ir pardavėjas/teikėjas yra tikri, jai kuris nors ar abu netikri, tai apie tai gražinamas pranešimas, jai abu tikri, tai toliau patikrina ar kliento-vartotojo sąskaitoje yra pakankamai pinigų, jai pinigų trūksta, tai gražina informaciją pardavėjui/ teikėjui, kad apmokėjimas negalimas. Jei kliento-vartotojo sąskaitoje yra pakankamai pinigų tai iš kliento-vartotojo sąskaitos reikimą sumą perveda ir pardavėjo/ teikėjo sąskaitą, ir apie apmokėjimą, informuojamas pardavėjas/teikėjas. Taip pat suregistruojama visą reikiama informacija duomenų bazėje.

Sistemos kliento-vartotojas (universiteto studentas ar darbuotojas) norėdamas įnešti į savo sąskaitą pinigus kreipiasi į buhalterį, kuris įmokėtą į universiteto sąskaitą pinigų sumą perveda į kliento-vartotojo sąskaitą (įmoką gali būti gynais, pervedimu ir pan.).

Buhalteris tvarko ne tik įmokas, bet taip pat ir visą finansinę informaciją, kaip kad: sąskaitas, jų paskirstymą klientams-vartotojams, sąskaitų balansą ir pan. Buhalteris taip pat lieka dalį administratoriaus funkcijų, kurios yra glaudžiai susijusios su jo veikla, kad vartotojų informacijos tvarkymas ir pan.

Administratorius prižiūri elektroninių mokėjimų kompiuterinį tinklą, tvarko ir prižiūri duomenų bazę, rūpinasi, kad visa sistema dirbtų stabiliai ir be sutrikimų.

Visi sistemos vartotojai (universiteto studentai ir personalas) gali peržiūrėti savo sąskaitas per internetą, naudodami interneto naršyklę, iš bet kurios pasaulio vietos. Vartotojai gali paržiūrėti pinigų pervedimus į savo sąskaitas, peržiūrėti atliktų apmokėjimų sąskaitas, peržiūrėti neapmokėtas sąskaitas ir jas apmokėti, jai sąskaitoje yra pakankamai pinigų. Jai vartotojas sugalvoja apmokėti sąskaitą, jis pasirenka norimą sąskaitą, paspaudžia apmokėjimo mygtuką, įveda turimą apsaugos kodą, ir patvirtina apmokėjimą. Toliau universiteto mokėjimo sistema visą reikiamą kliento ir sąskaitos informaciją nusiunčia į serverį, serveris patikrina ar vartotojas tikras (pagal gautą informaciją ir apsaugos kodą), jai klientas netikras, tai apmokėjimas nevykdoma ir apie tai informuojama, jai vartotojas tikras, tada yra patikrinama ar yra pakankamai pinigų vartotojo sąskaitoje, jai yra pakankamai pinigų tai įvykdo apmokėjimą ir apie tai informuoja vartotoją, jai nėra – tai informuoja vartotoją, kad apmokėjimas negalimas. Taip pat vartotojas gali peržiūrėti išsamią sąskaitų istoriją, bei savo sąskaitos balansą. Visą atėjusi naujų sąskaitų, įmokų į savo sąskaitą ir apmokėjimų

informacija yra išsiunčiama į kliento nurodytą elektroninį paštą. Todėl sistemos vartotojas pastarąją informaciją gali peržiūrėti ir neprisijungęs prie apmokėjimų sistemos, tiesiog pasinaudodamas savo elektroniniu paštu.

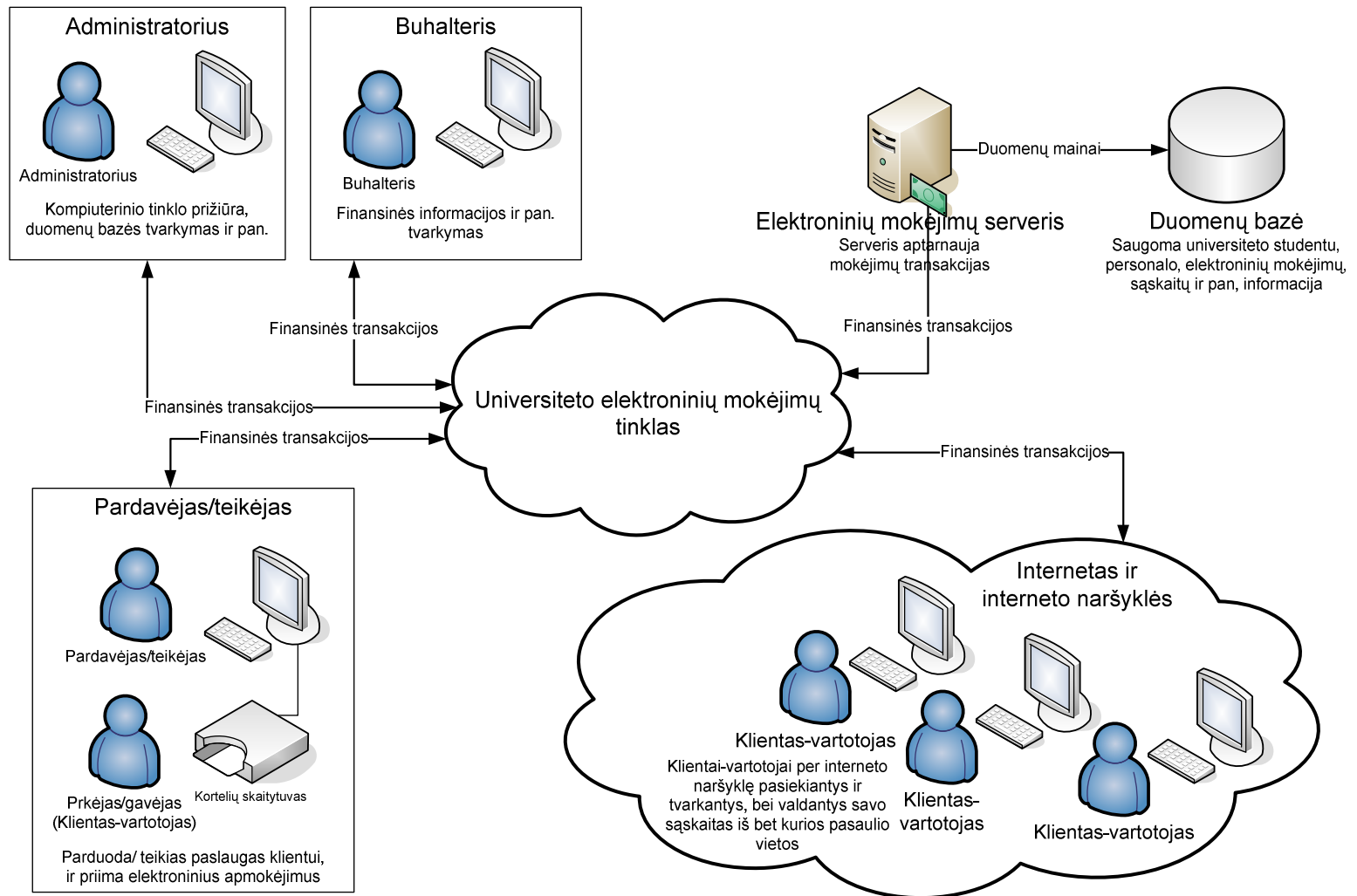
Serveris vykdo duomenų mainus su duomenų baze, registruoja ir tvarko informaciją duomenų bazėje, tvarko sistemos vartotojų autentiškumo tikrinimo procedūrą, tikrina sąskaitas ar jose yra pakankamai pinigų, jai yra pakankamai pinigų, tai juos perveda į pardavėjų/teikėjų sąskaitas (esant apmokėjimo poreikiui), organizuoja finansinės informacijos pateikimo, bei peržiūros operacijas, išsiuntinėja pranešimus ir patvirtinimus, į vartotojų elektroninius paštus, ir pan.

Visą finansinių transakcijų informacija yra siunčiama panaudojant SSL saugumą užtikrinantį protokolą, ir saugiais tiesioginio sujungimo TCP/IP kanalais. Visą finansinių transakcijų informacija, vykstanti vidiniame universiteto kompiuterių tinkle „keliauja“ link serverio ir atgal per vidinį universiteto kompiuterių tinklą (intranetą), jį galime tiesiog pavadinti „Universiteto elektroninių mokėjimų tinklu“, o informacijos mainai su išore vykdomi per globalų kompiuterių tinklą (internetą). Visi duomenys yra saugomi duomenų bazėje, o po to tik apdorojami, tokiu būdu yra apsisaugoma nuo svarbios informacijos praradimo sutrikus sistemos darbui.

Sistemą sudarančios sąsajos: administratoriaus, buhalterio, pardavėjo/teikėjo ir kliento-vartotojo (interneto naršyklės) sąsajos

Projektuojamos „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos koncepcinė schema (diagrama) (Paveikslas 3.14-15.) pateikto žemiau esančiame paveiksle.

„Elektroninių mokėjimų universitetinėje programinėje įrangoje“ sistemos išdėstymo vaizdo koncepcinė schema



Paveikslas 3.2-1. „Elektroninių mokėjimų universitetinėje aplinkoje programinėje įrangoje“ sistemos išdėstymo vaizdo koncepcinė diagrama.

3.2.4. Vartotojai

3.2.4.1. Vartotojai

Sudarytas sąrašas vartotojų, kurie betarpiškai naudosis sistema - „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“.

Lentelė 3.2-1. Vartotojas buhalteris.

Vartotojo kategorija:	Buhalteris
Vartotojo sprendžiami uždaviniai:	Įmokų priėmimas Įmokėtų sumų patvirtinimas Įmokėtos sumos pervedimas į kliento-vartotojo sąskaitą Sąskaitų rengimas Sąskaitų paskirstymas Sąskaitų tvirtinimas Klientų-vartotojo aptarnavimas Finansinės informacijos tvarkymas Finansinių ataskaitų formavimas
Patirtis dalykinėje srityje:	Patyręs
Patirtis informacinėse technologijose:	Patyręs
Papildomos vartotojo charakteristikos:	Finansinio pobūdžio informacijos tvarkymo išmanymas
Apmokymo poreikis:	Reikia
Amžiaus grupė:	30-80

Lentelė 3.2-2. Vartotojas administratorius

Vartotojo kategorija:	Administratorius
Vartotojo sprendžiami uždaviniai:	Elektroninių mokėjimų kompiuterinio tinklo priežiūra Elektroninių mokėjimų programinės įrangos priežiūra Duomenų bazės priežiūra Sistemos stabilumo užtikrinimas
Patirtis dalykinėje srityje:	Patyręs

Patirtis informacinėse technologijose:	Patyręs
Papildomos vartotojo charakteristikos:	Informacinių technologijų išmanymas
Apmokymo poreikis:	Nėra
Amžiaus grupė:	30-80

Lentelė 3.2-3. Vartotojas pardavėjas/teikėjas.

Vartotojo kategorija:	Pardavėjas/Teikėjas
Vartotojo sprendžiami uždaviniai:	Identifikavimas Sąskaitos parinkimas Duomenų tikrumo patikrinimas Apmokėjimo priėmimas Kliento, pardavėjo sąskaitos informacijos siuntimas į serverį Kliento-vartotojo apmokėjimo patvirtinimas Savo sąskaitų peržiūrėjimas Sąskaitų apmokėjimas Piniginiai pervedimai Apmokėjimo patvirtinimas
Patirtis dalykinėje srityje:	Naujokas, patyręs
Patirtis informacinėse technologijose:	Naujokas, patyręs
Papildomos vartotojo charakteristikos:	Finansinio pobūdžio informacijos tvarkymo išmanymas
Apmokymo poreikis:	Reikia
Amžiaus grupė:	18-80

Lentelė 3.2-4. Vartotojas klientas-vartotojas.

Vartotojo kategorija:	Klientas-vartotojas (t.y.. pirkėjas/gavėjas, klientas ir t.t., sistema klientams skirta dalimi besinaudojanti žmogus)
Vartotojo sprendžiami uždaviniai:	Identifikavimas Sąskaitos parinkimas Savo sąskaitų peržiūrėjimas Sąskaitų apmokėjimas Piniginiai pervedimai Apmokėjimo/pervedimo patvirtinimas
Patirtis dalykinėje srityje:	Naujokas, patyręs
Patirtis informacinėse technologijose:	Naujokas, patyręs

Papildomos vartotojo charakteristikos:	-
Apmokymo poreikis:	Reikia
Amžiaus grupė:	18-80

3.2.4.2. Vartotojo prioritetai

Vartotojų grupėms yra suteikiami prioritetai, nes tokiu atveju galima efektyviai nustatyti, kurių vartotojų teikiama informacija yra svarbiausi ir į kurią reikia betarpiškai atsižvelgti, o kurių nėra itin svarbi.

Lentelė 3.2-5. Vartotojo prioritetai.

Vartotojų kategorijos	Prioritetai
Buhalteris	Svarbus vartotojas
Administratorius	Svarbus vartotojas
Pardavėjas	Svarbus vartotojas
Vartotojas	Antraeilis vartotojas

Pastaba:

Svarbus vartotojas – jo nuomonė ir pageidavimai svarbiausi.

Antraeilis vartotojas – jei kyla konfliktas su pirmąja grupe, tai jų nuomonė turi didesnę įtaką.

3.2.4.3. Vartotojo charakteristikos

Apžvelgus analogiškas sistemas matėme kad jomis naudojasi dažniausiai studentai, jie turi galimybę ir gali susirasti žmogų kuris galės matyti jų sąskaitas ir apmokėti. Jas gali apmokėti taip pat ir šeimos nariai. Galima daryti prielaidą kad sistema naudosis pagrinde studentai. Nors kadangi sistema yra kuriama universitetinei aplinkai bus naudotojų ir dėstytojų. Apskritai visi vartotojai universitetinėje aplinkoje, kaip studentai ir dėstytojai turi pakakama žinių bagažą kad galėtu naudotis šita mūsų kuriama sistema. Pakankamai, susiveda į tokį konkretizavimą, kad vartotojas turētu mokėti pasijungti kompiuterį, jame mokėti pasileisti internetinė naršyklę, prisijungti prie internetinio puslapio ir prieiti prie savo duomenų ir žinoma atlikti pervedimus, apmokėjimus. Tuos dalykus žino kiekvienas žmogus nors kiek besinaudojantis kompiuteriu. Taip yra iš vienos pusės, tačiau norėdami pritraukti ir

tuos kurie neturi, galbūt tiek reikalingų žinių, mes stengsimės sukurti sistema tokia, kad ji būtų lengvai suprantama ir paprasta naudojime. Tada vartotojai turėtų turėti noro, poreikio atsidaryti prisijungti prie sistemos, o jau ten pajus paprastumą naudojime.

Taigi šita sistema turėtų pasirinkti tas žmonių ratas, kurie nori greitai ir saugiai atlikti apmokėjimus, savo sąskaitų peržiūrą ir t.t. Ir suprantame kad tai būtų žmones taupantys savo laiką ir pinigus.

3.2.4.4. Vartotojo problemos

Tokios problemos kaip greitis, patikimumas, saugumas, patogumas; problemos kurias turėtų išspręsti sistema. Realus pavyzdys, situacija iš gyvenimo, atsiskaitymai atliekami grynais, kai mokėtojas turi suskaičiuoti reikiamą pinigų sumą, o po to pardavėjas vėl ją paskaičiuoti ir dar jei reikia suskaičiuoti grąžą, tai užtrunka labai ilgai ir gaištama bereikalingai daug laiko. Dar blogesnis atvejis, kai apmokėjimui už paslaugas ir pan. išduodamas kvitas, kurį reikia kažkur apmokėti, o po to vėl grąžinti, čia yra apkraunami ir klientai ir darbuotojai, nes mokėtojas sugaišta daug laiko kol apmoka, o darbuotojams reikia suregistruoti, kas ir kada sumokėjo ar nesumokėjo. Tai labai nepatogu, todėl kuriama sistema apmokėjimą leis atlikti vos per kelias sekundes, o ne per kelias valandas ir daugiau, kaip t.y.. dabar. O be to vartotojas, bet kada galės gauti išsamią savo finansinę informaciją, kurios dabar negali gauti, nes paprasčiausiai niekas individualiai neranka duomenų, kiek kur ir koks asmuo išleido, nebent pats asmuo renka tokią savo informaciją. Be to sistema, bet kuriuo laikų rodys realią universiteto aplinkoje vykstančių finansinių operacijų apimtį (ten kur ji bus įdiegta), ne taip kaip dabar, kai visa finansinių operacijų informacija gaunama dienos ar mėnesio pabaigoje, po to kai atliekami „dideli“ ir labai varginantys skaičiavimai.

Sukurta programinė įranga leis pasinaudojant studento ar darbuotojo pažymėjimu atlikti apmokėjimą už paslaugas (tiesiog pinigai bus pervedami iš vienos sąskaitos į kitą), be to bus kur kas saugiau nes nereikės turėti grynųjų pinigų, kurios gali kas nors pavogti. Taip pat bus galima peržiūrėti savo sąskaitos likutį, atlikti pervedimus. Visą finansinių transakcijų informacija yra siunčiama panaudojant SSL saugumą užtikrinantį protokolą, ir saugiais tiesioginio sujungimo TCP/IP kanalais. Duomenys yra saugomi duomenų bazėje, o po to tik apdorojami, tokiu būdu yra apsaugoma nuo svarbios informacijos praradimo sutrikus sistemos darbui.

3.2.4.5. Vartotojo tikslai

„Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos programinė įrangos prototipas. Šita sukurta apmokėjimų programinė įranga, galės būti panaudota atlikti įvairioms finansinėms operacijoms universitetinėje aplinkoje, kaip pavyzdį būtų galima paminėti, tokias kuriamos programinės įrangos galimybes: priimti iš studentų ar darbuotojų apmokėjimus už: kopijavimo, spausdinimo, maitinimo, apsipirkinėjimą knygyne ir pan. paslaugas. Sukurta programinė įranga leis pasinaudojant studento ar darbuotojo pažymėjimu atlikti apmokėjimą už paslaugas. Peržiūrėti savo sąskaitos likutį, atlikti pervedimus elektroninio apmokėjimo sistemos teikiamomis galimybėmis. Taigi šiame darbe bus koncentruojamasi daugiau ties elektroninių apmokėjimų atsiskaitymo galimybėmis bei finansinės informacijos tvarkymo funkcijomis ir galimybėmis, t.y.. bus koncentruojami ties kuriamos sistemos branduoliu, nei ties kitomis kuriamos sistemos teikiamomis paslaugomis.

Naudojant sukurta programinę įrangą galima bus ją panaudoti ir kituose panašiuose projektuose, kuriose bus poreikis realizuoti programinę įrangą, susijusią su elektroniniais mokėjimais. Panaudojant sukurta programinę rangą (sistemą), bus padidintas elektroninių mokėjimų universitetinėje aplinkoje efektyvumas ir našumas. Manytume, kiekvienas vartotojas pradėjęs ja naudotis įvertins tokius sistemos privalumus.

3.3. Architektūros pateikimas

3.3.1. Architektūros pateikimas

Pateikiant architektūrinį „Elektroninių mokėjimų universitetinėje aplinkoje programina įranga“ sistemos vaizdą, naudojami skirtingi architektūriniai vaizdai, kurie parodo skirtingus sistemos architektūros aspektus. Architektūros pateikimo tikslas yra surinkti ir pateikti svarbius architektūrinius sprendimus, kurie buvo atlikti tyrinėjant ir projektuojant sistemą, ir kuriuos galima atlikti kuriamoje sistemoje.

„Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos architektūra pateikiama keliais vaizdais: panaudojimo atvejų (PA), statiniu, dinaminiu, išdėstymo ir duomenų. Šie vaizdai yra pateikiami kaip NATIONAL Rosią modeliai naudojant unifikuota modeliavimo kalbą (UML). Sistemos architektūra pateikiama remiantis RUP (NATIONAL Unified Process) rekomendacijomis. Sistemos specifikacija pateikta vaizdais, kuriems įgyvendinti reikia UML diagramų.

Lentelė 3.3-1. Architektūros pateikimo vaizdai ir modeliai.

Vaizdai	Modeliai	Aprašas
Panaudojimo atvejų vaizdas	Panaudojimo atvejų diagrama	Pateikiami esminiai panaudojimo atvejai arba scenarijai.
Sistemos statinis vaizdas	Sistemos skaidymo į paketus diagramos Klasių diagramos	Pateikiama sistemos išskaidymas į paketus kiekvienam paketui pateikiamas jo trumpas aprašymas ir klasių diagramos.
Sistemos dinaminis vaizdas	Veiklos diagramos Bendradarbiavimo diagramos	Pateikiamos sąveikos parinkta veiklos, bei sąveikai atvaizduoti pasirinkta bendradarbiavimo diagramos.
Išdėstymo vaizdas	Koncepcinė diagrama Išdėstymo diagrama	Aprašoma techninės įrangos, kurioje sistema bus išdėstyta ir veiks, konfigūracija bei sistemos komponentai.
Duomenų vaizdas	Duomenų bazės modelio	Pateikiamas duomenų bazės

	diagrama Duomenų bazės schema	modelis ir schema.
--	----------------------------------	--------------------

3.3.2. Architektūros tikslai ir apribojimai

Apžvelgiami projektuojamos „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos architektūros tikslai ir apribojimai. Šiuo atveju pagrindinis tikslas būtų panaudoti tokią kuriamos sistemos architektūrą, kuri leistų tinkamai atlikti elektroninius mokėjimus universitetinėje aplinkoje. Kadangi tinkama elektroninių mokėjimų realizacija yra pagrindinis kuriamos sistemos aspektas.

Projektuojamos „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos architektūros esminiai tikslai bei juos įtakojantys esminiai reikalavimai:

- Sistema bus pateikta kaip uždaro kodo, nekomercinė programinė įranga.
- Sistemą bus galima naudoti mokymo tikslas, tik gavus atitinkamų asmenų leidimą.
- Sistemą turi būti galima panaudoti plėtojant šį projektą ar kuriant kitus panašius projektus ir panašios srities.
- Sistemos turi būti suprojektuota taip, kad ją galima būtų lengva išplėsti ar prijungi naujus modulius.
- Sistema turi turėti galimybę pasiimti duomenis iš KTU akademinės informacinės sistemos, taip pat turi būti atspari tos sistemos sąsajos pasikeitimams.
- Tiri turi galimybę sinchronizuoti duomenis tarp sistemos duomenų bazės ir KTU akademinės informacinės sistemos.
- Sistema neturi leisti neautorizuotiems vartotojams prie jos prisijunkti.
- Sistema bus realizuota kliento-serverio modelyje. Sistemos serverinė dalis bus realizuota serveryje, o klientinė – veiks internetinėje naršyklėje (pvz.: Internet Explorer, Mozilla, Opera).
- Sistemoje bus panaudota Microsoft .NET Framework, Microsoft ASP.NET ir Microsoft C#.NET technologijos.
- Sistemos duomenų bazei bus panaudota Microsoft SQL Server duomenų bazių valdymo sistema.
- Sistema turi turėti atskirą buhalterio posistemį.
- Sistema turi turėti atskira atskirą pardavėjo-teikėjo posistemį.
- Sistema turi turėti atskirą kliento-vartotojo posistemį.

- Sistema turi užtikrinti tinkamą transakcijų apdorojimą.
- Sistema turi būti patogi vartotojui.
- Sistemoje turi būti tinkamai užtikrintas duomenų ir informacijos saugumas.
- Sudarant sistemos architektūrą, turi būti atsižvelgta į būtinas programos vykdymo charakteristikas, apibrėžtas reikalavimų specifikacijoje, kur yra išsamiai aprašyti visi sistemos reikalavimai.
- Išsamiai sistemos reikalavimai aprašyti skyriuose: *3.11. Funkciniai reikalavimai*, *3.12. Reikalavimai duomenims*, *3.13. Nefunkciniai reikalavimai*.

3.4. Priimti techniniai sprendimai

3.4.1. Projekto įgyvendinimo planai ir kokybės vertinimas

3.4.1.1. Įgyvendinimas

Projektuojama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ programinė įranga būtų realizuojama panaudojant Microsoft .NET platformą. Ir tiesiogiai leidžiančią pilnai pasinaudoti šios platformos galimybėmis Microsoft Visual Studio 2005.NET paketu. Nes čia yra suteikiama galimybė panaudoti, ne tik C/C++ ar JAVA programavimo kalbas, bet ir tokias pažangias programavimo kalbas, kaip Visual C#.NET, Visual Basic.NET, ASP.NET, Visual C++.NET, Visual J#.NET programavimo kalbas. Tai leidžia kurti itin greitai ir patikimai kokybišką programinę įrangą, kurioje bus įgyvendinti visi lūkesčiai, ties sistemos patikimumu, greitumu ar dizainu, ir pan. Sistemos realizacijai konkrečiai pasirenkamos Microsoft Visual C#.NET ir Microsoft Visual ASP.NET programavimo kalbos.

Microsoft Visual Studio 2005. NET teikiamos galimybės:

- Tobulinti ir derinti daugiapakopes serverio programas unifikuotoje programavimo aplinkoje.
- Kurti sprendimus, skirtus „SQL Server 2005“, naudojant integruotą vaizdinį duomenų bazės modelį ir pranešimo įrankius.
- Užtikrinti programinės įrangos saugumo galimybes.
- Kurti kliento/serverio programas, naudojant voratinklio paslaugas ir integruotus projektavimo įrankius, kad būtų galimybė turėti prieigą prie nuotolinių duomenų.

Sistemos duomenų bazės įgyvendinimui yra parinktas Microsoft SQL Server 2005 duomenų bazių valdymo sistema. Kadangi Microsoft SQL Server 2005 yra būsimųjų kartų sprendimas, kaip valdyti ir analizuoti duomenis, didelį duomenų ir analitinių programų saugą, keičiamumą ir pasiekiamumą, be to, padedantis jas kurti, programuoti ir valdyti.

Microsoft SQL Server 2005 suteikia galimybę valdyti integruotus duomenis ir priimti analizės sprendimą, kurie, bet kada gali padėti:

- Kurti, diegti ir valdyti programas, kurios yra saugesnės, keičiamos ir patikimos.

- Didinti informacinių technologijų produktyvumą, supaprastinant duomenų bazės programų kūrimą, diegimą ir valdymą.
- Duomenis naudoti keliose platformose, programose ar įrenginiuose, kad būtų paprasčiau prijungti vidines ir išorines sistemas.
- Valdyti informaciją nedarant žalos veikimui, pasiekiamumui, keičiamumui bei saugai

Microsoft SQL Server 2005 apribojimai:

- Neribotas procesorių skaičius
- RAM atmintis priklauso nuo operacinės sistemos galimybių
- 64bitų palaikymas
- Neribotas duomenų bazės dydis
- Didelių duomenų bazių palaikymas
- Duomenų atkūrimas
- Itin lengvas valdymas
- Geras auditas, autentifikavimas ir autorizacija
- Duomenų šifravimas ir rakto valdymas

Aptarnavimo serveris Microsoft Windows Server 2003, parinktas todėl, kad jis specialiai sukurta taip, kad veiktų kartu ir tolydžiai su kitais duomenimis ir programomis visoje naudojamose IT aplinkoje, taip leisdamas sumažinti einamųjų operacijų sąnaudas, užtikrindama itin patikimą ir saugią IT infrastruktūrą Microsoft Windows Server 2003 tikrai padeda pasiekti daugiau, naudojant mažiau išteklių.

Microsoft Windows Server 2003 yra patikimiausia ir saugiausia serverio platforma. Ši serverio platforma užtikrina vienijantį bendrųjų tarnybų sluoksnį IT, programų ir informacijos darbuotojų darbo infrastruktūroms. Microsoft Windows Server 2003 privalumai:

- Bendrasis programų kūrimo modelis
- Bendrasis katalogų ir saugos modelį
- Bendrosios duomenų tarnybos
- Integruotas duomenų perkėlimą į talpyklą
- Integruotas paskirstytų operacijų valdymas
- Integruota diagnostika
- Integruotos valdymo tarnybos
- Integruotos daugialypės terpės ir bendradarbiavimo tarnybos

Klientinė programos (administratoriaus, buhalterio, pardavėjų/paslaugų teikėjų) realizuojamos kaip Windows OS programos, nes:

- Nes daugiausia technologijų sukurtą būtent šiai OS.
- Patogi ir turtinga vartotojo sąsaja
- Greitas programų veikimas
- Itin didelės programos funkcionalumo ir kokybės galimybės
- Lengvas supratimas ir vartojimas
- Didelis saugumas ir patikimumas
- Stabilus darbas.

Klientinė programa skirta eiliniams vartotojams, bus realizuota per internetą ir interneto naršyklę. Kad vartotojai galėtų, bet kada ir iš bet kur peržiūrėti savo finansinę informaciją. Tačiau vartotojų naudojamos naršyklė turi palaikyti ASP.NET technologiją.

Duomenų mainai su duomenų baze ir duomenų bazėje yra vykdomi, kai yra gaunamas, iš kurio nors kliento atitinkamas nurodymas, ar vykdomos tam tikros nustatytos užduotys.

3.4.1.2. Reikalavimų inžinerijos metodika

Projektuojamos „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ programinės įrangos reikalavimų surinkimui ir analizei bus taikoma populiari ir itin plačiai taikoma Volere šablono metodika ir Microsoft Office paketo priemonės. Volere šablonai yra tuo geri, kad jie yra tarsi reikalavimų „konteineriai“. Todėl analizuojant reikalavimų aprašus ir suskirstant juos pagal tipus, reikalavimų atpažinimas (užfiksavimas) ir įgijimas tampa daug paprastesnis. Taikant pastarąją metodiką pagrindinis dėmesys yra skiriamas sistemos apribojimams (nors tai nėra tiesioginiai reikalavimai, tačiau turi didelę įtaką kuriamai sistemai bei jos teikiamiems veikimo rezultatams), funkcinius ir nefunkcinius reikalavimus, projekto išeią.

Reikalavimų išgavimas yra svarbiausia veikla, skirta išsiaiškinti informacijos sistemos vartotojo poreikius, juos vienareikšmiškai ir tiksliai užregistruoti sudarant vartotojo poreikių specifikaciją, kuri toliau yra naudojama kuriant informacinę sistemą. Nes tik nuo tinkamo užsakovo ir vartotojų reikalavimų išgavimo ir įvertinimo priklausys kuriamos sistemos sėkmingas įgyvendinimas ir eksploatavimas.

3.4.1.3. Projektavimo metodika

Projektuojamos „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ programinės įrangos architektūros projektavimas bus atliekamas pasinaudojant MagicDraw ir Microsoft Office Visio priemonėmis (kadangi t.y.. pilnaverčiai, išbandyti produktai, puikiai tinkantys programinės įrangos projektavimui), kurios padės sukurti realizuojamo projekto dokumentaciją remiantis Unified Modeling Language (UML) notacija. Tokiu būdu yra teikiama galimybė pakartotinai panaudoti projekte surastus problemų sprendimus, tęsiant ir plėtojant šį projektą, ar rengiant ir plėtojant susijusius projektus.

„Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ bus projektuojama remiantis evoliucinio metodo principu. Nes taikant šį projektavimo metodą su užsakovu yra dirbama nuo pradinės specifikacijos iki galutinės programos versijos. To pasėkoje visas projektas tampa pilnai suderintas su užsakovu ir nebekyla jokių nesklandumų, ir realizuotas projektas tenkina visus užsakovo reikalavimus.

3.4.1.4. Išplečiamumas

Kadangi informacinės technologijos nestovi vietoje ir nuolat tobulėja. Tad nuolat atsiranda naujos technologijos ar ištobulinamos jau esamos, todėl tobulėjant technologijoms žmonių poreikiai ir keliami reikalavimai sistemoms taip pat auga, juk nieks nenori naudoti atsilikusios nuo pažangos sistemos, kad ir kokia ji tobula būtų. Todėl ir kuriama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ negali atsilikti nuo informacinių technologijų pažangos. Todėl kuriant „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ programinę įrangą naudojami objektinio projektavimo ir programavimo principai, o be to „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ teiks programines sąsajas, kad ateityje būtų galima lengvai prijungti naujus modulius, išplečiančius sistemos ir jos programinės įrangos funkcionalumo galimybes, tai bus atliekam itin lengvai, nes tai jau buvo numatyta projektavimo stadijoje.

3.4.2. Projekto įgyvendinimo etapai

Siekiant, kad „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektas būtų atliktas kuo greičiau ir perspektyviau yra numatyti projekto darbų įgyvendinimo etapai su atlikimo terminais, kurie yra pateikti Lentelė 3.4-1. lentelėje.

Lentelė 3.4-1. Projekto įgyvendinimo etapai.

Eil. Nr.	Etapas	Trukmė, mėn.
1.	Projekto metodologijos ir technologijų analizė	1,5
2.	Paraiška	0,5
3.	Darbų planas	0,5
4.	Reikalavimų specifikacijos	3,5
5.	Programinės įrangos architektūros specifikacija	1
6.	Detali programinės įrangos architektūros specifikacija	1
7.	Programinės įrangos realizacija	2
8.	Testavimo planas - testavimas	1
9.	Vartotojo dokumentacija	1
10.	Programinės įrangos diegimas	0,5
11.	Pilnai užbaigtas projektas	12,5

Projekto įgyvendinimo etapų sudaromųjų elementų paaiškinimas:

1. Projekto metodologijos ir technologijų analizė - literatūros, interneto šaltinių studija, skirta pasirengti specifikavimui, projektavimui, susipažinti su projektuojamos sistemos taikymo sritimis, pasauliniais pasiekimais toje srityje.
2. Paraiška – projekto tikslų bei reikalavimų suderinimas tarp jo užsakovo, vadovo ir vykdytojų;
3. Darbų planas – projekto realizavimo darbų eigos, jų atlikimo terminų nustatymas.
4. Reikalavimų specifikacija – projekto pagrindinių apribojimų nustatymas, funkcinių ir nefunkcinių reikalavimų sudarymas, išeigos apskaičiavimas.
5. Programinės įrangos architektūros specifikacija – projekto architektūros sudarymas panaudojimo atvejų, procesų ir realizacijos aspektais per UML diagramas.
6. Detali programinės įrangos architektūros specifikacija – projekto detalios architektūros projektavimas.
7. Programinės įrangos realizacija – realizuojama projekte numatyta programinė įranga.

8. Testavimo planas - testavimas – projekto testavimo darbų eigos sudarymas, testavimo metodų ir kriterijų parinkimas, testavimas.
9. Vartotojo dokumentacija – išsamus naudojimosi sistema aprašymas.
10. Programinės įrangos diegimas – programinės įrangos įdiegimas ir sistemos parengimas naudojimui.
11. Pilnai užbaigtas projektas – gautinai užbaigtas projektas, paruošta visa reikiama programinė įranga ir dokumentacija, bei viskas atiduota užsakovui.

3.5. Numatoma eksploatacijos aplinka

Numatoma eksploatacijos aplinka, kurioje dirbs vartotojas planuojama kas bus universitetinėje aplinkoje. Kiekvienas vartotojas galės prisijungti iš kompiuterio per internetinį ryšį prie šitos sistemos. Todėl šitoje vietoje nesusidaro sunkumų, nes vartotojas dažniausiai pasirenka pats aplinka kurioje norėtų dirbti su sistema, (darbo vieta mobili) žinoma neatmetama tikimybe kad toji vartotojo pasirinkta aplinka neretai gali būti triukšminga. Tačiau kas liečia sistemos posistemių pvz. spausdinimo priemonė gali būti šiek tiek atokiau nuo vartotojo, nes kitaip reiktų prie kiekvieno kompiuterio, kiek jų bebūtų diegti po spausdintuvą. Žinoma kol kas tai yra sunkumas tačiau, manoma kad jis bus išspręstas. Pvz. kiekvienai kompiuterių klasės po vieną spausdintuvą. Daugiau sunkumų nenumatoma kadangi vartotojas dirbs pagrinde su sistema prie jos prisijungęs, nebent pats kompiuteris iš kurio jis prisijungęs yra problematiškas ir sunkiai veikiantis. Nors iš kitos puses vartotojas gali pasirinkti ir kitą kompiuteris, nes jis nėra griežtai pririštas prie vietos.

3.6. Sprendimų įtaka sistemos architektūrai

Apžvelgiami kuriamo „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos architektūros įtakojami kokybės faktoriai. Yra nagrinėjami tokie kokybės faktoriai kaip: išplečiamumas, pernešamumas, patikimumas.

Išplečiamumas – kadangi projektuojant yra naudojama tokia sistemos architektūra, kuri leidžia be didelio vargo prijungti naujus modulius. Sistemos projektavimas remiasi komponentiniu kūrimu leidžia nesunkiai papildyti sistemą naujais komponentais, pakartotinai juos panaudoti kitose sistemose ar panašiuose projektuose.

Pernešamumas – sisteminių ir vartotojų duomenų perkėlimas, nesunkiai įgyvendinamas nes architektūra leidžia perkelti duomenys į kita sistema, su perkėlimu susiję tokie duomenys kaip:

Egzistuojančių sistemos vartotojų duomenys; duomenys susiję su prisijungimu, su naudojama sesija, su saugumu, egzistuojančios sąskaitos (neapmokėtos)

Sistemos konfiguracioniai duomenys; sistemos programiniai „griaučiai“ sistemos struktūra realizuojantis duomenys, tam kad būtų galima ją įdegti ir panaudoti pasinaudojus nauju aparatūriniu vienetu.

Patikimumas, saugumas - siekiant užtikrinti kuriamos sistemos saugumą, panaudoti Microsoft kompanijos sukurti programinės įrangos eksploatavimo saugumą užtikrinantys saugos bibliotekų paketai (*System.NET Security paketas, System.Security, System.Security.AccessControl, System.Security.Authentication, System.Security.Cryptography, System.Security.Cryptography.Pkcs, ir kt.*), kurie suteikia sistemai saugų ir patikimą egzistavimą ir funkcionavimą.

3.7. Projekto technologiniai apribojimai

3.7.1. Apribojimai sprendimui

Šiame skyriuje apžvelgsime išankstinius projekto „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos apribojimus kaip sistemos kūrimo eigai taip ir charakteristikoms.

Lentelė 3.7-1. Apribojimai sprendimui.

Išankstiniai apribojimai kuriamai sistemai	Apribojimo atsiradimo priežastys	Reikalavimo išpildymas
Galimybė dirbti su sistema pasinaudojus tokiomis naršyklėmis kaip: Internet Explorer(5.0, 6.0, 7.0), Firefox, Opera.	Vartotojams reikės prisijunginėti prie sistemos pasinaudojus tokiomis naršyklėmis	Pilnai
Sistema turi būti pritaikyta dirbti žmonėms kurie nėra praėję kompiuterinio raštingumo kursų	Norint pritraukti daugiau vartotojų	Pakankamai
Turi būti vartotojo prisijungimo sritis	Tam kad vartotojui nereiktu specialiai kažkur dar jungtis	Pilnai
Turi būti administratoriaus prisijungimo sritis	Kad administratorius galėtų administruoti	Pilnai
Sistema turi būti patalpinta nutolusiame serveryje	Tam kad prie jos galėtų prisijungti nutolę vartotojai	Pakankamai
Kiekvienam užsiregistravusiam vartotojui turi būti sukuriama vieta serveryje	Kažkur vartotojai turi saugoti savo duomenys	Pilnai
Kiekvienam prisijungusiam vartotojui turi būti sukuriama saugi sesija	Saugumui užtikrinti	Pilnai
Esant kokiam nors sistemos gedimui, turi būti išvedamas pranešimas, kur	Kad vartotojai būtų informuoti apie tai kas	Dalinai

kreiptis	vyksta su sistema	
Vartotojams prisiregistravusiems sistemoje, bet pamiršusiems savo prisijungimo vardą bei slaptažodį reikia užtikrinti kad jie jį saugiai gautu ir vėl galētu naudotis sistema	Kad tokioje situacijoje esantys vartotojai galētu vėl naudotis sistema	Dalinai
Kuriama sistema turi atlikti finansines operacijas universitetinėje aplinkoje	Kadangi ji kuriama pagrinde universitetiniam aptarnavimui	Pakankamai
Sukurta programine įranga turi būti prieinama(tam tikriems asmenims) kad galima būtų ją panaudoti tolimesniam vykdymui	Norint padaryti ją toliau vystomą	Pakankamai

Pastaba: Reikalavimų išpildymų paaiškinimai:

- **Pilnai** – sistema pilnai atitinka įvardinta kriterijų, viskas kuo jame buvo reikalaujama yra išpildyta ir funkcionuoja
- **Pakankamai** – įvardinti reikalavimai yra išpildyti, kai kuriuos dalys paliktos nebaigtos, nes tam reikia laiko kad jas ištestuoti
- **Dalinai** – reikalavimai išpildyti tiek kiek įmanoma, šitoje vietoje jų išpildymas priklauso ir nuo vartotojo veiklos

3.7.2. Diegimo aplinka

Apžvelgiame projektuojamos „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos diegimo aplinkos apribojimus.

Technologinė, programine, fizinė aplinkos

- *Duomenų bazė.* Bus diegiama duomenų bazės valdymo MS SQL Server serveris,. Kadangi „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistema bus pritaikyta būtent šio tipo duomenų bazės valdymo serveriui. Nes, duomenų bazės valdymo MS SQL Server serveris puikiai užtikrina didelį duomenų patikimumą, saugumą, greitą duomenų apdorojimą, ir pan., kas

eksploatuojant „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemą yra labai svarbu.

- *Operacinė sistema.* Kuriama sistema „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ pritaikyta MS Windows operacinei sistemai, todėl serveryje ir kompiuteriuose turėtų būti naudojama MS Windows operacinė sistema.
- *Programinėje įrangoje naudojamos technologijos.* „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistema bus sukurta naudojant pažangiausią šių dienų programinės įrangos kūrimo technologiją MS .NET Framework. Diegimo aplinkoje (serveryje, buhalterijų ir pardavėjų/teikėjų kompiuteriuose) turėtų bus diegiama MS .NET Framework technologijos naujausia versija su visais atnaujinimais, tam, kad būtų užtikrintas sėkmingas „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos darbas.
- *Internetinė naršyklė.* „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos vartotojų sąskaitų informaciją bus galima peržiūrėti ir per interneto naršyklę. Vartotojai turėtų turėti naršykles palaikančias ASP technologiją
- *Kompiuterinis tinklas.* Kompiuterinis tinklas Kauno Technologijos Universitete yra pakankamai greitas, tačiau siekiant kokybiškai įdiegti sistemą, ir kad ji galėtų saugiai ir patikimai veikti, gali tekti pertvarkyti. Nes patikimas sistemos veikimas ir geras saugumo užtikrinimas yra kritiškai svarbus. Pertvarkymas numatomas kai kurių komponentų Kauno Technologijos Universiteto kompiuteriniame tinkle.
- *Serveris.* Serveriui taikomi tokie reikalavimai, kaip kad turi būti pakankama aparatūrinė dalis kad aptarnauti universitetinės aplinkos vartotojus, manoma kad laukui bėgant reikės didesnio dar kad būtų orientuotas į didesnę vartotojų segmentą.
- Spausdintuvas
- Kortelių skaitytuvas

3.7.3. Bendradarbiaujančios sistemos

Kuriamos programinės įrangos sisteminė aplinka kurioje ji veiks, gyvuos yra numatoma universitetas. Sukurta programinė įranga leis pasinaudojant studento ar darbuotojo pažymėjimu atlikti apmokėjimą už paslaugas (tiesiog pinigai bus pervedami iš vienos sąskaitos į kitą), be to bus kur kas saugiau, nes nereikės turėti grynųjų pinigų, kurios gali kas nors pavogti. Taip pat bus galima peržiūrėti savo sąskaitos likutį, atlikti pervedimus „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos programinė įrangos teikiamomis galimybėmis.

Įmanomas būtų bendradarbiavimas su SEB Vilniaus bankas, Hansabankas, DnB Nord bankas, Snoras bankas, ir kitų bankų teikiamomis elektroninėmis paslaugomis. Bankų teikiamos galimybės yra geros, tačiau šiuo atveju nepakankamai geros, pagrindinė problema yra ta, kad bankų yra daug, o studentų ir darbuotojų universitete yra taip pat labai daug, todėl negalima varžyti pasirinkimo laisvės kokį banką reikia asmeniui pasirinkti. Be to, kiekvienas bankas naudoja savo individualia finansinių operacijų sistemą, todėl yra labai sunkų suderinti visus bankus. Be to, finansinės transakcijos tarp pardavėjo ir banko trunka labai ilgai. Atsiskaitymų informacija gaunama ne iškart, tik nustatytais terminais.

3.7.4. Komerciniai specializuoti programų paketai

Charakterizuojami specializuoti į kuriamą „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistema numatomi įtraukti programų paketai, kurie bus reikalingi kuriamos sistemos funkcionalumui užtikrinti yra tokie:

- *System.Security paketas* [52] – aprūpina realaus laiko saugumo struktūromis sistemą, įskaitant pagrindines klasių teises.
- *System.NET Security paketas* [51] - aprūpina tinklo srautų komunikaciją tarp nutolusių tinklo taškų.
- *System.Security.AccessControl paketas* [53] – aprūpina programos elementus kontroliniais leidimais, ir audituoja saugumą susijusių saugių objektų.
- *System.Security.Authentication paketas* [54] – suteikia enumeracijos aprašus saugumo ryšiams. Į šią enumeraciją įeina: [CipherAlgorithmType](#), [ExchangeAlgorithmType](#), [HashAlgorithmType](#), ir [SslProtocolType](#).
- *System.Security.Permissions paketas* [57] – aprašo klasių kontrolės teisių operacijoms ir resursų polisams.

- *System.Security.Cryptography paketas* [55] – suteikia kriptografijos servisu, įskaitant saugų duomenų kodavimą ir dekodavimą, ir daug kitų operacijų, tokių kaip hašingavimas, atsitiktinių skaičių generavimas, ir pranešimų autentifikacija ir t.t.
- *System.Security.Cryptography.Pkcs paketas* [56] – suteikia programuojamiems elementams Viešojo rakto kriptografijos standartą (PKCS), įskaitant metodus duomenų parašui, raktų apsisikeitimui, sertifikatų prašymus, viešojo rakto kriptografiją ir dekriptografiją, ir kitas saugumo funkcijas.
- *System.Security.Principal paketas* [59] – aprašo pagrindinius objektus, kurie atstovauja saugumo kontekstą pagal kodo vykdymą.
- *System.Security.RightsManagement paketas* [60] – suteikia klasių teisių valdymo sukuriant aplikacijos kontekstą tokį, kuris yra išsaugomas [*Package*](#) ar [*XpsDocument*](#).
- *System.Security.Policy paketas* [58] – apima kodo grupes, narystės (membership) sąlygas, ir požymius. Šie trys klasių tipai naudojami sukuriant taisykles pagal vykdomą saugumo polisą sistemoje. Akivaizdus klasės indėlis į saugumo polisą ir narystės (membership) sąlygas yra komutuojamas; kartu sukuriant poliso sąlygas ir suteikiant leidimus. Poliso ygiai ir kodo grupės yra struktūrinės polisų hierarchijoje. Kodo grupės inkapsuluotos taisyklių ir sutvarkytos polisų hierarchijos.

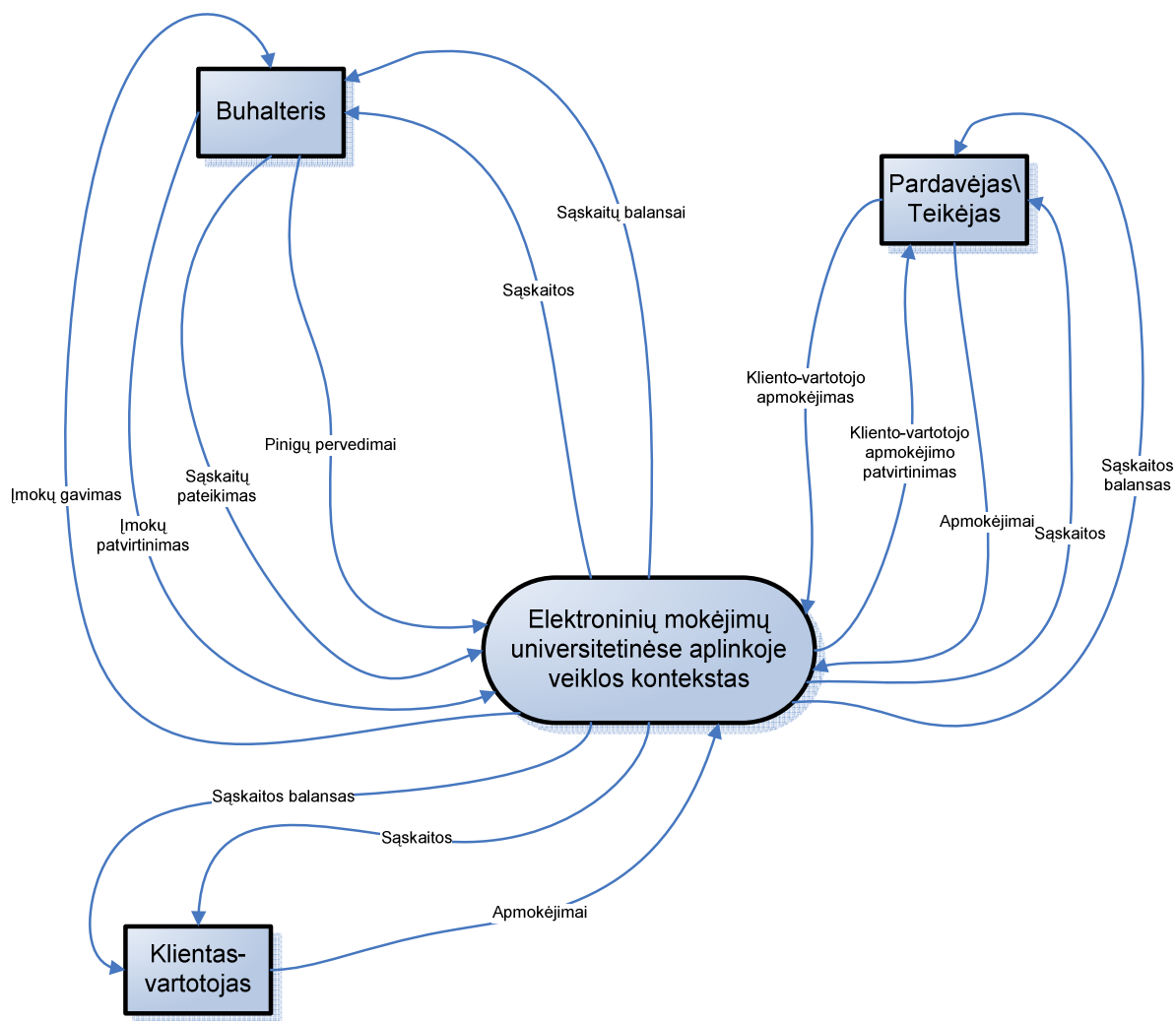
Kadangi sistemos funkcionalumo užtikrinimas yra labai svarbus aspektas, tam yra pasitelkiama daugiausiai dėmesio ir pasirūpinama naujausiais programų paketais užtikrinančiais saugumą.

3.8. Veiklos kontekstas

Nagrinėjama elektroninių mokėjimų universitetinėje aplinkoje veiklos sričiai apibrėžti naudojama "Konteksto diagrama". Šią sritį tenka iširti, kad būtų galima sukurti sistemą („Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“). Veiklos kontekstas apima plačiau, nei kuriamos sistemos atliekamos funkcijos. Kad geriau suprastume darbą (veiklą), kuriam turi talkinti sistema, ir galėtume sukurti tinkamos sistemos, kuri „įsirašys“ į aplinką. Veiklos kontekstas apibrėžia dominančią veiklą ir jos naudojamus bei formuojamus informacijos srautus. Veiklos „atsakomybė“ prasideda kai informacijos srautas įeina į sistemą ir baigiasi, kai rezultatinis srautas išeina iš sistemos. Išorinės esybės diagramoje modeliuoja kaimynines (gretimas) sistemas (tai aplinkinės sistemos arba žmonės).

Sąsaja tarp kaimyninių sistemų ir veiklos konteksto nusako, dėl kokių priežasčių kaimyninė sistema mus domina. Informacijos srautai tarp gretimų sistemų ir veiklos srities nusako, už ką yra atsakinga nagrinėjama veikla ir už ką - kaimyninės sistemos.

Žemiau esančiame paveiksle (Paveikslas 2.10-1.) yra pateiktas elektroninių mokėjimų universitetinėje aplinkoje sistemos veiklos kontekstas.



Paveikslas 3.8-1. Veiklos konteksto diagrama.

3.9. Veiklos padalinimas

Sudaromas veiklos įvykių sąrašas, kuris apima visus veiklos įvykius, už kuriuos yra atsakinga nagrinėjama elektroninių mokėjimų universitetinėje aplinkoje veikla. Veiklos įvykiai - tai vartotojo išskiriami veiksmai, atliekami veiklos metu. Reakcija (atsakymas) į kiekvieną įvykį atvaizduoja veiklos dalį, įeinančią į bendrą veiklą sudarančias funkcijas.

Įvykių sąrašą sudaro: 1) Įvykio pavadinimas; 2) Įeinantys ir išeinantys informacijos srautai, kurie "lydi" įvykį. Veiklos padalinimo paskirtis - identifikuoti veiklos "gabaliukus", kurių pagrindu būtų galima nustatyti reikalavimus. Veiklos įvykių pagrindu toliau galima remtis atliekant sistemos detalią analizę ir projektavimą. Veiklos įvykių sąrašas pateiktas Lentelė 3.9-1.

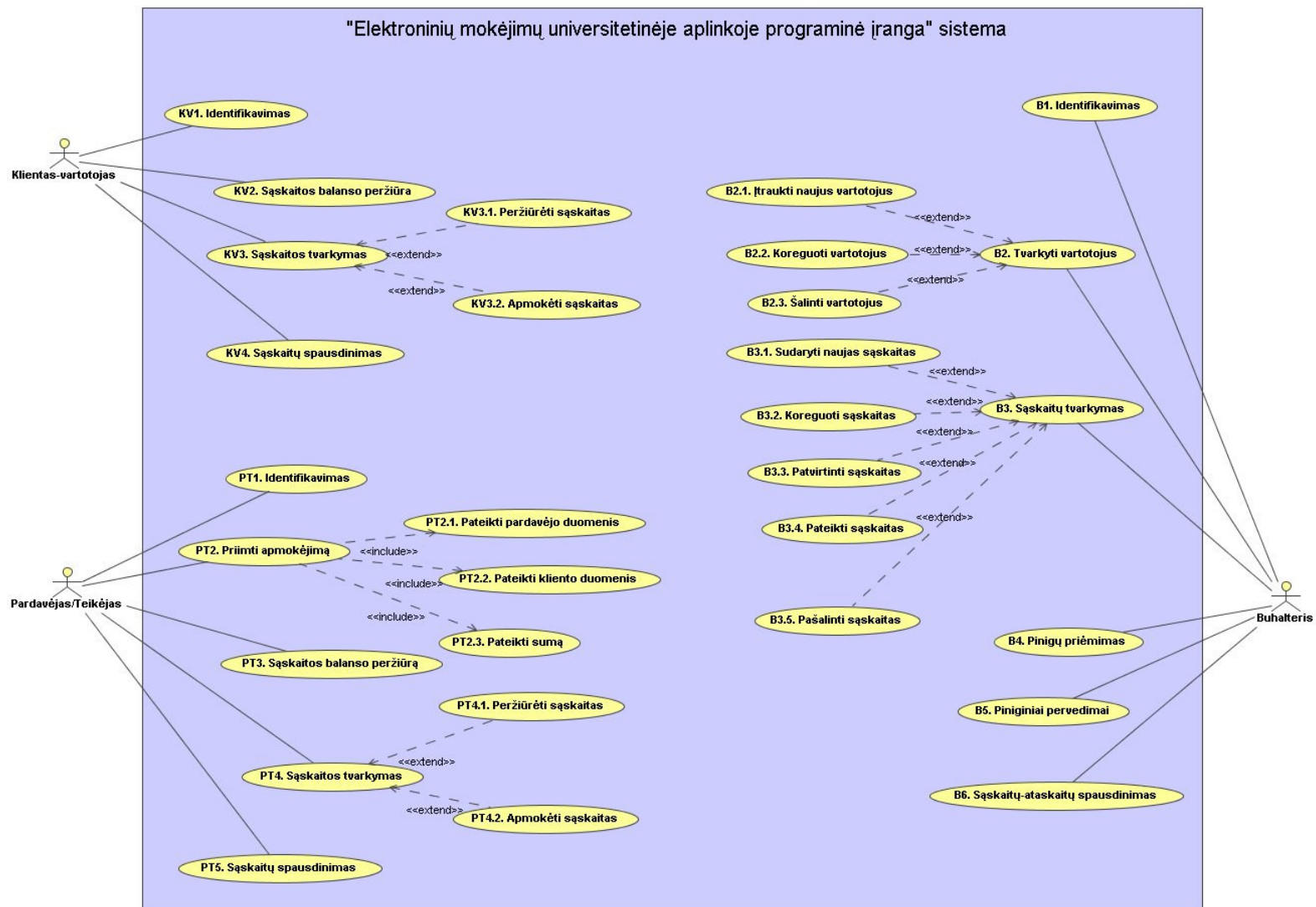
Lentelė 3.9-1. Veiklos įvykių sąrašas.

Eil. Nr.	Įvykio pavadinimas	Įeinantys(in)/išeinantys(out) informacijos srautai
1.	Buhalteris gauna įmokas	Įmokų gavimas (out)
2.	Buhalteris patvirtina įmokas	Įmokų patvirtinimas (in)
3.	Buhalteris pateikia sąskaitas	Sąskaitų pateikimas (in)
4.	Buhalteris perveda į gavėjų sąskaitas pinigus	Pinigų pervedimai (in)
5.	Buhalteris gauna sąskaitas	Sąskaitos (out)
6.	Buhalteris gauna sąskaitų balanso informaciją	Sąskaitų balansas (out)
7.	Pardavėjas\Teikėjas priima kliento-vartotojo apmokėjimą	Kliento-vartotojo apmokėjimas (in)
8.	Pardavėjas\Teikėjas gauna kliento-vartotojo apmokėjimo patvirtinimą	Kliento-vartotojo apmokėjimo patvirtinimas(out)
9.	Pardavėjas\Teikėjas gauna sąskaitos balanso informaciją	Sąskaitos balansas (out)
10.	Pardavėjas\Teikėjas gauna sąskaitas	Sąskaitos (out)
11.	Pardavėjas\Teikėjas apmoka sąskaitas	Apmokėjimas (in)
12.	Klientas-vartotojas gauna sąskaitos balanso informaciją	Sąskaitos balansas (out)
13.	Klientas-vartotojas gauna sąskaitas	Sąskaitos (out)
14.	Klientas-vartotojas apmoka sąskaitas	Apmokėjimas (in)

3.10. Panaudojimo atvejai

„Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos ir vartotojo ribas nusako panaudojimo atvejų diagrama (Paveikslas 3.10-1.). Šios sistemos panaudojimo atvejų diagrama sudaryta įvertinant kiekvieną išskirtą veiklos įvykį ir kuriamos sistemos indelį šio įvykio atžvilgiu.

Pastaba: Paveikslas 3.10-1. paveiksle pateiktas buhalteris kartu apima ir administratoriaus funkcijas, kadangi projektuojamoje sistemoje jie yra glaudžiai susiję, be to jų atliekamos funkcijos yra susijusios (funkcijos priklauso nuo suteiktų vartotojui teisių).



Paveikslas 3.10-1. Panaudojimo atvejų diagrama.

3.11. Funkciniai reikalavimai

Pateikiami „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos funkciniai reikalavimai, kurie nusako sistemos funkcionalumą. Apibrėžia, ką sistema turi daryti ir kokiais duomenimis manipuliuoti, kad atliktų numatytas funkcijas. Funkciniai reikalavimai atitinka sistemos vartotojų veiklos reikalavimus, kurie nusako kokius darbus galėtų atlikti „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojama sistema.

Funkciniai reikalavimai „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamai sistemai:

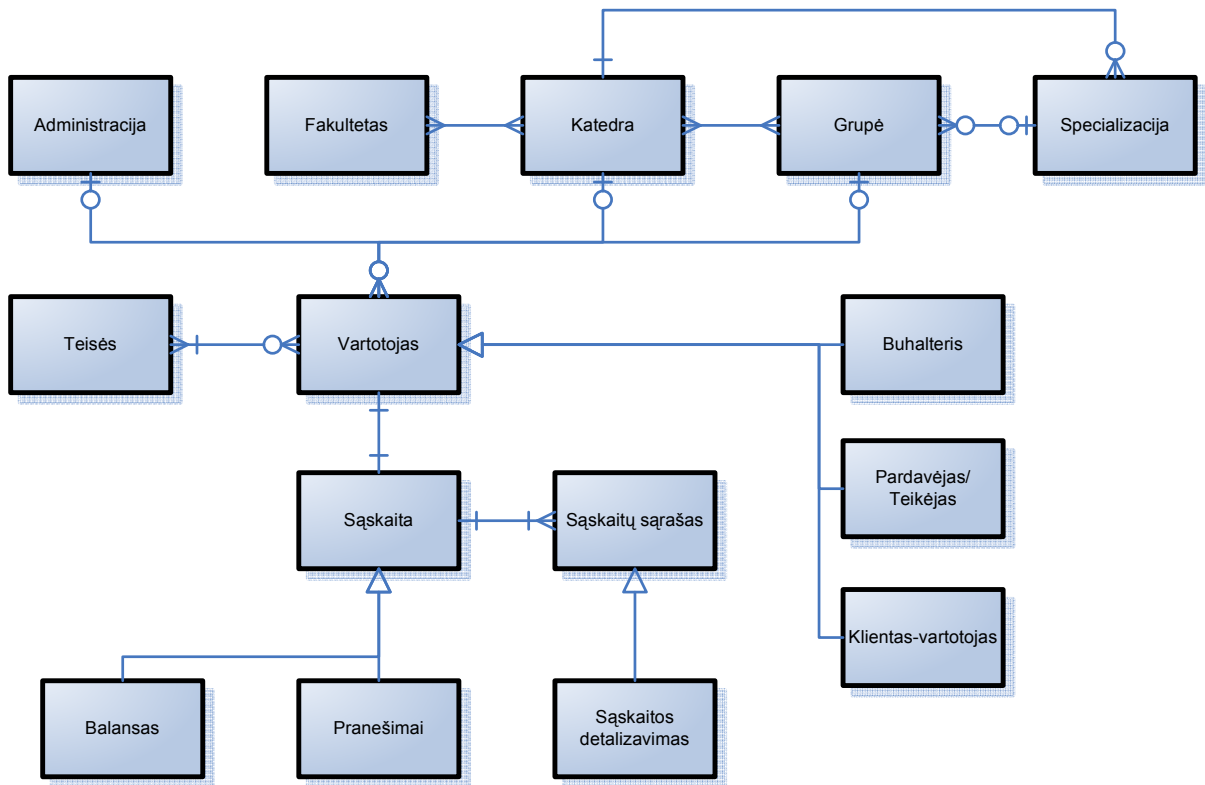
- Sistemoje turi būti galimybė tvarkyti vartotojų asmeninius (kontaktinius) duomenis.
- Sistemos vartotojų asmeninę (kontaktinę) informaciją gali tvarkyti tik buhalteris.
- Sistema turi leisti registruoti naujus vartotojus.
- Sistema turi leisti vartotojams priskirti teisas.
- Sistema turi leisti koreguoti vartotojų asmeninius (kontaktinius) duomenis.
- Sistema turi leisti pašalinti ja nebesinaudojančius vartotojus neturinčius neapmokėtų sąskaitų, ir pinigų sąskaitoje.
- Sistema neturi leisti pašalinti vartotojų, kurie turi neapmokėtų sąskaitų, ir/ar pinigų sąskaitoje.
- Daugumos vartotojų (Kauno Technologijos Universiteto studentų ir personalo) informacija turi būti susinchronizuota su Kauno Technologijos Universiteto akademinė informacine sistema.
- Sistema turi leisti buhalteriu vartotojų sąrašus filtruoti pagal įvairius parametrus.
- Sistema turi leisti buhalteriu rikiuoti vartotojų sąrašus.
- Sistema turi leisti buhalteriu atlikti vartotojų paiešką.
- Sistema turi rodyti buhalteriu detalią vartotojų informaciją.
- Visiems vartotojams turi būti priskirtos atitinkamos teisės naudojantis sistema.
- Sistemos pardavėjas/teikėjas turi tik pardavėjo/teikėjo teises.
- Sistemos klientas-vartotojas turi tik kliento-vartotojo teises.
- Sistemos buhalteris turi tik visas teises, t.y.. buhalterio teisę kuri apima visas teises..
- Visos sistemos funkcijos turi būti prieinamos su atitinkamomis vartotojų teisėmis.
- Visa sistemos informacija turi būti prieinamos su atitinkamomis vartotojų teisėmis.
- Sistema naudojami tik autorizuoti vartotojai.
- Prie sistemos gali prisijungti tik autorizuoti vartotojai.
- Vartotojo prisijungimo vardas turi atitikti jo identifikacinį numerį.

- Sistemos vartotojo slaptažodis turi būti netrumpesnis nei 6 simboliai.
- Vartotojo neteisingai bandymų prisijungti iš eilės skaičius, turi būti limituotas iki 10.
- Praėjus valandai po vartotojo blokavimo (kai jis neteisingai bandė prisijungti prie sistemos 10 kartų iš eilės), sistema turi atblokuoti vartotoją.
- Vartotojas turi turėti galimybę pasikeisti prisijungimo slaptažodį.
- Sistema turi leisti tvarkyti sąskaitas.
- Visas sistemos sąskaitas gali tvarkyti tik buhalteris.
- Sistema turi leisti sudaryti naujas sąskaitas.
- Sistema turi leisti koreguoti sąskaitas.
- Pateikiamos sąskaitos turi būti patvirtinamos.
- Visos gauto sąskaitos turi būti patvirtintos.
- Buhalteris turi turėti galimybę, pateikti sąskaitas vartotojams.
- Sistema turi leisti pašalinti sąskaitas.
- Sąskaitose turi būti pateikiama vartotojų kontaktinė informacija.
- Sąskaitose turi būti pateikiama pateikėjų kontaktinė informacija.
- Sąskaitose turi būti aiškiai įvardinti sudarymo data.
- Sąskaitose turi būti aiškiai įvardinti apmokėjimo pabaigos data.
- Sistema turi registruoti ar sąskaitos yra apmokėtos ar ne.
- Sistema turi leisti buhalteriu sąskaitas filtruoti pagal įvairius parametrus.
- Sistema turi leisti buhalteriu rikiuoti sąskaitas.
- Sistema turi leisti buhalteriu atlikti sąskaitų paiešką.
- Sistema turi rodyti buhalteriu detalią sąskaitų informaciją.
- Sistema turi leisti buhalteriu atspausdinti sąskaitas.
- Sistema turi leisti buhalteriu atspausdinti finansines ataskaitas.
- Sistema turi leisti buhalteriu atspausdinti bendrus finansinius rodiklius (ataskaitas).
- Sistema turi leisti buhalteriu atspausdinti atskirų grupių finansinius rodiklius (ataskaitas).
- Sistema turi leisti buhalteriu atspausdinti atskirų individų finansinius rodiklius (ataskaitas).
- Sistema turi leisti buhalteriu atspausdinti vartotojų sąrašus su finansiniais rodikliais (ataskaitas).
- Buhalteris turi turėti galimybę atlikti piniginius pervedimus tarp sąskaitų, jai sąskaitoje iš kurio pervedama yra pakankamai pinigų.
- Buhalteris neturi turėti galimybę atlikti piniginius pervedimus tarp sąskaitų, jai sąskaitoje iš kurio pervedama yra nėra pakankamai pinigų.
- Sistemoje turi būti galimybė vartotojams tvarkyti savo duomenys
- Prie sistemos gali prisijungti tik autorizuotas vartotojas.

- Sistema turi leisti klientui-vartotojui peržiūrėti savo sąskaitos balansą.
- Sistema turi leisti vartotojui peržiūrėti savo sąskaitos sąskaitas.
- Sistema turi leisti vartotojui atspausdinti savo apmokėtas arba peržiūrėtas sąskaitas.
- Sistema turi leisti vartotojui peržiūrėti savo sąskaitos sąskaitas.
- Sistema turi rodyti klientui-vartotojui detalią sąskaitos informaciją.
- Sistema turi leisti klientui-vartotojui atlikti sąskaitų paiešką.
- Sistema turi leisti pardavėjui/tiekėjui tvarkytis su apmokėjimais.
- Sistema turi leisti pardavėjui/tiekėjui pateikti pardavėjo duomenys .
- Sistema turi leisti pardavėjui/tiekėjui pateikti kliento duomenys.
- Sistema turi leisti pardavėjui/tiekėjui pateikti sumą kuri turi būti apmokėta
- Sistema turi leisti pardavėjui/tiekėjui tvarkytis su apmokėjimais.
- Sistema turi leisti pardavėjui/tiekėjui peržiūrėti savo sąskaitos balansą.
- Sistema turi leisti pardavėjui/tiekėjui tvarkytis sąskaitas.
- Sistema turi leisti pardavėjui/tiekėjui apmokėti, peržiūrėti sąskaitas.
- Sistema turi leisti pardavėjui/tiekėjui sąskaitų spausdinimą.

3.12. Reikalavimai duomenims

Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos dalykinė srities objektų specifikacija (Paveikslas 3.12-1.), kuri siejasi su kuriamą sistema. Tai atitinka kuriamos sistemos pradinį duomenų modelio variantą.



Paveikslas 3.12-1. Pradinis duomenų modelis

3.13. Nefunkciniai reikalavimai

3.13.1.Reikalavimai sistemos išvaizdai

Nustatomi bendri reikalavimai „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos vartotojo sąsajai.

Sistemos išvaizdos reikalavimai „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamai sistemai:

- Informatyvi, bet neperkrauta ir lengvai skaitoma sąsaja.
- Neįkyri vartotojo sąsaja.
- Lengvai valdoma sąsaja.
- Nesudėtingas meniu.
- Klientui-vartotojui sistemos teikiamos funkcijos turi būti prieinamos per interneto naršyklę.

3.13.2.Reikalavimai panaudojamumui

Nustatomi reikalavimai „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos panaudojamumui - panaudojimo paprastumui (lengvumui).

Panaudojamumo reikalavimai „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamai sistemai:

- Sistema turi būti nesudėtinga išmokti naudotis (buhalteriams, pardavėjams/teikėjams, klįjantams-vatotojams ir pan.).
- Sistema turi būti lengva naudotis (klientams-vartotojams ir pan.).
- Sistemoje turi būti įdiegta visapusiška pagalba vartotojui.
- Produktas turi padėti vartotojui nedaryti klaidų
- Sistemoje turi būti galimybė taisyti padarytas klaidas.
- Kur įmanoma, naudoti iškrentančius sąrašus.
- Kur įmanoma, naudoti pasirinkimo (Taip/Ne ir pan.) punktus.

3.13.3.Reikalavimai vykdymo charakteristikoms

Nustatomi reikalavimai „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos vykdymo charakteristikoms.

Vykdymo charakteristikų reikalavimai „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamai sistemai:

- Visi skaičiai rodomi su fiksuotu kiekiu skaitmenų po kablelio.
- Efektyvus resursų panaudojimas.
- Efektyvus užduočių vykdymas.
- Didelis vartotojų aptarnavimo pajėgumas.
- Greitas transakcijų vykdymas.
- Transakcijų neprarandamumas.
- Sistemos talpumas.
- Sistemos patikimumas.
- Sistemos pasiekiamumas.
- Išplečiamumas.

3.13.4.Reikalavimai veikimo sąlygoms

Nustatomi reikalavimai „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos veikimo (veikimo aplinkai) sąlygos.

Veikimo sąlygų reikalavimai „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamai sistemai:

- Sistema turėtų būti naudojama kambario temperatūroje.
- Sistema turėtų būti naudojama esant geram apšvietimui.
- Sistema turėtų būti naudojama ne drėgnoje aplinkoje.
- Sistema negali būti naudojama aplinkoje kur didelis triukšmas.
- Vibracija, jos lygis.
- Sistema negali būti numesta ar kritus ant žemes.

3.13.5.Reikalavimai sistemos priežiūrai

Nustatomi reikalavimai „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos priežiūrai.

Sistemos priežiūros reikalavimai „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamai sistemai:

- Veikimas kelių platformų pagrindu.
- Kitos kalbos pritaikymas.
- Duomenų vaizdavimo formos keitimas.
- Kitos kalbos pritaikymas.
- Nenumatytos prielaidos sistemos veikimui.

3.13.6.Reikalavimai saugumui

Nustatomi reikalavimai „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos saugumui. Tai vienas iš sudėtingiausių reikalavimų ir, be to, susijęs su didele rizika, jei jų nepaisoma.

Saugumo reikalavimai „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamai sistemai:

- Sistema turi užtikrinti duomenų konfidencialumą.
- Sistema turi užtikrinti duomenų vientisumą.
- Kurti naujus sistemos vartotojus gali tik buhalteris.
- Sistema turi neleisti neautorizuotiems vartotojams prisijungti prie sistemos ir ja naudotis.
- Sistema turi nutraukti vartotojo darbą, jei jis tam tikrą laiką ja nesinaudoja.
- Sistema neleidžia įvesti klaidingos informacijos.
- Buhalterinius sistemos darbus gali atlikti tik buhalteris.
- Pardavėjas/teikėjas gali valdyti tik su jo veikla tiesiogiai susijusią informaciją – duomenis.
- Klientas-vartotojas sistemoje gali valdyti tik savo sąskaitas.

3.14. Architektūros specifikacija

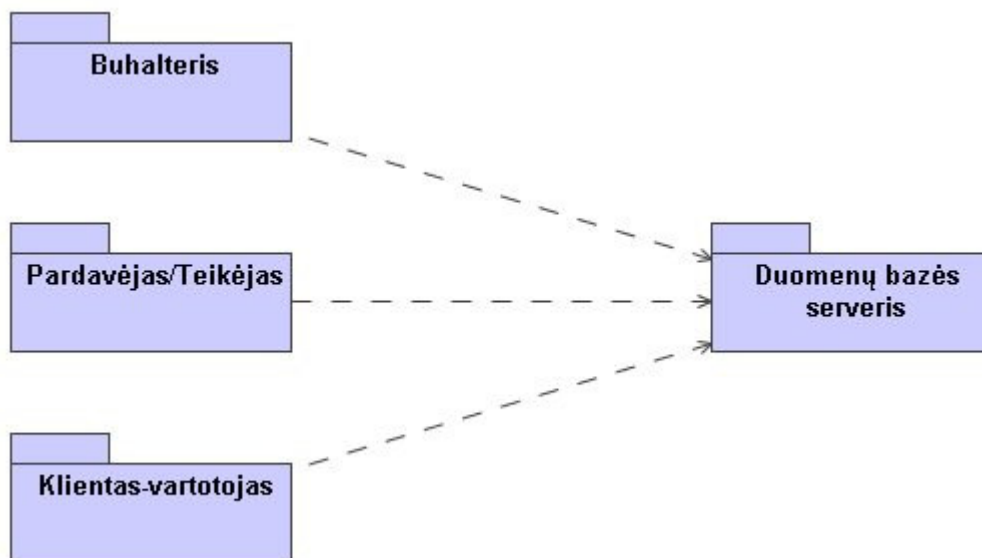
3.14.1. Sistemos statinis vaizdas

Aprašoma „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos loginė sistemos struktūra. Pateikiamas sistemos išskaidymas į paketus ir juos sudarančias klases.

3.14.1.1. Paketai

Projektuojama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistema yra suskaidyta į 4 paketus aukščiausiam lygyje.

Pastaba: Paketas buhalteris kartu apima ir administratoriaus funkcijas, kadangi projektuojamoje sistemoje buhalteris ir administratorius yra glaudžiai susiję, be to jų atliekamos funkcijos yra susijusios (funkcijos priklauso nuo suteiktų vartotojui teisių).

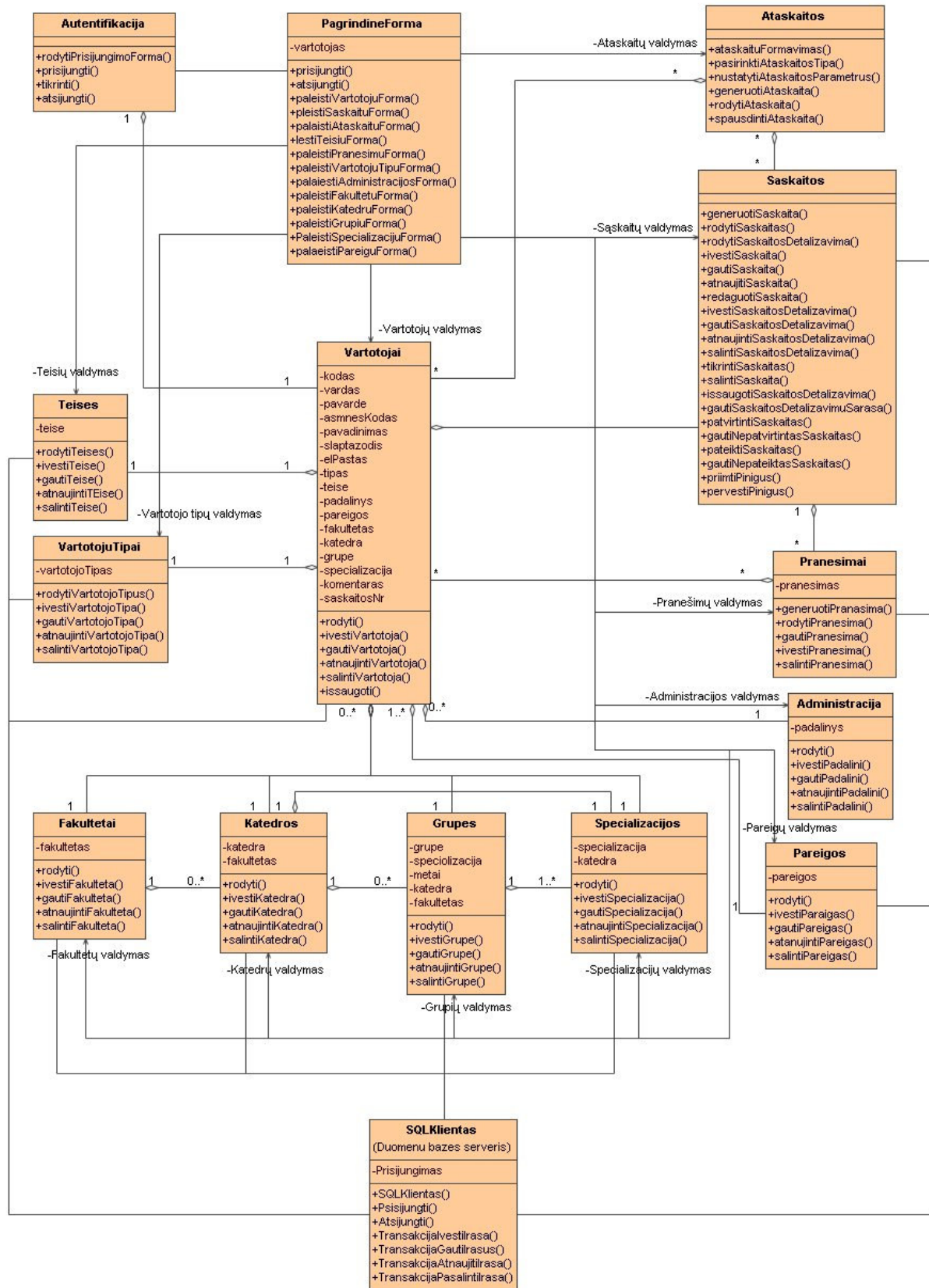


Paveikslas 3.14-1. Sistemos išskaidymas į paketus aukščiausiam lygyje diagrama.

3.14.1.2. Paketų detalizavimas

3.14.1.2.1. „Buhalteris“ paketo detalizavimas

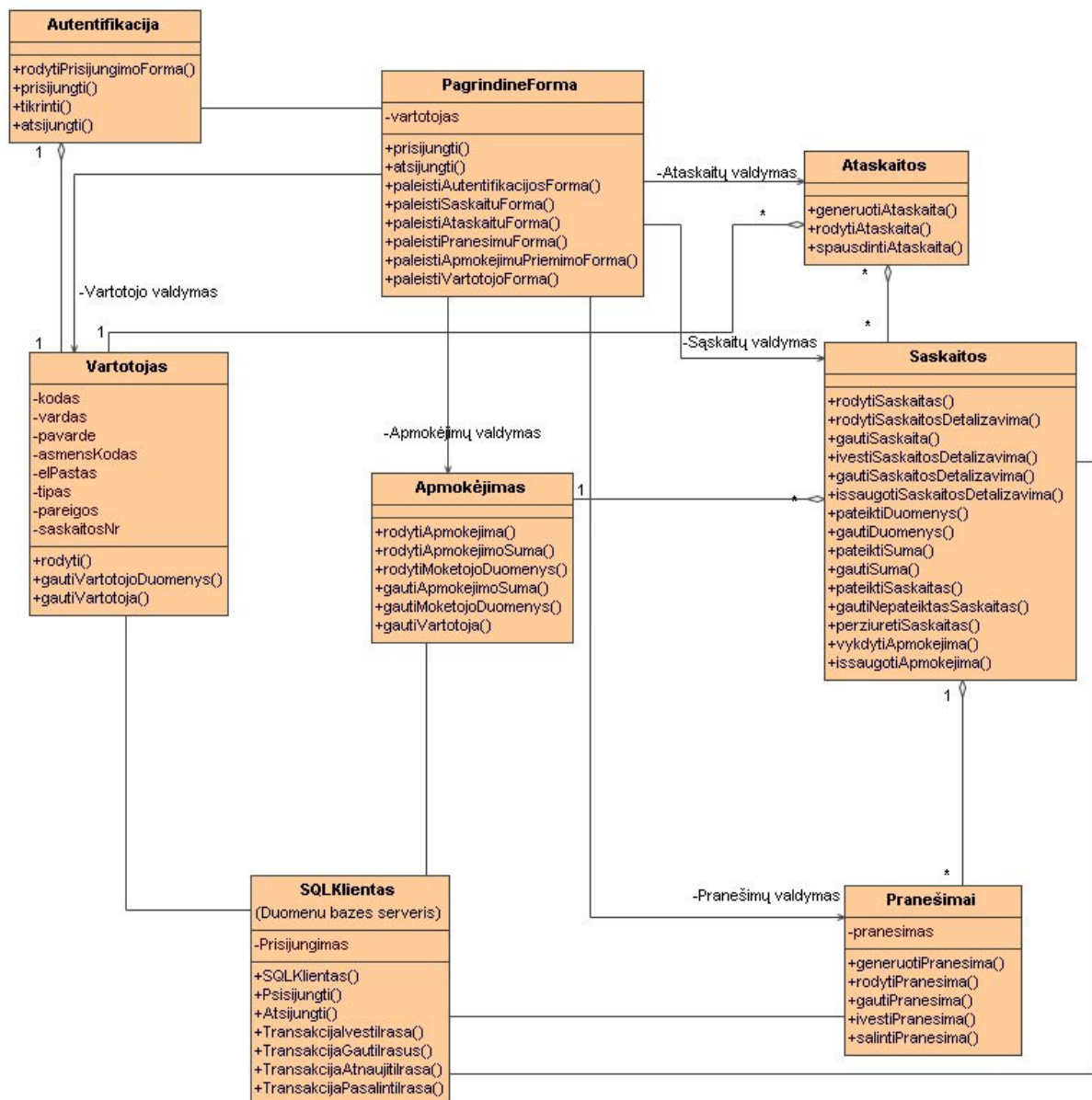
Buhalteris paketas skirtas „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos buhalterio (administraciniai) daliai aptarnauti. Paketas buhalteris kartu apima ir administratoriaus funkcijas, kadangi projektuojamoje sistemoje buhalteris ir administratorius yra glaudžiai susiję, be to jų atliekamos funkcijos yra susijusios (funkcijos priklauso nuo suteiktų vartotojui teisių). Šį paketą sudaro buhalterio (administracinę) dalį aptarnaujančios klasės. Naudojant šio paketo klases galima prisijungti prie buhalterio (administracinės) dalies, tvarkyti vartotojus ir sąskaitas, aptarnauti sąskaitas, tvarkyti pranešimus, priskirti vartotojams teises bei vartotojų tipus bei juos tvarkyti, priskirti vartotojams bei tvarkyti administracinį suskirstymą (t.y.. padaliniai, profesijos, fakultetai, katedros, grupės, specializacijos). Klasių diagrama pateikta paveiksle (Paveikslas 3.14-2.).



Paveikslas 3.14-2. Paketo Buhalteris klasių diagrama.

3.14.1.2.2. „Pardavėjas/Teikėjas“ paketo detalizavimas

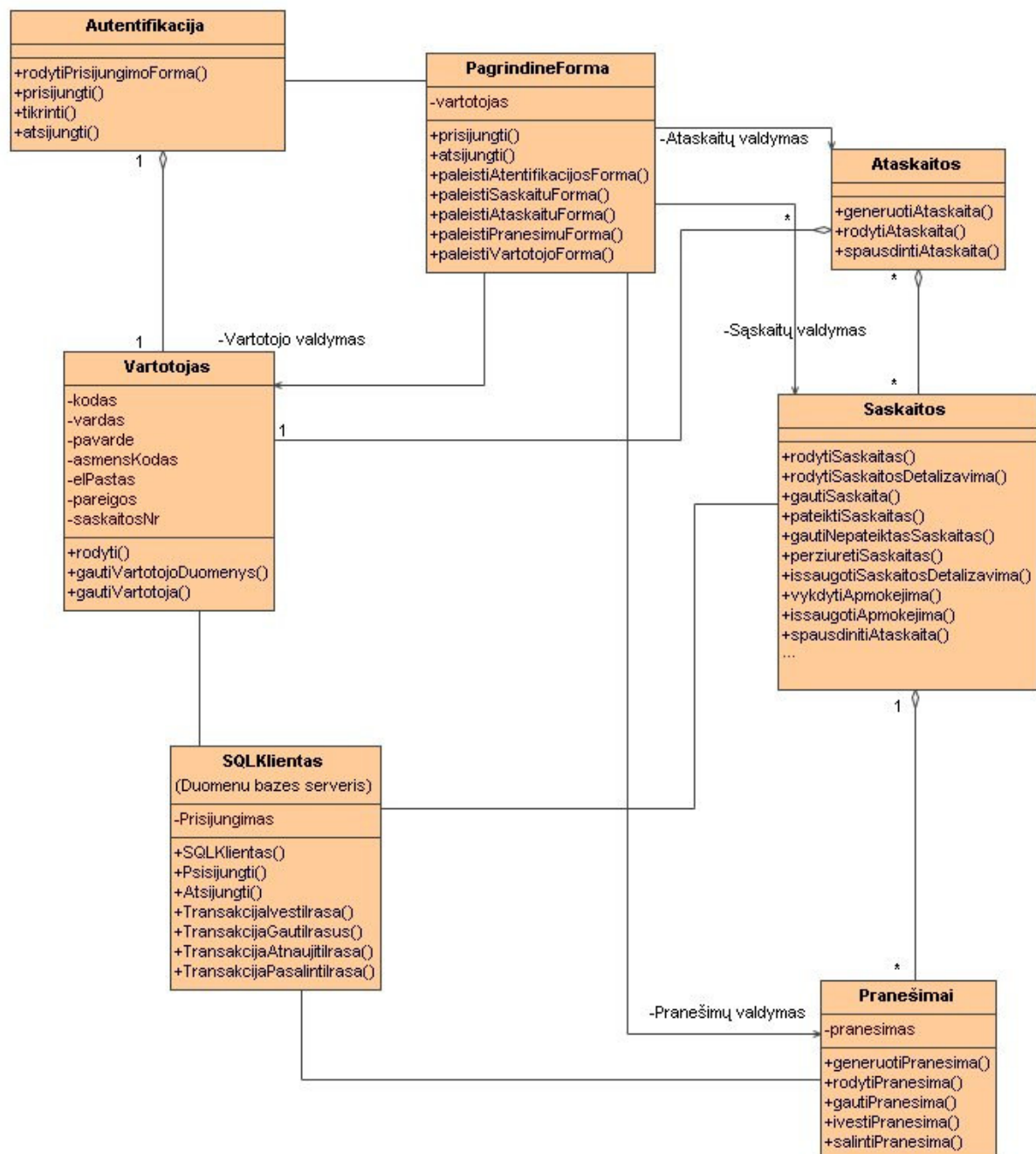
Pardavejas/Tiekėjas paketas skirtas „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos pardavėjo/tiekėjo daliai aptarnauti. Šį paketą sudaro pardavėjo/tiekėjo dalį aptarnaujančios klasės. Naudojant šio paketo klases galima prisijungti prie pardavėjo/tiekėjo dalies, tvarkyti savo sąskaitas, aptarnauti sąskaitas(peržiūrėti sąskaitų balansą, atsispausdinti), tvarkyti pranešimus, priimti apmokėjimą(pateikti sumą). (Klasių diagrama pateikta paveiksle (Paveikslas 3.14-3.).



Paveikslas 3.14-3. Paketo *Pardavėjas/Teikėjas* klasių diagrama.

3.14.1.2.3. „Klientas-vartotojas“ paketo detalizavimas

Klientas-vartotojas paketas skirtas „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos kliento-vartotojo daliai aptarnauti. Šį paketą sudaro kliento-vartotojo dalį aptarnaujančios klasės. Naudojant šio paketo klases galima prisijungti prie kliento-vartotojo dalies, tvarkyti savo sąskaitas, aptarnauti sąskaitas, tvarkyti pranešimus. (Klasių diagrama pateikta paveiksle (Paveikslas 3.14-4.).



Paveikslas 3.14-4. Paketo *Klientas-vartotojas* klasių diagrama.

3.14.1.2.4. „Duomenų bazės serveris“ paketo detalizavimas

Duomenų bazės serveris paketas skirtas duomenų bazės abstrakcijos klasėms. Tai klasės skirtos darbui su duomenų baze. Klasės naudojamos daugumoje paketų. Klasių diagrama pateikta paveiksle (Paveikslas 3.14-5.).

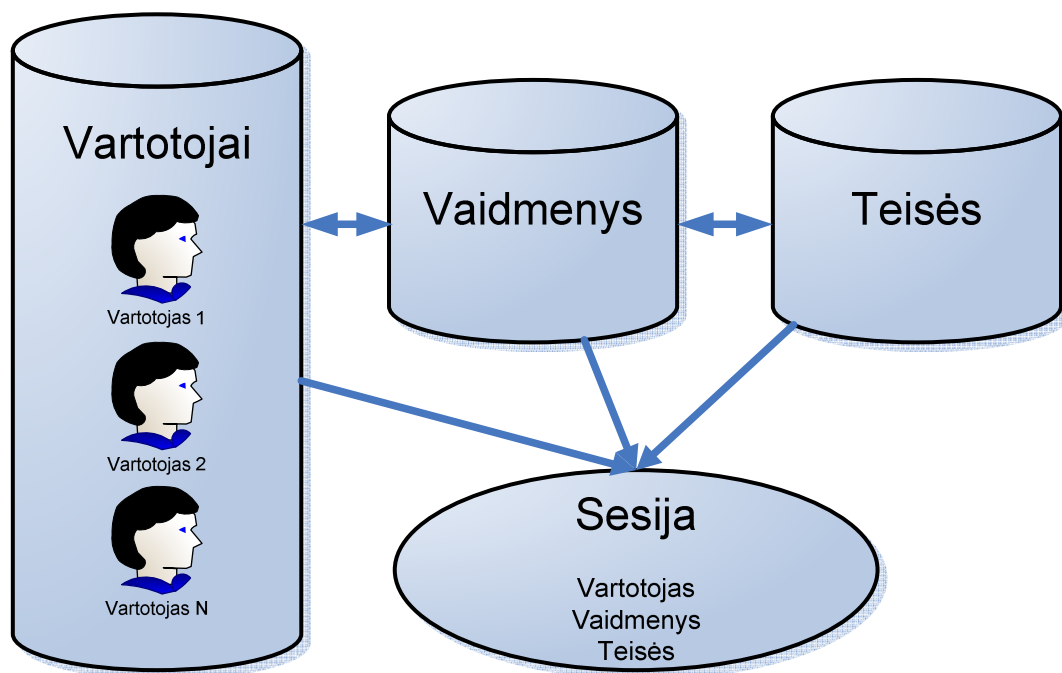


Paveikslas 3.14-5. Paketo *Duomenų bazės serveris* klasių diagrama.

3.14.2. Sistemos saugumo užtikrinimas

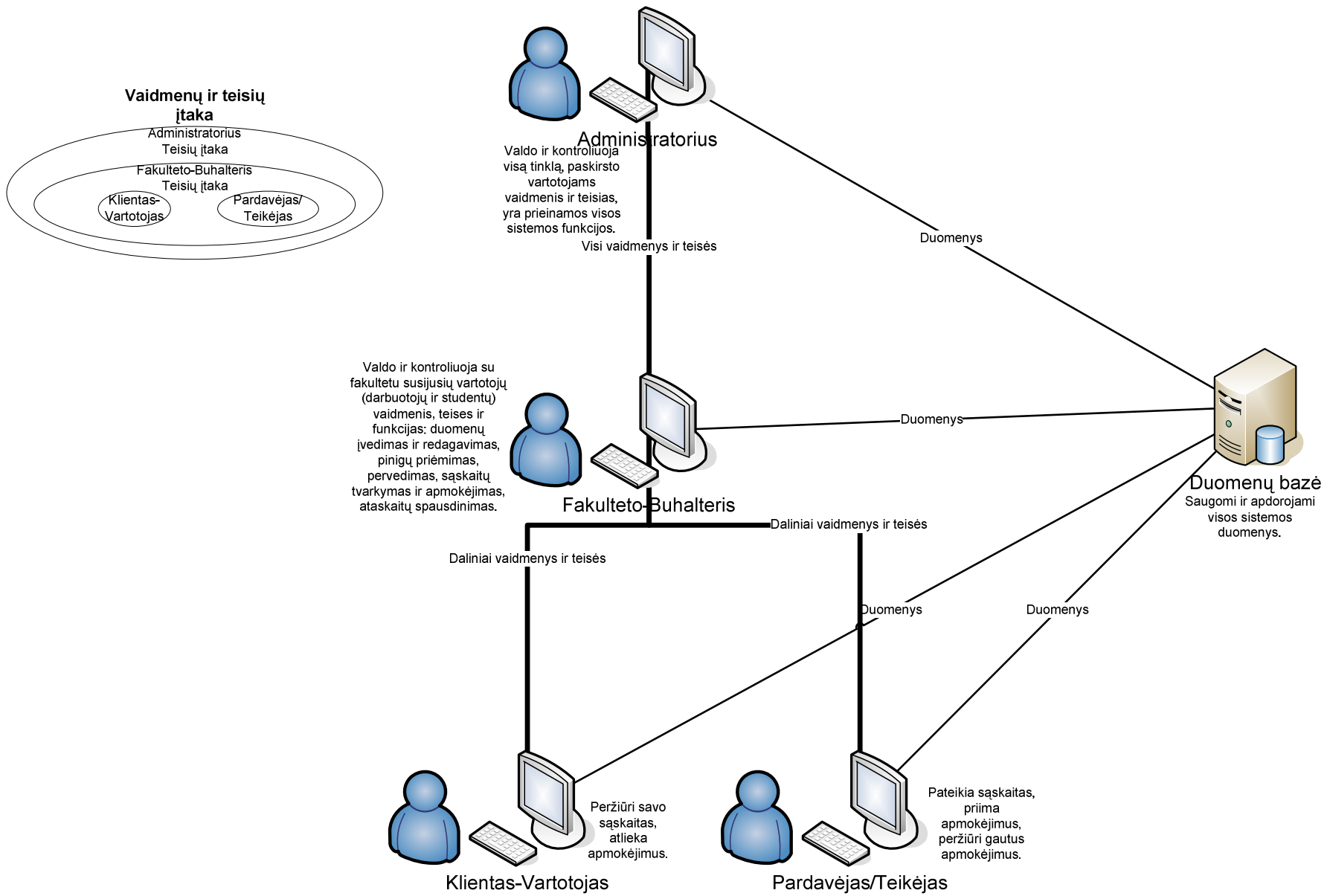
Siekiant užtikrinti „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemoje esančios informacijos konfidencialumą, būtina įdiegti tam tikrus saugumo principus. Pagrindė saugumo problemos yra susijusio su saugoma informacija, tad siekiant užtikrinti saugų darbą su sistema, į sistema turi būti įdiegta vartotojo prieigos kontrolė.

Naudojant prieigos kontrolę, kiekvienam sistemoje esančiam vartotojui, turi būti priskirti, tam tikri vaidmenys ir tuo vaidmenis atitinkančios teisės. Pastaruosius vartotojas naudos siekdamas gauti ar apdoroti kažkokią informaciją, kuri jam yra prieinama pagal jam suteiktas teises. Taikant prieigos kontrolės principą visi sistemoje esantys vartotojai turi būti identifikuoti, ir sesijos metu (darbo su sistema metu) gaus tam tikrus vaidmenis ir teises, iš to išplaukia, kad sistemoje, turi būti saugoma ne tik finansinė vartotojų informacija su vartotojai, bet ir saugumą užtikrinanti informacija, kaip kad vartotojo identifikaciniai duomenys, vaidmenys, teisės. Visa tai supaprastintai pateikta paveiksle (Paveikslas 3.14-6.).



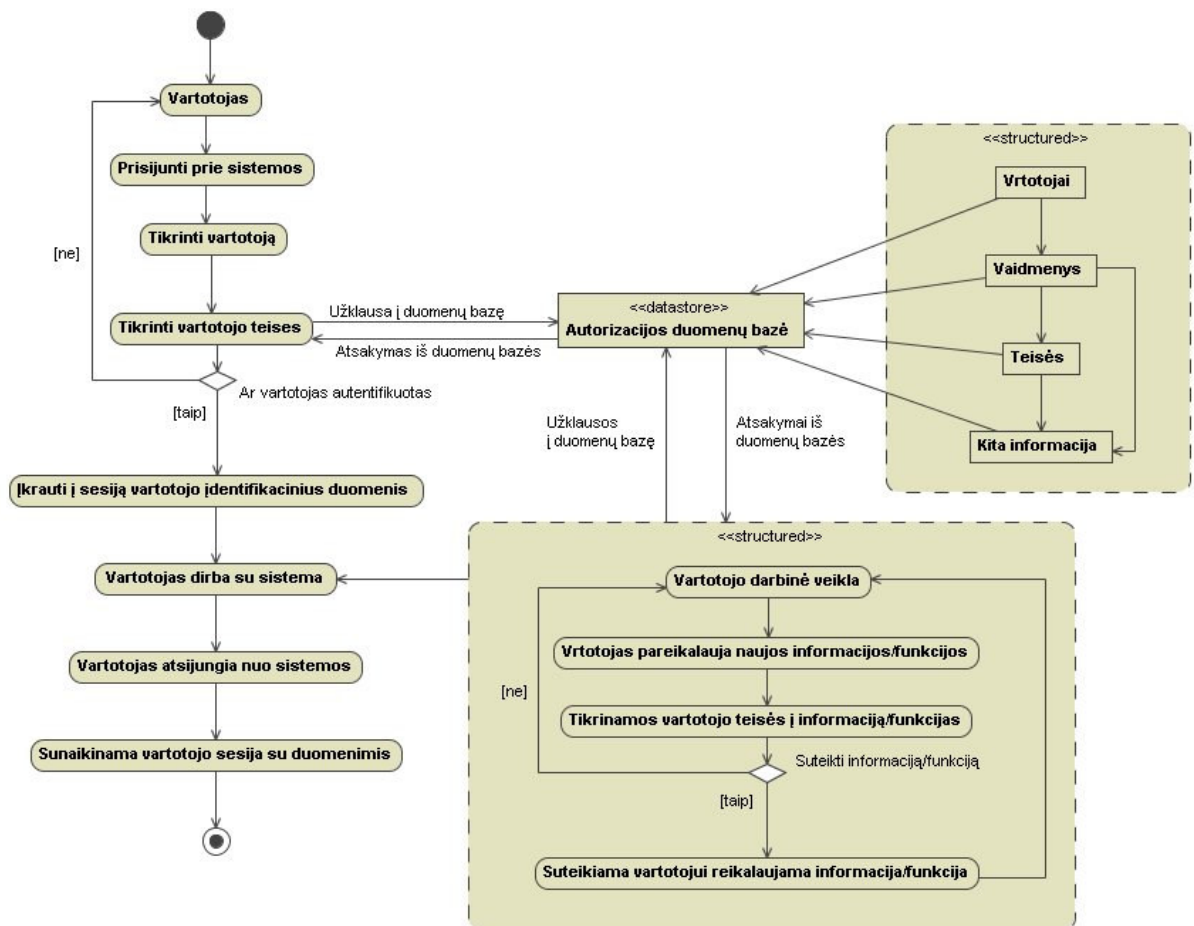
Paveikslas 3.14-6. Prieigos kontrolės duomenų schema.

Kad būtų įmanoma kontroliuoti sistemą, turinčią paskirstyto prieigos kontrolę, visi sistemos vartotojai, turi pagal tam tikrą vartotojų hierarchiją, kuri suteikia vartotojams tam tikrus vaidmenis ir teises, kaip parodyta žemiau pateiktam paveiksle (Paveikslas 3.14-7.).



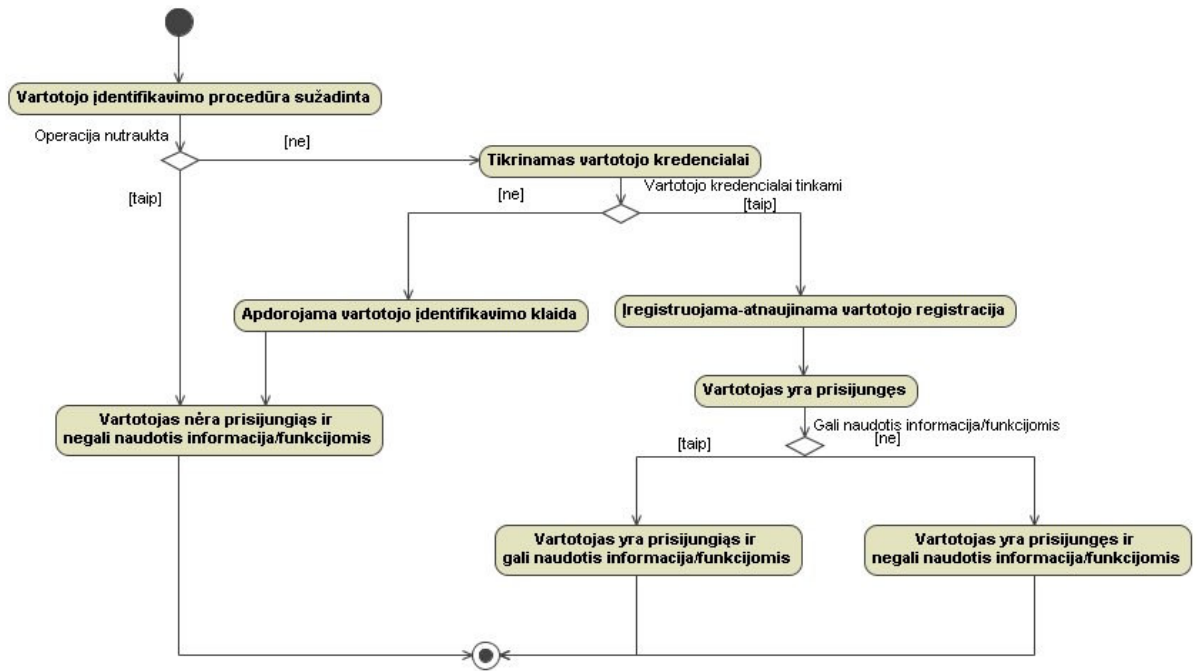
Paveikslas 3.14-7. Vartotojų, vaidmenų ir teisių hierarchinis pasiskirstymas.

Visi vartotojai dirbantys su sistema turi būti autorizuoti, bei darbo metu turi būti nuolat tikrinama ar dar tas pats vartotojas dirba su sistema, aišku nuolatinis tikrinimas labai apkrautą sistemą, ir sukeltų nepageidaujamas problemas vartotojams, tad tokie tikrinimai, turi būti atliekami kai vartotojas pareikalauja, kažkokių tai naujų duomenų ar funkcijų darbo metu, kaip galėtų atrodyti tokia kontrolė, yra pateikta žemiau pateiktame paveiksle (Paveikslas 3.14-8.).



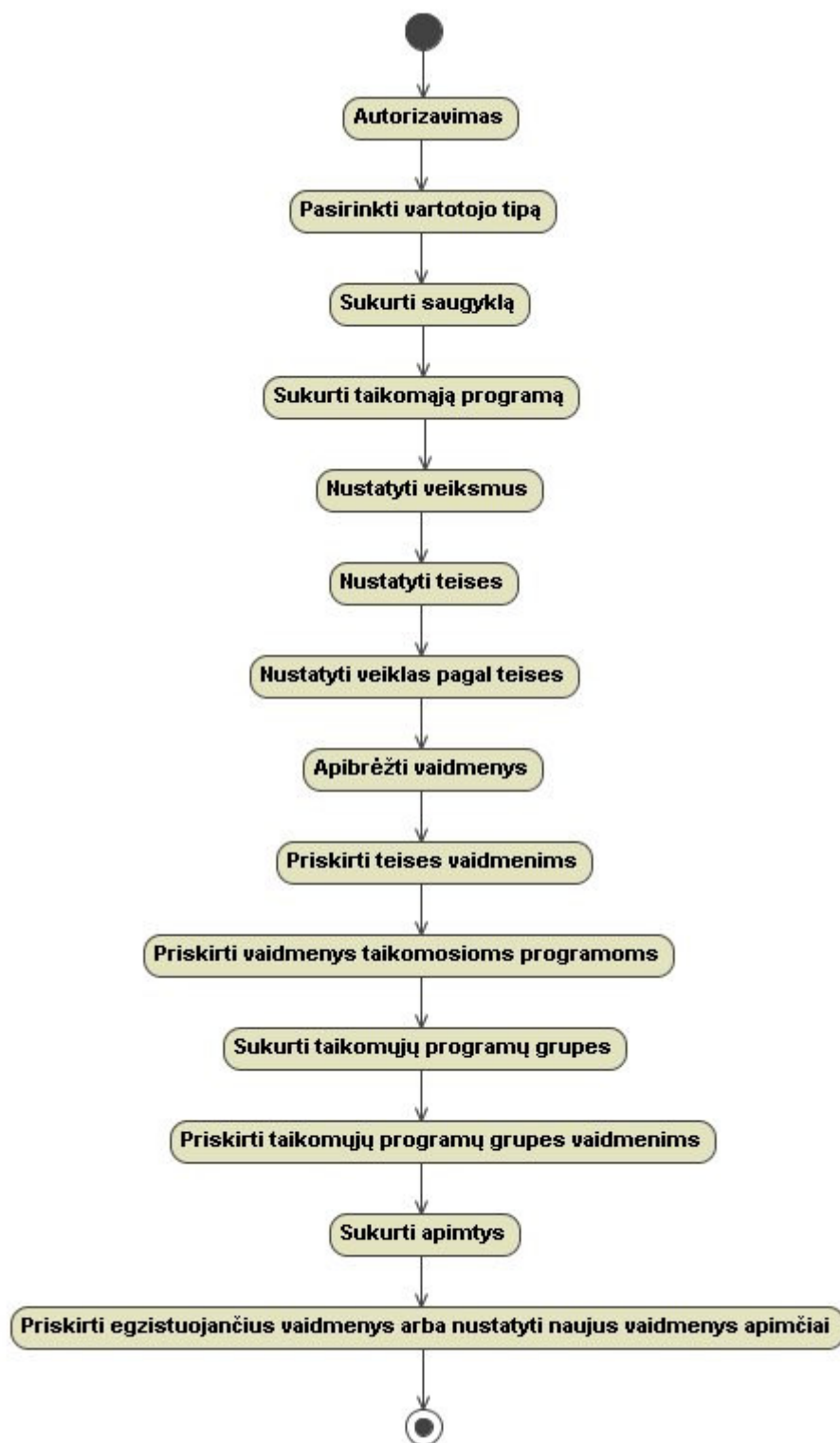
Paveikslas 3.14-8. Vartotojo prieigos kontrolė.

Sistemoje visi dirbantys vartotojai turi būti identifikuoti, jei vartotojas yra neidentifikuotas, arba jis dėl kažkokių priežasčių darbo sesijos metu prarado savo identifikacinius duomenis, jis negali dirbti su sistema, jei identifikacinius duomenis praras darbo sesijos metu, jis turės jungtis prie sistemos iš naujo. Tad turi būti vykdoma nuolatinė vartotojo kontrolė, kuri kartu tikrina netik ar jis yra prisijungęs prie sistemos, bet ir jo galimybes naudotis tam tikra informacija ar funkcijomis, visa tai tikrinama, kai vartotojas paprašo naujos informacijos ar funkcijos(-ų), kai vykdoma vartotojo identifikacinė kontrolė, parodyta žemiau pateiktam paveiksle (Paveikslas 3.14-9.).



Paveikslas 3.14-9. Vartotojo identifikacinė kontrolė.

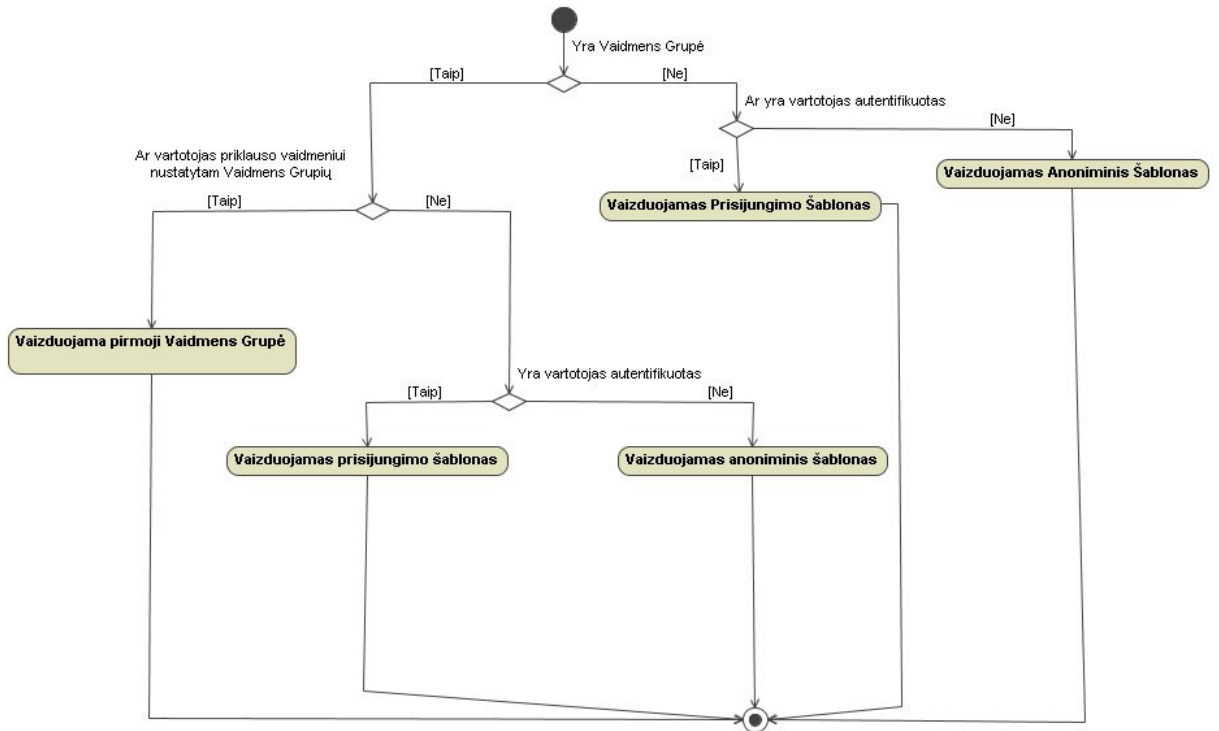
Sistemoje esantys ir dirbantys vartotojai turi būti autorizuojami, todėl visa tam reikiam informacija turi būti saugoma darbinėje duomenų saugykloje (sesijoje). Žemiau pateiktam paveiksle (Paveikslas 3.14-10.) vaizduojami žingsniai įtraukiantys saugyklos kūrimo, taikomųjų programų, teisių, priskirtų veiksmų teisėms, nustatytus vaidmenis, priskirtus veiksmų vaidmenis, nustatytų vaidmenų taikomosioms programoms, priskirtų vartotojų/vartotojo grupių vaidmenims ir sukūrimo apimčių veiklas.



Paveikslas 3.14-10. Žingsniai po autorizavimo.

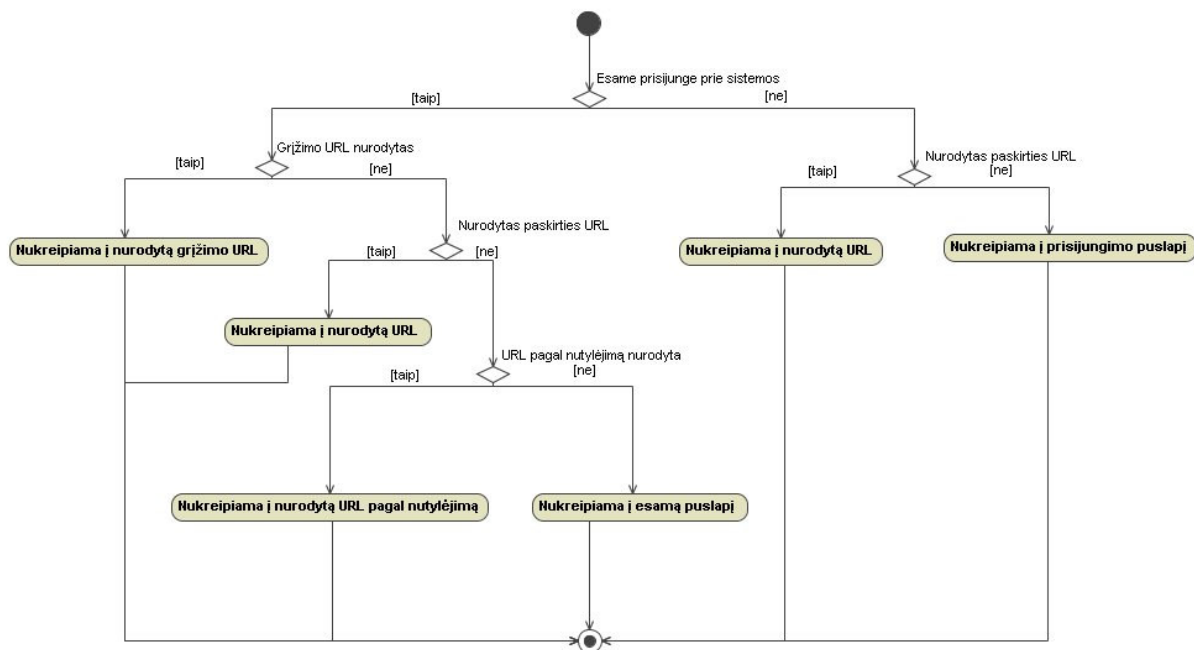
Sistemos vartotojams yra priskirti tam tikri vaidmenys, kuriais remiantis yra atrenkamos atitinkamos teisės skirtos vartotojams (Paveikslas 3.14-11.). Pradėdama nuo tikrinimo ar vartotojas priklauso Vaidmens Grupei ar ne, nuo to priklauso ar jis nukreipiamas į anoniminius vartotojus ar ne. Nukreiptas į anoniminius vartotojus, gauna tik anoniminems

virtotojams priskirta informaciją. Nukreiptas į autorizuotų virtotojų Vaidmens Grupę jam yra vaizduojama informacija skirta autorizuotam virtotojui.



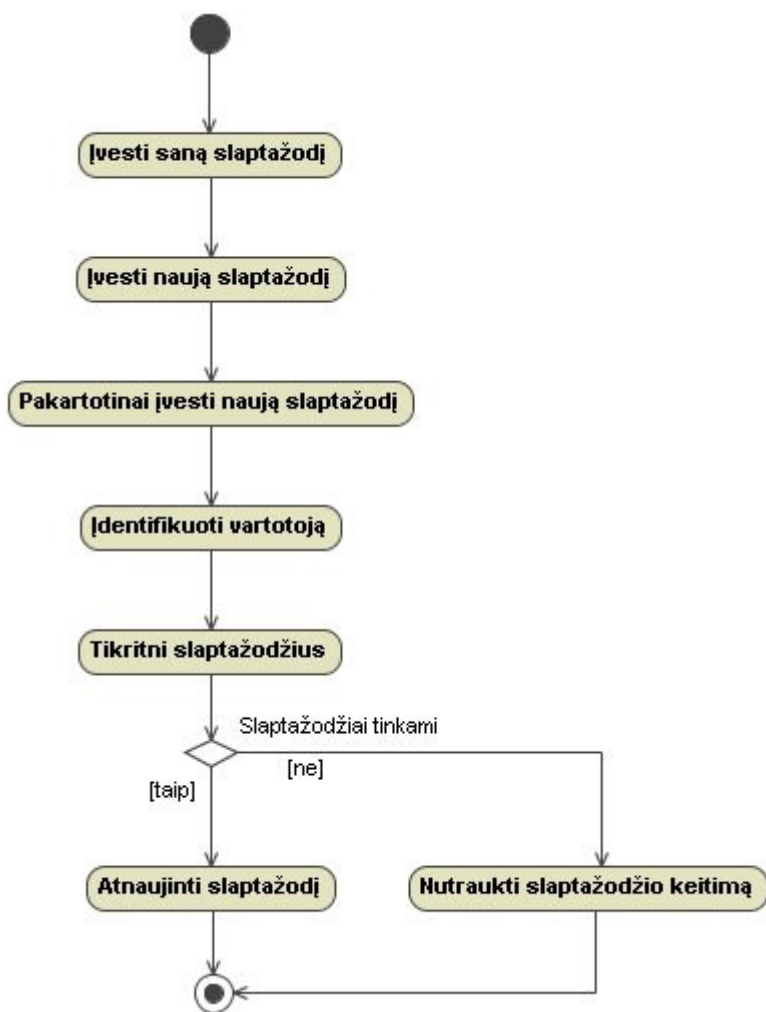
Paveikslas 3.14-11. Autorizavimas pagal Vaidmenų Grupes.

Kadangi virtotojo prieigos kontrolė nuolat vyksta, tai sistemos internetiniai puslapiai kontroliuojami turėtų būti kaip parodyta žemiau pateiktam paveiksle (Paveikslas 3.14-12.). Jei virtotojas nėra prisijungęs prie sistemos tai jis bus nukreipiamas į atitinkamą puslapį (kai nutrūksta prisijungimas), arba į prisijungimo puslapį. Jei virtotojas yra prisijungęs, tai jam pateikiami puslapiai pagal vykdomą darbinę veiklą.



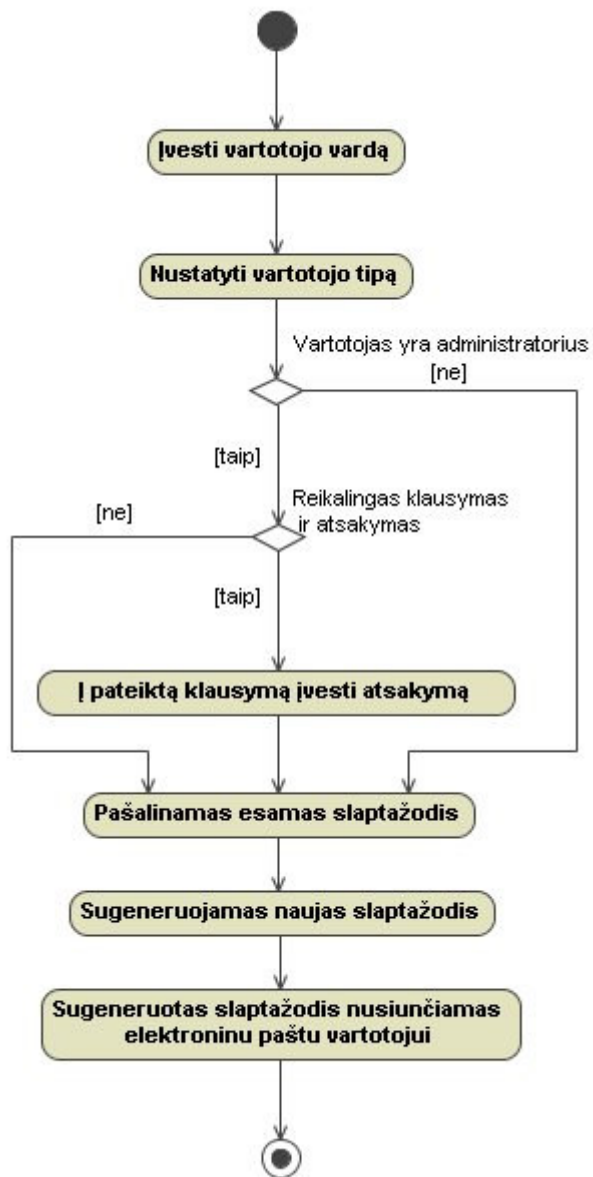
Paveikslas 3.14-12. Prieigos kontrolė sistemos taikymas internetiniams puslapiams.

Visi sistemoje esantys vartotojai yra identifikuojami vartotojo vardas ir slaptažodžiu, vartotojo vardas ir slaptažodis yra priskiriamas jo sukūrimo metu. Tačiau jei vartotojui nepatinka jo turimas slaptažodis jis jį gali pasikeisti, pasinaudodamas slaptažodžio pakeitimo funkcija, pateikta Paveikslas 3.14-13.



Paveikslas 3.14-13. Vartotojo slaptažodžio pakeitimas.

Jei vartotojas pamiršo savo slaptažodį, tai jis gali būti atkurtas, slaptažodį atkurti gali tik įgaliooti asmenys, turintys tam reikiamas teises (pvz. administratorius), tai yra reikalinga, kad būtų užtikrintas tinkamas sistemos saugumas. Tačiau tik sistemos administratoriai (turintys tam reikiamas teises) gali atsikurti slaptažodį patys, kaip tai turėtų būti realizuota parodyta žemiau pateiktam paveiksle (Paveikslas 3.14-14.)



Paveikslas 3.14-14. Vartotojo slaptažodžio atstatymas.

3.14.3. Išdėstymo vaizdas

3.14.3.1. Išdėstymo vaizdas

Sukurtą „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojama sistema, bus panaudota atlikti įvairioms finansinėms operacijoms universitetinėje aplinkoje, kaip pavyzdį būtų galima paminėti, tokias kuriamos programinės įrangos galimybes: priimti iš studentų ar darbuotojų apmokėjimus už: kopijavimo, spausdinimo, maitinimo, apsipirkinėjimą knygyne ir pan. paslaugas. Sukurta programinė įranga leis pasinaudojant studento ar darbuotojo pažymėjimu atlikti apmokėjimą už paslaugas. Peržiūrėti savo sąskaitos likutį, atlikti pervedimus „Elektroninių mokėjimų universitetinėje aplinkoje programinės įrangos“ teikiamomis galimybėmis.

Toliau pateikiama aprašymas, kaip visa tai turėtų atrodyti ir funkcionuoti.

Klientas-vartotojas norėdamas susimokėti už prekes ar suteiktas paslaugas pas pardavėją, ar paslaugų teikėją (pardavėją/teikėją), kai pardavėjas/ teikėjas suformuoja sąskaitą ir paprašo kliento-vartotojo apmokėti, klientas-vartotojas pateikia mokėjimo kortelę (t.y.. studento ar darbuotojo pažymėjimą, kuris atitinka mokamąją kortelę). Pardavėjas/teikėjas parbraukia kortelę per mokėjimo kortelių nuskaitymo įrenginį, toliau sistema prie kliento-vartotojo informacijos gautos iš mokomosios kortelės prideda pardavėjo/teikėjo bei pastarojo sąskaitos informaciją ir toliau visa informacija per universitetinį mokėjimų (universiteto vidinį) tinklą išsiunčiama į elektroninių mokėjimų serverį, kuris patikrina ar klientas-vartotojas ir pardavėjas/teikėjas yra tikri, jai kuris nors ar abu netikri, tai apie tai gražinamas pranešimas, jai abu tikri, tai toliau patikrina ar kliento-vartotojo sąskaitoje yra pakankamai pinigų, jai pinigų trūksta, tai gražina informaciją pardavėjui/ teikėjui, kad apmokėjimas negalimas. Jei kliento-vartotojo sąskaitoje yra pakankamai pinigų tai iš kliento-vartotojo sąskaitos reikimą sumą perveda ir pardavėjo/ teikėjo sąskaitą, ir apie apmokėjimą, informuojamas pardavėjas/teikėjas. Taip pat suregistruojama visą reikiama informacija duomenų bazėje.

Sistemos kliento-vartotojas (universiteto studentas ar darbuotojas) norėdamas įnešti į savo sąskaitą pinigus kreipiasi į buhalterį, kuris įmokėtą į universiteto sąskaitą pinigų sumą perveda į kliento-vartotojo sąskaitą (įmoka gali būti grynais, pervedimu ir pan.).

Buhalteris tvarko ne tik įmokas, bet taip pat ir visą finansinę informaciją, kaip kad: sąskaitas, jų paskirstymą klientams-vartotojams, sąskaitų balansą ir pan. Buhalteris taip pat

lieka dalį administratoriaus funkcijų, kurios yra glaudžiai susijusios su jo veikla, kad vartotojų informacijos tvarkymas ir pan.

Administratorius prižiūri elektroninių mokėjimų kompiuterinį tinklą, tvarko ir prižiūri duomenų bazę, rūpinasi, kad visa sistema dirbtų stabiliai ir be sutrikimų.

Visi sistemos vartotojai (universiteto studentai ir personalas) gali peržiūrėti savo sąskaitas per internetą, naudodami interneto naršyklę, iš bet kurios pasaulio vietos. Vartotojai gali paržiūrėti pinigų pervedimus į savo sąskaitas, peržiūrėti atliktų apmokėjimų sąskaitas, peržiūrėti neapmokėtas sąskaitas ir jas apmokėti, jai sąskaitoje yra pakankamai pinigų. Jai vartotojas sugalvoja apmokėti sąskaitą, jis pasirenka norimą sąskaitą, paspaudžia apmokėjimo mygtuką, įveda turimą apsaugos kodą, ir patvirtina apmokėjimą. Toliau universiteto mokėjimo sistema visą reikiamą kliento ir sąskaitos informaciją nusiunčia į serverį, serveris patikrina ar vartotojas tikras (pagal gautą informaciją ir apsaugos kodą), jai klientas netikras, tai apmokėjimas nevykdoma ir apie tai informuojama, jai vartotojas tikras, tada yra patikrinama ar yra pakankamai pinigų vartotojo sąskaitoje, jai yra pakankamai pinigų tai įvykdo apmokėjimą ir apie tai informuoja vartotoją, jai nėra – tai informuoja vartotoją, kad apmokėjimas negalimas. Taip pat vartotojas gali peržiūrėti išsamią sąskaitų istoriją, bei savo sąskaitos balansą. Visą atėjusi naujų sąskaitų, įmokų į savo sąskaitą ir apmokėjimų informacija yra išsiunčiama į kliento nurodytą elektroninį paštą. Todėl sistemos vartotojas pastarąją informaciją gali peržiūrėti ir neprisijungęs prie apmokėjimų sistemos, tiesiog pasinaudodamas savo elektroniniu paštu.

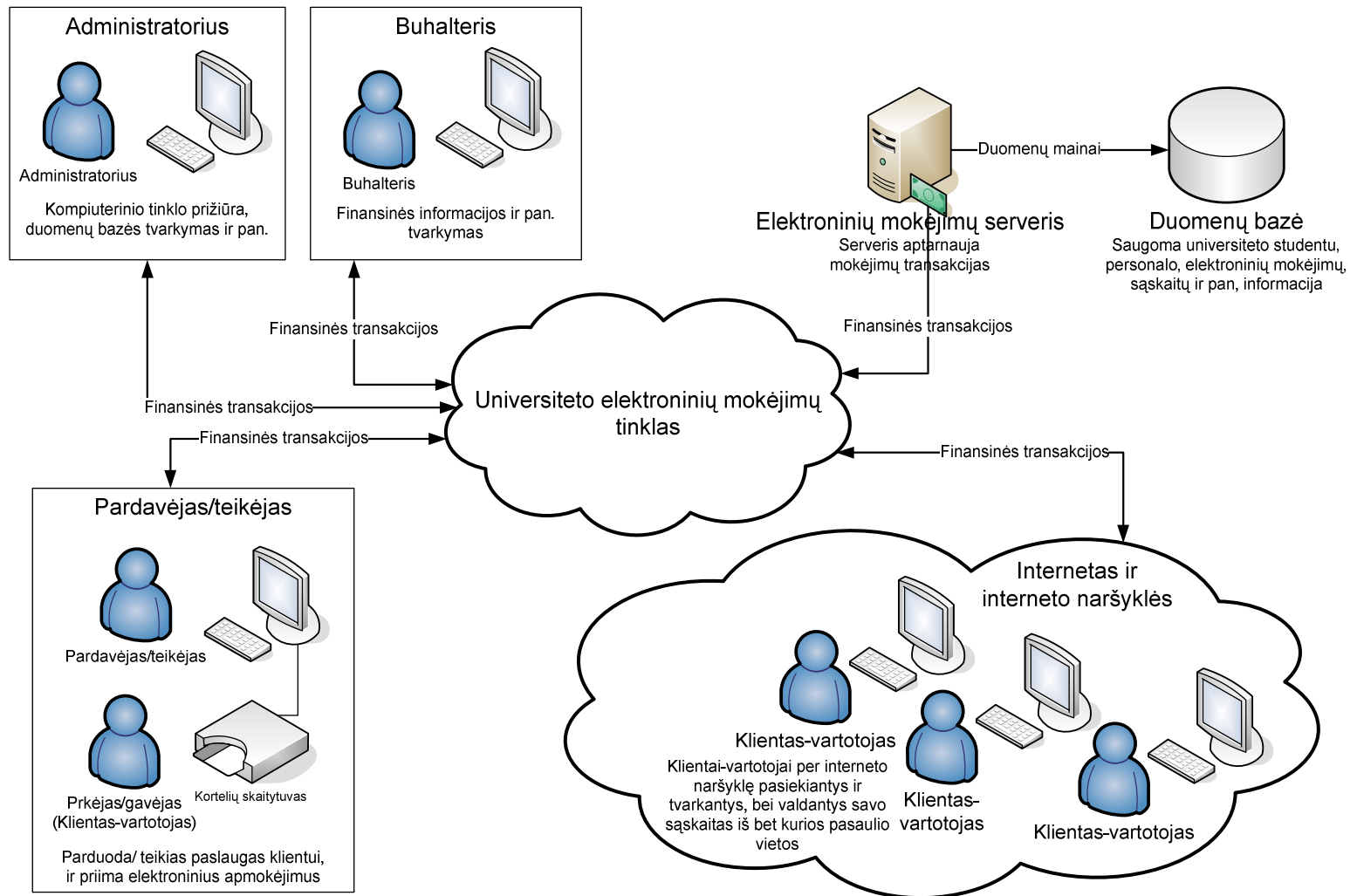
Serveris vykdo duomenų mainus su duomenų baze, registruoja ir tvarko informaciją duomenų bazėje, tvarko sistemos vartotojų autentiškumo tikrinimo procedūrą, tikrina sąskaitas ar jose yra pakankamai pinigų, jai yra pakankamai pinigų, tai juos perveda į pardavėjų/teikėjų sąskaitas (esant apmokėjimo poreikiui), organizuoja finansinės informacijos pateikimo, bei peržiūros operacijas, išsiuntinėja pranešimus ir patvirtinimus, į vartotojų elektroninius paštus, ir pan.

Visą finansinių transakcijų informacija yra siunčiama panaudojant SSL saugumą užtikrinantį protokolą, ir saugiais tiesioginio sujungimo TCP/IP kanalais. Visą finansinių transakcijų informacija, vykstanti vidiniame universiteto kompiuterių tinkle „keliauja“ link serverio ir atgal per vidinį universiteto kompiuterių tinklą (intranetą), jį galime tiesiog pavadinti „Universiteto elektroninių mokėjimų tinklu“, o informacijos mainai su išore vykdomi per globalų kompiuterių tinklą (internetą). Visi duomenys yra saugomi duomenų bazėje, o po to tik apdorojami, tokiu būdu yra apsisaugoma nuo svarbios informacijos praradimo sutrikus sistemos darbui.

Sistemą sudarančios sąsajos: administratoriaus, buhalterio, pardavėjo/teikėjo ir kliento-vartotojo (interneto naršyklės) sąsajos

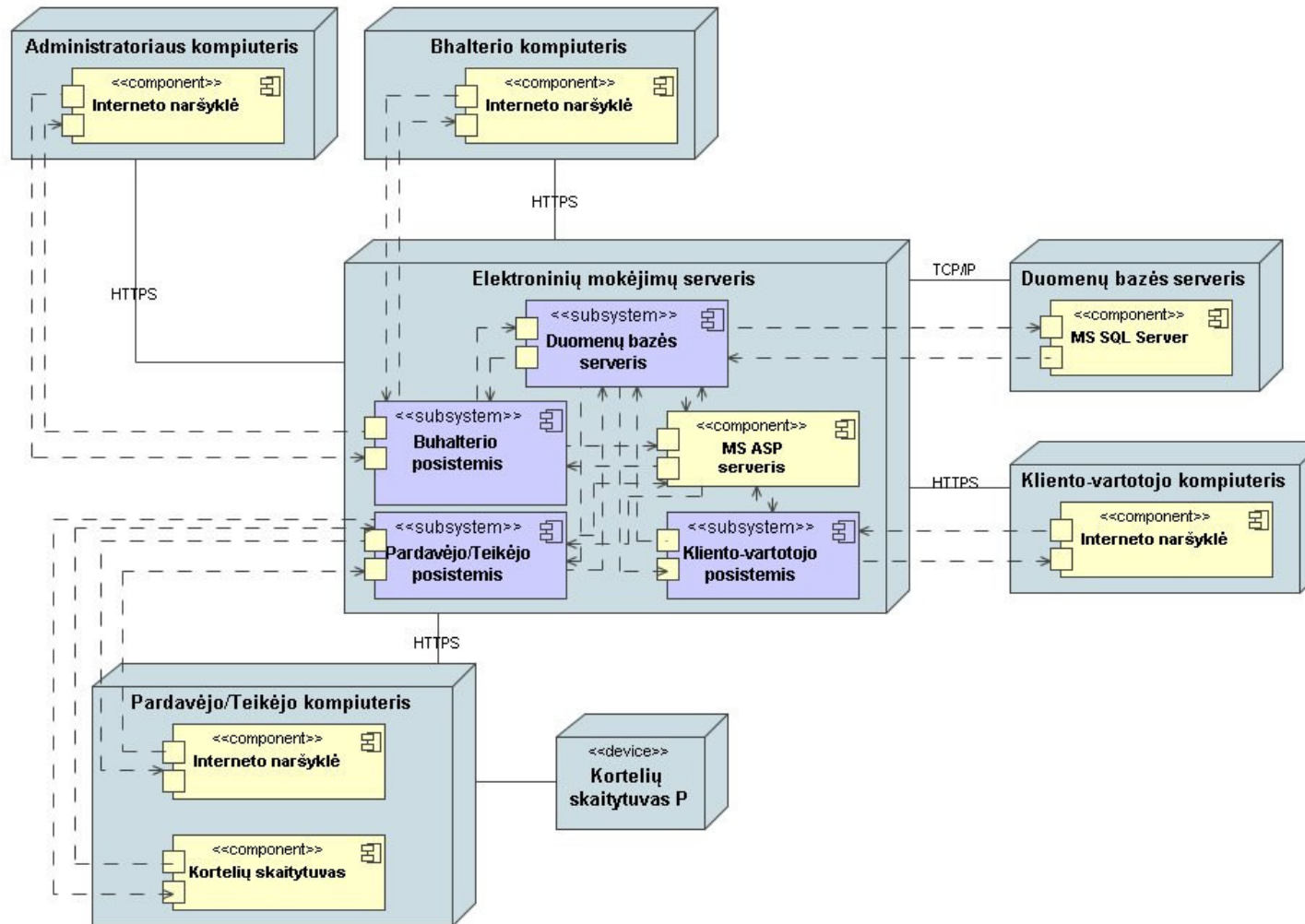
Projektuojamos „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos išdėstymo koncepcinė schema (diagrama) (Paveikslas 3.14-15.), bei sistemos išdėstymo vaizdo diagrama (Paveikslas 3.14-16.), pateikto žemiau esančiuose paveiksluose.

„Elektroninių mokėjimų universitetinėje programinė įranga“ sistemos išdėstymo vaizdo koncepcinė schema



Paveikslas 3.14-15. „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos išdėstymo vaizdo koncepcinė diagrama.

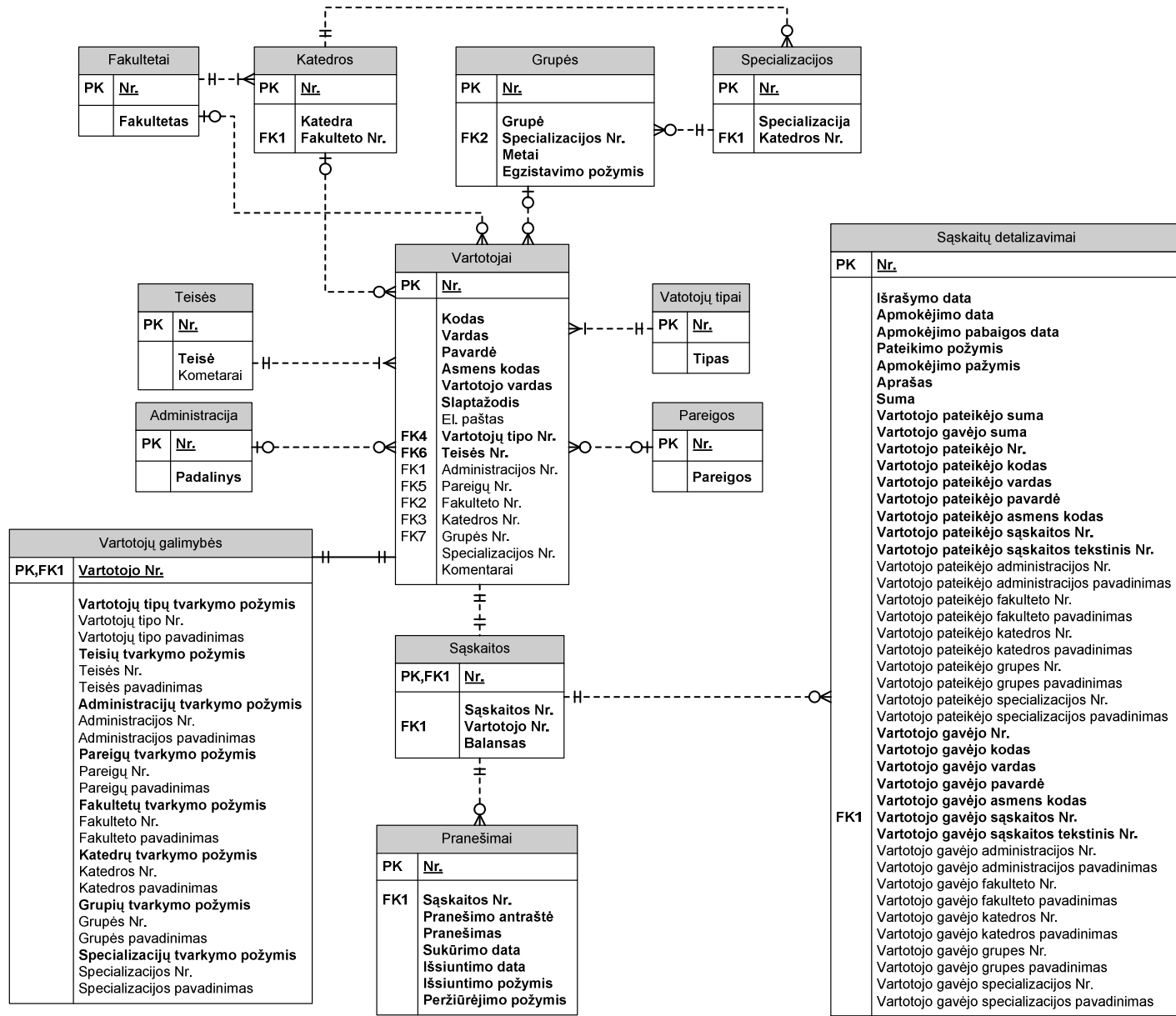
„Elektroninių mokėjimų universitetinėje programinė įranga“ sistemos išdėstymo vaizdo schema



Paveikslas 3.14-16. „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos išdėstymo vaizdo diagrama.

3.15. Duomenų modelis

Projektuojamos „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos duomenų valdymo sistemai yra pasirinkta Microsoft SQL Server Enterprise 2005 duomenų bazės valdymo sistema. Kuriamos projektuojamos sistemos duomenų bazės modelio vaizdas pateiktas žemiau esančiame paveiksle (Paveikslas 3.15-1.).



Paveikslas 3.15-1. Duomenų bazės modelis.

Projektuojamos „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos duomenų bazės modelyje (Paveikslas 3.15-1.) esančių esybių aprašymai yra pateikti sekančioje lentelėje (Lentelė 3.15-1.).

Lentelė 3.15-1. Duomenų bazės modelio esybės.

Esybė	Aprašymas
Fakultetai	Saugoma informacija apie KTU fakultetus.
Katedros	Saugoma informacija apie KTU fakultetų katedras.
Grupės	Saugoma informacija apie KTU studentų grupes.
Specializacijos	Saugoma informacija apie KTU mokslo profesines specializacijas.
Administracija	Saugoma informacija apie administracijos padalinius.
Vartotojai	Saugoma informacija apie sistemos vartotojus.
Vartotojų tipai	Saugoma informacija apie sistemos vartotojų tipus.
Teisės	Saugoma informacija apie sistemos vartotojų teises naudojantis sistema.
Vartotojų galimybės	Saugoma informacija sistemos vartotojų galimybės naudojantis sistema.
Pareigos	Saugoma informacija apie vartotojų pareigas.
Sąskaitos	Saugoma informacija apie sistemos vartotojų sąskaitas.
Sąskaitų detalizavimai	Saugoma vartotojų sąskaitų detalizavimo informacija.
Pranešimai	Saugomi sistemos vartotojams skirti pranešimai.

3.16. Projekto išeiga

3.16.1. Atviri klausimai

Projektuojant „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemą išaiškėja įvairios vietos, kurios taip ir vadinasi neturinčios sprendimo problemos arba klausimai dar neturintys atsakymo, tačiau tos vietos yra svarbios sistemai ir jos užtikrintam funkcionavimui.

Projektuojant tokio tipo sistema didelis faktorius lemiantis jos saugu veikimą yra saugumo užtikrinimas. Nors jam užtikrinti ir yra pasitelkiamos saugiausios bibliotekos, tačiau niekas negali būti 100% būti užtikrintu kad laikui bėgant jos veiks ant tiek gerai kaip tarkime šiandien. Todėl šioje vietoje yra paliktas toks atviras klausimas, prie kurio sprendimo bus grįžtama manau dar ne vieną kartą.

Kitas ne mažiau svarbus sprendimas, kuriam yra ieškoma atsakymo yra sistemos perkeliamumo. Kitaip sakant kokios yra galimybės ir priemonės numatyti sistemos vystimosi ir plečiamumo progresą. Nors sistema ir veiks, atlikinės vartotojų užklausas, tačiau perkeliant ją ant kito pagrindo atsiras sunkumų. Todėl ir yra sprendžiamas klausimas kaip sistema projektuoti labiau lanksčia, labiau universalia, kad vėliau ji galėtų būti tobulinama.

3.16.2. Egzistuojantys sprendimai

3.16.2.1. Pagamintos sistemos, kurios gali būti nupirktos

Projektas yra unikalus ir pasaulyje elektroninių apmokėjimų universitetinėje aplinkoje pilnai užbaigtų sistemų nėra daug, dažniausiai t.y.. įvairaus pobūdžio prototipai. Elektroninių mokėjimų universitetinėje aplinkoje sistemos į rinką nepatenka, todėl tokios sistemos kopijos faktiškai neįmanoma nusipirkti dėl saugumui keliamų reikalavimų, nes vargu ar kas norės parduoti savo sistemą, o vėliau turėti begales problemas dėl saugumo. Todėl kiekvienu atveju sistemos yra itin unikalios savo architektūra, nors atliekamo funkcijos yra itin panašios. Tačiau bendradarbiaujant universitetams, šiuo atveju Kauno Technologijos Universitetui su universitetais turinčiais elektroninių mokėjimų universitetinėje aplinkoje sistemas, būtų mokslo tikslais, inovacijų tikslais, ar gero bendradarbiavimo tikslais, būtų įmanoma įsigyti

šias sistemas, ar dalį jų teikiamų paslaugų, arba jų pilnas ar dalines specifikacijas, bei kitas sistemų sudedamąsias dalis. Priklausomai nuo bendradarbiavo tarp universitetų ir jų vidaus taisyklių, bei saugumo reikalavimų.

Pasaulyje yra nemažai koledžų ir universitetų, kurie naudodami naujausias technologijas yra sukūrę ir įsidięę bei naudoja apmokėjimo universitetinėje aplinkoje sistemas. Kurios teikia paslaugas nuo galimybės susimokėti už mokslą iki apsipirkinėjimų elektroninėse parduotuvėse. Tad toliau apžvelgsime geriausias elektroninių mokėjimų universitetinėje aplinkoje sistemas:

- *Electronics Shop and Pay (ESP) sistema*

Electronics Shop and Pay[2] (ESP) sistema Harvard universitete yra itin plačiau naudojama, kaip žinetinklio pagrindinis įrankis užsisakinėjant prekes bei paslaugas, bei jas apmokant, t.y.. atsiskaitant už prekes ir paslaugas su atitinkamais pardavėjais.

Sistema leidžia iš vienos vietos pasiekti daugelio pardavėjų svetaines, įskaitant pagrindines paslaugas, su mokslu susijusias paslaugas, WRM Scientific[3](t.y. elektroninė parduotuvė prekiaujanti pramonei, vyriausybei, mokslo įstaigoms, mokslininkams, elektronika, farmacijos įstaigoms ir pan. reikalinga laboratorine įranga ir priemonėmis). Visa informacija yra patalpinta specialiame pardavėjų sąrašė, kuris kitaip gali būti vadinamas Harvard-specifiniu katalogu, kuriame prekių kainos yra pateikiamos su ženklia nuolaida, nei tiesiogiai pas pardavėjus.

- *Financial Management Service(FMS) sistema*

Financial Management Service[5] (FMS) sistema yra itin plačiai naudojama Indiana universitete. Tai yra įžymiausia sistema tarp universitetinių apmokėjimo sistemų, ir gal būt tobuliausia.

FSM sistema veikia žinetinklio pagrindu, todėl visos sistemos teikiamos paslaugos yra prieinamos per naršyklę. Esminis šios sistemos bruožas yra tas, kad sistema teikia itin daug paslaugų sistemos vartotojams, todėl yra sukurti specialūs servisai skirti aptarnauti vartotojus.

- *E-payment sistema*

E-payment[7] yra apmokėjimo sistema taikoma Trent universitete. Šios universiteto studentai gali susimokėti sąskaitas už mokslą būdami namuose ar kokioje nors kitokioje buveinėje naudodamiesi internetu.

Studentai gali sumokėti mokesčius naudodamiesi, bet kokia populiaria atsiskaitymo sistema, nes visos populiarios finansines paslaugas teikiančios sistemos yra integruotos Trent universiteto IT skyriaus į E-payment sistemą.

Taip pat galima laisvai atsiskaityti naudojantis Visa, MASTERCARD aptarnaujamas American Express aptarnavimo centru.

- *Stanford ePay sistema*

Stanford ePay[9] yra Stanford universiteto elektroninio sąskaitų išrašymo ir elektroninio apmokėjimo sistema. Sistema teikia galimybę studentams ar autorizuotiems mokėtojams pamatyti savo universitetines sąskaitas ir jas apmokėti. Mokėtojai apmokėti sąskaitas gali 24 valandas per parą, t.y.. bet kuriuo paros metu.

Universitetinę sąskaitą sudaro: mokestis už mokslą, nuoma, maitinimas, sveikatos draudimas, medicininė priežiūra, gyvenamosios vietos mokesčiai, pašto paslaugos, kabelinės televizijos paslaugos. Tačiau negalima susimokėti baudos už negražintas laiku į biblioteką ar pamestas knygas, parkingo paslaugas, įvairių renginių bilietus, knygas.

- *SDePay sistema*

SDePay[10] yra elektroninių sąskaitų pateikimo ir elektroninio apmokėjimo sistema. Ši sistema aptarnauja visus studentus ir jų sąskaitas ir sąskaitų apmokėjimus, kurie mokosi South Dakota State universitete ir South Dakota Public universitete.

Studentams visos Universitetinės sąskaitos yra pateikiamos tik SDePay sistemoje, o elektroniniu paštu gauna tik pranešimą, apie atėjusias sąskaitas. Sąskaitos gali būti: mokestis už mokslą, maitinimą, gyvenamąją vietą, honorarai ir t.t..

- *Enroll & Pay sistema*

Enroll & Pay[12] yra Kansas universiteto elektroninių sąskaitų pateikimo ir elektroninio apmokėjimo sistema.

Studentai naudojantys Enroll & Pay sistema gali peržiūrėti jiems pateiktas sąskaitas ir apmokėti tiesiogiai sistemoje. Sąskaitos gali būti: mokestis už mokslą, gyvenamąją vietą, nuomą, baudos už laiku negražintas knygas ir pan.). Sąskaitas galima apmokėti kreditinėmis kortelėmis, čekiais, ar populiariomis elektroninio apmokėjimo paslaugas teikiančiomis sistemomis.

- *Tupay sistema*

TUpay[14] yra oficiali sąskaitų išrašymo ir apmokėjimų sistema aktyviems-įtrauktiems į sąrašą studentams Temple universitete. Sąskaitų tvirtinimai gali būti atliekami tik realiu laiku. Studentas gauna e-laišką pranešimą, nes tai oficiali Temple universiteto sistema vartotojus informuoja kad jų e-sąskaitos yra nagrinėjamos. Prie TUpay sistemos galima prieiti adresu[15] arba prisijungti prie OWLnet tiesiogiai[16]. Galima mokėti realiu laiku elektroniniais pinigais be tikrinimo arba išsaugojimų arba kreditine kortele.[14]

- *John Hopkins universitetas elektroninio apmokėjimo sistemą*

John Hopkins universitetas pristato e-sąskaitų apmokėjimo sistemą. Elektroninė sąskaita yra oficiali priemonė generuoti sąskaitas už mokslą visiems Peabody studentams. Tai reiškia, kad popierinės sąskaitos nėra siunčiamos paštu užsiregistravusiems studentams, nebent specialiai reikia[17].

- *Oregonos universitetas QuickPay elektroninio apmokėjimo sistema*

Oregonos universitetas naudoja QuickPay[18], elektroninę sistemą kad visi studentai galėtų atsispausdinti jų sąskaitas, atlikti apmokėjimus realiu laiku, nustatyti turinčius asmenys kaip autorizuotus mokėtojus, kurie gali daryti tą patį. Per QuickPay sistemą, elektroninis laiškas siunčiamas kiekvienam studentui į oficialu universiteto elektroninį paštą[18].

3.16.2.2. Pagaminti komponentai, kurie gali būti panaudoti

Projektas yra unikalus ir pasaulyje elektroninių apmokėjimų universitetinėje aplinkoje pilnai užbaigtų sistemų nėra daug, dažniausiai t.y.. įvairaus pobūdžio prototipai. Elektroninių mokėjimų universitetinėje aplinkoje sistemos į rinką nepatenka, todėl tokios sistemos kopijos faktiškai neįmanoma nusipirkti dėl saugumui keliamų reikalavimų, nes vargu ar kas norės parduoti savo sistemą, o vėliau turėti begales problemas dėl saugumo. Todėl kiekvienu atveju sistemos yra itin unikalios savo architektūra, nors atliekamos funkcijos yra itin panašios. Tačiau bendradarbiaujant universitetams, šiuo atveju Kauno Technologijos Universitetui su universitetais turinčiais elektroninių mokėjimų universitetinėje aplinkoje sistemas, būtų mokslo tikslais, inovacijų tikslais, ar gero bendradarbiavimo tikslais, būtų įmanoma įsigyti šias sistemas, ar dalį jų teikiamų paslaugų, arba jų pilnas ar dalines specifikacijas, bei kitas sistemų sudedamąsias dalis. Priklausomai nuo bendradarbiavo tarp universitetų ir jų vidaus taisyklių, bei saugumo reikalavimų. Jai kas nors iš sukurtų sistemų būtų įsigyta (elektroninių mokėjimų sistemų sąrašas su apibūdinimais pateiktas „3.16.2.1. Pagamintos sistemos,

kurios gali būti nupirktos“ skyrelyje), tokiu atveju būtų panaudotos įsigytų sistemos ar jų dalys, kuriant „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamą sistemą.

Kuriant ir eksploatuojant „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamą sistemą yra itin svarbu užtikrinti gerą šios sistemos saugumą. Nes neužtikrinus tinkamo šios sistemos saugumo, jos nebus galima eksploatuoti. Siekiant užtikrinti kuriamos sistemos saugumą, bus panaudoti Microsoft kompanijos sukurti programinės įrangos eksploatavimo saugumą užtikrinantys saugos bibliotekų paketai. Nes šie paketai laidžia užtikrinti itin didelį sistemos saugumo lygį, kas šiuo atveju yra labai aktualu. Toliau pateikiami saugos bibliotekų paketai, su trumpais apibūdinimais.

System.Web.Security paketas

System.Web.Security paketą[50] - sudaro klasės, kurios yra skirtos realizuoti ASP.NET saugumą žiniatinklio serviso aplikacijose.

Membership klasė yra naudojama ASP.NET aplikacijose patikrinti vartotojo kredencialus ir valdyti vartotojo parinktis, tokias kaip slaptažodis ir elektroninio pašto adresas. *Roles* klasė suteikia valdymo autorizaciją aplikacijos pagrindinėms vartotojų grupėms su paskirtomis teisėmis į žiniatinklio aplikacijas.

Abi *Membership* ir *Roles* klasės dirba su aprūpintu, klasės kreipiasi į jį prašydamos duomenų iš duomenų saugyklos, kad susigražintų narystę (membership) ir teisią (role) į informaciją. Narystės (membership) ir teisių (role) informacija gali būti išsaugota Microsoft SQL Server duomenų bazėje panaudojant [SqlMembershipProvider](#) ir [SqlRoleProvider](#) klases; Aktyviojoje direktorijoje (Active Directory) panaudojant [ActiveDirectoryMembershipProvider](#) ir [AuthorizationStoreRoleProvider](#) klases, ar parinktų duomenų šaltinyje tam panaudojant [MembershipProvider](#) ir [RoleProvider](#) klases.

Vartotojas sukonfigūruoja ASP.NET narystę (membership) panaudodamas [membership Element \(ASP.NET Settings Schema\)](#). Kada aplikacija naudoja narystės (membership) priėjimą, ASP.NET sukuria *Membership* klasę, kurią galima panaudoti užklausančiam narystės (membershi) informacijos. Aprūpintojas specifikuojamas įgyvendinant [MembershipUser](#) klasę su turima informacija apie vartotojus prie prieinamo puslapio. Taip pat leidžiamas pasirinktinis *MembershipUser* klasės panaudojimas aplikacijose.

ASP.NET teises (roles) galima sukongfigūruoti panaudojant [roleManager Element \(ASP.NET Settings Schema\)](#). ASP.NET sukuria *ROLES* turinčią informaciją apie teisių (role) narystę (membership) einamojo vartotojo.

ASP.NET aprūpintojo serviso kontrolė yra interaktyvi su *Membership* klase ir *Roles* klase. [Login](#), [CreateUserWizard](#), ir [ChangePassword](#) kontrolė dirba su *Membership* klase, kad būtų supaprastintas sukūrimas žiniatinklio aplikacijos autentifikacijos, ir [LoginView](#) kontroliuoja naudojamus teisių (role) šablonus, priskiriant vartotojus specifinėms žiniatinklio vartotojų grupėms.

- *System.NET Security paketas*

System.NET Security paketas[51] - aprūpina tinklo srautų komunikaciją tarp nutolusių tinklo taškų.

- *System.Security paketas*

System.Security paketas[52] – aprūpina realaus laiko saugumo struktūromis sistema, įskaitant pagrindines klasių teises.

- *System.Security.AccessControl paketas*

System.Security.AccessControl paketas[53] – aprūpina programos elementus kontroliniais leidimais, ir audituoja saugumą susijusių saugių objektų.

- *System.Security.Authentication paketas*

System.Security.Authentication paketas[54] – suteikia enumeracijos aprašus saugumo ryšiams. Į šią enumeraciją įeina: [CipherAlgorithmType](#), [ExchangeAlgorithmType](#), [HashAlgorithmType](#), ir [SslProtocolType](#).

- *System.Security.Cryptography paketas*

System.Security.Cryptography paketas[55] – suteikia kriptografijos servigus, įskaitant saugų duomenų kodavimą ir dekodavimą, ir daug kitų operacijų, tokių kaip hašingavimas, atsitiktinių skaičių generavimas, ir pranešimų autentifikacija ir t.t.

- *System.Security.Cryptography.Pkcs paketas*

System.Security.Cryptography.Pkcs paketas[56] – suteikia programuojamiems elementams Viešojo rakto kriptografijos standartą (PKCS), įskaitant metodus duomenų

parašui, raktų apsiketimui, sertifikatų prašymus, viešojo rakto kriptografiją ir dekriptografiją, ir kitas saugumo funkcijas.

- *System.Security.Permissions paketas*

System.Security.Permissions paketas[57] – aprašo klasių kontrolės teisių operacijoms ir resursų polisams.

- *System.Security.Policy paketas*

System.Security.Policy paketas[58] – apima kodo grupes, narystės (membership) sąlygas, ir požymius. Šie trys klasių tipai naudojami sukuriant taisykles pagal vykdomą saugumo polisą sistemoje. Akivaizdus klasės indėlis į saugumo polisą ir narystės (membership) sąlygas yra komutuojamas; kartu sukuriant poliso sąlygas ir suteikiant leidimus. Poliso ygiai ir kodo grupės yra struktūrinės polisų hierarchijoje. Kodo grupės inkapsuluotos taisyklių ir sutvarkytos polisų hierarchijos.

- *System.Security.Principal paketas*

System.Security.Principal paketas[59] – aprašo pagrindinius objektus, kurie atstovauja saugumo kontekstą pagal kodo vykdymą.

- *System.Security.RightsManagement paketas*

System.Security.RightsManagement paketas[60] – suteikia klasių teisių valdymo sukuriant aplikacijos kontekstą tokį, kuris yra išsaugomas [Package](#) ar [XpsDocument](#).

3.16.2.3. Galimas pakartotinas panaudojimas

Yra nemažai elektroninių mokėjimų universitetinėje aplinkoje sistemų, tačiau jas nusipirkti ir panaudoti pakartotinai kažką sutaupius yra sudėtinga dėl saugumui keliamų reikalavimų, nes vargu ar kas norės perduoti savo sistemą, o vėliau turėti begales problemas dėl saugumo. Tačiau yra aptikta nemažai sistemų tokio pobūdžio, kurios būtų tinkamos jeigu jas perpanaudoti. Kadangi beveik visos jos jeigu apibendrint turi panašias, o kartais ir tokias pat funkcijas, kas liečia architektūros, tai ji gali ir skirtis.

Toliau apžvelgsime elektroninių mokėjimų universitetinėje aplinkoje sistemas, kurios galėtų būti panaudotos pakartotinai:

- *Harvard's Electronic Shop and Pay*

Electronics Shop and Pay[2] (ESP) sistema Harvard universitete yra plačiau naudojama, kaip žiniatinklio pagrindinis įrankis užsisakinėjant prekes bei paslaugas, bei jas apmokant, t.y.. atsiskaitant už prekes ir paslaugas su atitinkamais pardavėjais.

Sistema suteikia galimybę iš vienos vietos pasiekti daugelio pardavėjų svetaines, įskaitant pagrindines paslaugas, su mokslu susijusias paslaugas, WRM Scientific[3] (t.y. elektroninė parduotuvė prekiaujanti pramonei, vyriausybei, mokslo įstaigoms, mokslininkams, elektronika, farmacijos įstaigoms ir pan. reikalinga laboratorine įranga ir priemonėmis).

- *Financial Management Service*

Financial Management Service[5] (FMS) sistema yra plačiai naudojama Indiana universitete. Tai yra įžymiausia sistema tarp universitetinių apmokėjimo sistemų, ir gal būt tobuliausia. FSM sistema veikia žiniatinklio pagrindu, todėl visos sistemos teikiamos paslaugos yra prieinamos per naršyklę. Esminis šios sistemos bruožas yra tas, kad sistema teikia itin daug paslaugų sistemos vartotojams, todėl yra sukurti specialūs servisai skirti aptarnauti vartotojus.

- *E-payment*

E-payment[7] yra apmokėjimo sistema taikoma Trent universitete. Šios universiteto studentai gali susimokėti sąskaitas už mokslą būdami namuose ar kokioje nors kitokioje buveinėje naudodamiesi internetu.

Studentai gali sumokėti mokesčius naudodamiesi, bet kokia populiaria atsiskaitymo sistema, nes visos populiarios finansines paslaugas teikiančios sistemos yra integruotos Trent universiteto IT skyriaus į E-payment sistemą.

Taip pat galima laisvai atsiskaityti naudojantis Visa, MASTERCARD aptarnaujamam American Express aptarnavimo centru.

- *Stanford ePay*

Stanford ePay[9] yra Stanford universiteto elektroninio sąskaitų išrašymo ir elektroninio apmokėjimo sistema. Sistema teikia galimybę studentams ar autorizuotiems mokėtojams pamatyti savo universitetines sąskaitas ir jas apmokėti. Mokėtojai apmokėti sąskaitas gali 24 valandas per parą, t.y.. bet kuriuo paros metu.

- *SdePay*

SDePay[10] yra elektroninių sąskaitų pateikimo ir elektroninio apmokėjimo sistema. Ši sistema aptarnauja visus studentus ir jų sąskaitas ir sąskaitų apmokėjimus, kurie mokosi South Dakota State universitete ir South Dakota Public universitete.

SDePay sistema turi iš tiek daug galimybių, skirtų peržiūrėti ir apmokėti sąskaitas, be to šia sistema itin paprasta naudotis. Todėl nenuostabu, kad ši sistema susilaukė itin didelio dėmesio iš vartotojų, ir dabar paprasto – popierinės sąskaitos nebespausdinamo ir nebesiuntinėjamos studentams ir jų rėmėjams.

- *Enroll & Pay*

Enroll & Pay[12] yra Kansas universiteto elektroninių sąskaitų pateikimo ir elektroninio apmokėjimo sistema. Kansas Universiteto Enroll & Pay sistema kol kas gali pateikti tik universitetinis sąskaitas, ir priimti už jas apmokėjimus, bet planuojama, kad ateityje, studentai turės galimybę apmokėti ir kitokio pobūdžio sąskaitas, užsisakyti įvairias paslaugas.

- *Temple universiteto TUpay sistema*

Tupay[14] yra oficiali sąskaitų išrašymo ir apmokėjimų sistema aktyviems-įtrauktiems į sąrašą studentams Temple universitete. Sąskaitų tvirtinimai gali būti atliekami tik realiu laiku. Sąskaitos apmokamos per TUpay: apmokėjimas priimamas e-čekiu arba kreditinė kortelė; E-čekis yra greitas ir saugus metodas elektroniniu būdu apmokėti e-sąskaitą.

- *John Hopkins universiteto elektroninio apmokėjimo sistema*

John Hopkins universitetas pristato e-sąskaitų apmokėjimo sistemą[17]. Elektroninė sąskaita yra oficiali priemonė generuoti sąskaitas už mokslą visiems Peabody studentams. Tai reiškia, kad popierinės sąskaitos nėra siunčiamos paštu užsiregistravusiems studentams, nebent specialiai reikia.

- *Oregono universiteto elektroninio apmokėjimo sistema QuikPAY*

Oregonos universitetas naudoja QuickPay[18], elektroninę sistemą kad visi studentai galėtų atsispausdinti jų sąskaitas, atlikti apmokėjimus realiu laiku, nustatyti turinčius asmenys kaip autorizuotus mokėtojus, kurie gali daryti tą patį. Per QuickPay sistemą, elektroninis laiškas siunčiamas kiekvienam studentui į oficialu universiteto elektroninį pašta.

Apžvelgę trumpai egzistuojančius sprendimus tinkančius pakartotiniam panaudojimui, matome kad yra sukurta įvairiausių variantų, žinoma juos panaudojus būtų tikrai sutaupyta laiko, be to ir kitokių resursų.

3.16.3.Naujos problemos

3.16.3.1. Problemos diegimo aplinkai

„Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos diegimas gali sukelti tam tikras problemas. Pagrindinės problemos, kurias gali kelti pastarosios sistemos įdiegimas, yra susijusios su įdiegimo aplinkoje egzistuojančiomis sistemomis. Toliau pateikiamos „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos įtakos jos diegimo aplinkai:

- *Operacinės sistemos problema.* „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistema bus pritaikyta MS Windows operacinei sistemai, todėl serveryje(-iuose), ir kompiuteriuose (buhalterių, pardavėjų/teikėjų) turėtų būti naudojama MS Windows operacinė sistema. Taip pat MS Windows operacinė sistema neturėtų būti senesnė, nei MS Windows 2000, priešingu atveju gali kilti tam tikrų problemų, nes kai, kurie komponentai gali neveikti.
- *Duomenų bazės problema.* „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistema siekiant, kad sėkmingai ir našiai bei saugiai dirbtų, ir tinkamai saugotų ir apdorotų duomenis, turėtų būti įdiegtas duomenų bazės valdymo MS SQL Server serveris, kiek įmanoma naujesnės versijos. Kadangi „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistema bus pritaikyta būtent šio tipo duomenų bazės valdymo serveriui. Nes, duomenų bazės valdymo MS SQL Server serveris puikiai užtikrina didelį duomenų patikimumą, saugumą, greitą duomenų apdorojimą, ir pan., kas eksploatuojant „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemą yra labai svarbu.
- *Programinėje įrangoje naudojamos technologijos problema.* „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistema bus sukurta naudojant pažangiausią šių dienų programinės įrangos kūrimo technologiją MS .NET Framework, kadangi ši programinė įranga užtikrina su ja sukurtų produktų puikų funkcionavimą. MS .NET Framework technologija yra nuolat atnaujinama, todėl šioje technologijoje atlikti atnaujinimai „Elektroninių

mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos kūrimo metu bus automatiškai įtraukti į kuriamą sistemą. Todėl diegimo aplinkoje (serveryje, buhalterijų ir pardavėjų/teikėjų kompiuteriuose) turėtų būti įdiegta MS .NET Framework technologijos naujausia versija su visais atnaujinimais, tam, kad būtų užtikrintas sėkmingas „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos darbas.

- *Internetinės naršyklės problema.* „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos vartotojų sąskaitų informaciją bus galima peržiūrėti ir per interneto naršyklę. Todėl vartotojai galės naudotis, bet kokia operacine sistema. Tačiau vartotojai turėtų turėti naršyklės palaikančias ASP technologiją. Priešingu atveju vartotojai šia peržiūros galimybe gali negalės pasinaudoti.
- *Kompiuterinio tinklo problema.* Kauno Technologijos Universitete egzistuoja išvystytas kompiuterinis tinklas, tačiau siekiant kokybiškai įdiegti sistemą, ir kad ji galėtų saugiai ir patikimai veikti, gali tekti pertvarkyti. Nes patikimas sistemos veikimas ir geras saugumo užtikrinimas yra kritiškai svarbus, nes jai nebus užtikrintas sistemos patikimas veikimas ir saugumas, jos eksploatacija galės būti nutraukta. Todėl siekiant tai užtikrinti (patikimą sistemos veikimą ir saugumą), gali tekti pertvarkyti, kai kuriuos Kauno Technologijos Universiteto kompiuterinio tinklo komponentus, ar patį tinklą papildyti naujais komponentais.

3.16.3.2. Įtaka jau instaliuotoms sistemoms

„Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos programinė įranga neturėtų daryti žymios įtakos jau instaliuotoms sistemoms. Tačiau minimali įtaka yra neišvengiama. Toliau pateikiamos charakteristikos „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojama sistema įtakos jau instaliuotas sistemas:

- Kadangi „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos programinė įranga bus padaryta su MS .NET Framework technologija, ši technologija neapkrauna sistemos (gal tik šiek tiek, tačiau t.y.. visiškai nepastebima, nebent sistema yra sąlyginai labai silpna, kita vertus tokia silpna sistema yra netinkama šios technologijos ir su ja sukurtos programinės įrangos eksploatavimui, ir ją reikia keisti). Naudojant MS .NET Framework technologija

yra reikalaujamą naudoti, saugumo sumetimais naujausią versiją su visais atnaujinimais. Todėl jai bus įdiegta pastarosios technologijos senesnė versija, ją teks atnaujinti iki naujausios versijos su visais atnaujinimais. Periodinis MS .NET Framework technologijos atnaujinimas žalos nedaro, nes kiekviena naujesnė versija turi visas senesnės versijos funkcijas, o, be to, kiekviena nauja versija ir atnaujinimai pašalina senosios versijos trūkumus, tuo pačiu ir „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos trūkumus, susijusius su MS .NET Framework technologija.

- „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistema nekenks kitoms įdiegtoms programoms, tačiau saugumo sumetimais, gali tekti pašalinti kai kurias programas, kurios gali kelti saugumo problemas.
- „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistema neapkraus sistemos į kurią ji bus įdiegta, nebent ja iškart naudotųsi labai daug vartotojų, tokiu atveju kiltų sistemos apkrautumo problemos.
- „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistema nuolat (periodiškai) bendraus su Kauno Technologijos Universiteto akademinė informacinė sistema. Iš jos bus atnaujinti: studentų, darbuotojų, fakultetų, katedrų, grupių sąrašai, susijusi informacija ir t.t. Todėl kuriama sistema gali padidinti egzistuojančio Kauno Technologijos Universiteto akademinės informacinės sistemos apkrautumą.
- „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistema padidins serverio apkrautumą, į kurią ji bus įdiegta, ir su ją tiesiogiai susijusio kompiuterinio tinklo dalie apkrautumą. Todėl teks pagerinti centrinio serverio ir su juo glaudžiai susijusios kompiuterinės dalies techninius duomenis.
- „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistema klientai-vartotojai šia sistema galės naudotis tik per interneto naršyklę, todėl jų naudojamai programinei įrangai jokios neigiamos įtakos neturės. (Buhalteriai ir pardavėjai šia sistema naudosis ir per interneto naršyklę ir klasikiniu būdu.)

3.16.3.3. Neigiamas vartotojų nusiteikimas

Neigiamas vartotojo nusiteikimas galėtų kilti tokiuose vietose kaip; *sistemos greیتaveika, menių išdėstymas, struktūra, pasiekiamumas, saugumas, sistemos galimybe atsistatyti po neigiamų padarinių.*

Toliau aptarsime juos detaliau:

- *Sistemos greیتaveika,*

Tai yra ta silpnoji sistemos vieta, kur dažniausiai jaustųsi trukumas. Kodėl gi taip atsitinka, na tai yra susiję su dviem parametrais; aptarnaujamų vartotojų kiekių duotuoju momentu ir siunčiamų užklausų kiekiu. Kitaip sakant sistema veikia gerai tol kol prie jos prisijungusių vartotojų skaičius neviršija nustatyto, kai tik jis išauga nes jį sunku numatyt, sistemos veikimas sulėtėja. Taip pat galimas vartotojų nepasiteikimas dėl pačios prieigos prie interneto, kadangi kaip žinoma ne visi interneto tiekėjai turi užsitikrinę greitą internetą. Žinoma yra dėdamos pastangos kad tas vietas pagerinti. Pritaikomos efektyvesnės technologijos, priemonės.

- *Menu išdėstymas, struktūra*

Sistemos komponentų išdėstymas, langų struktūra, grafinė sąsaja be abėjo turėjo ir turi itakos sistemos vartotojų nusiteikimui. Kadangi ne visiems vartotojams patinka vienoks arba kitoks sistemos išdėstymas ir tai gali vartotojui sukelti neigiamų emocijų. Tačiau bandoma kūrėti daugiau mažiau patogų interfeisą ir išdėstymą.

- *Saugumas*

Sistemos saugumo niekada nebus per daug, kadangi kartais projektuotojai negali nuspėti vartotojų mastymo ir apgalvoti vienokios ar kitokios situacijos sprendimo jai neatsitikus. Vartotojai gali sugalvoti įvairiausių būdų, kaip kad prisijungus prie savo apsaugotos srities bandyti išėiti iš jos spaudžiant *back* mygtuką *explorerio* lange. To žinoma sistema neleis daryt nes tai kenkia saugumui, bet vartotojui tai gali pasirodyti nepatogu, nes jis visada buvo pripratęs būtent taip elgtis. Bet tokios vietos yra išsprendžiamos tiesiog pačioje sistemoje sukūrus mygtuką *back*.

3.16.3.4. Kliudantys diegimo aplinkos apribojimai

Galimos problemos susijusios su diegimo aplinkos apribojimais yra; *nepakankami sistemos techninės dalies resursai, diegimo aplinkos pasikeitimas, aplinkos sąlygos.*

- *Nepakankami sistemos techninės dalies resursai,*

Tai yra susiję su aparatūrine dalimi, kai serveris yra nepakankamai galingas kad galėtų aptarnauti tam tikru momentu galbūt padidėjusi vartotojų skaičių. Tuomet tokia aplinka tampa kliudanti sistemai tinkamai veikti. Dar vienas techninės dalies apribojimas galėtų būti pačio vartotojo naudojamos sistemos greیتaveika, kai jo naudojama sistema lėtai veikia dėl jo turimų kompiuterinės dalies komponentų vėlinimo.

- *Diegimo aplinkos pasikeitimas*

Kai sistema yra perkeliama į kitą veikimui skirtą aplinką, ir tada kyla aibė klausimų ar tai ne kliūtis sistemos veikimui. Dažniausiai tai būna susiję kaip su sistemos diegimo klausimais taip ir su internetinio ryšio apribojimais.

- *Aplinkos sąlygos*

Galimos kliūtys susijusios su aplinkos kur bus diegiama sistema sąlygom, t.y.. drėgmės daug turinti aplinka, sulieta, daug magnetinių laukų. Dėl šitų išvardintų veiksmų sistemai veikiančiai tokioje aplinkoje gali tekti veikti kritinėmis sąlygomis, dėl ko gali žymiai kristi jos greیتaveika.

3.16.3.5. Galimos naujos sistemos sukeltos problemos

Galimos naujos sistemos sukeltos problemos gali būti susijusios su; *virtotojų aptarnavimo trukme, sistemos techniniu pasiruošimu, programinės įrangos išbaigtumu:*

- *Vartotojų aptarnavimo trukme, sistemos techninis pasiruošimas*

Šitas punktas susijęs taipogi su aparatūrine dalimi, kai serveris yra nepakankamai galingas kad galėtų aptarnauti tam tikru momentu galbūt padidėjusį vartotojų skaičių. Tuomet tokia aplinka tampa kliudanti sistemai tinkamai veikti. Bet šitas rodiklis visą laiką yra plaukiojantis, nes jis priklausomas nuo vartotojų skaičiaus. Todėl sistemos techninis pasiruošimas gali būti pakankamas tik tam tikrom sąlygomis esant, kitokiom jis bus apribotas.

- *Programinės įrangos išbaigtumas*

Susijęs su programinio kodo efektyviu panaudojimu kad kuo mažiau naudotu sistemos aparaturinių resursų, optimizuotas naudojimas ir programinio kodo eilučių mažinimas, kad viskas būtų greičiau vykdoma, išlaikant tą pačią funkcionalumo esmę.

3.16.4.Sistemos vystymo etapai

Siekiant, kad projektas „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ būtų atliktas ir elektroninių mokėjimų universitetinėje aplinkoje sistema būtų sukurta kuo greičiau ir perspektyviau yra numatyti sistemos vystymo etapai, kurie yra pateikti Lentelė 3.16-1.

Lentelė 3.16-1. Sistemos vystymo etapai.

Eil. Nr.	Fazės pavadinimas	Komponentai
1.	Duomenų bazės kūrimas	Duomenų bazė
2.	Serverio posistemis	Serverio posistemio komponentas
3.	Buhalterio posistemis	Buhalterio posistemio komponentas
4.	Pardavėjo\ Teikėjo posistemis	Pardavėjo/ Tekėjo posistemio komponentas
5.	Kliento-vartotojo posistemis	Kliento-vartotojo posistemio komponentas

3.16.5.Pritaikymas

3.16.5.1. Reikalavimai esamų duomenų perkėlimui

Lentelė 3.16-2. Reikalavimai esamų duomenų perkėlimui.

Eil. Nr.	Pritaikymo veikos (darbu) sąrašas
1.	Sistemai diegti reikalinga aparatūrinė įranga
2.	Aparatūrinės dalies konfigūravimas
3.	Aparatūrinės dalies patikrinimas
4.	Reikalingų programinių įrangai atnaujinimų gavimas
5.	Atnaujinimų diegimas
6.	Programinės įrangos diegimas
7.	Testavimas
8.	Kilusių nesklandumų likvidavimas
9.	Pakartotinis diegimas(esant reikalui)

Sistemai diegti reikalinga aparatūrinė įranga – kadangi sukurta sistema turi veikti ant kažkokio techninio pagrindo, t.y. ją turi kažkas vykdyti, tam yra naudojama techninė, aparatūrinė įranga. Šio etapo metu ta reikalinga įranga yra surandama ir panaudojama programinės įrangos įdiegimui.

Aparatūrinės dalies konfigūravimas – žinant, kad programinė įranga gali turėti vienokių ar kitokių pasekmių diegiant ją į aparatūrinę dalį, dažnai prireikia techninės dalies pakeitimų.

Aparatūrinės dalies patikrinimas – tam, kad nereiktų tuščiai ir akiai paleidinėti programinės įrangos, galimas žingsnis aparatūrinės įrangos tiesiog patikrinimas po jos konfigūravimo.

Reikalingų programinių įrangai atnaujinimų gavimas – šis etapas reikalingas tam, kad surandame atnaujinimų, kad programinė įranga, kuri bus reikalinga geresniam suderinamumui.

Atnaujinimų diegimas – vykdomas rastų atnaujinimų diegimas tam, kad programinė įranga veiktų kuo geriau ant techninio pagrindo.

Testavimas – testavimo etapas tiesiog būtinas, kad patikrinti programinės įrangos sklandų veikimą, techninio pagrindo atžvilgiu su įdiegtais atnaujinimais.

Kilusių nesklaidumų likvidavimas – ištestavus dažnai atsiranda prieš tai nepastebėtos klaidos, todėl jos likviduojamos šio etapo metu.

Pakartotinis diegimas(esant reikalui) – šis etapas tam atvejui, jeigu kartais nepavyktų viskas sklandžiai iš pirmo karto, nes praktika rodo kad dažnai jis praverčia.

3.16.5.2. Reikalingas duomenų transformavimas perkeliant į naują sistemą

Pateikiamas duomenų transformavimo uždavinių sąrašas bei identifikuojami duomenys, kurie turės būti perkelti į naują „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemą. Kai tik papildomas terminų žodynas, galima kelti klausimą, kur šią sąvoką charakterizuojantys duomenys bus saugomi ir ar nauja sistema įtakos jų struktūrą, ar bus reikalingas transformavimas?

Pateikiama reikalingų duomenų transformavimas keliant į naują sistemą susideda iš duomenų apie:

- vartotojų prisijungimo sesijas(sesijų informacija, duomenys)
- vartotojų asmeninės sąskaitas(apmokėtas, neapmokėtas)
- vartotojų prisijungimo informacijos(slaptažodžiai, prisijungimo vardai)
- vartotojų atliktų veiksmų istorijas
- gauti pranešimai, išsiųsti pranešimai
- prisijungimo srities nustatymo parametrų informacija
- su tinklo struktūra susiję
- reikalingi tinkliniam bendradarbiavimui
- sistemos diegimui reikalingi duomenys

- informacija apie reikalingus diegimui įrenginius

3.16.6. Rizikos

3.16.6.1. Galimos sistemos kūrimo rizikos

Visi projektai yra rizikingi, bet skirtingu laipsniu, ne išimtis ir „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektas. Todėl valdant projektą būtina numatyti būdus, kaip kuo anksčiau identifikuoti momentus, kuomet rizika pradeda kelti problemas. Todėl pateikiamas sąrašas rizikingų faktorių (Lentelė 3.16-3.), šalia pateikiant tikimybinį įvertinimą, kad rizika sukels problemas.

Lentelė 3.16-3. Sistemos kūrimo rizikos.

Eil. Nr.	Rizikos faktorius	Tikimybė	Įtaka
1.	Projekte dalyvaujančių personalo patirtis ir sugebėjimai.	Vidutiniška	Rimta
2.	Reikalavimų pasikeitimas.	Vidutiniška	Rimta
3.	Architektūros pasikeitimas.	Vidutiniška	Rimta
4.	Bendradarbiaujančių sistemų sąsajos pasikeitimai.	Žema	Rimta
5.	Poreikis kardinaliai pakeisti sistemą ar sistemos posistemę kai ji jau sukoduota.	Vidutiniška	Rimtas
6.	Pakartotinai naudojami komponentai su klaidom, prasta dokumentacija.	Vidutiniška	Rimta
7.	Neefektyvios CASE priemonės.	Žema	Rimta
8.	Techninės įrangos gedimas.	Vidutiniška	Leistina
9.	Techninės įrangos pristatymo vėlavimas.	Žema	Leistina
10.	Programinė įranga neveikia vartotojo kompiuteriuose.	Žema	Rimta
11.	Programinė įranga neveikia vartotojų naršyklėse.	Žema	Rimta
12.	Projekto personalo pasitraukimas dėl ligos ir pan.	Žema	Leistina

3.16.6.2. Atsitiktinumų (rizikų) valdymo planas

Projekto „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ rizikos faktoriai ir numatomi planai problemoms spręsti (rizikos sprendimo būdai) pateikti Lentelė 3.16-4.

Lentelė 3.16-4. Sistemos kūrimo atsitiktinumų (rizikų) valdymo planas.

Eil. Nr.	Rizikos faktorius	Problemos sprendimas
1.	Projekte dalyvaujančių personalo patirtis ir sugebėjimai.	Pasirinkti ir duot išspręsti personalui tuos uždavinius, su kuriais jie jau yra susidūrę, turi nors mažiausios patirties. Jei duodami nauji uždaviniai, reikia skirti pakankamai daug laiko įsigilinimui į probleminę sritį.
2.	Reikalavimų pasikeitimas.	Reikia su užsakovu derinti reikalavimus ir nuo pradžių juos suskirstyti pagal prioritetus. Nesvarbius reikalavimų pasikeitimus galima atidėti sekančiai programinės įrangos versijai. Ir aišku pasilikti pakankamą laiko rezervą svarbiems reikalavimų pakeitimams įgyvendinti.
3.	Architektūros pasikeitimas.	Numatyti laiko rezervą projekto plane. Naudoti CASE įrankius.
4.	Bendradarbiaujančių sistemų sąsajos pasikeitimai.	Kurti bendradarbiaujančią sistemą wrapper komponentą, tuo minimaliai sumažinant tiesioginį bendravimą su bendradarbiaujančiomis sistemomis.
5.	Poreikis kardinaliai pakeisti sistemą ar sistemos posistemę kai ji jau sukoduota.	Bandyti išsiaiškinti su užsakovu, ar tokie pakeitimai tikrai labai svarbūs ir ar jų negalima atidėti sekančiai programos versijai. Reikia derinant reikalavimus naudoti oficialius dokumentus ant kurių būtų užsakovo parašas rodantis, kad jis su tinka su reikalavimais ir jei jie keisis tai bus skiriama

		daugiau lėšų ir/arba laiko jiems įvykdyti.
6.	Pakartotinai naudojami komponentai su klaidom, prasta dokumentacija.	Pasirinkti patikimus komponentus.
7.	Neefektyvios CASE priemonės.	Rinktis labiau žinomas CASE priemones, numatyti alternatyvias CASE priemones.
8.	Techninės įrangos gedimas.	Numatyti rezervinę techninę įrangą. Jei tokia rizika didelė, papildomai skirti laiko tos įrangos pakeitimui, perėjimui iš vienos darbi vietos į rezervinę.
9.	Techninės įrangos pristatymo vėlavimas.	Naudoti techninės įrangos emulavimą.
10.	Programinė įranga neveikia vartotojo kompiuteriuose.	Numatyti vartotojo kompiuteriuose naudojamą programinę įrangą, pasiūlyti įsidiegti atnaujinimus, naujas versijas.
11.	Programinė įranga neveikia vartotojų naršyklėse.	Nustatyti vartotojo naršyklės, pasiūlyti įsidiegti atnaujinimus, naujas versijas.
12.	Projekto personalo pasitraukimas dėl ligos ir pan.	Reikia numatyti tokias galimybes, teisingai suplanuoti atvejus personalui pasitraukus, palikti laiko rezervą arba numatyti rezervinius resursus.

3.16.7. Perspektyviniai reikalavimai

Šiame skyriuje pateiksime „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojama sistemos reikalavimus funkcinis, nefunkcinis kurie gali būti įvertinti perspektyvinėje sistemos versijoje. Tai yra nerealizuotus reikalavimus, kurie laukia savo eilės kad būtų realizuoti.

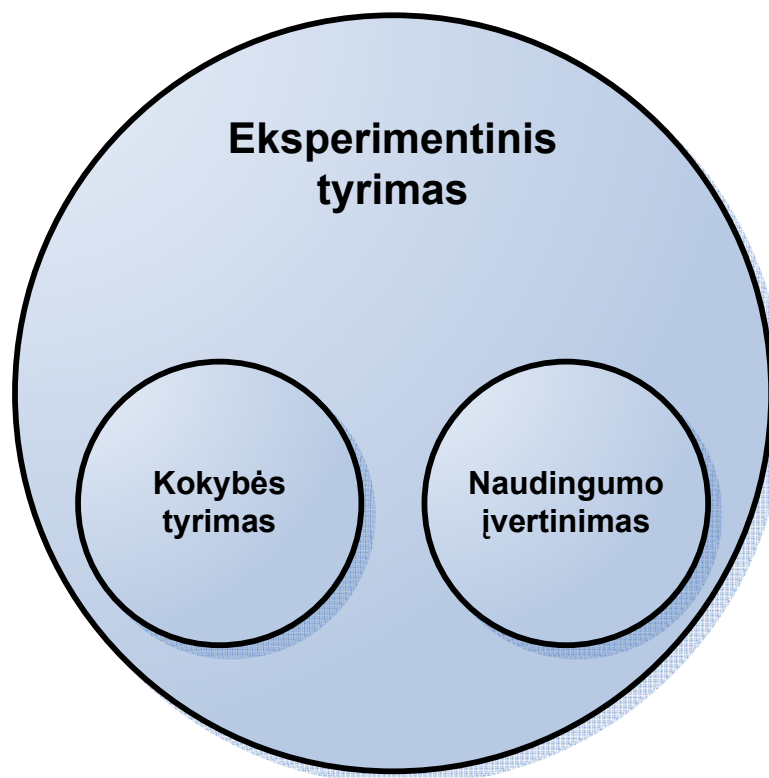
Perspektyviniai reikalavimai „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamai sistemai:

- Duomenų integralumas su kitomis sistemomis.
- Sistemos išvaizdai suteikti dinaminę struktūrą, tam, kad užkrovus per skirtingų rezoliucijų langus rodytų vienodai.
- Sistemoje sukurti “banerių” tam kad joje galima būtų reklamuoti sistema arba naudingas nuorodas.

4. EKSPERIMENTINĖ TIRIAMOJI DALIS

4.1. Įvadas

Eksperimentinio tyrimo metu atliekamas magistrinio darbo metu sukurtos „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos eksperimentinis tyrimas.



Paveikslas 4.1-1. Eksperimentinio tyrimo aspektai.

Sistemos eksperimentinis tyrimas atliekamas dviem aspektais (kaip pateikta aukščiau esančiam paveiksle (Paveikslas 4.1-1.)): pagal kokybės kriterijus ir atliekant realizuotos sistemos naudingumo įvertinimą.

4.2. Programų sistemos kokybės vertinimo charakteristikos

Sukurtos „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos programinės įrangos kokybės vertinimui naudosim ISO 9126 standartą [61]. Jis pateikia kokybės modelį, kuris yra pritaikomas, bet kokiai programinei įrangai. Žemiau esančiame paveiksle (Paveikslas 4.2-1.) pateikti šeši pagrindiniai šį modelį sudarantys produkto kokybės kriterijai.



Paveikslas 4.2-1. ISO 9126 modelis.

Pagrindinius kriterijus sudarantys subkriterijai ir jų aprašymai pateikiami žemiau esančioje lentelėje (Lentelė 4.2-1.).

Lentelė 4.2-1. ISO 9126 modelio kriterijai ir subkriterijai.

Kriterijus	Subkriterijus	Apibrėžimas
Funkcionalumas	Tinkamumas	Programinės įrangos atributai nurodantys esančias ir teisingas funkcijas reikalingas tam tikrai užduočiai.
	Tikslumas	Programinės įrangos atributai nurodantys gražinamus teisingus ar sutartus rezultatus, ar efektus.
	Bendradarbiavimas	Programinės įrangos atributai nurodantys

		galimybes sistemai sąveikauti su specifikuotom sistemom.
	Atitikimas	Programinės įrangos atributai kurie padaro programinę įrangą atitinkančią standartus, konvencijas ar įstatymus ir panašius aprašus.
	Apsauga	Programinės įrangos atributai saugantys nuo neteisėto panaudojimo ir priėjimo, nesvarbu tyčinio ar ne, prie programos ar duomenų.
Patikimumas	Išbaigtumas	Programinės įrangos atributai nurodantys klaidų dažnį programinėje įrangoje
	Pakantumas klaidoms	Programinės įrangos atributai nurodantys sistemos galimybes išlaikyti tam tikrą veikimo lygį įvykus klaidoms.
	Atstatomumas	Programinės įrangos atributai nurodantys galimumą atstatyti normalų veikimo darbą ir atgaminti duomenis tiesiogiai paveiktus klaidų bei laiką ir pastangas reikalingas tai atlikti.
Lengvumas naudoti	Suprantamumas	Programinės įrangos atributai nurodantys vartotojui reikalingas pastangas, kad suprastų loginius principus ir jų pritaikymą.
	Išmokstamumas	Programinės įrangos atributai nurodantys vartotojui reikalingas pastangas, kad išmokti naudotis programa.
	Darbingumas	Programinės įrangos atributai nurodantys vartotojo pastangas atliekant operacijas ir operacijų kontrolę.
Efektyvumas	Laiko naudojimas	Programinės įrangos atributai nurodantys sistemos veikimo laikus ir atliekamų operacijų skaičius per tam tikrą laiką.
	Resursų naudojimas	Programinės įrangos atributai nurodantys sunaudojamų resursų kiekį ir kiek laiko jie

		yra naudojami.
Palaikomumas	Analizuojamumas	Programinės įrangos atributai nurodantys reikalingą pastangų kiekį siekiant nustatyti netikslumus, klaidų priežastis ar identifikuoti dalis, kurias reikia modifikuoti.
	Keičiamumas	Programinės įrangos atributai nurodantys pastangų kiekį reikalingą modifikavimui, klaidų pašalinimui ar aplinkos pakeitimams.
	Stabilumas	Programinės įrangos atributai nurodantys riziką apie nenumatytus efektus atliekant modifikavimą.
	Testuojamumas	Programinės įrangos atributai nurodantys pastangas reikalingas ištestuoti programinę įrangą.
Pernešamumas	Pritaikomumas	Programinės įrangos atributai nurodantys galimybes pritaikyti skirtingoms aplinkoms nenaudojant jokių kitokių priemonių kaip tos, kurios priklauso tai programinei įrangai.
	Instaliuojamumas	Programinės įrangos atributai nurodantys pastangas reikalingas instaliuoti programinę įrangą specifikuotoje aplinkoje.
	Prisitaikymas	Programinės įrangos atributai, kurie leidžia programinei įrangai laikytis standartų ir konvencijų susijusių su portabilumu.
	Pakeičiamumas	Programinės įrangos atributai nurodantys galimybes naudoti šią programinę įrangą vietoj kitos programinės įrangos tos programinės įrangos aplinkoje.

Remiantis šiais kriterijais įvertintos sistemos kokybė turėtų būti neprastesnė kaip vidutiniška.

4.3. Sistemos kokybės tyrimas

„Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos kokybės vertinimas buvo atliekamas pagal 4.2. *Programų sistemos kokybės vertinimo charakteristikos* skyriuje apibrėžtus ISO 9126 standarto kriterijus. Remiantis šiais kriterijais įvertintos sistemos kokybė turėtų būti neprastesnė kaip vidutiniška

Kokybės subkriterijų vertinimo parametrai apibrėžti remiantis sistemai iškeltais reikalavimais, o vertinimas atliekamas stebint sistemos darbą, ir analizuojant įvairią statistinę sistemos registruojamą informaciją.

Kiekvienas kokybės subkriterijus susideda iš vieno ar daugiau jį nusakančių parametrų. Kiekvienas parametras įvertintas dešimtbalėje skalėje: 1 – kokybė labai bloga arba toks parametras išvis nerealizuotas, 10 – kokybė labai gera.

Žemiau pateiktoje lentelėje (Lentelė 4.3-1.) pateikiami „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos kokybės vertinimo rezultatai. Kiekvienas kokybės kriterijus įvertinamas pagal visus jį sudarančius subkriterijus paskui apskaičiuojant bendrą kriterijaus įvertinimą.

Lentelė 4.3-1. „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos kokybės vertinimas pagal ISO 9216 standartą.

Funkcionalumas		
<i>Subkriterijus</i>	<i>Subkriterijaus parametrų aprašymai</i>	<i>Įvertinimas</i>
Tinkamumas	Sistema realizuoja visus nurodytus pradinis vartotojo reikalavimus sistemai.	8
	Sistema realizuoja visus iš vartotojo surinktus funkcinis reikalavimus.	9
	Sistema realizuoja visus iš vartotojo surinktus nefunkcinis reikalavimus.	9
Tikslumas	Sistemos atliekami apmokėjimų skaičiavimai yra teisingi.	9
	Sistemos atliekami pinigų pervedimai yra teisingi.	9
Bendradarbiavimas	Sistema realizuoja duomenų importavimą iš kitų susijusių sistemų.	7
	Sistema realizuoja duomenų eksportavimą į kitas	8

	susijusias sistemas.	
Atitikimas	Sistemos formuojamos ataskaitos atitinka įstatymų numatytus reikalavimus.	8
	Sistema užtikrina įstatymiškai privalomą asmeninių duomenų saugumą/neplatinimą.	8
Apsauga	Administraciniai ir paprasti sistemos vartotojai turi prisijungimo vardus ir slaptažodžius.	7
	Sistemos DB apsaugota slaptažodžiu.	10
	Visi sistemos atliekami duomenų persiuntimai per WEB servisus atliekami tik po sėkmingos vartotojo identifikacijos.	10
	Vartotojų slaptažodžiai yra koduojami.	8
Bendras		8,4
Patikimumas		
<i>Subkriterijus</i>	<i>Subkriterijaus parametrų aprašymai</i>	<i>Įvertinimas</i>
Išbaigtumas	Ne daugiau kaip viena klaida per 7 dienas.	5
Pakantumas klaidoms	Visos klaidos registruojamos ir apie jas pranešama vartotojui.	8
	Klaida kuriame nors sistemos modulyje arba vartotojo sąsajos formoje neapriboja galimybių naudotis likusia sistemos dalimi.	8
Atstatomumas	Įvykus klaidai sistema neleidžia koreguoti duomenų. Įrašomi tik pilni duomenų paketai. Palaikomos transakcijos.	5
Bendras		6,5
Lengvumas naudoti		
<i>Subkriterijus</i>	<i>Subkriterijaus parametrų aprašymai</i>	<i>Įvertinimas</i>
Suprantamumas	Vartotojo sąsaja nenaudoja jokių neįprastų formų ar būdų. Tie patys vartotojo sąsajos valdymo principai išlaikomi visoje sistemoje.	9
	Kur reikia pateikiami paaiškinimai pačioje	6

	virtotojo sąsajoje.	
Išmokstamumas	Vartotojas susipažįsta ir išmoksta naudotis pagrindinėmis sistemos funkcijomis per vieną dieną (8 valandas).	8
	Sistema turi vadovą pradedančiajam.	9
	Sistema turi detalų vartotojo vadovą.	9
Darbingumas	Sistemoje naudojamas lengvai suprantamas, intuityvus ir pastovus vartotojo sąsajos grafinis dizainas.	9
	Vartotojui pateikiami pranešami apie visus jo atliekamus veiksmus. Ar jie buvo sėkmingi ar nesėkmingi ir kodėl.	9
Bendras		8,4
Efektyvumas		
<i>Subkriterijus</i>	<i>Subkriterijaus parametrų aprašymai</i>	<i>Įvertinimas</i>
Laiko naudojimas	Duomenų atsinaujinimas po vartotojo sukūrimo neužtrunka 5 min	8
	Duomenų atsinaujinimas po vartotojo trinimo neužtrunka 3 min.	8
	Įprastinės vartotojo operacijos neužtrunka ilgiau kaip 5 sekundes.	8
Resursų naudojimas	Laikas kai sistema 100% apkrauna jai paskirtą WEB serverį neviršija 1 valandos per parą.	8
	Duomenų bazės apkraunamumas nedidėja eksponentiškai su kiekvienu prisijungusių vartotojų kreipimuisi tuo pačiu metu.	9
Bendras		8,2
Palaikomumas		
<i>Subkriterijus</i>	<i>Subkriterijaus parametrų aprašymai</i>	<i>Įvertinimas</i>
Analizuojamumas	Prie visų klaidų pranešimų pateikiama informacija, kurioje vietos ji įvyko.	6

	Visos sistemos funkcijos yra tiksliai dokumentuotos. Viena funkcija vykdoma viename modulyje.	7
Keičiamumas	Norint atlikti pakeitimus sistemoje nereikia būti susipažinus su visa sistemos realizacija.	7
	Daugelį sistemos veikimo parametrų galima keisti per parametrus. Nereikalaujant sistemos perkompiliavimo.	7
Stabilumas	Pakeitimai viename sistemos modulyje neiššaukia pakeitimų ar klaidų kitame.	4
Testuojamumas	Automatiniai sistemos testai apima didžiąją dalį sistemos funkcijų.	5
	Sistema nereikalauja specialios aplinkos sukongūravimo ir paleidimo prieš pradant jos testavimą.	4
Bendras		5,71
Pernešamumas		
<i>Subkriterijus</i>	<i>Subkriterijaus parametrų aprašymai</i>	<i>Įvertinimas</i>
Pritaikomumas	Sistema gali dirbti Windows (98, 2000, XP, 2003, Vista) operacinėse sistemose.	9
	Sistema gali būti naudojama Linux/Unix tipo operacinėse sistemose.	1
Instaliuojamumas	Lengvas sistemos WEB dalies diegimas	7
	Lengvas sistemos servisų diegimas	9
	Lengvas sistemos parametrų konfigūravimas	7
Prisitaikymas	Naudojami standartiniai protokolai duomenų mainams sistemos viduje.	8
Pakeičiamumas	Sistema turi didžiąją dalį funkcinių galimybių pasitaikančių alternatyviuose sprendimuose todėl gali būti naudojama vietoje jų.	8
Bendras		7,0

Aukščiau pateiktoje lentelėje (Lentelė 4.3-1.) kokybės vertinimas susideda iš 6 pagrindinių kriterijų ir 21 subkriterijaus. Kiekvienas subkriterijus nusakomas vieno ar daugiau parametro.

Konkretoaus **subkriterij**o reikšmė apskaičiuojama pagal žemiau pateiktą (1) formulę:

$$SBK = \frac{\sum pr}{pk} \quad (1)$$

kur SBK – subkriterijus,

pr – parametro reikšmė,

pk – parametrų kiekis.

Konkretoaus **kriterij**o kokybės įvertinimas randamas pasinaudojant 2-a formule:

$$K = \frac{\sum sk}{skk} \quad (2)$$

kur K – kriterijus,

sk – subkriterijus,

skk – subkriterijų kiekis.

Bendras sistemos kokybės įvertinimas gaunamas pasinaudojant 3-čia formule:

$$B\bar{I} = \frac{\sum K}{kk} \quad (3)$$

kur $B\bar{I}$ – bendras įvertis,

K – kriterijus,

kk – kriterijų kiekis.

Po visų atliktų skaičiavimų gautas bendras sistemos kokybės įvertis **lygus 7,3**. Gautas rezultatas rodo, kad sistema yra vidutiniškos kokybės.

Atskirų sistemos kokybės kriterijaus rezultatai ganėtinai skirtingi; blogiausiai įvertintas sistemos palaikomumo kriterijus. Pagrindinė problema yra ta, kad parametro pakeitimai viename sistemos modulyje neiššaukia pakeitimų ar klaidų kitame įvertinimas, nėra geras, tai

bet kokie pakeitimai sistemoje, padaryti mažiau su ja susipažinusių žmonių, gali iššaukti vienos ar daugiau sistemos dalių neveikimą. Tačiau funkcionalumo kriterijus, kuriame nusakomas sistemos atitikimas vartotojų lūkesčiams, pateiktiems parametrams, buvo įvertintas net 8,4 balo.

4.4. Sistemos naudingumo tyrimas

Atliekamas realizuotos „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ programų sistemos teikiamo naudingumo įvertinimas. Vertinimas buvo atliktas lyginant realizuotą sistemą su 2.4. *Egzistuojantys sprendimai* skyrelyje analizuotomis egzistuojančiomis sistemomis: *Harvard's Electronic Shop and Pay*, *Financial Management Service*, *E-payment*, *Stanford ePay*, *SDePay*, *Enroll & Pay*, *TUpay* sistema, *John Hopkins universiteto* elektroninio apmokėjimo sistema *E-Billing*, *Oregono universiteto* elektroninio apmokėjimo sistema *QuikPAY*.

Programų sistemos naudingumo įvertis remiasi programos galimybių įvertinimu. Kiekvienos programos realizuotos galimybės įtaka naudingumui buvo skaičiuojama remiantis svertiniu naudingumo įverčiu, kuris buvo gautas išanalizavus egzistuojančias sistemas ir realizuotą sistemą.

Žemiau pateikiama lentelė (Lentelė 4.4-1.), kurioje palyginamas egzistuojančių sistemų naudingumas su realizuotos sistemos „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ naudingumu (vertinamos tik svarbiausios sistemų galimybės).

Lentelė 4.4-1. Egzistuojančių sistemų naudingumo palyginimas su realizuota sistema („Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“).

Eil. Nr.	Galimybės	Sistemos										
		Harvard's Electronic Shop and Pay	Financial Management Service	E-payment	Stanford ePay	SDePay	Enroll & Pay	TUpay	E-Billing	QuikPAY	Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga	Naudingumo įverčiai
		<i>Harvard University</i>	<i>Indiana University</i>	<i>Trent University</i>	<i>Stanford University</i>	<i>South Dakota State University</i>	<i>University of Kansas</i>	<i>University of Temple</i>	<i>Johns Hopkins University</i>	<i>University of Oregon</i>	<i>Kauno technologijos universitetas</i>	
1.	Vartotojo autentifikacija	+	+	+	+	+	+	+	+	+	+	10
2.	Informacijos konfidencialumas	+	+	+	+	+	+	+	+	+	+	10
3.	Peržiūrėti sąskaitas	+	+	+	+	+	+	+	+	+	+	10
4.	Apmokėti įvairias sąskaitas	+	+			+		+	+	+	+	7
5.	Apmokėti įvairias sąskaitas susijusias su universiteto teikiamomis paslaugomis.	+	+		+	+	+	+	+	+	+	9
6.	Apmokėti universitetines sąskaitas už mokslą	+	+	+	+	+	+	+	+	+	+	10
7.	Automatinis sąskaitų apmokėjimas	+	+		+						+	4
8.	Automatiškai tvarkoma sąskaitų informacija	+	+		+	+					+	5
9.	Automatinis sąskaitų apmokėjimo formų pildymas						+				+	2
10.	Sąskaitos balanso kontrolė	+	+	+	+	+	+	+	+	+	+	10
11.	Sąskaitų istorija	+	+	+	+	+	+	+	+	+	+	10
12.	Išsami visų rūšių sąskaitų istorija			+	+		+	+			+	5
13.	Automatiniai pranešimai elektroniniu paštu apie naujas sąskaitas	+	+		+	+	+	+	+	+	+	9
14.	Automatiniai elektroninių apmokėjimų patvirtinimai elektroniniu paštu	+	+		+	+	+	+	+	+	+	9
15.	Veikla 24 valandas per parą ir 7 dienas per savaitę	+	+		+	+					+	5
16.	Veikla darbo dienomis ir darbo laiku	+	+	+	+	+	+	+	+	+	+	10
17.	Autorizuoti mokėtojai			+	+	+	+	+	+	+	+	8
18.	Tiesioginis prisijungimas prie sistemos iš vartotojo elektroninio pašto					+				+	+	3
19.	Peržiūrėti partnerių siūlomas prekes/paslaugas	+										1
20.	Apsipirkti partnerių svetainėse(arba lokaliai)/pirkti prekes ar paslaugas	+										1
Apskaičiuoti įvertinimai												
Galimybių skaičius, vnt.		16	14	9	15	15	13	13	12	13	18	20
Svorinis galimybių įvertis		120	118	83	124	125	112	117	112	115	136	138
Naudingumas (%)		87	86	60	90	91	81	85	81	83	99	100
Realizuotos galimybės (%)		80	70	45	75	75	65	65	60	65	90	

Anksčiau pateiktos lentelės (Lentelė 4.4-1.) apatinės dalies („Apskaičiuoti įvertinimais“), skaičiavimo formulės.

Skaičiavimo formulės:

- *Galimybių skaičius* – tai realizuotų galimybių skaičius. Ties kiekviena galimybę prie atitinkamos elektroninių apmokėjimų sistemos rašoma „ - nerealizuota, arba „+“ - realizuota. „+“ suma ir sudarys galimybių skaičių.
- *Svorinis galimybių įvertis* – prie kiekvienos galimybės yra nurodytas naudingumo įvertis nustatytas iš sistemų galimybių analizės.

$$SGI = \sum (w \cdot c),$$

kur SGI – svorinis galimybių įvertis,

w – naudingumo įvertis,

c – galimybės realizavimo reikšmė („ -“ arba „+“).

- *Naudingumas* – naudingumas parodo procentinį realizuotų galimybių naudingumą lyginant su maksimaliai įvertintu naudingumu.

$$N = \frac{SGI}{PN} \cdot 100\%$$

kur N – naudingumas,

SGI – svorinis galimybių įvertis,

$PN = \sum w$ - maksimalus svorinis galimybių įvertis = 138.

- *Realizuotos galimybės* – parodo procentinį realizuotų galimybių skaičių lyginant su maksimaliu galimybių skaičiumi.

$$RG = \frac{GS}{VG} \cdot 100\%$$

kur RG – realizuotos galimybės,

GS – galimybių skaičius,

$$VG = \sum g - \text{viena galimybė, išviso galimybių yra 20.}$$

- *Vidutinis naudingumas* – parodo visų tiriamų sistemų bendrą naudingumo vidurkį.

$$VN = \frac{\sum N}{SS},$$

kur VN – vidutinis naudingumas,

N – naudingumas,

SS – tiriamų sistemų skaičius, išviso tiriamų sistemų yra 10.

- *Vidutinis realizuotų galimybių kiekis* – parodo visų tiriamų sistemų bendrą realizuotų galimybių vidurkį.

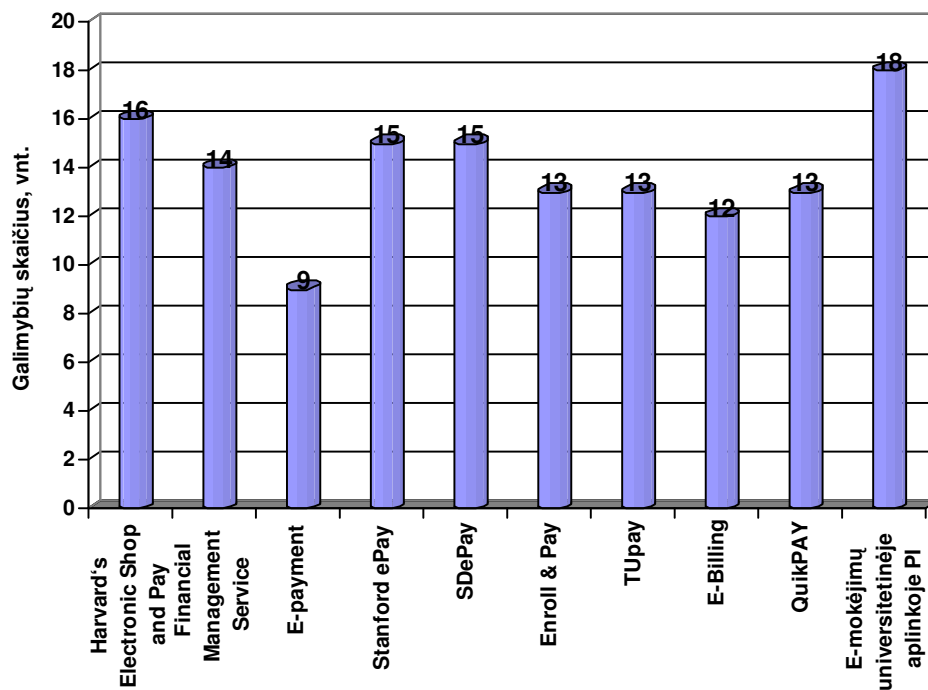
$$VRG = \frac{\sum RG}{SS},$$

kur VRG – vidutinis realizuotų galimybių kiekis,

RG – realizuotos galimybės,

SS – tyrimų sistemų skaičius, išviso tyrimų sistemų yra 10.

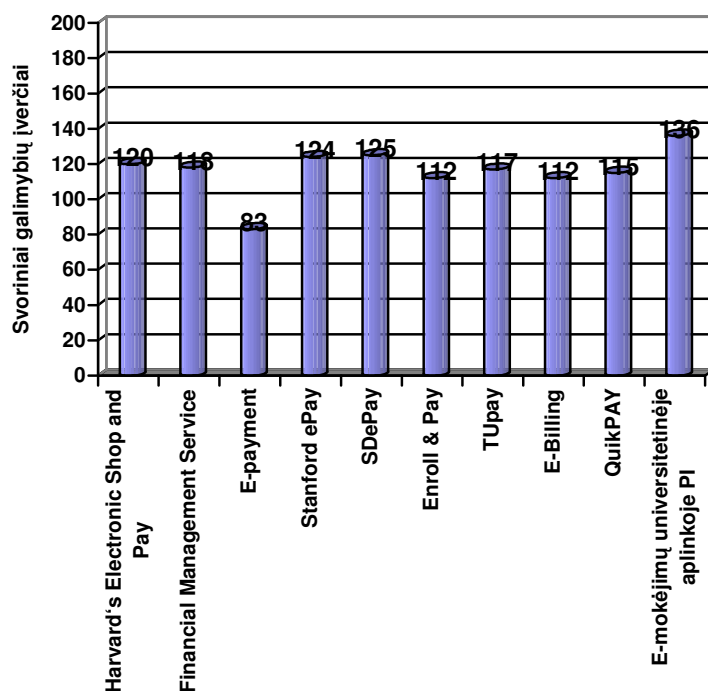
Žemiau pateiktame grafike (Paveikslas 4.4-2.), parodomi visų analizuojamų elektroninių mokėjimų universitetinėje aplinkoje sistemų galimybių kiekiai.



Paveikslas 4.4-1. Sistemų galimybių palyginimas.

Pateiktame grafike (Paveikslas 4.4-1.) matome, kad mūsų realizuota sistema turi daugiausiai galimybių (18). Panašų realizuotų galimybių kiekį turi tik trys sistemos: *Harvard's Electronic Shop and Pay* (16), *Stanford ePay* (15) ir *SDePay* (15). Visos kitos nagrinėtos sistemos turi gerokai mažiau galimybių. Tačiau šis grafikas visiškai neįvertina realizuotų galimybių naudingumo.

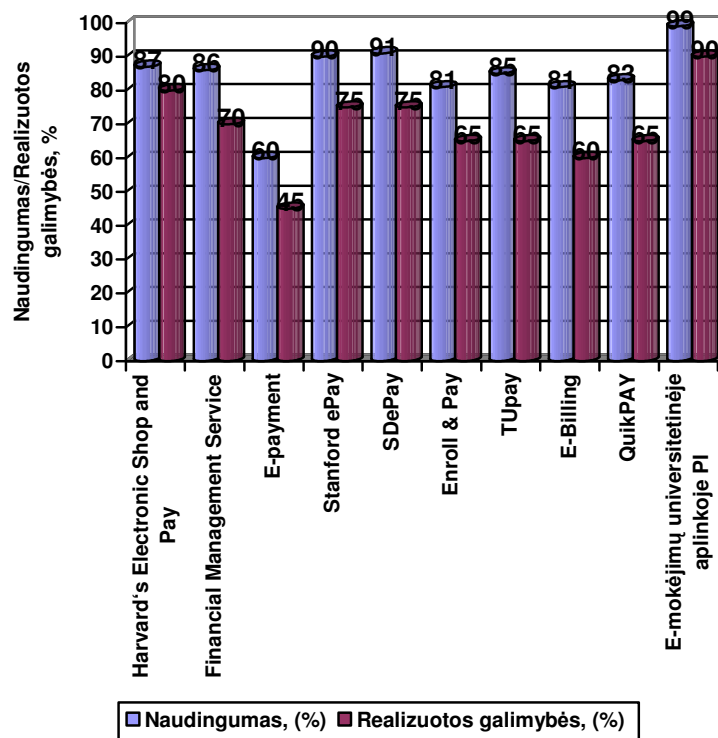
Žemiau pateiktam grafike (Paveikslas 4.4-2.), parodomi visų analizuojamų elektroninių mokėjimų universitetinėje aplinkoje sistemų surinkti svoriniai galimybių įverčiai.



Paveikslas 4.4-2. Sistemų naudingumo svorinių įverčių palyginimas.

Pateiktame grafike (Paveikslas 4.4-2.) matome, kad mūsų realizuota sistema turi didžiausią įvertį (136 balai). Panašų įvertį turi tik dvi sistemos: *SDePay* (125 balai) ir *Stanford ePay* (124 balai), bei *Harvard's Electronic Shop and Pay* (120 balų). Visos kitos nagrinėtos sistemos turi gerokai mažesnius įvertinimus. Tačiau šis grafikas visiškai neįvertina realizuotų galimybių skaičiaus ir nepalygina svorinių įverčių taškų su maksimaliai galimomis reikšmėmis.

Žemiau pateikiamas grafikas (Paveikslas 4.4-3.), kuriame atvaizduojamas palyginimas tarp procentinio realizuoto sistemų naudingumo ir procentinio realizuotų galimybių skaičiaus.



Paveikslas 4.4-3. Sistemų naudingumo ir realizuotų galimybių procentinis palyginimas.

Pateiktame grafike matome, kad iš visų nagrinėtų sistemų, mūsų realizuota sistema suteikia didžiausią naudingumo procentą (99%) ir taip pat realizuoja daugiausiai nagrinėtų galimybių (90%). Artimiausios konkurentės, kaip ir pagal anksčiau pateiktą grafiką, naudingumo atžvilgiu išlieka *SDePay* (91%) ir *Stanford ePay* (90%), bei *Harvard's Electronic Shop and Pay* (87%), realizuotų galimybių atžvilgiu *Harvard's Electronic Shop and Pay* (80%) ir *SDePay* (75%), bei *Stanford ePay* (75%).

Reikia pastebėti, kad *SDePay* sistema suteikia beveik panašų naudingumo procentą 91% vietoj 99% liginant su mūsų kurta sistema, realizuodama beveik 15% mažiau galimybių, atitinkamai 75% vietoj 90%. Taip pat pastebime, kad vienintelės mūsų sistemos naudingumo ir realizuotų galimybių skaičiaus procentai viršija kitas sistemas.

Taip pat galim pastebėti, kad visų sistemų bendras vidutinis naudingumas yra 84,3% („Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos naudingumas 99%, t.y.. 14,7% didesnis už vidurkį). Visų sistemų bendras vidutinis realizuotų galimybių skaičius yra 69% („Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos realizuotų galimybių skaičius 90%, t.y.. 21% didesnis už vidurkį).

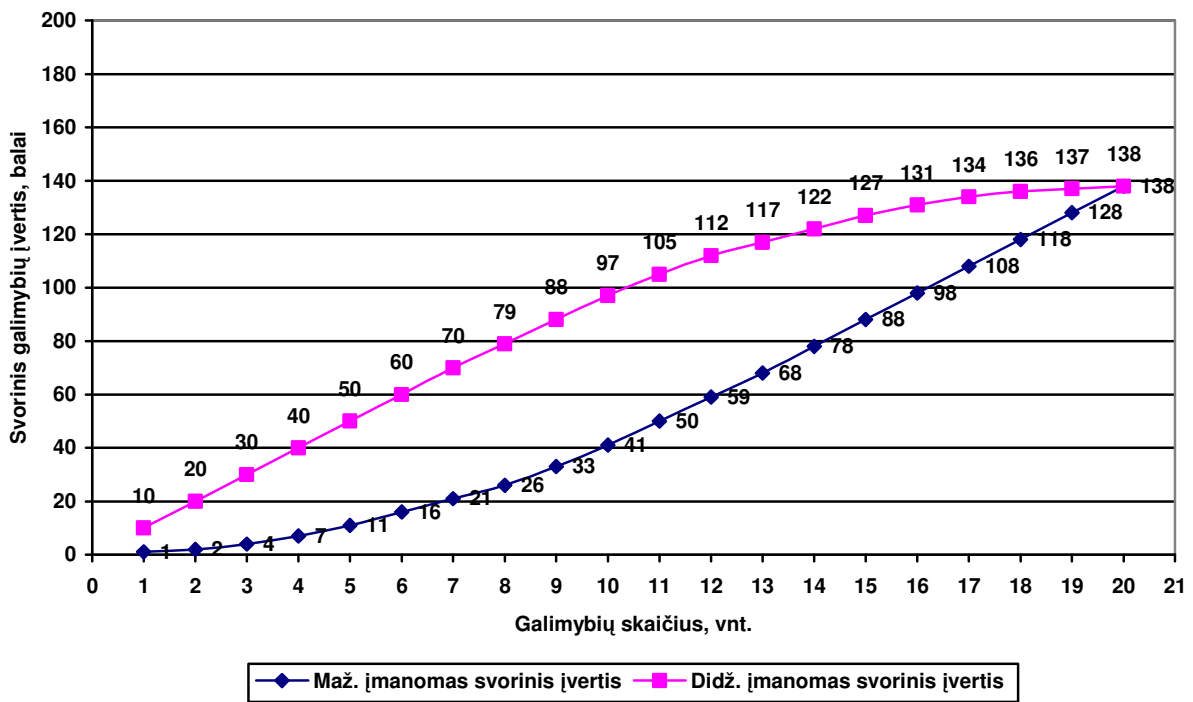
Kad būtų aiškiau kodėl taip yra sudarysime realizuotų galimybių ir jų teikiamų svorinių naudingumo įverčių priklausomybių grafiką.

Grafiko sudarymui naudojama žemiau pateiktą lentelę (Lentelė 4.4-2.). Šioje lentelėje atvaizduoti du duomenų masyvai: vienas jų nurodo mažiausius įmanomus svorinius galimybių naudingumo įverčius kitas didžiausius įmanomus svorinius galimybių naudingumo įverčius, priklausomai nuo galimybių kiekio.

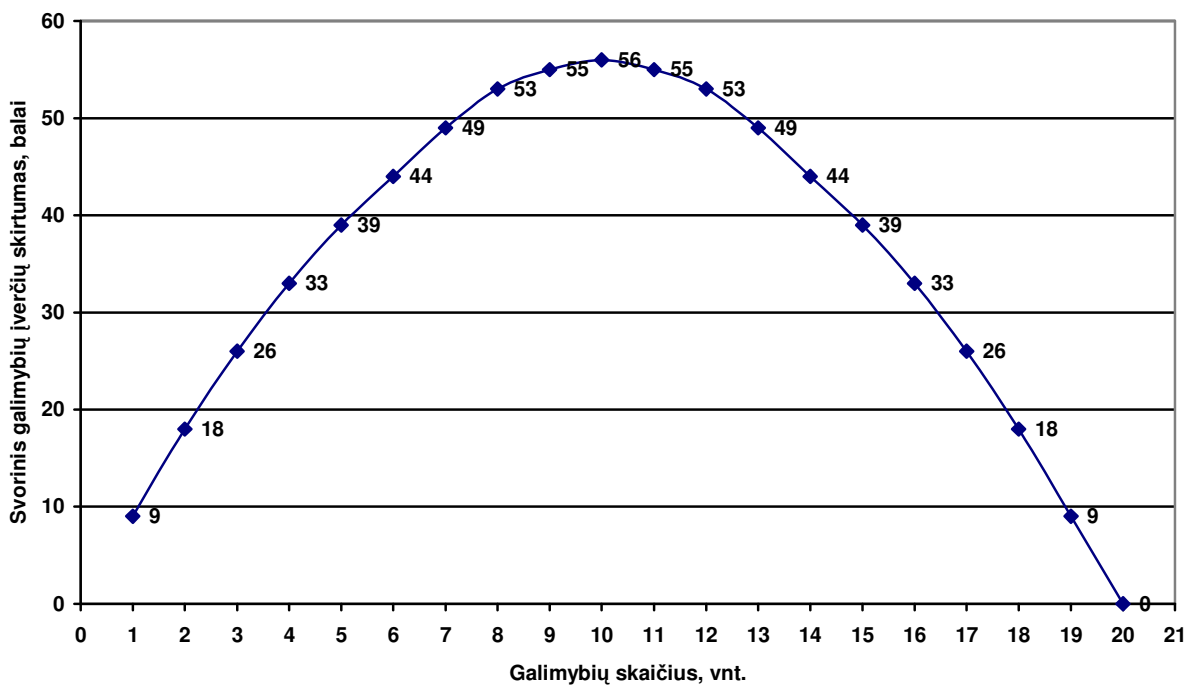
Lentelė 4.4-2. Galimybių skaičiaus ir svorinio naudingumo įverčio priklausomybė.

Galimybių skaičius	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Maž. įmanomas svorinis įvertis, balai	1	2	4	7	11	16	21	26	33	41	50	59	68	78	88	98	108	118	128	138
Didž. įmanomas svorinis įvertis, balai	10	20	30	40	50	60	70	79	88	97	105	112	117	122	127	131	134	136	137	138
Svorinių įverčių skirtumas, balai	9	18	26	33	39	44	49	53	55	56	55	53	49	44	39	33	26	18	9	0

Pagal lentelės (Lentelė 4.4-2.) duomenis nubraižomas žemiau pateiktas realizuotų galimybių ir svorinių naudingumo įverčių grafiką (Paveikslas 4.4-4.), bei realizuotų galimybių ir svorinių įverčių įtaka bendram sistemos naudingumui grafikas (Paveikslas 4.4-5).



Paveikslas 4.4-4. Realizuotų galimybių ir svorinių naudingumo įverčių grafikas.



Paveikslas 4.4-5. Realizuotų galimybių ir svorinių naudingumo įverčių įtaka bendram sistemos naudingumui.

Žiūrėdami į paveikslė (Paveikslas 4.4-4.) pateikto grafiko, maksimalių įverčių kreivę matome, kad nuo tam tikro taško papildomos realizuotos galimybės beveik nepadidina bendro sistemos naudingumo. Būtent toks, kai kurių galimybių, mažas įtakos turėjimas bendram sistemos naudingumui (tai parodyta paveikslė (Paveikslas 4.4-5.)), paaiškina kodėl *SDePay* sistema realizuodama beveik 15% (75%) mažiau galimybių pasiekia beveik tokį patį naudingumą (91%) (skirtumas 8%, lyginant su „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistema), arba *TUpay* sistema realizuodama beveik 25% (65%) mažiau galimybių pasiekia panašų naudingumą (85%) (skirtumas 14%, lyginant su „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos) *SDePay* sistema arba *TUpay* sistema realizuoja tik pačias naudingiausias galimybes, kai tuo tarpu mūsų realizuota „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistema realizuoja ir ne tokias naudingas savybes.

Tokių, mažai naudos nešančių, galimybių realizavimas nėra teigiamas dalykas, nes bet koks papildomas sistemos funkcionalumas įneša papildomą sudėtingumą ir klaidų tikimybę, o jei funkcija nenaudinga, tai papildomas sudėtingumas neatneša jokios finansinės naudos. Tačiau kita vertus ir mažai naudingos funkcijos yra dažnai reikalingos, nes jos neretai sumažina sistemos sudėtingumą vartotojo atžvilgiu (tačiau sistemos struktūros atžvilgiu sistemos sudėtingumą padidina). Apibendrinus galima teigti, kad ir mažai naudingos funkcijos bendrą sistemos naudingumą padidina. Visa tai apibendrinant galima teigti jog sistemoje nėra tiesioginės priklausomybės tarp sistemos teikiamo naudingumo ir jos realizuojamų funkcijų skaičiaus.

Tolesniuose sistemos tobulinimo etapuose reikėtų lanksčiau ir tiksliau įvertinti projektuojamų sistemos galimybių naudingumą.

4.5. Tolimesnės sistemos tobulinimo, plėtojimo galimybės

Atliekant sistemos kokybės ir naudingumo tyrimus bei įvertinant tyrimų metu gautus rezultatus, bei stebint vartotojų darbą su sistema, buvo nustatytos tokios „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos tobulinimo ir plėtojimo galimybės.

Siekiant padidinti sistemos kokybę bei naudingumą, reikėtų supaprastinti vartotojų sąsajas, kadangi sistemoje yra realizuota nemažai funkcijų, tačiau jų naudingumas gali būti dar didesnis. Tai pat reiktų daugumą pavienių funkcijų apjungti į apibendrintus funkcijų rinkinius, tai padidintų sistemos struktūros sudėtingumą, tačiau sistemos efektyvumas ženkliai padidėtų. Į sistemą reiktų įvesti daugiau įvairių sąskaitų tipų pildymo bei pateikimo šablonų. Siekiant padidinti sistemos patikimumą reikėtų į sistemą įtraukti papildomo sąskaitos balanso kaitos kontrolės funkciją. Taip pat į sistemą reikėtų įtraukti daugiau analizuojamojo pobūdžio statistinių funkcijų, pvz. statistinio pobūdžio informacija galėtų būti pateikiama ne vien lentelių pavidalu, bet ir grafiniu būdu.

„Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos realizavimo metu Microsoft kompanija išleido naujesnę MS .NET Framework 3.5 versiją, MS ASP.NET 3.5 versiją, taip pat MS SQL Server naują versiją. Vienas iš pagrindinių patobulinimų naujose versijose yra greitesnis ir funkcionalesnis duomenų rinkinių apdorojimas. Kadangi „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemoje duomenų apdorojimo funkcijos ypač plačiai naudojamos, tai naujos galimybės leistų pagreitinti sistemos veikimą ir duomenų apdorojimo kokybę, bei efektyvumą. Be kita ko perėjus prie naujų naudojamų komponentų versijų atsivertų didesnės saugumo ir vartotojo sąsajos pagerinimo galimybės. Taip pat pereinant prie naujesnių versijų reikėtų skirti nemažai dėmesį sistemos architektūros atnaujinimui, pritaikant naujas Microsoft komponentų galimybes, kas leistų stipriai supaprastinti sistemos palaikomumo problemas.

4.6. Apibendrinimas

Eksperimentinio tyrimo metu buvo atliekamas magistrinio darbo metu sukurtos „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos eksperimentinis tyrimas.

Sistemos eksperimentinis tyrimas buvo atliekamas dviem aspektais: pagal kokybės kriterijus ir atliekant realizuotos sistemos naudingumo įvertinimą.

Atliekant „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos kokybės tyrimą buvo remiamasi apibrėžtais ISO 9126 standarto kriterijais. Buvo vertinamas kiekvienas iš 6 pagrindinių šį modelį sudarančių produkto kokybės kriterijų. Ir detalizuojama iki pagrindinius kriterijus sudarančių 21 subkriterijo.

Sistemos bendras kokybės įvertinimas gavosi 7,3 dešimtbaleje sistemoje. Gautas rezultatas leidžia teigti, kad įvertintos sistemos kokybė yra neprastesnė kaip vidutiniška.

Peržvelgus atskirus vertinimo kriterijus pastebėta, kad prasčiausiai įvertintas sistemos palaikomumas. Šis kriterijus apima tokius subkriterijus kaip: analizuojamumas, keičiamumas, stabilumas ir kt. Tiesiog yra svarbu kad sistema lengvai analizuojama ir pritaikyta keitimams, na ir žinoma keitimai neturėtų iššaukti sistemos nestabilumo arba lūžimo pakeitus viena ar kitą jos modulį. Atsižvelgiant į tai, tobulinant sistemą siūloma peržiūrėti sistemos klasių architektūrą.

Funkcionalumo parametras buvo įvertintas 8,4 balo, kas rodo, kad sistemos atitikimas vartotojų lūkesčiams bei pateiktiems parametrams yra geras. Kadangi sistema veikia pagal užduotus vartotojo reikalavimus, taip pat visus nurodymus vykdo tiksliai ir greitai. Tokio tipo sistemoms yra labai svarbu tikslumas, našumas, greitis.

Atliekant „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos kokybės tyrimą, buvo tyrinėjamos realizuotos sistemos galimybės, analizuotų užsienio universitetuose esančių sistemų atžvilgiu. Analizei buvo pasirinktos vienos iš geriausių elektroninio apmokėjimo universitetinėje aplinkoje sistemų: *Harvard's Electronic Shop and Pay*, *Financial Management Service*, *E-payment*, *Stanford ePay*, *SDePay*, *Enroll & Pay*, *TUpay* sistema, *John Hopkins universiteto* elektroninio apmokėjimo sistema *E-Billing*, *Oregono universiteto* elektroninio apmokėjimo sistema *QuikPAY*. Taigi iš viso buvo tiriama 10 sistemų. Naudingumo tyrimui buvo išskirta 20 apibendrintų apžvelgtų sistemų naudingiausių ir populiariausių galimybių. Pagal kurias buvo vertinamos visos pasirinktos sistemos.

Naudingumo tyrimo metu buvo tiriamos sistemos pagrindinės galimybės, naudingumo svorinai įverčiai, sistemų naudingumas pagal galimybes, realizuotų galimybių ir svorinių įverčių priklausomybė.

Naudingumo tyrimo metu buvo nustatyto, jog daugiausiai galimybių turi magistrinio darbo metu realizuota „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistema (18), artimiausias sistema yra *Harvard's Electronic Shop and Pay* (16). Tačiau pagal naudingumo svorinius įverčius daugiausiai naudingumo įverčio balų surinko realizuota sistema (136 balai), panašų įvertinimą turi tik dvi sistemos: *SDePay* (125 balai) ir *Stanford ePay* (124 balai), bei *Harvard's Electronic Shop and Pay* (120 balų). Pagal naudingumą naudingiausia sistema yra „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ (99% naudingumas ir 90% realizuotos galimybės), artimiausios konkurentės, naudingumo atžvilgiu yra *SDePay* (91%) ir *Stanford ePay* (90%), bei *Harvard's Electronic Shop and Pay* (87%), realizuotų galimybių atžvilgiu *Harvard's Electronic Shop and Pay* (80%) ir *SDePay* (75%), bei *Stanford ePay* (75%). Tačiau atsižvelgiant į realizuotas galimybes ir galimybių svorinius naudingumo įverčius, galima teigti kad norint pasiekti pakankamai didelį sistemos naudingumo lygį yra nebūtina realizuoti daug funkcijų (galimybių), iš esmės užtektų realizuoti tik pačias svarbiausias, nes kuo daugiau sistema turi galimybių, tuo kiekviena papildoma galimybė faktiškai įneša vis mažesnę naudingumą.

Tokių, mažai naudos nešančių, galimybių realizavimas nėra teigiamas dalykas, nes bet koks papildomas sistemos funkcionalumas įneša papildomą sudėtingumą ir klaidų tikimybę, o jei funkcija nenaudinga, tai papildomas sudėtingumas neatneša jokios finansinės naudos. Tačiau kita vertus ir mažai naudingos funkcijos yra dažnai reikalingos, nes jos neretai sumažina sistemos sudėtingumą vartotojo atžvilgiu (tačiau sistemos struktūros atžvilgiu sistemos sudėtingumą padidina), kas dažnai padidina sistemos patrauklumą bei vertę. Apibendrinus galima teigti, kad ir mažai naudingos funkcijos bendrą sistemos naudingumą padidina. Visa tai apibendrinant galima teigti jog sistemoje nėra tiesioginės priklausomybės tarp sistemos teikiamo naudingumo ir jos realizuojamų funkcijų skaičiaus.

Tolesniuose sistemos tobulinimo etapuose reikėtų lanksčiau ir tiksliau įvertinti projektuojamos sistemos klasių architektūrą ir galimybių naudingumą.

5. IŠVADOS

1. Atlikus egzistuojančių sistemų, skirtų elektroninių mokėjimų universitetinėje aplinkoje uždavinio sprendimui, analizę nustatyta, kad būtent universitetinės aplinkos sistemose yra taikomos naujausios eksperimentinės elektroninių mokėjimų technologijos. Todėl darbe keliamas uždavinys iširti, technologijų įtaką sistemų funkcionalumui.
2. Yra aktualu, remiantis sistemų tyrimo požiūriu, kurti elektroninių mokėjimų universitetinėje aplinkoje sistemą. Poreikį kurti lėmė specifiniai sistemos funkciniai reikalavimai, bei sistemos individualumo savybės.
3. Programų sistemos kūrimo metodai ir priemonės pasirinktos metodų lyginamosios analizės pagrindu, atsižvelgiant į sprendžiamo uždavinio specifiką bei metodų privalumus. Reikalavimų specifikavimo detalumą užtikrina *Volere* šablonas. Objektinis projektavimo metodas pasirinktas dėl dinaminių projektavimo aspektų, leidžiančių pasiekti reikiamą detalumo lygį, atliekant į vartotoją orientuotus kūrimus. Projektavimo etapo lankstumui ir patikimumui užtikrinti, pasirinkti paketai *MagicDraw*, *Microsoft Office*, bei *Unified Modeling Language (UML)* notacija, užtikrinanti vieningą ir nuoseklų specifikavimą bei projektavimą.
4. Išanalizavus vartotojo išskeltus reikalavimus sistemai, bei egzistuojančias sistemas, nustatyti penkti probleminiai sistemos realizavimo uždaviniai:
 - Elektroniniai mokėjimai universitetinėje aplinkoje
 - Sąskaitų ir apmokėjimų valdymas
 - Centralizuotas finansinės informacijos valdymas
 - Vartotojų kontrolė finansinės informacijos aplinkoje
 - Elektroninių mokėjimų sistemos integracija universitetinėje aplinkojeVisiems jiems pateikti galimi sprendimo variantai.
5. Atlikus universiteto veiklos analizę buvo suformuluotas sprendžiamas uždavinys, apibrėžta tyrimo sritis ir objektas, suformuluotas dalykinės srities aprašas. Remiantis atlikta analize suformuluoti kuriamai sistemai keliami funkciniai ir nefunkciniai reikalavimai, kuriais vadovaujantis atliktas sistemos projektavimas.

6. Projektuojant sistemą pasirinkta trijų lygių architektūra: vartotojo sąsajos, valdymo ir duomenų bazės. Remiantis šia architektūra, priimta, kad vartotojas per vartotojo sąsają gali operuoti duomenimis esančiais duomenų bazėje naudodamasis sistemos teikiamu funkcionalumu.
7. Remiantis analitinėje ir projektinėje dalyse suformuluotais reikalavimais ir priimtais projektiniais sprendimais, realizuota elektroninių mokėjimų universitetinėje aplinkoje sistema, kurią sudaro moduliai:
 - Buhalterio (buhalterio – administratoriaus) modulis
 - Pardavėjo/Teikėjo modulis
 - Kliento – vartotojo modulis
 - Duomenų bazės serverio modulis
8. Remiantis ISO 9126 standarto rekomenduojamais kriterijais buvo atliktas sistemos kokybės vertinimo tyrimas. Jo metu gautas bendras sistemos kokybės įvertinimo balas lygus 7,3. Rezultatas leidžia teigti, kad įvertintos sistemos kokybė yra neprastesnė kaip vidutiniška. Peržvelgus atskirus vertinimo kriterijus pastebėta, kad prasčiausiai įvertintas sistemos palaikomumas. Tai yra dėl to kad sistema turi būti lengvai analizuojama ir pritaikyta keitimams, na ir žinoma keitimai neturėtų iššaukti sistemos nestabilumo arba lūžimo pakeitus vieną ar kitą jos modulį. Atsižvelgiant į tai, tobulinant sistemą siūloma peržiūrėti sistemos klasių architektūrą. Funkcionalumo parametras buvo įvertintas 8,4 balo, kas rodo, kad sistemos atitikimas vartotojų lūkesčiams bei pateiktiems parametrams yra geras. Kadangi sistema veikia pagal užduotus vartotojo reikalavimus, tai pat visus nurodymus vykdo tiksliai ir greitai.
9. Atlikus sistemos ir egzistuojančių panašių sistemų analizę, gauti panašių sistemų teikiamų galimybių svoriniai įverčiai. Remiantis šiais įverčiais buvo atliktas sukurtos elektroninių mokėjimų universitetinėje aplinkoje sistemos naudingumo tyrimas. Sudarius naudingumo ir realizuotų galimybių procentinio palyginimo su kitomis sistemomis grafiką, nustatyta, kad sukurta elektroninių mokėjimų universitetinėje aplinkoje sistema suteikia daugiausia naudingumo ir realizuoja didžiausią galimybių skaičių. Taip pat pastebėta, jog nagrinėtų sistemų suteikiamų galimybių procentas įvairiai viršija jų teikiamo naudingumo procentą. Norint išsiaiškinti šio skirtumo

priežastį buvo sudarytas realizuotų galimybių ir svorinių naudingumo įverčių grafikas, kuris parodė, kad kai kurios galimybės turi tik nedidelę įtaką bendram sistemos naudingumui. Dėl pastarosios priežasties sistemoje nėra tiesioginės priklausomybės tarp sistemos teikiamo naudingumo ir jos realizuojamų funkcijų skaičiaus.

10. Atlikus sistemos vertinimą pastebėtos jos tobulinimo kryptys. Tolesniuose sistemos tobulinimo etapuose reikėtų lanksčiau ir tiksliau įvertinti projektuojamų sistemos architektūra ir galimybių naudingumą
11. Žinios ir įgūdžiai pritaikyti sprendžiant praktinio pobūdžio problemą, praeiti visi programinės įrangos kūrimo etapai, paruošta pilna projekto dokumentacija ir realizuota elektroninių mokėjimų universitetinėje aplinkoje sistema. Sistemos tyrimo rezultatai patvirtino programų sistemos efektyvumą.

6. LITERATŪRA

1. VILIALO, Matt. ePayment >> Modern Times. *Insue of Campus Technology* [interaktyvus]. 2005, rugpjūtis [žiūrėta 2006-11-10]. Prieiga per internetą: <<http://campustechnology.com/article.asp?id=11520>>
2. *Harvard's Electronics Shop and Pay*: Digital cash software of university. *Harvard University*. [žiūrėta 2006-11-10]. Prieiga per internetą: <<http://able.harvard.edu/esp/>>
3. VWR Scientific. Žiūrėta [2006-11-10]. Prieiga per internetą: <<http://www.vwrsp.com/about/index.cgi>>
4. HLW. Harvard's Electronics Shop and Pay. *Financial Office* [interaktyvus]. 2005, balandis [žiūrėta 2006-11-10]. Prieiga per internetą: <<http://www.law.harvard.edu/administration/financial/news/2004/06/ESPMemo.php>>
5. Financial Management Service: Digital cash software of university. Indiana University. [žiūrėta 2006-11-10]. Prieiga per internetą: <<https://fdrs.fms.indiana.edu/fdrs/>>
6. Financial Management Service. *Indiana University* [interaktyvus]. 2006, rugpjūtis [žiūrėta 2006-11-10]. Prieiga per internetą: <<http://www.fms.indiana.edu/fis/home.asp>>
7. E-payment: Digital cash software of university. *Trent University*. [žiūrėta 2006-11-10]. Prieiga per internetą: <<http://www.trentu.ca/admin/finance/>>
8. E-payment at Trent Lunched. *Trent University* [interaktyvus]. 2005, rugpjūtis [žiūrėta 2006-11-10]. Prieiga per internetą: <<http://www.trentu.ca/news/daily/archive/050816epayment.htm>>
9. ALI, Student Financial Services and The Cashier's Office. Stanford ePay. Stop standing in line and go online!. *Stanford University* [interaktyvus]. 2006, rugsėjis [žiūrėta 2006-11-11]. Prieiga per internetą: <<http://co.stanford.edu/students/stanfordepay/>>
10. SDePay: Digital cash software of university. *South Dakota State University*. [žiūrėta 2006-11-11]. Prieiga per internetą: <<http://courseinfo.sdstate.edu/Spring06PayInfo/>>
11. SDSU CASHIERS OFFICE, SDePay – Electronic Billing & Payment Services

- for Student Tuition and Fees. *South Dakota State University* [interaktyvus]. 2006, liepa [žiūrėta 2006-11-11]. Prieiga per internetą: <<http://courseinfo.sdstate.edu/Spring06PayInfo/SDePayBrochure.pdf>>
12. Enroll & Pay: Digital cash software of university. *University of Kansas* [žiūrėta 2006-11-11]. Prieiga per internetą <<https://sa.ku.edu/psp/saku/?cmd=login>>
 13. GARDNER, D.; GERGEN, J.; MODIN, D.; ir GOBLE, D., *eCommerce Project Overview* [interaktyvus]: projekto apžvalga. University of Kansas, 2005, kovas [žiūrėta 2006-11-11]. Prieiga per internetą: <<http://www.comptroller.ku.edu/eCommerce.pdf>>
 14. Temple universiteto TUpay sistema. *Bursar's Office*, Temple University. [žiūrėta 2006-11-10]. Prieiga per internetą: <<http://www.temple.edu/bursar/current/current.htm>>
 15. TUpay: Digital cash software of university . *University of Temple* [žiūrėta 2006-11-10]. Prieiga per internetą: <<http://tupay.temple.edu/>>
 16. TUpay central system: Digital cash software of university . *University of Temple* [žiūrėta 2006-11-10]. Prieiga per internetą: <<http://ownlnet.temple.edu/>>
 17. Electronic Bill Presentment & Online Student Payment Systema. *Johns Hopkins University*. [žiūrėta 2006-11-10]. Prieiga per internetą: <<http://www.peabody.jhu.edu/597>>
 18. Oregon University QuikPAY: Digital cash software of university. University of Oregon. [žiūrėta 2006-11-12]. Prieiga per internetą: <<http://baowww.uoregon.edu/Student/EbillIntro.htm> >
 19. DARA, J.; GUNDEMONI, L. *Credit Cardr Security and E-Payment: Master's Thesis*. Lulea University of Technology Information Systems Sciences [Lulea], 2006. 52 p. [žiūrėta 2006-11-11] Prieiga per internetą : <<http://epubl.ltu.se/1653-0187/2006/23/LTU-PB-EX-0623-SE.pdf> >
 20. Transakcija. Vikipedija [interaktyvus]. 2006, liepa [žiūrėta 2006-11-11]. Prieiga per internetą: <<http://lt.wikipedia.org/wiki/Transakcija>>
 21. BRUCKER, X. F. *Primitive-Based Payment System for Flexible Value Transfer in the Personal Router: Master's Thesis*. Massachusetts Institutec of Technology. Engineering System Division. [Massachusetts], 2002. 154 p. [žiūrėta 2006-11-11]Prieiga per internetą:

- <http://itc.mit.edu/itel/students/papers/brucker_thesis.pdf>
22. KURKI, O. *Modeling The Workload of an Electronic Commerce Payment System*: Master's Thesis. Helsinki University of Technology, Laboratory of Information Processing Science, [Helsinki]. 2000, 63 p. [žiūrēta 2006-11-12] Prieiga per internetą: <<http://stg.cs.hut.fi/publications/kurki-00.pdf>>
 23. GRADWELL, P. *A survey of current techniques and issues surrounding the security of electronic (financial) transactions*: Mokslinis straipsnis. University of Bath, Department of Computer Science [Bath], 2001, 18 p. [žiūrēta 2006-11-12] Prieiga per internetą: <<http://peter.gradwell.com/phd/writings/security-survey.pdf>>
 24. XUE, F. *A Secure Anonymous and Scalable Digital cash System*: Master's Thesis. McGill University, School of Computer Science. [Montreal], 2002, 109 p. [žiūrēta 2006-11-12] Prieiga per internetą: <<http://www.cs.mcgill.ca/~crepeau/PDF/memoire-feng.pdf>>
 25. Fischer, M. *Towards a Generalized Payment Model for Internet Services*: Master's Thesis. Technical University of Vienna, Information Systems Institute. [Vienna], 2002, 125 p. [žiūrēta 2006-11-12]. Prieiga per Internetą: <<http://www.infosys.tuwien.ac.at/teaching/thesis/online/Fischer/Fischer.pdf>>
 26. SADEGHI, A.-R.; SCHNEIDER, M. *Electronic Payment Systems*, Springer Berlin / Heidelberg, Germany, 2003, 462 p. ISBN: 978-3-540-40465-1.
 27. KNIBERG, H. *What Makes a Micropayment Solution Succeed*: Master's Thesis. Kungliga Tekniska Hogskolan, Institution for Applied Informatikon Technology. [Stocholm], 2002. 68 p. [žiūrēta 2006-11-11]Prieiga per internetą: <<http://www.kniberg.com/henrik/thesis/pdf/What-makes-a-micropayment-solution-succeed.pdf>>
 28. PUHRERFELLNER, M. *An implementation of the Millicent micro-payment protocol and its application in a pay-per-view business model*: Master's Thesis. Technische Universitat Wien, Systems Group. [Wien], 2000. 86 p. [žiūrēta 2006-11-11]Prieiga per internetą: <<http://www.infosys.tuwien.ac.at/teaching/thesis/online/Puehrerfellner/mpuehrer.pdf>>
 29. MANDAGI, R. K. *Comparison of Current On-line Payment Technologies*: Master's Thesis. Linkoping Institute of Technology, Dept of Electrical Engineering. [Linkoping], 2006, 95 p. [žiūrēta 2006-11-12] Prieiga per

- internetā: <www.diva-portal.org/diva/getDocument?urn_nbn_se_liu_diva-7383-1__fulltext.pdf>
30. ORRIE GARTNER, G.; MADJDPOUR, A.; MCMURRAY, B.; SUBRAMANIAN, S. *Program upon Payment Processors Implications of Visa's Cardholder Information Security* : A capstone paper submitted as partial fulfillment of the requirements for the degree of Masters. University of Colorado, Interdisciplinary Telecommunications. [Kolorado], 2003, 13 p. [žiūrēta 2006-11-12]. Prieiga per internetā: <http://spot.colorado.edu/~sicker/students/Capstone_CISP.pdf>
 31. ROCKINGER, R.; BAUMEISTER, H. *BABSy: Basic Agent Framework Billing System*: mokslinis straipsnis. Universitat Munchen, Institut fur Informatik. [Munchen], 2000, 7p.[žiūrēta 2006-11-12]. Prieiga per internetā: <<http://www.pst.ifi.lmu.de/~baumeist/publications/mama2000.pdf>>
 32. KIM, B.; LAAS, C.; O'GILVIE, S.; YIP, A. *Anonymity Tools for the Internet*: tarptautinēs konferencijas pranešimų medžiaga. [Massachusetts], 2001, 36 p. [žiūrēta 2006-11-12] Prieiga per internetā: <<http://www-swiss.ai.mit.edu/6095/student-papers/spring01-papers/anonymity.pdf>>
 33. SEYS, S., DIAZ, C., DE WIN, B., NEASSENS, V., GOEMANS, C., CLEASSENS, J., MORKAU, W. APES. *Anonimity and Privacy in Electronics Servines*: Master's Thesis. IWT Strategische Technologieen Woor Welzijn en Welvaart ,[Sedas]. 2001, 73 p. [žiūrēta 2006-11-11] Prieiga per internetā: <<http://www.cosic.esat.kuleuven.be/publications/article-450.pdf>>
 34. Pfleeger, C., P., Pfleeger, S., L. *Security in Computing*. Prentice-Hall, Upper Saddle River, 2006, 845 p. ISBN: 978-0-13-239077-4
 35. Interneto Protokolas. Wikipedija [interaktyvus]. 2006, gegužē [žiūrēta 2006-11-12]. Prieiga per internetā: <http://lt.wikipedia.org/wiki/Interneto_protokolas>
 36. STEPHEN, A, T. *SSL and TLS essentials for Securing the Web*, John Wiley and Sons Inc, New York, USA, 2000, 224 p., ISBN: 0471383546
 37. Transport Layer Security. Wikipedija [interaktyvus]. 2008, rugsējis [žiūrēta 2006-04-01]. Prieiga per internetā: <http://en.wikipedia.org/wiki/Secure_Sockets_Layer>
 38. Wagner, D., Schneier, B. *Analysis of the SSL 3.0 protocol. In The Second USENIX Workshop on Electronic Commerce Proceedings - Volume 2:*

- mokslinis straipsnis. University of California, Counterpane Systems [Berkeley], 1996. 4 p. [žiūrēta 2008-04-23]. Prieiga per internetā : <http://portal.acm.org/ft_gateway.cfm?id=1267171&type=pdf&coll=GUIDE&dl=GUIDE&CFID=67498716&CFTOKEN=76836184>
39. SCHWARZ, T. *COEN 350 Kerberos*, Santa Clara University, Computer Engineering, [Santa Clara], 2003, [žiūrēta 2006-11-12] . Prieiga per internetā: <<http://www.cse.scu.edu/~tschwarz/coen350/kerberos.html>>
 40. MIT, *Kerberos: The Network Authentication Protocol*, MIT. 2006, lapkritis [žiūrēta 2006-11-12]. Prieiga per internetā: <<http://web.mit.edu/kerberos/>>
 41. OSBORN, S., SANDHU, R., MUNAWER, Q. *Configuring Role-Based Access Control to Enforce Mandatory and Discretionary Access Control Policies*: mokslinis straipsnis. The University of Western Ontario, George Mason University [London, Fairfax], 2000. 22 p. [žiūrēta 2008-04-01]. Prieiga per internetā : <[http://www.list.gmu.edu/journals/tissec/dac00\(org\).pdf](http://www.list.gmu.edu/journals/tissec/dac00(org).pdf)>
 42. Sandhu, R.S., Coyne, E.J., Role-based Access control models. *IEEE Computer* 29, 2 [interaktyvus]. 1996, 47p [žiūrēta 2008-04-01] Prieiga per internetā: <<http://ieeexplore.ieee.org/iel1/2/10411/00485845.pdf>>
 43. Bell, D. Secure computer systems: In *Proceedings on 3rd Annual Computer Security Application Conferenc* [interaktyvus]. 1987, 35 p. [žiūrēta 2008-04-01] Prieiga per internetā: <<http://www.albany.edu/acc/courses/ia/classics/bellpadula2.pdf>>
 44. Lee, T. Using mandatory integrity to enforce „commercial“ security. In *Proceedings of IEEE Symposium on Security and Privacy* [interaktyvus]. 1988, 10 p [žiūrēta 2008-04-01] Prieiga per internetā: <<http://www.users.cs.york.ac.uk/~fiona/PUBS/InforsidSSI04.pdf>>
 45. Sandhu, R.S., Bhamidipati, V., ir Munawer, Q. The ADBAC97 model for role-based administration of roles. *ACM Trans. Inf. Syst. Secur.* [interaktyvus]. 1999, 31p [žiūrēta 2008-04-01] Prieiga per internetā: <[http://www.list.gmu.edu/journals/acm/rbac97\(org\).pdf](http://www.list.gmu.edu/journals/acm/rbac97(org).pdf)>
 46. JANSSON, F.; LARISON, M. *Portal Payment Possibilities for a start-up company*: Master's Thesis. Royal Institute of Technology, Department of Teleinformatics. [Stockholm], 2000, 152 p. [žiūrēta 2006-11-12]. Prieiga per internetā: <http://www.e.kth.se/~e95_mla/ppp.pdf>
 47. VOSS, B. *UniCert* [interaktyvus]. 2004, rugsējis [žiūrēta 2006-11-12]. Prieiga

- per internetą: <<http://rcswww.urz.tu-dresden.de/~unicert/e/unicert.htm>>
48. Brokat, Twister. *Brokat* [interaktyvus]. 2006, lapkritis [žiūrėta 2006-11-12]. Prieiga per internetą: <<http://www.brokat.com/>>
 49. VerSign Team. About VeriSign. *VeriSign* [interaktyvus]. 2006, liepa [žiūrėta 2006-11-12]. Prieiga per internetą: <<http://www.verisign.com/verisign-inc/index.html>>
 50. *System.Web.Security Namespace*: MSDN library. *Microsoft*. [žiūrėta 2007-03-10]. Prieiga per internetą: <<http://msdn2.microsoft.com/en-us/library/system.web.security.aspx>>
 51. *System.Net.security Namespace*: MSDN library. *Microsoft*. [žiūrėta 2007-03-10]. Prieiga per internetą: <<http://msdn2.microsoft.com/en-us/library/system.net.security.aspx>>
 52. *System.Security Namespace*: MSDN library. *Microsoft*. [žiūrėta 2007-03-10]. Prieiga per internetą: <<http://msdn2.microsoft.com/en-us/library/system.security.aspx>>
 53. *System.Security.Accesscontrol Namespace*: MSDN library. *Microsoft*. [žiūrėta 2007-03-10]. Prieiga per internetą: <<http://msdn2.microsoft.com/en-us/library/system.security.accesscontrol.aspx>>
 54. *System.Security.Authentication Namespace*: MSDN library. *Microsoft*. [žiūrėta 2007-03-10]. Prieiga per internetą: <<http://msdn2.microsoft.com/en-us/library/system.security.authentication.aspx>>
 55. *System.Security.Cryptography Namespace*: MSDN library. *Microsoft*. [žiūrėta 2007-03-10]. Prieiga per internetą: <<http://msdn2.microsoft.com/en-us/library/system.security.cryptography.aspx>>
 56. *System.Security.Cryptography.Pkcs Namespace*: MSDN library. *Microsoft*. [žiūrėta 2007-03-10]. Prieiga per internetą: <<http://msdn2.microsoft.com/en-us/library/system.security.cryptography.pkcs.aspx>>
 57. *System.Security.Permissions Namespace*: MSDN library. *Microsoft*. [žiūrėta 2007-03-10]. Prieiga per internetą: <<http://msdn2.microsoft.com/en-us/library/system.security.permissions.aspx>>
 58. *System.Security.Policy Namespace*: MSDN library. *Microsoft*. [žiūrėta 2007-03-10]. Prieiga per internetą: <<http://msdn2.microsoft.com/en-us/library/system.security.policy.aspx>>
 59. *System.Security.Principal Namespace*: MSDN library. *Microsoft*. [žiūrėta 2007-

03-10]. Prieiga per internetą: <<http://msdn2.microsoft.com/en-us/library/system.security.principal.aspx>>

60. *System.Security.Rightsmanagement Namespace*: MSDN library. *Microsoft*. [žiūrėta 2007-03-10]. Prieiga per internetą: <<http://msdn2.microsoft.com/en-us/library/system.security.rightsmanagement.aspx>>
61. *ISO 9126: The Standard of Reference*. Centre for Software Engineering. [žiūrėta 2008 04 15]. Prieiga per internetą: <<http://www.cse.dcu.ie/essiscope/sm2/9126ref.html>>

7. TERMINŲ IR SANTRUMPŲ ŽODYNAS

Sąvoka	Paaiškinimas
Projekto dalių sąvokos:	
Buhalteris	Projektavimo terminas apibūdinantis sistemos vartotojus ir sistemos dalis, kurie(-ios) atlieka funkcijas susijusias su buhalterija, finansais ir administravimu. Apima vartotojus: buhalteris, administratorius, finansistas ir pan. Šie vartotojai šios dalies funkcijomis gali naudotis priklausomai nuo jiems suteiktų teisių.
Pardavėjas/Teikėjas	Projektavimo terminas apibūdinantis sistemos vartotojus ir sistemos dalis, kurie(-ios) atlieka funkcijas susijusias su prekyba, paslaugų teikimu. Apima vartotojus: pardavėjas, taikėjas.
Klientas - vartotojas	Projektavimo terminas apibūdinantis sistemos vartotojus ir sistemos dalis, kurie(-ios) atlieka funkcijas susijusias su eilinio kliento- vartotojo naudojimąsi sistema. Apima vartotojus: klientus – vartotojus.
Kitos sąvokos:	
Administracija	Tvarkanti visas grupes ir katedras.
American Express	Elektroninės bankininkystės mokomųjų kreditinių kortelių rūšis.
Autorizuotas mokėtojas	Studento kviestinis mokėtojas, galintis apmokėti studento sąskaitas.
Balansas	Kiek kokių sąskaitų apmokėta, neapmokėta
Buhalteris	Asmuo atliekantis vartotojų sąskaitų apskaitą,

	tvarkymu
Discover	Elektroninės bankininkystės mokomųjų kreditinių kortelių rūšis.
Duomenų bazė	Organizuotas (susistemintas, metodiškai sutvarkytas) duomenų rinkinys, kuriuo galima individualiai naudotis elektroniniu ar kitu būdu.
eCheck	Elektroninis čekis.
E-čekis	Elektroninis čekis.
E-laiškas	Elektroninis laiškas.
Enkriptavimas	Duomenų užkodavimas.
Fakultetas	Fakultetas kuriam priklauso nurodyta grupė
Grupė	Grupė vartotojų priklausančių tai specializacijai
ID	Identifikacinis numeris.
Integravimo testavimas	Komponentų, sujungtų į atskiras grupes testavimas, sukuriant sistemas ar posistemas.
Katedra	Apjungia kelias specializacijas
Kerberos	Tinklo autentifikavimo protokolas.
Klientas-vartotojas	Sistema naudojantysis asmuo, prisiregistravęs sistemoje
MASTERCARD	Elektroninės bankininkystės mokomųjų kreditinių kortelių rūšis.
Microsoft .NET Framework	„.NET Framework“ - tai tvarkyklių ir įrankių rinkinys, būtinas norint, kad kompiuteryje veiktų programos, sukurtos naudojant „.NET“ technologiją.
Microsoft ASP.NET	Technologija web-aplikacijų ir web-servisų kūrimui. Ji yra Microsoft.NET platformos sudedamoji dalis
PA	Panaudojimo atvejai.
Pardavėjas/tiekėjas	Paslaugos tiekėjas.
Pranešimai	Pranešimai skirti vartotoji, apie sistemos naudojimąsi arba susiję su sąskaitų apmokėjimu.
RUP	Rational Unifikuotas Procesas (Rational Unified

	Process)
Sąskaita	Apmokėta, neapmokėta pinigų srautų ataskaita
Sąskaitos detalizavimas	Surašyta daugiau duomenų apie vartotojo sąskaitą
Sąskaitų sąrašas	Keletas sąskaitų, už kelias paslaugas
Specializacija	Sritis kurioje specializuojamasi(biblioteka, įmokos už mokslą)
SSL	Security Socket Layer (SSL) - tai yra kriptografinis internetinis saugumo protokolas.
TCP/IP	Standartinis duomenų perdavimo protokolų rinkinys, kurio pagrindu veikia internetas bei daugelis privačių komercinių tinklų. Tinklo protokolas, duomenų perdavimui.
Teisės	Priskirtos teisės kurių reikia laikytis norit naudotis sistema
TLS	Transport Layer Security (TLS) - tai yra kriptografinis internetinis saugumo protokolas.
UML	Unifikuota modeliavimo kalba (Unified Modeling Language)
URL	Uniform Resource Locator (URL) - unifikuotas interneto išteklių adresai.
Vartotojas	Sistemos naudotojas bendresniu atveju
Visa	Elektroninės bankininkystės mokomųjų kreditinių kortelių rūšis.
Web programa	Žiniatinklio portalas pateikiantis informaciją susijusią su elektroniniais apmokėjimais, leidžiantis pasinaudoti su elektroniniais apmokėjimais susijusiomis funkcijomis, bei suteikiantis galimybę prieiti prie organizacijos informacinių resursų.

8. PRIEDAI

8.1. PRIEDAS 1: Apklausa

Prašome įvertinti žemiau pateiktas „elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ sistemos charakteristikas skalėje nuo 1 iki 10, 1 – žemiausias įvertinimas, 10 aukščiausias įvertinimas.

Lentelė 8.1-1. Sistemos kokybės vertinimas pagal ISO 9216 standartą anketa.

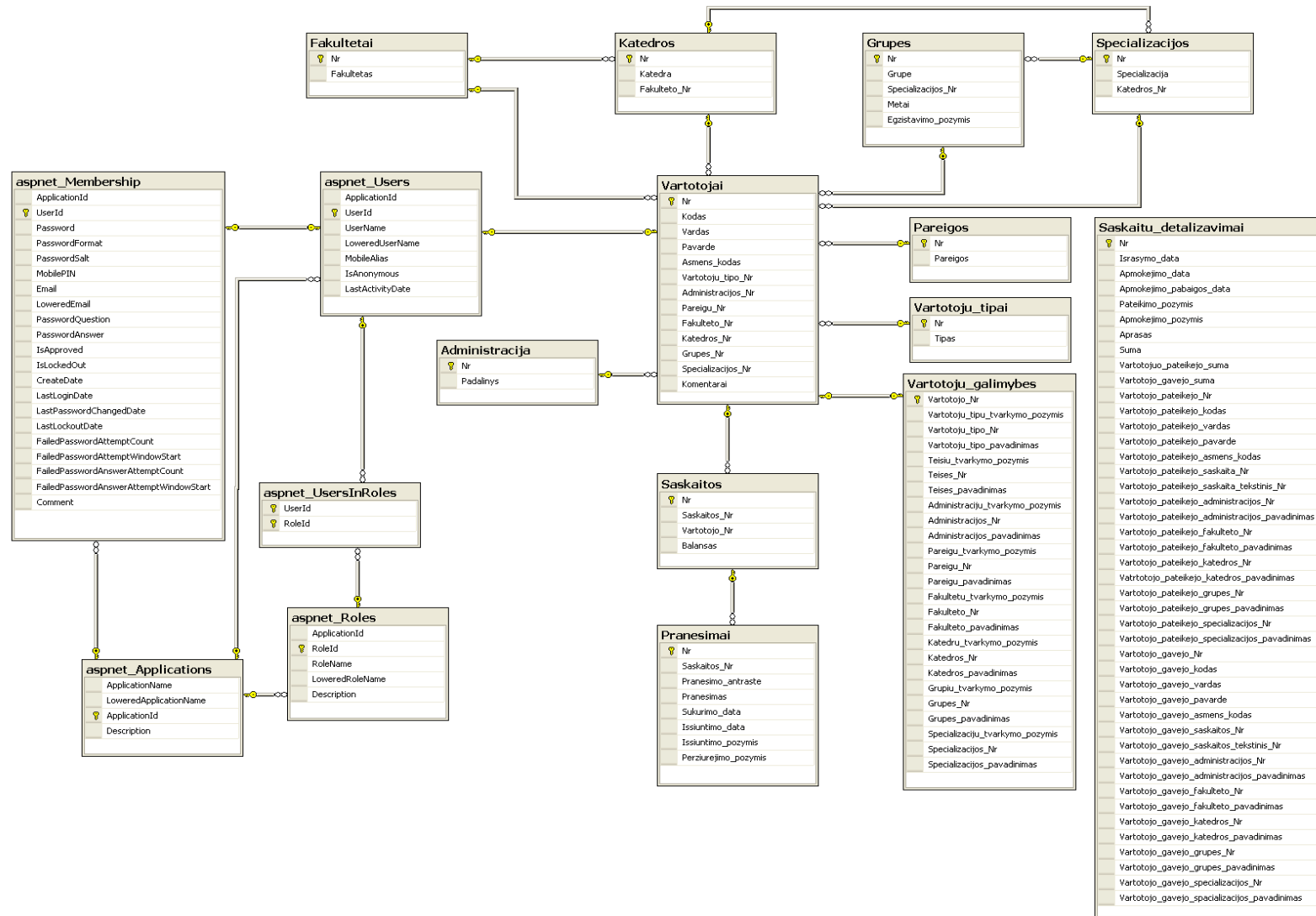
Funkcionalumas		
<i>Subkriterijus</i>	<i>Subkriterijaus parametrų aprašymai</i>	<i>Įvertinimas</i>
Tinkamumas	Sistema realizuoja visus nurodytus pradinis vartotojo reikalavimus sistemai.	
	Sistema realizuoja visus iš vartotojo surinktus funkcinius reikalavimus.	
	Sistema realizuoja visus iš vartotojo surinktus nefunkcinius reikalavimus.	
Tikslumas	Sistemos atliekami apmokėjimų skaičiavimai yra teisingi.	
	Sistemos atliekami pinigų pervedimai yra teisingi.	
Bendradarbiavimas	Sistema realizuoja duomenų importavimą iš kitų susijusių sistemų.	
	Sistema realizuoja duomenų eksportavimą į kitas susijusias sistemas.	
Atitikimas	Sistemos formuojamos ataskaitos atitinka įstatymų numatytus reikalavimus.	
	Sistema užtikrina įstatymiškai privalomą asmeninių duomenų saugumą/neplatinimą.	
Apsauga	Administraciniai ir paprasti sistemos vartotojai turi prisijungimo vardus ir slaptažodžius.	
	Sistemos DB apsaugota slaptažodžiu.	

	Visi sistemos atliekami duomenų persiuntimai per WEB servisus atliekami tik po sėkmingos vartotojo identifikacijos.	
	Vartotojų slaptažodžiai yra koduojami.	
Bendras		
Patikimumas		
<i>Subkriterijus</i>	<i>Subkriterijaus parametrų aprašymai</i>	<i>Įvertinimas</i>
Išbaigtumas	Ne daugiau kaip viena klaida per 7 dienas.	
Pakantumas klaidoms	Visos klaidos registruojamos ir apie jas pranešama vartotojui.	
	Klaida kuriame nors sistemos modulyje arba vartotojo sąsajos formoje neapriboja galimybių naudotis likusia sistemos dalimi.	
Atstatomumas	Įvykus klaidai sistema neleidžia koreguoti duomenų. Įrašomi tik pilni duomenų paketai. Palaikomos transakcijos.	
Bendras		
Lengvumas naudoti		
<i>Subkriterijus</i>	<i>Subkriterijaus parametrų aprašymai</i>	<i>Įvertinimas</i>
Suprantamumas	Vartotojo sąsaja nenaudoja jokių neįprastų formų ar būdų. Tie patys vartotojo sąsajos valdymo principai išlaikomi visoje sistemoje.	
	Kur reikia pateikiami paaiškinimai pačioje vartotojo sąsajoje.	
Išmokstamumas	Vartotojas susipažįsta ir išmoksta naudotis pagrindinėmis sistemos funkcijomis per vieną dieną (8 valandas).	
	Sistema turi vadovą pradedančiajam.	
	Sistema turi detalų vartotojo vadovą.	

Darbingumas	Sistemoje naudojamas lengvai suprantamas, intuityvus ir pastovus vartotojo sąsajos grafinis dizainas.	
	Vartotojui pateikiami pranešami apie visus jo atliekamus veiksmus. Ar jie buvo sėkmingi ar nesėkmingi ir kodėl.	
Bendras		
Efektyvumas		
<i>Subkriterijus</i>	<i>Subkriterijaus parametru aprašymai</i>	<i>Įvertinimas</i>
Laiko naudojimas	Duomenų atsinaujinimas po vartotojo sukūrimo neužtrunka 5 min	
	Duomenų atsinaujinimas po vartotojo trinimo neužtrunka 3 min.	
	Įprastinės vartotojo operacijos neužtrunka ilgiau kaip 5 sekundes.	
Resursų naudojimas	Laikas kai sistema 100% apkrauna jai paskirtą WEB serverį neviršija 1 valandos per parą.	
	Duomenų bazės apkraunamumas nedidėja eksponentiškai su kiekvienu prisijungusių vartotojų kreipimuisi tuo pačiu metu.	
Bendras		
Palaikomumas		
<i>Subkriterijus</i>	<i>Subkriterijaus parametru aprašymai</i>	<i>Įvertinimas</i>
Analizuojamumas	Prie visų klaidų pranešimų pateikiama informacija, kurioje vietoje ji įvyko.	
	Visos sistemos funkcijos yra tiksliai dokumentuotos. Viena funkcija vykdoma viename modulyje.	
Keičiamumas	Norint atlikti pakeitimus sistemoje nereikia būti susipažinus su visa sistemos realizacija.	
	Daugelį sistemos veikimo parametru galima keisti per parametrus. Nereikalaujant sistemos perkompiliavimo.	

Stabilumas	Pakeitimai viename sistemos modulyje neiššaukia pakeitimų ar klaidų kitame.	
Testuojamumas	Automatiniai sistemos testai apima didžiąją dalį sistemos funkcijų.	
	Sistema nereikalauja specialios aplinkos sukonfigūravimo ir paleidimo prieš pradėdant jos testavimą.	
Bendras		
Pernešamumas		
<i>Subkriterijus</i>	<i>Subkriterijaus parametrų aprašymai</i>	<i>Įvertinimas</i>
Pritaikomumas	Sistema gali dirbti Windows (98, 2000, XP, 2003, Vista) operacinėse sistemose.	
	Sistema gali būti naudojama Linux/Unix tipo operacinėse sistemose.	
Instaliuojamumas	Lengvas sistemos WEB dalies diegimas	
	Lengvas sistemos servisų diegimas	
	Lengvas sistemos parametrų konfigūravimas	
Prisitaikymas	Naudojami standartiniai protokoliai duomenų mainams sistemos viduje.	
Pakeičiamumas	Sistema turi didžiąją dalį funkcinių galimybių pasitaikančių alternatyviuose sprendimuose todėl gali būti naudojama vietoje jų.	
Bendras		

8.2. PRIEDAS 2: „Elektroninių mokėjimų universitetinėje programinė įranga“ duomenų bazės schema



Paveikslas 8.2-1. Duomenų bazės schema.

8.3. PRIEDAS 3: Sistemos dinaminis vaizdas

Aprašoma „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos dinaminis vaizdas. Pateikiamas sistemos veiklos diagramos.

8.3.1. Sistemos dinaminio vaizdo diagramos

8.3.1.1. Veiklos diagramos

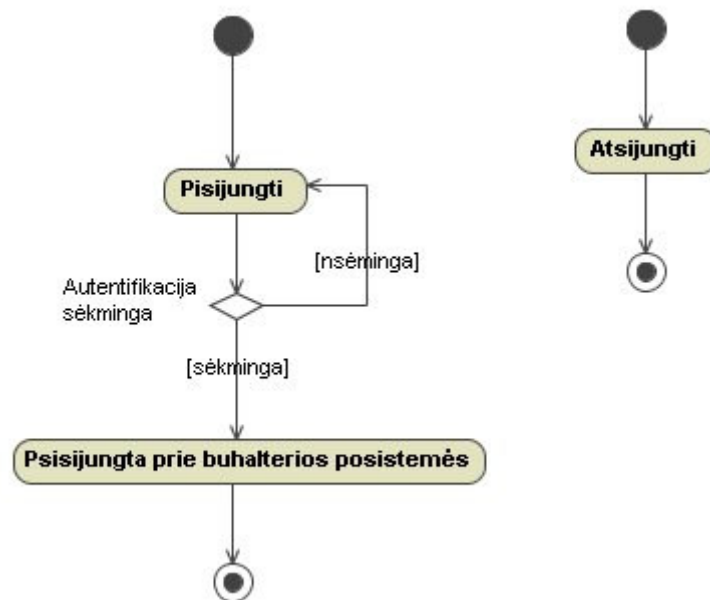
Pateikiamos „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos vartotojo(-ų) (buhalterio, pardavėjo/teikėjo, kliento-vartotojo), sistemos saugumo užtikrinimo veiklos diagramos.

8.3.1.1.1. Buhalterio veiklos diagramos

Pateikiamos „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos buhalterio veiklos diagramos.

8.3.1.1.1. „B1. Identifikavimas“ veiklos diagrama

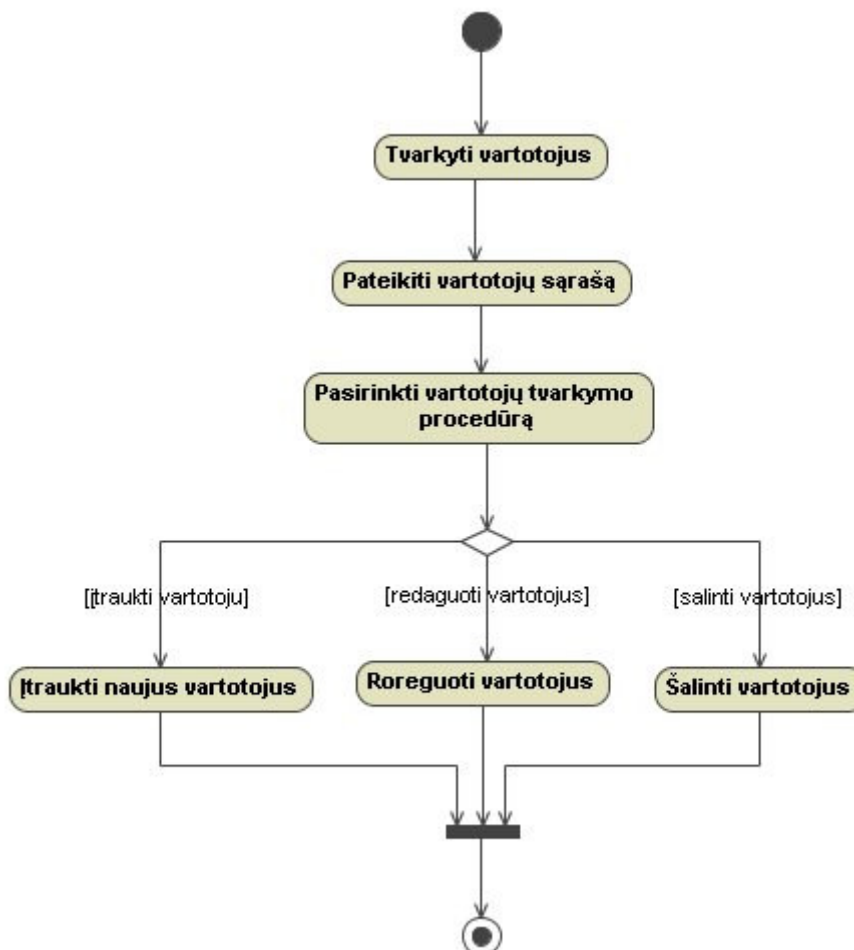
Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos *B1. Identifikavimas* veiklos diagrama.



Paveikslas 8.3-1. *B1. Identifikavimas* veiklos diagrama.

8.3.1.1.1.2. „B2. Tvarkyti vartotojus“ veiklos diagrama

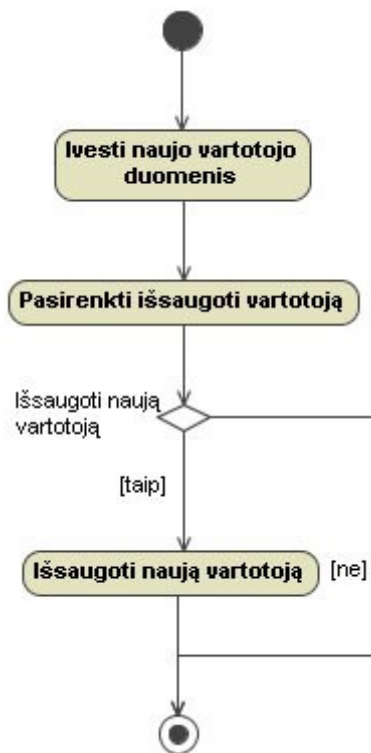
Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos B2. *Tvarkyti vartotojus* veiklos diagrama.



Paveikslas 8.3-2. B2. *Tvarkyti vartotojus* veiklos diagrama.

8.3.1.1.1.3. „B2.1. Įtraukti naujus“ veiklos diagrama

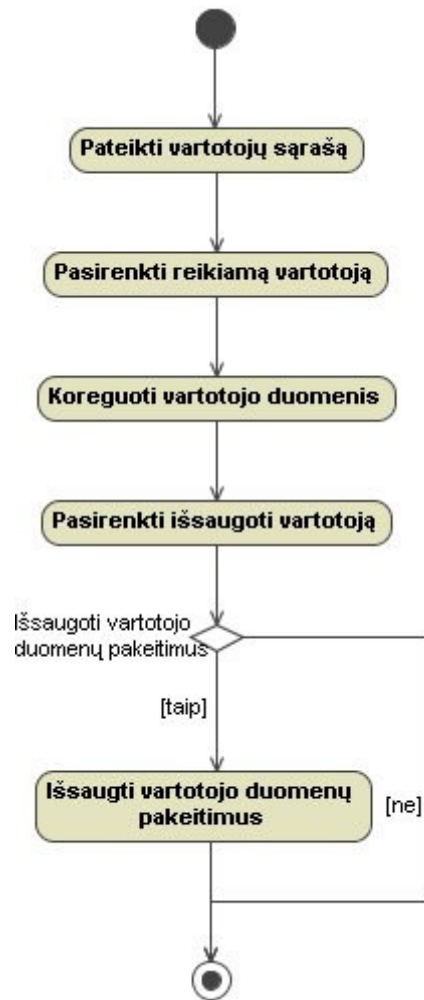
Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos B2.1. *Įtraukti naujus vartotojus* veiklos diagrama.



Paveikslas 8.3-3. B2.1. *Įtraukti naujus vartotojus* veiklos diagrama.

8.3.1.1.1.4. „B2.2. Koreguoti vartotojus“ veiklos diagrama

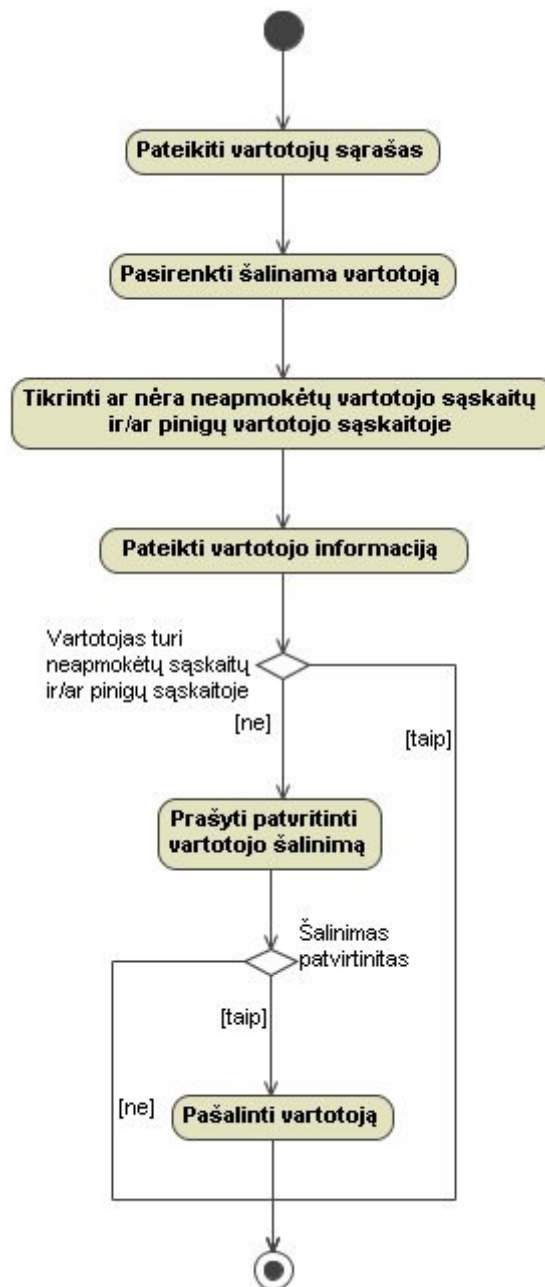
Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos B2.2. *Koreguoti vartotojus* veiklos diagrama.



Paveikslas 8.3-4. B2.2. *Koreguoti vartotojus* veiklos diagrama.

8.3.1.1.1.5. „B2.3. Šalinti vartotojus“ veiklos diagrama

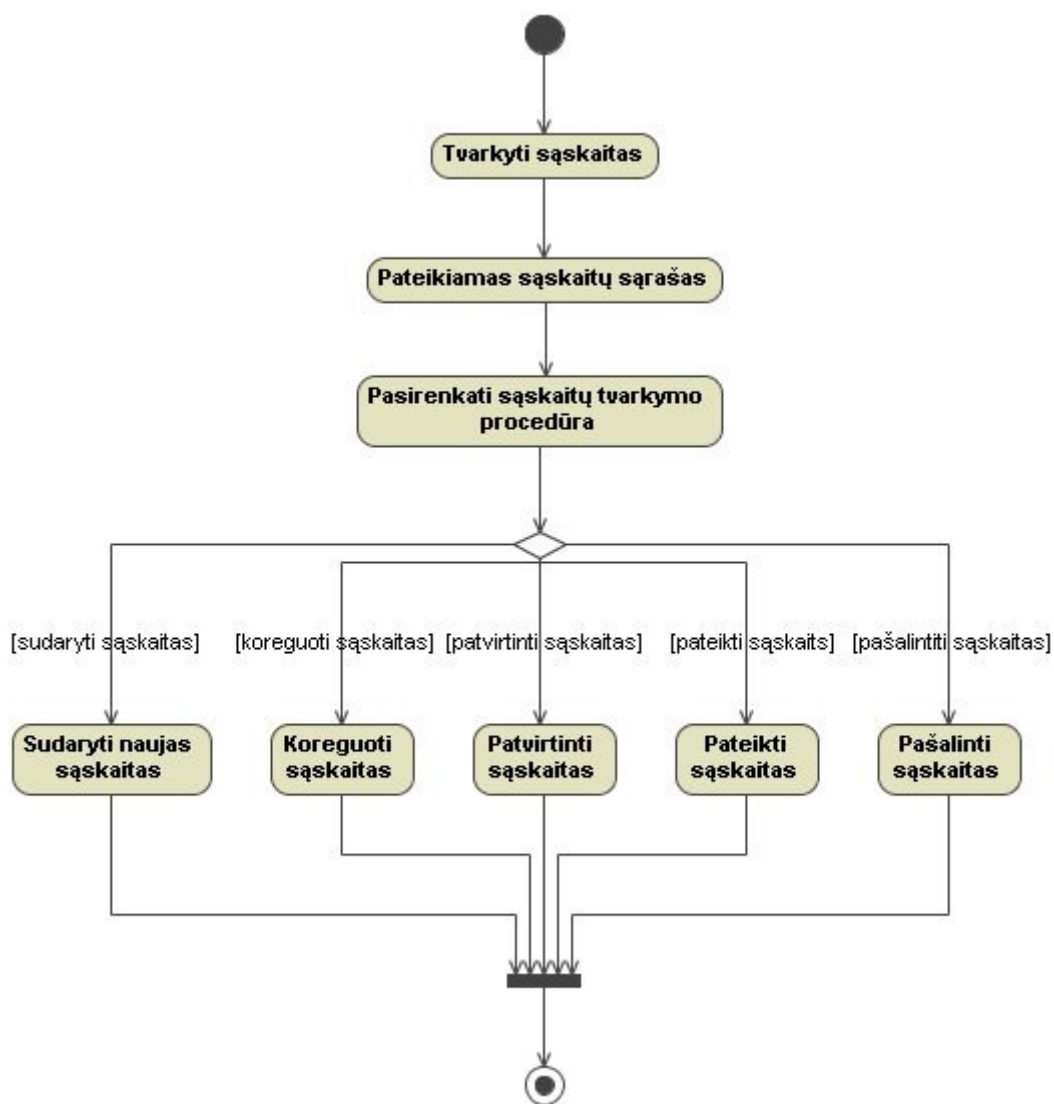
Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos B2.3. Šalinti vartotojus veiklos diagrama.



Paveikslas 8.3-5. B2.3. Šalinti vartotojus veiklos diagrama.

8.3.1.1.1.6. „B3. Sąskaitų tvarkymas“ veiklos diagrama

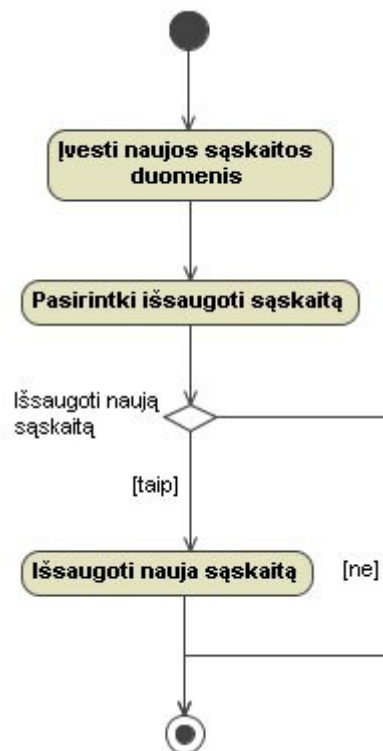
Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos *B3. Sąskaitų tvarkymas* veiklos diagrama.



Paveikslas 8.3-6. B3. Sąskaitų tvarkymas veiklos diagrama.

8.3.1.1.1.7. „B3.1. Sudaryti naujas sąskaitas“ veiklos diagrama

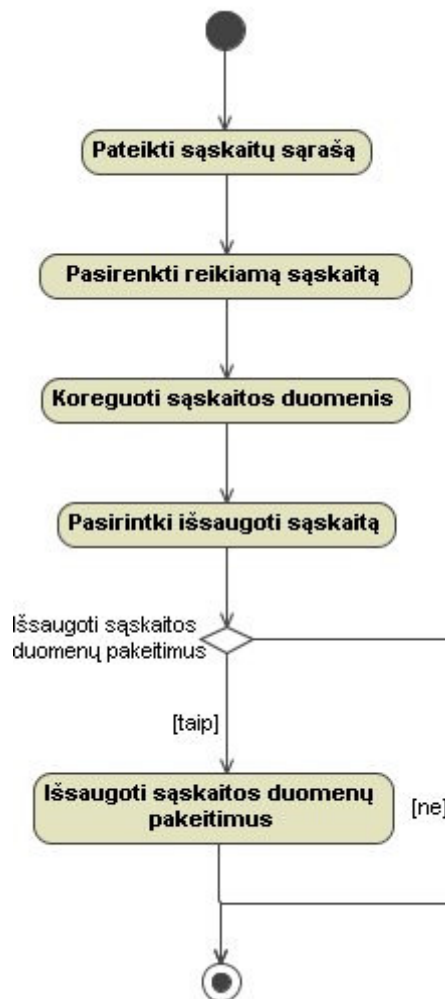
Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos *B3.1. Sudaryti naujas sąskaitas* veiklos diagrama.



Paveikslas 8.3-7. *B3.1. Sudaryti naujas sąskaitas* veiklos diagrama.

8.3.1.1.1.8. „B3.2. Koreguoti sąskaitas“ veiklos diagrama

Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos B3.2. *Koreguoti sąskaitas* veiklos diagrama.



Paveikslas 8.3-8. B3.2. *Koreguoti sąskaitas* veiklos diagrama.

8.3.1.1.1.9. „B3.3. Patvirtinti sąskaitas“ veiklos diagrama

Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos *B3.3. Patvirtinti sąskaitas* veiklos diagrama.



Paveikslas 8.3-9. *B3.3. Patvirtinti sąskaitas* veiklos diagrama.

8.3.1.1.10. „B3.4. Pateikti sąskaitas“ veiklos diagrama

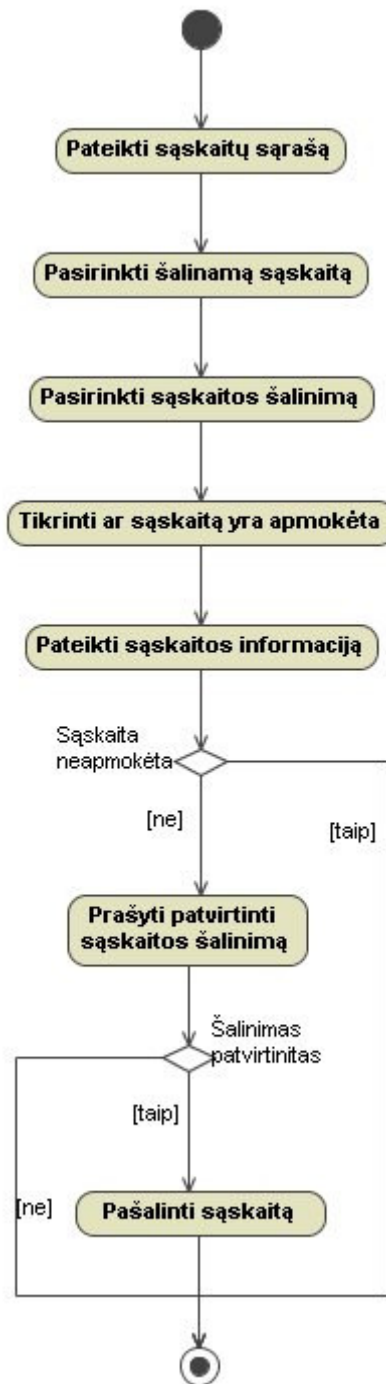
Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos B3.4. Pateikti sąskaitas veiklos diagrama.



Paveikslas 8.3-10. B3.4. Pateikti sąskaitas veiklos diagrama.

8.3.1.1.11. „B3.5. Pašalinti sąskaitas“ veiklos diagrama

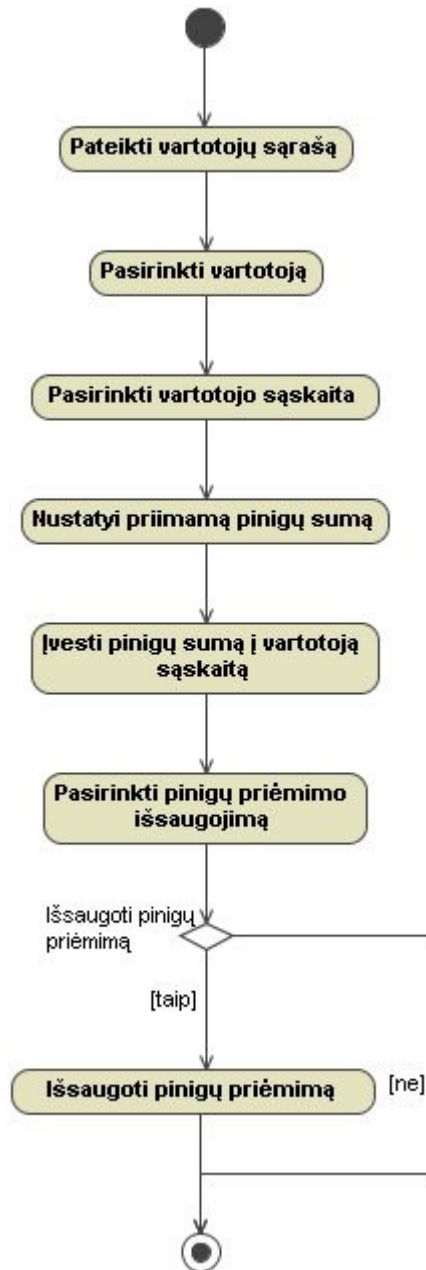
Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos B3.5. Pašalinti sąskaitas veiklos diagrama.



Paveikslas 8.3-11. B3.5. Pašalinti sąskaitas veiklos diagrama.

8.3.1.1.1.12. „B4. Pinigų priėmimas“ veiklos diagrama

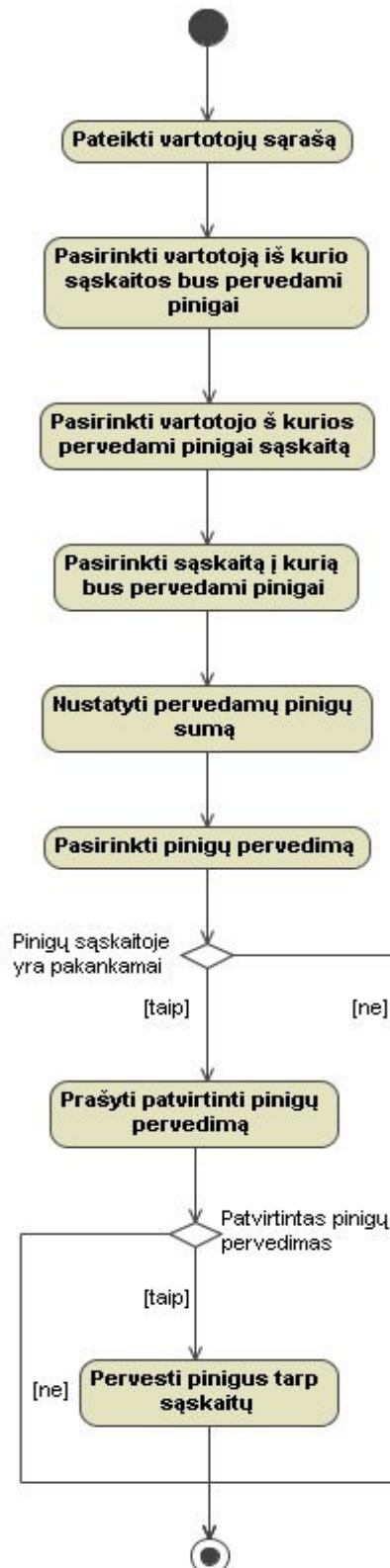
Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos *B4. Pinigų priėmimas* veiklos diagrama.



Paveikslas 8.3-12. *B4. Pinigų priėmimas* veiklos diagrama.

8.3.1.1.13. „B5. Piniginiai pervedimai“ veiklos diagrama

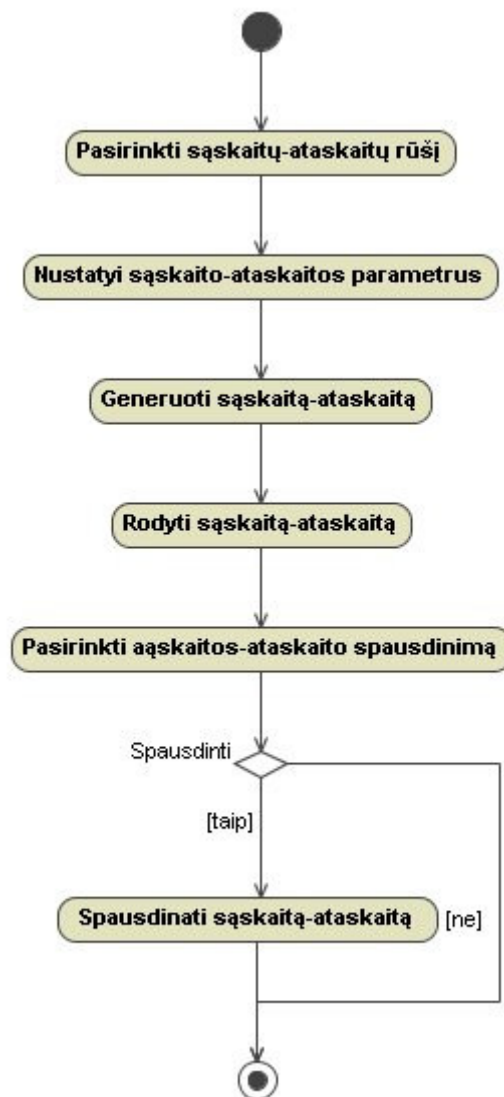
Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos *B5. Piniginiai pervedimai* veiklos diagrama.



Paveikslas 8.3-13. *B5. Piniginiai pervedimai* veiklos diagrama.

8.3.1.1.14. „B6. Sąskaitų-ataskaitų spausdinimas“ veiklos diagrama

Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos B6. *Sąskaitų-ataskaitų spausdinimas* veiklos diagrama.



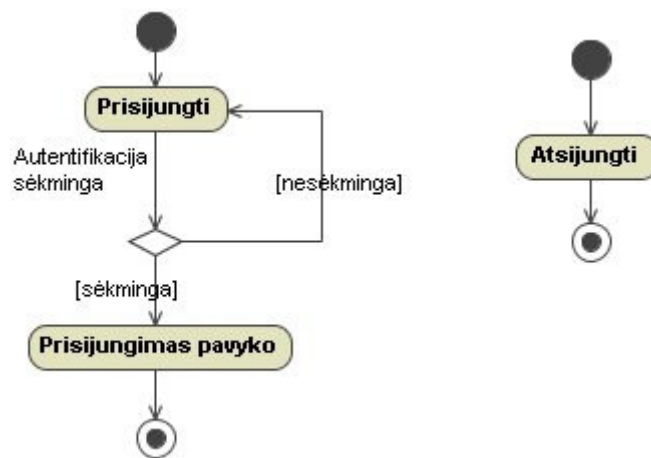
Paveikslas 8.3-14. B6. *Sąskaitų-ataskaitų spausdinimas* veiklos diagrama.

8.3.1.1.2. Pardavėjo/Teikėjo veiklos diagramos

Pateikiamos „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos pardavėjo/teikėjo veiklos diagramos.

8.3.1.1.2.1. „PT1. Identifikavimas“ veiklos diagrama

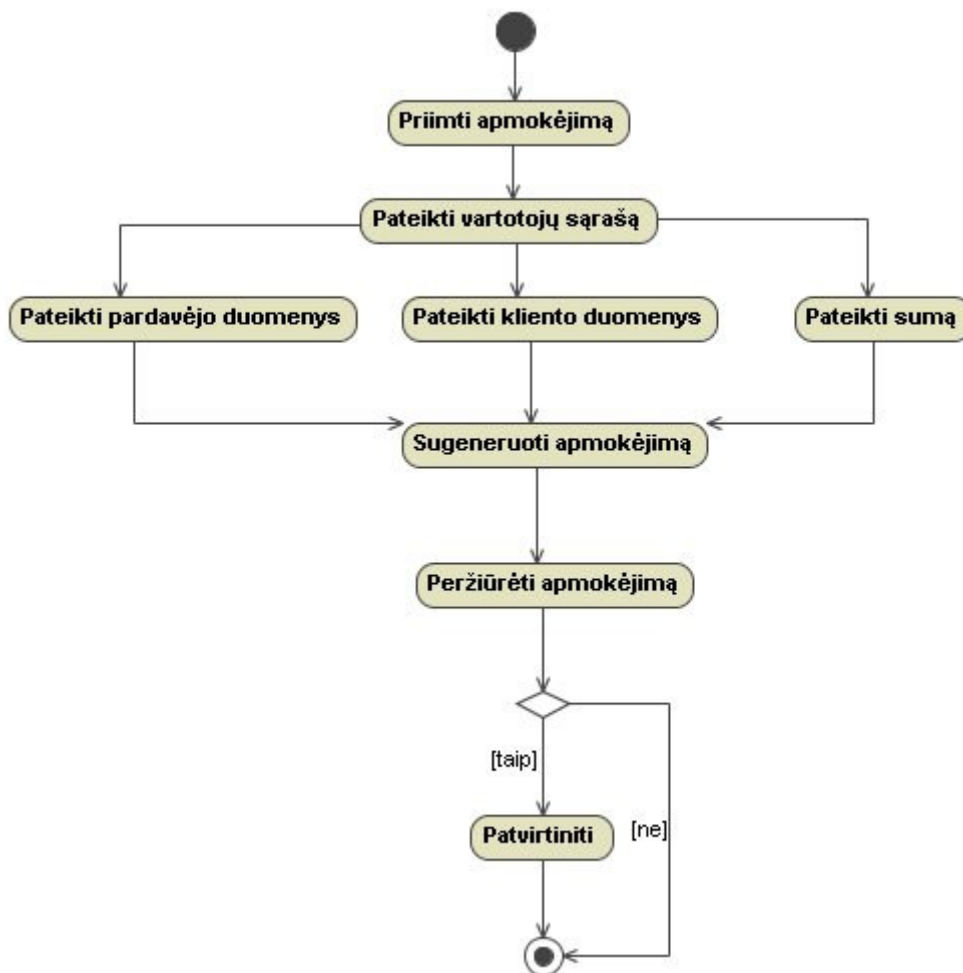
Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos *PT1. Identifikavimas* veiklos diagrama.



Paveikslas 8.3-15. *PT1. Identifikavimas* veiklos diagrama.

8.3.1.1.2.2. „PT2. Priimti apmokėjimą“ veiklos diagrama

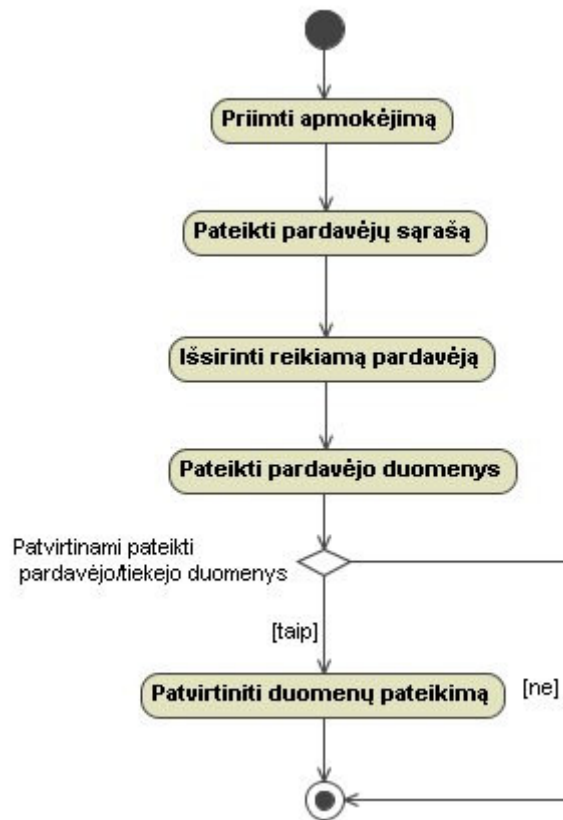
Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos *PT2. Priimti apmokėjimą* veiklos diagrama.



Paveikslas 8.3-16. *PT2. Priimti apmokėjimą* veiklos diagrama.

8.3.1.1.2.3. „PT2.1. Pateikti pardavėjo duomenis“ veiklos diagrama

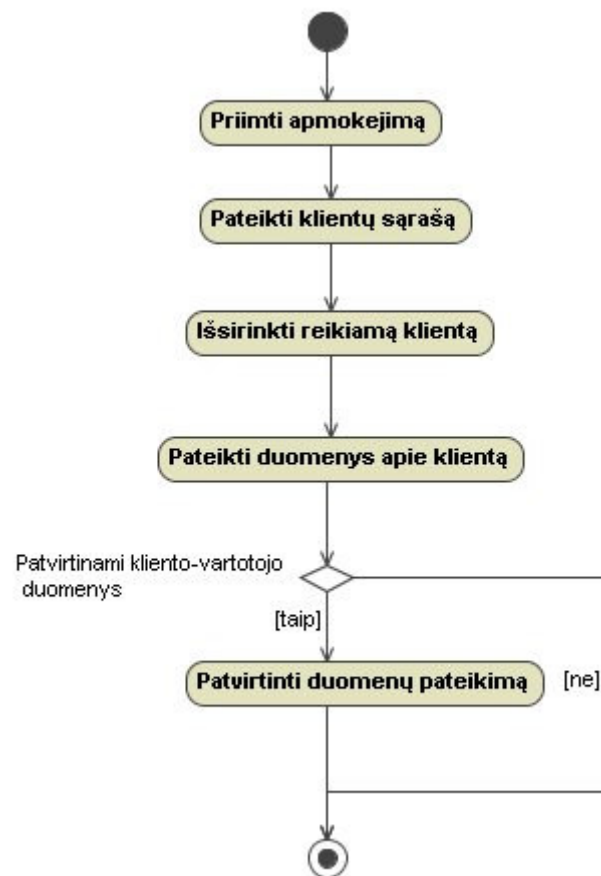
Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos *PT2.1. Pateikti pardavėjo duomenis* veiklos diagrama.



Paveikslas 8.3-17. *PT2.1. Pateikti pardavėjo duomenis* veiklos diagrama.

8.3.1.1.2.4. „PT2.2. Pateikti kliento duomenis“ veiklos diagrama

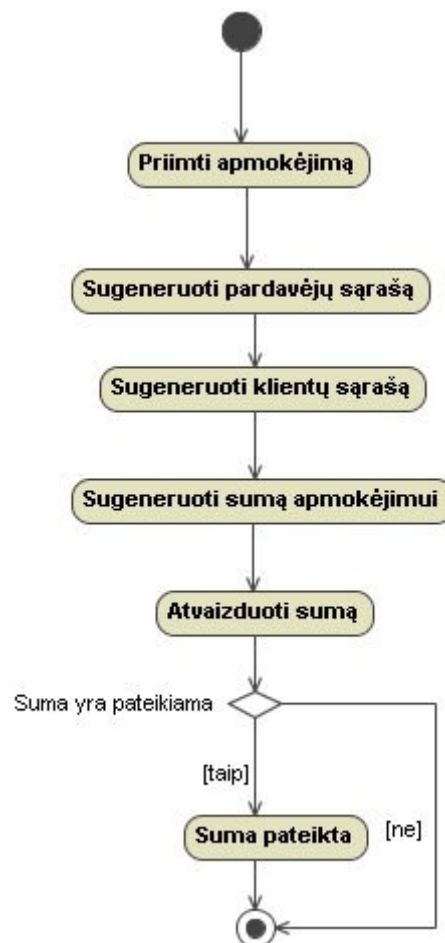
Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos *PT2.2. Pateikti kliento duomenis* veiklos diagrama.



Paveikslas 8.3-18. *PT2.2. Pateikti kliento duomenis* veiklos diagrama.

8.3.1.1.2.5. „PT2.3. Pateikti sumą“ veiklos diagrama

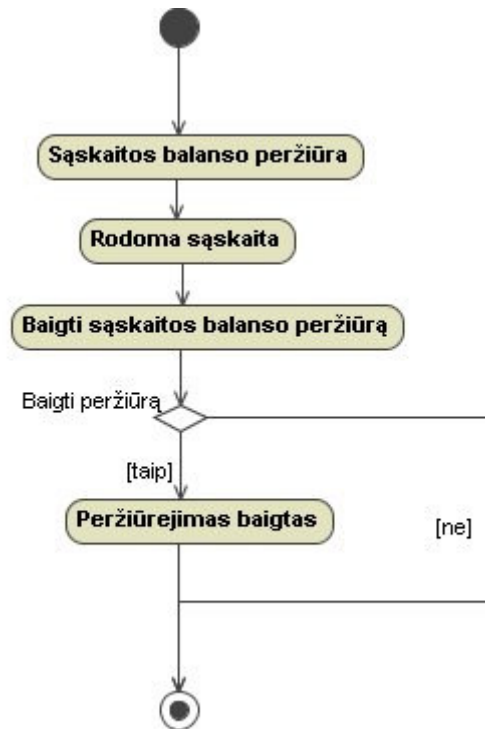
Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos *PT2.3. Pateikti sumą* veiklos diagrama.



Paveikslas 8.3-19. *PT2.3. Pateikti sumą* veiklos diagrama.

8.3.1.1.2.6. „PT3. Sąskaitos balanso peržiūra“ veiklos diagrama

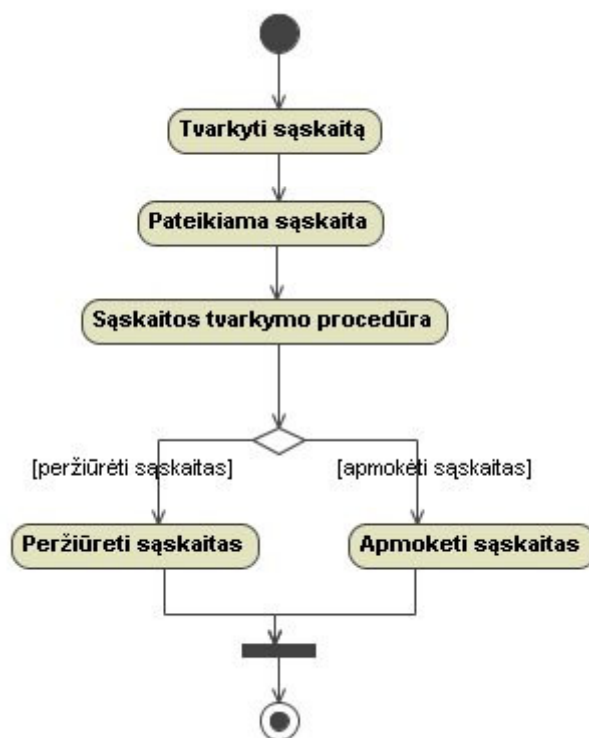
Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos *PT3. Sąskaitos balanso peržiūra* veiklos diagrama.



Paveikslas 8.3-20. *PT3. Sąskaitos balanso peržiūra* veiklos diagrama.

8.3.1.1.2.7. „PT4. Sąskaitos tvarkymas“ veiklos diagrama

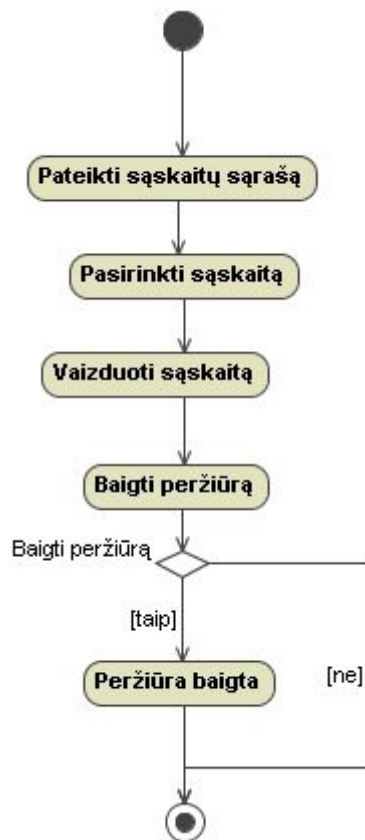
Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos *PT4. Sąskaitos tvarkymas* veiklos diagrama.



Paveikslas 8.3-21. *PT4. Sąskaitos tvarkymas* veiklos diagrama.

8.3.1.1.2.8. „PT4.1. Peržiūrėti sąskaitas“ veiklos diagrama

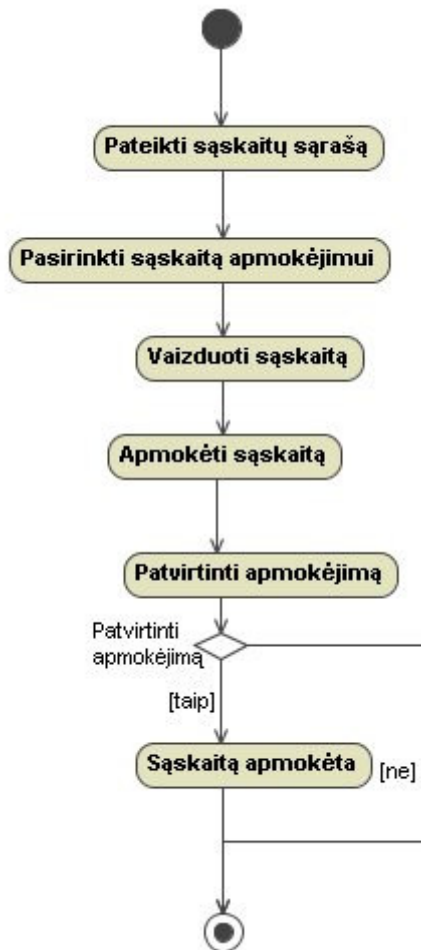
Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos *PT4.1. Peržiūrėti sąskaitas* veiklos diagrama.



Paveikslas 8.3-22. *PT4.1. Peržiūrėti sąskaitas* veiklos diagrama.

8.3.1.1.2.9. „PT4.2. Apmokėti sąskaitas“ veiklos diagrama

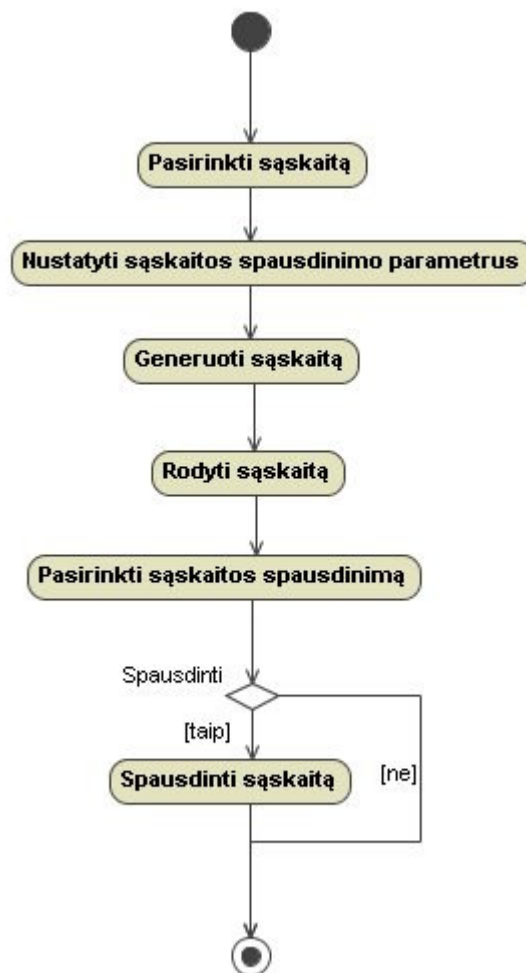
Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos *PT4.2. Apmokėti sąskaitas* veiklos diagrama.



Paveikslas 8.3-23. *PT4.2. Apmokėti sąskaitas* veiklos diagrama.

8.3.1.1.2.10. „PT5. Sąskaitų spausdinimas“ veiklos diagrama

Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos *PT5. Sąskaitų spausdinimas* veiklos diagrama.



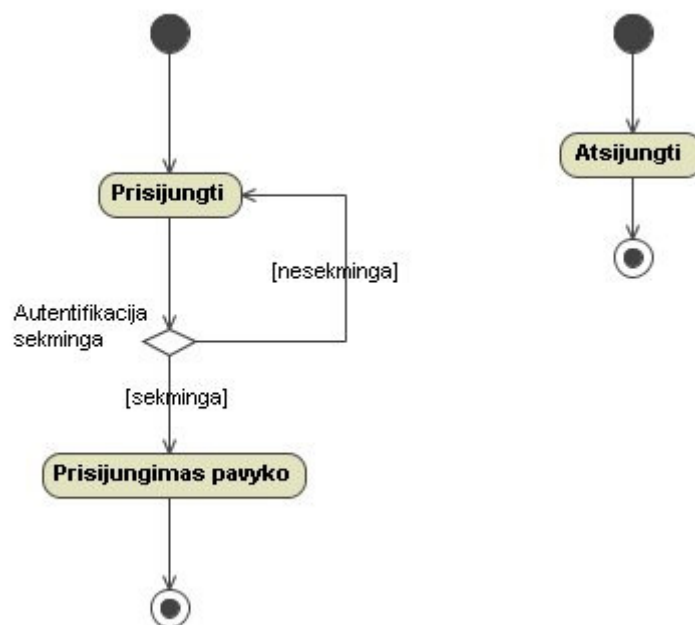
Paveikslas 8.3-24. *PT5. Sąskaitų spausdinimas* veiklos diagrama.

8.3.1.1.3. Kliento-vartotojo veiklos diagramos

Pateikiamos „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos kliento-vartotojo veiklos diagramos.

8.3.1.1.3.1. „KV1. Identifikavimas“ veiklos diagrama

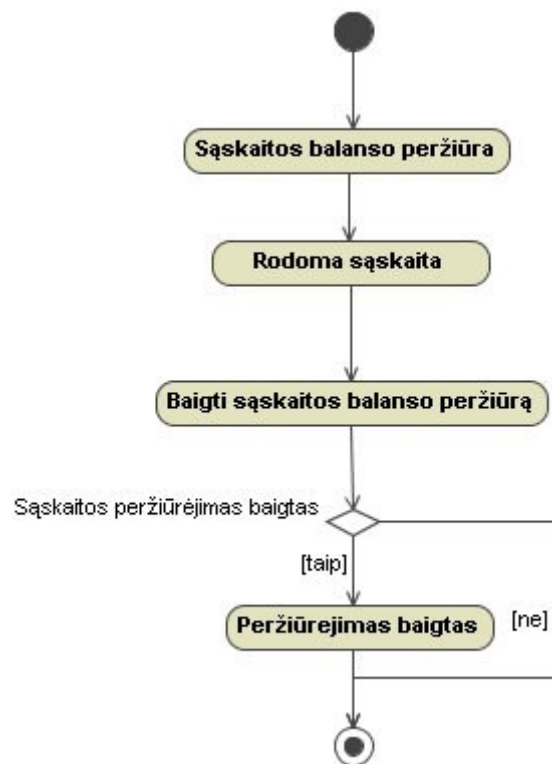
Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos *KV1. Identifikavimas* veiklos diagrama.



Paveikslas 8.3-25. *KV1. Identifikavimas* veiklos diagrama.

8.3.1.1.3.2. „KV2. Sąskaitos balanso peržiūra“ veiklos diagrama

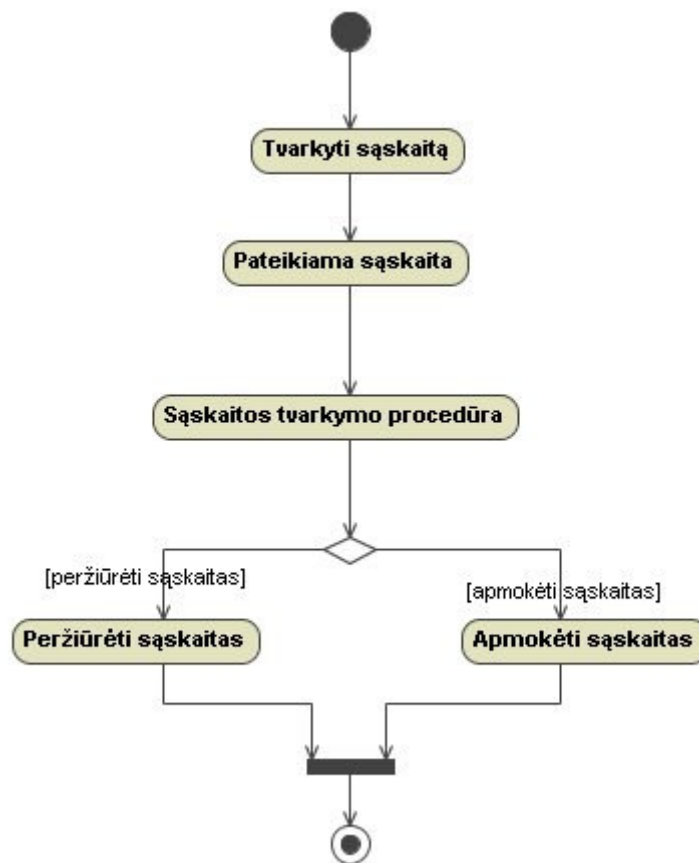
Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos KV2. *Sąskaitos balanso peržiūra* veiklos diagrama.



Paveikslas 8.3-26. KV2. *Sąskaitos balanso peržiūra* veiklos diagrama.

8.3.1.1.3.3. „KV3. Sąskaitos tvarkymas“ veiklos diagrama

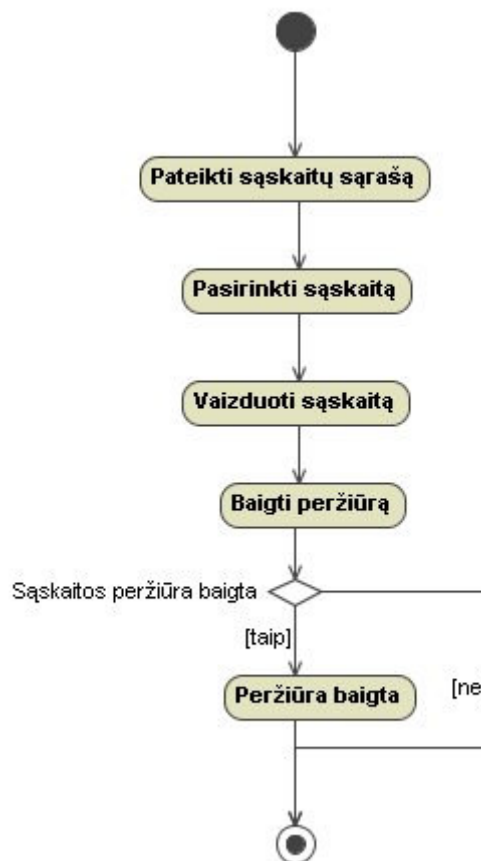
Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos *KV3. Sąskaitos tvarkymas* veiklos diagrama.



Paveikslas 8.3-27. *KV3. Sąskaitos tvarkymas* veiklos diagrama.

8.3.1.1.3.4. „KV3.1. Peržiūrėti sąskaitas“ veiklos diagrama

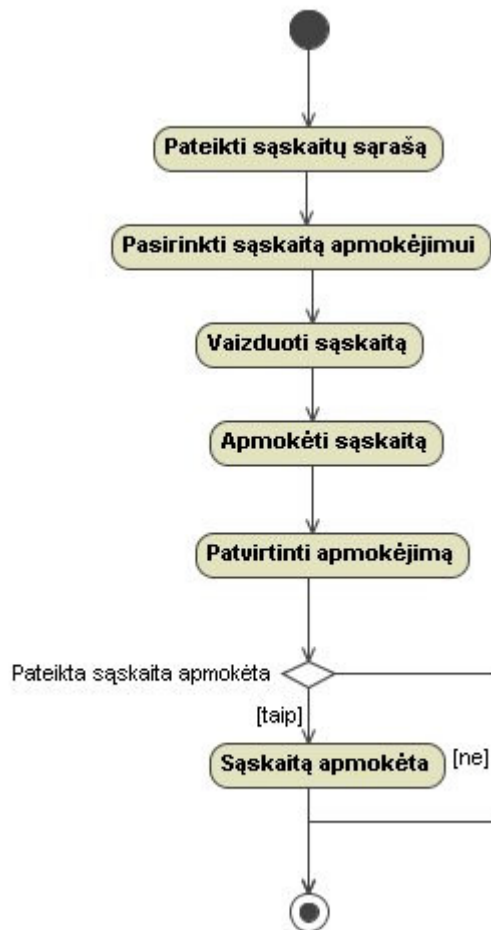
Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos *KV3.1. Peržiūrėti sąskaitas* veiklos diagrama.



Paveikslas 8.3-28. KV3.1. Peržiūrėti sąskaitas veiklos diagrama.

8.3.1.1.3.5. „KV3.2.Apmokėti sąskaitas“ veiklos diagrama

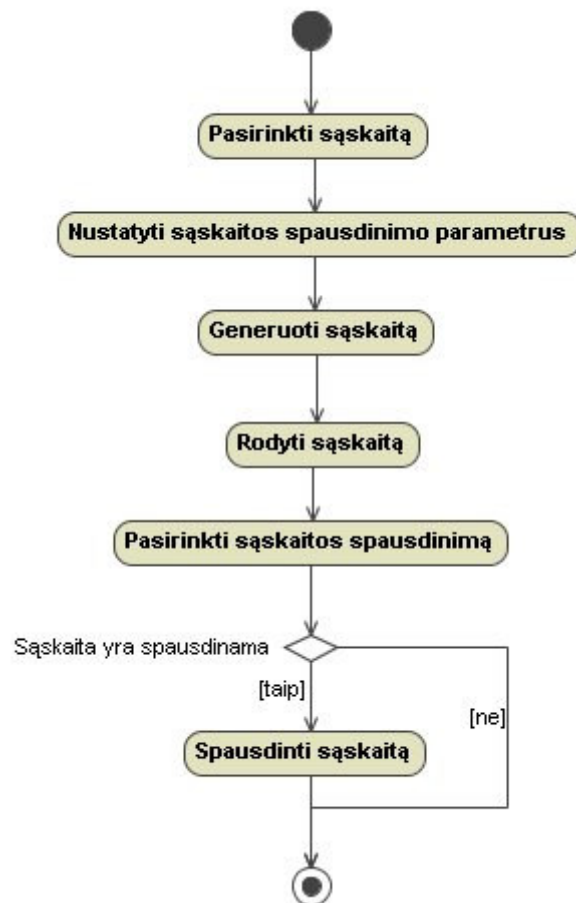
Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos *KV3.2.Apmokėti sąskaitas* veiklos diagrama.



Paveikslas 8.3-29. *KV3.2.Apmokėti sąskaitas* veiklos diagrama.

8.3.1.1.3.6. „KV4. Sąskaitų spausdinimas“ veiklos diagrama

Pateikiama „Elektroninių mokėjimų universitetinėje aplinkoje programinė įranga“ projektuojamos sistemos KV4. *Sąskaitų spausdinimas* veiklos diagrama.



Paveikslas 8.3-30. KV4. *Sąskaitų spausdinimas* veiklos diagrama.