



**KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
PROGRAMŲ INŽINERIJOS KATEDRA**

Vytautas Gavėnavičius

**Krašto apsaugos sistemos personalo duomenų
analizės ir kontrolės posistemė**

Magistro darbas

**Darbo vadovas
doc. dr. J. Adomavičius**

KAUNAS, 2006

**KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
PROGRAMŲ INŽINERIJOS KATEDRA**

**TVIRTINU
Katedros vedėjas
doc. dr. E. Bareiša
2006-01-10**

**Krašto apsaugos sistemos personalo duomenų
analizės ir kontrolės posistemė**

Informacinių technologijų magistro baigiamasis darbas

**Vadovas
doc. dr. J. Adomavičius
2006-01-10**

**Recenzentas
doc. dr. K. Baniulis
2006-01-10**

**Atliko
IFN - 3 gr. stud.
V. Gavėnavičius
2006-01-06**

KAUNAS, 2006

**INFORMACINIŲ TECHNOLOGIJŲ STUDIJŲ PROGRAMOS (62107T103)
KVALIFIKACINĖ KOMISIJA**

Pirmininkas: Kazys Kavaliauskas, dr. (UAB „Baltic Software Solutions“)

Sekretorius: Antanas Lenkevičius, docentas

Nariai: Jonas Kazimieras Maticikas, docentas

Kęstutis Motiejūnas, docentas

Bronius Paradauskas, docentas

Dalius Rubliauskas, docentas

docentas Arūnas Tomkevičius, dr. (Lietuvos ir Vokietijos UAB „IBS BALTIC“)

SANTRAUKA LIETUVIŲ IR ANGLŲ KALBOMIS (SUMMARY)

Sėkmingą kiekvienos įstaigos ar organizacijos veiklą lemiantis veiksnys yra gerai ir tinkamai organizuotas personalo valdymas ir apskaita. Šiuo metu Krašto apsaugos sistemos (toliau – KAS) personalą administruojantys darbuotojai duomenų apskaitai ir analizei naudojami KAS personalo apskaitos informacinės sistemos Baltpers teikiamomis paslaugomis. Magistro darbe atlikta analizė parodė, kad Baltpers sistemoje realizuotos duomenų kontrolės ir analizės priemonės netenkina KAS personalo specialistų tam tikrų poreikių.

Magistro darbe siūloma, kaip praktiškai realizuoti specifinius vartotojų poreikius – taikant Visual Basic for Applications programavimo kalbą sukurti nesunkiai įsisavinamą ir nebrangią personalo duomenų analizės ir kontrolės posistemės programinę įrangą. Tokia posistema galės naudotis KAS personalo specialistai, taip pat KAS finansų ir logistikos sričių specialistai, savo užduotims atlikti naudojantys personalo duomenis. Posistemė suteiks vartotojams galimybę pagal atitinkamai parinktus tikrinimo parametrus tikrinti personalo asmens kodo teisingumą, formuoti hierarchinį KAS organizacijos pavadinimą, skaičiuoti personalo darbo stažą, analizuoti ir tikrinti etatų užimtumą.

Vytautas Gavėnavičius

Personnel Data Analysis and Control Subsystem of National Defence System

Successful factor of the performance of each company or organisation is a well and adequately organised management and record keeping of personnel. At present, the staff administrating the personnel records of the National defence system of Lithuania (hereinafter - NDS) for recording and analysis of personnel data users Baltpers information system for personnel account for NDS. The analysis carried out in the Master's paper proved that the data controls and functions implemented in Baltpers system do not correspond with certain needs of the NDS personnel specialists.

The Master's paper suggests practical implementation of the specific needs of the users. It suggests applying Visual Basic for Application programming language with the help of which is easily assimilated and inexpensive composition of software of personnel data analysis and controlling subsystem. Such system might be used by personnel specialists of NDS, as well as military finance and logistics specialists, who use personnel data for carrying out their tasks. This subsystem will provide an opportunity for users to verify the personnel ID, form top-down organisational title of the NDS, calculate personnel work standing, analyse and check staff occupation on the basis of selected survey parameters.

TURINYS

SANTRUMPOS.....	8
LENTELIŲ SĄRAŠAS	9
PAVEIKSLŲ SĄRAŠAS	9
ĮVADAS	10
1. ANALITINĖ DALIS	12
1.1. Probleminės srities aplinkos analizė	12
1.2. Galimų problemų sprendimų būdų apžvalga	14
1.3. IS kūrimo būdų analizė	16
1.4. Programavimo priemonių analizė	19
2. PROGRAMINĖS ĮRANGOS PROJEKTAS	22
2.1. Projekto vykdymo sritis	22
2.1.1. Organizacijos veiklos srities aprašymas	22
2.1.2. Posistemės veiklos kontekstas	23
2.1.3. Veiklos padalinimas	23
2.1.4. Posistemės vartotojai	24
2.1.5. Veiklos informacinės architektūros modelis	25
2.2. Reikalavimai projektuojamai Posistemėi	26
2.2.1. Reikalavimų gavimo būdai	26
2.2.2. Projekto apribojimai	26
2.2.3. Vartotojo reikalavimai	27
2.2.4. Prototipai	28
2.3. Posistemės specifikacija	28
2.3.1. Programinės įrangos architektūra	28
2.3.1.1. Panaudojimo atvejų vaizdas	29

2.3.1.2.	Programos loginis vaizdas	30
2.3.1.3.	Procesų vaizdas	30
2.3.1.4.	Išdėstymo vaizdas	31
2.3.1.5.	Realizacijos vaizdas	32
2.3.2.	Posistemės reikalavimų specifikacija	33
2.3.2.1.	Funkcinių reikalavimų specifikavimas	33
2.3.2.2.	Nefunkcinių reikalavimų specifikavimas	33
2.3.3.	Posistemės struktūros specifikacija	35
2.3.4.	Duomenų struktūrų specifikacija	36
2.3.5.	Programinės įrangos komponentų specifikavimas	39
2.3.5.1.	Vartotojo sąsajos komponentas	40
2.3.5.2.	Veiklos paslaugų komponentas	41
2.3.5.3.	Duomenų valdymo komponentas	44
2.3.6.	Vartotojo sąsajos specifikacija	45
2.3.6.1.	Reikalavimai vartotojo sąsajai	45
2.3.6.2.	Vartotojo sąsajos realizacija	46
2.3.7.	Programinės įrangos kodavimas	47
2.4.	Projekto grafikas ir išlaidų planas	47
2.4.1.	Posistemės kūrimo biudžetas ir resursai	47
2.4.2.	Projekto išlaidos	48
2.4.3.	Projekto grafikas	48
2.5.	Vartotojo dokumentacija	48
3.	TYRIMO DALIS	51
3.1.	Projekto išeiša	51
3.1.1.	Atviri klausimai (problemos)	51
3.1.2.	Problemos diegimo aplinkai	51

3.1.3.	Neigiamas vartotojų nusiteikimas	51
3.2.	Rizikos įvertinimo ir mažinimo planas	51
3.2.1.	Posistemės kūrimo rizikos įvertinimas	51
3.2.2.	Rizikos valdymo ir mažinimo planas	52
3.3.	Produkto kokybės įvertinimas	53
3.4.	Posistemės tobulinimo planas	54
4.	EKSPERIMENTINĖ DALIS	55
4.1.	Programinės įrangos testavimas	55
4.1.1.	PĮ testavimo metodai	55
4.1.2.	PĮ testavimo planas	56
4.1.3.	Testavimo procedūros ir rezultatai	57
4.2.	Programinės įrangos diegimas	58
	IŠVADOS	59
	NAUDOTA LITERATŪRA	60
	TERMINŲ ŽODYNAS	61
	PRIEDAI	

SANTRUMPOS

ADO (angl. MS ActiveXData Objects) – biblioteka, leidžianti programuotojams su kitomis programinėmis aplinkomis kurti ir valdyti (administruoti) Access duomenų bases;

API – taikomųjų programų sąsaja;

Baltpers – Krašto apsaugos sistemos personalo apskaitos informacinė sistema;

CASE - kompiuterizuota programų inžinerija;

CDB – centrinė duomenų bazė;

COM (angl. Component Object Model) – komponentinis objektų modelis;

DAO (angl. MS Data Access Object) – Access programos turima biblioteka, leidžianti kurti, palaikyti ir valdyti DB;

DB – duomenų bazė;

DBVS – duomenų bazių valdymo sistema;

ERD - esybių - ryšių diagrama;

GUI – grafinė vartotojo sąsaja;

IA – informacinė architektūra;

IS (angl. Information System) – informacinė sistema;

ISA – informacinės sistemos architektūra;

KAS – krašto apsaugos sistema;

MS – sutrumpintas Microsoft kompanijos pavadinimas;

ODBS (angl. Open Database Connectivity) - programa sujungianti skirtingas duomenų bases;

OLE (angl. Object Linking and Embedding) – objektų sąryšiai ir įterpimas;

OS (angl. Operating System) – operacinė sistema;

PĮ – programinė įranga;

SQL – speciali reliacinių DB tvarkymo kalba (užklausų kalba);

TCP/IP – tinklo protokolas duomenų perdavimui;

UML (angl. Unified Modelling Language) – objektiškai orientuoto modeliavimo kalba;

UCM (angl. Use Case Model) – panaudojimo atvejų modelis;

VBA – Microsoft Office taikomuosiose programose integruota Visual Basic kalba;

VB – objektinio programavimo kalba Visual Basic;

VIA – veiklos informacinė architektūra.

Lentelių sąrašas

1. lentelė. Veiklos įvykių sąrašas	24
2 lentelė. Vartotojas – personalo specialistas	25
3 lentelė. Vartotojas – personalo padalinio vadovas	25
4 lentelė. Duomenų struktūrų modelio esybės	39
5 lentelė. Projekto resursai	48
6 lentelė. Posistemės kūrimo rizikos	53
7 lentelė. Rizikos valdymo ir mažinimo planas	54
8 lentelė. Posistemės vartotojo sąsajos testavimo veiksmų planas	57
9 lentelė. Posistemės funkcijų testavimo veiksmų planas	57
10 lentelė. Atitikties GUI standartui testavimo veiksmų planas	58
11. lentelė. Pradinių duomenų failas	65
12 lentelė. Funkcijos „Asmens kodo tikrinimas“ testo rezultatai	65
13 lentelė. Funkcijos „Tarnybos laikas“ kontrolinio testo rezultatai	65
14 lentelė. Prisijungimo formos kontrolinio testo rezultatai	66
15 lentelė. Formos „Asmens Kodo“ Tikrinimas testo rezultatai	66

Paveikslų sąrašas

1 pav. Organizacijos veiklos personalo srityje panaudojimo atvejų (Use Case) diagrama	22
2 pav. Veiklos konteksto diagrama	23
3 pav. Posistemės kūrimo technologinis procesas	25
4 pav. Duomenų analizės ir kontrolės proceso VIA modelis.....	26
5 pav. Posistemės veiklos „Duomenų analizė ir kontrolė“ uždavinių modelis.....	29
6 pav. Posistemės išskaidymas į paketus aukščiausiam lygyje	30
7 pav. Vartotojo bendradarbiavimo su Posisteme sekų diagrama.....	31
8 pav. Fizinė Posistemės išdėstymo diagrama	32
9 pav. Posistemės realizacijos modelis	33
10 pav. Posistemės suskirstymas į komponentus	36
11 pav. Duomenų struktūrų klasių diagrama	37
12 pav. Vartotojo sąsajos paketo klasių diagrama	41
13 pav. Paketo Vartotojo Paslaugos klasių diagrama.....	42
14 pav. Paketo Funkcijos klasių diagrama.....	43
15 pav. Paketo Saugumas klasių diagrama	44
16 pav. Paketo Duomenų Importas Eksportas klasių diagrama	45
17 pav. Pagrindinė forma „Duomenų kontrolės posistemė“	46

ĮVADAS

Sėkmingai kiekvienos įstaigos ar organizacijos veiklai labai svarbus yra žmogiškųjų išteklių, arba personalo, veiksnys. Tinkamai organizuotas personalo valdymas ir apskaita yra vieni iš lemiamų veiksnių siekiant kuo geresnių įstaigos ar organizacijos veiklos rezultatų.

Krašto apsaugos sistemos (toliau — KAS) personalą administruojantys darbuotojai apskaitydami ir analizuodami personalo duomenis naudojami KAS personalo registracijos, šaukimo, tarnybos ir mobilizacinio rezervo apskaitos informacinės (toliau — Baltpers) sistemos paslaugomis.

Baltpers sistema sukurta įgyvendinant Lietuvos Respublikos Krašto apsaugos ministerijos ir Švedijos nacionalinės karinės prievolės administracijos sutartį. Sistemos sukūrimu rūpinosi Švedijos vyriausybė. Šis tarptautinis projektas yra įgyvendintas trijų Baltijos valstybių gynybos institucijose. Baltpers sistemoje saugoma informacija apie KAS personalą (Lietuvoje gyvenančius būtiniosios karinės tarnybos šauktinius, karo prievolinius, rezervistus bei profesionaliosios karinės tarnybos karius ir pan.), taip pat informacija apie KAS padalinius ir organizacijas. Panaudota kliento — serverio architektūra. Visa informacija kaupiama centrinėje duomenų bazėje.

Vienas pagrindinių reikalavimų duomenims yra jų patikimumas ir tikslumas. Daugelis kuriamų informacinių sistemų prieš jas projektuojant nėra pakankamai ištiriamos. Praeina daug laiko, kol sukuriamas tokia sistema, ir tik tada pamatoma, kad ji funkcionuoja ne taip, kaip tikėtasi, arba netenkina visų vartotojų poreikių. Taip projektų kūrėjai praranda daug laiko, dideles investicijas. Baltpers sistemoje realizuotos standartinės duomenų kontrolės funkcijos netenkina tam tikros grupės KAS personalo specialistų, rengiančių norminius dokumentus, skaičiuojančių pensijas ir pan. poreikių. Jiems reikalingos papildomos personalo duomenų analizės ir kontrolės funkcijos, kurios nėra įdiegtos Baltpers sistemoje. Be to, personalo duomenys turi būti teisingi ir patikimi.

Tad magistro darbo (toliau — Darbas) tiriamasis objektas yra KAS personalo duomenų analizė ir kontrolė, o tyrimo tikslas – ištirti personalo duomenų analizės ir tikrinimo procesų kompiuterizavimo galimybę.

Darbo rezultatai pateikiami magistro darbo apraše (toliau – Dokumentas). Dokumento paskirtis – pateikti probleminės srities analizę ir problemų sprendimo būdus, aprašyti sukurtos programinės įrangos (toliau — PI) techninius — projektinius sprendimus, atlikti PI eksperimentinį tyrimą ir ištirti jos kokybę, taip pat aprašyti, kaip dirbti su programa.

Teorines prielaidas sukurti reikiamą programinę įrangą sudaro projektuotojo ir projekto vykdytojo magistro studijų metu įgytos duomenų bazių ir informacinių sistemų, objektinio programavimo, programavimo inžinerijos, sistemų architektūros, formalių specifikacijų analizės ir kitų studijų modulių žinios. Aptariamą problemą sprendimą palengvina didelis problema nagrinėjančios teorinės literatūros kiekis.

Darbo tema yra labai aktuali praktine prasme KAS personalo srityje dirbantiems specialistams ir vadovams, nes specifiniams vartotojų poreikiams tenkinti personalo apskaitos sistema Baltpers neturi realizuotų funkcijų ir reikalingus skaičiavimus vartotojai turi atlikti rankiniu būdu. Galima sakyti, kad sukurta PĮ yra praktinė specifinių vartotojų poreikių realizacija.

Darbo naujumą sudaro tai, kad kuriama nauja programinė įranga, kokios KAS personalo specialistai dar neturėjo.

Darbo privalumas — sukurta PĮ galės naudoti ne tik KAS personalo, bet ir kitų veiklos sričių (finansų, logistikos ir pan.) specialistai, kurie savo užduotims atlikti naudoja personalo duomenis.

Buvo nuspręsta kurti personalo duomenų analizės ir kontrolės programinę įrangą. PĮ sukurti buvo parengtas projektas:

- atlikta išsami esamos programinės įrangos analizė, išskirti jos privalumai bei trūkumai;
- išanalizuoti vartotojų reikalavimai, iš jų išskirti funkciniai ir nefunkciniai reikalavimai;
- sudarytas programinės įrangos sistemos projektas;
- sudarytas programinės įrangos architektūrinis modelis;
- pagal išskirtas posistemes sudarytos programos klasių struktūros ir nustatyti klasių tarpusavio ryšiai;
- numatytos galimos duomenų struktūros;
- sudarytas projekto atlikimo grafikas;
- nustatytos projekto rizikos ir jų išvengimo priemonės;
- sukurta vartotojo sąsaja;
- sudarytas sistemos testavimo planas ir nustatyti testavimo metodai jam realizuoti;
- parengta vartotojo dokumentacija.

Idėjos iliustracijai sukurta personalo duomenų analizės ir kontrolės programinė įranga. Visa tai detaliau išdėstoma DOKUMENTE.

Prieš įgyvendinant projektą pirmiausia reikia atlikti problemų ir problemų sprendimo būdų detalią analizę.

1. ANALITINĖ DALIS

1.1. Probleminės srities aplinkos analizė

Šioje projekto dalyje detaliam analizuojami projektuojamos sistemos ir problemų sprendimo būdai, programos pirminių duomenų ir rezultatų prasmė, pagrindžiamas bei įvertinamas priimtas sprendimas, kaip spręsti problemas, parenkamas arba kuriamas uždavinio sprendimo algoritmas.

Kaip minėta šio Dokumento įvade, pagrindinė problema yra ta, kad Baltpers sistemoje realizuotos duomenų kontrolės funkcijos netenkina tam tikros grupės KAS personalo specialistų poreikių. Be to, Krašto apsaugos sistemoje įdiegtos ir funkcionuoja buhalterinę apskaitą, personalo apskaitą, turto apskaitą, biudžeto planavimą, logistiką ir kitas Krašto apsaugos sistemos veiklos sritis bei valdymą kompiuterizuojančios informacinės sistemos (toliau — IS). Baltpers sistemoje saugomus KAS asmenų ir organizacijų duomenis naudoja programų parengimo ir biudžeto planavimo sistema (toliau — PPBS VIS) ir logistikos valdymo sistema (toliau — LogVis). Į kitas KAS informacines sistemas importuojami personalo duomenys turi būti teisingi ir patikimi. Detaliau išanalizuosime problemas, susijusias su personalo asmens kodo teisingumu, etatų užimtumo tikrinimu, tikslu KAS padalinių pavadinimų formavimu ir tarnybos laiko apskaičiavimu.

Baltpers sistema sudaryta taip, kad būtų galima apdoroti ir išsaugoti didelį kiekį duomenų, susijusių su kasdien atliekamu darbu ir taip palengvinti varginantį ir daug laiko atimantį rankų darbą. Kad informacija būtų viską apimanti ir tinkama darbui, duomenys į sistemą turi būti įvedami teisingi ir tikslūs. Duomenims įvesti numatytos specialios formos, nusakančios įvedamos informacijos rūšį. Apie kiekvieną konkretų asmenį kaupiama informacija, kuri yra prieinama jos naudotojams. Baltpers duomenų bazėje (toliau – DB) duomenims saugoti ir valdyti naudojama Access aplinka. Access programa leidžia kiekvienam DB lentelės laukui nustatyti lauko savybes (charakteristikas) [11].

Su Access galima atlikti šias duomenų kontrolės funkcijas:

- kontroliuoti, kad įvedant duomenis į lentelės lauką šio lauko reikšmė būtų įvesta, t.y. laukui būtų suteikta būtina reikšmė (angl. Required);
- nustatyti bendrą simbolių kiekį, reikalingą atvaizduoti skaitinei reikšmei, t.y. skaitinės reikšmės tikslumą (angl. Precision) ir skalę (angl. Scale);
- kontroliuoti (tikrinti) įvedamų lauko reikšmių korektiškumą, t.y. nustatyti tikrinimo taisyklę (angl. Validation Rule) ir tikrinimo tekstą (angl. Validation Text), pvz. <100);
- laukui nustatyti įvesties kaukę (angl. Input Mask), kad nebūtų įmanoma įvesti klaidingų duomenų ir taip padidinant įvedamų duomenų patikimumą.

Access programa turi daug įvairių funkcijų, skirtų apdoroti įvairių tipų duomenis (pvz. Sum, Avg, Max, Min ir kt.) [3]. Tačiau tam tikrai grupei KAS personalo specialistų (rengiančių

norminius dokumentus, skaičiuojančių pensijas ir pan.) reikalingos papildomos duomenų analizės ir kontrolės funkcijos, kurios nėra įdiegtos Baltpers sistemoje. Personalo specialistų specializuotiems poreikiams (skaičiavimams, duomenų analizei ir kontrolei) standartinių Access funkcijų nebeužtenka. Labai aktualu tapo personalo apskaitos sistemos funkcionalumo praplėtimas.

Baltpers sistemoje kiekvienas įrašas apie KAS personalą registruojamas tos dienos data, pvz. 2002-05-21. Tuo tarpu personalo socialinės saugos ir pensijų skyriaus specialistai, kuriems reikia skaičiuoti pensijas, apskaičiuoti darbuotojo stažą pagal darbuotojo priėmimo į darbą ir atleidimo iš darbo datas ir rezultata gauti metais, mėnesiais ir dienomis. Šiuo metu tarnybos stažo apskaičiuojamas pagal priėmimo ir atleidimo datą rankiniu būdu. Kompiuterizuota tarnybos stažo apskaičiavimo funkcija padės spręsti minėtą problemą.

Kai darbuotojas priimamas į darbą, personalo specialistas įrašo į Baltpers sistemą krašto apsaugos sistemos darbuotojo asmens kodą. Asmens kodas tampa pagrindiniu darbuotojo identifikatoriumi (toliau – ID). Vėliau šio asmens duomenys apdorojami ir modifikuojami remiantis asmens kodu, todėl labai svarbu, kad į sistemą įvestas asmens kodas būtų teisingas, t.y. atitiktų būtent tam asmeniui Valstybinio asmens duomenų registro suteiktą kodą. Standartinės Access funkcijos tik nustato bendrą simbolių kiekį, reikalingą atvaizduoti skaitinei reikšmei, t.y. skaitinės reikšmės tikslumą. Todėl reikalingas asmens kodo teisingumo patikrinimas pagal metodiką, kurios panaudojimą reglamentuoja Lietuvos standartas RS-1185-91. Paanalizuosime asmens kodo teisingumo tikrinimo metodiką. Tą pačią dieną gimusiems asmenims suteikiamas eilės Nr., kuris įrašomas į Gyventojų registrą. Kontrolinis skaitmuo yra skaičiuojamas „modulio“ metodu naudojantis I ir II laipsnio svoriais.

I laipsnio svoris: 1 234567 891

II laipsnio svoris: 3 456789 123

Kiekvienas pirmųjų dešimt asmens kodo skaitmenų dauginamas iš atitinkamo I laipsnio svorio skaitmens, o sandaugos sumuojamos. Gauta suma dalijama iš 11. Jei liekana nelygi 10, tai liekanos reikšmė ir yra kontrolinis skaitmuo. Jei liekana yra lygi 10, kiekvienas pirmųjų asmens kodo skaitmenų yra dauginamas iš II laipsnio svorio skaitmens, o sandaugos sumuojamos. Gauta suma dalijama iš 11. Jeigu liekana nelygi 10, gauta liekana ir yra kontrolinis skaitmuo, o jeigu liekana lygi 10, tai kontrolinis skaitmuo lygus 0. Šį metodą galėtų pailiustruoti toks pavyzdys: skaičiuojamas atsargos karininko, gimusio 1932 m. lapkričio 24 d., kuriam suteiktas eilės Nr. 106, kontrolinis skaitmuo.

	3 3 2 1 1 2 4 1 0 6
<u>I laipsnio svoris</u>	<u>1 2 3 4 5 6 7 8 9 1</u>
Sandaugos:	3 6 4 12 8 6
	6 5 28 0

Sandaugos sumuojamos: $3+ 6+ 6+ 4 + 5 + 12 + 28 + 8 + 0 + 6 = 78$ (1)

Rezultatas dalinamas iš 11: $78:11= 7 + (\text{liekana } 1)$(2)

Tokiu būdu kontrolinis skaitmuo lygus 1, o asmens kodas - 3 321124 106 1.

Kompiuterizuotas asmens kodo tikrinimas pagal „modulio“ metodiką padės išspręsti minėtą problemą.

KA sistema yra hierarchinio pobūdžio, t.y. kariuomenę sudaro pajėgos, pajėgas sudaro brigados, brigadas sudaro batalionai ir t.t. iki pat smulkesnių struktūrinių vienetų - skyrių ir būrių. Tokioje sistemoje kiekvienas poskyris priklauso tam tikrą pavadinimą turinčiam skyriui ir to skyriaus pavadinimas prisideda prie poskyrio pavadinimo ir t.t. Kai organizacinė struktūra yra sudėtinga (keleto lygmenų) dažnai personalo specialistams, rengiantiems tokius norminius dokumentus kaip ministro įsakymus ar ministerijos sekretoriaus potvarkius, reikalinga formuoti pilną padalinio pavadinimą, nurodant visą hierarchinį to padalinio pavaldumą aukštesniems hierarchiniams organizacijos padaliniais. Be to, padalinio pavaldumas kartais keičiasi, todėl konkrečiai padalinio egzistavimo datai gali būti skirtingas to padalinio hierarchinis pavadinimas. Šiuo metu hierarchinio pavadinimo formavimo procedūra atliekama rankiniu būdu. Standartinės Access funkcijos nesiūlo jokių šios problemos sprendimų būdų. Realizuota organizacijos pilno pavadinimo formavimo funkcija padės spręsti šią problemą.

Kai asmuo priimamas dirbti į Krašto apsaugos sistemą, personalo specialistas turi tikrinti etato užimtumą ir etatų kiekį skyrimo į pareigas dienai. Etato užimtumas turi būti tikrinamas ir atleidžiant darbuotoją iš darbo. Tikrinant nustatoma, ar neviršytas etatų kiekis, ar pareigybės etatas neužimtas, ar tokia pareigybė galiojanti. Kadangi personalo apskaitos sistemoje tokia tikrinimo funkcija neįdiegta, personalo specialistas tai turi atlikti rankiniu būdu. Kompiuterizuotas etato užimtumo tikrinimas padės išspręsti minėtą problemą.

1.2. Galimų problemų sprendimų būdų apžvalga

Galimi Dokumento 2.1 skyriuje išvardintų problemų sprendimo būdai:

- 1) modifikuoti Baltpers sistemą, papildant ją reikiamomis funkcijomis;
- 2) sukurti atskirą duomenų analizės ir kontrolės posistemę (toliau — Posistemė), realizuojančią vartotojui reikalingas specializuotas funkcijas;
- 3) panaudoti komercinius (Off-the Shelf) programinius paketus, kuriuose yra personalo valdymo moduliai ir yra realizuotos duomenų kontrolės funkcijos. Kadangi KAS vartotojams reikalingos specifinės duomenų analizės ir kontrolės funkcijos, todėl nė viena pagaminta ar nupirktą sistema nepritaikoma. Tad šito problemos sprendimo būdo daugiau nebeanalizuosime.

1-mo sprendimo būdo privalumai:

- mažesnė projekto realizavimo kaina.

1-mo sprendimo būdo trūkumai:

- Baltpers sistema tampa sudėtingesne;
- papildomos funkcijos reikalingos tik nedideliam skaičiui vartotojų, todėl likusiems vartotojams tai tik apsunkintų darbą su sistema;
- kadangi Baltpers sistema sukurta tarptautinio projekto rėmuose, jos modifikavimą riboja atitinkamos teisinės procedūros.

2-ro sprendimo būdo privalumai:

- paprastesnis darbas su programa, nes patogesnė vartotojo sąsaja;
- programos autonomiškumas, įgalinantis jungtis prie įvairių .mdb formato duomenų bazių;
- Posistemės funkcijų išplečiamumas, nepriklausantis nuo visos Baltpers sistemos modifikavimo;
- patogus programinių objektų modifikavimas.

2-ro sprendimo būdo trūkumai:

- didesnės Posistemės kaip atskiros IS sukūrimo išlaidos;
- reikalingas papildomas Baltpers duomenų pasiekimo (importo/eksporto) mechanizmas.

Įvertinus galimų sprendimo būdų privalumus ir trūkumus nuspręsta pasirinkti antrą problemų sprendimo būdą – sukurti atskirą duomenų analizės ir kontrolės posistemę.

Nagrinėjamo uždavinio aspektų analizė:

- socialiniai aspektai: sukurta Posistemė turėtų pagerinti personalo specialistų darbo sąlygas (mažiau rankinio darbo);
- juridiniai aspektai: KA sistemoje yra priimti visi reikalingi teisės aktai reglamentuojantys personalo administravimą ir apskaitą.
- ryšys su kitais objektais ar uždaviniais: Posistemės kūrimas susijęs su Baltpers sistema.

Posistemėje bus naudojami .mdb formato duomenys importuojami iš išorinio šaltinio — Baltpers DB, prijungiant jos failus prie Posistemės išoriniais ryšiais (saitais). Toks išorinių duomenų šaltinių prijungimo mechanizmas leis organizuoti vartotojų grupinį Baltpers DB naudojimą. Šiuo būdu visi DB duomenų pakeitimai iš karto perduodami visiems prisijungusiems vartotojams. Naudojantis ODBC standartu galima pasiekti ir kitus duomenų bazių serverius [8].

Atlikta analizė leidžia suformuluoti pagrindinį magistro darbo tikslą - sudaryti personalo duomenų analizės ir kontrolės posistemės programinės įrangos projektą (toliau — Projektas) ir sukurti reikalingą programinę įrangą, padėsiančią spręsti Dokumente jau paminėtas problemas.

1.3. IS kūrimo būdų analizė

Kad galėtume kurti programinės įrangos projektą, pirmiausiai turime išanalizuoti pagrindinius IS kūrimo būdus, ištirti jų privalumus, trūkumus bei galimybes. IS projektavimo pagrindas yra teisingas jos modelio atvaizdavimas. Labai svarbus konkretaus modeliavimo metodo parinkimas. Projektavimui gali būti naudojami įvairūs metodai. Egzistuoja ir visą IS gyvavimo ciklą apimantys projektavimo metodai, ir specialios paskirties metodai. Detaliau panagrinėsiu keletą populiariausių IS kūrimo būdų.

1.3.1. Naudojant struktūrinį - funkcinį metodą pagrindinis dėmesys skiriamas IS funkcijoms specifikuoti. Sistema skaidoma į aibę sąveikaujančių modulių, kurių kiekvienas turi aiškiai apibrėžtas funkcijas. Struktūriniame programavime naudojamos CASE - kompiuterizuota programų inžinerija. CASE sistema gali sukurti išbaigtą programų sistemą pagal diagramas ir kitą projektavimui reikalingą informaciją. Tačiau CASE priemonės nekuria PĮ, jos paverčia grafinį sistemos projektą tekstiniu projektu. Išbaigto grafinio projekto sukūrimas reikalauja tiek pat daug laiko ir pastangų, kaip ir programos kūrimas įprastiniu būdu.

1.3.2. Tuo tarpu objektiškai orientuotas IS kūrimo metodas pirmiausia dėmesį skiria probleminės srities objektams identifikuoti, po to sutelkia aplink juos funkcijas [7, 10]. Tai leidžia lengviau pritaikyti PĮ keičiantis vartotojo reikalavimams. Patogaus procesų grafinio atvaizdavimo dėka objektinė technologija paplitusi gana plačiai. Tačiau augant modernių šiuolaikinių IS apimtims bei sudėtingumui objektinis atvaizdavimas tampa vis sudėtingesnis. Objektiškai orientuotas IS kūrimas apima veiklos srities analizę ir modeliavimą, IS projektavimą, objektiškai orientuotų duomenų modelių sudarymą ir objektinį programavimą. Objektinė modeliavimo technologija IS atvaizdavimui naudoja šiuos modelius:

- objektų modelį, aprašantį objektus, klases ir jų ryšius (statinės sistemos savybės);
- funkcinį modelį, aprašantį sistemoje vykstančias duomenų transformacijas (vaizduoja skaičiavimus);
- dinaminį modelį, aprašantį sąveiką tarp objektų klasių (savybės keičiasi laiko atžvilgiu).

Objektų modelis susideda iš klasių diagramų ir objektų - egzempliorių diagramų. Dinaminį modelį sudaro būsenų diagramos (state diagram) ir įvykių sekos diagramos (event trace diagrams). Būsenos diagrama yra grafas, kurio viršūnės yra būsenos, o lankai – įvykių iššaukti perėjimai tarp būsenų. Įvykių diagrama nurodo sistemos veiklos metu atsirandančių įvykių, kurie sieja konkrečių objektų aibę, seką. Funkcinį modelį sudaro duomenų srautų diagramos (angl. DFD) [4]. Duomenų srautų diagrama yra grafas, kurio viršūnės yra procesai, o lankai – duomenų srautai. Šie trys modeliai yra tarpusavyje susieti.

1.3.3. Architektūrinis organizacijos veiklos modelis reikalingas sistemos dalių integralumui užtikrinti. Organizacijos veiklos informacinės architektūros (toliau - VIA) modeliavimas skirtas IS, atitinkančių realius veiklos poreikius, projektavimo ir realizavimo metodams plėtoti. Organizacijos veikla nagrinėjama iš skirtingų pozicijų, išskiriant skirtingo pobūdžio veiklos dalykines sritis, vadinamas veiklos domenais. Pagrindiniai organizacijos veiklos domenai, kurių visuma ir sąveikos užtikrina organizacijos funkcionavimą, yra šie:

- verslo procesų domenas (angl. BD) – ekonominę ir gamybinę veiklą vykdančios organizacijos dalies (valdymo funkcijos, ekonominė veikla) informaciniai poreikiai ir reikalavimai;

- informacijos domenas (angl. DD) – duomenys, žinios ir tikslai, jų saugojimo ir perdavimo organizacijos padaliniais procesais;

- informacinių procesų arba taikomųjų uždavinių domenas (angl. IPD) – organizacijoje atliekami skaičiavimai, sprendimo priėmimo procesai.

- technologinių procesų domenas (toliau – TPD) – organizacijos dalies, atliekančios produkto gamybą (ar formavimą) ir produkto gamybos procesų informaciniai poreikiai ir reikalavimai;

- darbo vietų domenas (angl. DVD) – darbo vietų visuma ir informaciniai reikalavimai darbo vietose.

Veiklos domenai sąveikauja tarpusavyje. Skirtingų domenų tarpusavio informacinės sąveikos užtikrinimas ir pagrindinių domenų sąveikos su išorine aplinka realizavimas reiškia organizacijos veiklos integravimą.

1.3.4. IS komponentinis modelis naudojamas integruotoms organizacijos IS kurti. Komponentinės technologijos panaudojimas kuriant PĮ įgalina panaudoti programinius komponentus kaip nepriklausomus programinius vienetus. Tokia technologija leidžia taikomas programas ar IS sudarinėti iš komponentų, kuriuos gali gaminti skirtingi įrangos gamintojai. Šiuo atveju kiekvienas komponentas su kitais komponentais bendrauja per pagrindinį komponentą, t. y. kiekvieno komponento veikimas priklauso tik nuo pagrindinio komponento ir tiesiogiai nepriklauso nuo kitų komponentų, o pakeitus IS komponentinę sudėtį kinta tik pagrindinis komponentas – kiti komponentai lieka nepakitę. IS projekto lygmens komponentai projektuojami pagal modeliu pagrįstą (angl. model-driven) projektavimo paradigmą, kurioje komponentai paveldi veiklos proceso modelio aprašus. Organizacijos IS komponentams ir sąsajoms tarp jų identifikuoti siūloma nauja grafinė notacija – komponentinis sistemos modelis. Šis modelis apjungia VIA modelio ir darbų sekos modelio savybes. Darbų sekos modelis yra kuriamas per veiklos sąveikų modelį. Šis modelis atvaizduoja organizacijos veiklos procesus ir išreiškia juos veiklos komponentais ir darbų seka tarp tų veiklų.

Šiuo metu pagrindinė ir populiariausia Microsoft komponentinė technologija yra COM, kurios pagrindu buvo sukurtos kitos (pvz. COM+, DCOM, ActiveX) technologijos [7]. Komponentinės technologijos privalumai:

- IS projektavimą ar jo dalį gali atlikti tik minimalias programavimo žinias turintys žmonės;
- tas pats komponentas kaip nepriklausomas modulis gali būti panaudotas daugiau nei viena sykių esamoje IS arba kitose IS.

Tačiau tokio metodo technologija turi ir trūkumų:

- programinio kodo lygmenyje metodas aprašo tik būtinas IS programinių komponentų savybes (atributus bei metodus), o kitas savybes turi sukurti specialiųjų žinių turintis programuotojas;
- jei IS kuriama kita kalba negu C++, Java ar OtMG IDL, specifikuojant komponentus bei sąsajas kodą generuoti reikia rankiniu būdu ar ieškoti įrankio, galinčio dirbti su reikiama kalba.

Išvada. Kiekvienas iš išvardytų metodų analizuoja tik ribotą IS charakteristikų kiekį, todėl konkretus metodas negali užtikrinti, kad visos įmanomos sistemos būsenos ar elgsenos charakteristikos bus įvertintos. Kelių metodų integracija (IS kūrimo informacinės architektūros modelio pagrindu bei komponentinio IS projektavimo apjungimas) yra geriausias sprendimas, tačiau jis pasiteisina tik kuriant integruotas IS. Kadangi kuriama Posistemė nebus sudėtingos struktūros, jai sukurti pasirenkamas paprastesnis objektiškai orientuotas IS projektavimo metodas.

IS gyvavimo ciklo pasirinkimas susijęs su turimos PĮ galimybėmis. Pasirinktas tradicinis “iš viršaus žemyn” (angl. top-down) arba krioklio gyvavimo ciklas [9]. Jame išskiriami analizės, projektavimo ir realizavimo etapai.

PĮ sukurti buvo parengtas programinės įrangos projektas, kuriame aprašyti šie klausimai:

1. Projekto paraiškos rengimas - projekto tikslų ir reikalavimų suderinimas tarp projekto užsakovo ir projekto vykdytojo;
2. Projekto aplinkos analizė ir vartotojų poreikių surinkimas;
3. Reikalavimų Posistemėi specifikavimas – projekto funkcinių ir nefuncinių reikalavimų sudarymas ir pagrindinių apribojimų nustatymas;
4. Posistemės architektūros specifikavimas (eskizinis arba loginis projektavimas);
5. Detalios programinės įrangos architektūros specifikacija (fizinis projektavimas) ;
6. Programinės įrangos realizacija (kodavimas – programinio kodo rašymas);
7. Posistemės testavimas - testavimo eigos aprašymas, testavimo kriterijų, metodų parinkimas, testavimo darbai;
8. Posistemės įdiegimas – programinės įrangos instaliavimas vartotojų darbo kompiuteriuose;

9. Vartotojo dokumentacijos parengimas.

1.4. Programavimo priemonių analizė

Šiame analizės etape numatomos programavimo priemonės, kurios vėliau gali būti tikslinamos. Pasirinkdami programavimo priemones vadovautasi šias pagrindiniais kriterijais:

- suderinamumas su kitomis sistemoje naudojamomis programomis;
- darbo su priemonėmis patogumas;
- kaina;
- mokėjimas naudoti programavimo priemones;
- programinės įrangos charakteristikos (greičio, dydžio ir pan.);
- įsigijimo galimybės ir sąnaudos.

1.4.1. Kuriamos IS aplinkos analizei ir modeliams kurti gali būti naudojamos CASE priemonės. Reikalingas produktas, kuris tiktų nedidelius darbo su kompiuteriu įgūdžius turinčiam vartotojui ir pasižymėtų intuityvia bei „draugiška“ vartotojo sąsaja. Išanalizuota keletas CASE priemonių privalumai ir trūkumai. Pasirinktas Microsoft programinis paketas Visio 2003, nes:

- MS Visio yra Microsoft Office sudėtinė dalis;
- vartotojo sąsaja tokia kaip ir Microsoft Office;
- visos MS Visio figūros aprašomos formulėmis ir todėl galima keisti jų parametrus;
- visi brėžinių tipų paveikslėliai yra vaizdūs, paprastai galima pasirinkti reikiamą tipą;
- galima kurti savo figūras ir šablonus, kuriuose būtų figūrų rinkiniai;
- patogu kurti DB, nes iš nubraižytos DB schemos galima generuoti programos kodą visoms populiariausioms DBVS (Access, FoxPro, Oracle ir kt.);
- sukurtus brėžinius ir diagramas galima „nutempti“ į Word dokumentą, Excel lentelę ar PowerPoint programą;
- programinis produktas plačiai paplitęs;
- programinis produktas nėra brangus ir yra įdiegtas projekto vykdytojo kompiuteryje.

Kaip Visio trūkumą galima paminėti programos reiklumą kompiuterio resursams.

IS modeliuojant su MS Visio galima apimti pagrindinius IS gyvavimo ciklo etapus (veiklos procesų modeliavimą, veiklos objektų modeliavimą, IS modeliavimą).

1.4.2. Projektui atlikti gali būti panaudotos įvairios programavimo kalbos, palaikančios tekstinių masyvų apdorojimą. Dideliems duomenų rinkiniams tvarkyti ir ataskaitoms rengti naudojamos tokios priemonės kaip Access, Oracle ir pan. Baltpers sistemoje naudojama duomenų bazių valdymo sistema (toliau — DBVS) Microsoft Access, todėl svarbus suderinamumas su MS Access programa. Pasirinkta patogi ir plačiai naudojama projektavimo – programavimo terpė

Microsoft Access [8, 11]. Microsoft Access (toliau - MS Access) yra viena populiariausių vidutinės apimties reliacinė DBVS. Joje duomenys pateikiami kaip dvimačių lentelių rinkinys. MS Access priemonėmis sukurta reliacinė DB išsaugoma faile, kuriam suteikiamas plėtinys mdb. Šiame faile saugomos ne tik normalizuotos dvimatės lentelės su duomenimis, bet ir kiti objektai. Su MS Access taip pat galima kurti tiek autonomines, tiek tinklines taikomas programas, tiek programas, skirtas dirbti sistemose „Klientas – serveris“. MS Access turi daug pagalbinių funkcijų: užtikrina duomenų korektiškumo ir vientisumo kontrolę, įgalina atlikti įvairius matematinius, statistinius skaičiavimus, leidžia įslaptinti tam tikrus duomenis ir uždrausti nesankcionuotą kreiptį į juos. Vartotojas, naudodamas tokias Access priemones kaip makrokomandas, SQL ar Visual Basic kalbą gali kurti savąją meniu sistemą ir taikomas programas, skirtas specifiniams poreikiams tenkinti. Minėtoms funkcijoms atlikti MS Access turi specialius instrumentus – grafinius dizainerius, vedlius (meistrus), išraiškų konstruktorius, programų generatorius ir kt. Access turi makrokomandas (angl. Macros), skirtas veiksmų sekoms aprašyti. Tad specializuota taikomoji programa bus kuriama Access aplinkoje. Programos kūrimas kaip Windows programos teikia šiuos privalumus:

- greitas programos veikimas;
- turtinga vartotojo sąsaja;
- nebūtinai nuolatinis prisijungimas prie duomenų tinklo (duomenys gali būti sinchronizuojami su pagrindine DB tik tam tikrais momentais).

Tačiau Windows programų kūrimas turi ir trūkumų:

- PĮ reikia įdiegti kiekviename vartotojo kompiuteryje ir dėl to pasidaro sudėtingesnis diegimas;
- modifikuojant programą reikia atnaujinti ją visuose kompiuteriuose ir dėl to sudėtingėja programinio produkto palaikymas.

Minėtų problemų sprendimas – programos patalpinimas web serveryje. Tada vartotojas turi jungtis prie serverio ir paleisti programą. Paleista programa patikrina ar jau yra tokia programa įdiegta vartotojo kompiuteryje, o jei nėra - įsidiegtų ir pradėtų veikti. Programą kiekvieną kartą paleidžiant ji pasitikrintų, ar nėra atnaujinimo. Atnaujinimai turi įsidiegti automatiškai. Tokį variantą numatoma realizuoti per kitus Projekto vystymo etapus, kai bus gautas užsakovo pritarimas.

Vartotojui reikalingas funkcijas galima programuoti naudojant Visual Basic programavimo kalbą [6], kadangi ją galima instaliuoti į Microsoft Access aplinką. Tuo tarpu MS Access aplinkoje jau yra integruota Visual Basic for Applications (toliau — VBA) programa. VBA yra programavimo aplinka, kurią galima išsikviesti iš jau egzistuojančios taikomosios programos (aplikacijos) ir su ja kurti programinius komponentus veikiančius egzistuojančioje aplikacijoje [5].

Tad iš turimų alternatyvų projekto realizacijai pasirinkta VBA programavimo kalba. Kiti VBA pasirinkimo motyvai:

- ši kalba yra Microsoft produktas, suderinamas su Microsoft Windows aplinka;
- tai objektiškai orientuota, populiarė programavimo kalba, kurioje yra visos reikalingos tekstinių duomenų apdorojimo funkcijos;
- kalba pasižymi paprastumu, programuojant lengvai prijungiamos Windows operacinės sistemos funkcijos, su jos pagalba galima modifikuoti MS Office paketo programas;
- VBA yra lanksti, turi daug bibliotekų, didelę elementų ir plačią palaikymo bazę (archyvus, vartotojų konsultavimą, forumus, pagalbą) turinti programavimo kalba;
- su VBA galima parengti programos pultą Access duomenų bazei (toliau – DB) tvarkyti, t.y. įvesti duomenis, atlikti automatinę duomenų paiešką, pašalinti nereikalingus įrašus, išvesti duomenis iš DB į VB programą;
- programinius produktus sukurtus su VBA nesudėtinga modifikuoti;
- programos derinimo galimybė;
- sistemos stabilumas išlaikomas pasitaikius programavimo klaidoms;
- su šia programavimo kalba Informacinių technologijų magistrantūros studentai supažindinami studijų metu ir Posistemės kūrėjas turi darbo šioje aplinkoje patyrimą, todėl jam nereikės iš naujo prisitaikyti prie kokios nors naujos programavimo aplinkos;
- ši programavimo priemonė nekainuoja, ji jau yra integruota į Microsoft Office paketus (pvz., Microsoft Office XP pakete integruota VBA 6.3).

Su VBA bus parašyti moduliai (angl. Modules) – rezultatai gražinančios funkcijos [12]. Modulis yra failas, turintis kodą priklausantį formai ar ataskaitai. Modulyje gali būti laikomas nepriklausomas kodas.

Sukurta PĮ gali būti pateikta kompaktinėje plokštelėje (toliau – CD) diske kartu su reikiama dokumentacija arba pateikta internete. Gali būti sukurta internetinė svetainė ir joje pateikta išsami Projekto dokumentacija.

2. PROGRAMINĖS ĮRANGOS PROJEKTAS

Projekto tikslas — panaudojant Access aplinką ir VBA programavimo priemones sukurti nesunkiai įsisavinamą, nebrangią programinę įrangą, leidžiančią vartotojui atlikti Baltpers sistemoje esančių personalo duomenų analizę ir patikrinimą.

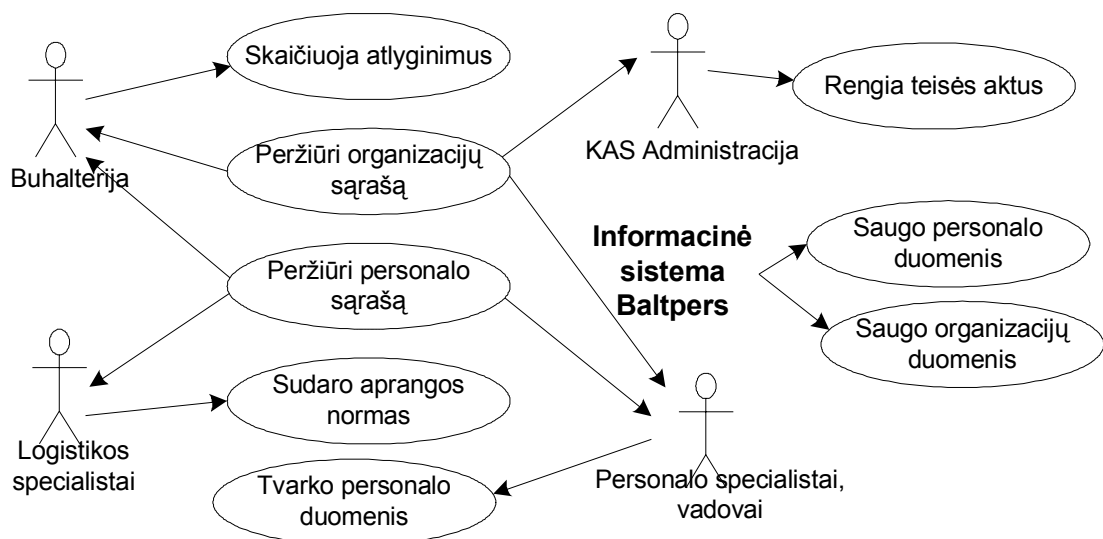
Pagrindiniai Projekto uždaviniai yra šie:

- suformuluoti reikalavimus kuriamai Posistemei;
- apibrėžti naudojamų duomenų struktūrą;
- apibrėžti projektuojamos programų sistemos architektūrą;
- specifikuoti programinius modulius ir objektus;
- parengti išsamią vartotojo dokumentaciją.

2.1. Projekto vykdymo sritis

2.1.1. Organizacijos veiklos srities aprašymas

Veiklos srities informacinių poreikių aprašymui naudojamas grafinis modelis – Use Case Model (toliau – UCM) [7]. KAS padalinių veiklą personalo srityje aprašanti UCM pateikta 1 pav.

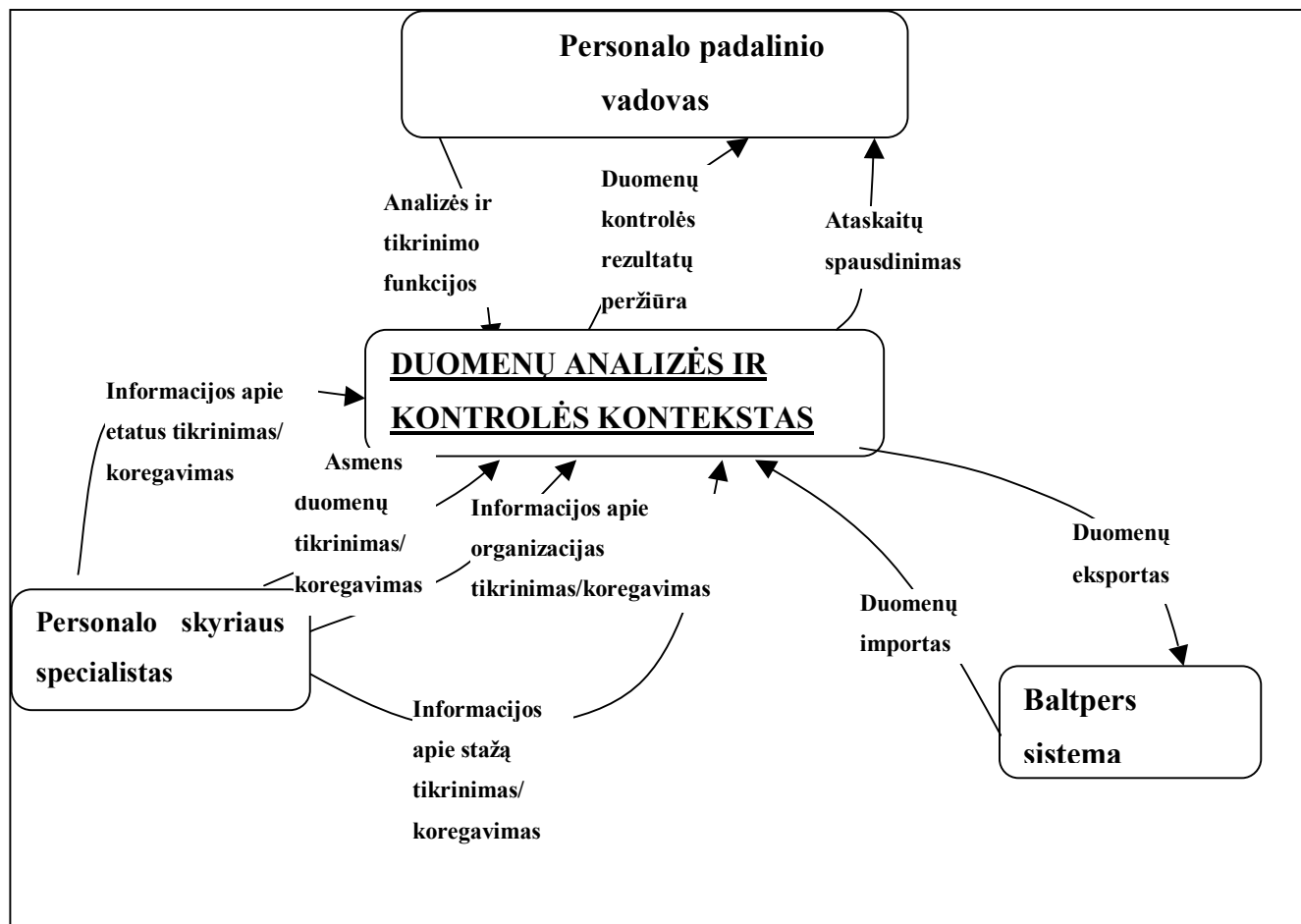


1 pav. Organizacijos veiklos personalo srityje panaudojimo atvejų (Use Case) diagrama

Kadangi šio darbo tema susijusi tik su personalo specialistų ir vadovų specializuotų poreikių realizavimu, tolesnis Projektas iš esmės apims kompiuterizuojamą veiklos sritį – personalo ir organizacijų duomenų analizę ir kontrolę. Veiklos produktas – patikrintų personalo duomenų ataskaitos.

2.1.2. Posistemės veiklos kontekstas

Siekiant sukurti programų sistemą reikia iširti personalo specialistų ir vadovų veiklos sritį. Veiklos konteksto diagrama apibrėžia dominančią veiklą ir jos naudojamus bei formuojamus informacijos srautus.



2 pav. Veiklos konteksto diagrama

2.1.3. Veiklos padalinimas

Veiklos padalinimo paskirtis - identifikuoti veiklos "gabaliukus" ir informacijos srautus, kurių pagrindu būtų galima nustatyti vartotojų reikalavimus. Veiklos įvykių pagrindu remiamasi atliekant sistemos detalią analizę ir projektavimą.

Veiklos įvykių sąrašas

1 lentelė

įil. √r.	Įvykio pavadinimas	Įeinantys(in)/ išeinantys (out) informacijos srautai
1	Į Posistemę importuojami personalo ir KAS organizacijų duomenys iš sistemos Baltpers	Personalo duomenys (in)

2	Vadovas nurodo tikrinimo funkcijas	Tikrinimo funkcijos (in)
3	Vadovas peržiūri duomenų analizės ir kontrolės rezultatus	Rezultatai (out)
4	Specialistas tikrina/koreguoja informaciją apie organizacijas	Organizacijos (in)
5	Personalo specialistas tikrina informaciją apie etatus	Etatai (in)
6	Personalo specialistas tikrina/koreguoja asmens duomenis	Asmens duomenys (in)
7	Specialistas tikrina/koreguoja informaciją apie tarnybos trukmę	Tarnybos trukmė (in)
8	Vadovas peržiūri duomenų kontrolės (skaičiavimo) rezultatus	Rezultatai (out)
9	Personalo vadovas peržiūri duomenų kontrolės rezultatus	Ataskaita (out)
10	Posistemė eksportuoja patikrintus duomenis į Baltpers(į failą)	Patikrinti duomenys (out)

2.1.4. Posistemės vartotojai

Pagrindiniai Posistemės vartotojai bus personalo specialistas ir personalo padalinio vadovas.

Vartotojas - personalo specialistas

2 lentelė

<i>Vartotojo kategorija</i>	<i>Personalo skyriaus specialistas</i>
Vartotojo sprendžiami uždaviniai:	- duomenų apie asmenis ir organizacijas kontrolė; - informacijos apie asmenis, organizacijas, etatus ir tarnybos trukmę koregavimas; - duomenų patikros rezultatų peržiūra.
Patirtis dalykinėje srityje:	Įprastas darbuotojas, srities specialistas
Patirtis informacinėse technologijose:	Naujokas, patyręs
Papildomos vartotojo charakteristikos:	Vartotojas turi mokėti dirbti su personalo IS Baltpers, būti susipažinęs su tarnybą KAS, personalo apskaitą reglamentuojančiais teisės aktais

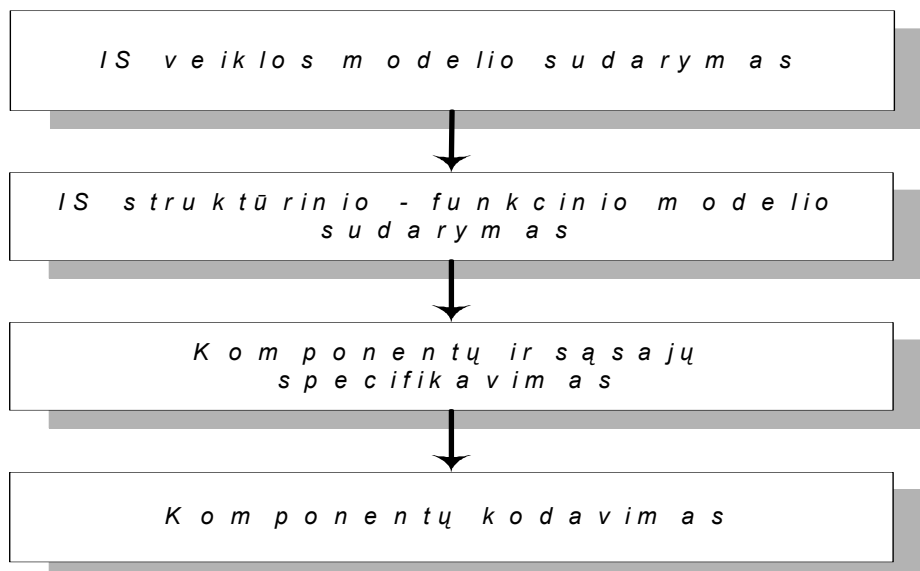
Vartotojas – personalo padalinio vadovas

3 lentelė.

<i>Vartotojo kategorija</i>	<i>Padalinio vadovas (Personalo departamento direktorius, Personalo skyriaus vedėjas)</i>
Vartotojo sprendžiami uždaviniai:	- duomenų koregavimo tvirtinimas; - duomenų kontrolės ataskaitų peržiūra ir spausdinimas; - gautų rezultatų analizė. - ataskaitų spausdinimas popieriuje.
Patirtis dalykinėje srityje:	Patyręs, srities specialistas
Patirtis informacinėse technologijose:	Naujokas, patyręs
Papildomos vartotojo charakteristikos:	Vartotojas turi mokėti dirbti su personalo IS Baltpers, gerai žinoti tarnybą KA sistemoje, KAS personalo administravimą reglamentuojančius teisės aktus

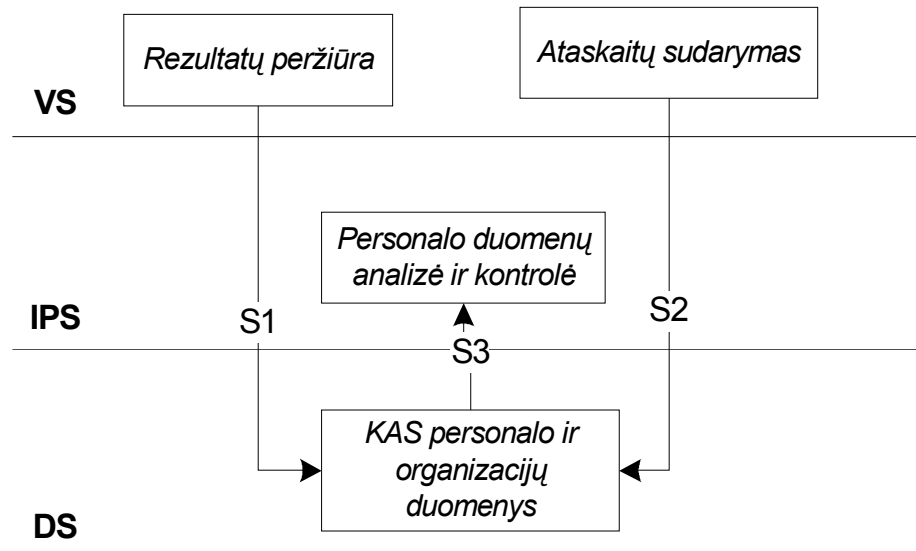
2.1.5. Veiklos informacinės architektūros modelis

Pagrindiniai Posistemės kūrimo etapai pavaizduoti 3 pav.



3 pav. Posistemės kūrimo technologinis procesas

Posistemės modeliavimo etape sprendžiama kas turi būti padaryta, o projektavimo etape – kaip tai turi būti padaryta. Ir struktūrinėje, ir objektinėje IS kūrimo technologijoje sudaromas kompiuterizuojamos organizacijos veiklos modelis. Veiklos modelio paskirtis yra aprašyti duomenų transformacijas sistemoje. Pagal Posistemės veiklos uždavinių modelį sudaromas veiklos informacinės architektūros modelis. Posistemės kiekvieną veiklos modelio elementą priskiriame vienam iš keturių VIA modelio sričių (4 pav.). Mūsų atveju technologinių procesų sritis yra tuščia (gamybinių procesų nėra). S1, S2 ir S3 - sąsajos tarp VIA srities komponentų. Pagal sudarytą VIA jau bus galima specifiuoti programos komponentus. Naudojami žymėjimai: VS – veiklos sritis, IPS – informacinių procesų sritis, DS – duomenų sritis.



4 pav. Duomenų analizės ir kontrolės proceso VIA modelis

2.2. Reikalavimai projektuojamai Posistemei

2.2.1. Reikalavimų gavimo būdai

Reikalavimai gaunami vykdamas:

- vartotojų apklausą ir tuo būdu nustatant funkcinis reikalavimus;
- analizuojant panašaus pobūdžio PĮ privalumus ir trūkumus;
- nustatant nefunkcinius reikalavimus.

2.2.2. Projekto apribojimai

Apribojimai technologinei ir fizinei diegimo aplinkai:

- Posistemė turi veikti Windows šeimos operacinėse sistemose;
- Krašto apsaugos sistema turi savo kompiuterinį tinklą (LAN-a);
- reikalingas bet kokios markės personalinis kompiuteris.

Posistemės PĮ turi veikti šias charakteristikas turinčiuose kompiuteriuose:

- operacinė sistema: MS Windows 9x/2000/ ME, NT, XP;
- Intel Pentium 166MHz procesorius;
- darbinė atmintis (RAM): \geq 128 MB;
- ne mažiau 5 MB laisvos vietos (erdvės) kietajame diske.

Vidinė DB sukurta SQL kalba pagal reliacinių DB standartus. Baltpers duomenų pasiekimas galimas per ODBC sąsają.

Bendradarbiaujančios sistemos. Sistema Baltpers ir Posistemė bendradarbiauja su kitomis KAS IS, t.y. su PPBS VIS ir LogVis. Minėtos sistemos sukurtos išimtinai Krašto apsaugos sistemos reikmėms. Tarp minėtų komunikuojančių IS sprendžiami integravimo ir apsikeitimo duomenimis uždaviniai.

Komerciniai specializuoti programų paketai. Komercinių programinių paketų nebus taikoma.

Apribojimai darbo vietos aplinkai. Identifikuojame šias fizines Posistemės kompiuterizuotos darbo vietos charakteristikas:

- darbo vieta yra stacionari;
- darbo vietoje turi būti spausdintuvas (A4 formato lazerinis spausdintuvas lokalus arba tinklinis);
- kadangi Baltpers naudojamų asmens duomenų saugumą reglamentuoja Asmens duomenų apsaugos įstatymas, tai Posistemėje turi būti įdiegtas autorizuotas vartotojų prisijungimas.

Apribojimai darbo su Posisteme scenarijui:

- darbo su Posisteme pradžioje vartotojas turi prisijungti prie išorinio duomenų šaltinio – Baltpers (personalo duomenų) bazės lentelių ir prisijungus dirbama su naujausiais personalo duomenimis;
- modifikuoti duomenys ir skaičiavimo rezultatai realaus laiko režime atnaujinami ir Baltpers duomenų bazėje;
- pasinaudojant Access priemonėmis rezultatai gali būti eksportuojami į pasirinkto formato failą.

2.2.3. Vartotojo reikalavimai

Atlikus vartotojų apklausą nustatyta, kad Posistemės vartotojas nori programos, leidžiančios jam:

- pasirinkti duomenų analizės ar tikrinimo funkciją;
- tikrinti personalo duomenis (atlikti skaičiavimus);
- tikrinti organizacinių struktūrų duomenis ir formuoti pilną organizacijos pavadinimą;
- skaičiuoti personalo tarnybos trukmę (stažą);
- tikrinti etatų užimtumą;
- leisti įvesti ir koreguoti duomenis;
- peržiūrėti duomenų kontrolės rezultatus ekrane;
- leisti koreguoti personalo duomenis;
- generuoti ir leisti atsispausdinti vartotojo užklausas tenkinančias ataskaitas;
- išsaugoti atliktų personalo ir organizacinių struktūrų duomenų pakeitimus;
- importuoti/eksportuoti duomenis iš/į Baltpers.

Remiantis vartotojo poreikių analize formuluojami funkciniai ir nefunkciniai reikalavimai projektuojamai Posistemai.

2.2.4. Prototipai

PĮ projektavimo procese būna svarbus prototipų vaidmuo. Kad susidarytų bendras programos vaizdas prototipą tikslinga kurti pradinėje projektavimo stadijoje. Tolesnėse projekto stadijose prototipo panaudojimas tampa nereikalingas. Kadangi mūsų atveju programinis produktas yra palyginus nedidelis, todėl prototipo kūrimas nebus atsiperkantis ir prototipas nebus kuriamas.

2.3. Posistemės specifikacija

Posistemės specifikavimo tikslas – apibūdinti kuriamą Posistemę bei nubrėžti jos realizavimo gaires. Sudarant specifikacijas nustatomi ir fiksuojami:

- pradiniai programos duomenys,
- naudojamų duomenų struktūros ir vaizdavimo būdai,
- gaunami jų darbo rezultatai, jų tarpusavio ryšys ir apdorojimo reikalavimai,
- sistemos ribojimai (reakcijos laikas, vykdymo trukmė, tikslumas, klaidų apdorojimo būdai);
- įvertinamas galimas ryšys su kita programine įranga.

Kuriama Posistemė nepriklauso kritinių – laiko sistemų klasei, todėl formalios specifikacijos nebus naudojamos.

2.3.1. Programinės įrangos architektūra

Dėl spartaus technologijų vystymosi ir didesnių vartotojų reikalavimų esminiu dalyku tampa programų sistemos architektūra (PĮ komponentės, ryšiai tarp komponentių ir duomenų struktūrų). Architektūra turi būti tiksliai apibrėžta, kad atradus sistemos klaidą bet koks programuotojas sugebėtų ją greitai ištaisyti net jei programos autorius klaidos taisyme nedalyvautų.

Architektūros apribojimai:

- kuriama nekomercinė PĮ ir ji bus pateikta kaip atviro kodo programinė įranga;
- PĮ bus kuriama VBA kalba;
- Posistemė turės imti duomenis iš .mdb formato failų ir išsaugoti duomenis į .mdb formatą;
- architektūra bus parenkama taip, kad ją galima būtų lengva išplėsti ar prijungti naujus komponentus;

Duomenų importui ir eksportui gali būti reikalingi papildomi architektūriniai sprendimai.

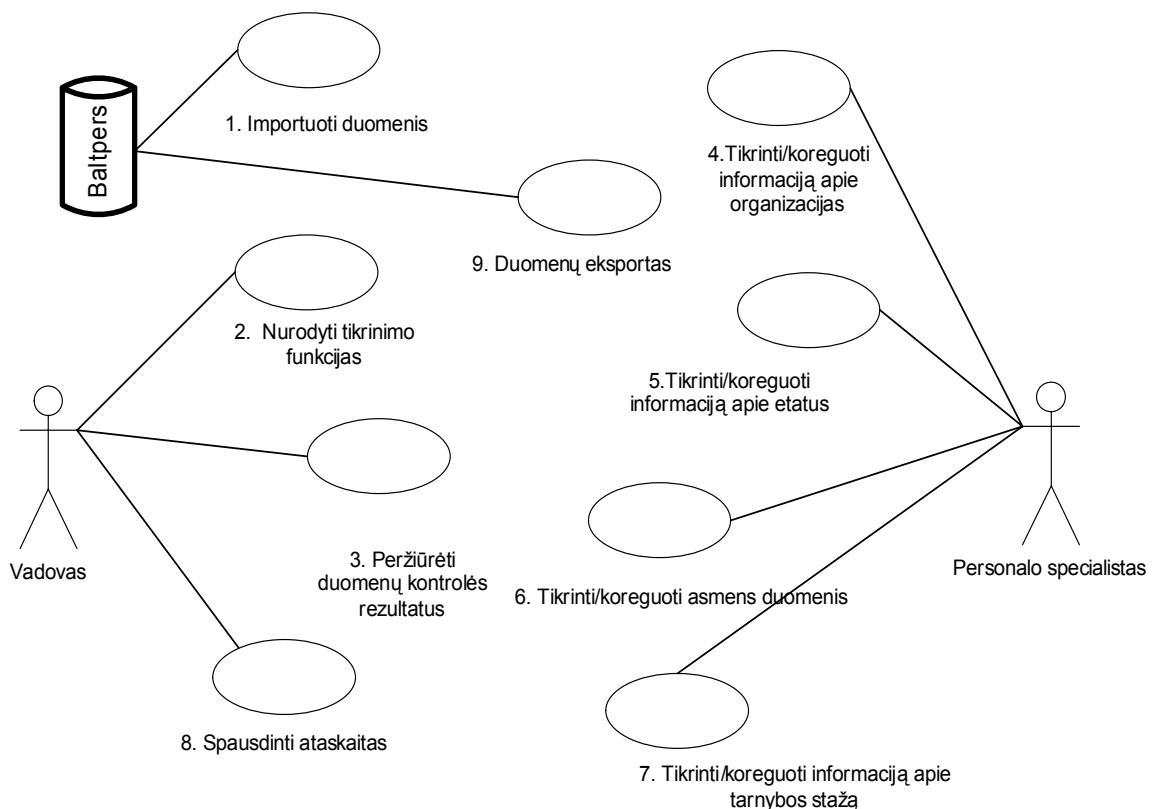
PĮ architektūra modeliuojama keliais aspektais (tarpusavyje susijusiais, bet skirtingais požiūriais):

- panaudojimo atvejų vaizdu (panaudojimo atvejų UML diagrama);

- loginiu programos vaizdu (klasių diagramos, klasių apjungimas į paketus);
- procesų vaizdu (būsenų kaitų diagramos, sekų diagramos, bendradarbiavimo diagramos);
- išdėstymo vaizdu (išdėstymo diagrama);
- realizavimo vaizdu.

2.3.1.1. Panaudojimo atvejų vaizdas

UCM arba panaudojimo atveju diagrama apibrėžia ir analizuoja projektuojamos Posistemės ribas, svarbiausius veiklos sferas, veiklos procesus ar veiklos sąveikas tiek organizacijos viduje, tiek ir išorėje.



5 pav. Posistemės veiklos “Duomenų analizė ir kontrolė” uždavinių modelis

Ši diagrama apibūdina ir pagrindines IS savybes – personalo duomenų importą, personalo duomenų tikrinimą, analizę ir koregavimą, ataskaitų sudarymą, peržiūrą ir spausdinimą, DB atnaujinimą koreguotais duomenimis, duomenų eksportą.

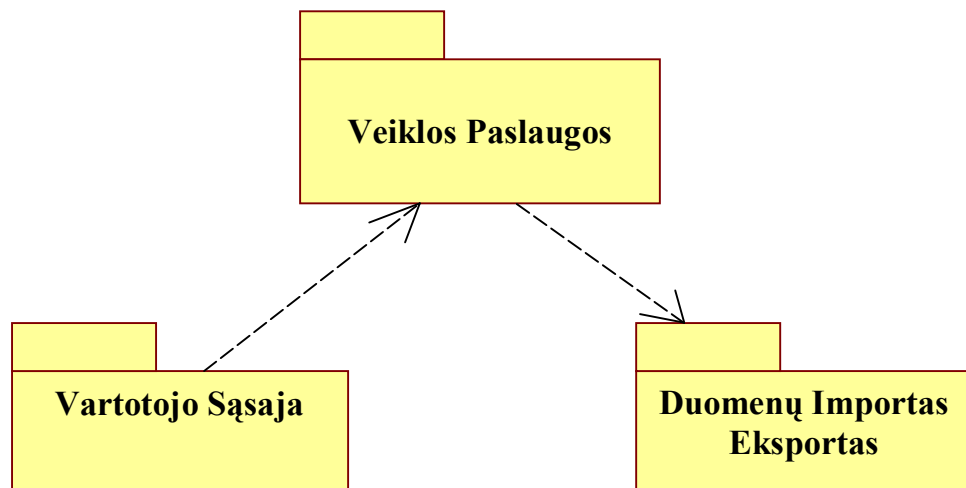
2.3.1.2. Programos loginis vaizdas

Architektūros aprašymas (modeliavimas) pradedamas nuo PĮ architektūros loginio (statinio) vaizdo, kuris parodo objektus ir jų ryšius kuriuo nors konkrečiu sistemos funkcionavimo momentu.

Atsižvelgiant į vartotojo reikalavimus sudarome Posistemės loginį modelį (6 pav.), kuriame išskiriame šiuos pagrindinius programos paketus, atitinkančius tam tikrą veiklos tipą:

- Vartotojo Sąsaja paketą (vartotojo sąsajos projektavimas);
- Veiklos Paslaugos paketą (Posistemės funkcionalumo projektavimas);
- Duomenų Importas Eksportas paketą (duomenų struktūrų projektavimas).

Paketas yra klasių grupė. Pagal UML teoriją paketas yra logiškai susietų UML elementų rinkinys. Paketai yra pagrindiniai sistemos komponentai. Tad Posistemės kūrimo projektas iš esmės apims išvardintų komponentų projektavimą. Paketas Veiklos Paslaugos yra svarbiausias paketas.



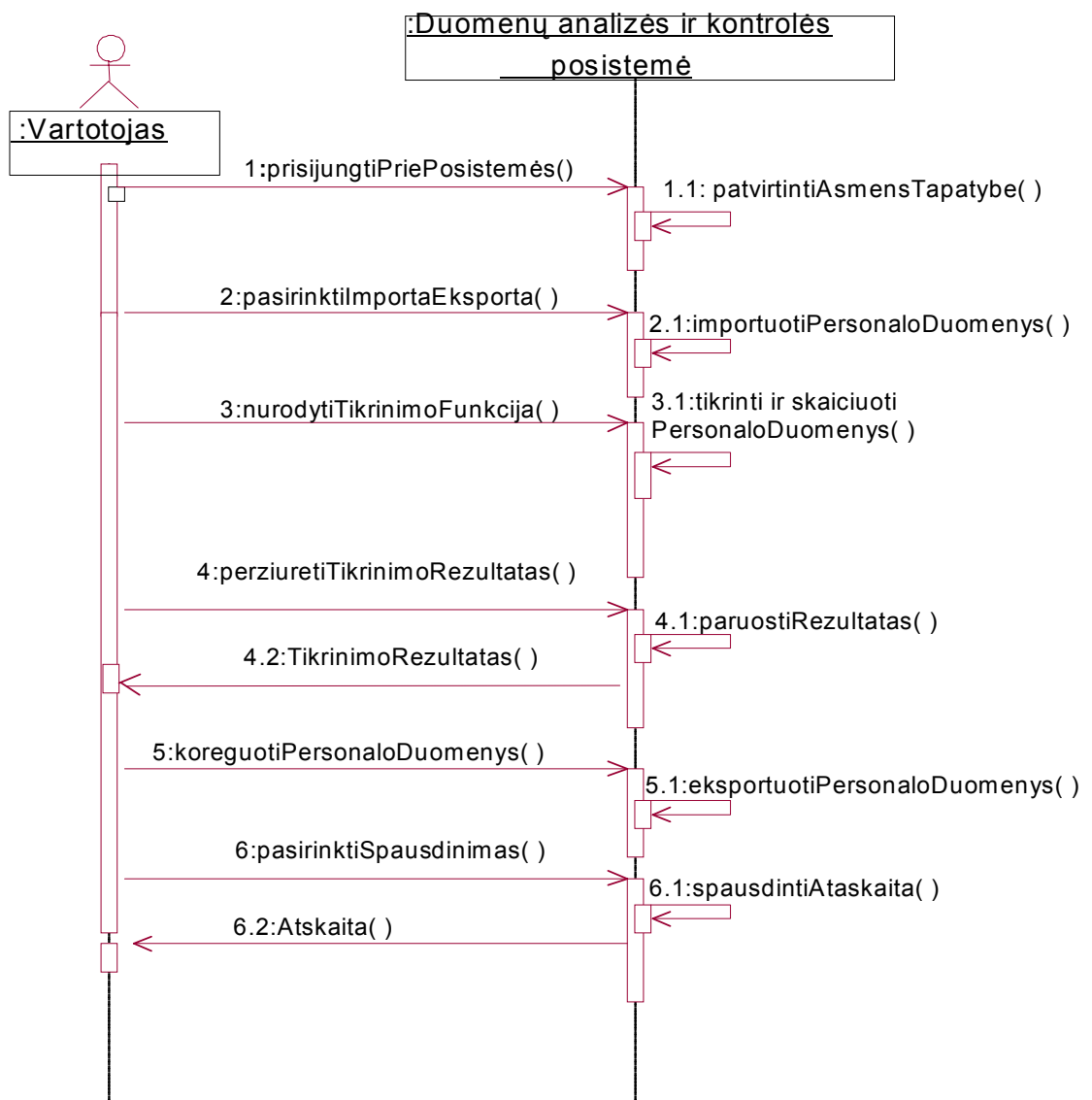
6 pav. Posistemės išskaidymas į paketus aukščiausiame lygyje

Veiklos Paslaugos paketą sudaro valdančios klasės, realizuojančios visą Posistemio funkcionalumą, t.y. personalo vadovo ir personalo skyriaus specialisto atliekamas funkcijas. Architektūriniu požiūriu Veiklos Paslaugos paketą tikslinga išskaidyti į šiuos pagrindinius subpaketus (subkomponentus): Saugumas, Vartotojo Paslaugos ir Funkcijos. Paketų detalizavimas į juos sudarančias klases ir klasių realizacija pateikta Dokumento 2.3.5. poskyryje.

2.3.1.3. Procesų vaizdas

IS procesų (dinaminis) vaizdas analizuoja sistemos objektų sąveiką laike. Jis modeliuojamas sąveikos (interaction), būsenų (state) ir veiklos (activity) diagramomis.

Būsenų diagramos naudojamos parodyti vieno objekto veikimą daugelyje panaudos atvejų. Sąveikos diagramos vaizduoja objektų tarpusavio sąveiką. Sekų (sequence) diagramos naudojamos parodyti viename panaudos atvejyje dalyvaujančius objektus ir rodančios pranešimų (metodų kvietimų) chronologinę tvarką (seką), o bendradarbiavimo (collaboration) diagramos naudojamos analizuoti atskirų objektų vaidmenį duotame panaudos atvejyje. Veiklos diagramos naudojamos parodyti kelių sąveikaujančių objektų veikimą. Projektuojamos Posistemės procesų sąveikai iliustruoti pasirenkama viena iš diagramų – vartotojo bendradarbiavimo su Posisteme sekų diagrama (7 pav.).

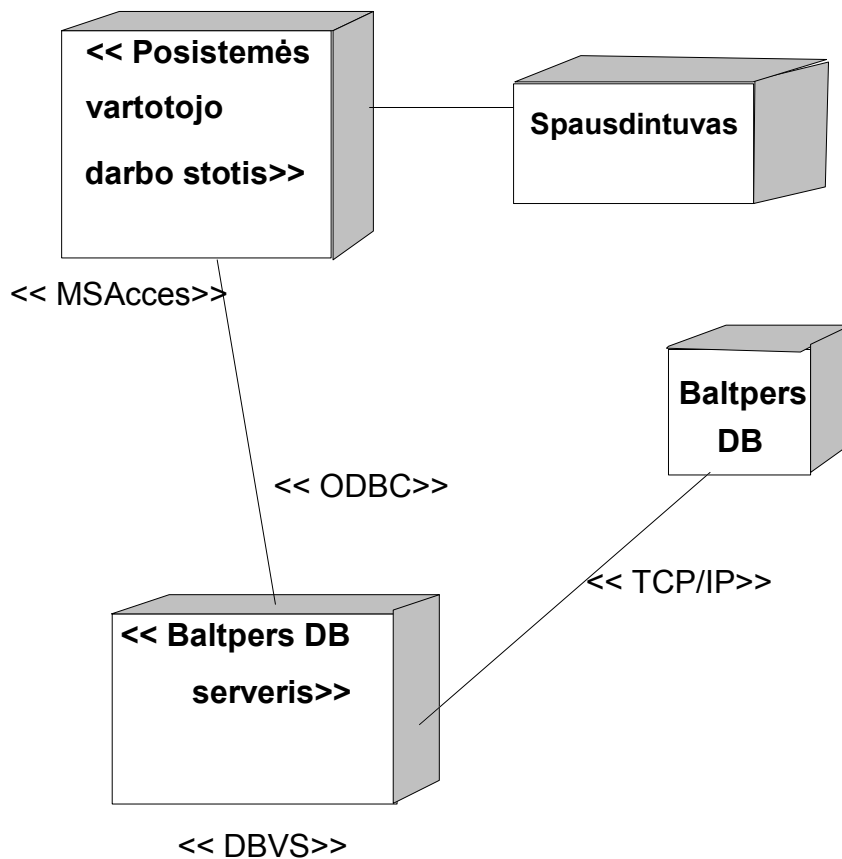


7 pav. Vartotojo bendradarbiavimo su Posisteme sekų diagrama

2.3.1.4. Išdėstymo vaizdas

Išdėstymo (angl. deployment) vaizdas parodo techninės įrangos, kurioje veiks programinė įranga, konfigūraciją, komponentus ir ryšius tarp PI (angl. software) ir techninės įrangos (angl.

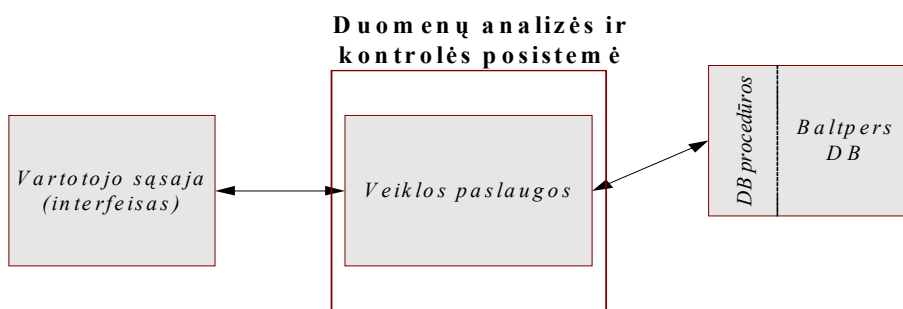
hardware).



8 pav. Fizinē Posistemēs iždēstymo diagrama

2.3.1.5. Realizācijas vaizdas

Posistemē bus realizuota daugiasluoksnēs (triju slukšņu) arhitektūros pagrindu.



9 pav. Posistemės realizacijos modelis

Duomenų bazės sluoksnyje saugomi Posistemėi reikalingi duomenys. Naudojama reliacinė DBVS. Duomenų paėmimui iš DB, modifikavimui bei naujų duomenų įterpimui į DB naudojamos Access procedūros. Bendravimas su DB bus vykdomas vien tik per importo ir eksporto procedūras.

Veiklos paslaugų sluoksnyje vykdomos taikomosios srities taisyklės (Posistemės kompiuterizuota veikla). Reikalingi duomenys paimami iš DB naudojantis procedūromis, apdorojami remiantis taikomosios srities taisyklėmis, o gauti rezultatai perduodami vartotojo sąsajos sluoksniui. Skaičiavimo rezultatai procedūrų pagalba patalpinami į Baltpers DB (arba išsaugomi atskirame faile). Kadangi tie patys duomenų nešėjai yra naudojami skirtinguose veiklos procesuose, todėl tikslingas šio architektūros sluoksnio išskyrimas į atskirus posluoksnius: Vartotojo Paslaugos, Saugumas, Funkcijos.

Vartotojo sąsajos sluoksnis skirtas rezultatams apipavidalinti ir juos pateikti vartotojui.

2.3.2. Posistemės reikalavimų specifikacija

Įvertinus vartotojų reikalavimus, uždavinio analizės rezultatus ir PĮ architektūros modelius, apibrėžiami funkciniai ir nefunkciniai reikalavimai kuriamai Posistemėi.

2.3.2.1. Funkcinių reikalavimų specifikavimas

Funkciniai reikalavimai – sistemos paslaugų aprašymai. Turi būti realizuotos šios funkcijos:

- duomenų gavimas (importas) iš Baltpers DB;
- duomenų siuntimas (eksportas) į Baltpers DB;
- karinio personalo asmens kodo patikrinimas (tikrinti, ar įvestas į sistemą kodas atitinka to asmens dokumentuose (pase, tapatybės kortelėje) įrašytam asmens kodui);
- organizacijos pilno pavadinimo formavimas;
- tarnybos stažo apskaičiavimas;
- etatų užimtumo (laisvo/užimto etato) tikrinimas;
- duomenų koregavimas;
- ataskaitų formavimas ir spausdinimas.

2.3.2.2. Nefuncinių reikalavimų specifikavimas

Nefunkciniai reikalavimai – paslaugų ar funkcijų apribojimai.

Reikalavimų Posistemės išvaizdai specifikavimas – vartotojui turi būti suteikta paprasta, lengvai skaitoma sąsaja.

Reikalavimų Posistemės panaudojamumui specifikavimas:

- lengvai suprantama pagalba (Help-as), paaiškinanti kaip naudotis programiniu produktu;
- paprastas panaudojimas, nereikalaujantis specialaus personalo specialistų apmokymo.

Patikimumo reikalavimų specifikavimas:

- patikimumo matavimas – IS funkcionalumo pradžios trukmė neturi viršyti penkių minučių per vieną valandą;
- klaidų apdorojimas – klaidos atveju Posistemė turi išduoti klaidos pranešimą;
- Posistemės veikimo atkūrimas – IS turi būti sukurta iš atskirų modulių ir įvykus klaidai viename iš modulių, kiti moduliai neturi prarasti funkcionalumo;
- iškilusios išimtinės situacijos turi būti tinkamai apdorotos programos viduje, apie iškilusias problemas informuojant vartotoją;
- programinis produktas turi patikimai veikti Microsoft šeimos operacinėse sistemose.

Reikalavimų veikimo sąlygoms specifikavimas – Posistemės integruotumas.

Reikalavimų Posistemės priežiūrai specifikavimas - keičiantis personalo administravimą reglamentuojantiems teisės aktams, turi būti galimybė adaptuoti Posistemę prie naujų poreikių.

Reikalavimų duomenų saugumui specifikavimas:

- duomenų konfidencialumas ir vientisumas;
- neturi būti galimybės neautorizuotiems naudotojams prisijungti prie Posistemės;
- galimybė audituoti su Posisteme atliktas operacijas (funkcijas);
- vartotojas turi naudoti prisijungimo slaptažodžius;
- kompiuteris turėtų prašyti vartotojo iš naujo prisijungti po tam tikro nustatyto nesinaudojimo laiko (pavyzdžiui 5 minučių).

Reikalavimų PĮ saugumui specifikavimas: PĮ produktas turi nekelti grėsmės kitai PĮ, duomenims, aparatinei įrangai, vartotojo sveikatai, netrukdyti kitų sistemų darbui, nesukelti MS Office „lūžių“.

Kultūriniai-politiniai reikalavimai - vartotojo sąsaja turi būti realizuota lietuvių kalba.

Teisiniai reikalavimai:

- kuriant programinį produktą turi būti vadovaujama LR asmens duomenų apsaugos įstatymu;
- kuriamas programinis produktas turi būti teisėtas;
- produkto modifikavimas, platinimas, kopijavimas turi būti suderintas su programos autoriumi.

Reikalavimų našumui specifikavimas:

- skaičiavimo operacijų laikas turi būti ribotas, o jei per tam tikrą laiko operacijos neatliekamos, vartotojas apie turi gauti pranešimą.

- operatyvinės atminties naudojimas serveryje neturi viršyti 32 MB operatyvinės atminties;
Reikalavimų kokybei specifikuojimas – produkto kokybę turi užtikrinti programinės kūrimo priemonės.

Tikslumo reikalavimų specifikuojimas – leistinų paklaidų nėra.

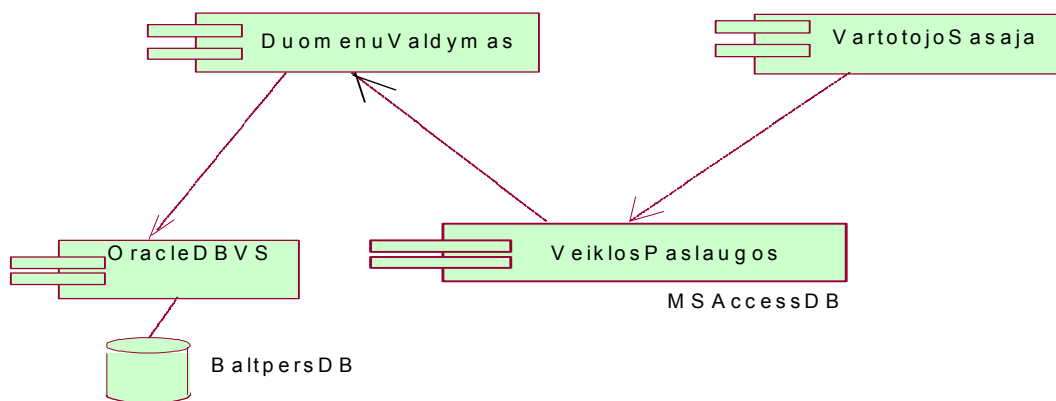
Kitų nefunkcinių reikalavimų specifikuojimas:

- IS palaikymas – numatyti IT ar IS specialisto konsultacijas dėl iškilusių neaiškumų;
- taikomųjų programų suderinamumas – reikalingas suderinimas su Microsoft ir kita PI;
- aptarnavimas ir priežiūra – administravimui naudoti standartines DB valdymo priemones;
- modernizavimas – IS turi būti perkeliama į naujesnes ar patobulintas Windows operacinių sistemų versijas. Į kitas OS nebus perkeliama.

Operacinis scenarijus (Posistemės veiksmų scenarijus iš vartotojo pozicijų) Vartotojui turi būti sukurtas pagrindinis meniu. Duomenys įvedami į lentelės sukurtas formas, turinčias papildomų atributų: mygtukų, iškrentančių sąrašų su galimomis laukų reikšmėmis ir pan. Formų langai atitinkamai apipavidalinti, laukų reikšmės išdėstomos laisva tvarka ir vaizduojamos vartotojui patogiu formatu. Rezultatai turi būti spausdinami atskaitose.

2.3.3. Posistemės struktūros specifikuojimas

Posistemės struktūrą geriausiai vaizduoja komponentinė diagrama (10 pav.). Komponentas yra kodo modulis. Komponentų diagrama yra klasių diagramos fizinis analogas. Posistemę struktūriškai būtų galima suskirstyti į VeiklosPaslaugos, DuomenuValdymas ir VartotojoSasaja komponentus.



10 pav. Posistemės suskirstymas į komponentus

Veiklos paslaugų (skaičiavimo) komponentas turi būti atsakingas už pagrindinių Posistemės funkcijų realizavimą, t.y. asmens kodo tikrinimą, organizacijos pavadinimo formavimą, etatų užimtumo tikrinimą, tarnybos stažo skaičiavimą, rezultatų peržiūrą, atitinkamų ataskaitų formavimą. Komponento realizaciją turi sudaryti visos klasės iš paketų Vartotojo Paslaugos, Saugumas, Funkcijos. Komponentas bus naudojamas kitų programinės sistemos komponentų.

Duomenų valdymo komponentas turi būti atsakingas už duomenų srautų valdymą, duomenų išsaugojimą, užkrovimą, importavimą iš Baltpers ir jų eksportavimą. Komponento realizaciją turi sudaryti visos klasės iš paketo Duomenų Importas Eksportas.

Vartotojo sąsajos valdymo komponentas turi būti atsakingas už vartotojo sąsajos valdymą, duomenų įvedimo ir vaizdavimo operacijas, langų slėpimo operacijas, spalvas ir pan. Komponento realizaciją turi sudaryti visos klasės iš paketo Vartotojo Sąsaja.

Kiekvieno komponento darbui reikalingi kompiuterio operatyvinės atminties resursai.

2.3.4. Duomenų struktūrų specifikacija

Specifikuojama:

- naudojamų duomenų struktūra;
- duomenų vaizdavimo, saugojimo ir gavimo būdai.

Duomenų struktūra. Duomenų struktūra pasirenkama remiantis objektiniu modeliu ir informacijos reikalingos išsaugoti duomenims poreikiu. Loginė duomenų struktūra aprašoma šiais struktūriniais vienetais: duomenų elementais, duomenų agregatais, įrašais, failais ir DB. Fizinė duomenų struktūra aprašoma šiais struktūriniais vienetais: fiziniiais įrašais, duomenų rinkiniais (failais), fizine DB. Duomenų struktūroms modeliuoti naudojamas esybių-ryšių modeliavimas, skirtas projektuoti duomenų bazines. Esybių-ryšių modeliavimo metu identifikuojami svarbūs organizacijos objektai (vadinami esybėmis), esybių savybės (atributai) ir tarpusavio ryšiai.

Posistemės naudojamų duomenų struktūra parenkama pagal Baltpers duomenų struktūrą. Baltpers loginės DB struktūra realizuota naudojant reliacinį modelį. Saugomi šie duomenų tipai: Text, Double, Long Integer, Date/Time, Number.

Duomenų vaizdavimo būdas. Loginiam duomenų modeliui grafiškai vaizduoti dažniausiai naudojama ovalinė diagrama ir stačiakampė diagrama (duomenų elementai vaizduojami stačiakampiais, o ryšiai tarp elementų - linijomis) [9].

Vaizdavimui pasirenkame antrą būdą. Čia objektas – daikto, asmens ar reiškinių apibūdinimas (įrašo DB ekvivalentas), atributas – objekto savybė duotu laiko momentu (lauko ekvivalentas), metodas – veiksmai, kuriuos turi atlikti objektas. Duomenys laikomi .mdb formato faile. Posistemės

vidinės DB reliacinio modelio ekrano vaizdas pateiktas 1 priede. Posistemės duomenų struktūrą klasių diagrama pateikta 11 pav.



11 pav. Duomenų struktūrą klasių diagrama

Pagrindinės klasės yra Tikrinimas, Asmuo, Etatas, TarnybosLaikas, Organizacija. Duomenų modelyje esančių esybių aprašymai pateikti 4 lentelėje.

Duomenų struktūrų modelio esybės

4 lentelė

<i>Esybė</i>	<i>Aprašymas</i>
Vartotojai	Saugoma informacija apie Posistemės vartotojus, prisijungimo vardus, vartotojų tipus. (Pagal vartotojo tipą nustatomos vartotojui prieinamos Posistemės funkcijos).
Vadovas	Saugoma informacija apie personalo vadovus
Specialistas	Saugoma informacija apie personalo specialistus
Asmuo	Saugoma informacija apie KAS dirbantį personalą
Etatas	Informacija apie KAS etatus (kiek ir kokių yra padaliniuose etatų)
Organizacija	Saugoma informacija apie KAS struktūrinius padalinius (kokie padaliniai yra, kam jie pavaldūs, kada įkurti)
Stažas	Saugomos kiekvieno KAS dirbančio asmens priėmimo į darbą ir atleidimo iš darbo datos
Tikrinimas	Saugomi duomenų tikrinimo funkcijų sąrašas

Duomenų saugojimas. IS duomenų struktūroms saugoti naudojama MS Access. Kiekvienai klasei saugoti DB sudaroma atskira lentelė. DB bus galima papildyti naujomis lentelėmis. Leistina į lenteles įterpti naujus laukus, bet neleistina redaguoti lentelių struktūros, t.y. keisti laukų pavadinimų.

Duomenų pasiekimas. Visual Basic aplinkoje DB failams pasiekti yra keletas būdų. Galima sukurti paprastą DB aplikaciją, kuri įgalintų peržiūrėti DB įrašus. Siekiant sukurti tokią aplikaciją, reikia įterpti duomenų kontrolės funkciją į naują formą ir ją sujungti su DB. Galime susikurti DB failą naudodami DB aplikaciją arba galime naudoti DB failus, kurie eina kartu su VB programa. Tokia duomenų kontrolė galės dirbti tik su limituota duomenų rūšimi ir griežtai tik su MS Access DB. Yra ir galingesnis duomenų pasiekimo mechanizmas veikiantis VB aplinkoje – ADO, kuris remiasi ActiveX komponentų panaudojimu ir gali dirbti įvairiose platformose (skirtingose PC sistemose) su skirtingomis programinėmis kalbomis ir gali pasiekti daug skirtingų duomenų rūšių (pvz. e-mail tekstas, Interneto naršyklės, grafika, reliacinės ir nereliacinės DB). Kadangi Projekto uždavinys yra apdoroti MS Access duomenis, tai mūsų uždavinio sprendimui užteks panaudoti paprastesnį pirmą būdą. Naudosime duomenų susiejimo (eksporto-importo) mechanizmą, kurio pagalba Posistemėje bus sukuriama Baltpers DB lentelių kopijos su analogiška struktūra. Jas bus galima tvarkyti MS Access priemonėmis. Posistemės vartotojas prieš importuojant lenteles turės atverti importo paskirties DB (mūsų atveju - Baltpers DB). Reikia tik naudojantis standartinėmis Windows funkcijomis nustatyti ryšius tarp lentelių ir importuoti arba eksportuoti duomenis. Tokiu būdu kiekvienas atnaujinimas lentelėse fiksuojamas ir Baltpers DB, o pakeitimai DB fiksuojami ir susietose Posistemės lentelėse.

Prisijungimas prie DB bus vykdomas naudojant prisirišimo prie DB mechanizmą - DAO (Data Access Objects). Naudojama biblioteka MS DAO 3.6 Object Library. DAO yra optimalus dirbant su Microsoft Access kuriamomis bylomis, bet taip pat gali prieiti prie ODBC duomenų šaltinių per Microsoft Jet duomenų bazių jungtuvą. Jei kompiuteryje yra instaliuota ODBC (programa, kuri sujungia skirtingas duomenų bazines), reikalui esant galima turėti ryšį ir su kito tipo duomenų bazėmis, pvz., Oracle, My SQL.

Posistemės apdorotą informaciją numatoma saugoti šiose lentelėse:

- Lentelė: Asmuo. Lentelės paskirtis – saugoti informaciją apie asmenį. Atributai: PersonalCode – Pavardė– Vardas–KodoEilėsNr - Kiti atributai. Lentelė turi paprastą pirminį raktą – AsmensKodas;

- Lentelė: Organizacija. Paskirtis – saugoti informaciją apie organizaciją. Atributai: OrgNo – Organame – DateStart – DateEnd - PavaldumoNo. Paprastas pirminis raktas – OrgNo.

- Lentelė Etatas. Paskirtis – saugoti informaciją apie etatą (pareigybę). Atributai: EtatoID – Organo – EtatoPavadinimas – EtatuKiekis – UzimtaEtatu – StartDate – EndDatev. Paprastas pirminis raktas – EtatoID.

- Lentelė: Tarnybos laikas. Paskirtis – saugoti duomenis apie ištarnautą laiką. Atributai: StazoID – PersonalCode – Pavardes – Vardas – PriemimoData – AtleidimoData – EndDatev - Kiti atributai. Paprastas pirminis raktas – StazoID.

Lentelė: Tikrinimas. Paskirtis – saugoti duomenis apie duomenų tikrinimo operacijas. Atributai: ID – PersonalCode – Pavadinimas – OrgNo – Statusas – Etatas – Stazas –Specialistas – Vadovas. Pirminis raktas – FunkcijosID.

Lentelė: Vadovas. Paskirtis – saugoti duomenis apie personalo vadovus. Lentelė turi paprastą pirminį raktą – VadovoID.

Lentelė: Specialistas. Paskirtis – saugoti duomenis apie personalo specialistus. Lentelė turi paprastą pirminį raktą – SpecialistoID.

Lentelė: Vartotojai. Paskirtis –saugoti duomenis apie Posistemės vartotojus. Pirminis raktas – ID.

Reikalavimai importuojamiems duomenims. Kad Posistemės apskaičiuoti rezultatai būtų teisingi, sistemoje Baltpers duomenys turi atitikti šiuos patikimumo reikalavimus: duomenys turi būti aktualūs (visuose padaliniuose iš anksto turi būti į DB suvesti visi etatai), teisiškai atlikti visi paskyrimai į pareigas, kiekvieno padalinio užimtų ir laisvų etatų skaičius Baltpers - e turi sutapti su užimtų ir laisvų etatų skaičiumi padalinio etatų sąrašė, neturi būti personalo priskirto panaikintam padaliniui ir neturi būti padalinio be priskirto personalo.

2.3.5. Programinės įrangos komponentų specifikavimas

Šioje Posistemės kūrimo fazėje aprašomos PĮ paketus sudarančios klasės, klasių tarpusavio asociacijos, apibrėžiami metodai ir jų sąveikos algoritmai. Atliekami šie žingsniai:

- detalizuojant Projekto 5.3. skyriuje aprašytus modelius išskiriamos klasių operacijos;
- sukuriama algoritmus aprašančios operacijos;
- aprašoma klasių reakcija į išorinius įvykius;
- aprašomos klasių ryšiai (asociacijos);
- apibrėžiami klasių atvaizdavimai;
- nusprendžiama dėl klasių priklausomybės moduliams.

Šiame Projekto skyriuje naudojami šie terminai:

- objektų klasės - objektų (formų, modulių, lentelių ir pan.) tipai;
- metodai ar funkcijos - objektų operacijos;
- objektų atributai – kintamieji.

PĮ architektūroje galima išskirti šiuos klasių tipus:

- bazinės klasės (pagrindinių klasių grupė visoms kitoms klasėms);
- funkcinės klasės (atlieka įvairias specializuotas funkcijas);
- valdytojų klasės (Managers classes) (skirtos duomenų valdymui, vartotojo sąsajos valdymui);
- duomenų struktūrų klasės;
- langų klasės (Window classes) (bazinės programos langų vaizdavimo bei valdymo klasės).

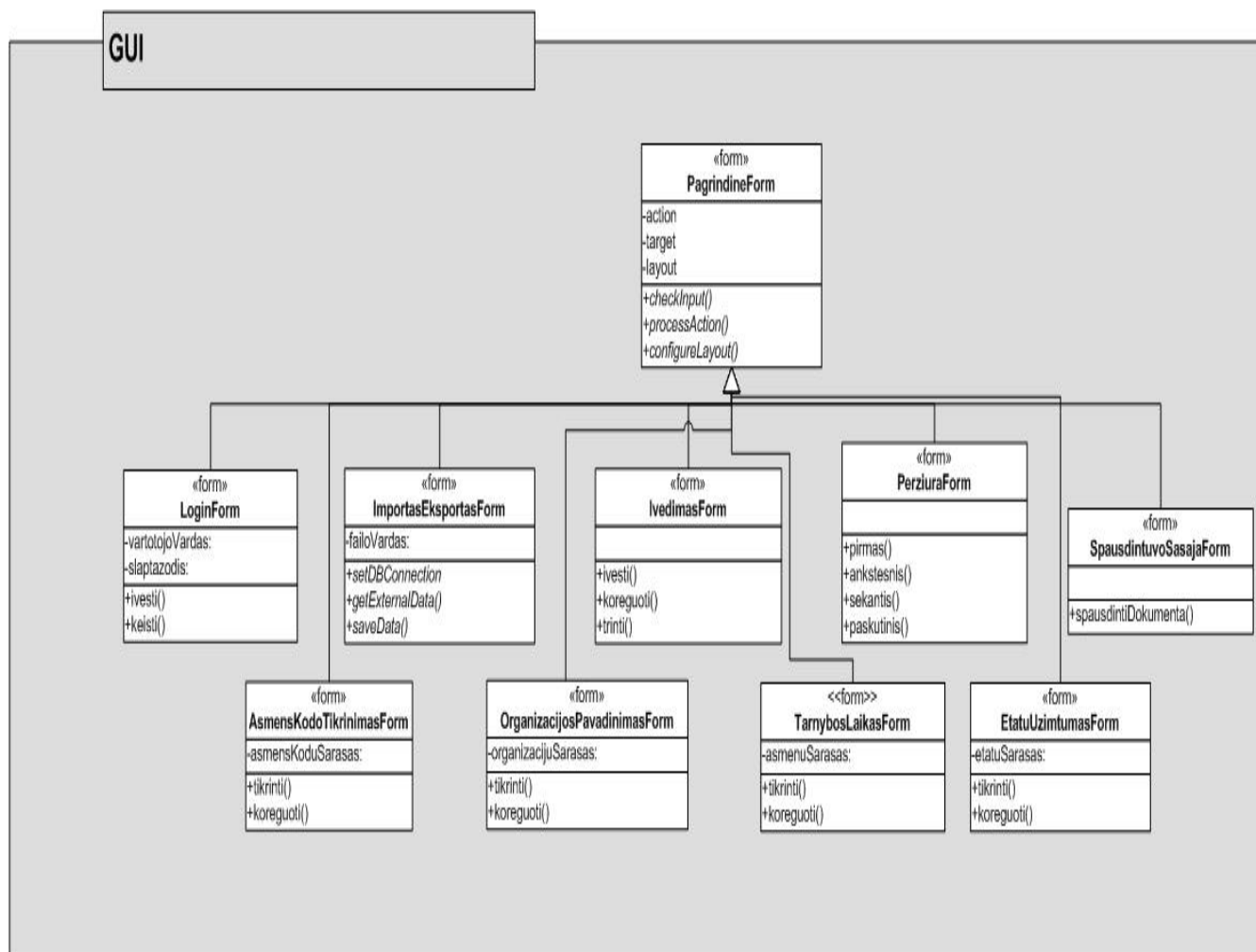
Ne visos Posistemėje naudojamos klasės yra vaizduojamos architektūros modelyje. Praleidžiami standartiniai sąveikos elementai (pvz. “OK”, “Baigti” mygtukai, statinis tekstas, paveikslukai ir kt.). Specifikuojame architektūros modelį sudarančius komponentus (modulius, paketus, klases).

2.3.5.1. Vartotojo sąsajos komponentas

Specifikuojami komponentai naudoja tam tikras sąsajas, kurios taip pat turi būti aprašytos. Vartotojo sąsajos pakete pateikiamos formos (ribinės klasės), per kurias Posistemės vartotojai bendrauja su Posisteme. Tai – prisijungimo forma, personalo duomenų tikrinimo formos, personalo duomenų koregavimo formos, ataskaitų spausdinimas ir kt. Formos turi papildomų atributų: mygtukų, iškrentančių sąrašų su galimomis laukų reikšmėmis ir kt. Formų langai atitinkamai

apipavidalinti, laukų reikšmės jame išdėstomos laisva tvarka ir vaizduojamos vartotojui patogiu formatu. Duomenų įvedimui skirtos formos sudarytos iš vienos ar kelių Posistemės DB lentelių.

Vartotojo sąsajos struktūra pateikiama paketo Vartotojo Sąsaja klasių diagrama (12 pav.).



12 pav. Vartotojo sąsajos paketo klasių diagrama

Klasių diagrama rodo sistemą sudarančias klases ir ryšius tarp klasių.

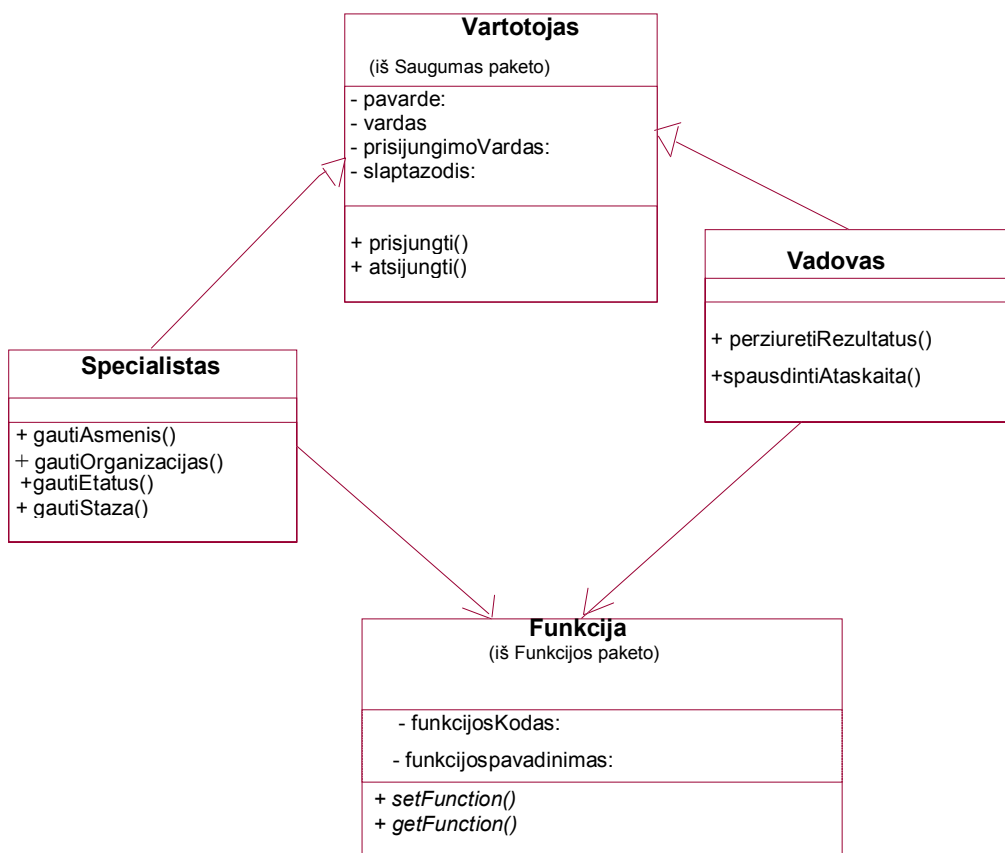
Vartotojo sąsajos pakete yra tik sistemos vaizdavimo lygio klasės. Šiame pakete esančios klasės visoms funkcijoms atlikti naudoja klases esančias Veiklos Paslaugos pakete. Vartotojo sąsajoje skaičiavimai neatliekami. Komponentas sąveikauja su Veiklos Paslaugos komponentu.

2.3.5.2. Veiklos paslaugų komponentas

Veiklos Paslaugos paketą sudaro valdančios klasės, realizuojančios visą Posistemės funkcionalumą t.y. personalo vadovo ir personalo skyriaus specialisto atliekamas funkcijas. Šitas klases naudoja kitų paketų klases. Architektūriniu požiūriu Veiklos Paslaugos paketą tikslinga

išskaidyti į tris pagrindinius subpaketus, t.y. Saugumas, Vartotojo Paslaugos, Funkcijos. Toliau pateikiamos veiklos paslaugų komponentą sudarančių klasių diagramos.

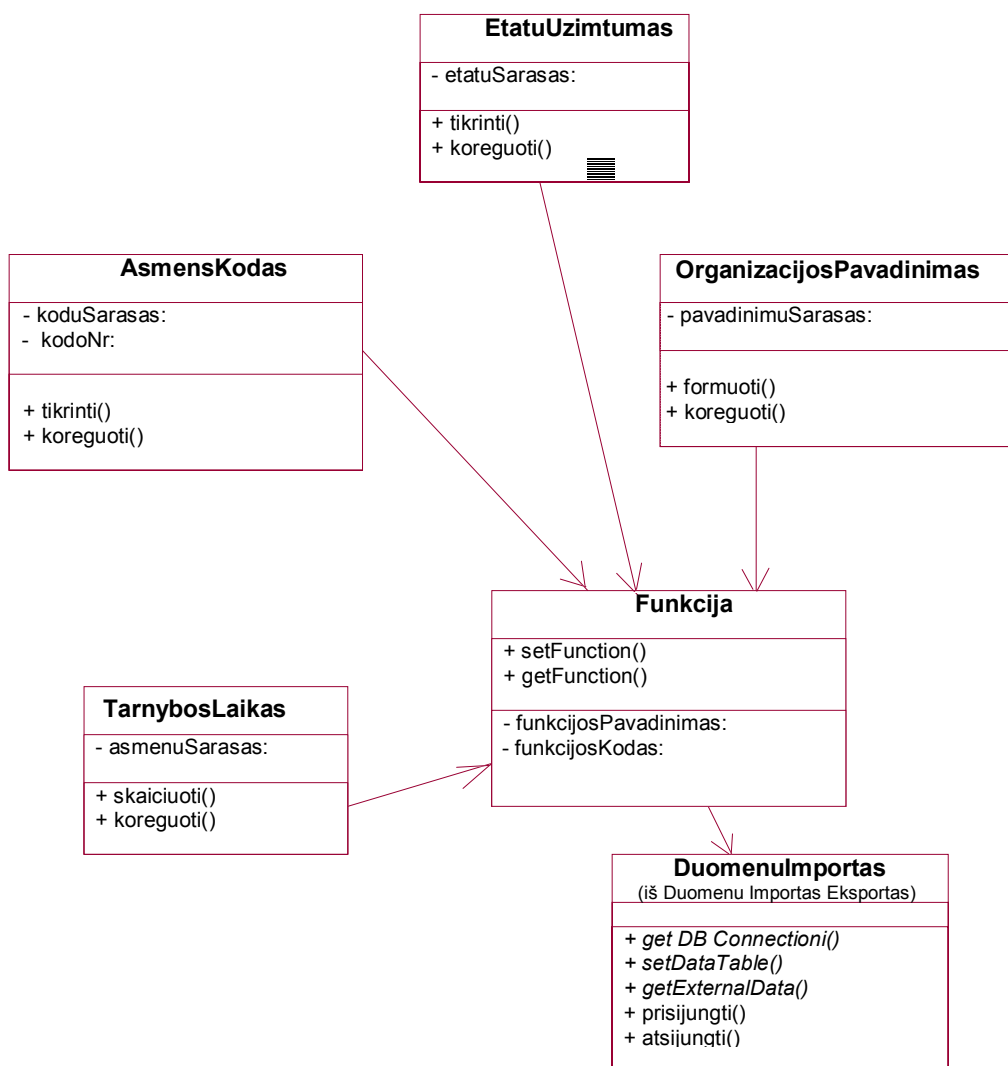
Paketas Vartotojo Paslaugos. Paketo detalizavimas pateikiamas 13 pav.



13 pav. Paketo Vartotojo Paslaugos klasių diagrama

Klasė Vadovas skirta personalo vadovui pasiekti jo funkcijas. Klasė atsakinga už duomenų tikrinimo ir skaičiavimo rezultatų peržiūrą, ataskaitų gavimą ir spausdinimą. Klasė Specialistas skirta personalo specialistui pasiekti jo funkcijas ir atsakinga už duomenų apie asmenis, organizacijas, etatus bei tarnybos laiką gavimą, jų patikrinimą ar perskaičiavimą.

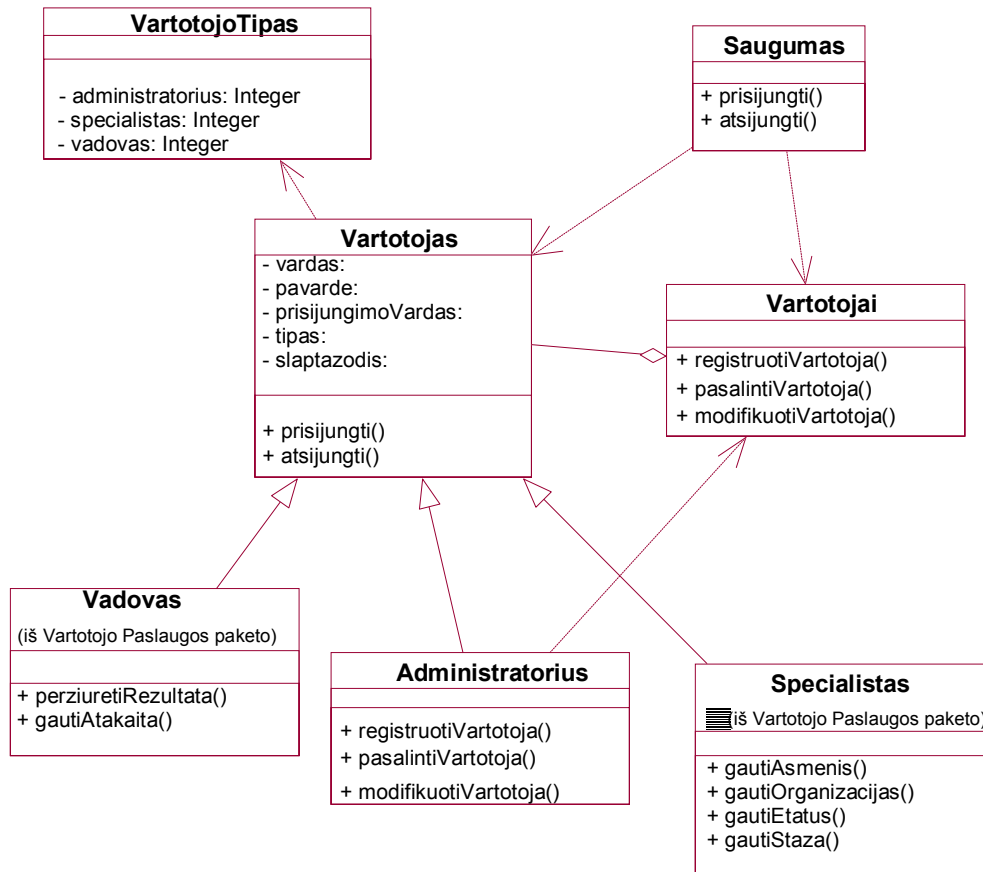
Paketas Funkcijos. Paketo detalizavimas pateikiamas 14 pav.



14 pav. Paketo Funkcijos klasių diagrama

Klasė Funkcija skirta Posistemės vartotojui išsikviesti reikiamą tikrinimo (skaičiavimo) funkciją. Klasės AsmensKodas, EtatuUzirtumas, OrganizacijosPavadinimas ir TarnybosLaikas realizuoja duomenų tikrinimo ir koregavimo funkcijas. Be to TarnybosLaikas funkcija (rezultatas išreikštas metais, mėnesiais ir dienomis) didina ataskaitų informatyvumą. Laisvo/užimto etato tikrinimo funkcija automatizuoja tikrinimo operaciją. Klasė Funkcija naudojami klasės Duomenulimportas paslaugomis.

Paketas Saugumas. Paketo detalizavimas pateikiamas 15 pav.

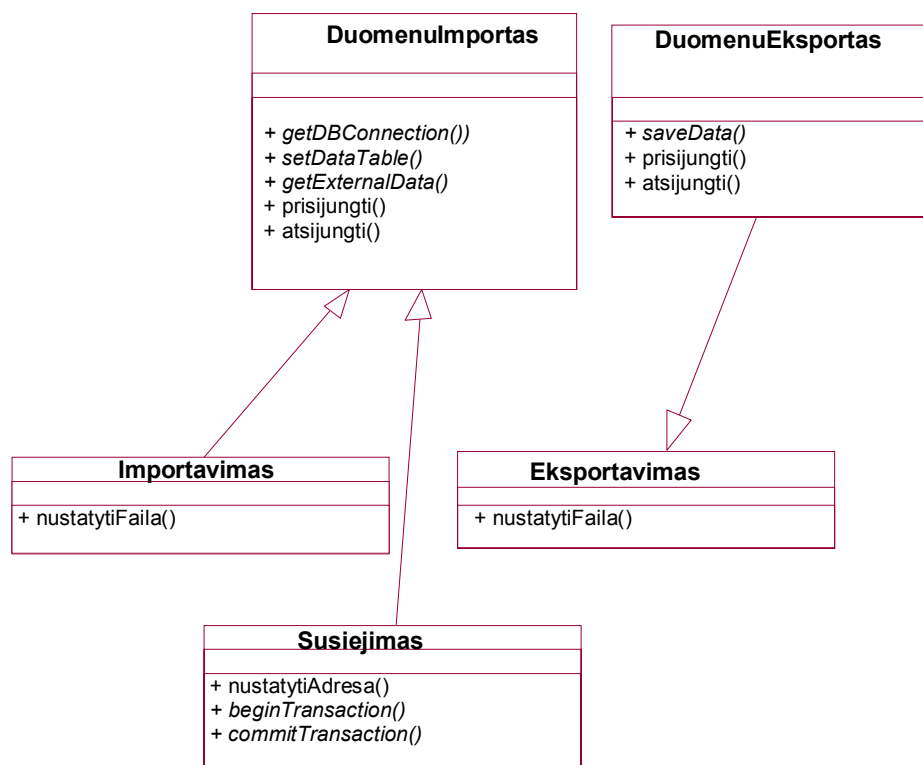


15 pav. Paketo Saugumas klasių diagrama

Pakete pateikiamos klasės skirtos vartotojų prisijungimui prie Posistemės ir vartotojų administravimui. Klasė Administratorius skirta administratoriui pasiekti jo funkcijas ir atsakinga už naujų vartotojų kūrimą, šalinimą, redagavimą. Klasė Saugumas teikia vartotojui galimybę prisijungti, taip pat autentifikuoja Vartotoją. Klasė VartotojoTipas apibrėžia vartotojų tipus pagal jų turimas teises. Klasė Vartotojas atsakinga už prisijungimą, atsijungimą, informacijos apie vartotoją laikymą. Klasė Vartotojai Klasės teikia funkcija skirtas Posistemės administratoriui valdyti vartotojus.

2.3.5.3. Duomenų valdymo komponentas

Komponento realizaciją sudaro visos klasės iš paketo Duomenų Importas Eksportas. Paketo detalizavimas pateikiamas 16 pav.



16 pav. Paketo Duomenų Importas Eksportas klasių diagrama

Paketas atsakingas už duomenų importą iš Access dokumentų, iš Baltpers sistemos ir už patikrintų bei koreguotų duomenų eksportą į minėtą sistemą arba į atskirus failus. Paketo klases naudoja Veiklos Paslaugos paketo klases. Paketas yra statiskai prijungiamas prie Veiklos Paslaugos komponento. Klasė Importavimas skirta importuoti duomenis iš įvairių formatų failų (.mdb, .txt, xls ir kt.). Klasė Eksportavimas skirta eksportuoti duomenis iš Posistemės į įvairių formatų failus (.mdb, .txt, xls ir kt.). Klasė Susiejimas skirta Posistemės lentelių susiejimui su kitų DB lentelėmis. Importo ir eksporto sąsajos leidžia esant poreikiui prijungti ir naujus duomenų formatus. Aprašytų komponentų klasių detalizavimas gali būti pateiktas atskiru PĮ detalios architektūros dokumentu.

2.3.6. Vartotojo sąsajos specifikacija

2.3.6.1. Reikalavimai vartotojo sąsajai

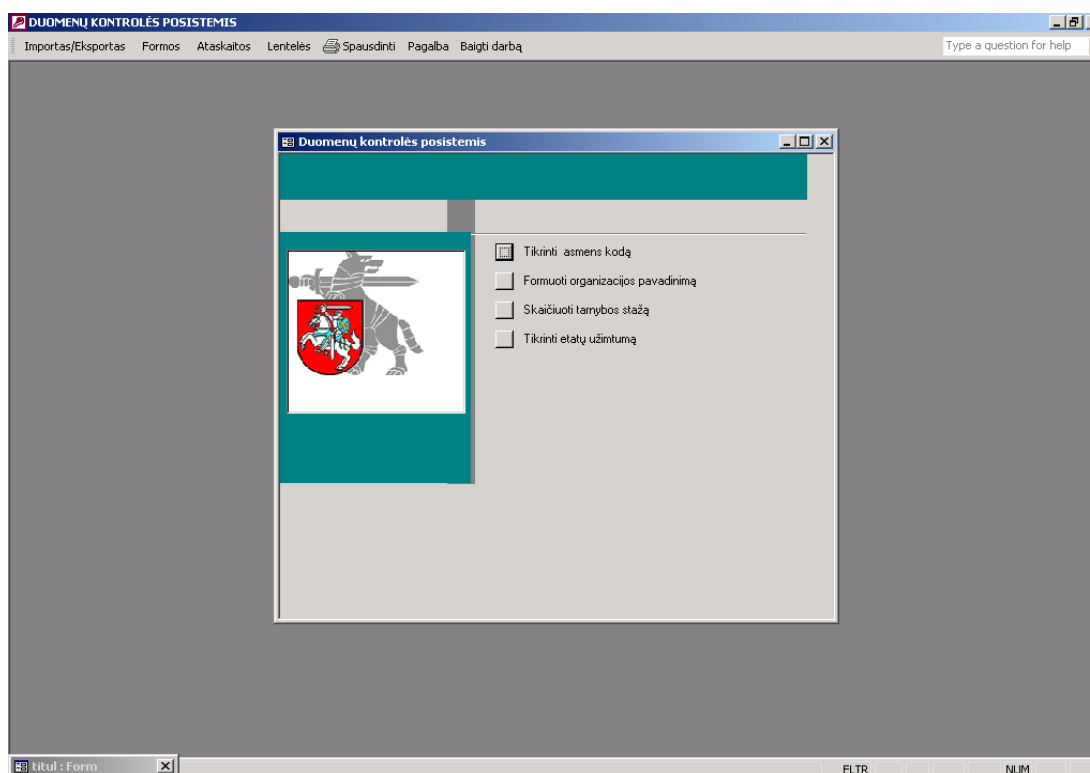
- Vartotojui turi būti suteikta paprasta ir suprantama, lengvai valdoma sąsaja;
- Turi būti langų tipo vartotojo sąsaja;
- Kiek įmanoma lankstesnis vartotojo sąsajos keitimas pagal vartotojo poreikius;
- Langai turi būti neperkrauti nereikalingais elementais, o informacija neturi būti perteklinė;
- Keletas peržiūros langų vienu metu;
- Vartotojas gali perkelti darbinus langus, nusistatyti „karštus“ klavišus (angl. shortcuts);

- Paprastas klaidų apdorojimas;
- Aiškūs vartotojo pranešimai turi būti formuojami grafiniu bei skaitmeniniu būdu;
- Suprantama ir lakoniška pagalba (Help-as), paaiškinanti kaip naudotis programiniu produktu.

Toliau vartotojo sąsają numatoma tobulinti atsižvelgiant į vartotojų pastabas (pvz. ekrane saugoti tik tai svarbiausią informaciją, kiek įmanoma lanksčiau keisti vartotojo sąsają pagal vartotojo poreikius, siekti, kad informacija būtų išdėstyta nuosekliai, logiškai, struktūriškai ir pan.).

2.3.6.2. Vartotojo sąsajos realizacija

Vartotojo sąsaja - grafinė vartotojo sąsaja (GUI). Projektuojant vartotojo sąsają pasirenkama dialogo struktūra (ekranas, interaktyvūs objektai ir pan.). Sąsaja turi leisti suformuluoti bet kokią užduotį, kurią gali įvykdyti programų sistema. Vartotojo sąsaja suprojektuota taip, kad būtų patogus ir paprastas naudojimas, naudojami terminai suprantami ir nereikalaujantys profesinio patyrimo. Meniu yra realizuotas punktas „Pagalba“. Suaktyvinus šį meniu punktą matoma programinio produkto dokumentacija. Posistemės pagrindinės formos vaizdas pateiktas 17 paveiksle, o per vartotojo sąsają pasiekiamų funkcijų navigavimo planas pateiktas 2 priede.



17 pav. Pagrindinė forma „Duomenų kontrolės posistemė“

2.3.7. Programinės įrangos kodavimas

Rašant programos tekstą pateikiami visų modulinėje schemeje parodytų procedūrų ir funkcijų aprašymai. Be to visų komponentų specifikacija turi būti pateikta pagal UML notaciją (angl. notation). Naudojama programavimo kalba Visual Basic for Applications. PĮ kodavimas apima:

- programos teksto rašymą moduluose asocijuotuose su atitinkamomis formomis;
- programos modulių apjungimą;
- sintaksės klaidų paiešką ir taisymą;
- logikos klaidų paiešką ir taisymą.

Programa buvo rašoma prisilaikant bendrųjų programinės įrangos kodo rašymo taisyklių. Programuojami keturi moduliai, skirti aptarnauti (teikti funkcijas) šioms formoms: “Asmens kodo tikrinimas”, “Etatų užimtumas”, “Organizacijos pavadinimas”, “Tarnybos laikas”. Moduliai turi sąveikauti tokiu būdu, kad visa IS būtų saugi, greitai, patikima dirbant su maksimaliu apkrovimu. Naudotasi MicrosoftActiveX Data Objects 2.1 Library biblioteka. Modulio viduje programinis kodas rašomas procedūromis (funkcijomis). Kiekvienam įvykiui sudarytas metodas, kuris vykdomas įvykus tam tikram įvykiui. Standartinių elementų programuoti atskirai nereikia, nes jų metodai aprašyti minėtos bibliotekos klasėse. Programoje yra realizuotas klaidų apdorojimas. Vartotojas apie klaidas informuojamas pranešimais. Funkcijų programinio kodo tekstai pateikiami 4 priede.

2.4. Projekto grafikas ir išlaidų planas

2.4.1. Posistemės kūrimo biudžetas ir resursai

Kadangi Projektas buvo vykdomas studijų metu, projekto biudžetas ir finansavimas nebuvo numatyti. Visus reikalingus resursus ir priemones Projekto vykdytojas turi. Finansavimas nėra būtinas.

Projekto resursai

5 lentelė

<i>Eil. Nr.</i>	<i>Resurso pavadinimas</i>	<i>Resursų kiekis (vnt.)</i>
1.	Projekto vadovas	1
2.	Projektuotojas	1
3.	Programuotojas	1
4.	Testuotojas	1
5.	Kompiuteris	1
6.	PĮ Microsoft Visual Basic for Applications 6.3	1
7.	PĮ MS Access 2003	1
8	PĮ Internet Explorer (nuo Version 6.0 ir aukščiau)	1
9.	PĮ MS Visio 2003	1

2.4.2. Projekto išlaidos

Projekto išlaidos. Projekto išlaidos yra minimalios, nes Projektas vykdomas kaip magistro studijų darbas. Pagrindines Projekto išlaidas sudaro laikas, skirtas Projektui kurti bei programai rašyti.

Laiko ir kainos apribojimai. Kadangi tai mokomasis Projektas, yra ribojamas tik Projekto atlikimo laikas – magistrantūros studijų pabaiga. Kainos apribojimų nėra.

Konkurencija rinkoje. Konkurencijos rinkoje nėra, nes Projektas nėra komercinis.

Projekto išlaidų ir pelno santykis. Projektas nesiekiantis pelno. Projektas duos netiesioginę naudą – suteiks galimybes vartotojams realizuoti jiems reikalingus specifinius poreikius.

2.4.3. Projekto grafikas

Pagrindiniai Projekto etapai ir jų atlikimo terminai:

- 2003-12-01 – 2004-01-30 suformuluota Projekto tema, iškelti tikslai, uždaviniai, pasirinktas projekto vadovas;
- 2004-02-01 – 2004-06-30 atlikta Projekto analizė, nustatyti pradiniai reikalavimai (funkciniai ir nefunkciniai);
- 2004-09-01 – 2004-12-20 apibrėžta pradinė programos struktūra, apgalvoti galimi objektai, galutinai nustatytas naudotinas objektinis modelis;
- 2005-02-01 – 2005-06-30 sudaryta reikalavimų ir PĮ architektūros specifikacija, suprogramuoti programos komponentai ir jie integruoti tarpusavyje, suprogramuota vartotojo sąsaja;
- 2005-09-01 – 2005-12-23 parengta Projekto dokumentacija, sudaryta Posistemės vartotojo dokumentacija, atliktas Posistemės testavimas, darbas paruoštas pristatyti kvalifikacinei komisijai.

2.5. Vartotojo dokumentacija

Posistemės vartotojo dokumentacija pateikiama spausdinta (4 priedas), kompaktinėje plokštelėje, Posistemės pagalbos failuose (Help-e) kartu su Posistemės PĮ. Kadangi Posistemė dar nėra visiškai išbaigta, todėl pateikiama preliminarini vartotojo dokumentacija:

Vartotojo dokumentaciją sudaro šios pagrindinės dalys :

- techninės įrangos aprašymas;
- Posistemės aprašymas;
- programos įdiegimo instrukcija;

- prisijungimas prie personalo duomenų bazės;
- Posistemės funkcinis aprašymas;
- Posistemės vartotojo vadovas.

Žemiau pateikiamas trumpas aprašymas, kaip naudotis programos meniu:

- **Menu.**
 - **Importas/Eksportas** – įkelti, eksportuoti personalo duomenis.
 - **Linked Table Manager** – susieti Posistemę su DB lentelėmis.
 - **Duomenų importas** – įkelti personalo duomenis iš Baltpers.
 - **Duomenų eksportas (Ctrl+E)** – eksportuoti skaičiavimų rezultatus.
 - **Formos** – duomenų formų valdymas.
 - **Switchboard** – iškviesti pagrindinę mygtukų formą.
 - **Asmens kodo tikrinimas** – iškviesti formą “Asmens kodo tikrinimas”.
 - **Organizacijos pavadinimas** – iškviesti formą “Organizacijos pavadinimas”.
 - **Tarnybos laikas** – iškviesti formą “Tarnybos laikas”.
 - **Etatu užimtumas** – iškviesti formą “Etatu užimtumas”.
 - **Ataskaitos** – ataskaitų peržiūra ir spausdinimas.
 - **Asmens kodo tikrinimas** – generuoti ataskaitą “Asmens kodo tikrinimas”.
 - **Darbuotojo turimas stažas** – generuoti ataskaitą “Darbuotojo turimas stažas”.
 - **Laisvi/užimti etatai** – generuoti ataskaitą “Laisvi/užimti etatai”.
 - **Pilnas organizacijos pavadinimas** – generuoti ataskaitą “Pilnas organizacijos pavadinimas”.
 - **Lentelės** – duomenų lentelių valdymas.
 - **Asmens kodo tikrinimas** - iškviesti lentelę “Asmens kodo tikrinimas”.
 - **Etatu užimtumas** – iškviesti lentelę “Etatu užimtumas”.
 - **Organizacijos pavadinimas** – iškviesti lentelę “Organizacijos pavadinimas”
 - **Tarnybos laikas** – iškviesti lentelę “Tarnybos laikas”
 - **Switchboard Items** – rodyti mygtukų formoje sukurtus objektus
 - **Spausdinti (Ctrl+P)** – leidžia atsispausdinti iškviestus objektus
 - **Pagalba** – vartotojo pagalba

- **Pagalba** (*Ctrl+H*) - paleisti pagalbos Posistemę.
- **Apie programą** – rodyti informaciją apie Posistemę.
- **Baigti darbą** (*Ctrl+X*) – išeiti iš Posistemės.

Naudojimosi programa pavyzdys (Funkcija „Asmens kodo tikrinimas“):

- paleidžiame nuorodą į failą Duoemenu-kontroles_posisteme.mdb - pasirodo pagrindinė forma „Duomenų kontrolės posistemė“;
- spaudžiame mygtuką „Tikrinti asmens kodą“ – pasirodo forma „Asmens kodo tikrinimas“;
- su mygtuku „Record“ pasirenkame norimą įrašą arba ranka į formos laukus suvedame duomenis apie reikiamą asmenį;
- spaudžiame mygtuką „Tikrinti asmens kodą“;
- tikrinimo rezultatas automatiškai atsiranda formos lauke „Tikrinimo rezultatas“;
- rezultatą galima atspausdinti, nuspaudus mygtuką „Print“;
- rezultatą galima išsaugoti, aktyvavus programos meniu punktą „File/Save“.

Parengtos vartotojo dokumentacijos turėtų užtekti pradedant dirbti su Posisteme, todėl organizuoti vartotojų apmokymų nėra būtinybės. Kadangi programa yra nedidelė ir nesudėtinga, rengti specialaus Posistemės administratoriaus vadovo nenumatoma.

3. TYRIMO DALIS

Šioje Dokumento dalyje tiriamos su Projektu susijusios problemos ir rizikos, pateikiamas PĮ kokybės įvertinimas, kokybės tobulinimo galimybių tyrimas, siūlomų patobulinimų pagrindimas ir realizavimas.

3.1. Projekto išeiga

3.1.1. Atviri klausimai (problemos)

Svarbūs Posistemei klausimai arba problemos, neturinčios aiškaus sprendimo yra šie:

- Lietuvos Respublikos Seimas planuoja priimti Asmens duomenų apsaugos įstatymo pakeitimus ir šiuo metu yra dar neaišku, kas ir kada bus keičiama;
- Baltpers sistemos valdytojas nėra nusprendęs, kaip personalo apskaitos sistema bendradarbiaus su PPBS VIS ir LogVis sistemomis, kas turės įtakos Posistemės vystymui.

3.1.2. Problemos diegimo aplinkai

Baltpers duomenų bazė patalpinta centriniame serveryje. Posistemės programinė įranga diegiama naudotojo kompiuteryje. Jei bus užtikrinti keliami reikalavimai techninei įrangai ir duomenų perdavimo tinklui, problemų ir potencialių konfliktų diegiant Posistemę neturėtų kilti.

3.1.3. Neigiamas vartotojų nusiteikimas

Kadangi Posistemės vartotojai yra personalo specialistai, turintys tik labai ribotas žinias informacinių technologijų srityje, todėl galimas vartotojų neigiamas nusiteikimas dirbti su kuriama Posisteme tol, kol vartotojai įsisavins ir pripras prie naujo programinio produkto. Parengta PĮ vartotojo dokumentacija padės spręsti problemą ir suteiks pilną informaciją apie programą bei jos naudojimą.

3.2. Rizikos įvertinimo ir mažinimo planas

3.2.1. Posistemės kūrimo rizikos įvertinimas

Posistemės kūrimo riziką galima išskirti į keletą grupių.

Projekto rizika. Ją apsprendžia:

- reikalavimų specifikacijos pasikeitimai realizavimo fazėje (reikalavimų pasikeitimai gali būti įtakoti programos kūrimo proceso);
- PĮ architektūros klaidos.

Su projekto vykdytojais susijusi rizika. Ją apsprendžia:

- projekte dalyvaujančio personalo patirtis ir sugebėjimai;
- Projekto vadovo ar vykdytojų pasikeitimas.

Su vartotoju susijusios rizikos. Ją apsprendžia:

- užsakovo reikalavimų pasikeitimai (reikalavimų pasikeitimai pareikalautų laiko ir lėšų jiems įdiegti).

Techninė rizika. Ją apsprendžia:

- kompiuterio gedimas (pvz., kompiuterio standaus disko gedimas turėtų didelės įtakos projekto eigai, kadangi gali būti prarasti jau sukaupti projekto duomenys).

Proceso rizika. Ją apsprendžia:

- gauti papildomi darbai (papildomi darbai labai apsunkintų projekto baigimą laiku);
- aplinkos pasikeitimas (negalima pasinaudoti kokia nors darbo priemone).

Posistemės kūrimo rizikos

6 lentelė

<i>Nr.</i>	<i>Rizikos faktorius</i>	<i>Tikimybė</i>	<i>Įtakos įvertinimas*</i>
1.	Reikalavimų specifikacijos pasikeitimai realizavimo fazėje (projekto rizika)	Žema	10
2.	Kompiuterio gedimas (techninė rizika)	Vidutiniška	9
3.	Projekte dalyvaujančio personalo patirtis ir sugebėjimai (su projekto vykdytojais susijusi rizika)	Žema	8
4.	Projekto vadovo ar vykdytojų pasikeitimas (su projekto vykdytojais susijusi rizika)	Žema	8
5.	PĮ Architektūros klaidos (projekto rizika)	Vidutiniška	8
6.	Užsakovo reikalavimų pasikeitimai (su vartotoju susijusi rizika)	Vidutiniška	7
7.	Papildomi darbai (proceso rizika)	Vidutiniška	5
8.	Aplinkos pasikeitimas (proceso rizika)	Žema	5

*Įtakos įvertinimo skalė: 1 – rizika nesukelia problemų, 5 – leistinas problemos įtakos lygis, 10 - rizika sukelia rimtas problemas.

3.2.2. Rizikos valdymo ir mažinimo planas

Sudaromas rizikos valdymo ir mažinimo planas (7 lentelė).

Rizikos valdymo ir mažinimo planas

7 lentelė

<i>Nr.</i>	<i>Rizikos faktorius</i>	<i>Problemos sprendimas</i>
1.	Reikalavimų specifikacijos pasikeitimai realizavimo fazėje	Kiekvienu atveju su užsakovu derinti reikalavimų pasikeitimus. Svarbių reikalavimų pasikeitimų įgyvendinimui projekto realizavimo fazėje numatyti reikiamą laiko rezervą
2.	Kompiuterio gedimas	Reikėtų periodiškai susikurti Projektui svarbiausios kopijas į CD-R laikmenas ar į kitus standžiuosius diskus. Tokiu atveju prarastos informacijos kiekis būtų nedidelis.
3.	Projekte dalyvaujančio personalo patirtis ir sugebėjimai	Personalui duoti uždavinius bei naudoti PĮ, su kuria personalas jau turi darbo patirties arba numatyti laiko rezervą Projekte dalyvaujančiam personalui įsigilinti į probleminius uždavinius Mūsų atveju Projektą vykdo vienas komandos narys, kuris turi sukaupęs pakankamai patirties projekto įvykdymui.
4.	Projekto vykdytojo ligos arba pasikeitimas	Projekte suplanuoti laiko rezervą arba numatyti rezervinius parengto personalo resursus Nors projekto komandą sudaro vienas narys, tačiau jo aplinkoje yra nemažai kolegų iš kurių, reikalui esant, jis gali gauti pagalbą. Projektui plečiantis komanda turėtų didėti.
5.	PĮ architektūros klaidos	Rinktis žinomas ir plačiau naudojamas CASE priemonės, o taip pat numatyti rezervinių CASE priemonių rinkinį.
6.	Užsakovo reikalavimų pasikeitimas	Projekto vykdymo eigą periodiškai reikalinga derinti su užsakovu KAS Personalo departamentu, kad reikalavimų pasikeitimai nebūtų netikėti ar kritiniai.
7.	Papildomi darbai	Atsiradus papildomų darbų, pirmiausiai dirbti prie svarbiausių projekto vietų, kad vartotojui atrodytų, jog praktiškai visas funkcionalumas egzistuoja ir projektas bus pristatytas laiku, likusį funkcionalumą įdiegiant vėliau.
8.	Aplinkos pasikeitimai	Projekto plane numatyti kai kuriuos darbus, kad dėl aplinkos pasikeitimo galimos rizikos įtaka projektui būtų mažiausia.

3.3. Produkto kokybės įvertinimas

Posistemės programinės įrangos architektūra įtakoja šiuos į Posistemės kokybės parametrus:

- išplėtimo galimybę, t.y. apibrėžta sąsaja duomenų importui/eksportui leidžia atsiradus poreikiui prijungti naujus duomenų importo/eksporto šaltinius;
- patikimumą, t.y. Posistemėje realizuotos duomenų apsaugos priemonės minimizuoja Posistemės funkcionalumo pažeidimų tikimybę;
- PĮ naujumą, t.y. Posistemės vartotojai naudosis tik naujausia Posistemės PĮ versija.

PI produkto kokybė buvo vertinama testavimo metu bei įvertinant vartotojų atsiliepimus (anketos vartotojams). Posistemė laikoma tiek kokybiška, kiek teisingi skaičiavimo rezultatai (programinio produkto veiksmų rezultatai).

Išvada: sudaryta Posistemės PI architektūra iš esmės tenkina pagrindinius kokybės reikalavimus specifikuotus šio projekto 2.3.2.skyriuje.

3.4. Posistemės tobulinimo planas

Priklausomai nuo vartotojų poreikių Posistemė bus vystoma, papildant ją naujomis funkcijomis. Numatome tokį Posistemės tobulinimo planą:

- vartotojų (personalo specialistų ir vadovų) apmokymas;
- programinės įrangos naudojimas ir priežiūra (tolesnis programinio produkto naudojimas ir priežiūra priklausys nuo Projekto užsakovo pageidavimų);
- Posistemės funkcionalumo papildymas vartotojui reikalingomis funkcijomis;
- duomenų importo galimybės praplėtimas - Posistemės galimybė dirbti su įvairių formatų (ne tik .mdb) duomenų bazėmis;
- platesnis Posistemės pritaikomumas (ne tik personalo srityje dirbantiems specialistams);
- vartotojo sąsajos tobulinimas ir supaprastinimas (pagal vartotojų informaciją).

4. EKSPERIMENTINĖ DALIS

Šioje Dokumento dalyje atliekamas sukurtos PĮ bei jos patobulinimų eksperimentinis tyrimas – testavimas, aprašomas Posistemės diegimas.

4.1. Programinės įrangos testavimas

Testavimo tikslas - paruošti testinių duomenų rinkinius, ištestuoti PĮ ir įvertinti gautus rezultatus (sulyginti su apskaičiuotais rezultatais). Testavimo plano tikslas – pateikti procedūras ir strategijas, kuriomis naudojantis būtų galima ištestuoti kuriama/sukurtą PĮ ir užtikrinti pasiektus užsibrėžtus kokybės reikalavimus. Siekiama, kad Posistemė būtų aukštos kokybės ir be defektų. Testavimo metu bus tikrinamos kodavimo klaidos, blogo veikimo bei loginės klaidos. Testavimui atlikti naudojami resursai nesiskiria nuo PĮ kūrimui naudotų resursų, t.y. bus naudojamas personalinis kompiuteris. Testavimo personalą sudaro Projekto vykdytojas. MS Access naudojama kaip testavimo aplinka. Daugiau jokių specialių įrankių ar aplinkų nenaudojama.

Minimalūs techniniai apribojimai reikalingi programų sistemos testavimui: 16MB operatyviosios atmintinės (RAM) ir 15'' vaizduoklis. Testavimas bus vykdomas Windows 2000, Windows XP operacinių sistemų platformose. Kitose OS platformose ir kitų PĮ įtaka testuojama nebus.

4.1.1. PĮ testavimo metodai

Pagrindiniai testavimo metodai yra šie:

- statinis programos testavimas. Testuojant tikrinama parašytos programos kodo sintaksės ir semantikos korektiškumas, kintamųjų panaudojimas, nepertekliškumas, optimalumas, teisingų duomenų struktūrų bei programavimo elementų naudojamas;

- struktūrinis vienetų (pavienių modulių) testavimas. Tai "Baltos dėžės" testavimas. Naudojami testiniai atvejai gaunami iš programos struktūros. Po vieną testuojami visi programiniai moduliai. Testavimas atliekamas su testiniais duomenimis Sudėtingesnis komponentas gali būti testuojamas atskiromis paprogramėmis, o paprastesni programos elementai gali būti testuojami tiesiog juos naudojant aukštesnio lygio komponentuose, ir įsitikinant, kad jie veikia korektiškai. Struktūrinis testavimas taikomas pasirinktinai keliems programų sistemos komponentams arba klasių metodams;

- funkcinis vienetų testavimas. Tai "Juodos dėžės" testavimas kai remiamasi įėjimo, išėjimo ir vidinių būsenų kintamaisiais. Testavimo atvejai gaunami remiantis sistemos specifikacija. Funkcinis testavimas taikomas pasirinktinai keliems programų sistemos komponentams arba klasių metodams.

- virtotojo sąsajos testavimas. Testuojama virtotojo sąsaja einant per visus PĮ komponentus;

- integravimo testavimas. Jis atliekamas išbaigtai sistemai, sudarytai iš integruotų komponentų. Toks testavimas turėtų būti „juodos dėžės“ testavimas su testais, gautais iš specifikacijų. Metodo sunkumas yra klaidų lokalizavimas, tačiau palaipsninis integravimo testavimas mažina šią problemą;

- validavimo testavimas. Apibrėžiami kriterijai, koku atveju testavimo metu gauti rezultatai mus tenkins. Validavimo testavimas yra „juodos dėžės“ tipo ir atliekamas pilnai išbaigtam produktui. Jo metu programos darbas sutikrinamas su reikalavimų specifikacijos dokumentu.

- virtotojo (viešas) testavimas. Šis metodas apima bendravimą su virtotoju tikslu įsitikinti ar PĮ yra tinkama virtotojui. Naudojamas „Juodos dėžės“ testavimas – įvedami duomenys ir gauti rezultatai lyginami su duomenimis apskaičiuotais iš anksto. Atsiradus problemų klaidos taisomos.

- „High- order“ testavimas Šis metodas apima atsistatymo, sistemos apsaugos, stresinis testavimas (kaip sistema reaguoja į didelį virtotojų apkrovimą) ir vykdymo testavimas.

4.1.2. PĮ testavimo planas

Virtotojo sąsajos testavimui sudarytas 8 lentelėje pateiktas planas, funkcijų testavimui – 9 lentelėje pateiktas planas, o GUI standarto atitikimo testavimui – 10 lentelėje pateiktas veiksmų planas.

Posistemės virtotojo sąsajos testavimo veiksmų planas 8 lentelė

<i>Testavimo tikslas</i>	Patikrinti, ar sistemos sąsaja atitinka numatytus sąsajos reikalavimus, ar virtotojui yra patogų dirbti su Posisteme.
<i>Technologija</i>	Sukurti ir įvykdyti testavimo atvejus kiekvienai Posistemės formai ir visiems joje esantiems objektams. Tikrinama: - norimi veiksmai ir perėjimai nuo lango prie lango, nuo lauko prie lauko (naudojantis klaviatūra ir pelės pagalba); - ar visi langų objektai (menu, objektų dydžiai, vieta lange ir kiti) atitinka numatytus standartus.
<i>Užbaigimo kriterijai</i>	Visi langai yra patikrinti ir atitinka numatytus standartus.

Posistemės funkcijų testavimo veiksmų planas 9 lentelė

<i>Testo tikslas</i>	Patikrinti, ar sukurtos Posistemės funkcijos veikia teisingai, ar duomenys yra išsaugomi teisingai.
<i>Apimtis</i>	Testavimas turi apimti visų numatytų ir aprašytų Posistemės reikalavimų testavimą
<i>Technologija</i>	Sukurti ir įvykdyti testavimo atvejus kiekvienai funkcijai, tikrinant su teisingais ir klaidingais duomenimis. Šiame teste turi būti patikrinta: - teisingų duomenų atveju gaunami norimi rezultatai; - neteisingų duomenų atveju yra rodomas klaidos pranešimas; - kiekviena funkcija yra realizuota. Posistemėje yra įvedami duomenys virtotojo sąsajos pagalba ir yra analizuojami gauti rezultatai.
<i>Užbaigimo kriterijai</i>	Visi norimi testo atvejai yra patikrinti, netikslumai ir klaidos aprašytos.

<i>Objektas</i>	Testavimo sąlygos, atvejais	Laukiami rezultatai
<i>Laukų pavadinimai</i>	Patikrinti spalvas, dydžius, pavadinimus, gramatiką, vietą ekrane	Objektai atitinka Posistemės reikalavimus ir GUI standartą
<i>Mygtukai</i>	Patikrinti pavadinimus, dydžius, poziciją ekrane, gramatiką	Objektai atitinka Posistemės reikalavimus ir GUI standartą
<i>Duomenų įkrovimas</i>	Patikrinti ar užkraunami tie langai, kurie turi būti	Objektai atitinka Posistemės reikalavimus ir GUI standartą
<i>Pranešimai</i>	Patikrinti pranešimų tekstus, gramatiką	Objektai atitinka Posistemės reikalavimus ir GUI standartą
<i>Bendras lango vaizdas</i>	Patikrinti ar ekrane yra visi įvedimo laukai, mygtukai	Objektai atitinka Posistemės reikalavimus ir GUI standartą

4.1.3. Testavimo procedūros ir rezultatai

Modulių testavimą galima atlikti ir naudojantis specialiai testavimui skirta programa. Posistemės PĮ nėra sudėtinga, todėl testuotojas fiziškai be jokios specialios PĮ pagalbos perskaito analizuojamus rezultatus, o po to palygina savo išvadas su programos pateiktais rezultatais. Testavimui buvo paruošti testinių duomenų rinkiniai, teoriškai paskaičiuoti atsakymai. Pradiniai testavimo duomenys saugomi tekstiniuose failuose, o rezultatai išvedami į ekraną (arba į failą). Testavimo rezultatams saugoti naudojami Microsoft Excel ar Access failai. Buvo atlikti šie PĮ testai:

- struktūrinis pavienių modulių testavimas:
- paduodami įėjimo (testiniai) duomenys;
- stebimi modulių išėjimų duomenys;
- išėjimo duomenys palyginami su laukiamais (rankiniu būdu paskaičiuotais) rezultatais.
- vartotojo sąsajos komponentų testavimas:
- parametrų įvedimas iš klaviatūros;
- duomenų importas į .mdb formato failus ir importuotų duomenų palyginimas su šaltinio duomenimis;
- valdymo komponentų testavimas:
- duomenų nuskaitymas;
- atskirų meniu punktų funkcionalumas.
- integracinis testavimas – testuojami programos servais ir aukštesnio lygio vartotojo sąsajos komponentai. Testo metu buvo tikrinamas bendras programos funkcionalumas, t.y. Baltpers lentelių prijungimo mechanizmas, duomenų importas ir eksportas, programos veikimas iškviečiant formas, peržiūrint ir spausdinant ataskaitas, programinių objektų išsaugojimo ir modifikavimo galimybės.

- virtotojo (viešas) testavimas. Posistemės virtotojai testavo su personalo duomenimis realioje darbo aplinkoje. Virtotojai rašo ataskaitas apie surastas klaidas ir pateikia pasiūlymus.

Testinių rinkinių variantai ir testavimo rezultatai pateikti 3 priede.

4.2. Programinės įrangos diegimas

Pirmą kartą vykdant programą reikia ją įdiegti. Kadangi PĮ yra realizuota kaip Windows programa, todėl PĮ reikia įdiegti kiekviename virtotojo darbo vietos kompiuteryje. Reikia nukopijuoti paleidžiamuosius sistemos failus (bylas) į virtotojo sukurtą katalogą ir sukurti šaukinį (nuorodą) pagrindinei paleidžiamajai bylai. Programos pakete (kataloge „POSISTEMĖ“) yra programos failas, nuoroda į programos failą, atsarginė programos kopija ir informacijos apie MS Access darbo grupę failas. Nuorodą gali susikurti ir pats virtotojas. Pilnas PĮ diegimo aprašymas pateikiamas virtotojo instrukcijoje (5 priedas).

Tobulinant Posistemę galima realizuoti ir web sprendimą – PĮ patalpinti web puslapyje ir informuoti virtotojus, kad programą galima paleisti surenkant web adresą. Paleidus programą ji įsidiegtų virtotojo kompiuteryje, o paleidus sekantį kartą tikrintų ar nėra programos atnaujinimo ir jei taip, tai automatiškai atnaujintų PĮ.

IŠVADOS

1. Darbo tikslas įvykdytas – parengtas personalo duomenų analizės ir kontrolės posistemės programinės įrangos projektas ir sukurta programinė įranga, padėsianti spręsti vartotojui iškilusias problemas.

2. Atlikus KAS personalo apskaitos srities analizę, buvo įvertinti vartotojų poreikiai, nustatyti programinei įrangai keliami vartotojo reikalavimai, pasirinktos programavimo priemonės.

3. Įvykdytas projekto uždavinys – panaudojant Access aplinką ir Visual Basic for Applications programavimo priemones sukurta nesunkiai įsisavinama, nebrangi PI, leidžianti vartotojui atlikti Baltpers sistemoje esančių personalo duomenų analizę ir patikrinimą. Funkcijoms programuoti panaudotos Access terpėje integruotos VBA kalbos priemonės leido mažiausiomis sąnaudomis įgyvendinti projekto specifikaciją.

4. Atlikta projekto praktinė realizacija - sukurta personalo duomenų analizės ir kontrolės posistemė, kuria gali naudotis KAS personalo srities specialistai, taip pat ir kitose krašto apsaugos sistemos veiklos srityse dirbantys ir savo užduotims atlikti personalo duomenis naudojantys finansų, logistikos ir pan. specialistai.

5. Sukurta Posistemė suteikia vartotojams galimybę pagal atitinkamai parinktus tikrinimo parametrus tikrinti personalo asmens kodo teisingumą, formuoti hierarchinį KAS organizacijos pavadinimą, skaičiuoti personalo darbo stažą, analizuoti ir tikrinti etatų užimtumą. Sukurtas programinis produktas naudingas vartotojui, nes mažina personalo specialistų rankinio darbo apimtį, taupo jų darbo laiką ir teikia tikslesnius darbo rezultatus.

6. Atliktas Posistemės testavimas padėjo rasti ir pašalinti trūkumus bei silpnąsias vietas. Teigiami pilnai integruotos programų sistemos testavimo rezultatai leidžia pradėti eksploatuoti Posistemį.

7. Parengta Posistemės vartotojo dokumentacija – funkcinis aprašymas, įdiegimo dokumentas, ir darbo su Posisteme vadovas.

8. Numatytos galimybės tobulinti Posistemę, papildant ją naujomis funkcijomis.

9. Numatyta Posistemės bandomoji eksploatacija KAS personalo specialistų darbo vietose.

NAUDOTA LITERATŪRA

- [1] Adomavičius J., Pečiukonienė M., Sekliuckis V. Informatika 2. Algoritmai ir jų diegimas. Mokomoji knyga. – Kaunas, Technologija, 2001. – 82 p.
- [2] Gudas S., Žobakas T. Organizacijų veiklos procesų gyvavimo ciklų modeliai. 1999. [interaktyvus]. [žiūrėta 2005-12-15], prieiga per internetą: <http://www.leidykla.vu.lt/inetleid/inf-mok/10/str7.html>
- [3] Gulbinienė E., Petrauskienė J. Duomenų bazių valdymo sistema MS Access. Mokomoji knyga. – Šiauliai, Šiaurė, 2004.– 149 p.
- [4] *How to draw Data Flow Diagrams* [interaktyvus]. [žiūrėta 2005-12-15], prieiga per internetą: <http://www.smartdraw.com/resources/centers/software/>.
- [5] Orvis Dž. Viljam. Visual Basic for Applications by example. – Maskva, Kijevas, Binom, 1995.– 510 p.
- [6] Ostreika A. Programavimo Visual Basic pagrindai. – Kaunas, Technologija, 2003. – 228 p.
- [7] Rumbaugh J., Blaha M., Premerlani W., Eddy F., Lorenzen W. Object – Oriented Modeling and Design. - Prentice – Hall, 1991. – 500 p.
- [8] Scott Barker F. Access 2002 Power Programming. – Indianapolis, USA, 2002.– 991 p.
- [9] Sekliuckis V., Gudas S., Garšva G. Informacijos sistemos ir duomenų bazės. – Kaunas, Technologija, 2003. – 338 p.
- [10] Surma E., Nogis R. Objektiškai orientuoti kompiuterinio projektavimo esiniai. – Vilnius, Technika, 1998. – 75 p.
- [11] Vidžiūnas A., Marčiulygienė R. Access XP: taikomųjų duomenų bazių projektavimo pagrindai. - K., Smaltija, 2003.– 318 p.
- [12] *Visual Basic for MS Access* [interaktyvus]. [žiūrėta 2005-12-22], prieiga per internetą: <http://www.functionx.com/vbaccess/>
- [13] Williams S., Kindel C. *The Component Object Model: A Technical Overview*. [žiūrėta 2005-12-30], Microsoft Corporation, prieiga per internetą: http://msdn.microsoft.com/library/en-us/dncomg/html/msdn_comppr.asp

TERMINŲ ŽODYNAS

Pavadinimas	Paaiškinimas
Aplikacija (angl. Applications programme)	taikomoji programinė įranga
„Baltos dėžės“ principas	objektų kūrimo principas, pagal kurį objektas kuriamas iš gerai žinomų komponentų
Baltpers	KAS personalo registracijos, šaukimo, tarnybos ir mobilizacinio rezervo apskaitos informacinė sistema
Darbas	magistrinis darbas
Dokumentas	magistrinio darbo aprašas
Failas	įvardyta visų duoto tipo ar analogišku būdu sudarytų įrašų visuma
Funkcija	vykdo programoje tam tikrą užduotį ir rezultatą perduoda programai
Kodas	programos tekstas kompiuterio veiksmams valdyti
Makrokomanda (angl. Macros)	automatizuotas tam tikrų komandų sekos įvykdymas vieno klavišo ar mygtuko paspaudimu
Meniu	vartotojo iškviečiama, naudojama ir vykdoma operacijų ir funkcijų sistema
Modulis	programiniai komponentai sudaromi, norint apjungti Access komandas bei makrokomandas su Microsoft VB programomis
Projektas	programinės įrangos projektas
PĮ testavimas	programos darbo rezultato teisingumo patikrinimas su žinomais testiniais duomenimis
Specifikacija	sistemos funkcionalumo aprašymas formaliais metodais
Teisingumo lentelė	lentelė, nurodanti sąryšį tarp įėjimo ir išėjimo duomenų aibių
Visual Basic	objektinio programavimo kalba programoms rengti OS Microsoft Windows