



KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
KOMPIUTERIŲ KATEDRA

Orudž Alijev

Gaminio savikainos apskaičiavimo informacinė sistema

Magistro darbas

Darbo vadovas

doc.dr. E.Kazanavičius

Konsultantas

A. Žvironas

Kaunas
2004

TURINYS

1.	PRATARMĖ.....	4
2.	ĮVADAS	5
3.	ANALITINĖ DALIS	8
3.1	Uždavinio formulavimas.....	8
3.2	Uždavinio aplinkos analizė.....	8
3.3	Informacijos sistema, kūrimo būdai ir etapai.....	12
3.3.1	Informacijos sistema	12
3.3.2	Architektūra grindžiamas IS projektavimas	12
3.3.3	CASE priemonių analizė	13
3.4	Uždavinio analizės rezultatai	17
4.	PROGRAMINĖS ĮRANGOS PROJEKTAS.....	18
4.1	Reikalavimų projektuojamai sistemai specifikuojimas.....	18
4.1.1	Įvadinė dalis	18
4.1.2	Sistemos funkciniai reikalavimai	19
4.1.3	Nefunkciniai reikalavimai.....	22
4.1.4	Prototipai.....	23
4.1.5	Reikalavimai sąsajai	24
4.1.6	Projekto apribojimai	24
4.1.7	Duomenų šroutai	24
4.2	Duomenų struktūra	27
4.3	Sistemos architektūra	33
4.3.1	Sistemos sudedamosios dalys	33
4.3.2	Vartotojo sąsajos architektūra.....	35
4.4	Programinių modulių specifikacija.....	38
4.5	Testavimo medžiaga	40
4.5.1	Testavimo strategija.....	40
4.5.2	Testavimo planas	40
4.5.3	Testavimo eiga.....	42
5.	VARTOTOJO DOKUMENTACIJA.....	44
5.1	Sistemos funkcinis aprašymas	44
5.2	Sistemos vadovas.....	45
5.2.1	Sistemos struktūra.....	45
5.2.2	Vartotojo atmintinė.....	46
5.2.2.1	Bendri darbo su sistema aspektai.....	46
5.2.2.2	Pagrindinis vartotojo langas.....	48
5.2.2.3	Duomenų bazė	49
5.2.2.4	Rezultatai	58
5.2.2.5	Grafika	63
5.2.2.6	Ataskaitos.....	67
5.2.2.7	Pagalba.....	69
5.2.2.8	Sistemos priežiūra.....	70
5.3	Sistemos instaliavimo dokumentas.....	71
5.4	Sistemos administratoriaus vadovas	72
5.4.1	Sistemos priežiūra.....	72
5.4.2	Vartotojų ir jų grupių teisių tvarkymas.....	74
5.4.2.1	Sistemoje realizuotos grupės ir jų aprašymas	74
5.4.2.2	Grupių ir vartotojų tvarkymas.....	76
5.4.3	Ryšys su kitomis sistemomis	76

6.	KOKYBĖS VERTINIMAS	77
6.1	Funkcionalumo faktoriai	77
6.2	Mobilumo faktoriai	78
6.3	Kokybės faktoriai, susiję su tolesniu sistemos vystymusi	78
7.	IŠVADOS	80
8.	LITERATŪRA	81
9.	SUMMARY	83
10.	TERMINŲ IR SANTRUMPŲ ŽODYNAS	84
11.	PRIEDAI	85

1. PRATARMĖ

Nepaisant visų pastangų, programinės įrangos krizė didėja. Problema yra ne techninė įranga (*angl. hardware*) - kompiuterių greitis ir galingumas auga fenomenaliai, bet programinės įrangos kūrimo procesas. Sukurti programas, kurios išnaudoja kompiuterių galimybes yra didesnė problema, kaip geresnių kompiuterių kūrimas.

Šiuolaikinė CASE sistema iš tiesų gali sukurti išbaigtą, tinkančią vartojimui programų sistemą pagal diagramas ir kitą projektavimui reikalingą informaciją. CASE priemonė – tai integruota informacinės sistemos kūrimo aplinka.

Šia sistema projektuojama CASE sistemos **ProVision Workbench** (Proforma Corporation) paketų, kuris visiškai apima visą informacinės sistemos gyvavimo ciklą (IS GC). Pagrindinis šaltinis generuojamiems vartotojo sąsajos langų prototipams - biznio objektų modelyje apibrėžiami objektai, jų metodai, atributai bei ryšiai tarp objektų.

ProVision Workbench paketas leidžia naudoti sukurtų objektų duomenis reliacinių duomenų bazių valdymo sistemose (DBVS), tokiose kaip Oracle, Visual Basic, Access ir kt., kuriose gali būti atliekamos informacinės sistemos funkcijos. DDL (*angl. data definition language*) interpretuoja SQL CREATE sakinius į lenteles, kurios gali būti perskaitomos kokios nors vartotojo reliacinės duomenų bazės.

Pagrindinis **ProVision Workbench** paketo privalumas yra tai, kad IS kūrimo gyvavimo ciklas sudaroma pagal strategiją “nuo sąvokos iki programos kodo”.

2. ĮVADAS

Išėjimas į tarptautinę rinką, ekonominių ryšių su užsienio šalimis plėtojimas susijęs su esminiu produkcijos kokybės ir techninio lygio kėlimu. Tai, savo ruožtu, reikalauja ir atitinkamo gamybos techninio lygio, nes produkcijos kaina tarptautinėje rinkoje lemia šios produkcijos maksimalų galimos gamybos lygį. Kuo mažesnės gamybos sąnaudos, gaminio kaina ir savikaina, tuo didesnė galimybė įsitvirtinti tarptautinėje rinkoje.

Todėl iškyla būtinumas suderinti veiklos reinžinerijos planavimą, su IT reinžinerija, naudojant modernias, našias informacines technologijas. Taigi reikia naudoti kompiuterizuotus finansinius– ekonominius skaičiavimus, nes:

- trumpėja projektavimo trukmė greito informacijos pasikeitimo, operatyvios projekto būklės kontrolės dėka,
- atpinga projektavimas, nes išvengiama daugelio klaidų, mažėja papildomi skaičiavimai ir išlaidos,
- pagerėja projekto kokybė, nes pateikiami keli variantai.

Paskutinė naujovė struktūriniame programavime yra CASE - kompiuterizuota programų inžinerija. CASE priemonių pagalba kompiuteris valdo funkcinės dekompozicijos procesą, analizuodamas grafines schemas (diagramas), kuriose nurodomos paprogramės ir jų tarpusavio sąveikos. Šiuolaikinė CASE sistema iš tiesų gali sukurti išbaigtą, tinkančią vartojimui programų sistemą pagal diagramas ir kitą projektavimui reikalingą informaciją.

Nepaisant visų pastangų, programinės įrangos krizė didėja. Problema yra ne techninė įranga (*hardware*) - kompiuterių greitis ir galingumas auga fenomenaliai, bet programinės įrangos kūrimo procesas. Sukurti programas, kurios išnaudoja kompiuterių galimybes yra didesnė problema, kaip geresnių kompiuterių kūrimas.

Struktūrinis programavimas labai pagerino programinės įrangos kokybę, tačiau išaiškėjo ir jo trūkumai. Vienas iš rimčiausių trūkumų - neįmanoma užbaigti viso programų paketo projekto anksčiau, negu ji yra įdiegiama. Tik įdiegus programinį produktą paaiškėja, kad problema buvo blogai suskirstyta į modulius, todėl tenka perprogramuoti viską iš naujo. Kuo didesnė sistema, tuo didesnė tikimybė, kad teks keisti programų sistemos struktūrą.

Šiuo metu programavime ir informacinių sistemų (IS) projektavime vyrauja objektiškai orientuotas (OO) požiūris, metodai ir priemonės. Objektinis požiūris pakeitė tradiciniu laikytą struktūrinį- funkcinį požiūrį. OO technologijoje procedūros yra vadinamos metodais, o duomenų elementai vadinami kintamaisiais, kadangi jų reikšmės kinta laikui

bėgant. Vienas pirmųjų OO požiūrio metodas buvo Objektinė modeliavimo technologija (OMT), kurios autorius - James Rumbaugh.

Objektiškai orientuota technologija remiasi trimis pagrindinėmis sąvokomis:

- objektai (*objects*),
- pranešimai (*messages*),
- klasės (*classes*).

Objektų modelis susideda iš klasių diagramų (*class diagrams*) ir objektų diagramų (*instance diagrams*). Klasių diagrama yra grafas, kurio viršūnės yra objektų klasės, o lankai - santykiai tarp objektų klasių. Pagrindinis objektiškai orientuotų sistemų skirtumas yra tai, kad objektinės sistemos remiasi objekto sąvoka, kuri reiškia esybę, susietą su operacijomis, kurias apima šis objektas. Objektas susideda iš atributų ir metodų.

Objekto atributai yra analogiški esybių atributams, saugomiems reliacinėse duomenų bazėse, ir aprašo objektą struktūros aspektu. Esminis bruožas yra tai, kad objektai objektinėje duomenų bazėje susieti su žymiai didesniu informacijos kiekiu, kaip esybė reliacinėje DB, ir todėl yra sudėtingesni informaciniai vienetai. Objektiniame DB projektavime reikia suderinti IS struktūrinės savybės ir elgesio ypatybes. Akivaizdu, kad objektiškai orientuotos IS projektavime reikia griežtesnių metodų, kaip reliacinės DB pagrindu kuriamų IS projektavimas.

Šiuo metu Lietuvos įmonėse yra plačiai naudojamos kompiuterizuotos informacinės sistemos. Tarp jų gan gerai žinomos valdymo sistemos „Stekas – apskaita“, „Praima“, „Konto“ ir daugelis kitų. Dauguma jų gana universalios ir tinka įvairiems įmonės valdymo uždaviniams spręsti. Šios sistemos pasižymi dideliu sudėtingumu bei aukšta kaina. Įmonės darbuotojų apmokymui dirbti šiomis sistemomis tenka skirti gana daug lėšų ir laiko.

Todėl reikalingos paprastesnės ir pigesnės informacinės sistemos. Būtent dėl šios priežasties šio darbo tikslas yra sudaryti ne didelės ir ne sudėtingos gaminio savikainos apskaičiavimo informacinės sistemos programinės įrangos projektą, kuris apimtų pagrindinius didžiųjų sistemų privalumus bei vengtų jų trūkumų.

Šia sistema projektuojama CASE sistemos **ProVision Workbench** (Proforma Corporation) paketų, kuris visiškai apima visą IS GC. **ProVision Workbench** paketo privalumas yra tai, kad IS kūrimo gyvavimo ciklas sudarytas pagal strategiją „nuo sąvokos iki programos kodo“.

ProVision Workbench paketas leidžia naudoti sukurtų objektų duomenis reliacinių duomenų bazių valdymo sistemose (DBVS), tokiose kaip Oracle, Visual Basic, Access ir kt., kuriose gali būti atliekamos IS-os funkcijos. DDL (*data definition language*) interpretuoja

SQL CREATE sakinius į lenteles, kurios gali būti perskaitomos kokios nors vartotojo reliacinės duomenų bazės.

Programinei įrangai buvo paruoštas projektas:

- Atlikta išsami esamos programinės įrangos analizė, išskirti jos privalumai ir trūkumai.
- Išanalizuoti vartotojo reikalavimus, įvertinti sistemos vartotojo poreikius, patirtį ir sugebėjimo faktorius.
- Išaiškinti sistemai keliamus funkcinis ir ne funkcinis reikalavimus; sistema turi dirbti patikimai ir saugiai.
- Rizikos šaltiniai ir jų išvengimo priemonės išanalizuojami pradinėje projektavimo stadijoje.
- Sukurta vartotojo sąsaja. Jai sukurti naudoti prototipai ir esamos programinės įrangos analizės medžiaga.
- Sudarytas projekto atlikimo grafikas.
- Sudarytas sistemos testavimo planas ir išaiškinti stebėjimo ir atsiskaitymo mechanizmai.

Vartotojo reikalavimai projekte yra apibrėžti, naudojant natūralią kalbą, lenteles ir diagramas, vertinant sistemos vartotojo poreikius, patirtį ir sugebėjimus. Vartotojas turi turėti galimybę ieškoti arba visus pradinius duomenų bazės rinkinius, arba išsirinkti poaibį iš jų.

3. ANALITINĖ DALIS

3.1 Uždavinio formulavimas

Pagrindinis darbo uždavinys: Sukurti informacinę sistemą, kuri apskaičiuotų gaminio savikainą.

3.2 Uždavinio aplinkos analizė

Egzistuoja pakankamai daug programinės įrangos paketų, atliekančių visas funkcijas, reikalingas užduotyje nurodytoms bendrovės veiklos sritims automatizuoti. Todėl visų pirma išanalizuosime jau esamą programinę įrangą, išstirsime jos galimybes, nustatysime privalumus ir trūkumus. Analizės medžiagą panaudosime savos sistemos kūrimui.

Žemiau pateikiami kelių panašaus tipo programinių paketų trumpi aprašymai, parodydami pagrindinius jų privalumus ir trūkumus.

"Rivilė - SOLO"

Gamintojas: UAB „Rivilė”.

Operacinė sistema: Win98, Win NT 4.0 (Regional Setting – Lithuania, Multilanguage support – Baltic) su Service Pack 4 arba aukštesne versija, Windows 2000 Server arba Profesional.

Palaikomi bylų formatai: ataskaitos perkeliamos į *.xls bei tekstinius formatus.

Aprašymas:

Sistema **"Rivilė - SOLO"** - tai apskaitos sistema smulkiąjam verslui, apimanti buhalterinę, finansinę ir verslo valdymo apskaitą. Ji turi įprastas grafinės Windows terpės sąsajos galimybes, o Kliento/Serverio (Client/Server-SQL) technologija bei transakcijų mechanizmas užtikrina duomenų patikimumą ir stabilumą. Sistema **"Rivilė - SOLO"** pritaikyta Lietuvos vartotojui. Patogi vartotojo sąsaja leidžia greitai ir patogiai dirbti su šia programa [8].

Sistemos privalumai ir galimybės:

- Patogi vartotojo sąsaja.

Programa jungia visus pagrindinius apskaitos modulius: didžiąją knygą, klientų skolas, pirkimus, pardavimus, atsiskaitymus, atsargų operacijas, gamybą ir operacijas su ilgalaikiu turtu.

- Galima komplektuoti ir su atlyginimų skaičiavimo moduliū.

Sistemos trūkumai:

- Sudėtinga sistema.
- Didelė sistemos kaina.

„STEKAS – apskaita”

Gamintojas: UAB „Stekas”.

Operacinė sistema: Win95, Win98, WinNT, Win2000.

Palaikomi bylų formatai: „STEKAS – apskaita” (*.pdb, ataskaitos perkeliomos į *.xls).

Aprašymas:

Tai integruota buhalterinės apskaitos programa skirta vesti įmonės pilnai buhalterinei apskaitai. Sistema leidžia pildyti pagrindinius buhalterinės apskaitos pirminius dokumentus ir gauti įvairias finansinių rodiklių ataskaitas, reikalingas įmonės ūkinei-finansinei veiklai atspindėti. Programa suskirstyta į atskirus modulius (didžioji knyga, pirkimai, pardavimai, atsargos, ilgalaikis turtas, gamyba, sutartys) pagal darbo, vartotojo poreikių pobūdį. Moduliai yra integruoti, t.y. viename jų įvedus informaciją, ji automatiškai patenka į kitus. Jokio dokumento nereikia vesti po kelis kartus [1].

Sistemos privalumai ir galimybės:

- Suvesti pirkimo, pardavimo sąskaitas-faktūras, apmokėjimus.
- Priklausomai nuo to ką perkate, parduodate, kam ir iš kur mokate pinigus, kompiuteris sukuria atitinkamą sąskaitų korespondenciją.
- Gauti bet kuriuo metu ir bet kurio laikotarpio įmonės balansą, pelno nuostolio ataskaitą įvairius finansinius rodiklius.
- Žodynuose įvedama pastovi įvairaus pobūdžio informacija. Ji įvedama vieną kartą ir pastoviai naudojama vedant pirminius dokumentus.
- Visa informacija į programą vedama operacijų pavidalu.
- Visą operacijų pavidalu suvestą apskaitos informaciją, įvairiais pjūviais galima pamatyti ataskaitų pagalba.

Sistemos trūkumai:

- Sudėtinga sistema.
- Didelė sistemos kaina.

KONTO

Gamintojas: UAB „Labbis”.

Operacinė sistema: Win95, Win98, WinNT 4.0, Win2000.

Palaikomi bylų formatai: KONTO (ataskaitos perkeliamos į *.xls bei tekstinius formatus).

Aprašymas:

Tai kompiuterizuota buhalterinės apskaitos programa, pritaikyta darbui Windows operacinėje aplinkoje. Ji skirta biudžetinių įstaigų, akcinių bendrovių bei individualių įmonių apskaitai vesti. Gali būti taikomas FIFO arba LIFO metodai. Patogi vartotojo sąsaja leidžia greitai ir patogiai dirbti su šia programa. Patogus navigacijos ekranas prekės judėjimui, dinamikai ir jos būsenai stebėti bei analizuoti [2].

Sistemos privalumai ir galimybės:

- Pilna apskaita iki balanso.
- Yra atsargų modulis.
- Skirta tiek didmeninei, tiek mažmeninei prekybai.
- Pritaikyta ir individualių įmonių apskaitai.
- Neribojamas įmonių ar jų vidinių padalinių skaičius.
- Darbas viename AK ir jų tinkle.

Sistemos trūkumai:

- Sudėtinga sistema.
- Didelė sistemos kaina.

Resvera

Gamintojas: UAB „Resvera”.

Operacinė sistema: Win95, Win98.

Palaikomi bylų formatai: ataskaitos perkeliamos į *.xls.

Aprašymas:

Tai kompiuterizuota buhalterinės apskaitos programa, pritaikyta darbui Windows operacinėje aplinkoje. Pilnas buhalterinės apskaitos iki balanso programų paketas [4].

Sistemos privalumai ir galimybės:

- Galima skaičiuoti prekių savikainą.
- Galima spausdinti sąskaitas faktūras ir važtaraščius.
- Galima daryti tarptautinius ir vietinius pavedimus.
- Galima gauti visas finansines ataskaitas (apyvartos žiniaraščius).
- Galima matyti visus skolininkus.
- Galima gauti įmonės balansą.
- Galima matyti prekių pardavimų, pirkimų, mokėjimų ataskaitas grafiškai.

Sistemos trūkumai:

- Sudėtinga sistema.
- Didelė sistemos kaina.

Centas

Gamintojas: UAB "EDLONTA"

Operacinė sistema: Win95, Win98.

Palaikomi bylų formatai: ataskaitos perkeliamos į *.xls bei *.doc.

Aprašymas:

Tai materialinių vertybių ir buhalterinės apskaitos programa, turinti materialinių vertybių apskaitos, gamybos, darbo užmokesčio, ilgalaikio turto modulius. Naudojami įvairūs žinynai, įvairios peržiūros funkcijos, mokėjimų atsiskaitymo operacijos ir kt. Patogi vartotojo sąsaja leidžia greitai ir patogiai dirbti su šia programa. [9].

Sistemos privalumai ir galimybės:

- Pirkimai - pardavimai. Atsiskaitymų apskaita.
- Sandėlių apskaita. Neribotas sandėlių skaičius. Vidiniai judėjimai.
- Paslaugos. Suteiktų paslaugų registravimas.
- Sąskaitų planas. Operacijų žurnalas. Korespondencijų ataskaitos. Didžioji knyga. Balansas.
- Darbo užmokesčio apskaita.
- Ilgalaikio turto apskaita
- Kasos ir banko apskaita.

Sistemos trūkumai:

- Sudėtinga sistema.
- Didelė sistemos kaina.

Dauguma minėtų programų netinkamos, nes nevysiškai atitinka vartotojų poreikius. Šios sistemos pasižymi dideliu sudėtingumu bei aukšta kaina. Įmonės darbuotojų apmokymui dirbti šiomis sistemomis tenka skirti gana daug lėšų ir laiko.

Todėl atsiranda poreikis paprastesnėms ir pigesnėms informacinėms sistemoms atsirasti. Būtent dėl šios priežasties šio darbo tikslas yra sudaryti nedidelės ir nesudėtingos gaminio savikainos apskaičiavimo informacinės sistemos programinės įrangos projektą, kuris apimtų pagrindinius didžiųjų sistemų privalumus bei vengtų jų trūkumų.

3.3 Informacijos sistema, kūrimo būdai ir etapai

3.3.1 Informacijos sistema

Informacijos sistema (IS) surenka, apdoroja, saugo, analizuoja ir paskirsto informaciją, turinčią konkrečią paskirtį organizacijos veikloje.

Informacijos sistemos apdoroja įeigą (duomenis) ir suformuoja išeigą (ataskaitas, suvestines), kuriuos nusiunčia vartotojui arba kitai sistemai. Informacijos sistema susideda iš techninės įrangos, programinės įrangos duomenų bazės ir telekomunikacijų IS sudėtyje taip pat yra ir atitinkamos kvalifikacijos personalas. Tradicinis reikalavimų apibrėžimas: “reikalavimas turi apibrėžti **ka** sistema turi atlikti, nenurodant **kaip** tai turi vykti”. Reikalavimų dokumentas turi būti sudarytas, atsižvelgiant į nagrinėjamus klausimus.

Reikalavimai skirstomi į tokias stambias grupes:

- Funkciniai reikalavimai - specifikuoja kiekvienos sąsajos atlikimo savybes, apibūdinančias įeigos procesą ir išeigos procesą ;
- Nefunkciniai reikalavimai specifikuoja numatomų (projektuojamų) sistemos funkcijų savybes – apibrėžia kokybines funkcijų charakteristikas. Todėl nefunkciniai reikalavimai dar vadinami kokybės reikalavimais;
- Vadybiniai reikalavimai - tarpinė riba tarp reikalavimų sistemai ir sutarties sudarymo sąlygų. Planuojant naujas IS ir organizuojant jų kūrimo darbus, nepakanka funkcinį ir nefunkcinį reikalavimų.
- Bendri reikalavimai sistemai - produkto apibrėžimas, vartotojų apibūdinimas, vartotojų tikslai ir reikalavimai, vartotojų problemų apibūdinimas, analogiškų sistemų apibūdinimas,

Reikalavimams specifikuoti dažniausiai naudojami grafiniai modeliai yra kontekstinės diagramos, duomenų srautų diagramos (angl. Data Flow Diagrams (DFD)), esybių - ryšių diagrama (angl. Entity Relationship Diagrams (ERD)), įvykių-funkcijų sąrašai, objektinio modeliavimo kalbos UML modelis *Use Case*, kurį vadiname informacinių poreikių arba taikomųjų uždavinių modeliu [11].

3.3.2 Architektūra grindžiamas IS projektavimas

Viena iš pažangiausių veiklos procesų ir taikomųjų programų integravimo metodologijų vadinama “architektūriniu modeliavimu” ar “architektūra grindžiamas IS projektavimas”. Veiklos informacinė architektūra apima bendros sistemos struktūros, sistemos komponentų, loginių jų ryšių ir išoriškai matomų savybių modeliavimą (projektavimą). Organizacijų informacinės architektūros modeliavimas tiesiogiai skirtas

informacijos sistemų, atitinkančių realius veiklos poreikius, projektavimo ir realizavimo metodams vystyti.

Komponentinis projektavimas teoriškai turi daug privalumų, iš kurių svarbiausias – pakartotino komponentų panaudojimo galimybė. Dėl šios savybės padidėja produktyvumas, palaikymo ir modifikavimo galimybės, o lygiagrečiai sumažėja projekto kūrimo ciklas ir kaštai.

IS projekto lygmens komponentai projektuojami pagal modeliu pagrįstą (*model-driven*) projektavimo paradigmą, kurioje komponentai paveldi aprašus iš veiklos proceso modelio. Aptariamas metodas aprašo architektūrinio IS projektavimo etapą, kuriame identifikuojami IS projekto komponentai ir jų sąsajos [19].

IS projekto komponentai yra skirstomi į:

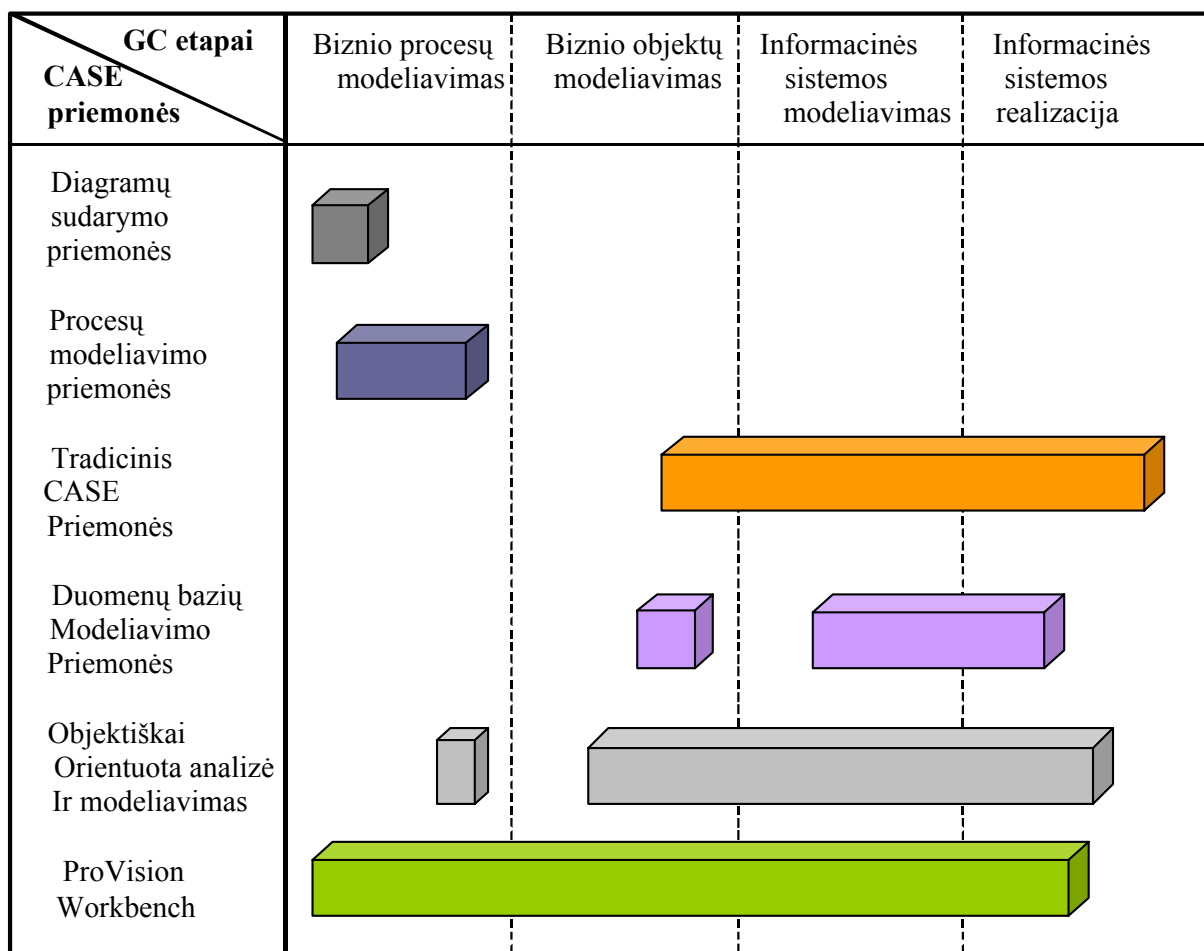
- vartotojo sąsajos komponentus (menu, ekrano formos, ataskaitos),
- duomenų komponentus (duomenų bazėse ar duomenų saugykloje talpinami informacijos vienetai),
- funkcinius komponentus (skaičiavimai ir taikomųjų uždavinių logika).

Organizacijos informacijos sistemos komponentėms ir sąsajoms tarp jų identifikuoti siūlomas naujas grafinė žymėjimas – komponentinis sistemos modelis. Šis modelis apjungia veiklos informacinės architektūros (VIA) modelio ir darbų sekos modelio savybes.

3.3.3 CASE priemonių analizė

CASE (*computer aided software engineering*) - tai priemonė, kuri padeda programinės įrangos inžinieriams kurti, palaikyti, vystyti programinę įrangą. Kai kurios CASE priemonės yra skirtos tik diagramų braižymui. Kitos sutelktos į programinės įrangos realizavimą, kodo generavimą. Tačiau patogiausia yra dirbti su viena organizacijos veiklos modeliavimo priemone, kuri apimtų visą informacinės sistemos (IS) kūrimo gyvavimo ciklą.

Žemiau esančiame 1 paveiksle pateikiami daugelio tipų modeliavimo priemonių palyginimai;



Ipav. CASE priemonių analizė

Iš diagramos matome, kad palyginus su daugelio tipų modeliavimo priemonėmis, ProVision Workbench paketas plačiausiai apima pagrindinius gyvavimo ciklo etapus:

- Biznio procesų modeliavimas ir reinžinerija (*BPR*);
- Biznio objektų modeliavimas;
- Informacinės sistemos modeliavimas;
- Informacinės sistemos realizacija.

Diagramų sudarymo priemonės (*angl. diagraming tools*). Šios priemonės skirtos tikrai biznio diagramų braižymui. Jos negali sukurtų diagramų praplėsti, transformuoti į kitus formatus. Diagramų sudarymo priemonės atlieka tik nedidelę BPR etapo dalį. (Visio, Micrografx Floxcharter, Corel Flow ir kt.)

Procesų modeliavimo ir imitavimo priemonės (*angl. process modeling and simulation*). Tai priemonės su sudėtingais procesų imitatoriais. Kadangi šios priemonės yra per daug sudėtingos ir brangios biznio procesų modeliavimui, jos yra naudojamos statistikoje ir operacijų tyrinėjimuose. Daugelis šio tipo priemonių yra pagrįstos specializuota ir ribota

IDEF (*angl. Interactive development environment functions*) - interaktyvios vystymosi aplinkos funkcijos metodologija. (BPwin, Optima, Promodel ir kt.)

Tradicinės CASE priemonės (*angl. traditional CASE*), kurios apima informacinės sistemos modeliavimo, IS realizavimo etapus ir dalį biznio objektų modeliavimo etapo. Bet kadangi tradicinės CASE priemonės buvo kuriamos COBOL programos kodo pagrindu, o šiuo metu vyrauja kliento/serverio sistemos, jos jau nebenaudojamos, bet jų principai yra panaudojami kitose modeliavimo priemonėse. (ADW/Cool and IEF/Composer from Sterling Software ir kt.)

Duomenų bazių modeliavimo priemonės (*angl. database modeling*). Šios priemonės buvo kuriamos semantinių duomenų modeliavimui su griežtai pabrėžiamu duomenų bazių generavimu. Tačiau jos buvo kuriamos ne kaip organizacijos modeliavimo priemonės, todėl jas sunku perprojektuoti, kad būtų galima modeliuoti procesus. (Erwin, Power Designer, Silverrun ir kt.)

Objektiškai orientuota analizė ir modeliavimas (*angl. object oriented analysis and design (OO A&D)*). Šio tipo priemonės skirtos procesų dekompozicijai, sistemoms orientuotų taikomųjų uždavinių modeliavimui. Taip pat šiek tiek palaiko informacijos dokumentavimą, kuris reikalingas biznio modelių kūrimui ir biznio sistemų išvedimui iš šių modelių. (Rational, Select, Paradigm Plus ir kt.)

ProVisionWorkbench - tai organizacijos modeliavimo priemonė, kuri apjungia BPR ir OO A&D į vieną, integruotą modeliavimo priemonę. Paketas palaiko geresnes biznio modeliavimo galimybes, ypač unikalus biznio sąveikų modelis (*angl. business interaction model*) ir populiarios darbų sekos (*angl. workflow model*) ir biznio objektų (*angl. business object model*) modeliai. Tai leidžia apimti BPR ir biznio objektų modeliavimo etapus. Palaikant reliacinių lentelių modelį (*angl. relational model*), vartotojo sąsajos (*angl. user interface model*) ir kitų UML (*angl. unified modeling language*) modelių kūrimą, paketas apima visą informacinės sistemos modeliavimo etapą. Taip pat dėka galimybės generuoti programinius kodus, paketas patenka ir į realizacijos etapo dalį.

CASE paketų bendra problema yra bandymas išspręsti kaip pritaikyti integruotą "nuo sąvokos iki programos kodo" sprendimą biznio sistemos kūrimui. "Nuo sąvokos iki programos kodo" strategija iki šio laiko dar nebuvo realizuota jokiame pakete. Bet technologijos, analizės ir projektavimo metodai ir beveik nepriklausoma biznio procesų reinžinerijos (BPR) sritis yra vis daugiau vystomi, kad šią strategiją būtų įmanoma įgyvendinti. ProVision Workbench paketas, prieš tai įvertinęs trūkumus ankstesnių paketų, kurie bandė realizuoti strategiją, "nuo sąvokos iki programos kodo" idėją priartino prie realybės.

ProVision Workbench

Gamintojas: Proforma Corporation

Metododika:

- Booch
- Core
- Information Engineering
- Jacobson Use Cse
- Martin/Odell
- Object Thinking
- OMT
- Rummler-Brache
- Shlaer/Mellor
- UML

Modeliai:

- Tikslų modelis (*Goal*)
- Organizacijos modelis (*Organization*)
- Aplinkos modelis (*Location*)
- Procesų modelis (*Process*)
- Sistemos modelis (*System*)
- Įvykių modelis (*Event*)
- Biznio sąveikų modelis (*Bussiness interaction*)
- Darbų sekos modelis (*Workflow*)
- Panaudojimo atvejų modelis (*Use Case*)
- Objektų modelis (*Object*)
- Potipis modelis (*Subtype*)
- Būsenų modelis (*State*)
- Sąveikų modelis (*Interaction*)
- Metodų modelis (*Method*)

Kodo generavimas:

- C++,
- Coad
- Power Builder,
- Visual Basic,

- VisualAge Smalltalk,
- Erwin (SML).

Reikalavimai techninei ir programinei įrangai:

- Windows 95, Windows 98 ir Windows NT
- 486 arba Pentium tipo PC procesoriai,
- 32 Mb RAM atminties,
- Ne mažiau kaip 30 Mb HDD disko atminties paketo instaliavimui ir 50 Mb disko atminties darbui,
- VGA suderinamas ekranas.

3.4 Uždavinio analizės rezultatai

Dauguma jau sukurtų programinės įrangos paketų, iš vienos pusės, turi daugybę mūsų kompiuterizuojamai įmonei bereikalingų funkcijų. Iš kitos pusės, jau egzistuojančiose programose nėra kai kurių specifinių įmonei funkcijų. Sudėtinga sistema ir didelė sistemos kaina.

Dėl šių priežasčių yra nutarta suprojektuoti ir realizuoti savitą programinę įrangą, atliekančią visas uždavinyje nurodytas funkcijas. Tiek vartotojo sąsaja, tiek ir duomenų struktūros bus pritaikytos specifinėms įmonės funkcijos atlikti.

Informacinės sistemos kūrimui pasirenkame CASE paketą - **Pro Vision Workbench**

3.1. Pagrindinis šaltinis generuojamiems vartotojo sąsajos langų prototipams - biznio objektų modelyje apibrėžiami objektai, jų metodai, atributai bei ryšiai tarp objektų. Pro Vision Workbench 3.1 paketas leidžia naudoti sukurtų objektų duomenis reliacinių duomenų bazių valdymo sistemose (*DBVS*), tokiose kaip Oracle, Visual Basic, Access ir kt., kuriose gali būti atliekamos IS funkcijos ar panašiai.

4. PROGRAMINĖS ĮRANGOS PROJEKTAS

Pagrindiniai projekto tikslai yra šie:

- Sukurti nesunkiai įsisavinamą, nebrangią programinę įrangą, leidžiančią vartotojui apskaičiuoti gaminio savikainą.
- Įvertinti vartotojo poreikius, patirtį ir sugebėjimus.
- Sudaryti lengvai valdomą ir programuojamą, lankstų objektinį-programinį modelį, kuris įgalintų programuotoją greitai ir kokybiškai kurti bei modifikuoti programinę įrangą.

Projekto keliami uždaviniai yra šie:

- Išsiaiškinti vartotojo norus ir pageidavimus programinei įrangai.
- Išsiaiškinti projekto ypatumus.
- Išsiaiškinti reikalavimus naudojamai programavimo kalbai.
- Apibrėžti PĮ architektūrinį modelį.
- Apibrėžti PĮ objektinį modelį.
- Apibrėžti PĮ duomenų struktūrų modelį.
- Apibrėžti vartotojo sąsają.
- Nustatyti sistemos testavimo būdus.
- Atsižvelgiant į anksčiau minėtus punktus, sudaryti projekto realizavimo grafiką.

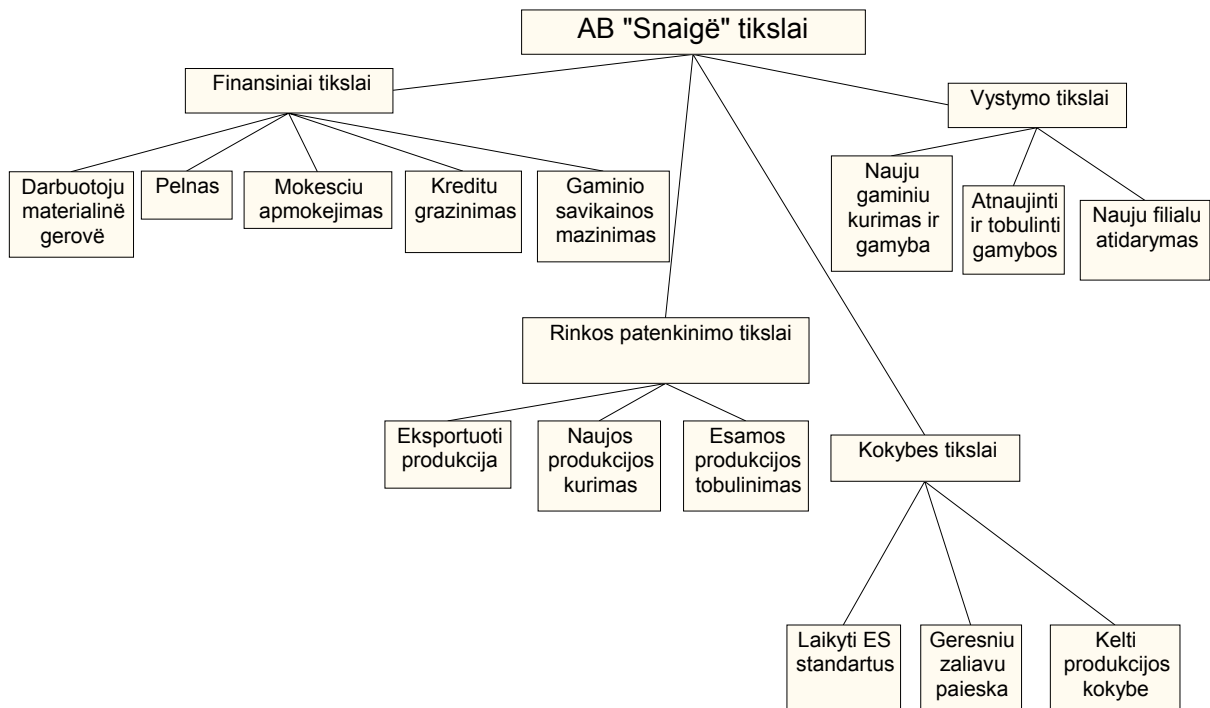
3.1 Reikalavimų projektuojamai sistemai specifikavimas.

4.1.1 Įvadinė dalis

Reikalavimų specifikavimo dokumento tikslas – parašyti reikalavimus kompiuterizuojamai informacijos sistemai.

Akcinė Bendrovė (AB) „Snaigė“, kaip ir kiekviena gamybinė įmonė turi savitus veiklos reikalavimus; tai bendrovė - serijiniu gamybiniu būdu gaminanti įvairių modelių „Snaigė“ šaldytuvus. Atsižvelgiant į tai, kad gamyba yra specifinė, reikalavimų specifikavimo išsiaiškinimui žemiau pateikiame AB „Snaigė“ tikslų modelį (*Goal Modeler*), sudarytą **ProVision Workbench** paketu.

Kaip matome (2 Pav.) gamybos tikslai suskirstyti į grupes, ir pogrupius; nors jie glaudžiai susieti tarp savęs.



2 Pav. AB „Snaigė” tikslų modelis (Goal Modeler)

Mes nagrinėsime grupės „Finansiniai tikslai” pogrupio „gaminio savikainos mažinimas”.

Numatomas produktas turi apskaičiuoti gaminio savikainą, kuri susideda iš materialinės sąnaudos, darbo užmokesčio, atsiskaitymų su „Sodra“, amortizacinių atskaitymų ir kitų išlaidų.

Programinio produkto vartotojas yra buhalterinės apskaitos skyrius.

Keliamiems reikalavimams pasiekti naudosime tokį planą:

- Vartotojų apklausa bei panašaus pobūdžio programinės įrangos privalumų ir trūkumų analizė.
- Nefunkcinių reikalavimų nustatymas.
- Prototipų kūrimas.

4.1.2 Sistemos funkciniai reikalavimai

Programinė įranga yra skirta jos vartotojui (mūsų atveju buhalterinės apskaitos skyriui), todėl pirmiausiai ir išsiaiškinsime jos poreikius.

Vartotojo poreikiai kuriamai programinei įrangai:

Vartotojas nori programos, kuri leistų jam apskaičiuoti gaminio savikainą, esant reikalui, redaguoti, papildyti arba ištrinti nereikalingus įrašus. Vartotojas taip pat nori peržiūrėti, atspausdinti suskaičiuotus rezultatus.

Gaminio savikainos skaičiavimo procesas:

Pradiniai duomenys:

- Materialinės sąnaudos:
 - Pagrindinės medžiagos
 - Komplektuojami gaminiai
 - Elektros energija
 - Kuras, vanduo.
- Darbo užmokestis:
 - Pagrindinių darbininkų
 - Pagalbinių darbininkų
 - Vadovų.
- Atsiskaitymai socialinio draudimui
- Amortizaciniai atskaitymai
- Kitos išlaidos

Gaminio savikaina = Materialinis sąnaudos + darbo užmokestis + atsiskaitymai

SODRAI + amortizaciniai atskaitymai + kitos išlaidos.

Žemiau detaliau aprašome įeities duomenų sudėtis:

Materialinės sąnaudos:

Pagrindinės medžiagos:

- Juodieji metalai,
- Spalvotieji metalai,
- Polimerai ir plastmasės,
- Cheminės medžiagos,
- Tara įpakavimui.

Komplektuojami gaminiai:

- kompresoriai,
- kapiliarai,
- šaldymo agentas,
- kondensatoriai,
- garintuvai.

Elektros energija:

- Jėginės elektros energijos poreikis,
- Apšvietimo elektros energijos poreikis.

Kuras, vanduo, suspaustas oras:

- Kuro poreikis,
- Vandens poreikis,
- Suspausto oro poreikis.

Darbo užmokestis:

Pagrindinių darbininkų:

- Pagrindinių darbininkų specialybė,
- Pagrindinių darbininkų kategorija,
- Pagrindinių darbininkų skaičius,
- Valandinis atlygis,
- Efektyvus darbo laiko fondas,
- Tarifinis darbo užmokestis,
- Pagrindinis darbo užmokestis,
- Papildomas darbo užmokestis.

Pagalbinių darbininkų:

- Pagalbinių darbininkų specialybė,
- Pagalbinių darbininkų kategorija,
- Pagalbinių darbininkų skaičius,
- Valandinis atlygis,
- Efektyvus darbo laiko fondas,
- Tarifinis darbo užmokestis,
- Pagrindinis darbo užmokestis,
- Papildomas darbo užmokestis.

Vadovų:

- Vadovų skaičius,
- Mėnesinis darbo užmokestis,
- Metinis darbo užmokestis.

Atsiskaitymai Sodrai:

- atskaičiuojama nuo darbo užmokesčio.

Amortizaciniai atskaitymai:

- Įrengimų nusidėvėjimas,
- Įrangos nusidėvėjimas,
- Gamybinio pastato nusidėvėjimas.

Kitos išlaidos:

- įvertinamos išlaidos nuo gamybinio kapitalo.

Vartotojo keliami uždaviniai kuriamai sistemai:

- Pagrindinis vartotojo sąsajos langas turi atspindėti esminę duomenų bazės struktūrą, t.y. duomenų bazės lenteles, užklausas, ataskaitas t.t.
- Projekte nenaudoti meniu komandos, nes tai papildomai apkrautų sąsają.
- Vartotojo sąsajos langus rengti naudojant valdymo, pav. CommandButton mygtukais.
- Atskirti duomenų bazės lenteles, užklausas, ataskaitas, grafikos elementus, vartotojo pagalbos langelius į atskiras grupes, naudojant pav. Frame objektus. Duomenų bazės lentelėse naudoti redagavimo, papildymo, veiksmų atšaukimo priemones. Vartotojas turi turėti galimybę ieškoti arba visus pradinius duomenų bazės rinkinius, arba išsirinkti iš jų reikalingus.
- Kiekvienam lentelių laukui (*field*) privaloma paskirti saugomų duomenų tipus (*Data Type*), lauko ilgį (*Field Size*), formatus (*Format*) t.t. Pav. lauko piniginės reikšmės vartotojui bus parodyta “100.000,00 Lt” formatu.
- Įvedamų duomenų korektiškumui tikrinti nurodoma speciali tikrinimo taisyklė (*Validation Rule*) ir pranešimo tekstas (*Validation Text*), išpėjantis vartotoją, kai bandoma įvesti neleistinus duomenis.
- Sistema turi leisti vartotojui ištaisyti klaidas.
- Pranešimų projektavimas yra labai svarbus, reikalinga standartinės klaidų pranešimo kortelės (*Message Box*).
- Sistema turi leisti vartotojui įvesti duomenis, naudojant standartinę kortelę informacijai įvesti (*InputBox*).
- Skaičiavimo rezultatai turi būti tikslūs, be klaidos.

4.1.3 Nefunkciniai reikalavimai

Išnagrinėjus panašaus tipo programinę įrangą bei jos reikalavimus sistemoms, planuojami programinės bei aparatinės įrangos reikalavimai būtų tokie:

Programine įranga:

- Duomenų bazių valdymo sistema (DB/VS): Microsoft Access;
Pagrindinis privalumas yra ta, kad ši sistema įeina į Microsoft Office programų rinkinį ir prieinama daugeliui vartotojų. Microsoft Access tinkama kurti

nedidelėms DB, kurias sudaro dešimtys ar šimtai tūkstančių įrašų ir kuriomis naudojasi nedidelis vartotojų skaičius.

- Informacinės sistemos modeliavimo terpė: CASE paketas- Pro Vision Workbench 3.1;
Pagrindinis šaltinis generuojamiems vartotojo sąsajos langų prototipams - biznio objektų modelyje apibrėžiami objektai, jų metodai, atributai bei ryšiai tarp objektų. Pro Vision Workbench 3.1 paketas leidžia naudoti sukurtų objektų duomenis reliacinių duomenų bazių valdymo sistemose (*DBVS*), tokiose kaip Oracle, Visual Basic, Access ir kt., kuriose gali būti atliekamos IS-os funkcijos ar pan.
- Programavimo kalba: Visual Basic 6;
Patogi objektinė programavimo kalba, palaikančios vizualinį programavimą, taip pat Pro Vision Workbench paketas leidžia generuoti ir eksportuoti modelių ir objektų programinį kodą į Visual Basic aplinką.
- Operacinė sistema: Microsoft Windows 98, NT, 2000, XP;
Tai plačiai vietinėje rinkoje naudojama operacinė sistema.
- Prototipo darymas: MS Visual Basic;
VB yra patogi ir lengvai leidžianti projektuoti vartotojo sąsajas sistema. Sistemos kūrėjas gerai susipažinęs su šia sistema, todėl jam nereikės iš naujo prisitaikyti prie kokios nors naujos programavimo aplinkos.

Aparatinė įranga:

IBM PC tipo personalinis kompiuteris. Minimalūs reikalavimai: Procesorius Celeron 450MHz, 2 GB HDD, 128 MB RAM.

- Spausdintuvas (ataskaitoms spausdinti).

Kiti nefunkciniai reikalavimai:

- Programa turi užtikrinti Reliacinių duomenų modelio reikalavimus; taip vadinamą duomenų vientisumą ("Data Integrity").
- Numatyti duomenų kopijų kūrimo galimybę;
- Planuojamas programos dydis iki 10Mb.
- Visos panaudotos taikomosios programos turi būti suderintos.

4.1.4 Prototipai

Programinės įrangos projektavimo procese labai svarbus yra prototipų vaidmuo. Prototipai padeda išsiaiškinti papildomus sistemos funkcinius ir nefunkcinius reikalavimus,

pateikia pradinį sistemos vaizdą. Buvo sukurti keli prototipai papildomų reikalavimų išgavimui, bei vartotojo sąsajos pobūdžio nustatymui.

Reikalavimai, išgauti prototipų kūrimo metu:

- Vartotojo sąsajos funkcionalumo praplėtimui naudoti prototipe “Pagrindinis langas” sukurtus objektus.
- Pagrindiniame manipuliavimo lange parodyti esminis duomenų bazės struktūrą.
- Neapkrauti tiesioginio manipuliavimo langų per dideliu informacijos kiekiu, tam tikslui atsisakoma naudoti meniu komandos.
- Sąsajos languose, atvaizduojančios duomenų bazės lenteles, naudojamos redagavimo, papildymo, veiksmų atšaukimo priemonės.
- Duomenų bazės lentelių, užklausų atvaizdavimui naudoti DBGrid objektus.

4.1.5 Reikalavimai sąsajai

1. Grafinė vartotojo sąsaja turi būti patogi, lengvai suprantama ir informatyvi;
2. Programa turi pateikti kokybišką, vartotojų nevarginantį tekstūromis padengtą vaizdą.
3. Duomenų įvedimas, redagavimas, peržiūrėjimas;
4. Ataskaitų peržiūrėjimas ekrane ir spausdinimas;
5. Diagnostika (klaidų pranešimų aprašymas) – jeigu funkcija vykdoma neteisingai turi išsiterpti aiškus pranešimas apie tai, jog funkcija atliekama neteisingai.
6. Sistemos pranešimai turi būti mandagūs, glausti, nuoseklūs ir konstruktyvūs.

4.1.6 Projekto apribojimai

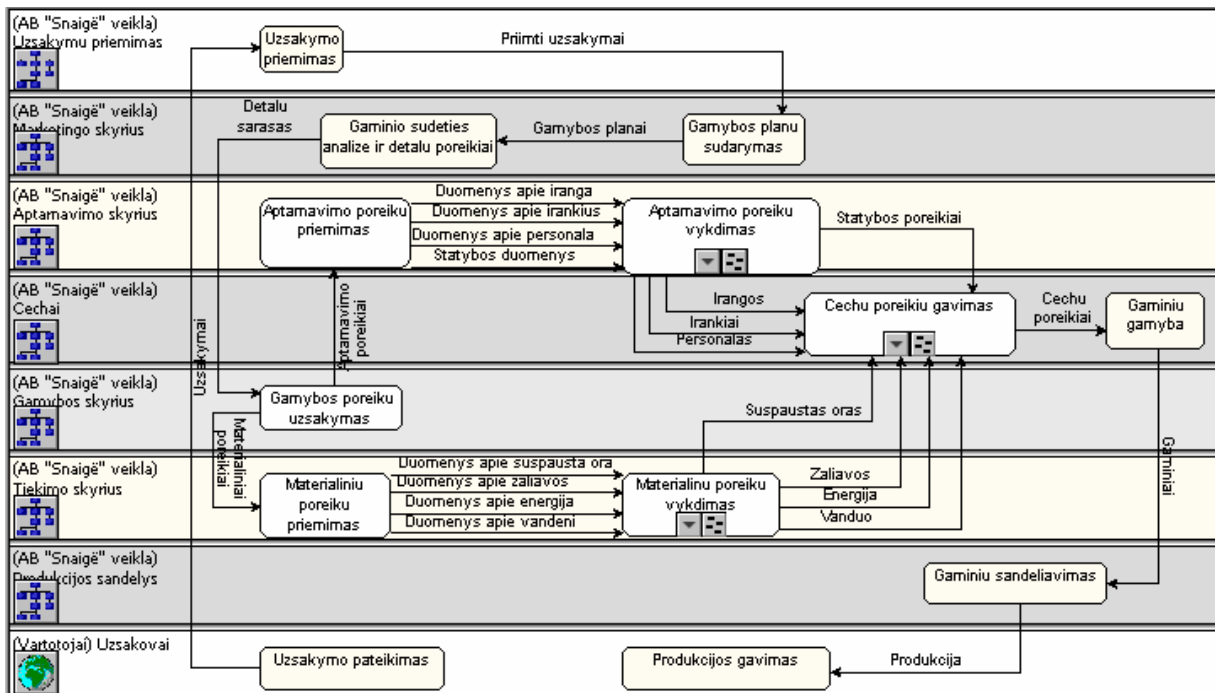
1. Bus naudojama Microsoft Access 2000 duomenų bazė, todėl reikalinga MS Office 2000 (AB „Snaigė” ši programinė įranga jau yra įdiegta).
2. Papildomai reikalingą įsigyti Microsoft Visual Studio rinkinio programa Visual Basic 6.0
3. Techninė įranga - procesorius, monitorius, klaviatūra, pelė ir spausdintuvas.

3.1.7 Duomenų srautai

Duomenų srautų analizė atliekama naudojant darbų sekos modelį (*workflow model*), nes į šį modelį yra įtraukiamos pagrindinės organizacinės struktūros, kurios atlieka tam tikrą

darbų seką. Naudojant darbų sekos modeliavimą leidžiama sukurti detalesnį veiklų modelį, kuris apima vartotojo biznio procesą. Šis modelis atvaizduoja biznio procesus išreiškiant juos veiklos komponentais ir darbų seka tarp tų veiklų. Darbų sekos modelis koncentruotas į darbų seką nuo biznio pradžios iki galo. Tai atspindi pagrindinis aukščiausio lygio modelis. Jei koks nors procesas ar darbas yra sudėtinis kitų darbų atžvilgiu, tai šiam procesui yra sukuriamas detalesnis žemesnio lygio darbų sekos modelis.

Pateikiame pirmąjį lygmens gamybos poreikių organizavimo darbų sekos modelį (*workflow model*) (3 Pav.)



3 Pav. Gamybos poreikių organizavimo darbų sekos modelis (pirmo lygmens)

Užsakovai (tai firmos arba atskiri individai) pateikia užsakymą užsakymo priėmimo skyriui. Marketingo skyrius sudaro gamybos planą, gaminio sudėties ir poreikių analizę. Gamybos skyrius užsako gamybos poreikius tokia tvarka:

Gamybos materialiniai poreikiai: (Tiekimo skyriui)

- Duomenys apie žaliavas;
- Duomenys apie energiją;
- Duomenys apie kurą, vandenį;
- Duomenys apie suspaustą orą.

Gamybos aptarnavimo poreikiai: (Aptarnavimo skyriui)

- Duomenys apie įrangą;
- Duomenys apie įrankius;
- Duomenys apie aptarnaujantį personalą;
- Duomenys apie statybos poreikius.

Tiekimo skyrius teikia materialinius poreikius gamybiniams cechams:

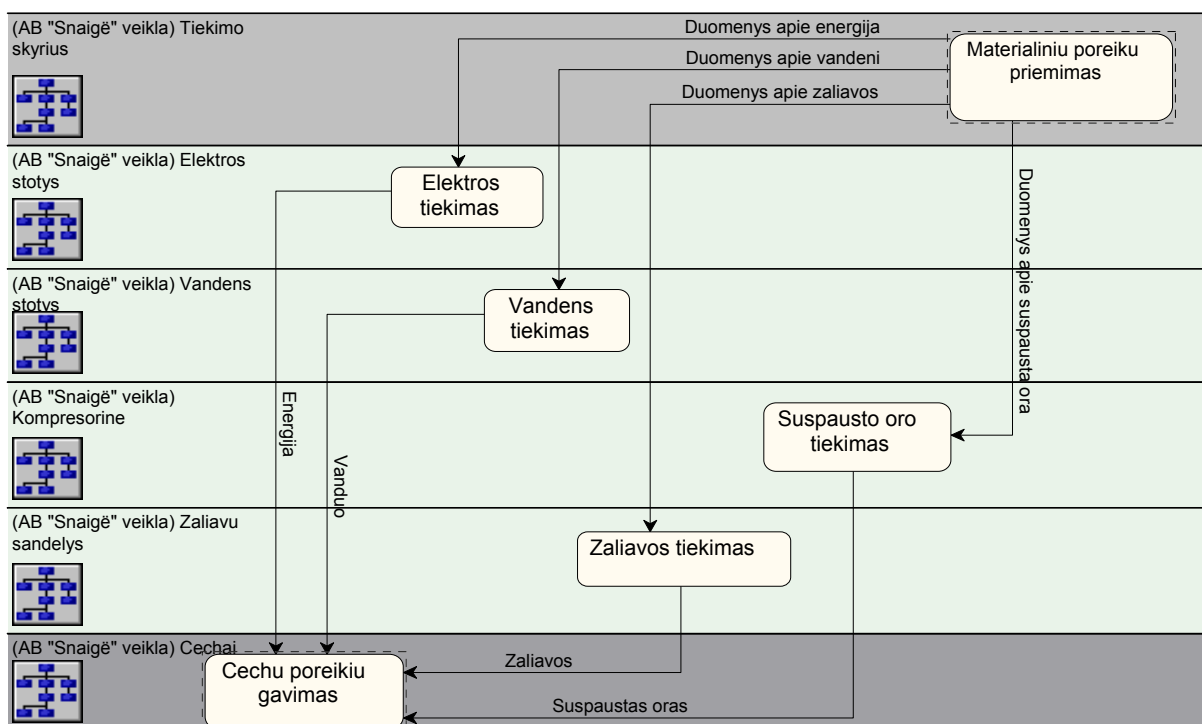
- Žaliavos;
- Elektros energija;
- Kuras, vanduo;
- Suspaustas oras.

Aptarnavimo skyrius teikia aptarnavimo poreikius gamybiniams cechams:

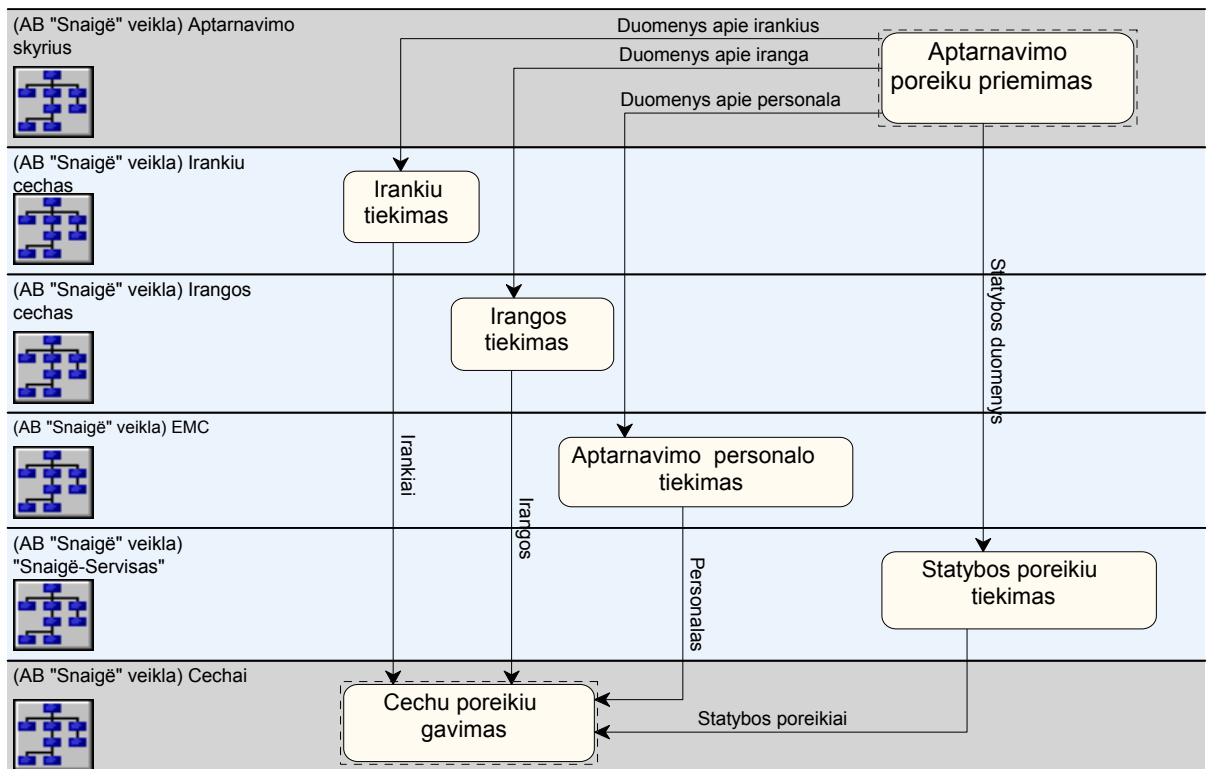
- Įrangos;
- Įrankius;
- Statybos poreikius;
- Aptarnaujantį personalą.

Gamybiniai cechai gauna gamybos poreikius ir vykdo gaminių gamybą. Čia vyksta pagrindiniai gamybiniai procesai. Gatava produkcija patenka į produkcijos sandėlį, iš kur užsakovas perka gatavą produkciją.

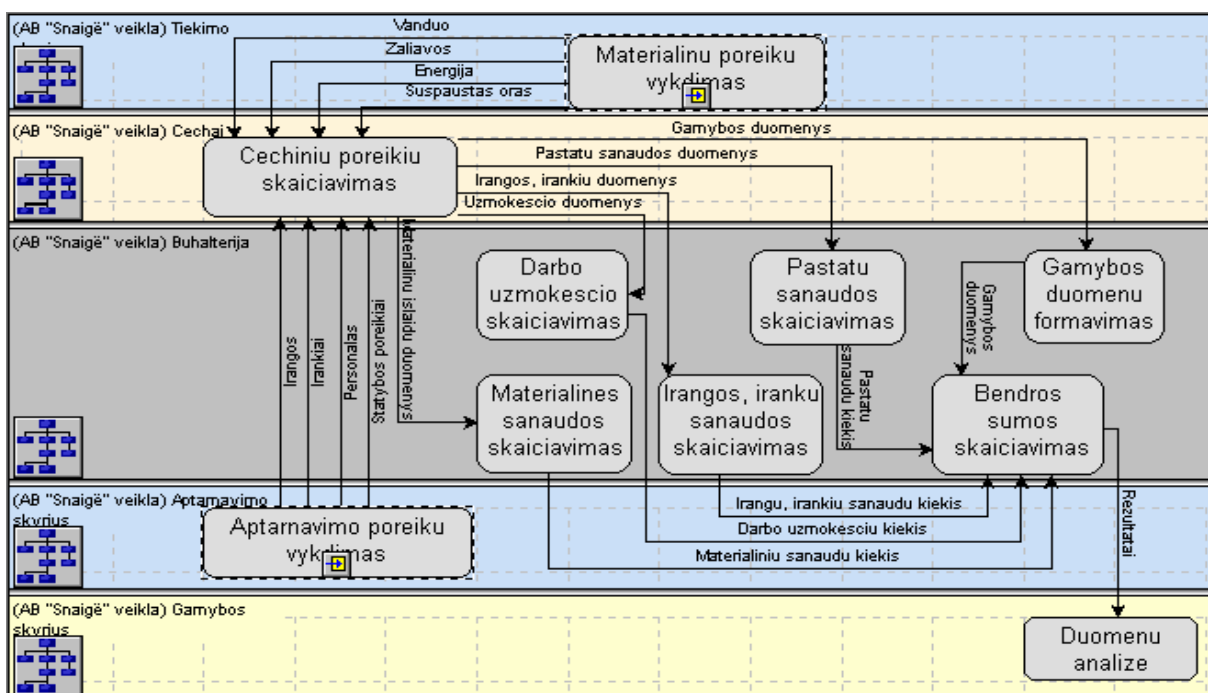
Žemiau pateikiami antrojo lygmens darbų sekos modeliai: materialinių poreikių organizavimas (4 Pav.) ir aptarnavimo poreikių organizavimas(5 Pav.).



4 Pav. Materialinių poreikių organizavimo darbų sekos modelis (antrasis lygmuo)



5 Pav. Aptarnavimo poreikių organizavimo darbų sekos modelis (antrasis lygmuo)



6 Pav. Cechų poreikių organizavimo darbų sekos modelis (antrasis lygmuo)

Cechų poreikių organizavimo darbų sekos modelis (6 Pav.) yra kuriamos programinės įrangos taikomųjų uždavinių modelių UCM (*use case model*) šaltinis. Šis modelis parodo pagrindinius duomenų srautus, skaičiuojant gaminio savikainą.

4.2 Duomenų struktūra

Duomenų struktūros analizei pasirenkame **ProVision Workbench 3.1** paketo Biznio objektu modelį (*Modelers/Object modeler*). Šis modelis atitinka klasių modelį.

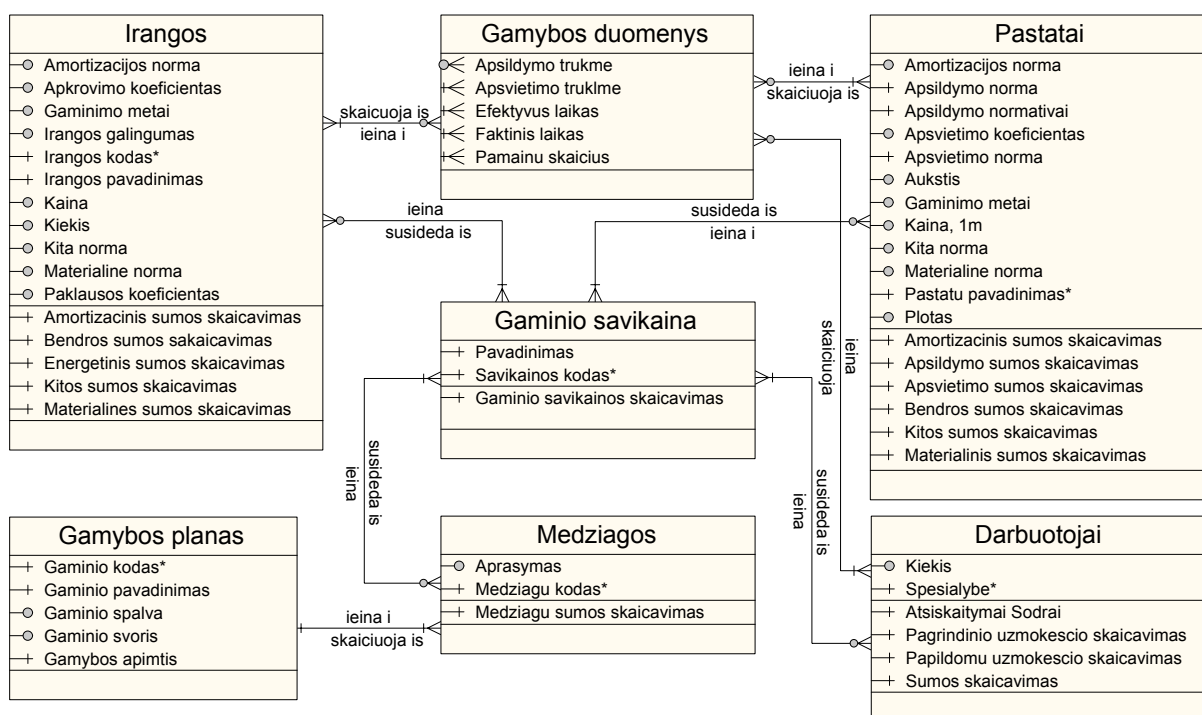
Biznio objektų modelis leidžia vartotojui apibrėžti detalią biznio objektų informaciją (savybes). Objektų modelis sudaro pagrindinį biznio sferos objektų komponentų aprašą. Tai leidžia suprasti vidinius ryšius tarp biznio objektų ir greitai pamatyti visus jų komponentus. Objektu modelis taip pat parodo kiekvieno objekto tokias savybes:

- atributai (*attributes*),
- metodai arba funkcijos (*method*).

Taip pat yra su objektų savybėmis, kurios yra apibrėžiamos, bet vizualiai modelyje nesimato:

- būsenos (*states*) - tai objekto pereinamos būsenos vykstant biznio veiklai,
- potipiai (*subtypes*) - čia nurodomi objekto potipiai.

Objektų modelyje (7 pav.) pavaizduotos objektų savybės, atributai ir metodai.



7 Pav. Objektų modelis

Objektas „Gaminio savikaina“:

Atributai:

- Pavadinimas,
- Savikainos kodas.

Metodai:

- Gaminio savikainos skaičiavimas.

Asociacija (association):

- Susideda iš „Irangos“,

- Susideda iš „Pastatų”,
- Susideda iš „Medžiagos”,
- Susideda iš „Darbuotojų”.

Objektas „Gamybos duomenys”:

Atributai;

- Apšildymo trukmė,
- Apšvietimo trukmė,
- Efektyvus laikas,
- Faktinis laikas,
- Pamainų skaičius.

Asociacija (association);

- Įeiną į „Pastatus”,
- Įeiną į „Įrangas”,
- Įeiną į „Darbuotojus”.

Objektas „Gamybos planas”:

Atributai;

- Gaminio kodas,
- Gaminio pavadinimas,
- Gaminio spalva,
- Gaminio svoris,
- Gamybos apimtis.

Asociacija (association);

- Įeiną į „Medžiagas”.

Objektas „Įrangos”:

Atributai;

- Amortizacijos norma,
- Apkrovimo koeficientas,
- Gaminimo metai,
- Įrangos galingumas,
- Įrangos kodas,
- Įrangos pavadinimas,
- Kaina,
- Kiekis,
- Kita norma,
- Materialinė norma,

- Paklausos koeficientas.

Metodai;

- Amortizacinis sumos skaičiavimas,
- Bendros sumos skaičiavimas,
- Energetinis sumos skaičiavimas,
- Kitos sumos skaičiavimas,
- Materialinis sumos skaičiavimas.

Asociacija (association);

- Skaičiuoja iš „Gamybos duomenų”,
- Įeiną į „Gaminio savikainą”.

Objektas „Darbuotojai”:

Atributai;

- Kiekis,
- Specialybė.

Metodai;

- Atsiskaitymai „Sodrai“,
- Pagrindinių užmokesčių skaičiavimas,
- Papildomų užmokesčių skaičiavimas.

Asociacija (association);

- Skaičiuoja iš „Gamybos plano”,
- Įeiną į „Gaminio savikainą”.

Objektai „Darbininkai”, „Vadovai, specialistai” paveldi superklasės „Darbuotojai” atributus ir metodus, bet turi ir nuosavus atributus (8 Pav.).

Objektas „Darbininkai”:

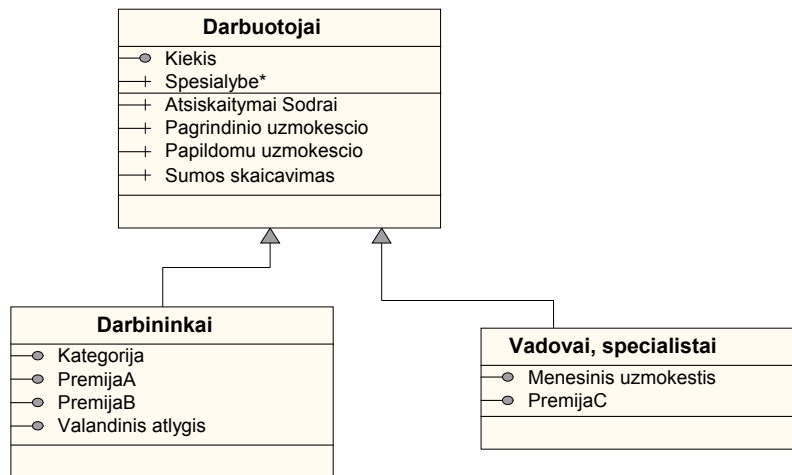
Atributai;

- Kategorija,
- PremijaA
- PremijaB
- Valandinis atlygis.

Objektas „Vadovai, specialistai”:

Atributai;

- Mėnesinis užmokestis,
- PremijaC.



8 Pav. Objekto „Darbuotojai” potipių modelis (SubType Model).

Objektas „Medžiagos”:

Atributai;

- Aprašymas,
- Medžiagų kodas.

Metodai;

- Medžiagų sumos skaičiavimas.

Asociacija (association);

- Skaičiuoja iš „Gamybos duomenų”,

įeiną į „Gaminio savikainą”.

Objektai „Pagrindinės medžiagos” ir „Komplektuojami gaminiai” paveldi superklasės „Medžiagos” atributus ir metodus, taip pat turi savus atributus ir metodus.(9 Pav.)

Objektas „Pagrindinės medžiagos”:

Atributai;

- Atliekų kaina,
- Atliekų norma,
- Medžiagų kaina,
- Medžiagų pavadinimas,
- Svoris.

Metodai:

- Atliekų kainos skaičiavimas,
- Medžiagų kainos skaičiavimas.

Objektas „Komplektuojami gaminiai”:

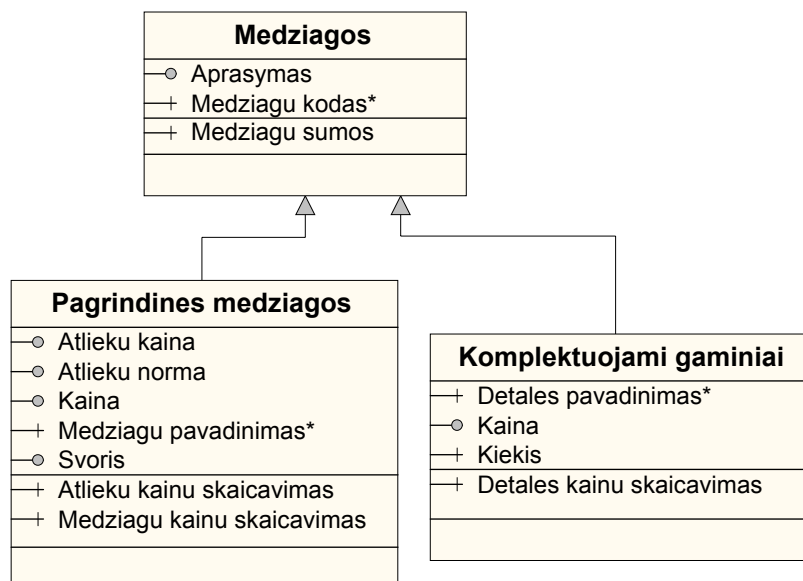
Atributai;

- Detalių pavadinimas,
- Kaina,

- Kiekis.

Metodai;

- Detalių kainų skaičiavimas.



9 Pav. Objekto „Medžiagos” potipių modelis (SubType Model).

Objektas „Pastatai”:

Atributai;

- Amortizacijos norma,
- Apšildymo norma,
- Apšildymo normatyvai,
- Apšvietimo koeficientas,
- Apšvietimo norma,
- Aukštis,
- Gaminimo metai,
- Kaina,
- Kita norma,
- Materialine norma,
- Pastatų pavadinimas,
- Plotas.

Metodai;

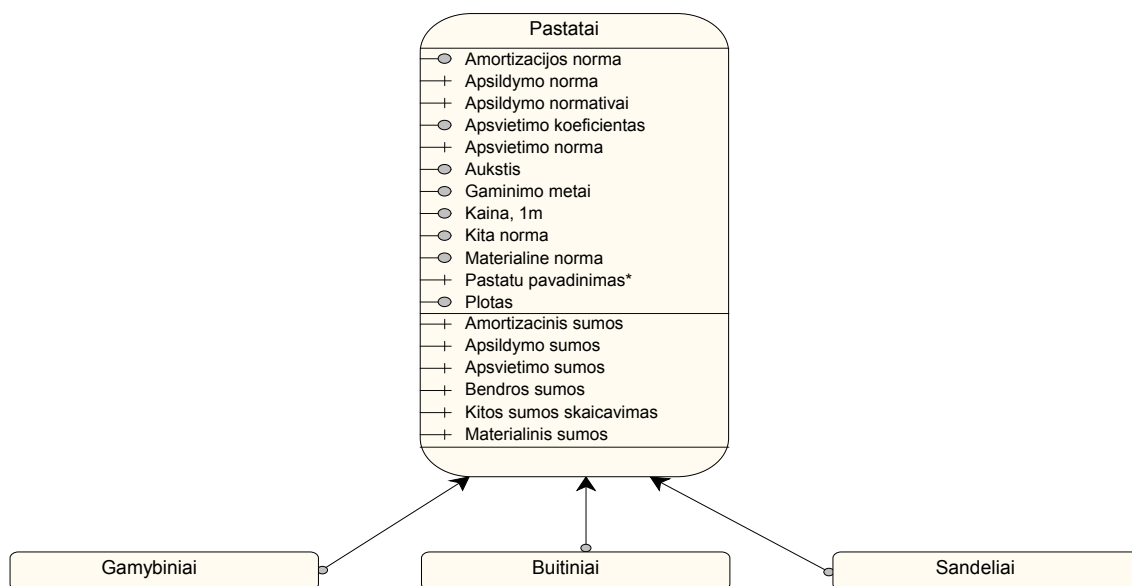
- Amortizacinis sumos skaičiavimas,
- Apšildymo sumos skaičiavimas,
- Apšvietimo sumos skaičiavimas,
- Bendros sumos skaičiavimas,
- Kitos sumos skaičiavimas,

- Materialinės sumos skaičiavimas.

Asociacija (association);

- Skaičiuoja iš „Gamybos duomenų“,
- Įeiną į „Gaminio savikainą“.

Objektai „Gamybiniai“, „Buitiniai“ ir „Sandėliai“ paveldi superklasės „Pastatai“ visus atributus ir metodus (10 Pav.).



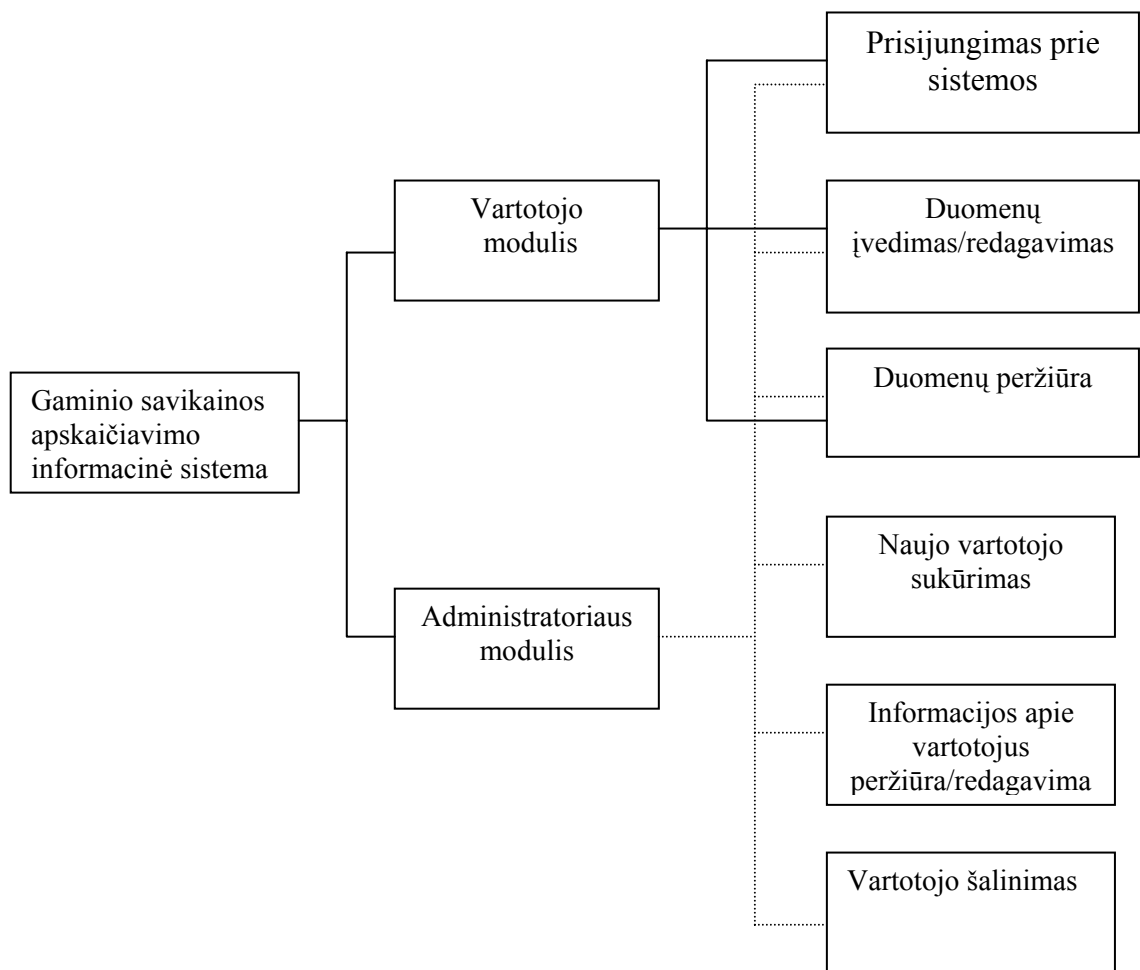
10 Pav. Objekto „Pastatai“ potipių modelis (SubType Model).

4.3 Sistemos architektūra

4.3.1 Sistemos sudedamosios dalys

Kadangi šia sistema naudosis kelių tipų vartotojai, išskiriami du pagrindiniai programiniai moduliai:

- Vartotojo modulis
- Administratoriaus modulis

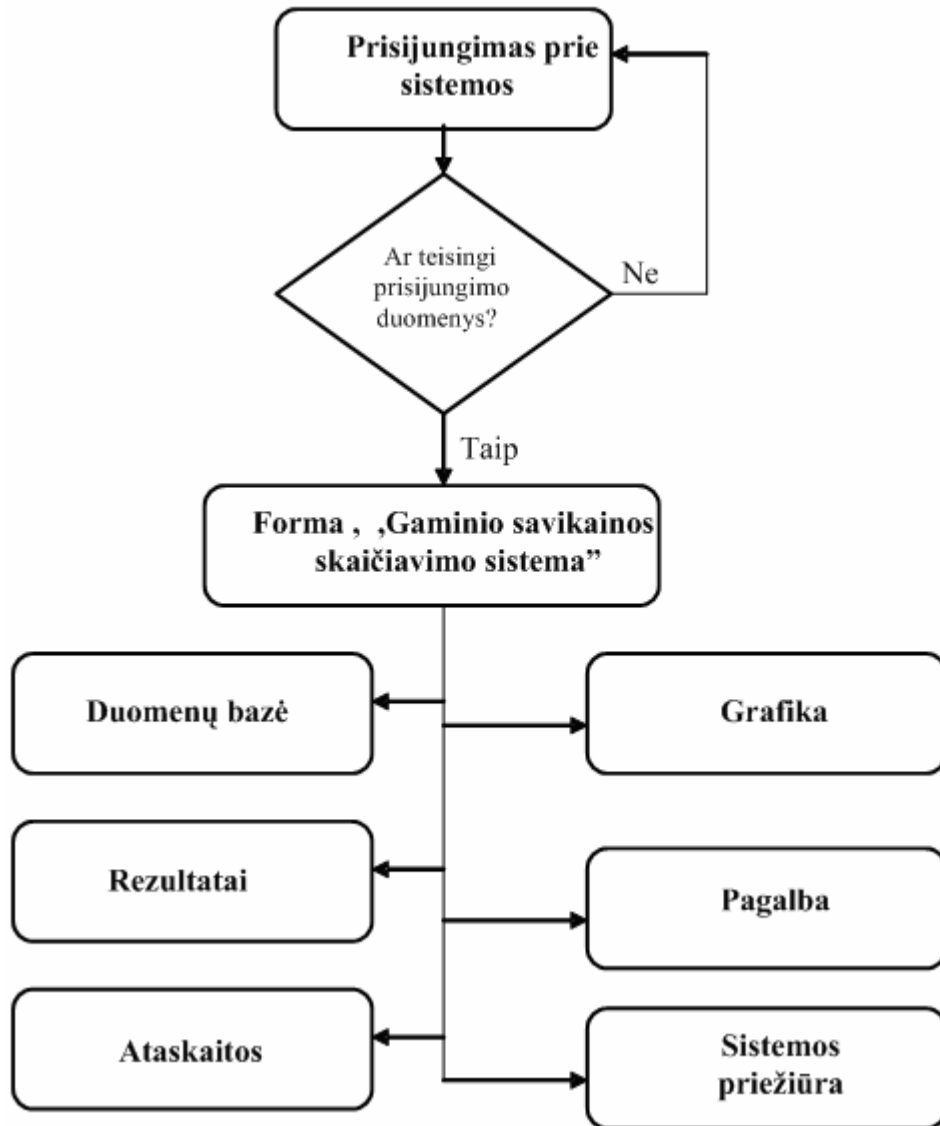


11 Pav. Sistemos struktūros modelis

Vartotojo modulį sudaro dvi dalys:

- Vartotojų, kurie turi teisę peržiūrėti duomenis, modelis;
- Vartotojų, kurie turi teisę redaguoti/įvesti/ peržiūrėti duomenis, modelis;

4.3.2 Vartotojo sąsajos architektūra



12 Pav. Komponentų architektūros sistema.

Šioje schemoje parodytas sistemos pagrindo – vartotojo sąsajos komponentų išdėstymas.

Sąsajos komponentai:

- Prisijungimas prie sistemos – vartotojo prisijungimo informacijos tikrinimo funkcijos
- Forma "Gaminio savikainos apskaičiavimo sistema" – funkcija, formuojanti pagrindinę vartotojo sąsają. Naudoja klasę Form ir jos metodus.
- Duomenų bazė – formos, naudojamos duomenų įvedimui, redagavimui, peržiūrai. Naudoja klasę Form ir jos metodus.

- Rezultatai – formos, naudojamos duomenų apdorojimui ir duomenų peržiūrėjimui. Naudoja klasę Form ir jos metodus.
- Ataskaitos – ataskaitų generavimo failai. Naudoja klasę Reports ir jos metodus.
- Grafika – formos, naudojamos duomenų peržiūrėjimui. Naudoja klasę Form ir jos metodus.
- Pagalba – forma, aprašanti kaip naudotis pagalba. Naudoja klasę Form ir jos metodus.
- Sistemos priežiūra – sistemos administratoriaus vadovas, nurodanti kaip prižiūrėti sistemą.

Komponentas “Prisijungimas prie sistemos”. Vartotojo prisijungimo prie sistemos langas ir priemonė vartotojo identifikacijai. Išveda vartotojo prisijungimo langą. Pagal įvestą informaciją (vartotojo vardas, slaptažodis) patikrina ar toks vartotojas yra, kokiai grupei jis priklauso, kokios jo teisės.

Forma „Gaminio savikainos skaičiavimo sistema”. Tai pagrindinis vartotojo sąsajos langas, naudojantis klasę Form ir jos metodus. Valdymo mygtukai (*Command Button*) pagal paskirtį surūšiuoti į atskiras grupes; tam tikslui naudoti Frame objektai.

Kiekvienas mygtukas turi savo paskirtį; atidaro arba uždaro reikalingus objektus.

Mygtukai suaktyvinami su pele:

- Pasirenkamas reikalingas mygtukas užvedant pelės žymeklį.
- Suaktyvinamas - paspaudus kairiąją pelės mygtuką.

Komponentas „ Duomenų bazė”. Tai formos, naudojamos duomenų įvedimui, redagavimui, peržiūrai. Duomenys suformuojami ir išvedami į ekraną formose DBGrid (*Data Bound Grid*) objekto pavidalu.

Formų sudedamosios dalys:

- Mygtukai „Papildyti”, “Ištrinti”, „Redaguoti”, „Išsaugoti” ir “Baigti”
- Mygtukai, valdantys ryšius (*angl. association*) tarp objektų.
- DBGrid laukai:
 - Tekstui įvesti,
 - Sveikiesiems skaičiams įvesti,
 - Realiesiems skaičiams įvesti,
 - Valiutą išreiškiantiems skaičiams įvesti,
 - Datas įvedimui.

Apribojimai. Vykdamas duomenų įvedimą ar redagavimą būtini formos laukai turi būti užpildyti, kitaip duomenys nebus išsaugomi duomenų bazėje (jei vartotojas neužpildė reikiamų laukų, jam, saugant duomenis išvedamas pranešimas).

Vartotojas taip pat gauna pranešimus, jei įvedami nekorektiški duomenys.

Vartotojas įveda (redaguoja) duomenis į DBGrid laukus ir spaudžia vieną iš šių mygtukų:

Papildyti – leidžiama įvesti naujus duomenis, sudaroma nauja duomenų bazės lentelės eilutė.

Redaguoti – pasirinktą informaciją galima redaguoti.

Išsaugoti – baigus redagavimą informacija bus išsaugota.

Ištrinti – vartotojas gali ištrinti nereikalingus įrašus, prieš tai gaunant įspėjimus – pranešimus.

Baigti – uždaro formą.

Komponentas „ Rezultatai”. Formos, naudojamos duomenų apdorojimui ir duomenų peržiūrėjimui. Duomenys įvesti komponente „ Duomenų baze” čia apdorojami ir išvedami į ekraną. Vartotojas duomenų apdorojimui neturi įtakos; gali peržiūrėti visą arba dali informacijos.

Formų sudedamosios dalys:

- Valdymo mygtukai, įvykdantys objektų metodus.
- DBGrid objektai:
- Mygtukas „Baigti”

Apribojimai: Vartotojas negali redaguoti, papildyti, ištrinti informacijos. Ši informacija skirta tik peržiūrėjimui. Jei neuždarytas komponentas „Duomenų bazė”, neatsidarys forma „Rezultatai”.

Komponentas „Ataskaitos” - tai klasė, su kuria generuojamos ataskaitos. Kiekvienai ataskaitai sukuriamas atskiras failas, kuriame, naudojantis klasės Reports metodais, suformuojama ataskaita.

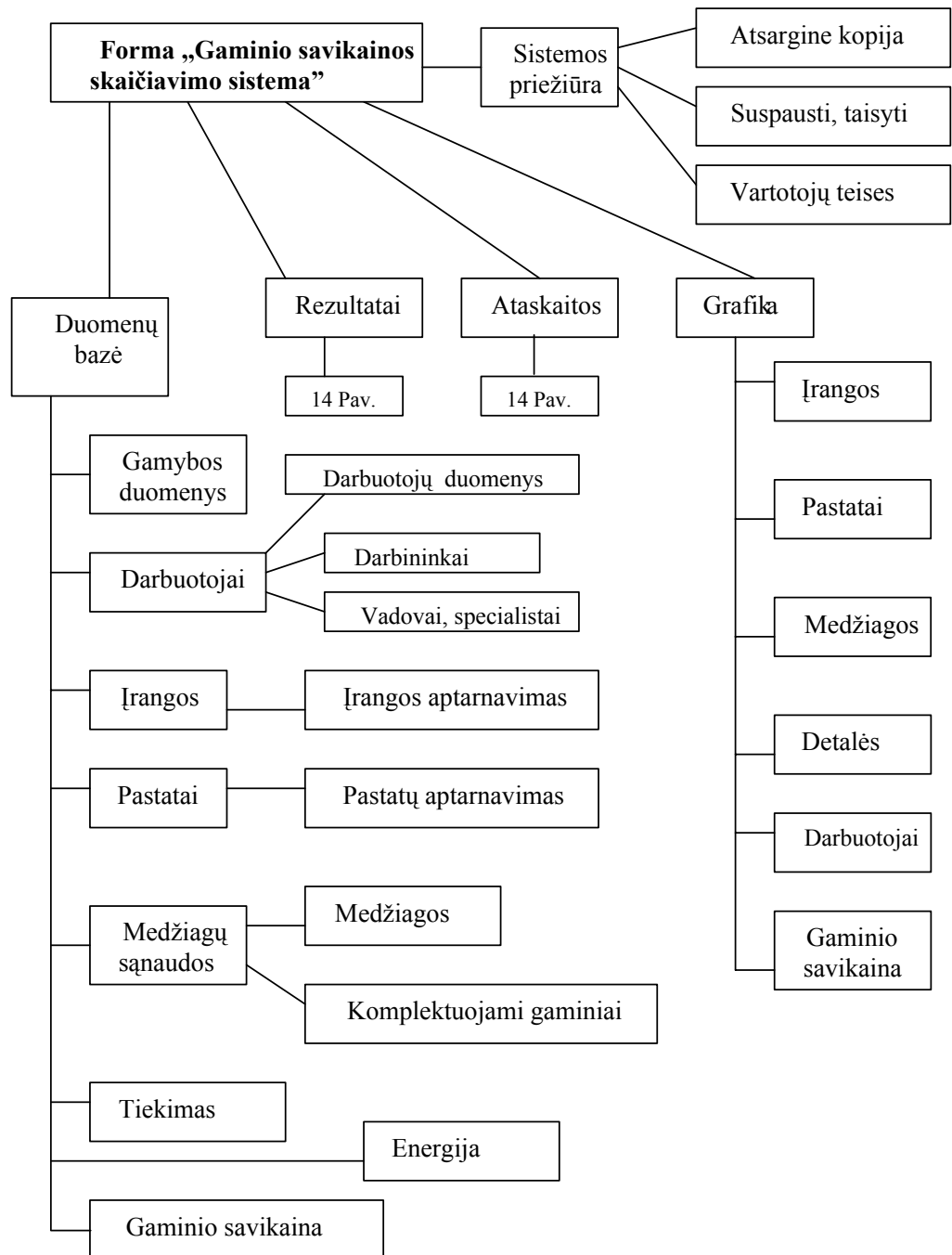
Pastabos. Formuoja ataskaitos langą. Ataskaitoje išvedama visa išrinkta iš duomenų bazės lentelės informacija. Vartotojas gali atsispausdinti ataskaitos turinį.

Išrinkus iš sąrašo ataskaitą, iškviečiama ataskaitą generuojanti funkcija. Kiekvienai ataskaitai kviečiama vis kita funkcija, kuri formuoja ataskaitą pagal iš anksto užduotus kriterijus.

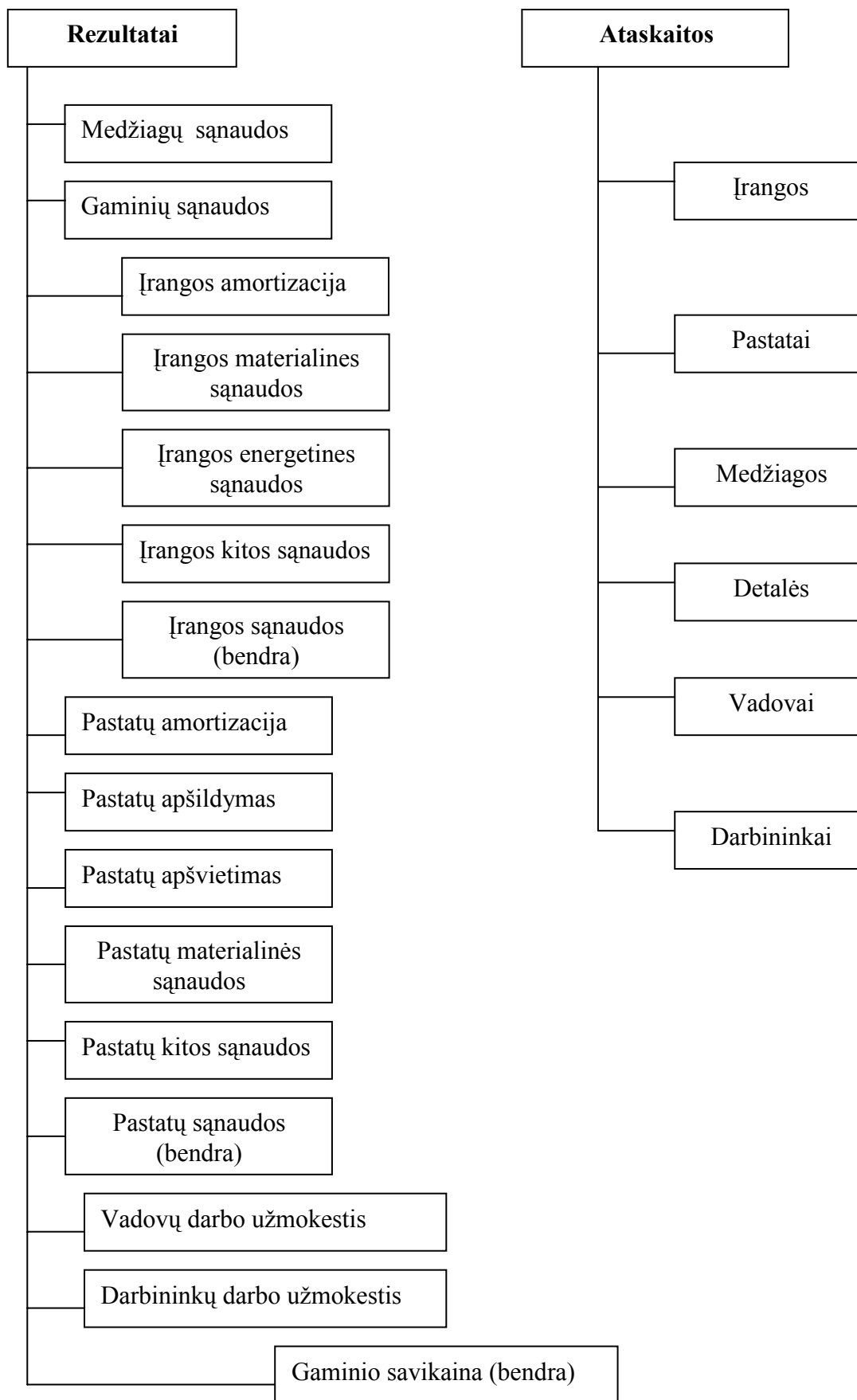
Komponentas „Grafika”. Formos, naudojančios klasę Form ir jos sąsają su OLE (*Object Linking and Embedding*) protokolu. OLE protokolas leidžia dviem programoms automatiškai keistis duomenimis. Šis protokolas leidžia vartotojui be specialaus pasirengimo peržiūrėti ir tvarkyti reikalingus objektus, parengtus „Microsoft Office” programų paketais „Word”, „Excel”, „Power Point” t.t.

Komponentas „Sistemos priežiūra”. Forma skirta sistemos administratoriui. Sistemos administratorius čia dirba su programines įrangos failais, nustato vartotojų ir grupių teises, pašalina arba sukuria naują vartotoją.

4.4 Programinių modulių specifikacija



13 Pav. Programines įrangos objektu duomenų struktūra.



14 Pav. Programinės įrangos objektu duomenų struktūra.

4.5 Testavimo medžiaga

4.5.1 Testavimo strategija

Čia aprašomos testavimo strategijos, kurios bus atliekamos testuojant sistemą.

Struktūrinis testavimas

Kadangi žinoma programos struktūra ir jos veikimas, struktūrinis testavimas bus atliekamas kodavimo metu, tikrinant kiekvieno naujai parašyto sistemos komponento veikimą. Tikrinimą sudarys įvairių duomenų kombinacijų padavimas testuojamam objektui ir jo išduodamų rezultatų stebėjimas.

Funkcinis testavimas

Tikrinamos atskiros sistemos dalys ir sistema kaip viena visuma. Siekiama išsiaiškinti visas galimas klaidas, prieš diegiant sistemą pas užsakovą.

Sistemos integralumo testavimas

Sistemos integralumas bus tikrinamas diegimo laikotarpiu ir iškart po sistemos įdiegimo. Bus vedami įvairūs duomenys, lyginami laukiami ir gauti rezultatai, tikrinama ar sistema korektiškai veikia užsakovo informacinės sistemos aplinkoje.

Validavimas

Bus atliekamas įdiegus sistemą pas užsakovą. Per 1-2 mėn. užsakovas naudosis sistema, bus fiksuojamos klaidos ir papildomi vartotojo pageidavimai, reikalingi sistemai tobulinti.

4.5.2 Testavimo planas

Testuojama sistema susideda iš duomenų bazės bei vartotojo sąsajos. Testuojant atskirus sąsajos modulius patikrinamas kiekvieno sąsajos elemento veikimas, tikrinama, ar teisingai įvedami duomenys ir daromi pakeitimai duomenų bazėje, ar korektiškai veikia pati sąsaja.

Bus testuojami šie sąsajos elementai:

- Prisijungimo langas - vedant skirtingus prisijungimo vardus ir slaptažodžius.
- Pagrindinis langas.

Pagrindinis langas sudarytas iš valdymo mygtukų (*Command Button*). Šie mygtukai pagal paskirtį surūšiuoti į atskirtas grupes Frame, kuriose yra visi komponentai. Kiekvienas mygtukas turi savo paskirtį: atidaro arba uždaro reikalingus objektus.

Komandinių mygtukų grupės, atskirtos Frame objektų:

- Duomenų bazė
- Rezultatų langas

- Ataskaitos langas
- Grafikos langas
- Pagalbos langas
- Sistemos priežiūra.

Mygtukai suaktyvinami paspaudus ant jo kairiąją pelės mygtuką. Kiekvienas mygtukas atidaro forma.

Vartotojo sąsaja su pagrindinėmis formomis ir ataskaitomis atrodo taip:

Duomenų bazė: Pastatai, pastatų aptarnavimas, įrangos, įrangos aptarnavimas, vadovai ir specialistai, darbininkai, darbuotojų duomenys, medžiagos, komplektuojami gaminiai, gaminio savikaina.

Rezultatai: Pagrindinės medžiagos, komplektuojami gaminiai, įrangos amortizacija, įrangos energetinis sąnaudos, įrangos materialinės sąnaudos, įrangos kitos sąnaudos, įrangos sąnaudos (bendra), darbininkų darbo užmokesčiai, vadovų darbo užmokesčiai, pastatų amortizacija, pastatų apšildymas, pastatų apšvietimas, pastatų materialinės sąnaudos, pastatų kitos sąnaudos, pastatų sąnaudos (bendra), gaminio savikaina (bendra).

Ataskaitos: Įrangos, pastatai, medžiagos, detalės, darbininkai, vadovai.

Pagalba: Darbuotojų paieška, medžiagų paieška, detalių paieška, pastatų paieška, įrangos paieška, energijos paieška.

Grafika: Įrangos, pastatai, detalės, medžiagos, darbuotojai, gaminio savikaina

Sistemos priežiūra: Pasirinkus Duomenų bazės langus, vartotojas, priklausomai nuo teisių, gali atlikti šiuos veiksmus:

Papildyti -

Ištrinti -

Redaguoti –

Baigti redagavimą –

Išeiti –

Kiekvienose duomenų bazių formose vartotojas gali nustatyti ryšį (*association*) tarp lentelių. Pavyzdžiui, vartotojas, pasirinkęs formą „**Įrangos**“, iš tos formos gali patekti į kitas formas:

Įrangos aptarnavimas –

Energija –

Pagrindiniai duomenys –

Gaminio savikaina –

Pasirinkęs formas „**Rezultatai**“, vartotojas gali tik peržiūrėti informaciją. Redagavimo galimybės vartotojas neturi. Pavyzdžiui, pasirinkęs formą „**Vadovų, specialistų darbo užmokestis**“, vartotojas gali peržiūrėti informaciją:

Pagrindinis darbo užmokestis ;

Atskaitymai „Sodrai“ ;

Bendra suma ;

4.5.3 Testavimo eiga

Detalus atskirų sistemos dalių testavimo aprašymas.

Prisijungimo langas.

Vedami teisingi ir neteisingi vartotojų vardai ir slaptažodžiai. Paspaudus mygtuką OK, jei duomenys neteisingi, gaunamas pranešimas, jei teisingi, užkraunama pagrindinė vartotojo sąsaja.

Pagrindinė vartotojo sąsaja.

Tikrinamas vaikščiojimas po valdymo mygtukus su pele. Jei pagal teises Command Button mygtukai vartotojui neturi būti prieinami, jis turi būti neaktyvus. Suaktyvinus mygtuką, turi atsirasti atitinkamas langas.

Duomenų bazės formų bendri objektai:

Mygtukas „Papildyti“ – leidžia įvesti naujus duomenis.

Mygtukas „Ištrinti“ – ištrina aktyvius duomenis.

Mygtukas „Išsaugoti“ - duomenis perduoda į DBVS

Mygtukas „Baigti“ – uždaro formas.

Visos duomenų bazės lentelės vartotojui parodomos DBGrid objekto pavidalu. Kiekvienas DBGrid objekto laukas turi savo duomenų tipą, griežtai apibrėžtą DBVS lentelėse.

Teksto įvedimo laukas – į šį lauką vedama simbolinė informacija. Kiekvienas toks laukas turi dydžio apribojimą (priklausomai nuo dydžio DBVS lentelėje).

Sveikų skaičių įvedimo laukas – į šį lauką vedami tik sveikieji skaičiai. Kiekvienas toks laukas turi dydžio apribojimą (priklausomai nuo dydžio DBVS lentelėje).

Skaičių su slankiu kableliu įvedimo laukas – analogiškas prieš tai aprašytam, tik leidžia papildomai įvesti simbolį, skiriantį sveiką ir trupmeninę skaičiaus dalis.

Datos įvedimo laukas – skirtas įvesti datai. Tikrinamas įvedamų metų, mėnesio, dienos teisingumas. Turi apibrėžtas ribas.

Perėjimas tarp DBGrid objekto laukų vykdomas su pele arba klaviatūra („Enter“). Jei neužpildomi būtini formos laukai arba kuris nors laukas užpildomas neteisingai – išvedamas klaidos pranešimas.

Ataskaitos. Pasirinkus šį meniu punktą, galima pasirinkti reikalingas ataskaitas. Kai kuriose iš jų dar reikia įvesti arba pasirinkti reikalingus kriterijus. Pasirinkus kriterijus, reikia paspausti mygtuką „Ataskaita“, tada bus išvedama į ekraną ataskaitos informacija.

5. VARTOTOJO DOKUMENTACIJA

Vartotojo vadovo dokumentas skirtas sistemos vartotojui. Aprašoma sistemos galimybės, paskirtis, atliekamos funkcijos bei jų panaudojimas, pateikiamos instrukcijos sistemos diegimui bei palaikymui.

5.1 Sistemos funkcinis aprašymas

Ši programinė įranga kompiuterizuotai apskaičiuoja gaminio savikainą. Atlieka atitinkamus skaičiavimus, saugoja duomenis, spausdina reikiamas ataskaitas. Pateikia programos sąsają su reikiamais dialogo langais (lietuvių kalba). Apsaugoja duomenis nuo svetimų asmenų slaptažodžiais.

Informacijos sistema skirta dirbti įvairiems vartotojams, kuriems yra ribojamos galimybės prieiti prie su jų darbu nesusijusios informacijos. Vartotojų grupės - buhalteris ir skyriaus vadovas. Visa pagrindinė informacija pateikta duomenų bazėje; vartotojas - buhalteris gali esant reikalui redaguoti, papildyti, įvesti naują informaciją, gerai susipažinęs su šia programa. Apskaičiuotus rezultatus buhalteris ir skyriaus vadovas negali redaguoti. Tai gali atlikti tik sistemos administratorius.

Ši programinė įranga palengvina įmonės buhalterijos darbuotojų darbą. Naudodamasis programine įranga, darbuotojas gali atlikti šias funkcijas:

- suvesti ir koreguoti pirminius duomenis;
- suvesti ir koreguoti normatyvinę informaciją;
- atlikti reikalingus skaičiavimus;
- peržiūrėti rezultatus;
- formuoti ataskaitas.

Ši programinė įranga atlieka tokius skaičiavimus:

- medžiagų sąnaudos;
- atliekų sąnaudos;
- komplektuojamų gaminių sąnaudos;
- pastatų sąnaudos:
 - pastatų nusidėvėjimo sąnaudos;
 - pastatų apšildymas;
 - pastatų apšvietimas;
 - pastatų materialinės sąnaudos;
 - kitos materialinės sąnaudos.

- Įrangos sąnaudos:
 - įrangos nusidėvėjimo sąnaudos;
 - įrangos energetinis sąnaudos;
 - įrangos materialinės sąnaudos;
 - kitos materialinės sąnaudos.
- Darbo užmokesčio sąnaudos:
 - pagrindinės darbo užmokesčio sąnaudos;
 - papildomos darbo užmokesčio sąnaudos;
 - atskaitymai;
 - bendros darbo užmokesčio sąnaudos.

5.2 Sistemos vadovas

5.2.1 Sistemos struktūra

Šiame skyriuje aptarsime sistemos struktūrą bei ją sudarančius sistemos komponentus ir ryšius. Sistema sudaryta iš pagrindinio lango, pagrindinės formos, formų bei pranešimų formų, kurių kiekvienas yra susietas tarpusavyje.

Pagrindinis sistemos langas tarnauja kaip visos sistemos pagrindas. Pagrindiniame lange su mygtukais galima atsidaryti formas, ataskaitas, rezultatus. Taigi, sistemos pagrindinis langas tarnauja kaip sistemos grafinio pateikimo pagrindas.

Pagrindinis sistemos langas susideda iš komponentų:

- Duomenų bazė;
- Rezultatai;
- Ataskaitos;
- Grafika;
- Pagalba;
- Servisas.

Pasirinkus atitinkamą mygtuką – komponentą, atsidaro vartotojo sąsajos formos, kuriose surūšiuoti objektai pagal paskirtį. Atitinkamai nuo prisijungusio prie sistemos vartotojo turimų teisių, kai kurie pagrindinės formos punktai gali būti neaktyvus. Tai yra vartotojai turi priėjimą tik prie jiems skirtų formų ir ataskaitų. Pasirinkus žemesnio lygio punktą, yra iškviečiama atitinkama forma, kurioje vartotojas gali atlikti tam tikrus veiksmus.

Formos sąlyginai suskirstytos į:

- Duomenų apdorojimo formos;
- Duomenų peržiūros formos;

- Rezultatų – užklausų išvedimo formos;
- Ataskaitų formavimo parametrų nustatymo formos;
- Grafikos elementų apdorojimo formos.

Bet kuri forma iš šių formos grupių išskviečiama pagrindinėje formoje pasirinkus atitinkamą mygtuką. Vartotojas atidarytoje duomenų apdorojimo formoje gali atlikti veiksmus su tam tikrais duomenimis. Atidarytoje duomenų peržiūros ir rezultatų – užklausų išvedimo formose vartotojas pagal tam tikrus kriterijus gali peržiūrėti duomenis. Grafikos elementus apdorojimo formose vartotojas be specialaus pasirengimo gali peržiūrėti ir tvarkyti reikalingus objektus parengtus „Microsoft Office“ programų paketais „Word“, „Excel“, „Power Point“ t.t.

Dirbant su sistema vartotojui yra išvedamos pranešimo formos t.y. pranešimai, kuriuos galima suskirstyti:

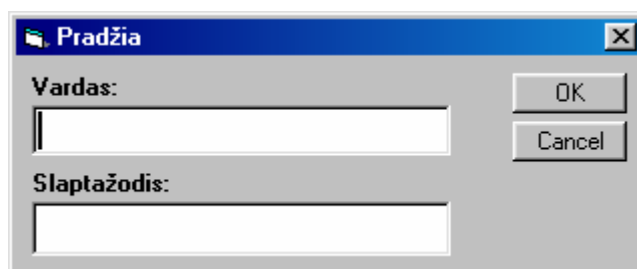
- Dėmesio – pranešimai su atitinkamu informaciniu tekstu;
- Perspėjimas – pranešimai, perspėjantys vartotoją apie turimą pasirinkimą;
- Patvirtinimas – patvirtinimo reikalaujantys pranešimai;
- Klaida – pranešimai įspėjantys, kad neįmanoma atlikti kai kurių veiksmų.

5.2.2 Vartotojo atmintinė

Šiame skyriuje bus pateikiamas detalus sistemos naudojimo galimybių ir funkcijų aprašymas, pranešimai apie klaidas bei kiti darbo su sistema aspektai.

5.2.2.1 Bendri darbo su sistema aspektai

Kiekvienam vartotojui pradėdant dirbti su sistema, pateikiamas vartotojo prisijungimo langas, kuriame jis turi įvesti prisijungimo vardą ir slaptažodį.



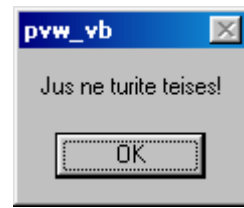
14 Pav. Prisijungimo langas

Vartotojas, suvedęs prisijungimo slaptažodį, gali spausti mygtuką „OK“, kad prisijungtų prie sistemos arba „Cancel“, kad nutrauktų autorizuotą prisijungimą ir nebetęstų

darbo su sistema. Jeigu suvesti duomenys teisingi, pasirodys forma (pav. 15), jeigu suvesti duomenys neteisingi pasirodys forma (pav. 16).



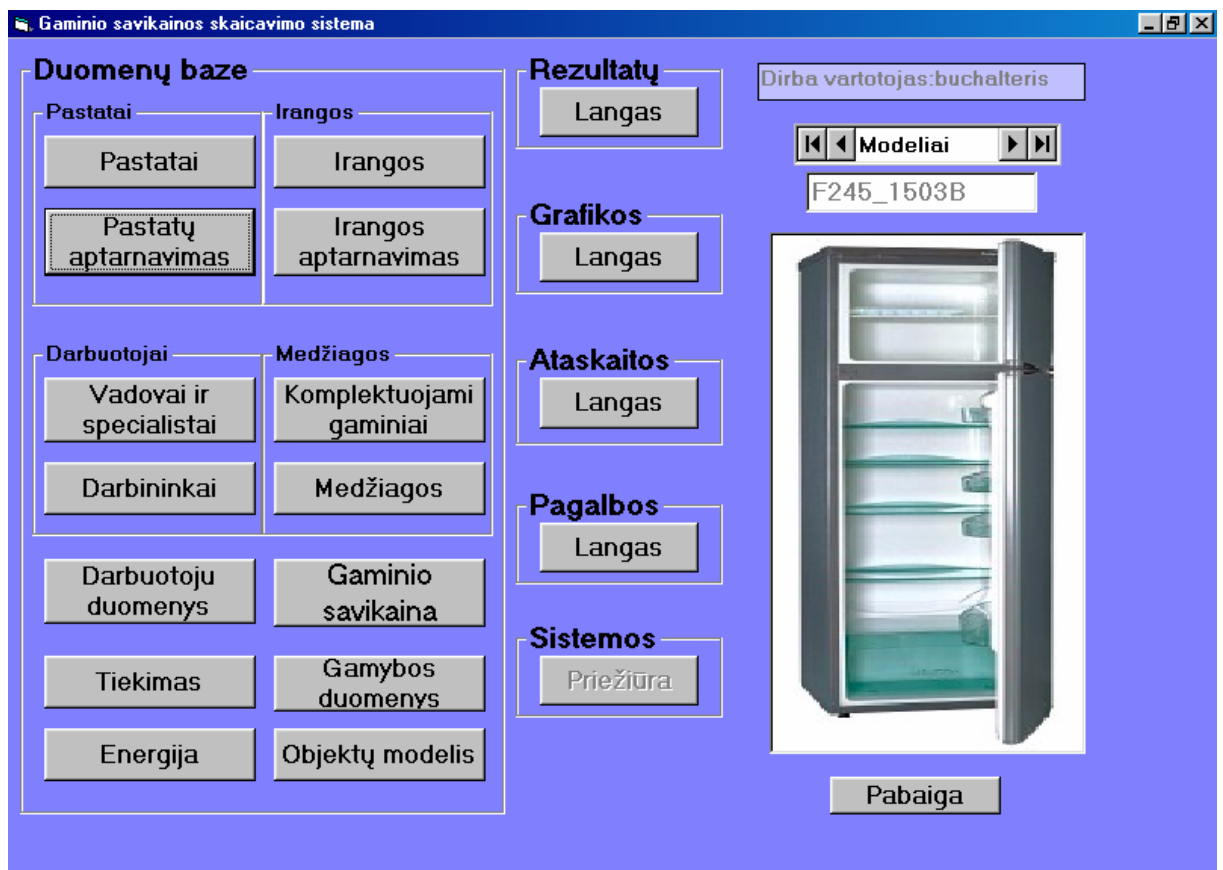
Pav.15 Pranešimo langas



Pav.16 Pranešimo langas

Tuo atveju jei vartotojas blogai įvedė prisijungimo vardą ir slaptažodį ir bando prisijungti prie sistemos, jam pranešama, kad įvesti neteisingi duomenys, ir sistema atsijungia.

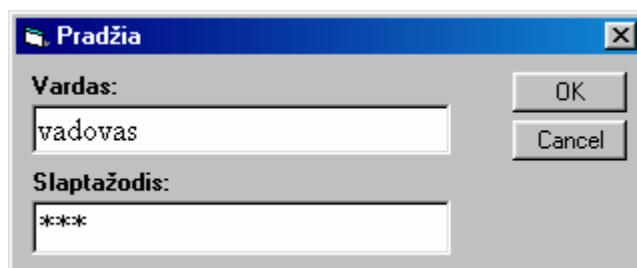
Kai vartotojas teisingai įvedą vardą ir slaptažodį, vartotojas prisijungia prie sistemos ir atsidaro pagrindinis sąsajos langas.



Pav.17 Pagrindinis sąsajos langas

5.2.2.2 Pagrindinis vartotojo langas

Prisijungus vartotojui, aktyvūs būna tik tie sąsajos punktai, prie kurių vartotojas turi priėjimą. Pavyzdžiui, prisijungus vartotojui, kurio vardas „vadovas“, pagrindinėje formoje matoma, kad kai kurie mygtukai yra neaktyvūs (pilkos spalvos). Prie šių punktų vartotojas prieiti negali (tai yra jis negali nei įvesti, nei redaguoti, nei trinti duomenų). Jis gali suaktyvinti tik punktus, kurie ryškūs.



18 pav. Prisijungimo langas



19 pav. Pagrindinis sąsajos langas, kai vartotojas yra „vadovas”

Pagrindinis sąsajos langas yra sudarytas iš pagrindinių punktų:

- Duomenų bazė – formos, naudojamos duomenų įvedimui, redagavimui, peržiūrai. Naudoja klasę Form ir jos metodus.
- Rezultatai – formos, naudojamos duomenų apdorojimui ir duomenų peržiūrėjimui. Naudoja klasę Form ir jos metodus.

- Ataskaitos – ataskaitų generavimo failai. Naudoja klasę Reports ir jos metodus.
- Grafika – formos, naudojamos duomenų peržiūrėjimui. Naudoja klasę Form ir jos sąsaja su OLE (*Object Linking and Embedding*) protokolą.
- Pagalba – forma, aprašanti kaip naudotis pagalba. Naudoja klasę Form ir jos metodus.
- Sistemos priežiūra - duomenų bazės priežiūra, vartotojų ir grupių nustatymas vartotojų ir grupių teisių priskyrimas.

5.2.2.3 Duomenų bazė

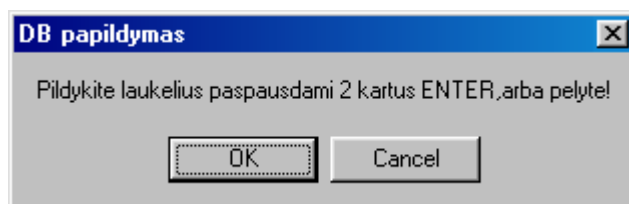
Tai pagrindinis sąsajos lango punktas, naudojamas duomenų įvedimui, redagavimui, peržiūrai. Duomenys suformuojami ir išvedami į ekraną formose DBGrid (*Data Bound Grid*) objekto pavidalu. Pasirinkus atitinkamą mygtuką, atsidaro reikalingos formos. Tai yra kitos formos:

- Gamybos duomenys;
- Pastatai;
- Pastatų aptarnavimas;
- Įrangos;
- Įrangos aptarnavimas;
- Darbuotojų duomenys;
- Darbininkų darbo užmokesčio sąnaudos;
- Vadovų ir specialistų darbo užmokesčio sąnaudos;
- Medžiagos;
- Komplektuojami gaminiai;
- Medžiagų ir komplektuojamų gaminių tiekimas;
- Energetinės sąnaudos;
- Gaminio savikaina.

Kiekviename “Duomenų bazės” puslapyje yra šie duomenų apdorojimo mygtukai:

Papildyti

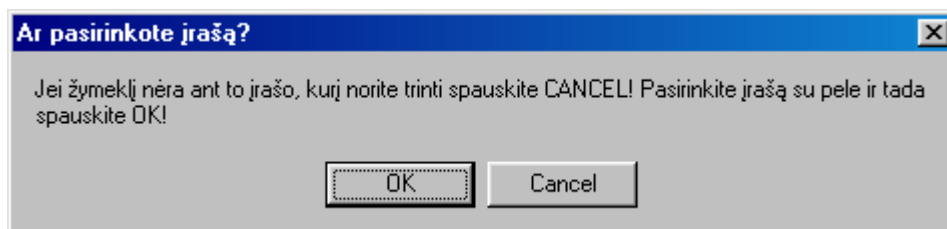
-tai naujo įrašo įvedimas į DBGrid laukus. Duomenų įvedimo laukeliai atsidariusioje DBGrid lentelėje neredaguojami, kol nepasirinktas šis mygtukas. Pasirinkus šį mygtuką, žymeklis nustatomas į pirmąjį duomenų įvedimo laukelį. Vartotojas gali įvesti naujus duomenis. Jei vartotojas paspaudžia šį mygtuką, gauna pranešimą:



20 Pav. Pranešimo langas

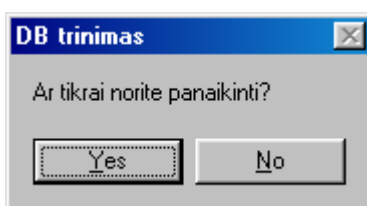
Pasirinkus mygtuką „OK”, žymeklis nustatomas į pirmąjį duomenų įvedimo laukelį, pasirinkus „Cancel”, išiname iš redagavimo režimo.

Ištrinti - šiuo mygtuku vartotojas gali ištrinti visą eilutę DBGrid lentelėje, prieš tai gaunant perspėjimą:



21 Pav. Pranešimo langas

Jei vartotojas pasirenka „OK”, gauna antrąjį perspėjimą:



22 Pav. Pranešimo langas

Vartotojas, pasirinkęs „DB trinimas” formoje mygtuką „Yes”, DBGrid lentelės eilutė ištrinama, pasirinkus mygtuką „No”, išiname iš redagavimo režimo.

Redaguoti - vartotojas gali redaguoti atskirus duomenų bazės laukus, jei yra poreikis. Šis mygtukas leidžia suaktyvinti lauką, kurį pasirenka vartotojas. Pasirinkus šį mygtuką formos teksto laukose (*TextBox*), suaktyvinamas įrašas:

Pasirinkite įrašą, redaguokite, po to spauskite mygtuką :Išsaugoti

23 Pav. Pranešimas

Išsaugoti - vartotojas po operacijos „Redaguoti” ir „Papildyti” turi pasirinkti šį mygtuką, kuris išsaugoja įvestus duomenys.

Baigti - vartotojas išina iš atitinkančios formos.

Forma „Gaminio savikaina” yra (20 pav.) pagrindinė, taip vadinama sąryšių forma. Vartotojas gali patekti tiesiogiai į kitas duomenų bazės lenteles. Formos laukai yra:

- Gaminio pavadinimas;
- Sąnaudos kodas;
- Sąnaudų pavadinimas.

Gaminis	Sąnaudos kodas	Pavadinimas
RF315_1503A	darbpg	Pagrindiniu darbininku
RF315_1503A	darbpl	Pagalbiniu darbininku
RF315_1503A	irangsan	Įrangos sanaudas
F245_1503B	kompqam_245	Komplektuojami gamii
RF270_1503A	kompqam_270	Komplektuojami gamii
RF315_1503A	kompqam_315	Komplektuojami gamii
RF360_1501A	kompqam_360	Komplektuojami gamii

24 pav. Forma „Gaminio savikaina”

Paspaudęs mygtuką „Vadovai ir specialistai”, vartotojas atidaro formą „Vadovai ir specialistai”.

Sankod	Pareiga	Kiekis	Mėnesinis
vadspe	Gamybos vadovas	1	1.000,00
vadspe	Mecanikas	1	800,00
vadspe	Meistras	3	750,00
vadspe	Technologas	2	800,00

25 Pav. Forma „Vadovai ir specialistai”

DBGrid laukų vardai:

- Sąnaudos kodas;
- Pareigos;
- Kiekis;
- Mėnesinis darbo užmokesčiai;
- Premija %;

Forma „Vadovai ir specialistai” turi sąryši su sekančiais formomis:

- Gaminio savikaina;
- Gamybos duomenys;
- Pagrindiniai duomenys.

Pasirinkus mygtuką „Darbininkai”, pagrindiniame lange atsidaro forma „Darbininkai”

The screenshot shows a software window titled "Darbininkai". Inside, there is a table with the following data:

Sankod	Specialybe	Kategorija	Kiekis	Val. atlygis	Premi
darbpg	Saltkalvis1	3	2	2,00 Lt	
darbpg	Saltkalvis2	4	3	2,10 Lt	
darbpg	Surinkejas1	3	12	2,00 Lt	
darbpg	Surinkejas2	4	10	2,05 Lt	
darbpg	Suvirintojas	4	2	2,10 Lt	
darbpl	Frezuotojas	4	1	2,10 Lt	
darbpl	Tekintojas	4	1	2,10 Lt	

Below the table are five buttons: "Papildyti", "Ištrinti", "Redaguoti", "Išsaugoti", and "Baigti". To the right of the table are three buttons: "jeina į Gaminio savikainą", "dirba Gamyboje", and "skaičiuoja iš Pagrindinių duomenų".

26 Pav. Forma „Darbininkai”

DBGrid laukų reikšmės yra šios:

- Sąnaudos kodas;
- Specialybė;
- Kategorija;
- Kiekis;
- Valandinis atlygis;
- Premijos dydis %.

Ši forma turi sąryšį:

- Gaminio savikaina;
- Gamybos duomenys;
- Pagrindiniai duomenys.

Pasirenkame pagrindiniame lange mygtuką „Medžiagos”, patenkame į formą „Pagrindinės medžiagos” (27 pav.)

Formos laukų vardai:

The screenshot shows a software interface for managing materials. At the top, there are two buttons: 'Tiekimas' and 'ieina i Gaminio savikaina'. Below them is a table titled 'Medžiagos' with the following data:

sankod	Pavadinimas	Svoris
pmedziagos_245	Chemines medziagos_	4,5
pmedziagos_245	Juodeji metalai_245	15,3
pmedziagos_245	Polimerai ir plastmases	15,2
pmedziagos_245	Spalvoti metalai_245	6,3

At the bottom of the window, there are five buttons: 'Papildyti', 'Ištrinti', 'Redaguoti', 'Išsaugoti', and 'Baigti'.

27 Pav. Forma „Pagrindinės medžiagos”

- Sąnaudos kodas;
- Medžiagų pavadinimas;
- Svoris;
- Kaina;
- Atliekamų medžiagų norma;
- Atliekamų medžiagų kaina.

Forma „Pagrindinės medžiagos” susiriša su formomis:

- Gaminio savikaina;
- Tiekimas.

Pasirenkame formą „Komplektuojami gaminiai” (28 pav.)

Formos DBGrid laukų pavadinimai:

- Sąnaudos kodas;
- Gaminio, detalių pavadinimai;
- Kiekis;
- Kaina.

Forma „Komplektuojami gaminiai” turi sąryšį:

- Gaminio savikaina;
- Tiekimas.

Komplektuojami_gaminiai

Tiekimas įeina į Gaminio savikainą

Komplektuojami gaminiai				
	Sankod	Pavadinimas	Kiekis	Kaina
▶	kompagam_245	Briauna	2	2,92 L
▶	kompagam_245	Kompresorius	1	129,00 L
▶	kompagam_245	Saldymo agentas	1	2,85 L
▶	kompagam_245	Vamzdelis	1	3,82 L
▶	kompagam_245	Vamzdelis 8FJA	1	7,18 L
▶	kompagam_245	Vamzdelis 8F.IA175 18	2	2,57 L

Papildyti Ištrinti Redaguoti Išsaugoti Baigti

28 pav. forma „Komplektuojami gaminiai”

Pasirenkame mygtuką „Įrangos”, atsidaro forma „Įrangos“ (29 pav.):

Įrangos

Įrangos				
	Irkodas	Pavadinimas	Gammetai	Kiekis
▶	mazverthel	Helis ballionose	2002	2
▶	mazvertmo	Movos	1998	45
▶	mazvertmov	Movos	1992	50
▶	mazvertpn	Pneumoaparatai	1982	4
▶	mazvertpne	Pneumoaparatai	1975	3
▶	mazverttik	Tikrinimo iranga	1991	4

Įrangos aptarnavimas

naudojasi Energija

skaičiuoja is Pagrindinių duomenų

įeina į Gaminio savikainą

Papildyti Ištrinti Redaguoti Išsaugoti Baigti

29 pav. . forma „Įrangos”

Formos „Įrangos” laukų vardai yra:

- Įrangos kodas;
- Pavadinimas;
- Gaminimo metai;
- Kiekis;
- Kaina;
- Amortizacijos norma;

- Materialinė norma;
- Kitos normos;
- Paklausos koeficientas;
- Įrangos galingumas;
- Apkrovimo koeficientas.

Forma „Įrangos“ susiriša su kitomis formomis:

- Įrangos aptarnavimas;
- Energija;
- Gaminio savikaina;
- Pagrindiniai duomenys.

Pasirenkame mygtuką „Pastatai“ pagrindiniame lange, atsidaro forma „Pastatai“ (30 pav.):

Pastatai				
Sankod	Pavadinimas	Gammetai	Plotas	Aukst
▶ pastsan	buitinis	1963	486,5	
pastsan	gamybinis	1963	2900,6	
pastsan	sandeliai	1963	223,9	

30 pav. Forma „Pastatai“

DBGrid laukų kiti pavadinimai:

- Sąnaudos kodas;
- Pastatų pavadinimas;
- Gaminimo metai;
- Pastatų plotas;
- Pastatų aukštis;
- Kaina, 1m³;
- Amortizacijos norma;
- Materialinė norma;
- Kitos normos;
- Apšildymo norma;

- Normatyvai.

Formos sąryšis su kitomis formomis:

- o Energija
- o Pastatų aptarnavimas;
- o Pagrindiniai duomenys.

Forma „Gamybos duomenys” (31 pav.) taip pat yra sąryšių forma.

The screenshot shows a window titled "Pagrindinis_duomenys" with a table of production data. The table has four columns: "Gaminio pavadinimas", "Svoris", "Apimtis", and "Apšil". The data rows are:

Gaminio pavadinimas	Svoris	Apimtis	Apšil
F245_1503B	46	30000	
RF270_1503A	63	50000	
RF315_1503A	63	60000	

Below the table are five buttons: "Papildyti", "Ištrinti", "Redaguoti", "Išsaugoti", and "Baigti". To the right of the table are four buttons: "Lentele:Įrangos", "Lentele:Vadovai ir specialistai", "Lentele:Pastatai", and "Lentele:Darbininkai".

31 pav. Forma „Gamybos duomenys”

Fomos DBGrid laukų pavadinimai:

- Gaminio pavadinimas;
- Gaminio svoris;
- Gamybos apimtis;
- Apšildymo trukmė;
- Apšvietimo trukmė;
- Elektros energijos kaina;
- Suspausto oro kaina;
- Vandens kaina;
- Faktinis darbo laikas
- Efektyvus darbo laikas;
- Darbo pamainų skaičius;
- Atskaitymai „Sodrai“.

Sąryšis su kitomis formomis:

- o Įrangos;
- o Pastatai;
- o Vadovai ir specialistai;
- o Darbininkai.

Vartotoją – buhalterį supažindiname su tuo, kad esant reikalui pakeistų duomenis, tai reikia atlikti būtent šios formos DBGrid laukuose. Formose „Įrangos“, „Pastatai“, „Darbininkai“, „Vadovai ir specialistai“ reliacinės duomenų bazės taisyklės neleis tai padaryti. Žemiau pateiksime pavyzdį kaip pakeisti, **redaguoti duomenis**.

1. Mygtukų „Gamybos duomenys“ pasirenkame formą „Gamybos duomenys“
2. Paspaudžiame mygtuką „Redaguoti“, suaktyvėja įrašas: „Pasirinkite įrašą, redaguokite, po to spauskite mygtuką: „Išsaugoti“ (32 pav.)
3. Pasirenkame įrašą DBGrid laukose, kuri redaguosime, pav. Lauke „Šilenkaina“ vietoje „2,90 Lt“ įvedame kitą skaičių.
4. Užrašus kitą skaičių, spaudžiame mygtuką „Išsaugoti“.
5. Pasirenkame mygtuką „Baigti“ arba mygtuku baigiame redagavimą.

Gamybos duomenys				
	Svoris	Apimtis	Apšildtrukme	Šilenkaina
▶	46	30000	4320	2,90 Lt
	63	50000	4320	2,90 Lt
	63	60000	4320	2,90 Lt

Pasirinkite įrašą, redaguokite, po to spauskite mygtuką :Išsaugoti

Papildyti Ištrinti Redaguoti Išsaugoti Baigti

32 pav. Forma „Gamybos duomenys“ redagavimo režime

Vartotojas – buhalteris taip pat turi teisę įvesti **naują įrašą**, trinti nereikalingus įrašus. Žemiau pateiksime pavyzdį, kaip buhalteris gali įvesti naują įrašą visa eilute DBGrid laukuose.

1. Pasirenkame mygtuką pav. „Komplektuojami gaminiai“
2. Atsidarius langui „Komplektuojami gaminiai“, spaudžiame mygtuką „Papildyti“.
3. Lange „DB papildymas“ (20 pav.) pasirenkame mygtuką „OK“.
4. Žymeklis atsistoja naujame DBGrid lentelių eilutės pirmajame stulpelyje (33 pav.)
5. Lauke „Sankod“ įvedame pirmąjį įrašą, po to užpildome eilės tvarka visą DBGrid lauko eilutę. Perėjimas tarp stulpelių vykdomas pele, klaviatūra „Enter“ arba žymeklio stumdymo per vieną poziciją klavišais.
6. Užpildęs paskutinį įrašo lauką spaudžiame mygtuką „Išsaugoti“.
7. Mygtukų „Baigti“ arba mygtuku uždarome langą.

Komplektuojami gaminiai

Sankod	Pavadinimas	Kiekis	Kaina
kompgam_360	Saldymo agentas	1	4,15 L
kompgam_360	Vamzdelis	2	3,82 L
kompgam_360	Vamzdelis 8FJA175.18	1	7,18 L
kompgam_360	Vamzdelis 8FJA175.20	1	2,57 L
*			

33 pav. Naujo įrašo įvedimo pavyzdys

Pastaba: Jei duomenų laukų reikšmės neužpildytos arba užpildytos neteisingai, duomenys nebus išsaugoti. Tam tikslui sistema siunčia pranešimus. Mūsų pavyzdyje laukas „Pavadinimas” identifikuojantis laukas - jo reikšmė negali kartotis. Lauko „Kiekis” ir „Kaina” reikšmė negali būti ≤ 0 .

5.2.2.4 Rezultatai

Formos, naudojamos duomenų peržiūrėjimui. Duomenys įvesti komponente „Duomenų baze” čia apdorojami ir išvedami į ekraną. Vartotojas duomenų apdorojimui negali turėti įtakos; gali peržiūrėti visą arba dali informacijos.

Pastaba: Duomenų saugumo sumetimais, kol neuždarytos komponento „Duomenų baze” formos, neatsidarys komponentas „Rezultatai”.

Pagrindiniame sąsajos lange paspaudus mygtuką „Rezultatai” atsidaro forma „Rezultatai” (34 pav.) Šia sąsaja gali naudotis visi vartotojai. Čia duomenų bazės užklausos rezultatai sugrupuojami į rėmelius „Medžiagos”, „Darbo užmokesčiai”, „Įrangos sąnaudos”, „Pastatų sąnaudos” vartotojui patogia forma. Vartotojas, išsirinkęs norimą mygtuką, gali peržiūrėti atskirus arba bendrai suskaičiuotus rezultatus.

Pavyzdžiui, „Įrangos sąnaudos” paskirstytos į grupes:

- Įrangos amortizacija;
- Įrangos energetinės sąnaudos;
- Įrangos materialines sąnaudos;

- Įrangos kitos materialinės sąnaudos;
- Bendros įrangos sąnaudos.

„Darbo užmokesčiai”, „Įrangos sąnaudos”, „Pastatų sąnaudos” rezultatai apjungti į bendrą užklausą „Darbo užmokesčiai, įrangos ir pastatų sąnaudos”.

Bendra suma suskaičiuoti rezultatai pateikti formoje „Gaminio savikaina (bendra)”.

Vartotojas į duomenų bazę sugrįžta mygtuku „Atgal į Duomenų Bazę”

The screenshot shows a window titled "Rezultatai" with a light blue background. It contains three main panels:

- Medžiagos (Materials):**
 - Pagrindinės medžiagos
 - Komplektuojami gaminiai
- Įrangos sąnaudos (Equipment Expenses):**
 - Įrangos amortizacija
 - Įrangos energetinės sąnaudos
 - Įrangos materialines sąnaudas
 - Įrangos kitos materialinės sąnaudos
 - Bendros įrangos sąnaudos
- Darbo užmokesčiai (Labor Costs):**
 - Darbininkų darbo užmokesčiai
 - Vadovų specialistu darbo užmokesčiai

At the bottom of the window, there are three buttons:

- Darbo užmokesčiai, įrangos ir pastatų sąnaudas
- Gaminio savikaina (bendra)** (highlighted with a dotted border)
- Atgal į Duomenų Bazę

34 pav. Forma „Rezultatai”

Pasirinkęs mygtuką „Pagrindinės medžiagos”, atsidaro langas „Pagrindinės medžiagos” (35 pav.) Vartotojas šiame lange gali:

- Su mygtuku „Medžiagų kainos skaičiavimas” peržiūrėti medžiagų kainos sumą.
- Su mygtuku „Bendra suma” peržiūrėti bendros medžiagų kainos rezultatus.

Pastaba: Bendra suma = Medžiagų kaina – atliekų kaina.

- „Baigti” mygtukų išeiti iš formos.

Medžiagų sąnaudos					
Medpav	Svoris	Kaina	Medzagu kaina	Atnorma	
Chemines medziagos_245	4,5	9,85 Lt	44,33 Lt	0,01	
Juodeji metalai_245	15,3	3,10 Lt	47,43 Lt	0,15	
Polimerai ir plastmasės_245	15,2	4,12 Lt	62,62 Lt	0,01	
Spalvoti metalai_245	6,3	3,80 Lt	23,94 Lt	0,15	
Tara ipakavimui_245	4,8	2,95 Lt	14,16 Lt	0,1	
Chemines medziagos_270	6,5	9,85 Lt	64,03 Lt	0,01	
Juodeji metalai_270	23	3,10 Lt	71,30 Lt	0,15	
Polimerai ir plastmasės_270	24,2	4,12 Lt	99,70 Lt	0,01	
Spalvoti metalai_270	8,3	3,80 Lt	31,54 Lt	0,15	
Tara ipakavimui_270	6,9	2,95 Lt	20,36 Lt	0,1	
Chemines medziagos_315	6,5	9,85 Lt	64,03 Lt	0,01	
Juodeji metalai_315	22,45	3,10 Lt	69,60 Lt	0,15	
Polimerai ir plastmasės_315	24,5	4,12 Lt	100,94 Lt	0,01	
Spalvoti metalai_315	8,6	3,80 Lt	32,68 Lt	0,15	

Medžiagų kainos skaičiavimas Bendra suma 47371240.26410 Lt Baigti

35 pav. „Pagrindinės medžiagos“

Pasirinkus mygtuką „Darbininkų darbo užmokesčiai“, atsidaro rezultatų langas „Darbininkų darbo užmokesčiai“ (36 pav.)

Darbininkų darbo užmokestis				
Specialybė	Kategorija	Kiekis	Valandinis	Efu
Saltkalvis2	4	3	2,10 Lt	
Surinkejas1	3	12	2,00 Lt	
Surinkejas2	4	10	2,05 Lt	
Suvirintojas	4	2	2,10 Lt	
Frezuotojas	4	1	2,10 Lt	

Pagrindinis darbo užmokestis Atskaitymai Sodrai Bendra suma 256383.64420 Lt Baigti

36 pav. Darbininkų darbo užmokesčiai

Vartotojas gali peržiūrėti rezultatus:

- Pagrindiniai darbo užmokesčiai;
- Atskaitymai „Sodrai“;
- Bendra darbo užmokesčio suma.

Paspaudus mygtuką „Bendros įrangos sąnaudas”, atsidaro forma „Įrangos bendra”.

The screenshot shows a window titled "Įrangos bendra" with a table titled "Bendros įrangos sąnaudos". The table has four columns: "Irpavard", "Amortsuma", "Energsuma", and "Matsum". Below the table, there is a "Bendra suma" button, a text box containing "254466.69710 Lt", and a "Baigti" button.

Irpavard	Amortsuma	Energsuma	Matsum
► Helis ballionose	4,52 Lt	370,88 Lt	6,76
Movos	90,00 Lt	6.954,02 Lt	135,00
Movos	120,00 Lt	7.726,69 Lt	120,00
Pneumoaparatai	256,00 Lt	1.483,52 Lt	96,00
Pneumoaparatai	81,00 Lt	1.205,36 Lt	27,00
Tikrinimo iranga	800,00 Lt	1.854,40 Lt	240,00
Vakuumsiuurbliai	18.900,00 Lt	4.238,64 Lt	10.800,00
Vakumsiuurbliai	16.200,00 Lt	4.238,64 Lt	10.800,00
Nutekejimo tikrinimo ira	4.000,00 Lt	529,83 Lt	1.600,00
Ultragarsinis suvirinimo	8.500,00 Lt	529,83 Lt	3.400,00
Uzpildymo irenginys	15.000,00 Lt	678,18 Lt	4.000,00

Bendra suma 254466.69710 Lt **Baigti**

37 pav. Forma „ Įrangos bendra”

Čia pateikti rezultatai:

- Įrangos amortizacijos suma;
- Įrangos energetinių sąnaudų suma;
- Įrangos materialinių sąnaudų suma;
- Įrangos kitos sąnaudos;
- Bendra suma.

Mygtukas „Bendros pastatų sąnaudos” atidaro forma „Pastatų sąnaudos”

The screenshot shows a window titled "Pastatų sąnaudos" with a table titled "Bendros pastatų sąnaudos". The table has five columns: "Pastpavard", "Amortsuma", "Matsuma", "Kitasuma", and "A". Below the table, there is a "Pastatų sąnaudos(bendra)" button, a text box containing "88445.58960 Lt", and a "Baigti" button.

Pastpavard	Amortsuma	Matsuma	Kitasuma	A
► buitinis	2.298,71 Lt	1.103,38 Lt	735,59 Lt	
gamybinis	24.225,81 Lt	12.112,91 Lt	9.690,33 Lt	
sandeliai	1.870,01 Lt	935,01 Lt	748,01 Lt	

Pastatų sąnaudos(bendra) 88445.58960 Lt **Baigti**

38 pav. Forma „Pastatų sąnaudos”

Čia pateikti rezultatai:

- Pastatų amortizacijos suma;
- Pastatų materialinių sąnaudų suma;
- Pastatų kitos sąnaudos;

- Pastatų apšvietimo sąnaudos;
- Pastatų apšildymo sąnaudos;
- Bendra suma.

Mygtukas „Darbo užmokesčiai, įrangos ir pastatų sąnaudos” atidaro formą:

Darbo užmokesčiai, įrangos ir pastatų sąnaudas			
Pavadinimas	Visodarbininku_suma	Visovadovu_suma	Irangsuma
Pagrindiniu darbininku darb	97.360,88 Lt		
Pagrindiniu darbininku darb	25.557,23 Lt		
Pagrindiniu darbininku darb	16.226,81 Lt		
Pagalbiniu darbininku darbc	8.519,08 Lt		
Pagalbiniu darbininku darbc	8.519,08 Lt		
Irangos sanaudas			20.460,61 Lt
Irangos sanaudas			10.418,69 Lt
Irangos sanaudas			27.535,36 Lt
Irangos sanaudas			8.199,66 Lt
Irangos sanaudas			27.678,18 Lt
Irangos sanaudas			19.229,83 Lt
Irangos sanaudas			9.329,83 Lt
Irangos sanaudas			52.838,64 Lt
Irangos sanaudas			55.538,64 Lt
Irangos sanaudas			3.534,40 Lt
Irangos sanaudas			1.385,36 Lt
Irangos sanaudas			2.091,52 Lt
Irangos sanaudas			8.286,69 Lt
Irangos sanaudas			7.539,02 Lt
Irangos sanaudas			400,26 Lt
Pastatu sanaudas			
Pastatu sanaudas			
Pastatu sanaudas			

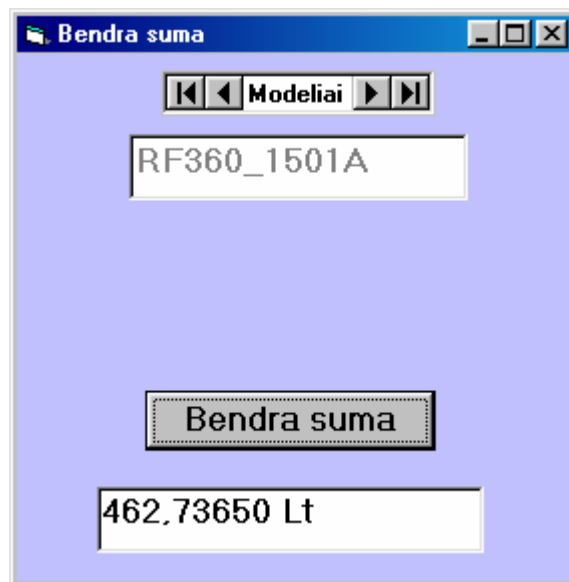
Suma 701436.63180 Lt Baigti

39 pav. Forma „Darbo užmokesčiai, įrangos ir pastatų sąnaudos”

Čia pateikiami rezultatai:

- Vadovų ir specialistų darbo užmokesčiai;
- Darbininkų darbo užmokesčiai;
- Bendros įrangos sąnaudos;
- Bendros pastatų sąnaudos;
- Bendra suma.

Pasirinkęs mygtuką „Gaminio savikaina (bendra)”, vartotojas atidaro formą „Bendra suma”. Čia pateikiami bendri rezultatai, gaminio savikaina.



40 pav. Forma „Bendra suma”

Objekto Data „Modeliai” mygtukais vartotojas pasirenka modelio pavadinimą, po to paspaudžia komandinį mygtuką „Bendra suma”. Tekstiniame lauke pasirodo bendra suma. Pav. „462,73650 Lt”.

5.2.2.5 Grafika

Formos, naudojančios klasę Form ir jos sąsają su OLE (*Object Linking and Embedding*) protokolu. OLE protokolas leidžia dviem programoms automatiškai keistis duomenimis. Šis protokolas leidžia vartotojui be specialaus pasirengimo peržiūrėti ir tvarkyti reikalingus objektus, parengtus „Microsoft Office” programų paketais „Word”, „Excel”, „Power Point” t.t.

Mes naudosime „Microsoft Office” programų paketo „Excel” sąsają su OLE protokolu. „Excel” skaičiuoklė yra puiki diagramų kūrimo priemonė, kuri vaizdžiai pateikia grafinę informaciją vartotojui.

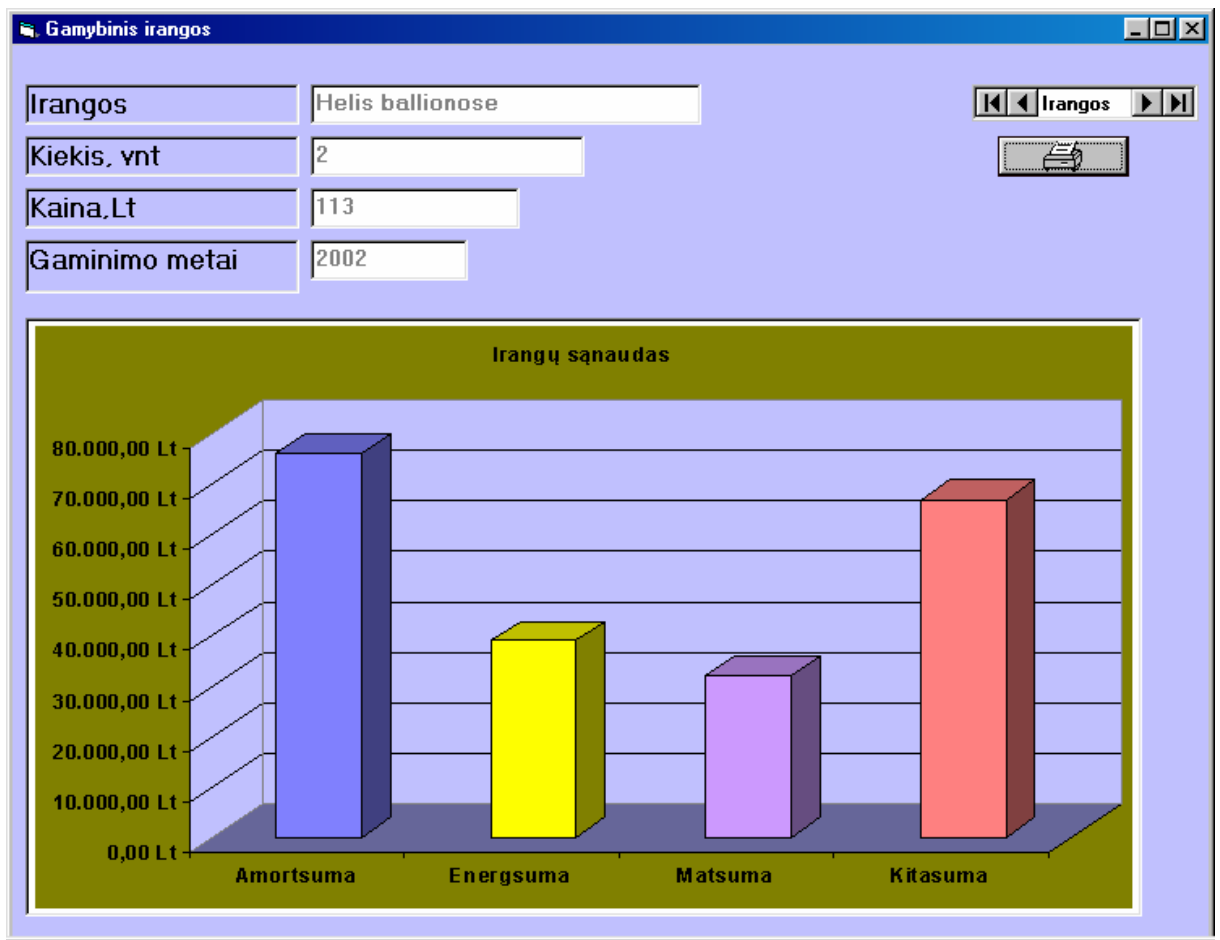
Pagrindiniame sąsajos lange pasirenkame komandinį mygtuką „Grafikos langas”, atsidaro forma „Grafika” (41 pav.) Šiame lange išdėstyti taip pat mygtukai: „Įrangos”, „Pastatai”, „Detalės”, „Medžiagos”, „Darbuotojai”, „Atgal į duomenų bazę”.

Mygtukas „Atgal į duomenų bazę” uždaro formą, vartotojas grįžta į pagrindinį sąsajos langą. Likę mygtukai atidaro naujas formas pagal paskirtį.



41 pav. Forma „Grafika”

„Grafikos” lange (41 pav.) pele pasirenkame mygtuką „Įrangos”. Atsidaro forma „Gamybos įrangos”.



42 pav. Forma „Gamybos įranga

Šiame lange pateikiama informacija apie:

- Įrangos pavadinimą;
- Kiekį, vnt;
- Kainą, Lt

- Gaminimo metus.

Data objekto „Įrangos” mygtukų grupės reikšmės:

◀ - Vienu įrašu atgal;

▶ - Vienu įrašu pirmyn;

◀◀ - Į pirmą įrašą;

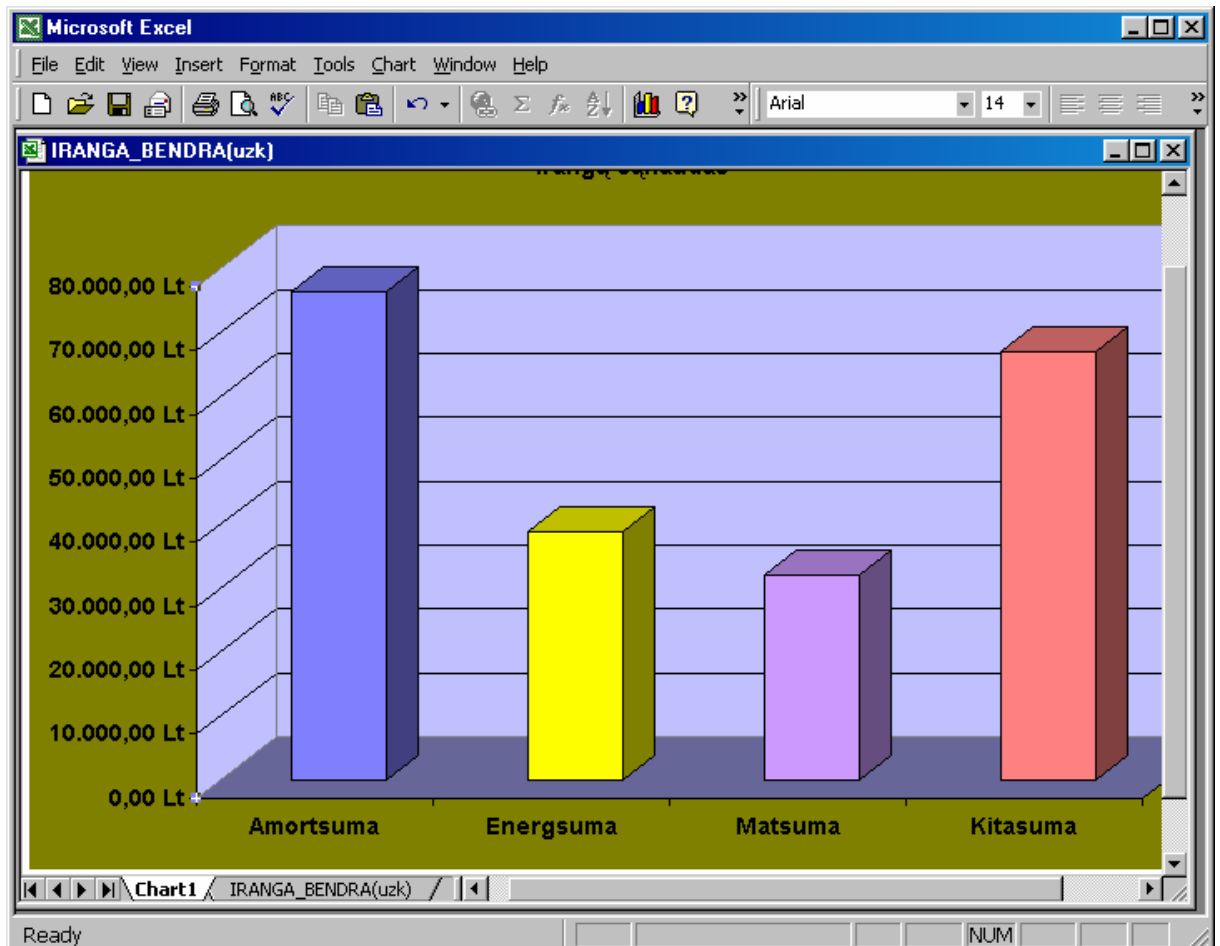
▶▶ - Į paskutinį įrašą.



- šis mygtukas leidžia atspausdinti formą.

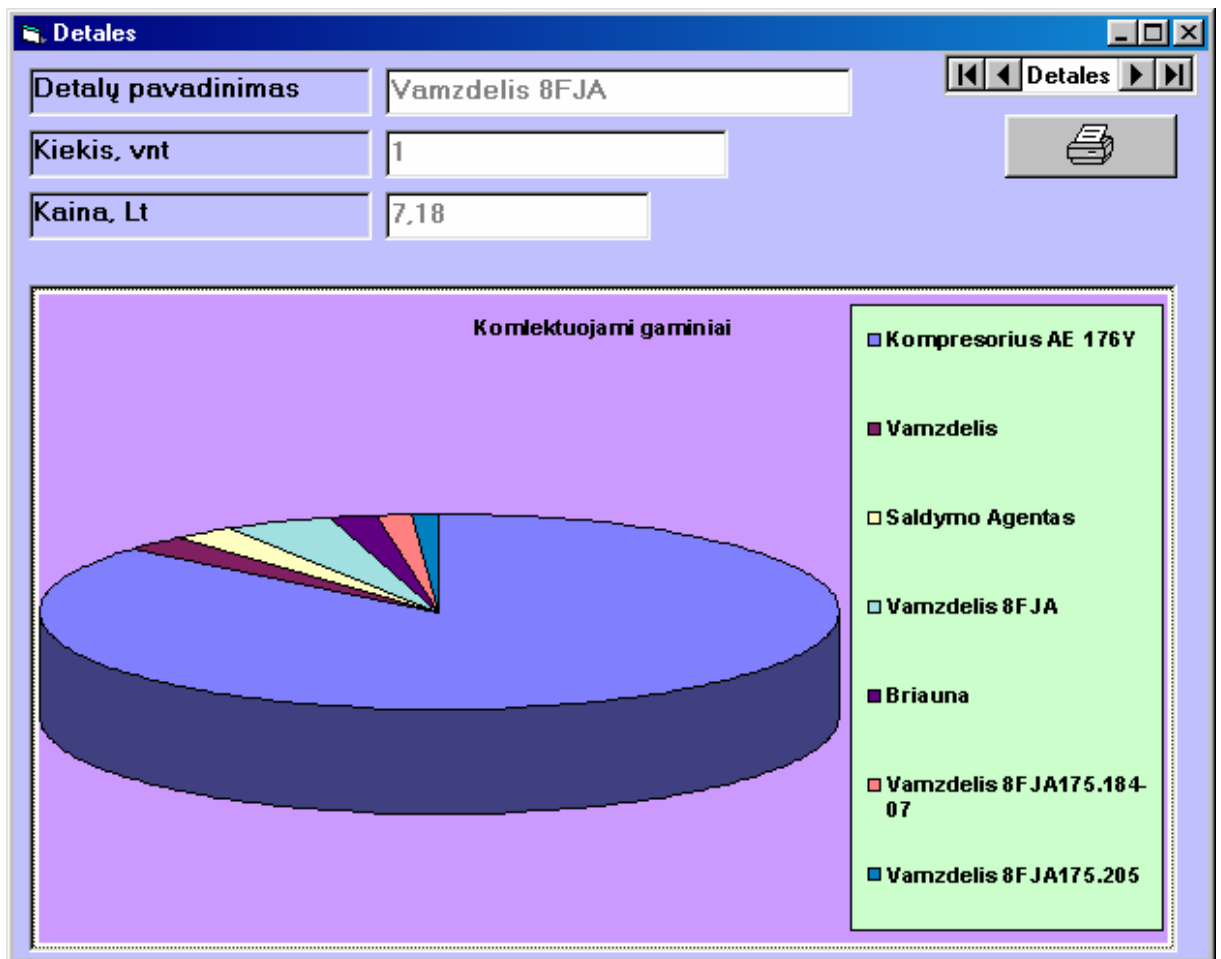
Diagramą „Įrangos sąnaudos” vartotojas gali atidaryti su „Excel” skaičiuokle. Tam tikslui reikia ant diagramos paspausti dešinę peles mygtuką, atvertame kontekstiniame meniu pasirinkti „Open” komandą.

Diagramą galima atidaryti taip pat „Excel” lentelių pavidalu, tam reikia „Excel” būsenų eilutėje perjungti mygtuką „Įranga_bendra(uzk)” (43 pav.)



43 pav. „Excel” diagrama

Vartotojui pasirinkus mygtuką „Detalės”, atsidaro forma „Detalės”.



44 pav. forma „Detalės”

Šiame lange pateikiama informacija apie:

- Detalių pavadinimą;
- Kiekį, vnt.;
- Kainą, lt.

Kontekstiniame meniu pasirenkame „Open” komandą, perjungiamo mygtuką „Kompgam_šnaudas” „Excel” būsenų juostoje.

Žemiau pateikiame diagramą „Excel” lentelių pavidalu. (45 pav.)

	A	B	C	D	E
1	Sanaudos kodas	Pavadinimas	Kiekis_vnt	Kaina_Lt	Visakaina
2	kompgam	Kompresorius AE 176Y	1	138,000 Lt	11.040.000,00 Lt
3	kompgam	Vamzdelis	1	3,82000 Lt	305.600,00 Lt
4	kompgam	Saldymo Agentas	1	4,13000 Lt	330.400,00 Lt
5	kompgam	Vamzdelis 8FJA	1	7,18000 Lt	574.400,00 Lt
6	kompgam	Briauna	1	2,92000 Lt	233.600,00 Lt
7	kompgam	Vamzdelis 8FJA175.184-07	1	2,57000 Lt	205.600,00 Lt
8	kompgam	Vamzdelis 8FJA175.205	2	,8000 Lt	128.000,00 Lt
9					
10					
11					
12					
13					
14					
15					
16					
17					
18					
19					
20					
21					
22					
23					

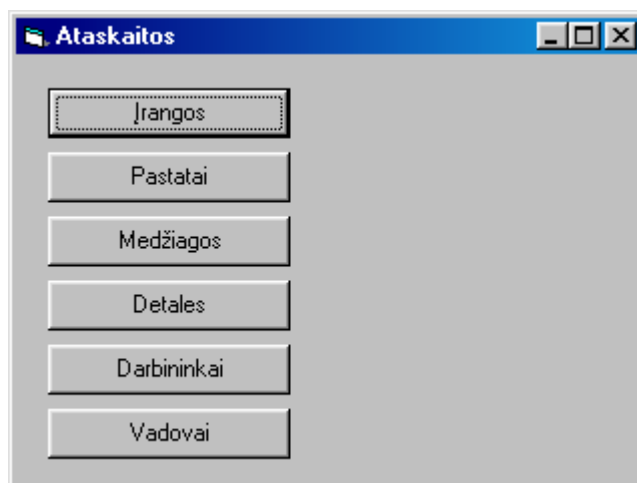
45 pav. „Excel” lentelė

5.2.2.6 Ataskaitos

Kiekvienai ataskaitai sukuriamas atskiras failas, kuriame, naudojantis klasės Reports metodais, suformuojama ataskaita. Tai suvestiniai duomenys apie:

- Bendras įrangos sąnaudas;
- Bendras pastatų sąnaudas;
- Medžiagų sąnaudas;
- Gaminio, detalių sąnaudas;
- Vadovų, specialistų darbo užmokesčius;
- Darbininkų darbo užmokesčius.

Pagrindiniame lange pasirinkęs mygtuką „Ataskaitų langas”, vartotojas atidaro formą „Ataskaitos” (46 pav.):



46 pav. Forma „Ataskaitos”

Paspaudę mygtuką „Detalės”, atidarome ataskaitą „Detalės” (47 pav.)

Pavadinimas	Kiekis:	Kaina:	Bendra suma:
Briauna	2	2,92 Lt	175.200,00 Lt
Kompresorius	1	129,00 Lt	3.870.000,00 Lt
Saldymo agentas	1	2,85 Lt	85.500,00 Lt
Vamzdelis	1	3,82 Lt	114.600,00 Lt
Vamzdelis 8FJA	1	7,18 Lt	215.400,00 Lt
Vamzdelis 8FJA175.184-07	2	2,57 Lt	154.200,00 Lt
Briauna	1	2,92 Lt	146.000,00 Lt

47 pav. Ataskaita „Detalės”



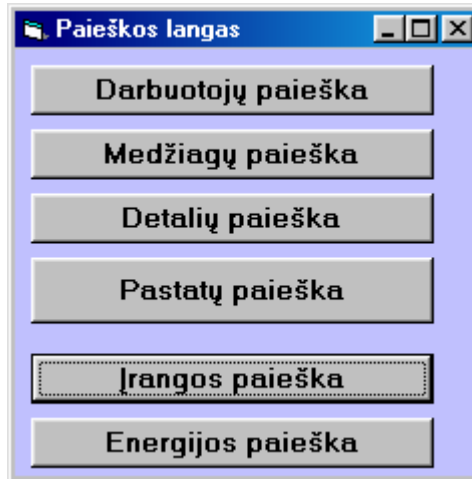
- mygtukas, skirtas spausdinti ataskaitas.



- mygtukas, skirtas eksportuoti ataskaitą HTML formatu.

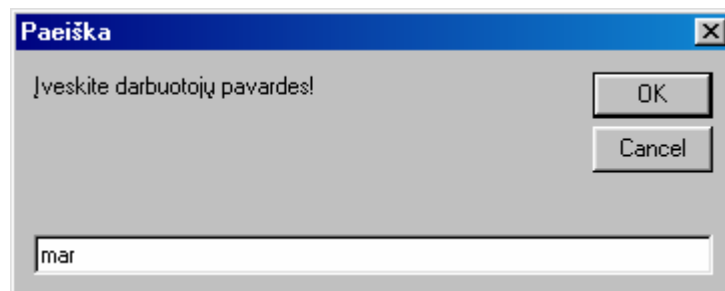
5.2.2.7 Pagalba

Šiame skyriuje trumpai paaiškinsime, kaip naudotis pagalbos langu. Vartotojas, pasirinkęs mygtuką „Pagalbos langas“, patenka į formą „Paieškos langas“. Čia vykdomas darbuotojų, medžiagų, detalių, pastatų, įrangos, energijos paieška (48 pav.)



48 pav. „Paieškos langas“

Pasirenkame, pavyzdžiui, mygtuką „Darbuotojų paieška“. Lange „Paieška“ (49 pav.) įvedame darbuotojų pavardę, spaudžiame mygtuką „OK“



49 pav. „Paieška“

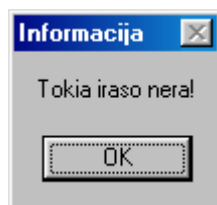
Jei sistema randa darbuotoją, jį parodo DBGrid lentelė (50 pav.)



Askod	Pavardė	Specialybė	Op.pav
0030	Adamaitene Roma	Surinkejas2	Kronšteinu prisu
0057	Adamaitis Valdas	Surinkejas1	Kronšteinu tvirtin
0011	Cizius Antanas	Gamybos vadovas	Cecho vadovas
0020	Juozaitis Kestas	Saltkalvis2	Duru prisukimas
0013	Krasauskas Rimas	Meistras	Pamainos meis
0032	Martinaitene Rita	Saltkalvis2	Duru prisukimas
0065	Mironovas Vitas	Frezuotojas	Frezavimo darb
0014	Muraska Algis	Meistras	Pamainos meis
0015	Petraitis Pranas	Saltkalvis1	Briaunu prisukin

50 pav. Paieškos rezultatas

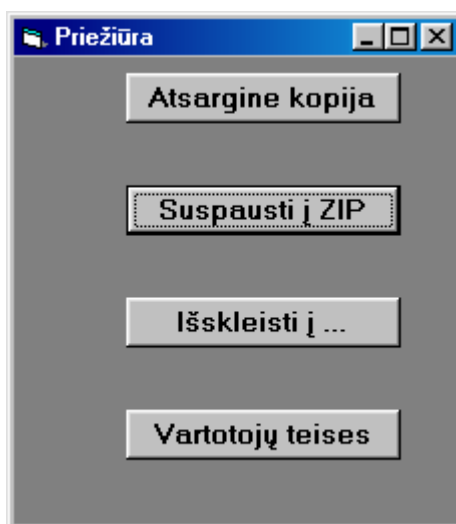
Jei sistema neranda darbuotojo, siunčia pranešimą (51 pav.)



51 pav. Pranešimas

5.2.2.8 Sistemos priežiūra

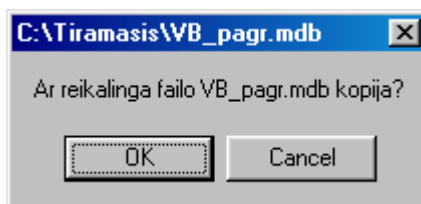
Mygtukas „Sistemos priežiūra“ atidaro forma „Priežiūra“, skirtos tik sistemos administratoriui. Sistemos administratorius įvedęs vartotojo vardą ir slaptažodį, patenka į forma „Priežiūra“:



52 pav. Forma „Priežiūra“

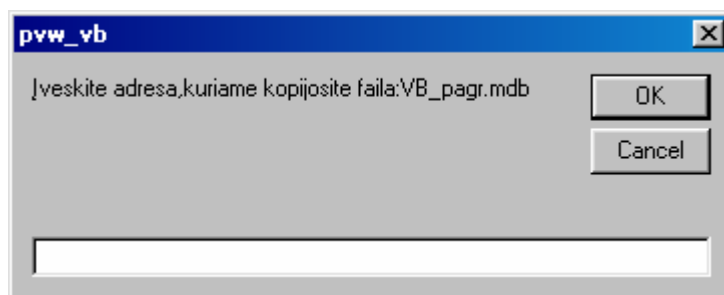
Atsargine kopija. Atsarginės duomenų kopijas daromos operacinės sistemos priemonėmis. Rekomenduojama įsitikinti, kad su sistema tuo metu niekas nedirba. Tuo įsitikinti galima patikrinus katalogą, kuriame yra programos failai. Jame neturėtų būti failo Vb_pagr.ldb.

Vartotojas paspaudes mygtuką „Atsargine kopija“ gauna pranešimą:



53 pav. Sistemos pranešimas

Jei pasirenkame mygtuką „OK“, sistema prašo įvesti adresą kuriame kopijuosime:



54 pav. Sistemos pranešimas

Įvedęs adresą failas bus kopijuotas nurodytu adresu.

Suspausti į ZIP ir išskleisti... Duomenų bazės ir programos failų suspaudimas bei išskleidimas Šį veiksmą gali atlikti tik administruojantys vartotojai, tai yra vartotojai, priklausantys grupei “admins” nurodant savo vartotojo vardą bei slaptažodį. Plačiau apie parašyta 4.4.1 skyrelyje “Sistemos priežiūra”.

Vartotojų teises. Sukurti, panaikinti, ar pakoreguoti vartotojus bei jų teises galima prisijungus prie sistemos panaudojant administruojantiems vartotojams skirtą nuorodą, bei nurodžius “admins” grupei priklausantį vartotojo vardą bei slaptažodį. Plačiau apie tai parašyta 4.4.2.2 skyrelyje “Grupių ir vartotojų tvarkymas”.

5.3 Sistemos instaliavimo dokumentas

Šiame skyriuje aprašoma sistemos paruošimo darbui vietiniuose bei nutolusiuose kompiuteriuose instrukcija, reikalinga papildoma programinė įranga bei minimalūs reikalavimai aparatinei įrangai.

Minimalūs techniniai įrangos reikalavimai yra šie:

- IBM PC tipo personalinis kompiuteris. Procesorius Celeron 450MHz, 2 GB HDD, 128 MB RAM, klaviatūra, pelė;
- Spalvotas monitorius (800x600 skiriamoji geba);
- Microsoft Access 2000;
- MS Visual Basic 6;
- Operacinė sistema: Microsoft Windows 98, NT, 2000, XP;
- Laisvos vietos diske: ne mažiau 10 MB.

AB “Snaigė” buhalterijos skyriuje suinstaliuota Microsoft Office 2000 paketas su MS Access 2000 programa. Galima naudoti ir aukštesnes Microsoft Access versijas, palaikančias Access 2000 formatą.

Skyriuje reikia instaliuoti Microsoft Visual Studio paketo Visual Basic 6 programinę įrangą; kaip įdiegti šią programinę įrangą, žiūrėkite papildomai Microsoft Visual Studio instaliavimo dokumente.

Visa informacinė sistema sudaro tokie failai:

VB_pagr.mdb – šiame faile yra saugomi duomenys ir priėjimui prie duomenų naudojamos lentelės, užklausos; konvertuotas į Visual Basic 6 aplinką su Visual Data Manager.

Pvw_vb.vbp – programos kodo ir prototipo generavimo failas, sugeneruotas ProVision Workbench 3.1 modeliavimo terpėje.

Programoje taip pat yra failų su plėtiniais: *.frm, *.cls, *.frx, *.dsr, *.tmp, kurie yra kataloge „Visbasic”.

Savikaina.exe – savarankiška programa.

Norint paruošti programą darbui, reikia atlikti tokius veiksmus:

1. nukopijuoti katalogą „Tiriamasis” į diską (C:\Tiramasis);
2. sukurti nuorodą į programą.

Sukurti nuorodą vartotojams rekomenduojame tokiu būdu:

1. pažymėti failą Savikaina.exe;
2. paspausti dešinę pelės klavišą. Atsiradusiame kontekstiniame meniu pasirinkti punktą “Create Shortcut”;
3. pažymėti failo nuovadą bei paspausti dešinę pelės klavišą. Atsiradusiame kontekstiniame meniu pasirinkti punktą “Properties”
4. atsiradusiame lange pasirinkti “Shortcut” puslapį. “Target” lauke esančią informaciją turėtų atrodyti taip: „C:\TIRAMASIS\SAVIKAINA.exe”
5. Lauke “Start in” turėtų būti reikšmė " C:\TIRAMASIS”
6. paspausti “OK” mygtuką.

5.4 Sistemos administratoriaus vadovas

Šis skyrius skirtas sistemos administratoriams. Jame nurodoma kokius veiksmus reikia atlikti prižiūrint sistemą, kaip tvarkyti vartotojus ir jų teises.

5.4.1 Sistemos priežiūra

Siekiant užtikrinti nepriekaištingą sistemos funkcionavimą, reikia atlikti sistemos priežiūros darbus. Sistemos priežiūra turėtų būti atliekama periodiškai. Visą sistemos priežiūrą galima suskirstyti į šiuos darbus:

- duomenų bazės suspaudimas;

- atsarginių duomenų bazės kopijų darymas;
- duomenų bazės atstatymas iš atsarginės kopijos;
- operacinės sistemos ir kompiuterio priežiūra.

Duomenų bazės suspaudimas. Šį veiksmą galima atlikti tik prisijungus prie sistemos panaudojant administruojantiems vartotojams skirtą nuorodą. Šį veiksmą gali atlikti tik administruojantys vartotojai, tai yra vartotojai, priklausantys grupei “admins”.

Prisijungus prie sistemos, pagrindiniame lange pasirenkame mygtuką „Sistemos priežiūra”, toliau mygtuką „Suspausti į ZIP”. Po to reikalinga nurodyti failą, kuris spaudžiamas ir suspausti., Po suspaudimo sumažės duomenų bazės failo VB_pagr.mdb apimtis.

Suspaudimo proceso metu yra atliekama duomenų defragmentacija, kuri atsiranda sistemos darbo metu, pašalinami tarpiniai laikini duomenys, ištaisomos pastebėtos duomenų bazės klaidos. Todėl po suspaudimo visa sistema veiks greičiau ir stabiliau.

Prieš atliekant suspaudimo veiksmą reikia įsitikinti, kad su sistema nedirba kiti vartotojai. Suspaudimo metu duomenų bazė tampa neprieinama vartotojams.

Suspaudimo veiksmą rekomenduojame atlikti nerečiau kaip kartą per savaitę.

Atsarginių duomenų bazės kopijų darymas. Atsarginės duomenų kopijos daromos operacinės sistemos priemonėmis. Rekomenduojama įsitikinti, kad su sistema tuo metu niekas nedirba, tai yra duomenų bazė yra uždaryta. Tuo įsitikinti galima patikrinus katalogą, kuriame yra programos failai (pavyzdžiui, C:\Tiramasis). Jame neturėtų būti failo VB_pagr.ldb Šis failas yra laikinas ir sukuriamas MS Access, kai yra atidaroma duomenų bazė. Uždarius duomenų bazę, šis failas automatiškai yra panaikinamas.

Įsitikinę, kad duomenų bazė uždaryta, galime daryti atsarginę kopiją. Atsarginės kopijos daromos nukopijuojant sistemos failus į kitą diską, kitą to paties disko katalogą arba į kitą kompiuterį. Galima sistemos failų kopijas įrašyti į CD-R, magnetinius diskelius, ar kitas išorines duomenų saugojimo priemones.

Šiuose failuose yra saugomi tiek duomenys, tiek ir pati programa. Prieš darant atsarginę kopiją, rekomenduojama atlikti duomenų bazės suspaudimą. Tai sumažins failų dydžius ir leis taupyti resursus.

Atsargines duomenų ir visos sistemos kopijas rekomenduojame daryti ne rečiau kaip kartą per savaitę.

Duomenų bazės atstatymas iš atsarginės kopijos. Duomenų bazės atstatymas atliekamas operacinės sistemos priemonėmis. Rekomenduojama įsitikinti, kad su sistema tuo metu niekas nedirba, tai yra duomenų bazė yra uždaryta.

Duomenų bazės ir visos sistemos atstatymui reikalinga nukopijuoti vėliausiai darytos duomenų bazės kopijos failą VB_pagr.mdb į darbinį sistemos katalogą (pavyzdžiui, c:\Tiramasis\). Po to rekomenduojama pabandyti prisijungti prie sistemos per administruojantiems vartotojams skirtą nuorodą, nurodant “admins” grupei priklausantį vartotojo vardą bei atitinkamą slaptažodį. Prisijungus atlikti duomenų suspaudimą. Jeigu operacija baigėsi sėkmingai, galima uždaryti programą ir leisti ja naudotis paprastiesiems vartotojams.

Duomenų bazės ir visos sistemos atstatymą rekomenduojama atlikti tik sugedus naudojamai sistemai, kuomet negalima prisijungti prie sistemos, sistema funkcionuoja nestabiliai, arba gaunami klaidingi rezultatai

Operacinės sistemos ir kompiuterio priežiūra. Bet kokia programinė įranga nepriekaištingai funkcionuoti gali tik jeigu nepriekaištingai dirba kompiuterio techninė įranga bei operacinė sistema. Todėl reikalinga jas nuolat prižiūrėti. Ši priežiūra nėra šio projekto dalis, todėl detaliau nenagrinėjama.

Paminėsime keletą aspektų, kurie yra svarbūs mūsų nagrinėjamai sistemai:

- reikalinga sekti laisvą disko vietą, kad jos nepritrūktų duomenų bazei;
- užtikrinti stabilų elektros tiekimą, kad dingus elektrai nebūtų sugadintas diskinis kaupiklis;
- apsaugoti kompiuterį nuo virusų, kad nebūtų sugadinti duomenys.

5.4.2 Vartotojų ir jų grupių teisių tvarkymas

Vienas iš keliamų saugumo reikalavimų yra užtikrinti, kad vartotojas galėtų prieiti tik prie jam reikalingos informacijos ir atlikti tik jam pavestas funkcijas. Pagal atliekamas funkcijas ir reikalingą informaciją visi įmonės darbuotojai yra suskirstyti į grupes. Kiekvienai grupei yra nustatyta, kokią informaciją šios grupės nariai turi pasiekti ir kokias funkcijas turi sugebėti atlikti. Atitinkamai kiekvienam duomenų bazės bei sistemos elementui yra priskirta kokios grupės vartotojai kokius veiksmus gali atlikti su šiais elementais.

5.4.2.1 Sistemoje realizuotos grupės ir jų aprašymas

AB “Snaigė” buhalterijos skyriaus darbuotojai, pagal kompiuterizuojamos įmonės veiklos specifiką, yra suskirstyti į šias grupes:

1. Admins – administruojantys vartotojai;
2. Buchalteris – finansinius - ekonominius skaičiavimus atliekantis darbuotojai;
3. Vadovas – buhalterijos skyriaus vadovas.

Toliau pateiksime kiekvienos grupės objektų, prie kurių konkrečios grupės nariai turi priėjimą, teisių nustatymus, kurie užtikrina normalų sistemos darbą, duomenų saugumą ir patikimumą. Administruojantys vartotojai (priklausantys “admins” grupei) gali šias teises keisti pasikeitus poreikiams. Tačiau šis procesas reikalauja ypač didelio atidumo, kadangi nekorektiškai priskyrus teises, galima pažeisti sistemos saugumą bei funkcionalumą.

Vartotojų grupė Admins. Šios grupės nariais yra administruojantys vartotojai, turintys pilną priėjimą prie visų sistemos objektų. Jie gali kurti naujus sistemos komponentus, pritaikyti naujoms sąlygoms jau sukurtus, keisti grupės teises, vartotojų priklausomumą grupėms, suspausti duomenų bazę, eksportuoti duomenis į kitas sistemas ir t.t. Kadangi sistemos objektų yra labai daug, todėl jų neišvardinsime.

Vartotojų grupė Buhalteris. Šios grupės nariai atlieką pagrindinį darbą: Gerai susipažinę su sistema, dirba su duomenų bazės elementais, patikrina duomenų bazę, redaguoja, esant reikalui, keičia, įveda naujus, trina nereikalingus, senus įrašus DBGrid lentelėje. Buhalteris redaguoja grafikos elementus, t.y. dirba su MS „Excel” elektronine lentele, peržiūri rezultatus, ataskaitas. Kadangi pagrindinis tikslas yra darbas su duomenų baze, pateiksime objektų pavadinimus, su kuriais dirbs vartotojas:

- Gamybos duomenys;
- Pastatai;
- Pastatų aptarnavimas;
- Įrangos;
- Įrangos aptarnavimas;
- Darbuotojų duomenys;
- Darbininkų darbo užmokesčio sąnaudos;
- Vadovų ir specialistų darbo užmokesčio sąnaudos;
- Medžiagos;
- Komplektuojami gaminiai;
- Medžiagų ir komplektuojamų gaminių tiekimas;
- Energetinis sąnaudos;
- Gaminio savikaina.

Vartotojų grupė Vadovas. Šios grupės narys yra buhalterijos skyriaus vadovas, galintis peržiūrėti visą informaciją. Jis gali tik peržiūrėti visus rezultatus, atidaryti rezultatų,

ataskaitas, grafikos ir pagalbos langus. Kadangi sistemos objektų yra labai daug, todėl jų neišvardinsime.

5.4.2.2 Grupių ir vartotojų tvarkymas

Vartotojų grupės apima pakankamai siauras įmonės veiklos sritis. Tai yra svarbu, nes realiai su sistema dirba konkretūs vartotojai, kurie gali priklausyti vienai arba kelioms grupėms. Jeigu grupės funkcijos bus labai išplėstos, gali atsirasti situacija, kad vartotojas gaus teises, kurių jam nereikia savo darbo funkcijoms atlikti. Tai kelia pavojų duomenų saugumo principui, kad vartotojas turi prieiti tik prie jam reikalingų duomenų.

Sukurti, panaikinti ar pakoreguoti grupes galima prisijungus prie sistemos, panaudojant administruojantiems vartotojams skirtą nuorodą bei nurodžius “admins” grupei priklausantį vartotojo vardą bei slaptažodį

5.4.3 Ryšys su kitomis sistemomis

Specialūs ryšiai su kitomis sistemomis priemonės šiame projekte nenumatyti. Tačiau sistemos realizacija, reikalui esant, leidžia pasinaudoti Microsoft Access programine įranga ir eksportuoti reikiamus duomenis į kitas sistemas. Pageidaujamą duomenų formatą galima suformuoti su užklausa. Šiuos veiksmus gali atlikti bet kuris “admins” grupei priklausantis vartotojas, nors šiek tiek susipažinęs su MS Access programine įranga.

Galima prie duomenų bazės prisijungti iš kitų duomenų bazių valdymo sistemų per ODBC. Reikia nurodyti “admins” grupei priklausantį vartotoją ir slaptažodį.

6. KOKYBĖS VERTINIMAS

AB „Snaigė” gaminio savikainos apskaičiavimo informacinės sistemos kokybės vertinimo dokumentas skirtas apžvelgti kokybės faktorius ir, remiantis jais, įvertinti sukurtos informacinės sistemos kokybę.

6.1 Funkcionalumo faktoriai

Funkcionalumo faktoriai yra šie: korektiškumas, patikimumas, efektyvumas, integralumas, panaudojamumas.

Korektiškumas – tai kokybės faktorius atspindintis programos atitikimą specifikacijai ir vartotojo poreikių tenkinimą. Remiantis bandymų rezultatais, galima teikti, informacinės sistemos yra korektiškos. Buvo sukurti keli prototipai papildomų reikalavimų išgavimui bei vartotojo sąsajos pobūdžio nustatymui. Ji atitinka visus apibrėžtus reikalavimus ir visiškai tenkina vartotojo poreikius apibrėžtose srityse.

Patikimumas – tai kokybės faktorius, kuriuo apibrėžiamas numatomo funkcionalumo tenkinimas su reikalaujama tikslumu. Apibrėžtas funkcionalumas tenkinamas 100 %. Reikalaujamas tikslumas - 100 %. Skaitinių reikšmių ir rezultatų tikslumas fiksuojamas iki žemiausio piniginio vieneto ir naudojamų smulkiausių matavimo vienetų. Sistema yra apsaugota nuo nepatikimų vartotojų veiksmų:

- Įvedamų duomenų korektiškumui tikrinti nurodoma speciali tikrinimo taisyklė (*Validation Rule*) ir pranešimo tekstas (*Validation Text*). Todėl duomenų korektiškumo pažeisti per vartotojo sąsajos elementus neįmanoma.
- Duomenų bazė apsaugota nekorektiškų veiksmų. Pavyzdžiui, negalima atidaryti forma „Rezultatai”, kol atidarytas „Duomenų bazė”, per paieškos langą patekus į duomenų bazę, redagavimo mygtukai pasidaro neaktyvūs.
- Neleidžiama įvesti nekorektiškų duomenų. Jei informacinis laukas susideda tik iš skaitmenų, tai vartotojas galės įvesti tik skaitmenis, jei iš raidžių – tik raides ir t.t.

Saugumas nusako autorizuotą priėjimą prie programinės įrangos ar jos duomenų. Kiekvienas sistemos vartotojas turi tik autorizuotą priėjimą prie sistemos, jos duomenų bei veiksmų su sistema. Kiekvienas sistemos vartotojas turi prisijungimo vardą ir slaptažodį, kuriais prisijungęs prie sistemos, gali dirbti tik su jam reikalingais duomenimis ir atlikti tik jam pavestas funkcijas. Todėl saugumas vertinamas kaip aukštas.

Efektyvumas apibrėžia santykį tarp kompiuterinių resursų, reikalingų programinei įrangai funkcionuoti ir programinės įrangos įmonei teikiamos naudos. Įdiegta kompiuteryje programa užima apie 5 MB kietame diske. Papildomai užimama vieta priklauso nuo duomenų bazės dydžio. Palyginti su turimais resursais, programinės įrangos naudojami resursai nėra dideli. Ši informacinė sistema žymiai sumažina darbuotojų darbo laiko sąnaudas, skaičiuojant gaminio savikainą, padeda išvengti skaičiavimo klaidų. Įmonė jau turi įsigijusi visą reikiamą techninę ir programinę įrangą, o nauja informacinė sistema papildomų resursų nereikalauja, todėl efektyvumas vertinamas kaip aukštas.

Sąsajos paprastumas įvertinamas kaip pastangos, reikalingos išmokti dirbti su sistema, ja naudotis, išmokti paruošti duomenis bei interpretuoti rezultatus.

Kvalifikuotam vartotojui užtenka perskaityti vartotojo vadovo dokumentą bei apie 3 valandas peržiūrėti sistemą, ir jis pasiruošęs vykdyti veiklą. Pradinių duomenų paruošimas atliekamas tik vieną kartą, o po to lieka tik jais pasinaudoti. Apie galimas klaidas vartotojas yra išpėjamas.

6.2 Mobilumo faktoriai

Galima išskirti tokius mobilumo faktorius: pernešamumas, pakartotinis panaudojimas, susiejamumas.

Pernešamumas nusako pastangas, reikalingas programinės įrangos perkėlimui iš vienos techninės konfigūracijos į kitą arba iš vienos programinės aplinkos į kitą. AB "Snaigė" duomenų bazė gali būti perkeliama į kitą platformą ar į kitą programinę aplinką. Tai galima atlikti vykdant reinžineriją tiriamai duomenų bazei. Vartotojo sąsajos ir funkcionalumo perkėlimui iš MS Visual Basic į kitą programinę įrangą priemonių nėra.

Pakartotinas panaudojimas apibrėžia programos perkėlimo galimybę į kitą taikomąją aplinką. AB "Snaigė" gaminio savikainos apskaičiavimo IS yra realizuota MS Visual Basic, todėl pakartotinai panaudota gali būti tik MS Visual Basic aplinkoje. Duomenų bazę pakartotinai galima panaudoti eksportuojant jos struktūrą ir duomenis į kitą duomenų bazių valdymo sistemą.

Susiejamumas nusako pastangas, reikalingas sistemos apjungimui su kita sistema. Gaminio savikainos apskaičiavimo IS susiejamumo su kitomis sistemomis nepalaiko.

6.3 Kokybės faktoriai, susiję su tolesniu sistemos vystymusi

Kokybės faktoriai, susiję su tolesniu vystymusi, yra šie: palaikomumas, lankstumas.

Palaikomumas nusako pastangas reikalingas nustatyti ir ištaisyti sistemos klaidas. Ši kokybės faktorių dar galima susieti su lankstumu. Gaminio savikainos apskaičiavimo IS, realizuota MS Visual Basic aplinkoje, užtikrina greitą klaidų aptikimą. Programos kodas yra saugomas dalimis ir susietas su konkrečia forma. Įvykus klaidai, programos vykdymas sustabdomas, ir žymeklis pastatomas į tą kodo eilutę, kurioje įvyko klaida.

Lankstumas nusako pastangas, reikalingas programų sistemos modifikavimui. Sistemos modifikavimas bei jos išplečiamumas galimas, nes nėra labai sudėtingas. Šiam tikslui reikalingos MS Access duomenų bazės ir Visual Basic for Applications priemonių išmanymas. Egzistuojanti sistema lengvai gali būti modifikuojama.

7. IŠVADOS

- Išanalizuoti dauguma programinės įrangos paketų, šiuo metu plačiai naudojamų Lietuvos įmonėse. Dauguma jų gana universalūs ir tinka įvairiems įmonės valdymo uždaviniams spręsti. Šios sistemos pasižymi dideliu sudėtingumu bei aukšta kaina. Įmonės darbuotojų apmokymui dirbti šiomis sistemomis tenka skirti daug lėšų ir laiko.
- Būtent dėl šios priežasties šio darbo tikslas buvo sudaryti nedidelį ir nesudėtingą gaminio savikainos apskaičiavimo informacinės sistemos programinės įrangos projektą, kuris apimtų pagrindinius didžiųjų sistemų privalumus bei vengtų jų trūkumų.
- Ši sistema projektuojama CASE sistemos **ProVision Workbench** (Proforma Corporation) paketu, kuris apima visą informacinės sistemos gyvavimo ciklą (IS GC). **ProVision Workbench** paketo privalumas yra tas, kad IS kūrimo gyvavimo ciklas sudarytas pagal strategiją “nuo sąvokos iki programos kodo”.
- **ProVision Workbench** paketas leidžia naudoti sukurtų objektų duomenis reliacinių duomenų bazių valdymo sistemose (DBVS), tokiose kaip Oracle, Visual Basic, Access ir kt., kuriose gali būti atliekamos informacinės sistemos funkcijos. DDL (*data definition language*) interpretuoja SQL CREATE sakinius į lenteles, kurios gali būti perskaitomos kokios nors vartotojo reliacinės duomenų bazės.
- Pagrindinis šaltinis generuojamiems vartotojo sąsajos langų prototipams - biznio objektų modelyje apibrėžiami objektai, jų metodai, atributai bei ryšiai tarp objektų.
- Tyrimai, atlikti programinės įrangos kūrimo metu, parodo, kad CASE sistemos sąsaja su duomenų bazių valdymo sistemomis (DBVS), mūsų atveju MS Access ir objektinė programavimo kalba Visual Basic 6.0, pagreitina ir atpigina nedidelės ir nesudėtingos programinės įrangos kūrimo procesą.

8. LITERATŪRA

1. Stekas [interaktyvus]. [žiūrėta 2003-10-27]. Prieiga per internetą: <http://www.stekas.lt/>
2. Konto. Apie. [interaktyvus]. [žiūrėta 2003-10-27]. Prieiga per internetą: <http://www.konto.lt/>
3. Buhalterinės apskaitos paketas [interaktyvus]. [žiūrėta 2003-10-27]. Prieiga per internetą: <http://www.rl.lt/soft/programa.php3?prgid=24>
4. Resvera [interaktyvus]. [žiūrėta 2003-10-27]. Prieiga per internetą: <http://www.resvera.lt/>
5. EMAN [interaktyvus]. [žiūrėta 2003-10-27]. Prieiga per internetą: http://www.equinox.lt/prod_eman.php
6. VISION [interaktyvus]. [žiūrėta 2003-10-27]. Prieiga per internetą: http://www.equinox.lt/prod_vision.php
7. FINVALDA [interaktyvus]. [žiūrėta 2003-10-27]. Prieiga per internetą: <http://www.fvs.lt/finvalda.htm>
8. Rivilė [interaktyvus]. [žiūrėta 2003-10-27]. Prieiga per internetą: http://www.rivile.lt/rivile_lt/default.asp
9. Centas [interaktyvus]. [žiūrėta 2003-11-11]. Prieiga per internetą: <http://www.edlonta.lt/centas.htm>, <http://www.edlonta.lt/Vadovas.pdf>
10. Pragma [interaktyvus]. [žiūrėta 2003-11-11]. Prieiga per internetą: <http://www.proringas.lt/puslapis2.htm>,
http://www.avilura.lt/index.php?USER_ID=pragma
11. S.Lauesen Software Requirements: styles and techniques. – Samfudslitteratur, Frederiksberg, 1999. – 191p. ISBN 87-593-0794-3
12. S.Gudas Veiklos analizė ir informacinių poreikių specifikavimas, Kaunas, Naujasis lankas, 2002. 96 p.
13. James & Suzanne Robertson Volere Requirements Specification Template, Edition 8, Principals of the Atlantic Systems Guild, London, 2001, [interaktyvus]. [žiūrėta 2002-10-20]. Prieiga per internetą: www.systemsguild.com
14. Reikalavimų specifikavimo dokumento forma. [Interaktyvus]. [žiūrėta 2002-10-20]. Prieiga per internetą: <http://www.2ics.hawaii.edu/~johnson/413/lectures/5.2.html>
15. P.P.Chen The Entity-Relationship Model: Towards a Unified view of Data, ACM Transactions on Database Systems, Vol.1. No.1, March 1976, pp.9 –36
16. V.Sekliuckis, G.Garšva, S.Gudas. Duomenų bazės, Kaunas: Naujasis lankas, 2000. 94 p.

17. R. Butkienė. Informacijos sistemų projektavimas Oracle Designer/2000 priemonėmis, Kaunas, Technologija, 1998, 165 p.
18. S. Gudas. The Information Architecture Framework for Enterprise Integration. – Databases & Information Systems. Proceedings of the 4th IEEE International Baltic Workshop. Vol. 12. Vilnius, Technika, 2000, p. 168-175.
19. S. Gudas. Architektūra grindžiamas IS projektavimas. [interaktyvus]. [žiūrėta 2002-10-20]. Prieiga per internetą: <ftp://isd.ktu.lt/ISD/Gudas/DBIS/>

9. SUMMARY

This system is designed by package of CASE system entitled Pro Vision Workbench, which absolutely includes all Life Cycle of Information System. The creation life cycle of Information System is composed according to strategy from concept to program code is the necessity of Pro Vision Workbench package.

The created objects of data in relation Data Base Management System (DBMS), for instance, Oracle, Visual Basic, Access and etc., in which the functions of IS can be done, are allowed to use by Pro Vision Workbench package. The sentences of SQL CREATE are interpreted to tables by DDL (data definition language), and any relational Data Base of user can read these tables.

The basic source of generating windows prototypes of user link is the definable objects, its methods, attributes and relationship between objects in business objects model.

Research, which was carried out of software, show the link of CASE system between Data Base Management System (DBMS), in this case, MS Access and the objective programming language of Visual Basic 6.0 accelerate and reduced the price of created process small and simple software.

10. TERMINŲ IR SANTRUMPŲ ŽODYNAS

- CASE – Computer aided software engineering
- DFD – duomenų srautų diagramos (angl. Data Flow Diagrams)
- ERD – esybių - ryšių diagrama (angl. Entity Relationship Diagrams)
- IS – informacinė sistema
- DB – duomenų bazė.
- IT – informacinės technologijos
- PĮ – Programinė įranga
- OMT – Objektinė modeliavimo technologija
- DBVS – duomenų bazių valdymo sistema
- IS GC – informacinės sistemos kūrimo gyvavimo ciklas
- DDL – Data definition language
- OO A&D – Objektiškai orientuota analizė ir modeliavimas (angl. Object oriented analysis and design)
- UML – Unified modeling language
- OLE – Object Linking and Embedding

11. PRIEDAI

```
CREATE TABLE GAMYBASUSIDEDABADARBININKAI
( GAMYBASUSIDEDABADARBININKAI_ID INTEGER NOT NULL
, FK_ASMENS_KODAS INTEGER
, FK_SPESIALYBE INTEGER
, PRIMARY KEY (GAMYBASUSIDEDABADARBININKAI_ID)
, FOREIGN KEY FK_GAMYBA (FK_ASMENS_KODAS)
  REFERENCES GAMYBA
  ON DELETE CASCADE
, FOREIGN KEY FK_DARBININKAI (FK_SPESIALYBE)
  REFERENCES DARBININKAI
  ON DELETE CASCADE
);
```

```
CREATE TABLE GAMYBA
( ASMENS_KODAS INTEGER NOT NULL
, GIMIMO_DATA TIME
, OPERACIJA VARCHAR(100)
, PAVARDE VARCHAR(100)
, PRIMARY KEY (ASMENS_KODAS)
);
```

```
CREATE TABLE DARBININKAI
( SPESIALYBE INTEGER NOT NULL
, KATEGORIJA VARCHAR(100)
, KIEKIS INTEGER
, PREMIJAA VARCHAR(15)
, PREMIJAB VARCHAR(100)
, VALANDINIS_ATLYGIS VARCHAR(10)
, PRIMARY KEY (SPESIALYBE)
);
```

```
CREATE TABLE PAGRINDINIS_DUOMENYS
( APSILDYMO_TRUKME INTEGER
, APSVIETIMO_TRUKLME INTEGER
, EFEKTYVUS_LAIKAS INTEGER
, ELEKTROS_KAINA VARCHAR(10)
, FAKTINIS_LAIKAS INTEGER
, GAMINIO_SVORIS VARCHAR(15)
, GAMYBOS_APIMTIS VARCHAR(4)
, PAMAINU_SKAICIUS INTEGER
, SILUMOS_KAINA VARCHAR(10)
, GAMINIO_PAVADINIMAS INTEGER NOT NULL
, PRIMARY KEY (GAMINIO_PAVADINIMAS)
);
```

```
CREATE TABLE VADOVAISPECIALINIOSAVIKAINA
( VADOVAISPECIALINIOSAVIKAINA_ID INTEGER NOT NULL
, FK_SPECIALIBE INTEGER
```

```

, FK_SAVIKAINOS_KODAS INTEGER
, PRIMARY KEY (VADOVAISPECIALINIOSAVIKAINA_ID)
, FOREIGN KEY FK_VADOVAI_SPECIALISTAI (FK_SPECIALIBE)
  REFERENCES VADOVAI_SPECIALISTAI
  ON DELETE CASCADE
, FOREIGN KEY FK_GAMINIO_SAVIKAINA (FK_SAVIKAINOS_KODAS)
  REFERENCES GAMINIO_SAVIKAINA
  ON DELETE CASCADE
);

```

```

CREATE TABLE VADOVAI_SPECIALISTAI
( SPECIALIBE INTEGER NOT NULL
, KIEKIAI INTEGER
, MENESINIS_UZMOKESTIS VARCHAR(10)
, PREMIJA VARCHAR(15)
, PRIMARY KEY (SPECIALIBE)
);

```

```

CREATE TABLE GAMINIO_SAVIKAINA
( SAVIKAINOS_KODAS INTEGER NOT NULL
, PAVADINIMAS VARCHAR(100)
, PRIMARY KEY (SAVIKAINOS_KODAS)
);

```

```

CREATE TABLE PAGR_MEDZIAGOS
( ATLIEKU_KAINA VARCHAR(10)
, ATLIEKU_NORMA VARCHAR(15)
, KAINA VARCHAR(10)
, SVORIS VARCHAR(15)
, MEDZIAGU_PAVADINIMAS INTEGER NOT NULL
, PRIMARY KEY (MEDZIAGU_PAVADINIMAS)
);

```

```

CREATE TABLE GAMYBASUSIDEDAISPECIALISTAI
( GAMYBASUSIDEDAISPECIALISTAI_ID INTEGER NOT NULL
, FK_ASMENS_KODAS INTEGER
, FK_SPECIALIBE INTEGER
, PRIMARY KEY (GAMYBASUSIDEDAISPECIALISTAI_ID)
, FOREIGN KEY FK_GAMYBA (FK_ASMENS_KODAS)
  REFERENCES GAMYBA
  ON DELETE CASCADE
, FOREIGN KEY FK_VADOVAI_SPECIALISTAI (FK_SPECIALIBE)
  REFERENCES VADOVAI_SPECIALISTAI
  ON DELETE CASCADE
);

```

```

CREATE TABLE ENERGIJA
( APIBUDINIMAS VARCHAR(100)
, ENERGIJOS_PAVADINIMAS VARCHAR(100)
, TIEKEJAI VARCHAR(100)
, ENERGIJA_ID INTEGER NOT NULL
, PRIMARY KEY (ENERGIJA_ID)
);

```

```

);

CREATE TABLE GAMINIOSAVIKAINADARBININKAI
( GAMINIOSAVIKAINADARBININKAI_ID INTEGER NOT NULL
, FK_SAVIKAINOS_KODAS INTEGER
, FK_SPESIALYBE INTEGER
, PRIMARY KEY (GAMINIOSAVIKAINADARBININKAI_ID)
, FOREIGN KEY FK_GAMINIO_SAVIKAINA (FK_SAVIKAINOS_KODAS)
  REFERENCES GAMINIO_SAVIKAINA
  ON DELETE CASCADE
, FOREIGN KEY FK_DARBININKAI (FK_SPESIALYBE)
  REFERENCES DARBININKAI
  ON DELETE CASCADE
);

CREATE TABLE GAMINIOSAVIKAIEINAIPASTATAI
( GAMINIOSAVIKAIEINAIPASTATAI_ID INTEGER NOT NULL
, FK_SAVIKAINOS_KODAS INTEGER
, FK_PASTATU_PAVADINIMAS INTEGER
, PRIMARY KEY (GAMINIOSAVIKAIEINAIPASTATAI_ID)
, FOREIGN KEY FK_GAMINIO_SAVIKAINA (FK_SAVIKAINOS_KODAS)
  REFERENCES GAMINIO_SAVIKAINA
  ON DELETE CASCADE
, FOREIGN KEY FK_PASTATAI (FK_PASTATU_PAVADINIMAS)
  REFERENCES PASTATAI
  ON DELETE CASCADE
);

CREATE TABLE PASTATAI
( PASTATU_PAVADINIMAS INTEGER NOT NULL
, AMORTIZACIJOS_NORMA VARCHAR(15)
, APSILDYMO_NORMA VARCHAR(15)
, APSILDYMO_NORMATIVAI INTEGER
, APSVIETIMO_KOEFICIENTAS VARCHAR(15)
, APSVIETIMO_NORMA INTEGER
, AUKSTIS VARCHAR(15)
, GAMINIMO_METAI INTEGER
, KAINA_1M VARCHAR(10)
, KITA_NORMA VARCHAR(15)
, MATERIALINE_NORMA VARCHAR(15)
, PLOTAS VARCHAR(15)
, PRIMARY KEY (PASTATU_PAVADINIMAS)
);

CREATE TABLE PASTATU_APTARNAVIMAS
( PASTABOS VARCHAR(100)
, RUSIS VARCHAR(100)
, TRUKME VARCHAR(100)
, APTARNAVIMO_KODAS INTEGER NOT NULL
, PRIMARY KEY (APTARNAVIMO_KODAS)
);

```

```

CREATE TABLE PAGRINDINISDUOUOJAISIRANGOS
  ( PAGRINDINISDUOUOJAISIRANGOS_ID INTEGER NOT NULL
  , FK_GAMINIO_PAVADINIMAS INTEGER
  , FK_IRANGOS_KODAS INTEGER
  , PRIMARY KEY (PAGRINDINISDUOUOJAISIRANGOS_ID)
  , FOREIGN KEY FK_PAGRINDINIS_DUOMENYS
(FK_GAMINIO_PAVADINIMAS)
  REFERENCES PAGRINDINIS_DUOMENYS
  ON DELETE CASCADE
  , FOREIGN KEY FK_IRANGOS (FK_IRANGOS_KODAS)
  REFERENCES IRANGOS
  ON DELETE CASCADE
);

```

```

CREATE TABLE IRANGOS
  ( IRANGOS_KODAS INTEGER NOT NULL
  , AMORTIZACIJOS_NORMA VARCHAR(10)
  , APKROVIMO_KOEFICIENTAS VARCHAR(10)
  , GAMINIMO_METAI INTEGER
  , IRANGOS_GALINGUMAS VARCHAR(10)
  , IRANGOS_PAVADINIMAS VARCHAR(100)
  , KAINA VARCHAR(10)
  , KIEKIS INTEGER
  , KITA_NORMA VARCHAR(10)
  , MATERIALINE_NORMA VARCHAR(10)
  , PAKLAUSOS_KOEFICIENTAS VARCHAR(15)
  , PRIMARY KEY (IRANGOS_KODAS)
);

```

```

CREATE TABLE GAMINIOSAVIKAISIEINAIRANGOS
  ( GAMINIOSAVIKAISIEINAIRANGOS_ID INTEGER NOT NULL
  , FK_SAVIKAINOS_KODAS INTEGER
  , FK_IRANGOS_KODAS INTEGER
  , PRIMARY KEY (GAMINIOSAVIKAISIEINAIRANGOS_ID)
  , FOREIGN KEY FK_GAMINIO_SAVIKAINA (FK_SAVIKAINOS_KODAS)
  REFERENCES GAMINIO_SAVIKAINA
  ON DELETE CASCADE
  , FOREIGN KEY FK_IRANGOS (FK_IRANGOS_KODAS)
  REFERENCES IRANGOS
  ON DELETE CASCADE
);

```

```

CREATE TABLE PAGRINDINISDUOISPECIALISTAI
  ( PAGRINDINISDUOISPECIALISTAI_ID INTEGER NOT NULL
  , FK_GAMINIO_PAVADINIMAS INTEGER
  , FK_SPECIALIBE INTEGER
  , PRIMARY KEY (PAGRINDINISDUOISPECIALISTAI_ID)
  , FOREIGN KEY FK_PAGRINDINIS_DUOMENYS
(FK_GAMINIO_PAVADINIMAS)
  REFERENCES PAGRINDINIS_DUOMENYS
  ON DELETE CASCADE
  , FOREIGN KEY FK_VADOVAI_SPECIALISTAI (FK_SPECIALIBE)

```



```

REFERENCES VADOVAI_SPECIALISTAI
ON DELETE CASCADE
);

CREATE TABLE TIEKIMAS
( PASTABOS VARCHAR(100)
, SANDELIO_PAVADINIMAS VARCHAR(100)
, VADOVAS VARCHAR(100)
, REGNR INTEGER NOT NULL
, PRIMARY KEY (REGNR)
);

CREATE TABLE IRANGOSAPTARNAALINGAIRANGOS
( IRANGOSAPTARNAALINGAIRANGOS_ID INTEGER NOT NULL
, FK_APTARNAVIMO_KODAS INTEGER
, FK_IRANGOS_KODAS INTEGER
, PRIMARY KEY (IRANGOSAPTARNAALINGAIRANGOS_ID)
, FOREIGN KEY FK_IRANGOS_APTARNAVIMAS
(FK_APTARNAVIMO_KODAS)
REFERENCES IRANGOS_APTARNAVIMAS
ON DELETE CASCADE
, FOREIGN KEY FK_IRANGOS (FK_IRANGOS_KODAS)
REFERENCES IRANGOS
ON DELETE CASCADE
);

CREATE TABLE IRANGOS_APTARNAVIMAS
( APTARNAVIMO_KODAS INTEGER NOT NULL
, APTARNAVIMO_RUSIS VARCHAR(100)
, PASTABOS VARCHAR(100)
, TRUKME VARCHAR(100)
, PRIMARY KEY (APTARNAVIMO_KODAS)
);

CREATE TABLE ENERGIJAREIKALOJASIPASTATAI
( ENERGIJAREIKALOJASIPASTATAI_ID INTEGER NOT NULL
, FK_ENERGIJA_ID INTEGER
, FK_PASTATU_PAVADINIMAS INTEGER
, PRIMARY KEY (ENERGIJAREIKALOJASIPASTATAI_ID)
, FOREIGN KEY FK_ENERGIJA (FK_ENERGIJA_ID)
REFERENCES ENERGIJA
ON DELETE CASCADE
, FOREIGN KEY FK_PASTATAI (FK_PASTATU_PAVADINIMAS)
REFERENCES PASTATAI
ON DELETE CASCADE
);

CREATE TABLE TIEKIMASTEIKIAPAGRMEDZIAGOS
( TIEKIMASTEIKIAPAGRMEDZIAGOS_ID INTEGER NOT NULL
, FK_REGNR INTEGER
, FK_MEDZIAGU_PAVADINIMAS INTEGER
, PRIMARY KEY (TIEKIMASTEIKIAPAGRMEDZIAGOS_ID)

```

```

    , FOREIGN KEY FK_TIEKIMAS (FK_REGNR)
      REFERENCES TIEKIMAS
      ON DELETE CASCADE
    , FOREIGN KEY FK_PAGR_MEDZIAGOS (FK_MEDZIAGU_PAVADINIMAS)
      REFERENCES PAGR_MEDZIAGOS
      ON DELETE CASCADE
  );

CREATE TABLE ENERGIJAREIKALDOJASIIRANGOS
  ( ENERGIJAREIKALDOJASIIRANGOS_ID INTEGER NOT NULL
  , FK_ENERGIJA_ID INTEGER
  , FK_IRANGOS_KODAS INTEGER
  , PRIMARY KEY (ENERGIJAREIKALDOJASIIRANGOS_ID)
  , FOREIGN KEY FK_ENERGIJA (FK_ENERGIJA_ID)
      REFERENCES ENERGIJA
      ON DELETE CASCADE
  , FOREIGN KEY FK_IRANGOS (FK_IRANGOS_KODAS)
      REFERENCES IRANGOS
      ON DELETE CASCADE
  );

CREATE TABLE PAGRINDINISDUOOJAISPASTATAI
  ( PAGRINDINISDUOOJAISPASTATAI_ID INTEGER NOT NULL
  , FK_GAMINIO_PAVADINIMAS INTEGER
  , FK_PASTATU_PAVADINIMAS INTEGER
  , PRIMARY KEY (PAGRINDINISDUOOJAISPASTATAI_ID)
  , FOREIGN KEY FK_PAGRINDINIS_DUOMENYS
(FK_GAMINIO_PAVADINIMAS)
      REFERENCES PAGRINDINIS_DUOMENYS
      ON DELETE CASCADE
  , FOREIGN KEY FK_PASTATAI (FK_PASTATU_PAVADINIMAS)
      REFERENCES PASTATAI
      ON DELETE CASCADE
  );

CREATE TABLE TIEKIMASTEIKIAOJAMIGAMINIAI
  ( TIEKIMASTEIKIAOJAMIGAMINIAI_ID INTEGER NOT NULL
  , FK_REGNR INTEGER
  , FK_DETALES_PAVADINIMAS INTEGER
  , PRIMARY KEY (TIEKIMASTEIKIAOJAMIGAMINIAI_ID)
  , FOREIGN KEY FK_TIEKIMAS (FK_REGNR)
      REFERENCES TIEKIMAS
      ON DELETE CASCADE
  , FOREIGN KEY FK_KOMPLEKTUOJAMI_GAMINIAI
(FK_DETALES_PAVADINIMAS)
      REFERENCES KOMPLEKTUOJAMI_GAMINIAI
      ON DELETE CASCADE
  );

CREATE TABLE KOMPLEKTUOJAMI_GAMINIAI
  ( DETALES_PAVADINIMAS INTEGER NOT NULL
  , KAINA VARCHAR(10)

```

```

, KIEKIS INTEGER
, PRIMARY KEY (DETALES_PAVADINIMAS)
);

CREATE TABLE GAMINIOSAVIKAIJAMIGAMINIAI
( GAMINIOSAVIKAIJAMIGAMINIAI_ID INTEGER NOT NULL
, FK_SAVIKAINOS_KODAS INTEGER
, FK_DETALES_PAVADINIMAS INTEGER
, PRIMARY KEY (GAMINIOSAVIKAIJAMIGAMINIAI_ID)
, FOREIGN KEY FK_GAMINIO_SAVIKAINA (FK_SAVIKAINOS_KODAS)
REFERENCES GAMINIO_SAVIKAINA
ON DELETE CASCADE
, FOREIGN KEY FK_KOMPLEKTUOJAMI_GAMINIAI
(FK_DETALES_PAVADINIMAS)
REFERENCES KOMPLEKTUOJAMI_GAMINIAI
ON DELETE CASCADE
);

CREATE TABLE PASTATUAPTARNALINGAPASTATAI
( PASTATUAPTARNALINGAPASTATAI_ID INTEGER NOT NULL
, FK_APTARNAVIMO_KODAS INTEGER
, FK_PASTATU_PAVADINIMAS INTEGER
, PRIMARY KEY (PASTATUAPTARNALINGAPASTATAI_ID)
, FOREIGN KEY FK_PASTATU_APTARNAVIMAS
(FK_APTARNAVIMO_KODAS)
REFERENCES PASTATU_APTARNAVIMAS
ON DELETE CASCADE
, FOREIGN KEY FK_PASTATAI (FK_PASTATU_PAVADINIMAS)
REFERENCES PASTATAI
ON DELETE CASCADE
);

CREATE TABLE PAGRINDINISDUOISDARBININKAI
( PAGRINDINISDUOISDARBININKAI_ID INTEGER NOT NULL
, FK_GAMINIO_PAVADINIMAS INTEGER
, FK_SPESIALYBE INTEGER
, PRIMARY KEY (PAGRINDINISDUOISDARBININKAI_ID)
, FOREIGN KEY FK_PAGRINDINIS_DUOMENYS
(FK_GAMINIO_PAVADINIMAS)
REFERENCES PAGRINDINIS_DUOMENYS
ON DELETE CASCADE
, FOREIGN KEY FK_DARBININKAI (FK_SPESIALYBE)
REFERENCES DARBININKAI
ON DELETE CASCADE
);

CREATE TABLE GAMINIOSAVIKAIPAGRMEDZIAGOS
( GAMINIOSAVIKAIPAGRMEDZIAGOS_ID INTEGER NOT NULL
, FK_SAVIKAINOS_KODAS INTEGER
, FK_MEDZIAGU_PAVADINIMAS INTEGER
, PRIMARY KEY (GAMINIOSAVIKAIPAGRMEDZIAGOS_ID)
, FOREIGN KEY FK_GAMINIO_SAVIKAINA (FK_SAVIKAINOS_KODAS)

```

```
REFERENCES GAMINIO_SAVIKAINA
ON DELETE CASCADE
, FOREIGN KEY FK_PAGR_MEDZIAGOS (FK_MEDZIAGU_PAVADINIMAS)
REFERENCES PAGR_MEDZIAGOS
ON DELETE CASCADE
);
```