

**VILNIAUS UNIVERSITETAS  
KAUNO HUMANITARINIS FAKULTETAS**

**INFORMATIKOS KATEDRA**

Verslo informatikos studijų programa

Kodas 62109P101

**ROBERTAS BUTKEVIČIUS**

**MAGISTRO BAIGIAMASIS DARBAS**

**INFORMACIJOS VALDYMO KONTAKTŲ CENTRUOSE ANALIZĖ**

Kaunas 2007

**VILNIAUS UNIVERSITETAS  
KAUNO HUMANITARINIS FAKULTETAS**

**INFORMATIKOS KATEDRA**

**ROBERTAS BUTKEVIČIUS**

**MAGISTRO BAIGIAMASIS DARBAS**

**INFORMACIJOS VALDYMO KONTAKTŲ CENTRUOSE ANALIZĖ**

Leidžiama  
ginti \_\_\_\_\_

Magistrantas \_\_\_\_\_

(parašas)

Darbo vadovas \_\_\_\_\_

(parašas)

doc. Vytautas Rudžionis

(darbo vadovo mokslo laipsnis, mokslo pedagoginis vardas,  
vardas ir pavardė)

Darbo įteikimo data \_\_\_\_\_

Registracijos Nr. \_\_\_\_\_

Kaunas 2007

## TURINYS

<b>SANTRUMPŲ SĄRAŠAS</b> .....	<b>4</b>
<b>PAVEIKSLŲ SĄRAŠAS</b> .....	<b>4</b>
<b>LENTELIŲ SĄRAŠAS</b> .....	<b>5</b>
<b>SANTRAUKA</b> .....	<b>6</b>
<b>ĮVADAS</b> .....	<b>7</b>
<b>1. KONTAKTŲ CENTRŲ ORGANIZACINĖS IR TECHNOLOGINĖS STRUKTŪROS ANALIZĖ</b> .....	<b>10</b>
1.1 KONTAKTŲ CENTRŲ ORGANIZACINĖ STRUKTŪRA .....	10
1.2 KONTAKTŲ CENTRŲ TECHNOLOGINĖ STRUKTŪRA .....	15
1.2.1 AUTOMATINIŲ KONTAKTŲ CENTRŲ TECHNOLOGIJOS .....	19
1.2.2 Kalbos Atpažinimas.....	19
1.2.3 Kalbos atpažinimas hierarchinėse struktūrose.....	19
1.2.4 Kalbos atpažinimo sistemų charakteristika .....	20
1.2.5 Klaidų skaičius kalbos atpažinime .....	20
1.3 KALBOS ATPAŽINIMO MODELIAI.....	22
1.3.1 HMM modelis kalbos atpažinime.....	23
1.3.2 HMM modelio architektūra.....	24
1.3.3 Neuroniniu tinklu paremtos kalbos atpažinimas .....	26
1.3.4 Dinaminiu laiko iškraipymu (DLI) paremtas kalbos atpažinimas.....	26
1.4 KALBOS ATPAŽINIMO PROGRAMINĖ ĮRANGA.....	27
1.4.1 HTK programinė įranga.....	27
1.4.2 Microsoft Speech Server programinė įranga .....	28
1.4.3 IBM WebSphere programinė įranga.....	30
1.5 ESAMŲ KONTAKTŲ CENTRŲ TECHNOLOGINIŲ SPRENDIMŲ TRŪKUMAI....	31
<b>2. AUTOMATINIO KONTAKTŲ CENTRO TECHNOLOGINIO SPRENDIMO VARIANTAS</b> .....	<b>32</b>
2.1 PROJEKTO STRUKTŪRA.....	33
2.2 INFORMACIJOS SISTEMAI KELIAMI REIKALAVIMAI.....	36
<b>3. PROJEKTO REALIZACIJAI REIKALINGŲ BALSO KOMANDŲ ATPAŽINIMO TIKSLUMO TYRIMAS</b> .....	<b>37</b>
3.1 BANDYMAS SU HTK PRODUKTU.....	37
3.2 BANDYMAS SU MS SPEECH SERVER PRODUKTU.....	39
3.3 TYRIMO IŠVADOS.....	42
<b>4. AUTOMATINIO KONTAKTŲ CENTRO DUOMENŲ BAZĖS SĄSAJOS SPRENDIMAS</b> .....	<b>43</b>
4.1 TECHNINĖ UŽDUOTIS.....	43
4.2 KOMPIUTERIZUOJAMŲ FUNKCIJŲ HIERARCHIJA .....	44
4.3 KONCEPCINIS OBJEKTO MODELIS (ER-DIAGRAMA).....	45
4.4 PROJEKTUOJAMOS SISTEMOS DUOMENŲ SRAUTŲ DIAGRAMA.....	47
4.5 SISTEMOS DINAMIKOS APRAŠYMAS (DIALOGO SCENARIJUS) .....	48
4.6 KLASIFIKAVIMO IR KODAVIMO SISTEMOS APRAŠYMAS.....	49
4.7 DUOMENŲ BAZĖS PROJEKTAS .....	50
4.8 INFORMACIJOS APDOROJIMO, PAIEŠKOS IR IŠRINKIMO TECHNOLOGIJOS APRAŠYMAS .....	52
4.9 SISTEMOS ARCHITEKTŪROS APRAŠYMAS (STRUKTŪRINĖ SCHEMA) .....	56
4.10TESTAVIMO DUOMENŲ APRAŠYMAS.....	57
4.11VARTOTOJO VADOVAS.....	61
4.12PROGRAMUOTOJO VADOVAS.....	68
<b>IŠVADOS IR PASIŪLYMAI</b> .....	<b>69</b>
<b>LITERATŪROS SĄRAŠAS</b> .....	<b>72</b>
<b>PRIEDAI</b> .....	<b>74</b>

## SANTRUMPŲ SĄRAŠAS

DB – duomenų bazė  
IS – informacijos sistema  
KAS – kalbos atpažinimo sistemos  
MS – Microsoft  
HMM – paslėptas Markovo modelis  
ASR – Automatinė kalbos atpažinimo sistema

## PAVEIKSLŲ SĄRAŠAS

1 pav. UAB „Lintel“ struktūra .....	13
2 pav. „Lietuvos Telekomo“ TSQL sistemos programų ir modulių tarpusavio sąsaja .....	17
3 pav. AB „Lietuvos Telekomo“ TSQL sistemos sąsaja su kitų sistemų programomis .....	18
4 pav. Būklės keitimasis Paslėptajame Markovo modelyje (pavyzdys). .....	24
5 pav. Bendroji HMM modelio sudėtis. ....	25
6 pav. Microsoft balso serveris MSS 2004 R2 2005. ....	28
7 pav. MS Speech Server sąveika su duomenų bazėmis. ....	29
8 pav. IBM WebSphere balso serveris .....	30
9 pav. Projekto CallMagic 2007 struktūra. ....	33
10 pav. HTT bandymų metodika su moterimis. ....	38
11 pav. HTT bandymų metodika su vyrais. ....	38
12 Pav. MS Speech Server bandymų metodika dvigarsiais. ....	39
13 pav. MS Speech Server bandymų metodika komandomis. ....	41
14 pav. Kompiuterizuojamų funkcijų hierarchija .....	44
15 pav. ER modelis .....	46
16 pav. Duomenų srautų diagrama. ....	47
17 pav. Dialogo scenarijus .....	48
18 pav. Duomenų bazės loginė schema MS Visio aplinkoje .....	50
19 pav. DB loginė schema. ....	51
20 pav. Sistemos struktūros diagrama. ....	56
21 pav. Pagrindinis sistemos langas .....	61
22 pav. Vartotojų administravimas .....	62
23 pav. Įmonių administravimas .....	63
24 pav. Gatvių, miestų ir veiklos rūšių administravimas .....	64
25 pav. Įmonių direktorių administravimas .....	64
26 pav. Gyventojų administravimas .....	65
27 pav. Paieškos konsolė. ....	66
28 pav. Vartotojų istorija. ....	67

## LENTELIŲ SĄRAŠAS

1 lentelė. Didžiausių Amerikos kontaktų centrų sąrašas.....	10
2 lentelė. Apytikris klaidingų žodžių skaičius. ....	21
3 lentelė. HTT Testavimo rezultatai.....	39
4 lentelė. MS Speech Server testavimo rezultatai su dvigarsiais. ....	40
5 lentelė. MS Speech Server testavimo rezultatai, bei transkripcijos. ....	41
6 lentelė. Informacijos išrinkimo technologijos aprašymas .....	52
7 lentelė. Testavimo duomenų pavyzdžiai .....	57

## **SANTRAUKA**

BUTKEVIČIUS, Robertas. (2007) *The informational management analysis in call centers*. Graduation paper. Kaunas: Vilnius University, Kaunas Faculty of Humanities, Department of Informatics. 104 p.

### **SUMMARY**

The current paper is concentrated on the organizational and technological structures of contact centers. Here we familiarize with speech recognition systems and models used in them. The accomplished experiment ascertained accuracy of speech recognition for the chosen systems. A simplified prototype of contact center was projected and its tests performed.

The work object of MA paper is organizational and technological structure of contact centers.

The aim of this paper – to inspect the organizational and technological structure of contact centers in Lithuania and abroad, compare them by identifying the major formation problems for organizational and technological structure of contact centers in Lithuania.

Research methods used in the paper. The research examines and reviews methods of technologies used for the contact center's information systems in Lithuania and abroad, together generalizing them. Research material is evaluated by the methods of deduction, synthesis and induction. Research method is applied in research for accuracy in speech recognition systems. Analysis of scientific articles, system evaluation methodology is performed together with systemization, comparative analysis of systems. System functional components, usable methods, software, possibilities of development are compared. A creation of a simplified and generalized system model is performed, where principal guide lines, major system elements and performed processes are presented.

Significance of the received results. The paper provides generalized models, useful for projection of a new system, and those of information system architecture and course of activity at contact centers. The proposed architectural model provides principal guide lines and elements of the system. The offered simplified model of contact centers CallMagic 2007, is realized in SQL data base, a representative client part is created with the help of Microsoft Visual FoxPro.

## IVADAS

Šiuolaikiniame versle labai svarbu greitai gauti reikiamą informaciją reikiamu laiku reikiamoje vietoje, greitai atlikti įvairius rinkos tyrimus siekiant išsiaiškinti kintančius rinkos poreikius. Telefoninės paslaugos yra viena iš paslaugų rūšių, padedančių užtikrinti spartesnę informacijos cirkuliaciją. Siekiant apdoroti didelį kiekį skambučių ir patenkinti kuo didesnio kiekio klientų, ieškančių informacijos, poreikius, yra steigiami Kontaktų centrai – „fizinė vieta, kurioje priimami „ateinantys“ arba „išeinantys“ (skambinami) didžiuliai skambučių srautai“. Kontaktų centruose naudojamos technologijos suteikia galimybę valdyti skambučius nepastebimu greičiu ir nepriklausomai nuo skambučių geografinės vietos. Vieni pagrindinių faktorių, lemiančių kontaktų centrų veiklos efektyvumą, yra jų valdymo struktūra ir naudojamos technologijos. Kaip ir kiekviename versle, taip ir Kontaktų centrų veikloje, svarbu yra parinkti optimalią organizacinę struktūrą, kuri būtų priderinta prie tiek prie įmonės parametrų, tiek prie rinkos situacijos.

*Baigiamojo darbo objektas* – kontaktų centrų informacijos valdymas.

Baigiamojo darbo *tikslas* – išnagrinėti, informacijos valdymo kontaktų centruose problemas ir sudaryti šių ekonomiškai pagrįstą problemų sprendimo modelį.

*Darbo uždaviniai:*

1. Išanalizuoti kontaktų centrų organizacinės ir technologinės struktūros formavimo patirtį.
2. Išanalizuoti organizacinės ir technologinės struktūros ypatumus Lietuvoje.
3. Aptarti TSQL sandarą ir vystymo kryptis.
4. Palyginti kontaktų centrų turimą patirtį Lietuvoje ir užsienyje.
5. Identifikuoti Lietuvoje steigiamų kontaktų centrų pagrindines technologines bei organizacines problemas.
6. Remiantis identifikuotomis problemomis nustatyti sąsajos informaciniams srautams reikalavimus.
7. Remiantis nustatytais sąsajos informaciniams srautams reikalavimais, suprojektuoti kontaktų centro duomenų srautų informacinę sistemą.
8. Atlikti tyrimą, nustatant balso atpažinimų sistemų tikslumą su numatytais informacinės sistemos tiksliniais žodžiais - komandomis.
9. Realizuoti kontaktų centro prototipinę informacinę sistemą.

*Darbo struktūra.* Darbą sudaro trys pagrindinės dalys. Pirmoje dalyje yra išanalizuojama esama kontaktų centrų organizacinė ir technologinė struktūra. Antroje dalyje pasiūlomas supaprastintas automatinio kontaktų centro technologinis sprendimo variantas. Trečioje dalyje

suprojektuojama ir realizuojama sąsaja tarp kontaktų centro duomenų bazės ir kalbos atpažinimo sistemos.

*Darbe naudoti literatūros šaltiniai.* Rengiant darbą buvo panaudota 19 literatūros šaltinių. Nagrinėti naujausi užsienio ir Lietuvos mokslinių tyrimų, sistemų prototipų, metodologijos aprašymai, moksliniai straipsniai. Šių literatūros šaltinių detalus sąrašas pateiktas darbo pabaigoje. Dauguma šaltinių yra publikuoti ir turi prieigą per Internetą. Nuorodos į šiuos šaltinius pateiktos kiekvienos citatos gale.

*Tyrimo metodai.* Tyrimo metu atliekama kontaktų centrų informacinėse sistemose naudojamų technologijų užsienyje ir Lietuvoje, metodų apžvalga ir apibendrinimas. Įvertinama tyrimų medžiaga dedukcijos, sintezės ir indukcijos metodais. Bandymo metodas pritaikytas balso atpažinimo sistemų tikslumo tyrime. Atliekama mokslinių straipsnių, sistemų vertinimo metodologijos analizė ir sisteminimas, sistemų lyginamoji analizė. Palyginamos sistemų funkcinės komponentės, naudojami metodai, programinė įranga, vystymo galimybės. Atliekamas supaprastinto ir apibendrinto sistemos modelio kūrimas, kuriame išskiriamos bendros gairės, pagrindiniai sistemos elementai ir atliekami procesai.

*Darbo rezultatų teorinė reikšmė.* Darbe pateikti apibendrinti kontaktų centrų informacinės sistemos architektūros ir veiksmų eigos modeliai, naudingi projektuojant naują sistemą. Pasiūlytame architektūros modelyje nurodytos pagrindinės sistemos gairės ir elementai.

*Darbo rezultatų praktinė reikšmė.* Pasiūlytas ir suprojektuotas supaprastintas kontaktų centro modelis CallMagic 2007, realizuotas SQL duomenų bazėje, pavyzdinę klientinę dalį sukuriant Microsoft Visual FoxPro pagalba.

*Rezultatų aprobavimas.* 11-ojoje tarpuniversitetinėje magistrantų konferencijoje „Informacinės technologijos 2006“ buvo perskaitytas ir publikuotas straipsnis tema „Kontaktų centrų organizacinės ir technologinės struktūros analizė“.

*Darbo apribojimai ir sunkumai.* Nagrinėjamos kontaktų centrų technologijos dėl savo specifikos yra labai sudėtingos, todėl Lietuvoje mažai pritaikomos. Kontaktų centrų technologijų integravimas ir sistemos elementų parinkimas atliekamas siekiant supaprastinti tiriamą modeliuojamą sistemą iki plačiai pritaikomo kontaktų centrų technologinio modelio. Testuojant balso atpažinimo sistemų tikslumą buvo naudotas ne telefoninis garsinio ryšio kanalas, kas paprastai naudojama kontaktų centrų technologijose, o asmeninis kompiuteris su tiesiogine garsinio ryšio perdavimo įranga, siekiant sumažinti garso iškrypimus, ir atlikti tik sistemos tiesioginio balso atpažinimo funkcijas.

*Darbo struktūra ir apimtis.* Darbą sudaro kontaktų centrų organizacinę ir technologinę struktūros analizę nagrinėjanti teorinė dalis, analitinė dalis, bei eksperimentinė dalis, kurioje pasiūlytas supaprastintas sistemos modelis, ir realizuotas duomenų bazės projektas. Darbe pateikta



28 paveikslai, 7 lentelės. Darbo apimtis 104 puslapis. Modelių schemas buvo sudaromos naudojantis MS VISIO For Enterprise Architects programiniu paketu. Darbo pabaigoje pateiktos išvados ir pasiūlymai, prieduose – realizacijos programinių kodų pavyzdžiai.

# 1. KONTAKTŲ CENTRŲ ORGANIZACINĖS IR TECHNOLOGINĖS STRUKTŪROS ANALIZĖ

## 1.1 KONTAKTŲ CENTRŲ ORGANIZACINĖ STRUKTŪRA

### Užsienio kontaktų centrai

Nagrinėjant užsienio patirtį (Call centres establishment guidelines: the information of NSW department of Commerce office of information and communications technology), kontaktų centrų organizacinės struktūros formavime, galima pastebėti, kad užsienyje kontaktų centrai, kuriame dirba apie kelias dešimtis darbuotojų, vadovauja kontaktų centro vadybininkas, personalo vadybininkas taip pat yra paskiriamas komandos lyderis. Tuo tarpu, kontaktų centruose, kuriuose dirba daugiau nei 50 operatorių: būna 3-4 komandos lyderiai, vyr.operatorius, specialistas – konsultantas, personalo vystymo specialistas, darbo jėgos poreikį analizuojantis specialistas. Kontaktų centro vadovas priklausomai nuo kontaktų centro dydžio turi keletą pavaduotojų, kurie atsakingi už tam tikras specifines sritis. Tad galima teigti, kad užsienio kontaktų centrų valdymas yra orientuotas į naujausias vadybos tendencijas, akcentuojant žmogiškųjų išteklių svarbą, lankstumą, demokratiškumą, liberalumą. Didžiausių Amerikoje kontaktų centrų sąrašą:

1 lentelė

### Didžiausių Amerikos kontaktų centrų sąrašas

Pavadinimas	Adresas	Darbuotojų skaičius
American Express	20022 N. 31st Ave. Phoenix, AZ 85027	7,000
Discover Financial Services	2402 W. Beardsley Rd. Phoenix, AZ 85027	3,700
Bank of America	1825 E. Buckeye Rd. Phoenix, AZ 85034	2,200
e-Telecare Global Solutions	8901 E. Raintree Dr. Scottsdale, AZ 85260	2,200
JP Morgan Chase	100 W. University Dr. Tempe, AZ 85218	2,000
USAA	1 Norterra Parkway Phoenix, AZ 85027	2,000
The Vanguard Group	14321 N. North Sight Blvd. Scottsdale, AZ 85260	1,500
Charles Schwab & Co., Inc.	2423 E Lincoln Dr Phoenix, AZ 85016	1,337
A Blue Cross Blue Shield of Arizona	2444 W. Las Palmaritas Dr. Phoenix, AZ 85021	1,230
Verizon Wireless	6955 W. Morelos Pl. Chandler, AZ 85226	1,100
Bank One	1515 W. 14th St. Tempe, AZ 85281	1,000

Šaltinis: (Largest Call Centres)

Kaip matome iš aukščiau pateiktos lentelės, savo veikloje kontaktų centrus dažnai steigia didelės ir garsios pasaulinės kompanijos. Yra daug kontaktų centrų pritaikymo sričių, tokių kaip bankai, oro linijos, ir t.t. Tačiau visų kontaktų centrų organizacinė struktūra yra panaši.

Žemiau nagrinėsime kontaktų centrus Lietuvoje, kurios struktūra yra labai panaši į pasaulinių kontaktų centrų organizacines struktūras, tačiau darbuotojų skaičiumi dar toli gražu neprilygsta.

### **Lietuvos kontaktų centrai**

Nagrinėjant **Lietuvos kontaktų centro UAB „Lintel“ organizacinę struktūrą**, įmonėje galima išskirti 4 pagrindinius procesus, kurie ir sąlygoja įmonės struktūrą: paslaugų kūrimas, paslaugų pardavimas įmonėms, paslaugų teikimas ir finansų valdymas (žr. pav.). Įmonėje iš viso veikia 5 paslaugų teikimo skyriai. Praėjusiais metais dėl ekonominės būtinybės – nuostolingų paslaugų teikimo buvo panaikintas Vilniaus paslaugų teikimo skyrius. Kurti naujas ir modifikuoti esamas paslaugas yra įsteigtas rinkodaros ir informacinių technologijų skyriai. Naujų paslaugų pardavimu rūpinasi įmonėje funkcionuojantis pardavimų sektorius. Reikia pabrėžti, kad kiekvieno padalinio darbas UAB „Lintel“ yra grindžiamas komandiniu darbu, darbuotojų tarpusavio glaudžiu bendradarbiavimu. Įmonės valdymas yra demokratinio pobūdžio, kiekvienas darbuotojas yra vertinamas. Įmonėje nuolat yra vykdomi darbuotojų mokymai ir kvalifikacijos kėlimas. UAB „Lintel“ generalinis direktorius taip pat turi keletą pavaduotojų, kurie yra atsakingi už įmonės paslaugų plėtrą, pardavimus, paslaugų teikimą ir finansų valdymą. Galima pastebėti, kad UAB „Lintel“ šiuo atžvilgiu taip siekia neatsilikti nuo naujausių kontaktų centrų veiklos organizavimo tendencijų. To pavyzdys, yra tai, kad paslaugų teikimo procesas UAB „Lintel“ taip pat remiasi komandiniu darbu. Kiekvienai paslaugai teikti yra formuojamos kelios komandos, yra paskiriami komandų lyderiai, yra vertinamas bendras komandos darbas, lyginami atskirų komandų rezultatai. Pavyzdžiui (Kauno paslaugų teikimo skyriaus susirinkimo vykusio 2005 02 01 medžiaga. Kaunas), UAB „Lintel“ paslaugai 1525 teikti yra suformuotos keturios komandos. Vertinant komandų darbą teikiant paslaugas, yra skaičiuojamas bendras pardavimų procentas nuo atsakytų skambučių. Galima teigti, kad UAB „Lintel“ organizacinė struktūra pasižymi tipinėmis kontaktų centrui užsienyje būdingomis savybėmis: struktūra yra vertikalaus hierarchinio pobūdžio, pasižyminti lankstumu, demokratišku valdymu, komandinio darbo, žmogiškųjų išteklių vystymo akcentavimu. Kita vertus, reikalinga pastebėti, kad šiuolaikiniame versle yra svarbu nuolat reaguoti į aplinkos pokyčius, įmonės veiklos rezultatus todėl svarbu užtikrinti, kad kontaktų centrų struktūra būtų keičiama atsižvelgiant į mikro ir makroaplinkos pokyčius. Reikia pabrėžti, kad UAB „Lintel“ nesiekia išlaikyti statiškos struktūros. To pavyzdys yra tas, kad 2004 m. dėl skyriaus veiklos nuostolingumo buvo panaikintas UAB „Lintel“ Vilniaus paslaugų teikimo skyrius. Apžvelgiant tai, kokie iš viso yra išskiriami kontaktų centrų tipai, galima paminėti (Klenke M. (2004) Blueprint for Call center

success: building an effective organizational structure and team environment) tris kontaktų centrų organizacinės struktūros tipus: hierarchinio pobūdžio, mažiau formalizuotą (pagrįstą didelių komandų sudarymu ir šių komandų vadovų paskyrimu, siekiama minimizuoti atstumą tarp aukščiausio lygio vadovų ir žemesnio lygio darbuotojų) ir komandinę struktūrą (komandos veikia savarankiškai, kiekviena komanda yra kompetentinga atlikti jai pavestas užduotis), ir matricinę struktūrą (vadybininkai atliekantys daugialypes pareigas, perduoda dalį atsakomybės savo pavaldiniams, pats koncentruodamasis ties specialiu užduočių atlikimu. Šiuo metu galima išskirti šias kontaktų centrų organizavimo pagrindinius akcentus (Klenke M. (2004) Blueprint for Call center success: building an effective organizational structure and team environment):

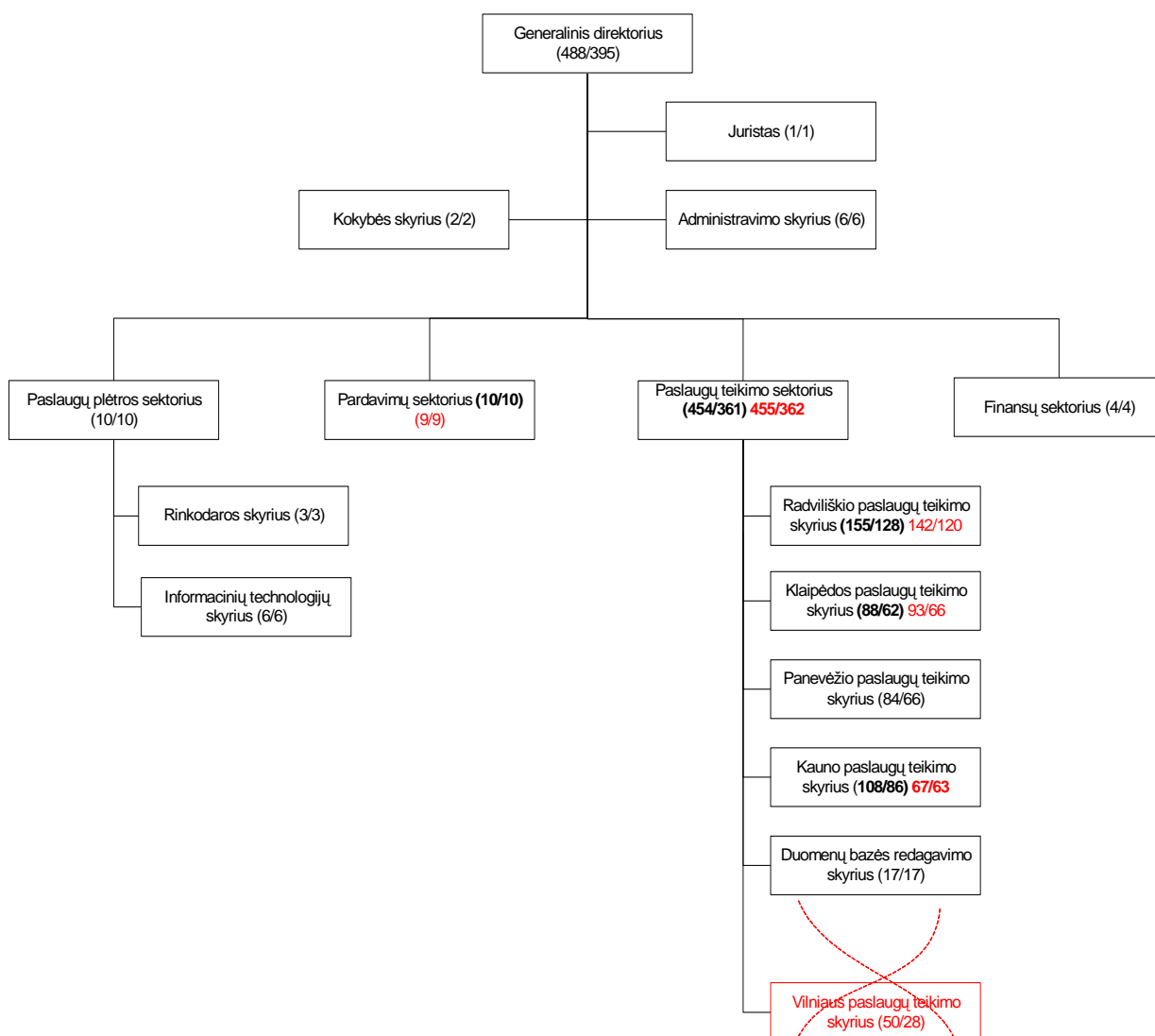
- **Globališkumą.** Dauguma kontaktų centrų pasaulyje yra išsidėstę už šalies, kurioje yra kontaktų centro būstinė, ribų.

- **Lankstumą.** kontaktų centrai turi lanksčiai reaguoti į aplinkos pokyčius, diegti naujausias technologijas. kontaktų centrų struktūros lankstumas gali būti realizuojamas sudarant galimybes centro operatoriams dirbti namuose, samdant pamaininius darbuotojus ir pan.

- **Struktūros plokštumą.** Ši struktūros savybė reiškia, kad kontaktų centrų darbuotojams yra suteikiama daugiau įgaliojimų ir veiklos laisvės.

- **Tinkliškumą.** Pasaulyje pastebima kontaktų centrų tarpusavio tinklų apjungimo, kooperavimosi tendencija, įgalinanti kontaktų centrus sumažinti veiklos kaštus. Ši kontaktų centrų struktūros savybė reiškia, kad paslaugos turi būti teikiamos įvairiomis kalbomis, taip siekiant efektyviau patenkinti kliento poreikius.

## UAB „Lintel“ struktūra



Šaltinis: (Kauno paslaugų teikimo ir duomenų bazių redagavimo skyrių susirinkimo, vykusio 2004 08 05 medžiaga)

### 1 pav. UAB „Lintel“ struktūra

Vertinant UAB „Lintel“ organizacinę struktūrą šių trijų išskiriamų organizacinės struktūros tipų kontekste, galima pastebėti, kad UAB „Lintel“ struktūra yra mišraus pobūdžio: UAB „Lintel“ struktūroje galima aptikti įvairių KONTAKTŲ centrų struktūrų tipų bruožų, išskyrus, tik matricinės. Apskritai reikia pastebėti, kad matricinė valdymo struktūra yra retai naudojama organizuojant KONTAKTŲ centrų veiklą. UAB „Lintel“ struktūra yra hierarchinio pobūdžio (padalinių vadovai yra pavaldūs direktoriui, padalinių vadovams yra pavaldūs konsultantai ir kt.). Kita vertus, UAB „Lintel“ struktūros formalizacijos lygis nėra aukštas. Kiekviename įmonės padalinyje yra siekiama suburti darbo komandas, kurioms yra deleguojama nemažai funkcijų ir

suteikiama didesnė veiklos laisvė. Be to, atsižvelgiant į rinkos poreikius, yra keičiamas komandos narių skaičius, komandai iškeliamos naujos užduotys. Kaip atskirą komandą, kuri turi nemažą veiklos laisvę galima paminėti informacinių technologijų skyrių, kuris nors ir yra pavaldus aukščiausio lygio vadovams, tačiau šis skyrius turi nemažai erdvės vystant esamas ir kuriant naujas KONTAKTŲ centrų paslaugas. Visumoje, galima teigti, kad UAB „Lintel“ organizacinė struktūra nėra pakankamai orientuota į anksčiau minėtas naujausias globalias KONTAKTŲ centrų organizacinės struktūros formavimo tendencijas. Įmonei reikėtų daugiau dėmesio skirti globališkumo ir įvairiapusiškumo principų įgyvendinimui, steigiant centro filialus kitose šalyse ir praplečiant kalbų, kuriomis gali būti teikiama paslauga ratą. Taip pat įmonei reikėtų siekti išplėsti savo partnerių, su kuriais įmonė galėtų kooperotis ir apjungti savo tarpusavyje tinklus, skaičių. Kita vertus, reikia pastebėti, kad įmonė yra nemažai pažengusi, orientuojantis į struktūros lankstumo ir struktūros plokštėjimo tendencijas.

## **1.2 KONTAKTŲ CENTRŲ TECHNOLOGINĖ STRUKTŪRA**

Vertinant **KONTAKTŲ centrų technologinio vystymosi aspektus išsivysčiusiose šalyse**, galima pastebėti, kad užsienyje kontaktų centrų vystymas yra neatsiejamai susijęs su Interneto plėtra, naujų technologijų (IP telefonijos ir kt.) panaudojimu kontaktų centrų veikloje. V.Europos šalyse, JAV jau seniai yra praeitis kontaktų centrų vystymo etapas, kuomet kontaktų centro veikla buvo sufokusuota tik į telefoninių skambučių atsakinėjimą. Galimybė susikalbėti per Internetą, vykdyti asinchronines ir video konferencijas, išsiplėtusios informacijos paieškos galimybės ir kitos technologinės naujovės leido kontaktų centrams lanksčiau ir greičiau aptarnauti savo klientus. Pavyzdžiui, klientas gali pateikti klausimus kontaktų centrams Internetu ir gauti greitą atsakymą. Galima pastebėti, kad kalbinės technologijos pasaulyje šiuo metu yra vyraujanti technologija, kurios naudojimą galima suderinti su įvairiais naujausiais technologiniais prietaisais: PDA, Tablet PCs, smart phones, and desktop PCs. Taip pat Prognozuojama (Rosenberg A. (2004) *IP Myths, Call centers realinies*), kad 2009 m. 80% kontaktų centrų agentų naudosis IP telefonija paremtomis technologijomis, palyginti su dabartiniiais 10%. Naudojamos sudėtingos klausimų ir atsakymų paieškos identifikuoja dažniausiai užduodamus klausimus ir pastarieji yra skelbiami „Dažniausiai užduodamų klausimų skyrelyje. Taip pat yra prognozuojama (Informacijos ir paslaugų centro samprata), kad netolimoje ateityje kontaktų centrų teikiamų paslaugų procesas vyks kaip tiesioginė elektroninė transakcija duomenų bazėje, kuomet paslaugai teikti nebus reikalingas operatorius. Kita vertus, reikia pastebėti, kad vien tik Interneto panaudojimas teikiant kontaktų centrų paslaugas reiškia, kad ne bus visiškai patenkinti kliento poreikiai. kontaktų centrai turi sudaryti galimybes savo klientams tiesiogiai pabendrauti su operatoriumi. Straipsnyje, pristatant naujausias kontaktų centrų technologines naujoves, yra nagrinėjamos dviejų pasaulyje žinomų kompanijų – Unveil ir Avaya Inc. tiekiamą programinę įrangą KONTAKTŲ centrams. Unveil Inc. kompanija kontaktų centrų automatizavimui siūlo du produktus (The features and benefits of Conversation Suite):

- **Unveil Conversation Studio** – tai IBM Eclipse® pagrindu sukurta programinė įranga, kurios pagalba kontaktų centro analitikai gali vystyti virtualaus operatoriaus galimybes.
- **Unveil Conversation Manager** – ši sistema įgyvendina su Conversation studio sukurtus kalbinius variantus. Sistema gali veikti bet kokiame J2EE pagrindu veikiančiame serveryje (pavyzdžiui, tokiuose kaip BEA WebLogic and IBM Websphere). Unveil siūlomų sprendimų pagrindiniai bruožai ir nauda (The features and benefits of Conversation Suite) yra adaptyvus mokymasis, šnekamosios kalbos supratimas, galimybė šalia automatinio operatoriaus paslaugų pasinaudoti ir realaus operatoriaus pagalba. Galima pastebėti, kad Unveil Inc. kompanijos siūloma automatizavimo įranga kontaktų centrams leidžia taupyti darbuotojų darbo laiką, didinti darbuotojų

darbo produktyvumą, sudaro galimybes gauti papildomų pajamų (panaudojant sutaupyta laiką papildomoms pajamoms uždirbti arba steigiant kontaktų centrus kaip papildomą verslo rūšį, taip siekiant išskirti įmonę iš konkurentų).

Kita JAV kompanija siūlanti kontaktų centrams (New Avaya Contact Center Solution Brings Intelligent Communications to Outbound Customer Service) automatizavimo įrangą, prognostines skambinimo technologijas, yra Avaya Inc. Šios kompanijos naudojama techninė įranga padeda efektyviau vykdyti technines apklausas, telefoninius pardavimus. **Pronexus Inc.** Kanados kompanija (The products and solutions of Pronexus Inc. Company) Pronexus Inc. siūlo kontaktų centrams kompiuterinės telefonijos produktą [VBVoice™](#) (bruožai: automatinis kalbos atpažinimas, Text-to-Speech), produktą [VBSALT™](#), skirtą kurti sudėtingas kalbines programas. Tarptautinė kompanija Stratasoft Inc. (Call center software offered by Stratasoft Inc. Company) siūlo kontaktų centrams prognostinį rinkiklį, virtualaus kontaktų centro StrataDial®.VC2™ platformą, technologijas, suteikiančias kontaktų centro agentams dirbti tiesiai iš namų, kontaktų centro darbo kokybės vertinimo technologiją StrataVoice™ ir kt.

**Vertinant UAB „Lintel“ naudojamų technologijų spektrą** su paminėtų kompanijų gaminiais, galima pastebėti, kad UAB „Lintel“ kol kas dar turi nemažai erdvės vystyti savo technologinę bazę, ypač kalbos automatizavimo srityje. UAB „Lintel“ su klientais tiesiogiai bendrauja operatoriai, nėra teikiama automatinio operatoriaus paslaugų. Kaip vieną iš galimybių siekiant efektyviau organizuoti UAB „Lintel“ veiklą naudojant naujausias technologijas, galima išskirti virtualaus kontaktų centro padalinio, kurio darbuotojai dirbtų tik iš namų įsteigimą. Tam tikslui, pavyzdžiui, galima būtų panaudoti virtualaus kontaktų centro StrataDial®.VC2™ platformą, technologiją. Taip pat kaip vienas iš pasiūlymų UAB „Lintel“ tobulinti savo technologinę bazę, būtų kūrimas įvairių kalbinių programų kūrimas ir adaptyvus jų įgyvendinimas. Įmonė galėtų pati susikurti automatinę bazę su įvairiais kalbiniais variantais ir šią bazę nuolat tobulinti. Taip pat UAB „Lintel“, vis labiau Lietuvoje plečiantis Internetinėms technologijoms, galėtų labiau išplėsti ir Interneto panaudojimą, teikiant paslaugas. Pavyzdžiui, įmonė galėtų įsteigti Internetinį padalinį, kuris atsakinėtų vien į elektroninio pašto žinutes.

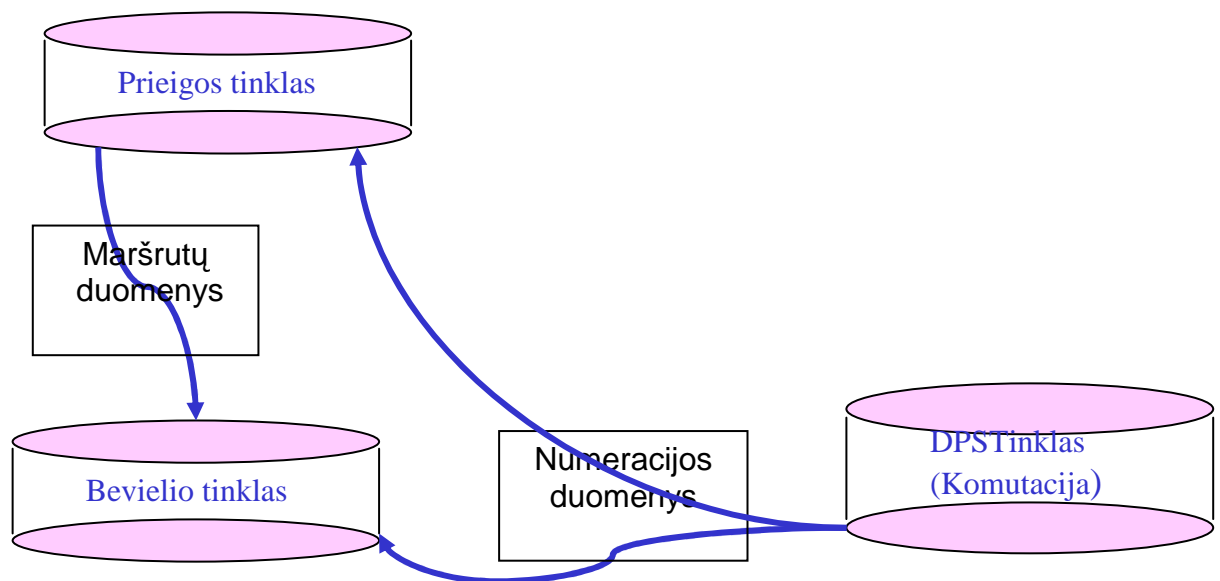
Nagrinėjant AB „Lietuvos telekomo“ TSQL sistemą, naudojamą duomenų perdavimui, galima pastebėti, kad **„Lietuvos telekomo“ TSQL sistema** (Štarolis S., Varkalys G. (2001), p. 3) valdo prieigos (Prieiga), bevielio (Bevielis), perdavimo sistemų, duomenų perdavimo ir komutacijos (DPS tinklas) tinklų bei numeracijos duomenis. Galima pastebėti, kad šiuo metu jau yra aprašyta ir valdoma didžioji dalis „Lietuvos telekomo“ prieigos, bevielio ir komutacijos tinklo duomenų (Štarolis S., Varkalys G. (2001), p. 3):

- Prieigos tinkle yra sudaryti maršrutai, atlikta tinklo resursų analizė,



- Bevieliame tinkle yra įgyvendintas terminalų programavimo palaikymas ir numeracijos „namų zonos“ aprašymas,
- Komutacijos tinkle yra atlikta tinklo elementų lokacijos kontrolė, numeracijos intervalų valdymas.

Kita vertus, dar nemažai funkcijų TSQL tinkle nėra įgyvendinta. Reikalinga prieigos tinkle –įvesti linijų matavimų rezultatus ir atlikti jų analizę. Bevieliame tinkle reikalinga aprašyti GSM tinklo abonentinis terminalus. DPS tinkle reikalinga aprašyti maršrutus, įvesti loginį lygmenį (srautų aprašymas). Kanalizacijos tinkle reikalinga sukurti tinklo modulį, išvystyti interfeisus į kitas sistemas. Visumoje, galima teigti, kad nors nemažai darbų kuriant „Lietuvos telekomo“ TSQL sistemą yra atlikta, tačiau ją dar reikalinga tobulinti, įvedant naujas funkcijas visuose tinklo moduluose. Apibendrinta „Lietuvos telekomo“ TSQL sistemos struktūra yra pateikiama 1 pav. Prieigos tinklo modulis surištas su bevielio tinklo moduliu dėl Prieigos tinklo maršrutų duomenų perdavimui iš Prieigos tinklo duomenų bazės į Bevielio tinklo duomenų bazę. Tuo tarpu, modulis DPSTinklas susijęs su Prieigos tinklo ir Bevielio tinklo moduliu dėl numeracijos duomenų perdavimo iš DPSTinklas į Prieigos tinklo ir Bevielio tinklo duomenų bazę.

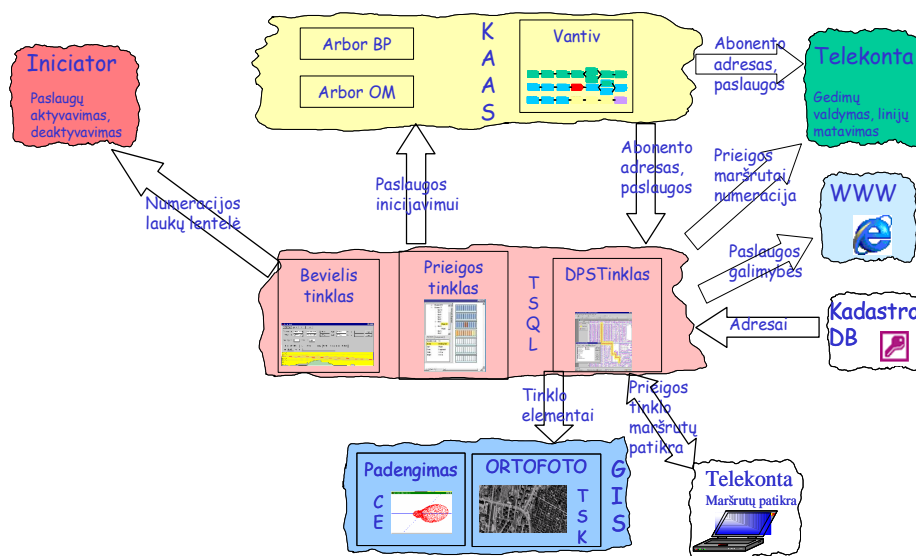


Šaltinis: (Štarolis S., Varkalys G. (2001), p. 14)

## 2 pav. „Lietuvos Telekomo“ TSQL sistemos programų ir modulių tarpusavio sąsaja

Kaip matome iš 2 pav. TSQL sistema yra sujungta sąsajomis su kitų sistemų programomis ir moduliais: Klientų apskaitos ir aptarnavimo sistemos (KAAS) Vantiv ir Arbor OM programomis, Geografinės informacinės sistemos GIS „taškavimo“ aplikacija, Telekontos gedimų šalinimo sistema, Telekontos maršrutų patikros sistema, Kadastrinių adresų duomenų bazė, Interfeisu į WWW puslapį, „Iniciator“ sistema. Šių sąsajų paskirtis yra įvairi: perduoti numeracijos

laukų duomenis, Prieigos tinklo maršrutų, tinklo elementų būsenų pažymėjimo (gedimas, rekonstrukcija, ...) duomenis, vieningo formato kadastrinius vietovės adresų duomenis, DSL Takas paslaugos suteikimo analizės duomenis ir kt.



Šaltinis: (Štarolis S., Varkalys G. (2001), p. 15)

### 3 pav. AB „Lietuvos Telekomo“ TSQ L sistemos sąsaja su kitų sistemų programomis

Reikia pastebėti, kad Lietuvos telekomas planuoja vystyti tiek vidinius, tiek interfeisinius TSQ L sistemos ryšius, tiek. Vystant vidinius (Štarolis S., Varkalys G. (2001), p. 17) TSQ L ryšius, plečiamas sąryšis tarp DPSTinklo modulio ir Prieigos tinklo modulio, įvedus DPSTinklas modulyje Prieigos įrangos aprašymą. Taip pat plečiamas sąryšis tarp DPSTinklo modulio ir Bevielio tinklo modulio, įvedant DPSTinklas modulyje Bevielio bazinių stočių įrangos aprašymą. Tuo tarpu, vystant TSQ L sistemos interfeisinius ryšius, bus siekiama vystyti ryšius su klientų apskaitos ir aptarnavimo sistemos (KAAS) Vantiv programa Tinklo valdymo sistema, Linijų matavimo sistema, „Adress Book“ duomenų baze, Geografinės informacinės sistemos (GIS) „Celurar Expert“ programa. Apibendrintai galima teigti, kad „Lietuvos telekomo“ TSQ L sistema turi būti vystoma tobulinant tiek jos vidinę struktūrą, tiek sąryšius su aplinkinėmis sistemomis.

## **1.2.1 AUTOMATINIŲ KONTAKTŲ CENTRŲ TECHNOLOGIJOS**

### **1.2.2 Kalbos Atpažinimas**

Siekiant sumodeliuoti automatinio kontaktų centro idėją, naudosis balso atpažinimo technologijomis, kurios jau yra taikomos sudėtingose kontaktų centrų sistemose, tačiau kurios, kaip matome iš aukščiau pateiktos technologinės analizės, Lietuvoje dar nėra praktikuojamos dėl jų sudėtingumo, bei aptarnavimo bei naudojimo kaštų.

Kalbos atpažinimas (daugelyje kontekstų taip pat žinomas sąvoka automatinis kalbos atpažinimas, kompiuterinės kalbos atpažinimas ar dar kitaip, beje klaidingai laikomas balso atpažinimu) yra procesas paverčiantis, konvertuojantis kalbos signalą į žodžių seką ir vykdomas kompiuterinės programos algoritmo pagalba.

Kalbos atpažinimas taikomas pastaraisiais metais pasirodžiusiose taikomose programose:

- tokiose kaip telefono rinkimas balsu (pvz., užklausa „Skambinti į namus“),
- skambučio paieška (pvz., „Norėčiau skambinti kito žmogaus sąskaita“),
- nesudėtingų duomenų įvedimas (pvz., kredito kortelės numerio įvedimas),
- struktūrinių dokumentų paruošimas (pvz., rentgenologinės ataskaitos),
- suteikiama galimybė valdyti namų apyvokos reikmenis ir ieškoti turiniu paremtos kalbinės audio informacijos.

Balso atpažinimas arba kalbančiojo atpažinimas yra susijęs procesas, kurio tikslas nustatyti kalbantį asmenį priešingai tam kas sakoma.

Technologine prasme, šiandieninės techninės srities knygos pabrėžia Paslėptojo Markovo modelio reikšmę kaip esminę technologiją. Dinamiško programavimo, neuroninių tinklų ir intelektualinis mokymosi metodas buvo intensyviai tiriama sritis 1980-1990 metais.

### **1.2.3 Kalbos atpažinimas hierarchinėse struktūrose**

Bėgant laikui, ir plečiantis informaciniams ištekliams vis labiau susiduriame su problema, kuomet tampa vis sunkiau valdyti sudėtingas hierarchines struktūras. Kalbos atpažinimas šioje srityje labai padeda. Ši problema ypatingai aktuali kontaktų centruose, kuomet hierarchijos valdymas atliekamas mygtukų paspaudimu. Esant sudėtingai struktūrai skambinantysis gali prarasti kantrybę ir neišlaukti reikiamos informacijos vien dėl to, kad norint prie jos prieiti reikia atlikti sudėtingo meniu valdymą, dėl šios priežasties krenta ne tik kontaktų centro našumas, bet ir prarandami klientai. Įdiegus kalbos atpažinimą kontaktų centre, ši problema išsisprendžia. Skambinantysis iš karto gali pasakyti ko ieško, ir iš karto gauti atsakymus pagal ieškomus žodžius.

#### **1.2.4 Kalbos atpažinimo sistemų charakteristika**

Kalbos atpažinimo sistemų veikla yra dažniausiai apibūdinama tikslumo ir greičio prasme. Tikslumas matuojamas žodžių klaidų dažniu, tuo tarpu greitis yra ypatingai įdomi technologija.

Dauguma kalbos atpažinimo sistemos vartotojų pritarę, jog diktavimo mašinos labai tikslingai veikia valdomose sąlygose. Tačiau dalis painiavos atsiranda iš įvairaus termino „kalbos atpažinimas“ ir „dikcija“ naudojimo.

Nuo kalbančiojo priklausančios diktavimo sistemos, kurioms reikalingas trumpas apmokymas, gali užfiksuoti nepertraukiamą kalbą su plačiu žodžių visumos, leksikos naudojimu kalbant normaliu tempu ir labai dideliu žodžių tikslumu. Dauguma komercinių bendrovių teigia, kad atpažinimo technologijos gali pasiekti 98% - 99% tikslumą (vienas, du žodžiai iš šimto gali būti neteisingi) jeigu dirbama optimaliose sąlygose. Tokiomis sąlygomis laikomi testavimo subjektai, kurie turi:

- 1) susieti kalbančiojo charakteristiką su apmokymo duomenimis,
- 2) tiksliai kalbančiojo kalbą adaptuoti
- 3) šis procesas turi vykti aiškioje aplinkoje (pvz., biure). Šiuo būdu galime suprasti kodėl kai kurie programos vartotojai, ypač turintys akcentą, mato, kad programos supratimo greitis gali būti gana mažesnis nei 98% - 99%.

Kitos, riboto žodyno, sistemos, kurioms nereikia jokio apmokymo, gali atpažinti nedidelį skaičių žodžių (pvz. dešimt skaičių). Tokios sistemos yra tinkamos nustatyti skambučius ir jų paskyrimo vietą didelėse kompanijose. (Speech recognition. From Wikipedia, the free encyclopedia)

#### **1.2.5 Klaidų skaičius kalbos atpažinime**

Lentelėje apačioje pateiktas apytikris neteisingų žodžių procentumas (t.y. procentinis klaidingų žodžių skaičius) moderniausiose automatinėse kalbos atpažinimo sistemose.

Nukrypimai pasitaikantys dėl triukšmo trukdžių ir akcento gana nemažai padidina klaidų skaičių. Pvz. stipraus japonų ar ispanų kalbos akcento kalbant angliškai apdorojimas gali būti 3-4 kartus klaidingesnis nei toks pats gražia anglų kalba kalbančiojo kalbos apdorojimas (Tomokiyo, 2001). O pridėjus automobilio triukšmą 10dB SNR klaidų gali pasitaikyti 2-4 kartus daugiau.

**Apytikris klaidingų žodžių skaičius.**

<b>Užduotis</b>	<b>Žodynas</b>	<b>Klaidų skaičius %</b>
TI Digits	11 (0-9)	0,5
Wall Street Journal skaitoma kalba	5000	3
Wall Street Journal skaitoma kalba	20000	3
TV naujienų transliacija	64000+	10
Telefoninė kalba (CTS)	64000+	20

Apytikris klaidingų žodžių skaičius (% neatpažintų žodžių) paskelbtų ištyrus 2006 m. kalbos atpažinimo sistemas įvairiais tiklais; TV naujienų transliacijose ir telefoninėje kalboje remiamasi apmokymais ir testavimu.

Šaltinis: (Automatic Speech Recognition)

Bendraja prasme, šie klaidų skaičiai kiekvienais metais vis mažėja, kadangi kalbos atpažinimo sistemos nuo jų atsiradimo daug pasikeitė ir patobulėjo. Tokių sistemų tikslumas per pastarąjį dešimtmetį padidėjo apie 10% dėl algoritminių poslinkių ir Moore dėsnio. (Automatic Speech Recognition)

### ***1.3 KALBOS ATPAŽINIMO MODELIAI***

Beveik visos šiuo metu rinkoje siūlomos KAS (Kalbos Atpažinimo Sistemos) remiasi paslėptaisiais Markovo modeliais (Hidden Markov Models - HMM). Jais aprašomas fonemų ir alofonų tarimas bei šių tarimų trukmė. Markovo modeliai jau taikomi tris dešimtmečius, ir jų populiarumas tik didėja. "HMM artinys turi neginčijamų pranašumų. Matematikos požiūriu jis yra labai patogus, lengvai suprantamas ir įgyvendinamas", - tvirtina Masačusetso technologijos instituto kompiuterijos laboratorijos vadovas Viktoras Zue.

Be abejo, esama ir kitų artinių. Kitos KAS kūrėjos - kompanijos Sensory ir Fonix (Solt Leik Sitis, Jutos valstija) pasiūlė vadinamųjų neuroninių tinklų veikimo principą. Neuroniniai tinklai geriau nei HMM atpažįsta atskirus žodžius, tik pirmiau juos reikia "pamokyti". Anot E.Soule'o, neuroniniam tinklui reikia "pasakyti", kokių žodžių jis turėtų ieškoti ir kokių neieškoti. Tik tuomet jis nustato esminius požymius, pagal kuriuos ieškomus žodžius galima atskirti nuo panašiai skambančiųjų.

Kai reikia atpažinti ne pavienius žodžius, o sakinius ar skaičių sekas, paslėptieji Markovo modeliai yra pranašesni už neuroninius tinklus. Dabartinė Sensory kalbos atpažinimo sistema Fluent Speech taiko hibridinį - neuroninių tinklų ir paslėptųjų Markovo modelių - artinį.

Kad būtų sukurta duomenų bazė, su kuria reiktų lyginti išstartus žodžius, HMM irgi reikia "pamokyti". Kai kurias užduotis atlikti sistemą moko pats kalbėtojas. Tuomet atpažinimo sistema mokės atpažinti būtent to žmogaus kalbą. Kitais atvejais kalbančiųjų gali būti daug. Tuomet sistemos kūrėjai parenka daugelio žmonių kalbos pavyzdžių ir sudaro mokymo duomenų vidurkį. (Pokalbis su mašina)

### 1.3.1 HMM modelis kalbos atpažinime

HMM modelio populiarumą įtakoja tai, kad šias sistemas galima automatiškai apmokyti, jos yra paprastos ir kompiuteriu lengvai valdomos. Kalbos atpažinimo procese, paslėptasis Markovo modelis išvestų seką  $n$ -dimencinių realios reikšmės vektorių, kurių  $n$  yra, sakykim, 13, išvedant tokius  $n$  kas kiekvieną 10 milisekundės. Vektoriai, vėlgi labai paprastu atveju, susidėtų iš garso absorbcijos koeficientų, kurie gaunami paėmus trumpalaikio kalbos lango Fourier transformaciją ir atsiejus spektrą naudojantis kosinuso transformacija, o tuomet paėmus pirmąjį (ir patį svarbiausią) koeficientą. Paslėptasis Markovo modelis turi statistinį paskirstymą, kuris vadinamas įstrižinio nukrypimo Gauso metodu, kuris priskiria kiekvieną pastebėtą vektorių. Kiekvienas žodis, arba (daugiau bendrinėms kalbos atpažinimo sistemoms) kiekviena fonema, turės skirtingą išvesties paskirstymą; Paslėptasis Markovo modelis žodžių ar fonemų sekai yra sukuriamas sujungiant individualiai „apmokytus“ paslėptuosius Markovo modelius skirtingiems žodžiams ir fonemoms.

Viršuje apibūdinti svarbiausi HMM modelio elementai. Modernios kalbos atpažinimo sistemos naudoja keletą standartinių technikų kombinaciją siekiant pagerinti pagrindinio viršuje apibūdinto modelio rezultatus. Tipinei didelio leksikos kiekio sistemai reikėtų, kad kontekstas:

- Būtų priklausomas nuo telefono.
- Naudotų garso absorbcijos normalizavimo funkciją, su kurios pagalba gali sunorminti skirtingų kalbėtojų ir įrašymo sąlygas;
- Tolesniam kalbėtojo normalizavimui naudotų vokalinio trakto ilgio normalizavimą (VTLN), kai normalizuojamas vyro-moters balsas ir maksimalaus panašumo linijinė regresija (MLLR) adaptuojant paprastus kalbėtojus.
- Šios sistemos savybės yra taip vadinami delta ir delta-delta koeficientai, kurie pagauna kalbos dinamiką ir gali naudoti heteroskedastinės kreivės diskriminacinę analizę (HLDA);
- Galima praleisti delta ir delta-delta koeficientus ir naudoti sandūrą bei LDA paremtą projekciją, po kurios seka tikriausiai heteroskedastinės kreivės diskriminacinė analizė arba apmokymo technikos, kurios vykdomos pagal grynai statistinį modelį ir taip nustatomi HMM parametrai.
- Pavyzdžiais gali būti maksimali bendrinė informacija (MMI), minimali klasifikacinė klaida (MCE) ir minimali telefono klaida (MPE).

Dekoduojant kalbą (šis terminas naudojamas apibūdinti kas atsitinka kai sistema išgirsta naują dikciją ir turi apdoroti tinkamiausią jai sakinį) tikriausiai bus naudojamas Viterbi algoritmas, kuris leidžia surasti geriausią sprendimą, be to galima pasirinkti tarp dinamiško paslėptojo Markovo

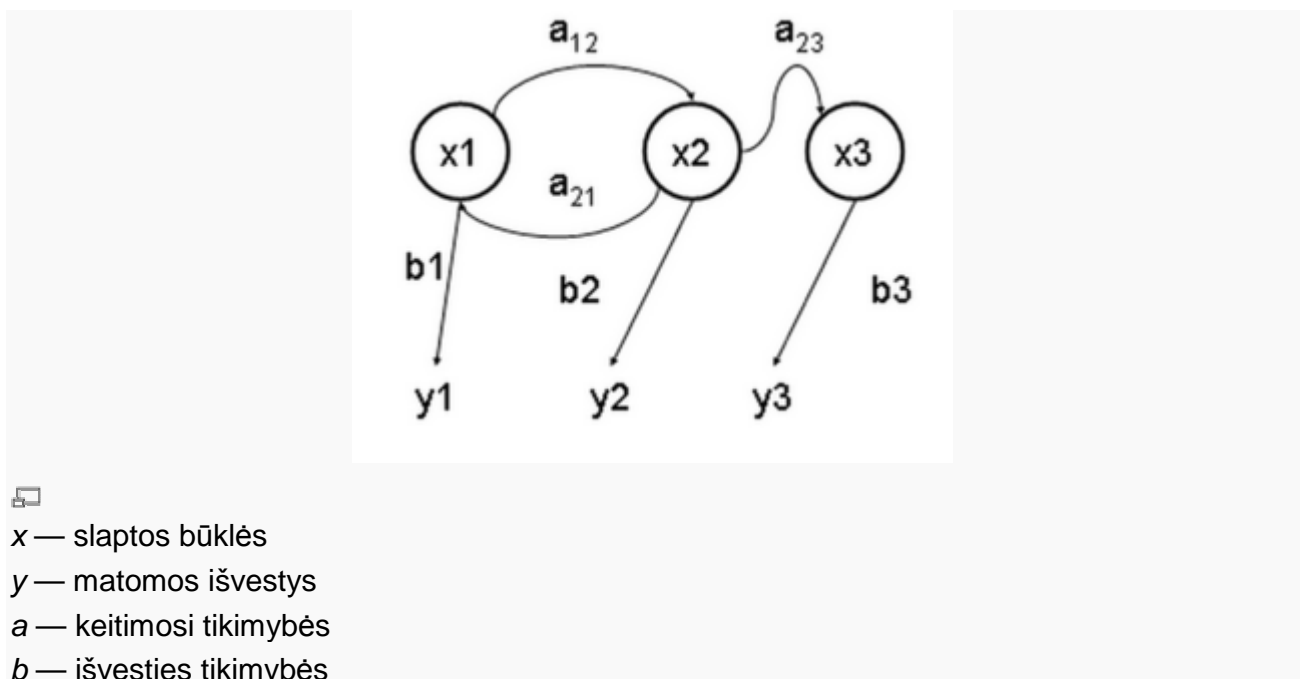
modelio, kuriame yra tiek akustinė tiek ir kalbos modelio informacija, arba rinktis statistinį modelį. (Speech recognition. From Wikipedia, the free encyclopedia)

### 1.3.2 HMM modelio architektūra.

Paslėptasis Markovo modelis (HMM) yra statistinis modelis, kuriame sumodeliuota sistema laikoma Markovo procesu su nežinomais parametrais, ir čia svarbu nustatyti tuos slaptuosius parametrus iš žinomų parametrų. Išimto modelio parametrai tuomet gali būti naudojami atlikti tolesnę analizę, pvz. modelio atpažinimo taikymas. HMM gali būti laikomas paprasčiausiu dinaminiu Bajeso tinklu.

- Paprastame Markovo modelyje padėtis yra stebėtoju matoma ir būklės pasikeitimo galimybės yra vieninteliai parametrai.
- Paslėptajame Markovo modelyje, būklė nėra tiesiogiai matoma, tačiau būklės įtakojami kintamieji čia matomi.
- Kiekvienoje būklėje paskirstoma tikimybė. Tačiau leksemų sekos HMM modelyje vis dėlto suteikia šiek tiek informacijos apie sekos būklę.

Paslėptieji Markovo modeliai yra ypač gerai žinomi dėl jų taikymo atpažįstant laikinus modelius, tokius kaip kalba, raštas, gestų atpažinimas, muzikos partitūros sekimas ir bioinformatika.



$x$  — slaptos būklės

$y$  — matomos išvestys

$a$  — keitimosi tikimybės

$b$  — išvesties tikimybės

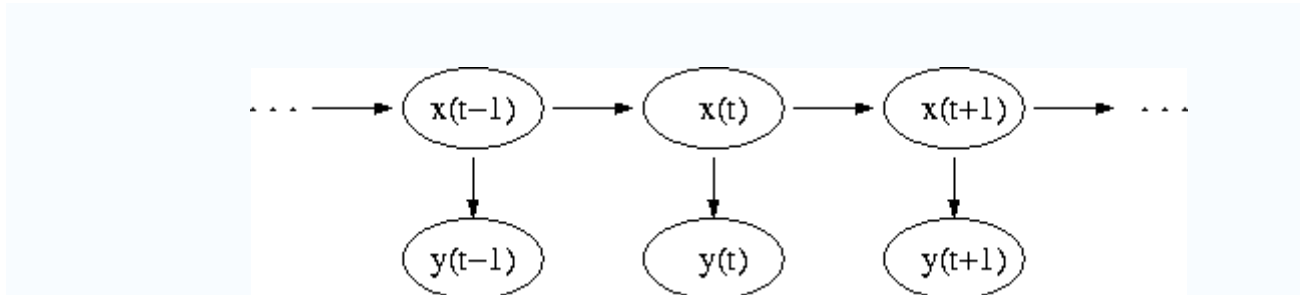
Šaltinis: (Hidden Markov Model. From Wikipedia, the free encyclopedia)

**4 pav. Būklės keitimasis Paslėptajame Markovo modelyje (pavyzdys).**



Žemiau pateikta diagrama nurodo bendrąją HMM modelio sudėtį. Kiekviena ovali forma yra atsitiktinis kintamasis, kuris gali priimti keletą reikšmių. Atsitiktinis kintamasis  $x(t)$  yra slaptąjo kintamojo reikšmė laike  $t$ . Atsitiktinis kintamasis  $y(t)$  yra matomo kintamojo reikšmė laike  $t$ . Diagramos rodyklės pažymi sąlygines priklausomybes.

Iš diagramos matome, kad slaptąjo kintamojo  $x(t)$  (laike  $t$ ) reikšmė priklauso tik nuo slaptąjo kintamojo  $x(t - 1)$  (laike  $t - 1$ ) reikšmės. Tai yra Markovo ypatybė. Panašiai ir matomo kintamojo  $y(t)$  reikšmė priklauso tik nuo slaptąjo kintamojo  $x(t)$  (abu laike  $t$ ) reikšmės.



Šaltinis: (Hidden Markov Model. From Wikipedia, the free encyclopedia)

**5 pav. Bendroji HMM modelio sudėtis.**

#### **Matomos sekos tikimybė.**

Sekos stebėjimo tikimybė  $Y = y(0), y(1), \dots, y(L - 1)$  ilgyje  $L$  yra:

$$P(Y) = \sum_X P(Y | X)P(X)$$

Čia suma pakartoja visas galimas slaptosios kreipties sekas  $X = x(0), x(1), \dots, x(L - 1)$ .  $P(Y)$  Bruto jėgos apskaičiavimas yra sunkiai valdomas daugelyje realių gyvenimiškų problemų atveju, kadangi galimų slaptųjų kreipčių seka gali būti labai didelė. (Hidden Markov Model. From Wikipedia, the free encyclopedia)

### **1.3.3 Neuroniniu tinklu paremtos kalbos atpažinimas**

Kitas akustinio modeliavimo metodas yra neuroninių tinklų naudojimas. Jie gali atlikti daug sudėtingesnes atpažinimo užduotis. Vietoj to, kad būtų naudojama bendraja prasme, kalbos atpažinimo sistema gali susidoroti su prastos kokybės, triukšmingais duomenimis ir nepriklausomai nuo kalbėtojo. Tokios sistemos gali pasiekti didesnę tikslumą nei HMM paremtos sistemos, kadangi neribojami nei apmokymo nei leksikos duomenys. Dar bendresnis metodas naudojant neuroninius tinklus yra fonemų atpažinimas. Tai aktyvi tyrimo sfera, tačiau bendrai tariant, jos rezultatai yra geresni nei HMM. Egzistuoja taip pat ir NN-HMM hibridinės sistemos, kurios naudoja neuroninių tinklų dalį atpažįstančią fonemas ir paslėptojo Markovo modelio dalį modeliuojančią kalbą.

### **1.3.4 Dinaminiu laiko iškraipymu (DLI) paremtas kalbos atpažinimas**

Dinaminis laiko iškraipymas yra metodas, kuris jau seniai buvo naudojamas kalbos atpažinimui, tačiau šiai laikais yra dažnai keičiamas sėkmingesniu HMM paremtu modeliu. Dinaminis laiko iškraipymas yra algoritmas skirtas apskaičiuoti panašumus tarp dviejų sekų, kurios gali skirtis laiko arba greičio atžvilgiu. Pvz., ėjimo panašumus galima atskirti netgi jeigu vienoje video juostoje žmonės ėjo lėtai ir jeigu kitoje juostoje, abu žmonės ėjo greičiau arba netgi jeigu jie darė pagreitėjimus arba lėtino tempą viso stebėjimo metu. DLI naudojamas video, audio ir grafiniais pavidalais – taipogi su DLI pagalba galima analizuoti bet kokius duomenis, kuriuos galima įtraukti į linijinį pavaizdavimą.

Gerai žinomas jos taikymas yra automatinis kalbos atpažinimas ir sugebėjimas susitvarkyti su skirtingais kalbos greičiais. Bendraja prasme, tai metodas kuris padeda kompiuteriui rasti optimalų atitikimą tarp dviejų pateiktų sekų (pvz. laiko serijų) su tam tikrais apribojimais, t.y. sekos yra „iškraipomos“ nelinejiniu būdu, kad jos viena su kita sutaptų. Šis sekos sulyginimo metodas dažnai naudojamas paslėptojo Markovo modelio kontekste. (Speech recognition. From Wikipedia, the free encyclopedia)

## **1.4 KALBOS ATPAŽINIMO PROGRAMINĖ ĮRANGA**

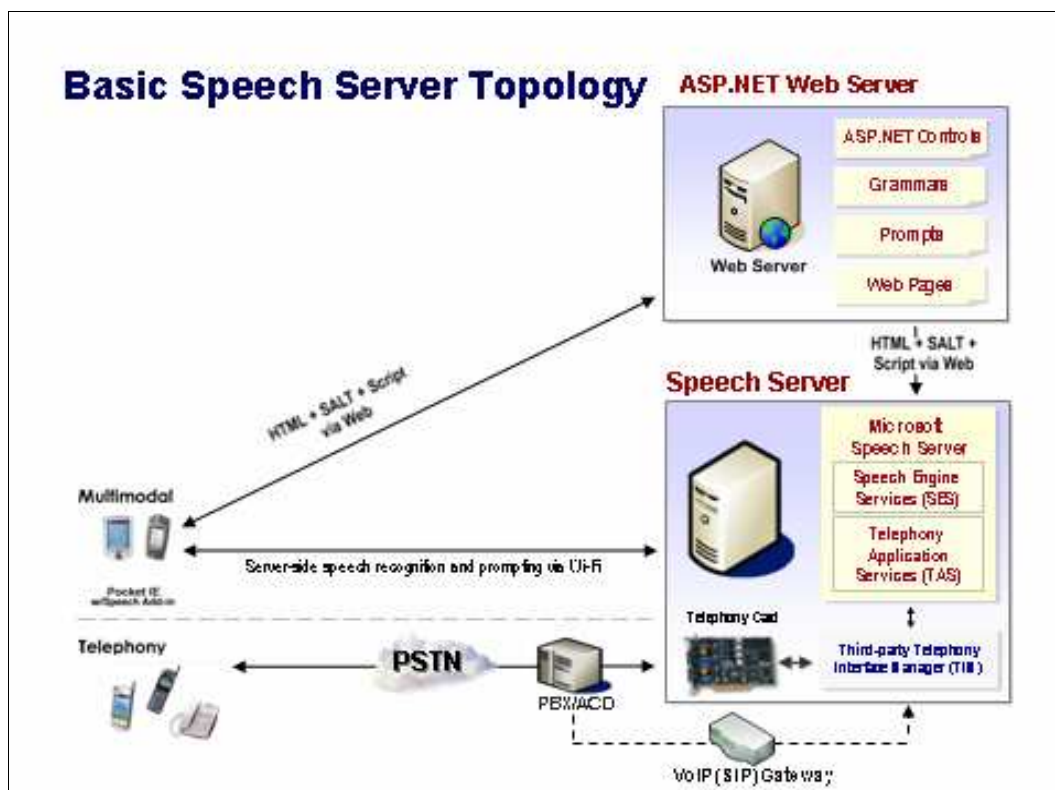
### **1.4.1 HTK programinė įranga**

Paslėptojo Markovo modelio programinė įranga (HTK) yra nešiojama įranga skirta kurti ir valdyti paslėptąjį Markovo modelį. HTK yra pirmiausiai naudojama kalbos atpažinimo tyrimui, nors ji naudojama ir keletui kitų sričių, įskaitant tyrimus apie kalbos sintezę, simbolių atpažinimą ir DNA sekas. HTK naudojama daugelyje pasaulinių svetainių.

HTK sudaro bibliotekos modelių rinkinys ir įranga C programinės kalbos formoje. Įranga suteikia sudėtingų galimybių analizuoti kalbą, HMM apmokymą, testavimą ir rezultatų analizę. Įranga palaiko HMM naudodama tiek nepertraukiamo tankio Gauso metodų mišinį, tiek ir pavienes distribucijas ir gali būti naudojama sukurti sudėtingas HMM sistemas. HTK mechanizmą sudaro daugybė dokumentų ir pavyzdžių.

HTK buvo sukurta Mašininio intelektualumo laboratorijoje (anksčiau žinomoje kaip Kalbos vizijos ir Robotų grupė) priklausančioje Kembridžo universiteto inžinerijos katedrai (CUED), kur ši įranga buvo naudojama sukurti didžiulę CUED kalbos atpažinimo sistemą. 1993-iais metais Entropic Research Laboratory Inc. įgijo teises parduoti HTK ir HTK tobulėjimas buvo pilnai perduotas Entropic laboratorijos atsakomybėn 1995 m., kai susikūrė Entropic Cambridge Research Laboratory Ltd. HTK Entropic pardavė dar iki 1999 m., kai Microsoft nusipirko Entropic. Microsoft dabar jau atgavo teises į HTK, atidavė CUED ir teikia pagalbą CUED, kad ši vėl HTK padalintų ir tobulėtų per HTK3 svetainę. (What is HTK?)

## 1.4.2 Microsoft Speech Server programinė įranga



Šaltinis: (Balso dialogų technologijos (bdt): kada prakalbinsime kompiuterį lietuviškai?)  
**6 pav. Microsoft balso serveris MSS 2004 R2 2005.**

**MS Speech Server programinės įrangos raida** (Balso dialogų technologijos (bdt): kada prakalbinsime kompiuterį lietuviškai?):

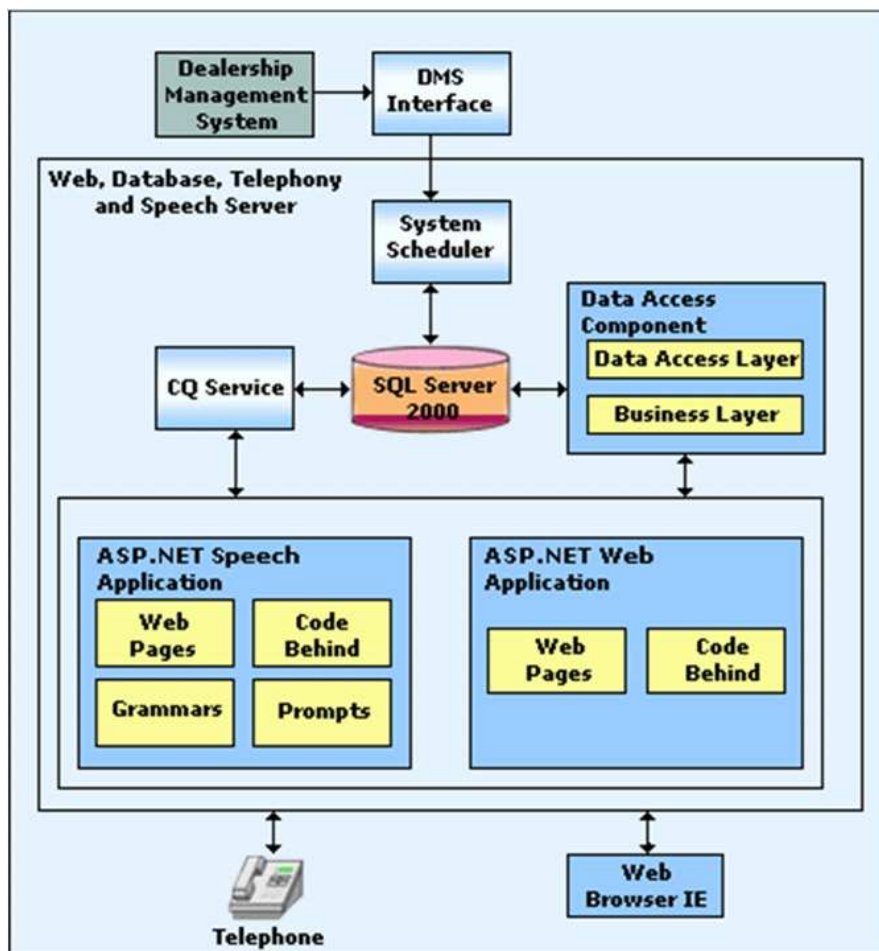
### Microsoft balso serveris MSS 2004

- projektuotas 11 metų, pirmoje versijoje viena kalba (US English)
- trys pagrindiniai vartotojai: stacionarus PC, planšetinis (tablet) PC, mobilus įtaisas
- interneto serveris, balso serveris, telefonija, projektavimo priemonės
- balso serveryje yra SES ir TAS

### Microsoft balso serveris MSS 2004 R2

- trys kalbos ( US English, US French, US Spain )
- internetinė telefonija (VoIP)

Žemiau pateiktame paveiksle matome, kaip MS Speech Server programinė įranga komunikuoja su duomenų baze SQL Server, kurią ir naudosime Kontaktų Centrų programinės įrangos prototipo kūrimo. Dėl šios priežasties kalbos atpažinimui ir buvo nuspręsta naudoti būtent šia kalbos atpažinimo programinę įrangą. (Balso dialogų technologijos (bdt): kada prakalbėsime kompiuterį lietuviškai?)

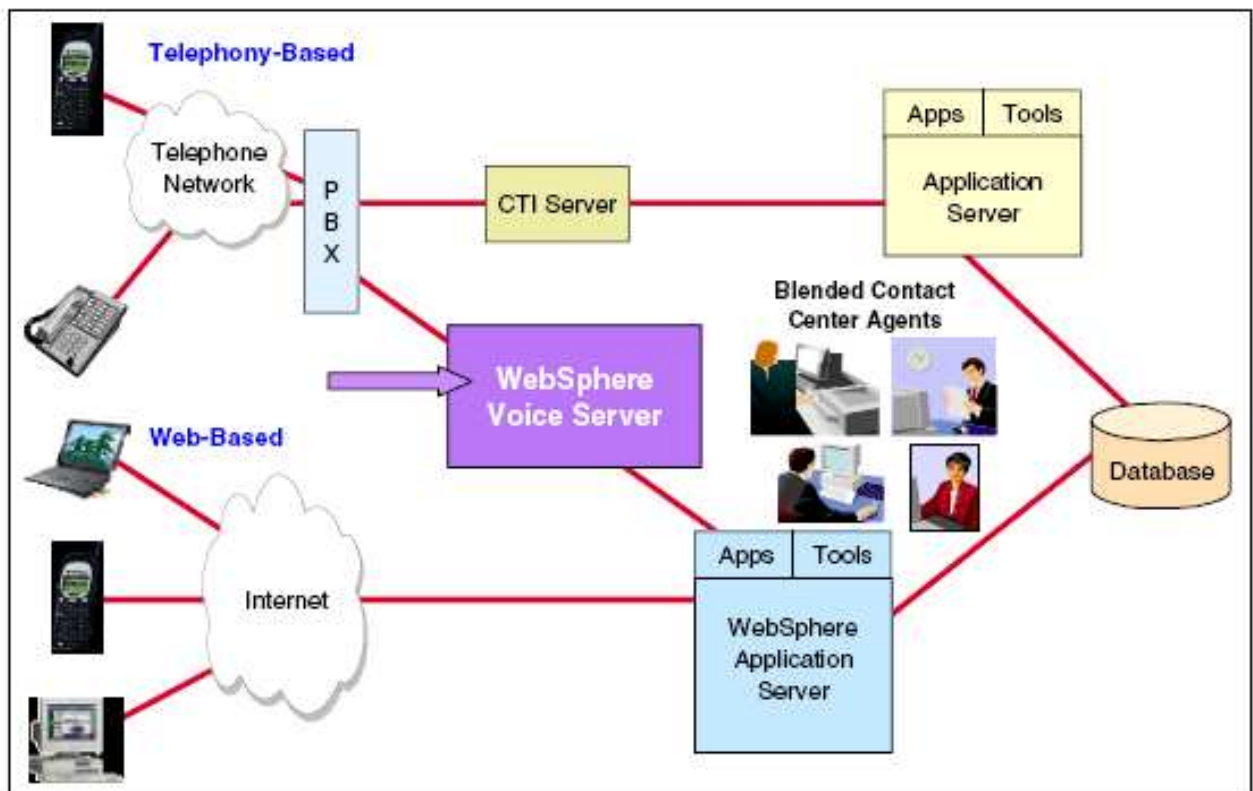


Šaltinis: (Balso dialogų technologijos (bdt): kada prakalbėsime kompiuterį lietuviškai?)

7 pav. MS Speech Server sąveika su duomenų bazėmis.

### 1.4.3 IBM WebSphere programinė įranga

IBM Websphere yra programinės įrangos produktų šeima, skirta aplikacijų infrastruktūrai, verslo sistemų integravimui, portalams, el. komercijai, mobiliam prieigai ir prieigai balsu.



Šaltinis: (Balso dialogų technologijos (bdt): kada prakalbėsime kompiuterį lietuviškai?)

8 pav. IBM WebSphere balso serveris

## **1.5 ESAMŲ KONTAKTŲ CENTRŲ TECHNOLOGINIŲ SPRENDIMŲ TRŪKUMAI**

Nagrinėjant kontaktų centrus technologiniu požiūriu buvo nustatę keletas trūkumų, dėl kurių sudėtingėja kontaktų centrų plėtra Lietuvoje. Šie trūkumai yra:

- **Esamų rinkoje programinių produktų sudėtingumas.** Labai dažnai Lietuvos rinkos poreikiams tenkinti nereikia tokių sudėtingų ir problematiškų kontaktų centrų technologinių sprendimų, tačiau keletą naujausiomis technologijomis paremtų modulių, tokių kaip kalbos atpažinimas, visgi norėtusi pritaikyti Lietuvos kontaktų centrų veikloje. Tačiau nėra tokio supaprastinto automatinio kontaktų centro technologinio sprendimo, ir norint įdiegti šias naujoves reikia rinktis būtent šiuos sudėtingus produktus.

- **Esamų rinkoje programinių produktų brangumas.** Dėl esamų rinkoje programinių produktų sudėtingumo stipriai išauga šių produktų diegimo ir palaikymo kaštai. Dėl šios priežasties mažoje Lietuvos rinkoje esantiems kontaktų centrams tiesiog neapsimoka diegti tokias galingas sistemas, nes šie atnaujinimai tiesiog neatsipirks.

- **Vartotojo interfeiso sudėtingumas.** Dėl esamų produktų sudėtingumo vartotojo interfeisas pasidaro ypatingai sudėtingas, kas visiškai nėra reikalinga paprastus Lietuvos rinkos poreikius tenkinančiuose kontaktų centruose. Įdiegus tokį sudėtingą automatinio kontaktų centro vartotojo interfeiso variantą Lietuvoje bus prarandami klientai, kurių ir taip nėra labai daug.

- **Pritaikomumas Lietuvių kalbai.** Dideli ir sudėtingi kontaktų centrų programiniai produktai dažniausiai yra pritaikomi didelių šalių kalboms, tokioms kaip anglų ar vokiečių. Lietuvių kalbai pritaikytų kontaktų centrų programinių produktų rinkoje nėra išvis.

## **2. AUTOMATINIO KONTAKTŲ CENTRO TECHNOLOGINIO SPRENDIMO VARIANTAS**

Atlikus kontaktų centrų organizacinės ir technologinės struktūros analizę ir įvertinus trūkumus buvo nuspręsta pasiūlyti savo kontaktų centro technologinio sprendimo variantą, sumodeliuoti projektą sukuriant savitą kontaktų centrąs programinę įrangą pasižyminčia galimybėmis, kurios būdingos dideliems kontaktų centrų programiniams paketams, tačiau išlaikant kiek įmanoma paprastesnę struktūrą, paliekant tik esminius dalykus.

Projekto pavadinimas CallMagic 2007.

Programinė įranga apims šias pagrindines sritis:

- Vartotojų registravimą;
- Įmonių kontaktės informacijos administravimą ir kaupimą
- Gyventojų kontaktės informacijos administravimą ir kaupimą
- Balso atpažinimą ir interpretavimą i kodinę sistemą
- Kontaktų paiešką pagal balso atpažinimo sistemos kodus.
- Vartotojų istorijos peržiūrą

Sumodeliuoti sąsają tarp kontaktų centro duomenų bazių ir kalbos atpažinimo sistemos kompiuterizuojant visos minėtos sritys išskyrus balso atpažinimo ir interpretavimo modulius, kuriems bus naudojamas jau sukurtas programinis paketas MS Speech Server.

Sukurti pavyzdinį tekstinį vartotojo interfeisą, kuriame būtų naudojamos tos pačios kodinės komandos, kaip ir gautos iš balso atpažinimo ir interpretavimo sistemos, o rezultatai pateikiami, tokie, kokius pateiks automatinio balso operatorius.



## 2.1 PROJEKTO STRUKTŪRA



Šaltinis: sukurta autoriaus.

**9 pav. Projekto CallMagic 2007 struktūra.**

Aukščiau pateiktame paveiksle matome vartotojų, administratorių ir duomenų bazių kontaktų centro technologinių sistemų sąveiką tarpusavyje. Projekto sudedamosios dalys:

- 2 duomenų bazės: kontaktų duomenų bazė, garsinės informacijos duomenų bazė
- 2 veikėjai: sistemos administratorius, vartotojas
- 2 technologinės sistemos: skambučių valdymo centras, balso atpažinimo ir interpretavimo sistema.
- Įvairios užklauskos ir duomenų bazių ryšiai.

**Kontaktų duomenų bazė.** Šioje duomenų bazėje kaupiama visa kontaktų centro informacija. Šiuo atveju joje bus laikoma įmonių kontaktinė informacija, tokia kaip, telefonas, adresas, pavadinimas, imones rekvizitai, direktoriaus informacija, ir gyventojų informacija, tai vardas pavardė, gyvenamoji vieta kontaktai. Šioje duomenų bazėje taip pat kaupiami kiekvieno paieškos kriterijaus sisteminiai kodai pagal kuriuos joje bus vykdoma paieška. Šie kodai automatiškai priskiriami įvedant naują kontaktinę eilutę, ir yra identiški garsinės duomenų bazės paieškos kriterijų kodams.

**Garsinės informacijos duomenų bazė.** Šioje duomenų bazėje kaupiama garsinė paieškos kriterijų informacija. Jai priklauso kiekvieno ieškomo žodžio garsinės transkripcijos, taip pat šia kaupiama informacija apie pasikeitimus paieškos kriterijuose (pvz. Pasikeite Įmones pavadinimas, ir vartotojas ieško pagal senąjį pavadinimą, o rezultatus gauna jau su nauju pavadinimu). Šioje duomenų bazėje taip pat kaupiami kiekvieno paieškos kriterijaus sisteminiai kodai iš kurių kalbos atpažinimo sistema atpažinusi paieškos kriterijų pasakytą balsu, formuos užklauską į kontaktų duomenų bazę. Šie kodai automatiškai priskiriami įvedant naują kontaktinę eilutę ir garsinę informaciją jos paieškai, ir yra identiški kontaktų duomenų bazės paieškos kriterijų kodams.

**Sistemos administratorius.** Sistemos administratorius koreguoja ir prideda naują kontaktinę informaciją į kontaktų duomenų bazę ir taip pat koreguoja ir priskiria transkripcijas ar pasikeitimus kiekvienam paieškos kriterijui.

**Vartotojas.** Telefonu skambina į kontaktų centrą ir balsu skelbia užklauskas jai, kur rezultatus į šias užklauskas gauna taipgi balsu. Vartotojas taip pat internetu gali registruoti naują vartotojo profilį, ir užsiregistravęs peržiūrėti savo kontaktinės informacijos paieškų istoriją.

**Skambučių valdymo centras.** Tai telefoninė stotis valdanti organizacijos skambučius, taipogi nukreipianti vartotojus į automatinį kontaktų centrą.

**Balso atpažinimo ir interpretavimo sistema.** Tai sistema suprantanti telefonu skambinančiojo vartotojo kalbą, pagal garsinėje duomenų bazėje esančias transkripcijas, taip

pat perduodanti vartotojui garsinę informaciją apie paieškos rezultatus gautus iš kontaktų duomenų bazės.

**Įvairios užklausos ir duomenų bazių ryšiai.** Šiame projekte duomenų bazės komunikuoja SQL užklausų pagalba. Vartotojai užklausas formuoja tiesiog balsu. Administratorius turi atskirą grafinį vartotojo interfeisą, kuriame koreguoja duomenų bazių informaciją.

## **2.2 INFORMACIJOS SISTEMAI KELIAMI REIKALAVIMAI**

Projekto sukūrimui ir proceso kompiuterizavimui bus naudojami programiniai paketai: *Microsoft Office Visio for Enterprise Architects*, *Microsoft Visual FoxPro 9.0*. Serveriui bus naudojama *MS SQL Server 2005* serverio tipo duomenų bazės sistema.

Sistemos projektavimas atliekamas *Microsoft Office Visio for Enterprise Architects* pagalba.

Kuriamos sistemos architektūra bus pritaikyta *MS Speech Server* kalbos atpažinimo priemonei. Tai sistema turinti pilną integraciją į duomenų bases, taipgi gerai komunikuojanti su kitais Microsoft produktais, ir turinti labai platų ir išsamų šios sistemos pagalbos ir paaiškinimų spektrą.

Kuriamos informacinės sistemos DBVS pasirinkta *Microsoft Visual FoxPro 9.0*, nes tai gana paprasta, bet ir daug galimybių teikianti DBVS. Jos pilnai užtenka kompiuterizuoti pasirinktą veiklos sritį. Be to, ji lengvai sąveikauja su kitais Microsoft Office paketais.

Šie trys programiniai paketai lengvai sąveikauja tarpusavyje – *Microsoft Office Visio for Enterprise Architects* aplinkoje sukurtą duomenų bazės schemą galima lengvai konvertuoti į *MS SQL Server 2005* duomenų bazę, ir atvirkščiai.

### **3. PROJEKTO REALIZACIJAI REIKALINGŲ BALSO KOMANDŲ ATPAŽINIMO TIKSLUMO TYRIMAS**

Planuojamam projektuoti kontaktų centro modelio tikslumui didžiausią įtaką turi kalbos atpažinimo sistema, nuo kurios gebėjimo tiksliai interpretuoti skambinančiojo kalbą į kodinę sistemą ir priklausys viso projekto naudingumas, ir praktinis pritaikomumas.

Buvo nuspręsta bandymo būdu atlikti dviejų programinių produktų kalbos atpažinimo tikslumo tyrimą. Šie programiniai produktai yra:

- HTK – HMM (Hidden Markov Model) Markovo modelio kalbos atpažinimo priemonė;
- MS Speech Server. Tai plačiai paplitusi Serverio tipo kalbos atpažinimo sistema, Microsoft Office Communication Server programinio paketo dalis.

Šio tyrimo tikslas, nustatyti bendrųjų projektuojamos kontaktų centro informacinės sistemos komandų atpažinimo tikslumą, ir statistiškai įvertinant klaidų kiekį surasti procentinę šio tikslumo išraišką.

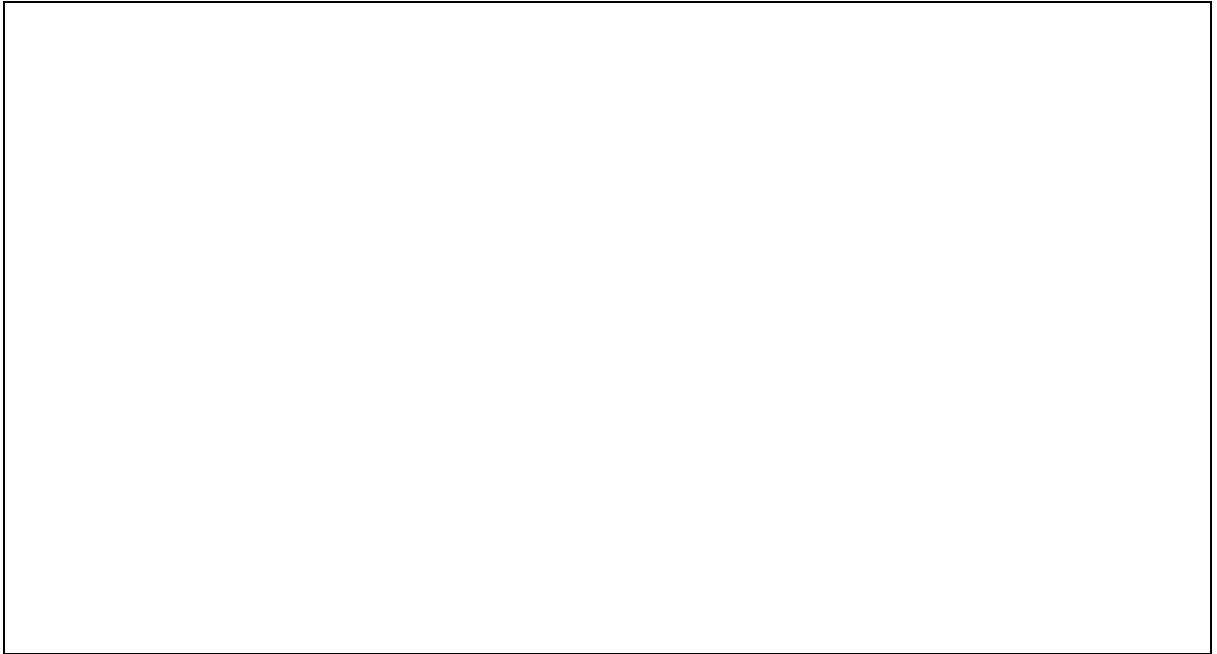
#### **3.1 BANDYMAS SU HTK PRODUKTU**

Kalbos atpažinimui buvo naudota 5 – 6 būsenų HMM grandinės.

Pirmojo programinio produkto HTK bandymo metu buvo pasirinktos tokios pagrindinės komandos:

- Pirmyn
- Atgal
- Į pradžią
- Į galą
- Tęsti
- Pauzė

Bandymui atlikti buvo pasirinktos 225 moterys, ir 125 vyrai diktoriai. Bandymo metodika pateikta žemiau.



Šaltinis: sukurta autoriaus.

**10 pav. HTT bandymų metodika su moterimis.**



Šaltinis: sukurta autoriaus.

**11 pav. HTT bandymų metodika su vyrais.**

Norint atlikti pasirinktų komandų testavimą, būtina nustatyti šių komandų lietuviškų garsų transkripcijas į anglų kalbos garsus. Žemiau pateiktoje lentelėje matome šių komandų transkripcijas, bei testavimo rezultatus.

**HTT Testavimo rezultatai.**

<b>Žodis</b>	<b>Vyrai, %</b>	<b>Moterys, %</b>	<b>Transkripcija</b>
Pirmyn	87	84	PEERRMEEN
Atgal	91	89	AATGAAL
Į pradžią	78	81	EE PRAADJA
Į galą	81	76	EE GAALAA
Tęsti	86	85	TASTEE
Pauzė	82	78	PAAUZYE

Šaltinis: sukurta autoriaus.

Iš gautų rezultatų matome, jog vyrų ištartos komandos beveik visais atvejais buvo atpažintos nežymiai tiksliau.

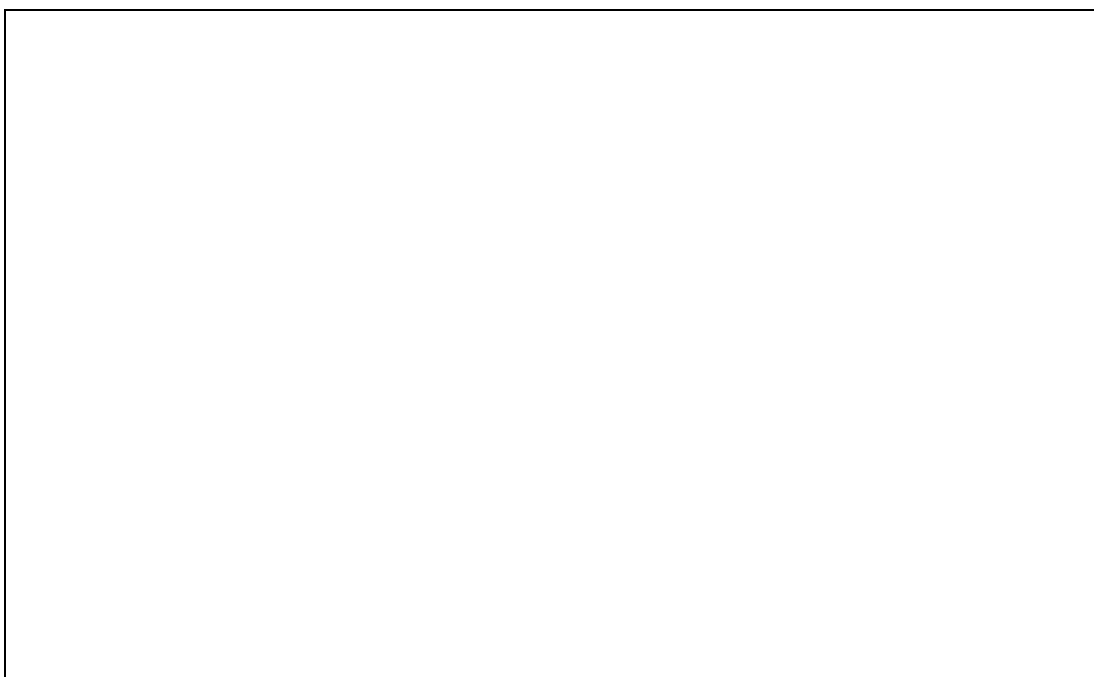
### **3.2 BANDYMAS SU MS SPEECH SERVER PRODUKTU**

Antrojo programinio produkto bandymas buvo vykdomas dviem etapais.

**Pirmojo** etapo metu buvo atliktas sistemos gebėjimas atpažinti tam tikrus sunkiai atpažįstamus dvigarsius:

- TI
- DI
- TA
- DA

Bandymui atlikti buvo pasirinkti 4 vyrai diktoriai. Bandymo metodika pateikta žemiau.



Šaltinis: sukurta autoriaus.

### **12 Pav. MS Speech Server bandymų metodika dvigarsiais.**

**MS Speech Server testavimo rezultatai su dvigarsiais.**

<b>Žodis</b>	<b>Vyrai, %</b>
TI	18
DI	12
TA	99,5
DA	3

Šaltinis: sukurta autoriaus

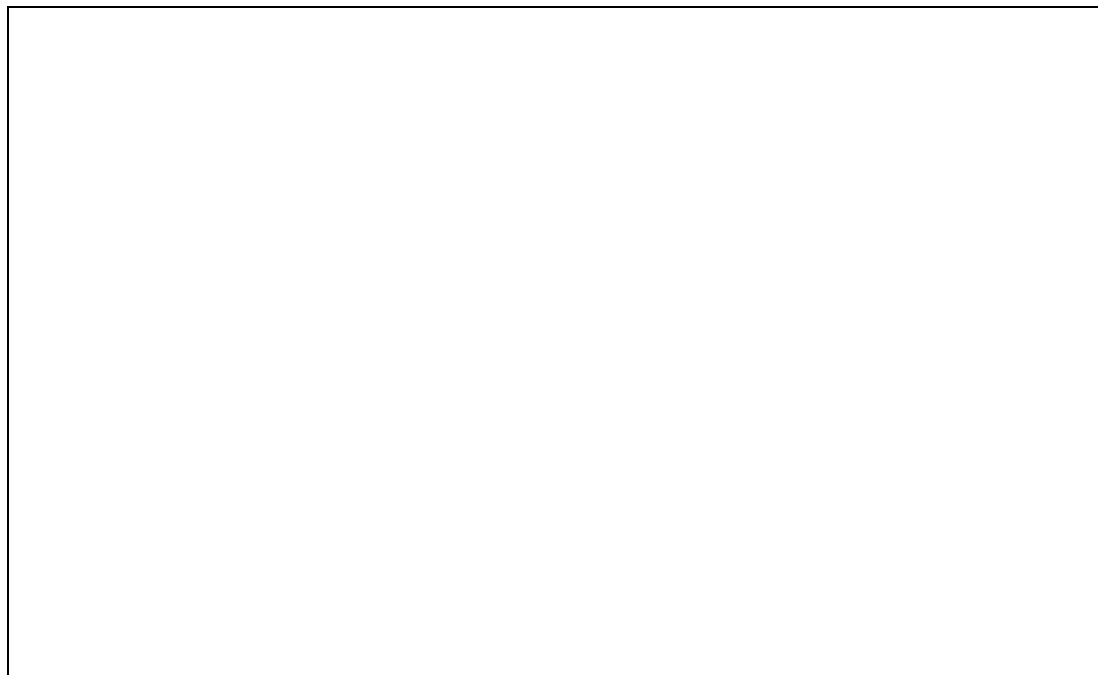
Iš gautų rezultatų matome, jog sistema labai sunkiai atpažįsta dvigarsį DA, ir labai gerai atpažįsta dvigarsį TA, kai tuo tarpu fonetiškai sunkesnius dvigarsius kaip TI ir DI atpažįsta vienodai tiksliai, ir gerokai tiksliau už dvigarsį DA.

**Antrojo** etapo metu kaip ir prieš tai buvusiame HTT bandyme buvo pasirinktos pagrindinės likusios projektuojamos sistemos komandos. Šios komandos mūsų sistemoje simbolizuoja kiekvieną galimą kontaktų paieškos kriterijų, tad labai svarbu, kad sistema jas atpažintų tiksliai. Šios komandos yra:

- Įmonės,
- Gyventojai,
- Įmonės kodas
- Įmonės pavadinimas
- Miestas
- Gatvė
- Veiklos rūšis,
- Gyventojų pavardė
- Direktorius pavardė

Bandymui buvo pasirinkti 2 vyrai diktoriai, ir 2 moterys. Bandymo metodika pateikta žemiau.





Šaltinis: sukurta autoriaus.

**13 pav. MS Speech Server bandymų metodika komandomis.**

5 lentelė.

**MS Speech Server testavimo rezultatai, bei transkripcijos.**

<b>Žodis</b>	<b>Vyrai, %</b>	<b>Moterys, %</b>	<b>Transkripcija</b>
Įmonės,	87	84	EEMONYES
Gyventojai,	91	89	GEEVENNTOIAI
Įmonės kodas	78	81	EEMONYES KKODDASS
Įmonės pavadinimas	81	76	EEMONYES PPAVVADDINNIMMASS
Miestas	86	85	MYEEESTASS
Gatvė	82	78	GGATTVIE
Veiklos rūšis,	81	89	VA KLOSS ROOSHISS
Gyventojo pavardė	86	81	GEEVENNTOIO PAVARDIE
Direktoriaus pavardė	82	76	DIRECTORAUS PAVARDE

Šaltinis: sukurta autoriaus.

### 3.3 *TYRIMO IŠVADOS.*

- Iš gautų rezultatų atliekant tiek HTK tiek MS Speech Server kalbos atpažinimo tikslumo tyrimą galime daryti išvadą, jog pasirinktos komandos pilnai gali būti naudojamos mūsų projekte.

- Kontaktų centrų prototipo kūrime naudosime MS Speech Server programine įranga, dėl jos gebėjimo geriau komunikuoti su duomenų bazėmis ir dėl platesnių aprašymų ir pagalbos resursų.

- HTK programinė įranga buvo atmesta dėl to, jog ji yra daugiau pritaikyta mokomiesiems tikslams, nei realiam produktui išbaigti.

- Diktorių vyrų kalbą sistema atpažįsta nežymiai tiksliau nei moterų, tačiau skirtumas yra toks nedidelis, kad realios įtakos sistemoms naudojimuisi jis neturės.

- Nors ir kiek yra pažengusios garso atpažinimo technologijos, tačiau iki šimtaprocentinio tikslumo ar bent artimo jam dar dar toli.

- Visos atpažinimo sistemos labiausiai išvystytos yra Anglų kalba, tad norint atpažinti žodžius išartus lietuviškai, šiems žodžiams reikia parinkti teisingas transkripcijas.

## 4. AUTOMATINIO KONTAKTŲ CENTRO DUOMENŲ BAZĖS SĄSAJOS SPRENDIMAS

### 4.1 TECHNINĖ UŽDUOTIS

#### 1. TEMA:

Informacijos valdymo kontaktų centruose prototipas

#### 2. SUPROJEKTUOTI, REALIZUOTI IR PARUOŠTI FUNKCIJAS:

- 2.1. Įmonių kontaktinių duomenų administravimą,
- 2.2. Pirkėjų kontaktinių duomenų administravimą,
- 2.3. Įmonių direktorių kontaktinių duomenų administravimą
- 2.4. Vartotojų informacijos administravimą
- 2.5. paieškos kriterijų administravimą
- 2.6. Kontaktinių duomenų paiešką demonstracinėje konsolėje
- 2.7. Paieškos duomenų registravimą vartotojui
- 2.8. Išvedimo informacijos ruošimą:

- Paieškos rezultatų peržiūra
- Vartotojo paieškos istorijos peržiūra

#### 3. PARUOŠTI SISTEMOS NAUDOJIMO DOKUMENTUS:

- 3.1. Vartotojo naudojimo instrukciją
- 3.2. Instrukciją administratoriui

#### 4. REIKALAVIMAI PROJEKTAVIMUI, PROGRAMINEI IR

#### TECHNINEI ĮRANGAI

- 4.1. Projektavimui naudoti paketą Microsoft Office Visio for Enterprise Architects
- 4.2. Duomenų bazių valdymo sistema: Visual FoxPro v. 9.0, serverio programinė įranga MS SQL Server 2000.

4.3. Sistema turi funkcionuoti Windows 9X/2000/XP/VISTA aplinkoje.

4.4. *Minimalūs reikalavimai techninei įrangai:*

Serveriui: Pentium4 1.6GHz, 512 MB RAM, 80Gb HDD,

Klientui: Pentium3 800MHz, 128Mb RAM, Spausdintuvas, kiti į standartinę

kompiuterio konfigūraciją įeinantys aksesuarai

4.5. Reikalavimai programinei įrangai:

- MS Office paketas (rekomenduojama 2003 metų versija)
- Spausdintuvo tvarkyklė

#### 5. SISTEMOS TESTAVIMAS IR ĮVERTINIMAS

- 5.1 Paruošti kontrolinį duomenų rinkinį sistemos testavimui.
- 5.2 Įvertinti testavimo rezultatus ir pateikti juos magistrinio darbo aprašyme

#### 6. REIKALAVIMAI DARBO PRISTATYMIUI:

- 7.1. Pateikti darbo aprašą pagal pateiktą magistrinio darbo struktūrą
- 7.2. Pateikti kompaktinį diską su programų paketu, kontrolinio pavyzdžio duomenimis, bakalaurinio darbo tekstu.

7.3. Gynimo metu pademonstruoti visas galimas sistemos funkcijas.

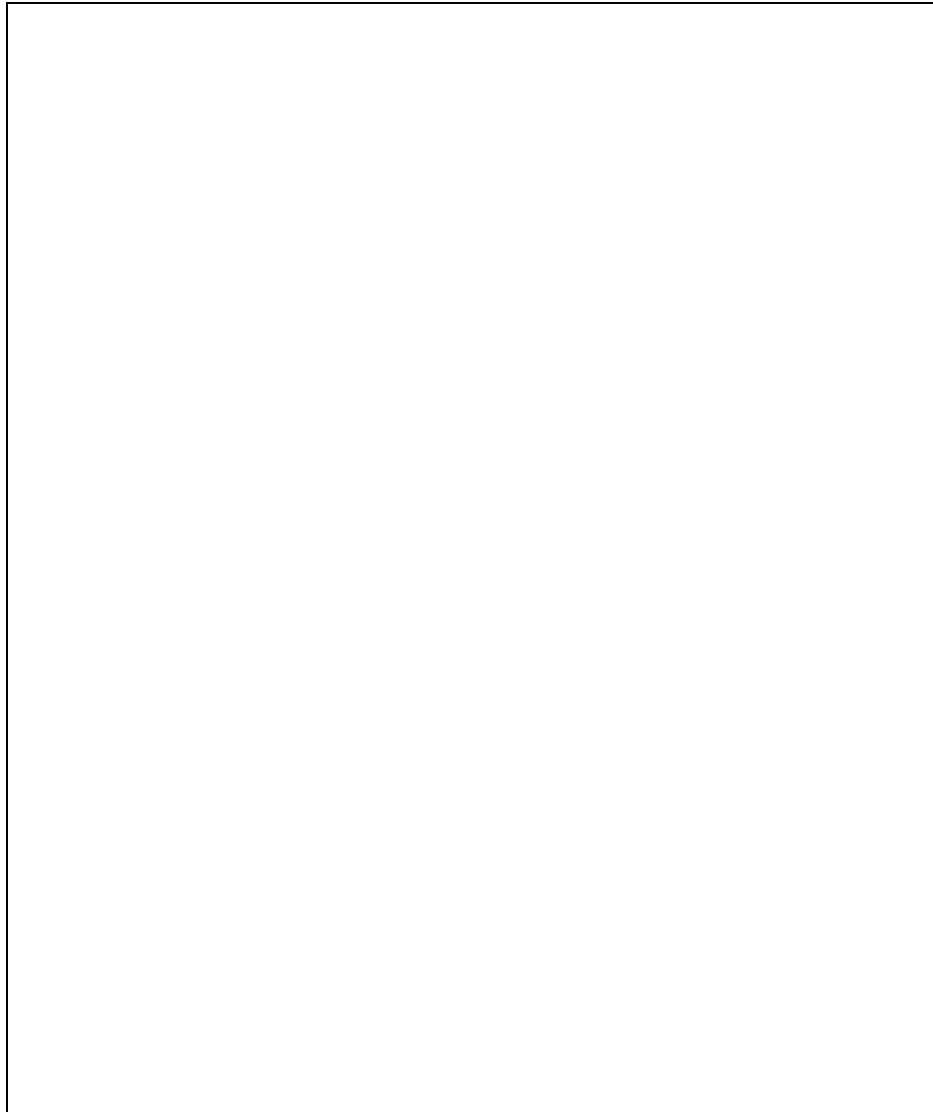
Robertas Butkevičius –VIM2gr. studentas

## **4.2 KOMPIUTERIZUOJAMŲ FUNKCIJŲ HIERARCHIJA**

Kontaktų centrų informacinės sistemos prototipo paskirtis – registruoti naujus kontaktinius duomenis, administruoti vartotojo profilį, pateikti vartotojo informaciją, atlikti paiešką pagal kintamą kriterijų skaičių.

Žemiau pateiktoje schemoje pavaizduota funkcijų hierarchija grafiškai apibūdina pasirinktų kompiuterizuoti sričių funkcijas.

Funkcijų hierarchijos diagrama atspindi visu skambinančiojo veiksmų procesus.



Šaltinis: sukurta autoriaus

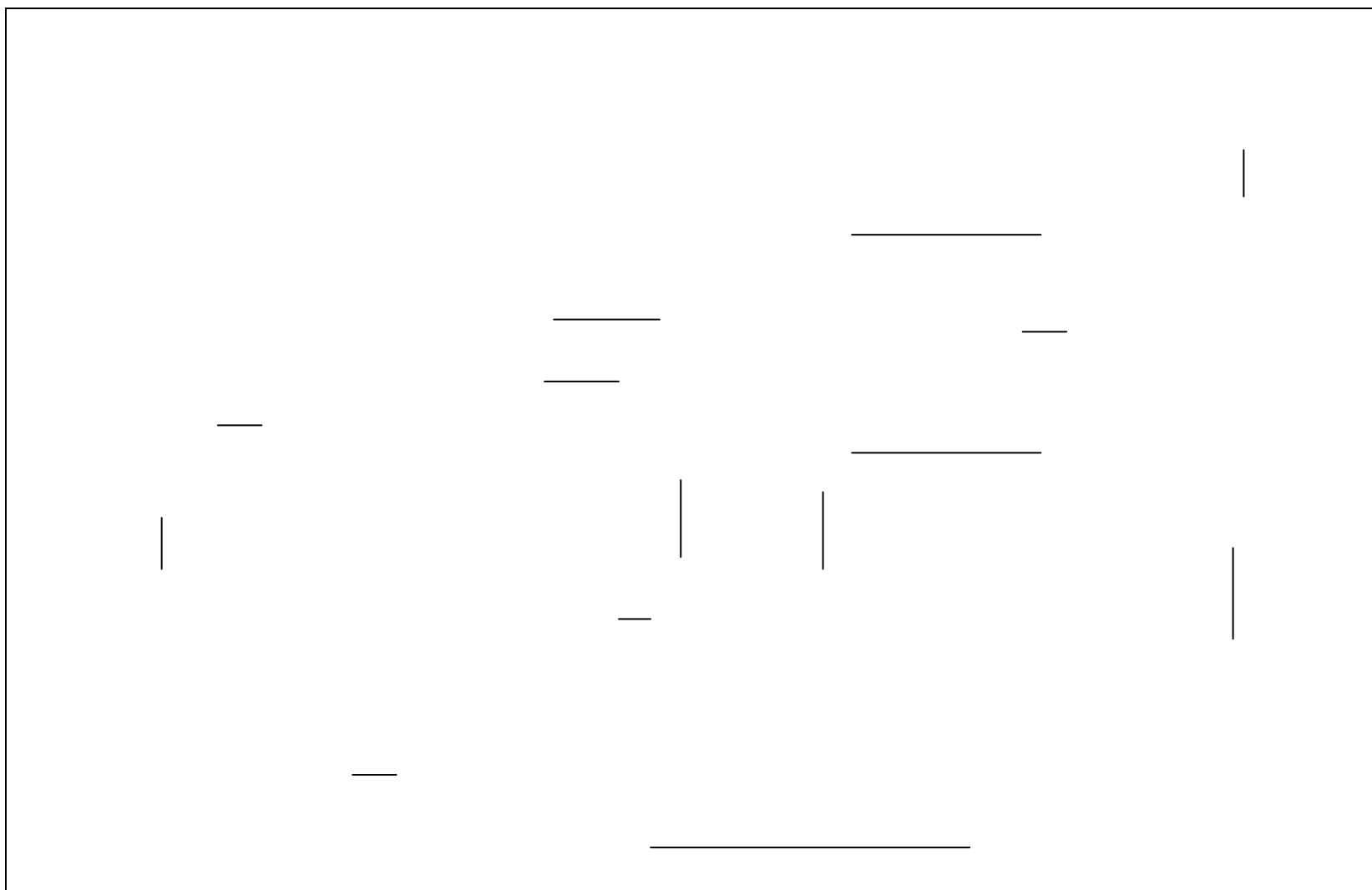
**14 pav. Kompiuterizuojamų funkcijų hierarchija**

### **4.3 KONCEPCINIS OBJEKTO MODELIS (ER-DIAGRAMA)**

Esybių modeliavimo pagrindas – diagrama, skirta sistemos statinei daliai atvaizduoti. Populiariausias esybių modeliavimo metodas – ER modelis.

ER schema apibūdina ryšius tarp lentelių. Objektai pasižymintys panašiomis savybėmis sudaro vieną klasę, vadinamą esybe. ER diagramoje priklausomybė atvaizduojama ryšiu tarp esybių. Ryšiai tarp esybių nusakomi pavadinimais apibūdinančiais santykį tarp jų (Sekliuckis Vitolis, Garšva Gintautas, Gudas Saulius, 2003, p. 141-144).

Žemiau pateiktame paveiksle ER modelyje yra atvaizduojamos esybės sujungtos ryšiais.



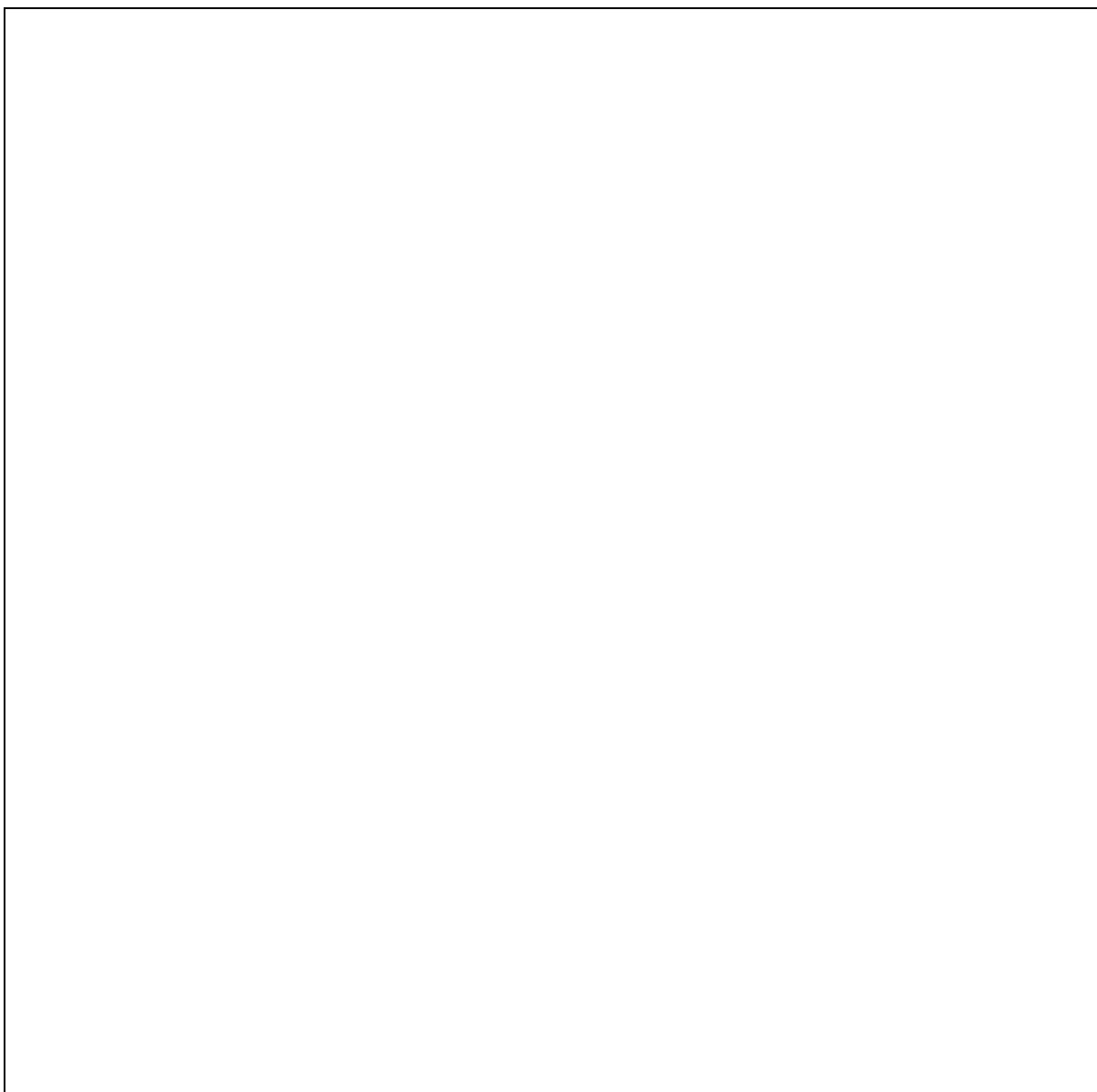
Šaltinis: sukurta autoriaus

15 pav. ER modelis

#### **4.4 PROJEKTUOJAMOS SISTEMOS DUOMENŲ SRAUTŲ DIAGRAMA**

Projektuojamos sistemos duomenų srautų diagrama padeda geriau išsiaiškinti sistemoje kompiuterizuojamas funkcijas, jų eiliškumą. Duomenų srautų diagrama padeda nustatyti informacijos srautus – kas kam teikia informaciją ir duomenis, kur jie keliauja, kur kaupiami.

DSD vaizduoja funkcijas, aprašytas aukščiau pateiktame paveiksle funkcijų hierarchijos diagramoje. Ji taip pat apima duomenų bazės lenteles, kuriose saugomi duomenys, ir išorinius objektus kurie dalyvauja srautų formavime.

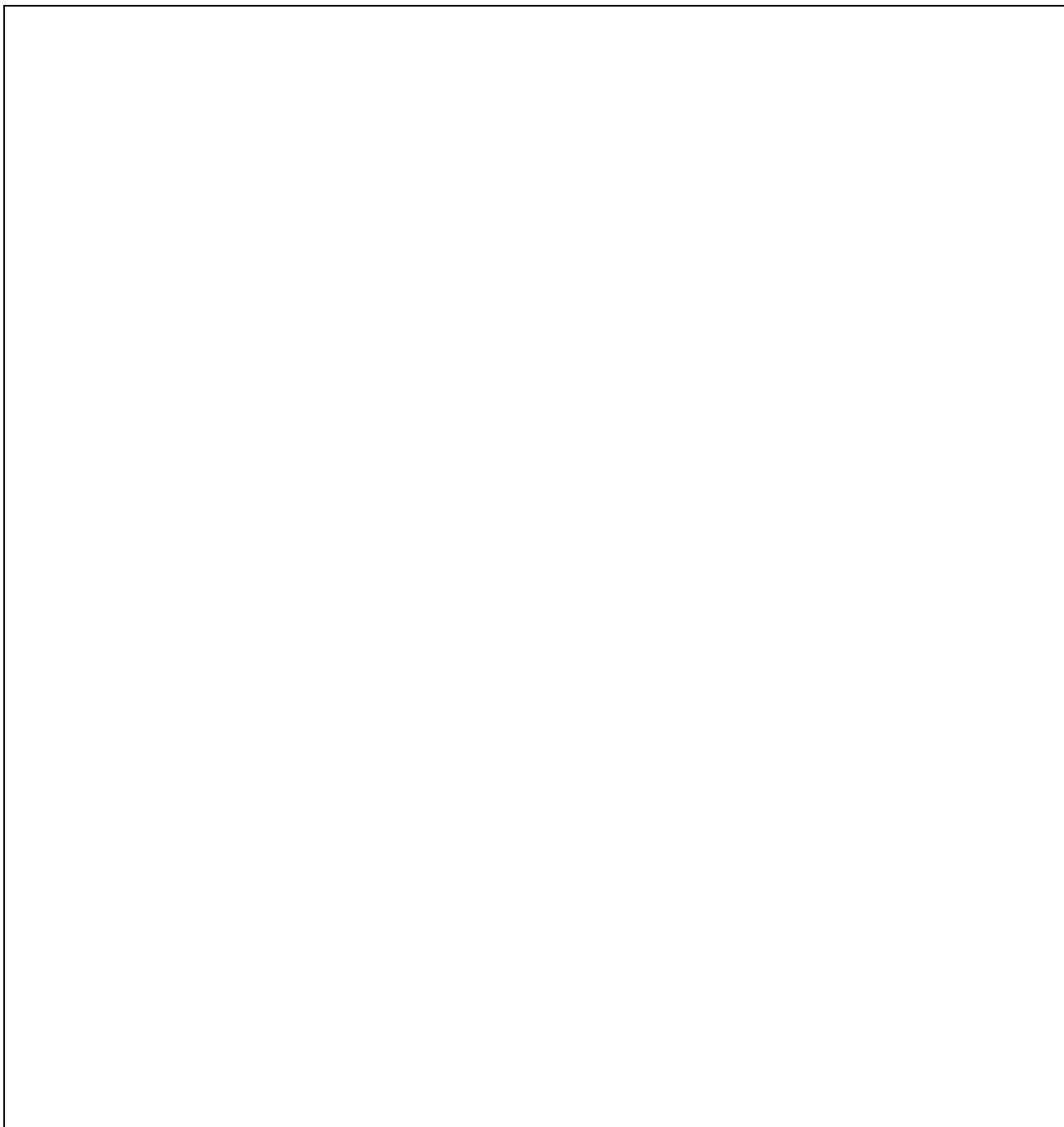


Šaltinis: sukurta autoriaus

**16 pav. Duomenų srautų diagrama**

#### **4.5 SISTEMOS DINAMIKOS APRAŠYMAS (DIALOGO SCENARIJUS)**

Paveiksle matome kuriamo kontaktų centro dialogo scenarijų. Skambinantysis visų pirma susijungia su sistema, išklauso pranešimą apie galimas paieškas, ir viena iš jų pasirenka. Vėliau pereina prie paieškos kriterijų, vėliau juos įvardina atskirai po vieną, ir galiausiai gauna rezultatus apie paiešką.



Šaltinis: sukurta autoriaus

**17 pav. Dialogo scenarijus**



#### 4.6 KLASIFIKAVIMO IR KODAVIMO SISTEMOS APRAŠYMAS

Kadangi informacijos unikalumas yra vienas pagrindinių reliacinių duomenų bazių kūrimo punktų, informacijos klasifikavimas ir kodavimas kuriant IS yra neišvengiamas. Tai yra atliekama norint užtikrinti kad informacija nesidubliuos kuriant projektą CallMagic2007.

Taip pat svarbu, kad kiekvienas modulis galėtų vienas su kitu komunikuoti jam žinomu būdu, dėl to svarbiausia paieškos užklausų informacija koduojama sekančiai:

Visų pirma yra nustatoma kokią paieška bus vykdoma, ir įvedamas pasirinkimo kodas. Jie yra:

- [gv] – gyventojų paieška
- [im] – įmonių paieška

Vėliau pereinama pie paieškos kriterijų įvedimo, kur taip pat yra įvedama pasirinkimo kodas. Jie yra:

- Įmonių paieškai:
  - [o] – įmonės kodas
  - [p] – įmonės pavadinimas
  - [r] – veiklos rūšis
  - [m] - miestas
  - [t] - gatvė
  - [k] – direktoriaus vardas ir pavardė
- Gyventojų paieškai:
  - [d] – gyventojų ID
  - [m] - miestas
  - [t] - gatvė
  - [k] – gyventojų vardas ir pavardė

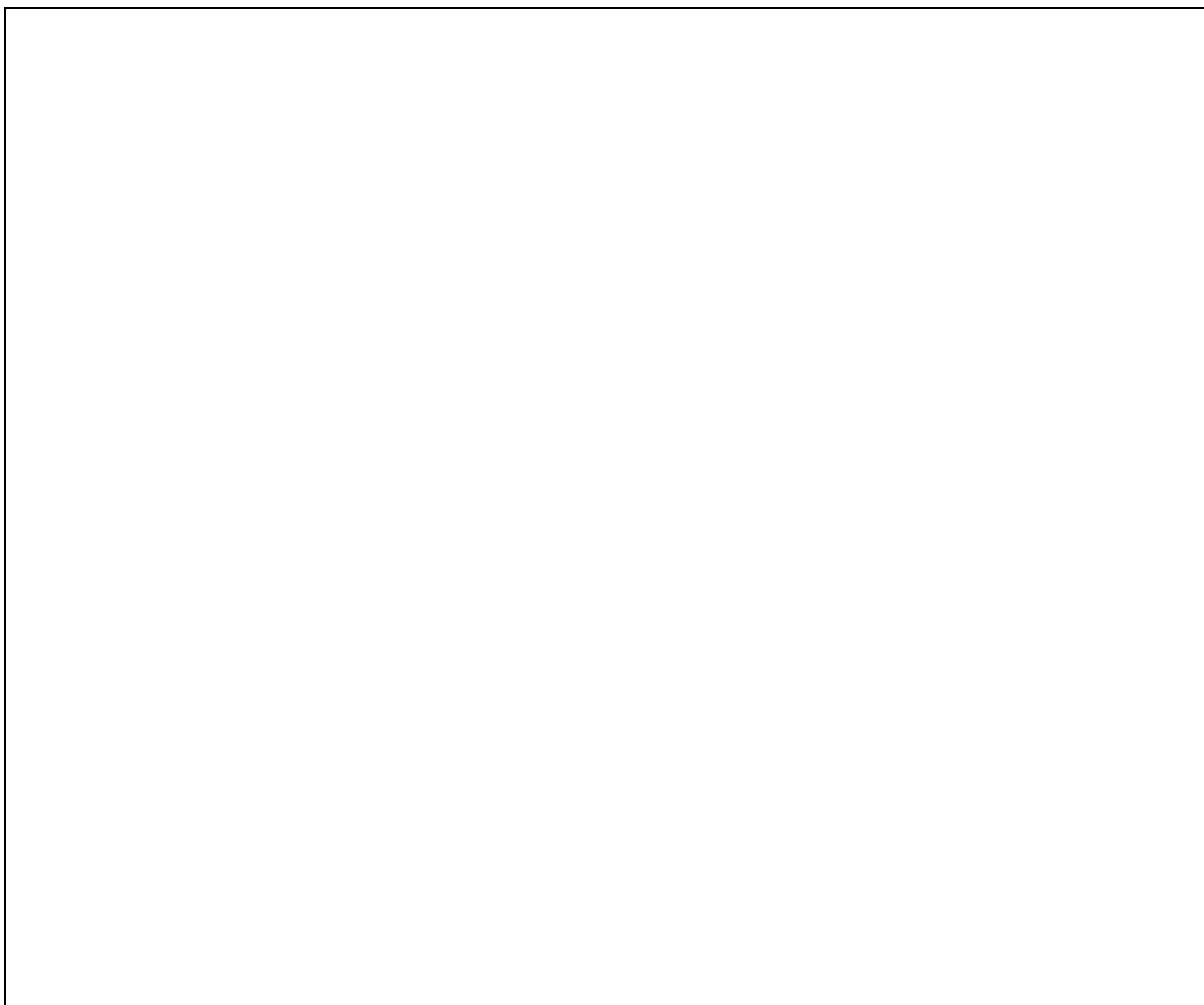
Kiekvienas iš minėtų kriterijų vėliau yra koduojamas unikaliais atitinkamo kriterijaus skaičiais, kurie taip pat yra perduodami ir į garsinės informacijos duomenų bazę.

Kuriant šį projektą duomenų kiekvieno paieškos kriterijaus duomenų bazėje buvo nuspręsta naudoti automatiškai generuojantis sveikąjį skaičių SQL duomenų tipas INT IDENTITY.

#### **4.7 DUOMENŲ BAZĖS PROJEKTAS**

Žemiau pateiktame paveiksle pavaizduota duomenų bazės loginė schema kuri suformuota MS Visio Enterprise Architect ir MS SQL Server 2005 Server Management Studio aplinkose.

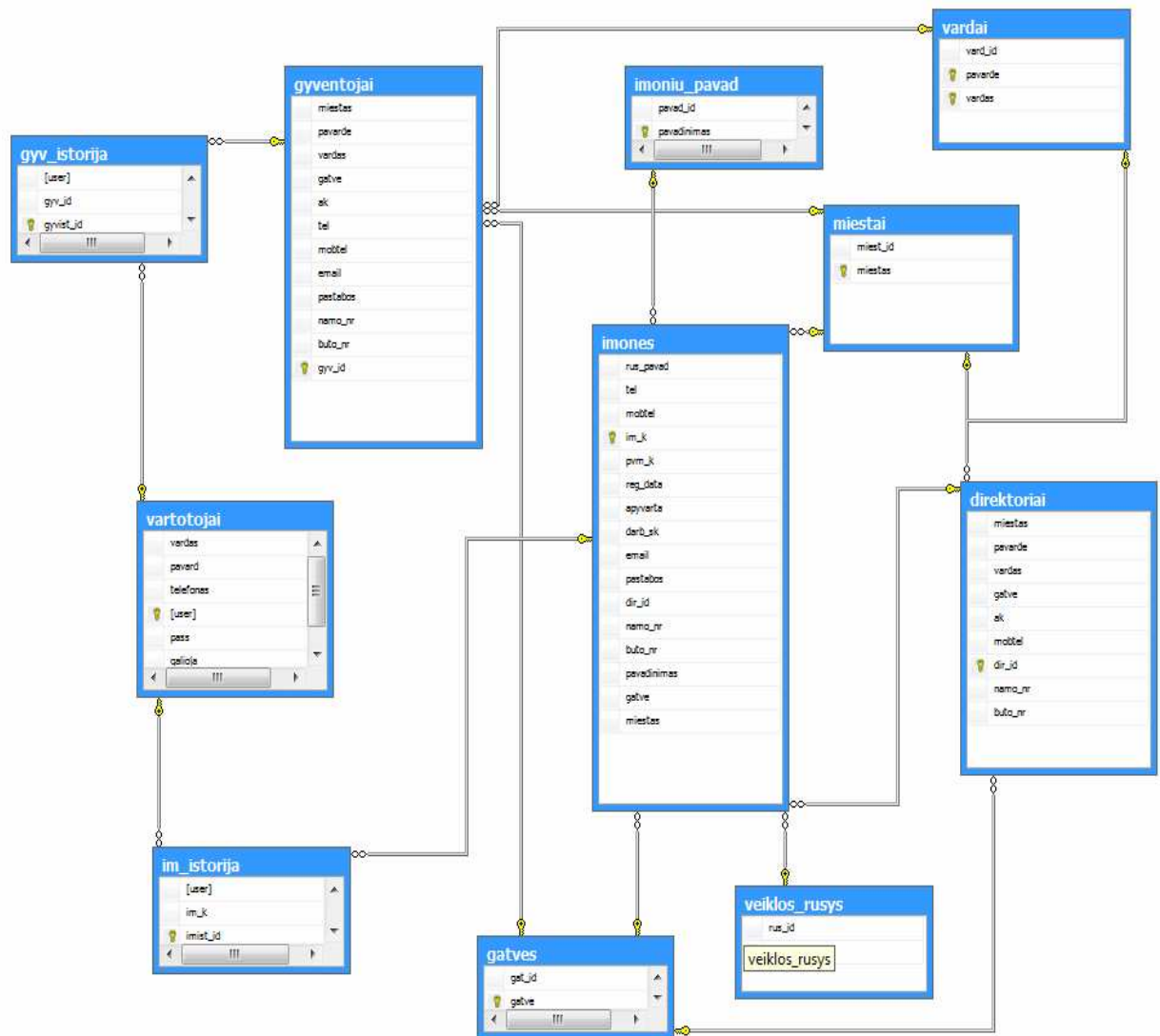
Sudaryta 11 lentelių sujungtų tarpusavio ryšiais.



Šaltinis: sukurta autoriaus

**18 pav. Duomenų bazės loginė schema MS Visio aplinkoje**

Suprojektuotos sistemos loginė schema MS SQL Server 2005 Server Management Studio aplinkoje matoma žemiau pateiktame paveiksle.



Šaltinis: sukurta autoriaus

19 pav. DB loginė schema

#### 4.8 INFORMACIJOS APDOROJIMO, PAIEŠKOS IR IŠRINKIMO TECHNOLOGIJOS APRAŠYMAS

Žemiau esamoje lentelėje pateikiama informacija apie kuriamos informacinės sistemos išrinkimo technologiją, ir atlikimo algoritmus.

6 lentelė

##### Informacijos išrinkimo technologijos aprašymas

Pavad.	Aprašymas	SQL tekstas
dirv	Išrinkimo procedūra direktoriaus informacijai pateikti	<pre>SELECT TOP (100) PERCENT dir_id, pavarde, vardas, ak, mobtel, gatve, namo_nr, buto_nr, miestas FROM dbo.direktoriai ORDER BY pavarde</pre>
Gyvist orija	Išrinkimo procedūra gyventojų istorijos pateikimui	<pre>SELECT TOP (100) PERCENT dbo.gyv_istorija.gyvist_id, dbo.gyventojai.pavarde, dbo.gyventojai.vardas, dbo.gyventojai.ak, dbo.gyventojai.miestas, dbo.gyventojai.gatve, dbo.gyventojai.namo_nr, dbo.gyventojai.buto_nr, dbo.gyventojai.tel, dbo.gyventojai.mobtel, dbo.gyventojai.email, dbo.gyventojai.pastabos, dbo.gyventojai.gyv_id, dbo.gyv_istorija.[user] FROM dbo.gatves INNER JOIN dbo.gyventojai ON dbo.gatves.gatve = dbo.gyventojai.gatve INNER JOIN dbo.miestai ON dbo.gyventojai.miestas = dbo.miestai.miestas INNER JOIN dbo.vardai ON dbo.gyventojai.pavarde = dbo.vardai.pavarde AND dbo.gyventojai.vardas = dbo.vardai.vardas INNER JOIN dbo.gyv_istorija ON dbo.gyventojai.gyv_id = dbo.gyv_istorija.gyv_id ORDER BY dbo.gyv_istorija.gyvist_id</pre>
Gyvv	Išrinkimo procedūra gyventojų informacijos pateikimui	<pre>SELECT TOP (100) PERCENT gyv_id, pavarde, vardas, ak, gatve, namo_nr, buto_nr, miestas, tel, mobtel, email, pastabos FROM dbo.gyventojai ORDER BY pavarde</pre>
I	Išrinkimo procedūra įmonės informacijos pateikimui	<pre>SELECT TOP (100) PERCENT dbo.imones.pavadinimas AS a, dbo.imones.im_k AS b, dbo.imones.pvm_k AS c, dbo.imones.gatve AS d, dbo.imones.namo_nr AS e, dbo.imones.buto_nr AS f, dbo.imones.miestas AS g, dbo.imones.tel AS h, dbo.imones.mobtel AS i, dbo.imones.email AS j, dbo.imones.reg_data AS k, dbo.imones.apyvarta AS l, dbo.imones.darb_sk AS m, dbo.imones.pastabos AS n, dbo.imones.rus_pavad AS o, dbo.direktoriai.vardas AS p, dbo.direktoriai.pavarde AS r, dbo.direktoriai.mobtel AS s FROM dbo.direktoriai INNER JOIN dbo.imones ON dbo.direktoriai.dir_id = dbo.imones.dir_id INNER JOIN dbo.imoniu_pavad ON dbo.imones.pavadinimas = dbo.imoniu_pavad.pavadinimas ORDER BY a</pre>
Imist	Išrinkimo procedūra įmonių informacijos pateikimui	<pre>SELECT TOP (100) PERCENT dbo.im_istorija.imist_id, dbo.imones.pavadinimas, dbo.imones.im_k, dbo.imones.pvm_k, dbo.imones.gatve, dbo.imones.namo_nr, dbo.imones.buto_nr, dbo.imones.miestas, dbo.imones.tel, dbo.imones.mobtel, dbo.imones.email, dbo.imones.reg_data, dbo.imones.apyvarta, dbo.imones.darb_sk, dbo.imones.pastabos, dbo.direktoriai.pavarde, dbo.direktoriai.vardas,</pre>

Pavad.	Aprašymas	SQL tekstas
		<pre> dbo.direktorai.mobtel AS dirmob, dbo.imones.rus_pavad, dbo.im_istorija.[user] FROM    dbo.direktorai INNER JOIN         dbo.imones ON dbo.direktorai.dir_id = dbo.imones.dir_id INNER JOIN         dbo.gatves ON dbo.imones.gatve = dbo.gatves.gatve INNER JOIN         dbo.imoniu_pavad ON dbo.imones.pavadinimas =         dbo.imoniu_pavad.pavadinimas INNER JOIN         dbo.miestai ON dbo.imones.miestas = dbo.miestai.miestas INNER JOIN         dbo.vardai ON dbo.direktorai.pavarde = dbo.vardai.pavarde AND         dbo.direktorai.vardas = dbo.vardai.vardas INNER JOIN         dbo.veiklos_rusys ON dbo.imones.rus_pavad =         dbo.veiklos_rusys.rus_pavad INNER JOIN         dbo.im_istorija ON dbo.imones.im_k = dbo.im_istorija.im_k ORDER BY dbo.im_istorija.imist_id </pre>
Gyvatranka_p	Išrinkimo procedūra su paieškos kriterijais. Skirta gyventojų išrinkimui su kintamu ir nepriklausomų nuo eilės tvarkos paieškos kriterijų skaičiumi.	<pre> set ANSI_NULLS ON set QUOTED_IDENTIFIER ON go ALTER PROCEDURE [dbo].[gyvatranka_p]     @gyvid int,     @miestid int,     @gatid int,     @vardid int,     @vart char(10) AS BEGIN drop table gyvatranka4; SELECT    dbo.gyventojai.pavarde, dbo.gyventojai.vardas, dbo.gyventojai.ak,         dbo.gyventojai.miestas, dbo.gyventojai.gatve, dbo.gyventojai.namo_nr,         dbo.gyventojai.buto_nr,         dbo.gyventojai.tel, dbo.gyventojai.mobtel, dbo.gyventojai.email,         dbo.gyventojai.pastabos, dbo.gyventojai.gyv_id, dbo.vardai.vard_id,         dbo.miestai.miest_id,         dbo.gatves.gat_id into gyvatranka FROM    dbo.gatves INNER JOIN         dbo.gyventojai ON dbo.gatves.gatve = dbo.gyventojai.gatve INNER JOIN         dbo.miestai ON dbo.gyventojai.miestas = dbo.miestai.miestas INNER JOIN         dbo.vardai ON dbo.gyventojai.pavarde = dbo.vardai.pavarde AND         dbo.gyventojai.vardas = dbo.vardai.vardas if @gyvid &lt;&gt; 0 SELECT * into gyvatranka1 from gyvatranka WHERE (gyvatranka.gyv_id = @gyvid) else SELECT * into gyvatranka1 from gyvatranka if @miestid &lt;&gt; 0 SELECT * into gyvatranka2 from gyvatranka1 WHERE (gyvatranka1.miest_id = @miestid) else SELECT * into gyvatranka2 from gyvatranka1 if @gatid &lt;&gt; 0 SELECT * into gyvatranka3 from gyvatranka2 WHERE (gyvatranka2.gat_id = @gatid) else SELECT * into gyvatranka3 from gyvatranka2 if @vardid &lt;&gt; 0 SELECT * into gyvatranka4 from gyvatranka3 WHERE (gyvatranka3.vard_id = @vardid) else </pre>

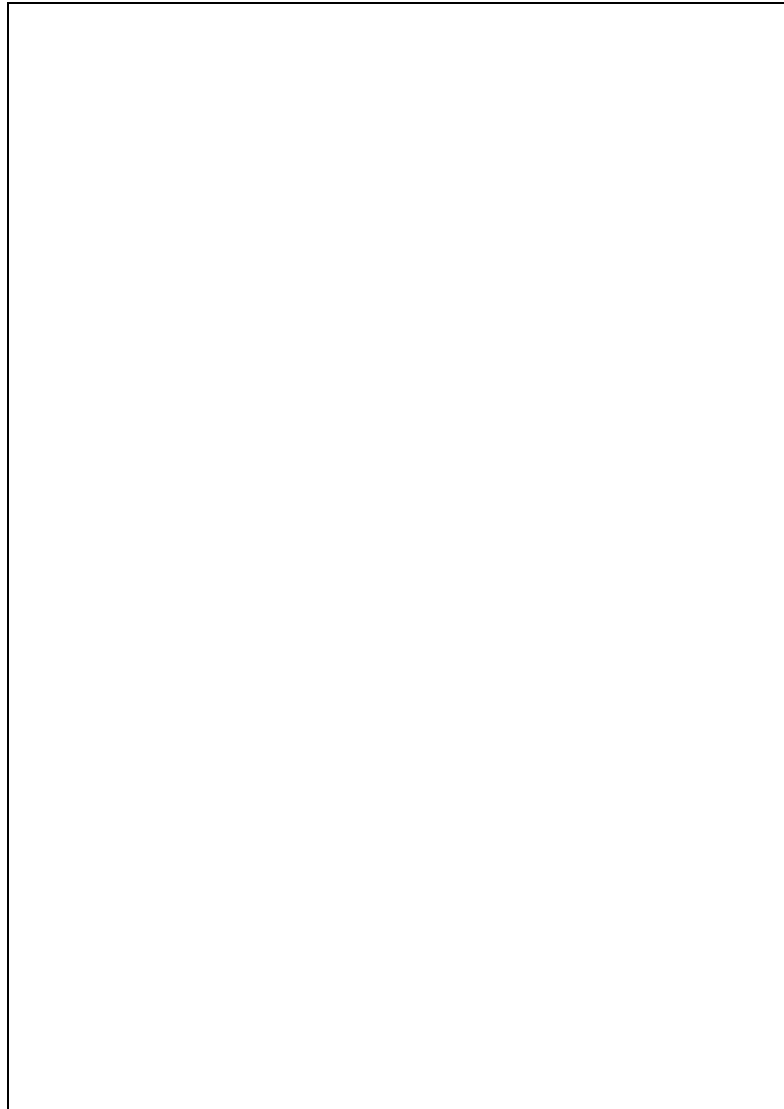
Pavadinimas	Aprašymas	SQL tekstas
		<pre> SELECT * into gvatranka4 from gvatranka3 drop table gvatranka; drop table gvatranka1; drop table gvatranka2; drop table gvatranka3; insert into gyv_istorija ([user], gyv_id) SELECT    dbo.vartotojai.[user], dbo.gvatranka4.gyv_id FROM      dbo.vartotojai CROSS JOIN           dbo.gvatranka4 WHERE     (dbo.vartotojai.[user] = @vart) END </pre>
Imatranka_p	Išrinkimo procedūra su paieškos kriterijais. Skirta įmonių išrinkimui su kintamu ir nepriklausomu nuo eilės tvarkos paieškos kriterijų skaičiumi.	<pre> set ANSI_NULLS ON set QUOTED_IDENTIFIER ON go ALTER PROCEDURE [dbo].[imatranka_p] @imk char(15), @rusid int, @miestid int, @gatid int, @pavamid int, @vardid int, @vart char(10) AS BEGIN drop table imatranka6; SELECT    dbo.imones.pavadinimas, dbo.imones.im_k, dbo.imones.pvm_k, dbo.imones.gatve, dbo.imones.namo_nr, dbo.imones.buto_nr, dbo.imones.miestas, dbo.imones.tel,           dbo.imones.mobtel, dbo.imones.email, dbo.imones.reg_data, dbo.imones.apyvarta, dbo.imones.darb_sk, dbo.imones.pastabos, dbo.direktorai.pavarde,           dbo.direktorai.vardas, dbo.direktorai.mobtel AS dirmob, dbo.imones.rus_pavad, dbo.veiklos_rusys.rus_id, dbo.miestai.miest_id, dbo.gatves.gat_id,           dbo.imoniu_pavad.pavad_id, dbo.vardai.vard_id into imatranka FROM      dbo.direktorai INNER JOIN           dbo.imones ON dbo.direktorai.dir_id = dbo.imones.dir_id INNER JOIN           dbo.gatves ON dbo.imones.gatve = dbo.gatves.gatve INNER JOIN           dbo.imoniu_pavad ON dbo.imones.pavadinimas = dbo.imoniu_pavad.pavadinimas INNER JOIN           dbo.miestai ON dbo.imones.miestas = dbo.miestai.miestas INNER JOIN           dbo.vardai ON dbo.direktorai.pavarde = dbo.vardai.pavarde AND dbo.direktorai.vardas = dbo.vardai.vardas INNER JOIN           dbo.veiklos_rusys ON dbo.imones.rus_pavad = dbo.veiklos_rusys.rus_pavad if @imk &lt;&gt; '' SELECT * into imatranka1 from imatranka WHERE (imatranka.im_k = @imk) else SELECT * into imatranka1 from imatranka if @rusid &lt;&gt; 0 SELECT * into imatranka2 from imatranka1 WHERE (imatranka1.rus_id = @rusid) else SELECT * into imatranka2 from imatranka1 if @miestid &lt;&gt; 0 SELECT * into imatranka3 from imatranka2 WHERE (imatranka2.miest_id = @miestid) </pre>

Pavard.	Aprašymas	SQL tekstas
		<pre> else SELECT * into imatranka3 from imatranka2 if @gaid &lt;&gt; 0 SELECT * into imatranka4 from imatranka3 WHERE (imatranka3.gat_id = @gaid) else SELECT * into imatranka4 from imatranka3 if @pavidid &lt;&gt; 0 SELECT * into imatranka5 from imatranka4 WHERE (imatranka4.pavid_id = @pavidid) else SELECT * into imatranka5 from imatranka4 if @vardid &lt;&gt; 0 SELECT * into imatranka6 from imatranka5 WHERE (imatranka5.vard_id = @vardid) else SELECT * into imatranka6 from imatranka5 drop table imatranka; drop table imatranka1; drop table imatranka2; drop table imatranka3; drop table imatranka4; drop table imatranka5; insert into im_istorija ([user], im_k) SELECT  dbo.vartotojai.[user], dbo.imatranka6.im_k FROM    dbo.vartotojai CROSS JOIN         dbo.imatranka6 WHERE   (dbo.vartotojai.[user] = @vart) END </pre>

Šaltinis: sukurta autoriaus

#### **4.9 SISTEMOS ARCHITEKTŪROS APRAŠYMAS (STRUKTŪRINĖ SCHEMA)**

Paveiksle pavaizduota kuriamo kontaktų centro struktūros schema:



Šaltinis: sukurta autoriaus

**20 pav. Sistemos struktūros diagrama**



#### 4.10 TESTAVIMO DUOMENŲ APRAŠYMAS

Programos išbaigimas neįmanomas be jos išbandymo su testuojamaisiais duomenimis, kadangi šio etapo metu išaiškėja visos sistemos klaidos ir trūkumai.

Kontaktų centrų prototipas buvo išbandomas su šiais pateiktais duomenimis.

7 lentelė

#### Testavimo duomenų pavyzdžiai

##### Veiklos\_rusys

Rus_ID	Rus_pavad
4	Automobiliai
5	Buitine technika
6	Foto prekes
1	Kompiuteriai
3	Reklama
2	Statybos

##### Miestai

Miest_id	miestas
7	Alytus
1	Kaunas
3	Klaipeda
9	Palanga
5	Panevezys
4	Siauliai
6	Taurage
8	Trakai
2	Vilnius

##### Gatves

Gat_id	Gatve
3	Dainavos
5	Kipro
8	Laisves al.
9	Laisves pr
6	Marvels
4	Partizanu
2	Savanoriu
7	Tiskeviciaus
1	Verkiu

##### Vartotojai

Vardas	Pavarde	Telefonas	User	Pass	galioja
Robertas	Butkevicius	867613223	admin	admin	2007.06.16 00:00:00
Andrius	Butkevicius	867613222	andrius	andrius	2007.06.16 00:00:00
Jonas	Jonaitis	857546556	jonas	jonas	2007.06.17 00:00:00
Petras	Petraitis	852345445	petras	petras	2007.06.17 00:00:00

##### Pavad\_id Pavadinimas

60	AB "Senukai"
49	UAB "Anis"
61	UAB "DMAX"

##### Imonių\_pavad

63	UAB "Kalis"
2	UAB "Lapes"
3	UAB "Lietuvalis"
1	UAB "Raivila"
62	UAB "Sonex"

##### Vardai

Vard_id	Pavarde	vardas
1	Butkevicius	Robertas
10	Eimantas	Nekrosius
9	Jonas	Biliunas
5	Juozas	Petraitis
3	Jurate	Miskinyte
7	Justas	Justiniskis
8	Kestas	Kubilius

imoniupav

2	Petras	Cvirka
4	Tadas	Blinda
6	Zana	Dark

### Imones

Rus_pavad	Tel	Mobtel	Im_k	Pvm_k	Darb_sk	Dir_id	Namo_nr	Buto_nr	Pavadinimas	Gatve	miestas
Kompiuteriai	865323465	865323465	126528564	LT42126528564	100	13	6	45	UAB "Lapes"	Verkiu	Kaunas
Kompiuteriai	8653526665	8653526665	135246954	LT42135246954	200	1	2	54	UAB "Lietuvalis"	Dainavos	Vilnius
Buitine technika	865236665	865236665	172411545	LT42172411545	300	1	52	6	UAB "Raivila"	Marveles	kaunas
Statybos	8653255665	8653255665	210032658	LT42210032658	52	14	63	435	UAB "Raivila"	Kipro	Vilnius
Buitine technika	8653325665	8653325665	236584236	LT42236584236	41	13	258	54	UAB "Raivila"	Partizanu	Klaipeda
Kompiuteriai	8652365256	8652365256	256472365	LT42256472365	62	1	123	6	UAB "Lietuvalis"	Verkiu	kaunas
Buitine technika	8653228465	8653228465	439514856	LT42439514856	235	13	5	6	UAB "Raivila"	verkiu	Kaunas
Foto prekes	456456456	456456456	521263258	LT23654586	20	14	4	0	UAB "Kalis"	Laisves pr	Klaipeda
Statybos	8653223652	8653223652	652951236	LT42652951236	425	1	2	345	UAB "Sonex"	Kipro	Vilnius

### Gyventojai

Miestas	Pavarde	Vardas	Gatve	AK	Tel	Mobtel	Email	Namo_nr	Buto_nr	Gyv_id
Kaunas	Butkevicius	Robertas	Verkiu	38112190714	867613223	867613223	robertas@delfi.lt	6	94	8
Klaipeda	Eimantas	Nekrosius	Partizanu	38112192562	86761322	86761322	eim@takas.lt	5	6	9
Kaunas	Jonas	Biliunas	Partizanu	38546100125	855546556	855546556	jonas@takas.lt	6	10	10
Kaunas	Juozas	Petraitis	Marveles	38512051448	865412345	865412345		4	65	11
Kaunas	Juozas	Petraitis	Savanoriu	35685120412	856412356	856412356		4a	20	12
Vilnius	Jurate	Miskinyte	Laisves al.	48106130252	865213456	865213456	jm@takas.lt	4H	45	13
Palanga	Justas	Justiniskis	Marveles	48522135689	864512345	864512345		4d	6	14
Vilnius	Kestas	Kubilius	Laisves pr	35212456999	854694654	854694654		6	2	15
Kaunas	Zana	Dark	Laisves al.	47912150205	837256489	837256489		6	25	16

### Direktoriai

Miestas	Pavarde	Vardas	Gatve	AK	Mobtel	Dir_id	Namo_nr	Buto_nr
Kaunas	Butkevicius	Robertas	Verkiu	38112514164	865623112	1	6	94
Vilnius	Petras	Cvirka	Verkiu	38112360714	865621212	9	6	56
Vilnius	Butkevicius	Robertas	Dainavos	38115236714	865623522	10	85	654
Kaunas	Juozas	Petraitis	Laisves al.	38112190714	865621852	13	6	0
Vilnius	Zana	Dark	Marveles	38112190714	865236112	14	6	12
Kaunas	Jonas	Biliunas	Laisves pr	356251402	65494654	15	6	7

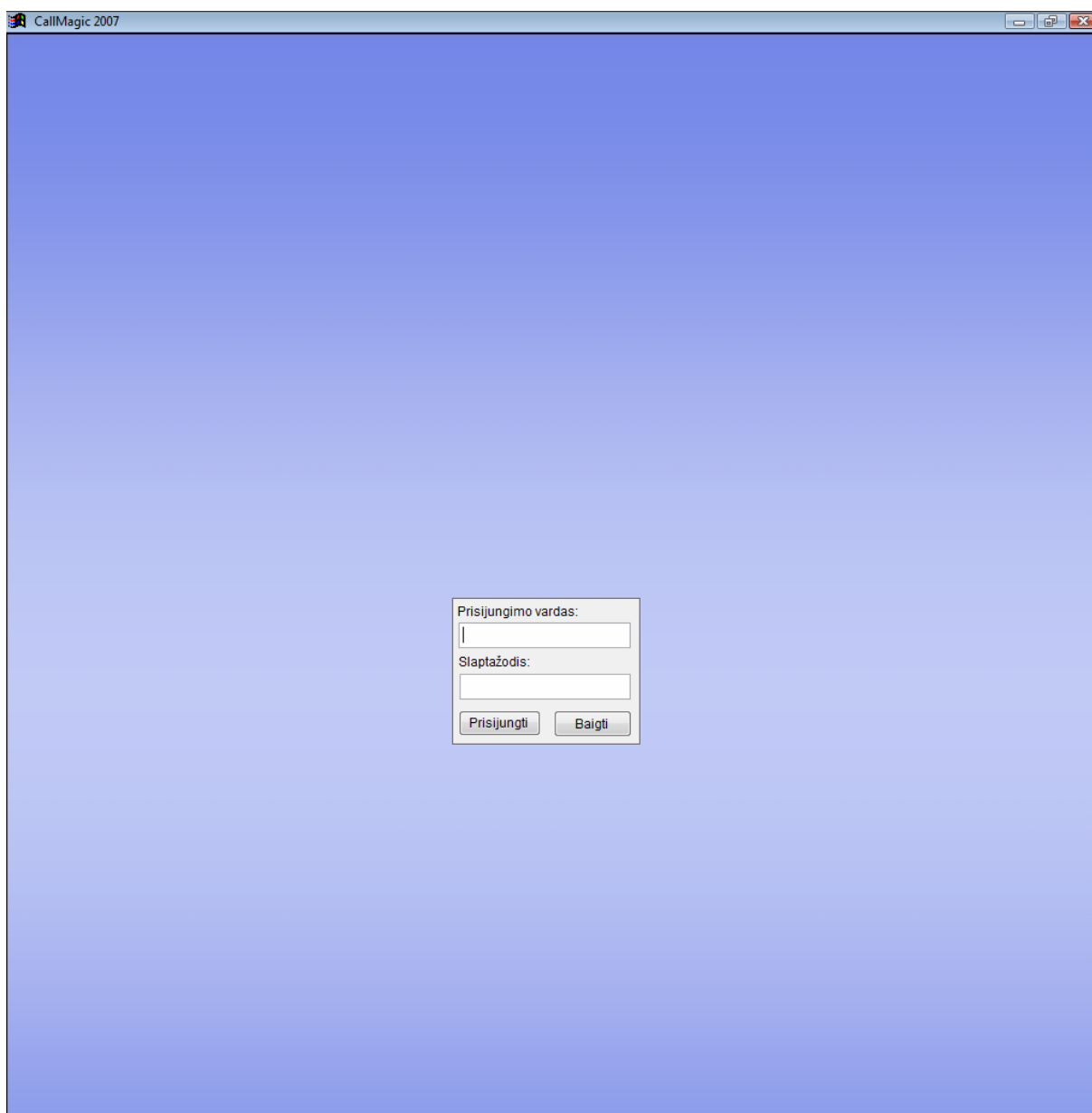
Šaltinis: sukurta autoriaus

#### 4.11 VARTOTOJO VADOVAS

Kontaktų centrų prototipo paskirtis yra administruoti į duomenų bazę patenkančius duomenis, o taip pat atvaizduoti paieškos rezultatus pagal užklausas, vykdomas įvedus sisteminius kodus, kad parodyti kaip kalbos atpažinimo modulis turės komunikuoti su kuriama duomenų baze.

Norint pradėti darbą reikia paleisti sistemos paleidžiamąjį failą callmagic.exe

Žemiau matome pagrindinį sistemos langą kuriame prašoma įvesti prisijungimo duomenis.



Šaltinis: sukurta autoriaus

**21 pav. Pagrindinis sistemos langas**

Įvedus teisingą prisijungimą sistema prisijungs prie duomenų bazės, ir leis pradėti dirbti.

Pasirinkus meniu punktą: ADMINISTRAVIMAS/Vartotojų administravimas matomas žemiau pateiktas langas, kuriame mes galime daryti šuos veiksmus:

- Įvesti naują vartotoją
- Keisti esamo vartotojo duomenis

Vardas	Pavardė	Tel.	User	Pass	Galiuja iki
Robertas	Butkevicius	867613223	admin	admin	2007.06.16 1
Andrius	Butkevicius	867613222	andrius	andrius	2007.06.16 1
Jonas	Jonaitis	857546556	jonas	jonas	2007.06.17 1
Petras	Petrakis	852345445	petras	petras	2007.06.17 1

Šaltinis: sukurta autoriaus

**22 pav. Vartotojų administravimas**

Pasirinkus meniu punktą: ADMINISTRAVIMAS/Įmonių administravimas matomas žemiau pateiktas langas, kuriame mes galime daryti šuos veiksmus:

- Įvesti naują įmonę
- Keisti esamos įmonės duomenis duomenis

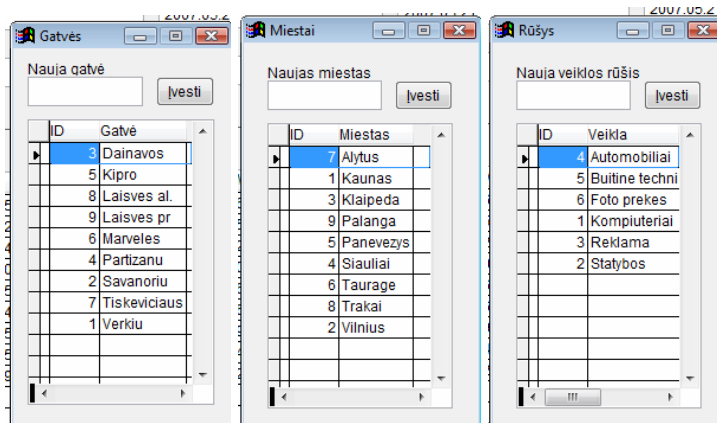
	Pavadinimas	Im.K.	PVM.K.	Gatvė	Namo Nr.	Buto Nr.	Miestas	Tel.	Mob tel.	E
▶	UAB "Lapes"	126528564	LT42126528564	Verkiu	6	45	Kaunas	865323465	865323465	
	UAB "Lietuvalis"	135246954	LT42135246954	Dainavos	2	54	Vilnius	8653526665	8653526665	
	UAB "Raivila"	172411545	LT42172411545	Marveles	52	6	kaunas	865236665	865236665	
	UAB "Raivila"	210032658	LT42210032658	Kipro	63	435	Vilnius	8653255665	8653255665	
	UAB "Raivila"	236584236	LT42236584236	Partizanų	258	54	Klaipėda	8653325665	8653325665	
	UAB "Lietuvalis"	256472365	LT42256472365	Verkiu	123	6	kaunas	8652365256	8652365256	
	UAB "Raivila"	439514856	LT42439514856	verkiu	5	6	Kaunas	8653228465	8653228465	
	UAB "Kalis"	521263258	LT23654586	Laisves pr	4	0	Klaipėda	456456456	456456456	
	UAB "Sonex"	652951236	LT42652951236	Kipro	2	345	Vilnius	8653223652	8653223652	

Šaltinis: sukurta autoriaus

### 23 pav. Įmonių administravimas

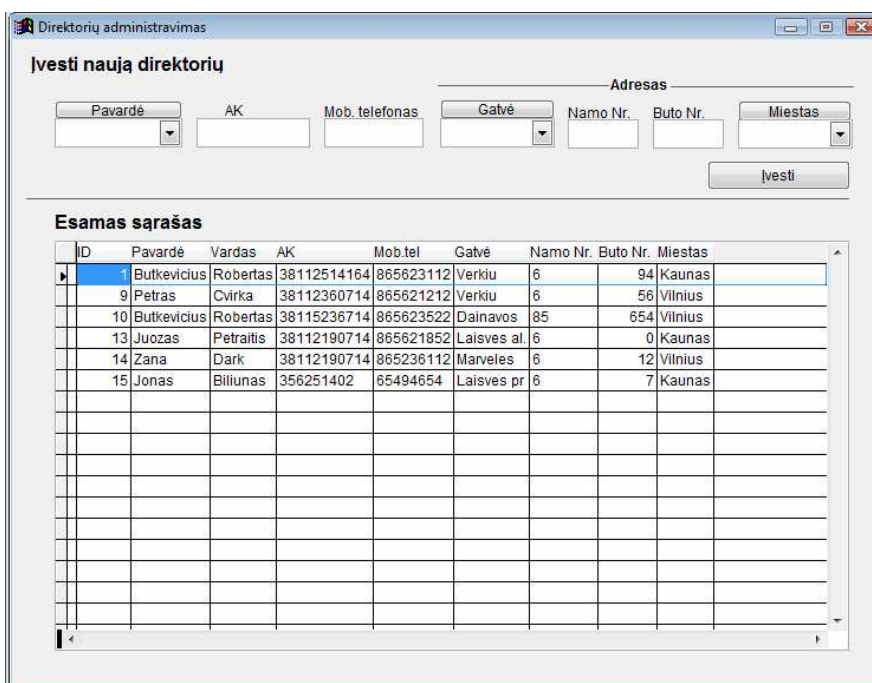
Gatvė, miestas, direktorius, veiklos rūšis pasirenkami iš iškrentančio meniu. Jeigu reikalingo varianto sąraše nėra, tuomet reikia paspausti ant atitinkamo užrašo, kur bus atidaromi langai, su naujos atitinkamos informacijos įvedimu, bei esamos informacijos koregavimu.

Šie langai pateikti žemiau:



Šaltinis: sukurta autoriaus

**24 pav. Gatvių, miestų ir veiklos rūšių administravimas**



Šaltinis: sukurta autoriaus

**25 pav. Įmonių direktorių administravimas**

Pasirinkus meniu punktą: ADMINISTRAVIMAS/Gyventojų administravimas matomas žemiau pateiktas langas, kuriame mes galime daryti šuos veiksmus:

- Įvesti naują gyventoją
- Keisti esamo gyventoją duomenis



Gyventojų administravimas

**Įvesti naują gyventoją**

Adresas

Pavardė  AK  Mob. telefonas  Gatvė  Namų Nr.  Buto Nr.  Miestas

Telefonas  EMAIL  Pastabos

**Esamas sąrašas**

ID	Pavardė	Vardas	AK	Gatvė	Namų Nr.	Buto Nr.	Miestas	Tel.	Mob tel.	El.
8	Butkevicius	Robertas	38112190714	Verkiu	6	94	Kaunas	867613223	867613223	ro
9	Eimantas	Nekrosius	38112192562	Partizanu	5	6	Klaipeda	86761322	86761322	ei
10	Jonas	Biliunas	38546100125	Partizanu	6	10	Kaunas	855546556	855546556	jo
11	Juozas	Petrailis	38512051448	Manvelės	4	65	Kaunas	865412345	865412345	
12	Juozas	Petrailis	35685120412	Savanoriu	4a	20	Kaunas	856412356	856412356	
13	Jurate	Miskinyte	48106130252	Laisvės al.	4H	45	Vilnius	865213456	865213456	jr
14	Justas	Justiniskis	48522135689	Manvelės	4d	6	Palanga	864512345	864512345	
15	Kestas	Kubilius	35212456999	Laisvės pr	6	2	Vilnius	854694654	854694654	
16	Zana	Dark	47912150205	Laisvės al.	5	25	Kaunas	837256489	837256489	

Šaltinis: sukurta autoriaus

## 26 pav. Gyventojų administravimas

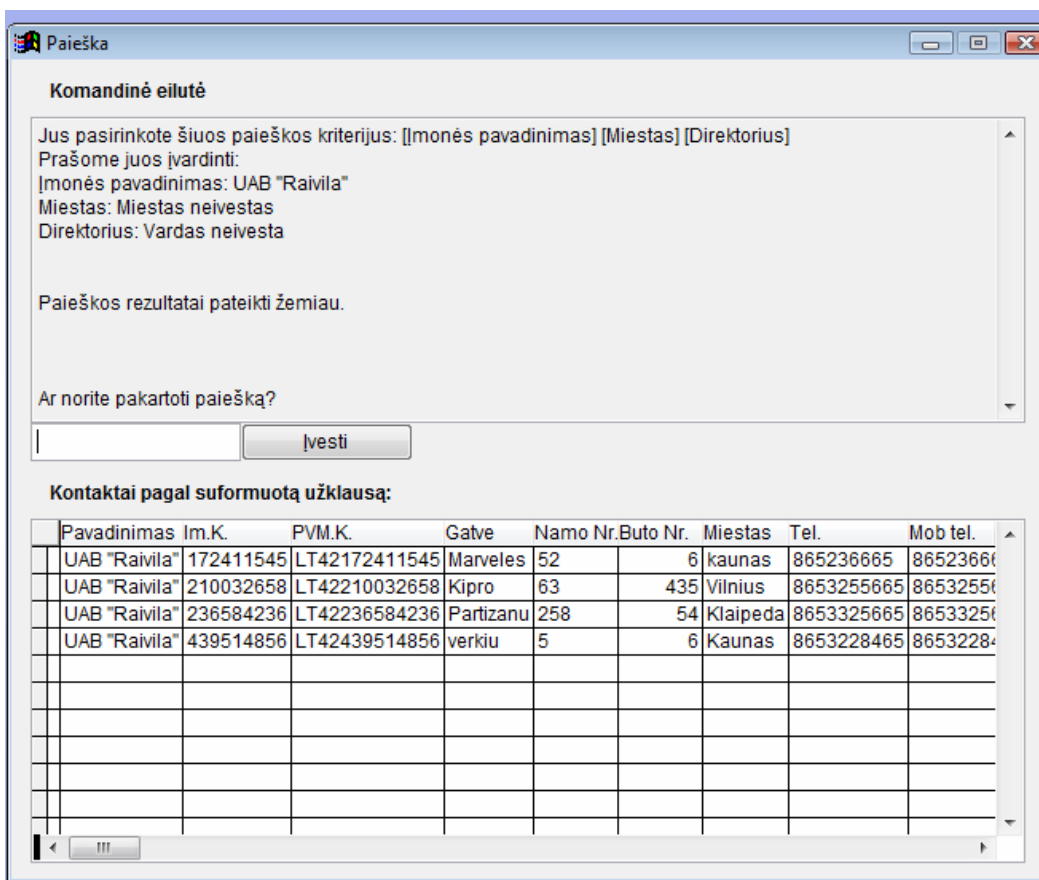
Pavardė, gatvė, miestas pasirenkami iš iškrentančio meniu. Jeigu reikalingo varianto sąrašė nėra, tuomet reikia paspausti ant atitinkamo užrašo, kur bus atidaromi langai, su naujos atitinkamos informacijos įvedimu, bei esamos informacijos koregavimu. Kurių pavyzdžiai jau buvo pateikti aukščiau.

Pasirinkus meniu punktą: Paieška, bus atidaromas demonstracinis kodinės informacijos paieškos langas, kuris pateiktas žemiau. Šis langas iš esmės yra tekstine konsolė, atspindinti kaip vyksta informacijos paieška kontaktų centre, ir kaip pateikiami rezultatai.

Šioje konsolėje vykdoma paieška pagal kodus, kurie išvardinti aukščiau esančiame skyriuje apie kodavimo paaiškinimą.

Norint išbandyti šios konsolės veikimą būtina žinoti ieškomos informacijos kodus. Juos galima pasižiūrėti aukščiau pateiktame skyriuje apie testavimo duomenis.

Žemiau pateiktas pavyzdys, kaip sistema ieško įmonės, kurios pavadinimo kodas yra: 1



Šaltinis: sukurta autoriaus

27 pav. Paieškos konsolė

Pasirinkus meniu punktą: Istorija, matome žemiau pateiktą langą, kuriame saugoma visa atitinkamo vartotojo ieškota kontaktinių duomenų istorija. Šiame lange vartotojas įmonių ir gyventojų kontaktinių duomenų paieškos istoriją mato atskirai, kuria taipgi atskirai gali ir išvalyti.

Vartotojo istorija

Vartotojo: admin, įmonių paieškos istorija

ID	Pavardė	Vardas	AK	Gatvė	Namo Nr.	Buto Nr.	Miestas	Tel.	Mob tel.
15	UAB "Raivila"	172411545	LT42172411545	Marveles	52	6	kaunas	865236665	865236665
16	UAB "Raivila"	210032658	LT42210032658	Kipro	63	435	Vilnius	8653255665	8653255665
17	UAB "Raivila"	236584236	LT42236584236	Partizanų	258	54	Klaipėda	8653325665	8653325665
18	UAB "Raivila"	439514856	LT42439514856	verkiu	5	6	Kaunas	8653228465	8653228465
19	UAB "Lapes"	126528564	LT42126528564	Verkiu	6	45	Kaunas	865323465	865323465
20	UAB "Lietuvalis"	135246954	LT42135246954	Dainavos	2	54	Vilnius	8653526665	8653526665
21	UAB "Raivila"	172411545	LT42172411545	Marveles	52	6	kaunas	865236665	865236665
22	UAB "Raivila"	210032658	LT42210032658	Kipro	63	435	Vilnius	8653255665	8653255665
23	UAB "Raivila"	236584236	LT42236584236	Partizanų	258	54	Klaipėda	8653325665	8653325665
24	UAB "Lietuvalis"	256472365	LT42256472365	Verkiu	123	6	kaunas	8652365256	8652365256
25	UAB "Raivila"	439514856	LT42439514856	verkiu	5	6	Kaunas	8653228465	8653228465

Išvalyti

Vartotojo: admin, gyventojų paieškos istorija

Header1	Header1	Header1	Header1	Header1	Header1	Header1	Header1	Header1	Header1	Header1
8	Butkevicius	Robertas	38112190714	Kaunas	Verkiu	6	94	867613223	867613223	rot
9	Jonas	Biliunas	38546100125	Kaunas	Partizanų	6	10	855546556	855546556	jon
10	Juozas	Petraitis	38512051448	Kaunas	Marveles	4	65	865412345	865412345	
11	Juozas	Petraitis	35685120412	Kaunas	Savanorių	4a	20	856412356	856412356	
12	Zana	Dark	47912150205	Kaunas	Laisves al.	6	25	837256489	837256489	
13	Butkevicius	Robertas	38112190714	Kaunas	Verkiu	6	94	867613223	867613223	rot
14	Jonas	Biliunas	38546100125	Kaunas	Partizanų	6	10	855546556	855546556	jon
15	Juozas	Petraitis	38512051448	Kaunas	Marveles	4	65	865412345	865412345	
16	Juozas	Petraitis	35685120412	Kaunas	Savanorių	4a	20	856412356	856412356	
17	Zana	Dark	47912150205	Kaunas	Laisves al.	6	25	837256489	837256489	

Išvalyti

Šaltinis: sukurta autoriaus

## 28 pav. Vartotojų istorija

#### **4.12 PROGRAMUOTOJO VADOVAS**

Ši duomenų bazė yra realizuota naudojantis MS Visual FoxPro 9.0, serverio duomenų baze pasirenkant MS SQL Server 2005, todėl gali būti lengvai koreguojama kiekvieno kas turi įdiegęs atitinkamas programas.

Pagrindinės panaudotos funkcijos:

- Select
- Insert
- Update
- Delete

Duomenų bazėje yra 11 lentelių, 11 formų, 5 poschemės (view), viena pagrindinė programa ir 1 meniu.

Apsaugos: visose formose yra realizuota duomenų kontrolė nuo rakto dubliavimosi, taip pat nuo negalimų arba skaitmeninių reikšmių įvedimo, datos laukai kontroliuojami kad nebūtų galima įvesti vėlesnės nei sisteminės datos. Naudojama nemažai combo box įvedimo laukų tai padeda kontroliuoti įvedamus duomenis.

## IŠVADOS IR PASIŪLYMAI

- Kontaktų centrų plėtra yra neatsiejama nuo naujausių technologijų vystymosi. kontaktų centrų veikla yra daugialypė, pasireiškianti paslaugų teikimu Internetu, IP telefonijos pritaikymu paslaugų teikimo metu, paieškos Internete galimybėmis, asinchroninių ir video konferencijų rengimu ir kt.

- Kaip rodo kontaktų centrų praktika yra daug kontaktų centrų pritaikymo sričių, tačiau jų visų organizacinė struktūra panaši. Pirmaujanti pasaulyje pagal kontaktų centrų steigimą yra Jungtinės Amerikos Valstijos. Kontaktų centrų išsivystymą šioje šalyje byloja šiame sektoriuje dirbančių darbuotojų skaičius.

- UAB “Lintel” organizacinė struktūra yra orientuota į pasaulyje plintančias kontaktų centrų organizavimo tendencijas. Komandinio darbo akcentavimas, lankstumas, efektyvi komunikacija tarp aukščiausio lygio vadovų ir žemesnio rango darbuotojų rodo, kad UAB “Lintel” siekia kurti šiuolaikinio modernaus kontaktų centro struktūros tipą. Kita vertus, UAB “Lintel” kai kuriais aspektais reikėtų tobulinti savo organizacinę struktūrą, akcentuojant įvairiapusiškumo, tinkliškumo ir globališkumo aspektus, kuriuos galima būtų realizuoti kooperuojantis su partneriais užsienyje.

- UAB “Lintel” siekia diegti naujausias kontaktų centrų technologines naujoves, tačiau kol kas šis naujų technologijų perėmimo lygis nėra pakankamai aukštas. Įmonei derėtų didesnę dėmesį skirti programų, padedančių automatizuoti įmonės veiklą kūrimui, Interneto galimybių išplėtimu teikiant paslaugas, virtualių operatorių projektavimu.

- Pasaulyje žinomos kontaktų centrų programinę įrangą gaminančios kompanijos siekia orientotis į kontaktų centrų automatizavimą didinančių programų kūrimą, virtualių operatorių projektavimą, centrų darbo kokybės parametrų vertinimą ir kitus centro veiklos efektyvumą didinančius ir kaštus mažinančius procesus.

- “Lietuvos telekomo” naudojama tinklo TSQL nėra pakankamai išvystyta ir turi būti tobulintina, atliekant patobulinimus visose tinklo moduluose: prieigos tinkle, bevieliam tinkle ir komutacijos tinkle.

- Pasaulyje yra keletas kalbos atpažinimo taikymo sričių. Automatizuotas kontaktų centras – viena jų. Jo tikslas supaprastinti informacijos pateikimą vartotojams, sumažinant aptarnavimo kaštus.

- Be kalbos atpažinimo įdiegimo vartotojui darosi labai sunku suvaldyti sudėtingus hierarchinius meniu, todėl šis meniu valdymo principas labai veiksmingas automatiniuose kontaktų centruose.

- Yra keletas kalbos atpažinimo modelių, tačiau šiuo metu labiausiai paplitęs ir naudojamas yra paslėptas Markovo modelis kurio populiarumą įtakoja tai, kad šias sistemas galima automatiškai apmokyti, jos yra paprastos ir kompiuteriu lengvai valdomos.

- Susipažinta su kalbos atpažinimo priemonėmis, tokiomis kaip HTK, bei MS Speech Server veikimo principais, ir nustatyta, kad MS Speech Server turi gerą integraciją į duomenų bazes, tokias kaip SQL, bei turi didelius aprašymų resursus.

- Nagrinėjant kontaktų centrus technologiniu požiūriu buvo nustatę keletą trūkumų, dėl kurių sudėtingėja kontaktų centrų plėtra Lietuvoje. Šie trūkumai yra: Esamų rinkoje programinių produktų sudėtingumas; Esamų rinkoje programinių produktų brangumas; Vartotojo interfeiso sudėtingumas; Pritaikomumas Lietuvių kalbai.

- Atlikus kontaktų centrų organizacinės ir technologinės struktūros analizę ir įvertinus trūkumus, buvo nuspręsta pasiūlyti savo kontaktų centro technologinio sprendimo variantą, sumodeliuoti projektą sukuriant savitą kontaktų centrams programinę įrangą pasižyminčia galimybėmis, kurios būdingos dideliems kontaktų centrų programiniams paketams, tačiau išlaikant kiek įmanoma paprastesnę struktūrą, paliekant tik esminius dalykus. Projekto pavadinimas CallMagic 2007. Projektas apima šias pagrindines sritis: Vartotojų registravimą; Įmonių kontaktės informacijos administravimą ir kaupimą; Gyventojų kontaktės informacijos administravimą ir kaupimą; Balsu atpažinimą ir interpretavimą į kodinę sistemą; Kontaktų paiešką pagal balsu atpažinimo sistemos kodus; Vartotojų istorijos peržiūrą.

- Projekte veikia keturi pagrindiniai moduliai: Skambučių valdymo centras, Balsu atpažinimo ir interpretavimo sistema, Kontaktų duomenų bazė bei garsinės informacijos duomenų bazė.

- Šiame darbe bus projektuojamas automatinio kontaktų centro duomenų bazės sąsajos sprendimas pritaikytas kalbos atpažinimo priemonei MS Speech Server..

- Projekte vartotinių kalbos atpažinimo priemonių tikslumo tyrime gautos išvados:
  - Iš gautų rezultatų atliekant tiek HTK tiek MS Speech Server kalbos atpažinimo tikslumo tyrimą galime daryti išvadą, jog pasirinktos komandos pilnai gali būti naudojamos mūsų projekte.

- Kontaktų centrų prototipo kūrime naudosime MS Speech Server programine įranga, dėl jos gebėjimo geriau komunikuoti su duomenų bazėmis ir dėl platesnių aprašymų ir pagalbos resursų.
- HTK programinė įranga buvo atmesta dėl to, jog ji yra daugiau pritaikyta mokomiesiems tikslams, nei realiam produktui išbaigti.
- Diktorių vyrų kalbą sistema atpažįsta nežymiai tiksliau nei moterų, tačiau skirtumas yra toks nedidelis, kad realios įtakos sistemos naudojimuisi jis neturės.
- Nors ir kiek yra pažengusios garso atpažinimo technologijos, tačiau iki šimtaprocentinio tikslumo ar bent artimo jam dar dar toli.
- Visos atpažinimo sistemos labiausiai išvystytos yra Anglų kalba, tad norint atpažinti žodžius išstartus lietuviškai, šiems žodžiams reikia parinkti teisingas transkripcijas.
- Vienas iš svarbiausių punktų kompiuterizuojamoje dalyje yra informacijos kodavimas. Tai yra tam, kad garsinės informacijos ir kontaktų informacijos duomenų bazės moduliai komunikuotų vienas su kitu.
  - Projektavimui pasirinkta ši programinė įranga: MS Visio Enterprise Architect, MS Visual FoxPro v9, MS SQL Server 2005.
  - Šiuo metu realizuoti šie moduliai: Įmonių administravimas, Gyventojų administravimas, Paieška pagal užduotus kodus, Vartotojų administravimas, Vartotojo istorijos peržiūra ir valymas.

## LITERATŪROS SĄRAŠAS

1. Informacijos ir paslaugų centro samprata, [interaktyvus], Vilnius [žiūrėta 2005 10 30]. Prieiga per Internetą: <http://www.rtn.lt/rtn/0004/centras.html>
2. Štarolis S., Varkalys G. (2001) „Lietuvos telekomo“ tinklo SQL sistemos vystymo strategija. Vilnius. P 3 – 17. (Štarolis S., Varkalys G. 2001)
3. Kauno paslaugų teikimo ir duomenų bazių redagavimo skyrių susirinkimo, vykusio 2004 08 05 medžiaga. Kaunas.
4. Call centres establishment guidelines: the information of NSW department of Commerce office of information and communications technology. [interaktyvus]. Australia [žiūrėta 2005 11 01]. Prieiga per Internetą: <http://www.oit.nsw.gov.au/Guidelines/4.3.5.o-CC-Establishment.asp>
5. Kauno paslaugų teikimo skyriaus susirinkimo vykusio 2005 02 01 medžiaga. Kaunas.
6. Woudstra A., Huber, H., Michalczuk K. (2004) Call centers in distance education [interaktyvus]. [Žiūrėta 2005 11 14]. Prieiga per Internetą: [http://cde.athabasca.ca/online\\_book/ch12.html](http://cde.athabasca.ca/online_book/ch12.html)
7. Klenke M. (2004) Blueprint for Call center success: building an effective organizational structure and team environment [interaktyvus]. [Žiūrėta 2005 11 15]. Prieiga per Internetą: [www.theKontaktųcenterschool.com/articles/BlueprintforKontaktųCenterSuccess.pdf](http://www.theKontaktųcenterschool.com/articles/BlueprintforKontaktųCenterSuccess.pdf)
8. The features and benefits of Conversation Suite [interaktyvus]. [Žiūrėta 2005 10 31]. Prieiga per Internetą: [http://www.unveil.com/solutions/solutions\\_features.shtml](http://www.unveil.com/solutions/solutions_features.shtml)
9. Rosenberg A. (2004) IP Myths, Call centers realinies [interaktyvus]. [Žiūrėta 2005 11 02]. Prieiga per Internetą: [http://www.Kontaktųcentermaga\\_zine.com/showArticle.jhtml?articleID=163104835](http://www.Kontaktųcentermaga_zine.com/showArticle.jhtml?articleID=163104835)
10. The products and solutions of Pronexus Inc. Company [interaktyvus]. [Žiūrėta 2005 11 02]. Prieiga per Internetą: <http://www.pronexus.com/create/applications/Kontaktų-center.asp>
11. Call center software offered by Stratasoft Inc. Company [interaktyvus]. [Žiūrėta 2005 11 02]. Prieiga per Internetą: [http://www.strata\\_soft.com/products/virtual-Kontaktų-center.asp](http://www.strata_soft.com/products/virtual-Kontaktų-center.asp)



12. New Avaya Contact Center Solution Brings Intelligent Communications to Outbound Customer Service [interaktyvus]. [Žiūrėta 2005 11 02]. Prieiga per Internetą: <http://www.avaya.com/gcm/master-usa/en-us/corporate/pressroom/pressreleases/2005/pr-050926.htm>
13. Speech recognition. From Wikipedia, the free encyclopedia [interaktyvus]. Žiūrėta [2007 05.19]. Prieiga per Internetą: [http://en.wikipedia.org/wiki/Speech\\_recognition](http://en.wikipedia.org/wiki/Speech_recognition)
14. Automatic Speech Recognition [interaktyvus]. [Žiūrėta 2007 05 19]. Prieiga per Internetą: <http://www.cs.colorado.edu/~martin/SLP/Updates/9.pdf>
15. Pokalbis su mašina. [interaktyvus]. Kaunas [Žiūrėta 2007 05 19]. Prieiga per Internetą: <http://rtn.elektronika.lt/rtn/0204/pokalbis.html/>
16. Hidden Markov Model. From Wikipedia, the free encyclopedia. [interaktyvus]. [Žiūrėta 2007 05 19]. Prieiga per Internetą: [http://en.wikipedia.org/wiki/Hidden\\_Markov\\_model](http://en.wikipedia.org/wiki/Hidden_Markov_model)
17. What is HTK? [interaktyvus]. Kalifornija [Žiūrėta 2007 05 19]. Prieiga per Internetą: <http://htk.eng.cam.ac.uk/>
18. Balso dialogų technologijos (bdt): kada prakalbinsime kompiuterį lietuviškai? [interaktyvus]. Vilnius [Žiūrėta 2007 05 19]. Prieiga per Internetą: <http://www3.lrs.lt/docs2/ERRBYOVC.PPT>
19. Largest Call Centers. [interaktyvus]. [Žiūrėta 2007 05 21]. Prieiga per Internetą: <http://phoenix.about.com/od/empllarge/a/callcenter2005.htm>

## **PRIEDAI**

1 priedas. Pavyzdinės paieškos konsolės įvedimo metodo algoritmas.....	<b>75</b>
2 priedas. Duomenų bazės lentelių struktūros ataskaita. ....	<b>84</b>

## 1 priedas. Pavyzdinės paieškos konsolės įvedimo metodo algoritmas.

```
x = thisform.text2.value

rysys=SQLCONNECT("call","", "")
*****

IF (xx = 'gv') AND (faze = 3) then

IF NOT (kd == '') THEN
gyvid = VAL(thisform.text2.value)
sqlxec(rysys, "select gyv_id from gyventojai where gyv_id = ?gyvid",
"gyvidcur")
IF USED('gyvidcur') then
if (gyvidcur.gyv_id <> 0 )then
thisform.edit1.value = thisform.edit1.value + STR(gyvidcur.gyv_id) +
CHR(13)
ELSE
thisform.edit1.value = thisform.edit1.value + 'ID neivestas' + CHR(13)
gyvid = 0
endif
ELSE
thisform.edit1.value = thisform.edit1.value + 'ID neivestas' + CHR(13)
gyvid = 0
ENDIF
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
kd = ''
faze = 2
else

IF NOT (km == '') THEN
miestid=VAL(thisform.text2.Value)
sqlxec(rysys, "select miestas, miest_id from miestai where
miestai.miest_id = ?miestid", "miestidcur")
IF USED('miestidcur') AND (miestidcur.miest_id <> 0 )
thisform.edit1.value = thisform.edit1.value + miestidcur.miestas + CHR(13)
ELSE
thisform.edit1.value = thisform.edit1.value + 'Miestas neivestas' + CHR(13)
miestid = 0
ENDIF
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
km = ''
faze = 2
else

IF NOT (kt == '') THEN
gatid=VAL(thisform.text2.Value)
sqlxec(rysys, "select gatve, gat_id from gatves where gatves.gat_id =
?gatid", "gatidcur")
IF USED('gatidcur') AND (gatidcur.gat_id <> 0 )
thisform.edit1.value = thisform.edit1.value + gatidcur.gatve + CHR(13)
ELSE
thisform.edit1.value = thisform.edit1.value + 'Gatvė neivesta' + CHR(13)
gatid= 0
ENDIF
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
```

```

kt = ''
faze = 2
else

IF NOT (kk == '') THEN
vardid=VAL(thisform.text2.Value)
sqlexec(rysys, "select pavarde, vardas, vard_id from vardai where
vardai.vard_id = ?vardid", "vardidcur")
IF USED('vardidcur') AND (vardidcur.vard_id <> 0 )
thisform.edit1.value = thisform.edit1.value + vardidcur.pavarde +
vardidcur.vardas + CHR(13)
ELSE
thisform.edit1.value = thisform.edit1.value + 'Pavardē ir vardas neivesti'
+ CHR(13)
vardid = 0
ENDIF
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
kk = ''
faze = 2
endif
endif
ENDIF
ENDIF

endif
*****
IF (xx = 'gv') AND (faze = 2) then
thisform.text2.Value = ''
faze = 3
IF NOT (kd == '') THEN
thisform.edit1.value = thisform.edit1.value + 'Gyventojo ID: '
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
else

IF NOT (km == '') THEN
thisform.edit1.value = thisform.edit1.value + 'Miestas: '
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
else

IF NOT (kt == '') THEN
thisform.edit1.value = thisform.edit1.value + 'Gatvė: '
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
else

IF NOT (kk == '') THEN
thisform.edit1.value = thisform.edit1.value + 'Pavarde ir vardas: '
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
ELSE
faze = 4
endif
endif
ENDIF
ENDIF

endif

```

```

*****
IF (xx = 'gv') AND (faze = 5) then
IF thisform.text2.Value = 'taip'
thisform.Destroy
thisform.Init
endif
endif

*****

IF (xx = 'gv') AND (faze = 4) then
sqlexec(rysys, "gvatranka_p '"+STR(gyvid)+"', '"+STR(miestid)+"',
'"+STR(gatid)+"', '"+STR(vardid)+"', '"+var+""")
sqlexec(rysys, "select * from gvatranka4", "rezultatai")
thisform.grid1.ColumnCount = 11
thisform.grid1.recordsource = "rezultatai"
thisform.grid1.AutoFit
Thisform.edit1.value = Thisform.edit1.value + CHR(13)+ CHR(13)+ 'Paieškos
rezultatai pateikti žemiau.'+ CHR(13)+ CHR(13)+ CHR(13)+ CHR(13)+ 'Ar
norite pakartoti paiešką? '
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
faze = 5
thisform.grid1.columncount = 11
thisform.grid1.column1.header1.caption = 'Pavardė'
thisform.grid1.column2.header1.caption = 'Vardas'
thisform.grid1.column3.header1.caption = 'AK'
thisform.grid1.column4.header1.caption = 'Miestas'
thisform.grid1.column5.header1.caption = 'Gatvė'
thisform.grid1.column6.header1.caption = 'Namo Nr.'
thisform.grid1.column7.header1.caption = 'Buto Nr.'
thisform.grid1.column8.header1.caption = 'Tel.'
thisform.grid1.column9.header1.caption = 'Mob tel.'
thisform.grid1.column10.header1.caption = 'EMAIL'
thisform.grid1.column11.header1.caption = 'Pastabos'
endif
*****
IF (xx = 'gv') AND (faze = 1) then
IF AT('d', x)>0 then
kd = ' [Gyventojo ID]'
else

IF AT('m', x)>0 then
km = ' [Miestas]'
endif
IF AT('t', x)>0 then
kt = ' [Gatvė]'
endif
IF AT('k', x)>0 then
kk = ' [Pavarde ir vardas]'
endif
endif

IF (kd == '') AND (km == '') AND (kt == '') AND (kk == '') then
Faze = 1
else
thisform.edit1.value = 'Jus pasirinkote šiuos paieškos kriterijus:'
+kd+km+kt+kk + chr(13);
+'Prašome juos įvardinti:' + chr(13)
thisform.edit1.SelStart = 1
thisform.edit1.refresh

```

```

IF NOT (kd == '') THEN
thisform.edit1.value = thisform.edit1.value + 'Gyventojų ID: '
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
thisform.text2.Value = ''
else

IF NOT (km == '') THEN
thisform.edit1.value = thisform.edit1.value + 'Miestas: '
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
thisform.text2.Value = ''
else

IF NOT (kt == '') THEN
thisform.edit1.value = thisform.edit1.value + 'Gatvė: '
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
thisform.text2.Value = ''
else

IF NOT (kk == '') THEN
thisform.edit1.value = thisform.edit1.value + 'Pavarde ir vardas: '
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
thisform.text2.Value = ''
endif
endif
ENDIF
ENDIF
faze=3

endif

endif
*****
*****
*****
*****

IF (xx = 'im') AND (faze = 3) then

IF NOT (ko == '') THEN
imk = thisform.text2.value
sqlxec(rysys, "select im_k from imones where im_k = ?imk", "imkcur")
IF USED('imkcur') then
if (imkcur.im_k <> ' ')then
thisform.edit1.value = thisform.edit1.value + imkcur.im_k + CHR(13)
ELSE
thisform.edit1.value = thisform.edit1.value + 'Kodas neivestas' + CHR(13)
imk = ''
endif
ELSE
thisform.edit1.value = thisform.edit1.value + 'Kodas neivestas' + CHR(13)
imk = ''
ENDIF
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
ko = ''
faze = 2

```

```

else

IF NOT (kp == '') THEN
pavadid=VAL(thisform.text2.Value)
sqlxec(rysys, "select pavadinimas, pavad_id from imoniu_pavad where
imoniu_pavad.pavad_id = ?pavadid", "pavadidcur")
IF USED('pavadidcur') then
IF (pavadidcur.pavad_id <> 0 )
thisform.edit1.value = thisform.edit1.value + pavadidcur.pavadinimas +
CHR(13)
ELSE
thisform.edit1.value = thisform.edit1.value + 'Pavadinimas neivestas' +
CHR(13)
pavadid = 0
endif
ELSE
thisform.edit1.value = thisform.edit1.value + 'Pavadinimas neivestas' +
CHR(13)
pavadid = 0
ENDIF
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
kp = ''
faze = 2
else

IF NOT (km == '') THEN
miestid=VAL(thisform.text2.Value)
sqlxec(rysys, "select miestas, miest_id from miestai where
miestai.miest_id = ?miestid", "miestidcur")
IF USED('miestidcur') AND (miestidcur.miest_id <> 0 )
thisform.edit1.value = thisform.edit1.value + miestidcur.miestas + CHR(13)
ELSE
thisform.edit1.value = thisform.edit1.value + 'Miestas neivestas' + CHR(13)
miestid = 0
ENDIF
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
km = ''
faze = 2
else

IF NOT (kt == '') THEN
gatid=VAL(thisform.text2.Value)
sqlxec(rysys, "select gatve, gat_id from gatves where gatves.gat_id =
?gatid", "gatidcur")
IF USED('gatidcur') AND (gatidcur.gat_id <> 0 )
thisform.edit1.value = thisform.edit1.value + gatidcur.gatve + CHR(13)
ELSE
thisform.edit1.value = thisform.edit1.value + 'Gatvė neivesta' + CHR(13)
gatid= 0
ENDIF
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
kt = ''
faze = 2
else

IF NOT (kr == '') THEN
rusid=VAL(thisform.text2.Value)

```

```

sqlexec(rysys, "select rus_pavad, rus_id from veiklos_rusys where
veiklos_rusys.rus_id = ?rusid", "rusidcur")
IF USED('rusidcur') AND (rusidcur.rus_id <> 0 )
thisform.edit1.value = thisform.edit1.value + rusidcur.rus_pavad + CHR(13)
ELSE
thisform.edit1.value = thisform.edit1.value + 'Rūšis neivesta' + CHR(13)
rusid = 0
ENDIF
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
kr = ''
faze = 2
else

IF NOT (kk == '') THEN
vardid=VAL(thisform.text2.Value)
sqlexec(rysys, "select pavarde, vardas, vard_id from vardai where
vardai.vard_id = ?vardid", "vardidcur")
IF USED('vardidcur') AND (vardidcur.vard_id <> 0 )
thisform.edit1.value = thisform.edit1.value + vardidcur.pavarde +
vardidcur.vardas + CHR(13)
ELSE
thisform.edit1.value = thisform.edit1.value + 'Vardas neivesta' + CHR(13)
vardid = 0
ENDIF
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
kk = ''
faze = 2
endif
endif
ENDIF
ENDIF
ENDIF
ENDIF

endif
*****
IF (xx = 'im') AND (faze = 2) then
thisform.text2.Value = ''
faze = 3
IF NOT (ko == '') THEN
thisform.edit1.value = thisform.edit1.value + 'Įmonės kodas: '
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
else

IF NOT (kp == '') THEN
thisform.edit1.value = thisform.edit1.value + 'Įmonės pavadinimas: '
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
else

IF NOT (km == '') THEN
thisform.edit1.value = thisform.edit1.value + 'Miestas: '
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
else

IF NOT (kt == '') THEN
thisform.edit1.value = thisform.edit1.value + 'Gatvė: '

```



```

thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
else

IF NOT (kr == '') THEN
thisform.edit1.value = thisform.edit1.value + 'Veiklos rūšis: '
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
else

IF NOT (kk == '') THEN
thisform.edit1.value = thisform.edit1.value + 'Direktorius: '
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
ELSE
faze = 4
endif
endif
ENDIF
ENDIF
ENDIF
ENDIF

endif
*****
IF (xx = 'im') AND (faze = 5) then
IF thisform.text2.Value = 'taip'
thisform.Destroy
thisform.Init
endif
endif

*****

IF (xx = 'im') AND (faze = 4) then
sqlxec(rysys, "imatranka_p '"+imk+"', '"+STR(rusid)+"',
'"+STR(miestid)+"', '"+STR(gatid)+"',;
'"+STR(pavadid)+"', '"+STR(vardid)+"', '"+vart+"'" )
sqlxec(rysys, "select * from imatranka6", "rezultatai")
thisform.grid1.ColumnCount = 18
thisform.grid1.recordsource = "rezultatai"
thisform.grid1.AutoFit
Thisform.edit1.value = Thisform.edit1.value + CHR(13)+ CHR(13)+ 'Paieškos
rezultatai pateikti žemiau.'+ CHR(13)+ CHR(13)+ CHR(13)+ CHR(13)+ 'Ar
norite pakartoti paiešką? '
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
faze = 5
thisform.grid1.columnCount = 18
thisform.grid1.column1.header1.caption = 'Pavadinimas'
thisform.grid1.column2.header1.caption = 'Im.K.'
thisform.grid1.column3.header1.caption = 'PVM.K.'
thisform.grid1.column4.header1.caption = 'Gatve'
thisform.grid1.column5.header1.caption = 'Namo Nr.'
thisform.grid1.column6.header1.caption = 'Buto Nr.'
thisform.grid1.column7.header1.caption = 'Miestas'
thisform.grid1.column8.header1.caption = 'Tel.'
thisform.grid1.column9.header1.caption = 'Mob tel.'
thisform.grid1.column10.header1.caption = 'EMAIL'
thisform.grid1.column11.header1.caption = 'Iregistruota'
thisform.grid1.column12.header1.caption = 'Apyvarta'

```

```

thisform.grid1.column13.header1.caption = 'Darb. sk.'
thisform.grid1.column14.header1.caption = 'Pastabos'
thisform.grid1.column15.header1.caption = 'Dir.Vardas'
thisform.grid1.column16.header1.caption = 'Dir.Pavarde'
thisform.grid1.column17.header1.caption = 'Dir.Mob.Tel'
thisform.grid1.column18.header1.caption = 'Veikla'
endif
*****
IF (xx = 'im') AND (faze = 1) then
IF AT('o', x)>0 then
    ko = ' [Įmonės kodas]'
else

IF AT('p', x)>0 then
    kp = ' [Įmonės pavadinimas]'
endif
IF AT('m', x)>0 then
    km = ' [Miestas]'
endif
IF AT('t', x)>0 then
    kt = ' [Gatvė]'
endif
IF AT('r', x)>0 then
    kr = ' [Veiklos rūšis]'
endif
IF AT('k', x)>0 then
    kk = ' [Direktorius]'
endif
endif

IF (ko == '') AND (kp == '') AND (km == '') AND (kt == '') AND (kr == '')
AND (kk == '') then
Faze = 1
else
thisform.edit1.value = 'Jus pasirinkote šiuos paieškos kriterijus:'
+ko+kp+km+kt+kr+kk + chr(13);
+'Prašome juos įvardinti:' + chr(13)
thisform.edit1.SelStart = 1
thisform.edit1.refresh
IF NOT (ko == '') THEN
thisform.edit1.value = thisform.edit1.value + 'Įmonės kodas: '
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
thisform.text2.Value = ''
else

IF NOT (kp == '') THEN
thisform.edit1.value = thisform.edit1.value + 'Įmonės pavadinimas: '
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
thisform.text2.Value = ''
else

IF NOT (km == '') THEN
thisform.edit1.value = thisform.edit1.value + 'Miestas: '
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
thisform.text2.Value = ''
else

IF NOT (kt == '') THEN

```

```

thisform.edit1.value = thisform.edit1.value + 'Gatvė: '
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
thisform.text2.Value = ''
else

IF NOT (kr == '') THEN
thisform.edit1.value = thisform.edit1.value + 'Veiklos rūšis: '
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
thisform.text2.Value = ''
else

IF NOT (kk == '') THEN
thisform.edit1.value = thisform.edit1.value + 'Direktorius: '
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
thisform.text2.Value = ''
endif
endif
ENDIF
ENDIF
ENDIF
ENDIF
faze=3
endif
endif
endif
IF faze = 0 then
IF x = 'im' THEN
thisform.edit1.value = 'Jus pasirinkote ieškoti įmonių informacijos.' +
CHR(13) + ;
'Pasirinkite pagal kokius paieškos kriterijus vykdysite paiešką.'+ CHR(13)
+'Galimi variantai:'+ ;
CHR(13) + CHR(13) + '[o] Įmonės kodas'+ CHR(13) + '[p] Įmonės
pavadinimas]' + CHR(13) + '[r] Veiklos rūšis' + CHR(13) + '[m] Miestas' +
CHR(13) + ;
'[t] Gatvė' + CHR(13) + '[k] Direktoriau pavardė ir vardas'
faze = 1
xx= 'im'
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
ENDIF
IF x = 'gv' THEN
thisform.edit1.value = 'Jus pasirinkote ieškoti gyventojų informacijos.' +
CHR(13) + ;
'Pasirinkite pagal kokius paieškos kriterijus vykdysite paiešką.'+ CHR(13)
+'Galimi variantai:'+ ;
CHR(13) + CHR(13) + '[d] Gyventojų ID'+ CHR(13) + '[m]Miestas' + CHR(13) +
'[t] Gatvė' + CHR(13) + '[k] Pavardė ir vardas'
faze = 1
xx='gv'
thisform.edit1.SelStart = LEN(thisform.edit1.VALUE)+1
thisform.edit1.refresh
endif
thisform.text2.value = ''
endif

*****

SQLDISCONNECT (rysys)

```

## 2 priedas. Duomenų bazės lentelių struktūros ataskaita.

# Table Report

## *Logine DB Schema.vsd*

### Database summary

Target DBMS: Microsoft SQL Server  
Number of tables: 14  
Number of views: 5  
Number of columns: 176  
Number of indexes: 1  
Number of foreign keys: 15  
Last build date: Not built

Extended attributes:  
Filegroup PRIMARY

Tables	Columns	Indexes	Foreign keys	Notes
veiklos_rusys	2	0	0	1
vardai	3	0	0	1
sysdiagrams	5	1	0	1
miestai	2	0	0	
imonių_pavad	2	0	0	
imones	16	0	5	
imatranka6	23	0	0	
im_istorija	3	0	2	
gvatranka4	15	0	0	
gyventojai	12	0	3	
gyv_istorija	3	0	2	
gatves	2	0	0	
direktoriai	9	0	3	

Views	Columns
dirv	9
gyvistorija	14
gyvv	12
I	18
imistorija	20

### direktoriai

Owner: dbo  
Target DB name: call  
Number of columns: 9  
Number of indexes: 0  
Number of foreign keys: 3  
Codes: 0  
Type: Table

Extended attributes:  
OnFileGroup PRIMARY  
Clustered PK Yes

Columns	Data type	Allow NULLs	Value/Range
<b>miestas (FK)</b>	char(10)	Not allowed	
<b>pavarde (FK)</b>	char(20)	Not allowed	
<b>vardas (FK)</b>	char(20)	Not allowed	
<b>gatve (FK)</b>	char(20)	Not allowed	
<b>ak</b>	char(11)	Not allowed	
<b>mobtel</b>	char(20)	Allowed	
<b>dir_id</b>	int identity	Not allowed	
<b>namo_nr</b>	char(10)	Allowed	
<b>buto_nr</b>	int	Allowed	

Foreign keys	Child	Parent
<b>gatves_direktoraii_FK1</b>	gatve	gatves.gatve
<b>miestai_direktoraii_FK1</b>	miestas	miestai.miestas
<b>vardai_direktoraii_FK1</b>	pavarde vardas	vardai.pavarde vardai.vardas
<b>FK_imones_direktoraii</b>	imones.dir_id	dir_id

Column details	
<b>1. miestas (FK)</b>	
Physical data type:	char(10)
Allow NULLs:	Not allowed
<b>2. pavarde (FK)</b>	
Physical data type:	char(20)
Allow NULLs:	Not allowed
<b>3. vardas (FK)</b>	
Physical data type:	char(20)
Allow NULLs:	Not allowed
<b>4. gatve (FK)</b>	
Physical data type:	char(20)
Allow NULLs:	Not allowed
<b>5. ak</b>	
Physical data type:	char(11)
Allow NULLs:	Not allowed
<b>6. mobtel</b>	
Physical data type:	char(20)
Allow NULLs:	Allowed
<b>7. dir_id</b>	
Physical data type:	int identity
Allow NULLs:	Not allowed
<b>8. namo_nr</b>	
Physical data type:	char(10)
Allow NULLs:	Allowed
<b>9. buto_nr</b>	
Physical data type:	int
Allow NULLs:	Allowed

Foreign key details (child)		
<b><u>gatves_direktoraii_FK1</u></b>		
<b>Definition:</b>	<b>Child</b> gatve	<b>Parent</b> gatves.gatve
<b>Relationship type:</b>	Non-Identifying	
<b>Cardinality:</b>	One -to- Zero-or-More	
<b>Allow NULLs:</b>	Not allowed	
<b>Verb phrase:</b>	hasgatves_direktoraii_FK1	

Inverse phrase: is of  
 Ref. Integrity on update: Cascade  
 Ref. Integrity on delete: No action

**miestai direktoriai FK1**

<b>Definition:</b>	<b>Child</b> miestas	<b>Parent</b> miestai.miestas
<b>Relationship type:</b>	Non-Identifying	
<b>Cardinality:</b>	One -to- Zero-or-More	
<b>Allow NULLs:</b>	Not allowed	
<b>Verb phrase:</b>	hasmiestai_direktoriai_FK1	
<b>Inverse phrase:</b>	is of	
<b>Ref. Integrity on update:</b>	Cascade	
<b>Ref. Integrity on delete:</b>	No action	

**vardai direktoriai FK1**

<b>Definition:</b>	<b>Child</b> pavarde vardas	<b>Parent</b> vardai.pavarde vardai.vardas
<b>Relationship type:</b>	Non-Identifying	
<b>Cardinality:</b>	One -to- Zero-or-More	
<b>Allow NULLs:</b>	Not allowed	
<b>Verb phrase:</b>	hasvardai_direktoriai_FK1	
<b>Inverse phrase:</b>	is of	
<b>Ref. Integrity on update:</b>	Cascade	
<b>Ref. Integrity on delete:</b>	No action	

**dirv**

Notes: 1  
 Owner: dbo  
 Target DB name: call  
 Number of columns: 9  
 Number of indexes: 0  
 Number of foreign keys: 0  
 Codes: 0  
 Type: View

Columns	Data type	Allow NULLs	Value/Range
dir_id	int identity	Not allowed	
pavarde	char(20)	Not allowed	
vardas	char(20)	Not allowed	
ak	char(11)	Not allowed	
mobtel	char(20)	Allowed	
gatve	char(20)	Not allowed	
namo_nr	char(10)	Allowed	
buto_nr	int	Allowed	
miestas	char(10)	Not allowed	

**Column details**

**1. dir\_id**  
 Physical data type: int identity  
 Allow NULLs: Not allowed

**2. pavarde**  
 Physical data type: char(20)  
 Allow NULLs: Not allowed

**3. vardas**

Physical data type: char(20)  
 Allow NULLs: Not allowed

**4. ak**

Physical data type: char(11)  
 Allow NULLs: Not allowed

**5. mobtel**

Physical data type: char(20)  
 Allow NULLs: Allowed

**6. gatve**

Physical data type: char(20)  
 Allow NULLs: Not allowed

**7. namo\_nr**

Physical data type: char(10)  
 Allow NULLs: Allowed

**8. buto\_nr**

Physical data type: int  
 Allow NULLs: Allowed

**9. miestas**

Physical data type: char(10)  
 Allow NULLs: Not allowed

**View definition**

```
CREATE VIEW dbo.dirv
AS
SELECT TOP (100) PERCENT dir_id, pavarde, vardas, ak, mobtel, gatve, namo_nr, buto_nr, miestas
FROM dbo.direktoriai
ORDER BY pavarde
```

**gatves**

Owner: dbo  
 Target DB name: call  
 Number of columns: 2  
 Number of indexes: 0  
 Number of foreign keys: 0  
 Codes: 0  
 Type: Table

**Extended attributes:**

OnFileGroup PRIMARY  
 Clustered PK Yes

Columns	Data type	Allow NULLs	Value/Range
gat_id	int identity	Not allowed	
gatve	char(20)	Not allowed	

Foreign keys	Child	Parent
gatves_direktoriai_FK1	direktoriai.gatve	gatve
gatves_gyventojai_FK1	gyventojai.gatve	gatve
gatves_imones_FK1	imones.gatve	gatve

**Column details**

**1. gat\_id**

Physical data type: int identity  
 Allow NULLs: Not allowed

## 2. gatve

Physical data type: char(20)  
Allow NULLs: Not allowed

## gyv\_istorija

Owner: dbo  
Target DB name: call  
Number of columns: 3  
Number of indexes: 0  
Number of foreign keys: 2  
Codes: 0  
Type: Table

Extended attributes:  
OnFileGroup PRIMARY  
Clustered PK Yes

Columns	Data type	Allow NULLs	Value/Range
user (FK)	char(10)	Not allowed	
gyv_id (FK)	int	Not allowed	
gvvist_id	int identity	Not allowed	

Foreign keys	Child	Parent
gyventojai_gyv_istorija_FK1	gyv_id	gyventojai.gyv_id
vartotojai_gyv_istorija_FK1	user	vartotojai.user

Column details
<b>1. user (FK)</b> Physical data type: char(10) Allow NULLs: Not allowed
<b>2. gyv_id (FK)</b> Physical data type: int Allow NULLs: Not allowed
<b>3. gvvist_id</b> Physical data type: int identity Allow NULLs: Not allowed

Foreign key details (child)		
<b><u>gyventojai_gyv_istorija_FK1</u></b>		
Definition:	<b>Child</b> gyv_id	<b>Parent</b> gyventojai.gyv_id
Relationship type:	Non-Identifying	
Cardinality:	One -to- Zero-or-More	
Allow NULLs:	Not allowed	
Verb phrase:	hasgyventojai_gyv_istorija_FK1	
Inverse phrase:	is of	
Ref. Integrity on update:	No action	
Ref. Integrity on delete:	No action	

<b><u>vartotojai_gyv_istorija_FK1</u></b>		
Definition:	<b>Child</b> user	<b>Parent</b> vartotojai.user
Relationship type:	Non-Identifying	
Cardinality:	One -to- Zero-or-More	



**Allow NULLs:** Not allowed  
**Verb phrase:** hasvartotojai\_gyv\_istorija\_FK1  
**Inverse phrase:** is of  
**Ref. Integrity on update:** Cascade  
**Ref. Integrity on delete:** No action

## gyventojai

**Owner:** dbo  
**Target DB name:** call  
**Number of columns:** 12  
**Number of indexes:** 0  
**Number of foreign keys:** 3  
**Codes:** 0  
**Type:** Table

**Extended attributes:**  
**OnFileGroup** PRIMARY  
**Clustered PK** Yes

Columns	Data type	Allow NULLs	Value/Range
miestas (FK)	char(10)	Not allowed	
pavarde (FK)	char(20)	Not allowed	
vardas (FK)	char(20)	Not allowed	
gatve (FK)	char(20)	Not allowed	
ak	char(11)	Allowed	
tel	char(10)	Not allowed	
mobtel	char(10)	Allowed	
email	char(20)	Allowed	
pastabos	char(30)	Allowed	
namo_nr	char(10)	Allowed	
buto_nr	int	Allowed	
gyv_id	int identity	Not allowed	

Foreign keys	Child	Parent
gatves_gyventojai_FK1	gatve	gatves.gatve
miestai_gyventojai_FK1	miestas	miestai.miestas
vardai_gyventojai_FK1	pavarde vardas	vardai.pavarde vardai.vardas
gyventojai_gyv_istorija_FK1	gyv_istorija.gyv_id	gyv_id

Column details	
<b>1. miestas</b> (FK)	
Physical data type:	char(10)
Allow NULLs:	Not allowed
<b>2. pavarde</b> (FK)	
Physical data type:	char(20)
Allow NULLs:	Not allowed
<b>3. vardas</b> (FK)	
Physical data type:	char(20)
Allow NULLs:	Not allowed
<b>4. gatve</b> (FK)	
Physical data type:	char(20)
Allow NULLs:	Not allowed
<b>5. ak</b>	
Physical data type:	char(11)
Allow NULLs:	Allowed
<b>6. tel</b>	
Physical data type:	char(10)

Allow NULLs:	Not allowed
<b><u>7. mobtel</u></b>	
Physical data type:	char(10)
Allow NULLs:	Allowed
<b><u>8. email</u></b>	
Physical data type:	char(20)
Allow NULLs:	Allowed
<b><u>9. pastabos</u></b>	
Physical data type:	char(30)
Allow NULLs:	Allowed
<b><u>10. namo_nr</u></b>	
Physical data type:	char(10)
Allow NULLs:	Allowed
<b><u>11. buto_nr</u></b>	
Physical data type:	int
Allow NULLs:	Allowed
<b><u>12. gvv_id</u></b>	
Physical data type:	int identity
Allow NULLs:	Not allowed

#### Foreign key details (child)

##### gatves\_gyventojai FK1

<b>Definition:</b>	<b>Child</b> gatve	<b>Parent</b> gatves.gatve
<b>Relationship type:</b>	Non-Identifying	
<b>Cardinality:</b>	One -to- Zero-or-More	
<b>Allow NULLs:</b>	Not allowed	
<b>Verb phrase:</b>	hasgatves_gyventojai_FK1	
<b>Inverse phrase:</b>	is of	
<b>Ref. Integrity on update:</b>	Cascade	
<b>Ref. Integrity on delete:</b>	No action	

##### miestai\_gyventojai FK1

<b>Definition:</b>	<b>Child</b> miestas	<b>Parent</b> miestai.miestas
<b>Relationship type:</b>	Non-Identifying	
<b>Cardinality:</b>	One -to- Zero-or-More	
<b>Allow NULLs:</b>	Not allowed	
<b>Verb phrase:</b>	hasmiestai_gyventojai_FK1	
<b>Inverse phrase:</b>	is of	
<b>Ref. Integrity on update:</b>	Cascade	
<b>Ref. Integrity on delete:</b>	No action	

##### vardai\_gyventojai FK1

<b>Definition:</b>	<b>Child</b> pavarde vardas	<b>Parent</b> vardai.pavarde vardai.vardas
<b>Relationship type:</b>	Non-Identifying	
<b>Cardinality:</b>	One -to- Zero-or-More	
<b>Allow NULLs:</b>	Not allowed	
<b>Verb phrase:</b>	hasvardai_gyventojai_FK1	
<b>Inverse phrase:</b>	is of	
<b>Ref. Integrity on update:</b>	Cascade	
<b>Ref. Integrity on delete:</b>	No action	

## gyvistorija

**Notes:** 2  
**Owner:** dbo  
**Target DB name:** call  
**Number of columns:** 14  
**Number of indexes:** 0  
**Number of foreign keys:** 0  
**Codes:** 0  
**Type:** View

Columns	Data type	Allow NULLs	Value/Range
gyvist_id	int	Not allowed	
pavarde	char(20)	Not allowed	
vardas	char(20)	Not allowed	
ak	char(11)	Allowed	
miestas	char(10)	Not allowed	
gatve	char(20)	Not allowed	
namo_nr	char(10)	Allowed	
buto_nr	int	Allowed	
tel	char(10)	Allowed	
mobtel	char(10)	Allowed	
email	char(20)	Allowed	
pastabos	char(30)	Allowed	
gyv_id	int identity	Not allowed	
user	char(10)	Not allowed	

### Column details

#### 1. gyvist\_id

**Physical data type:** int  
**Allow NULLs:** Not allowed

#### 2. pavarde

**Physical data type:** char(20)  
**Allow NULLs:** Not allowed

#### 3. vardas

**Physical data type:** char(20)  
**Allow NULLs:** Not allowed

#### 4. ak

**Physical data type:** char(11)  
**Allow NULLs:** Allowed

#### 5. miestas

**Physical data type:** char(10)  
**Allow NULLs:** Not allowed

#### 6. gatve

**Physical data type:** char(20)  
**Allow NULLs:** Not allowed

#### 7. namo\_nr

**Physical data type:** char(10)  
**Allow NULLs:** Allowed

#### 8. buto\_nr

**Physical data type:** int  
**Allow NULLs:** Allowed

#### 9. tel

**Physical data type:** char(10)

Allow NULLs:	Allowed
<b>10. mobtel</b>	
Physical data type:	char(10)
Allow NULLs:	Allowed
<b>11. email</b>	
Physical data type:	char(20)
Allow NULLs:	Allowed
<b>12. pastabos</b>	
Physical data type:	char(30)
Allow NULLs:	Allowed
<b>13. gyv_id</b>	
Physical data type:	int identity
Allow NULLs:	Not allowed
<b>14. user</b>	
Physical data type:	char(10)
Allow NULLs:	Not allowed

#### View definition

```

CREATE VIEW dbo.gyvistorija
AS
SELECT TOP (100) PERCENT dbo.gyv_istorija.gyvist_id, dbo.gyventojai.pavarde, dbo.gyventojai.vardas,
dbo.gyventojai.ak, dbo.gyventojai.miestas, dbo.gyventojai.gatve,
    dbo.gyventojai.namo_nr, dbo.gyventojai.buto_nr, dbo.gyventojai.tel, dbo.gyventojai.mobtel,
dbo.gyventojai.email, dbo.gyventojai.pastabos, dbo.gyventojai.gyv_id,
    dbo.gyv_istorija.[user]
FROM    dbo.gatves INNER JOIN
    dbo.gyventojai ON dbo.gatves.gatve = dbo.gyventojai.gatve INNER JOIN
    dbo.miestai ON dbo.gyventojai.miestas = dbo.miestai.miestas INNER JOIN
    dbo.vardai ON dbo.gyventojai.pavarde = dbo.vardai.pavarde AND dbo.gyventojai.vardas =
dbo.vardai.vardas INNER JOIN
    dbo.gyv_istorija ON dbo.gyventojai.gyv_id = dbo.gyv_istorija.gyv_id
ORDER BY dbo.gyv_istorija.gyvist_id

```

### gyvv

Notes:	1
Owner:	dbo
Target DB name:	call
Number of columns:	12
Number of indexes:	0
Number of foreign keys:	0
Codes:	0
Type:	View

Columns	Data type	Allow NULLs	Value/Range
gyv_id	int identity	Not allowed	
pavarde	char(20)	Not allowed	
vardas	char(20)	Not allowed	
ak	char(11)	Allowed	
gatve	char(20)	Not allowed	
namo_nr	char(10)	Allowed	
buto_nr	int	Allowed	
miestas	char(10)	Not allowed	
tel	char(10)	Allowed	
mobtel	char(10)	Allowed	
email	char(20)	Allowed	
pastabos	char(30)	Allowed	

## Column details

### 1. gyv\_id

Physical data type: int identity  
Allow NULLs: Not allowed

### 2. pavarde

Physical data type: char(20)  
Allow NULLs: Not allowed

### 3. vardas

Physical data type: char(20)  
Allow NULLs: Not allowed

### 4. ak

Physical data type: char(11)  
Allow NULLs: Allowed

### 5. gatve

Physical data type: char(20)  
Allow NULLs: Not allowed

### 6. namo\_nr

Physical data type: char(10)  
Allow NULLs: Allowed

### 7. buto\_nr

Physical data type: int  
Allow NULLs: Allowed

### 8. miestas

Physical data type: char(10)  
Allow NULLs: Not allowed

### 9. tel

Physical data type: char(10)  
Allow NULLs: Allowed

### 10. mobtel

Physical data type: char(10)  
Allow NULLs: Allowed

### 11. email

Physical data type: char(20)  
Allow NULLs: Allowed

### 12. pastabos

Physical data type: char(30)  
Allow NULLs: Allowed

## View definition

```
CREATE VIEW dbo.gyv  
AS  
SELECT TOP (100) PERCENT gyv_id, pavarde, vardas, ak, gatve, namo_nr, buto_nr, miestas, tel, mobtel, email,  
pastabos  
FROM dbo.gyventojai  
ORDER BY pavarde
```

Notes: 1  
Owner: dbo  
Target DB name: call

**Number of columns:** 18  
**Number of indexes:** 0  
**Number of foreign keys:** 0  
**Codes:** 0  
**Type:** View

Columns	Data type	Allow NULLs	Value/Range
a	char(20)	Not allowed	
b	char(15)	Not allowed	
c	char(20)	Allowed	
d	char(20)	Not allowed	
e	char(10)	Not allowed	
f	int	Allowed	
g	char(10)	Not allowed	
h	char(10)	Not allowed	
i	char(10)	Allowed	
j	char(20)	Allowed	
k	datetime	Allowed	
l	money	Allowed	
m	int	Allowed	
n	char(30)	Allowed	
o	char(20)	Not allowed	
p	char(20)	Not allowed	
r	char(20)	Not allowed	
s	char(20)	Allowed	

#### Column details

##### 1. a

**Physical data type:** char(20)  
**Allow NULLs:** Not allowed

##### 2. b

**Physical data type:** char(15)  
**Allow NULLs:** Not allowed

##### 3. c

**Physical data type:** char(20)  
**Allow NULLs:** Allowed

##### 4. d

**Physical data type:** char(20)  
**Allow NULLs:** Not allowed

##### 5. e

**Physical data type:** char(10)  
**Allow NULLs:** Not allowed

##### 6. f

**Physical data type:** int  
**Allow NULLs:** Allowed

##### 7. g

**Physical data type:** char(10)  
**Allow NULLs:** Not allowed

##### 8. h

**Physical data type:** char(10)  
**Allow NULLs:** Not allowed

##### 9. i

**Physical data type:** char(10)  
**Allow NULLs:** Allowed

##### 10. j

**Physical data type:** char(20)  
**Allow NULLs:** Allowed

**11. k**

Physical data type: datetime  
 Allow NULLs: Allowed

**12. l**

Physical data type: money  
 Allow NULLs: Allowed

**13. m**

Physical data type: int  
 Allow NULLs: Allowed

**14. n**

Physical data type: char(30)  
 Allow NULLs: Allowed

**15. o**

Physical data type: char(20)  
 Allow NULLs: Not allowed

**16. p**

Physical data type: char(20)  
 Allow NULLs: Not allowed

**17. r**

Physical data type: char(20)  
 Allow NULLs: Not allowed

**18. s**

Physical data type: char(20)  
 Allow NULLs: Allowed

**View definition**

```
CREATE VIEW dbo.I
AS
SELECT TOP (100) PERCENT dbo.imones.pavadinimas AS a, dbo.imones.im_k AS b, dbo.imones.pvm_k AS c,
dbo.imones.gatve AS d, dbo.imones.namo_nr AS e,
      dbo.imones.buto_nr AS f, dbo.imones.miestas AS g, dbo.imones.tel AS h, dbo.imones.mobtel AS i,
dbo.imones.email AS j, dbo.imones.reg_data AS k,
      dbo.imones.apyvarta AS l, dbo.imones.darb_sk AS m, dbo.imones.pastabos AS n, dbo.imones.rus_pavad
AS o, dbo.direktorai.vardas AS p,
      dbo.direktorai.pavarde AS r, dbo.direktorai.mobtel AS s
FROM   dbo.direktorai INNER JOIN
      dbo.imones ON dbo.direktorai.dir_id = dbo.imones.dir_id INNER JOIN
      dbo.imoniu_pavad ON dbo.imones.pavadinimas = dbo.imoniu_pavad.pavadinimas
ORDER BY a
```

**im\_istorija**

Owner: dbo  
 Target DB name: call  
 Number of columns: 3  
 Number of indexes: 0  
 Number of foreign keys: 2  
 Codes: 0  
 Type: Table

**Extended attributes:**

OnFileGroup PRIMARY  
 Clustered PK Yes

Columns	Data type	Allow NULLs	Value/Range
user (FK)	char(10)	Not allowed	
im_k (FK)	char(15)	Not allowed	
imist_id	int identity	Not allowed	

Foreign keys	Child	Parent
imones_im_storija_FK1	im_k	imones.im_k
FK_im_istorija_vartotojai	user	vartotojai.user

Column details	
<b>1. user (FK)</b>	
Physical data type:	char(10)
Allow NULLs:	Not allowed
<b>2. im_k (FK)</b>	
Physical data type:	char(15)
Allow NULLs:	Not allowed
<b>3. imist_id</b>	
Physical data type:	int identity
Allow NULLs:	Not allowed

#### Foreign key details (child)

##### imones\_im\_storija\_FK1

<b>Definition:</b>	<b>Child</b> im_k	<b>Parent</b> imones.im_k
<b>Relationship type:</b>	Non-Identifying	
<b>Cardinality:</b>	One -to- Zero-or-More	
<b>Allow NULLs:</b>	Not allowed	
<b>Verb phrase:</b>	hasimones_im_storija_FK1	
<b>Inverse phrase:</b>	is of	
<b>Ref. Integrity on update:</b>	No action	
<b>Ref. Integrity on delete:</b>	No action	

##### FK\_im\_istorija\_vartotojai

<b>Definition:</b>	<b>Child</b> user	<b>Parent</b> vartotojai.user
<b>Relationship type:</b>	Non-Identifying	
<b>Cardinality:</b>	One -to- Zero-or-More	
<b>Allow NULLs:</b>	Not allowed	
<b>Verb phrase:</b>	hasFK_im_istorija_vartotojai	
<b>Inverse phrase:</b>	is of	
<b>Ref. Integrity on update:</b>	Cascade	
<b>Ref. Integrity on delete:</b>	No action	

## imistorija

<b>Notes:</b>	2
<b>Owner:</b>	dbo
<b>Target DB name:</b>	call
<b>Number of columns:</b>	20
<b>Number of indexes:</b>	0
<b>Number of foreign keys:</b>	0
<b>Codes:</b>	0
<b>Type:</b>	View

Columns	Data type	Allow NULLs	Value/Range
imist_id	int	Not allowed	
pavadinimas	char(20)	Not allowed	



im_k	char(15)	Not allowed
pvm_k	char(20)	Allowed
gatve	char(20)	Not allowed
namo_nr	char(10)	Not allowed
buto_nr	int	Allowed
miestas	char(10)	Not allowed
tel	char(10)	Not allowed
mobtel	char(10)	Allowed
email	char(20)	Allowed
reg_data	datetime	Allowed
apyvarta	money	Allowed
darb_sk	int	Allowed
pastabos	char(30)	Allowed
pavarde	char(20)	Not allowed
vardas	char(20)	Not allowed
dirmob	char(20)	Allowed
rus_pavad	char(20)	Not allowed
user	char(10)	Not allowed

#### Column details

##### 1. imist\_id

Physical data type: int  
Allow NULLs: Not allowed

##### 2. pavadinimas

Physical data type: char(20)  
Allow NULLs: Not allowed

##### 3. im\_k

Physical data type: char(15)  
Allow NULLs: Not allowed

##### 4. pvm\_k

Physical data type: char(20)  
Allow NULLs: Allowed

##### 5. gatve

Physical data type: char(20)  
Allow NULLs: Not allowed

##### 6. namo\_nr

Physical data type: char(10)  
Allow NULLs: Not allowed

##### 7. buto\_nr

Physical data type: int  
Allow NULLs: Allowed

##### 8. miestas

Physical data type: char(10)  
Allow NULLs: Not allowed

##### 9. tel

Physical data type: char(10)  
Allow NULLs: Not allowed

##### 10. mobtel

Physical data type: char(10)  
Allow NULLs: Allowed

##### 11. email

Physical data type: char(20)  
Allow NULLs: Allowed

##### 12. reg\_data

Physical data type: datetime  
Allow NULLs: Allowed

### 13. apyvarta

Physical data type: money  
Allow NULLs: Allowed

### 14. darb\_sk

Physical data type: int  
Allow NULLs: Allowed

### 15. pastabos

Physical data type: char(30)  
Allow NULLs: Allowed

### 16. pavarde

Physical data type: char(20)  
Allow NULLs: Not allowed

### 17. vardas

Physical data type: char(20)  
Allow NULLs: Not allowed

### 18. dirmob

Physical data type: char(20)  
Allow NULLs: Allowed

### 19. rus\_pavad

Physical data type: char(20)  
Allow NULLs: Not allowed

### 20. user

Physical data type: char(10)  
Allow NULLs: Not allowed

## **View definition**

```
CREATE VIEW dbo.imistorija
AS
SELECT TOP (100) PERCENT dbo.im_istorija.imist_id, dbo.imones.pavadinimas, dbo.imones.im_k,
dbo.imones.pvm_k, dbo.imones.gatve, dbo.imones.namo_nr,
    dbo.imones.buto_nr, dbo.imones.miestas, dbo.imones.tel, dbo.imones.mobtel, dbo.imones.email,
dbo.imones.reg_data, dbo.imones.apyvarta, dbo.imones.darb_sk,
    dbo.imones.pastabos, dbo.direktorai.pavarde, dbo.direktorai.vardas, dbo.direktorai.mobtel AS dirmob,
dbo.imones.rus_pavad, dbo.im_istorija.[user]
FROM    dbo.direktorai INNER JOIN
    dbo.imones ON dbo.direktorai.dir_id = dbo.imones.dir_id INNER JOIN
    dbo.gatves ON dbo.imones.gatve = dbo.gatves.gatve INNER JOIN
    dbo.imoniu_pavad ON dbo.imones.pavadinimas = dbo.imoniu_pavad.pavadinimas INNER JOIN
    dbo.miestai ON dbo.imones.miestas = dbo.miestai.miestas INNER JOIN
    dbo.vardai ON dbo.direktorai.pavarde = dbo.vardai.pavarde AND dbo.direktorai.vardas =
dbo.vardai.vardas INNER JOIN
    dbo.veiklos_rusys ON dbo.imones.rus_pavad = dbo.veiklos_rusys.rus_pavad INNER JOIN
    dbo.im_istorija ON dbo.imones.im_k = dbo.im_istorija.im_k
ORDER BY dbo.im_istorija.imist_id
```

## **imones**

Owner: dbo  
Target DB name: call  
Number of columns: 16  
Number of indexes: 0  
Number of foreign keys: 5  
Codes: 0  
Type: Table

Extended attributes:

OnFileGroup PRIMARY  
 Clustered PK Yes

Columns	Data type	Allow NULLs	Value/Range
rus_pavad (FK)	char(20)	Not allowed	
tel	char(10)	Not allowed	
mobtel	char(10)	Allowed	
im_k	char(15)	Not allowed	
pvm_k	char(20)	Allowed	
reg_data	datetime	Allowed	
apyvarta	money	Allowed	
darb_sk	int	Allowed	
email	char(20)	Allowed	
pastabos	char(30)	Allowed	
dir_id (FK)	int	Not allowed	
namo_nr	char(10)	Not allowed	
buto_nr	int	Allowed	
pavadinimas (FK)	char(20)	Not allowed	
gatve (FK)	char(20)	Not allowed	
miestas (FK)	char(10)	Not allowed	

Foreign keys	Child	Parent
FK_imones_direktorai	dir_id	direktorai.dir_id
gatves_imones_FK1	gatve	gatves.gatve
FK_imones_imoniu_pavad1	pavadinimas	imoniu_pavad.pavadinimas
FK_imones_miestai	miestas	miestai.miestas
FK_imones_veiklos_rusys	rus_pavad	veiklos_rusys.rus_pavad
imones_im_storija_FK1	imistorija.im_k	im_k

Column details	
<b>1. rus_pavad (FK)</b>	
Physical data type:	char(20)
Allow NULLs:	Not allowed
<b>2. tel</b>	
Physical data type:	char(10)
Allow NULLs:	Not allowed
<b>3. mobtel</b>	
Physical data type:	char(10)
Allow NULLs:	Allowed
<b>4. im_k</b>	
Physical data type:	char(15)
Allow NULLs:	Not allowed
<b>5. pvm_k</b>	
Physical data type:	char(20)
Allow NULLs:	Allowed
<b>6. reg_data</b>	
Physical data type:	datetime
Allow NULLs:	Allowed
<b>7. apyvarta</b>	
Physical data type:	money
Allow NULLs:	Allowed
<b>8. darb_sk</b>	
Physical data type:	int
Allow NULLs:	Allowed
<b>9. email</b>	
Physical data type:	char(20)
Allow NULLs:	Allowed

<b><u>10. pastabos</u></b>	
Physical data type:	char(30)
Allow NULLs:	Allowed
<b><u>11. dir_id</u></b> (FK)	
Physical data type:	int
Allow NULLs:	Not allowed
<b><u>12. namo_nr</u></b>	
Physical data type:	char(10)
Allow NULLs:	Not allowed
<b><u>13. buto_nr</u></b>	
Physical data type:	int
Allow NULLs:	Allowed
<b><u>14. pavadinimas</u></b> (FK)	
Physical data type:	char(20)
Allow NULLs:	Not allowed
<b><u>15. gatve</u></b> (FK)	
Physical data type:	char(20)
Allow NULLs:	Not allowed
<b><u>16. miestas</u></b> (FK)	
Physical data type:	char(10)
Allow NULLs:	Not allowed

#### Foreign key details (child)

##### FK imones direktoriai

<b>Definition:</b>	<b>Child</b> dir_id	<b>Parent</b> direktoriai.dir_id
<b>Relationship type:</b>	Non-Identifying	
<b>Cardinality:</b>	One -to- Zero-or-More	
<b>Allow NULLs:</b>	Not allowed	
<b>Verb phrase:</b>	hasFK_imones_direktoriai	
<b>Inverse phrase:</b>	is of	
<b>Ref. Integrity on update:</b>	No action	
<b>Ref. Integrity on delete:</b>	No action	

##### gatves imones FK1

<b>Definition:</b>	<b>Child</b> gatve	<b>Parent</b> gatves.gatve
<b>Relationship type:</b>	Non-Identifying	
<b>Cardinality:</b>	One -to- Zero-or-More	
<b>Allow NULLs:</b>	Not allowed	
<b>Verb phrase:</b>	hasgatves_imones_FK1	
<b>Inverse phrase:</b>	is of	
<b>Ref. Integrity on update:</b>	Cascade	
<b>Ref. Integrity on delete:</b>	No action	

##### FK imones imoniu pavad1

<b>Definition:</b>	<b>Child</b> pavadinimas	<b>Parent</b> imoniu_pavad.pavadinimas
<b>Relationship type:</b>	Non-Identifying	
<b>Cardinality:</b>	One -to- Zero-or-More	
<b>Allow NULLs:</b>	Not allowed	
<b>Verb phrase:</b>	hasFK_imones_imoniu_pavad1	
<b>Inverse phrase:</b>	is of	

Ref. Integrity on update: Cascade  
 Ref. Integrity on delete: No action

**FK imones miestai**

**Definition:** **Child** miestai **Parent** miestai.miestas

**Relationship type:** Non-Identifying  
**Cardinality:** One -to- Zero-or-More  
**Allow NULLs:** Not allowed  
**Verb phrase:** hasFK\_imones\_miestai  
**Inverse phrase:** is of  
**Ref. Integrity on update:** Cascade  
**Ref. Integrity on delete:** No action

**FK imones veiklos rusvs**

**Definition:** **Child** rus\_pavad **Parent** veiklos\_rusys.rus\_pavad

**Relationship type:** Non-Identifying  
**Cardinality:** One -to- Zero-or-More  
**Allow NULLs:** Not allowed  
**Verb phrase:** hasFK\_imones\_veiklos\_rusys  
**Inverse phrase:** is of  
**Ref. Integrity on update:** Cascade  
**Ref. Integrity on delete:** No action

**imoniu\_pavad**

**Owner:** dbo  
**Target DB name:** call  
**Number of columns:** 2  
**Number of indexes:** 0  
**Number of foreign keys:** 0  
**Codes:** 0  
**Type:** Table

**Extended attributes:**  
**OnFileGroup** PRIMARY  
**Clustered PK** Yes

Columns	Data type	Allow NULLs	Value/Range
pavad_id	int identity	Not allowed	
pavadinimas	char(20)	Not allowed	

Foreign keys	Child	Parent
FK_imones_imoniu_pavad1	imones.pavadinimas	pavadinimas

Column details
<b><u>1. pavad_id</u></b> <b>Physical data type:</b> int identity <b>Allow NULLs:</b> Not allowed
<b><u>2. pavadinimas</u></b> <b>Physical data type:</b> char(20) <b>Allow NULLs:</b> Not allowed

**miestai**

**Owner:** dbo  
**Target DB name:** call  
**Number of columns:** 2  
**Number of indexes:** 0  
**Number of foreign keys:** 0  
**Codes:** 0  
**Type:** Table

**Extended attributes:**  
**OnFileGroup** PRIMARY  
**Clustered PK** Yes

Columns	Data type	Allow NULLs	Value/Range
miest_id	int identity	Not allowed	
miestas	char(10)	Not allowed	

Foreign keys	Child	Parent
miestai_direktoraii_FK1	direktoraii.miestas	miestas
miestai_gyventojaii_FK1	gyventojaii.miestas	miestas
FK_imones_miestai	imones.miestas	miestas

Column details	
<b>1. miest_id</b>	
Physical data type:	int identity
Allow NULLs:	Not allowed
<b>2. miestas</b>	
Physical data type:	char(10)
Allow NULLs:	Not allowed

## sysdiagrams

**Notes:** 1  
**Owner:** dbo  
**Target DB name:** call  
**Number of columns:** 5  
**Number of indexes:** 1  
**Number of foreign keys:** 0  
**Codes:** 0  
**Type:** Table

**Extended attributes:**  
**OnFileGroup** PRIMARY  
**Clustered PK** Yes

Columns	Data type	Allow NULLs	Value/Range
name (U1)	sysname	Not allowed	
principal_id (U1)	int	Not allowed	
diagram_id	int identity	Not allowed	
version	int	Allowed	
definition	image	Allowed	

Indexes	Columns	Sort order
UK_principal_name (U1)	principal_id name	Ascending Ascending

Column details	
<b>1. name (U1)</b>	
Physical data type:	sysname
Allow NULLs:	Not allowed
<b>2. principal_id (U1)</b>	

Physical data type: int  
 Allow NULLs: Not allowed

**3. diagram\_id**

Physical data type: int identity  
 Allow NULLs: Not allowed

**4. version**

Physical data type: int  
 Allow NULLs: Allowed

**5. definition**

Physical data type: image  
 Allow NULLs: Allowed

**Index details**

**UK\_principal\_name**

Column(s): principal\_id (Asc)  
 name (Asc)

Unique: Yes

Extended attributes:

OnFileGroup	PRIMARY
CLUSTERED	No
IGNORE_DUP_KEY	No
FILLFACTOR	0
PAD_INDEX	No
DROP_EXISTING	No
STATISTICS_NORECOMPUTE	No
SORT_IN_TEMPDB	No
ONLINE	No
ALLOW_ROW_LOCKS	Yes
ALLOW_PAGE_LOCKS	Yes

**vardai**

Notes: 1  
 Owner: dbo  
 Target DB name: call  
 Number of columns: 3  
 Number of indexes: 0  
 Number of foreign keys: 0  
 Codes: 0  
 Type: Table

Extended attributes:

OnFileGroup	PRIMARY
Clustered PK	Yes

Columns	Data type	Allow NULLs	Value/Range
vard_id	int identity	Not allowed	
pavarde	char(20)	Not allowed	
vardas	char(20)	Not allowed	

Foreign keys	Child	Parent
vardai_direktoraii_FK1	direktoraii.pavarde	pavarde
	direktoraii.vardas	vardas
vardai_gyventojaii_FK1	gyventojai.pavarde	pavarde
	gyventojai.vardas	vardas

**Column details**

**1. vard\_id**

Physical data type: int identity

Allow NULLs: Not allowed

**2. pavarde**

Physical data type: char(20)  
Allow NULLs: Not allowed

**3. vardas**

Physical data type: char(20)  
Allow NULLs: Not allowed

## vartotojai

Notes: 1  
Owner: dbo  
Target DB name: call  
Number of columns: 6  
Number of indexes: 0  
Number of foreign keys: 0  
Codes: 0  
Type: Table

Extended attributes:  
OnFileGroup PRIMARY  
Clustered PK Yes

Columns	Data type	Allow NULLs	Value/Range
vardas	char(20)	Not allowed	
pavard	char(20)	Not allowed	
telefonas	char(20)	Not allowed	
user	char(10)	Not allowed	
pass	char(10)	Not allowed	
galioja	datetime	Not allowed	

Foreign keys	Child	Parent
vartotojai_gyv_istorija_FK1	gyv_istorija.user	user
FK_im_istorija_vartotojai	im_istorija.user	user

### Column details

**1. vardas**

Physical data type: char(20)  
Allow NULLs: Not allowed

**2. pavard**

Physical data type: char(20)  
Allow NULLs: Not allowed

**3. telefonas**

Physical data type: char(20)  
Allow NULLs: Not allowed

**4. user**

Physical data type: char(10)  
Allow NULLs: Not allowed

**5. pass**

Physical data type: char(10)  
Allow NULLs: Not allowed

**6. galioja**

Physical data type: datetime  
Allow NULLs: Not allowed



## veiklos\_rusys

Notes: 1  
Owner: dbo  
Target DB name: call  
Number of columns: 2  
Number of indexes: 0  
Number of foreign keys: 0  
Codes: 0  
Type: Table

Extended attributes:  
OnFileGroup PRIMARY  
Clustered PK Yes

Columns	Data type	Allow NULLs	Value/Range
rus_id	int identity	Not allowed	
rus_pavad	char(20)	Not allowed	

Foreign keys	Child	Parent
FK_imones_veiklos_rusys	imones.rus_pavad	rus_pavad

### Column details

#### 1. rus\_id

Physical data type: int identity  
Allow NULLs: Not allowed

#### 2. rus\_pavad

Physical data type: char(20)  
Allow NULLs: Not allowed