

**KAUNO TECHNOLOGIJOS UNIVERSITETAS  
INFORMATIKOS FAKULTETAS  
INFORMACIJOS SISTEMŲ KATEDRA**

**Paulius Paužas**

**Viešbučių rezervavimo informacinė sistema**

Magistro darbas

**Vadovas  
doc. L. Nemuraitė**

**KAUNAS, 2005**

**KAUNO TECHNOLOGIJOS UNIVERSITETAS  
INFORMATIKOS FAKULTETAS  
INFORMACIJOS SISTEMŲ KATEDRA**

**Viešbučių rezervavimo informacinė sistema**

Informacinių technologijų magistro baigiamasis darbas

**Kalbos konsultantė**  
Lietuvių kalbos katedros lektorė  
dr. J. Mikelionienė

**Vadovas**  
doc.L.Nemuraitė

**Recenzentas**

**Atliko**  
IFM 9/3 gr. stud.  
P.Paužas

**KAUNAS, 2005**

# TURINYS

1. ĮVADAS.....	6
2. VIEŠBUČIŲ INFORMACINĖS SISTEMOS REIKALAVIMŲ IR ĮGYVENDINIMO TECHNOLOGIJŲ ANALIZĖ.....	8
2.1. Darbo tikslas ir uždaviniai.....	8
2.2. Kompiuterinių programų apžvalga.....	8
2.2.1. Microsoft Visual Studio .NET.....	8
2.2.2. .NET Framework.....	10
2.2.3. ADO.NET ( <i>ActiveX Data Objects</i> ).....	10
2.2.4. ADO.NET privalumai.....	12
2.2.5. Microsoft Visual C# .NET.....	13
2.2.6. ASP.NET.....	13
2.2.7. SQL Server 2000.....	14
2.3. Naudojamų programų apibendrinimas.....	14
2.4. Analogiškų sistemų apibūdinimas.....	15
2.4.1. Trumpas palyginimas JAVA(J2EE) ir .NET technologijų.....	16
2.4.1.1. Kodo eilučių kiekio palyginimas.....	17
2.4.1.2. Transakcijos atsakymo laikas.....	17
2.4.1.3. Interneto taikomosios programos sistemos darbo išbandymas.....	18
2.5. Bendri reikalavimai sistemai.....	20
2.5.1. Produkto apibrėžimas.....	20
2.5.2. Vartotojų apibūdinimas.....	20
2.5.3. Funkciniai reikalavimai sistemai.....	20
2.5.4. Nefunkciniai reikalavimai sistemai.....	21
3. VIESBUČIŲ INFORMACINĖS SISTEMOS MODELIS.....	23
3.1. Formaliai specifikuojamos sistemos aprašymas.....	23
3.2. Formalus aprašymas.....	24
3.3. Viešbučio rezervavimo sistemos taikomųjų uždavinių modelis.....	37
3.4. Sistemos architektūra.....	39
3.5. Kliento aplinkos „žemėlapis“.....	40
3.6. Darbuotojų aplinkos „žemėlapis“.....	40
3.7. Viešbučių sistemos klasių diagrama.....	41
3.8. Viešbučių sistemos klasių aprašymas.....	42
3.9. Viešbučių rezervavimo sistemos diegimo architektūra.....	52
3.10. Programos objektų aprašymas.....	52
3.11. Duomenų bazės schema.....	55
3.12. Naudojamų poschemių (View) struktūros specifikacija.....	56
3.13. Sistemos programinė realizacija.....	57
3.13.1. Prototipinė viešbučių rezervavimo sistema.....	57
3.13.2. Projektuojama viešbučių rezervavimo sistema.....	58
4. VARTOTOJO DOKUMENTACIJA.....	59
4.1. Sistemos funkcinis aprašymas.....	59
4.2. Rezervavimo sistemos kliento dalies vadovas.....	60
4.2.1. Prisijungimas prie sistemos interneto naršykle.....	60
4.2.2. Prisijungimas prie sistemos naudojant WAP technologiją.....	66
4.3. Rezervavimo sistemos administravimo vadovas.....	66

4.4. Sistemos instaliavimo dokumentas .....	71
5. PRODUKTO KOKYBĖS ĮVERTINIMAS .....	72
6. IŠVADOS .....	74
7. LITERATŪRA .....	75
8. PRIEDAI .....	76
8.1. Sukurtos saugomos procedūros .....	76

## **Summary**

In this work the Hotel Reservation system is created and developed. The development starts with setting functional and non-functional requirements for the system and identifying the end users: Administrator, Secretary, Manager, Internet user. The system is realized with suitable technologies from Microsoft .NET namely ASP.NET, ADO.NET, Visual C#.NET and is available both through stationary (personal computer) and mobile (cell phone) devices. The system is tested and documented in detail for any end user. It fulfilled all the requirements and expectations of the end users.

## Paveikslų sąrašas

1	pav. Microsoft .NET platforma .....	9
2	pav. .NET Framework architektūra .....	10
3	pav. ADO.NET architektūra .....	11
4	pav. ADO.NET kliento/serverio architektūra .....	12
5	pav. Kodo eilučių skaičiaus palyginimas .....	17
6	pav. Puslapių perdavimas per sekundę .....	18
7	pav. Maksimalus prisijungusių klientų skaičius .....	19
8	pav. Pateikta puslapų per sekundę, 2 procesorių serveriai .....	19
9	pav. Taikomųjų uždavinių modelis .....	37
10	pav. Sistemos architektūra .....	39
11	pav. Kliento aplinkos “žemėlapis” .....	40
12	pav. Darbuotojų aplinkos “žemėlapis” .....	40
13	pav. Viešbučių sistemos administravimo programos klasių diagrama .....	41
14	pav. Viešbučių sistemos rezervavimo programos klasių diagrama .....	42
15	pav. Viešbučių rezervavimo sistemos diegimo architektūra .....	52
16	pav. Duomenų bazės schema .....	55
17	pav. Pagrindinis klientų prisijungimo puslapis .....	60
18	pav. Kambarių apžiūrėjimas .....	61
19	pav. Naujo vartotojo registracija .....	62
20	pav. Prisijungimas .....	63
21	pav. Klientų užsakymų aplinka .....	64
22	pav. Naujas užsakymas .....	65
23	pav. WAP prisijungimas .....	66
24	pav. Rezervavimo sistemos administracinės programos prisijungimas .....	66
25	pav. Administravimo sistema .....	67
26	pav. Kambarių administravimo langas .....	67
27	pav. Kambarių administravimo langas 2 .....	68
28	pav. Kambarių patvirtinimo langas .....	69
29	pav. Užsakymų administravimo langas .....	69
30	pav. Užsakymų administravimas, filtravimo galimybės .....	70
31	pav. Sugeneruota ataskaita pagal pasirinktus filtrus .....	71
32	pav. Kokybės produkto įvertinimo grafikas .....	73

## Lentelių sąrašas

1	lentelė. Naudojamos programos .....	14
2	lentelė. Panašios sistemos .....	16
3	lentelė. Kodo eilučių kiekio palyginimas .....	17
4	lentelė Transakcijos atsakymo laikas .....	17
5	lentelė. Reikalavimai Visual Studio .NET 2003 [4] programai .....	21
6	lentelė. Viešbučių sistemos klasių aprašymas .....	42
7	lentelė. Programos objektų aprašymas .....	52
8	lentelė. Naudojamų poschemių (View) struktūros specifikacija .....	56
9	lentelė. Produkto kokybės įvertinimas .....	72

# 1. ĮVADAS

Darbo tikslas – sukurti viešbučių užsakymo informacinę sistemą, kuri leistų vartotojams rasti informaciją apie viešbučių paslaugas, rezervuoti vietas viešbutyje internetu ar mobiliais įrenginiais, o viešbučio darbuotojams – tvarkyti informaciją apie paslaugas, patvirtinti užsakymus ir pan.

Sistemos kūrimas motyvuojamas tuo, kad žmonės darosi vis mobilesni, reikalinga sistema, kuri palengvintų viešbučių administracijos darbą ir supaprastintų klientams rezervavimo procesą. Šiuo metu daugumai žmonių internetas yra laisvai prieinamas ir manoma, kad ateityje tokių žmonių dar padaugės. Internetas populiarės, todėl tikslinga plėsti paslaugų teikimą internetu.

Kuriant sistemą stengiamasi ją padaryti kuo informatyvesnę ir interaktyvesnę. Klientas galės apžiūrėti kiekvieno kambario paveikslukus ar pasiskaityti papildomą informaciją. Dabar daugumoje esančių sistemų apie teikiamas paslaugas interneto puslapyje informacija yra pateikiama statiškai, nėra galimybės pamatyti rezervuojamo kambario pavyzdžių.

Kuriamoje sistemoje kiekvieno kambario informacija ir paveikslukai yra saugomi duomenų bazėje, kurie yra pateikiami dinamiškai, klientui apžiūrint teikiamas paslaugas ar rezervuojant. Tokiu būdu viešbučio darbuotojai, naudodamiesi administracine programa, gali greitai keisti informaciją apie teikiamas paslaugas ar pridėti naujų paslaugų. Tad naudojant šią sistemą nereikia perkrauti ar perkompiliuoti programos įdiegus naujas paslaugas ar padarius pakeitimus.

Taip pat didėjant sąsajos įrenginių aibe, reikia paslaugų, kurias būtų galima perkelti į kitus įrenginius, pavyzdžiui, mobiliuosius telefonus. Todėl kuriamoje viešbučių rezervavimo sistemoje bus galimybė klientams mobiliuoju telefonu užsisakyti norimo tipo kambarį ar konferencijų salę.

Sistema buvo projektuojama taip, kad turėtų pagrindines funkcijas, kurios būdingos paslaugų užsakymo informacinėms sistemoms. Todėl ją galima nesunkiai pritaikyti kitiems tikslams. Pasirinkta C# programavimo kalba, kuri tinka tiek interneto sistemoms, tiek taikomosioms programoms ir leidžia naudoti visus teikiamus .NET technologijos privalumus (*remoting*, *web services* ir kitus).

Viešbučių rezervavimo sistema suprojektuota naudojant Microsoft naujausias technologijas (MS.NET), kurios atskleis šių technologijų privalumus ir pranašumus prieš kitas technologijas. MS .NET technologijos suteikia galimybę plačiai pritaikyti sukurtą sistemą. Sistema bus pasiekama ne tik įprastais kompiuteriais, bet ir kitais įrenginiais (*smart devices*) ar mobiliaisiais

telefonais. Tokiu būdu sistema bus skirta platesniam vartotojų ratui, o tai ir yra kuriamos sistemos tikslas – pritraukti kuo daugiau klientų ir palengvinti rezervavimo procesą.

Darbo dėstomoji dalis susideda iš keturių pagrindinių dalių: 1. Viešbučių informacinės sistemos reikalavimų ir įgyvendinimo technologijų analizės, 2. Viešbučių informacinės sistemos modelio, 3. Vartotojo dokumentacijos ir 4. Produkto kokybės analizės.

Pirmojoje dalyje yra aptariami darbo tikslai ir uždaviniai, trumpai apžvelgiamos visos .NET technologijos, kurios yra pasirinktos sistemos realizacijai, įvertinami jų pranašumai prieš kitas technologijas bei argumentuojamas jų tinkamumas būtent Viešbučių rezervavimo sistemos kūrimui. Kalbama apie šias technologijas: Microsoft Visual Studio.NET kaip patogią aplinką .NET taikomosioms programoms kurti, ADO.NET kaip technologiją duomenų išgavimui iš MS SQL Server 2000, ASP.NET kaip priemonę internetinei sąsajai sukurti bei Visual C#.NET kaip programavimo kalbą, kuri realizuoja sistemą bei tinkamai išnaudoja .NET technologijų privalumus. Taip pat apibūdinamos analogiškos sistemos bei suformuluojami bendri reikalavimai būsimai sistemai: apibrėžiamas produktas, apibūdinami vartotojai, išvardinami tiek funkciniai, tiek nefunkciniai reikalavimai sistemai.

Antrojoje dalyje pateikiamas formalus sistemos aprašas bei išsamus Viešbučių rezervavimo sistemos taikomųjų uždavinių modelis, sistemos architektūra. Sudaromi klientų bei darbuotojų aplinkų žemėlapiai. Išskiriamos Viešbučių sistemos klasės ir jos aprašomos. Papildomai pateikiamos šios diagramos bei lentelės su aprašais: Viešbučių rezervavimo sistemos diegimo architektūra, programos objektų aprašymas, duomenų bazės schema, naudojamų poschemių struktūros specifikacija. Toliau demonstruojama sistemos realizacija, vartotojų matomi langai bei smulkiai aprašomos galimos jų funkcijos. Taip pat aprašomas prototipinis sistemos kūrimo būdas, kadangi jis buvo pasirinktas kaip tinkamiausias aiškiems vartotojų poreikiams išskirti sistemos kūrimo procese.

Trečiojoje dalyje sistema dokumentuojama vartotojui. Ji susideda iš sistemos funkcinio aprašymo, rezervavimo sistemos kliento dalies vadovo, rezervavimo sistemos administratoriaus vadovo bei sistemos instaliavimo dokumento.

Ketvirtoje dalyje atliekamas produkto kokybės įvertinimas.

## **2. VIEŠBUČIŲ INFORMACINĖS SISTEMOS REIKALAVIMŲ IR ĮGYVENDINIMO TECHNOLOGIJŲ ANALIZĖ**

Šiame skyriuje pateikiamas sprendžiamo uždavinio aktualumo ir naujų sprendimo variantų poreikio pagrindimas, formuluojama sistemos kūrimo užduotis. Vystantis naujoms technologijoms įmonėms yra svarbu perimti ir įdiegti naujas technologijas, nes tai padeda atlaikyti konkurenciją. Viešbučio rezervavimo sistema bus kuriama pasinaudojant naujausiomis technologijomis, kurios padidins viešbučių rezervavimo efektyvumą, pagerins informacijos tiekimo kokybę ir kiekybę.

### **2.1. Darbo tikslas ir uždaviniai**

Sukurti sistemą, kuri aptarnautų viešbučio užsakymų priėmimą, naudojantis Microsoft .NET technologija.

Viešbučio apgyvendinimo sistema leis:

- Klientui rezervuoti kambarius ar konferencijų salę internete (kompiuteriu ar mobiliuoju telefonu);
- Peržiūrėti ar gauti kitą išsamią informaciją apie rezervuojamus apartamentus;
- Saugoti duomenų bazėje duomenis apie pastovius klientus, kuriems bus taikomos nuolaidos.

Sistema turi turėti galimybę integruotis su vidine viešbučio informacine sistema.

### **2.2. Kompiuterinių programų apžvalga**

#### **2.2.1. Microsoft Visual Studio .NET**

Naudojantis Visual Studio .NET programų kūrėjai gali:

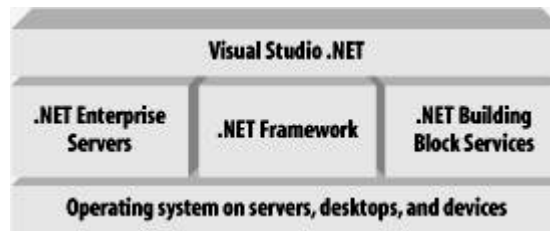
- Sukurti naujos kartos internetą.
- Greitai ir efektyviai sukurti galingas programas
- Apimti įvairias platformas ar įrenginius

Visual Studio .NET [3] yra kūrimo aplinka, kurią naudojant galima sukurti programas ir integruoti per XML tinklo paslaugas (*services*). Tai leidžia programoms dalintis duomenimis per



interneta, XML tinklo paslaugos įgalina kūrėjus parašyti programas nauju ar esamu kodu, priklausomai nuo platformos, programavimo kalbos ar objekto modelio.

**Microsoft .NET platforma** susideda iš 5 pagrindinių komponentų:



**1 pav. Microsoft .NET platforma**

Žemiausiame sluoksnyje yra operacinė sistema, kuri gali būti viena iš įvairių Windows operacinių sistemų: Windows XP, Windows 2000, Windows Server 2003, Windows ME, ir Windows CE. Viena iš .NET strategijų yra integruoti kuo daugiau naujos kartos įrenginių (*smart devices*).

Aukštesniame nei operacinė sistema lygmenyje yra įvairūs serverių produktai, kurie pagreitina didelių verslo sistemų sukūrimą. Serverių produktus sudaro: Application Center, BizTalk Server, Commerce Server, Exchange Server, Host Integration Server, Internet Security, Acceleration Server, ir SQL Server.

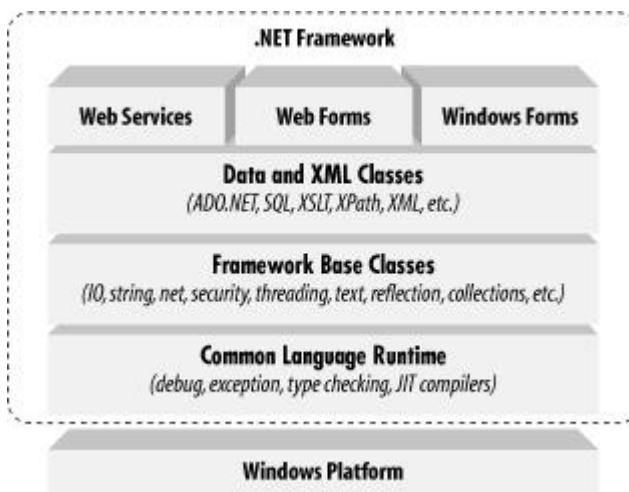
Viršutinį sluoksnį sudaro .NET architektūros kūrimo įrankis Visual Studio .NET (VS.NET), kuris įgalina greitą internetinių paslaugų (*web services*) ir kitų programų kūrimą. VS.NET yra integruota kūrimo aplinka (*Integrated Development Environment (IDE)*), kuri palaiko keturias skirtingas kalbas ir palaiko klaidų taisymą (*debugging*) tarp skirtingų kalbų.

.NET Framework yra kūrimo ir vykdymo infrastruktūra, kuri pakeičia verslo programų kūrimą Windows sistemoje. .NET Framework apima CLR (*common language runtime*) ir dažnai naudojamų klasių struktūrą, kurios gali būti panaudotos visų .NET kalbų.

### 2.2.2. .NET Framework

Microsoft .NET Framework įgyvendina tokius projekto tikslus, kurie yra kartu praktiški ir toli siekiantys. Pagrindiniai Microsoft .NET Framework tikslai yra komponentų palaikymas, kalbų integravimas, taikomųjų programų bendravimas per tinklą, supaprastintas kūrimas ir failų išdėstymas, pagerintas patikimumas ir didesnis saugumas.

#### .NET Framework architektūra



2 pav. .NET Framework architektūra

### 2.2.3. ADO.NET (*ActiveX Data Objects*)

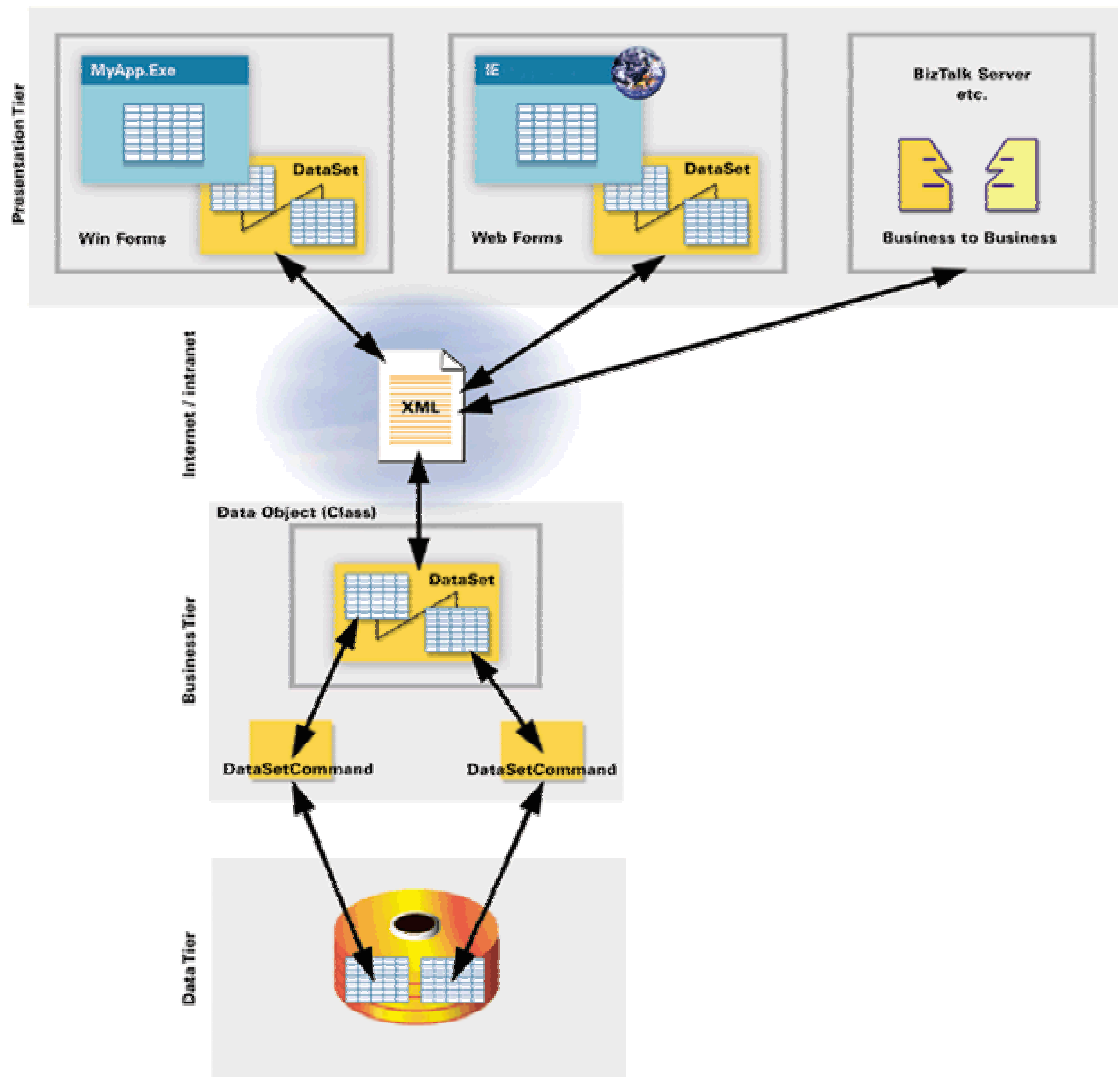
ADO.NET [1] perėjo nuo COM-grindžiamų įrašų aibių (*COM-based recordset*) ir naudoja XML kaip transportavimo duomenų formatą.

Kadangi XML yra nuo platformos nepriklausantis, ADO.NET prasiplėtė ir tapo prieinama visoms programoms, kurios gali koduoti/iš koduoti XML.

Tai didžiulis ADO.NET privalumas, nes COM-pagrindū įrašai (*COM-based recordset*) nėra nepriklausomi nuo platformos.

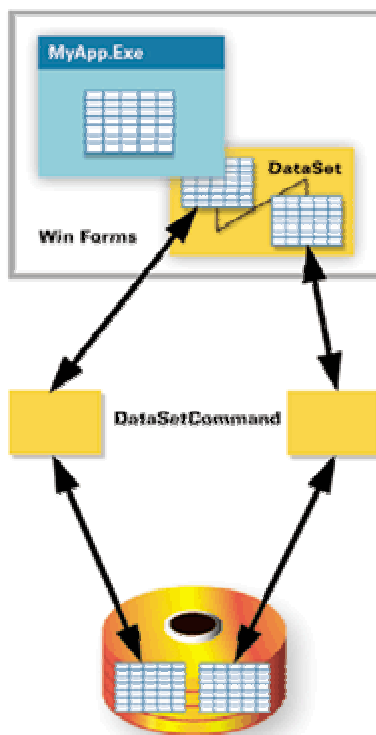
ADO.NET teikia daug privalumų kuriant verslo programas.

*ADO.NET architektūra* [2]:



### 3 pav. ADO.NET architektūra

ADO.NET gali dirbti per tinklo protokolus, naudojant XML (aukščiau diagramoje), arba labiau tradiciniu būdu – kliento/serverio architektūra:



4 pav. ADO.NET kliento/serverio architektūra

### 2.2.4. ADO.NET privalumai

Sąveika su kitomis aplinkomis (*Interoperability*)

Galimybė susisiekti per nevienalytes aplinkas.

Išplečiamumas (*Scalability*)

Galimybė aptarnauti didėjantį klientų skaičių, neįtakojant sistemos darbo.

Produktyvumas (*Productivity*)

Galimybė greitai patobulinti sudėtingas duomenų taikomasias programas naudojant ADO.NET komponentų objektinį modelį.

Efektyvumas (*Performance*)

Patobulinta ankstesnė ADO versija, duomenų serveris nepatirs „užsikimšimo“ (*bottleneck*) efekto, kai kreipinių kiekis padidėja.

### 2.2.5. Microsoft Visual C# .NET

Microsoft Visual C# .NET [5] yra visapusiškas įrankių rinkinys kuriant XML tinklo paslaugas ir Microsoft .NET – sujungtas programos tarp Microsoft Windows® ir tinklo.

Tai galingas kūrimo paketas, kuris naudoja komponentiškai orientuotą C# kūrimo kalbą, taip pat teikia pradedantiesiems ir pažengusiems kūrėjams (su C++ ar Java patirtimi) modernią kalbą ir aplinką, kuriant naujos kartos programas. Visual C# .NET 2003 teikia aukščiausią funkcionalumą srautiniams (*streamlining*) biznio procesams, įskaitant:

1. Modernus dizainas – plėtojimas ir išdėstymo palaikymas kuriant ar naudojant tinklo paslaugas (*Web services*);
2. Formų projektavimas ir vizualios kontrolės (*visual controls*) kuriant Windows programas;
3. Autorizavimo įrankiai ir paslaugos (*services*) kuriant galingus Microsoft .NET serverinius (*server-based*) sprendimus;
4. Perkėlimo įrankiai, kurie perkelia Java pagrindu darytus projektus į Microsoft .NET kūrimo aplinką.

Su Visual C# .NET kūrėjai gali sukurti sprendimus plačiausiam diapazonui klientų, įskaitant Windows tinklą, ir nešiojamus ar įtvirtintus (*embedded*) prietaisus. Naudojant šią aiškia programavimo kalbą ir įrankius, kūrėjai gali efektyviai perkelti savo esamus C++ ar Java kalbų įgūdžius ir žinias į .NET aplinką.

### 2.2.6. ASP.NET

ASP.NET yra programavimo rėmas, paremtas '*common language runtime*', kuris kuriant galingas interneto programas gali būti panaudotas serveriuose. ASP.NET yra tobulesnis negu ankstesni interneto puslapių kūrimo modeliai ir pasižymi tokiais pranašumais:

- Patobulintas veikimas – ASP.NET yra '*common language runtime*' kodas veikiantis serveryje, todėl tai leidžia *just-in-time* kompiliaciją, vietinę optimizaciją.
- *Word-Class* įrankių palaikymas – WYSIWYG redagavimas, *drag-and-drop* serverio valdymas ir automatinis išdėstymas.

- Galingumas ir lankstumas – kadangi ASP.NET yra paremta '*common language runtime*', todėl juo gali naudotis visa platforma.
- Paprastumas – su ASP.NET yra paprasta atlikti įprastas užduotis, nuo paprasto formų pateikimo ir kliento autorizavimo iki išdėstymo ir puslapio konfigūracijos.
- Lengvas valdymas – ASP.NET naudoja tekstu paremtą hierarchinę konfigūracijos sistemą, kuri palengvina ypatybių keitimą serverio aplinkoje ar interneto programose.
- Saugumas - su integruotu Windows autentiškumo patvirtinimu ir programos konfigūracija, programos yra saugios.

### 2.2.7. SQL Server 2000

SQL Server 2000 yra skirtas patenkinti informacijos laikymo reikalavimams didelėms informacijos apdorojimo sistemoms ir komerciniams interneto puslapiams, taip pat atitinka lengvo naudojimosi reikalavimus individualiems poreikiams ar smulkiam verslui.

SQL Server 2000 teikia dvi pagrindines paslaugas: Microsoft .NET platformai ir tradicinei dviejų pakopų kliento/serverio aplinkai. Pirma paslauga yra *SQL Server service*, kuri yra didelio galingumo (*high-performance*), daug keičiamos (*highly scalable*) reliacinės duomenų bazės variklis. Kita paslauga yra *SQL Server 2000 Analysis Services*, kuri teikia įrankius analizei duomenų, laikomų duomenų sandėliuose ir duomenų mugėse (*data marts*), sprendimams palaikyti.

## 2.3. Naudojamų programų apibendrinimas

1 lentelė. Naudojamos programos

Programinės priemonės pavadinimas	Paskirtis
Microsoft Visual Studio .NET	Tai yra kūrimo aplinka, kurią naudojant galima sukurti ir integruoti programas remiantis XML tinklo paslaugų ( <i>services</i> ) architektūra. Ji leidžia programoms dalintis duomenimis per internetą. XML tinklo paslaugos leidžia kūrėjams rašyti

	programas įvairiomis kalbomis arba naudoti jau esamas programas bei jas integruoti nepriklausomai nuo platformos, programavimo kalbos ar objekto modelio
<b>.NET Framework</b>	Apima CLR ( <i>Common Language Runtime</i> ) ir dažnai naudojamų klasių struktūras, kurias gali naudoti visos .NET kalbos
<b>Crystal Reports for .NET</b>	Tai ataskaitų kūrimo standartas Microsoft Visual Studio .NET programoms.
<b>ADO.NET (ActiveX Data Objects)</b>	Duomenų bazių prieigos ir duomenų apdorojimo programa.
<b>Microsoft Visual C# .NET</b>	Visapusiškas įrankių rinkinys XML tinklo paslaugų, .NET sujungtomis taikomosiomis programomis <i>windows</i> ir <i>web</i> formų kūrimui, naudojant komponentiškai - orientuota C# kūrimo kalbą.
<b>ASP.NET</b>	Naudojama kurti galingas dinamines interneto sistemas
<b>MS SQL Server 2000</b>	DBVS, skirta duomenų saugojimui ir apdorojimui

## 2.4. Analogiškų sistemų apibūdinimas

Lietuvoje nedaug viešbučių naudoja internetinio vietų rezervavimo sistemą, kuri yra integruota su vidine viešbučio informacine sistema. Daugelyje viešbučių, kuriuose yra įdiegta internetinio vietų rezervavimo sistema, ji yra labai paprasta, turi labai nedaug galimybių. Nėra galimybės apžiūrėti rezervuojamų patalpų, prastas duomenų saugumas. Tačiau egzistuoja Delfi viešbučių portalas (<http://hotels.delfi.lt>), kuris yra aukštos kokybės, labai informatyvus, patogus naudotis, kuo nepasižymi atskirų viešbučių rezervavimo sistemos.

## 2 lentelė. Panašios sistemos

<i>Delfi viešbučiai ir kaimo turizmas</i> ( <a href="http://hotels.delfi.lt">http://hotels.delfi.lt</a> )	Tai viešbučių portalas. Čia galima užsakyti viešbučių kambarius įvairiuose miestuose. Įdiegta viešbučių paieška. Detali informacija apie pasirinktą viešbutį: žemėlapis, nurodantis viešbučio buvimo vietą, aplinkinių vietovių aprašymas su paveikslukais, susisiekimas ir automobilių stovėjimo aikštelės, bendra informacija apie viešbutį, specialiomis piktogramomis pavaizduotos teikiamos paslaugos, paslaugos verslui taip pat parodytos specialomis piktogramomis, bendravimo kalbos – išvardinta kokiomis kalbomis galima bendrauti viešbutyje. Tačiau ši sistema nėra visiškai integruota su vidinėmis viešbučių sistemomis, negalima pažiūrėti kiek ir kokie kambariai laisvi, informacija apie kambarius yra teikiama tikrai tam tikrai grupei (vieno kambario, dviejų kambarų apartamentai), nėra informacijos apie atskirus kambarius.
<i>Viešbutis „Dvaras“</i> ( <a href="http://www.dvaras.lt">http://www.dvaras.lt</a> )	tai keturių žvaigždučių viešbutis, tačiau internetiniame puslapyje trūksta detalios informacijos apie rezervuojamus kambarius, nėra kambarių nuotraukų.
<i>Viešbutis „Congress“</i> ( <a href="http://www.congress.lt">http://www.congress.lt</a> )	šio viešbučio internetinis puslapis yra pakankamai informatyvus, galima pamatyti viešbučio fasado, vaizdo pro langą, registratūros, banketų, konferencijų galimybių nuotraukas.

### 2.4.1. Trumpas palyginimas JAVA(J2EE) ir .NET technologijų

(Platesnį aprašymą galima rasti: <http://www.neathosting.net/hartmann/netbenchmark.pdf>)

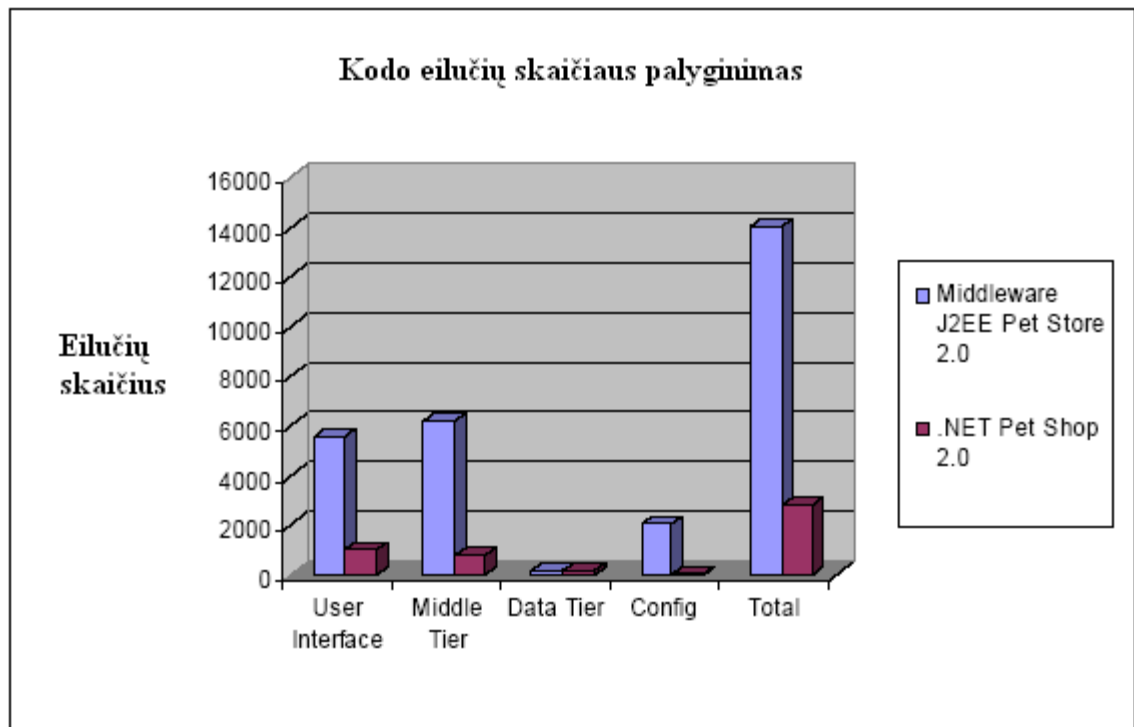
Kuriant šią sistemą yra akcentuojamas technologinis skirtumas nuo analoginių sistemų, todėl norint išvelgti skirtumą reikia palyginti .NET framework technologiją su kitomis panašiomis technologijomis. Alternatyvi .NET framework technologijai yra JAVA (J2EE), kurių palyginimas yra pateiktas (2002 Middleware Company – <http://www.middleware-company.com>).



### 2.4.1.1. Kodo eilučių kiekio palyginimas

3 lentelė. Kodo eilučių kiekio palyginimas

Elementai	.NET Pet Shop 2.0	Middleware J2EE Pet Store 2.0
User Interface	1,002	5,567
Middle Tier	795	6,187
Data Tier	197	197
Configuration	102	2,053
<b>Total</b>	<b>2,096</b>	<b>14,004</b>



5 pav. Kodo eilučių skaičiaus palyginimas

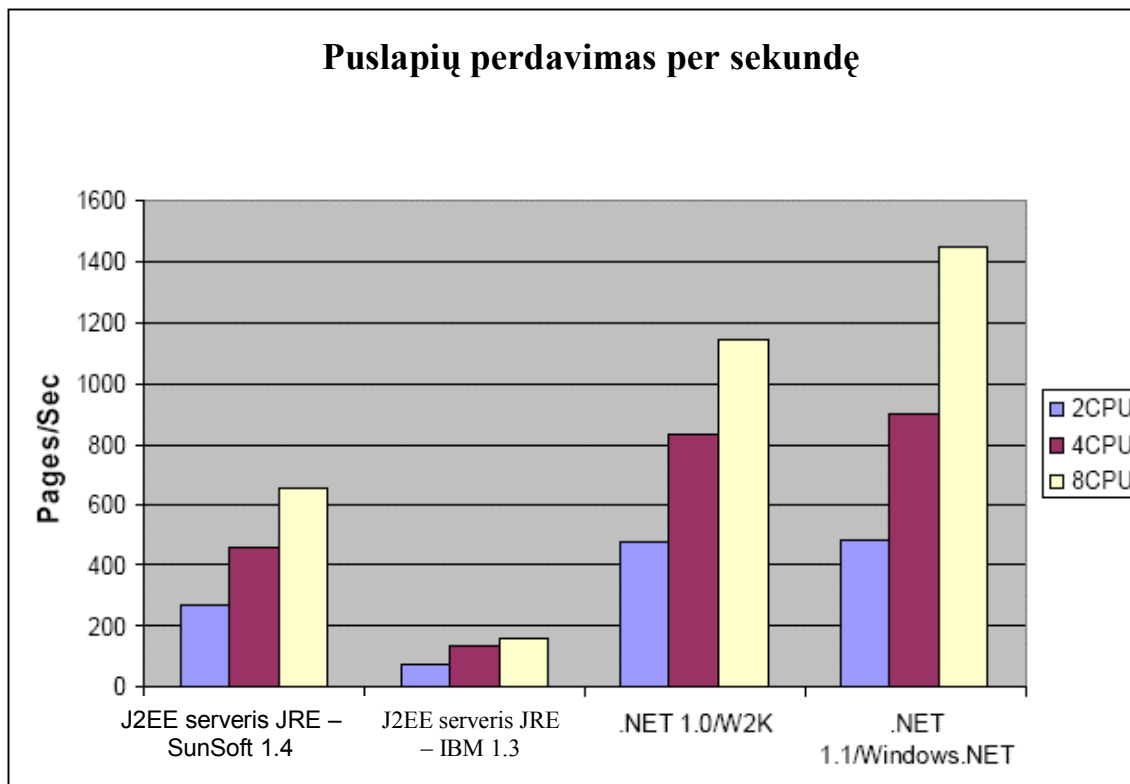
### 2.4.1.2. Transakcijos atsakymo laikas

4 lentelė Transakcijos atsakymo laikas

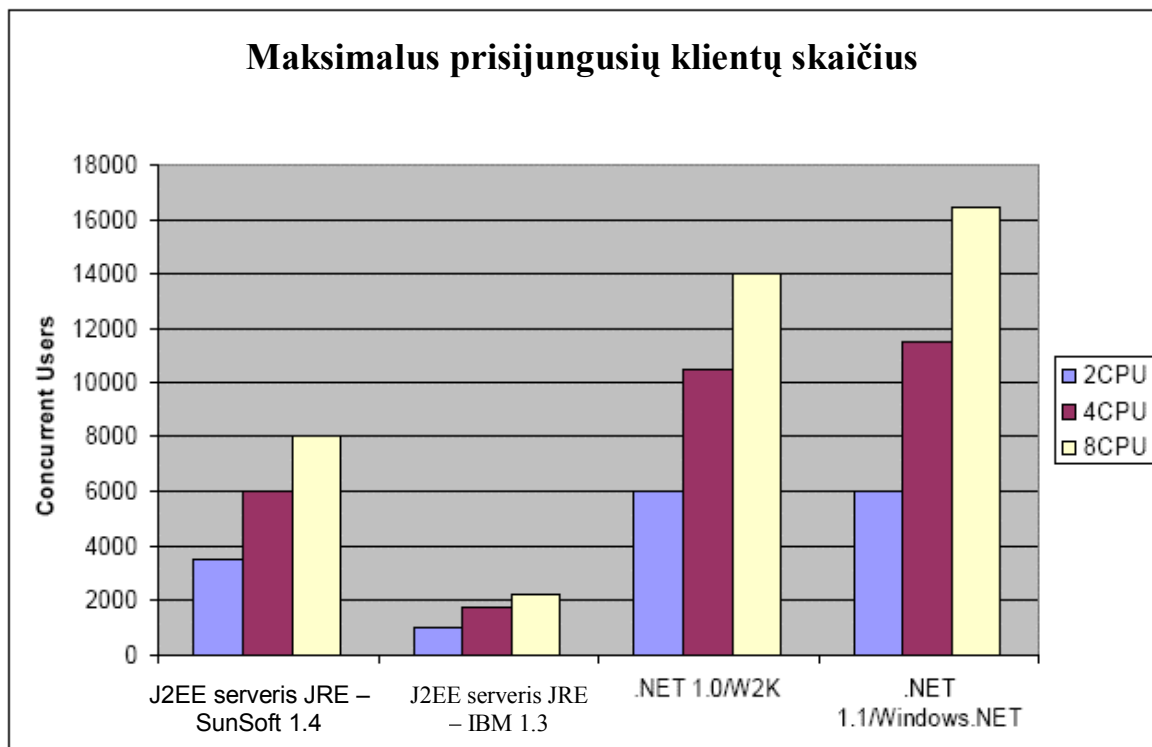
Vartotojų apkrovimas	J2EE serveris JRE – SunSoft 1.4	J2EE serveris JRE – IBM 1.3	.NET 1.0/ Windows 2000 Serveris	.NET 1.1/ Windows.NET Serveris
500	0.012	0.293	0.010	0.008
750	0.013	1.050	0.010	0.009
1000	0.014	3.025	0.010	0.010
2000	0.178	-	0.010	0.013

3000	0.970	-	0.012	0.016
4000	4.550	-	0.020	0.022
5000	-	-	1.237	0.391
6000	-	-	2.952	2.569

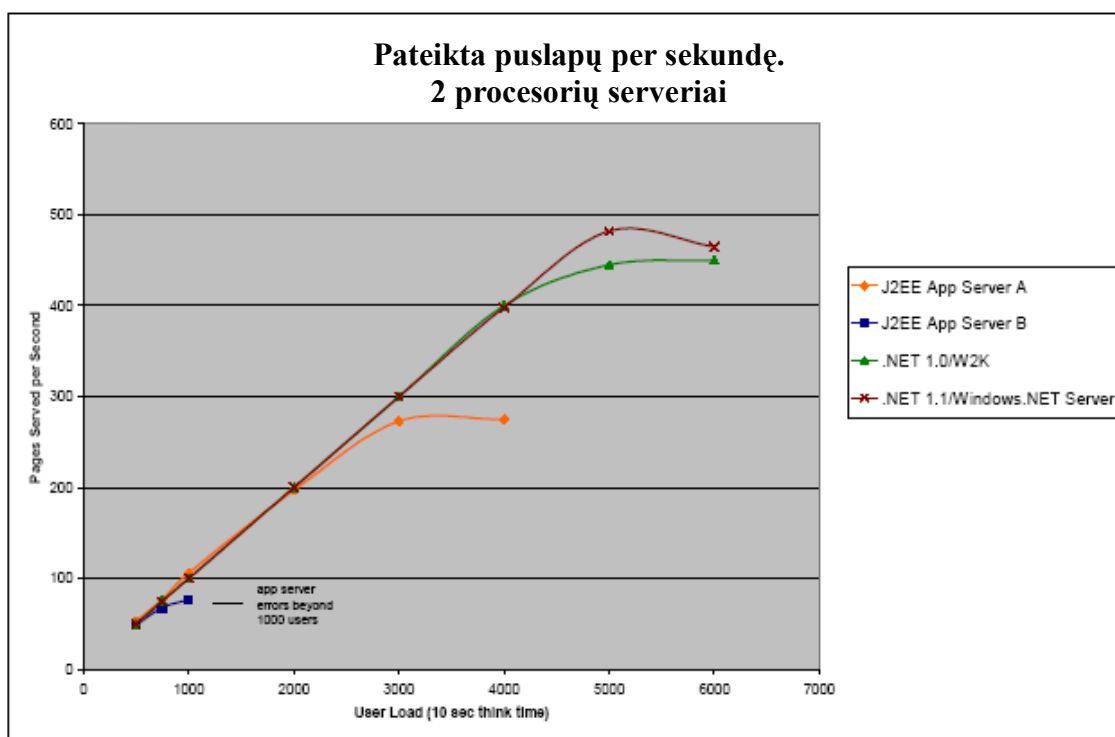
### 2.4.1.3. Interneto taikomosios programos sistemos darbo išbandymas



6 pav. Puslapių perdavimas per sekundę



7 pav. Maksimalus prisijungusių klientų skaičius



8 pav. Pateikta puslapų per sekundę, 2 procesorių serveriai

## **2.5. Bendri reikalavimai sistemai**

### **2.5.1. Produkto apibrėžimas**

Bus kuriama viešbučio rezervavimo sistema, kurios pagalba viešbučio klientai internetu galės apžiūrėti ir rezervuoti vietas norimai datai.

### **2.5.2. Vartotojų apibūdinimas**

Projektuojamos sistemos vartotojas turi turėti kompiuterio ir interneto naudojimo pagrindus. Sistemoje bus keturių tipų vartotojai:

Klientas

Sekretorė

Vadybininkas

Sistemos administratorius

### **2.5.3. Funkciniai reikalavimai sistemai**

Sistema turi leisti vykdyti tokias funkcijas:

1. Peržiūrėti informaciją apie kambarius
2. Užsakyti kambarį.
3. Patvirtinti užsakymą.
4. Rezervuoti konferencijų salę.
5. Patvirtinti konferencijų salę.
6. Atšaukti nepatvirtintą užsakymą.
7. Registruoti klientą.
8. Sukurti darbuotojo prisijungimo vardą.
9. Keisti kliento informaciją.
10. Keisti darbuotojo informaciją.
11. Keisti užsakymų informaciją.
12. Administruoti kambarių duomenis .
13. Administruoti apartamentų duomenis.
14. Filtruoti duomenis.
15. Generuoti ataskaitas.

## 2.5.4. Nefunkciniai reikalavimai sistemai

### 1) Reikalavimai grafinei vartotojo sąsajai. GUI – grafinė vartotojo sąsaja:

Programa turi būti padaryta atsižvelgiant į naujausias dizaino technologijas. Įvairiapusė įrankių panelė, “karšti klavišai” (*hot-key*), “pop-up” meniu bei spalvota informacija.

### 2) Reikalavimai diagnostikai (klaidų pranešimų aprašymui)

Jeigu funkcija vykdoma neteisingai turi išsiterpti aiškus pranešimas apie tai.

### 3) Reikalavimai sisteminei programinei ir techninei įrangai

#### 5 lentelė. Reikalavimai Visual Studio .NET 2003 [4] programai

Procesorius	450-megahertz (MHz) Pentium II-Klasės procesorius, 600-MHz Pentium III-klasės procesorius rekomenduojamas
Operacinė sistema	Visual Studio .NET 2003 gali būti instaliuota į vieną iš šių sistemų: <ul style="list-style-type: none"><li>• Microsoft Windows® Server 2003</li><li>• Windows XP Professional</li><li>• Windows XP Home Edition</li><li>• Windows 2000 Professional (reikia SP3)</li><li>• Windows 2000 Server (reikia bent SP3)</li></ul> Programos gali būti naudojamos šitose sistemose: <ul style="list-style-type: none"><li>• Windows Server 2003</li><li>• Windows XP Professional</li><li>• Windows XP Home Edition</li><li>• Windows 2000 (rekomenduojamas Service Pack 2)</li><li>• Windows Millennium Edition (Windows Me)</li><li>• Windows 98</li><li>• Microsoft Windows NT® 4.0 (reikalingas Service Pack 6a)</li></ul>

	Windows 95 (naudojant Visual C++® .NET)
Atmintis	<ul style="list-style-type: none"> <li>• Windows Server 2003: 160 Megabaitų (MB) RAM'o</li> <li>• Windows XP Professional: 160 MB RAM'o</li> <li>• Windows XP Home Edition: 96 MB RAM'o</li> <li>• Windows 2000 Professional: 96 MB RAM'o</li> </ul> <p>Windows 2000 Server: 192 MB RAM'o</p>
Kietas diskas	<ul style="list-style-type: none"> <li>• 900 MB sisteminiame diske, 3.3 gigabaitų (GB) laisvos vietos instaliuojamame diske.</li> </ul> <p>Papildomai 1.9 GB laisvos vietos nebūtinai MSDN bibliotekos dokumentacijai.</p>
Diskas	CD-ROM ar DVD-ROM diskas
Displėjus	Super VGA (1024 x 768) ar aukštesnė skiriamoji geba su 256 spalvomis
Pelė	Microsoft pelė ar suderinamas nurodantis įrenginys

**4) Reikalavimai programos funkcionavimui:** kompiuteris, kuriame veiktų .NET programos (žiūrėti aukščiau lentelėje) ir išėjimas į internetą.

**5) Reikalavimas klientui:** kompiuteris su interneto naršykle, kuri palaiko XML, ir išėjimas į internetą.

**6) Saugumas:** neteikti klientų informacijos neįgaliesiems asmenims.

**7) Reikalavimai IS palaikymui:** konsultacijų teikimas sistemos eksploatavimo metu.

**8) Pakartotinio panaudojimo reikalavimai:** programa turėtų būti tinkama kitiems viešbučio filialams arba kitiems viešbučiams.

### 3. VIESBUČIŲ INFORMACINĖS SISTEMOS MODELIS

#### 3.1. Formaliai specifikuojamos sistemos aprašymas

- Viešbučio rezervavimo sistema leidžia visiems vartotojams peržiūrėti laisvas vietas:
  - vieno,
  - dviejų,
  - ar trijų kambarių apartamentų.
- Vykdoma vartotojų registracija.
- Registruotų vartotojų prisijungimas.
- Vartotojas privalo būti užsiregistravęs ir prisijungęs, kad gautų:
  - rezervuoti,
  - peržiūrėti rezervuotus,
  - atšaukti rezervacijas,
- Rezervuoti galima tik laisvus:
  - vieno,
  - dviejų,
  - ar trijų kambarių apartamentus.
- Peržiūrėti rezervuotus apartamentus:
  - vieno,
  - dviejų,
  - ar trijų kambarių apartamentų.
- Atšaukti rezervaciją įvedant datą:
  - vieno,
  - dviejų,
  - ar trijų kambarių apartamentų.
- Suskaičiuoti rezervuotų apartamentų kainą.

### 3.2. Formalus aprašymas

Formalus aprašymas buvo realizuotas pasinaudojus „SUM“ programa ir Z formalus specifikavimo kalba. „SUM“ programa patikrina parašytą specifikaciją, todėl kuriant programą pagal šia specifikaciją yra išvengiama klaidų. Toks programų kūrimo metodas leidžia sutaupyti laiko ir pinigų, kurių reikia taisant programos klaidas ir atliekant testavimą.

#### *Viesbutis*

```
// Viesbutio rezervavimo sistemos specifikacija
// Atliko: Paulius Pauzas IFM 9/3
// vardu_aibe - visu galimu vardu aibe
// elektroninis_pastas - visu galimu elektroniniu pastu aibe
// login_aibe - visu galimu loginu aibe
// pass_aibe - visu galimu slaptazodziu aibe
// rez_data - visu galimu rezervavimo datu aibe
[vardu_aibe, elektroninis_pastas, login_aibe, pass_aibe, rez_data]

// apibreziamos konstantos:
// kamb_1_kiekis - isviso viesbutyje esanciu vieno kambario nuomojama apartamentu;
// kamb_2_kiekis - isviso viesbutyje esanciu dviejų kambariu nuomojama apartamentu;
// kamb_3_kiekis - isviso viesbutyje esanciu triju kambariu nuomojama apartamentu;
// kaina_1_kambario - vieno kambario apartamento kaina;
// kaina_2_kambario - dviejų kambariu apartamento kaina;
// kaina_3_kambario - triju kambariu apartamento kaina.
kamb_1_kiekis:
kamb_2_kiekis:
kamb_3_kiekis:
kaina_1_kambario:
kaina_2_kambariu:
kaina_3_kambariu:

// sukuriamos grupes:
// vard - loginas su vardu
// email - loginas su emailu
// kamb_1_data - vieno kambario apartamentu rezervavimo data su loginu
// kamb_2_data - dviejų kambariu apartamentu rezervavimo data su loginu
// kamb_3_data - triju kambariu apartamentu rezervavimo data su loginu
// login_temp - cia bus laikinai saugomas loginas prisijungus, o atsijungus istrinamas.
state
  vard: login_aibe  vardu_aibe
  login: login_aibe  pass_aibe
  email: login_aibe  elektroninis_pastas
  kamb_1_data: rez_data  login_aibe
  kamb_2_data: rez_data  login_aibe
  kamb_3_data: rez_data  login_aibe
  login_temp:      login_aibe

  dom vard  dom login
```



```
dom login  dom email
dom email  dom vard
```

```
// priskiriamos pradines reiksmes
init
```

```
  vard'
  login'
  email'
  kamb_1_data'
  kamb_2_data'
  kamb_3_data'
  kamb_1_kiekis 20
  kamb_2_kiekis 15
  kamb_3_kiekis 10
  kaina_1_kambario 60
  kaina_2_kambariu 150
  kaina_3_kambariu 300
  login_temp'
```

```
// zinute, kuri bus isvedama prisijungus
```

```
pavyko
```

```
pr!: string
```

```
pr!  "sekmingai prisijungta"
```

```
// zinute, kuri bus isvedama nepavykus prisijungti
```

```
nepavyko
```

```
pr!: string
```

```
pr!  "klaida prisijungiant"
```

```
// teisingas prisijungimas prie sistemos,
```

```
// ivedamas loginas ir slaptazodis turi sutapti su login aibeje esanciu logino ir passwordo
pora,
```

```
// loginas laikinai irasomas i login_temp
```

```
PrisPavyko
```

```
state
```

```
state'
```

```
l?: login_aibe
```

```
p?: pass_aibe
```

```
pavyko
```

```
p?  ran( l?  login)
```

```
login_temp' l?
```

```
// neteisingas prisijungimas,  
// /ivedamas loginas ir slaptazodis nesutampa su login aibeje esanciu logino ir passwordo  
pora,
```

```
PrisNepavyko
```

```
state
```

```
state'
```

```
l?: login_aibe
```

```
p?: pass_aibe
```

```
nepavyko
```

```
p? ran ( l? login)
```

```
login_temp'
```

```
Prisijungimas (PrisPavyko PrisNepavyko)
```

```
// operacija loginas, kuria sudaro pavykias arba nepavykes prisijungimas
```

```
op loginas
```

```
Prisijungimas
```

```
// teisingo prisiregistravimo zinute
```

```
pavykoPrisiregistruoti
```

```
pr!: string
```

```
pr! "sekmingai prisiregistruota"
```

```
// neteisingo prisiregistravimo zinute
```

```
nepavykoPrisiregistruoti
```

```
pr!: string
```

```
pr! "klaida prisiregistruojant"
```

```
// teisingas registravimas
```

```
RegPavyko
```

```
state
```

```
state'
```

```
v?: vardu_aibe
```

```
l?: login_aibe
```

```
p?: pass_aibe
```

```
e?: elektroninis_pastas
```

```
pavykoPrisiregistruoti
```

```
l? dom login e? ran email
```

```
vard' vard (l?, v?)
```

```
login' login (l?, p?)
```

```
email' email (l?, e?)
```

```
// neteisingas registravimas
```

```
RegNePavyko
```

```
state
state'
v?: vardu_aibe
l?: login_aibe
p?: pass_aibe
e?: elektroninis_pastas
nepavykoPrisiregistruoti

l? dom login e? ran email
```

*Registravimas (RegPavyko RegNePavyko)*

```
op registr
Registravimas
```

```
// sekmingo rezervavimo zinute
```

```
pavyko_uzsakyti
pr!: string
```

```
pr! "sekmingai uzrezervuota"
```

```
// nesekmingo rezervavimo zinute. kai nera vietu
```

```
ne_pavyko_uzsakyti
pr!: string
```

```
pr! "nera vietu"
```

```
// sekmingo rezervavimo zinute, kai vartotojas neprisijunges
```

```
ne_pavyko_uzsakyti2
pr!: string
```

```
pr! "rezervuoti galima tik prisijungus"
```

```
// sekmingas 1 kambario apartamento rezervavimas
```

```
kambario_1_Uzsakymas
```

```
state
state'
d?: rez_data
pavyko_uzsakyti
```

```
⌈ (login_temp      )
login_temp  dom login
kamb_1_kiekis  Ÿ ( d?  kamb_1_data)
kamb_1_data'  kamb_1_data  (d?, login_temp)
```

```
// nesekmingas 1 kambario apartamento rezervavimas
kambario_1_Uzsakymas_nepavyko
state
state'
d?: rez_data
ne_pavyko_uzsakyti

T̄ (login_temp      )
login_temp  dom login
kamb_1_kiekis  Ÿ ( d?   kamb_1_data)
```

```
// nesekmingas 1 kambario apartamento rezervavimas
kambario_1_Uzsakymas_nepavyko2
state
state'
ne_pavyko_uzsakyti2

login_temp
```

```
Rezervavimas_1_kambario (kambario_1_Uzsakymas
kambario_1_Uzsakymas_nepavyko kambario_1_Uzsakymas_nepavyko2)
op rezerv1
Rezervavimas_1_kambario
```

```
// sekmingas 2 kambario apartamento rezervavimas
kambariu_2_Uzsakymas
state
state'
d?: rez_data
pavyko_uzsakyti

T̄ (login_temp      )
login_temp  dom login
kamb_2_kiekis  Ÿ ( d?   kamb_2_data)
kamb_2_data'  kamb_2_data  (d?, login_temp)
```

```
// nesekmingas 2 kambario apartamento rezervavimas
kambariu_2_Uzsakymas_nepavyko
state
state'
d?: rez_data
ne_pavyko_uzsakyti

T̄ (login_temp      )
login_temp  dom login
kamb_2_kiekis  Ÿ ( d?   kamb_2_data)
```

```

// nesekmingas 2 kambario apartamento rezervavimas
kambariu_2_Uzsakymas_nepavyko2
state
state'
ne_pavyko_uzsakyti2

login_temp

Rezervavimas_2_kambariu (kambariu_2_Uzsakymas
kambariu_2_Uzsakymas_nepavyko kambariu_2_Uzsakymas_nepavyko2)
op rezerv2
Rezervavimas_2_kambariu

// sekmingas 3 kambario apartamento rezervavimas
kambario_3_Uzsakymas
state
state'
d?: rez_data
pavyko_uzsakyti

T (login_temp )
login_temp dom login
kamb_3_kiekis Y ( d? kamb_3_data)
kamb_3_data' kamb_3_data (d?, login_temp)

// nesekmingas 3 kambario apartamento rezervavimas
kambario_3_Uzsakymas_nepavyko
state
state'
d?: rez_data
ne_pavyko_uzsakyti

T (login_temp )
login_temp dom login
kamb_3_kiekis Y ( d? kamb_3_data)

// nesekmingas 3 kambario apartamento rezervavimas
kambario_3_Uzsakymas_nepavyko2
state
state'
ne_pavyko_uzsakyti2

login_temp

Rezervavimas_3_kambario (kambario_3_Uzsakymas
kambario_3_Uzsakymas_nepavyko kambario_3_Uzsakymas_nepavyko)

```

```

op rezerv3
  Rezervavimas_3_kambario

// atsijungus istrinama is login_temp prisijungusiojo loginas
op log_out
  login_temp'

// ivedus data parodo laisvus 1 kambario apartamentus
op isvesti_1_kambario_laisvus
  d?: rez_data
  z!:

  z! kamb_1_kiekis Ė Ę ( d? kamb_1_data)

// ivedus data parodo laisvus 2 kambario apartamentus
op isvesti_2_kambariu_laisvus
  d?: rez_data
  z!:

  z! kamb_2_kiekis Ė Ę ( d? kamb_2_data)

// ivedus data parodo laisvus 3 kambario apartamentus
op isvesti_3_kambariu_laisvus
  d?: rez_data
  z!:

  z! kamb_3_kiekis Ė Ę ( d? kamb_3_data)

// zinutes, kurios bus isvedamos nepavykus perziureti rezervuotu kambariu.
ne_pavyko_perziureti
  pr!: string

  pr! "Jus nerezervavote "

ne_pavyko_perziureti2
  pr!: string

  pr! "perziuretu rezervyitus kambarius galima tik prisijungus"

// sekmingas perziurejimas jau rezervuotu 1 kambario patalpu,
perziureti_1_kambario_uzsakytus
  state
  state'
  pu!: rez_data

  T (login_temp      )

```

```
login_temp ran kamb_1_data
pu! dom (kamb_1_data login_temp )
```

```
// nesekmingas perziurejimas jau rezervuotu 1 kambario patalpu, kai nera prisiregistraves
uzrezervaves 1 kambario apartamentu
```

```
perziureti_1_kambario_uzsakytus_nepavyko
state
state'
ne_pavyko_perziureti
```

```
T (login_temp )
login_temp ran kamb_1_data
```

```
// nesekmingas perziurejimas jau rezervuotu 1 kambario patalpu, kai vartotojas
neprisiregistraves
```

```
perziureti_1_kambario_uzsakytus_nepavyko2
state
state'
ne_pavyko_perziureti2
```

```
login_temp
```

```
Perziureti_1_kambario_rezervuotus (perziureti_1_kambario_uzsakytus
perziureti_1_kambario_uzsakytus_nepavyko perziureti_1_kambario_uzsakytus_nepavyko2)
op perziurejimas1
Perziureti_1_kambario_rezervuotus
```

```
// sėkmingas perziurejimas jau rezervuotu 2 kambariu patalpu,
```

```
perziureti_2_kambariu_uzsakytus
state
state'
pu!: rez_data
```

```
T (login_temp )
login_temp ran kamb_2_data
pu! dom (kamb_2_data login_temp )
```

```
// nesekmingas perziurejimas jau rezervuotu 2 kambario patalpu, kai nera prisiregistraves
uzrezervaves 2 kambario apartamentu
```

```
perziureti_2_kambariu_uzsakytus_nepavyko
state
state'
ne_pavyko_perziureti
```

```
T (login_temp )
login_temp ran kamb_2_data
```

```

// nesekmingas perziurejimas jau rezervuotu 2 kambario patalpu, kai vartotojas nera
prisiregistruotes
    perziureti_2_kambariu_uzsakytus_nepavyko2
        state
        state'
        ne_pavyko_perziureti2

        login_temp

    Perziureti_2_kambariu_rezervuotus                (perziureti_2_kambariu_uzsakytus
perziureti_2_kambariu_uzsakytus_nepavyko    perziureti_2_kambariu_uzsakytus_nepavyko2)
// nesekmingas perziurejimas jau rezervuotu 2 kambario patalpu, kai nera prisiregistruotes
uzrezervaves 2 kambario apartamentu
    op perziurejimas2
        Perziureti_2_kambariu_rezervuotus

// perziurejimas jau rezervuotu 3 kambariu patalpu,
// sekmingas perziurejimas jau rezervuotu 3 kambariu patalpu.
    perziureti_3_kambariu_uzsakytus
        state
        state'
        pu!: rez_data

        T (login_temp          )
        login_temp    ran kamb_3_data
        pu!    dom (kamb_3_data    login_temp )

// nesekmingas perziurejimas jau rezervuotu 3 kambario patalpu, kai nera prisiregistruotes
uzrezervaves 3 kambario apartamentu
    perziureti_3_kambariu_uzsakytus_nepavyko
        state
        state'
        ne_pavyko_perziureti

        T (login_temp          )
        login_temp    ran kamb_3_data

// nesekmingas perziurejimas jau rezervuotu 3 kambario patalpu, kai vartotojas nera
prisiregistruotes
    perziureti_3_kambariu_uzsakytus_nepavyko2
        state
        state'
        ne_pavyko_perziureti2

        login_temp

```



```

Perziureti_3_kambariu_rezervuotus          (perziureti_3_kambariu_uzsakytus
perziureti_3_kambariu_uzsakytus_nepavyko  perziureti_3_kambariu_uzsakytus_nepavyko2)
op perziurejimas3
Perziureti_3_kambariu_rezervuotus

// zinutes, kurios bus isvedamos atsakant rezervuotus kambarius.
pavyko_atsakyti
pr!: string

pr!  "Sekmingai atsakyta rezervacija nurodytu laiku "

ne_pavyko_atsakyti
pr!: string

pr!  "Klaida, nera uzrezervuota siuo laiku. "

ne_pavyko_atsakyti2
pr!: string

pr!  "atsakyti rezervyitus kambarius galima tik prisijungus"

// sekmingai atsakyta jau rezervuotus 1 kabario patalpas, ivedant data
atsakyti_1_kambario
state
state'
d?: rez_data
pavyko_atsakyti

T̄ (login_temp      )
(d?, login_temp)  kamb_1_data
kamb_1_data'  kamb_1_data  (d?, login_temp)

// nesekmingai atsakyta jau rezervuotus 1 kabario patalpas, ivedant data, nera tokios datos
atsakyti_1_kambario_nepavyko
state
state'
d?: rez_data
ne_pavyko_atsakyti

T̄ (login_temp      )
(d?, login_temp)  kamb_1_data

// nesekmingai atsakyta jau rezervuotus 1 kabario patalpas, ivedant data, vartotojas
neprisijunges
atsakyti_1_kambario_nepavyko2
state

```

```

state'
d?: rez_data
ne_pavyko_atsakyti2

login_temp

Atsakyti_1_kambariu_rezervuotus (atsakyti_1_kambario
atsakyti_1_kambario_nepavyko atsakyti_1_kambario_nepavyko2)
op Atsakymas1
Atsakyti_1_kambariu_rezervuotus

// sekmingai atsakyta jau rezervuotus 2 kabariu patalpas, ivedant data
atsakyti_2_kambariu
state
state'
d?: rez_data
pavyko_atsakyti

T̄ (login_temp )
(d?, login_temp) kamb_2_data
kamb_2_data' kamb_2_data (d?, login_temp)

// nesekmingai atsakyta jau rezervuotus 2 kabariu patalpas, ivedant data, nera tokios datos
atsakyti_2_kambariu_nepavyko
state
state'
d?: rez_data
ne_pavyko_atsakyti

T̄ (login_temp )
(d?, login_temp) kamb_2_data

// nesekmingai atsakyta jau rezervuotus 2 kabariu patalpas, ivedant data, vartotojas
neprisijunges
atsakyti_2_kambariu_nepavyko2
state
state'
d?: rez_data
ne_pavyko_atsakyti2

login_temp

Atsakyti_2_kambariu_rezervuotus (atsakyti_2_kambariu
atsakyti_2_kambariu_nepavyko atsakyti_2_kambariu_nepavyko2)
op Atsakymas2
Atsakyti_2_kambariu_rezervuotus

```

```

// sėkmingai atsakyta jau rezervuotus 3 kambarių patalpas, įvedant data
atsakyti_3_kambariu
state
state'
d?: rez_data
pavyko_atstasyti

T̄ (login_temp      )
(d?, login_temp)  kamb_3_data
kamb_3_data'  kamb_3_data  (d?, login_temp)

// nesėkmingai atsakyta jau rezervuotus 3 kambarių patalpas, įvedant data, nėra tokios datos
atsakyti_3_kambariu_nepavyko
state
state'
d?: rez_data
ne_pavyko_atstasyti

T̄ (login_temp      )
(d?, login_temp)  kamb_3_data

// sėkmingai atsakyta jau rezervuotus 3 kambarių patalpas, įvedant data, vartotojas
neprisijungęs
atsakyti_3_kambariu_nepavyko2
state
state'
d?: rez_data
ne_pavyko_atstasyti2

login_temp

Atstasyti_3_kambariu_rezervuotus      (atsakyti_3_kambariu
atsakyti_3_kambariu_nepavyko  atsakyti_3_kambariu_nepavyko2)
op Atsakymas3
Atstasyti_3_kambariu_rezervuotus

// žinutes, kurios bus išvedamos nepavykus išvesti kainos
ne_pavyko_kaina
pr!: string

pr!  "Jūs nerezervavote "

ne_pavyko_kaina2
pr!: string

pr!  "peržiūrėtu rezervytais kambariais galima tik prisijungus"

```

```

// sėkmingas kainos išvedimas
kaina
state
state'
k!:

 $\bar{T}$  (login_temp )
(login_temp ran kamb_1_data) (login_temp ran kamb_1_data) (login_temp
ran kamb_1_data)
k! (( $\bar{Y}$  (kamb_1_data login_temp ))  $\oslash$  kaina_1_kambario)  $\cup$  (( $\bar{Y}$  (kamb_2_data
login_temp ))  $\oslash$  kaina_2_kambariu)  $\cup$  (( $\bar{Y}$  (kamb_3_data login_temp ))  $\oslash$ 
kaina_3_kambariu)

// nesėkmingas kainos išvedimas, nes nėra nei vieno rezervuoto apartamento
kaina_nepavyko
state
state'
ne_pavyko_kaina

 $\bar{T}$  (login_temp )
(login_temp ran kamb_1_data) (login_temp ran kamb_1_data) (login_temp
ran kamb_1_data)

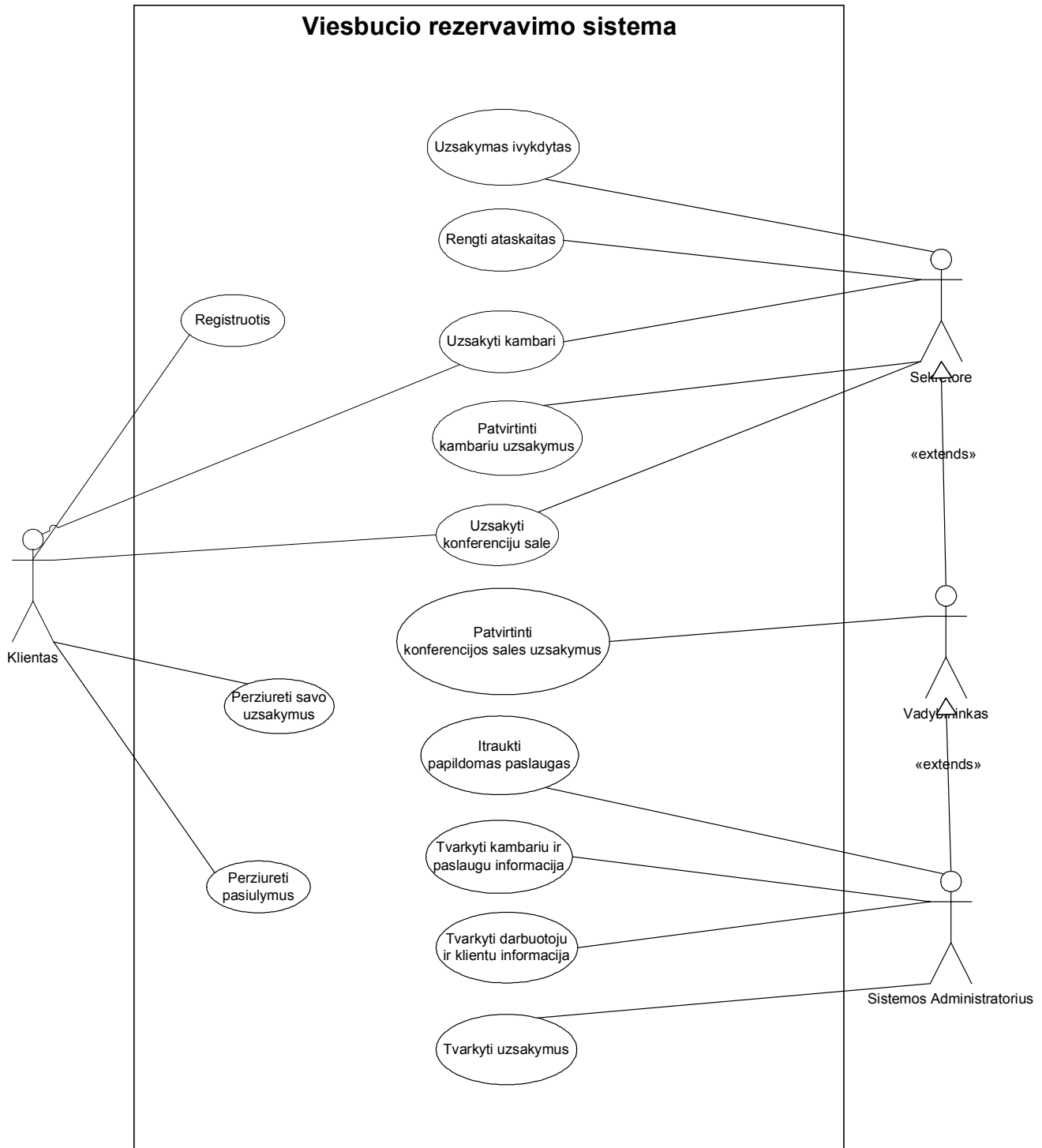
// nesėkmingas kainos išvedimas, nes neprisijunges
kaina_nepavyko2
state
state'
ne_pavyko_kaina2

login_temp

Isvesti_kaina (kaina kaina_nepavyko kaina_nepavyko2)
op is_kaina
Isvesti_kaina

```

### 3.3. Viešbučio rezervavimo sistemos taikomųjų uždavinių modelis



9 pav. Taikomųjų uždavinių modelis

Sistemos vartotojai:

16. Klientas.
17. Sekretorė.
18. Vadybininkas.
19. Sistemos administratorius

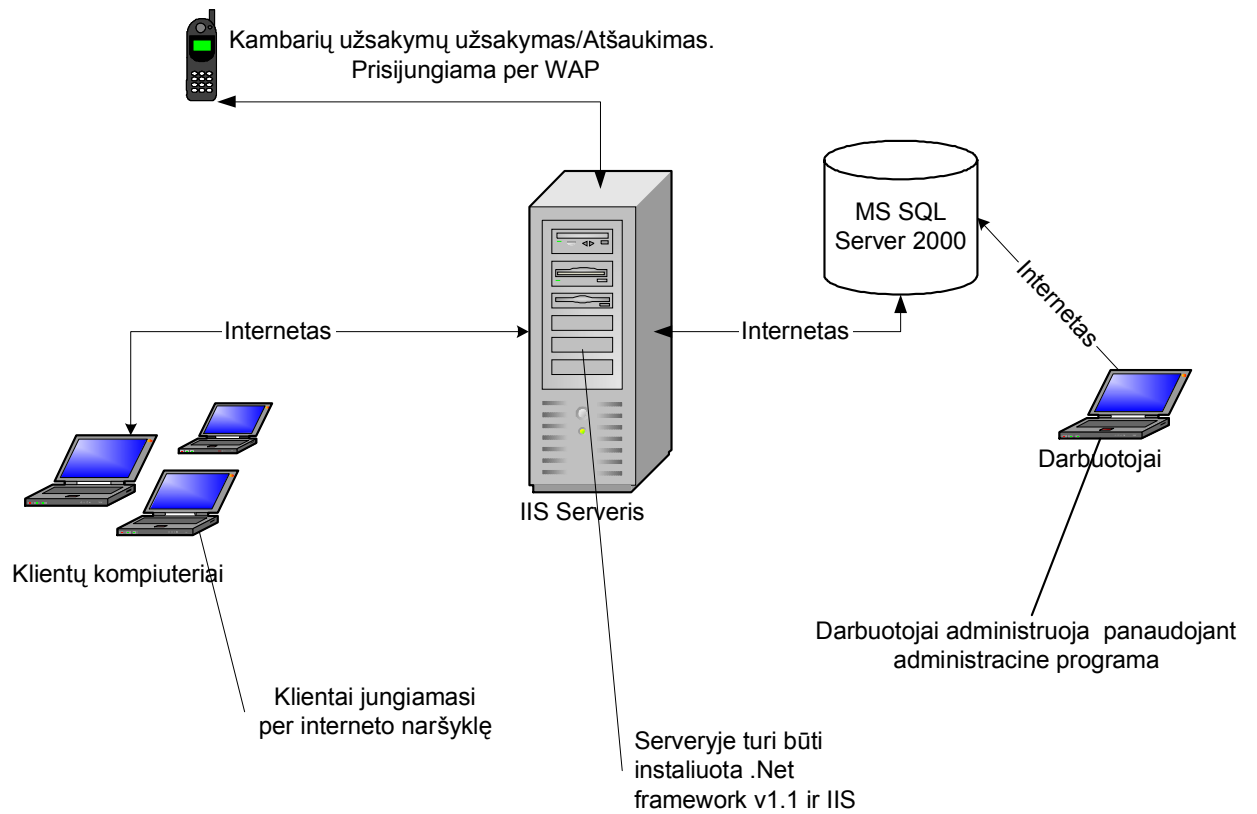
Atliekami veiksmai:

20. Registruotis
21. Peržiūrėti savo užsakymus
22. Peržiūrėti pasiūlymus
23. Užsakymas įvykdytas
24. Rengti ataskaitas
25. Užsakyti kambarį
26. Patvirtinti kambarių užsakymus
27. Užsakyti konferencijų salę
28. Patvirtinti konferencijos salės užsakymus
29. Įtraukti papildomas paslaugas
30. Tvarkyti kambarių ir paslaugų informaciją
31. Tvarkyti darbuotojų ir klientų informaciją
32. Tvarkyti užsakymus

Analizuojamos srities veiklos ribos:

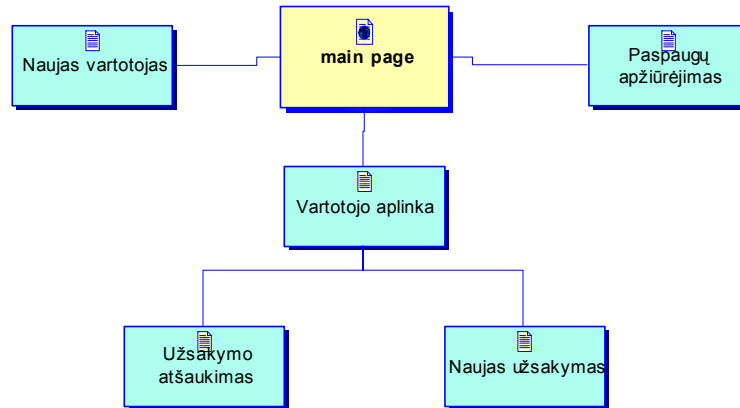
33. Viešbučio rezervavimo sistema

### 3.4. Sistemos architektūra



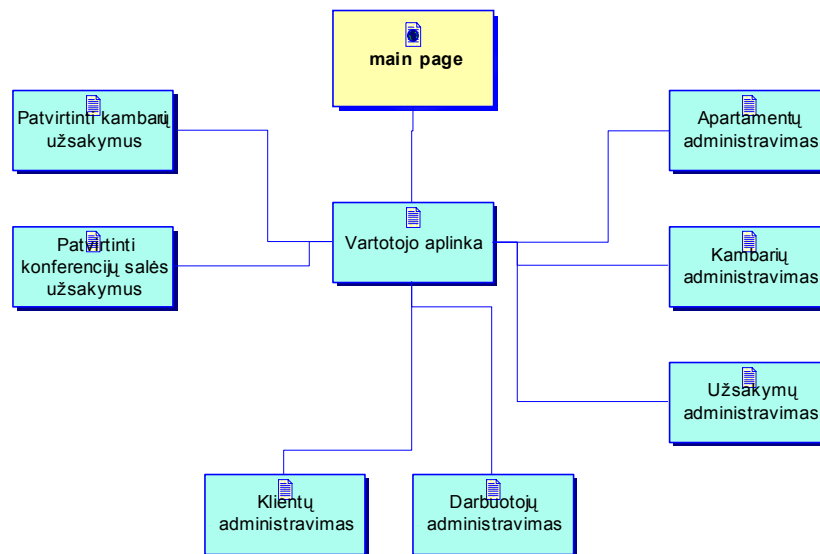
10 pav. Sistemos architektūra

### 3.5. Kliento aplinkos „žemėlapis”



11 pav. Kliento aplinkos “žemėlapis”

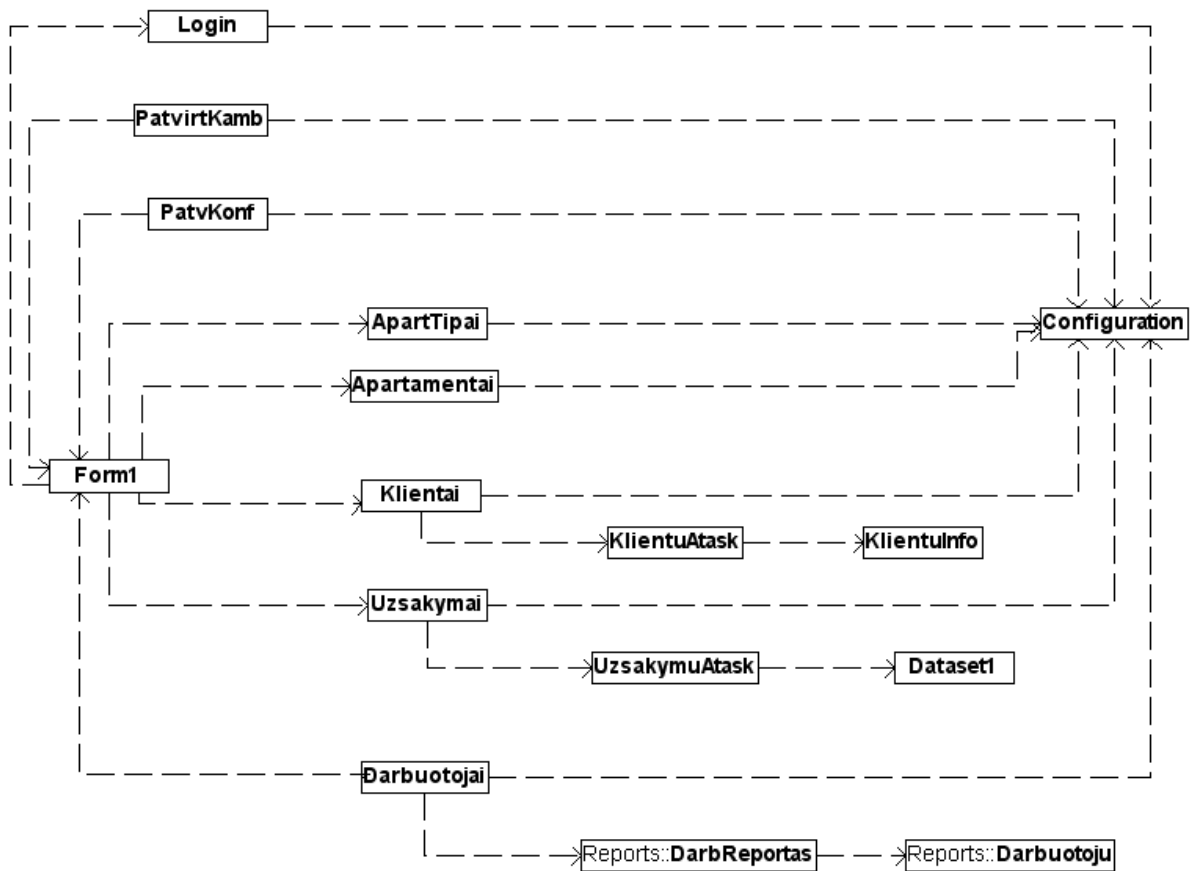
### 3.6. Darbuotojų aplinkos „žemėlapis”



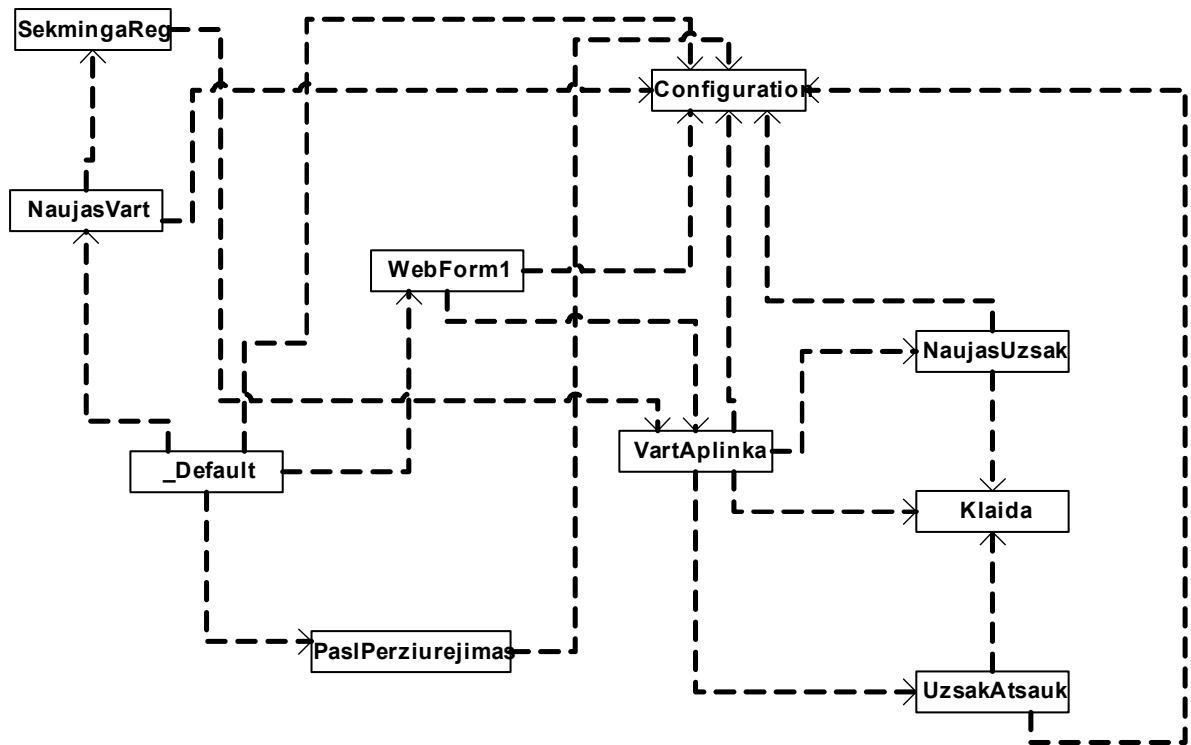
12 pav. Darbuotojų aplinkos “žemėlapis”



### 3.7. Viešbučių sistemos klasių diagrama



13 pav. Viešbučių sistemos administravimo programos klasių diagrama



14 pav. Viešbučių sistemos rezervavimo programos klasių diagrama

### 3.8. Viešbučių sistemos klasių aprašymas

6 lentelė. Viešbučių sistemos klasių aprašymas

Klasė	Trumpas aprašymas
<b>Administravimo programos klasių diagrama</b>	
<div style="border: 1px solid black; padding: 5px;"> <p style="text-align: center;"><b>Login</b></p> <pre> + _login : string = "" + _lygis : int = -1 -label1 : Label -label2 : Label -simpleButtonPL1 : SimpleButtonPL -textEditPL_Login : TextEditPL -textEditPL_Pass : TextEditPL -label_Klaida : Label -components : Container = null +Login() #Dispose(in disposing : bool) -InitializeComponent() -simpleButtonPL1_Click(in sender : object, in e : EventArgs)           </pre> </div>	<p>Pagrindinė klasė (24 pav.) iš kurios yra kviečiami kiti langai.</p>

Klasė	Trumpas aprašymas		
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center;">Form1</th> </tr> </thead> <tbody> <tr> <td> <pre> + _login : string = "" + _lygis : int = -1 + _vardas : string = "" -barManager1 : BarManager -barDockControlTop : BarDockControl -barDockControlBottom : BarDockControl -barDockControlLeft : BarDockControl -barDockControlRight : BarDockControl -dockManager1 : DockManager -dockPanel1_Container : ControlContainer -dockPanel2_Container : ControlContainer -hideContainerLeft : AutoHideContainer -dockPanel3_Container : ControlContainer -dockPanel_Vadybininkas : DockPanel -dockPanel_Admin : DockPanel -dockPanel_Sekretore : DockPanel -simpleButtonPL_PatKamb : SimpleButtonPL -simpleButtonPL_UzsAdmin : SimpleButtonPL -simpleButtonPL_KonfPatv : SimpleButtonPL -simpleButtonPL_KlientAdmin : SimpleButtonPL -simpleButtonPL_ApartAdmin : SimpleButtonPL -simpleButtonPL_DarbAdmin : SimpleButtonPL -bar1 : Bar -barButtonItem1 : BarButtonItem -simpleButtonPL_ApTipai : SimpleButtonPL -barEditItem1 : BarEditItem -repositoryItemTextEdit1 : RepositoryItemTextEdit -barAndDockingController1 : BarAndDockingController -components : IContainer  +Form1() #Dispose(in disposing : bool) -InitializeComponent() -Main() -Form1_Load(in sender : object, in e : EventArgs) -simpleButtonPL_PatKamb_Click(in sender : object, in e : EventArgs) -simpleButtonPL_KonfPatv_Click(in sender : object, in e : EventArgs) -simpleButtonPL_UzsAdmin_Click(in sender : object, in e : EventArgs) -barButtonItem1_ItemClick(in sender : object, in e : ItemClickEventArgs) -simpleButtonPL_KlientAdmin_Click(in sender : object, in e : EventArgs) -simpleButtonPL_DarbAdmin_Click(in sender : object, in e : EventArgs) -simpleButtonPL_ApartAdmin_Click(in sender : object, in e : EventArgs) -simpleButtonPL1_Click_1(in sender : object, in e : EventArgs) </pre> </td> </tr> </tbody> </table>	Form1	<pre> + _login : string = "" + _lygis : int = -1 + _vardas : string = "" -barManager1 : BarManager -barDockControlTop : BarDockControl -barDockControlBottom : BarDockControl -barDockControlLeft : BarDockControl -barDockControlRight : BarDockControl -dockManager1 : DockManager -dockPanel1_Container : ControlContainer -dockPanel2_Container : ControlContainer -hideContainerLeft : AutoHideContainer -dockPanel3_Container : ControlContainer -dockPanel_Vadybininkas : DockPanel -dockPanel_Admin : DockPanel -dockPanel_Sekretore : DockPanel -simpleButtonPL_PatKamb : SimpleButtonPL -simpleButtonPL_UzsAdmin : SimpleButtonPL -simpleButtonPL_KonfPatv : SimpleButtonPL -simpleButtonPL_KlientAdmin : SimpleButtonPL -simpleButtonPL_ApartAdmin : SimpleButtonPL -simpleButtonPL_DarbAdmin : SimpleButtonPL -bar1 : Bar -barButtonItem1 : BarButtonItem -simpleButtonPL_ApTipai : SimpleButtonPL -barEditItem1 : BarEditItem -repositoryItemTextEdit1 : RepositoryItemTextEdit -barAndDockingController1 : BarAndDockingController -components : IContainer  +Form1() #Dispose(in disposing : bool) -InitializeComponent() -Main() -Form1_Load(in sender : object, in e : EventArgs) -simpleButtonPL_PatKamb_Click(in sender : object, in e : EventArgs) -simpleButtonPL_KonfPatv_Click(in sender : object, in e : EventArgs) -simpleButtonPL_UzsAdmin_Click(in sender : object, in e : EventArgs) -barButtonItem1_ItemClick(in sender : object, in e : ItemClickEventArgs) -simpleButtonPL_KlientAdmin_Click(in sender : object, in e : EventArgs) -simpleButtonPL_DarbAdmin_Click(in sender : object, in e : EventArgs) -simpleButtonPL_ApartAdmin_Click(in sender : object, in e : EventArgs) -simpleButtonPL1_Click_1(in sender : object, in e : EventArgs) </pre>	<p>Prisijungimo langas (23 pav.). Įvedus teisingą prisijungimo vardą ir slaptažodį patenkama į Form1 pagrindinę formą.</p>
Form1			
<pre> + _login : string = "" + _lygis : int = -1 + _vardas : string = "" -barManager1 : BarManager -barDockControlTop : BarDockControl -barDockControlBottom : BarDockControl -barDockControlLeft : BarDockControl -barDockControlRight : BarDockControl -dockManager1 : DockManager -dockPanel1_Container : ControlContainer -dockPanel2_Container : ControlContainer -hideContainerLeft : AutoHideContainer -dockPanel3_Container : ControlContainer -dockPanel_Vadybininkas : DockPanel -dockPanel_Admin : DockPanel -dockPanel_Sekretore : DockPanel -simpleButtonPL_PatKamb : SimpleButtonPL -simpleButtonPL_UzsAdmin : SimpleButtonPL -simpleButtonPL_KonfPatv : SimpleButtonPL -simpleButtonPL_KlientAdmin : SimpleButtonPL -simpleButtonPL_ApartAdmin : SimpleButtonPL -simpleButtonPL_DarbAdmin : SimpleButtonPL -bar1 : Bar -barButtonItem1 : BarButtonItem -simpleButtonPL_ApTipai : SimpleButtonPL -barEditItem1 : BarEditItem -repositoryItemTextEdit1 : RepositoryItemTextEdit -barAndDockingController1 : BarAndDockingController -components : IContainer  +Form1() #Dispose(in disposing : bool) -InitializeComponent() -Main() -Form1_Load(in sender : object, in e : EventArgs) -simpleButtonPL_PatKamb_Click(in sender : object, in e : EventArgs) -simpleButtonPL_KonfPatv_Click(in sender : object, in e : EventArgs) -simpleButtonPL_UzsAdmin_Click(in sender : object, in e : EventArgs) -barButtonItem1_ItemClick(in sender : object, in e : ItemClickEventArgs) -simpleButtonPL_KlientAdmin_Click(in sender : object, in e : EventArgs) -simpleButtonPL_DarbAdmin_Click(in sender : object, in e : EventArgs) -simpleButtonPL_ApartAdmin_Click(in sender : object, in e : EventArgs) -simpleButtonPL1_Click_1(in sender : object, in e : EventArgs) </pre>			
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center;">Configuration</th> </tr> </thead> <tbody> <tr> <td> <pre> +ConnectionInfo() : string </pre> </td> </tr> </tbody> </table>	Configuration	<pre> +ConnectionInfo() : string </pre>	<p>Konfiguracijos klasė, joje laikoma informacija, kuri yra pakartotinai naudojama (pvz.: <i>SQL Connection string</i>).</p>
Configuration			
<pre> +ConnectionInfo() : string </pre>			
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center;">PatvKonf</th> </tr> </thead> <tbody> <tr> <td> <pre> -myDataSet : DataSet -gridControl1 : GridControl -gridView1 : GridView -Uzsakymas : GridColumn -Kambnr : GridColumn -AtvykData : GridColumn -IsvykData : GridColumn -UzsakData : GridColumn -Klogin : GridColumn -Kitainfo : GridColumn -simpleButtonPL_Patv : SimpleButtonPL -components : Container = null  +PatvKonf() #Dispose(in disposing : bool) -InitializeComponent() -PatvKonf_Load(in sender : object, in e : EventArgs) -simpleButtonPL_Patv_Click(in sender : object, in e : EventArgs) </pre> </td> </tr> </tbody> </table>	PatvKonf	<pre> -myDataSet : DataSet -gridControl1 : GridControl -gridView1 : GridView -Uzsakymas : GridColumn -Kambnr : GridColumn -AtvykData : GridColumn -IsvykData : GridColumn -UzsakData : GridColumn -Klogin : GridColumn -Kitainfo : GridColumn -simpleButtonPL_Patv : SimpleButtonPL -components : Container = null  +PatvKonf() #Dispose(in disposing : bool) -InitializeComponent() -PatvKonf_Load(in sender : object, in e : EventArgs) -simpleButtonPL_Patv_Click(in sender : object, in e : EventArgs) </pre>	<p>Konferencijos salės užsakymų patvirtinimo langas</p>
PatvKonf			
<pre> -myDataSet : DataSet -gridControl1 : GridControl -gridView1 : GridView -Uzsakymas : GridColumn -Kambnr : GridColumn -AtvykData : GridColumn -IsvykData : GridColumn -UzsakData : GridColumn -Klogin : GridColumn -Kitainfo : GridColumn -simpleButtonPL_Patv : SimpleButtonPL -components : Container = null  +PatvKonf() #Dispose(in disposing : bool) -InitializeComponent() -PatvKonf_Load(in sender : object, in e : EventArgs) -simpleButtonPL_Patv_Click(in sender : object, in e : EventArgs) </pre>			

Klasė	Trumpas aprašymas			
<table border="1" data-bbox="266 262 821 690"> <thead> <tr> <th data-bbox="266 262 821 296">PatvirtKamb</th> </tr> </thead> <tbody> <tr> <td data-bbox="266 296 821 579">           -myDataSet : DataSet            -gridControl1 : GridControl            -gridView1 : GridView            -KambNr : GridColumn            -Uzsakymas : GridColumn            -Atvykdata : GridColumn            -IsvykData : GridColumn            -UzsakData : GridColumn            -KlientoLoginas : GridColumn            -KitaInfo : GridColumn            -repositoryItemSpinEdit1 : RepositoryItemSpinEdit            -simpleButtonPL_Patvirtinti : SimpleButtonPL            -components : Container = null         </td> </tr> <tr> <td data-bbox="266 579 821 690">           +PatvirtKamb()            #Dispose(in disposing : bool)            -InitializeComponent()            -PatvirtKamb_Load(in sender : object, in e : EventArgs)            -simpleButtonPL_Patvirtinti_Click(in sender : object, in e : EventArgs)         </td> </tr> </tbody> </table>	PatvirtKamb	-myDataSet : DataSet -gridControl1 : GridControl -gridView1 : GridView -KambNr : GridColumn -Uzsakymas : GridColumn -Atvykdata : GridColumn -IsvykData : GridColumn -UzsakData : GridColumn -KlientoLoginas : GridColumn -KitaInfo : GridColumn -repositoryItemSpinEdit1 : RepositoryItemSpinEdit -simpleButtonPL_Patvirtinti : SimpleButtonPL -components : Container = null	+PatvirtKamb() #Dispose(in disposing : bool) -InitializeComponent() -PatvirtKamb_Load(in sender : object, in e : EventArgs) -simpleButtonPL_Patvirtinti_Click(in sender : object, in e : EventArgs)	Kambarių užsakymų patvirtinimo langas (27 pav.)
PatvirtKamb				
-myDataSet : DataSet -gridControl1 : GridControl -gridView1 : GridView -KambNr : GridColumn -Uzsakymas : GridColumn -Atvykdata : GridColumn -IsvykData : GridColumn -UzsakData : GridColumn -KlientoLoginas : GridColumn -KitaInfo : GridColumn -repositoryItemSpinEdit1 : RepositoryItemSpinEdit -simpleButtonPL_Patvirtinti : SimpleButtonPL -components : Container = null				
+PatvirtKamb() #Dispose(in disposing : bool) -InitializeComponent() -PatvirtKamb_Load(in sender : object, in e : EventArgs) -simpleButtonPL_Patvirtinti_Click(in sender : object, in e : EventArgs)				
<table border="1" data-bbox="266 772 789 1180"> <thead> <tr> <th data-bbox="266 772 789 806">ApartTipai</th> </tr> </thead> <tbody> <tr> <td data-bbox="266 806 789 1068">           -myDataSet : DataSet            -gridControl1 : GridControl            -gridView1 : GridView            -apart : GridColumn            -ap_kaina : GridColumn            -ap_info : GridColumn            -apartid : GridColumn            -ap_pav : GridColumn            -repositoryItemImageEdit1 : RepositoryItemImageEdit            -simpleButtonPL_Delete : SimpleButtonPL            -simpleButtonPL_Save : SimpleButtonPL            -components : Container = null         </td> </tr> <tr> <td data-bbox="266 1068 789 1180">           +ApartTipai()            #Dispose(in disposing : bool)            -InitializeComponent()            -ApartTipai_Load(in sender : object, in e : EventArgs)            -simpleButtonPL_Save_Click(in sender : object, in e : EventArgs)         </td> </tr> </tbody> </table>	ApartTipai	-myDataSet : DataSet -gridControl1 : GridControl -gridView1 : GridView -apart : GridColumn -ap_kaina : GridColumn -ap_info : GridColumn -apartid : GridColumn -ap_pav : GridColumn -repositoryItemImageEdit1 : RepositoryItemImageEdit -simpleButtonPL_Delete : SimpleButtonPL -simpleButtonPL_Save : SimpleButtonPL -components : Container = null	+ApartTipai() #Dispose(in disposing : bool) -InitializeComponent() -ApartTipai_Load(in sender : object, in e : EventArgs) -simpleButtonPL_Save_Click(in sender : object, in e : EventArgs)	Apartamentų tipų administravimo langas
ApartTipai				
-myDataSet : DataSet -gridControl1 : GridControl -gridView1 : GridView -apart : GridColumn -ap_kaina : GridColumn -ap_info : GridColumn -apartid : GridColumn -ap_pav : GridColumn -repositoryItemImageEdit1 : RepositoryItemImageEdit -simpleButtonPL_Delete : SimpleButtonPL -simpleButtonPL_Save : SimpleButtonPL -components : Container = null				
+ApartTipai() #Dispose(in disposing : bool) -InitializeComponent() -ApartTipai_Load(in sender : object, in e : EventArgs) -simpleButtonPL_Save_Click(in sender : object, in e : EventArgs)				
<table border="1" data-bbox="266 1270 821 1787"> <thead> <tr> <th data-bbox="266 1270 821 1304">Apartamentai</th> </tr> </thead> <tbody> <tr> <td data-bbox="266 1304 821 1633">           -myDataSet : DataSet            -gridControl1 : GridControl            -gridView1 : GridView            -simpleButtonPL_Delete : SimpleButtonPL            -simpleButtonPL_Save : SimpleButtonPL            -simpleButtonPL_pav : SimpleButtonPL            -openFileDialog1 : OpenFileDialog            -repositoryItemImageEdit1 : RepositoryItemImageEdit            -repositoryItemImageComboBox1 : RepositoryItemImageComboBox            -repositoryItemSpinEdit1 : RepositoryItemSpinEdit            -KambarioNr : GridColumn            -apartid : GridColumn            -k_info : GridColumn            -k_pav : GridColumn            -components : Container = null         </td> </tr> <tr> <td data-bbox="266 1633 821 1787">           +Apartamentai()            #Dispose(in disposing : bool)            -InitializeComponent()            -Apartamentai_Load(in sender : object, in e : EventArgs)            -simpleButtonPL_Delete_Click(in sender : object, in e : EventArgs)            -simpleButtonPL_Save_Click(in sender : object, in e : EventArgs)            -simpleButtonPL_pav_Click(in sender : object, in e : EventArgs)         </td> </tr> </tbody> </table>	Apartamentai	-myDataSet : DataSet -gridControl1 : GridControl -gridView1 : GridView -simpleButtonPL_Delete : SimpleButtonPL -simpleButtonPL_Save : SimpleButtonPL -simpleButtonPL_pav : SimpleButtonPL -openFileDialog1 : OpenFileDialog -repositoryItemImageEdit1 : RepositoryItemImageEdit -repositoryItemImageComboBox1 : RepositoryItemImageComboBox -repositoryItemSpinEdit1 : RepositoryItemSpinEdit -KambarioNr : GridColumn -apartid : GridColumn -k_info : GridColumn -k_pav : GridColumn -components : Container = null	+Apartamentai() #Dispose(in disposing : bool) -InitializeComponent() -Apartamentai_Load(in sender : object, in e : EventArgs) -simpleButtonPL_Delete_Click(in sender : object, in e : EventArgs) -simpleButtonPL_Save_Click(in sender : object, in e : EventArgs) -simpleButtonPL_pav_Click(in sender : object, in e : EventArgs)	Kambarių administravimo langas (25 ir 26 pav.)
Apartamentai				
-myDataSet : DataSet -gridControl1 : GridControl -gridView1 : GridView -simpleButtonPL_Delete : SimpleButtonPL -simpleButtonPL_Save : SimpleButtonPL -simpleButtonPL_pav : SimpleButtonPL -openFileDialog1 : OpenFileDialog -repositoryItemImageEdit1 : RepositoryItemImageEdit -repositoryItemImageComboBox1 : RepositoryItemImageComboBox -repositoryItemSpinEdit1 : RepositoryItemSpinEdit -KambarioNr : GridColumn -apartid : GridColumn -k_info : GridColumn -k_pav : GridColumn -components : Container = null				
+Apartamentai() #Dispose(in disposing : bool) -InitializeComponent() -Apartamentai_Load(in sender : object, in e : EventArgs) -simpleButtonPL_Delete_Click(in sender : object, in e : EventArgs) -simpleButtonPL_Save_Click(in sender : object, in e : EventArgs) -simpleButtonPL_pav_Click(in sender : object, in e : EventArgs)				

Klasė	Trumpas aprašymas
<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;"> <p style="text-align: center;"><b>Klientai</b></p> <pre> -myDataSet : DataSet -gridControl1 : GridControl -gridView1 : GridView -simpleButtonPL1 : SimpleButtonPL -simpleButtonPL2 : SimpleButtonPL -simpleButtonPL3 : SimpleButtonPL -repositoryItemTextEdit1 : RepositoryItemTextEdit -KLogin : GridColumn -Kpass : GridColumn -Kvardas : GridColumn -K_pavard : GridColumn -K_email : GridColumn -K_tel : GridColumn -K_pris : GridColumn -repositoryItemCalcEdit1 : RepositoryItemCalcEdit -repositoryItemHyperLinkEdit1 : RepositoryItemHyperLinkEdit -components : Container = null +Klientai() #Dispose(in disposing : bool) -InitializeComponent() -Klientai_Load(in sender : object, in e : EventArgs) -simpleButtonPL_Delete_Click(in sender : object, in e : EventArgs) -simpleButtonPL_Save_Click(in sender : object, in e : EventArgs) -simpleButtonPL_Ataskaita_Click(in sender : object, in e : EventArgs) </pre> </div>	<p>Klientų administravimo langas.</p>
<div style="border: 1px solid black; padding: 5px;"> <p style="text-align: center;"><b>KlientuAtask</b></p> <pre> -Detail : DetailBand -klientuInfo1 : KlientuInfo +klientuInfo2 : KlientuInfo -pageHeaderBand1 : PageHeaderBand -xrLabel1 : XRLabel -xrLabel2 : XRLabel -xrLabel3 : XRLabel -xrLabel4 : XRLabel -xrLabel5 : XRLabel -xrLabel6 : XRLabel -xrLine1 : XRLine -xrLine2 : XRLine -groupHeaderBand1 : GroupHeaderBand -xrLabel7 : XRLabel -xrLabel8 : XRLabel -xrLabel9 : XRLabel -xrLabel10 : XRLabel -xrLabel11 : XRLabel -xrLabel12 : XRLabel -pageFooterBand1 : PageFooterBand -xrPageInfo1 : XRPageInfo -xrPageInfo2 : XRPageInfo -reportHeaderBand1 : ReportHeaderBand -xrLabel13 : XRLabel -xrLine3 : XRLine -components : Container = null +KlientuAtask() #Dispose(in disposing : bool) -InitializeComponent() </pre> </div>	<p>Klientų administravimo lange generuojamos ataskaitos šablonas.</p>

Klasė	Trumpas aprašymas																										
<table border="1" data-bbox="266 247 844 575"> <thead> <tr> <th data-bbox="266 247 844 279">KlientuInfo</th> </tr> </thead> <tbody> <tr> <td data-bbox="266 279 844 310">-tableKLogin : KLoginDataTable</td> </tr> <tr> <td data-bbox="266 310 844 342">+KlientuInfo()</td> </tr> <tr> <td data-bbox="266 342 844 373">#KlientuInfo(in info : SerializationInfo, in context : StreamingContext)</td> </tr> <tr> <td data-bbox="266 373 844 405">+KLogin() : KLoginDataTable</td> </tr> <tr> <td data-bbox="266 405 844 436">+Clone() : DataSet</td> </tr> <tr> <td data-bbox="266 436 844 468">#ShouldSerializeTables() : bool</td> </tr> <tr> <td data-bbox="266 468 844 499">#ShouldSerializeRelations() : bool</td> </tr> <tr> <td data-bbox="266 499 844 531">#ReadXmlSerializable(in reader : XmlReader)</td> </tr> <tr> <td data-bbox="266 531 844 562">#GetSchemaSerializable() : XmlSchema</td> </tr> <tr> <td data-bbox="266 562 844 594">+InitVars()</td> </tr> <tr> <td data-bbox="266 594 844 625">-InitClass()</td> </tr> <tr> <td data-bbox="266 625 844 657">-ShouldSerializeKLogin() : bool</td> </tr> <tr> <td data-bbox="266 657 844 688">-SchemaChanged(in sender : object, in e : CollectionChangeEventArgs)</td> </tr> </tbody> </table>	KlientuInfo	-tableKLogin : KLoginDataTable	+KlientuInfo()	#KlientuInfo(in info : SerializationInfo, in context : StreamingContext)	+KLogin() : KLoginDataTable	+Clone() : DataSet	#ShouldSerializeTables() : bool	#ShouldSerializeRelations() : bool	#ReadXmlSerializable(in reader : XmlReader)	#GetSchemaSerializable() : XmlSchema	+InitVars()	-InitClass()	-ShouldSerializeKLogin() : bool	-SchemaChanged(in sender : object, in e : CollectionChangeEventArgs)	Klientų ataskaitos naudojamų duomenų aprašymas.												
KlientuInfo																											
-tableKLogin : KLoginDataTable																											
+KlientuInfo()																											
#KlientuInfo(in info : SerializationInfo, in context : StreamingContext)																											
+KLogin() : KLoginDataTable																											
+Clone() : DataSet																											
#ShouldSerializeTables() : bool																											
#ShouldSerializeRelations() : bool																											
#ReadXmlSerializable(in reader : XmlReader)																											
#GetSchemaSerializable() : XmlSchema																											
+InitVars()																											
-InitClass()																											
-ShouldSerializeKLogin() : bool																											
-SchemaChanged(in sender : object, in e : CollectionChangeEventArgs)																											
<table border="1" data-bbox="266 674 844 1253"> <thead> <tr> <th data-bbox="266 674 844 705">Uzsakymai</th> </tr> </thead> <tbody> <tr> <td data-bbox="266 705 844 737">-myDataSet : DataSet</td> </tr> <tr> <td data-bbox="266 737 844 768">-gridControl1 : GridControl</td> </tr> <tr> <td data-bbox="266 768 844 800">-gridView1 : GridView</td> </tr> <tr> <td data-bbox="266 800 844 831">-Uzsakymas : GridColumn</td> </tr> <tr> <td data-bbox="266 831 844 863">-KambNr : GridColumn</td> </tr> <tr> <td data-bbox="266 863 844 894">-repositoryItemSpinEdit1 : RepositoryItemSpinEdit</td> </tr> <tr> <td data-bbox="266 894 844 926">-Atvykdata : GridColumn</td> </tr> <tr> <td data-bbox="266 926 844 957">-IsvykData : GridColumn</td> </tr> <tr> <td data-bbox="266 957 844 989">-UzsakData : GridColumn</td> </tr> <tr> <td data-bbox="266 989 844 1020">-KitaInfo : GridColumn</td> </tr> <tr> <td data-bbox="266 1020 844 1052">-Patvirtinta : GridColumn</td> </tr> <tr> <td data-bbox="266 1052 844 1083">-DLogin : GridColumn</td> </tr> <tr> <td data-bbox="266 1083 844 1115">-KLogin : GridColumn</td> </tr> <tr> <td data-bbox="266 1115 844 1146">-simpleButtonPL1 : SimpleButtonPL</td> </tr> <tr> <td data-bbox="266 1146 844 1178">-simpleButtonPL2 : SimpleButtonPL</td> </tr> <tr> <td data-bbox="266 1178 844 1209">-simpleButtonPL_Ataskaita : SimpleButtonPL</td> </tr> <tr> <td data-bbox="266 1209 844 1241">-repositoryItemCheckEdit1 : RepositoryItemCheckEdit</td> </tr> <tr> <td data-bbox="266 1241 844 1272">-components : IContainer</td> </tr> <tr> <td data-bbox="266 1272 844 1304">+Uzsakymai()</td> </tr> <tr> <td data-bbox="266 1304 844 1335">#Dispose(in disposing : bool)</td> </tr> <tr> <td data-bbox="266 1335 844 1367">-InitializeComponent()</td> </tr> <tr> <td data-bbox="266 1367 844 1398">-Uzsakymai_Load(in sender : object, in e : EventArgs)</td> </tr> <tr> <td data-bbox="266 1398 844 1430">-simpleButtonPL1_Click(in sender : object, in e : EventArgs)</td> </tr> <tr> <td data-bbox="266 1430 844 1461">-simpleButtonPL2_Click(in sender : object, in e : EventArgs)</td> </tr> <tr> <td data-bbox="266 1461 844 1493">-simpleButtonPL_Ataskaita_Click(in sender : object, in e : EventArgs)</td> </tr> </tbody> </table>	Uzsakymai	-myDataSet : DataSet	-gridControl1 : GridControl	-gridView1 : GridView	-Uzsakymas : GridColumn	-KambNr : GridColumn	-repositoryItemSpinEdit1 : RepositoryItemSpinEdit	-Atvykdata : GridColumn	-IsvykData : GridColumn	-UzsakData : GridColumn	-KitaInfo : GridColumn	-Patvirtinta : GridColumn	-DLogin : GridColumn	-KLogin : GridColumn	-simpleButtonPL1 : SimpleButtonPL	-simpleButtonPL2 : SimpleButtonPL	-simpleButtonPL_Ataskaita : SimpleButtonPL	-repositoryItemCheckEdit1 : RepositoryItemCheckEdit	-components : IContainer	+Uzsakymai()	#Dispose(in disposing : bool)	-InitializeComponent()	-Uzsakymai_Load(in sender : object, in e : EventArgs)	-simpleButtonPL1_Click(in sender : object, in e : EventArgs)	-simpleButtonPL2_Click(in sender : object, in e : EventArgs)	-simpleButtonPL_Ataskaita_Click(in sender : object, in e : EventArgs)	Užsakymų administravimo langas (28 ir 29 pav.).
Uzsakymai																											
-myDataSet : DataSet																											
-gridControl1 : GridControl																											
-gridView1 : GridView																											
-Uzsakymas : GridColumn																											
-KambNr : GridColumn																											
-repositoryItemSpinEdit1 : RepositoryItemSpinEdit																											
-Atvykdata : GridColumn																											
-IsvykData : GridColumn																											
-UzsakData : GridColumn																											
-KitaInfo : GridColumn																											
-Patvirtinta : GridColumn																											
-DLogin : GridColumn																											
-KLogin : GridColumn																											
-simpleButtonPL1 : SimpleButtonPL																											
-simpleButtonPL2 : SimpleButtonPL																											
-simpleButtonPL_Ataskaita : SimpleButtonPL																											
-repositoryItemCheckEdit1 : RepositoryItemCheckEdit																											
-components : IContainer																											
+Uzsakymai()																											
#Dispose(in disposing : bool)																											
-InitializeComponent()																											
-Uzsakymai_Load(in sender : object, in e : EventArgs)																											
-simpleButtonPL1_Click(in sender : object, in e : EventArgs)																											
-simpleButtonPL2_Click(in sender : object, in e : EventArgs)																											
-simpleButtonPL_Ataskaita_Click(in sender : object, in e : EventArgs)																											

Klasė	Trumpas aprašymas																																
<table border="1" data-bbox="266 298 605 1005"> <thead> <tr> <th>UzsakymuAtask</th> </tr> </thead> <tbody> <tr> <td>-Detail : DetailBand</td> </tr> <tr> <td>+dataset11 : DataSet1</td> </tr> <tr> <td>-groupHeaderBand1 : GroupHeaderBand</td> </tr> <tr> <td>-xrLabel1 : XRLabel</td> </tr> <tr> <td>-xrLabel2 : XRLabel</td> </tr> <tr> <td>-groupHeaderBand2 : GroupHeaderBand</td> </tr> <tr> <td>-xrLabel3 : XRLabel</td> </tr> <tr> <td>-xrLabel4 : XRLabel</td> </tr> <tr> <td>-xrLabel5 : XRLabel</td> </tr> <tr> <td>-xrLabel6 : XRLabel</td> </tr> <tr> <td>-xrLabel7 : XRLabel</td> </tr> <tr> <td>-xrLabel8 : XRLabel</td> </tr> <tr> <td>-xrLabel9 : XRLabel</td> </tr> <tr> <td>-xrLine1 : XRLine</td> </tr> <tr> <td>-xrLine2 : XRLine</td> </tr> <tr> <td>-xrLabel10 : XRLabel</td> </tr> <tr> <td>-xrLabel11 : XRLabel</td> </tr> <tr> <td>-xrLabel12 : XRLabel</td> </tr> <tr> <td>-xrLabel13 : XRLabel</td> </tr> <tr> <td>-xrLabel14 : XRLabel</td> </tr> <tr> <td>-xrLabel15 : XRLabel</td> </tr> <tr> <td>-xrLabel16 : XRLabel</td> </tr> <tr> <td>-pageFooterBand1 : PageFooterBand</td> </tr> <tr> <td>-xrPageInfo1 : XRPageInfo</td> </tr> <tr> <td>-xrPageInfo2 : XRPageInfo</td> </tr> <tr> <td>-reportHeaderBand1 : ReportHeaderBand</td> </tr> <tr> <td>-xrLabel17 : XRLabel</td> </tr> <tr> <td>-components : Container = null</td> </tr> <tr> <td>+UzsakymuAtask()</td> </tr> <tr> <td>#Dispose(in disposing : bool)</td> </tr> <tr> <td>-InitializeComponent()</td> </tr> </tbody> </table>	UzsakymuAtask	-Detail : DetailBand	+dataset11 : DataSet1	-groupHeaderBand1 : GroupHeaderBand	-xrLabel1 : XRLabel	-xrLabel2 : XRLabel	-groupHeaderBand2 : GroupHeaderBand	-xrLabel3 : XRLabel	-xrLabel4 : XRLabel	-xrLabel5 : XRLabel	-xrLabel6 : XRLabel	-xrLabel7 : XRLabel	-xrLabel8 : XRLabel	-xrLabel9 : XRLabel	-xrLine1 : XRLine	-xrLine2 : XRLine	-xrLabel10 : XRLabel	-xrLabel11 : XRLabel	-xrLabel12 : XRLabel	-xrLabel13 : XRLabel	-xrLabel14 : XRLabel	-xrLabel15 : XRLabel	-xrLabel16 : XRLabel	-pageFooterBand1 : PageFooterBand	-xrPageInfo1 : XRPageInfo	-xrPageInfo2 : XRPageInfo	-reportHeaderBand1 : ReportHeaderBand	-xrLabel17 : XRLabel	-components : Container = null	+UzsakymuAtask()	#Dispose(in disposing : bool)	-InitializeComponent()	<p>Užsakymų administravimo lange generuojamos ataskaitos šablonas (30 pav.).</p>
UzsakymuAtask																																	
-Detail : DetailBand																																	
+dataset11 : DataSet1																																	
-groupHeaderBand1 : GroupHeaderBand																																	
-xrLabel1 : XRLabel																																	
-xrLabel2 : XRLabel																																	
-groupHeaderBand2 : GroupHeaderBand																																	
-xrLabel3 : XRLabel																																	
-xrLabel4 : XRLabel																																	
-xrLabel5 : XRLabel																																	
-xrLabel6 : XRLabel																																	
-xrLabel7 : XRLabel																																	
-xrLabel8 : XRLabel																																	
-xrLabel9 : XRLabel																																	
-xrLine1 : XRLine																																	
-xrLine2 : XRLine																																	
-xrLabel10 : XRLabel																																	
-xrLabel11 : XRLabel																																	
-xrLabel12 : XRLabel																																	
-xrLabel13 : XRLabel																																	
-xrLabel14 : XRLabel																																	
-xrLabel15 : XRLabel																																	
-xrLabel16 : XRLabel																																	
-pageFooterBand1 : PageFooterBand																																	
-xrPageInfo1 : XRPageInfo																																	
-xrPageInfo2 : XRPageInfo																																	
-reportHeaderBand1 : ReportHeaderBand																																	
-xrLabel17 : XRLabel																																	
-components : Container = null																																	
+UzsakymuAtask()																																	
#Dispose(in disposing : bool)																																	
-InitializeComponent()																																	
<table border="1" data-bbox="266 1075 846 1398"> <thead> <tr> <th>Dataset1</th> </tr> </thead> <tbody> <tr> <td>-tableUzsakymas : UzsakymasDataTable</td> </tr> <tr> <td>+Dataset1()</td> </tr> <tr> <td>#Dataset1(in info : SerializationInfo, in context : StreamingContext)</td> </tr> <tr> <td>+Uzsakymas() : UzsakymasDataTable</td> </tr> <tr> <td>+Clone() : DataSet</td> </tr> <tr> <td>#ShouldSerializeTables() : bool</td> </tr> <tr> <td>#ShouldSerializeRelations() : bool</td> </tr> <tr> <td>#ReadXmlSerializable(in reader : XmlReader)</td> </tr> <tr> <td>#GetSchemaSerializable() : XmlSchema</td> </tr> <tr> <td>+InitVars()</td> </tr> <tr> <td>-InitClass()</td> </tr> <tr> <td>-ShouldSerializeUzsakymas() : bool</td> </tr> <tr> <td>-SchemaChanged(in sender : object, in e : CollectionChangeEventArgs)</td> </tr> </tbody> </table>	Dataset1	-tableUzsakymas : UzsakymasDataTable	+Dataset1()	#Dataset1(in info : SerializationInfo, in context : StreamingContext)	+Uzsakymas() : UzsakymasDataTable	+Clone() : DataSet	#ShouldSerializeTables() : bool	#ShouldSerializeRelations() : bool	#ReadXmlSerializable(in reader : XmlReader)	#GetSchemaSerializable() : XmlSchema	+InitVars()	-InitClass()	-ShouldSerializeUzsakymas() : bool	-SchemaChanged(in sender : object, in e : CollectionChangeEventArgs)	<p>Užsakymų ataskaitos naudojamų duomenų aprašymas.</p>																		
Dataset1																																	
-tableUzsakymas : UzsakymasDataTable																																	
+Dataset1()																																	
#Dataset1(in info : SerializationInfo, in context : StreamingContext)																																	
+Uzsakymas() : UzsakymasDataTable																																	
+Clone() : DataSet																																	
#ShouldSerializeTables() : bool																																	
#ShouldSerializeRelations() : bool																																	
#ReadXmlSerializable(in reader : XmlReader)																																	
#GetSchemaSerializable() : XmlSchema																																	
+InitVars()																																	
-InitClass()																																	
-ShouldSerializeUzsakymas() : bool																																	
-SchemaChanged(in sender : object, in e : CollectionChangeEventArgs)																																	
<table border="1" data-bbox="266 1472 821 1814"> <thead> <tr> <th>Darbuotojai</th> </tr> </thead> <tbody> <tr> <td>-myDataSet : DataSet</td> </tr> <tr> <td>-gridControl1 : GridControl</td> </tr> <tr> <td>-gridView1 : GridView</td> </tr> <tr> <td>-simpleButtonPL_Delete : SimpleButtonPL</td> </tr> <tr> <td>-simpleButtonPL_Save : SimpleButtonPL</td> </tr> <tr> <td>-simpleButtonPL1 : SimpleButtonPL</td> </tr> <tr> <td>-components : Container = null</td> </tr> <tr> <td>+Darbuotojai()</td> </tr> <tr> <td>#Dispose(in disposing : bool)</td> </tr> <tr> <td>-InitializeComponent()</td> </tr> <tr> <td>-Darbuotojai_Load(in sender : object, in e : EventArgs)</td> </tr> <tr> <td>-simpleButtonPL_Delete_Click(in sender : object, in e : EventArgs)</td> </tr> <tr> <td>-simpleButtonPL_Save_Click(in sender : object, in e : EventArgs)</td> </tr> <tr> <td>-simpleButtonPL1_Click(in sender : object, in e : EventArgs)</td> </tr> </tbody> </table>	Darbuotojai	-myDataSet : DataSet	-gridControl1 : GridControl	-gridView1 : GridView	-simpleButtonPL_Delete : SimpleButtonPL	-simpleButtonPL_Save : SimpleButtonPL	-simpleButtonPL1 : SimpleButtonPL	-components : Container = null	+Darbuotojai()	#Dispose(in disposing : bool)	-InitializeComponent()	-Darbuotojai_Load(in sender : object, in e : EventArgs)	-simpleButtonPL_Delete_Click(in sender : object, in e : EventArgs)	-simpleButtonPL_Save_Click(in sender : object, in e : EventArgs)	-simpleButtonPL1_Click(in sender : object, in e : EventArgs)	<p>Darbuotojų administravimo langas.</p>																	
Darbuotojai																																	
-myDataSet : DataSet																																	
-gridControl1 : GridControl																																	
-gridView1 : GridView																																	
-simpleButtonPL_Delete : SimpleButtonPL																																	
-simpleButtonPL_Save : SimpleButtonPL																																	
-simpleButtonPL1 : SimpleButtonPL																																	
-components : Container = null																																	
+Darbuotojai()																																	
#Dispose(in disposing : bool)																																	
-InitializeComponent()																																	
-Darbuotojai_Load(in sender : object, in e : EventArgs)																																	
-simpleButtonPL_Delete_Click(in sender : object, in e : EventArgs)																																	
-simpleButtonPL_Save_Click(in sender : object, in e : EventArgs)																																	
-simpleButtonPL1_Click(in sender : object, in e : EventArgs)																																	

Klasė	Trumpas aprašymas																															
<table border="1" data-bbox="266 296 605 982"> <thead> <tr> <th>Reports: <b>DarbReportas</b></th> </tr> </thead> <tbody> <tr><td>-Detail : DetailBand</td></tr> <tr><td>+darbuotoju1 : Darbuotoju</td></tr> <tr><td>-pageHeaderBand1 : PageHeaderBand</td></tr> <tr><td>-xrLabel1 : XRLabel</td></tr> <tr><td>-xrLabel2 : XRLabel</td></tr> <tr><td>-xrLabel3 : XRLabel</td></tr> <tr><td>-xrLabel4 : XRLabel</td></tr> <tr><td>-xrLabel5 : XRLabel</td></tr> <tr><td>-xrLabel6 : XRLabel</td></tr> <tr><td>-xrLabel7 : XRLabel</td></tr> <tr><td>-xrLine1 : XRLine</td></tr> <tr><td>-xrLine2 : XRLine</td></tr> <tr><td>-groupHeaderBand1 : GroupHeaderBand</td></tr> <tr><td>-xrLabel8 : XRLabel</td></tr> <tr><td>-xrLabel9 : XRLabel</td></tr> <tr><td>-xrLabel10 : XRLabel</td></tr> <tr><td>-xrLabel11 : XRLabel</td></tr> <tr><td>-xrLabel12 : XRLabel</td></tr> <tr><td>-xrLabel13 : XRLabel</td></tr> <tr><td>-xrLabel14 : XRLabel</td></tr> <tr><td>-pageFooterBand1 : PageFooterBand</td></tr> <tr><td>-xrPageInfo1 : XRPageInfo</td></tr> <tr><td>-xrPageInfo2 : XRPageInfo</td></tr> <tr><td>-reportHeaderBand1 : ReportHeaderBand</td></tr> <tr><td>-xrLabel15 : XRLabel</td></tr> <tr><td>-xrLine3 : XRLine</td></tr> <tr><td>-components : Container = null</td></tr> <tr><td>+DarbReportas()</td></tr> <tr><td>#Dispose(in disposing : bool)</td></tr> <tr><td>-InitializeComponent()</td></tr> </tbody> </table>	Reports: <b>DarbReportas</b>	-Detail : DetailBand	+darbuotoju1 : Darbuotoju	-pageHeaderBand1 : PageHeaderBand	-xrLabel1 : XRLabel	-xrLabel2 : XRLabel	-xrLabel3 : XRLabel	-xrLabel4 : XRLabel	-xrLabel5 : XRLabel	-xrLabel6 : XRLabel	-xrLabel7 : XRLabel	-xrLine1 : XRLine	-xrLine2 : XRLine	-groupHeaderBand1 : GroupHeaderBand	-xrLabel8 : XRLabel	-xrLabel9 : XRLabel	-xrLabel10 : XRLabel	-xrLabel11 : XRLabel	-xrLabel12 : XRLabel	-xrLabel13 : XRLabel	-xrLabel14 : XRLabel	-pageFooterBand1 : PageFooterBand	-xrPageInfo1 : XRPageInfo	-xrPageInfo2 : XRPageInfo	-reportHeaderBand1 : ReportHeaderBand	-xrLabel15 : XRLabel	-xrLine3 : XRLine	-components : Container = null	+DarbReportas()	#Dispose(in disposing : bool)	-InitializeComponent()	<p>Darbuotojų administravimo lange generuojamos ataskaitos šablonas.</p>
Reports: <b>DarbReportas</b>																																
-Detail : DetailBand																																
+darbuotoju1 : Darbuotoju																																
-pageHeaderBand1 : PageHeaderBand																																
-xrLabel1 : XRLabel																																
-xrLabel2 : XRLabel																																
-xrLabel3 : XRLabel																																
-xrLabel4 : XRLabel																																
-xrLabel5 : XRLabel																																
-xrLabel6 : XRLabel																																
-xrLabel7 : XRLabel																																
-xrLine1 : XRLine																																
-xrLine2 : XRLine																																
-groupHeaderBand1 : GroupHeaderBand																																
-xrLabel8 : XRLabel																																
-xrLabel9 : XRLabel																																
-xrLabel10 : XRLabel																																
-xrLabel11 : XRLabel																																
-xrLabel12 : XRLabel																																
-xrLabel13 : XRLabel																																
-xrLabel14 : XRLabel																																
-pageFooterBand1 : PageFooterBand																																
-xrPageInfo1 : XRPageInfo																																
-xrPageInfo2 : XRPageInfo																																
-reportHeaderBand1 : ReportHeaderBand																																
-xrLabel15 : XRLabel																																
-xrLine3 : XRLine																																
-components : Container = null																																
+DarbReportas()																																
#Dispose(in disposing : bool)																																
-InitializeComponent()																																
<table border="1" data-bbox="266 1052 846 1377"> <thead> <tr> <th>Reports: <b>Darbuotoju</b></th> </tr> </thead> <tbody> <tr><td>-tabledarbuot : darbuotDataTable</td></tr> <tr><td>+Darbuotoju()</td></tr> <tr><td>#Darbuotoju(in info : SerializationInfo, in context : StreamingContext)</td></tr> <tr><td>+darbuot() : darbuotDataTable</td></tr> <tr><td>+Clone() : DataSet</td></tr> <tr><td>#ShouldSerializeTables() : bool</td></tr> <tr><td>#ShouldSerializeRelations() : bool</td></tr> <tr><td>#ReadXmlSerializable(in reader : XmlReader)</td></tr> <tr><td>#GetSchemaSerializable() : XmlSchema</td></tr> <tr><td>+InitVars()</td></tr> <tr><td>-InitClass()</td></tr> <tr><td>-ShouldSerializedarbuot() : bool</td></tr> <tr><td>-SchemaChanged(in sender : object, in e : CollectionChangeEventArgs)</td></tr> </tbody> </table>	Reports: <b>Darbuotoju</b>	-tabledarbuot : darbuotDataTable	+Darbuotoju()	#Darbuotoju(in info : SerializationInfo, in context : StreamingContext)	+darbuot() : darbuotDataTable	+Clone() : DataSet	#ShouldSerializeTables() : bool	#ShouldSerializeRelations() : bool	#ReadXmlSerializable(in reader : XmlReader)	#GetSchemaSerializable() : XmlSchema	+InitVars()	-InitClass()	-ShouldSerializedarbuot() : bool	-SchemaChanged(in sender : object, in e : CollectionChangeEventArgs)	<p>Darbuotojų ataskaitos naudojamų duomenų aprašymas.</p>																	
Reports: <b>Darbuotoju</b>																																
-tabledarbuot : darbuotDataTable																																
+Darbuotoju()																																
#Darbuotoju(in info : SerializationInfo, in context : StreamingContext)																																
+darbuot() : darbuotDataTable																																
+Clone() : DataSet																																
#ShouldSerializeTables() : bool																																
#ShouldSerializeRelations() : bool																																
#ReadXmlSerializable(in reader : XmlReader)																																
#GetSchemaSerializable() : XmlSchema																																
+InitVars()																																
-InitClass()																																
-ShouldSerializedarbuot() : bool																																
-SchemaChanged(in sender : object, in e : CollectionChangeEventArgs)																																
<p><b>Rezervavimo programos klasių diagrama</b></p>																																

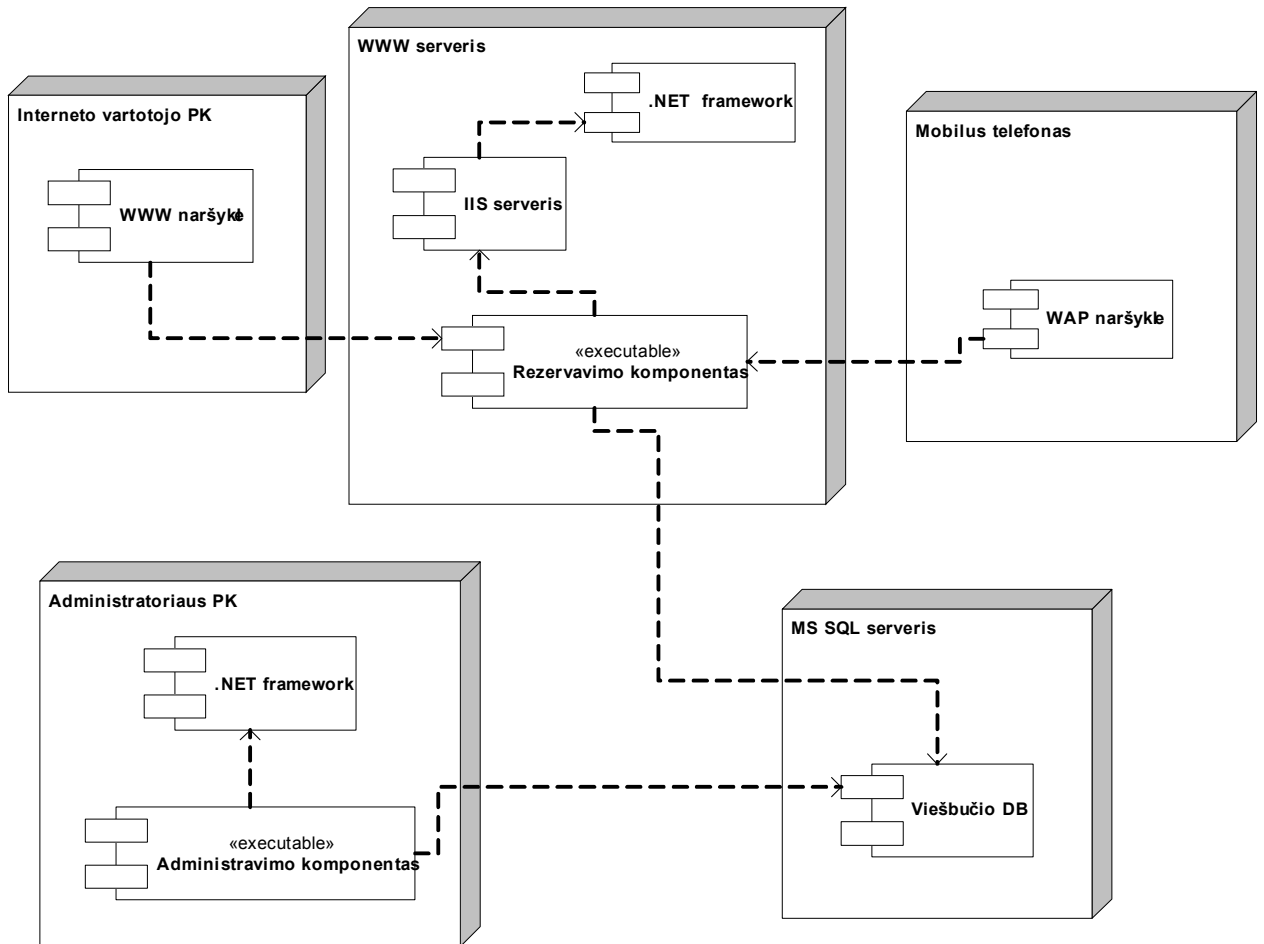


Klasė	Trumpas aprašymas		
<table border="1" data-bbox="266 254 673 619"> <thead> <tr> <th>WebForm1</th> </tr> </thead> <tbody> <tr> <td> #Label1 : Label  #Button2 : Button  #TextBox_Login : TextBox  #TextBox_Pass : TextBox  #Label2 : Label  #Label3 : Label  #Label_Klaida : Label  #Button3 : Button  #Button1 : Button  -Page_Load(in sender : object, in e : EventArgs)  #OnInit(in e : EventArgs)  -InitializeComponent()  -Button1_Click(in sender : object, in e : EventArgs)  -Button2_Click(in sender : object, in e : EventArgs)  -Button3_Click(in sender : object, in e : EventArgs) </td> </tr> </tbody> </table>	WebForm1	#Label1 : Label #Button2 : Button #TextBox_Login : TextBox #TextBox_Pass : TextBox #Label2 : Label #Label3 : Label #Label_Klaida : Label #Button3 : Button #Button1 : Button -Page_Load(in sender : object, in e : EventArgs) #OnInit(in e : EventArgs) -InitializeComponent() -Button1_Click(in sender : object, in e : EventArgs) -Button2_Click(in sender : object, in e : EventArgs) -Button3_Click(in sender : object, in e : EventArgs)	Pagrindinis rezervavimo sistemos puslapis(17 pav.), jame klientai gali prisijungti, registruotis ar pereiti į kambarių peržiūrėjimo puslapį.
WebForm1			
#Label1 : Label #Button2 : Button #TextBox_Login : TextBox #TextBox_Pass : TextBox #Label2 : Label #Label3 : Label #Label_Klaida : Label #Button3 : Button #Button1 : Button -Page_Load(in sender : object, in e : EventArgs) #OnInit(in e : EventArgs) -InitializeComponent() -Button1_Click(in sender : object, in e : EventArgs) -Button2_Click(in sender : object, in e : EventArgs) -Button3_Click(in sender : object, in e : EventArgs)			
<table border="1" data-bbox="266 659 474 747"> <thead> <tr> <th>Configuration</th> </tr> </thead> <tbody> <tr> <td>+ConnectionInfo() : string</td> </tr> </tbody> </table>	Configuration	+ConnectionInfo() : string	Rezervavimo sistemos konfigūracijos klasė, joje laikoma informacija, kuri yra pakartotinai naudojama(pvz.: <i>SQL Connection string</i> ).
Configuration			
+ConnectionInfo() : string			
<table border="1" data-bbox="266 898 1000 1436"> <thead> <tr> <th>NaujasVart</th> </tr> </thead> <tbody> <tr> <td> #Label1 : Label  #Label2 : Label  #Label3 : Label  #Label4 : Label  #Label5 : Label  #Label6 : Label  #Label7 : Label  #Button1 : Button  #TextBox_Login : TextBox  #TextBox_Pass1 : TextBox  #TextBox_Pass2 : TextBox  #TextBox_Vardas : TextBox  #TextBox_Pavarde : TextBox  #TextBox_Email : TextBox  #TextBox_Tel : TextBox  #Label_Klaida : Label  #Button2 : Button  -Page_Load(in sender : object, in e : EventArgs)  #OnInit(in e : EventArgs)  -InitializeComponent()  -Button2_Click(in sender : object, in e : EventArgs)  -Button1_Click(in sender : object, in e : EventArgs)  -registruoti_klienta(in a : string, in b : string, in c : string, in d : string, in e : string, in f : string) </td> </tr> </tbody> </table>	NaujasVart	#Label1 : Label #Label2 : Label #Label3 : Label #Label4 : Label #Label5 : Label #Label6 : Label #Label7 : Label #Button1 : Button #TextBox_Login : TextBox #TextBox_Pass1 : TextBox #TextBox_Pass2 : TextBox #TextBox_Vardas : TextBox #TextBox_Pavarde : TextBox #TextBox_Email : TextBox #TextBox_Tel : TextBox #Label_Klaida : Label #Button2 : Button -Page_Load(in sender : object, in e : EventArgs) #OnInit(in e : EventArgs) -InitializeComponent() -Button2_Click(in sender : object, in e : EventArgs) -Button1_Click(in sender : object, in e : EventArgs) -registruoti_klienta(in a : string, in b : string, in c : string, in d : string, in e : string, in f : string)	Naujo vartotojo sukūrimo langas(19 pav.).
NaujasVart			
#Label1 : Label #Label2 : Label #Label3 : Label #Label4 : Label #Label5 : Label #Label6 : Label #Label7 : Label #Button1 : Button #TextBox_Login : TextBox #TextBox_Pass1 : TextBox #TextBox_Pass2 : TextBox #TextBox_Vardas : TextBox #TextBox_Pavarde : TextBox #TextBox_Email : TextBox #TextBox_Tel : TextBox #Label_Klaida : Label #Button2 : Button -Page_Load(in sender : object, in e : EventArgs) #OnInit(in e : EventArgs) -InitializeComponent() -Button2_Click(in sender : object, in e : EventArgs) -Button1_Click(in sender : object, in e : EventArgs) -registruoti_klienta(in a : string, in b : string, in c : string, in d : string, in e : string, in f : string)			
<table border="1" data-bbox="266 1486 673 1654"> <thead> <tr> <th>SekmingaReg</th> </tr> </thead> <tbody> <tr> <td> #Button1 : Button  #Label1 : Label  -Page_Load(in sender : object, in e : EventArgs)  #OnInit(in e : EventArgs)  -InitializeComponent()  -Button1_Click(in sender : object, in e : EventArgs) </td> </tr> </tbody> </table>	SekmingaReg	#Button1 : Button #Label1 : Label -Page_Load(in sender : object, in e : EventArgs) #OnInit(in e : EventArgs) -InitializeComponent() -Button1_Click(in sender : object, in e : EventArgs)	Sėkmingai sukūrus naują vartotoją yra parodomas šis langas.
SekmingaReg			
#Button1 : Button #Label1 : Label -Page_Load(in sender : object, in e : EventArgs) #OnInit(in e : EventArgs) -InitializeComponent() -Button1_Click(in sender : object, in e : EventArgs)			

Klasė	Trumpas aprašymas			
<table border="1" data-bbox="266 279 722 835"> <thead> <tr> <th data-bbox="266 279 722 310">PasiPerziurejimas</th> </tr> </thead> <tbody> <tr> <td data-bbox="266 310 722 596"> <pre data-bbox="266 310 722 596"> -myDataSet1 : DataSet -myDataSet2 : DataSet -dv : DataView + hta : Hashtable = new Hashtable() #Button_Kamb : Button #Image1 : Image #Button_Apart : Button #ListBox2 : ListBox #Button_Filtras : Button #Button_Atgal : Button #Label_Info : Label #TextBox_info : TextBox #ListBox1 : ListBox </pre> </td> </tr> <tr> <td data-bbox="266 596 722 835"> <pre data-bbox="266 596 722 835"> -Page_Load(in sender : object, in e : EventArgs) #OnInit(in e : EventArgs) -InitializeComponent() -SurastiKamb() -Button_Kamb_Click(in sender : object, in e : EventArgs) -RodytiKambari() -RodytiApart() -uzkrautiTipus() -Button_Apart_Click(in sender : object, in e : EventArgs) -Button_Filtras_Click(in sender : object, in e : EventArgs) -Button_Atgal_Click(in sender : object, in e : EventArgs) </pre> </td> </tr> </tbody> </table>	PasiPerziurejimas	<pre data-bbox="266 310 722 596"> -myDataSet1 : DataSet -myDataSet2 : DataSet -dv : DataView + hta : Hashtable = new Hashtable() #Button_Kamb : Button #Image1 : Image #Button_Apart : Button #ListBox2 : ListBox #Button_Filtras : Button #Button_Atgal : Button #Label_Info : Label #TextBox_info : TextBox #ListBox1 : ListBox </pre>	<pre data-bbox="266 596 722 835"> -Page_Load(in sender : object, in e : EventArgs) #OnInit(in e : EventArgs) -InitializeComponent() -SurastiKamb() -Button_Kamb_Click(in sender : object, in e : EventArgs) -RodytiKambari() -RodytiApart() -uzkrautiTipus() -Button_Apart_Click(in sender : object, in e : EventArgs) -Button_Filtras_Click(in sender : object, in e : EventArgs) -Button_Atgal_Click(in sender : object, in e : EventArgs) </pre>	<p data-bbox="1097 205 1521 863">Kambarių apžiūrejimo langas(18 pav.)</p>
PasiPerziurejimas				
<pre data-bbox="266 310 722 596"> -myDataSet1 : DataSet -myDataSet2 : DataSet -dv : DataView + hta : Hashtable = new Hashtable() #Button_Kamb : Button #Image1 : Image #Button_Apart : Button #ListBox2 : ListBox #Button_Filtras : Button #Button_Atgal : Button #Label_Info : Label #TextBox_info : TextBox #ListBox1 : ListBox </pre>				
<pre data-bbox="266 596 722 835"> -Page_Load(in sender : object, in e : EventArgs) #OnInit(in e : EventArgs) -InitializeComponent() -SurastiKamb() -Button_Kamb_Click(in sender : object, in e : EventArgs) -RodytiKambari() -RodytiApart() -uzkrautiTipus() -Button_Apart_Click(in sender : object, in e : EventArgs) -Button_Filtras_Click(in sender : object, in e : EventArgs) -Button_Atgal_Click(in sender : object, in e : EventArgs) </pre>				
<table border="1" data-bbox="266 905 740 1199"> <thead> <tr> <th data-bbox="266 905 740 936">VartAplinka</th> </tr> </thead> <tbody> <tr> <td data-bbox="266 936 740 1045"> <pre data-bbox="266 936 740 1045"> #Button_Uzsakyti : Button #Button_Atsaukti : Button #DataGrid1 : DataGrid #Button1 : Button #Calendar1 : Calendar </pre> </td> </tr> <tr> <td data-bbox="266 1045 740 1199"> <pre data-bbox="266 1045 740 1199"> -Page_Load(in sender : object, in e : EventArgs) -uzkrautiDuomenis() #OnInit(in e : EventArgs) -InitializeComponent() -Button_Uzsakyti_Click(in sender : object, in e : EventArgs) -Button_Atsaukti_Click(in sender : object, in e : EventArgs) -Button1_Click(in sender : object, in e : EventArgs) </pre> </td> </tr> </tbody> </table>	VartAplinka	<pre data-bbox="266 936 740 1045"> #Button_Uzsakyti : Button #Button_Atsaukti : Button #DataGrid1 : DataGrid #Button1 : Button #Calendar1 : Calendar </pre>	<pre data-bbox="266 1045 740 1199"> -Page_Load(in sender : object, in e : EventArgs) -uzkrautiDuomenis() #OnInit(in e : EventArgs) -InitializeComponent() -Button_Uzsakyti_Click(in sender : object, in e : EventArgs) -Button_Atsaukti_Click(in sender : object, in e : EventArgs) -Button1_Click(in sender : object, in e : EventArgs) </pre>	<p data-bbox="1097 863 1521 1226">Prisijungęs vartotojas patenka į šį puslapį(21 pav.).</p>
VartAplinka				
<pre data-bbox="266 936 740 1045"> #Button_Uzsakyti : Button #Button_Atsaukti : Button #DataGrid1 : DataGrid #Button1 : Button #Calendar1 : Calendar </pre>				
<pre data-bbox="266 1045 740 1199"> -Page_Load(in sender : object, in e : EventArgs) -uzkrautiDuomenis() #OnInit(in e : EventArgs) -InitializeComponent() -Button_Uzsakyti_Click(in sender : object, in e : EventArgs) -Button_Atsaukti_Click(in sender : object, in e : EventArgs) -Button1_Click(in sender : object, in e : EventArgs) </pre>				

Klasė	Trumpas aprašymas
<div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> <p style="text-align: center;"><b>NaujasUzsak</b></p> <pre> -myReader : SqlDataReader -myDataSet : DataSet + _hta : Hashtable = new Hashtable() + _htk : Hashtable = new Hashtable() #Label1 : Label #Label2 : Label #Calendar1 : Calendar #Button2 : Button #Button3 : Button #Label3 : Label #DataGrid1 : DataGrid #TextBox_Atykimas : TextBox #TextBox_Isvykimas : TextBox #RadioButton_Atykimas : RadioButton #DropDownList_Tipai : DropDownList #DropDownList_Kambariai : DropDownList #Button_Surasti : Button #Label_Klaida : Label #Button_Atgal : Button #Image1 : Image #Button_Pic : Button #Label_Pic : Label #RadioButton_Isvykimas : RadioButton </pre> <pre> -Page_Load(in sender : object, in e : EventArgs) #OnInit(in e : EventArgs) -InitializeComponent() -Calendar1_SelectionChanged(in sender : object, in e : EventArgs) -uzkrautiTipus() -patikrinimas() : bool -rezervavimas() +info() -DropDownList_Tipai_SelectedIndexChanged(in sender : object, in e : EventArgs) -Button_Surasti_Click(in sender : object, in e : EventArgs) -Button2_Click(in sender : object, in e : EventArgs) -Button_Atgal_Click(in sender : object, in e : EventArgs) -Button_Pic_Click(in sender : object, in e : EventArgs) -kambPaveikslukai() </pre> </div>	<p>Naujo užsakymo puslapis(22 pav.).</p>
<div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> <p style="text-align: center;"><b>UzsakAtsauk</b></p> <pre> #Calendar1 : Calendar #Button_Istrinti : Button #DropDownList_Uzsakymai : DropDownList #DataGrid1 : DataGrid #Label1 : Label #Button_Atgal : Button </pre> <pre> -Page_Load(in sender : object, in e : EventArgs) #OnInit(in e : EventArgs) -InitializeComponent() -sp() -Button_Istrinti_Click(in sender : object, in e : EventArgs) -Button_Atgal_Click(in sender : object, in e : EventArgs) </pre> </div>	<p>Užsakymų atšaukimo puslapis.</p>
<div style="border: 1px solid black; padding: 5px;"> <p style="text-align: center;"><b>Klaida</b></p> <pre> #Label1 : Label </pre> <pre> -Page_Load(in sender : object, in e : EventArgs) #OnInit(in e : EventArgs) -InitializeComponent() </pre> </div>	<p>Į šį puslapį yra atsiunčiama jei vartotojas bando atlikti kokius nors veiksmus po prisijungimo sesijos laiko pasibaigimo</p>

### 3.9. Viešbučių rezervavimo sistemos diegimo architektūra



15 pav. Viešbučių rezervavimo sistemos diegimo architektūra

### 3.10. Programos objektų aprašymas

7 lentelė. Programos objektų aprašymas

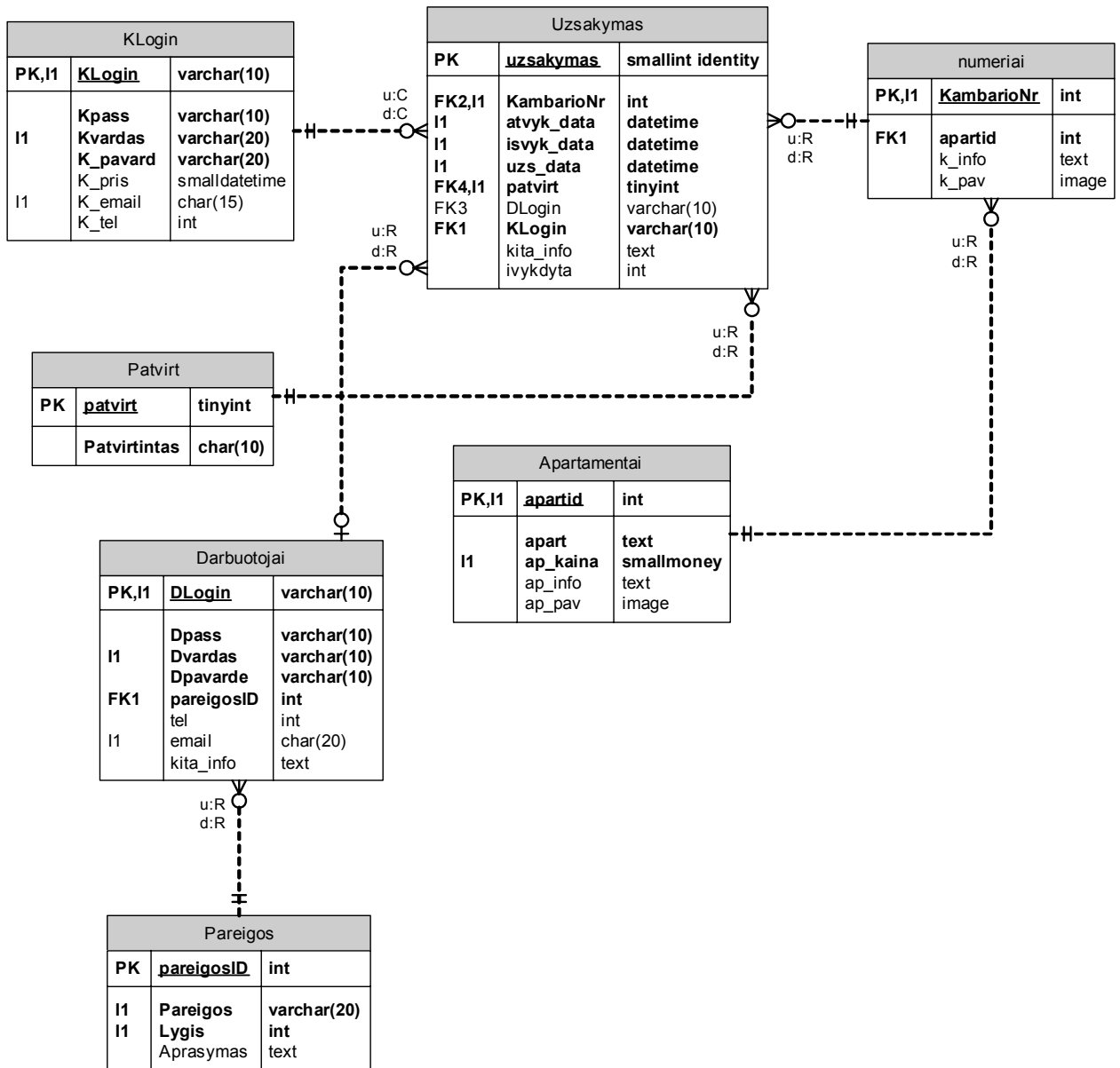
Formos pavadinimas	Duomenų šaltinis	Funkcijos
<b>Sistemos klientų dalis:</b>		
Prisijungimas	Stored procedūra: "uzsakymai"	Logino ir slaptažodžio sutikrinimas su "KLogin" lentelėje esančiais ir gražinimas 1 jei atitinka, kitu atveju - 0
Naujas vartotojas	Lentelė "Klogin"	[rašomas i lentelę "KLogin" naujas vartotojas

Vartotojo aplinka		Navigacija tarp formų
Užsakyti apartamentą	Stored procedūros: "uzsak_rez", "rezerv"	Stored procedūra "rezerv" suranda laisvą pasirinkto apartamento tipo pasirinktu laiko tarpu kambarį. Stored procedūra "uzsak_rez" sukuria užsakymą, irrašydama į lentelę "Uzsakymas" vartotojo prisijungimo vardą, atvykimo datą, išvykimo datą ir užsakymo datą.
Atšaukti užsakymą	Stored procedūra: "uzs_trinti"	Ištrina iš lentelės "Uzsakymas" vartotojo užsakymą.
<b>Sistemos darbuotojų dalis:</b>		
Prisijungimas	Stored procedūra: "loginasD"	Patikrinama ar vartotojo prisijungimo vardas ir slaptažodis sutampa su lentelėje "Darbuotojai" esančiu, jei taip, tada gražinamas to prisijungimo vardo lygis. Pagal gražinta lygį yra apribojamos tam tikros funkcijos. Administratorius gali naudotis visomis funkcijomis. Vadybininkas negali naudotis darbuotojų administravimo ir apartamentų administravimo funkcijomis. Sekretorė negali naudotis tomis funkcijomis, kuriomis negali ir vadybininkas, plius negali naudotis dar ir klientų administravimo funkcija. Sekretorė negali keisti užsakymu, užsakymų administravimo skiltyje.
Apdministravimo aplinka		Navigacija tarp formų
Patvirtinti kambarių užsakymus	Stored procedūros: "nepatvirtinti", "patvirtinti"	Užkrovus formą yra surandama nepatvirtinti kambarių užsakymai, kuriuos, pasinaudojant "stored" procedūra "patvirtinti", jie gali būti patvirtinti.
Patvirtinti konferencijos salės užsakymus	Stored procedūros: "nepatvirtinti_konf", "patvirtinti_konf"	Užkrovus formą yra surandama nepatvirtinti konferencijos salės užsakymai, kuriuos, pasinaudojant stored procedūra

		"patvirtint_konf", jie gali būti patvirtinti.
Užsakymų administravimas	Stored procedūros: "surastiusakymus"	Formai užsikrovus yra atvaizduojami visi užsakymai, kuriuos gali filtruoti pagal 3 filtrus: "kambario nr", "kliento login" ir ar užsakymas jau pasibaigęs ar bus vykdomas ateityje.
Klientų atdiministravimas	Lentelė: "Klogin"	Formai užsikrovus yra atvaizduojami visi užsiregistravę klientai, kuriuos galima keisti, trinti ar įrašyti naują.
Naujas klientas	Lentelė: "Klogin"	Įrašomas naujas klientas
Kliento duomenų keitimas	Lentelė: "Klogin"	Pasirinkto kliento duomenų keitimas
Apartamentų administravimas	Lentelė: "numeriai"	Apartamentų lentelės "numeriai" peržiūrėjimas ir keitimas
Darbuotojų administravimas	Užklausa: "darbuot"	Formai užsikrovus yra atvaizduojami visi užregistruoti darbuotojai kuriuos galima keisti, trinti ar įrašyti naują.
Naujas darbuotojas	Lentelė: "Pareigos"	Įrašomas naujas darbuotojas
Darbuotojo duomenų keitimas	Lentelė: "Pareigos"	Pasirinkto darbuotojo duomenų keitimas

### 3.11. Duomenų bazės schema

#### Viešbučio rezervavimo sistema



16 pav. Duomenų bazės schema

### 3.12. Naudojamų poschemių (View) struktūros specifikacija

8 lentelė. Naudojamų poschemių (View) struktūros specifikacija

<table border="1"> <thead> <tr> <th colspan="2">rezervuoti_kamb</th> </tr> </thead> <tbody> <tr> <td>KambarioNr</td> <td>int</td> </tr> <tr> <td>apartid</td> <td>int</td> </tr> <tr> <td>apart</td> <td>text</td> </tr> <tr> <td>atvyk_data</td> <td>datetime</td> </tr> <tr> <td>isvyk_data</td> <td>datetime</td> </tr> </tbody> </table>	rezervuoti_kamb		KambarioNr	int	apartid	int	apart	text	atvyk_data	datetime	isvyk_data	datetime	<pre>CREATE VIEW dbo.rezervuoti_kamb AS SELECT  dbo.Uzsakymas.KambarioNr, dbo.numeriai.apartid, dbo.Apartamentai.apart,         dbo.Uzsakymas.atvyk_data, dbo.Uzsakymas.isvyk_data FROM    dbo.Uzsakymas INNER JOIN         dbo.numeriai ON dbo.Uzsakymas.KambarioNr = dbo.numeriai.KambarioNr INNER JOIN         dbo.Apartamentai ON dbo.numeriai.apartid = dbo.Apartamentai.apartid</pre>										
rezervuoti_kamb																							
KambarioNr	int																						
apartid	int																						
apart	text																						
atvyk_data	datetime																						
isvyk_data	datetime																						
<table border="1"> <thead> <tr> <th colspan="2">kamb</th> </tr> </thead> <tbody> <tr> <td>KambarioNr</td> <td>int</td> </tr> <tr> <td>apart</td> <td>text</td> </tr> <tr> <td>apartid</td> <td>int</td> </tr> <tr> <td>k_info</td> <td>text</td> </tr> <tr> <td>k_pav</td> <td>image</td> </tr> </tbody> </table>	kamb		KambarioNr	int	apart	text	apartid	int	k_info	text	k_pav	image	<pre>CREATE VIEW dbo.kamb AS SELECT  dbo.numeriai.KambarioNr, dbo.Apartamentai.apart, dbo.numeriai.apartid,         dbo.numeriai.k_info, dbo.numeriai.k_pav FROM    dbo.Apartamentai INNER JOIN         dbo.numeriai ON dbo.Apartamentai.apartid = dbo.numeriai.apartid</pre>										
kamb																							
KambarioNr	int																						
apart	text																						
apartid	int																						
k_info	text																						
k_pav	image																						
<table border="1"> <thead> <tr> <th colspan="2">darbuot</th> </tr> </thead> <tbody> <tr> <td>DLogin</td> <td>varchar(10)</td> </tr> <tr> <td>Dpass</td> <td>varchar(10)</td> </tr> <tr> <td>Dvardas</td> <td>varchar(10)</td> </tr> <tr> <td>Dpavarde</td> <td>varchar(10)</td> </tr> <tr> <td>Pareigos</td> <td>varchar(20)</td> </tr> <tr> <td>tel</td> <td>int</td> </tr> <tr> <td>email</td> <td>char(20)</td> </tr> <tr> <td>kita_info</td> <td>text</td> </tr> </tbody> </table>	darbuot		DLogin	varchar(10)	Dpass	varchar(10)	Dvardas	varchar(10)	Dpavarde	varchar(10)	Pareigos	varchar(20)	tel	int	email	char(20)	kita_info	text	<pre>CREATE VIEW dbo.darbuot AS SELECT  dbo.Darbuotojai.DLogin, dbo.Darbuotojai.Dpass, dbo.Darbuotojai.Dvardas,         dbo.Darbuotojai.Dpavarde, dbo.Pareigos.Pareigos, dbo.Darbuotojai.tel,         dbo.Darbuotojai.email, dbo.Darbuotojai.kita_info FROM    dbo.Darbuotojai INNER JOIN         dbo.Pareigos ON dbo.Darbuotojai.pareigosID = dbo.Pareigos.pareigosID</pre>				
darbuot																							
DLogin	varchar(10)																						
Dpass	varchar(10)																						
Dvardas	varchar(10)																						
Dpavarde	varchar(10)																						
Pareigos	varchar(20)																						
tel	int																						
email	char(20)																						
kita_info	text																						
<table border="1"> <thead> <tr> <th colspan="2">report</th> </tr> </thead> <tbody> <tr> <td>uzsakymas</td> <td>smallint identity</td> </tr> <tr> <td>KambarioNr</td> <td>int</td> </tr> <tr> <td>apart</td> <td>text</td> </tr> <tr> <td>ap_kaina</td> <td>smallmoney</td> </tr> <tr> <td>KLogin</td> <td>varchar(10)</td> </tr> <tr> <td>Kvardas</td> <td>varchar(20)</td> </tr> <tr> <td>K_pavard</td> <td>varchar(20)</td> </tr> <tr> <td>K_email</td> <td>char(15)</td> </tr> <tr> <td>K_tel</td> <td>int</td> </tr> <tr> <td>K_pris</td> <td>smalldatetime</td> </tr> </tbody> </table>	report		uzsakymas	smallint identity	KambarioNr	int	apart	text	ap_kaina	smallmoney	KLogin	varchar(10)	Kvardas	varchar(20)	K_pavard	varchar(20)	K_email	char(15)	K_tel	int	K_pris	smalldatetime	<pre>CREATE VIEW dbo.report AS SELECT  dbo.Uzsakymas.uzsakymas, dbo.Uzsakymas.KambarioNr,         dbo.Apartamentai.apart, dbo.Apartamentai.ap_kaina, dbo.KLogin.KLogin,         dbo.KLogin.Kvardas, dbo.KLogin.K_pavard, dbo.KLogin.K_email,         dbo.KLogin.K_tel, dbo.KLogin.K_pris FROM    dbo.Uzsakymas INNER JOIN         dbo.KLogin ON dbo.Uzsakymas.KLogin = dbo.KLogin.KLogin INNER JOIN         dbo.Patvirt ON dbo.Uzsakymas.patvirt = dbo.Patvirt.patvirt INNER JOIN         dbo.numeriai ON dbo.Uzsakymas.KambarioNr = dbo.numeriai.KambarioNr INNER JOIN</pre>
report																							
uzsakymas	smallint identity																						
KambarioNr	int																						
apart	text																						
ap_kaina	smallmoney																						
KLogin	varchar(10)																						
Kvardas	varchar(20)																						
K_pavard	varchar(20)																						
K_email	char(15)																						
K_tel	int																						
K_pris	smalldatetime																						



		dbo.Apartamentai ON dbo.numeriai.apartid = dbo.Apartamentai.apartid																								
<table border="1"> <thead> <tr> <th colspan="2">uzsak_w</th> </tr> </thead> <tbody> <tr> <td>uzsakymas</td> <td>smallint identity</td> </tr> <tr> <td>KambarioNr</td> <td>int</td> </tr> <tr> <td>apart</td> <td>text</td> </tr> <tr> <td>ap_kaina</td> <td>smallmoney</td> </tr> <tr> <td>atvyk_data</td> <td>datetime</td> </tr> <tr> <td>isvyk_data</td> <td>datetime</td> </tr> <tr> <td>uzs_data</td> <td>datetime</td> </tr> <tr> <td>Patvirtintas</td> <td>char(10)</td> </tr> <tr> <td>DLogin</td> <td>varchar(10)</td> </tr> <tr> <td>KLogin</td> <td>varchar(10)</td> </tr> <tr> <td>kita_info</td> <td>text</td> </tr> </tbody> </table>	uzsak_w		uzsakymas	smallint identity	KambarioNr	int	apart	text	ap_kaina	smallmoney	atvyk_data	datetime	isvyk_data	datetime	uzs_data	datetime	Patvirtintas	char(10)	DLogin	varchar(10)	KLogin	varchar(10)	kita_info	text		<pre> CREATE VIEW dbo.uzsak_w AS SELECT  dbo.Uzsakymas.uzsakymas, dbo.Uzsakymas.KambarioNr,         dbo.Apartamentai.apart, dbo.Apartamentai.ap_kaina, dbo.Uzsakymas.atvyk_data,         dbo.Uzsakymas.isvyk_data, dbo.Uzsakymas.uzs_data,         dbo.Patvirt.Patvirtintas, dbo.Uzsakymas.DLogin, dbo.Uzsakymas.KLogin,         dbo.Uzsakymas.kita_info FROM    dbo.Uzsakymas INNER JOIN         dbo.Patvirt ON dbo.Uzsakymas.patvirt = dbo.Patvirt.patvirt INNER JOIN         dbo.numeriai ON dbo.Uzsakymas.KambarioNr = dbo.numeriai.KambarioNr INNER JOIN         dbo.Apartamentai ON dbo.numeriai.apartid = dbo.Apartamentai.apartid </pre>
uzsak_w																										
uzsakymas	smallint identity																									
KambarioNr	int																									
apart	text																									
ap_kaina	smallmoney																									
atvyk_data	datetime																									
isvyk_data	datetime																									
uzs_data	datetime																									
Patvirtintas	char(10)																									
DLogin	varchar(10)																									
KLogin	varchar(10)																									
kita_info	text																									
<table border="1"> <thead> <tr> <th colspan="2">kamb_patvirt</th> </tr> </thead> <tbody> <tr> <td>uzsakymas</td> <td>smallint identity</td> </tr> <tr> <td>KambarioNr</td> <td>int</td> </tr> <tr> <td>apartid</td> <td>int</td> </tr> <tr> <td>atvyk_data</td> <td>datetime</td> </tr> <tr> <td>isvyk_data</td> <td>datetime</td> </tr> <tr> <td>uzs_data</td> <td>datetime</td> </tr> <tr> <td>patvirt</td> <td>tinyint</td> </tr> <tr> <td>DLogin</td> <td>varchar(10)</td> </tr> <tr> <td>KLogin</td> <td>varchar(10)</td> </tr> <tr> <td>kita_info</td> <td>text</td> </tr> </tbody> </table>	kamb_patvirt		uzsakymas	smallint identity	KambarioNr	int	apartid	int	atvyk_data	datetime	isvyk_data	datetime	uzs_data	datetime	patvirt	tinyint	DLogin	varchar(10)	KLogin	varchar(10)	kita_info	text		<pre> CREATE VIEW dbo.kamb_patvirt AS SELECT  dbo.Uzsakymas.uzsakymas, dbo.Uzsakymas.KambarioNr,         dbo.numeriai.apartid, dbo.Uzsakymas.atvyk_data, dbo.Uzsakymas.isvyk_data,         dbo.Uzsakymas.uzs_data, dbo.Uzsakymas.patvirt, dbo.Uzsakymas.DLogin,         dbo.Uzsakymas.KLogin, dbo.Uzsakymas.kita_info FROM    dbo.Uzsakymas INNER JOIN         dbo.numeriai ON dbo.Uzsakymas.KambarioNr = dbo.numeriai.KambarioNr </pre>		
kamb_patvirt																										
uzsakymas	smallint identity																									
KambarioNr	int																									
apartid	int																									
atvyk_data	datetime																									
isvyk_data	datetime																									
uzs_data	datetime																									
patvirt	tinyint																									
DLogin	varchar(10)																									
KLogin	varchar(10)																									
kita_info	text																									

### 3.13. Sistemos programinė realizacija

#### 3.13.1. Prototipinė viešbučių rezervavimo sistema

Prototipui kurti panaudotos technologijos:

- C#:
  - Windows Forms .NET
  - ADO.NET
- Microsoft .NET Framework v1.1
- Visual Studio .NET 2003
- Crystal Reports for .NET
- Microsoft Visio for Enterprise Architects 2003
- Microsoft SQL Server 2000

Sukurtas sistemos prototipas, kuris apima apie 90% sistemos funkcijų, sudaro 4553 kodo eilučių (1012 kliento aplinkos eilučių, 3541 darbuotojo aplinkos eilučių) ir *crystal reports* ataskaitos generavimas

Prototipą sudaro kambarių rezervavimo dalis ir administracinė dalis.

Prototipas buvo kuriamas pagal Z specifikaciją, kas laido sumažinti galimų klaidų kiekį.

### **3.13.2. Projektuojama viešbučių rezervavimo sistema**

Sistemos kliento daliai kurti panaudots technologijos:

- C#:
  - ASP .NET
  - ADO.NET
- IIS
- Microsoft .NET Framework v1.1
- Visual Studio .NET 2003
- Microsoft Visio for Enterprise Architects 2003
- Microsoft SQL Server 2000

Rezervavimo sistemą sudaro rezervavimo sistema interneto naršykle, rezervavimo sistema mobiliuoju telefonu ir rezervavimo sistemos administracinė sistema

Sistemos administravimo programai kurti panaudotos technologijos:

- C#:
  - Windows Forms .NET
  - Developer Express WindowsForms
  - ADO.NET
- Microsoft .NET Framework v1.1
- Visual Studio .NET 2003
- Developer Express Reports
- Microsoft Visio for Enterprise Architects 2003
- Microsoft SQL Server 2000

## 4. VARTOTOJO DOKUMENTACIJA

Sistemoje klientas gali peržiūrėti, ką jam siūlo viešbutis, registruotis, o prisiregistravęs peržiūrėti savo užsakymus, jei tokių yra. Sekretorė patvirtina kambarių ir/ar konferencijų salės užsakymą, rengia ataskaitas, kai užsakymas įvykdytas (klienats atvažiuoja ir susimoka) įrašo, kad užsakymas įvykdytas. Vadybininkas gali patvirtinti konferencijų salės užsakymus, o administratorius gali tvarkyti užsakymus, tvarkyti darbuotojų ir kleintų informaciją, tvarkyti kambarių ir paslaugų informaciją, įtraukti papildomas paslaugas.

### 4.1. Sistemos funkcinis aprašymas

Viešbučių rezervavimo sistema yra skirta tiek viešbučio dantuotojams, tiek viešbučio klientams.

Viešbučio klientai gali užsisakyti norimo tipo kambarį ar konferencijų salę dviem būdais internete: per kompiuterį ar mobilųjį telefoną.

Interneto naršykle pasiekiamos sistemos funkcijos:

- Prieš registruodamasis sistemoje vartotojas gali apžiūrėti kambarius su jų aprašais bei paveiksliukais demonstracinėje sistemos versijoje.
- Jei vartotojas nori užsisakyti kambarį, tai jam teks čia pat internete prisiregistruoti, užpildant visus registracijoje prašomus asmens duomenų laukus.
- Prisijungimas.
- Kabarario užsisakymas.
- Užsisakamo kabarario paveiksliuko peržūrėjimas.
- Nepatvirtinto užsakymo atšaukimas

Mobiliuoju telefonu pasiekiamos sistemos funkcijos:

- Prisijungimas,
- Kabarario užsakymas,
- Nepatvirtinto kabarario užsakymo atšaukimas.

Darbuotojų administracinės sistemos funkcijos:

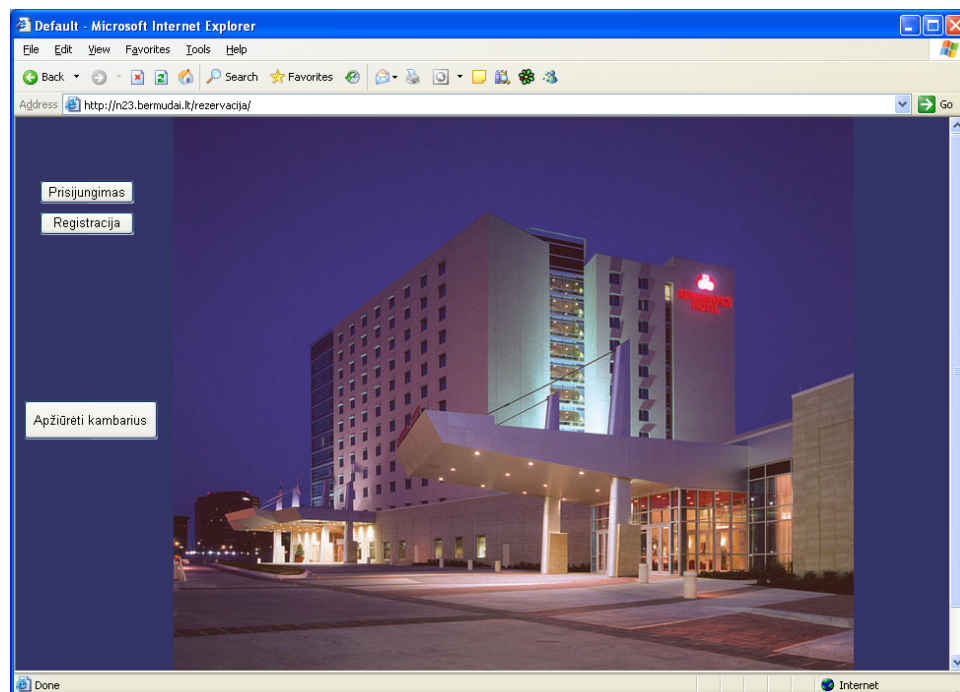
- Prisijungimas,
- Kambarių užsakymų patvirtinimas,
- Konferencijos salės užsakymų patvirinimas,

- Kambarių administravimas,
- Apartamentų tipų administravimas,
- Užsakymų administravimas,
- Klientų administravimas,
- Darbuotojų administravimas.
- Kiekviename lange filtravimo galimybės,
- Užsakymų ataskaitos generavimas pagal pasirinktus filtras,
- Klientų ataskaitos generavimas pagal pasirinktus filtras,
- Darbuotojų ataskaitos generavimas pagal pasirinktus filtras.

## 4.2. Rezervavimo sistemos kliento dalies vadovas

### 4.2.1. Prisijungimas prie sistemos interneto naršykle

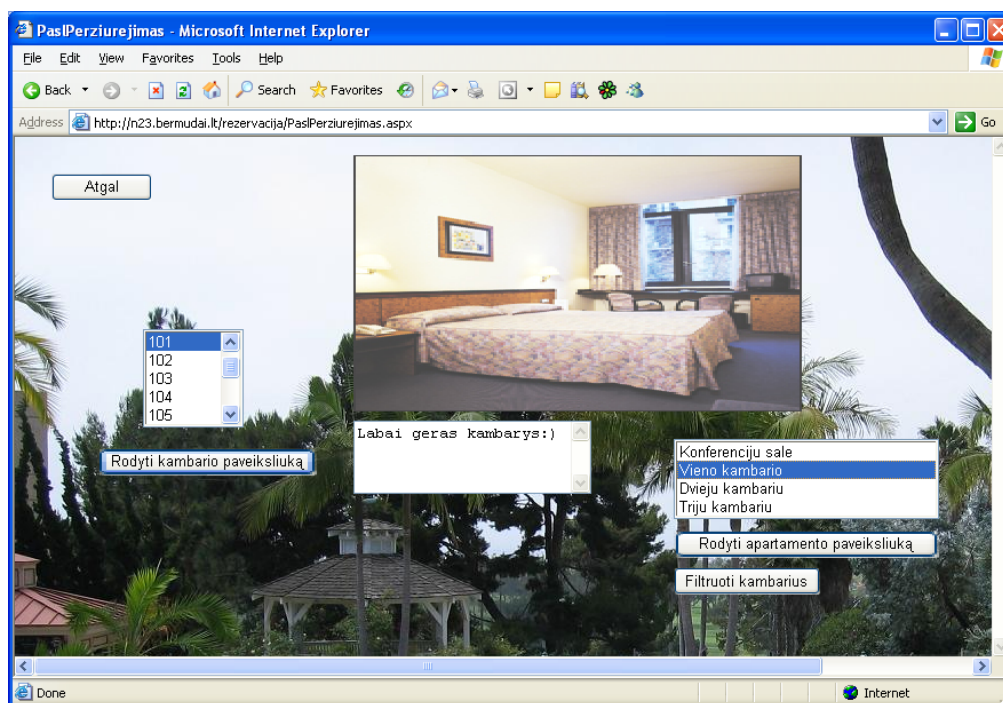
Suvedęs sistemos adresą internete, klientas pamato tokį prisijungimo puslapį:



**17 pav. Pagrindinis klientų prisijungimo puslapis**

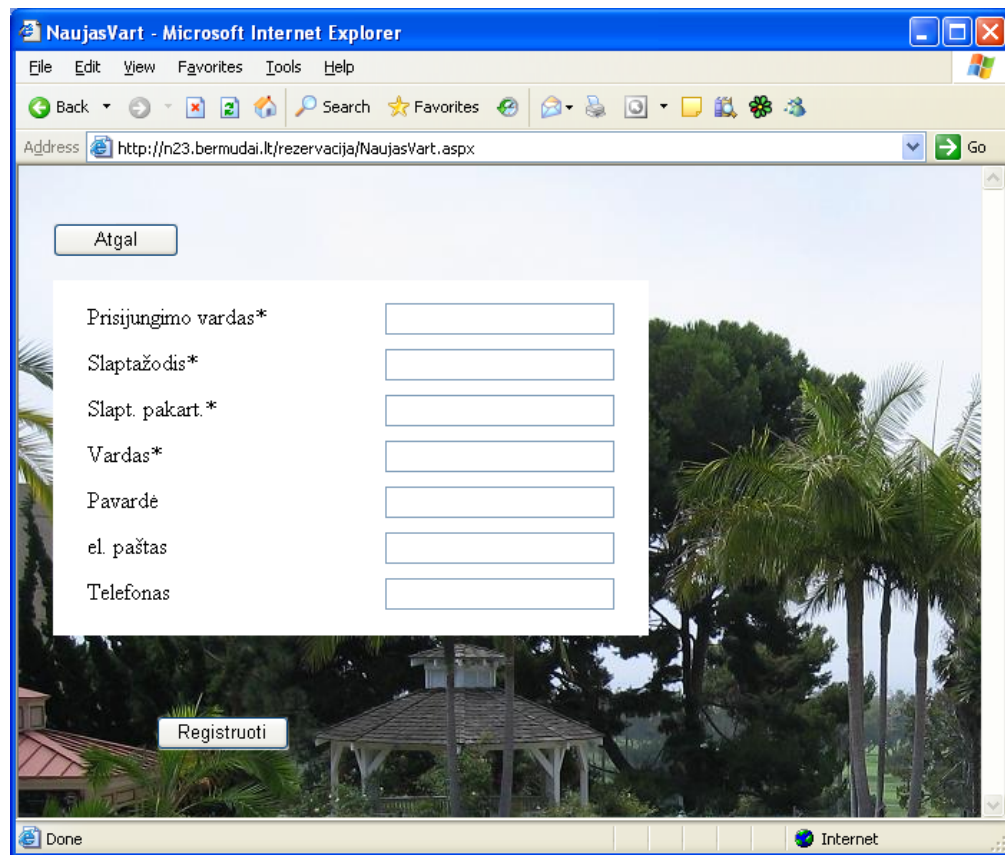
Galimi tokie veiksmai: jei vartotojas šiame puslapyje lankosi pirmą kartą ir jis nėra tikras, ar ši sistema atitiks jo poreikius, jis gali spausti mygtuką „Apžiūrėti kambarius“ ir tokiu būdu peržvelgti, ką viešbutis jam siūlo. Jei jis apsisprendžia, tuomet gali spausti mygtuką „Registracija“ ir atskirame lange suvesti sistemai reikalingus registracijos duomenis. Vėliau jis gali spausti mygtuką „Prisijungimas“ ir jungtis prie sistemos. Jei vartotojas šiame puslapyje jau

ne pirmą kartą, jis gali iškart jungtis prie sistemos ir kambarių peržiūrą bei užsakymų administravimą atlikti sistemos viduje:



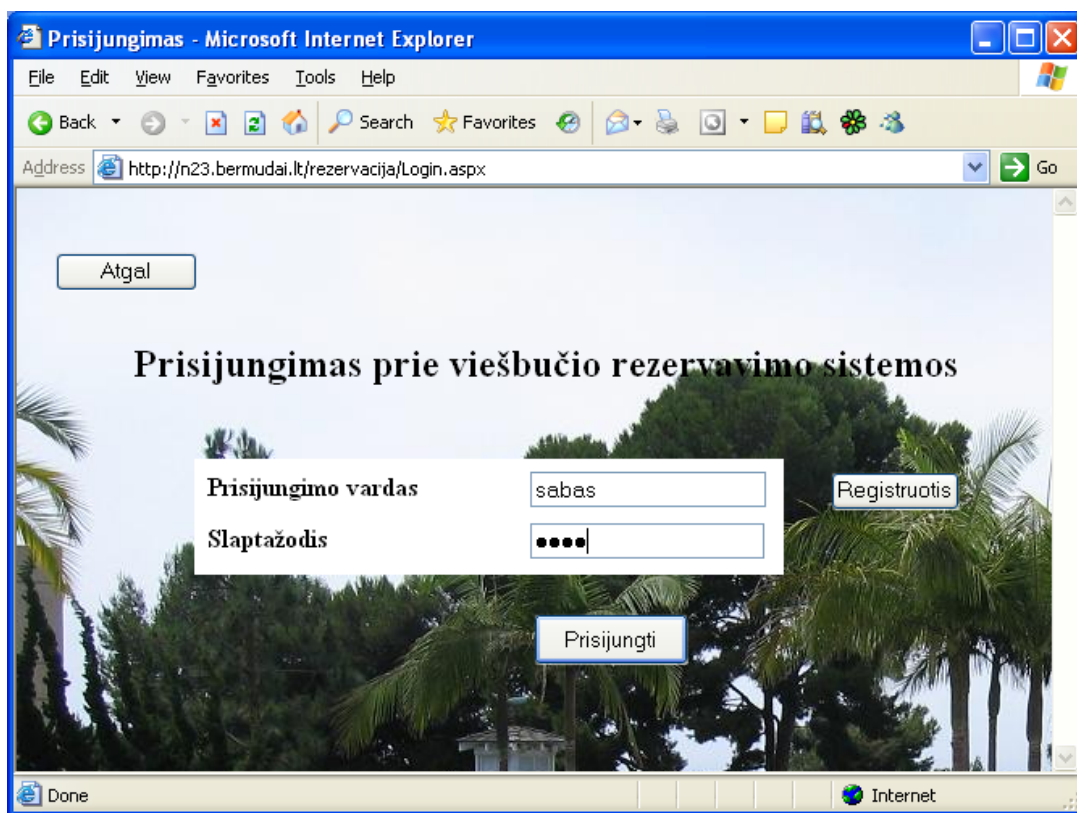
**18 pav. Kambarių apžiūrėjimas**

14pav. pavaizduotas sistemos vidus, kokį jį mato neprisiregistravęs klientas, norintis susipažinti su viešbučio siūlomais kambariais. Dešinėje jis gali pasirinkti, kokio tipo apartamento pageidauja: vieno kambario, dviejų kambarių, trijų, o galbūt konferencijų salės, taip pat paspaudęs mygtuką „Rodyti apartamento paveiksliuką“ jis gali pasižiūrėti, kaip atrodo šito tipo kambarys ar konferencijų salė. Paspaudęs mygtuką „Filtruoti kambarius“, kairėje klientas pamato sistemos jam atrinktus visus kambarius, turinčius paveiksliukus ir atitinkačius pasirinktą tipą. Čia klientas kairėje sukonkretina savo paiešką, pele pažymėdamas, kokį konkretų kambario numerį norėtų peržiūrėti. Paspaudęs mygtuką „Rodyti kambario paveiksliuką“, jis gali pamatyti konkretaus kambario nuotrauką bei paskaityti aprašą. Paspaudęs mygtuką „Atgal“ gali patekti į pradinį puslapį.



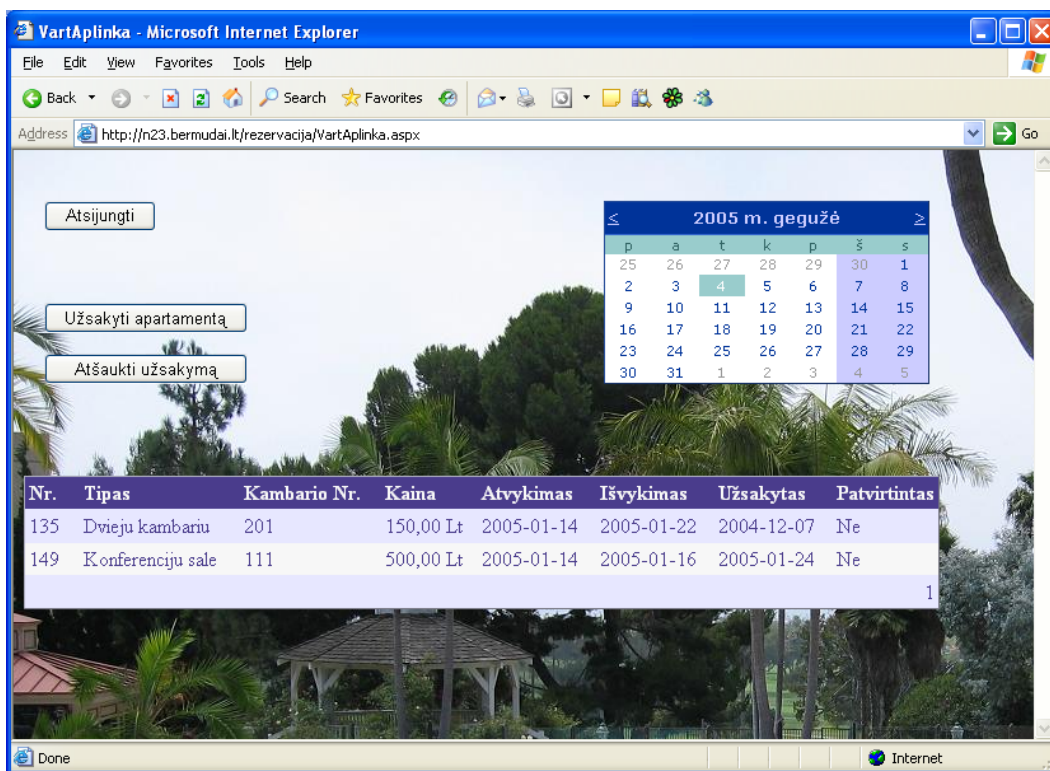
**19 pav. Naujo vartotojo registracija**

Šiame paveiksluke pavaizduota, kaip atrodo registracijos langas ir kokius duomenis besiregistruojantis klientas privalo pateikti sistemai. Žvaigždute yra pažymėti laukai, kuriuos būtina užpildyti. Jei vartotojas neužpildo visų reikiamų laukų ir spaudžia mygtuką „Registruoti“, tada yra parašomas pranešimas kad reikia užpildyti visus laukus, kurie pažymėti žvaigždute. Užpildęs duomenis, vartotojas spaudžia mygtuką „Registruoti“ ir nusiunčia savo duomenis į duomenų bazę. Prieš tai yra sutikrinami antras ir trečias slaptažodžių laukai, jei jie neatitinka, tai sistema praneša vartotojui, kad šie laukai nesutampa. Klientui tenka pataisyti duomenis ir vėl spausti mygtuką „Registruoti“.



**20 pav. Prisijungimas**

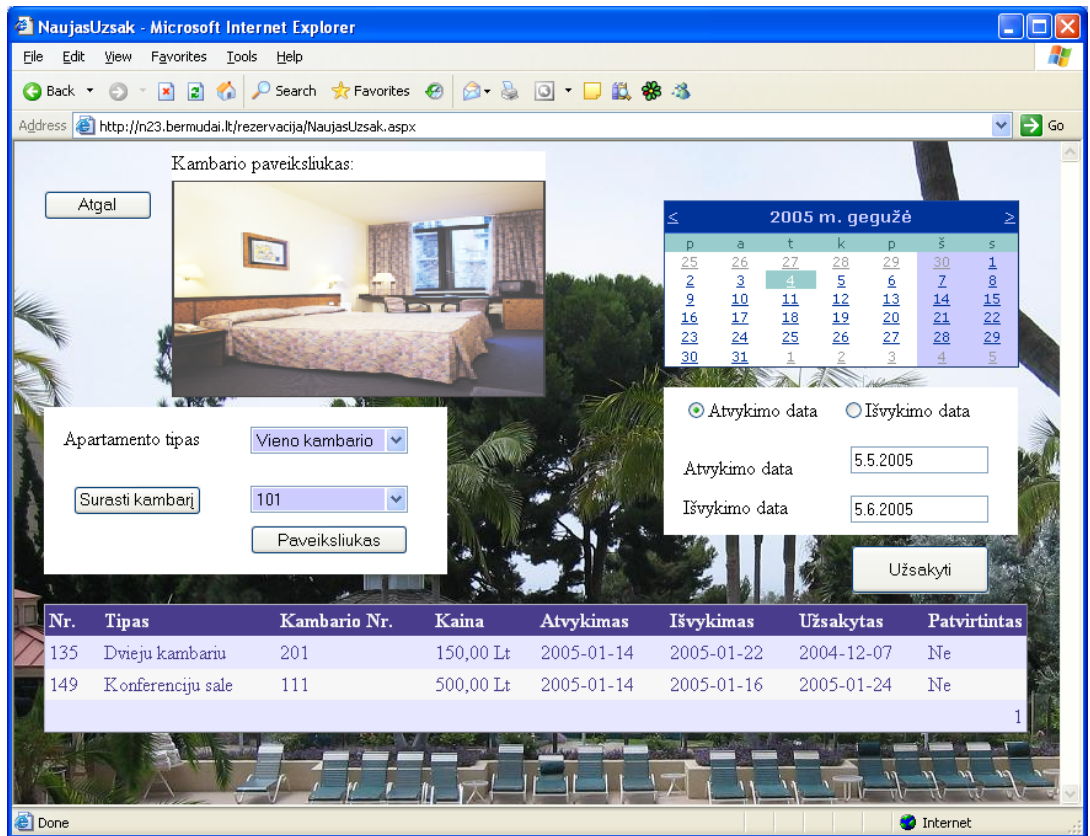
Jei klientas pasirenka prisijungimo prie sistemos mygtuką, jis yra nukreipiamas į prisijungimo langą. Čia suvedęs savo prisijungimo vardą ir slaptažodį bei paspaudęs mygtuką „Prisijungti“ patenka į sistemą. Neprisiregistravusio kliento patogumui yra pridėtas ir „Registruotis“ mygtukas, kuris nukreipia klientą į registracijos puslapį.



21 pav. Klientų užsakymų aplinka

Prisiregistravęs klientas papuola į užsakymų aplinką, kur savo patogumui mato kalendorių (pagalba planuojant kambario užsakymą konkrečiai datai). Taip pat šiame lange jis mato savo jau padarytus užsakymus bei turi galimybę papildomai užsisakyti apartamentą ar atšaukti užsakymą (kol šis dar nėra patvirtintas). Šias operacijas sistemoje jis gali atlikti paspaudęs mygtukus „Užsakyti apartamentą“ bei „Atšaukti užsakymą“.





22 pav. Naujas užsakymas

Čia yra užsakymo aplinka. Dešinėje esančiame kalendoriuje vartotojas turi galimybę nurodyti savo atvykimo ir išvykimo datą. Klientui datas pasirinkus, jos rodomos atitinkamuose laukeliuose. Jei klientą datos tenkina ir jis nepadarė klaidos, tuomet jam reikia pasirinkti apartamento tipą kairėje ir paspausti mygtuką „Surasti kambari“. Sistema suranda vartotojui prašomo tipo laisvus kambarius, leidžia pažiūrėti konkretaus kambario paveiksluką bei padaryti užsakymą dešinėje paspaudus „Užsakyti“ mygtuką.

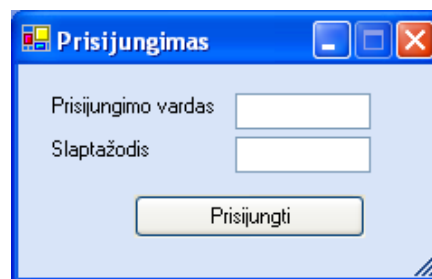
## 4.2.2. Prisijungimas prie sistemos naudojant WAP technologiją



23 pav. WAP prisijungimas.

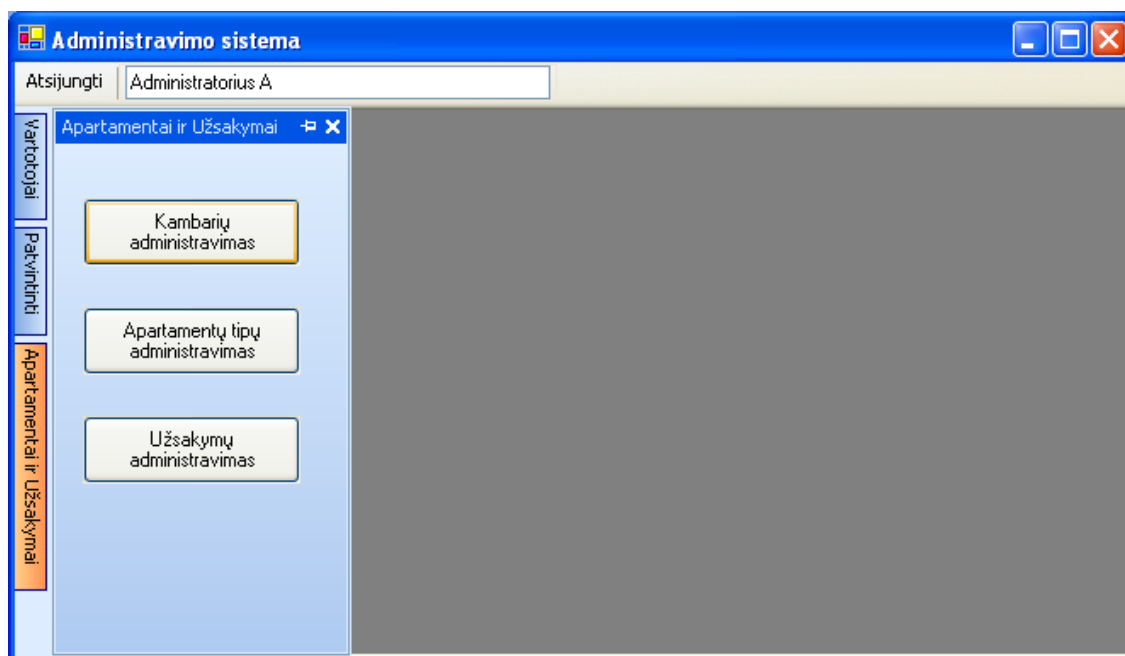
Prie sistemos galima prisijungti tik tuo atveju, jei jau yra vykdyta registracija internete. Tuomet su turimu prisijungimo vardu ir slaptažodžiu, vartotojas gali pasiekti sistemą mobiliuoju telefonu bei užsisakyti norimą kambarį.

## 4.3. Rezervavimo sistemos administravimo vadovas



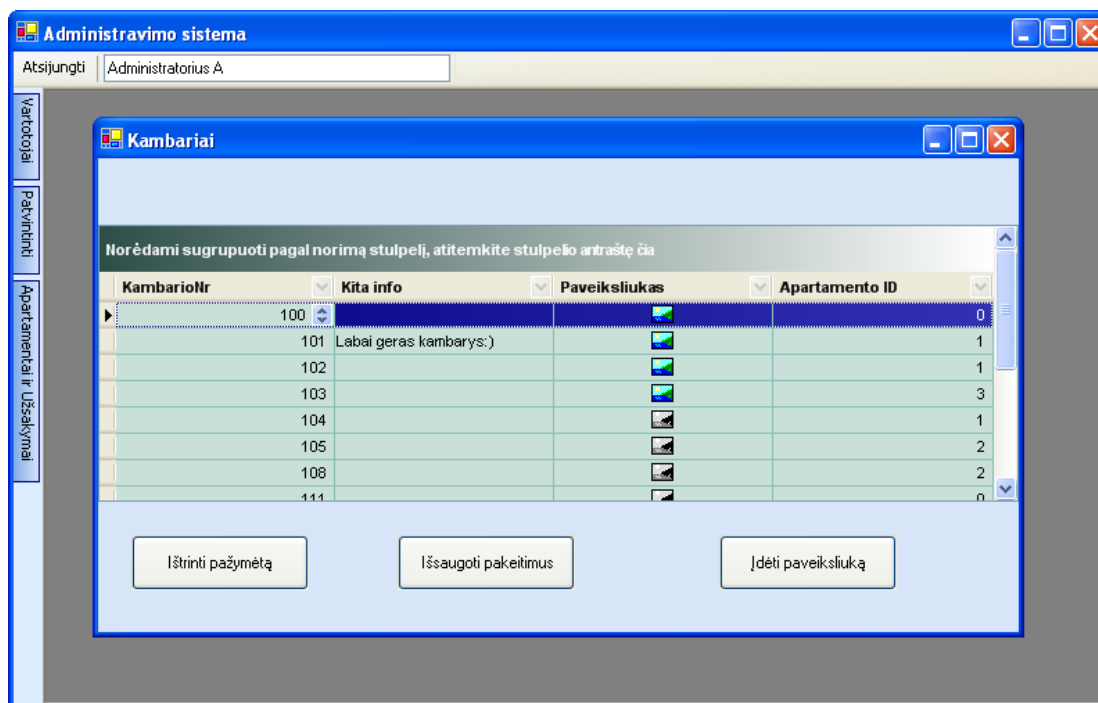
24 pav. Rezervavimo sistemos administracinės programos prisijungimas

Tokį prisijungimo langą mato sistemos administratorius. Kai jis suveda atitinkamus sistemos prašomus duomenis ir paspaudžia mygtuką „Prisijungti“, jis patenka į sistemą, kur gali atlikti visus administravimo darbus.



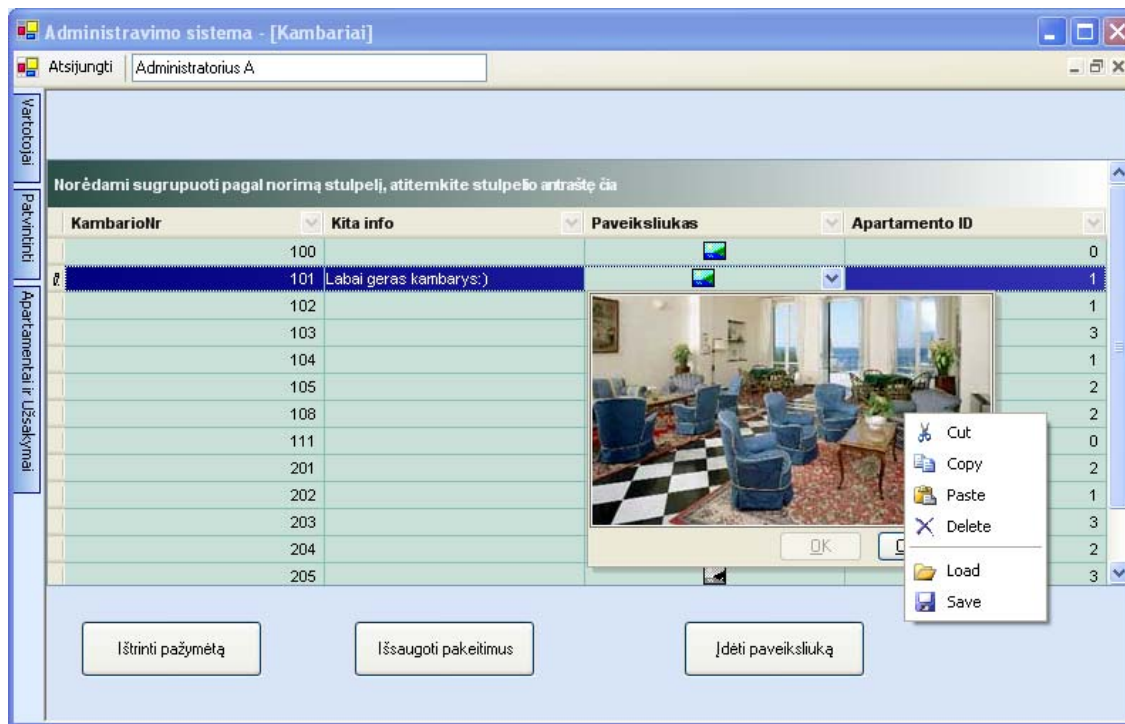
25 pav. Administravimo sistema

Taip atrodo administravimo sistema sėkmingai prisijungus. Užvedus pelę ant vieno iš mygtukų („Vartotojai“, „Patvirtinti“, „Apartamentai ir užsakymai“) atsidaro langas su funkciniais mygtukais. Paspaudus vieną iš mygtukų atsidaro naujas langas, kuris yra skirtas atlikti pasirinktai užduočiai.



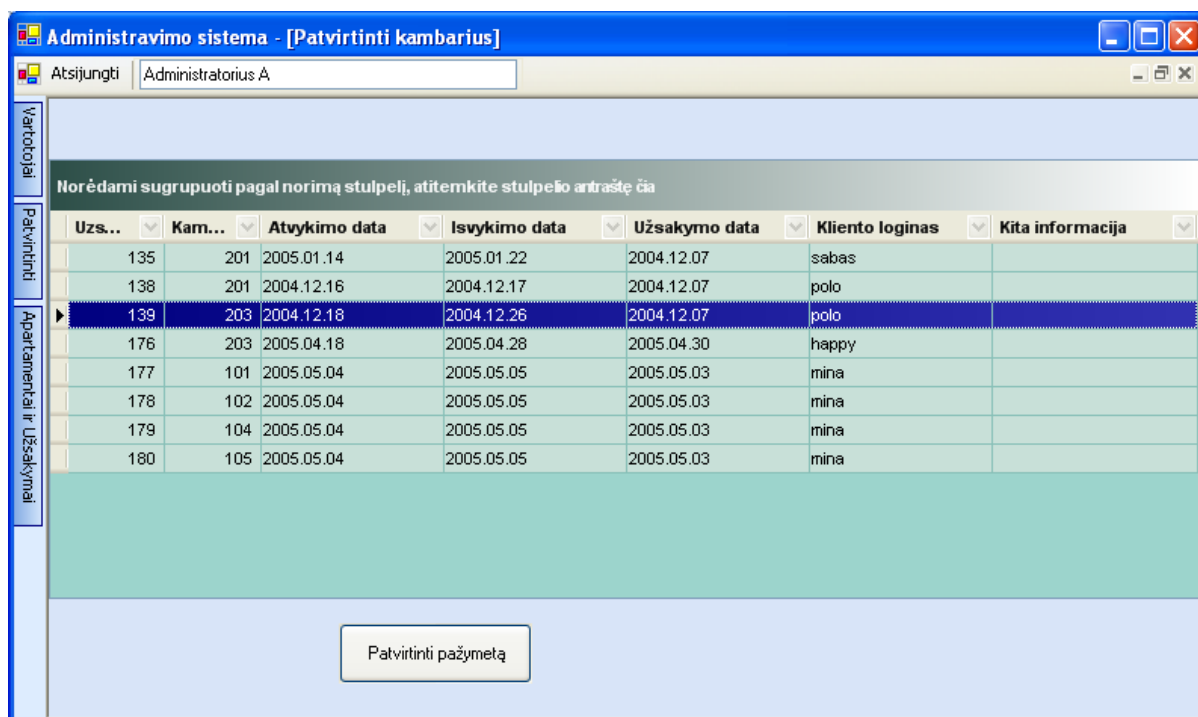
26 pav. Kambarių administravimo langas

Šiame lange yra matomi apartamentų duomenys, o mygtukų pavadinimai įvardina, kokias funkcijas administratorius gali atlikti: „Ištrinti pažymėtą“, „Išsaugoti pakeitimus“, „Įdėti paveiksluką“. Visos šios operacijos yra atliekamos su pažymėtu duomenų lauku.



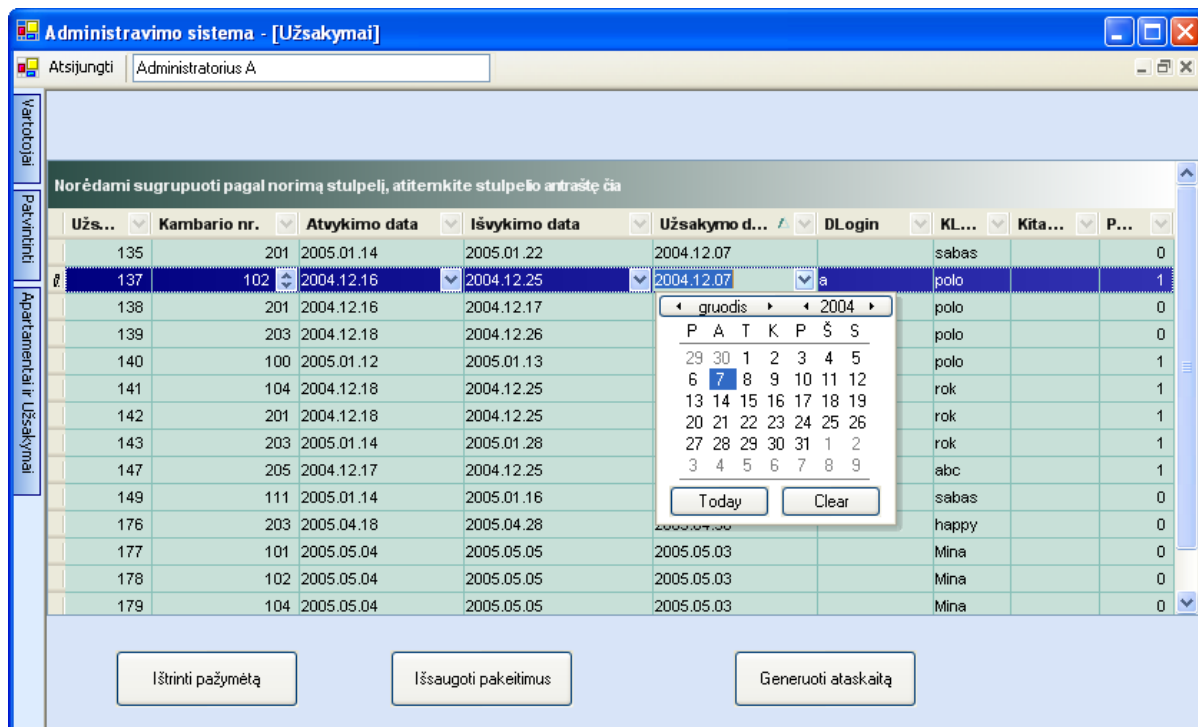
27 pav. Kambarių administravimo langas 2

Kambarių administravimo lange galima keisti kiekvieno kambario informaciją: priskirti, ištrinti ar nusikopijuoti kambario paveiksluką. Tokie patys paveikslukų priskyrimo veiksmai galimi ir apartamentų tipų administravimo dalyje.



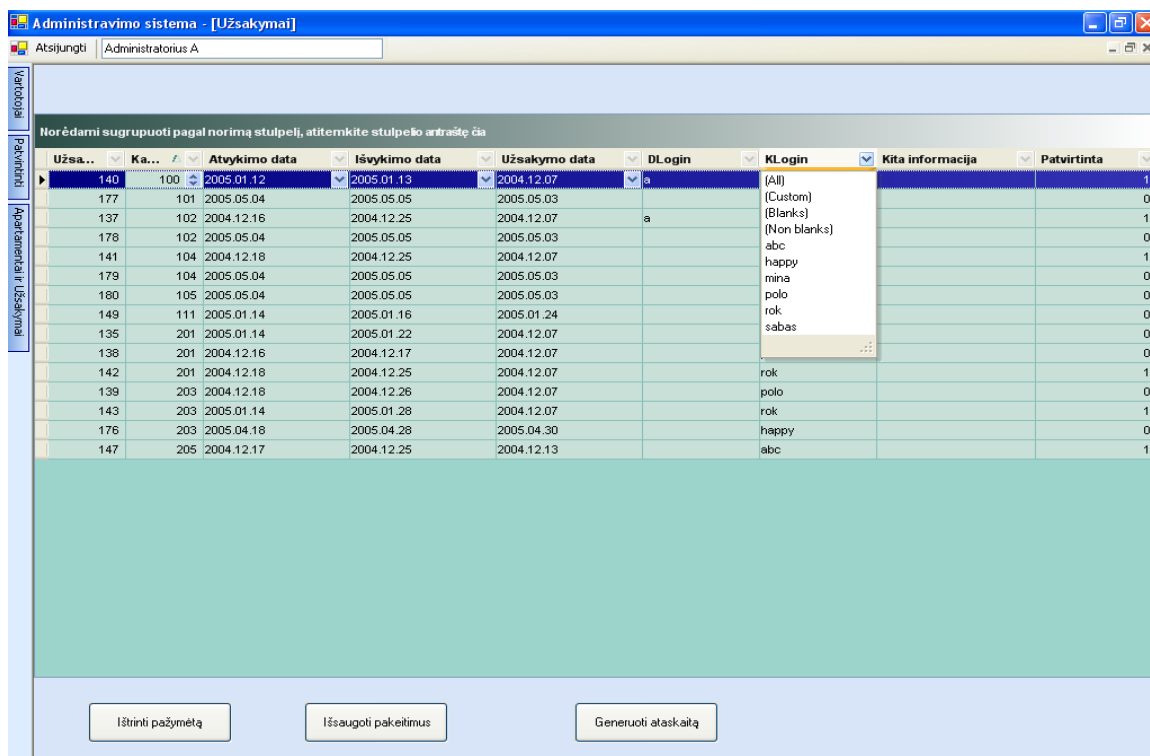
28 pav. Kambarių patvirtinimo langas

Kambarių patvirtinimo langas, čia galima patvirtinti pasirinktą kambarį, galimos įvairiausios filtravimo galimybės, kurios yra prie kiekvieno stulpelio. Kambarių patvirtinimo langas panašus į konferencijų salės patvirtinimo langą.



29 pav. Užsakymų administravimo langas

Užsakymų administravimo langas, čia galima administruoti atliktus užsakymus, galimi įvairiausi filtravimai (paspaudus stulpelio antraštėje esančią rodyklę, išlenda filtravimo kriterijaus pasirinkimas) ir pagal parinktus filtravimus ataskaitos generavimas:



30 pav. Užsakymų administravimas, filtravimo galimybės

Užsakymų ataskaita  
KLogin polo

Užsakymas	Kambario Nr	Atvykimo data	Išvykimo data	Užsakymo data	Patvirtinta	DLogin
140	100	2005.01.12 21:16:00	2005.01.13 21:16:00	2004.12.07 21:17:43	1	a
137	102	2004.12.16 21:16:00	2004.12.25 21:16:00	2004.12.07 21:17:04	1	a
138	201	2004.12.16 21:16:00	2004.12.17 21:16:00	2004.12.07 21:17:13	0	
139	203	2004.12.18 21:16:00	2004.12.26 21:16:00	2004.12.07 21:17:22	0	

Current Page No: 1      Total Page No: 1      Zoom Factor: 100%

**31 pav. Sugeneruota ataskaita pagal pasirinktus filtrus**

#### **4.4. Sistemos instaliavimo dokumentas**

Kad darbuotojas galėtų naudotis programomis, kompiuteryje turi būti instaliuotas .net framework 1.1, taip pat turi būti prijungtas prie interneto. Naudojamos operacinės sistemos gali būti: windows 98/2000/XP/2003.

Kad sistema veiktų reikia, kad serveriniame kompiuteryje būtų įdiegtas MS.NET Framework, IIS ir į IIS įdiegta ASP.NET viešbučių rezervavimo programos kodas.

ASP.NET Programos kodas į IIS instaliuojamas priskiriant direktorijai, kuriame yra šis kodas, internetini priėjimą (*Web sharing*) pasirinktu vardu.

## 5. PRODUKTO KOKYBĖS ĮVERTINIMAS

Buvo atlikta vartotojų reikalavimų patenkinimo analizė. Sistemos kokybę tyrė 3 ekspertai. Jie turėjo galimybę prisijungti prie sistemos 3 savaičių bėgyje, pateikti užsakymus ir pan. Kiekvieną savaitę vienas iš ekspertų dirbo kaip administratorius. Buvo vertinami funkciniai ir nefunkciniai reikalavimai.

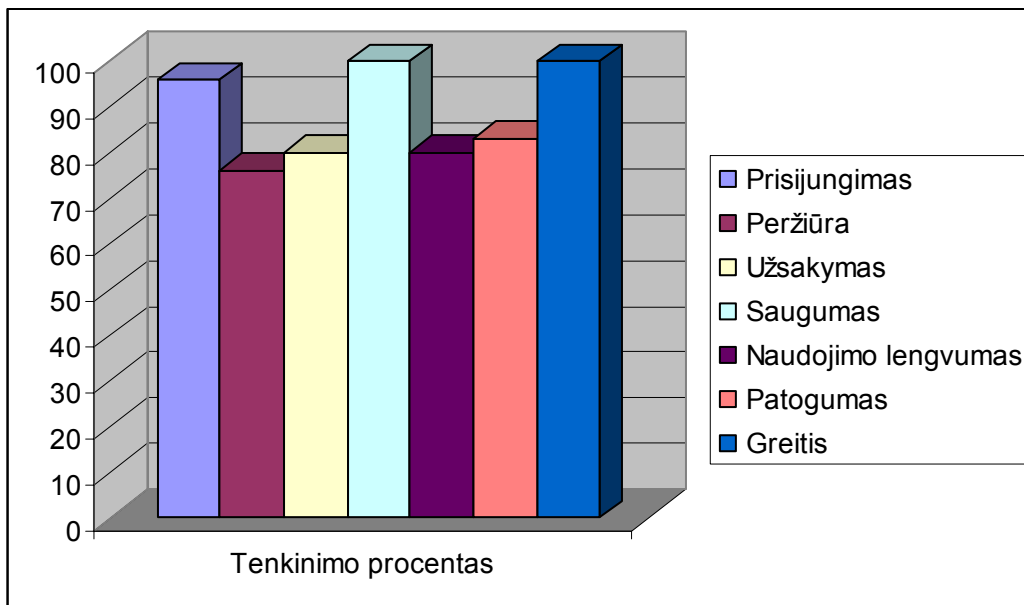
**9 lentelė. Produkto kokybės įvertinimas**

Reikalavimas	Ekspertas	Tenkinimo procentas
Prisijungimas	1 Renaldas Zajanckauskas	90
	2 Brigita Bagdonaitė	100
	3 Robertas Paužas	100
	<i>Vidurkis</i>	<i>96</i>
Peržiūra	1 Renaldas Zajanckauskas	80
	2 Brigita Bagdonaitė	70
	3 Robertas Paužas	80
	<i>Vidurkis</i>	<i>76</i>
Užsakymas	1 Renaldas Zajanckauskas	70
	2 Brigita Bagdonaitė	90
	3 Robertas Paužas	80
	<i>Vidurkis</i>	<i>80</i>
Saugumas	1 Renaldas Zajanckauskas	100
	2 Brigita Bagdonaitė	100
	3 Robertas Paužas	100
	<i>Vidurkis</i>	<i>100</i>
Naudojimo lengvumas	1 Renaldas Zajanckauskas	80
	2 Brigita Bagdonaitė	80
	3 Robertas Paužas	80
	<i>Vidurkis</i>	<i>80</i>
Patogumas	1 Renaldas Zajanckauskas	80
	2 Brigita Bagdonaitė	90
	3 Robertas Paužas	80
	<i>Vidurkis</i>	<i>83</i>
Greitis	1 Renaldas Zajanckauskas	100
	2 Brigita Bagdonaitė	100
	3 Robertas Paužas	100
	<i>Vidurkis</i>	<i>100</i>

Vertinime dalyvavo ekspertai:

Pavardė, vardas	Pareigos
Renaldas Zajanckauskas	IT magistrantas
Brigita Bagdonaitė	IT magistrantė
Robertas Paužas	Vadybininkas





**32 pav. Kokybės produkto įvertinimo grafikas**

## 6. IŠVADOS

- Remiantis panašių sistemų ir vartotojų poreikių analize, suformuluota viešbučių informacinės sistemos kūrimo užduotis, apimanti vartotojų registravimo, užsakymų sudarymo, įvairių tipų informacijos peržiūros ir tvarkymo funkcijas.
- Norint užtikrinti korektišką sistemos veikimą ir išvengti klaidų kūrimo metu, pagrindinės funkcijos buvo atidžiai išnagrinėtos ir formalizuotos Z kalba. Formalus aprašymas buvo realizuotas pasinaudojus „SUM“ programa.
- Pasirinktos .NET Framework realizavimo priemonės užtikrina reikiamus sistemos kokybės kriterijus:
  - palaiko saugią daugelio vartotojų veiklą
  - turi galimybę veikti internete, vietiniame tinkle ir mobiliuosiuose įrenginiuose
  - pritaikomumą kitos paskirties sistemoms
  - plečiamumą – sistemą nesunku plėsti
- Suprojektuota ir sugeneruota MS SQL 2000 duomenų bazė naudojant Microsoft Visio for Enterprise Architects 2003 (*Database Model Diagram*).
- Sukurtas sistemos prototipas, kuris apima apie 90% sistemos funkcijų, sudaro 4553 kodo eilučių (1012 kliento aplinkos eilučių, 3541 darbuotojo aplinkos eilučių) ir *crystal reports* ataskaitos pavyzdys.
- Rezervavimo sistemą sudaro viešbučių rezervavimo sistema (ASP.NET), viešbučių rezervavimo sistema mobiliuoju telefonu ir viešbučių rezervavimo sistemos administravimo programa (Windows Forms.NET, DevExpress)
- Sistema suteiks galimybes interneto vartotojams gauti informaciją apie viešbučių paslaugas ir jomis pasinaudoti; pagerins užsakymų aptarnavimą; padidins viešbučių klientų skaičių; palengvins administratoriaus darbą.

## 7. LITERATŪRA

1. MacDonald, R. Introducing ADO.NET. Iš *DNJ online* [interaktyvus]. 2001 [žiūrėta 2004-9-20]. Prieiga per internetą: <[http://www.dnjonline.com/article.aspx?ID=iss22\\_essentials](http://www.dnjonline.com/article.aspx?ID=iss22_essentials)>
  2. MacDonald, R. The architecture of ADO.NET. Iš *DNJ online* [interaktyvus]. 2001 [žiūrėta 2004-9-20]. Prieiga per internetą: <[http://www.dnjonline.com/articles/essentials/iss22\\_essentials\\_architecture.html](http://www.dnjonline.com/articles/essentials/iss22_essentials_architecture.html)>
  3. Microsoft Corporation. Microsoft Visual C# Developer Center 2005 m. – [žiūrėta 2005-02-20]. Prieiga per internetą: <<http://msdn.microsoft.com/vcsharp/productinfo/features/default.aspx>>
  4. Microsoft Corporation. Microsoft Visual C# Developer Center 2005 m. System Requirements for Visual Studio .NET 2003 – [žiūrėta 2005-02-20]. Prieiga per internetą: <<http://msdn.microsoft.com/vstudio/productinfo/sysreqs/default.aspx>>
  5. Microsoft Corporation. Microsoft Visual C# Developer Center 2005 m. Product Overview for Visual Studio .NET 2003– [žiūrėta 2005-02-20]. Prieiga per internetą: <<http://msdn.microsoft.com/vstudio/productinfo/overview/default.aspx>>
- Lam H., Thai T., *Dot.NET.Framework.Essentials.3rd.Edition*. O'Reilly, 2003. 384p.
6. Sekliuckis V., Gudas S., Garšva G. *Informacijos sistemos ir duomenų bazė: vadovėlis*. K.: Technologija, 2003. 338p.

## 8. PRIEDAI

### 8.1. Sukurtos saugomos procedūros

```
CREATE PROCEDURE loginas
@login varchar(10), @pass varchar(10), @ret int=0 out
AS
select @ret = count(KLogin)
from KLogin
where KLogin=@login and Kpass=@pass
GO
```

```
CREATE PROCEDURE loginasD
@login varchar(10), @pass varchar(10), @ret int=0 out
AS
if ((select count(DLogin) from Darbuotojai where DLogin=@login and Dpass=@pass)=1)
set @ret = (select Lygis from Pareigos where pareigosID= (select pareigosID from Darbuotojai
where DLogin=@login and Dpass=@pass) )
else set @ret=-1
GO
```

```
CREATE PROCEDURE nepatvirtinti
AS
select uzsakymas"Uzsakymas", KambarioNr"Kambario nr.", atvyk_data"Atvykimo data",
isvyk_data"Isvykimo data", uzs_data"Uzsakymo data", KLogin"Kliento loginas", kita_info"Kita
informacija"
from kamb_patvirt
where patvirt =0 and apartid!=0
GO
```

```
CREATE PROCEDURE nepatvirtinti_konf
AS
select uzsakymas"Uzsakymas", KambarioNr"Kambario nr.", atvyk_data"Atvykimo data",
isvyk_data"Isvykimo data", uzs_data"Uzsakymo data", KLogin"Kliento loginas", kita_info"Kita
informacija"
from kamb_patvirt
where patvirt =0 and apartid=0
GO
```

```
CREATE PROCEDURE patvirtint
@uz varchar(10), @dlogin varchar(10)
AS
update kamb_patvirt set patvirt=1, DLogin=@dlogin where uzsakymas=@uz and apartid!=0
GO
```

```
CREATE PROCEDURE patvirtint_konf
@uz varchar(10), @dlogin varchar(10)
```

```
AS
update kamb_patvirt set patvirt=1, DLogin=@dlogin where uzsakymas=@uz and apartid=0
GO
```

```
CREATE PROCEDURE rezerv
@data1 smalldatetime, @data2 smalldatetime, @apid int
AS
select KambarioNr from numeriai
where apartid = @apid and KambarioNr not in
(select distinct KambarioNr
from rezervuoti_kamb
where apartid = @apid and ( (@data1 between atvyk_data and isvyk_data) or (@data2 between
atvyk_data and isvyk_data)
or (atvyk_data between @data1 and @data2) or (isvyk_data between @data1 and @data2) ))
GO
```

```
CREATE PROCEDURE surastiuzsakymus
AS
select uzsakymas"Uzsakymas",KambarioNr, atvyk_data, isvyk_data, uzs_data"Uzsakymo
data",patvirt"Patvirtinta",DLogin, KLogin ,kita_info"Kita info" from Uzsakymas
GO
```

```
CREATE PROCEDURE uzsak_rez
@login varchar(10), @atvyk smalldatetime, @isvyk smalldatetime, @kambnr smallint
AS
if ((select count (*) from Uzsakymas
where (@kambnr=KambarioNr) and (((@atvyk between atvyk_data and isvyk_data) or (@isvyk
between atvyk_data and isvyk_data)
or (atvyk_data between @atvyk and @isvyk) or (isvyk_data between @atvyk and @isvyk) ))
=0)
insert into Uzsakymas(KambarioNr, atvyk_data, isvyk_data, uzs_data, KLogin)
values (@kambnr, @atvyk, @isvyk, GETDATE(), @login)
GO
```

```
CREATE PROCEDURE uzsakymai
@login varchar(10)
AS
select uzsakymas"Nr.", apart"Tipas", KambarioNr "Kambario Nr.", ap_kaina"Kaina", atvyk_data
"Atvykimas", isvyk_data "Išvykimas", uzs_data "Užsakytas",Patvirtintas
from uzsak_w
where KLogin =@login
GO
```

```
CREATE PROCEDURE uzs_trinti
@nr int
AS
DELETE FROM Uzsakymas WHERE uzsakymas =@nr
GO
```