

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
KOMPIUTERIŲ KATEDRA

Andrius Dmuchovskis

**Protingos namų informacijos valdymo sistemos
sudarymas ir tyrimas**

Magistro darbas

Darbo vadovas
prof. dr. E. Kazanavičius

Kaunas, 2012

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
KOMPIUTERIŲ KATEDRA

Andrius Dmuchovskis

**Protingos namų informacijos valdymo sistemos
sudarymas ir tyrimas**

Magistro darbas

Recenzentas

doc. dr. I. Mikuckienė
2012-05-28

Vadovas

prof. dr. E. Kazanavičius
2012-05-28

Atliko

IFM 0/1 gr. stud.

Andrius Dmuchovskis
2012-05-28

Kaunas, 2012

Turinys

1	ĮŽANGA	8
2	PROBLEMINĖS SRITIES ANALIZĖ.....	11
	2.1 Darbo objekto aprašymas	11
	2.2 Protingų būsto informacijos valdymo sistemų analizė.....	13
	2.3 Programinių agentų analizė.....	18
	2.3.1 Programinio agento aprašymas.....	18
	2.3.2 Programinių agentų, naudojamų protingose būsto informacijos valdymo sistemose, analizė	20
	2.3.3 Programinių agentų palyginimas ir pasirinkimas	22
	2.4 Agentų kūrimo programinės įrangos analizė.....	25
	2.5 Išvados.....	32
3	PROTINGOS NAMŲ INFORMACIJOS VALDYMO SISTEMOS MODELIS	33
	3.1 Protingos būsto informacijos valdymo sistemos funkcinis modelis	33
	3.2 Sistemos struktūra	35
	3.3 Duomenų mainų tarp sistemos agentų modelis.....	38
	3.4 Rekomendacijų agento darbo modelis	44
	3.5 Išvados.....	46
4	EKSPERIMENTINIS TYRIMAS	47
	4.1 Programinių agentų realizavimas sistemoje.....	47
	4.2 Rekomendacijų agento darbo tyrimas	48
	4.3 Sistemos realizacija	53
	4.4 Sistemos tobulinimo ir plėtros rekomendacijos	57
	4.5 Išvados.....	58
5	IŠVADOS	59
6	TERMINŲ IR SANTRUPMŲ ŽODYNAS	60
7	LITERATŪROS SĄRAŠAS	61

LENTELIŲ SĄRAŠAS

1 lentelė. Protingų būsto informacijos sistemų aspektų palyginimas.....	17
2 lentelė. Protingų būsto informacijos sistemos agentų palyginimas.....	24
3 lentelė. Agentų kūrimo programinės įrangos palyginimas.....	31
4 lentelė. Duomenų bazės lentelių aprašymai.....	38
5 lentelė. Šilumos kainų prognozių metodų palyginimas.....	50

PAVEIKSLŲ SĄRAŠAS

1 pav. Protingos būsto informacijos valdymo sistemos kontekstinė diagrama.....	12
2 pav. Agentų klasifikacijos pradinės savybės [13, 15].....	19
3 pav. Protingos būsto informacijos valdymo sistemos agentų bendravimo schema.....	23
4 pav. Agentų kūrimo programinės įrangos kalbų pasiskirstymo diagrama [29].....	26
5 pav. Agentų kūrimo programinės įrangos licencijų tipai [29].....	27
6 pav. Agentų kūrimo programinės įrangos operacinės sistemos [29].....	28
7 pav. Agentų kūrimo programinės įrangos taikymo sritys [29].....	28
8 pav. Agentų kūrimo programinės įrangos palaikymas [29].....	29
9 pav. Sistemos funkcinis modelis.....	34
10 pav. Sistemos struktūra.....	36
11 pav. Sistemos duomenų bazės loginė schema.....	38
12 pav. Sistemos paleidimo ir klaidų aptikimo sekos diagrama.....	40
13 pav. Orų prognozių gavimo sekos diagrama.....	41
14 pav. Šildymo stebėjimo ir valdymo sekos diagrama.....	42
15 pav. Rekomendacijų gavimo sekos diagrama.....	43
16 pav. Būsto suvartojamos šildymo energijos kiekio apskaičiavimo formulė [34].....	44
17 pav. Energijos suvartojimas šildant patalpas atsižvelgiant į šildymo sistemos efektyvumą ir šilumos netektis dėl nešildomų zonų [34].....	44
18 pav. Pastato komponentų atiduodamos šilumos apskaičiavimo formulė [34].....	45
19 pav. Individualaus pastato komponento šilumos netekties apskaičiavimo formulė.....	45
20 pav. Ryšiai tarp pagrindinių JADE architektūros elementų [14].....	47
21 pav. Šildymo tikrosios ir prognozuojamų kainų grafikas.....	49
22 pav. Pasirinkto šildymo kainų prognozių metodo skaičiavimo formulė.....	49
23 pav. Šilumos kainų prognozių metodo tikslumo priklausomybė nuo temp. prognozės tikslumo.....	50
24 pav. Rekomendacijų puslapis.....	51
25 pav. Rekomendacijos metodo parametrų keitimo puslapis.....	52
26 pav. Rekomendacijų rezultatų puslapis.....	52
27 pav. „Microsoft Exel“ ir protingos būsto informacijos valdymo rekomendacijų skaičiavimų rezultatų diagrama.....	52
28 pav. JADE „sniffer“ programos lango fragmentas, vaizduojantis siunčiamas agentų žinutes rekomendacijų teikimo metu.....	53
29 pav. Sistemos pradinis puslapis.....	54
30 pav. JADE „sniffer“ programos lango fragmentas, vaizduojantis siunčiamas agentų žinutes orų prognozės teikimo metu.....	55

31 pav. Šildymo puslapis	55
32 pav. JADE „sniffer“ programos lango fragmentas, vaizduojantis siunčiamas agentų žinutes šildymo stebėjimo metu	56
33 pav. Programos komandinis langas su klaidų aptikimo agento spausdinamomis žinutėmis	56

SANTRAUKA

Sparčiai augant modernių būstų skaičiui, taip pat sparčiai auga šių kompiuterizuotų būstų informacijos valdymo sistemų poreikis. Šios sistemos įvairiais jutikliais renka informaciją apie būsto patalpų temperatūrą, drėgmę ir kt. Ši informacija dažniausiai yra kaupiama skirtingais būdais atskirose posistemėse, todėl vartotojui nėra paranku stebėti kiekvienos posistemės duomenis.

Informacijos valdymo sistemos leidžia vartotojui stebėti visą savo kompiuterizuoto būsto sistemų informaciją vienoje vartotojo sąsajoje. Informacijos valdymo sistemos, panaudodamos esamus būsto informacijos sistemų duomenis, gali pateikti vartotojui išteklių naudojimo rekomendacijas, atsižvelgdamos į vartotojo suformuotus reikalavimus – paskaičiuoti mėnesio šildymo kainą vartotojui nurodant norimą patalpos temperatūrą įvertinant išteklių kainas.

Šio darbo objektas – informacijos valdymo sistema įterpta į kompiuterizuoto būsto sistemą. Darbo tikslas – sukurti protingos namų informacijos valdymo sistemos prototipą. Sistemą sudaro programiniai agentai, kurie pagal vartotojo poreikius sugeba surinkti ir pateikti informaciją vartotojui ir pagal tai priimti sprendimus. Darbe tiriamos programinių agentų savybės, informacijos, kurią naudoja agentai, surinkimo metodai, jų (agentų) klasifikacija ir tipologija, taikymas informacijos valdymo sistemose. Darbas susideda iš probleminės srities analizės, sistemos modelio ir eksperimentinio tyrimo.

SUMMARY

With the rapid growth of automated houses, there is same rapid growth of these houses information management systems needs. Systems with various sensors gather information about zones temperature, humidity etc., information about other systems resources (e.g. heating services, electricity prices) is also gathered. Most of the time this information is being saves by different methods in separate systems, and that is not very handy for a user to track all data from different systems sources.

Information management systems let the user to see all his automated house systems information in one graphical user interface. Information management systems, using existing house information systems data, can provide user resource usage recommendations, considered with users needs. E.g. calculate months heating costs, when user provides desired zone temperature, using resource prices to achieve this goal.

Master thesis object – information management system, working inside home automation system. Master thesis goal – to create smart house information managent systems prototype. System consists of software agents, which, according to users' needs, can gather and deliver information for user. Master thesis studies software agent features, agent information gathering methods, their (agents) classification and typology, usage in information management systems. Master thesis consists of three parts: problem analysis, system model and experiment.

1 IŽANGA

Darbo aktualumas. Šiuo metu ypač daugėja sričių, kuriose taikomos informacinės technologijos. Sparčiau populiarėja ir jų (informacinių technologijų) taikymas būstuose, todėl didėja ir kompiuterizuotų namų poreikis. Tai ypač gerina gyvenimo namuose kokybę. Augant kompiuterizuotų būstų skaičiui, taip pat sparčiai auga šių kompiuterizuotų būstų informacijos valdymo sistemų poreikis. Sistemos įvairiais jutikliais (angl. sensor) renka informaciją apie būsto zonų temperatūrą, drėgmę ir kt. Ši informacija dažniausiai yra kaupiama skirtingais būdais atskirose sistemose, todėl vartotojui nėra patogiu stebėti ir analizuoti kiekvienos atskiros sistemos duomenis. Dėl to atsiranda poreikis turėti protingas namų informacijos valdymo sistemas, kurios sujungia visų būsto sistemų duomenis ir juos apdorojusios centralizuotai pateikia vartotojui patogią formą (grafinės vartotojo sąsajos dėka).

Darbo naujumas. Šiuo metu rinkoje yra daug įvairių protingų būsto informacijos valdymo sistemų, dauguma jų atlieka panašaus pobūdžio funkcijas ir uždavinius, tokius kaip apšvietimo valdymas, šilumos reguliavimas, drėgmės stebėjimas ir kt. Dažniausiai šios sistemos tarpusavyje skiriasi palaikomų funkcijų gausa, jų valdymo sudėtingumu. Šių sistemų informacijos valdymas nėra aukšto lygio ir dažniausiai sistemos teikia tik įvairių išteklių suvartojimo diagramas, kas ne visada tenkina vartotojų poreikius priimant sprendimus. Kuriama sistema ne tik kaupia namų duomenis, bet ir analizuoja turimą informaciją – teikia šildymo sezono kaštų prognozes.

Sistemos „sumanumą“ (autonomiškumą, gebėjimą priimti sprendimus) galima pasiekti įvairiais būdais: panaudojant miglotąją logiką, neuroninius tinklus ar programinius agentus. Darbe pasirinktas sistemos „sumanumo“ įgyvendinimo būdas – programiniai agentai, dėl lengvo ir paprasto jų pritaikymo prie esamos sistemos, kuri naudoja programinius agentus JADE platformos aplinkoje. Protingose namų informacijos sistemose programiniai agentai nėra pakankamai dažnai naudojami, ir jų panaudojimas dažniausiai apsiriboja vienu agentu, kuris veikia tvarkaraščio principu (tam tikrais laiko intervalais atlieka užduotis, pvz. patikrina kambario temperatūrą). Kuriama sistema skiriasi tuo, kad ji naudoja ne vieną agentą, o daug tarpusavyje bendraujančių komandinių agentų, kurie bendrai atlieka įvairias užduotis – kuriama sistema yra daugiaagentė sistema. Vienas daugiaagentė sistemų privalumų yra paprastas plečiamumas ir atnaujinimas: į sistemą, nepertraukiant jos darbo, galima lengvai pridėti naujų ar atnaujintų programinių agentų su patobulintu funkcionalumu.

Darbo nagrinėjamos problemos. Kadangi kompiuterizuotų namų informacija dažniausiai yra renkama ir saugoma skirtingais būdais atskirose sistemose, vartotojui ir

sistemų kūrėjams kyla sunkumų norint peržiūrėti ir išanalizuoti atskirų sistemų duomenis. Informacijos valdymo sistemos leidžia vartotojui stebėti visą savo kompiuterizuoto būsto sistemų informaciją vienoje vartotojo sąsajoje.

Kita problema yra pats turimas informacijos panaudojimas, t.y. kokią naudą gali vartotojas gauti, panaudodamas savo turimus būsto duomenis. Informacijos valdymo sistemos, panaudodamos esamus būsto informacijos posistemų duomenis, gali pateikti vartotojui išteklių naudojimo sprendimus, atsižvelgdamos į vartotojo suformuotus reikalavimus – paskaičiuoti mėnesio šildymo kainą, kai pasirenkama patalpos temperatūra, įvertinant išteklių kainas.

Darbo objektas, tikslas ir uždaviniai. Darbo objektas – informacijos valdymo sistema įterpta į būsto kompiuterizavimo sistemą, sugebanti atlikti įvairias informacijos rinkimo ir apdorojimo funkcijas. Darbo tikslas – sukurti protingą namų informacijos valdymo sistemą ir atlikti jos tyrimą. Sistemą sudaro programiniai agentai, kurie pagal vartotojo poreikius sugeba surinkti ir pateikti informaciją vartotojui. Darbe tiriamos programinių agentų savybės, informacijos, kurią naudoja agentai, surinkimo metodai, jų kūrimo ir taikymo galimybės. Šiam tikslui pasiekti, reikia:

- atlikti protingų būsto informacijos valdymo sistemų analizę;
- atlikti programinių agentų, naudojamų šiose sistemose, analizę;
- sudaryti sistemos modelį, paremtą programinių agentų sistemos principu;
- atlikti informacijos valdymo sistemos modelio tyrimą ir jo įvertinimą;
- eksperimentiškai atlikti sistemos prototipo realizaciją;
- eksperimentiniu būdu atlikti sistemos funkcionavimo tyrimą ir įvertinimą.

Darbo dalių aprašymas. Darbas susideda iš probleminės srities analizės, protingos namų informacijos valdymo sistemos modelio ir eksperimentinio tyrimo. Probleminės srities analizė apima darbo objektą ir darbo probleminę sritį. Išanalizuotos esamos protingos būsto informacijos valdymo sistemos. Taip pat išanalizuoti programiniai agentai, jų klasės ir tipai, savybės, panaudojimas informacijos valdymo sistemose. Dar šioje dalyje atlikta programinių agentų kūrimo įrankių analizė.

Protingos namų informacijos valdymo sistemos modelio dalyje pateiktas problemos sprendimo modelis ir metodas. Sudaryti sistemos ir joje esančių programinių agentų modeliai.

Aprašytas rekomendacijų agento darbo modelis. Aprašyti duomenų mainai tarp atskirų protingos būsto informacijos valdymo sistemos agentų. Aprašyta sistemos struktūra.

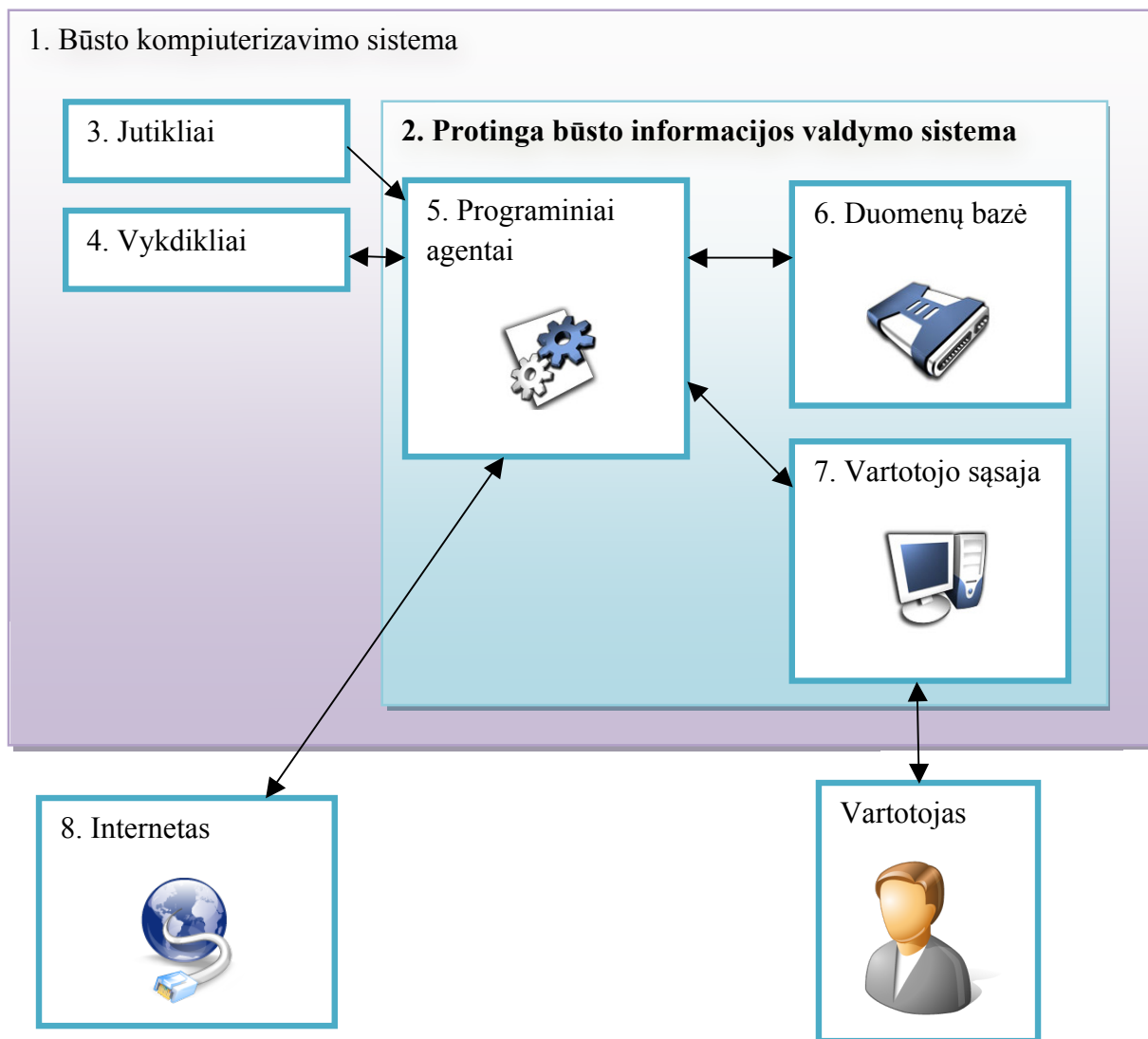
Eksperimentinėje dalyje aprašyta sistemos realizacija, programinių agentų realizavimas protingoje būsto informacijos valdymo sistemoje. Pateikta sistemos prototipo realizacija ir jos tyrimas. Taip pat pateikiamos tolimesnės sistemos tobulinimo ir plėtros rekomendacijos.

2 PROBLEMINĖS SRITIES ANALIZĖ

Darbo probleminė sritis – būsto kompiuterizavimo sistemų informacijos surinkimo ir valdymo būdai, jų tobulinimas ir taikymas. Pagrindinis probleminės srities analizės skyriaus tikslas yra išanalizuoti mokslinę literatūrą bei technines ataskaitas, kurios leidžia nustatyti probleminės srities sprendimo būdus. Probleminės srities analizės skyrius yra suskirstytas į poskyrius, kuriuose analizuojami atskiri probleminės srities aspektai: protingų būsto informacijos valdymo sistemų analizėje tiriami kitų mokslininkų atlikti darbai, esami produktai, jų privalumai ir trūkumai, galimi taikymo būdai darbe; programinių agentų analizėje tiriama agentų tipologija, savybės, jų galimi panaudojimo atvejai, bei teikiami privalumai; agentų kūrimo programinės įrangos analizėje analizuojami agentų kūrimo įrankiai ir šių technologijų panaudojimas kuriant agentines sistemas.

2.1 Darbo objekto aprašymas

Darbo objektas – būsto kompiuterizavimo sistemos dalis (posistemė), skirta informacijai rinkti, apdoroti, valdyti ir pateikti vartotojui (1 pav.). Kompiuterizuotos namų sistemos, pasinaudamos įvairiais jutikliais, sugeba rinkti informaciją iš aplinkos, taip pat su įvairiais vykdykliais (angl. actuator) jos gali valdyti kompiuterizuotus būsto prietaisus. Tačiau šios sistemos retai kada turi sprendimų priėmimo metodus ar algoritmus, dėl to atsiranda poreikis papildomų sistemų, kurios sugebėtų analizuoti esamą informaciją. Būsto informacijos valdymo sistemos veikia kartu su būsto kompiuterizavimo sistemomis, rinkdamos informaciją iš jutiklių ir vykdyklių ir išanalizavusios gaunamus duomenis pateikia rezultatus vartotojui. Protingų namų informacijos valdymo sistemų „intelektas“ gali būti įvairaus lygio – nuo paprastų namų išteklių sąnaudų grafikų sudarymo iki sudėtingų prognozių, nuspėjančių įvairius išteklių suvartojimo kaštus. Informacijos valdymo sistemos „išmanumas“ gali būti įgyvendinamas įvairiomis technologijomis – šiame darbe nagrinėjamos programinių agentų technologijos ir jų taikymo būdai tokiose sistemose. Programinių agentų technologija pasirinkta todėl, kad kuriama sistema bus taikoma būsto kompiuterizavimo sistemoje, sukurtoje programinių agentų pagrindu. Protinga namų informacijos valdymo sistema duomenis gali rinkti įvairiais būdais: gauti informaciją iš kompiuterizuoto būsto sistemos jutiklių ar vykdyklių, gauti duomenis iš vartotojo, pasinaudoti savo vidine duomenų baze ar rinkti informaciją internete.



1 pav. Protingos būsto informacijos valdymo sistemos kontekstinė diagrama

1. Būsto kompiuterizavimo sistema susideda iš įvairių jutiklių (temperatūros, drėgmės ir kt.), kurie renka informaciją apie būsto parametrus ir saugo ją duomenų bazėje. Taip pat pagal vartotojo reikalavimus sistema sugeba valdyti įvairius būsto parametrus (patalpų temperatūrą, apšvietimą ir kt.). Papildomai ši sistema gali turėti vartotojo sąsają. Būsto informacijos valdymo sistemos programiniai agentai gali panaudoti šios sistemos duomenis įvairiems namų ištekliams analizuoti ir atvaizduoti, tokiems kaip šilumos sąnaudų grafikai, elektros prietaisų veikimo laikas ir kt.

2. Protinga būsto informacijos valdymo sistema – šios sistemos paskirtis yra pagal vartotojo pageidavimus rinkti ir analizuoti informaciją iš kitų būsto sistemų ir interneto. Vienas iš šios sistemos atrinktos informacijos panaudojimo būdų – rekomendacijų apie būsto resursų naudojimą šildymo sezono metu pateikimas vartotojui.

3. Jutikliai grąžina informaciją apie įvairius būsto išteklius – kambario temperatūrą, drėgmę ir kt.

4. Vykdikliai valdo įvairius kompiuterizuotus prietaisus - šildytuvus, namų apšvietimą ir kt. Vykdikliai gali grąžinti informaciją apie savo būseną (įjungtas ar išjungtas).

5. Programiniai agentai atlieka pagrindines būsto informacijos valdymo sistemos funkcijas, tokias kaip informacijos rinkimas ir analizavimas, komunikavimas su vartotoju per vartotojo sąsają. Agentai informaciją gali gauti iš įvairių šaltinių: komunikuodami su kitomis sistemomis (būsto valdymo, virtuvės informacine sistema ir pan.), ieškodami informacijos internete, iš savo sistemos duomenų bazės, gaudami vartotojo teikiamą informaciją. Patys agentai sugeba tarpusavyje keistis duomenimis, reikalingais atlikti individualių agentų funkcijoms. Kiekvienas agentas turi savo specifinę paskirtį: komunikuoti su duomenų bazėmis, stebėti vartotojo sąsają, atlikti skaičiavimus, kurti ir prižiūrėti naujus agentus ir pan.

6. Duomenų bazė. Būsto informacijos valdymo sistemos duomenų bazės paskirtis yra saugoti sistemos darbui reikalingus duomenis: pasirinktus vartotojo nustatymus, skaičiavimų duomenis ir rezultatus ir kt.

7. Vartotojo sąsaja. Namų informacijos valdymo sistemos vartotojo sąsaja gauna ir pateikia informaciją vartotojui. Sąsaja su sistema bendrauja programinių agentų dėka. Vartotojui skirta informacija yra vizualiai pateikiama ekrane. Pati vartotojo sąsaja neturi galimybės prisijungti prie duomenų bazės, tačiau informacijos vizualizacijai reikalingus duomenis (paveikslėlius, spalvų paletes, šriftus, matmenis ir pan.) ji saugo savo bibliotekose.

8. Internetas. Būsto informacijos valdymo sistemos programiniai agentai internetu gali pasiekti skaičiavimams ar vartotojui reikalingą informaciją tokią kaip šildymo kaštai, lauko temperatūra ir pan.

2.2 Protingų būsto informacijos valdymo sistemų analizė

Protingos būsto informacijos valdymo sistemos neatsiejamai veikia kartu su būsto informacinėmis sistemomis. Šios sistemos sujungia naudojamą technologijas ir paslaugas namuose, biuruose ir mažuose pastatuose, kurių tikslas yra teikti geresnius apsaugos, komforto, komunikavimo ir energijos taupymo sprendimus išieškojant kuo mažiau vartotojo laiko ir pastangų [1, 2]. Šios sistemos su aplinka sąveikauja per mazgų tinklą, kiekvienas mazgas gali atlikti, priklausomai nuo funkcionalumo, kelis vaidmenis: jutikliai (angl. sensor) - mazgai, kurie renka informaciją iš aplinkos ir siunčia ją kitiems sistemos komponentams, vykdikliai (angl. actuator) - mazgai, kurie geba atlikti tam tikras operacijas (įjungti ar išjungti

šviesą), ir jutikliai-vykdikliai – mazgai, kurie gali atlikti tiek informacijos rinkimo, tiek valdymo operacijas [3, 4, 5]. Protingos būsto informacijos valdymo sistemos ne tik įgalina vartotoją stebėti savo būsto parametrus (temperatūrą, drėgmę ir kt.), bet ir užprogramuoti atitinkamus tvarkaraščius šviesios, šilumos valdymo sistemom ir pan. [1, 6] Informacijos valdymo sistemos taip pat kaupia duomenis apie būsto naudojamus resursus vėlesnėm reikmėm – resursų panaudos grafikų sudarymui, šildymo kaštų prognozėm ir pan. Būsto informacinės sistemos paskirtis – kontroliuoti įrenginius, valdančius įvairius būsto resursus (apšvietimo jungiklius, garažo vartus ir t.t.), bei teikti informaciją protingom būsto informacijos valdymo sistemom [1, 4, 6].

Būsto informacinės sistemos tarpusavyje apjungia namų ūkyje naudojamus elektrinius prietaisus [4, 6]. Šios sistemos gali atlikti daug įvairių funkcijų: namų kompiuterizavimas, namuose vykstančių įvykių valdymas - garso ir vaizdo sistemų valdymas, gėlių bei kiemo laistymo kontrolė, gyvūnų maitinimas, apšvietimo parinkimas sikingiems atvejams (žiūrint filmus, skaitant knygas, valgant ir kt.), bei namų robotų naudojimas [4]. Prietaisai gali būti apjungti kompiuterių tinklu ir būti valdomi asmeniniu kompiuteriu ar per atstumą - naudojant interneto ryšį [1].

Dažniausiai būsto informacinės sistemos yra diegiamos namo statymo metu [6], dėl to kad yra paprastesnis priėjimas prie sienų, elektros lizdų, tai taip pat įgalina kuo patogiau priderinti technologijas prie esamo namo išplanavimo. Sistemos dalių sujungimas laidas yra pranašensis dėl patikimumo, pasiekiamumo ir lankstumo [1]. Priderinant būsto informacinės sistemas jau pastatytiems namams dažniausiai naudojamos belaidės technologijos, taip sumažinant laidų tiesimo kaštus. Ryšys tarp prietaisų gali būti palaikomas tiek laidais, tiek radio ar infraraudonaisiais signalais[1, 3, 4, 5].

Dauguma būsto informacinių sistemų valdymas sudaromas iš dviejų pagrindinių komponentų: centrinio valdymo kompiuterio ir vartotojo sąsajų [1, 6]. Visos posistemės (pvz. apšvietimo valdymo posistemė) komunikuoja per centrinį valdymo kompiuterį [1, 4]. Tuo tarpu vartotojas, norėdamas kontroliuoti posistemas, prijungtas prie centrinio valdymo kompiuterio, naudoja vartotojo sąsajas, tokias kaip klaviatūros, lietimui jautrūs ekranai, delninukai ar asmeniniai kompiuteriai [1, 4]. Kiekviena būsto informacinė sistema tarpusavyje skiriasi intelekto lygmeniu, funkcijų užprogramavimo paprastumu, pritaikomumo lygmeniu, operacijų paprastumu, vartotojų sąsajų paprastumu, kaina ir kitomis ypatybėmis.

Neretai sistemos atnaujinimas kelia daug bėdų, nes vartotojam nėra patogiu ar visai neįmanoma atnaujinti savo būsto informacijos valdymo sistemos tik pasirodžius naujoms ar

atnaujintoms technologijoms, nes turima techninė ir programinė įranga dažniausiai būna glaudžiai susijusi [4, 7].

Rinkoje yra keletas labiausiai žinomų sistemų, viena iš jų „Home Control Assistant“ (HCA). HCA yra namų automatizavimo programa, paremta Windows platforma, kuri veikia su plačiu namų automatizavimo sąsajų spektru: apšvietimo, įrenginių, drėkinimo, šildymo ir kt. sistemų valdymu. Su HCA tvarkaraščiais ir „Green Mode“ (liet. „Žalioji režimas“) galima paprastai sutaupyti lėšas: sumažinant elektros sąnaudas, taupant vandenį, ir pagerinti saugumą automatizuojant namus [8]. „Home Control Assistant“ veikia su bet kokiais kompiuteriais, turinčiais Windows XP, Vista ar Windows 7 operacines sistemas. Sistema veikia su apšvietimo valdymo bei įrenginių posistemėm, kurios naudoja X10, INSTEON, UPB, belaidėmis ir Global Cache IR technologijomis [8]. HCA gali būti įdiegtas kartu su jau namuose egzistuojančia automatizavimo įranga. HCA sugeba nuskanuoti idiegtą INSTEON tinklą ir pridėti visus tinklui priklausančius prietaisus prie namo išplanavimo bylos [8]. UPB tinklai gali būti importuojami pasinaudojant UPB diegimo programa. Nauji automatizavimo sprendimai yra sukuriami veldių pagalba kuriant naujas bylas, pridedant prietaisus, sudarant tvarkaraščius.

Kita sistema „HouseLinc“. Ši sistema leidžia vartotojui valdyti ir automatizuoti apšvietimą bei kitas posistemas, naudojant asmeninį kompiuterį. „HouseLinc“ veikia su INSTEON ir X10 technologijomis [9]. Vartotojas gali sudarinėti tvarkaraščius ir scenarijus (pvz. apšvietimo stiprumas žiūrint filmą ar skaitant ir pan.). Komplekte yra „HouseLinc“ programinė įranga, „PoweLinc“ modemas su INSTEON USB sąsaja ir USB laidas [9]. Svarbu paminėti, kad tvarkaraščiai veikia tik įjungus kompiuterį – tai nėra paranku vartotojui.

Trečia sistema – „CasaTunes XLi“, kurios komplekte yra programinė ir techninė įranga, kuri sukuria namų garso sistemą namuose su audio prietaisų sąsajom ir interneto ryšiu. Techninę įrangą sudaro PCI-e garso plokštė [10], skirta asmeniniam kompiuteriui, kuris yra naudojamas kaip serveris garso skleidimui namų zonose. PCI-e garso plokštė gali groti išsaugotus muzikinius failus (mp3, m4p, midi ir wav formatais) 6 skirtingose namų zonose [10]. Programinė įranga leidžia vartotojui pasirinkti grotinčią muziką iš iki 5 skirtingų šaltinių vienu metu ar groti tą pačią muziką visose zonose [10]. Naudojant asmeninį kompiuterį ar bet kokį kitą prietaisą su prieiga prie interneto galima valdyti visas zonas ir kiekvieną zoną atskirai. PCI-e garso plokštė taip pat turi infraraudonųjų spidulių imtuvą, kurį galima suderinti su nuotolinio valdymo pultu [10].

Dar viena iš labiausiai žinomų rinkos sistemų yra „Web-Link II“. Ši sistema veikia kartu su „OmniLT“, „Omni Iie“ ar „OmniPro II“ [11] automatizavimo valdikliais. „Web-Link II“ leidžia vartotojui stebėti ir valdyti savo namus iš bet kurio prietaiso, turinčio prieigą prie interneto. Su „Web-Link II“ galima sukurti įvairius įvykius: pvz. įsijungus signalizacijai – vartotojui išsiųsti e-laišką su pranešimu ir pan. [11] „Web-Link II“ veikia su asmeniniais kompiuteriais su „Windows“ operacinėmis sistemomis [11] ir prie kompiuterių prijungtais bet kuriais „Omni“ klasės valdikliais. Protinga būsto informacijos valdymo sistema generuoja internetinius puslapius iš valdiklių informacijos, kuriuos galima peržiūrėti naudojantis vietiniu kompiuterių tinklu ar interneto ryšiu. Įdiegus „Web-Link II“ nereikia papildomų diegimų: „Web-Link II“ programinė įranga automatiškai suranda visus „Omni“ klasės įrenginius [11], prijungtus prie sistemos. „Web-Link II“ leidžia vartotojui stebėti savo nuosavybę su video kamerų pagalba, prireikus bet kada įrašinėti vaizdą – savarankiškai pasirenkant ar įvykus užprogramuotam įvykiui. Taip pat vartotojas gali gauti e-laiškus, kurie išsiunčiami iš suprogramuotų įvykių, ar naudotis susirašinėjimo žinučių sistema.

Kita populiari rinkos sistema – „HomeRunner RBI“ veikia su INSTEON ir X10 [12] technologijomis. Sistema veikia su savo atskiru prietaisu [12] ir įgalina vartotoją bet kuriuo metu priėti prie savo protingos namų informacijos valdymo sistemos naudojantis internetu. „HomeRunner RBI“ veikia „Windows“ aplinkoje ir leidžia vartotojui sudarinėti tvarkaraščius, įvykius ar scenarijus [12] (apšvietimo valdymo, temperatūros kontrolės ir pan.). Programinė įranga taip pat leidžia vartotojui keisti ar sukurti naują grafinę vartotojo sąsają.

Atlikus protingų būsto informacijos valdymo sistemų analizę buvo sudaryta produktų lentelė (1 lentelė), su esminiais programinės įrangos aspektais. Kiekvienas pasirinktas aspektas nusako mūsų kuriamai sistemai reikšmingas savybes: suderinamumas su operacinėmis sistemomis – galima pastebėti dažniausiai produktų (ir vartotojų) naudojamas operacines sistemas, prisijungimas prie sistemos – išvelgiamas pačios sistemos naudojimo būdas, esminės atliekamos funkcijos – parodo metodų ir algoritmų panaudojimą.

1 lentelė. Protingų būsto informacijos sistemų aspektų palyginimas.

Pavadinimas	Suderinamumas su operacinėmis sistemomis	Prisijungimas prie sistemos	Esminės atliekamos funkcijos	Papildomos pastabos
Home Control Assistant	Microsoft Windows Vista/7	Iš pagrindinio kompiuterio	Tvarkaraščiai	Informavimas SMS žinute ar e-laišku
HouseLine	Microsoft Windows 2000/XP/Vista/7	Iš pagrindinio kompiuterio	Tvarkaraščiai, resursų sąnaudų atvaizdavimas	
Casa Tools XLi	Microsoft Windows XP/Vista/Server 2003/Server 2008	Bet kurio prietaiso su prieiga prie interneto	Tvarkaraščiai (muzikos grojaraščių sudarymas)	
Web-Link II	Microsoft Windows XP/Server 2003	Bet kurio prietaiso su prieiga prie interneto	Tvarkaraščiai, resursų sąnaudų atvaizdavimas	Informavimas e-laišku
HomeRunner RBI	Microsoft Windows XP/Vista/7	Bet kurio prietaiso su prieiga prie interneto	Tvarkaraščiai	Galimybė susikurti savo grafinę vartotojo sąsają

Išvados. Išanalizavus protingas namų informacijos valdymo sistemas buvo nustatyta, kad dauguma jų veikia tvarkaraščio principu (reguliuojamas šildymas, apšvietimas tam tikru laiko momentu ir pan.) ar užprogramuoto įvykio principu (išsiunčiamas pranešimas vartotojui apie suveikusių signalizaciją ir kt.). Nebuvo nustatyta programinių agentų naudojimo esamos informacijos analizavimui ar rekomendacijų pateikimui. Taip pat nenustatyta daugiaagenčių sistemų, kurios galėtų išnaudoti visas programinių agentų teikiamas galimybes, tokias kaip – agentų bendravimas tarpusavyje siekiant tikslo, užduočių paskirstymas atitinkamiems agentams, lengvesnis sistemos taisymas ir atnaujinimas (tereikia atnaujinti ar pakeisti vieną sistemos agentą vietoje visos sistemos) ir kt. Visos analizuotos programos veikia tik „Microsoft Windows“ operacinių sistemų aplinkose, vartotojų patogumui gerinti ir pačios protingos būsto informacijos valdymo sistemos pritaikomumo mastui didinti, reikia atsižvelgti į galimybę sistemą naudoti su skirtingomis operacinėmis sistemomis. Prisijungimas prie šių sistemų varijuoja tarp prisijungimo iš pagrindinio kompiuterio ir priegos per internetą, šis aspektas labiau priklauso nuo kiekvienos individualios sistemos paskirties – pvz. sudarant apšvietimo tvarkaraščius nėra būtina prieiga per internetą, pakanka sudaryti tvarkaraštį vieną kartą iš pradinio kompiuterio ir vėliau, pagal reikalavimus, galima jį koreguoti, tačiau norint įjungti ar išjungti šviesas bet kuriuo metu – patogiausiai tai daryti per interneto priegą, taip galima valdyti prietaisus nebūnant šalia pagrindinio kompiuterio. Vis dėlto prieiga per internetą turi pranašumą, nes sistema yra pasiekiamą praktiškai iš bet kokios vietos – tiek iš namų, tiek už

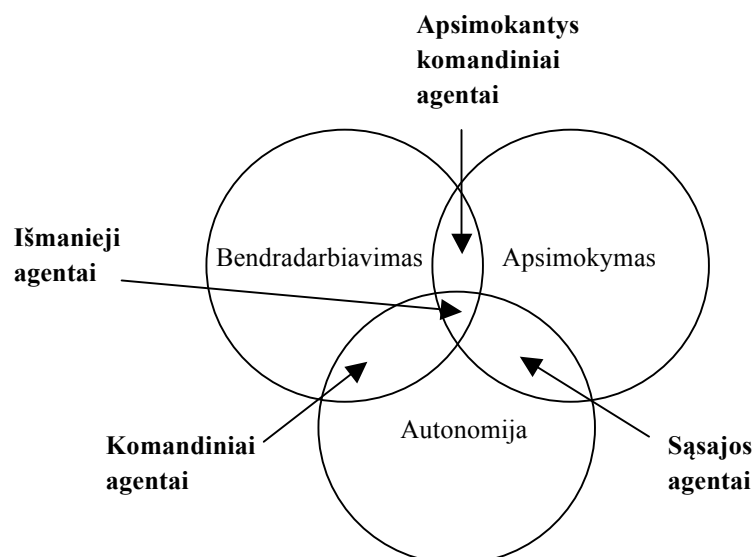
jų ribų. Verta paminėti, kad kai kurios sistemos geba informuoti vartotojus apie įvairius įvykius SMS žinute ar elektroniniu laišku.

2.3 Programinių agentų analizė

2.3.1 Programinio agento aprašymas

Terminas „agentas“ (arba „programinis agentas“) yra naudojamas įvairiose kompiuterinių technologijų srityse – duomenų bazėse, dirbtiniam intelektui, operacinėse sistemose, kompiuterių tinkluose. Nors literatūroje nėra vieningo termino „programinis agentas“ apibrėžimo, tačiau sprendžiant iš skirtingų aprašymų galima teigti, kad agentas yra programinės įrangos komponentas, kuris sugeba veikti autonomiškai, komunikuoti su jį valdančia sistema arba kitais agentais, siekti jam apibrėžtų tikslų ir gali nepertraukiamai veikti [13, 14, 15, 16, 17, 18, 19, 20]. Sistemoje gali veikti ir vienas agentas, kuris gali rinkti informaciją iš aplinkos ir, esant pageidavimui, komunikuoti su vartotoju, tačiau dažniausiai sistemas sudaro daugiau nei vienas agentas. Tokios sistemos yra vadinamos daugiaagentinėmis sistemomis (angl. multi-agent systems) ir susideda iš daugelio programinių agentų, kurie gali turėti skirtingus tikslus ir užduotis (pvz. – sistema turi bendrą tikslą, arba skirtingi agentai turi skirtingus tikslus)[13, 14, 15, 16, 21]. Šių sistemų agentai tarpusavyje sąveikauti gali tiesiogiai – komunikuodami vienas su kitu, arba netiesiogiai – reguodami į aplinką, kurioje jie veikia[14, 16]. Tarpusavyje agentai gali bendrauti pasinaudojant įvairias žinučių sistemas, kiekviena daugiaagentinė sistema gali turėti specifinę programinių agentų bendravimo kalbą (angl. ACL – agent communication language) [16]. Visos daugiaagentinės sistemos gali būti laikomos kaip turinčios dinamines aplinkas, nes sistemos agentai apgalvotai veikia aplinką nenuspėjamais būdais [15]. Dažniausiai praktikoje pasitaikanti daugiaagentinių sistemų panaudojimo sritis – informacijos valdymas [14, 19], tai galima paaiškinti su vis platesniu interneto panaudojimu įvairiose srityse, ko pasekoje atsiranda didžiuliai duomenų kiekiai, kuriuose nuolat reikia atlikti informacijos paieškas ar duomenų filtravimą.

Programiniai agentai yra klasifikuojami pagal pradines savybes (autonomija, apsimokymas, bendradarbiavimas) (2 pav.). Autonomija (angl. autonomy) – tai programinio agento gebėjimas atlikti savo užduotis be žmogaus įsikišimo[13]. Todėl agentai turi vidines būsenas ir tikslus, kurie užtinkrina uždavinio, kurį uždavė vartotojas, sprendimą. Apsimokymas (angl. learn) – ši savybė įgalina agentą apsimokyti reguojant ir/ar sąveikaujant su išorine aplinka[13]. Bendradarbiavimas (angl. cooperation) – tai programinių agentų gebėjimas komunikuoti su kitais agentais, be šios savybės daugiaagentės sistemos negalėtų egzistuoti.



2 pav. Agentų klasifikacijos pradinės savybės [13, 15]

Programiniam agentui turėtų būti įmanoma priskirti bent dvi pradinės savybės (2 pav.), priešingu atveju toks programinis komponentas nesiskirtų nuo paprastos programinės įrangos. Taigi, atsižvelgiant į pradinių savybių derinius, programinius agentus galima būtų suskirti į keletą klasių [13, 14]: komandiniai agentai, sąsajos agentai, mobilieji agentai, informacijos agentai, reaktyvūs agentai, hibridiniai agentai, sumanieji agentai.

Kuriamai sistemai labiausiai tinka komandinių, sąsajų ir reaktyvių agentų klasės:

Komandiniai agentai – pasižymi autonomija (savarankiškumu) ir bendradarbiavimu su kitais agentais, siekiant vartotojo užduotų tikslų. Šie agentai taip pat gali turėti ir apsimokymo savybę, tačiau ji nėra reikšminga komandinių agentų veikimui [13].

Sąsajos agentai – pasižymi autonomijos ir apsimokymo savybėmis, siekiant atlikti savo užduotis. Sąsajos agentai yra tarsi asmeninis asistentas, kuris bendrauja su vartotoju toje pačioje sistemos aplinkoje. Yra vienas esminis skirtumas tarp sąsajos ir komandinių agentų – sąsajos agentai bendrauja su vartotoju, komandiniai agentai – su kitais sistemos agentais, todėl kad bendrauti su vartotoju dažniausiai nereikia naudoti specifinės kalbos, kuri yra būtina tarpusavyje bendraujantiems agentams [13, 14].

Dažniausiai sąsajos agentai teikia palaikymo ir pagalbos paslaugas vartotojui, kuris mokosi naudotis specifine programine įranga (pvz. operacine sistema). Agentas stebi vartotojo atliekamus veiksmus su vartotojo sąsaja ir atitinkamai siūlo geresnius būdus atlikti tam tikrom užduotim [13], tokiom kaip greitesnis duomenų saugojimo būdas, naujų įrankių pristatymas, nenaudojamų įrankių šalinimas. Todėl galima būtų teigti, kad sąsajos agentas yra savarankiškas asmeninis asistentas, kuris, bendradarbiaujant su vartotoju, padeda atlikti

įvairias užduotis dirbant su programine įranga [13]. Sąsajos agentai dažniausiai apsimoko tokiais būdais, kurie pagerina agento pagalbą varotojui – stebint ir atkartojant vartotojo veiksmus (mokymąsis iš vartotojo), gaunant grįžtamąjį ryšį iš vartotojo (mokymąsis iš vartotojo), gaunant aiškius nurodymus iš vartotojo (mokymąsis iš vartotojo), ieškant patartimo iš kitų agentų (mokymąsis iš kitų agentų).

Sąsajos agentų bendravimas su kitais agentais (jei išvis toks bendravimas vyksta tarp sąsajos agentų ir kitų agentų) dažniausiai yra labai ribotas, toks kaip patarimo prašymas iš kitų agentų, neinančias detalesnį komunikavimą, kuris galėtų vykti tarp komandinių agentų [13].

Reaktyvūs agentai – yra išskirtinė agentų klasė, kuri veikia pagal veiksmo-atoveiksmio principą, priklausanti nuo sistemos, kurioje yra agentai, būsenos [13, 20]. Dažniausiai agento veiksmus sužadina įvairių jutiklių (šviesos, šilumos, drėgmės ir kt.) siunčiama informacija.

2.3.2 Programinių agentų, naudojamų protingose būsto informacijos valdymo sistemose, analizė

Kompiuterizuotų būstų sistemos susideda iš buitinių elektroninių prietaisų tinklų, kurie yra valdomi įvairių tipų informacinių sistemų [22, 23]. Protingos būsto informacijos valdymo sistemos yra atsakingos už kompiuterizuoto būsto informacijos rinkimą, apdorojimą ir valdymą. Kiekvienas prietaisas teikia paslaugas, kurios yra aprašomos operacijomis ir tiekia informaciją apie įvykius, kurie atspindi prietaiso būseną [22]. Paslaugos gali būti tokios, kaip kambario šildymas yra įjungiamas (suteikiama paslauga), kai vidaus temperatūra pasiekia 15 laipsnių (specifinės operacijos), išjungus šildymui yra siunčiamas pranešimas centriniam kompiuteriui apie įjungtą šildymą patalpoje (teikiama įvykio informacija apie pasikeitusią būseną). Visi prietaisai informaciją renka per jutiklius ir ją išsiunčia centriniam kompiuteriui, kuris yra atsakingas už duomenų rinkimą, apdorojimą ir sprendimų priėmimą [23].

Programinių agentų panaudojimas yra efektyvus sprendimo būdas protingose būsto informacijos valdymo sistemose, dėl kelių priežasčių [22, 23]. Pirma, „protinga“ sistema su centrinio valdymo architektūra nėra tinkama paskirstyto valdymo aplinkai [23], šioms aplinkoms tinka daugiaagentinės sistemos, kurios yra skirtos spręsti sudėtingiems, paskirstytų skaičiavimų uždaviniams [23]. Antra, kiekvienas „protingas“ namų apyvokos daiktas turi ribotą atmintį ir skaičiavimo galimybes, todėl yra paranku prietaisus valdyti su autonominiiais agentais, kurie neprivalo būti įrašyti tiesiai į prietaisą [23]. Taip pat augant kompiuterizuotų būstų gebėjimo prisitaikyti aplinkai poreikiui, atsirado šių būstų sistemų ypatybių reikalavimai, kurie yra lengvai įgyvendinami pasitelkiant agentų technologijas [22]: konfigūracija (angl. configurability) – sistema turėtų būti lengvai konfigūruojama tiek

administratorių, tiek vartotojų [22]; situacijos suvokimas (angl. context-awareness) – sistema turėtų sugebėti stebėti savo vidinę būseną (sudedamųjų dalių gedimus) [22]; autonominė rekonfiguracija (angl. autonomic reconfiguration) – sistema turėtų save automatiškai perkonfigūruoti, atsižvelgiant į sistemos pakitimus [22]; dinaminis prisitaikymas (angl. dynamic adaptation) – sistema turėtų sugebėti dinamiškai atlikti konfigūracijos ar rekonfigūracijos operacijas nenutraukiant savo darbo [22].

Protingų būsto informacijos valdymo sistemų naudojami agentai gali atlikti įvairias užduotis. Dauguma programinių agentų šiose sistemose atlieka autonominio prietaisų valdymo užduotis, nes tokiu būdu galima efektyviai autonomiškai valdyti skirtingus prietaisus, dažniausiai prietaisai yra valdomi tvarkaračio (pvz. apšvietimas tam tikrom valadom)ar regavimo į aplinką (šildymo įjungimas ar išjungimas, pasiekus tam tikrą kambario temperatūrą) principu[22, 23, 24, 25]. Taip pat naudojami sąsajos agentai, kurie užtikrina sistemos komunikavimą tarp vartotojų ar administratorių [22, 24]. Taip pat yra sistemų, kurios naudoja daugiaagentes sistemas atlikti savo funkcijom, tokiu būdu išnaudojant paskirstytų skaičiavimų taikymo privalumus (kiekvienas agentas yra autoniminis, keletas agentų gali bendrai siekti užsibrėžtų tikslų ir pan.) [24]. Dar viena priežastis, dėl kurios yra paplitęs daugiaagentinių sistemų naudojimas, - tai labai paprastas sistemos atnaujinimas ar naujų komponentų diegimas, kadangi tokiose sistemose agentai tarpusavyje bendrauja žinutėmis ir dažniausiai tiesiogiai nepaveikia vienas kito elgsenos, todėl keičiant vieną ar kitą agentą nenukenčia visa sistema[24].

Išvados. Išanalizavus programinius agentus, kurie yra naudojami protingose būsto informacijos valdymo sistemose buvo nustatyta keletas svarbių jų (agentų) panaudojimo atvejų. Dažniausiai priežastis, kodėl buvo nuspręsta sistemą kurti panaudojant programinius agentus – patogumas dirbant su atskirais prietaisais, galimybė kiekvieną sistemos sudedamosios dalies veikimo procesą padaryti autonominį, taip paspartinant bendrą sistemos darbą. Taip pat verta paminėti, kad norint efektyviai kontroliuoti sistemos komponentus, reikėtų naudoti daugiaagentines sistemas, kurios yra skirtos sudėtingų paskirstytų skaičiavimų užduočių sprendimui. Naudojant daugiaagentines sistemas užduotis galima paskirstytui keletui agentų, kurie tarpusavyje komunikuodami sieks atlikti užsibrėžtus tikslus. Kitas daugiaagentinių sistemų privalumas yra - nesudėtingas sistemos atnaujinimas ir išplėtimas. Dažniausios programinių agentų atliekamos funkcijos yra: prietaisų valdymas ir komunikavimo tarp sistemos ir vartotojų užtikrinimas.

2.3.3 Programinių agentų palyginimas ir pasirinkimas

Agentų tipų gali egzistuoti be galo daug, nes skirtingi programiai agentai gali pasižymėti skirtingomis savybėmis (vartotojo sąsajos agentui apsimokymo savybė yra svarbesnė už bendradarbiavimo savybę ir pan.). Todėl buvo sudarytas programinių agentų sąrašas, kurie yra naudojami protingoje būsto informacijos valdymo sistemoje:

Vartotojo sąsajos agentas – atsakingas už vartotojo įvedamos informacijos perdavimą kitiems sistemos agentams, bei rezultatų pateikimą vartotojo sąsajai. Vartotojo įvedamą informaciją agentas gauna stebėdamas vartotojo veiksmus grafinėje sąsajoje ir įvykus įvykiui (mygtuko paspaudimas) atlieka specifines užduotis (siunčia duomenis ir gauna rezultatus iš rekomendacijų agento ir juos pateikia vartotojui). Vartotojo sąsajos agentas gali bendrauti su šiais agentais: rekomendacijų agentas, duomenų bazės agentas, orų prognozių agentas, klaidų aptikimo agentas.

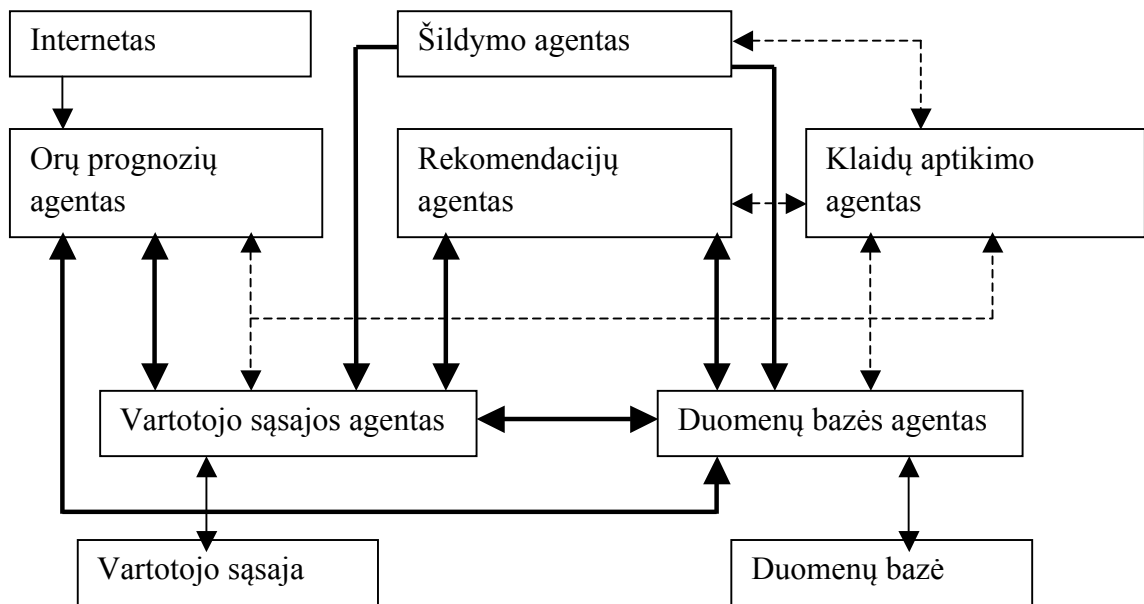
Rekomendacijų agentas – atlieka skaičiavimus, kurie padeda nuspėti būsto šildymo kaštus. Skaičiavimams reikalingus duomenis gauna iš vartotojo – per vartotojo sąsajos agentą, ir iš duomenų bazės – su duomenų bazės agento pagalba. Gautus rezultatus rekomendacijų agentas atiduoda vartotojo sąsajos agentui, kad šis atvaizduotų informaciją vartotojui. Rekomendacijų agentas gali bendrauti su šiais agentais: vartotojo sąsajos agentas, duomenų bazės agentas.

Duomenų bazės agentas – atsakingas už duomenų saugojimą ir jų pateikimą kitiems sistemos agentams. Duomenų bazės agentas dirba su sistemos duomenų bazėmis, kiti agentai neturi prieigos prie duomenų bazių. Duomenų bazės agentas gali bendrauti su šiais agentais: vartotojo sąsajos agentas, rekomendacijų agentas, orų prognozių agentas, klaidų aptikimo agentas.

Orų prognozių agentas – internete renka informaciją apie orų prognozes ir ją teikia vartotojui per vartotojo sąsajos agentą. Orų informacija taip pat saugoma duomenų bazėje rekomendacijų agento skaičiavimams. Orų prognozės vieta yra pasirenkama vartotojo ir saugoma sistemos duomenų bazėje. Orų prognozių agentas gali bendrauti su šiais agentais: vartotojo sąsajos agentas, duomenų bazės agentas.

Klaidų aptikimo agentas – stebi visus sistemos agentus ir atsiradus programiniams gedimams – juos pašalina. Siekiant aptikti gedimus, klaidų aptikimo agentas gali bendrauti su visais sistemos agentais.

Šildymo agentas – atlieka šilumos stebėjimo ir valdymo funkcijas.



3 pav. Protingos būsto informacijos valdymo sistemos agentų bendravimo schema

Tarpusavyje agentai gali bendrauti žinučių pagalba. Remiantis protingos būsto informacijos valdymo sistemos agentų aprašymu buvo sudaryta jų (agentų) bendravimo schema (3 pav.), kuri atvaizduoja komunikavimo ryšius tarp sistemos agentų. Iš šios schemos galima nustatyti, kad svarbiausi sistemos darbui agentai yra vartotojo sąsajos ir duomenų bazės agentai, nes su šiais agentais bendrauja visi sistemos programiniai agentai. Taip pat didelę svarbą rodo vartotojo sąsajos ir duomenų bazės agentų bendravimas, atitinkamai, su vartotojo sąsaja ir duomenų baze, tik per šiuos agentus sistema gauna informaciją iš vartotojo ir duomenų bazės. Likę agentai tarpusavyje nebendrauja, susiradę reikiamus uždavinio duomenis su duomenų bazės agento pagalba, panaudoja juos uždavinio atlikimui (šildymo kaštų skaičiavimui ir kt.) ir rezultatus išsiunčia vartotojui per vartotojo sąsajos agentą. Klaidų aptikimo agento bendravimas su kitais sistemos agentais šiek tiek skiriasi – pats bendravimas nepadedą atlikti jokių skaičiavimų, tačiau klaidų aptikimo agentui pastebėjus, kad kuris nors agentas nebeatsiliepia į jo (klaidų aptikimo agento) siunčiamas žinutes – nebeatsiliepiantis agentas yra pakeičiamas nauju ir taip ištaisomi galimi sistemos sutrikimai.

Siekiant palyginti protingos būsto informacijos valdymo sistemos agentus ir jų savybes buvo sudaryta agentų palyginimo lentelė (2 lentelė). Parinkti palyginimo lentelės aspektai apibūdina nagrinėjamą agentą: savybės – nusako agentui būdingas savybes (tiek esmines, tiek mažiau reikšmingas), agento klasė – kokiai klasei labiausiai tiktų priskirti pasirinktą agentą, paskirtis – agento atliekamos funkcijos.

2 lentelė. Protingų būsto informacijos sistemos agentų palyginimas

Agento pavadinimas	Savybės	Agento klasė	Paskirtis
Vartotojo sąsajos agentas	Bendradarbiavimas, apsimokymas, autonomija	Sąsajos agentas	Vartotojo informaciją teikti kitiems agentams, stebėti vartotojo veiksmus ir, atitinkamai, valdyti vartotojo sąsają, kitų agentų gautus rezultatus perduoti vartotojui
Rekomendacijų agentas	Bendradarbiavimas, autonomija, apsimokymas	Komandinis agentas	Nuspėti šilumos kaštus ir rezultatus pateikti vartotojui
Duomenų bazės agentas	Bendradarbiavimas, autonomija	Komandinis agentas	Tiekti agentų darbui reikiamus duomenis Saugoti agentų atsiunčiamus duomenis
Klaidų aptikimo agentas	Bendradarbiavimas, autonomija, apsimokymas	Reaktyvus agentas	Aptikti sistemos agentų klaidas ir jas ištaisyti
Orų prognozių agentas	Bendradarbiavimas, apsimokymas, autonomija	Komandinis agentas	Pasinaudojant internetu gauti orų prognozių duomenis
Šildymo agentas	Bendradarbiavimas, autonomija, apsimokymas	Komandinis agentas	Stebint patalpos temperatūrą reguliuoti šildymo valdymą

Išvados. Atlikus programinių agentų, naudojamų protingoje būsto informacijos valdymo sistemoje, analizę buvo nustatyta, kad visi sistemos agentai turi komandinių agentų klasei būdingų savybų – bendradarbiavimas ir autonomija. Visi agentai savo užduotis atlieka su kitų agentų pagalba, dėl to akivaizdu, kad didesnę dalis sistemos agentų priklauso komandinių agentų klasei. Taip pat dauguma programinių agentų turi apsimokymo savybę, nors ši savybė nėra labai aukšto sudėtingumo lygio – agentai apsimoko stebėdami aplinką (gaudami orų prognozes iš interneto ar sistemos duomenų bazės), bendraudami su vartotoju (gaudami informaciją apie įvairius būsto parametrus), bendraudami su kitais sistemos agentais (gaudami duomenis apie agento būklę). Be komandinių agentų klasės vyrauja dar dvi agentų klasės: sąsajos agentas – veikimo principas yra įprastas šiai klasei (agento bendravimas su vartotoju ir vartotojo stebėjimas), bei reaktyvus agentas – dažniausiai šio agento veiksmus sužadina aplinkos veiksniai (pasikeitusi temperatūra ir pan.), mūsų sistemos atveju šis agentas suveikia atsiradus sistemos vidiniam veiksniam, t.y. negavus atsako iškito agento – klaidų aptikimo agentas pakeičia neveikiantį agentą nauju. Nors dauguma sistemos agentų savybės ir klasės kartojasi, tačiau peržiūrėjus agentų paskirtį aiškiai matyti, kad kiekvienas programinis agentas turi savo individualias roles ir jų atliekami veiksmai nesidubliuoja. Kadangi sistemos

užduotys atliekamos su kelių agentų pagalba – sistemą galima vadinti daugiaagentine sistema. Agentai tarpusavyje bendrauja žinučių pagalba, dėl to reikėtų sudaryti tam skirtą bendravimo kalbą.

2.4 Agentų kūrimo programinės įrangos analizė

Daugiaagentinių sistemų programavimo kalbos, platformos ir plėtros įrankiai yra svarbi dalis, kuri gali paveikti tai, kaip gerai agentų technologijos yra įsisavinamos tarp skirtingų taikymo sričių [14, 26, 27]. Todėl galima būtų teigti, kad daugiaagentinių sistemų pasisekimas smarkiai priklauso nuo tinkamų programinių agentų kūrimo technologijų būvimo. Įvairios programavimo kalbos, programų bibliotekos ir plėtros įrankiai nulemia programinių agentų technologijų įsisavinimo ir pritaikymo greitumą ir paprastumą [26, 27].

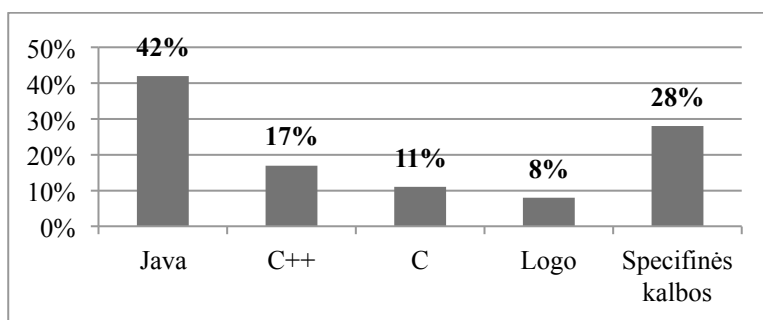
Daugiaagentės sistemos gali būti kuriamos ir realizuojamos su praktiškai bet kokia programavimo kalba. Objektinio programavimo kalbos yra vienos iš labiausiai tinkamų įrankių, kadangi agento sąvoka nedaug skiriasi nuo objekto sąvokos [14, 20]. Programiniai agentai ir objektai turi daug bendrų savybių, tokių kaip – kapsuliacija, paveldėjimas ir žinučių perdavimas [14, 20]. Tačiau taip pat programiniai agentai ir objektai tarpusavyje taip pat skiriasi – agentai yra autonomiški (programinis agentas pats nusprendžia ar atlikti kitų agentų reikalaujamus veiksmus), taip pat jie (agentai) gali turėti lanksčias elgsenas ir kiekvienas sistemos agentas turi nuosavą, atskirą giją [14, 20].

Programinių agentų kūrimo proceso metu dažniausiai susiduriama su dviem pagrindinėmis problemomis. Pirmoji problema yra kompetencija – kokiais būdais agentas įgauna žinias, kurios jam reikalingos priimančias sprendimus kada ir kaip padėti vartotojui [28]? Programinio agento kompetencija priklauso nuo sistemos programuotojo gebėjimų įgyvendinti visas užsibrėžtas agento ypatybes. Antroji problema yra pasitikėjimas – kaip galima būtų užtikrinti, kad vartotojas, skirdamas užduotį agentui, jausis patogiai [28] (pvz. programiniam agentui patikinti namų apšvietimo sistemos valdymą)? Pastarosios problemos aktualumas šiomis dienomis tampa vis mažesnis, kadangi su tobulėjančiomis programinių agentų kūrimo technologijomis pats agentų kūrimo procesas paprastėja ir agentai tampa patikimesni. Taip pat verta paminėti, kad plintant informacinėms technologijoms – vartotojai gali gauti daugiau žinių apie jas, kartu ir apie programinius agentus, ko pasekoje auga pasitikėjimas agentais.

Priežastys kodėl programinių agentų kūrėjai naudoja agentų kūrimo programinę įrangą yra tos pačios kaip ir programinės įrangos kūrėjų, dirbančių su objektiniu programavimu. Pastarieji dažniausiai teikia pirmenybę tam tikroms programų kūrimo aplinkoms [28] kaip pvz. „Java NetBeans“ ar „Microsoft Visual Basic“. Tokios programinės įrangos kūrimo

aplinkos suteikia tam tikrą abstrakcijos lygį, kuriame programuotojai gali kurti objektus [28], taip pat į šias aplinkas yra įtrauktos vizualaus programavimo ypatybės, kurios taupo laiką ir daro darbą lengvesnį ir patrauklesnį [28]. Papildomai programinės įrangos kūrimo aplinkos suteikia vartotojui testavimo ir klaidų taisymo terpes, leidžia programuotojams pakartotinai panaudoti klases (objektus), sukurtas kitų programuotojų [28].

Tačiau dabar esančios objektinio programavimo programinės įrangos kūrimo aplinkos palaiko ne visus agentų kūrimo aspektus – šios platformos neįgyvendina agentams būdingų savybių: agentų tarpusavio sąveikos taisyklių, agentų bendravimo kalbos [28]. Dėl šios priežasties atsirado agentų kūrimo programinė įranga, kuri suteikia tokią kurimo aplinką, kuri pilnai palaiko programinių agentų kūrimą, testavimą ir pakartotinį panaudojimą [28].

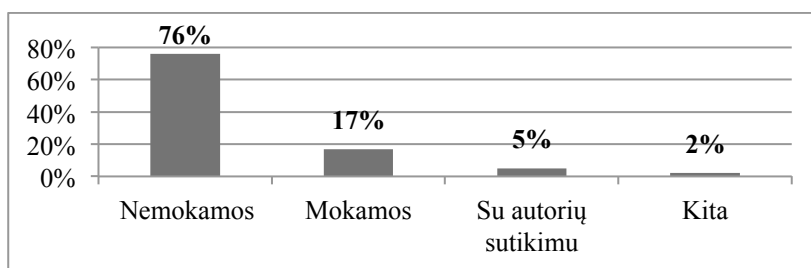


4 pav. Agentų kūrimo programinės įrangos kalbų pasiskirstymo diagrama [29]

Remiantis Cynthia Nikolai ir Gregory Madey atlikto agentų modeliavimo platformų tyrimo duomenimis [29] buvo sudaryta keletas diagramų, kurios apibrėžia svarbiausius agentų kūrimo programinės įrangos aspektus: naudojama programavimo kalba (4 pav.) – tai padeda pasirinkti keliamų programavimo kalbų reikalavimus atitinkantį produktą; produkto licencijos (5 pav.) – leidžia pasirinkti įrankį su palankiausiomis disponavimo taisyklėmis ir būdais; veikimui reikalinga operacinė sistema (OS) (6 pav.) – nusako sistemos kūrimui ir veikimui reikalingus OS tipus; įrankio taikymo sritis (7 pav.) – tai padeda pasirinkti labiausiai kuriamos sistemos paskirtį atitinkantį produktą; ir palaikymas (8 pav.) – leidžia įvertinti įrankio aptarnavimo aspektus ir darbo su produktu įsisavinimo kaštus [26] (pvz. neesant detaliam įrankio dokumentacijai – reikia skirti papildomų pastangų arba lėšų, norint išmokyti dirbti su įrankiu).

Iš agentų kūrimo programinės įrangos kalbų pasiskirstymo diagramos (4 pav.) matyti, kad didžiausią agentų kūrimo programinės įrangos rinkos dalį sudaro įrankiai, kurie veikia su „Java“ programavimo kalba. Tai galima būtų paaiškinti „Java“ programavimo kalbos paplitimu ir tarp įprastų programinės įrangos kūrimo įrankių [26], plačių panaudojimo galimybių – nebūtina „Java“ programų kurti kiekvienai operacinei sistemai atskirai, užtenka

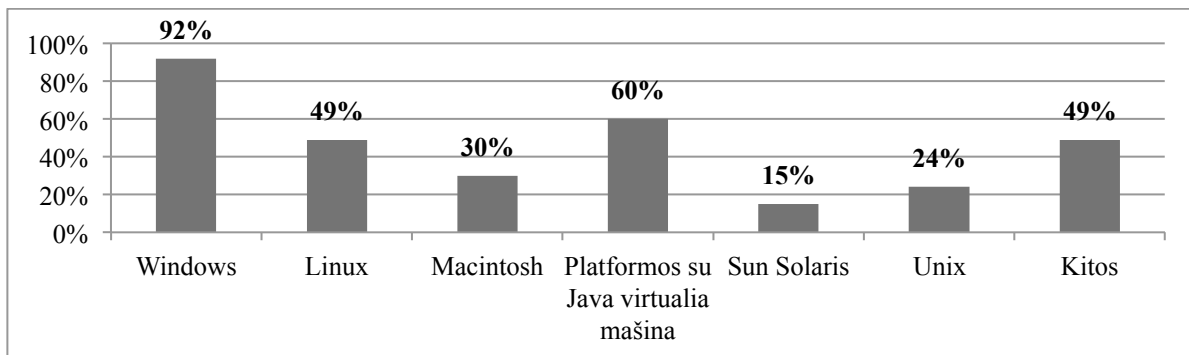
sukurti programą vienoje operacinės sistemos aplinkoje ir kitose sistemose „Java“ programa veiks analogiškai. Kita didelė agentų kūrimo programinės įrangos kalbų rinkos dalis priklauso specifinėm kalbom, dauguma šių programavimo kalbų yra specifiskai sukurtos naudojimui kartu su savo specifinės paskirties įrankiais [29]. Šių kalbų paplitimą tarp agentų kūrimo programinės įrangos galėtų lemti tai, kad ne visos programavimo kalbos gali tenkinti specifinius kuriamų agentinių sistemų reikalavimus, dėl to pastarajai problemai spręsti patogiausiai yra sukurti atskirą programavimo kalbą. Verta pastebėti kad bendra agentų kūrimo programinės įrangos kalbų suma iš diagramos sudaro 106%, tai nutinka dėl to, kad dalis įrankių gali veikti su keliomis skirtingomis programavimo kalbomis [29]. „C“, „C++“ ir „Java“ programavimo kalbos dažniausiai yra naudojamos bendrosios paskirties įrankių [29], tai rodo, kad didžioji dauguma agentų kūrimo programinės įrangos taikymo sričių yra pakankamai plačios ir tinkamos įvairiems agentinių sistemų kūrimo atvejams. Svarbu pabrėžti kad visos kalbos, atvaizduotos agentų kūrimo programinės įrangos diagramoje (4 pav.) yra naudojamos agentų programavimui, o ne pačių įrankių kūrimui [29].



5 pav. Agentų kūrimo programinės įrangos licencijų tipai [29]

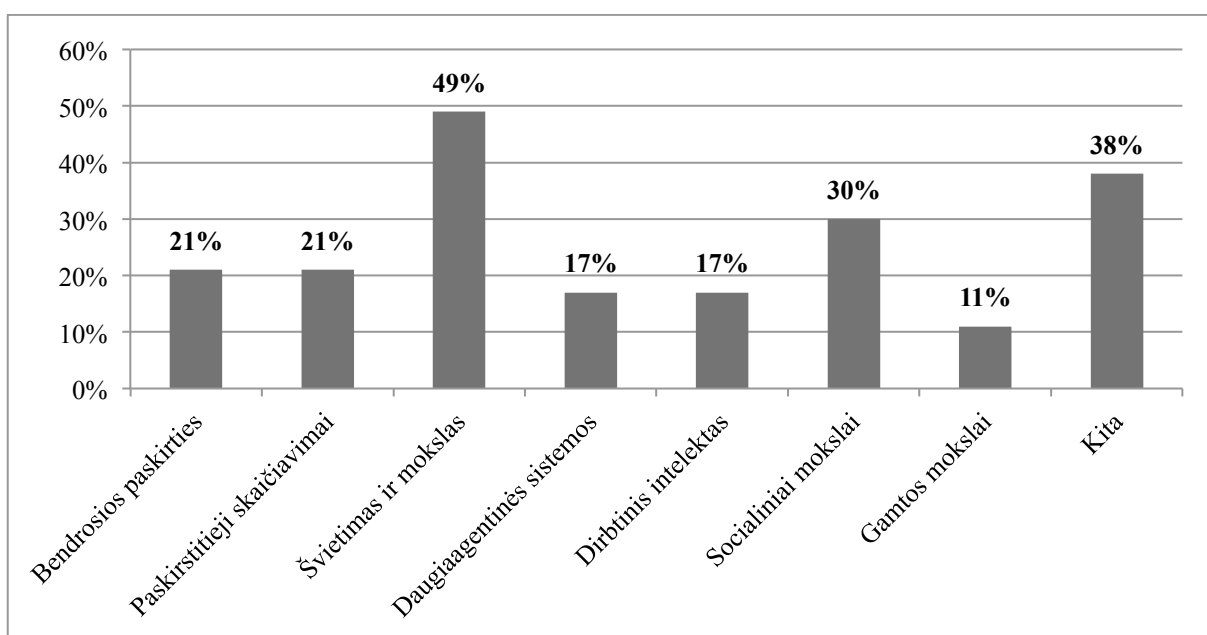
Agentų kūrimo programinė įranga gali turėti keletą licencijų tipų, tačiau, pažiūrėjus į licencijų tipų diagramą (5 pav.) aiškiai matyti, kad pagrinde vyrauja nemokamos licencijos tipas – jis sudaro net 76% visų licencijų tipų. Įrankių nemokamas licencijas sudaro atviro kodo (56%), uždaro kodo (9%) ir nemokamos, tenkinant papildomus reikalavimus (14%) (angl. free with restrictions) licencijos [29]. Atviro kodo licencijos įrankiai išleidžiami kartu su išeities kodais ir leidimais įvairioms įrankio modifikacijoms, uždaro kodo licencijos įrankiai neviešina savo išeities kodų [29]. Nemokamos, tenkinant papildomus reikalavimus, įrankių licencijos – yra nemokamos, jei tenkinamos visos numatytos agentų kūrimo programinės įrangos naudojimo sąlygos [29] (pvz. įrankis yra nemokamas, jei naudojamas akademiniais ar nekomerciniais tikslais). Galimai viena iš pagrindinių priežasčių, kodėl taip įrankių taip paplitusios atviro kodo licencijos yra ta, kad tai įgalina programinių agentų kūrėjus, esant reikalavimams, labai paprastai patobulinti savo turimus įrankius. Mokamų licencijų įrankiai įsigijami vartotojui sumokėjus įrankio autoriams. Mokamų licencijų agentų kūrimo programinė įranga sudaro santykiniai mažą rinkos dalį, t. y. apie 17%, greičiausiai ši

faktorių lemia tai, kad programinių agentų technologijos yra pakankamai naujos, o mokamų produktų kokybės standartai dažniausiai yra aukšti. Įrankiai, platinami pagal autorių sutikimo licenciją, užima 5% rinkos dalies, šiuo atveju agentų kūrimo programinė įranga platinama pagal jos autorių sutikimą, atsižveldiant į įrankio naudojimo aplinkybes.



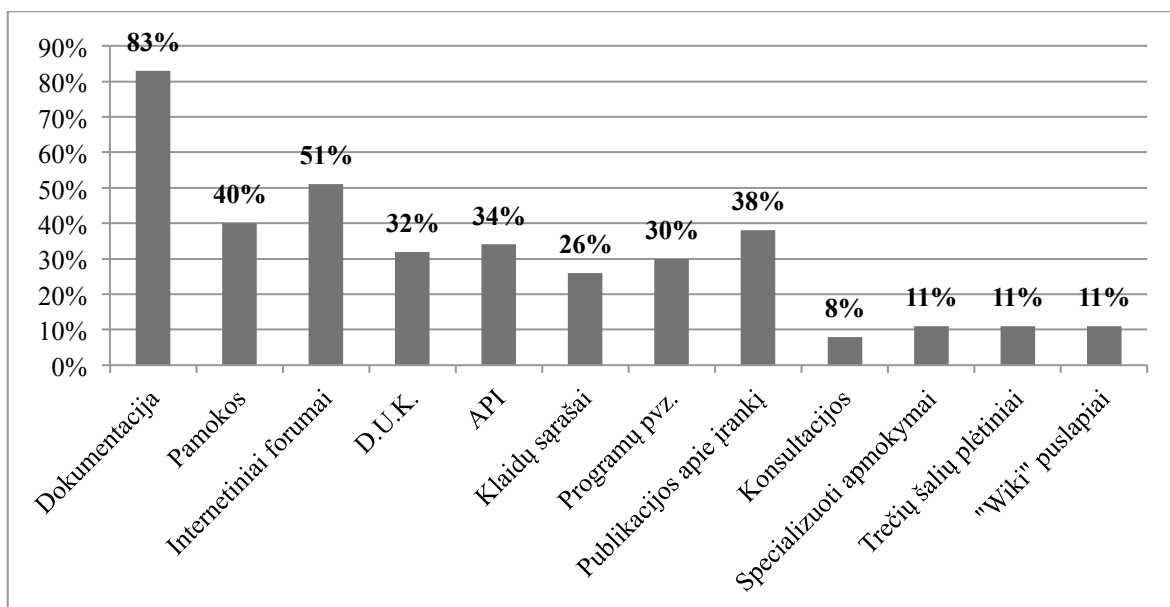
6 pav. Agentų kūrimo programinės įrangos operacinės sistemos [29]

Iš agentų kūrimo programinės įrangos operacinių sistemų diagramos (6 pav.) aiškiai matyti, kad labiausiai tarp įrankių paplitusi operacinė sistema – „Microsoft Windows“, tai greičiausiai yra todėl, kad „Windows“ yra labiausiai paplitusi operacinė sistema pasaulyje ir šis reiškinys atsispindi diagramoje. Taip pat pastebimas didelis populiarumas platformų, kurios palaiko „Java“ virtualią mašiną (t.y. tos sistemos, kuriose veikia „Java“ programos), tai galėtų lemti programavimo su „Java“ paprastumas ir platus pritaikymo operacinėms sistemoms mastas [29, 26]. Vertėtų paminėti tai, kad bendra diagramos elementų suma gerokai viršija 100% (t.y. 319%) – taip įvyksta dėl to, kad didelė dalis agentų kūrimo programinės įrangos gali būti naudojama su keliomis skirtingomis operacinėmis sistemomis.



7 pav. Agentų kūrimo programinės įrangos taikymo sritys [29]

Agentų kūrimo programinės įrangos taikymo sričių diagrama (7 pav.) parodo įrankių pagrindinių taikymo sričių pasiskirtymą, reikia paminėti kad tas pats įrankis dažniausiai turi kelias taikymo sritis [29], dėl to bendra diagramos elementų suma yra 204%. Didžiausią dalį (49%) sudaro švietimo ir mokslo sričių įrankiai, tai galima paaiškinti tuo, kad švietimo ir mokslo sferoje yra didelis susidomėjimas programinių agentų technologijomis [28], be to dauguma įrankių, taikomų kitose srityse, gali vienodai gerai būti taikomi ir pedagoginėse srityse [29]. Likusios taikymo sritys užima sąlyginiai panašias pozicijas. Mūsų eksperimentinė sistema yra daugiaagentinė sistema ir, matant iš agentų kūrimo programinės įrangos taikymo sričių diagramos, tokios paskirties įrankiai yra pakankamai gerai paplitę (17%). Kitos taikymo sritys sudaro 38% agentų kūrimo įrankių rinkos, tačiau šios sritys yra labai specifinės ir turi vos po kelis jų sričiai pritaikytus įrankius [29].



8 pav. Agentų kūrimo programinės įrangos palaikymas [29]

Iš agentų kūrimo programinės įrangos palaikymo diagramos (8 pav.) aiškiai matoma, kad įrankis gali turėti platų palaikymo priemonių spektrą. Aktualiausi palaikymo būdai (dokumentacija – 83%, pamokos – 40%, forumai – 51%, programų pvz. – 30%) yra plačiai paplitę, todėl įrankio įsisavinimo laikotarpis turėtų būti minimalus. Kadangi praktiškai visi įrankiai turi bent po kelis palaikymo būdus – bendra diagramos elementų procentinė suma yra didesnė už 100% (375%).

Išanalizavus agentų kūrimo programinę įrangą, atrinkome kelis populiariausius įrankius, vienas iš jų yra ABLE. ABLE yra „IBM T.J. Watson Research Center“ projektas [30], kuris yra Java pagrindu veikiantis įrankis, turintis komponentų bibliotekas sumaniųjų agentų (turinčių apsimokymo ir motyvacijos savybes) kūrimui [31]. ABLE platforma pateikia

pagrindinių Java klasių ir sąsajų rinkinius, kurie yra naudojami „JavaBeans“ (JavaBeans - pakartotinio panaudojimo programiniai Java komponentai) bibliotekų, vadinamų „AbleBeans“, sukūrimui [30]. Bibliotekos turi „AbleBeans“ komponentus, skirtus: tekstų ir duomenų bazių skaitymui ir rašymui, duomenų manipuliavimui, miglotiesiems skaičiavimams, neuroniniams tinklams [30] ir pan. Programinės įrangos kūrėjai turi galimybę prabėsti egzistuojančius „AbleBeans“, ar susikurti savo specifinius algoritmus [31]. ABLE veikia su bet kokia platforma, palaikančia Java 2 versiją [30].

Kitas įrankis – „Cougaar“. Šis agentų kūrimo įrankis yra „Java“ programinė įranga, palengvinanti sudėtingų, plataus masto ir paskirstytų skaičiavimų agentų kūrimo procesą [31]. „Cougaar“ yra labiau skirtas lanksčių ir išmanių tinklų kūrimui, o ne simuliacijai ar modeliavimui [31]. Ši atviro kodo agentų kūrimo programinė įranga turi platų demonstracijos, vizualizacijos ir valdymo komponentų spektrą [32]. Šis įrankis buvo sukurtas kaip dalis DARPA didelio masto agentinių sistemų, skirtų karinei logistikai, tyrimo projekto [32].

Dar vienas plačiai naudojamas programinių agentų kūrimo įrankis yra JADE. JADE siekia supaprastinti daugiaagentinių sistemų diegimą, panaudojant tarpinę programinę įrangą, kuri veikia pagal FIPA specifikacijas ir palaiko įvairias klaidų taisymo bei agentų dislokavimo funkcijas [14]. Agentų platforma gali veikti skirtingose mašinos (kurios nebūtinai turi turėti tas pačias operacines sistemas) ir platformos konfigūracija gali būti valdoma iš nutolusios vartotojo sąsajos [14]. Taip pat agentų platformos konfigūracija gali būti keičiama veikimo metu – reikiamu metu perkeliant agentus iš vienos mašinos į kitą [14]. JADE yra labiau skirtas lanksčių ir išmanių tinklų kūrimui, o ne simuliacijai ar modeliavimui [31]. Ši agentų kūrimo programinė įranga veikia su „Java“ platforma (reikalinga Java JDK-1.4 versija) [14]. JADE yra platinama „Telecom Italia“ pagal LGPL-2 licenciją [14]. Naujausia versija (4.0.1) buvo išleista 2010 liepą [31].

Dar vienas įrankis – „SimAgent“. Ši programinė įranga yra sukurta Aaron Sloman iš Birmingamo universiteto [33]. Šis įrankis yra skirtas greitai pritaikyti ir testuoti skirtingas agentų architektūras, kuriose agentas susideda iš kelių tarpusavyje kartu bendraujančių posistemų ir (agentas) yra aplinkoje, kurioje gali egzistuoti kiti agentai ar objektai [31]. Agentai gali turėti jutiklių ir vykdyklių, bendrauti su kitais agentais, taip pat agentai gali turėti ir hibridinių savybių – mechanizmų, leidžiančių bendrauti neuroniais tinklais [31]. „SimAgent“ naudoja „Pop-11“ programavimo kalbą ir „Poplog“ programinės įrangos kūrimo aplinką [33].

Atlikus agentų kūrimo programinės įrangos analizę buvo sudaryta įrankių palyginimo lentelė (3 lentelė). Įrankių lyginimui buvo pasirinkti šie aspektai: programavimo kalba – galima atrinkti įrankį, kuo optimaliau atitinkantį sistemos agentų kūrimo reikalavimus; licencijos tipas – šis aspektas leidžia pasirinkti įrankį su palankiausiais disponavimo terminais; operacinė sistema – palengvina agentų kūrimo programinės įrangos pritaikymą savo turimai aplinkai; taikymo sritis – tai padeda atsirinkti įrankį, kuris yra pritaikytas darbui su kuriama sistema; palaikymas – leidžia prognozuoti darbo su įrankiu įsisavinimo laiką.

3 lentelė. Agentų kūrimo programinės įrangos palyginimas

Pavadinimas	Programavimo kalba	Licencijos tipas	Operacinė sistema	Taikymo sritis	Palaikymas
ABLE	Able rule language (ARL)	Atviro kodo (nemokama akademiniam ir nekomerciniam tikslams)	Windows 95, Windows 98, Windows NT, UNIX, Java-2 JVM palaikančios platformos	Sumaniųjų programinių agentų kūrimas	D.U.K., pamokos, forumai, publikacijos apie įrankį, API, dokumentacija
Cougaar	Java	Cougaar atviro kodo licencija (COSL) – modifikuota atviro kodo licencija	Windows 98, Windows NT, Windows XP, Linux, Mac OS, Java 1.4 palaikančios platformos	Daugiaagentinės didelio masto, sudėtingos sistemos	D.U.K., pamokos, forumai, dokumentacija
JADE	Java	LGPL 2 versija	Bet kokia Java palaikanti platforma	Daugiaagentinės sistemos, susidedančios iš autonominių agentų	D.U.K., klaidų sąrašai, pamokos, programų pavyzdžiai, API, dokumentacija
SimAgent	Pop-11	Atviro kodo - nemokama	Windows NT, Windows XP, Windows Vista, Windows 7, Linux, UNIX, Mac OS	Įvairaus sudėtingumo aplinkose sąveikaujančių agentų tyrimas	Pamokos, dokumentacija, publikacijos apie įrankį, programų pavyzdžiai

Išvados. Palyginimui parinkti įrankiai (3 lentelė) pasižymėjo panašiomis savybėmis: visos licencijos leidžia naudotis įrankiais nemokamai, taip pat visi įrankiai turi platų palaikymo paslaugų spektrą. Esminiai skirtumai tarp agentų kūrimo programinės įrangos yra: skirtingos įrankių taikymo sritys ir programavimo kalbos. Vertėtų pastebėti tai, kad visi įrankiai gali būti naudojami su pakankamai plačiu operacinių sistemų spektru.

Protingos būsto informacijos valymo sistemos kūrimui pasirinkome JADE įrankį, dėl keleto svarių priežasčių. Visų pirma būsto kompiuterizavimo sistema, į kurią diegiama darbo metu sukurta sistema, yra sukurta JADE programinių agentų sistemos principu. JADE agentų kūrimo programinės įrangos paskirtis labiausiai atitiko mūsų kuriamos sistemos kriterijus –

kuriama sistema yra daugiaagentė sistema, šiuo aspektu tiktų ir Cougaar įrankis, tačiau jis labiau skirtas platesnio masto sistemoms, kas mūsų darbe nėra būtina. Taip pat JADE turi platų palaikymo būdų spektrą, kas palengvins pačios sistemos kūrimą. LGPL licencija įgalina kurti protingą būsto informacijos valdymo sistemą be griežtų disponavimo sąlygų. JADE kuriami agentai yra programuojami „Java“ programavimo kalba, dėl to sistema turi platų pritaikymo prie operacinių sistemų spektrą. Taip pat nustatyta kad JADE palaiko FIPA standartą, kuris užtikrina kuriamų programinių agentų suderinamumą su ateityje kuriamais agentais.

2.5 Išvados

Atlikus probleminės srities analizę buvo pasirinkti tinkamiausi analizuotų technologijų pritaikymo protingos būsto informacijos valdymo sistemos kūrimui būdai. Atlikus protingų būsto informacijos valdymo sistemų analizę buvo išsiaiškinti esminiai jų veikimo principai, palyginti jau sukurti produktai. Išanalizavus įvairius literatūros šaltinius nustatyta, kad programiniai agentai nėra naudojami šiose sistemose, o poreikis protingų būsto informacijos valdymo sistemų, naudojančių programinių agentų technologijas, yra vis didėjantis ir šioje srityje yra vystomos technologijos.

Taip pat atlikus programinių agentų analizę buvo atrinkti sistemos veikimui labiausiai tinkantys agentų tipai, kurie dažniausiai turi bendradarbiavimo ir autonomijos savybių. Išanalizavus literatūros šaltinius, buvo nustatyta, kad labiausiai tinka protingom namų informacijos valdymo sistemom ne tik pavieniai agentai, bet ir daugiaagentės sistemos. Daugiaagentinės sistemos yra pranašesnės tuo, kad jos yra skirtos spręsti sudėtingiems paskirstytų skaičiavimų uždaviniams, be to šias sistemas yra lengva atnaujinti ir išplėsti. Sistemą nusprestą kurti, kaip daugiaagentę sistemą, dėl akivaizdžių šio sprendimo būdo privalumų.

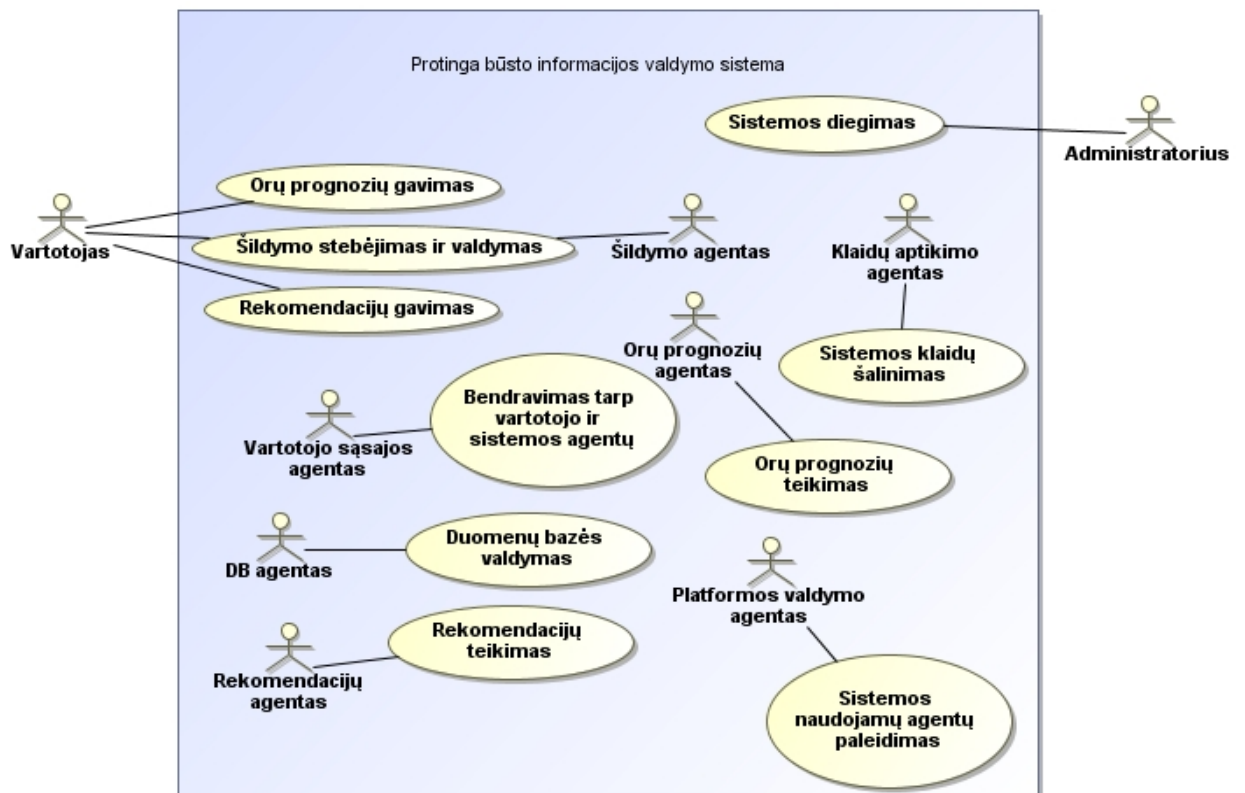
Atlikus programinės įrangos analizę išsiaiškinome esminius programinių agentų kūrimo įrankių skirtumus, privalumus ir trūkumus, ir sistemos kūrimui pasirinkome JADE įrankį. JADE įrankis buvo pasirinktas dėl savo paprasto naudojimo (naudoja Java programavimo kalbą), pilnai palaiko visiškai autonominių agentų kūrimą, pilnai palaiko daugiaagentinių sistemų kūrimą, taip pat palaiko FIPA ir ACL standartus, kurie užtikrina programinių agentų suderinamumą su vėlesnėmis sistemomis.

3 PROTINGOS NAMŲ INFORMACIJOS VALDYMO SISTEMOS MODELIS

Protingos namų informacijos valdymo sistemos modelio dalyje sudaromas sistemos modelis, kuris susideda iš protingos būsto informacijos valdymo sistemos funkcinio modelio, sistemos struktūros, duomenų mainų tarp sistemos agentų modelio ir rekomendacijų agento darbo modelio. Sistemos funkcinis modelis apibrėžia sistemos atliekamas funkcijas, vartotojų tipus taip pat nusako sistemos naudojamus programinius agentus. Sistemos struktūra atvaizduoja sistemos fizinius ir programinius (įskaitant ir agentus) elementus, ryšius tarp jų. Duomenų mainų tarp sistemos agentų modelis parodo informacijos apsikeitimo eigą tarp sistemos agentų funkcijų atlikimo metu, taip pat atvaizduojama sistemos duomenų bazės struktūra. Rekomendacijų agento darbo modelis aprašo šildymo sezono kaštų prognozių metodą, bei metodo naudojamų duomenų gavimą.

3.1 Protingos būsto informacijos valdymo sistemos funkcinis modelis

Sistemos funkcinis modelis (9 pav.) atvaizduoja sistemos panaudojimo galimybes, apibrėžia sistemos vartotojus ir jų galimybes – kokias funkcijas jie (vartotojai) gali atlikti. Protingos namų informacijos valdymo sistemos funkcinis modelis susideda iš kelių elementų: aktorių su kuriais sistema sąveikauja, pačios sistemos, panaudos atvejų ar paslaugų, kurias gali suteikti sistema bei ryšių tarp šių elementų. Reikia pastebėti, kad sistemos naudojami programiniai agentai yra atvaizduojami kaip vidiniai sistemos aktoriai, kurie sugeba atlikti įvairias funkcijas, tačiau šių programinių agentų negalima laikyti tikraisiais sistemos vartotojais.



9 pav. Sistemos funkcinis modelis

Kaip matoma iš sistemos funkcinio modelio (9 pav.), pagrindiniai sistemos vartotojai yra – vartotojas ir administratorius. Vartotojai yra skirstomi pagal atliekamas funkcijas ir gaunamas paslaugas. Sistemos agentai yra skirstomi pagal agento tiekiamas paslaugas ar atliekamas funkcijas.

Esminės vartotojo funkcijos yra: orų prognozių gavimas, šildymo stebėjimas ir valdymas, rekomendacijų gavimas. Orų prognozių gavimas – sistema vartotojui grafiškai atvaizduoja artimiausių dienų pasirinktos vietos orų prognozes, taip pat atvaizduojama dabartinė vietos lauko temperatūra ir oras (pvz. apsiniaukę, lietus ir pan.), orų prognozių vieta yra išsaugoma vėlesnėms reikmėms. Šildymo stebėjimas ir valdymas – sistema gali realiu laiku atvaizduoti esamą kambario temperatūrą, šildymo būseną (įjungta ar išjungta), taip pat vartotojas gali valdyti šildymą, keisdamas norimą kambario temperatūrą. Rekomendacijų gavimas – sistema gali pateikti vartotojui preliminaras pasirinkto mėnesio šildymo kainas, šiam tikslui pasiekti vartotojas turi nurodyti norimą kambario temperatūrą pasirinktam mėnesiui.

Pagrindinė administratoriaus atliekama funkcija yra sistemos diegimas. Sistemos diegimas – sistemos naudojamos programinės įrangos diegimas ir paleidimas: duomenų bazės valdymo sistemos paleidimas, protingos būsto informacinės sistemos naudojamų programinių agentų ir bibliotekų diegimas ir paleidimas.

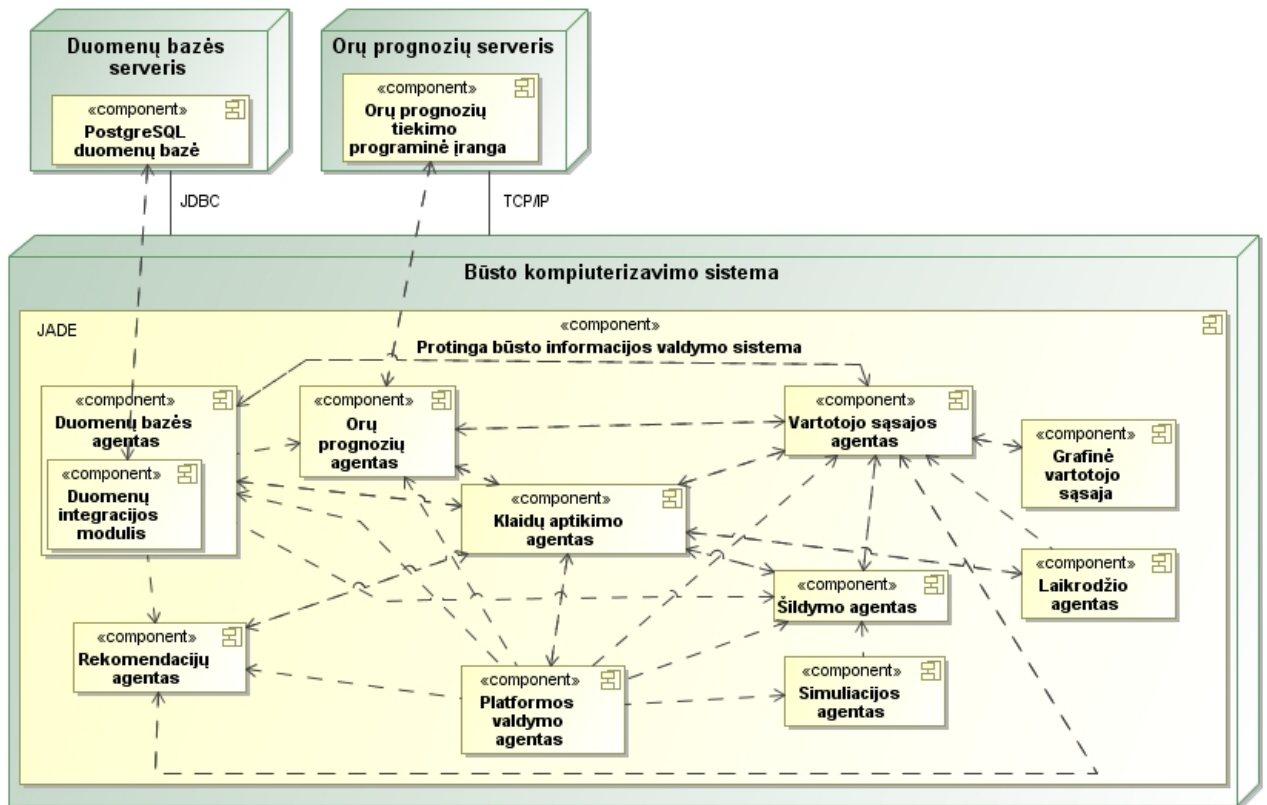
Be aktorių, kurie atvaizduoja sistemos vartotojus, sistemos funkcinis modelis (9 pav.) turi ir kitus aktorius, kurie yra atvaizduoti sistemos viduje, šie aktoriai parodo sistemos naudojamus programinius agentus. Kiekvienas agentas yra susietas su savo atliekamom funkcijom ir tiekiamom paslaugom. Šildymo stebėjimas ir valdymas – ši funkcija persidengia su vartotojui tiekiamom paslaugom, ją atliekantis šildymo agentas užtikrina paslaugos tiekimą vartotojui, stebėdamas temperatūrą ir atitinkamai priimdamas sprendimą (palyginant esamą temp. su vartotojo nustatyta norima temp.) ar reikia įjungti arba išjungti šildymą. Sistemos klaidų šalinimas – šią funkciją atlieka klaidų aptikimo agentas, nuolat tikrindamas ar visi sistemos programiniai agentai veikia, aptikus neveikiantį agentą – šis yra pakeičiamas nauju. Bendravimas tarp vartotojo ir sistemos agentų – vartotojo sąsajos agentas stebi vartotojo atliekamus veiksmus su grafine vartotojo sąsaja ir pagal vartotojo reikalavimus siunčia žinutes kitiems agentams apie turimus atlikti uždavinius, taip pat šis agentas pateikia iš kitų agentų gautus užduočių rezultatus vartotojui. Duomenų bazės valdymas – duomenų bazės agentas atlieka visus veiksmus su sistemos duomenų baze ir teikia informaciją kitiems agentams pareikalavus. Orų prognozių teikimas – šią funkciją atlieka orų prognozių agentas, rasdamas internete naujausias orų prognozes ir pateidamas jas vartotojui. Rekomendacijų teikimas – rekomendacijų agentas atlieka šildymo kainų prognozių skaičiavimus ir rezultatus perduoda vartotojui. Sistemos naudojamų agentų paleidimas – šią funkciją atlieka platformos valdymo agentas, kuris gali sukurti ar panaikinti bet kurį sistemos programinį agentą.

Išvados. Iš nubraižyto sistemos funkcinio modelio (9 pav.) galima pamatyti, kad sistemos vartotojai yra du: vartotojas ir administratorius. Pagrindinės sistemos funkcijos yra įvairių paslaugų teikimas vartotojui: orų prognozių gavimas, šildymo stebėjimas ir valdymas, rekomendacijų gavimas. Taip pat iš diagramos galima nustatyti tai, kad daugelį sistemos funkcijų atlieka programiniai agentai, o tai pagreitina praleidžiamą laiką dirbant su sistema ir pagerina praleidžiamo laiko kokybę, kas yra aktualu kiekvienam vartotojui. Taip pat kiekvienas programinis agentas atlieka savo atskiras užduotis, kurios nesidubliuoja, taip optimaliau išnaudojant sistemos turimus išteklius (neatliekant vienodų užduočių keliems agentams).

3.2 Sistemos struktūra

Sistemos struktūra (10 pav.) atvaizduoja protingą namų informacijos valdymo sistemą, į ją įtraukiant tiek fizinius elementus, tiek suprogramuotus komponentus. Protinga būsto informacijos vadymo sistema, susideda iš: duomenų bazės serverio, orų prognozių serverio ir kompiuterizuoto būsto serverio. Reikia pastebėti, kad naudojamas orų prognozių serveris ir jo programinė įranga nebuvo sukurta darbe – naudojamos internetinės orų prognozių tiekimo

paslaugos, tačiau šis elementas yra neatsiejama sistemos dalis. Kiekvienas sistemos realizacijos elementas savyje gali turėti kitus programinės įrangos komponentus – duomenų bazes, programinę įrangą ar programinius agentus. Elementai tarpusavyje yra sujungti ryšiais, kurie turi savo pavadinimus, komponentai tarpusavyje sujungti priklausomybės ryšiais, kurie atvaizduoja kokių komponentų tiekiami ištekliai (duomenys, rezultatai ir kt.) yra reikalingi komponento darbui.



10 pav. Sistemos struktūra

Duomenų bazės agentas pasinaudodamas duomenų integracijos moduliui atlieka duomenų bazės valdymo funkcijas – skaitymą, rašymą, trynimą. PostgreSQL duomenų bazės komponentas yra duomenų bazės serverio elemente, kuris su sistema bendrauja naudodamasis JDBC tvarkykle (angl. driver). Kaip matoma iš sistemos struktūros (10 pav.) dauguma protingos būsto informacijos valdymo sistemos naudojamų agentų turi priklausomybės ryšius su duomenų bazės agentu, taip yra dėl to, kad beveik visi agentų darbui reikalingi duomenys yra saugomi duomenų bazėje, taip kontroliuojant priėjimą prie sistemos duomenų.

Orų prognozių agentas bendrauja su orų prognozių tiekimo programine įranga per TCP/IP protokolą. Taip pat iš sistemos struktūros (10 pav.) matoma, kad orų prognozių agentas susietas priklausomybės ryšiais su keliais kitais agentais, svarbiausias yra ryšys su vartotojo sąsajos agentu – per jį yra pateikiama orų prognozė vartotojui.

Vartotojo sąsajos agentas yra vienintelis protingos būsto informacijos valdymo sistemos agentas, kuris gali bendrauti su grafine vartotojo sąsaja, taip paskirstant kiekvienam sistemos agentui specifines užduotis, kurios nesikartoja – kiti agentai neturi rūpintis informacijos atvaizdavimu vartotojui, jie gali orientuotis į savo atliekamas užduotis. Dėl šios priežasties praktiškai visi sistemos agentai yra sujungti su vartotojų sąsajos agentu, nes kiekvieno agento rezultatus grafinei vartotojo sąsajai perduoda tik vartotojo sąsajos agentas.

Klaidų aptikimo agentas yra susietas priklausomybės ryšiais su visais sistemos agentais, dėl savo atliekamų funkcijų – sistemos agentų stebėjimo ir klaidų taisymo. Platformos valdymo agentas taip pat susietas su visais protingos būsto informacijos valdymo sistemos agentais, kadangi jo atliekamos funkcijos sukuria ir paleidžia visus likusius programinius agentus. Klaidų aptikimo ir platformos agentų veikimas yra glaudžiai susijęs, nes sistemos programinių agentų klaidų šalinimas atliekamas apjungiant abiejų šių agentų vykdomas funkcijas.

Kiti sistemos agentai, kaip matoma iš sistemos struktūros (10 pav.), yra susieti priklausomybės ryšiais su tais komponentais, kurie yra reikalingi jų užduotims atlikti. Rekomendacijų agentas, gavęs skaičiavimui reikalingus duomenis iš duomenų bazės agento, atlieka šilumos kainų prognozes ir rezultatus perduoda vartotojo sąsajos agentui. Šilumos agentas priima sprendimus palygindamas temperatūras – vartotojo išsaugotą norimą temperatūrą su simuliacijos agento generuojamą temperatūrą ir, vartotojo sąsajos agentui pareikalavus, – rezultatus apie esamą šildymo būseną ir temperatūras gražina jam (vartotojo sąsajos agentui). Laikrodžio agentas veikia paprastu principu – kas sekundę siunčia vartotojo sąsajos agentui esamą laiką ir datą.

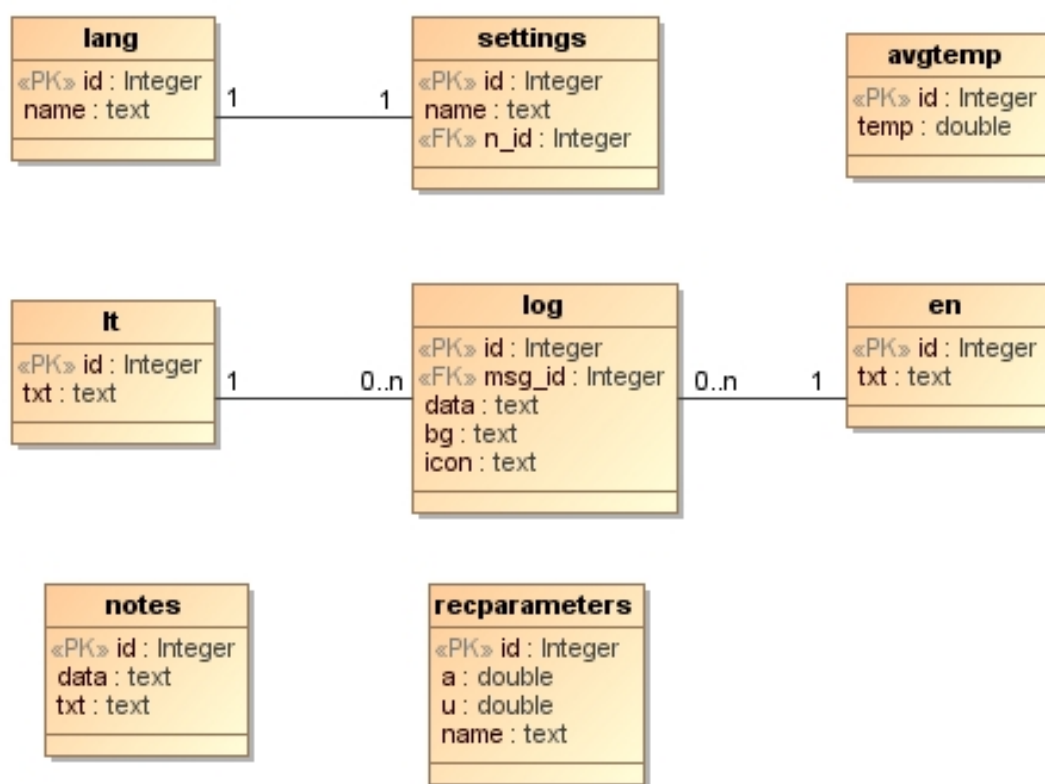
Išvados. Sudarius protingos būsto informacijos valdymo sistemos struktūrą (10 pav.) buvo nustatyta, kad sistemą sudarantys agentai atlieka savo užduotis pasinaudodami kitų agentų pagalba, nes agentai tarpusavyje yra susieti priklausomybės ryšiais, kurie atvaizduoja kitų komponentų (agentų) resursų panaudojimą agento darbui. Dėl to galima teigti kad sistema yra daugiaagentinė sistema. Tokią sistemą yra lengva atnaujinti ir tvarkyti, nes kiekvienas programinis agentas veikia autonomiškai ir jį galima lengvai perprogramuoti ar pakeisti, nesutrikdant visos sistemos veikimo. Šis aspektas yra svarbus, nes šiais laikais technologijos labai greitai tobulėja ir keičiasi ir kitokioms sistemoms yra sunku pritaikyti naujausias technologijas, joms vos tik pasirodžius. Taip pat struktūra (10 pav.) rodo kad sistema susideda iš kelių fizinių elementų: duomenų bazės serverio, kompiuterizuoto būsto sistemos ir orų prognozių serverio - šis serveris nėra realizuotas darbe, nes orų prognozių

agentas naudojami jau sukurtais ir egzistuojančiais internetines orų prognozių paslaugas tiekiančiais serveriais.

3.3 Duomenų mainų tarp sistemos agentų modelis

Duomenų mainų tarp sistemos agentų modelis susideda iš duomenų bazės loginės schemos, duomenų bazės lentelių aprašymų ir sekų diagramų, kurios atvaizduoja vykstančius informacijos mainus tarp agentų, atliekant pagrindines sistemos funkcijas.

Duomenų bazė naudoja PostgreSQL DBVS, beveik visi sistemai reikalingi duomenys yra saugomi duomenų bazėje. Protingos būsto informacijos valdymo sistemos duomenų bazę sudaro aštuonios lentelės (11 pav.).



11 pav. Sistemos duomenų bazės loginė schema

Žemiau pateiktos lentelės su duomenų bazių lentelių įrašais ir jų detalesniais aprašymais (4 lentelė).

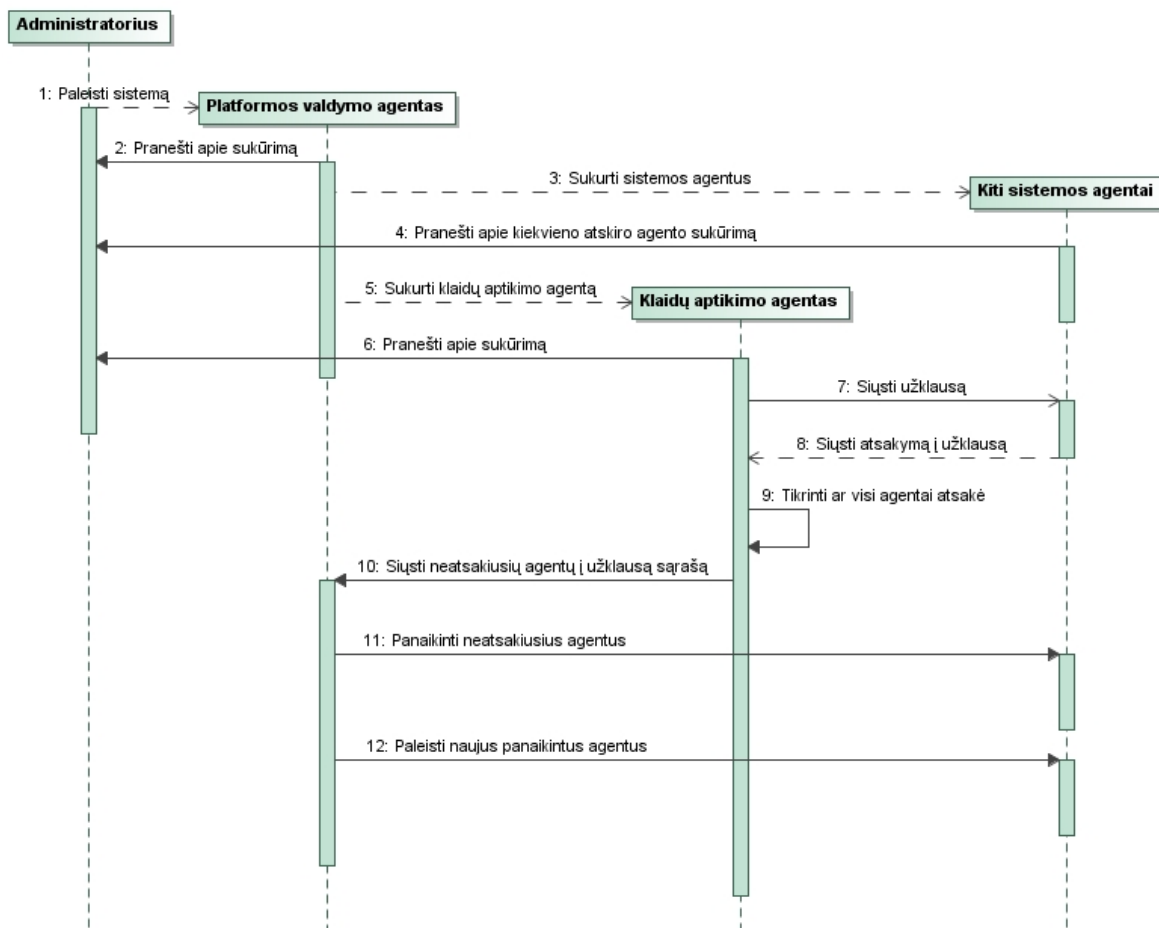
4 lentelė. Duomenų bazės lentelių aprašymai

„lang“ lentelės aprašymas		
Pavadinimas	Tipas	Aprašymas
id	Integer	kalbos numeris, aprašomas integer tipo (sveiko sk.) skaičiumi
name	text	kalbos pavadinimas

4 lentelė. Duomenų bazės lentelių aprašymai

„settings“ lentelės aprašymas		
Pavadinimas	Tipas	Aprašymas
id	Integer	nustatymo numeris, aprašomas integer tipo (sveiko sk.) skaičiumi
name	text	nustatymo pavadinimas
n_id	Integer	nuoroda į vartotojo išsaugoto nustatymo tipo numerį, informacijas apie išsaugotą nustatymą gaunama iš „lang“ lentelės
„avgtemp“ lentelės aprašymas		
Pavadinimas	Tipas	Aprašymas
id	Integer	mėnesio numeris, aprašomas integer tipo (sveiko sk.) skaičiumi
temp	double	mėnesio vidutinė temperatūra, saugoma double tipo (realaus sk.) skaičiumi
„lt“ lentelės aprašymas		
Pavadinimas	Tipas	Aprašymas
id	Integer	lietuvių kalba parašyto teksto numeris, aprašomas integer tipo (sveiko sk.) skaičiumi
txt	text	lietuvių kalba aprašytas tekstas
„en“ lentelės aprašymas		
Pavadinimas	Tipas	Aprašymas
id	Integer	anglų kalba parašyto teksto numeris, aprašomas integer tipo (sveiko sk.) skaičiumi
txt	text	anglų kalba aprašytas tekstas
„log“ lentelės aprašymas		
Pavadinimas	Tipas	Aprašymas
id	Integer	įrašo numeris, aprašomas integer tipo (sveiko sk.) skaičiumi
msg_id	Integer	įrašo teksto numeris, informacija apie įrašą gaunama iš en ir lt lentelių, priklausomai nuo vartotojo pasirinkto nustatymo
data	text	įrašo data
bg	text	kelias iki įrašo fono paveikslėlio
icon	text	kelias iki įrašo piktogramos paveikslėlio
„notes“ lentelės aprašymas		
Pavadinimas	Tipas	Aprašymas
id	Integer	užrašo numeris, aprašomas integer tipo (sveiko sk.) skaičiumi
data	text	užrašo data
txt	text	užrašo tekstas
„recparameters“ lentelės aprašymas		
Pavadinimas	Tipas	Aprašymas
id	Index	rekomendacijos agento naudojamo, šilumą praleidžiančio objekto numeris, aprašomas integer tipo (sveiko sk.) skaičiumi. Informacija apie objektus suveda vartotojas.
a	double	objekto plotas
u	double	objekto šiluminis laidumas
name	text	objekto pavadinimas

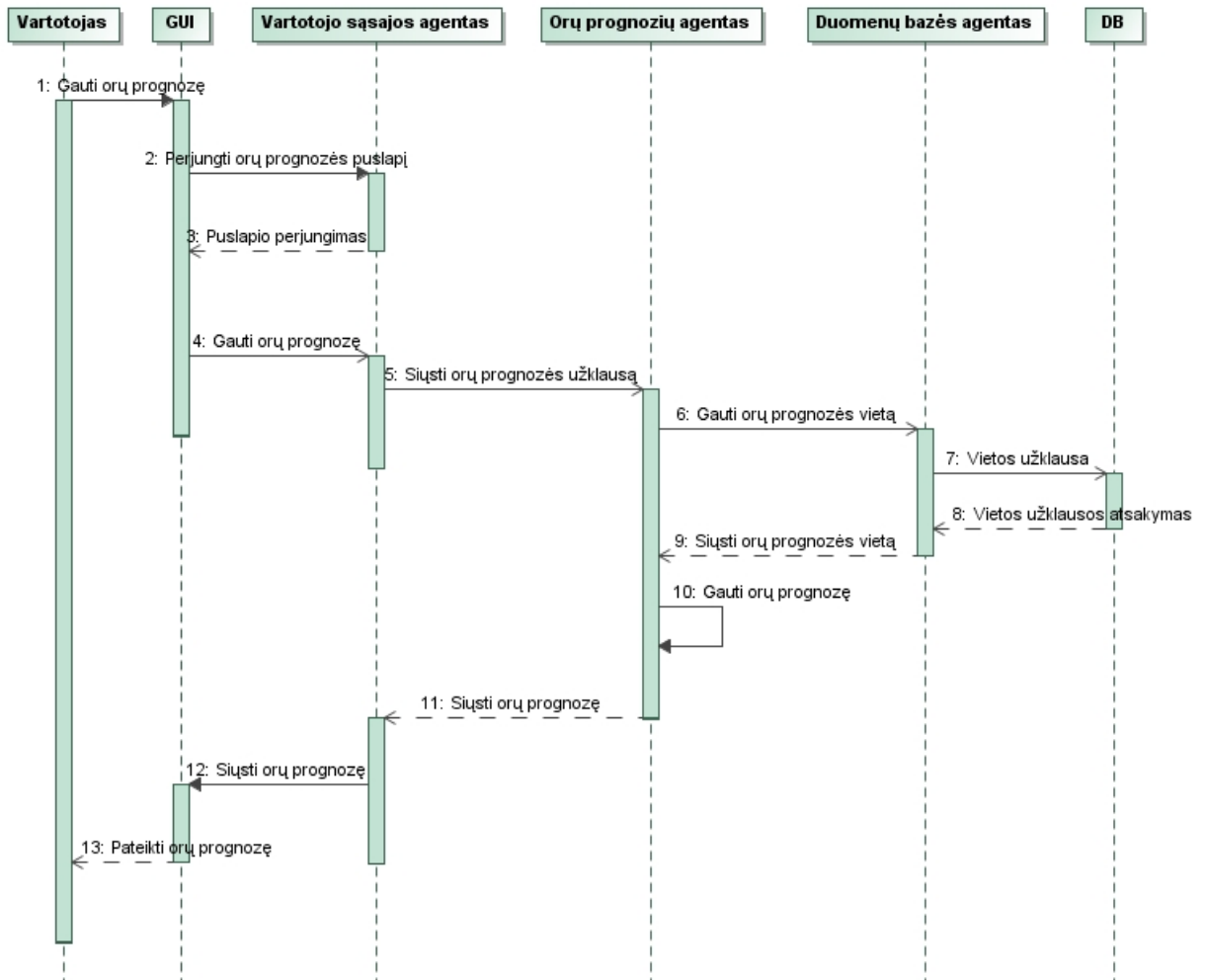
Veiksmų sekos diagramos parodo kaip ir kokia tvarka programiniai agentai veikia tarpusavyje, atliekant įvairias sistemos funkcijas. Nubraižytos veiksmų sekos diagramos atspindi informacijos mainus tarp sistemos programinių agentų, atliekant pagrindines sistemos funkcijas.



12 pav. Sistemos paleidimo ir klaidų aptikimo sekos diagrama

Sistemos paleidimo ir klaidų aptikimo sekos diagrama (12 pav.) atvaizduoja atliekamų veiksmų eiliškumą vykdant sistemos paleidimo ir klaidų aptikimo operacijas. Administratoriui paleidžiant protingą būsto informacijos valdymo sistemą yra sukuriamas platformos valdymo agentas, kuris praneša apie sistemos paleidimą administratoriui per komandinį langą. Toliau platformos valdymo agentas sukuria visus sistemos darbui reikalingus agentus, kurie atskirai praneša administratoriui apie savo sukūrimą. Tarp visų sukurtų sistemos programinių agentų yra sukuriamas ir klaidų aptikimo agentas, kuris, pranešęs apie savo sukūrimą administratoriui, kas tam tikrą laiko tarpą (pagal nutylėjimą kas 5 sekundes) siunčia užklausas kitiems sistemos agentams. Užklausą gavę agentai privalo atsakyti klaidų aptikimo agentui, taip parodydami kad jie ištiesų funkcionuoja. Praėjus laukimo terminui klaidų aptikimo agentas patikrina kurie sistemos agentai atsakė į užklausas ir, aptikus neatsakiusių agentų, - jų sąrašą pateikia platformos valdymo agentui. Gavęs

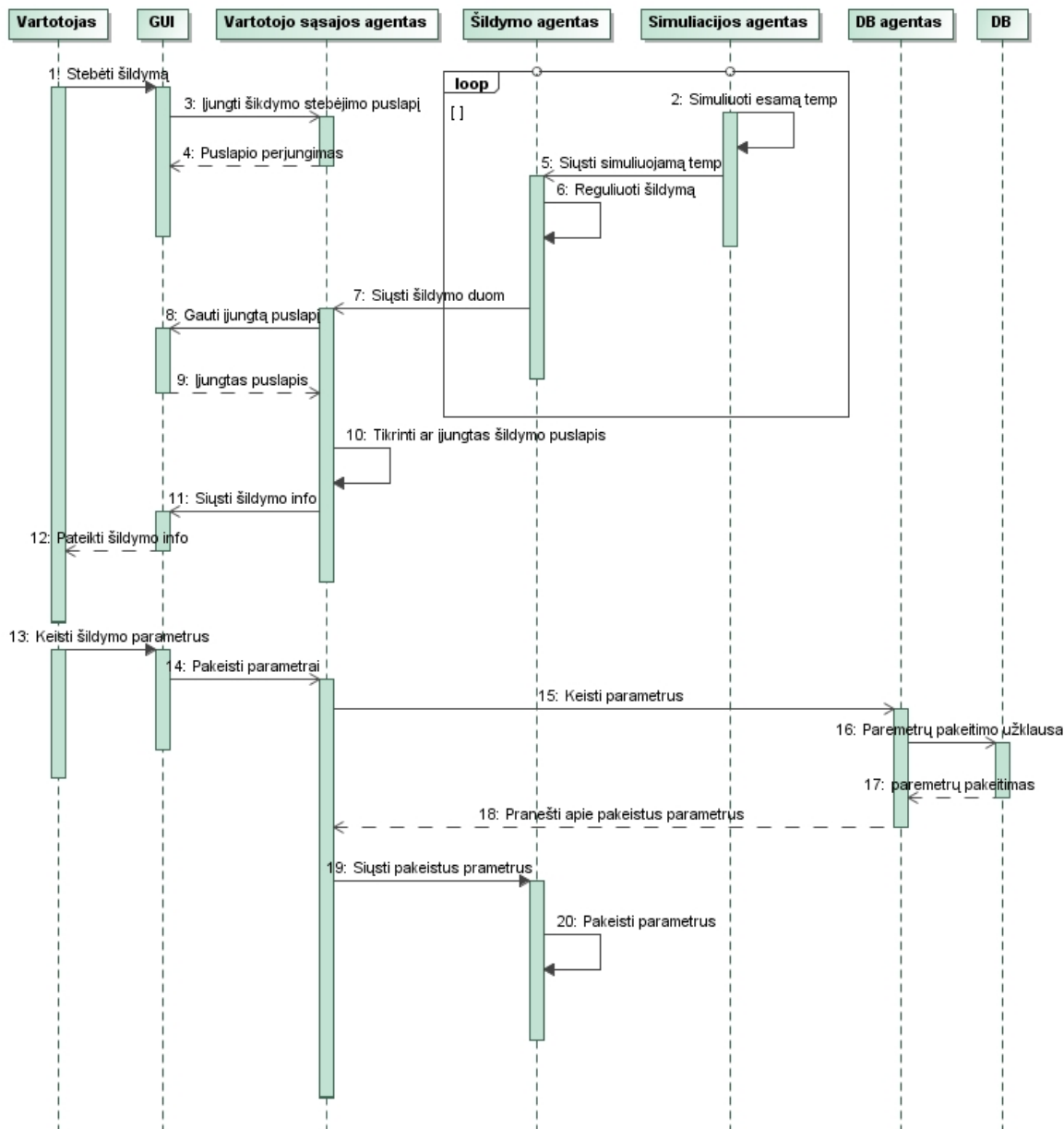
neatsakiusių į užklausą agentų sąrašą platformos valdymo agentas juos (agentus iš sąrašo) panaikina ir jų vietoje sukuria naujus agentus, taip užtikrinant sistemos veikimą.



13 pav. Orų prognozių gavimo sekos diagrama

Orų prognozių gavimo sekos diagrama (13 pav.) atvaizduoja atliekamų veiksmų eiliškumą vykdant orų prognozės teikimo operacijas. Vartotojui pasirinkus puslapį, kuriame rodoma orų prognozė, grafinė vartotojo sąsaja perduoda vartotojo sąsajos agentui, kad reikia perjungti sąsajos puslapį (jei puslapis nėra įjungtas). Įjungus puslapį su orų prognozėm, grafinė vartotojo sąsaja siunčia komandą vartotojo sąsajos agentui, kad reikia gauti orų prognozę. Vartotojo sąsajos agentas, gavęs komandą, siunčia žinutę su užklausa apie orų prognozę orų prognozių agentui. Toliau orų prognozės agentas siunčia užklausą duomenų bazės agentui apie vartotojo anksčiau pasirinktą ir išsaugotą orų prognozių vietą. Reguodamas į užklausą duomenų bazės agentas iš duomenų bazės suteikia informaciją apie orų prognozių vietą orų prognozių agentui. Turėdamas vietą orų prognozių agentas internete gauna orų prognozes artimiausiom dienom ir šią informaciją išsiunčia atgal vartotojo sąsajos agentui.

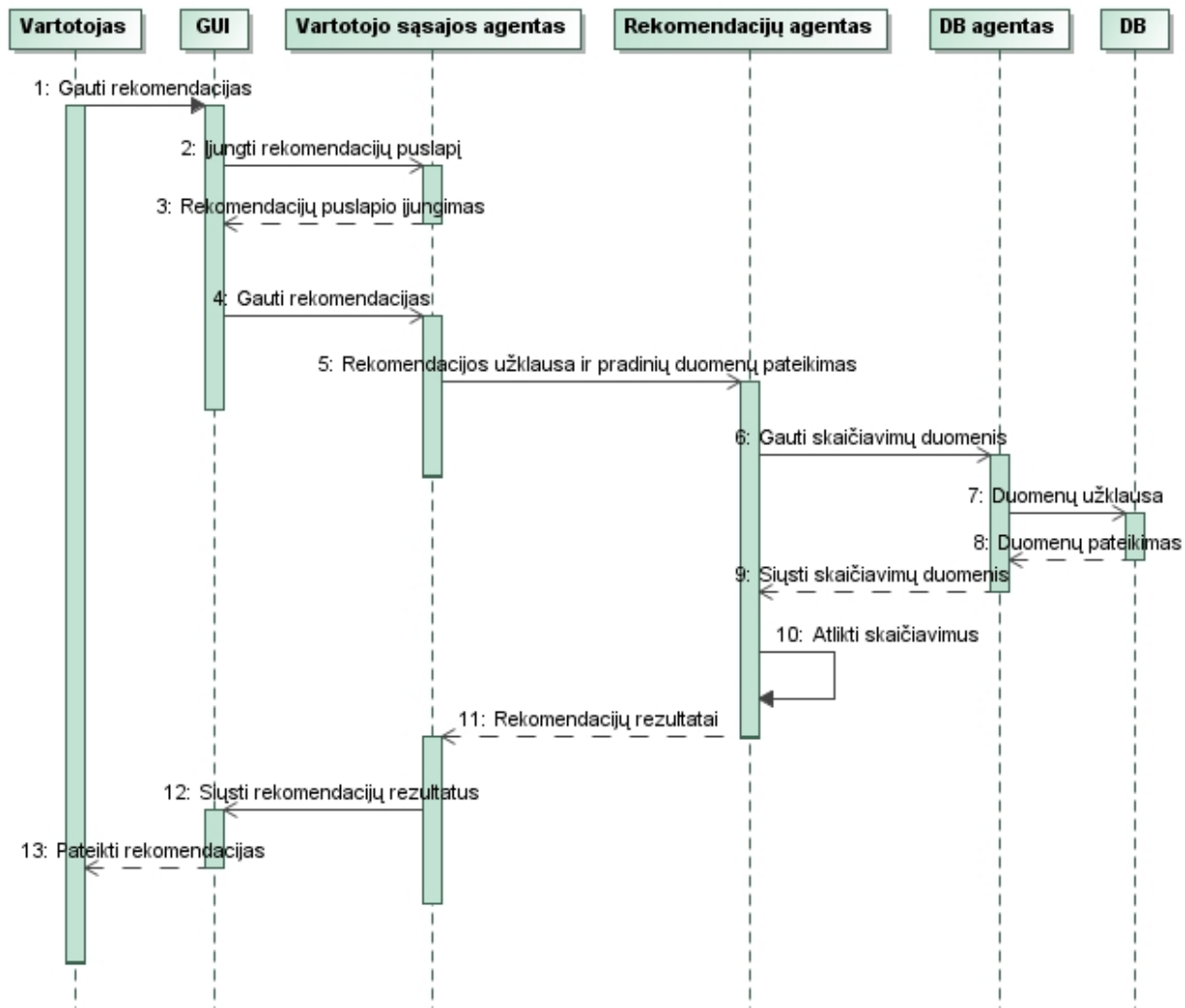
Gavęs orų prognozių duomenis, vartotojo sąsajos agentas juos (duomenis) išsiunčia grafinei vartotojo sąsajai, kuri vizualiai atvaizduoja juos ekrane vartotojui.



14 pav. Šildymo stebėjimo ir valdymo sekos diagrama

Šildymo stebėjimo ir valdymo sekos diagrama (14 pav.) atvaizduoja atliekamų veiksmų eiliškumą vykdant šildymo stebėjimo ir valdymo operacijas. Vartotojui pasirinkus šilumos stebėjimo puslapį, grafinė vartotojo sąsaja siunčia puslapio perjungimo komandą vartotojo sąsajos agentui, kuris perjungia puslapį. Simuliacijos agentas cikliškai simuliuoja kambario temperatūrą ir siunčia ją šildymo agentui, kuris, sureguliuavęs šildymą, šildymo duomenis išsiunčia vartotojo sąsajos agentui. Vartotojo sąsajos agentas, pasinaudodamas grafine sąsaja, patikrina ar šiuo metu yra įjungtas šildymo stebėjimo puslapis. Jeigu yra įjungtas šildymo stebėjimo puslapis – grafinė vartotojo sąsaja realiu laiku atvaizduoja šildymo duomenis

vartotojui. Vartotojui norint pakeisti šildymo parametrus (norimą kambario temp.) grafinė vartotojo sąsaja išsiunčia šiuos parametrus vartotojo sąsajos agentui, kuris juos persiunčia duomenų bazės ir šildymo agentui. Duomenų bazės agentas išsaugo pakeitimus sistemos duomenų bazėje, šildymo agentas – savo atmintyje.



15 pav. Rekomendacijų gavimo sekos diagrama

Rekomendacijų gavimo sekos diagrama (15 pav.) atvaizduoja atliekamų veiksmų eiliškumą vykdant rekomendacijų pateikimo operacijas. Vartotojui pasirinkus rekomendacijų puslapį, grafinė vartotojo sąsaja siunčia rekomendacijos puslapio įjungimo komandą vartotojo sąsajos agentui, kuris, gavęs komandą, perjungia puslapį. Įjungus rekomendacijų puslapį grafinė vartotojo sąsaja siunčia rekomendacijų gavimo komandą sąsajos agentui. Toliau vartotojo sąsajos agentas siunčia žinutę rekomendacijos agentui apie rekomendacijų gavimą. Siekiant atlikti skaičiavimus, rekomendacijos agentas siunčia žinutę duomenų bazės agentui su užklausa apie skaičiavimams reikalingus duomenis. Rekomendacijos agentas, gavęs skaičiavimų duomenis iš duomenų bazės agento ir atlikęs skaičiavimus, siunčia

rekomendacijų rezultatus vartotojo sąsajos agentui. Gautus rezultatus sąsajos agentas išsiunčia grafinei vartoto sąsajai, kad ši galėtų juos vizualiai pateikti vartotojui ekrane.

Išvados. Sudarius duomenų mainų tarp sistemos agentų modelį buvo nustatyta, kad sistema yra daugiaagentinė sistema, nes vizualiai matoma kaip protingos būsto informacijos valdymo sistemos programiniai agentai, bendraudami tarpusavyje, atlieka užduotis. Taip pat verta pabrėžti tai, kad agentai tarpusavyje bendrauja žinutėmis, kas palengvina jų atnaujinimą ir klaidų taisymą, nes agentai negali daryti įtakos vienas kito programiniam kodui. Taip yra dėl to, kad kiekvienas agentas į skirtingų pranešėjų skirtingas žinutes užprogramuotas elgtis individualiai. Toks žinučių apdorojimas taip pat sustiprina sistemos saugumą, nes ne kiekviena žinutė, atsiųsta programiniam agentui, negali iššaukti jo veiksmų.

3.4 Rekomendacijų agento darbo modelis

Rekomendacijų agento darbo modelis susideda iš šilumos kainų prognozių apskaičiavimo metodo ir skaičiavimams reikalingų duomenų gavimo būdų.

Šilumos kainų skaičiavimo metodas apskaičiuoja prognozuojamas pasirinkto mėnesio šildymo kainas atsižvelgiant į vartotojo parinktą norimą patalpų temperatūrą. Programiniam agentui gavus šildymo kainas (kWh kainas), jos yra padauginamos iš preliminarios prognozės, kuri nuspėja kiek kWh bus sunaudota šildymo sezono metu, simuliuojant šildymą. Simuliacijos metu gaunamos kWh yra apskaičiuojamos pagal formulę (16 pav.):

$$Q = \frac{Q_{šild.}}{\eta_{gam.}}$$

16 pav. Būsto suvartojamos šildymo energijos kiekio apskaičiavimo formulė [34]

Q – reikalingas energijos kiekis patalpų šildymui, kWh

$\eta_{gam.}$ – šilumos energijos gamybos efektyvumas. Mūsų eksperimente yra naudojamas centrinis šildymas, kurio gamybos efektyvumas yra apytiksliai 0.94 [34].

$Q_{šild.}$ energijos suvartojimas šildant patalpas ir atsižvelgiant į šildymo sistemos efektyvumą, bei šilumos netektis dėl nešildomų zonų [34] yra apskaičiuojamas pagal formulę (17 pav.):

$$Q_{šild.} = \frac{Q_{pat.}}{\eta_{pat.}}$$

17 pav. Energijos suvartojimas šildant patalpas atsižvelgiant į šildymo sistemos efektyvumą ir šilumos netektis dėl nešildomų zonų [34]

$Q_{pat.}$ – suvartojama energija patalpų šildymui, kWh

$\eta_{pat.}$ – patalpų šildymo sistemos efektyvumas

Šildymo sistemos efektyvumas $\eta_{pat.}$ įvertina šilumos netektis tiekiant, reguliojant, paskirstant šilumą [34]. Darbo metu bus naudojami vandeniui kaitinami radiatoriai patalpų šildymui, kurių efektyvumas apytiksliai yra 0,77 [34].

$Q_{pat.}$ energijos suvartojimas šildant patalpas apskaičiuojamas pagal formulę (18 pav):

$$Q_{pat.} = Q_{iš.sienų} + Q_{lubų} + Q_{grindų} + Q_{langų} + Q_{durų}$$

18 pav. Pastato komponentų atiduodamos šilumos apskaičiavimo formulė [34]

Pastato komponentų šilumos netektys atskirai kiekvienam komponentui yra apskaičiuojamos pagal formulę (29 pav.):

$$Q_i = \sum U_i A_i (T_{vidaus} - T_{lauko}) \Delta t / 1000$$

19 pav. Individualaus pastato komponento šilumos netekties apskaičiavimo formulė

Q_i – pastato komponento šilumos netektis, kWh

U_i – pastato komponento šiluminis laidumas, W/(m²K)

A_i – pastato komponento paviršiaus plotas, kuris ribojasi su išore, m²

T_{vidaus} – vidaus temperatūra, °C

T_{lauko} – lauko temperatūra, °C

Δt – laiko periodas, h

1000 – vienetų keitimas į kWh

Šildymo kainų prognozių metodas gali būti skaičiuojamas dviem būdais: atsižvelgiant į šildymo metu patiriamus nuostolius (16 pav.) ir neatsižvelgiant (18 pav.). Šildymo kainų prognozių apskaičiavimo metodui reikalingi duomenys: norima kambario temperatūra – gaunama iš vartotojo ar šildymo agento (gražinant nustatytą šildymo temperatūrą); kWh kainos – gaunamos iš sistemos duomenų bazės; pastato komponento šiluminis laidumas ir su išore besiribojantis paviršiaus plotas – gaunami vartotojui įvedant duomenis į sistemą; lauko temperatūra – vidutinė daugiametė mėnesio temperatūra, gaunama iš sistemos duomenų bazės.

3.5 Išvados

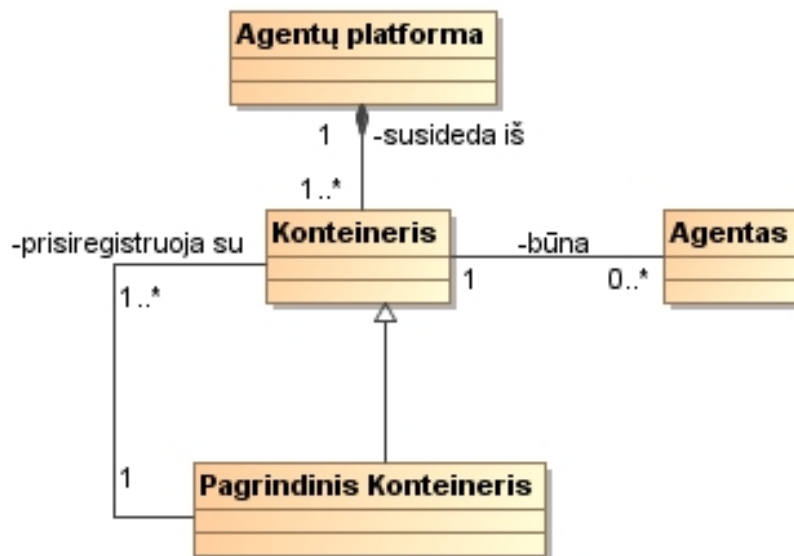
Protingos namų informacijos valdymo sistemos modelio dalyje pateikti sistemos funkcinis modelis (9 pav.), sistemos struktūra (10 pav.) ir duomenų mainų tarp sistemos agentų modelis, pagal kuriuos nustatyta jog protingos būsto informacijos valdymo sistemos programiniai agentai sudaro daugiaagentę sistemą. Tai galima teigti, nes sistemą sudarantys agentai, bendraudami tarpusavyje, atlieka sistemos užduotis. Tokią sistemą yra paprasčiau atnaujinti ir išplėsti. Taip pat nustatyta kokie agentai turėtų sudaryti sistemą ir kokias funkcijas jie atlieka: vartotojo sąsajos agentas – bendravimo tarp vartotojo ir kitų sistemos agentų užtikrinimas, šildymo agentas – šildymo stebėjimas ir valdymas, klaidų aptikimo agentas – sistemos klaidų šalinimas, orų prognozių agentas – orų prognozių teikimas, DB agentas – duomenų bazės valdymas, rekomendacijų agentas – rekomendacijų teikimas, platformos valdymo agentas – sistemos agentų paleidimas. Be to iš sistemos funkcinio modelio (12 pav.) nustatyti sistemos vartotojų tipai ir pagrindinės sistemos teikiamos paslaugos vartotojui: administratorius – sistemos diegimas, vartotojas – orų prognozių gavimas, šildymo stebėjimas ir valdymas, rekomendacijų gavimas. Pateiktas rekomendacijų agento darbo modelis, kuris aprašo šilumos kainų prognozių apskaičiavimo metodą ir šiam metodui reikalingų duomenų gavimo būdus.

4 EKSPERIMENTINIS TYRIMAS

Sistemos eksperimentinio tyrimo tikslas yra patikrinti sukurtą protingo būsto informacijos valdymo sistemą. Tikrinimo metu yra testuojamos pagrindinės sistemos atliekamos funkcijos vartotojui: orų prognozių gavimas, šildymo stebėjimas ir valdymas, rekomendacijų gavimas. Kiekvienos funkcijos rezultatai yra vizualiai atvaizduojami ekrane, pasinaudojant grafine vartotojo sąsaja. Taip pat šioje dalyje yra tiriamas rekomendacijų agento darbas.

4.1 Programinių agentų realizavimas sistemoje

20 pav. atvaizduoja pagrindinius architektūrinius JADE platformos elementus ir ryšius tarp jų. JADE platforma susideda iš agentų konteinerių, kurie gali būti pasklidę po kompiuterių tinklą. Agentai būna konteineriuose, kurie užtikrina agentų laikymui ir vykdymui reikalingas paslaugas. Taip pat egzistuoja specifinis konteineris – pagrindinis konteineris, jis atvaizduoja platformos pradinį tašką: tai pats pirmas paleidžiamas konteineris ir visi kiti vėliau paleidžiami konteineriai privalo prisijungti prie pagrindinio konteinerio prie jo prisiregistruojant.



20 pav. Ryšiai tarp pagrindinių JADE architektūros elementų [14]

Protingą būsto informacijos valdymo sistemą sudaro šie programiniai agentai:

- Šildymo agentas – reaktyvus agentas, stebėdamas aplinką (simuliuojamą temperatūrą) šildymo agentas priima atitinkamus sprendimus realiu laiku (išjungti ar įjungti patalpos šildymą, atsižvelgiant į vartotojo nustatymus).

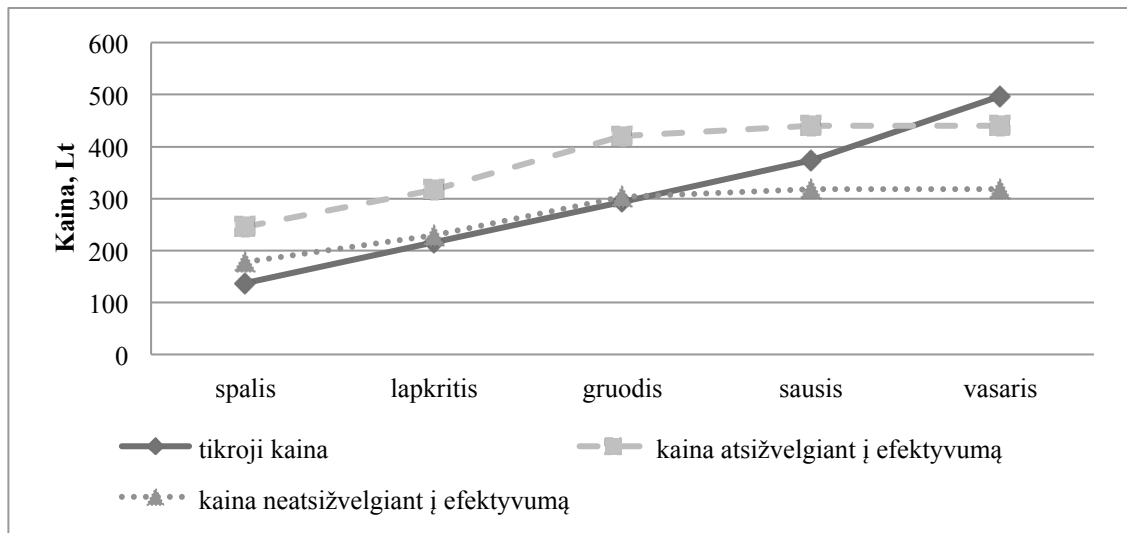
- Orų prognozių agentas – komandinis agentas, pasinaudodamas interneto ir kitų agentų pagalba sistemai teikia orų prognozių paslaugas.
- Vartotojo sąsajos agentas – sąsajos agentas, stebėdamas vartotojo veiksmus, atliekamus su grafine vartotojo sąsaja, juos interpretuoja ir praneša kitiems sistemos agentams apie vartotojo reikalaujamas atlikti užduotis. Užtikrina bendravimą tarp vartotojo ir sistemos agentų.
- Duomenų bazės agentas – komandinis agentas, suteikia prieigą prie sistemos duomenų bazės kitiems programiniams agentams. Atlieka visus veiksmus (rašymo, skaitymo, trinimo) su sistemos duomenų baze to pareikalavus iš kitų agentų.
- Laikrodžio agentas – komandinis agentas, turintis labai paprastą užduotį – rodyti dabartinį laiką ir datą grafinėje vartotojo sąsajoje.
- Rekomendacijų agentas – komandinis agentas, pasinaudodamas iš kitų agentų gaunamais duomenimis (vartotojo sąsajos ir duomenų bazės agentų) atlieka šildymo kaštų prognozių skaičiavimus ir rezultatus gažina vartotojui pateikti.
- Simuliacijos agentas – komandinis agentas, atliekantis simuliacijos rolę – kas nustatytą žingsnį sumuliuoja kambario temperatūrą ir sunčia ją šildymo agentui.
- Klaidų aptikimo agentas – reaktyvus agentas, stebi sistemos agentų veiklą siųsdamas jiems žinutes, negavus atsakymo – siunčia neatsakusių agentų pakeitimo komandą platformos valdymo agentui.
- Platformos valdymo agentas – komandinis agentas, sukuria visus sistemos agentus, taip pat gali bet kurį sistemos agentą išjungti ar sukurti naujus sistemos agentus.

Kaip matoma – dauguma protingos būsto informacijos valdymo sistemos programinių agentų priklauso komandinių agentų klasei, tai patvirtina teiginį, kad ši sistema yra daugiaagentinė sistema. Daugiaagentė sistema yra tokia sistema, kurios užduotis atlieka daug skirtingų, tarpusavyje bendraujančių, agentų.

4.2 Rekomendacijų agento darbo tyrimas

Rekomendacijų agento darbo tyrimas susideda iš sistemos modelio dalyje aprašyto rekomendacijos darbo modelio tyrimo ir rekomendacijų agento realizavimo sistemoje tyrimo. Rekomendacijos agentas naudoja šilumos kainų prognozavimo metodą, kuris gali būti

skaičiuojamas dviem variantais: prognozuoti būsto sunaudojamą šilumos kiekį atsižvelgiant į šilumos gamybos ir patalpų šildymo sistemos efektyvumą (16 pav.), ir prognozuoti šilumos suvartojimą neatsižvelgiant į efektyvumą (18 pav.). Siekiant palyginti abu šio metodo skaičiavimo būdus buvo sudarytas šildymo tikrosios ir prognozuojamų kainų grafikas (21 pav.).



21 pav. Šildymo tikrosios ir prognozuojamų kainų grafikas

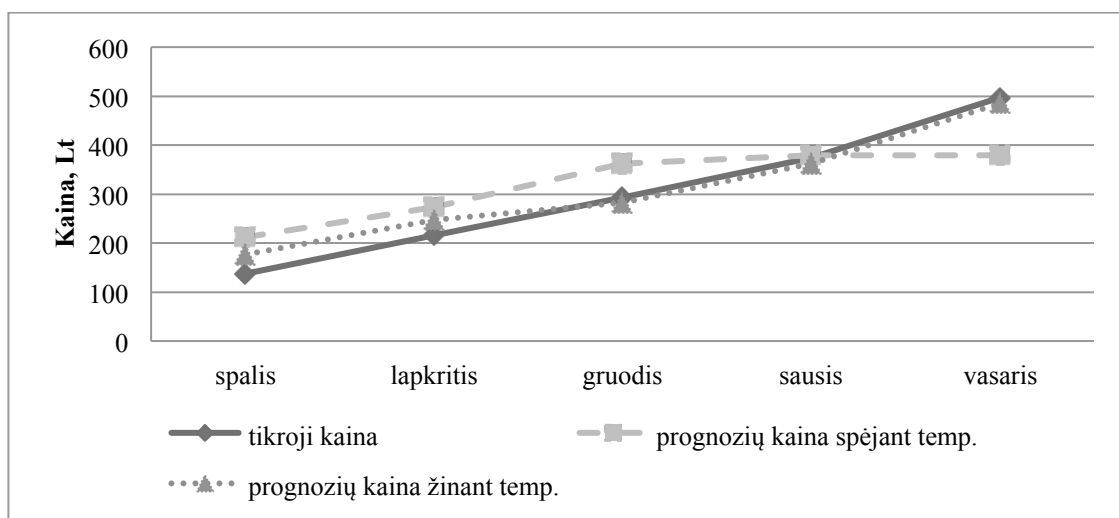
Šildymo tikrosios ir prognozuojamų kainų grafiko skaičiavimui panaudojome vidutines daugiametes Lietuvos oro temperatūras [35], Kauno raj. šilumos kainas [36] (eksperimentui naudojamas namas yra Kauno raj.), gautas sąskaitas už būsto šildymą (tikroji kaina). Taip pat į skaičiavimo metodą įtraukti ir specifiniai būsto parametrai: 27 m² su išore besiribojančių blokinių sienų, kurių šiluminis laidumas U yra 1,5 (W/m²K) [37]; 7,5 m² neizoliuotų plastikinių langų, kurių U = 3,3 (W/m²K). Verta pastebėti kad būstą supa kitos šildomos patalpos, kurių šildymas neįeina į gaunamas sąskaitas, dėl šios priežasties plotas, kuris ribojasi su išore yra mažas.

$$Q_{vid.} = \frac{Q + Q_{pat.}}{2}$$

22 pav. Pasirinkto šildymo kainų prognozių metodo skaičiavimo formulė

Iš grafiko (21 pav.) matyti kad abu skaičiavimo būdai turi paklaidas: atsižvelgiant į efektyvumą vidutiniškai 26,8 % nuo prognozuojamos sumos; neatsižvelgiant – 21,1 % nuo prognozuojamos sumos. Siekiant sudaryti kuo tikslesnes prognozes nuspręsta šildymo kainų prognozių metode taikyti abu skaičiavimo būdus (tiek atsižvelgiant į efektyvumą, tiek neatsižvelgiant) ir pateikiamose prognozėse naudoti abiejų metodų rezultatų vidurkį (22 pav.), kurio paklaida yra 22,7 % nuo prognozuojamos sumos. Metode naudojant abiejų variantų vidurkį, paklaida (22,7 %) išeina šiek tiek didesnė už prognozės skaičiavimo būdą

neatsižvelgiant į efektyvumą (21,1 %), tačiau tęsiant eksperimentą išaiškėjo, jog kuo tiksliau nuspėjant ateinančių mėnesių temperatūras – pasirinktas metodo skaičiavimas (naudojant abiejų variantų vidurkį) tampa ženkliai tikslesnis (23 pav.).



23 pav. Šilumos kainų prognozių metodo tikslumo priklausomybė nuo temp. prognozės tikslumo

Siekiant parodyti šilumos kainų prognozių metodo tikslumo priklausomybę nuo oro temperatūros prognozės, buvo sudarytas kitas kainų grafikas (23 pav.). Šiame grafike panaudojamos jau žinomos eksperimente naudojamų 2011-2012 metų mėnesių temperatūros [35], taip siekiant įvertinti metodo tisklumą kuo palankesnėmis sąlygomis (numatant kuo tikslesnes orų temp.). Iš grafiko (23 pav.) aiškiai matyti, kad prognozių kaina, žinant temperatūrą, yra žymiai artimesnė tikrajai kainai už prognozuojamą kainą spėjant temperatūrą (panaudojant vidutinę daugiametę temperatūrą). Prognozių vidutinė paklaida žinant temperatūrą yra 8,9 % nuo prognozuojamos sumos, numatant temperatūrą – 22,7 % nuo prognozuojamos sumos.

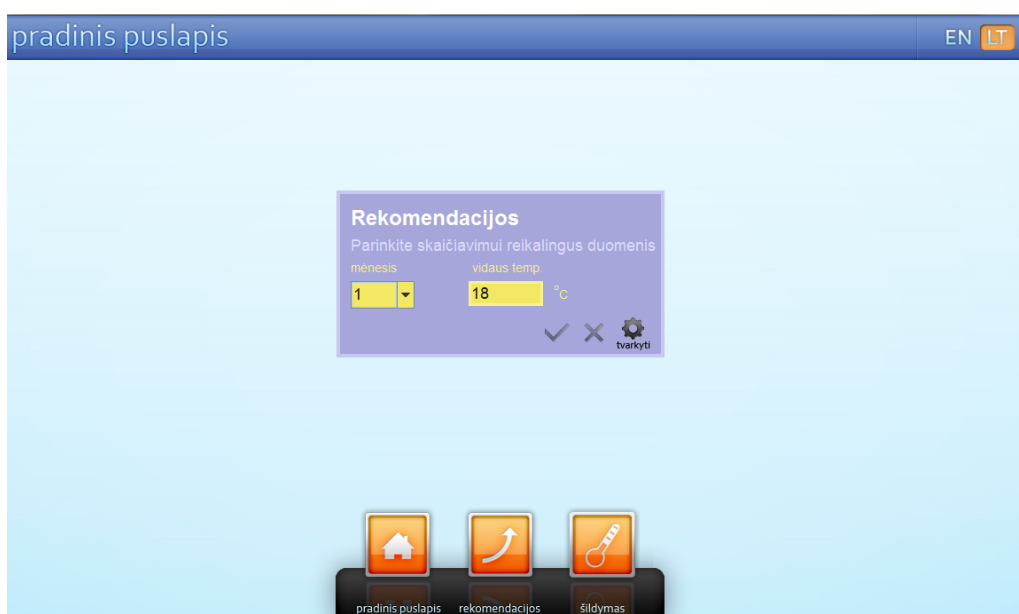
5 lentelė. Šilumos kainų prognozių metodų palyginimas

	Prognozė atsižvelgiant į šilumostiekimo ir sistemos efektyvumus (17 pav.)	Prognozė neatsižvelgiant į šilumostiekimo ir sistemos efektyvumus (19 pav.)	Prognozė imant ankstesnių kitų dviejų būdų rezultatų vidurkius (22 pav.)
Metodo tikslumas, skaičiavimui naudojant vid. daugiametę temp.			
Vidutinė paklaida	26,8 %	21,1 %	22,7 %
Metodo tikslumas, skaičiavimui naudojant tikslias temp.			
Vidutinė paklaida	18,2 %	15,9 %	8,9 %

Siekiant pasirinkti tinkamiausią rekomendacijų agento naudojamą metodą buvo sudaryta visų tiriamų metodų palyginimo lentelė (5 lentelė), kuri parodo kiekvieno varianto vidutinę

paklaidą nuo prognozuojamos sumos. Šilumos kainų prognozių metodui pasirinktas skaičiavimo būdas susideda iš dviejų metodų variantų rezultatų vidurkio (22 pav.), todėl kad tikslėjant orų temperatūros prognozėm šio metodo rezultatų paklaida tampa mažiausia.

Tiriant rekomendacijų agento veikimą sistemoje, atliekamas šio agento testavimas. Vartotojui pasirinkus „rekomendacijos“ meniu punktą yra įjungiamas rekomendacijų puslapis (24 pav.). Šis puslapis yra perjungiamas grafinėi vartotojo sąsajai pranešus vartotojo sąsajos agentui apie atliktą mygtuko „rekomendacijos“ paspaudimą. Norint gauti mėnesio šildymo kainos prognozę vartotojas turi pasirinkti šildymo mėnesį ir norimą kambario temperatūrą. Įjungiant rekomendacijos puslapį „vidaus temp.“ laukas yra užpildomas tokia temperatūros reikšme, kuri yra pasirinkta šildymo nustatymuose.



24 pav. Rekomendacijų puslapis

Norint pakeisti kitus rekomendacijos metodo parametrus, reikia įjungti rekomendacijos metodo parametrų keitimo puslapį (25 pav.), paspaudžiant mygtuką „tvarkyti“ (24 pav.). Įjungus puslapį visi tekstiniai laukai yra užpildomi jiems atitinkamais anksčiau išsaugotais duomenimis iš duomenų bazės. Šiame puslapyje varotojoas gali pakeisti būsto duomenis, reikalingus kainų prognozių skaičiavimams – objektų, besiribojančių su išore, plotus ir šiluminius laidumus. Išsaugant pakeitimus grafinė vartotojo sąsaja perduoda duomenis vartotojo sąsajos agentui, kuris juos žinute išsiunčia duomenų bazės agentui su liepimu įrašyti (duomenis) į duomenų bazę.

Elementas	plotas (m ²)	šiluminis laidumas (W/m ² K)
Išorinės sienos	27	1.5
Langai	7.5	3.3
Grindys	0	0.2
Lubos/stogas	0	1

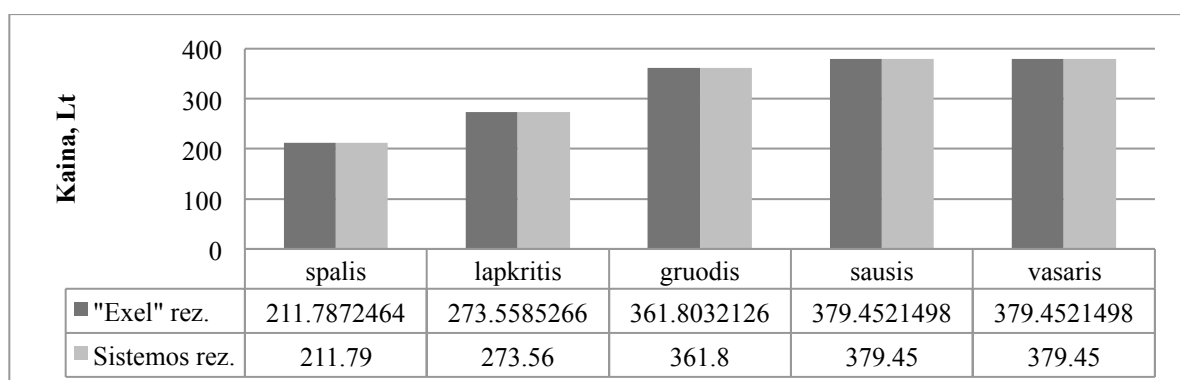
25 pav. Rekomendacijos metodo parametru keitimo puslapis

Rekomendacijų rezultatai yra pateikiami vartotojui ekrane rekomendacijų rezultatų puslapyje (26 pav.), pasinaudojant grafine vartotojo sąsaja.

Preliminari šildymo kaina
~379,45 Lt

26 pav. Rekomendacijų rezultatų puslapis

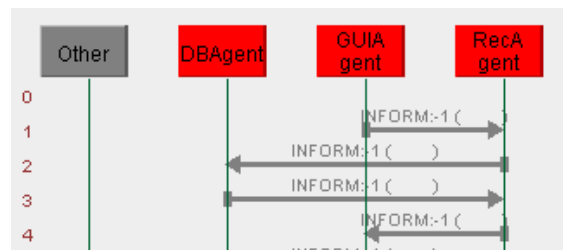
Siekiant patikrinti skaičiavimų tikslumą – sistemos gražinami rezultatai buvo palyginti su „Microsoft Exel“ programiniu paketu atliktais analogiškais skaičiavimais (27 pav.). Kaip matoma iš diagramos – rezultatai yra beveik identiški, mažos (šimtosios dalies tikslumu) paklaidos atsiranda dėl sistemos pateikiamo rezultato apvalinimo iki dviejų skaitmenų po kablelio.



27 pav. „Microsoft Exel“ ir protingos būsto informacijos valdymo rekomendacijų skaičiavimų rezultatų diagrama

Teikiant rekomendacijas protingos būsto informacijos valdymo sistemos programiniai agentai tarpusavyje siunčia ACL žinutes, taip atlikdami savo užduotis (28 pav.). Vartotojui

norint gauti rekomendacijas – grafinė vartotojo sąsaja perduoda rekomendacijų gavimo komandą vartotojo sąsajos agentui („GUIAgent“) kartu su vartotojo įvestais duomenimis. Komandą su duomenimis gavęs vartotojo sąsajos agentas juos (duomenis) persiunčia rekomendacijų agentui („RecAgent“) ACL žinute kartu pareikalaudamas šilumos kainų prognozių rezultatų. Kitus skaičiavimui reikalingus duomenis (šilumos kainą, mėnesio vidutines daugemetes temperatūras) rekomendacijos agentas gauna iš duomenų bazės agento („DBAgent“). Turėdamas visus reikalingus duomenis rekomendacijų agentas atlieka skaičiavimus ir rezultatus grąžina vartotojo sąsajos agentui, kuris, pasinaudodamas grafinė vartotojo sąsaja, juos atvaizduoja vartotojui.



28 pav. JADE „sniffer“ programos lango fragmentas, vaizduojantis siunčiamas agentų žinutes rekomendacijų teikimo metu

Atlikus rekomendacijų agento darbo tyrimą, buvo pasirinktas tinkamiausias šildymo kainų prognozių skaičiavimo metodas, susidedantis iš kainų skaičiavimo atsižvelgiant ir neatsižvelgiant į šildymo metu patiriamus nuostolius. Patikrintas rekomendacijų agento veikimas sistemoje. Testuojant rekomendacijų agento darbą sistemoje klaidų nebuvo aptikta.

4.3 Sistemos realizacija

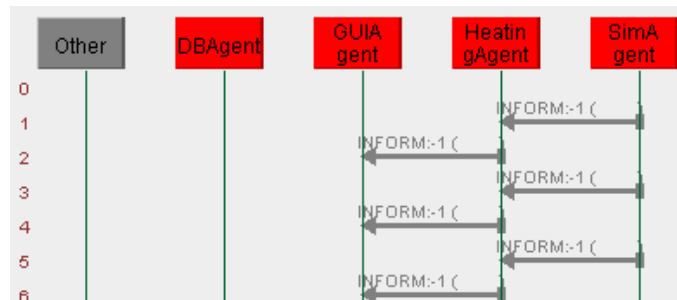
Sistemos realizacijos dalyje tiriamas sistemos agentų funkcionalumas, atliekant pagrindines sistemos funkcijas: orų prognozių gavimas, šildymo valdymas ir klaidų aptikimas. Vartotojui paleidus sistemą ekrane atvaizduojamas pradinis puslapis (29 pav.). Pradinio lango viršutinėje dešinėje dalyje sistema atvaizduoja pasirinktos vietos orų prognozes, paspaudus mygtuką „tvarkyti“ – vartotojas gali pakeisti ir užsaugoti kitą orų prognozės vietą. Orų prognozės atnaujinamos kaskart įjungiant pradinį puslapį. Kitos sistemos paslaugos pasiekiamos pasinaudojant lango apačioje esančiu meniu.



29 pav. Sistemos pradinis puslapis

Sistema vartotojui atvaizduoja orų prognozes, pasinaudodama programiniais agentais. Šios paslaugos teikimo metu atliekamos funkcijos atvaizduojamos ACL (angl. ACL – agent communication language) žinučių, kuriomis agentai tarpusavyje bendrauja, seka (30 pav.). Sekoms atvaizduoti naudojama JADE sniffer programa, kuri leidžia stebėti pasirinktus sistemos agentus ir jų sunčiamas žinutes tarpusavyje realiu laiku. Siekiant išvengti galimų funkcijų atlikimo klaidų, kiekvienas sistemos programinis agentas turi užprogramuotą žinučių gavimo ir išsiuntimo apdorojimo logiką, kuri tikrina siuntėjo arba gavėjo ir prašomų atlikti funkcijų derinius taip nurodant agentui kokias užduotis jam reikia atlikti. Atlikęs užduotis programinis agentas išsiunčia rezultatus jų reikalavusiam agentui. Orų prognozių teikimo metu, kaip matoma iš žinučių siuntimo sekos (30 pav.), vartotojo sąsajos agentas („GUIAgent“) siunčia žinutę orų prognozių agentui („WeatherAgent“) – 2 numeriu pažymėta eil., kuris, gavęs duomenų bazėje išsaugotą prognozių vietą ir duomenų bazės agento („DBAgent“) – 3 ir 15 nr. pažymėtos eil., internete suranda artimiausių dienų orų prognozes ir rezultatus išsiunčia atgal jų reikalavusiam vartotojo sąsajos agentui – 16 nr. pažymėta eil.

gavęs informaciją apie pasikeitusią temperatūrą ją išanalizuoja – sulygina ją su nustatyta kambario temperatūra ir atitinkamai priima sprendimą ar reikia išjungti ar įjungti šildymą. Atlikęs savo funkcijas šildymo agentas informaciją apie esamą temp. ir būseną išsiunčia vartotojo sąsajos agentui („GUIAgent“). Vartotojo sąsajos agentas, gavęs šildymo informaciją, patikrina ar grafinė vartotojo sąsaja atvaizduoja šildymo puslapį (siekiant išvengti sinchronizacijos problemų) ir, jeigu ekrane yra atvaizduojamas šildymo puslapis – šildymo informacija ekrane yra atnaujinama.



32 pav. JADE „sniffer“ programos lango fragmentas, vaizduojantis siunčiamas agentų žinutes šildymo stebėjimo metu

Testuojant klaidų aptikimo agentą, pasinaudojant JADE agentų valdymo panele, buvo tikslingai išjungiami protingos būsto informacijos valdymo sistemos programiniai agentai. Taip nustatant ar sėkmingai su nefunkcionuojančių agentų problemomis susitvarko klaidų aptikimo agentas. Aptikus nefunkcionuojančius agentus, klaidų aptikimo agentas jų sąrašą išspausdina programos komandiniame lange (33 pav.) ir siunčia ACL žinutę platformos valdymo agentui su pranešimu apie neveikiančius agentus. Platformos valdymo agentas, gavęs nefunkcionuojančių agentų sąrašą, juos pašalina iš sistemos ir į jų vietas sukuria naujus programinius agentus, taip užtikrinant sistemos veikimą.

```

---dead sarasas---
TimerAgent
HeatingAgent
---
PMAFeja CREATED AND STARTED NEW AGENT:TimerAgent ON CONTAINER Main-Container
PMAFeja CREATED AND STARTED NEW AGENT:HeatingAgent ON CONTAINER Main-Container
Hello World! My name is HeatingAgent
Hello World! My name is TimerAgent
2012-05-15 09.46.13 jade.core.messaging.MessagingService removeLocalAliases
INFO: Removing all local alias entries for agent TimerAgent
2012-05-15 09.46.13 jade.core.messaging.MessagingService removeGlobalAliases
INFO: Removing all global alias entries for agent TimerAgent
---dead sarasas---
TimerAgent
---
PMAFeja CREATED AND STARTED NEW AGENT:TimerAgent ON CONTAINER Main-Container
Hello World! My name is TimerAgent
    
```

33 pav. Programos komandinis langas su klaidų aptikimo agento spausdinamomis žinutėmis

Testuojant protingą būsto informacijos valdymo sistemą nebuvo aptikta klaidų. Taip pat nustatyta, kad sistema atlieka visas projektinėje dalyje apibrėžtas funkcijas. Kiekvienas sistemos programinis agentas tinkamai atlieka savo užduotis. Agentų rolės nesidubliuoja. Protingos būsto informacijos valdymo sistemos programiniam agentam atliekant jiems apibrėžtas užduotis – agentai savo tikslam pasiekti pasinaudoja kitų agentų pagalba tarpusavyje bendraudami ACL žinutėmis. Pagal tai galima teigti kad sistema yra daugiaagentinė sistema. Besikeisdami ACL žinutėmis sistemos agentai tarpusavyje keičiasi duomenimis ir komandomis, kurias reikia atlikti.

4.4 Sistemos tobulinimo ir plėtros rekomendacijos

Ateityje siekiant tobulinti ir plėtoti sistemą, bei gerinti vartotojo patirtį galima būti atlikti šias sistemos tobulinimo ir plėtros rekomendacijas:

- Kadangi rekomendacijų agento šildymo kainų prognozės tikslumas yra labai glaudžiai susijęs su viso mėnesio temperatūrų tikslumu – galima tobulinti mėnesio vidutinės temperatūros gavimo algoritmą, siekiant gauti kuo tikslesnes temperatūras. Taip būtų vartotojui teikiamos tikslesnės prognozės.
- Praplėsti šildymo valdymo funkcionalumą – įgalinti vartotoją kurti individualius šildymo tvarkaraščius. Taip suteikiant didesnę lankstumą vartotojui pasirenkant sau tinkamus šildymo nustatymus. Kartu šiuos tvarkaraščius susieti su rekomendacijos agento atliekamais prognozių skaičiavimais, taip vartotojui pateikiant skirtingų šildymo būdų (palaikant viena temp. ar šildant pagal tvarkaraštį) prognozuojamas kainas.
- Pridėti daugiau valdymo ir stebėjimo agentų, tokių kaip apšvietimo, drėgmės, vaizdo stebėjimo agentų. Taip praplečiant sistemos teikiamų paslaugų spektrą.
- Padaryti būsto resursų grafikų sudarymo programinį agentą, kuris galėtų vartotojui vizualiai ekrane atvaizduoti skirtingų būsto resursų (pvz. patalpų temperatūros, drėgmės statistika) grafikus. Kadangi dabar sistema šiuos resursų parametrus atvaizduoja ekrane realiu laiku, bet nėra jų atvaizdavimo grafikuose, daliai vartotojų toks peržiūros būdas būtų patogesnis. Grafikai būtų vaizduojami x, y koordinacių plokštumose.
- Išplėsti grafinės vartotojo sąsajos galimybes – padaryti prieigą prie sistemos internetinių tinklalapių principu. Taip būtų praplečiamas sistemos pasiekiamumas ir didinamas patogumas valdant pačią sistemą – vartotojui neprivaloma būti prie

konkreto sistemos valdymo kompiuterio, pakanka prisijungti prie sistemos iš bet koks prietaiso, turinčio interneto naršyklę. Kartu su šiuo vartotojo sąsajos patobulinimu reikėtų įdiegti ir vartotojų autentifikavimą, taip subalansuojant sistemos saugumą, atsiradus naujiems sistemos valdymo būdams iš skirtingų prietaisų.

4.5 Išvados

Eksperimentinio tyrimo metu buvo testuojamas protingos būsto informacijos valdymo sistemos veikimas. Testavimo metu buvo bandomos visos pagrindinės vartotojo atliekamos funkcijos us sistema: orų prognozių gavimas, šildymo stebėjimas ir valdymas, rekomendacijų gavimas. Atliktas rekomendacijų agento darbo tyrimas, kurio metu parinktas tinkamiausias šilumos kainų skaičiavimo metodas ir ištirtas rekomendacijų agento darbas sistemoje.

Testavimo metu nustatyta, kad užduotis atlieka aštuoni sistemos programiniai agentai: duomenų bazės, klaidų aptikimo, vartotojo sąsajos, rekomendacijų, laikrodžio, orų prognozių, platformos valdymo, šildymo agentai. Kiekvienas agentas atlieka savo unikalias užduotis – sistemos programinių agentų rolės nesidubliuoja. Sistemos atliekamos užduotys yra vykdomos pasinaudojant agentų bendradarbiavimu tarpusavyje, dėl to galima teigti, kad sistema – yra daugiaagentinė sistema. Patys agentai tarpusavyje bendrauja ACL žinutėmis.

Taip pat eksperimentinio tyrimo metu buvo sudarytos ir pateiktos sistemos tobulinimo ir plėtros rekomendacijos: orų prognozių, naudojamų šildymo kainų skaičiavime, tobulinimas; šildymo valdymo funkcionalumo praplėtimas; papildomų programinių agentų diegimas; grafinės vartotojo sąsajos galimybių išplėtimas.

Eksperimentinio tyrimo metu protingoje būsto informacijos valdymo sistemoje nebuvo aptikta klaidų.

5 IŠVADOS

1. Atlikus protingų būsto informacijos valdymo sistemų analizę nustatyta, kad programinių agentų technologijos nėra taikomos įvairiuose kompiuterizuotų būstų sistemose, tačiau jų poreikis šioje srityje sparčiai auga.
2. Atlikus programinių agentų ir jų panaudojimo protingose būsto informacijos valdymo sistemose analizę, buvo nustatyta, kad šiose sistemose naudojamų agentų paskirtis - prietaisų valdymas ir komunikavimas tarp sistemos ir vartotojų. Taip pat išanalizavus programinių agentų klasifikaciją ir tipus buvo pasiūlyti sistemai tinkamiausi agentai – komandiniai, sąsajos ir reaktyvūs agentai.
3. Sudarytas daugiaagentės sistemos modelis, kuris apibrėžia sistemos funkcionalumą, agentų roles, kurios yra skirtingos ir nesidubliuoja. Sistemą sudaro aštuonių tipų agentai: duomenų bazės, klaidų aptikimo, vartotojo sąsajos, rekomendacijų, laikrodžio, orų prognozių, platformos valdymo, šildymo valdymo.
4. Pasiūlytas mėnesio šildymo kainų prognozes skaičiuojantis metodas, kuris naudoja sudarytą suvartojamos šilumos skaičiavimo formulę, kurios pagrindu realizuotas rekomendacijų agentas, atliekantis šilumos skaičiavimą atsižvelgiant į patiriamus šilumos gavimo nuostolius.
5. Protinga namų informacijos valdymo sistema sukurta su JADE, JAVA, „PostgreSQL“ technologijomis. Dėl plačių šių technologijų pritaikymo galimybių – sukurta sistema yra daugiaplatformė. Taip pat su šiomis technologijoms yra sukurta būsto kompiuterizavimo sistema, į kurią yra diegiama darbo sistema – taip sutrumpėja sistemos kūrimo, klaidų taisymo ir testavimo laikas. JADE užtikrina lengvą sistemos atnaujinimą ir pritaikymą ateityje kuriamiems agentams, nes šis įrankis palaiko FIPA standartą.
6. Eksperimentinio tyrimo metu buvo ištirta realizuota sistema ir sudarytas sistemos tobulinimo ir plėtros rekomendacijų sąrašas. Eksperimentinio tyrimo metu nustatyta, kad sistema atlieka visas jai apibrėžtas funkcijas. Sukurtoje sistemoje neaptikta klaidų.

6 TERMINŲ IR SANTRUPMŲ ŽODYNAS

žodis ar santrumpa	paaiškinimas
ACL	(angl. agent communication language) Agentų bendravimo kalba. ACL žinutės yra FIPA agentų komunikavimo kalbos standartas.
DB	Duomenų bazė.
DBVS	Duomenų bazių valdymo sistema.
FIPA	(angl. Foundation for Intelligent Physical Agents) IEEE pripažintas agentų ir daugiaagentinių sistemų standartas.
GUI	(angl. graphical user interface) Grafinė vartotojo sąsaja.
JADE	(angl. Java Agent Development Framework) Agentų ir daugiaagentinių sistemų kūrimo įrankis, palaikantis FIPA ir ACL standartus, programuojamas Java programavimo kalba.
Java	Objektinio programavimo kalba, sukurta „Sun Microsystems“.
JavaBean	Pakartotinio panaudojimo java programinės įrangos komponentai.
LGPL-2	(angl. lesser general public licence) Laisvoji bendroji viešoji licencija yra laisvosios programinės įrangos licencija, taikoma programinėms bibliotekoms
OS	Operacinė sistema.
PMA	(angl. platform management agent) Platformos valdymo agentas.

7 LITERATŪROS SĄRAŠAS

- [1] Moraes, F. Pontificia Univ. Católica do Rio Grande do Sul (FACIN-PUCRS), Porto Alegre Amory, A. ; Calazans, N. ; Bezerra, E. ; Petrini, J. Using the CAN protocol and reconfigurable computing technology for Web-based smart house automation// Integrated Circuits and Systems Design, 2001, 14th Symposium on 2001, p. 38-43
- [2] F. Baumann, B. Jean-Bart, and A. Kung, P. Robin. “Electronic Commerce Services for Home Automation”, Dialog
- [3] Patricio, G. Fac. de Cienc. e Tecnol., Univ. Nova de Lisboa, Monte de Caparica, Portugal Gomes, L. Smart house monitoring and actuating system development using automatic code generation// Industrial Informatics, 2009. INDIN 2009. 7th IEEE International Conference on 23-26 June 2009, p. 256-261
- [4] Bischoff, U. Dept. of Comput., Lancaster Univ., Lancaster Sundramoorthy, V. ; Kortuem, G. Programming the smart home// Intelligent Environments, 2007. IE 07. 3rd IET International Conference on 24-25 Sept. 2007, p. 544-551
- [5] Al-Kuwari, A.M.A.H. Digital Ecosyst. & Bus. Intell. Inst., Curtin Univ. Perth, Perth, WA, Australia Ortega-Sanchez, C. ; Sharif, A. ; Potdar, V. User friendly smart home infrastructure: BeeHouse// Digital Ecosystems and Technologies Conference (DEST), 2011 Proceedings of the 5th IEEE International Conference on May 31 2011- June 3 2011, p. 257-262
- [6] Stauffer, H.B. Smart House, Upper Marlboro, MD Smart enabling system for home automation// Consumer Electronics, IEEE Transactions on May 1991, Volume: 37, Issue: 2, p. 29-35
- [7] Reis, M. Fac. de Cienc. e Tecnol., Univ. Nova de Lisboa, Lisbon, Portugal Campos-Rebelo, R. ; Gomes, L. Introducing model-based development within the reengineering of a Smart House controllers system// Industrial Informatics (INDIN), 2011 9th IEEE International Conference on 26-29 July 2011, p. 753-758
- [8] Home Control Assistant Version 10 User Guide [interaktyvus] Advanced Quonset Technology, Inc., 2010, birželis [žiūrėta 2011-09-18]. Priega per internetą:
<http://www.smarthome.com/manuals/126310.pdf>

- [9] HouseLinc Owner's Manual [interaktyvus] Smarthome, Inc., 2011, lapkritis [žiūrėta 2011-12-03]. Priega per internetą:
http://www.smarthome.com/manuals/HouseLinc_manual.pdf
- [10] CasaTunes XL User Guide [interaktyvus] CasaTools, LLC., 2008, spalvis [žiūrėta 2011-09-28]. Priega per internetą: <http://www.smarthome.com/manuals/9770.pdf>
- [11] HAI Web-Link II User's Guide [interaktyvus]HAI [žiūrėta 2011-10-15]. Priega per internetą: <http://www.smarthome.com/manuals/110908.pdf>
- [12] HomeRunner RBI User Manual [interaktyvus]JDS technologies, 2012, sausis [žiūrėta 2012-02-01]. Priega per internetą:
<http://www.jdstechologies.com/HomeRunnerRBI/HomeRunnerRBIUserManual.pdf>
- [13] Software Agents: An Overview [interaktyvus]Hyacinth S. Nwana,© Cambridge University Press, 1996 [žiūrėta 2012-03-01]. Priega per internetą:
<http://www.sce.carleton.ca/netmanage/docs/AgentsOverview/ao.html>
- [14] Bellifemine F., Caire G., Greenwood D., Developing multi-agent systems with JADE. John Wiley & Sons, Ltd 2007. 286 p.
- [15] Pranevičius H., Raudys Š., Rudžionis A., Rudžionis V., Ratkevičius K., Sakalauskaitė J., Makackas D., Agentinių sistemų modeliai [vadovėlis informatikos ir informatikos inžinerijos mokslo krypties doktorantams]. Kauno technologijos universitetas, 2008, 231 p.
- [16] Sharma, A. Dept. of Electr. & Comput. Eng., Univ. of Western Ontario, London, Ont. Capretz, M.A.M.Application maintenance using software agents// Source Code Analysis and Manipulation, 2001. Proceedings. First IEEE International Workshop on 2001, p. 55-64
- [17] Alex L.G. Hayzelden, John Bigham, eds., Software Agents for Future Communication Systems, Springer, 1998.
- [18] Buhler, P.A. Coll. of Charleston, SC, USA Huhns, M.N.Trust and persistence [software agents]// Internet Computing, IEEEVolume: 5, Issue: 2 p. 85-87
- [19] Stojkovic, V. Sch. of Comp., Math., & Nat. Sci., Morgan State Univ., Baltimore, MD Hongwei HuoSoftware Agents Action Securities// Computational Intelligence and Security, 2006 International Conference on3-6 Nov. 2006 p. 1026-1031

- [20] Alonso, F. Fac. de Informdtica, Univ. Politec. de Madrid, Madrid Fuertes, J.L. ; Martinez, L. ; Soza, H.Measuring the Social Ability of Software Agents// Software Engineering Research, Management and Applications, 2008. SERA '08. Sixth International Conference on20-22 Aug. 2008 p. 3-10
- [21] Z. Cheng, M. A. M. Capretz & M. Osano, “A Model for Negotiation Among Agents Based on the Transaction Analysis Theory,” in Proc. of the Second International Symposium on Theory,” in Proc. of the Second International Symposium on Theory,” in Proc. of the Second International Symposium on 427-433
- [22] Hamoui, F. LGI2P, Ecole des Mines d'Ales, Nimes, France Urtado, C. ; Vauttier, S. ; Huchard, M.SAASHA: A Self-Adaptable Agent System for Home Automation// Software Engineering and Advanced Applications (SEAA), 2010 36th EUROMICRO Conference on1-3 Sept. 2010, p. 227-230
- [23] Chen, W.H. Nat. Taipei Univ. of Technol., Taipei Tseng, W.S.An Agent Model for the Control of Smart Appliances// Fuzzy Information Processing Society, 2007. NAFIPS '07. Annual Meeting of the North American, 24-27 June 2007, p. 95-99
- [24] Karnouskos, S. de Holanda, T.N.Simulation of a Smart Grid City with Software Agents// Computer Modeling and Simulation, 2009. EMS '09. Third UKSim European Symposium on25-27 Nov. 2009
- [25] Kanamori, R. Nagoya Inst. of Technol., Nagoya, Japan Mizutani, N. ; Ito, T.Agent-based electrical power management model for houses equipped with storage battery and photovoltaic units// Service-Oriented Computing and Applications (SOCA), 2011 IEEE International Conference on12-14 Dec. 2011, p. 1-5
- [26] Shakshuki, E. Jodrey Sch. of Comput. Sci., Acadia Univ., Wolfville, NS, CanadaA methodology for evaluating agent toolkits// Information Technology: Coding and Computing, 2005. ITCC 2005. International Conference on4-6 April 2005, p. 391-396 Vol. 1
- [27] Camacho, D. Comput. Sci. Dept., Univ. Carlos III de Madrid, Leganes, Spain Aler, R. ; Castro, C. ; Molina, J.M.Performance evaluation of ZEUS, Jade, and SkeletonAgent frameworks// Systems, Man and Cybernetics, 2002 IEEE International Conference on6-9 Oct. 2002

- [28] Agent toolkits: A general overview of the market and an assesment of instructor satisfaction with utilizing tools in classroom [interaktyvus] Alexander Serenko, Bryan Detlor, Michael G. Degroote school of business McMaster university Hamilton, Ontario, 2002 [žiūrėta 2012-03-25]. Prieiga per internetą: http://foba.lakeheadu.ca/serenko/Agent_Toolkits_Working_Paper.pdf
- [29] Tools of the Trade: A Survey of Various Agent Based Modeling Platforms [interaktyvus] Cynthia Nikolai, Gregory Madey, Journal of Artificial Societies and Social Simulation vol. 12, no. 2 2, 2009 [žiūrėta 2012-03-21]. Prieiga per internetą: <http://jasss.soc.surrey.ac.uk/12/2/2.html>
- [30] Agent Building and Learning Environment (ABLE) [interaktyvus] IBM [žiūrėta 2012-03-28] Prieiga per internetą: <http://www.research.ibm.com/able/able.overview.shtml>
- [31] Survey of agent based modelling and simulation tools [interaktyvus] RJ Allan, Chadwick Library Science and Technology Facilities Council, 2010 [žiūrėta 2012-03-23]. Prieiga per internetą: <http://epubs.cclrc.ac.uk/bitstream/5601/DLTR-2010-007.pdf>
- [32] Cougaar Documentation [interaktyvus] Raytheon BBN Technologies [žiūrėta 2012-03-28] Prieiga per internetą: <http://cougaar.org/wp/documentation/>
- [33] The SimAgent TOOLKIT -- for Philosophers and Engineers (And Some Biologists, Psychologists and Social Scientists)(Also known as SIM_AGENT) [interaktyvus] Aaron Sloman, School of Computer Science, The University of Birmingham, 2010 [žiūrėta 2012-03-28] Prieiga per internetą: <http://www.cs.bham.ac.uk/research/projects/poplog/packages/simagent.html>
- [34] Calculation of power and energy needs for heating of buildings [interaktyvus] National Building Code of Finland, Part D5, Ministry of the Environment, Department of the Built Environment, 2012 [žiūrėta 2012-04-11] prieiga per internetą: http://www.puntofocal.gob.ar/notific_otros_miembros/fin43_t.pdf
- [35] Lietuvos hidrometeorologijos tarnybos prie Aplinkos ministerijos tinklalapis [interaktyvus] Lietuvos hidrometeorologijos tarnyba prie Aplinkos ministerijos, 2012 [žiūrėta 2012-04-17] Prieiga per internetą: <http://www.meteo.lt/>

- [36] Kauno rajono šilumos kainos [interaktyvus] Valstybinė kainų ir energetikos kontrolės komisija, 2012 [žiūrėta 2012-04-17] Prieiga per internetą: http://www.regula.lt/lt/siluma/zemelapis/kauno_raj
- [37] U-values and traditional buildings, *in situ* measurements and their comparisons to calculated values [interaktyvus] Dr. Paul Baker, Glasgow Caledonian University, 2011 sausis [žiūrėta 2012-04-17] Prieiga per internetą: <http://www.historic-scotland.gov.uk/hstp102011-u-values-and-traditional-buildings.pdf>