



KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS

Karolis Rušėnas

**VIDINIO DEBESŲ KOMPIUTERIJOS SPRENDIMO
AUTOMATIZAVIMAS IR RESURŲ VALDYMAS**

Baigiamasis magistro projektas

Vadovas
doc. dr. R. Butkienė

KAUNAS, 2015

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS

**VIDINIO DEBESŲ KOMPIUTERIJOS SPRENDIMO
AUTOMATIZAVIMAS IR RESURŲ VALDYMAS**

Baigiamasis magistro projektas
Informacinių sistemų inžinerijos studijų programa (kodas 621E15001)

Vadovas

doc. dr. R. Butkienė
2015-05-

Recenzentas

doc. dr. G. Vilutis
2015-05-

Projektą atliko

Karolis Rušėnas
2015-05-



KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS

(Fakultetas)

(Studento vardas, pavardė)

Informacinių sistemų inžinerijos studijų programa, 621E15001

(Studijų programos pavadinimas, kodas)

Baigiamojo projekto „Pavadinimas“
AKADEMINIO SAŽINGUMO DEKLARACIJA

20 ____ m. ____ d.

Kaunas

Patvirtinu, kad mano, **Karolio Rušėno**, baigiamasis projektas tema „VIDINIO DEBESŲ KOMPIUTERIJOS SPRENDIMO AUTOMATIZAVIMAS IR RESURSŲ VALDYMAS“ yra parašytas visiškai savarankiškai ir visi pateikti duomenys ar tyrimų rezultatai yra teisingi ir gauti sąžiningai. Šiame darbe nei viena dalis nėra plagijuota nuo jokių spausdintinių ar internetinių šaltinių, visos kitų šaltinių tiesioginės ir netiesioginės citatos nurodytos literatūros nuorodose. Įstatymų nenumatytų piniginių sumų už šį darbą niekam nesu mokėjęs.

Aš suprantu, kad išaiškėjus nesąžiningumo faktui, man bus taikomos nuobaudos, remiantis Kauno technologijos universitete galiojančia tvarka.

(vardą ir pavardę įrašyti ranka)

(parašas)

Rušėnas, K. Internal cloud deployment automation and resource management. *Final Degree Project of Master of Information Systems Engineering* / Supervisor Assoc. Dr. Rita Butkienė; Kaunas University of Technology, Faculty of Informatics.
Kaunas, 2015. 122 p.

SUMMARY

This project aims at systems that operate large pools of physical computing resources in datacenters. It also focuses on systems that automate OpenStack cloud computing operating system initial deployment. In analysis section we compare similar systems, analyze their strengths and weaknesses so we can define main functionalities for the new system that could manage datacenters, install and operate OpenStack so it could cover end-to-end installation.

In the following sections we investigate what steps are usually taken to find physical servers in the network, manually or semi-automatically install operating systems to servers, establish remote control. These flows are then documented and activity diagrams created. Using this knowledge we can then proceed to systems engineering.

Since OpenStack components are written in Python – this high level programming language was also chosen as a main tool for system development. There are numerous reasons to choose this language – from OpenStack python clients, HP IPMI python client to other libraries that were used to speed up system development (such as NMAP client for network scanning). Django was chosen as a core framework to provide structure for the applications. Since distributed application architecture was chosen – two systems were created: 1) SiteManager for local datacenter deployment which can find and install operating systems to servers and then carry on with OpenStack installation; 2) CloudMap for centralized automated information gathering which can aggregate data from multiple SiteManagers with their respective OpenStack environments.

System integration tests took place in several datacenters across the globe and results can be seen in one of the last sections. There are also some suggestions for further development such as deployment with Docker containers, configuration management using tools like Chef, Ansible and cloud capacity forecasting.

TURINYS

Lentelių sąrašas	8
Paveikslų sąrašas.....	10
Terminų ir santrumpų žodynas	13
Įvadas	14
1.1. OpenStack sistema	14
1.2. Darbo problematika ir aktualumas.....	15
1.3. Darbo tikslas ir uždaviniai	15
1.4. Darbo rezultatai ir jų svarba.....	15
1.4.1. Kam ir ką keičia šis sprendimas	16
1.5. Darbo struktūra	16
2. Probleminės srities analizė.....	17
2.1. Analizės tikslas	17
2.2. Tyrimo objektas, sritis ir problema.....	17
2.2.1. Sprendimų kūrimas OpenStack sistemai	17
2.2.2. Galimi problemos sprendimo būdai.....	19
2.3. OpenStack komponentų diegimo (veiklos proceso) analizė.....	20
2.4. Debesų kompiuterijos sprendimo priežiūros (veiklos proceso) analizė	21
2.5. Tyrimo objekto naudotojų analizė	23
2.6. Esamų problemos sprendimo metodų analizė.....	24
2.6.1. Mirantis Fuel.....	24
2.6.2. QuotaViewer	26
2.6.3. HP Systems Insight Manager (HP SIM).....	27
2.6.4. RackMonkey	27
2.6.5. TripleO ir Tuskar	29
2.6.6. Esamų sprendimų palyginimo išvados	29
2.7. Darbo tikslas, pagrindiniai uždaviniai, problemos ir siekiami privalumai	30
2.8. Siekiamo sprendimo apibrėžimas	31
2.9. Analizės išvados.....	32
3. SiteManager ir CloudMap Sprendimo reikalavimų specifikacija ir projektas.....	34
3.1. Reikalavimų specifikacija	34
3.1.1. Bendros sistemos funkciniai reikalavimai	34
3.1.2. Vartojimo (panaudos) atvejų diagramos.....	35
3.1.3. CloudMap sistemos panaudos atvejų diagrama.....	53
3.2. Sistemų nefunkciniai reikalavimai.....	59
3.3. Reikalavimų apibendrinimas	59
4. CLOUDMAP ir sitemanager sprendimo projektas.....	60

4.1. Sistemų loginė architektūra.....	60
4.1.1. CloudMap ir SiteManager sistemų veiklos diagramos.....	61
4.1.2. CloudMap gautos JSON tipo informacijos konversijos į <i>Dict</i> tipą pavyzdys	65
4.1.3. SiteManager serverių importavimo iš CSV failo veiklos diagrama	66
4.1.4. SiteManager serverių paieškos tinkle veiklos diagrama.....	67
4.1.5. SiteManager daugelio serverių operacinės sistemos atvaizdo priskyrimo veiklos diagrama.....	69
4.2. Duomenų bazės modelis	69
4.2.1. CloudMap duomenų bazės modelis.....	70
4.3. Detalus projektas.....	73
4.3.1. Komponentų architektūra	73
4.3.2. Diegimo modelis.....	74
4.4. Vartotojo sąsajų projektai	76
4.4.1. CloudMap pagrindinio puslapio sąsaja.....	76
4.4.2. CloudMap duomenų centrų puslapio sąsaja	79
4.4.3. CloudMap detalios informacijos apie SiteManager puslapio sąsaja	79
4.4.4. CloudMap detalios informacijos apie OpenStack aplinką puslapio sąsaja.....	80
4.4.5. CloudMap testų paleidimo vartotojo sąsaja.....	80
4.4.6. OpenStack virtualios mašinos kūrimo ir ištrynimo testavimo istorijos sąsajos langas	82
4.4.7. SiteManager serverių sąrašo langas.....	82
4.4.8. Serverių valdymo vartotojo sąsaja.....	86
4.4.9. Serverių spintų vartotojo sąsaja	87
4.4.10. SiteManager naujos OpenStack aplinkos kūrimo vartotojo sąsaja.....	88
4.4.11. SiteManager OpenStack nustatymų vartotojo sąsaja.....	88
4.4.12. SiteManager OpenStack serverių vartotojo sąsaja.....	89
4.4.13. SiteManager OpenStack hipervizorių vartotojo sąsaja.....	90
4.4.14. SiteManager OpenStack serverių grupavimo vartotojo sąsaja	91
4.4.15. SiteManager OpenStack virtualių mašinų šablonų vartotojo sąsaja.....	91
4.4.16. SiteManager OpenStack operacinių sistemų atvaizdų vartotojo sąsaja.....	92
4.4.17. SiteManager OpenStack virtualių mašinų vartotojo sąsaja	92
4.4.18. SiteManager OpenStack sistemos vidinių projektų langas.....	93
5. Sitemanager ir Cloudmap sprendimo realizacija ir testavimas.....	95
5.1. Sprendimo realizacijos ir veikimo aprašas	95
5.2. Testavimo modelis, duomenys, rezultatai.....	95
5.2.1. CloudMap automatinio testavimo modelis.....	95
5.2.2. Sistemoje vykstančių procesų įrašymas.....	107
5.2.3. Naudotojų atliktas visų sistemų integracijos testas.....	109
5.2.4. Testavimo rezultatai.....	110

6. Eksperimentinis CloudMap ir SiteManager sistemos tyrimas	112
6.1. Eksperimento planas	112
6.2. Eksperimento rezultatai	112
6.2.1. Išvados	114
6.3. Sprendimo veikimo ir savybių analizė, kokybės kriterijų įvertinimas	114
6.3.1. Sprendimo veikimo analizė	114
6.3.2. Sprendimo savybių analizė	115
6.3.3. Sprendimo kokybės kriterijų įvertinimas.....	115
6.4. Sprendimo taikymo rekomendacijos.....	115
7. Rezultatų apibendrinimas ir išvados	116
7.1. Rezultatų apibendrinimas.....	116
7.2. Išvados	116
7.3. Pasiūlymai tolimesniam sistemos tobulinimui.....	117
8. Literatūra	119
9. Priedai	120
9.1. SiteManager sistemos duomenų bazės schema.....	120

LENTELIŲ SĄRAŠAS

1. lentelė. Esamų sprendimų palyginimas.....	29
2. lentelė. Serverių informacijos peržiūrėjimo panaudos atvejis	35
3. lentelė. Detalios serverio informacijos peržiūrėjimo panaudos atvejis.....	36
4. lentelė. Informacijos redagavimo panaudos atvejis (serverio).....	36
5. lentelė. Atliktų veiksmų su serveriu informacijos peržiūrėjimo panaudos atvejis	36
6. lentelė. Serverio valdymo per IPMI sąsają panaudos atvejis.....	37
7. lentelė. Serverių importavimo iš CSV failo panaudos atvejis	37
8. lentelė. Tinklo skenavimo ir automatinės serverių paieškos panaudos atvejis.....	37
9. lentelė. Daugelio serverių automatinio konfigūravimo pagal pasirinktą operacinės sistemos atvaizdą panaudos atvejis.....	37
10. lentelė. Automatizuoto serverių tipo aptikimo panaudos atvejis	38
11. lentelė. OpenStack aplinkos duomenų peržiūrėjimo panaudos atvejis	38
12. lentelė. OpenStack aplinkos kūrimo panaudos atvejis.....	39
13. lentelė. Valdymo serverių diegimo panaudos atvejis.....	39
14. lentelė. Skaičiavimo serverių diegimo panaudos atvejis	40
15. lentelė. OpenStack serverių grupių kūrimo panaudos atvejis	41
16. lentelė. OpenStack serverių grupės redagavimo panaudos atvejis	41
17. lentelė. OpenStack serverių grupės ištrynimo panaudos atvejis	41
18. lentelė. SiteManager serverių grupių peržiūrėjimo panaudos atvejis	41
19. lentelė. SiteManager naujos serverių grupės kūrimo panaudos atvejis	42
20. lentelė. SiteManager serverių grupės duomenų redagavimas	42
21. lentelė. Jungimosi į virtualią mašiną per VNC konsolę panaudos atvejis	43
22. lentelė. Duomenų eksporto į CSV formatą panaudos atvejis.....	43
23. lentelė. SiteManager sistemoje esančių statinių duomenų peržiūrėjimo panaudos atvejis.....	43
24. lentelė. SiteManager sistemos statinių duomenų redagavimo panaudos atvejis.....	44
25. lentelė. Prisijungimo į SiteManager sistemą panaudos atvejis	44
26. lentelė. Atsijungimo iš SiteManager sistemos panaudos atvejis.....	45
27. lentelė. Serverių grupių informacijos gavimo panaudos atvejis	46
28. lentelė. Naujos serverių grupės kūrimo panaudos atvejis	46
29. lentelė. Serverių grupės redagavimo panaudos atvejis	47
30. lentelė. Serverių grupės ištrynimas	47
31. lentelė. Sistemos komponentų adresų ir protokolų gavimo panaudos atvejis.....	47
32. lentelė. Nustatytų limitų gavimo panaudos atvejis	47
33. lentelė. Hipervizorių informacijos gavimo panaudos atvejis.....	48
34. lentelė. Operacinių sistemų atvaizdų gavimo panaudos atvejis.....	48
35. lentelė. Sistemoje registruotų naudotojų sąrašo gavimo panaudos atvejis	48
36. lentelė. Projektų informacijos sąrašo gavimo panaudos atvejis.....	48
37. lentelė. Virtualių mašinų šablonų gavimo panaudos atvejis	49
38. lentelė. Virtualių mašinų sąrašo gavimo panaudos atvejis	49
39. lentelė. Prisijungimo į VNC konsolę panaudos atvejis.....	49
40. lentelė. Fizinį serverių informacijos surinkimo panaudos atvejis.....	51
41. lentelė. OpenStack aplinkų sąrašo gavimo panaudos atvejis.....	52
42. lentelė. Aplinkos hipervizorių informacijos gavimo panaudos atvejis.....	52
43. lentelė. Detalios OpenStack aplinkos servisų būklės analizės gavimo panaudos atvejis	52
44. lentelė. Duomenų persiuntimo į CloudMap sistemą panaudos atvejis	53
45. lentelė. OpenStack API testavimo panaudos atvejis.....	53
46. lentelė. OpenStack virtualios mašinos kūrimo ir sunaikinimo testavimo panaudos atvejis	53
47. lentelė. Prisijungimo į administratoriaus panelę panaudos atvejis	54
48. lentelė. Testavimo paskyros sukūrimo panaudos atvejis	54

49. lentelė. Testavimo paskyros redagavimo panaudos atvejis	55
50. lentelė. Duomenų centrų informacijos redagavimo panaudos atvejis.....	55
51. lentelė. SiteManager sistemų informacijos redagavimo panaudos atvejis.....	55
52. lentelė. OpenStack aplinkų informacijos redagavimo panaudos atvejis.....	55
53. lentelė. Testavimo paskyros priskyrimo panaudos atvejis.....	56
54. lentelė. OpenStack virtualios mašinos kūrimo ir trynimo testo panaudos atvejis	56
55. lentelė. Testų istorijos peržiūrėjimo panaudos atvejis	56
56. lentelė. OpenStack API testo paleidimo panaudos atvejis.....	56
57. lentelė. OpenStack aplinkų sąrašo peržiūrėjimo panaudos atvejis	57
58. lentelė. Detalios informacijos apie aplinką peržiūrėjimo panaudos atvejis.....	57
59. lentelė. OpenStack aplinkoje veikiančių servisų peržiūrėjimo panaudos atvejis.....	57
60. lentelė. SiteManager sistemų sąrašo peržiūrėjimo panaudos atvejis	57
61. lentelė. Detalios SiteManager informacijos peržiūrėjimo panaudos atvejis.....	57
62. lentelė. Duomenų centrų peržiūrėjimo panaudos atvejis	58
63. lentelė. Pagrindinio informacinio puslapio peržiūrėjimo panaudos atvejis	58
64. lentelė. Prisijungimo panaudos atvejis.....	58
65. lentelė. Atsijungimo panaudos atvejis	58
66. lentelė. „Environment“	70
67. lentelė. „Sitemanager“	71
68. lentelė. „Register“	71
69. lentelė. „Datacenter“	71
70. lentelė. „Country“	72
71. lentelė. „EnvironmentTestingAccount“	72
72. lentelė. „GlobalViewAccount“	72
73. lentelė. „NovaInstanceCreationTest“	72
74. lentelė. „Service“	72
75. lentelė. SiteManager sukūrimo su pilnais duomenimis testo aprašymas.....	99
76. lentelė. SiteManager kūrimo be OpenStack aplinkų testo aprašymas	99
77. lentelė. SiteManager kūrimo perduodant sistemos vardą testo aprašymas.....	99
78. lentelė. OpenStack aplinkos sukūrimo testo aprašymas	99
79. lentelė. OpenStack aplinkos sukūrimo perduodant nepilnus duomenis testo aprašymas	100
80. lentelė. Duomenų kairėje esančiam meniu perdavimo funkcijos testo aprašymas.....	100
81. lentelė. Suformuoto JSON objekto, skirto į pagrindinį informacinį puslapį testo aprašymas	100
82. lentelė. Servisų kūrimo funkcijos testo aprašymas	100
83. lentelė. Duomenų centrų informacijos perdavimo į žemėlapi testą aprašymas.....	100
84. lentelė. Kelių testų grupės paleidimo kartu testo aprašymas	101
85. lentelė. Nepilnų duomenų testo aprašymas.....	101
86. lentelė. SiteManager registracijos, kai nepateikiami fizinių serverių kiekiai testo aprašymas....	101
87. lentelė. Duomenų centro patikrinimo funkcijos neperduodant kodo testo aprašymas	101
88. lentelė. Duomenų centro funkcijos patikrinimo perduodant kodą testo aprašymas	102
89. lentelė. SiteManager sistemos būklės tikrinimo testo aprašymas.....	102
90. lentelė. Duomenų skaičiaus pradiniame faile testo aprašymas	102
91. lentelė. Pasaulio žemėlapi duomenų formavimo testo aprašymas.....	102
92. lentelė. Duomenų centro be koordinacijų testo aprašymas.....	102

PAVEIKSLŲ SĄRAŠAS

1. pav. OpenStack sistemos diagrama. Paimta iš http://www.openstack.org/software/	14
2. pav. OpenStack sistemos koncepcinė architektūra [6].....	18
3. pav. OpenStack diegimo panaudos atvejų diagrama	20
4. pav. Rankiniu būdu atliekamas OpenStack diegimas ir konfigūravimas.....	21
5. pav. OpenStack minimalios priežiūros panaudojimo atvejų diagrama.....	22
6. pav. Mirantis Fuel aptikęs serverius	24
7. pav. Mirantis Fuel – OpenStack aplinkos diegimo pradžia, aukšto patikimumo konfigūracijos pasirinkimas	25
8. pav. Galimybė rinktis įvairias OpenStack versijas Mirantis Fuel sistemoje.....	25
9. pav. Fuel - sistemos automatiniai testai	26
10. pav. RackMonkey sąsaja su serverių informacija.....	28
11. pav. RackMonkey serverių spintų informacija	28
12. pav. Automatizuotas OpenStack diegimas ir konfigūravimas	32
13. pav. SiteManager sistemos panaudos atvejų diagrama	35
14. pav. Valdymo serverių pasirinkimo langas.....	39
15. pav. Tinklo konfigūracijos parametrai	40
16. pav. Blokinio duomenų saugojimo komponento nustatymai.....	40
17. pav. Objektinio duomenų saugojimo komponento nustatymai.....	40
18. pav. Duomenų bazės konfigūravimas	40
19. pav. Naujos serverių grupės sukūrimo mygtukas	41
20. pav. Serverių atvaizdavimas OpenStack serverių grupėje	41
21. pav. SiteManager statinės informacijos meniu esanti nuoroda į serverių grupes	42
22. pav. Naujos serverių grupės SiteManager sistemoje kūrimo langas.....	42
23. pav. Parametrų pridėjimo sąsaja	43
24. pav. VNC konsolės atidarymas.....	43
25. pav. Duomenų iš lentelių eksportavimo į CSV sąsaja	43
26. pav. SiteManager sistemoje saugomi statiniai duomenų tipai	44
27. pav. SiteManager prisijungimo sąsaja	44
28. pav. Atsijungimui iš SiteManager skirtas meniu	45
29. pav. SiteManager ir OpenStack sistemų panaudos diagrama	46
30. pav. SiteManager, CloudMap ir OpenStack sistemų tarpusavio komunikacijos panaudos atvejų diagrama.....	51
31. pav. CloudMap sistemos panaudos atvejų diagrama	54
32. pav. CloudMap prisijungimo į administratoriaus panelę sąsaja	54
33. pav. Atsijungimui iš SiteManager skirtas meniu	59
34. pav. CloudMap ir SiteManager informacinių sistemų loginė architektūra.....	60
35. pav. SiteManager registracijos į CloudMap veiklos diagrama	61
36. pav. CloudMap pagrindinio lango veiklos diagrama.....	62
37. pav. Virtualios mašinos kūrimo ir ištrynimo testas.....	64
38. pav. SiteManager registracijos į CloudMap duomenų pavyzdys.....	66
39. pav. Informacijos apie serverius importavimas iš CSV failo.....	67
40. pav. Automatinės serverių paieškos tinklo veiklos diagrama	68
41. pav. Daugelio serverių operacinės sistemos atvaizdo konfigūravimo veiklos diagrama.....	69
42. pav. Pav. CloudMap duomenų bazės modelis, sugeneruotas per IntelliJ IDE.....	70
43. pav. Sistemos komponentų architektūra	73
44. pav. Sistemos elementų diegimo modelis	75
45. pav. CloudMap pagrindinio lango vartotojo sąsaja	77
46. pav. Išskleistas navigacijos meniu	78
47. pav. Papildomas meniu	79

48. pav. Duomenų centrų vartotojo sąsaja	79
49. pav. CloudMap sistemoje registruoto SiteManager detalios informacijos langas	80
50. pav. CloudMap sistemoje OpenStack aplinkos detalios informacijos langas	80
51. pav. OpenStack testavimo sąsajos langas	81
52. pav. OpenStack virtualios mašinos kūrimo testo pavyzdys	82
53. pav. OpenStack virtualios mašinos kūrimo ir ištrynimo testavimo istorijos sąsaja.....	82
54. pav. SiteManager serverių sąrašo langas	83
55. pav. Serverių valdymo mygtukai	83
56. pav. Serverio pridėjimo į SiteManager sistemą langas	83
57. pav. Serverių pridėjimo iš CSV failo sąsajos langas	84
58. pav. Serverių pridėjimo skenuojant tinklą sąsajos langas	84
59. pav. Serverių veiksmų mygtuko meniu.....	85
60. pav. „bulk image assign“ funkcijos sąsajos langas	85
61. pav. Serverių detali informacija SiteManager sistemoje	85
62. pav. Serverio, priskirto OpenStack aplinkai atvaizdavimas	85
63. pav. SiteManager serverių duomenų eksportavimo ir filtravimo sąsaja.....	86
64. pav. Serverio valdymo vartotojo sąsaja	86
65. pav. Serverio būklės sąsaja	87
66. pav. Serverių detalios informacijos vartotojo sąsaja.....	87
67. pav. Serverių spintos vaizdas	88
68. pav. OpenStack aplinkos kūrimo lentelė.....	88
69. pav. OpenStack aplinkos pagrindinių nustatymų langas	89
70. pav. Naujų serverių į OpenStack aplinką pridėjimo sąsaja.....	89
71. pav. Naujų skaičiavimo serverių pridėjimo langas	90
72. pav. Hipervizorių informacijos langas	90
73. pav. OpenStack serverių grupavimo sąsaja.....	91
74. pav. Virtualių mašinų šablonų peržiūrėjimo sąsaja	92
75. pav. OpenStack operacinių sistemų atvaizdų peržiūrėjimo sąsaja.....	92
76. pav. OpenStack aplinkoje sukurtų virtualių mašinų informacijos langas	93
77. pav. Metaduomenų langas.....	93
78. pav. Vidiniai projektai ir jų išnaudojami resursai	94
79. pav. Teisingų duomenų testavimui pavyzdys	97
80. pav. Duomenų testavimui pavyzdys, kai nėra informacijos apie OpenStack aplinkas.....	97
81. pav. Duomenų testavimui pavyzdys, kai perduodamas SiteManager sistemos pavadinimas.....	97
82. pav. Duomenų testavimui pavyzdys, kai trūksta „envs“ elemento	98
83. pav. Duomenų testavimui pavyzdys, kai pateikiami tik du elementai.....	98
84. pav. Pradinio duomenų failo fragmentas	98
85. pav. CloudMap automatinių testų programinio kodo padengimo rezultatas	103
86. pav. Specifinių CloudMap automatinių testų trukmė ir rezultatai	103
87. pav. CloudMap sąsajos langas su Django Debug Toolbar.....	104
88. pav. Detali Django Debug Toolbar SQL užklausų informacija.....	105
89. pav. SiteManager duomenų surinkimo ir persiuntimo funkcijos paleidimas per IPython konsolę	105
90. pav. CloudMap gauti duomenys iš SiteManager sistemos	106
91. pav. Struktūrizuotų duomenų fragmentas iš SiteManager registracijos	106
92. pav. CloudMap OpenStack aplinkos langas po registracijos.....	107
93. pav. SiteManager OpenStack aplinkos hipervizorių informacija	107
94. pav. Sistemoje vykstančių procesų įrašymo konfigūracija	108
95. pav. Pranešimų apie sistemos procesus veiklos diagrama	108
96. pav. Sistemos įrašai vykdant SiteManager registraciją į CloudMap	109
97. pav. Sistemų integracijos testo metu sukurtas SiteManager užsiregistruoja CloudMap sistemoje	109
98. pav. Detali OpenStack aplinkos informacija.....	110

99. pav. Produkcinės CloudMap realizacijos programos langas	112
100. pav. SiteManager serverių spintos vaizdas	113
101. pav. SiteManager detalios serverio informacijos langas.....	113
102. pav. Pav. SiteManager duomenų bazės modelis, sugeneruotas per IntelliJ IDE	120

TERMINŲ IR SANTRUMPŲ ŽODYNAS

FTP (File Transfer Protocol) – duomenų perdavimo protokolas.

HTTP (Hypertext Transfer Protocol) – pagrindinis duomenų perdavimo protokolas internete.

Debesis – plačiai naudojamas terminas, sutrumpinantis „debesų kompiuterijos“ terminą, nusako tam tikrą automatizacijos lygį infrastruktūroje.

SSH (Secure Shell) – kriptografinis duomenų perdavimo protokolas, skirtas saugioms komunikacijoms.

CPU (Central Processing Unit) - vienas iš pagrindinių kompiuterio sistemos komponentų, kuris rūpinasi programų siunčiamomis instrukcijomis.

vCPU (virtual CPU) - tai virtualus CPU vienetas, priskiriamas virtualioms mašinoms.

IPMI (Intelligent Platform Management Interface) – autonominė kompiuterio posistemė, kuri suteikia prieigą valdyti ir stebėti kompiuterio parametrus nepriklausomai nuo jo operacinės sistemos ar CPU tipo.

HP ILO (HP Integrated Lights-Out) – HP sukurta IPMI patobulinta sąsaja, kuri leidžia prisijungti prie serverio IPMI sąsajos per naršyklę, be to, turi papildomą RestAPI sąsają.

Hipervizorius – programa, kuri valdo virtualias mašinas ir operacines sistemas vienoje kompiuterinėje sistemoje.

API (Application Programming Interface) – viešai pasiekiamų funkcijų, metodų arba klasių apibrėžimas, kuriomis sistemos naudotojai gali pasiekti jos funkcijas.

Rest API (Representational State Transfer API) – tai programinės įrangos architektūra, nusakanti standartus ir geriausias praktikas, skirtas kurti didelio našumo internetines paslaugas.

VNC (Virtual Network Computing) – grafinė sąsaja, leidžianti nuotoliniu būdu valdyti serverius.

AD (Active Directory) – Microsoft sukurta sistema, autentifikuojanti ir autorizuojanti vartotojus į kompiuterius.

NMAP (Network Mapper) – Programa, skirta skenuoti tinklą. Dažnai naudojama tikrinant tinklo saugumą.

JSON (JavaScript Object Notation) – duomenų perdavimo standartas, kurį lengva skaityti tiek žmogui, tiek programai.

SDN (Software Defined Networks) - programinės įrangos kuriami kompiuterių tinklai.

IaaS (Infrastructure as a Service) – terminas, reiškiantis paslaugą, kai suteikiama infrastruktūra, kurioje galima diegti savo platformas, sistemas.

PaaS (Platform as a Service) – terminas, reiškiantis paslaugą, kai suteikiama platforma, kurioje galima talpinti savo taikomąsias programas (dažniausiai reikia įtraukti kiekvienai platformai specifinį konfigūracijų failą, kuris nurodo, kaip platforma turi paleisti taikomąją programą).

IVADAS

Šis darbas priklauso informacinių sistemų inžinerijos magistrantūros studijų programai. Darbo specializacija – debesų kompiuterijos [1] sprendimų diegimo automatizacija, resursų valdymas ir stebėjimas.

1.1. OpenStack sistema

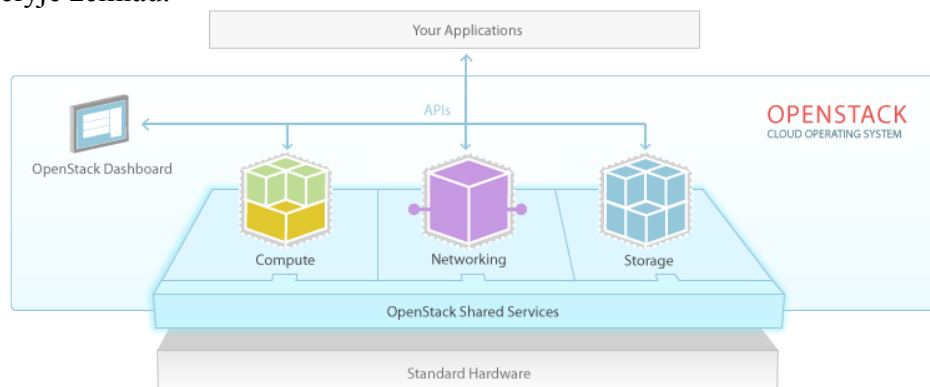
Debesų kompiuterija (*angl. Cloud Computing*) yra modelis, įgalinantis vartotoją greitai ir patogiai gauti jam reikalingus kompiuterių resursus. Šie resursai gali būti įvairūs: kompiuterių tinklai, serveriai, taikomosios programos, duomenų saugojimas. Gautos paslaugos turi būti nesunkiai valdomos, o sistemos administratorius turi arba visiškai arba tik minimaliai dalyvauti procesuose paslaugų užsakymo ir atsisakymo procesuose [1].

Debesų kompiuterijos sprendimai įgauna vis didesnę pagreitį tiek didelėse, tiek ir vidutinio dydžio bei mažose įmonėse. Pagal naujausius duomenis, 81% apklaustų įmonių iš Jungtinės Karalystės, JAV, Vokietijos, Prancūzijos ir Australijos, kurias apklausė *Forrester*, teigė, kad jau perkėlinėja arba planuoja perkelti svarbiausias taikomas programas į debesį per artimiausius du metus[2]. Šių įmonių tipus skiria tik debesies sprendimo įgyvendinimas. Mažose ir vidutinio dydžio įmonėse renkama dažniau išoriniai debesų kompiuterijos sprendimai – tokiu atveju nereikia skirti daug resursų debesies dizaino kūrimui, įrangos tiekimo ir palaikymo užtikrinimui, debesies sprendimo diegimui. Tačiau, viskas pasikeičia, kai reikalavimai išauga ir atsiranda poreikis dideliems resursų kiekiams arba tenka prisitaikyti prie vidinių reikalavimų [3], kurių negali patenkinti išoriniai tiekėjai – tuomet tenka kurti debesį įmonės viduje. Dažniausiai šie reikalavimai yra susiję su saugumo užtikrinimu, nes bankai, kaip ir kitos įmonės, kurios dirba tiesiogiai su žmonių duomenimis yra reguliuojami valstybiniu mastu. Šiuo metu ypač populiarus debesų kompiuterijos sprendimas – OpenStack, tai sistema, susidedanti iš savarankiškų komponentų, kurie, veikdami kartu, užtikrina prieigą prie turimų resursų per Rest API, užtikrina galimybę veikti nepriklausomai nuo kompiuterinės įrangos gamintojo (*hardware agnostic*) bet siūlo praktiškai visas paslaugas, kurios reikalingos sukurti savo nuosavą debesų kompiuterijos sprendimą.

Šis tiriamasis darbas yra glaudžiai susijęs su OpenStack debesų kompiuterijos sprendimu. Tai didžiulis projektas, vienas iš greičiausiai augančių atviro kodo projektų visame pasaulyje. Per šiek tiek daugiau nei keturis metus pasiekti rezultatai [4]:

- Įtraukta daugiau nei 18131 žmonių.
- 451 kompanija, kuri prisideda prie OpenStack kūrimo.
- Sistema naudojama 145 šalyse.
- Daugiau nei 20 milijonų programinio kodo eilučių.

Paprasčiausias OpenStack sistemos apibrėžimas būtų – debesų kompiuterijos operacinė sistema, kuri valdo didžiulius skaičiavimo, duomenų saugojimo ir tinklo valdymo resursus duomenų centruose. Visa tai yra valdoma per OpenStack Horizon komponentą arba API. Supaprastina diagrama paveikslėlyje žemiau.



1. pav. OpenStack sistemos diagrama. Paimta iš <http://www.openstack.org/software/>

1.2. Darbo problematika ir aktualumas

Nors OpenStack sistema yra pasiekusi jau gana brandžią gyvavimo fazę, tačiau vis dar nėra prieita prie vienos ar kelių pagrindinių papildomų sistemų, kurios padėtų valdyti turimus fizinius resursus, agreguoti jų informaciją bei automatizuotai paleisti diegimo programas turimuose resursuose dideliais mastais. Dideli mastai, šiuo atveju būtų po maždaug 600 serverių per 6 duomenų centrus visame pasaulyje, kuriuose būtų reikalinga keisti tokius parametrus kaip serverio vardas, tinklo konfigūracija (skirtinguose geografiniuose regionuose bus priskirti skirtingi potinkliai, taip pat gali būti poreikis perduoti į virtualias mašinas įvairius metaduomenis, grupuoti jas pagal projekto numerius, pagal operacines sistemas ir kitus, bizniui reikalingus parametrus. Tokių virtualių mašinų visame pasaulyje pirmojoje debesies fazėje tikimasi turėti 50 iki 96 tūkstančių. Su šia problema susiduria ne tik OpenStack, tačiau ir daugelis kitų šiuolaikinių sistemų, kurios nėra monolitinės, o įvairius savo komponentus laiko skirtinguose serveriuose arba virtualiose mašinose [5].

Kita problema, kuri atsiranda jau turint funkcionuojantį debesį – užtikrinimas, kad visi servisai veikia ir kad yra galimybė kurti virtualias mašinas. Taigi, reikalingas įrankis, kurio pagalba būtų galima matyti visas debesies pasiekiamumo zonas (*angl. Availability Zones*), servisus jose, jų būklę bei kitus pagrindinius duomenis, tokius kaip resursų utilizacija (kiek priskirta, kiek panaudota). Šis įrankis taip pat galėtų pasitarnauti žmonėms, kurie tiesiogiai dirba su debesies priežiūra, nes vartotojai, kreipdamiesi pagalbos dėl savo virtualių mašinų dažniausiai gali nežinoti, kuriuose duomenų centruose yra jų mašinos.

1.3. Darbo tikslas ir uždaviniai

Šio darbo tikslas yra išsaugoti OpenStack diegimui reikalingas žinias, kurios išliktų nepriklausomai nuo darbuotojų kaitos, bei pateikti instaliacijos ir priežiūros karkasus bei algoritmus, kurie galėtų būti toliau tobulinami kartoiant, surandant logines spragas, komponentų neatitikimus bei spręsti resursų būsenų stebėjimo ir valdymo problemą, kuri tampa vis aktualesnė, kai nebetinka standartiniai įrankiai tokie kaip *Nagios*, *Splunk* infrastruktūros stebėjimui bei valdymui.

Tiriamajame darbe taip pat siekiama identifikuoti OpenStack sistemos diegimo metu kylančias problemas dėl automatizacijos trūkumo. Didžiausios problemos kyla dėl didelio reikalingų komponentų kiekio ir jų konfigūracijų. Praktiškai tampa labai sudėtinga atlikti OpenStack diegimą kelis kartus iš eilės dėl įvairių priežasčių: kernelio neatitikimų, programų įrašymo tvarkos, ugniasienių nustatymų ir t.t., todėl dažnai diegimai baigiasi nesėkmingai. Diegimas turėtų susidėti iš automatizuotų žingsnių, kuriuos būtų galima pakartoti nepriklausomai nuo visos instaliacijos (turėtų būti galimybė atskirai diegti valdymo ir skaičiavimo serverius). Išsprendus šias problemas – organizacija gali vietoj kelių šimtų Linux, kompiuterių tinklų, duomenų bazių ir įvairių kitų specialybių inžinierių turėti tik po kelis visų sričių inžinierius, kurie paleistų automatizuotą debesų kompiuterijos sprendimo diegimą duomenų centruose.

Užsibrėžtam tikslui įvykdyti reikalinga atlikti šiuos uždavinius:

1. Išanalizuoti analogiškus sprendimus OpenStack diegimui, nustatant šių sprendimų pagrindines savybes (trūkumus ir privalumus).
2. Suprojektuoti sprendimus, kurie reikalingi automatizuoti diegimą ir valdyti bei stebėti turimus resursus.
3. Realizuoti sprendimą.
4. Eksperimentiškai ištirti sprendimą (testuoti tiek automatinių testų pagalba, tiek ir realioje aplinkoje)
5. Aprašyti gautus rezultatus.

1.4. Darbo rezultatai ir jų svarba

Kuriama sistema yra viena iš pagrindinių debesies dalių, be kurios nebūtų įmanoma sukurti žemiausios grandies infrastruktūros, priskiriamos IaaS (*angl. Infrastructure as a Service*) kategorijai. Šiuo metu rinkoje nėra sprendimų, kurie savyje turėtų visas reikalingas funkcijas arba tik dalinai jas

tenkina. Pavyzdžiui, rinkoje esantys produktai, kurie yra skirti diegti OpenStack sistemą į fizinius serverius – negali jų valdyti per IPMI sąsają, taigi negalima jiems nuotoliniu būdu priskirti operacinės sistemos atvaizdo, o reikia pradžioje rankiniu būdu prisijungti per terminalą ir sukonfigūruoti SSH raktų poras, kad vėliau galėtų atlikti diegimą. Taigi, sistemos diegėjams reiktų papildomų paskyrų ir teisių, kad galėtų jungtis į fizinius serverius, o dėl to gali atsirasti saugumo spragos.

Kita priežastis, kodėl svarbu sukurti šį sprendimą – rinkoje nėra įrankių, kurie galėtų sekti OpenStack instaliacijas įvairiuose duomenų centruose visame pasaulyje, taigi net pasirinkus kitą diegimui skirtą įrankį – jį reiktų modifikuoti arba šalia jo įdiegti atskirą servisą, kuris sektų jo komandas ir perduotų į kitą sistemą, kuri renka ir agreguoja duomenis iš visų duomenų centrų.

1.4.1. Kam ir ką keičia šis sprendimas

Šis sprendimas labiausiai pasitarnauja OpenStack diegėjui, kadangi ne tik išsaugomos žinios, reikalingos sėkmingam sistemos įdiegimui, tačiau ir sutaupoma daug laiko:

- Sutrumpina reikalingą laiką OpenStack įdiegimui nuo ~2 dienų iki 1 valandos.
- Suteikia sąsają naršyklėje, kurioje mato iškart iš visų komponentų apjungtą informaciją ir greitai gali reaguoti į iškilusias problemas.
- Nebereikia rankiniu būdu jungtis į kiekvieną serverį ir perkonfigūruoti, kad sekantį kartą krautųsi iš tinklo. Diegiant ~100 serverių vienu metu – sistema sutrumpina reikalingą laiką nuo keliasdešimt valandų iki kelių sekundžių.

Nors aukščiau pateikti punktai yra svarbūs, tačiau didžiausia šio sprendimo vertė yra žinių išsaugojimas ir panaudojimas. Naudojantis SiteManager (tai kuriamos sistemos, kuri bus atsakinga už OpenStack diegimą pavadinimas. Šis pavadinimas čia įtrauktas, kad vėliau būtų aiškiau, apie ką kalbama) sistema OpenStack diegimą gali atlikti kiekvienas, turintis bent minimalias žinias apie kompiuterių tinklus bei OpenStack komponentus, todėl įmonėje atsiranda galimybė OpenStack diegti ne porai didžiausius įgūdžius turinčius darbuotojus, o kelioms dešimtis technikų, kurie turi tik paviršutiniškas žinias apie OpenStack komponentus.

1.5. Darbo struktūra

Šis darbas susideda iš skyrių, tokių kaip analizė, kurios metu analizuojami panašūs produktai, siekiant pastebėti reikalingiausias funkcijas bei pagrindinius jų trūkumus. Taip pat duomenų perdavimo strategijos pasirinkimas, programavimo kalbos pasirinkimas, įvairių papildomų bibliotekų paieška, kurios padėtų pasiekti tikslą. Toliau, sprendimo specifikacijos skyriuje aptariama, kokios funkcijos turėtų įeiti į šias sistemas, nustatoma, kas įeina į šios sistemos kūrimą, taip nustatant ribas, iki kurių galima plėsti kuriamą sistemą pirmojoje debesies fazėje.

Šiame darbe taip pat pateikiamas sistemos projektas, kuriame grafiškai (veiklos diagramomis) pavaizduojami sistemoje vykstantys procesai, tokie kaip automatinė fizinių serverių paieška tinkle, operacinės sistemos atvaizdo priskyrimas ir kitos pagrindinės funkcijos. Projektavimo stadijoje taip pat pavaizduojama duomenų bazės schema ir lentelių aprašymai. Darbe taip pat parodomos sukurtos vartotojo sąsajos ir jų aprašymai.

Nors pateikiama sąsajos ir duomenų bazės aprašymai, tačiau šio darbo tikslas nėra atskleisti programiniame kode priimtų sprendimų, visa šių sprendimų logika atsispindi veiklos diagramose.

Testavimo skyriuje pateikiama medžiaga, kaip buvo testuojamas šis sprendimas. Kadangi yra dideli skirtumai tarp sistemos komponentų – nebuvo tikslinga visiem panaudos atvejams sukurti automatinius testus ir buvo pasitelktas rankinis naudotojų testavimas.

2. PROBLEMINĖS SRITIES ANALIZĖ

2.1. Analizės tikslas

Šiame skyriuje bus analizuojami panašūs produktai, jų pagrindinės teigiamos bei neigiamos savybės, taip siekiant išsiaiškinti norimo produkto savybes bei sukurti reikalavimus kuriamai sistemai. Nors kai kurie toliau pateikti įrankiai tiesiogiai negali atlikti visų reikalaujamų veiksmų – tačiau jie gali būti naudingi dėl savo kai kurių sprendimų (pavyzdžiui serverių spintų atvaizdavimo).

2.2. Tyrimo objektas, sritis ir problema

Objektas – sistemos, skirtos paruošti fizinius serverius OpenStack sistemos įdiegimui ir po to debesies priežiūrai rinkti reikalingus parametrus ir pats OpenStack sistemos diegimas (norint sukurti šią automatizavimo sistemą būtina įsigilinti į kai kuriuos diegimo metu vykdomus procesus). Diegimui reikalingi parametrai dažniausiai svarbūs ne tik palaikančiam personalui, tačiau ir verslo analitikams, kurie turi įvertinti minimalius debesies išnaudojimo resursus, ties kuriais yra pelninga turėti įmonėje šią sistemą ir kuomet reikia apjungti resursus su išoriniai debesų kompiuterijos tiekėjais (kurti hibridinį debesį).

Sritis – debesų kompiuterijai skirti sprendimai, apimantys tiek pradinį serverių konfigūravimą, tiek vėliau ateinančias debesies priežiūrai reikalingas funkcijas. Ši sritis yra nepaprastai plati, nes nėra vieno teisingo kelio atlikti diegimą arba vienareikšmiškai nuspręsti, jog pasirinkti komponentai yra geriausi. Šio darbo sritis apima tik būtiniausius komponentus tikslui pasiekti, tačiau nesigilinama į kitas, labiau specializuotas sritis, tokias kaip ELK (*ElasticSearch, LogStash, Kibana*) resursų stebėjimui arba *Chef* pačiai instaliacijai atlikti. Teoriškai, kuriama sistema gali būti bet kada išplečiama, įtraukiant aukščiau paminėtus įrankius ir perduodant arba paimant iš jų informaciją.

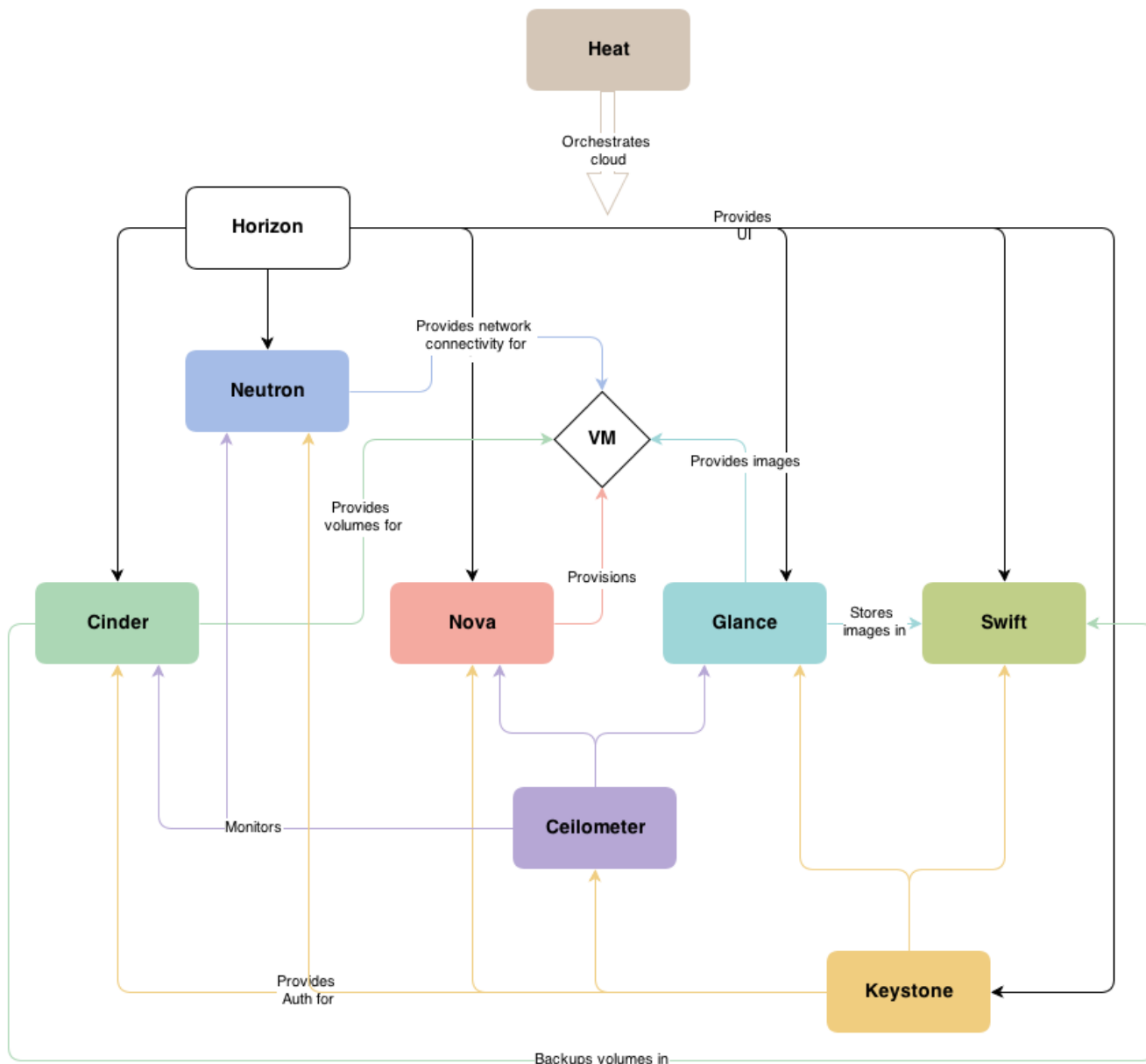
Yra kelios pagrindinės problemos, su kuriomis susiduriama. Pirmoji - šiuo metu daug įmonių skelbiasi apie jų naujus ir neįtikėtinus debesų sprendimus, tačiau realybė yra tokia, kad dauguma iš jų nesupranta, ką daro ir ką iš tikro reiškia debesų kompiuterijos paslaugos, tokios kaip IaaS ir PaaS. Jų sprendimai nėra pritaikyti dideliems poreikiams, jie sunkiai integruojami į jau egzistuojančią korporacijos infrastruktūrą. Nors ši technologija ir neturėtų taikytis prie senų standartų, tačiau taikytis turėtų įrankiai, kuriais po to turi naudotis šią sistemą palaikantys administratoriai ir paprasti vartotojai.

Kita problema yra žinių išsaugojimas. Ši problema OpenStack diegimo procese labai svarbi, nes konfigūruojant skirtingus komponentus ir jų sąveiką tarpusavyje – galima padaryti klaidų, kurios surasti ir ištaisyti kainuoja itin daug laiko. Taip pat, norint tobulinti procesą – svarbu jį turėti aprašytą (instaliacijos algoritmas tampa vienas iš geriausių variantų).

Taip pat prie problemų galima priskirti OpenStack sistemos būklės stebėjimą. Pati OpenStack sistema pateikia savo servisų būklę per API, tačiau standartinėje Horizon sąsajoje šie duomenys nerodomi. Trečiųjų šalių įrankiai, tokie kaip *BMC TrueSight* suteikia daugiau informacijos apie OpenStack aplinką, tačiau jų kaina didžiulė, o reali vertė – maža. Šiuos duomenis nėra sudėtinga pasiekti ir kaupti panaudojant vos kelias Rest tipo užklausas į Nova komponentą, o tuomet išsaugoti duomenų bazėje arba perduoti į centralizuotas ir strategines įmonės aliarmų ir įvykių stebėjimo sistemas, pvz. per *rsyslog*.

2.2.1. Sprendimų kūrimas OpenStack sistemai

Norint kurti sprendimus OpenStack, reikia susipažinti su šios sistemos koncepcine architektūra, kuri pavaizduota paveikslėlyje žemiau.



2. pav. OpenStack sistemos koncepcinė architektūra [6]

Iš šios diagramos matome, kad ši sistema susideda iš skirtingų komponentų, kurie tarpusavyje bendrauja per tam tikrus API kreipinius. Tokia architektūra leidžia ne tik laisviau tobulinti ir keisti pavienius komponentus nesudarant problemų kitiems komponentams, tačiau suteikia galimybę išviso pašalinti kai kurias OpenStack dalis ir vietoj jų įdėti savo sukurtus komponentus (jei to reikia). Tuomet, savo komponente tereikia atkartoti originalaus komponento API protokolus ir visa sistema net neaptiks, kad kažkas pasikeitė. Tokiu būdu OpenStack sistema galima pritaikyti praktiškai bet kokioms reikmėms, pavyzdžiui, vietoj įprastinio Neutron komponento, kuris valdo tinklo nustatymus virtualioms mašinoms (kuria L2, L3 tinklus) galima įdiegti trečiosios šalies sprendimus, kurie naudoja SDN kartos tinklus.

Tačiau, su dinamiškumu bei lankstumu atsiranda kita problema – neapibrėžtumas, kai galima turėti neribotus kiekius įvairiausių komponentų ir jų konfigūracijų. Norint, kad OpenStack veiktų – visi komponentai turi žinoti, kaip rasti vienas kitą. Šiuo atveju visu tuo turėtų rūpintis Keystone. Ši OpenStack posistemė užtikrina vartotojų autentifikaciją ir autorizaciją, taip pat savyje laiko informaciją apie galinius taškus (*angl. k. endpoints*), per kuriuos pasiekiamas bet kuris kitas komponentas. Tai taškas, per kurį vienas komponentas randa kitą, pvz. Nova kreipiasi į Glance, kai kuriama nauja virtuali mašina ir reikia operacinės sistemos atvaizdo. Tačiau, kaip visa ši informacija atsiranda Keystone? – Tai yra konfigūracijų pasekmė, todėl svarbu diegiant ir konfigūruojant komponentus nesupainioti IP adresų, o tai tampa sudėtingas uždavinys diegiant kelis šimtus serverių.

Taip pat, kuriant debesų kompiuterijos sprendimą korporacijoje, yra atsižvelgiama ir į kitas, išorines sistemas, kurios taip pat turi savo galinius taškus, per kuriuos išduoda arba priimą informaciją, taigi galimų konfigūracijų dar labiau padaugėja iki tokio lygio, kai tai padaryti rankinių būdu tampa praktiškai neįgyvendinama užduotis.

2.2.2. Galimi problemos sprendimo būdai

Žinių išsaugojimo problema sukuria puikų pretekstą automatizuoti kuo didesnę vykdomo proceso dalį. Tai leidžia ne tik inžinieriams kurti naujus, inovatyvius produktus, tačiau ir užtikrinti kokybės kontrolę (vykdyti automatizuotus testus). Taigi, problemos sprendimo būdas – kurti automatizuojantį įrankį (paverčiant itin sudėtingą diegimą ir konfigūravimą uždaviniu, kurį gali įveikti kiekvienas, bent šiek tiek su sistema susipažinęs technikas), kuris valdytų fizinius serverius, galėtų prisijungti prie OpenStack komponentų per API ir konsoliduotų daugelio aplinkų informaciją į centrinę sistemą, iš kurios būtų galima stebėti debesies būklę bei paleisti integracinius testus. Sprendimas taip pat turėtų apimti ir kitą problemą – OpenStack būklės stebėjimą, o šis uždavinys šiuo metu yra itin aktualus, dėl sudėtingos sistemos architektūros, kai nebeužtenka stebėti, ar veikia serveriai, o daug svarbiau tampa individualių servisų stebėjimas ir jų tarpusavio sąveika [7]. Kadangi diegimo metu ir po jo reikalinga galimybė stebėti debesies būklę – tiek automatizavimo, tiek ir stebėjimo funkcijos turėtų būti pasiekiamos per tą pačią sąsają.

Šiam sprendimui įgyvendinti, detalesni žingsniai būtų:

- Išanalizuoti galimybės serverių paruošimui OpenStack sistemai (valdyti juos per IPMI sąsają bei priskirti jiems operacinės sistemos atvaizdus nuotoliniu būdu).
- Išanalizuoti metodus, kurie leistų realiu laiku stebėti kompiuterių debesies parametrus, tokius kaip: sukurtų virtualių mašinų skaičius ir jų detalės, turimų resursų kiekis ir jų utilizacija (hipervizorių informacija), serverių grupių informacija ir jų valdymas.
- Suprojektuoti sąsają, kuri leistų matyti agreguotą informaciją apie debesies būklę iš daugelio duomenų centrų, galimybė matyti konkrečių servisų būklę ir kuriuose serveriuose jie yra.
- Sukurti šią informaciją teikiančią sistemą, pasiekiamą per įprastą internetinę naršyklę, kurios pagrindinį puslapį būtų galima naudoti dideliame ekrane.

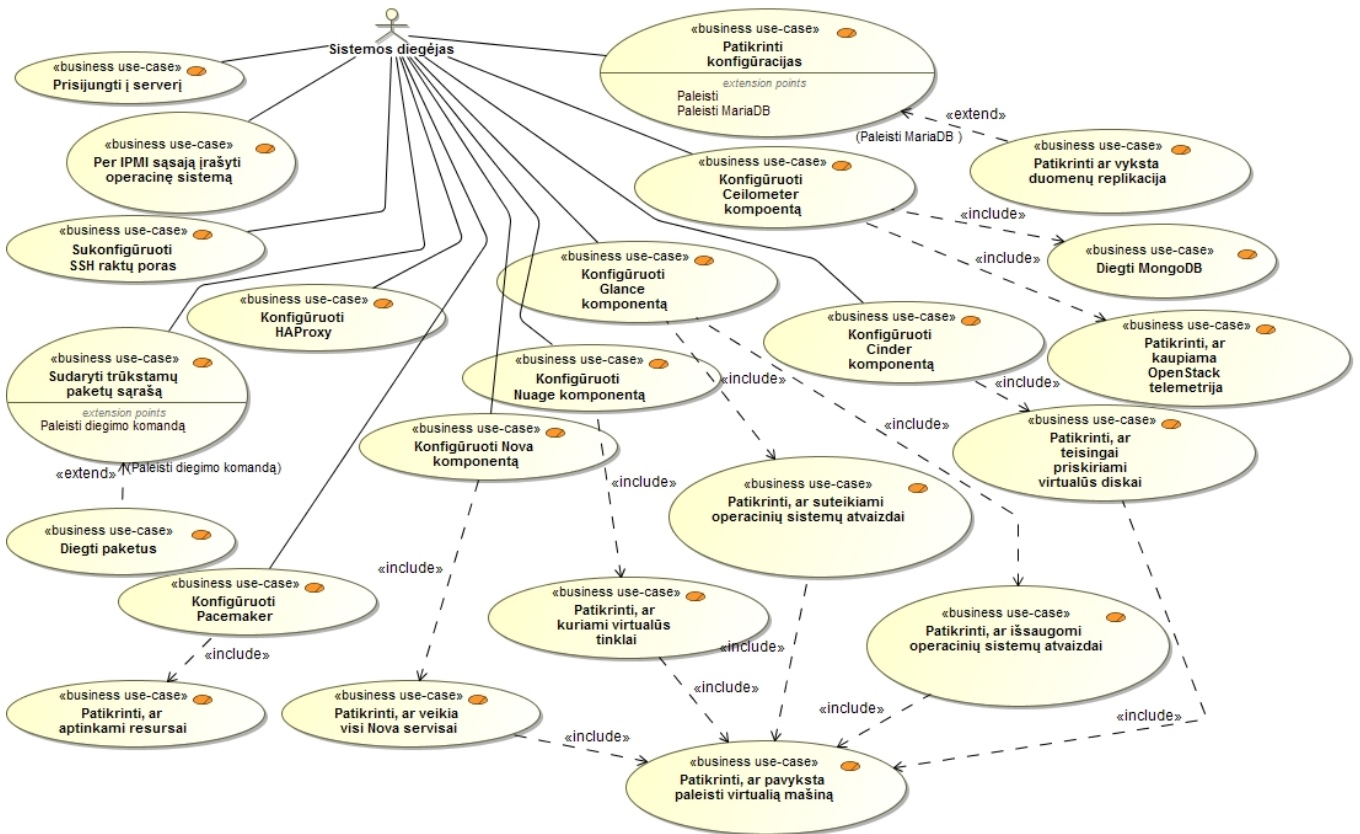
Tikslesni veiksmai, kuriuos reiktų atlikti yra:

- Sukurti dvi sistemas, duomenų centro ribose veikti skirtinga sistema – SiteManager ir jos veiklai priskirtos užduotys:
 - Įvertinti būdus serverių valdymui per IPMI sąsajas (gamintojas HP), tačiau žvelgiant į ateitį reikia pasilikti galimybes valdyti ir kitus serverius, pvz. Dell, taigi serverio valdymo modulis turėtų turėti prieš save kitą modulį, kuris nustato serverio gamintoją ir atiduoda reikiamą klasę.
 - Panaudojant HP serverių spintų funkcijas, kuriomis jie gali nustatyti savo buvimo vietą šasi, dinamiškai atvaizduoti naudotojo ekrane visą serverių spintą.
 - Tai pat pirmojoje sistemoje turi būti modulis skirtas stebėti OpenStack sistemą per jos API realiu laiku, suteikiant galimybę šią sistemą prižiūrintiems administratoriams visus reikalingus duomenis.
 - Prieš paleidžiant OpenStack sistemos įrašymą į serverius – galimybė nusiųsti papildomus metaduomenis į diegimo programas.
- Antroji sistema, kurios pavadinimas CloudMap ir jai priskiriami pagrindiniai uždaviniai:
 - Sukurti mechanizmą, kaip bus renkami duomenys iš duomenų centruose esančių SiteManager sistemų.

- Sukurti duomenų atvaizdavimą pasaulio žemėlapyje, kur būtų sužymėti duomenų centrai ir juose veikiančys servais.
- Sukurti modulį, skirtą OpenStack sistemos testavimui – pagrindiniai testai yra tikrinimas, ar API atsako ir sudėtingesnis, paliečiantis praktiškai visus OpenStack sistemos komponentus – virtualios mašinos sukūrimas ir ištrynimasis.

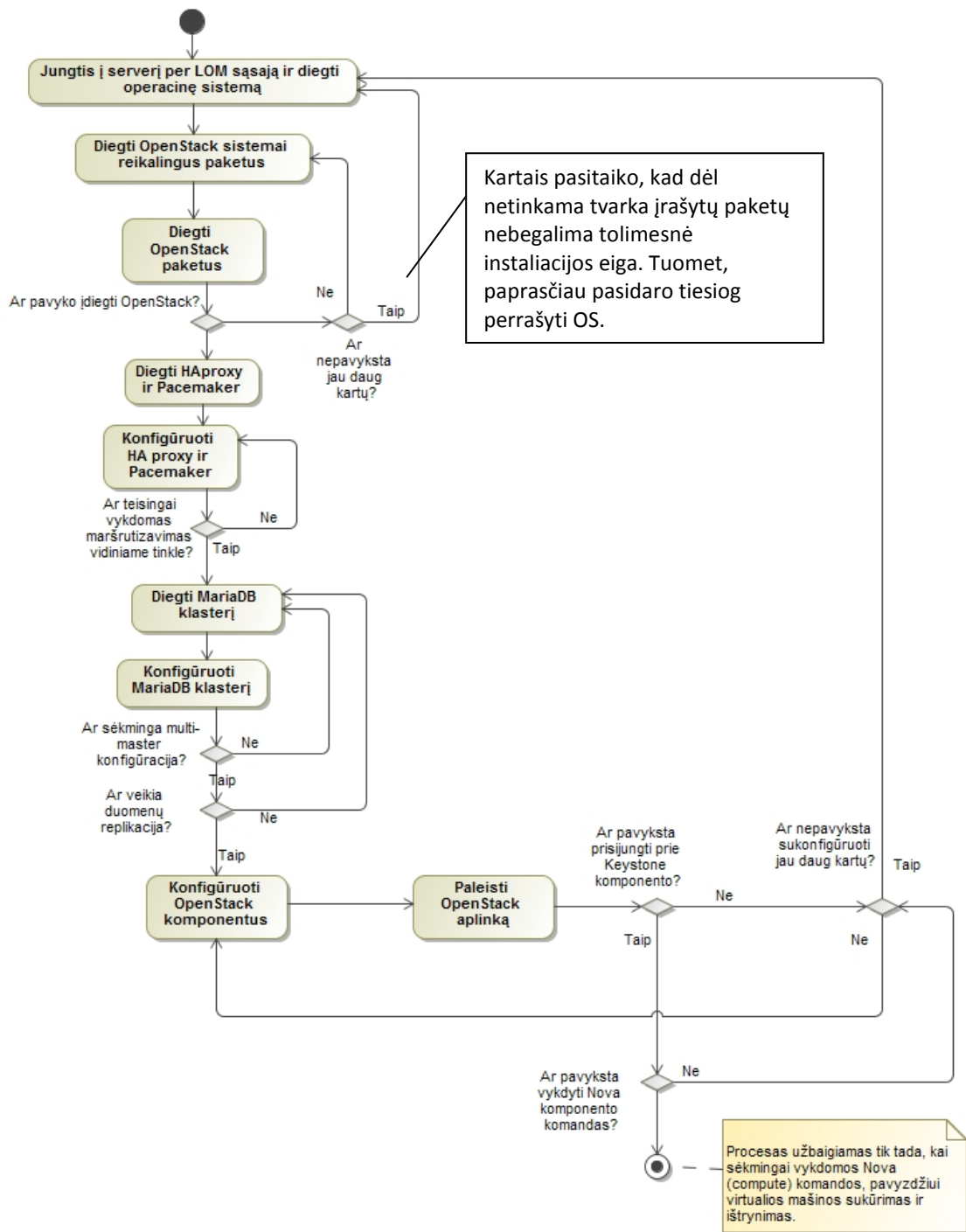
2.3. OpenStack komponentų diegimo (veiklos proceso) analizė

Šiame skyriuje analizuojamas OpenStack sistemos veiklos procesas. Šiam procesui atvaizduoti panaudojama panaudos atvejų diagrama bei veiklos diagrama, kurioje galime matyti pagrindinius veiksmus, kuriuos atlieka inžinierius, diegdamas ir konfigūruodamas OpenStack sistemą viename serveryje. Minimaliai, produkcijai skirtai OpenStack aplinkai yra reikalingi bent 3 valdymo serveriai, taigi šį procesą turėtų kartoti tris kartus, tačiau realybėje dar reiktų pridėti kelis šimtus skaičiavimo serverių, kurių diegimo procesas yra gana panašus. Paveikslėlyje žemiau pateikiama panaudos atvejų diagrama su veiksmais, kuriuose atlieka sistemos diegėjas, norėdamas sukurti vienkiančią OpenStack sistemą:



3. pav. OpenStack diegimo panaudos atvejų diagrama

Kuriama sistema turėtų automatizuoti šį procesą, sumažindama reikalingą darbą iki tiek, jog reiktų tik rankiniu būdu įvesti norimą sistemos konfigūraciją, nes priklausomai nuo fizinės serverių vietos bei kintančių reikalavimų (pvz. kompiuterių tinkluose išskirtų IP adresų režių ir potinklų negalima iš anksto numatyti keliems mėnesiams ar metams į priekį). Taigi, žemiau paveikslėlyje pateikta rankiniu būdu atliekamo OpenStack įrašymo veiklos diagrama:



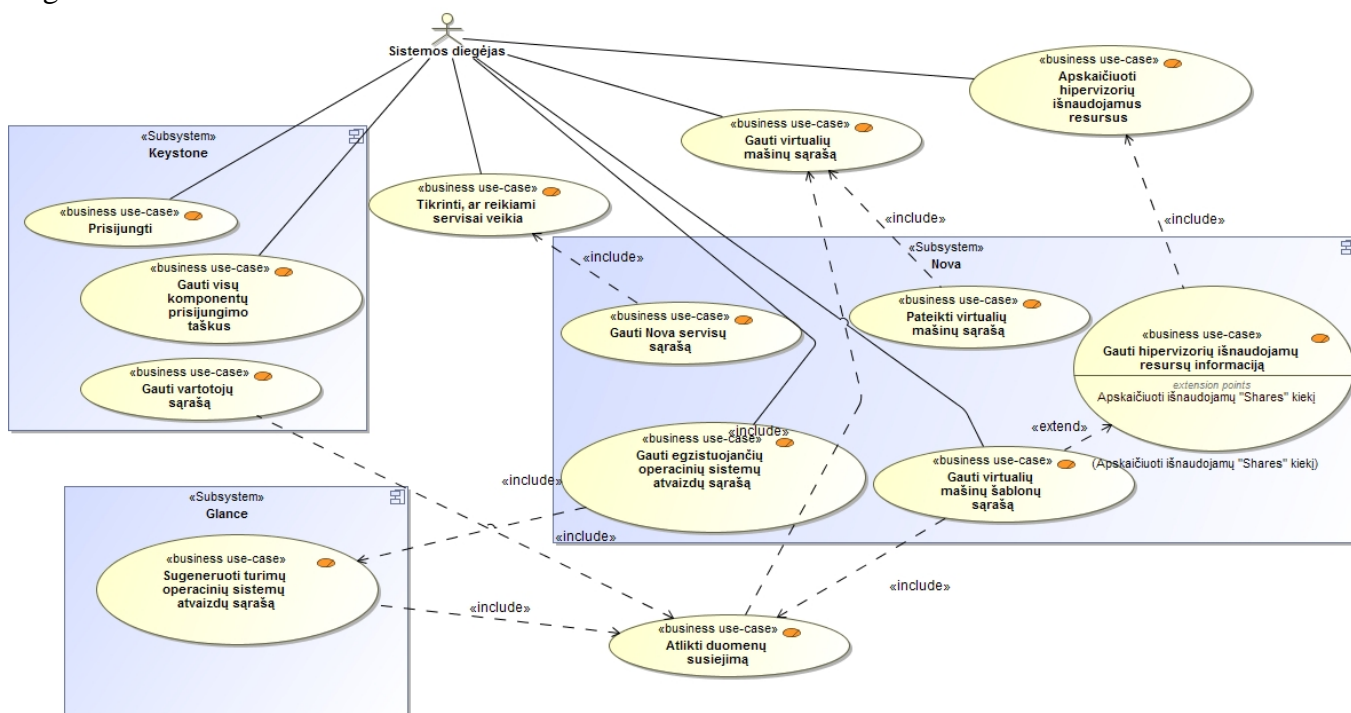
4. pav. Rankiniu būdu atliekamas OpenStack diegimas ir konfigūravimas

Iš šios diagramos matome, kad gana dažnai gali tecti pradėti viską iš naujo dėl itin daug reikalingų paketų, kurie gali pažeisti vienas kitą, pavyzdžiui kai kurie paketai dirba tik su griežtai nustatytos versijos kitais paketais ir naujesnė jų versija gali sugadinti jau esančius paketus ir t.t. Labai sudėtinga prisiminti ir dokumentuoti visus tokius neatitikimus. Dažnai net nebegali žinot, kodėl pavyko arba nepavyko sudiegti tam tikrus komponentus. Šis darbas yra lėtas ir erziantis, kurį pakartoti kelis šimtus kartų gali ir nori tik labai mažas kiekis žmonių.

2.4. Debesų kompiuterijos sprendimo priežiūros (veiklos proceso) analizė

Standartinė OpenStack sistemos architektūra susideda iš valdymo ir skaičiavimo serverių, kurie visi yra apjungti tarpusavyje ir veikia kaip viena sistema. Turint standartinį sprendimą – nėra labai sudėtinga stebėti jo būklę pasitelkiant Horizon komponentą, nors jis ir nėra pilnai tinkamas produkciniam naudojimui, nes neturi autorizuotų grupių kūrimo funkcionalumo, taigi, jei

vartotojams suteikiama prieiga stebėti išnaudojamus resursus – tuomet negalima uždrausti kurti ir naikinti virtualių mašinų. Taip pat tokie vartotojai galėtų prisijungti prie klientų virtualių mašinų per VNC konsolę, o panaudoję „rescue mode“ – galėtų ir patekti prie privačių duomenų. Kitas iššūkis – Horizon internetinė grafinė sąsaja neleidžia stebėti konkrečių OpenStack servisų būklės – tik atlikus tam tikras komandas (pavyzdžiui pareikalavus parodyti visus Glance virtualių mašinų šablonus) ir jom neįsivykčius – galima suprasti, kad yra neveikiančių komponentų. Kiti patikrinimai tampa dar sudėtingesni, pavyzdžiui servisų, kurie užtikrina, jog yra galimybė prisijungti prie virtualių mašinų per VNC konsolę neveikimas užfiksuojamas tik pabandžius tai atlikti, o šie veiksmai yra nerekomenduojami, jei virtuali mašina veikia. Šiuo atveju būtų galima susikurti testavimui skirtą virtualią mašiną ir per ją atlikti specifinius veiksmus, siekiant išsiaiškinti, servisų būklę. Kitas būdas – naudoti standartinius OpenStack API ir, pavyzdžiui per naršyklę su RestAPI papildiniais, tokiais kaip „Postman“ arba „Advanced REST Client“, atlikti užklausas į Keystone ir Nova komponentus. Pasitelkiant šablonus, tai būtų galima daryti gana nesudėtingai, tačiau užtruktų vis tiek daug laiko ir pareikalautų žmogiškųjų resursų, kurie tai darytų ir dokumentuotų. Šie sprendimai yra vienkartiniai, nepritaikyti daryti nuolat ir nesukuria perspėjimo sistemos, o dažniau gali tik patvirtinti vieną ar kitą faktą, kad sistema veikia arba neveikia. Sekančiame paveikslėlyje yra pavaizduojami pagrindiniai veiksmai, kuriuos atliekant galima patikrinti, ar OpenStack sistema veikia. Yra ir daugiau reikalingų veiksmų, tokių kaip RabbitMQ žinučių persiuntimo karkaso tikrinimas, tačiau čia specifiškai tikrinami tik OpenStack sistemos komponentai ir jų servais. Šiuos veiksmus gali atlikti tiek sistemos diegėjai, tiek ir palaikymo personalas. Žemiau pateikta sistemos priežiūros panaudos atveju diagrama:



5. pav. OpenStack minimalios priežiūros panaudojimo atveju diagrama

Iš diagramos matome, kad norint stebėti OpenStack būklę, neužtenka kreiptis į vieną komponentą, o reikia kreiptis iškart į kelis ir gautus duomenis apjungti. Kaip pavyzdį galima pateikti virtualių mašinų sąrašo peržiūrėjimą - norint gauti informaciją iš duomenų, reikia atlikti šiuos veiksmus:

1. **Gauti virtualių mašinų sąrašą** – šis sąrašas žmogui yra sudėtingai perskaitomas, nes vietoj hipervizorių pavadinimų, virtualių mašinų šablonų, vartotojų, kurie sukūrė jas yra pateikiami atitinkamos UUID reikšmės, kurios yra užkoduoti išoriniai raktai.
2. **Gauti virtualių mašinų šablonų sąrašą** – gavus šį sąrašą, su papildomom taikomosiomis programomis (pvz. Excel VLOOKUP funkcija) arba Python skriptais

galima susieti virtualių mašinų sąrašo UUID laukus su šablonų laukais ir gauti panaudoto šablono pavadinimą.

3. **Gauti vartotojų sąrašą** – taip pat kaip ir su šablonų sąrašu, vartotojai nėra iškart pateikti prie virtualių mašinų, o yra tik jų identifikaciniai numeriai, taigi nėra galimybių nustatyti, kas sukūrė tam tikrą virtualią mašiną prieš tai neapjungus šių sąrašų. Procedūra tokia pati – abu sąrašai su papildomomis taikomosiomis programomis apjungiami pagal UUID laukus.
4. **Gauti operacinių sistemų atvaizdų sąrašą** – šis sąrašas reikalingas tam, kad būtų galima žinoti, kokia operacinė sistema buvo panaudota kuriant virtualią mašiną. Be šio sąrašo apjungimo su virtualių mašinų sąrašu – nebūtų galimybės atskirti Windows nuo Linux, bei įvairių kitų *nix distribucijų. Apjungiami taip pat UUID laukai.

Visus šiuos žingsnius reikėtų kartoti kiekvieną kartą, kai norima gauti virtualių mašinų sąrašą. Tačiau sudėtingiau tampa su servisų patikrinimu. Šiuo atveju pirminis sąrašas gaunamas iš Keystone komponento – jame galima rasti pagrindinius komponentus, tokius kaip Keystone (jis pats), Glance, Nova, Cinder, Ceilometer, Swift. Šiame sąrašė ignoruojamas Neutron, nes jis reaguoja į kitas komandas ir yra konfigūruojamas bei prižiūrimas kitų komandų, tačiau jeigu būtų naudojamas standartinis Neutron, būtų galima prie jo prisijungti ir siųsti jam užklausas taip pat, kaip ir kitiems OpenStack komponentams. Tuomet, turint pirminį sąrašą, galima vieną po kito tikrinti jo elementus, pavyzdžiui, jeigu komponentas Glance – tuomet pabandoma gauti operacinių atvaizdų sąrašą, o jeigu Ceilometer – prašoma pateikti telemetrijos duomenis. Tačiau, tai tik valdymo serverių sudedamosios dalys ir jeigu aptinkamas Nova komponentas sąrašė – reikia prašyti jo turimų servisų sąrašo, o šis sąrašas gali būti kelių šimtų arba tūkstančių elementų ilgio. Kai kurie servais turi būti specialiai išjungti, pavyzdžiui „console-auth“ servisas turi veikti tik ant vieno valdymo serverio, o kitur turi būti išjungtas, taigi šiuos duomenis reikia papildomai apdoroti. Iš aukščiau pateiktos panaudos atvejų diagramos galime matyti, jog neautomatizuotas resursų ir servisų stebėjimas yra praktiškai neįveikiama užduotis, o jeigu bandyti tai spręsti didesniu darbuotojų skaičiumi – būtų itin neefektyvu.

2.5. Tyrimo objekto naudotojų analizė

Yra trys pagrindinės naudotojų grupės, kurių nariai naudosis šiomis sistemomis:

- Diegėjai – ši naudotojų grupė atsakinga už pradinį SiteManager sistemos diegimą į valdymo serverius, o tada su šios sistemos pagalba galėtų diegti kitus serverius su OpenStack sistema. Ši naudotojų grupė labiausiai suinteresuota šiomis funkcijomis:
 - Fizinų serverių paieška ir valdymas.
 - OpenStack sistemos diegimo metu perduodamais parametrais ir operacinės sistemos įrašymo metu vykstančių procesų atvaizdavimu SiteManager sistemoje. Ši grupė, tikėtina, kad nesinaudos CloudMap sistema, nebent tik tam, kad surastų jiems reikalingą SiteManager ir jo prisijungimo duomenis.
- Analitikai – ši naudotojų grupė yra atsakinga už išnaudojamų resursų stebėjimą pagal klientus arba biznio šakas. Vienas iš jų tikslų – nustatyti minimalų debesų kompiuterijos sprendimo išnaudojimą, kad jis būtų pelningas ar bent jau atsipirktų. Šia naudotojų grupę labiausiai domina:
 - SiteManager OpenStack išnaudojamų resursų langas, kur jie gali eksportuoti duomenis į CSV formatą ir tada jau analizuoti reikiama pjūviais.
 - CloudMap agreguotų duomenų vaizdas, kur matomos visos OpenStack aplinkos ir matomi visi turimi resursai per visus duomenų centrus.
- Palaikymo komanda – ši grupė atsakinga jau už įdiegtą ir paruoštą debesų kompiuterijos sprendimo palaikymą. Jie atsakingi už tai, kad būtų galima kurti

virtualias mašinas ir naudoti kitas OpenStack funkcijas. Jie taip pat atsakingi, kai, pavyzdžiui, vartotojas nebegali prisijungti prie savo virtualios mašinos, tuomet jie turėtų naudoti SiteManager, kad surastų šią virtualią mašiną ir su ja susijusius duomenis. Jiems taip pat būtų svarbi CloudMap sistema, tuomet jie galėtų matyti, kiek ir kokių servisų yra sutrikę globaliai.

2.6. Esamų problemos sprendimo metodų analizė

Debesų kompiuterijos pasaulyje, viena iš pagrindinių idėjų yra „automatizuoti viską“. Nieko keista, kai augant OpenStack sistemos naudotojui atsiranda vis daugiau automatizavimo projektų. Tačiau šių sprendimų kiekis yra išpūdingas – jų daugiau nei dvidešimt. Šiame skyriuje apžvelgsime keletą tokių projektų bei kitų sistemų, kurios atlieka tam tikrą dalį užsibrėžtų uždavinių.

Kai kurie toliau paminėti projektai, pavyzdžiui „Compass“ gali automatizuoti ne tik OpenStack diegimą, tačiau ir kitus, įmonei reikalingus, veiksmus.

2.6.1. Mirantis Fuel

Šis produktas buvo komercinis, o vėliau tapo atviro kodo produktu ir šios analizės metu – geriausiai išvystytas automatizavimo įrankis. Tačiau jis patiekiamas kartu su OpenStack distribucija ir nėra integruotas į OpenStack, o tik susijęs su juo. Taigi jis niekada nenaudoja naujausios OpenStack versijos, o šiek tiek modifikuotą. Pagrindinės funkcijos pateiktos toliau.

2.6.1.1. Automatinė fizinių ir virtualių serverių paieška tinkle

Ši funkcija suranda serverius, kuriuose numatyta krautis iš tinklo. Kai jie aptinkami – Fuel įtraukia juos į sąrašą ir sistemos administratorius gali juos naudoti, priskirti jiems roles debesyje. Fuel taip pat atlieka ir kompiuterių tinklo patikrinimus, nors ši funkcija šiame projekte nėra svarbi, nes bus naudojamas programinės įrangos būdu kuriamas tinklas (angl. Software Defined Network). Šis tinklas bus atitinkamai konfigūruojamas pagal virtualios mašinos tipą – vienos galės pasiekti produkcinę tinklą (*prod*), o kitos – sistemų kūrimui ir testavimui skirtą tinklą (*dev*). Vienintelis privalumas – prieš paleidžiant instaliaciją patikrinti tinkle esančius konfliktus ir juos pateikti sistemos administratoriui, kad būtų greičiau įdiegta sistema.

Ši funkcija tikrindama tinklą ir aptikusi serverius, parodo tai vartotojui tokiu formatu, kaip pateikta paveikslėlyje žemiau.

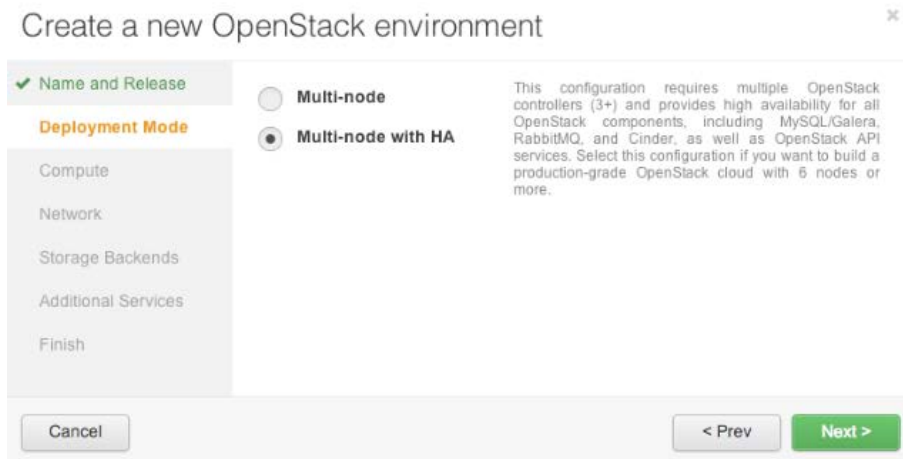


6. pav. Mirantis Fuel aptikęs serverius

Trūkumas – tinklo skenavimas yra agresyvus būdas ir sistemos administratorius visada turėtų perspėti atsakingą komandą apie planuojamą tinklo tikrinimą. Už saugumą atsakinga komanda turbūt neleistų jai veikti, todėl tektų ją išjungti.

2.6.1.2. Pagalba tinkamam konfigūracijos pasirinkimui

Tai papildomi aprašymai ir iš anksto numatyti scenarijai OpenStack sistemos diegimui. Diegimo pradžioje yra keli variantai, ar ši aplinka turės būti aukšto patikimumo, ar ne. Paveikslėlis žemiau.



7. pav. Mirantis Fuel – OpenStack aplinkos diegimo pradžia, aukšto patikimumo konfigūracijos pasirinkimas

Šiek tiek apie HA – tai *High Availability* trumpinys, reiškiantis šiek tiek kitokią konfigūraciją nei įprasta. Tai gerokai sudėtingesnė OpenStack konfigūracija, kai reikalingi trys arba daugiau valdantieji serveriai, kurie turi vienodus jiems priklausančius servisus. Tai reiškia, kad išsijungus net 2 serveriams – OpenStack sistema veiks be sutrikimų. Naudojant MariaDB Galera – visi duomenys bus prieinami iš vienintelio likusio serverio, o atsistačius likusiems – jie automatiškai sinchronizuos ir atsinaujins savo duomenų bazės lenteles.

Yra ir kitų konfigūracijų, ne tik čia nurodytos, tai būtų, pavyzdžiui, - *All in One* konfigūracija, kai visi reikalingi servisai veikia viename serveryje. Šioje konfigūracijoje vartotojas jungiasi į vieną serverį ir jame veikiančią OpenStack Horizon komponentą ir per jį kuria virtualias mašinas, o šios virtualios mašinos yra sukuriamos tame pačiame serveryje. Jeigu kyla klausimų, kam reikalinga tokia konfigūracija ir ar tai naudojama praktikoje – reiktų pabrėžti, kad dabartiniai serveriai gali pas save turėti maždaug 140-190 virtualių mašinų, o su šia konfigūracija nereikia visiškai spręsti jokių bėdų susijusių su tinklais.

Taip pat Fuel siūlo rinktis serverio operacinę sistemą (fizinio, į kurią bus įrašomas OpenStack), hipervizorių programinę įrangą, tinklo topologiją. Taip pat kita funkcija, kuri gali pasirodyti naudinga – šablonų kūrimas, kurie galėtų pagreitinti naujų OpenStack aplinkų diegimą.

Kita funkcija, kuri galėtų būti naudinga – galimybė rinktis įvairias OpenStack versijas. Kadangi ši sistema yra labai greitai plėtojama, reikia ją versijuoti (diegti atitinkamus komponentus pagal kitų komponentų versijas). Tačiau mūsų įmonės atžvilgiu ši funkcija nėra reikalinga, nes bus diegiama viena, patvirtinta versija, jau perėjusi daug testavimo fazių ir saugumo patikrinimų. Versijų paveikslėlis žemiau.

Releases		
OpenStack Release	Version	Status
Icehouse on CentOS 6.5	2014.1-5.1	Active
Icehouse on Ubuntu 12.04.4	2014.1-5.1	Active

8. pav. Galimybė rinktis įvairias OpenStack versijas Mirantis Fuel sistemoje

2.6.1.3. Siųsti komandas į OpenStack ir vykdyti operacijas debesyje

Mirantis turi galimybę keisti OpenStack konfigūraciją jau po įdiegimo ir paleidimo:

- Pridėti papildomus skaičiavimo serverius
- Priskirti daugiau rolių bet kuriam serveriui (įrašyti papildomus servisus, pavyzdžiui valdymo serveriuose įdiegti Ceilometer komponentą debesies parametrų rinkimui)
- Pašalinti serverį iš debesies

- Konfigūruoti fizinius diskus iškart daugelyje serverių
- Rinkti informaciją iš daugelio serverių apie jų būklę

2.6.1.4. Rinkti informaciją realiu laiku iš OpenStack aplinkos

Ši funkcija leidžia vykdyti automatinius testus įvairiems komponentams, pavyzdžiui tikrinimas, ar duomenys replikuojasi duomenų bazėse, ar žinučių perdavimo servisas veikia (RabbitMQ) ir kiti parametrai, paveikslėlis žemiau.

OpenStack Health Check				
		Expected Duration	Actual Duration	Status
<input checked="" type="checkbox"/> HA tests. Duration 30 sec - 8 min				
Check data replication over mysql	1-40 s.	20.4 s.	👍	
Check amount of tables in os databases is the same on each node	1-40 s.	19.3 s.	👍	
RabbitMQ availability	100 s.	—	🕒	
RabbitMQ queues availability	100 s.	—	🕒	
RabbitMQ messages availability	100 s.	—	🕒	
<input checked="" type="checkbox"/> Sanity tests. Duration 30 sec - 2 min				
Flavor list availability	20 s.	0.4 s.	👍	
Images list availability	20 s.	0.3 s.	👍	
Instance list availability	20 s.	0.4 s.	👍	

9. pav. Fuel - sistemos automatiniai testai

Tai gana naudinga informacija, tačiau, jeigu Mirantis yra įrašytas viename iš valdančiųjų serverių – yra natūralu, jog API užklausų atsakymo greitis yra didžiulis. Dėl kitų parametrų – jie naudingi tik instaliacijos metu, nes vėliau šią informaciją turėtų globaliai rinkti tam tikri agentai, kurie šiuos duomenis perduotų į tam skirtą bendrą sistemą. Kitas aspektas, į kurį reiktų atkreipti dėmesį – nėra OpenStack integracinių testų, kurie bandytų kurti virtualias mašinas arba įkelti operacinių sistemų atvaizdus į Glance komponentą.

2.6.1.5. Mirantis Fuel apžvalgos apibendrinimas

Taigi, šios sistemos privalumai:

- Galimybė rinkti ir matyti turimus fizinius ir virtualius resursus.
- Galimybė pasirinkti įvairias OpenStack konfigūracijas prieš instaliaciją.

Trūkumai:

- Automatinis tinklo tikrinimas gali sukelti saugumo problemų.
- Daugelis funkcijų, kurios pateikiamos vartotojui – mūsų įmonės atveju bus nereikalingos, nes nebus diegiamas nei duomenų saugojimo komponentas (Swift), nei tinklo komponentas (Neutron), nei skirtingos OpenStack versijos. Taip pat tokios funkcijos kaip fizinio serverio operacinės sistemos pasirinkimas irgi nebus reikalingas, nes tam iš anksto yra ruošama itin stabilus RedHat operacinės sistemos variantas.

2.6.2. QuotaViewer

Ši sistema buvo sukurta stebėti pirmajai OpenStack sistemos versijai. Ji nenaudoja OpenStack API, yra parašyta PHP kalboje ir tiesiogiai jungiasi į OpenStack duomenų bazę. Kadangi tai yra tik duomenų atvaizdavimo įrankis, saugumo problemos jungiantis į duomenų bazę yra išsprendžiamos apribojant paskyros teises.

Pagrindinės QuotaViewer funkcijos:

- OpenStack sistemos išnaudojamų resursų informacija
- Fizinį serverių sąrašas ir papildomi duomenys (RAM, VCPU, virtualių mašinų kiekis juose)
- Virtualių mašinų sąrašas su jų duomenimis.
- Operacinių sistemų atvaizdų sąrašas ir jų parametrai

QuotaViewer privalumai – gana paprasta sistema, kuri palaikymo komandai atskleidžia daug duomenų apie virtualias mašinas, resursų išnaudojimą. Taip pat yra funkcijos, skirtos eksportuoti duomenis į CSV tipo failą.

Trūkumai:

- Jungiantis tiesiai į duomenų bazę gali kilti problemų dėl blogai parašyto programinio kodo, taip pat nėra galimybės išnaudoti OpenStack duomenų valdymo klasių, kurios apjungia kai kuriuos duomenis ir leidžia iš duomenų išgauti žinias.
- Nėra galimybės ieškoti fizinių serverių ir jiems priskirti operacinės sistemos atvaizdų.
- Nėra galimybės diegti OpenStack.
- Sistemoje daug kur trūksta paaiškinimų (jų tiesiog nėra), taigi gali būti sudėtinga šia sistema naudotis nepatyrusiam naudotojui.
- Informacijos apie virtualias mašinas yra pateikta per daug, galima jas rūšiuoti pagal turimą RAM, atminties kiekį ar priskirtus CPU, tačiau palaikymo komandai užtektų vien tik virtualios mašinos šablono, taigi perteklinė informacija taip pat gali sudaryti problemų.
- Nėra galimybės prisijungti į VNC konsolę, jei virtuali mašina neužsikrauna.
- Nėra galimybės agreguoti informacijos iš daug OpenStack aplinkų.

2.6.2.1. QuotaViewer apžvalgos apibendrinimas

Ši sistema šiame sąrašė pateikta dėl to, jog ji yra vis dar naudojama įmonėje OpenStack parametrų peržiūrai. Ši OpenStack versija yra senesnė, kuri bus pakeista į naujausią versiją ir įrankis nebetiks naudoti. Iš šio įrankio galima spręsti apie pagrindinius informacijos šaltinius, kurie domina OpenStack palaikymo komandą, nes dirbant su jais – galima detaliai išsiaiškinti norimos kurti sistemos funkcijas.

2.6.3. HP Systems Insight Manager (HP SIM)

HP SIM sistema skirta valdyti fizinius serverius per IPMI sąsają, o tiksliau – HP ILO sąsają. Kadangi įmonės pirmojoje fazėje numatyta, jog visi serveriai bus perkami iš HP – naudojantis šia sistema galima būtų išvengti šio funkcionalumo kūrimo ir tiesiog nusipirkti šį įrankį. Ji turi tokias pageidaujamas funkcijas kaip serverių spintų atvaizdavimas bei nuotolinis jų valdymas (pavyzdžiui, nustatyti po serverio perkrovimo krautis iš tinklo). Ši sistema nėra skirta dirbti su OpenStack, ji tiesiog gali valdyti serverius, kaupti jų informaciją bei automatizuoti šiek tiek jų veiksmų.

Sistemos trūkumai:

- Tai komercinis produktas, kurį reikia pirkti.
- Galima valdyti tik HP gamintojo serverius.

2.6.4. RackMonkey

Ši sistema, taip pat kaip ir HP SIM, nėra skirta OpenStack diegimui arba valdymui. RackMonkey skirta kaupti informaciją apie fizinius serverius ir praktiškai bet ką, ką galima įdėti į serverių spintas (pavyzdžiui komutatorius, maršrutizatorius). Sąsaja yra gana paprasta ir pateikta paveikslėlyje žemiau:

RM Racks Devices Apps Config Reports Help View Rack... Find

Device Search
Found 4 devices which match www.

Device	.dom	Rack	Room	Role	Hardware	OS	Serial	Asset	Customer	Age	Notes
www1		A1 [5]	TFM4 in THDO	Web Server	Dell PowerEdge 2850	Ubuntu 8.04	T4567	-	Road Runner	1.8	Test data
www2		A2 [5]	TFM4 in THDO	Web Server	Dell PowerEdge 2850	Ubuntu 8.04	T4568	-	Road Runner	1.8	Test data
www3		A1 [7]	TFM4 in THDO	Web Server	Dell PowerEdge 2850	Ubuntu 8.04	T4569	-	Road Runner	1.8	Test data
www4		A2 [7]	TFM4 in THDO	Web Server	Dell PowerEdge 2850	Ubuntu 8.04	T4570	-	Road Runner	1.8	Test data

Showing 4 of 13 devices. Devices with a pink background aren't in service. [Show all](#)

Device name, serial number and asset number are searched. Searching is not case sensitive.

Connected from ::1 and not logged in. Generated by RackMonkey v1.2.5 at 2009-04-08 20:01 GMT.

10. pav. RackMonkey sąsaja su serverių informacija

Iš sąsajos matome, kad galima įsirašyti pagrindinę informaciją apie kiekvieną serverį. Taip pat yra galimybė peržiūrėti serverių spintų informaciją, paveikslėlis žemiau:

Rack A1 In TFM4		Rack A2 In TFM4	
20	sw1 (Catalyst 3560)	20	sw2 (Catalyst 3560)
19		19	
18		18	
17	mon1 (ProLiant DL365 G5)	17	mon2 (ProLiant DL365 G5)
16		16	
15		15	
14		14	
13		13	
12		12	appdev (Fire T2000)
11	app1 (Fire T2000)	11	app2 (Fire T2000)
10		10	
9		9	
8	www3 (PowerEdge 2850)	8	www4 (PowerEdge 2850)
7		7	
6	www1 (PowerEdge 2850)	6	www2 (PowerEdge 2850)
5		5	
4		4	
3	db1 (System p5 510Q)	3	db2 (System p5 510Q)
2		2	
1		1	

11. pav. RackMonkey serverių spintų informacija

Sistemoje itin naudinga funkcija - galima kaupti informaciją nepriklausomai nuo gamintojų ar įrenginių tipų.

RackMonkey privalumai:

- Galima nurodyti, kokios taikomosios programos veikia serveriuose.
- Galima nurodyti operacinės sistemos licencijos numerį prie serverio.
- Dublikatų paiešką pagal serijinį numerį, licenciją.
- Informacija pasiekama per API.

Trūkumai:

- Nėra galimybės automatizuotai rinkti serverių informaciją iš tinklo.
- Nėra galimybės valdyti serverių per IPMI sąsają.
- Nėra galimybės automatizuoti veiksmų su serveriais (nustatyti sekantį kartą krautis iš tinklo ir tuomet perkrauti serverių)

2.6.4.1. RackMonkey apžvalgos apibendrinimas

Nors tai visiškai skirtingas įrankis nuo kuriamos sistemos ir jis negali nei diegti OpenStack, nei valdyti serverių – vienas įdomiausių šio sprendimo bruožų – dinamiškas serverių spintų atvaizdavimas ir informacijos parodymas. Viena iš kuriamos sistemos funkcijų yra būtent tokių serverių spintų atvaizdavimas, taigi galima tvirtai nuspręsti, kad serveriai turėtų būti aptinkami automatiškai, o ne suvedami rankomis.

2.6.5. TripleO ir Tuskar

Ši sistema yra išskirtinė tuo, kad naudoja OpenStack įrankius diegti OpenStack sistemą, taigi reikia „turėti debesį, norint įdiegti debesį“. TripleO tai dedikuotas OpenStack diegimo įrankis, kuris dar vadinamas „žemuoju debesiu“ ir kuria naują OpenStack fiziniuose serveriuose, kuris vadinamas „viršutiniu debesiu“.

2.6.5.1. TripleO veikimo principas

Kadangi naudojami standartiniai OpenStack komponentai nusakyti, kaip atrodoys kuriamas OpenStack debesis – norimo debesies konfigūracija aprašoma Heat komponento šablone ir tada visas diegimas yra valdomas iš Heat. Skaičiavimo serveriai yra įdiegiami tiesiai į fizinius serverius naudojant Nova komponento „bare-metal (Ironic)“ funkciją. Tada, naudojamas PXE ir įdiegiami operacinių sistemų atvaizdai.

Diegėjai naudoja standartinius OpenStack įrankius:

- Keystone komponentą autentifikacijai ir autorizacijai.
- Horizon – grafinė vartotojo sąsaja.
- Nova konsolę.

Viskas vyksta lygiai taip pat, kaip būtų kuriamos virtualios mašinos OpenStack aplinkoje, taigi sistemos naudotojam praktiškai nėra jokių naujų funkcijų, kurios galėtų kelti neaiškumų.

TripleO skirtas didžiausių OpenStack aplinkų diegimam, atlieka nuolatinę integraciją daugelyje OpenStack aplinkų. Šį įrankį plėtoja tokios įmonės kaip HP, Red Hat bei daugelis kitų, jis yra integruotas į OpenStack, taigi įtrauktas į naujausią OpenStack versiją.

2.6.5.2. TripleO ir Tuskart apžvalgos apibendrinimas

Šis įrankis anksčiau ar vėliau bus naudojamas mūsų įmonės viduje, tačiau, nors ir yra daug tenkinančių funkcijų sprendimas, bet nėra būdo valdyti ir automatiškai aptikti daugelį fizinių serverių. Be to, šios sistemos kūrimo metu tai dar nėra pilnai išvystytas produktas. Jo išleidimo data buvo ketvirtas 2014 metų ketvirtis, tačiau Red Hat nukėlė jo išleidimą į 2015 pirmą ketvirtį, o tada dar vėliau.

2.6.6. Esamų sprendimų palyginimo išvados

Kuriama sistema yra didelio masto, todėl buvo atsižvelgta į daugelį kitų sprendimų. Šiame skyriuje nusakomi visos norimos funkcijos, kurios bus reikalingos diegiant ir prižiūrint OpenStack aplinką. Žemiau pateikta lentelė, kurioje įvertinami visų anksčiau pateiktų sprendimų funkcijų įvertinimai. Įrankiai įvertinami balais nuo 0 iki 2, kai 2 – tenkina pilnai, 1 – tenkina dalinai (pavyzdžiui Fuel turi galimybę ieškoti serverių tinkle, tačiau tik kai turi SSH raktą, su kuriuo gali prisijungti į juos), o 0 skiriamas toms sistemoms, kurios šios funkcijos neturi.

1. lentelė. Esamų sprendimų palyginimas

Palyginimo kriterijus	Reikalinga sistema	Fuel	QuotaViewer	RackMonkey	TripleO ir Tuskar	HP SIM
Automatinė serverių paieška tinkle be išankstinio SSH raktų apsikeitimo	2	1	0	0	0	2
Serverių spintų atvaizdavimas	2	0	0	2	0	2
Serverių valdymas per IPMI sąsają	2	0	0	0	0	2
Nepriklausomybė nuo serverių gamintojo	2	2	0	2	2	0
Automatizuotas	2	2	0	0	2	0

OpenStack diegimas						
Nestandartinių OpenStack sistemos komponentų diegimas	2	0	0	0	0	0
OpenStack sistemos parametrų peržiūra	2	2	1	0	2	0
OpenStack sistemos parametrų eksportavimas į CSV tipo failą	2	2	2	0	2	0
Papildomų OpenStack parametrų atvaizdavimas	2	0	0	0	0	0
Preiga prie OpenStack virtualių mašinų VNC konsolės	2	2	0	0	2	0
Informacijos apie visas OpenStack aplinkas įmonėje kaupimas ir atvaizdavimas	2	0	0	0	2	0
Pasaulio žemėlapis su visomis OpenStack aplinkomis (tikų ir sąrašas visų OpenStack aplinkų vienoje vietoje)	2	0	0	0	1	0
OpenStack sistemoje atskirų projektų išnaudojamų resursų atvaizdavimas	2	0	0	0	0	0
Rezultatai	26	11	3	4	13	6

Iš šios lentelės matome, jog tikslui pasiekti reikalinga sistema turėtų būti žymiai pranašesnė už atskirus sprendimus. Netgi pasirinkus naudoti atskiras sistemas vienu metu, siekiant turėti galimybę vykdyti visas funkcijas – būtų sudėtinga išlaikyti teisingus duomenis (be klaidų) keliuose sistemose vienu metu.

2.7. Darbo tikslas, pagrindiniai uždaviniai, problemos ir siekiami privalumai

Iš atliktos analizės matome, jog praktiškai visos sistemos, skirtos OpenStack diegimui naudoja tam tikrus šablonus, skirtus automatizuoti veiksmus. Pavyzdžiui, Mirantis Fuel, - nepavykus skaičiavimo serverio diegimui kartoja jį tol, kol jis pavyksta. Vėliau, galima išsaugoti sėkmingų diegimų konfigūracijas ir jas naudoti kitiems OpenStack diegimams. Šio darbo tikslas – sukurti OpenStack sistemos komponentų diegimui reikalingų žinių išsaugojimo skriptus, kurie automatizuoja patį diegimą ir OpenStack aplinkų stebėjimui skirtą sistemą. Šiam tikslui pasiekti, reikės išspręsti pagrindinius uždavinius:

- Sukurti sistemą, kuri leistų automatizuoti veiksmus su serveriais (aptikti serverius, diegti jiems operacines sistemas, valdyti per IPMI sąsają).
- Prisijungti prie OpenStack per šios sistemos API ir gauti informaciją apie jos išnaudojamus resursus.
- Agreguoti informaciją iš daugelio OpenStack aplinkų į vieną centralizuotą sistemą.

Problemos - įprastas, neautomatizuotas OpenStack diegimas iki dešimties serverių esančiai aplinkai trunka mažiausiai kelias dienas ir tai atlikti gali tik patyręs inžinierius, išmanantis visus OpenStack komponentus. Darbo rinkoje nėra daug tokių žmonių, kurie galėtų tai atlikti, o tai atlikti keliems tūkstančiams serverių tai taptų neįmanoma užduotimi. Taigi, šio tiriamojo darbo tikslas yra

sukurti sistemą, kuri galėtų automatizuotai diegti OpenStack sistemą perdavus jai tik pagrindinius parametrus, tokius kaip tinklo nustatymai ir t.t, o ši sistema būtų pritaikyta naudotis ne tik patyrusiems OpenStack inžinieriams, bet ir pradėjusiems. Taip pat kuriama sistema galėtų per API jungtis į OpenStack ir realiu laiku stebėti įvairius vidinius parametrus, pavyzdžiui resursų išnaudojimą. Kadangi, PXE ir TFTP technologijos yra gana plačiai taikomos automatizuotame diegime, jos bus naudojamos ir šiame sprendime [8].

Sistemos naudotojai yra inžinieriai, kuriantys komponentus OpenStack sistemai, analitikai ir viso infrastruktūros kūrimo departamento vadovai.

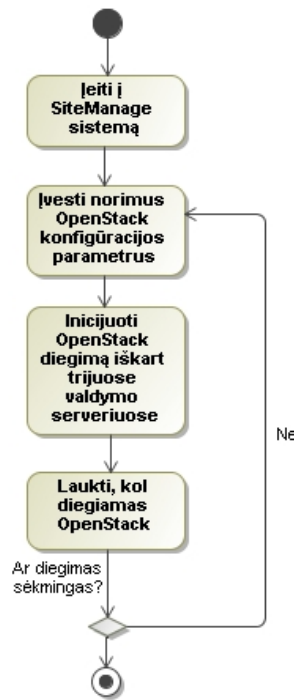
Pagrindiniai uždaviniai pagal grupes:

- Inžinierių grupė – reikalinga platforma, kuria būtų galima valdyti servius per ILO sąsają – keisti įrenginį, iš kurio užsikrauna OS, perkrauti serverį, išnaudoti jo vietos nustatymo galimybes (išnaudojant šią funkciją galima dinamiškai atkurti serverių spintų vaizdą vartotojo ekrane). Taip pat yra reikalavimas kuriant OpenStack aplinkas perduoti pradinis parametrus į skriptus, tokius kaip tinklo nustatymai, instaliacijos patikrinimai, serverių pradinės konfigūracijos patikrinimai.
- Analitikų grupė – ši klientų grupė yra orientuota į OpenStack aplinkos išnaudojimą, jiems svarbiausia yra informacija pagal projektus, kiek kas išnaudoja, kiek resursų yra laisvų ir kada gali reikėti pirkti papildomos įrangos. Kuriamos sistemos privalumai: informacija pagal projektus su papildomu „Shares“ parametru, kuris negalėtų būti atvaizduojamas kitose sistemose, eksportavimo į CSV tipo failus galimybė.
- Palaikymo komanda – jiems svarbiausia QuotaViewer stiliaus informacija, tačiau kuriama sistema taip pat turės galimybę suteikti prisijungimą į VNC konsolę, taip pat stebėti fizinius serverius bei serverių spintas, kuriose jie yra. Be to, nustačius, kad problema yra su fiziniu serveriu – jie galės įjungti LED lempuotę ant serverio (kad duomenų centro palaikymo komanda galėtų šalinti fizinius gedimus) arba tiesiog perkrauti serverį ir perrašyti OpenStack sistemą.

2.8. Siekiamo sprendimo apibrėžimas

Kuriamas sprendimas yra sistema, susidedanti iš dviejų posistemų, iš kurių viena atsakinga už serverių valdymą, automatizuotą OpenStack diegimą ir jo resursų išnaudojimo stebėjimą, o kita atsakinga už informacijos konsolidavimą bei OpenStack testavimą.

Ši sistema turėtų pastebimai sutrumpinti sistemos diegimą, taip pat daugelyje diegimo vietų panaikintų žmogiškosios klaidos faktorių, kai labai lengva supainioti daugelio komponentų tarpusavio komunikavimo konfigūraciją. Žemiau pateikta automatizuoto diegimo veiklos proceso diagrama:



12. pav. Automatizuotas OpenStack diegimas ir konfigūravimas

Iš šios diagramos matome, jog tampa labai nesudėtinga pradėti OpenStack instaliaciją. Net įvedus netinkamą konfigūraciją – serverius galima iš naujo diegti neskiriant tam daug laiko. Visas operacijas atlieka SiteManager serveris, taigi sistemos naudotojui net nereikia būti prisijungus prie SiteManager – jis gali pradėti instaliaciją ir užsiimti savo darbais.

2.9. Analizės išvados

Analizės metu nustatytos:

1. Tyrimo objektas – OpenStack yra didelė ir sudėtinga sistema, kuriai reikalingas ilgas ir kruopštus diegimas (didinamas inžinierių kiekis tik padidintų konfigūracijos klaidų galimybę). Geriausias būdas spręsti diegimo ir debesies būklės stebėjimą – procesą automatizuoti, naudojant pradinius kintamuosius, kurie perduodami į instaliacijos skriptą, kuris savo ruožtu juos pakartotinai panaudoja diegime. Taip pat šie aplinkos kintamieji (virtualus IP, lokali administracinė naudojimo paskyra) tuomet gali būti naudojami prisijungimams prie OpenStack API per kurį gaunama informacija apie turimus debesies resursus.
2. Naudotojų grupės ir kokios funkcijos jiems yra svarbiausios. Turint šią informaciją, galima teigti, jog bus reikalinga griežta autentifikacijos ir autorizacijos sistema, kai skirtingoms grupėms yra leidžiama atlikti tik joms priskirtas funkcijas. Šios funkcijos praktiškai visiškai persidengia tarp diegėjų ir palaikymo komandų, tačiau projektų vadovų grupei turėtų būti apribota prieiga prie serverių valdymo bei tam tikros informacijos apie virtualias mašinas.
3. Įvertinus rezultatus iš esamų sprendimų analizės matome, jog panašūs sprendimai, nors ir turintys dalį reikiamų funkcijas – niekaip negali atlikti pilno automatizuoto OpenStack diegimo. Tokiu atveju reiktų naudoti kelias sistemas vienu metu, kas dar padidintų diegimo sudėtingumą. Taip pat yra funkcijų, kurių negali pilnai atlikti nei viena sistema, taigi vis tiek reiktų kurti papildomus įrankius, kurie atliktų kai kurias funkcijas.

Taip pat analizės metu nustatyta, jog:

4. svarbu turėti fizinių resursų valdymo galimybes, kad būtų įmanoma atlikti šiuos veiksmus:

- a) Atlikti automatinę serverių paieška nurodytame potinklyje.
 - b) Importuoti serverių informaciją iš CSV failo.
 - c) Valdyti serverius per IPMI sąsają.
5. OpenStack diegimo automatizavimui nustatytos svarbiausios charakteristikos:
- a) Prieš diegiant sprendimą turi būti galimybė rinktis serverius iš turimų resursų. Šie resursai taip pat turėtų būti automatiškai patvirtinami, pavyzdžiui duomenų saugojimui skirtų serverių neturėtų būti galimybės pasirinkti kaip valdymo komponentų ir atvirkščiai. Nei viena iš alternatyvių sistemų neturi šios galimybės. Taigi kuriama sistema turėtų pagal nustatytus šablonus atpažinti serverius ir priskirti juos vienai iš trijų grupių:
 - Valdymo serveriai
 - Skaičiavimo serveriai
 - Duomenų saugojimo serveriai
 - b) Turėti galimybę realiu laiku stebėti OpenStack sistemos vidinius resursus:
 - Sukurtas virtualias mašinas ir su jomis susijusius duomenis bei meta-duomenis.
 - Stebėti kitus OpenStack sistemos parametrus, tokius kaip „flavors“, „images“, „networks“, „users“, „tenants“, „images“.
 - c) Agreguoti duomenis iš daugelio OpenStack aplinkų ir geografiškai matyti jų išsidėstymą pasaulyje bei veikiančių arba neveikiančių servisų būklę šiose aplinkose.

Taigi, sprendimo specifikacijos, sudarytos pagal esamų sprendimų analizę yra toliau projektuojamas ir realizuojamas tiriamajame darbe.

3. SITEMANAGER IR CLOUDMAP SPRENDIMO REIKALAVIMŲ SPECIFIKACIJA IR PROJEKTAS

Šiame skyriuje specifikuojami reikalavimai, pateikiami funkciniai ir nefunkciniai sistemos reikalavimai, panaudos atvejų diagramos. Didžiausias dėmesys skiriamas funkcijų išaiškinimui dėl didelės jų gausos bei skirtingo sudėtingumo lygio.

3.1. Reikalavimų specifikacija

Kuriama vidinio debesų kompiuterijos sprendimo ir resursų valdymo sistema (SiteManager ir CloudMap). Naudotojas, norėdamas atlikti OpenStack sistemos automatizuotą diegimą, turės prisijungti prie SiteManager sistemos per interneto naršyklę. Prisijungimo duomenys bus saugojami Active Directory serveryje, o AD grupė nusakys jo teises. Prisijungęs, galės įvairiais būdais į SiteManager duomenų bazę įrašyti fizinių serverių, paruošti jų operacines sistemas (automatizuotai daugeliui serverių arba po vieną) ir pradėti OpenStack sistemos diegimą.

Įdiegus OpenStack, SiteManager sistemoje diegėjų grupei bus galima matyti, ar diegimas įvyko be problemų ir stebėti kitus OpenStack sistemos parametrus, tokius kaip:

- Serverių spintas, kuriose yra serveriai, priskirti šiai OpenStack aplinkai.
- Bendrus aplinkos parametrus, tokius kaip versija, IP adresai, fizinių serverių kiekis.
- Hipervizoriai ir jų resursų išnaudojimas.
- Sistemos taškai, per kuriuos pasiekiami visi komponentai.
- OpenStack serverių grupės ir jų valdymas.
- Virtualių mašinų šablonai.
- Operacinių sistemų atvaizdai.
- Sukurtos virtualios mašinos ir jų informacija, VNC prieiga.
- Sistemoje registruoti naudotojai (Tik sisteminiai naudotojai, AD grupės ir jose esantys naudotojai nebus įtraukti).
- Sistemoje sukurti projektai, jų aprašymai ir jų išnaudojami resursai.

OpenStack palaikymo komanda ir analitikai SiteManager sistemoje turės ribotas galimybes veikti (priešingai nei diegėjai) – analitikai galės stebėti praktiškai visus duomenis, tačiau negalės jų redaguoti. Jų rolė šioje sistemoje bus informacijos rinkimas raportų kūrimui (kol nebus užbaigtas raportų automatizavimo projektas), o OpenStack palaikymo komanda dar turės papildomas galimybes veiksams su klientų turimomis virtualiomis mašinomis.

Esant bent vienam SiteManager, jo duomenys bus matomi CloudMap sistemoje. Šis procesas visiškai automatizuotas, taigi CloudMap sistemos naudotojai neturės suvedinėti jokios papildomos informacijos ir galės matyti visas SiteManager sistemas ir OpenStack aplinkas, kai tik jos bus sudiegtos. Į CloudMap sistemą naudotojai taip pat jungsis per AD autentifikaciją.

CloudMap sistema taip gali atlikti OpenStack sistemos testavimus:

- Tikrinti, ar pasiekiamas API.
- Tikrinti, ar pavyksta sukurti ir ištrinti virtualią mašiną.

3.1.1. Bendros sistemos funkciniai reikalavimai

Kuriamos sistemos galės atlikti šias pagrindines funkcijas:

- Valdyti fizinių serverių ir kitų, duomenų bazės modelyje apibrėžtų, elementų informaciją (kurti naujus elementus, redaguoti, trinti)
- Valdyti fizinius serverius per IPMI sąsają
- Vizualiai atvaizduoti serverių spintas su jose esančiais serveriais (dinamiškai nubrėžti serverių spintos paveikslėlių)
- Automatizuotai diegti OpenStack aplinką
- Realiu laiku stebėti OpenStack aplinkos parametrus (bei išimtiniais atvejais juos keisti)

- Agreguoti visų SiteManager sistemų duomenis į vieną centrinį tašką
- Leisti automatinius OpenStack sistemos testus

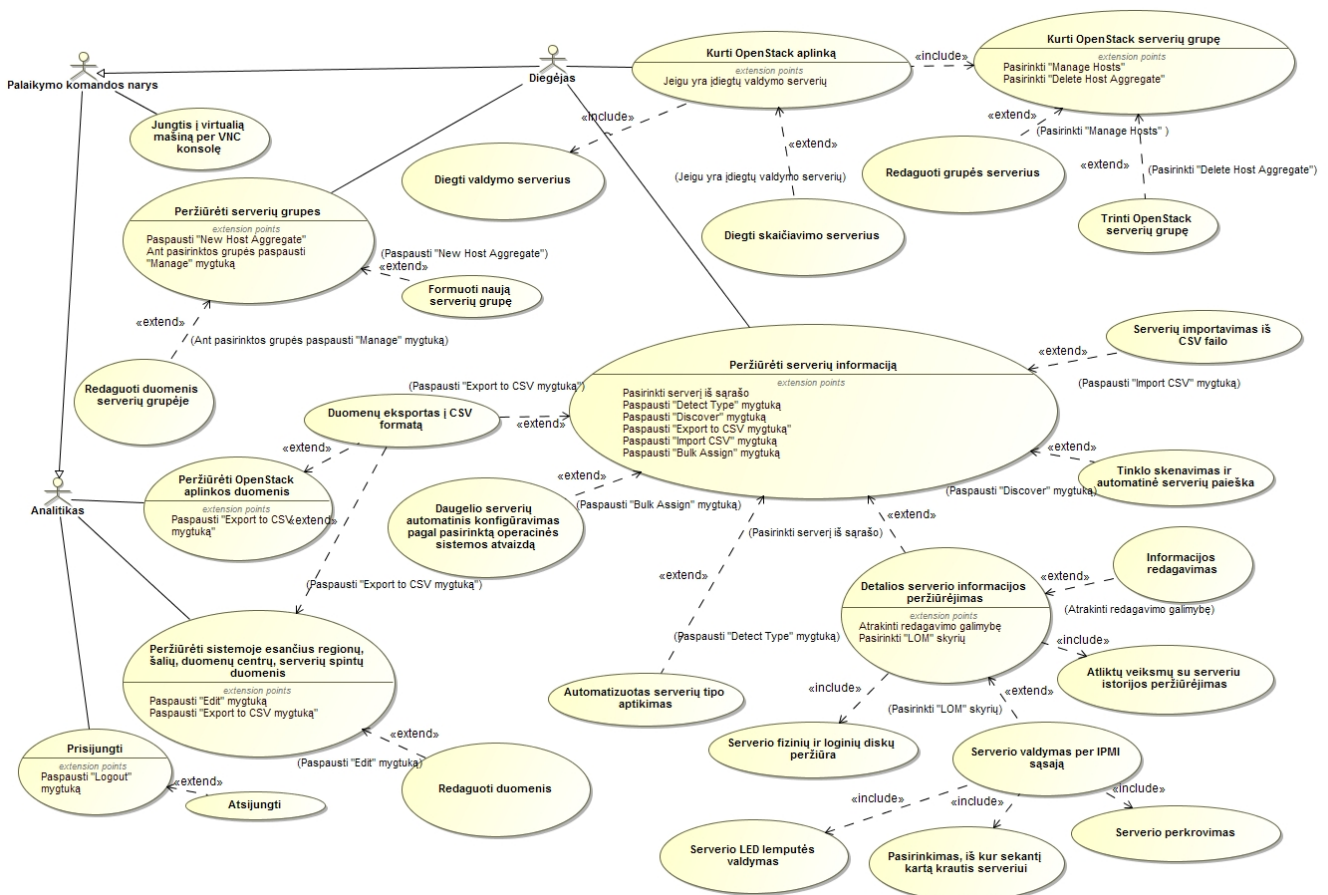
Diegti SiteManager sistemai taip pat bus paruoštas RPM paketas, skirtas automatizuoti daugelį veiksmų. CloudMap sistemai šiuo metu nėra numatoma sukurti RPM paketo, dėl to, kad planuojama ją turėti tik vieną per visą įmonės tinklą.

3.1.2. Vartojimo (panaudos) atvejų diagramos

Kuriamos sistemos panaudos atvejų diagramos pateiktos sekančiuose skyriuose. Diagramomis norima parodyti sistemos naudotojų ryšius su pačiomis sistemomis, sistemų tarpusavio ryšius (SiteManager ir CloudMap) bei šių kuriamų sistemų ir OpenStack ryšius.

3.1.2.1. SiteManager sistemos panaudos atvejų diagrama

Šioje diagramoje aprašomi veiksmai, kuriuos gali atlikti SiteManager sistemos naudotojai. Visas teises sistemoje turi diegėjas, jam tai reikalinga, nes šis naudotojas OpenStack diegimo metu gali panaudoti visas sistemos funkcijas. Jam reikalingos serverių paieškos funkcijos, jų valdymas, prieiga prie OpenStack sistemos diegimo funkcijų ir, galiausiai, OpenStack aplinkos parametrų peržiūrėjimas, nes reikia patikrinti, ar visa informacija sėkmingai įrašyta (operacinių sistemų atvaizdai, virtualių mašinų šablonai).



13. pav. SiteManager sistemos panaudos atvejų diagrama

Toliau pateikti papildomi panaudos atvejų aprašymai.

2. lentelė. Serverių informacijos peržiūrėjimo panaudos atvejais

Panaudojimo atvejais	Peržiūrėti serverių informaciją
Aprašymas	Diegėjui prisijungus į SiteManager sistemą – galima peržiūrėti sistemoje išsaugotus serverius bei jų papildomą informaciją, pridėti naujus serverius arba trinti jau esamus įrašus.

Aktoriai	Diegėjas.
Prieš-sąlyga	Diegėjas turi būti prisijungęs
Po-sąlyga	Naršyklės lange atsiranda serverių sąrašas su papildoma informacija stulpeliuose. Taip pat matomi mygtukai, kurių pagalba galima atlikti papildomus veiksmus su serveriais.

3. lentelė. Detalios serverio informacijos peržiūrėjimo panaudos atvejis

Panaudojimo atvejis	Detalios serverio informacijos peržiūrėjimas
Aprašymas	Diegėjui pasirinkus serverį iš sąrašo atsidaro naujas langas, kuriame pateikiama pasirinkto serverio detali informacija. Taip pat per šią sąsają diegėjas gali priskirti serveriui operacinės sistemos atvaizdą, koreguoti jo tipą bei pereiti į kitus, serverio istorijos bei valdymo skirtukus.
Aktoriai	Diegėjas.
Prieš-sąlyga	Diegėjas turi serverių sąrašo lange
Po-sąlyga	Naršyklės lange atsiranda serverio detalės su keturiais skirtukais: <ul style="list-style-type: none"> • Serverio pagrindiniams parametrų (aktyvus) • Serveriui atliktų veiksmų sąrašas. • Serverio valdymo per IPMI sąsają langas. • Serverio fizinių ir loginių diskų informacijos langas.

4. lentelė. Informacijos redagavimo panaudos atvejis (serverio)

Panaudojimo atvejis	Informacijos redagavimas
Aprašymas	Diegėjui atsidarius serverio detalios informacijos langą – yra galimybė keisti praktiškai visus, duomenų bazėje laikomus duomenis, tokius kaip serverio vardas, gamintojas, modelis ir t.t. (visą sąrašą galima rasti prie duomenų bazės aprašymo, žiūrint į serverio klasę). Pavyzdinis veiksmas: <ul style="list-style-type: none"> • Diegėjas pakeičia serverio pavadinimą. • Paspaudžia išsaugojimo mygtuką
Aktoriai	Diegėjas.
Prieš-sąlyga	Diegėjas atsidaręs detalią serverio informaciją. Serverio tam tikri parametrai yra apsaugoti, o norint juos pakeisti – reikia papildomai savo veiksmą aprašyti <i>ServiceNow</i> sistemoje.
Po-sąlyga	Informacija įrašoma į duomenų bazę.

5. lentelė. Atliktų veiksmų su serveriu informacijos peržiūrėjimo panaudos atvejis

Panaudojimo atvejis	Atliktų veiksmų su serveriu istorijos peržiūrėjimas
Aprašymas	Diegėjas atsidaro serverio detalios informacijos langą, tuomet atsidaro istorijos rodymo skirtuką. Šiame skirtuke gali peržiūrėti visus atliktus veiksmus, jų datą ir sistemos naudotoją, kuris atliko pakeitimą.
Aktoriai	Diegėjas.
Prieš-sąlyga	Diegėjui reikia peržiūrėti serverio informaciją, kad galėtų matyti, ar jau buvo šis serveris sukonfigūruotas ir paruoštas diegimui (galima patikrinti, ar priskyrimas operacinės sistemos atvaizdą buvo pasirinktas krovimasis iš tinklo ir perkrautas serveris)
Po-sąlyga	Parodoma detali serverio veiksmų istorija

6. lentelė. Serverio valdymo per IPMI sąsają panaudos atvejais

Panaudojimo atvejis	Serverio valdymas per IPMI sąsają
Aprašymas	Diegėjui valdo serverį per IPMI sąsają. Veiksmai, kuriuos gali atlikti diegėjas: <ul style="list-style-type: none"> • Perkrauti serverį. • Įjungti serverį. • Išjungti serverį. • Įjungti LED lempuotę ant serverio (tuomet duomenų centro personalas greičiau gali atpažinti serverį). • Nustatyti krovimąsi iš kito įrenginio, pavyzdžiui tinklo.
Aktoriai	Diegėjas.
Prieš-sąlyga	Diegėjui reikia nustatyti serverio sekantį krovimąsi iš tinklo. O tam reikia tai padaryti per serverio IPMI sąsają. Taip pat sistemos naudotojas turi pateikti galiojantį pakeitimo numerį iš <i>ServiceNow</i> sistemos.
Po-sąlyga	Serveriui siunčiamos komandos per IPMI sąsają iš SiteManager sistemos.

7. lentelė. Serverių importavimo iš CSV failo panaudos atvejais

Panaudojimo atvejis	Serverių importavimas iš CSV failo
Aprašymas	Diegėjui atidaro serverių sąrašo langą, paspaudžia „ <i>Import</i> “ mygtuką, atsidariusiame lange pasirenka failą, kurį nori įkelti, nurodo, kokią informaciją SiteManager sistema turėtų dar papildomai atnaujinti ir paspaudžia „ <i>Upload</i> “ mygtuką. SiteManager sistema tada nuskaity sąrašą, sudeda į duomenų bazę ir, priklausomai nuo pasirinktos reikšmės, atlieka papildomus veiksmus.
Aktoriai	Diegėjas.
Prieš-sąlyga	Diegėjui reikia iš CSV failo įkelti duomenis į SiteManager sistemą. Kadangi po vieną vesti keliasdešimt arba šimtus serverių būtų sudėtinga – jis gali naudotis šia funkcija.
Po-sąlyga	Iš CSV failo importuoti serveriai pasirodo serverių sąrašė.

8. lentelė. Tinklo skenavimo ir automatinės serverių paieškos panaudos atvejais

Panaudojimo atvejis	Tinklo skenavimo ir automatinės serverių paieškos panaudos atvejais
Aprašymas	Diegėjui atidaro serverių sąrašo langą, paspaudžia „ <i>Discovery</i> “ mygtuką. Tuomet įveda potinklį, IP adresą arba IP adresų režį, NMAP taikomajai programai skirtus argumentus, IPMI prisijungimo duomenis (pagal suvestus duomenis SiteManager sistema nusprendžia, ar šis serveris yra ieškomas, kad neprisidėti svetimų serverių). Taip pat, diegėjas nurodo tolimesnius veiksmus dėl informacijos atnaujinimo
Aktoriai	Diegėjas.
Prieš-sąlyga	Diegėjas žino arba įtaria nurodytame potinklyje arba IP adresų režyje esančius serverius. Naudodamasis serverių paieška pagal IP funkcija ir žinodamas IPMI prisijungimo duomenis, jis gali tuos serverius automatiškai pridėti į SiteManager duomenų bazę.
Po-sąlyga	Tinkle surasti serveriai pasirodo serverių sąrašė.

9. lentelė. Daugelio serverių automatinio konfigūravimo pagal pasirinktą operacinės sistemos atvaizdą panaudos atvejais

Panaudojimo atvejis	Daugelio serverių automatinis konfigūravimas pagal pasirinktą operacinės sistemos atvaizdą
----------------------------	--

Aprašymas	Diegėjui atidaro serverių sąrašo langą, pasirenka norimus konfigūruoti serverius ir paspaudžia iš „Actions“ mygtuko meniu „Assign Image“ nuorodą. Atsidaro naujas langas, kuriame pasirenka šiuos nustatymus: <ul style="list-style-type: none"> • Operacinės sistemos atvaizdą. • Ar nustatyti krovimąsi iš tinklo. • Ar perkrauti po konfigūravimo. • <i>ServiceNow</i> sistemoje galiojančio pakeitimo numerį. Taip pat diegėjas gali peržiūrėti sąrašą serverių, kurie buvo pasirinkti ankstesniame lange ir, jeigu reikia, pakoreguoti jį prieš vykdant komandą.
Aktoriai	Diegėjas.
Prieš-sąlyga	Diegėjui reikia sukonfigūruoti vieną ar daugiau serverių, o tam SiteManager sistemoje turi būti surastų fizinių serverių, kuriuos galima būtų konfigūruoti ir egzistuojančių operacinės sistemos atvaizdų.
Po-sąlyga	Serverių sąrašė matomi perkonfigūruoti serveriai pagal pasikeitusius operacinės sistemos atvaizdus. Jeigu nepavyko kai kurių serverių perkrauti arba nustatyti krovimosi iš tinklo – tai išvedama žinučių formatu į serverių sąrašo langą.

10. lentelė. Automatizuoto serverių tipo aptikimo panaudos atvejis

Panaudojimo atvejis	Automatizuotas serverių tipo aptikimas
Aprašymas	Diegėjui atidaro serverių sąrašo langą, pasirenka norimus konfigūruoti serverius ir paspaudžia iš „Actions“ mygtuko meniu „Detect Node Type“ nuorodą. SiteManager sistema kiekviename pasirinktame serveryje prisijungia per IPMI sąsają, gauna fizinių diskų konfigūraciją ir pagal nustatytus šablonus nustato, koks serverio tipas. Galimi variantai: <ul style="list-style-type: none"> • Controller – valdymui skirtas serveris. • Compute wit local storage – skaičiavimo serveris su papildoma vieta duomenų saugojimui. • Compute with shares storage – skaičiavimo serveris, kuris neturi papildomos vietos duomenų saugojimui. • Storage – duomenų saugojimo serveris
Aktoriai	Diegėjas.
Prieš-sąlyga	Diegėjui prieš diegiant OpenStack aplinką reikia nustatyti fizinių serverių tipus, nes kitaip SiteManager sistema neleidžia rinktis serverių diegimo metu (kurie neturi tipų).
Po-sąlyga	Serveriam priskiriami tipai, juos galima toliau naudoti OpenStack diegimo metu.

11. lentelė. OpenStack aplinkos duomenų peržiūrėjimo panaudos atvejis


Panaudojimo atvejis	Peržiūrėti OpenStack aplinkos duomenis
Aprašymas	Sistemos naudotojas prisijungia prie SiteManager, atidaro OpenStack aplinkų sąrašo langą, pasirenka norimą aplinką.
Aktoriai	Analitikas, Palaikymo komandos narys, Diegėjas (betkuris autentifikuotas sistemos naudotojas)
Prieš-sąlyga	Sistemos naudotojui reikia peržiūrėti OpenStack aplinkos duomenis.
Po-sąlyga	Atsidaro langas su OpenStack aplinkos parametrais skirtukuose: <ul style="list-style-type: none"> • Settings – šis langas skirtas parodyti aplinkos limitus, OpenStack versiją, diegimo fazę, priskirtus fizinius serverius, aplinkai priskirtus IP adresus. • Servers – šiame lange, dar neįvykdžius diegimo, galima rinktis


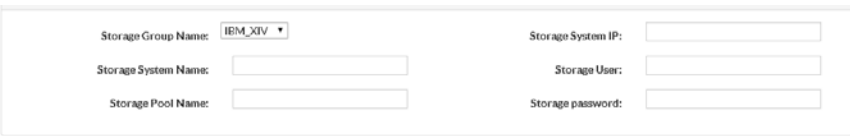
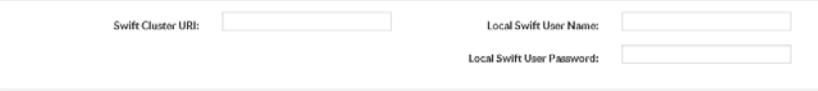
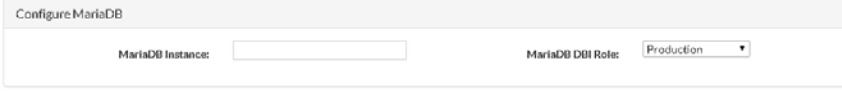
	<p>valdymo serverius, į kuriuos bus įdiegta OpenStack sistema. Vėliau, kai aplinka aktyvi – per šį skirtuką galima pridėti papildomus skaičiavimo serverius.</p> <ul style="list-style-type: none"> • Messages – šis skirtukas reikalingas OpenStack aplinkoje vykdytų veiksmų parodymui. Čia ateina informacija iš serverių ir pačio SiteManager, kai baigiama diegti papildomi serveriai arba atliekami kiti veiksmai. • Racks - pagal priskirtus fizinius serverius yra surandamos serverių spintos ir jų sąrašą galima rasti šiame lange. • Endpoints - OpenStack aplinkos komponentų administraciniai, išoriniai ir vidiniai adresai bei sąsajos. • Hypervisors - aplinkos hipervizoriai, jų tipai ir išnaudojami resursai. • Aggregates – serverių grupių OpenStack aplinkoje kūrimas, redagavimas ir trynimasis. • Flavors - virtualių mašinų šablonų sąrašas. • Images – operacinės sistemos atvaizdų sąrašas. • Instances – virtualių mašinų sąrašas. • Users – OpenStack Keystone komponente išsaugotų naudotojų sąrašas. • Tenants – OpenStack sistemoje registruoti projektai. <p>Sistemos naudotojas gali eiti į bet kurį skirtuką ir matyti visą informaciją.</p>
--	---

12. lentelė. OpenStack aplinkos kūrimo panaudos atvejis

Panaudojimo atvejis	Kurti OpenStack aplinką
Aprašymas	Diegėjas atidaro OpenStack aplinkų sąrašo langą, paspaudžia mygtuką „Create“, atsidariusiame lange įveda aplinkos pavadinimą, pasirenka versiją, kurią norėtų įdiegti, konfigūracijos tipą (šiuo metu yra tik vienas – didelio patikimumo) ir duomenų centrą, kuriame ši aplinka bus.
Aktoriai	Diegėjas.
Prieš-sąlyga	Diegėjui reikia sukurti naują OpenStack aplinką.
Po-sąlyga	SiteManager sistemoje sukuriama OpenStack aplinka. Tačiau ji neaktyvi, reikia pasirinkti valdymo serverius ir pradėti diegimą.

13. lentelė. Valdymo serverių diegimo panaudos atvejis

Panaudojimo atvejis	Diegti valdymo serverius
Aprašymas	<p>Diegėjas sukurtoje OpenStack aplinkoje atidaro „Servers“ skirtuką ir jame pasirenka tris fizinius serverius, kurių tipas atitinka valdymo serverių šabloną. Paveikslėlis žemiau:</p>  <p style="text-align: center;">14. pav. Valdymo serverių pasirinkimo langas</p> <p>Tada, nurodomi tinklo nustatymai, paveikslėlis žemiau:</p>


	 <p style="text-align: center;">15. pav. Tinklo konfigūracijos parametrai</p> <p>SiteManager sistema pagal serverių duomenis automatiškai tikrina ar sutampa serverių VLAN numeriai, nes priešingu atveju, po diegimo, serveriai negalėtų tarpusavyje keistis informacija. Atlikus tinklo konfigūracija, pasirenkami Block Storage parametrai, tai OpenStack Cinder komponentas, kuris atsakingas už blokinių resursų virtualizaciją ir jų pasiekiamumo per API užtikrinimą. Sąsajos paveikslėlis žemiau:</p>  <p style="text-align: center;">16. pav. Blokinio duomenų saugojimo komponento nustatymai</p> <p>Sekančiame žingsnyje nustatoma objektinio duomenų saugojimo konfigūracija – Swift API adresas bei prisijungimo duomenys, sąsajos paveikslėlis žemiau:</p>  <p style="text-align: center;">17. pav. Objektinio duomenų saugojimo komponento nustatymai</p> <p>Tada konfigūruojama duomenų bazė – MariaDB. Galima nurodyti, kuri MariaDB naudoti, nes tame pačiame serveryje gali būti įdiegtos kelios duomenų bazės. Taip pat gali būti pasirinktas diegimo tipas, ar tai būtų produkcinio lygmens duomenų bazė (pilnai apsaugota), kūrimo lygmens (palikti keli būdai prieiti prie duomenų) arba tai galėtų būti atstatomoji duomenų bazė po incidento arba avarijos. Sąsajos paveikslėlis žemiau:</p>  <p style="text-align: center;">18. pav. Duomenų bazės konfigūravimas</p>
Aktoriai	Diegėjas.
Prieš-sąlyga	Diegėjui reikia pradėti OpenStack sistemos diegimą. Tam dar reikia nurodyti galiojantį ServiceNow sistemos pakeitimo numerį.
Po-sąlyga	SiteManager pradeda diegti OpenStack sistemą.

14. lentelė. Skaičiavimo serverių diegimo panaudos atvejais


Panaudojimo atvejis	Diegti skaičiavimo serverius
Aprašymas	Diegėjas atidaro aktyvios OpenStack aplinkos „Servers“ skirtuką, jame spaudžia „Add“ mygtuką. Atsidariusiame lange pasirenka, į kuriuos serverius nori įdiegti OpenStack ir priskirti skaičiavimo serverių roles. Taip pat pasirenkami išoriniai ir valdymo tinklai.
Aktoriai	Diegėjas.
Prieš-sąlyga	Diegėjui reikia pridėti skaičiavimo serverių į aktyvią OpenStack aplinką.

Po-sąlyga	SiteManager įdiegia ir sukonfigūroja serverius. Serverių būklė matoma „Servers“ skirtuke.
------------------	---

15. lentelė. OpenStack serverių grupių kūrimo panaudos atvejis

Panaudojimo atvejis	Kurti OpenStack serverių grupę
Aprašymas	Diegėjas atidaro OpenStack aplinkos langą, jame pasirenka „Aggregates“ skirtuką, jame paspaudžia mygtuką „Create Host Aggregate“, kurio paveikslėlis žemiau:  19. pav. Naujos serverių grupės sukūrimo mygtukas Tada, įveda serverių grupės pavadinimą ir zoną, kurioje ji veiks ir spaudžia mygtuką „Add“.
Aktoriai	Diegėjas.
Prieš-sąlyga	Diegėjui reikia sukurti naują serverių grupę OpenStack aplinkoje.
Po-sąlyga	Sukurta serverių grupė pasirodo grupių sąrašė. Tuomet jai galima priskirti serverius.

16. lentelė. OpenStack serverių grupės redagavimo panaudos atvejis

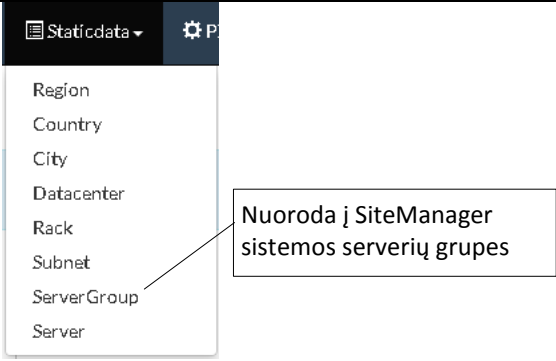
Panaudojimo atvejis	Redaguoti grupės serverius
Aprašymas	Diegėjas serverių grupės sąrašė prie norimo elemento paspaudžia mygtuką „More“ ir iš sąrašo pasirenka „Manage Hosts“. Tuomet, atsidariusiame lange pažymi serverius, kuriuos nori priskirti šiai grupei.
Aktoriai	Diegėjas.
Prieš-sąlyga	Diegėjui reikia priskirti serverius pasirinktai serverių grupei OpenStack aplinkoje.
Po-sąlyga	Redaguotas serverių sąrašas parodomas prie serverių grupių. Pavyzdys parodytas paveikslėlyje žemiau:  20. pav. Serverių atvaizdavimas OpenStack serverių grupėje Taip pat prie serverių grupės galima matyti ir metaduomenis.

17. lentelė. OpenStack serverių grupės ištrynimo panaudos atvejis

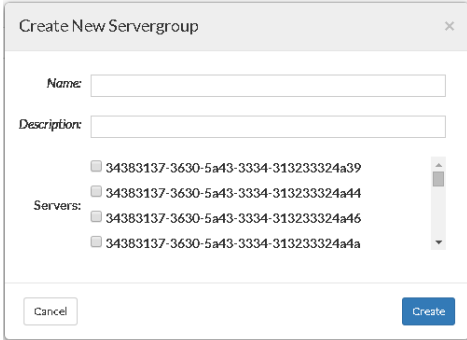
Panaudojimo atvejis	Trinti OpenStack serverių grupę
Aprašymas	Diegėjas serverių grupės sąrašė prie norimo elemento paspaudžia mygtuką „More“ ir iš sąrašo pasirenka „Delete Host Aggregate“. Serverių grupė pašalinama iš OpenStack (patys serveriai lieka niekaip nepaveikti).
Aktoriai	Diegėjas.
Prieš-sąlyga	Diegėjui reikia ištrinti serverių grupę iš OpenStack sistemos.
Po-sąlyga	Serverių grupė ištrinta.

18. lentelė. SiteManager serverių grupių peržiūrėjimo panaudos atvejis

Panaudojimo atvejis	Peržiūrėti serverių grupes
Aprašymas	Diegėjas pasirenka iš viršutinio meniu „Staticdata“, o tada „ServerGroup“. Meniu paveikslėlis pateiktas žemiau:

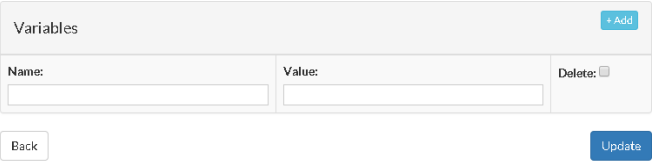
	 <p style="text-align: center;">21. pav. SiteManager statinės informacijos meniu esanti nuoroda į serverių grupes</p> <p>Serverių grupės reikalingos prieš pradėdant diegti OpenStack aplinką.</p>
Aktoriai	Diegėjas.
Prieš-sąlyga	Diegėjui reikia peržiūrėti serverių grupes.
Po-sąlyga	Atidaromas SiteManager serverių grupių sąrašas.

19. lentelė. SiteManager naujos serverių grupės kūrimo panaudos atvejis


Panaudojimo atvejis	Formuoti naują serverių grupę
Aprašymas	<p>Diegėjas atidaro SiteManager serverių grupių sąrašą. Tuomet, spaudžia mygtuką „Create“. Atsidariusiame lange įveda naujos serverių grupės pavadinimą, aprašymą ir pažymi serverius, kurie priklausys šiai grupei. Šąsajos paveikslėlis pateiktas žemiau:</p>  <p style="text-align: center;">22. pav. Naujos serverių grupės SiteManager sistemoje kūrimo langas</p> <p>Iš aukščiau pateikto paveikslėlio matome, jog kūrimo metu nėra galimybės numatyti serverių grupės parametrus.</p>
Aktoriai	Diegėjas.
Prieš-sąlyga	Diegėjui reikia sukurti naują serverių grupę SiteManager sistemoje.
Po-sąlyga	Sukurta nauja serverių grupė pasirodo sąrašė. Ją galima toliau redaguoti.

20. lentelė. SiteManager serverių grupės duomenų redagavimas


Panaudojimo atvejis	Redaguoti duomenis serverių grupėje
Aprašymas	<p>Diegėjas iš serverių grupių sąrašo pasirenka grupę ir atidaro jos detalią informaciją. Šiame lange diegėjas įveda papildomus parametrus, kurie vėliau bus naudojami diegiant OpenStack sistemą tam tikrai serverių grupei ir spaudžia mygtuką „Update“. Diegėjas taip pat gali ištrinti pasirinktus parametrus per šią sąsają. Šių papildomų parametrų kiekis yra neribojamas. Naudojamas principas „key:value“. Šąsajos pavyzdys paveikslėlyje žemiau:</p>

	 <p style="text-align: center;">23. pav. Parametru pridėjimo sąsaja</p>
Aktoriai	Diegėjas.
Prieš-sąlyga	Diegėjui reikia nustatyti papildomus parametrus serverių grupei.
Po-sąlyga	Serverių grupėje parodomi papildomi parametrai.

21. lentelė. Jungimosi į virtualią mašiną per VNC konsolę panaudos atvejis

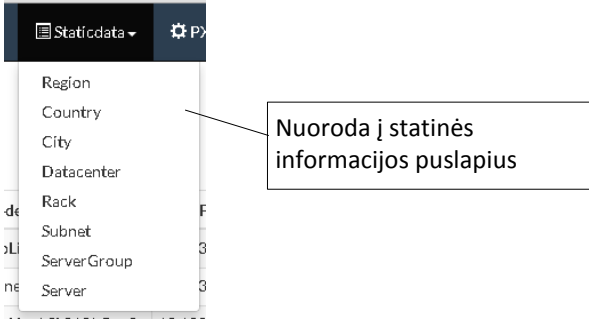
Panaudojimo atvejis	Jungtis į virtualią mašiną per VNC konsolę
Aprašymas	<p>Sistemos naudotojas atidaro virtualių mašinų sąrašą. Prie norimos virtualios mašinos paspaudžia mygtuką „VNC“. Atidaromas naujas langas, kuriame sukuriama konsolės tipo sąsaja per hipervizorių į virtualią mašiną. Mygtuko paveikslėlis pateiktas žemiau:</p>  <p style="text-align: center;">24. pav. VNC konsolės atidarymas</p>
Aktoriai	Diegėjas, palaikymo komandos narys
Prieš-sąlyga	<p>Sistemos naudotojui reikia patekti į virtualios mašinos VNC konsolę. Tai dažniausiai reikia padaryti, kai virtuali mašina nebegali pati užsikrauti ir būti pasiekama per SSH terminalą, tai gali nutikti dėl bėdų su operacinės sistemos kerneliu.</p> <p>Taip pat turi būti veikiantis Nova-ConsoleAuth komponentas, kuris užtikrina galimybes prisijungti prie virtualių mašinų per konsolę.</p>
Po-sąlyga	Atidaromas naujas langas su konsole į serverį.

22. lentelė. Duomenų eksporto į CSV formatą panaudos atvejis

Panaudojimo atvejis	Duomenų eksportas į CSV formatą
Aprašymas	<p>Sistemos naudotojas statinių duomenų puslapiuose arba OpenStack sistemos duomenų puslapiuose gali eksportuoti duomenis iš lentelių į CSV formato failą. Sąsajos paveikslėlis žemiau:</p>  <p style="text-align: center;">25. pav. Duomenų iš lentelių eksportavimo į CSV sąsaja</p> <p>Iš aukščiau pateikto paveikslėlio matome, jog galima atlikti ir daugiau veiksmų – eksportuoti į Excel, PDF formatus bei iškart spausdinti arba nukopijuoti duomenis.</p>
Aktoriai	Diegėjas, palaikymo komandos narys, analitikas
Prieš-sąlyga	Sistemos naudotojui reikia išsaugoti lentelės duomenis į atskirą failą.
Po-sąlyga	Sugeneruojamas duomenų failas ir pasiūloma naudotojui išsaugoti jį savo kompiuteryje.

23. lentelė. SiteManager sistemoje esančių statinių duomenų peržiūrėjimo panaudos atvejis

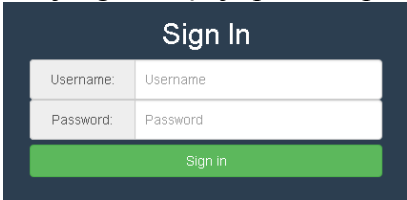
Panaudojimo atvejis	Peržiūrėti sistemoje esančius regionų, šalių, duomenų centrų, serverių spintų duomenis
----------------------------	--

Aprašymas	<p>Sistemos naudotojas iš meniu pasirenka „Staticdata“ nuorodą ir iš jos pasirenka norimos objektų grupės nuorodą. Atidaromas sąrašas. Galimi pasirinkimai paveikslėlyje žemiau:</p>  <p style="text-align: center;">26. pav. SiteManager sistemoje saugomi statiniai duomenų tipai</p>
Aktoriai	Diegėjas, palaikymo komandos narys, analitikas
Prieš-sąlyga	Sistemos naudotojui reikia peržiūrėti statinius duomenis SiteManager sistemoje.
Po-sąlyga	Atidaromas pasirinktų elementų sąrašas.

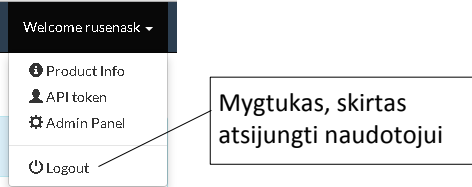
24. lentelė. SiteManager sistemos statinių duomenų redagavimo panaudos atvejis

Panaudojimo atvejis	Redaguoti duomenis
Aprašymas	Sistemos naudotojas atidaro pasirinktą elementą iš sąrašo ir jį redaguoja. Visi statiniai elementai SiteManager sistemoje yra redaguojami taip pat. Apsaugos uždėtos tik tuomet, kai informacijos redagavimas gali tiesiogiai daryti įtaką fiziniams serveriams, dėl to fizinių serverių informacijos redagavimui reikia papildomai suvesti ServiceNow galiojantį pakeitimo numerį. Siekiant išsaugoti pakeistus duomenis, sistemos naudotojas turi paspausti mygtuką „Update“.
Aktoriai	Diegėjas, palaikymo komandos narys, analitikas
Prieš-sąlyga	Naudotojas turi būti atidaręs detalią objekto informaciją ir turėti teisę keisti tą objektą. Teisių tikrinimas vykdomas tiek pačiame puslapyje JavaScript ir Django pagalba, tiek ir vidinėje programos dalyje prie metodų.
Po-sąlyga	Objekto informacija atnaujinama.

25. lentelė. Prisijungimo į SiteManager sistemą panaudos atvejis

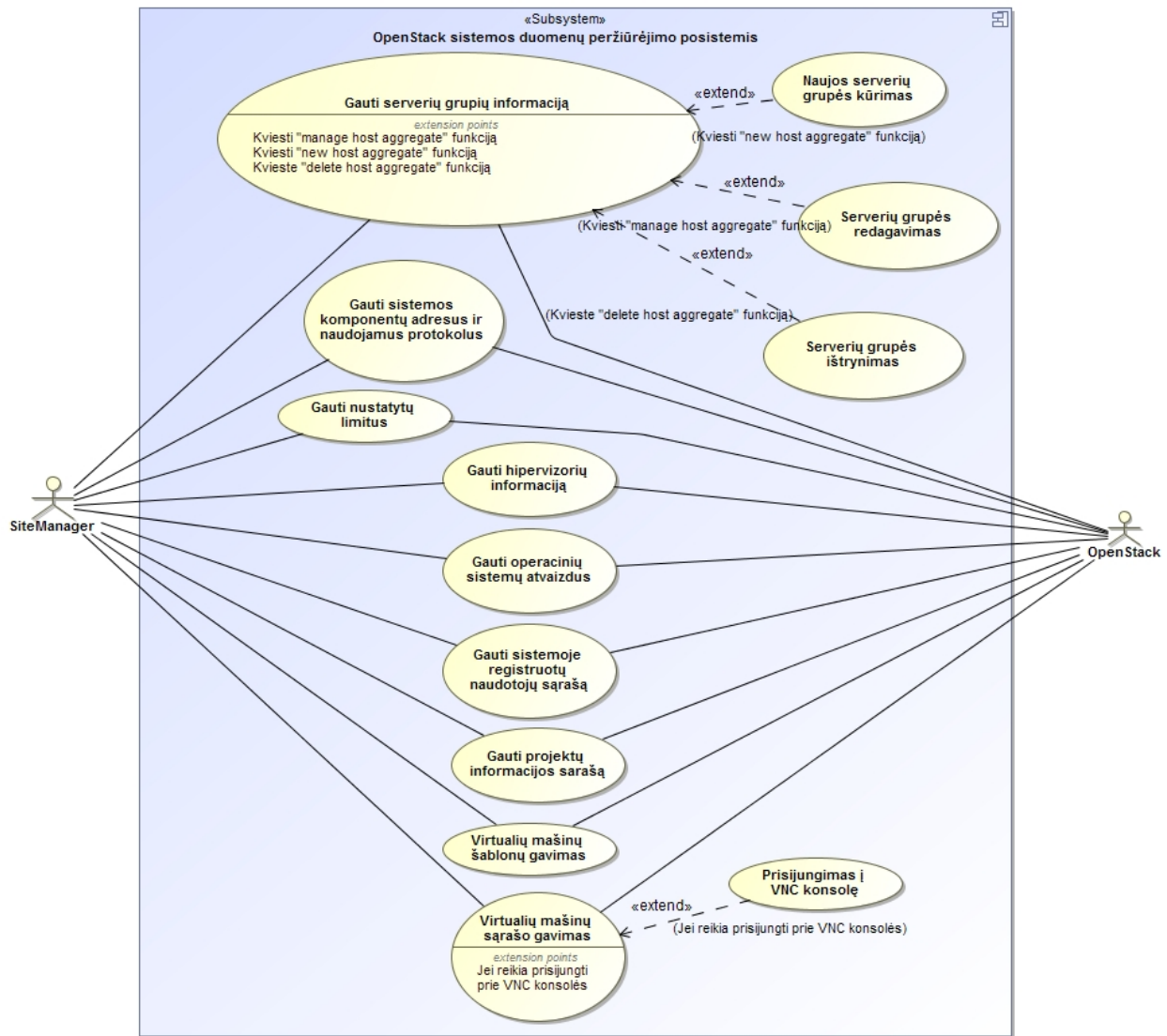
Panaudojimo atvejis	Prisijungti
Aprašymas	<p>Sistemos naudotojas įveda SiteManager adresą į naršyklę. Atidaromas prisijungimo langas. Prisijungimo sąsaja pateikta paveikslėlyje žemiau:</p>  <p style="text-align: center;">27. pav. SiteManager prisijungimo sąsaja</p>
Aktoriai	Diegėjas, palaikymo komandos narys, analitikas
Prieš-sąlyga	Naudotojas turi būti neprisijungęs
Po-sąlyga	Naudotojas prijungiamas į sistemą, sesija išsaugoma naršyklėje.

26. lentelė. Atsijungimo iš SiteManager sistemos panaudos atvejis

Panaudojimo atvejis	Atsijungti
Aprašymas	<p>Sistemos naudotojas paspaudžia ekrano viršuje, dešinėje esantį mygtuką „Welcome [naudotojo prisijungimo vardas]“ ir iš meniu pasirenka „Logout“. Meniu paveikslėlis pateiktas žemiau:</p>  <p style="text-align: center;">28. pav. Atsijungimui iš SiteManager skirtas meniu</p>
Aktoriai	Diegėjas, palaikymo komandos narys, analitikas
Prieš-sąlyga	Naudotojas turi būti atsijungęs
Po-sąlyga	Naudotojas atjungiamas nuo sistemos, jo sesija tampa negaliojanti.

3.1.2.2. SiteManager duomenų rinkimo realiu laiku iš OpenStack sistemos panaudos diagrama

Šiame skyriuje pateikta panaudos atvejų diagrama tarp veikiančių SiteManager ir OpenStack sistemų. Šia diagrama siekiama parodyti, kokie yra ryšiai tarp sistemų kai SiteManager sistemoje naudojamosi OpenStack duomenų peržiūrėjimo realiu laiku funkcijos. Diagrama pateikta paveikslėlyje žemiau:



29. pav. SiteManager ir OpenStack sistemų panaudos diagrama

Aukščiau pateiktoje diagramoje esantys panaudos atvejai nusako vidinių SiteManager sistemos funkcijų bendravimą su OpenStack sistemos komponentus per tam tikrus API. Toliau, pateikiami panaudos atvejų detalesni aprašymai.

27. lentelė. Serverių grupių informacijos gavimo panaudos atvejais

Panaudojimo atvejis	Serverių grupių informacijos gavimas
Aprašymas	Naudotojas, būdamas SiteManager sistemos OpenStack aplinkos informaciniame lange, pasirenka „Aggregates“ skirtuką. Tuomet, SiteManager, naudodamas savo vidinius prisijungimo duomenis, kreipiasi į OpenStack Nova komponento API, gauna serverių grupių sąrašą ir tuomet jį parodo vartotojui.
Aktoriai	SiteManager
Prieš-sąlyga	Naudotojui reikia peržiūrėti serverių grupių OpenStack sistemoje sąrašą.
Po-sąlyga	Serverių grupių sąrašas parodomas ekrane.

28. lentelė. Naujos serverių grupės kūrimo panaudos atvejis

Panaudojimo atvejis	Naujos serverių grupės kūrimas
Aprašymas	Naudotojas, iš SiteManager sistemos kurdamas naują OpenStack serverių grupę. Tada, SiteManager sistema nuskaito, naudotojo siūstą formą su

	pageidaujamais duomenimis. Kreipiamasi į OpenStack Nova komponentą ir per API siunčiami parametrai, pagal kuriuos reikia sukurti naują serverių grupę.
Aktoriai	SiteManager
Prieš-sąlyga	Naudotojui reikia sukurti naują OpenStack sistemoje serverių grupę.
Po-sąlyga	Naujai sukurta grupė yra parodoma serverių grupių sąrašė.

29. lentelė. Serverių grupės redagavimo panaudos atvejis

Panaudojimo atvejis	Serverių grupės redagavimas
Aprašymas	Naudotojui redaguojant serverių sąrašą OpenStack serverių grupėje, SiteManager kreipiasi į OpenStack Nova API ir gauna serverius, kurie priklauso tai grupei. Tuomet, sukuriama forma, kurioje naudotojas pasirenka, kuriuos serverius nori įtraukti arba pašalinti iš pasirinktos grupės. SiteManager pagal naują sąrašą siunčia užklausas į OpenStack sistemą ir atlieka pakeitimus serverių grupėje.
Aktoriai	SiteManager
Prieš-sąlyga	Naudotojui reikia redaguoti serverių grupės duomenis.
Po-sąlyga	Parodomas atnaujintas serverių sąrašas serverių grupėje.

30. lentelė. Serverių grupės ištrynimasis

Panaudojimo atvejis	Serverių grupės ištrynimasis
Aprašymas	Naudotojui pasirinkus OpenStack serverių grupės pašalinimą – SiteManager siunčia komandą į OpenStack sistemą dėl grupės ištrynimo.
Aktoriai	SiteManager
Prieš-sąlyga	Naudotojui reikia ištrinti OpenStack serverių grupę.
Po-sąlyga	Ištrinta grupė pašalinama iš sąrašo.

31. lentelė. Sistemos komponentų adresų ir protokolų gavimo panaudos atvejis

Panaudojimo atvejis	Sistemos komponentų adresų ir protokolų gavimas
Aprašymas	Naudotojas atidarius „Endpoints“ skirtuką – SiteManager sistema kreipiasi į OpenStack Keystone komponentą ir pareikalauja suteikti servisų katalogą. Šiame kataloge yra visų OpenStack komponentų adresai bei protokolai, kuriais galima juos pasiekti.
Aktoriai	SiteManager
Prieš-sąlyga	Naudotojui reikia gauti OpenStack komponentų adresus ir protokolus.
Po-sąlyga	Parodomas sąrašas su visais OpenStack komponentais ir kaip juos pasiekti.

32. lentelė. Nustatytų limitų gavimo panaudos atvejis

Panaudojimo atvejis	Nustatytų limitų gavimas
Aprašymas	Naudotojui atidarius pradinį OpenStack aplinkos nustatymų langą, SiteManager sistema kreipiasi į OpenStack Nova komponentą ir pareikalauja limitų sąrašo.
Aktoriai	SiteManager
Prieš-sąlyga	Naudotojui reikia OpenStack aplinkai nustatytų limitų sąrašo.
Po-sąlyga	Parodomi OpenStack aplinkos limitai.

33. lentelė. Hipervizorių informacijos gavimo panaudos atvejis

Panaudojimo atvejis	Hipervizorių informacijos gavimas
Aprašymas	Naudotojas, būdamas SiteManager sistemos OpenStack aplinkos informaciniame lange, pasirenka „Hypervisors“ skirtuką. Tuomet, SiteManager, naudodamas savo vidinius prisijungimo duomenis, kreipiasi į OpenStack Nova komponento API ir pareikalauja šių sąrašų: <ul style="list-style-type: none"> • Hipervizorių • Virtualių mašinų šablonų • Sukurtų virtualių mašinų Tada, susumuojami kiekvienam hipervizoriui tenkančių virtualių mašinų „Shares“ kiekiai. Apjungus šiuos duomenis su standartinė hipervizorių informacija (RAM, vCPU ir vidinės ilgalaikės atminties ištekliais) gaunamas galutinis duomenų paketas, kuris perduodamas į naudotojo naršyklę.
Aktoriai	SiteManager
Prieš-sąlyga	Reikia gauti hipervizorių informaciją.
Po-sąlyga	Gauta hipervizorių informacija ir apskaičiuotas išvestinis dydis „Shares“.

34. lentelė. Operacinių sistemų atvaizdų gavimo panaudos atvejis

Panaudojimo atvejis	Operacinių sistemų atvaizdų gavimas
Aprašymas	Naudotojas, būdamas SiteManager sistemos OpenStack aplinkos informaciniame lange, pasirenka „Images“ skirtuką. Tada, SiteManager kreipiasi į OpenStack Glance komponentą ir iš jo gauna atvaizdų sąrašą. Taip pat, kreipiasi į OpenStack Keystone komponentą ir iš jo gauna projektų (OpenStack sistemoje žinomų kaip „Tenants“) sąrašą. Turint šiuos sąrašus kiekvienam atvaizdui priskiriamas projektas, kuriame jis buvo sukurtas (tokiu atveju skirtingi projektai gali matyti tik jų pačių paruoštus operacinės sistemos atvaizdus).
Aktoriai	SiteManager
Prieš-sąlyga	Reikia atvaizduoti operacinės sistemos atvaizdų sąrašą.
Po-sąlyga	Sąrašas sukurtas, galima parodyti naudotojo naršyklėje.

35. lentelė. Sistemoje registruotų naudotojų sąrašo gavimo panaudos atvejis

Panaudojimo atvejis	Sistemoje registruotų naudotojų sąrašo gavimas
Aprašymas	Naudotojas, būdamas SiteManager sistemos OpenStack aplinkos informaciniame lange, pasirenka „Users“ skirtuką. Tada, SiteManager sistema kreipiasi į OpenStack Keystone komponentą ir iš jo gauna vidinių OpenStack naudotojų sąrašą.
Aktoriai	SiteManager
Prieš-sąlyga	Naudotojui reikia vidinių OpenStack sistemos naudotojų sąrašo.
Po-sąlyga	Vidinių OpenStack sistemos naudotojų sąrašas gautas, galima parodyti naršyklėje.

36. lentelė. Projektų informacijos sąrašo gavimo panaudos atvejis

Panaudojimo atvejis	Projektų informacijos sąrašo gavimas
Aprašymas	Naudotojas, būdamas SiteManager sistemos OpenStack aplinkos informaciniame lange, pasirenka „Tenants“ skirtuką. SiteManager kreipiasi į OpenStack Nova ir Keystone komponentus, gaunami du sąrašai – projektų sąrašas ir projektų resursų išnaudojimo sąrašas. Tada, pagal

	projekto identifikacinį numerį surandami jo išnaudojami resursai.
Aktoriai	SiteManager
Prieš-sąlyga	Naudotojui reikia OpenStack sistemoje esančių projekto sąrašo.
Po-sąlyga	Vidinių projektų sąrašas gautas ir priskirti jo išnaudoti resursai.

37. lentelė. Virtualių mašinų šablonų gavimo panaudos atvejis

Panaudojimo atvejis	Virtualių mašinų šablonų gavimas
Aprašymas	Naudotojas, būdamas SiteManager sistemos OpenStack aplinkos informaciniame lange, pasirenka „Flavors“ skirtuką. SiteManager kreipiasi į OpenStack Nova komponentą ir gauna virtualių mašinų šablonų sąrašą. Šiam sąrašui dar reikia pridėti naują išvestinį dydį „Shares“, todėl kiekvienam sąrašo elementui jis pridedamas.
Aktoriai	SiteManager
Prieš-sąlyga	Naudotojui reikia virtualių mašinų šablonų sąrašo.
Po-sąlyga	Sąrašas gautas iš OpenStack ir papildytas reikiamais duomenimis.

38. lentelė. Virtualių mašinų sąrašo gavimo panaudos atvejis

Panaudojimo atvejis	Virtualių mašinų sąrašo gavimas
Aprašymas	<p>Naudotojas, būdamas SiteManager sistemos OpenStack aplinkos informaciniame lange, pasirenka „Instances“ skirtuką. Tada, SiteManager kreipiasi į OpenStack Keystone, Nova, Glance komponentus ir iš jų gauna šiuos sąrašus:</p> <ul style="list-style-type: none"> • Virtualių mašinų sąrašą. • Virtualių mašinų šablonų sąrašą. • Išorinių ir vidinių sistemos naudotojų sąrašą. • OpenStack projektų sąrašą. • Operacinių sistemų atvaizdų sąrašą. <p>Turint šiuos sąrašus atliekami šie veiksmai:</p> <ul style="list-style-type: none"> • Pagal virtualios mašinos šablono identifikacinį numerį surandamas atitinkamo šablonų sąrašas ir pridedami nauji parametrai į virtualių mašinų sąrašą: šablono pavadinimas ir šablono detalės. • Pagal naudotojo identifikacinį numerį nustatoma ar tai vidinis, ar išorinis, o tada priskiriama tikroji reikšmė (nebe identifikacinis numeris iš atsitiktinių skaičių ir raidžių). • Pagal projekto identifikacinį numerį priskiriamas projekto pavadinimas prie virtualios mašinos aprašymo. • Pagal operacinės sistemos atvaizdo numerį sukuriama naujas parametras, kuris turi operacinės sistemos atvaizdo pavadinimą. <p>Turint papildytą reikiama informaciją virtualių mašinų sąrašą – SiteManager sistema jį perduoda sistemos naudotojui.</p>
Aktoriai	SiteManager
Prieš-sąlyga	Naudotojui reikia virtualių mašinų sąrašo.
Po-sąlyga	Sąrašas gautas iš OpenStack sistemos gavus reikalingą informaciją ir ją apjungus.

39. lentelė. Prisijungimo į VNC konsolę panaudos atvejis

Panaudojimo atvejis	Prisijungimas į VNC konsolę
----------------------------	-----------------------------

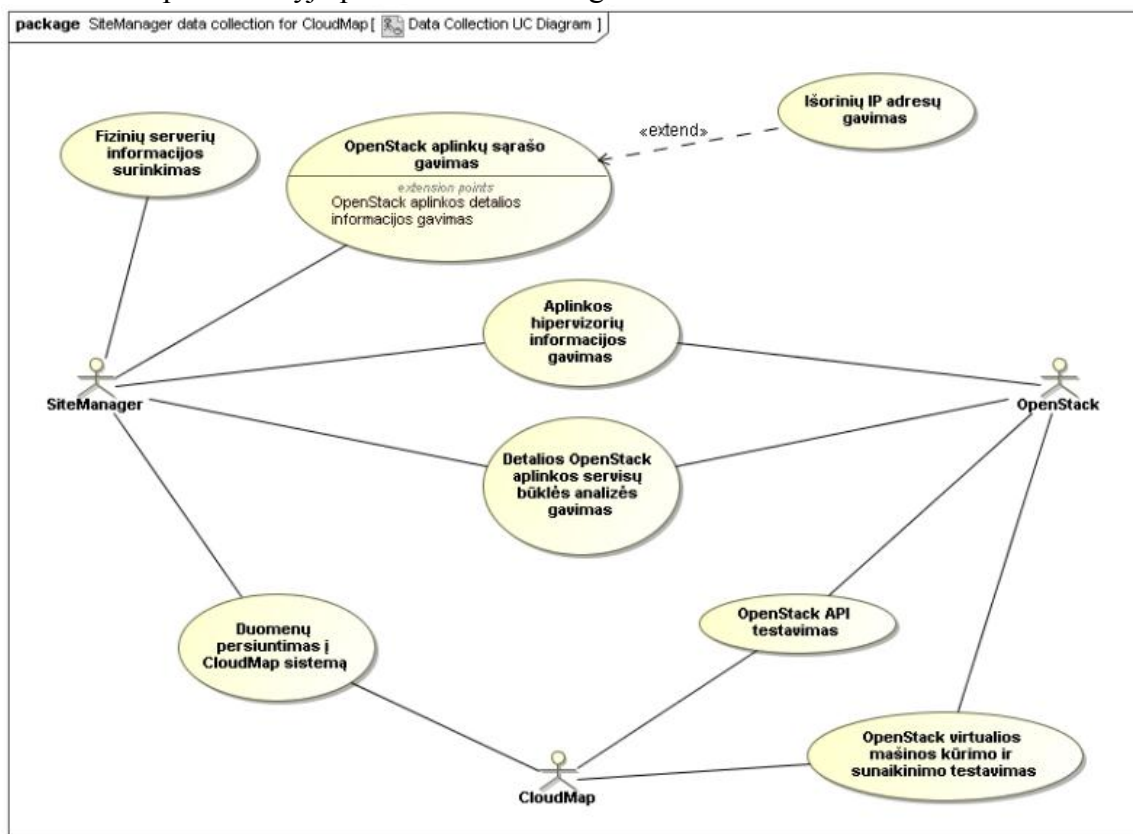
Aprašymas	Naudotojas atidaro SiteManager sistemoje virtualių mašinų sąrašą. Sąrašo krovimo metu, SiteManager sistema formuodama galutinį virtualių mašinų sąrašą, kreipiasi į OpenStack Nova komponentą ir kiekvienai mašinai kuria unikalų raktą, su kuriuo bus galima patekti į virtualią mašiną per hipervizorių.
Aktoriai	SiteManager
Prieš-sąlyga	Naudotojui reikia prisijungti į VNC konsolę.
Po-sąlyga	SiteManager sistema gauna prisijungimo raktą į kiekvieną virtualią mašiną, kuri sistemos naudotojas gali panaudoti jungiantis į konsolę tiesiai iš naršyklės.

3.1.2.3. SiteManager, CloudMap ir OpenStack tarpusavio komunikacijų panaudos atvejų diagrama

Šiame skyriuje aprašomi panaudos atvejai, nusakantys SiteManager, CloudMap ir OpenStack sistemų tarpusavio veikimą. SiteManager sistema, naudodama Celery ir Celerybeat sistemas, nustatytais intervalais kreipiasi į OpenStack, surenka duomenis ir persiunčia į CloudMap sistemą. CloudMap, turėdama informaciją apie OpenStack sistemas – gali paleisti integracinius testus. Pagrindiniai SiteManager veiksmai, atliekami ruošiant duomenis persiuntimui į CloudMap sistemą:

- Fizinių serverių informacijos surinkimas.
- Sukurtų OpenStack aplinkų sąrašo sudarymas.
- Kiekvienai OpenStack aplinkai atliekami veiksmai:
 - Hipervizorių informacijos gavimas.
 - Detalios OpenStack aplinkos servisų būklės analizė (gaunami visi Novos ir Keystone servais, Swift komponento būklė netikrinama).
- Duomenų persiuntimas į CloudMap sistemą.

Sekančiame paveikslėlyje pateikta veiklos diagrama:



30. pav. SiteManager, CloudMap ir OpenStack sistemų tarpusavio komunikacijos panaudos atvejų diagrama

Iš diagramos matome, jog SiteManager ir CloudMap sistemos sąveikauja su OpenStack sistema, tik skirtingais lygmenimis. Toliau lentelėse pateikiami detalizuoti panaudos atvejai:

40. lentelė. Fizinių serverių informacijos surinkimo panaudos atvejais

Panaudojimo atvejis	Fizinių serverių informacijos surinkimas
Aprašymas	SiteManager kreipiasi į duomenų bazę ir filtruoja serverius pagal jų tipus. Gautą sąrašą sudaro: <ul style="list-style-type: none"> • Informacijai saugoti skirtų serverių kiekis

	<ul style="list-style-type: none"> • Skaičiavimo serverių kiekis • Valdymo serverių kiekis • Visų turimų serverių kiekis
Aktoriai	SiteManager
Prieš-sąlyga	SiteManager sistema ruošia duomenis persiuntimui į CloudMap
Po-sąlyga	Paruošta serverių informacija

41. lentelė. OpenStack aplinkų sąrašo gavimo panaudos atvejis

Panaudojimo atvejis	OpenStack aplinkų sąrašo gavimas
Aprašymas	SiteManager kreipiasi į duomenų bazę ir gauna visas turimas OpenStack aplinkas. Net ir neaktyvi aplinka patenka į šį sąrašą, nes aplinkos diegimas užtrunka gana ilgą laiką, taigi iš CloudMap sistemos atsiranda galimybė stebėti ir įrašymo procesą. Taip pat kartu su aplinkų sąrašu gaunami ir papildomi duomenys, tokie kaip IP adresai, pavadinimai, duomenų centrai.
Aktoriai	SiteManager
Prieš-sąlyga	SiteManager sistema ruošia duomenis persiuntimui į CloudMap
Po-sąlyga	Paruoštas OpenStack aplinkų sąrašas

42. lentelė. Aplinkos hipervizorių informacijos gavimo panaudos atvejis

Panaudojimo atvejis	Aplinkos hipervizorių informacijos gavimas
Aprašymas	SiteManager, naudodamas savo vidinius prisijungimo duomenis, kreipiasi į OpenStack Nova komponento API ir pareikalauja šių sąrašų: <ul style="list-style-type: none"> • Hipervizorių • Virtualių mašinų šablonų • Sukurtų virtualių mašinų Tada, susumuojami kiekvienam hipervizoriui tenkančių virtualių mašinų „Shares“ kiekiai. Apjungus šiuos duomenis su standartinė hipervizorių informacija (RAM, vCPU ir vidinės ilgalaikės atminties ištekliais) gaunamas galutinis duomenų paketas.
Aktoriai	SiteManager
Prieš-sąlyga	SiteManager sistema ruošia duomenis persiuntimui į CloudMap
Po-sąlyga	Paruošti OpenStack hipervizorių duomenys

43. lentelė. Detalios OpenStack aplinkos servisų būklės analizės gavimo panaudos atvejis

Panaudojimo atvejis	Detalios OpenStack aplinkos servisų būklės analizės gavimas
Aprašymas	SiteManager kreipiasi į OpenStack Keystone komponento API, gauna visų servisų sąrašą. Tada, priklausomai nuo serviso tipo, naudoja tam tikrus sistemos modulius, kad galėtų komunikuoti su skirtingais OpenStack komponentais ir tikrinti, ar jie atsako į užklausas. Radus Nova komponentą – jis tikrinamas dar giliau ir kreipiantis tiesiogiai į jį, gaunamas detalus Nova servisų sąrašas (jų gali būti nuo dešimties iki kelių tūkstančių arba daugiau) ir šių servisų būklė. Tikrinant servisu – formuojamas sąrašas.
Aktoriai	SiteManager
Prieš-sąlyga	SiteManager sistema ruošia duomenis persiuntimui į CloudMap
Po-sąlyga	Paruošti OpenStack servisų būklės duomenys

44. lentelė. Duomenų persiuntimo į CloudMap sistemą panaudos atvejais

Panaudojimo atvejis	Duomenų persiuntimas į CloudMap sistemą
Aprašymas	SiteManager sistema konvertuoja turimus duomenis į JSON formatą ir persiunčia juos į CloudMap per HTTP užklausą.
Aktoriai	SiteManager
Prieš-sąlyga	SiteManager sistema turi paruoštus duomenis persiuntimui.
Po-sąlyga	Duomenys nusiųsti į CloudMap, gauto atsakymo numeris 200.

45. lentelė. OpenStack API testavimo panaudos atvejais

Panaudojimo atvejis	OpenStack API testavimas
Aprašymas	CloudMap kreipiasi į nurodytos aplinkos išorini IP adresą ir komunikuoja su OpenStack Nova komponentu. Siunčiama užklausa gauti virtualių mašinų šablonų sąrašą bei pradedamas matuoti laikas, per kiek atsakys. Gavus atsakymą, duomenys užfiksuojami.
Aktoriai	CloudMap
Prieš-sąlyga	CloudMap sistemoje turi būti inicijuotas testas
Po-sąlyga	Gauti testo rezultatai

46. lentelė. OpenStack virtualios mašinos kūrimo ir sunaikinimo testavimo panaudos atvejais

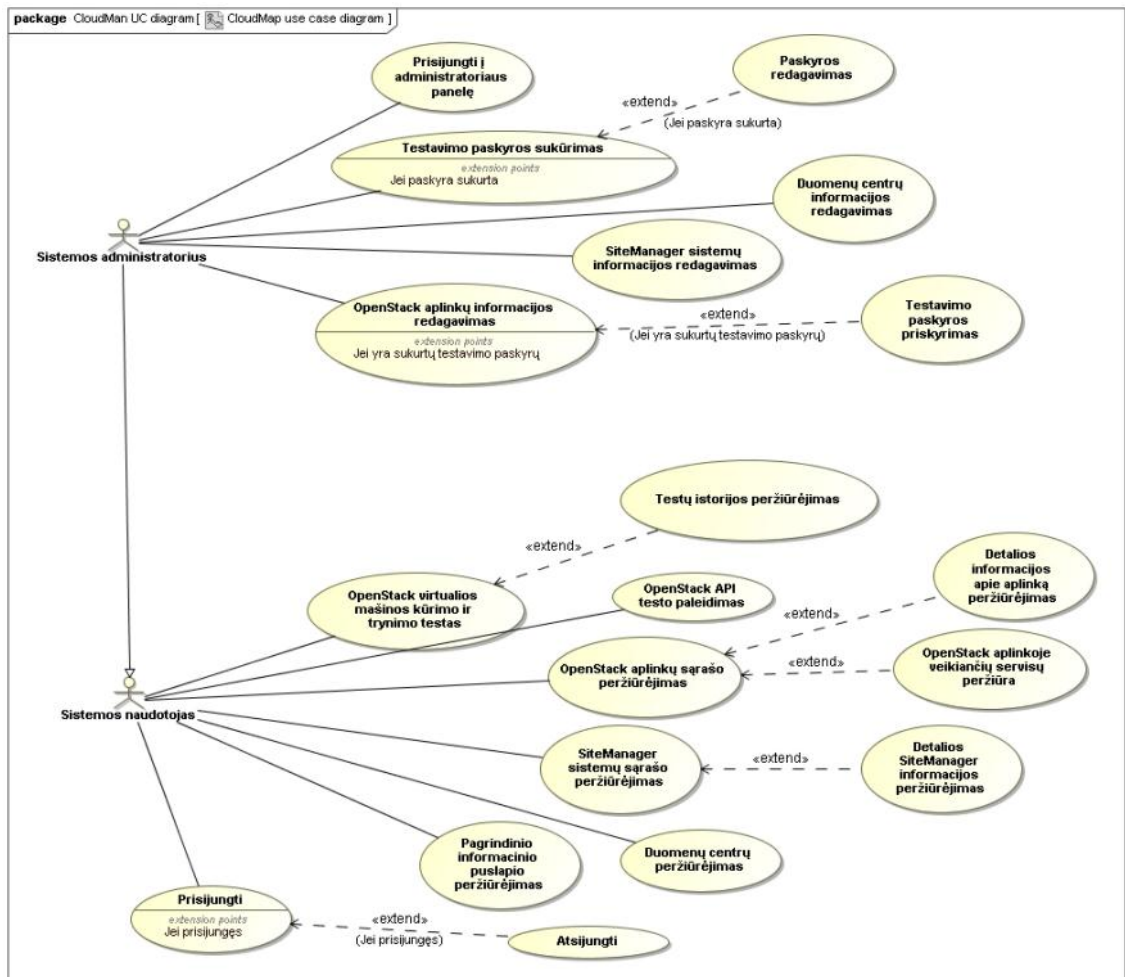
Panaudojimo atvejis	OpenStack virtualios mašinos kūrimo ir sunaikinimo testavimas
Aprašymas	CloudMap kreipiasi į nurodytos aplinkos išorini IP adresą ir komunikuoja su OpenStack Nova, Glance komponentais. Gaunami visi reikalingi parametrai virtualios mašinos sukūrimui. Sukuriama mašina, patikrinama ar ji pasiekė aktyvią būseną, tuomet ištrinama.
Aktoriai	SiteManager
Prieš-sąlyga	CloudMap sistemoje turi būti inicijuotas testas
Po-sąlyga	Testo rezultatai įrašomi duomenų bazėje

3.1.3. CloudMap sistemos panaudos atvejų diagrama

Šis skyrius skirtas parodyti sistemos naudotojų sąveiką su CloudMap sistema. Visi veiksmai vykdomi per naršyklę. Šios sistemos naudotojai skirstomi į dvi grupes: paprasti sistemos naudotojai ir sistemos administratorius. Įprastiniai sistemos naudotojai galės tik peržiūrėti informaciją, nes jiems nereikia kurti ar įvedinėti jokių duomenų patiems – jeigu SiteManager sistemų nustatymuose yra teisingai sukonfigūruotas CloudMap adresas, tuomet duomenys į šią sistemą turi būti atsiųsti automatiškai. Yra tik keli atvejai, kada naudotojui reikia prisijungti su administratoriaus paskyra:

1. OpenStack aplinka įdiegiama naujame, dar nenumatyname duomenų centre.
2. Naudotojas nori pakeisti duomenų centro informacinės eilutės poziciją žemėlapyje.
3. Įdiegus CloudMap sistemą reikia sukurti ir OpenStack aplinkoms priskirti testavimo paskyras (jos turės savo numatytus projektus sistemos viduje ir jiems priskirtus resursus, kurių užteks sukurti kelioms virtualioms mašinoms).
4. Naudotojui reikia priskirti specifinius pavadinimus SiteManager sistemoms.

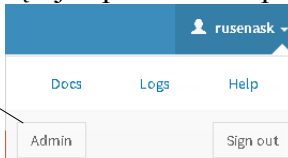
Žemiau pateikta CloudMap sistemos panaudos atvejų diagrama:



31. pav. CloudMap sistemos panaudos atvejų diagrama

Iš šios diagramos matome, jog pagrindinis šios sistemos tikslas – konsoliduoti informaciją ir ją pateikti sistemos naudotojui. Dėl šio pagrindinio tikslo yra skiriama daug laiko grafinės sąsajos tobulinimui, ko nereikėjo kuriant SiteManager sistemą. Sekančiose lentelėse detalizuojami CloudMap panaudos atvejai:

47. lentelė. Prisijungimo į administratoriaus panelę panaudos atvejis

Panaudojimo atvejis	Prisijungti į administratoriaus panelę
Aprašymas	Naudotojas paspaudžia ekrano viršuje, dešinėje esantį mygtuką. Iššokusioje lentelėje paspaudžia mygtuką „Admin“ ir patenka į administratoriaus panelę. Sąsajos paveikslėlis pateiktas žemiau: <div style="display: flex; align-items: center; justify-content: center;"> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">Prisijungimas į administratoriaus panelę</div>  </div>
	32. pav. CloudMap prisijungimo į administratoriaus panelę sąsaja
Aktoriai	Sistemos administratorius
Prieš-sąlyga	Naudotojo paskyra turi turėti administratoriaus privilegijas
Po-sąlyga	Įjungtama administratoriaus panelė

48. lentelė. Testavimo paskyros sukūrimo panaudos atvejis

Panaudojimo atvejis	Testavimo paskyros sukūrimas
----------------------------	------------------------------

Aprašymas	Administratorius, prisijungęs į administracinę panelę, pasirenka „Environment testing accounts“, tuomet paspaudžia „Add environment testing account“ ir suveda duomenis. Įvedus duomenis, spaudžia mygtuką „Save“.
Aktoriai	Sistemos administratorius
Prieš-sąlyga	Sistemos administratoriui reikia sukurti testavimo paskyrą
Po-sąlyga	Testavimo paskyra sukurta ir gali būti priskirta OpenStack aplinkoms

49. lentelė. Testavimo paskyros redagavimo panaudos atvejis

Panaudojimo atvejis	Paskyros redagavimas
Aprašymas	Administratorius, prisijungęs į administracinę panelę iš testavimo paskyrų sąrašo pasirenka paskyrą, tuomet atsidariusiame lange redaguoja duomenis ir tai atlikęs spaudžia mygtuką „Save“.
Aktoriai	Sistemos administratorius
Prieš-sąlyga	Sistemos administratoriui reikia redaguoti testavimo paskyros duomenis
Po-sąlyga	Testavimo paskyros duomenys redaguoti, OpenStack testavimo funkcijos naudoja atnaujintus duomenis

50. lentelė. Duomenų centrų informacijos redagavimo panaudos atvejis

Panaudojimo atvejis	Duomenų centrų informacijos redagavimas
Aprašymas	Administratorius, prisijungęs į administracinę panelę įeina į duomenų centrų sąrašą, tada pasirenka norimą duomenų centrą ir atsidariusiame lange redaguoja duomenis ir tai atlikęs spaudžia mygtuką „Save“.
Aktoriai	Sistemos administratorius
Prieš-sąlyga	Sistemos administratoriui reikia pakeisti duomenų centro duomenis
Po-sąlyga	Duomenų centro duomenys pakeisti

51. lentelė. SiteManager sistemų informacijos redagavimo panaudos atvejis

Panaudojimo atvejis	SiteManager sistemų informacijos redagavimas
Aprašymas	Administratorius, prisijungęs į administracinę panelę įeina į SiteManager sistemų sąrašą, tada pasirenka norimą SiteManager ir atsidariusiame lange redaguoja duomenis ir tai atlikęs spaudžia mygtuką „Save“.
Aktoriai	Sistemos administratorius
Prieš-sąlyga	Sistemos administratoriui reikia pakeisti SiteManager sistemos duomenis, pavyzdžiui suteikti jam pavadinimą
Po-sąlyga	SiteManager sistemos duomenys pakeisti

52. lentelė. OpenStack aplinkų informacijos redagavimo panaudos atvejis

Panaudojimo atvejis	OpenStack aplinkų informacijos redagavimas
Aprašymas	Administratorius, prisijungęs į administracinę panelę įeina į OpenStack aplinkų sąrašą, tada pasirenka norimą OpenStack aplinką ir atsidariusiame lange redaguoja duomenis ir tai atlikęs spaudžia mygtuką „Save“.
Aktoriai	Sistemos administratorius
Prieš-sąlyga	Sistemos administratoriui reikia pakeisti OpenStack aplinkos duomenis CloudMap sistemoje
Po-sąlyga	Aplinkos duomenys pakeisti

53. lentelė. Testavimo paskyros priskyrimo panaudos atvejis

Panaudojimo atvejis	Testavimo paskyros priskyrimas
Aprašymas	Administratorius, prisijungęs į administracinę panelę įeina į OpenStack aplinkų sąrašą, tada pasirenka norimą OpenStack aplinką ir atsidariusiame lange prie „OpenStack testing account“ laukelio iš sąrašo pasirenka norimą testavimo paskyrą ir tada spaudžia mygtuką „Save“.
Aktoriai	Sistemos administratorius
Prieš-sąlyga	Sistemos administratoriui reikia priskirti testavimo aplinką prie OpenStack aplinkos, kad būtų galima vykdyti testus
Po-sąlyga	Paskyra priskirta, testai šiai aplinkau naudos priskirtą testavimo paskyrą

54. lentelė. OpenStack virtualios mašinos kūrimo ir trynimo testo panaudos atvejis

Panaudojimo atvejis	OpenStack virtualios mašinos kūrimo ir trynimo testas
Aprašymas	Sistemos naudotojas kairėje esančiame meniu paspaudžia „OpenStack testing“, išsiskleidusiame meniu pasirenka „Instance creation“. Tuomet, pasirenka OpenStack aplinką, kurią nori testuoti ir spaudžia mygtuką „Launch“.
Aktoriai	Sistemos administratorius, sistemos naudotojas
Prieš-sąlyga	Sistemos naudotojas nori atlikti virtualios mašinos kūrimo testą OpenStack aplinkoje.
Po-sąlyga	Testas pradedamas, priklausomai nuo sistemos apkrovimo bei nustatytų taisyklių gali kisti

55. lentelė. Testų istorijos peržiūrėjimo panaudos atvejis

Panaudojimo atvejis	Testų istorijos peržiūrėjimas
Aprašymas	Sistemos naudotojas kairėje esančiame meniu paspaudžia „OpenStack testing“, išsiskleidusiame meniu pasirenka „Instance creation“. Tuomet, pasirenka OpenStack aplinką, kurios istoriją nori peržiūrėti ir spaudžia mygtuką „History“.
Aktoriai	Sistemos administratorius, sistemos naudotojas
Prieš-sąlyga	Sistemos naudotojui reikia peržiūrėti ankstesnių testų rezultatus
Po-sąlyga	Rezultatai parodomi ekrane, pagrindinė informaciją, kurią reikia parodyti: <ul style="list-style-type: none"> • Paskutinio sėkmingo testo data • Paskutinio nesėkmingo testo data • Paskutinio testo trukmė • Vidutinė testo trukmė

56. lentelė. OpenStack API testo paleidimo panaudos atvejis

Panaudojimo atvejis	OpenStack API testo paleidimas
Aprašymas	Sistemos naudotojas kairėje esančiame meniu paspaudžia „OpenStack testing“, išsiskleidusiame meniu pasirenka „API response time“. Tuomet, pasirenka OpenStack aplinką, kurią nori testuoti ir spaudžia mygtuką „Launch“.
Aktoriai	Sistemos administratorius, sistemos naudotojas
Prieš-sąlyga	Sistemos naudotojui reikia atlikti OpenStack aplinkos API atsakymui gauti reikalingo laiko testą
Po-sąlyga	Atliekamas testas, į naudotojo ekraną išvedama testo baigtis (sėkmingas arba nesėkmingas) ir laikas, per kurį gavo atsakymą. Testas gali būti laikomas nesėkmingu, jeigu per nustatytą laiką negaunamas atsakymas

	arba gaunamas blogas atsakymas (blogi prisijungimo duomenys arba nepasiekiamas API)
--	---

57. lentelė. OpenStack aplinkų sąrašo peržiūrėjimo panaudos atvejis

Panaudojimo atvejis	OpenStack aplinkų sąrašo peržiūrėjimas
Aprašymas	Sistemos naudotojas paspaudžia pele ant kairėje pusėje esančio meniu elemento „Environments“, išsiskleidžia visų OpenStack aplinkų sąrašas.
Aktoriai	Sistemos administratorius, sistemos naudotojas
Prieš-sąlyga	Sistemos naudotojui reikia peržiūrėti OpenStack aplinkų sąrašą.
Po-sąlyga	Sąrašas parodytas

58. lentelė. Detalios informacijos apie aplinką peržiūrėjimo panaudos atvejis

Panaudojimo atvejis	Detalios informacijos apie aplinką peržiūrėjimas
Aprašymas	Sistemos naudotojas paspaudžia pele ant kairėje pusėje esančio meniu elemento „Environments“, išsiskleidžia visų OpenStack aplinkų sąrašas. Tuomet, pasirinkęs norimą aplinką jis paspaudžia ant jos ir atsidaro detalus aprašas su hipervizorių informacija.
Aktoriai	Sistemos administratorius, sistemos naudotojas
Prieš-sąlyga	Sistemos naudotojui reikia peržiūrėti detalią OpenStack aplinkos informaciją
Po-sąlyga	Parodoma detali OpenStack aplinkos informacija

59. lentelė. OpenStack aplinkoje veikiančių servisų peržiūrėjimo panaudos atvejis

Panaudojimo atvejis	OpenStack aplinkoje veikiančių servisų peržiūra
Aprašymas	Sistemos naudotojas paspaudžia pele ant kairėje pusėje esančio meniu elemento „Environments“, išsiskleidžia visų OpenStack aplinkų sąrašas. Tuomet, pasirinkęs norimą aplinką jis paspaudžia ant jos ir atsidaro detalus aprašas su hipervizorių informacija. Žemiau, parodoma lentelė su šioje aplinkoje veikiančiais servisais.
Aktoriai	Sistemos administratorius, sistemos naudotojas
Prieš-sąlyga	Sistemos naudotojui reikia peržiūrėti aplinkoje veikiančius servisus
Po-sąlyga	Parodoma lentelė su servisais, jų būklėmis, paskutinis būklės pasikeitimo laikas bei serveris (jeigu tai iš Nova komponento), kuriame jis veikia.

60. lentelė. SiteManager sistemų sąrašo peržiūrėjimo panaudos atvejis

Panaudojimo atvejis	SiteManager sistemų sąrašo peržiūrėjimas
Aprašymas	Sistemos naudotojas paspaudžia pele ant kairėje pusėje esančio meniu elemento „Sitemanagers“, išsiskleidžia visų SiteManager sistemų sąrašas.
Aktoriai	Sistemos administratorius, sistemos naudotojas
Prieš-sąlyga	Sistemos naudotojui reikia peržiūrėti SiteManager sistemų sąrašą
Po-sąlyga	Parodomas SiteManager sistemų sąrašas.

61. lentelė. Detalios SiteManager informacijos peržiūrėjimo panaudos atvejis

Panaudojimo atvejis	Detalios SiteManager informacijos peržiūrėjimas
Aprašymas	Sistemos naudotojas paspaudžia pele ant kairėje pusėje esančio meniu elemento „Sitemanagers“, išsiskleidžia visų SiteManager sistemų sąrašas. Tuomet, pasirinkęs norimą SiteManager - jis paspaudžia ant jos ir atsidaro

	detalus aprašas su fizinių serverių kiekiais (pagal tipus), duomenų centras, kuriame jis veikia bei OpenStack aplinkos, kurios sukurtos šiame SiteManager.
Aktoriai	Sistemos administratorius, sistemos naudotojas
Prieš-sąlyga	Sistemos naudotojui reikia peržiūrėti detalią SiteManager sistemos informaciją
Po-sąlyga	Parodoma detali SiteManager sistemos informacija

62. lentelė. Duomenų centrų peržiūrėjimo panaudos atvejis

Panaudojimo atvejis	Duomenų centrų peržiūrėjimas
Aprašymas	Sistemos naudotojas paspaudžia pele ant kairėje pusėje esančio meniu elemento „Datacenters“, išsiskleidžia visų duomenų centrų sąrašas. Vartotojui, paspaudus ant bet kurio duomenų centro – atidaromas detalusis duomenų centrų sąrašas, kuriame yra nurodomas duomenų centro kodas, koordinatės bei registruotos SiteManager sistemos juose.
Aktoriai	Sistemos administratorius, sistemos naudotojas
Prieš-sąlyga	Sistemos naudotojui reikia peržiūrėti duomenų centrų sąrašą
Po-sąlyga	Parodomas duomenų centrų sąrašas

63. lentelė. Pagrindinio informacinio puslapio peržiūrėjimo panaudos atvejis


Panaudojimo atvejis	Pagrindinio informacinio puslapio peržiūrėjimas
Aprašymas	Naudotojas, sėkmingai prisijungęs prie sistemos yra automatiškai nukreipiamas į pagrindinį informacijos puslapį. Jeigu jis yra kitame lange – į jį gali pereiti paspaudęs „Dashboard“ mygtuką, kurį galima rasti kairėje esančiame meniu.
Aktoriai	Sistemos administratorius, sistemos naudotojas
Prieš-sąlyga	Sistemos naudotojui reikia peržiūrėti pagrindinį informacinį puslapį
Po-sąlyga	Parodomas pagrindinis informacijos puslapis, jame galima rasti: <ul style="list-style-type: none"> • Surinktų raportų kiekį • Sėkmingų testų rodiklį • Veikiančių servisų kiekį • Neveikiančių servisų kiekį • Žemėlapi su visais žinomais duomenų centrais ir juose veikiančių ir neveikiančių servisų kiekiais.

64. lentelė. Prisijungimo panaudos atvejis

Panaudojimo atvejis	Prisijungti
Aprašymas	Sistemos naudotojas įveda SiteManager adresą į naršyklę. Atidaromas prisijungimo langas.
Aktoriai	Sistemos administratorius, sistemos naudotojas
Prieš-sąlyga	Naudotojas turi būti neprisijungęs
Po-sąlyga	Naudotojas prijungiamas į sistemą, sesija išsaugoma naršyklėje.

65. lentelė. Atsijungimo panaudos atvejis

Panaudojimo atvejis	Atsijungti
Aprašymas	Sistemos naudotojas paspaudžia ekrano viršuje, dešinėje esantį mygtuką „[naudotojo prisijungimo vardas]“ ir iš meniu pasirenka „Sign out“. Meniu paveikslėlis pateiktas žemiau:

	 <p style="text-align: center;">33. pav. Atsijungimui iš SiteManager skirtas meniu</p>
Aktoriai	Sistemos administratorius, sistemos naudotojas
Prieš-sąlyga	Naudotojas turi būti atsijungęs
Po-sąlyga	Naudotojas atjungiamas nuo sistemos, jo sesija tampa negaliojanti.

3.2. Sistemų nefunkciniai reikalavimai

Nefunkciniai reikalavimai keliami SiteManager ir CloudMap sistemoms:

a) Reikalavimai techniniai įrangai:

- Serveryje, kuriame bus diegiamos sistemos, turi būti RedHat 6.5 operacinė sistema su jai priklausančiais standartiniais paketais.
- Serveryje taip pat turi būti galimybė diegti paketus iš nurodytų repozitorijų.
- Serveryje turi būti ryšys su Python PyPi paketų repozitorija.

b) Reikalavimai eksperimentui:

- CloudMap sistema turi būti pasiekama iš SiteManager sistemų per tinklą HTTP protokolu.
- Fiziniai serveriai, kuriuos norima pridėti į SiteManager duomenų bazę turi būti pasiekiami per tinklą.
- Siekiant išnaudoti valdymo per IPMI sąsają privalumus – serveriai turi būti HP gamintojo su ILO sąsajomis, o jų palaikoma versija turi būti naujesnė už 2.0.

c) Reikalavimai CloudMap sistemai:

- Pagrindinis informacinis langas turi turėti žemėlapi, kuriame atvaizduojami duomenų centrai ir OpenStack sistemų būklė juose.
- Pagrindinis informacinis langas turi gerai atrodyti dideliame ekrane (žemėlapio atvaizdavimui reikėtų naudoti vektorinę grafiką).
- Duomenys pagrindiniame informaciniame lange turi automatiškai atsinaujinti, kad nereiktų perkrauti naršyklės lango.

3.3. Reikalavimų apibendrinimas

CloudMap ir SiteManager sistemos yra įrankiai, kurie automatizuoja tam tikrus veiksmus arba suteikia greitesnę prieigą prie duomenų. Šioms sistemoms keliami reikalavimai (nedetalizuojant paties diegimo), kuriuos galutinis produktas turi tenkinti:

- Galimybė rinkti fizinių serverių informaciją ir nuotoliniu būdu juos valdyti
- Inicijuoti OpenStack sistemos diegimą
- Stebėti OpenStack aplinkos parametrus
- Prie sistemų turi būti galimybė jungtis su Active Directory paskyromis
- Turi būti galimybė atnaujinti programinį kodą po įdiegimo išsaugant duomenų bazėje esančius duomenis
- Turėti prieigą prie visų SiteManager sistemų sąrašo ir matyti jų duomenis
- Sukurti sąsają, kurioje būtų pasaulio žemėlapis ir jame išdėstyti duomenų centrai

4. CLOUDMAP IR SITEMANAGER SPRENDIMO PROJEKTAS

Sistemų pavadinimai ir vartotojo sąsajose naudojama anglų kalba, nes jomis naudojasi darbuotojai iš JAV, Jungtinės Karalystės, Indijos ir Singapūro.

Reikalavimams patenkinti nuspręsta sukurti dvi sistemas, todėl, esant poreikiui, – visada galima vieną iš jų pakeisti kitu produktu. Šio darbo kūrimo metu buvo planuojama SiteManager sistema, anksčiau ar vėliau, pakeisti kitu atviro kodo įrankiu, pvz. Redhat Tuskar, jei tai pasidarys reikalinga ir toks įrankis geriau tenkins reikalavimus.

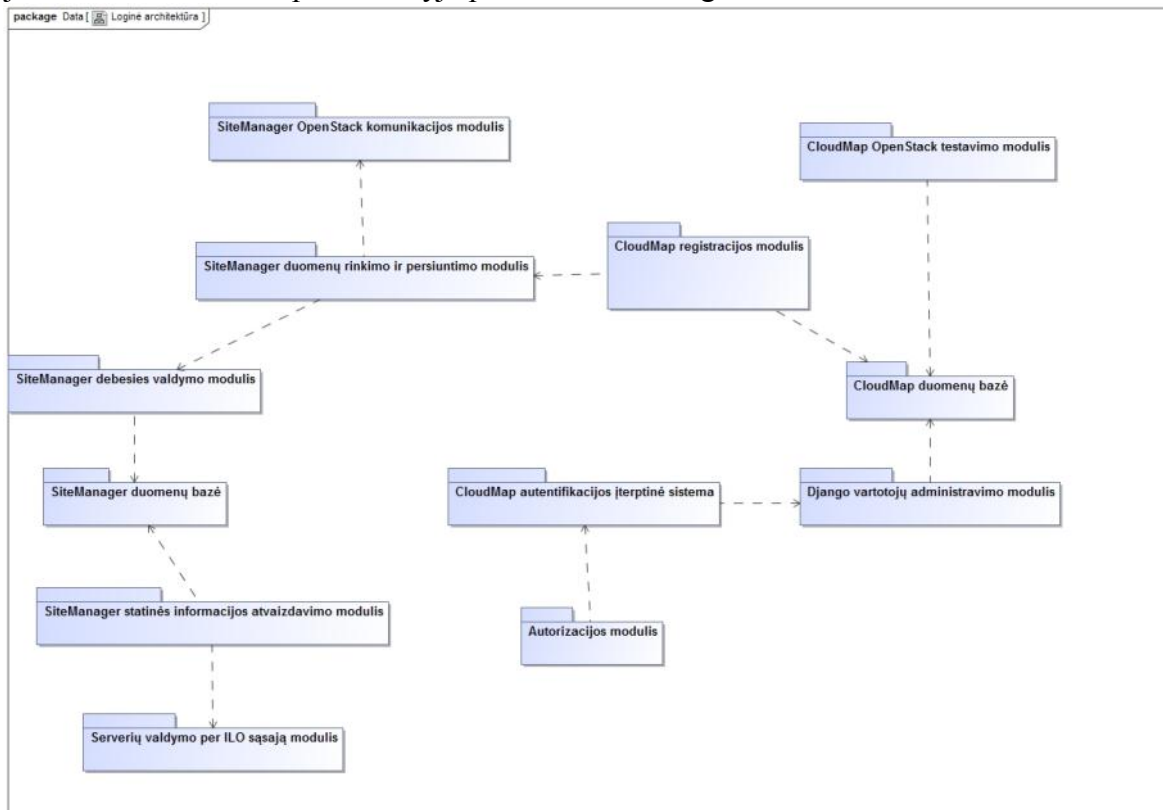
Šios dvi sistemos pasirinktos realizuoti su Python 2.6 kalba, naudojant Django 1.6 karkasą, taikomųjų programų stiliui pasitelkiant Twitter Bootstrap CSS karkasą. Sprendimams taip pat išnaudojama daugelis Python modulių, – tai vienas iš didžiausių šios programavimo kalbos privalumų, kurie įgalina programuotoją įveikti praktiškai bet kokią užduotį daug efektyviau nei kitomis programavimo kalbomis. Abu šie sprendimai buvo kurti pagal Django bendruomenėje nustatytas geriausias projektų kūrimo praktikas. Django filosofijoje projektas tai vidinių taikomųjų programų visuma. Taikomoji programa, tai programos modulis, turintis savo duomenų bazės modelius, atvaizdavimo funkcijų rinkinius, testavimo scenarijus, formas. Kartais, jie gali turėti ir URL kontrolei skirtas klases, taip dar labiau padidinant savarankiškumą. Naudojant tokią taikomųjų programų struktūrą tampa labai paprasta iš vieno projekto perkelti modulius į kitą projektą.

Sprendimui įdiegti bus reikalingas vienas serveris CloudMap sistemai ir bent po vieną serverį kiekviename duomenų centre, kurie išsidėstę įvairiose laiko zonose esančiose šalyse:

- JAV
- Jungtinėje Karalystėje
- Singapūre

4.1. Sistemų loginė architektūra

Sistemos loginė architektūra vaizduoja kaip atskiri sistemos moduliai sąveikauja tarpusavyje. Žemiau esančiame paveikslėlyje pavaizduota ši diagrama.



34. pav. CloudMap ir SiteManager informacinių sistemų loginė architektūra

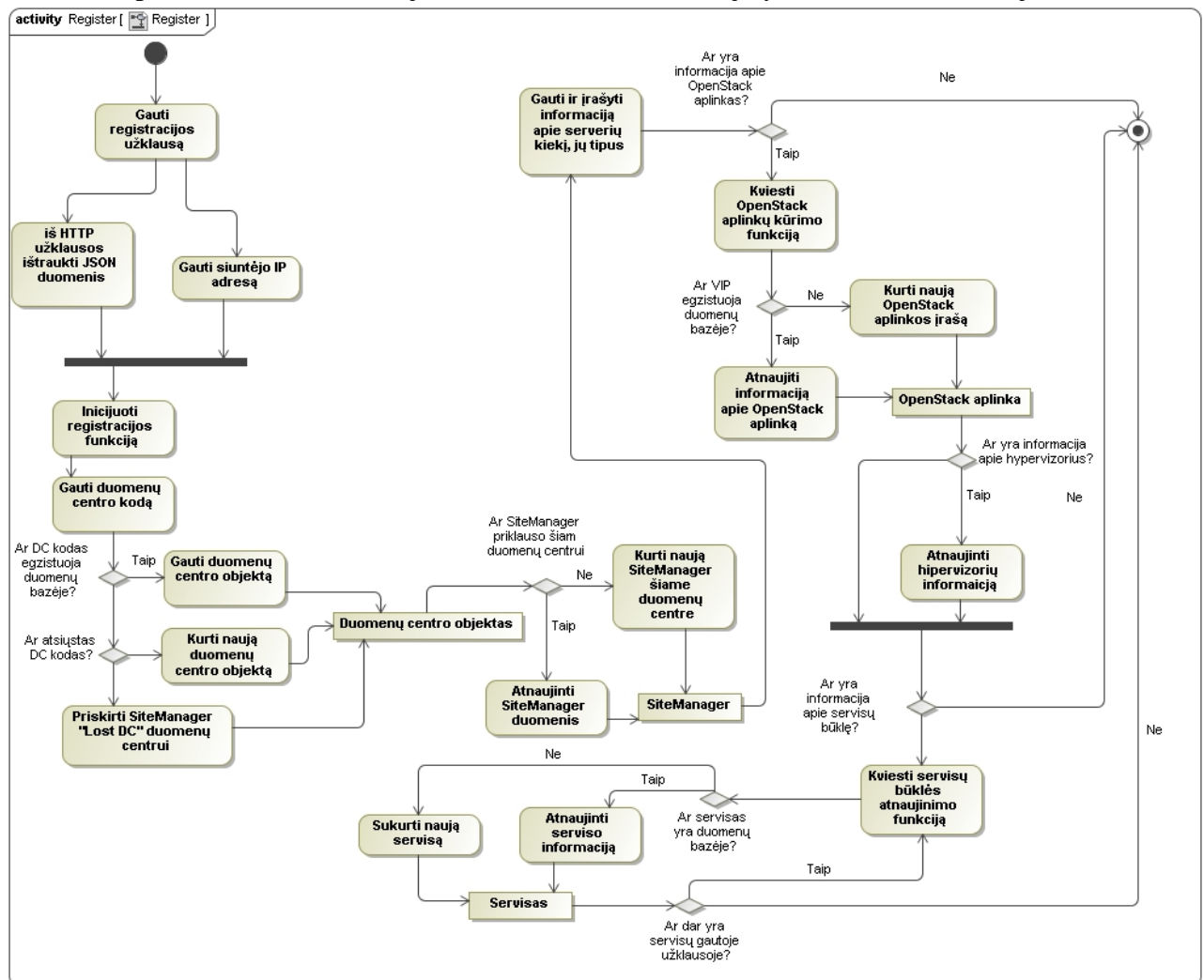
Iš jos matome, kad yra gana daug atskirų komponentų ir nėra vieno pagrindinio komponento, kuris atvaizduotų vartotojui visą bendros sistemos veiklą. Jie visi glaudžiai susiję, ir, pavyzdžiui, be SiteManager OpenStack komunikacijos modulio, nebūtų renkama informacija per OpenStack API ir dėl to praktiškai jokios naudingos informacijos nebūtų galima persiųsti į CloudMap sistemą. CloudMap registracijos modulis yra atsakingas ir už surinktos informacijos atvaizdavimą, todėl jis galėtų būti įvardintas kaip CloudMap sistemos pagrindinis komponentas, tačiau jis būtų visiškai nenaudojamas inžinierių, kurie naudoja SiteManager sistemą OpenStack aplinkos diegimui. Taigi, tam tikrus šios sistemos modulius naudos skirtingos vartotojų grupės.

4.1.1. CloudMap ir SiteManager sistemų veiklos diagramos

CloudMap sistema turi kelias pagrindines funkcijas, kurias galima vaizduoti veiklos diagramomis. Jos nėra skirtos pavaizduoti programinio kodo veikimą, tačiau puikiai nupasakoja algoritmo veikimą. Pagrindinė abiejų sistemų funkcija – automatizuoti kuo daugiau funkcijų, taigi nuo duomenų rinkimo, apdorojimo iki persiuntimo – stengiamasi visus žingsnius atlikti sistemos viduje automatiškai. SiteManager sistema, nors ir nėra visiškai automatizuotas įrankis – stengiasi sistemos naudotojui kaip įmanoma daugiau veiksmų automatizuoti ir pateikti kelis variantus, kaip įvykdyti tam tikrus užduotis (serverių paieška galima keliais būdais). Taip pat automatizuojamas OpenStack aplinkos diegimas, leidžiant pasirinkti galutinę konfigūraciją.

4.1.1.1. CloudMap registracijos veiklos diagrama

Sekančiame paveikslėlyje pavaizduota SiteManager sistemos registracija į CloudMap. Naudojamas HTTP protokolas ir suformuojama užklausa, kurios viduje yra JSON formato objektas.



35. pav. SiteManager registracijos į CloudMap veiklos diagrama

Visa tai vykdoma CloudMap pusėje, gavus duomenis iš SiteManager sistemos. Šio informacijos transporto privalumai:

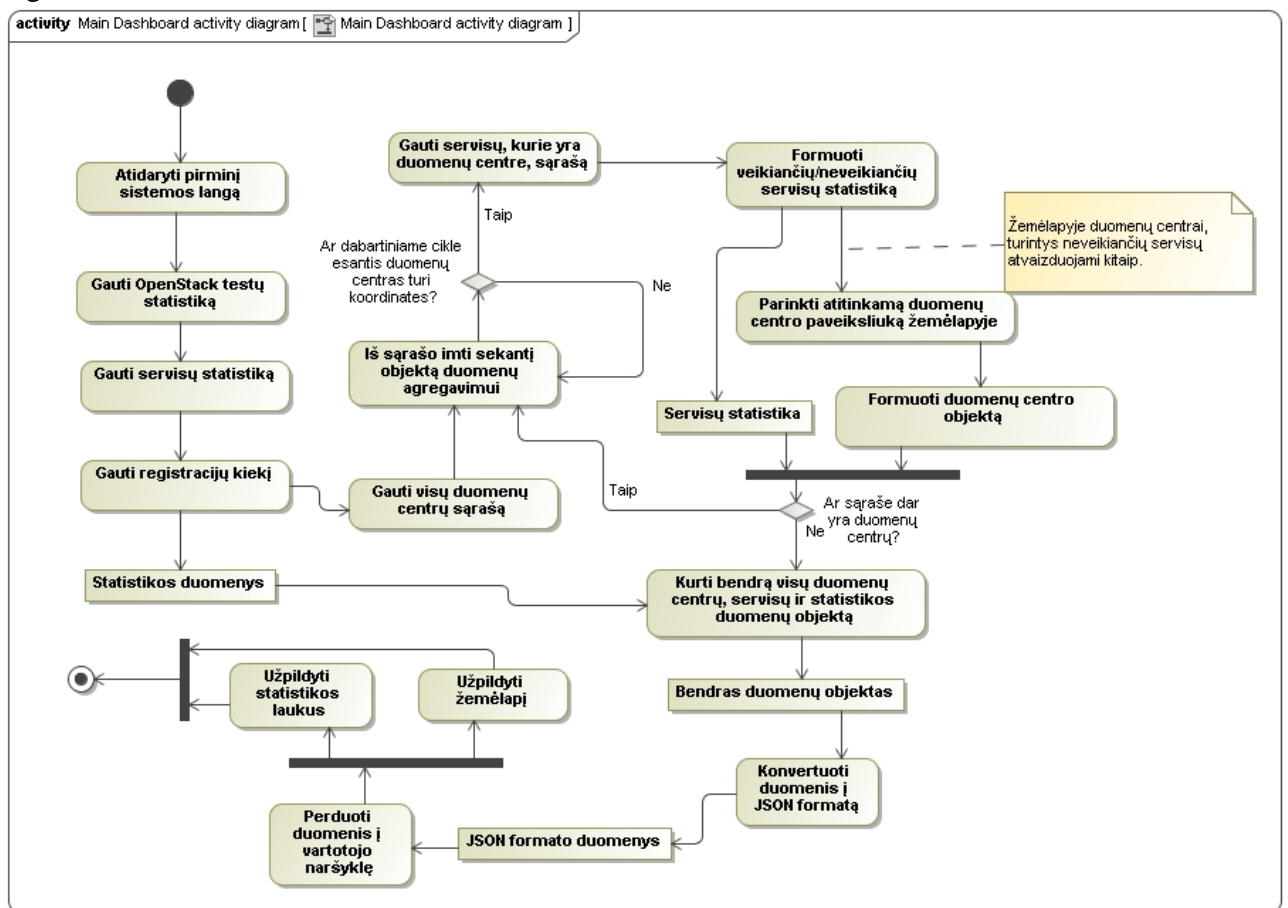
- JSON tipo duomenys užima mažiau vietos ir yra lengviau skaitomi nei XML tipo.
- Naudojant standartines Python bibliotekas, galima nesunkiai perversti JSON tipo duomenis į standartinius Python objektus, tokius kaip „list“ ir „dict“.
- Išlaikomas duomenų integralumas – visada atnaujinami visi duomenys: SiteManager fizinių resursų kiekiai, OpenStack aplinkos ir jų servaisai.

Trūkumai:

- Duomenų formatas gali sukelti problemų, jei informacija būtų atnaujinama iš kitos sistemos (ne SiteManager).

4.1.1.2. CloudMap pagrindinio lango duomenų rinkimo veiklos diagrama

Toliau pateikiame CloudMap sistemos pagrindinio lango atidarymo metu vykstančius procesus. Šie procesai turi būti kaip įmanoma labiau optimizuoti, siekiant sumažinti vykdomų užklausų kieki, nes ši informacija turi būti nuolat automatiškai atnaujinama. Pagrindinio lango veiklos diagrama žemiau:



36. pav. CloudMap pagrindinio lango veiklos diagrama

4.1.1.3. CloudMap integracinis OpenStack sistemos testavimas

Kita, didelę reikšmę turint CloudMap sistemos funkcija – integracinis OpenStack testavimas. Šio darbo rašymo metu buvo planuota sukurti du Nova komponento testus:

- API tikrinimas, ar atsako į užklausas ir laiko matavimas iki atsakymo gavimo.
- Virtualios mašinos sukūrimas ir sunaikinimas, taip pat išmatuojant visos operacijos laiko trukmę bei išsaugant istoriją buvusių testų. Šiame teste taip pat svarbu

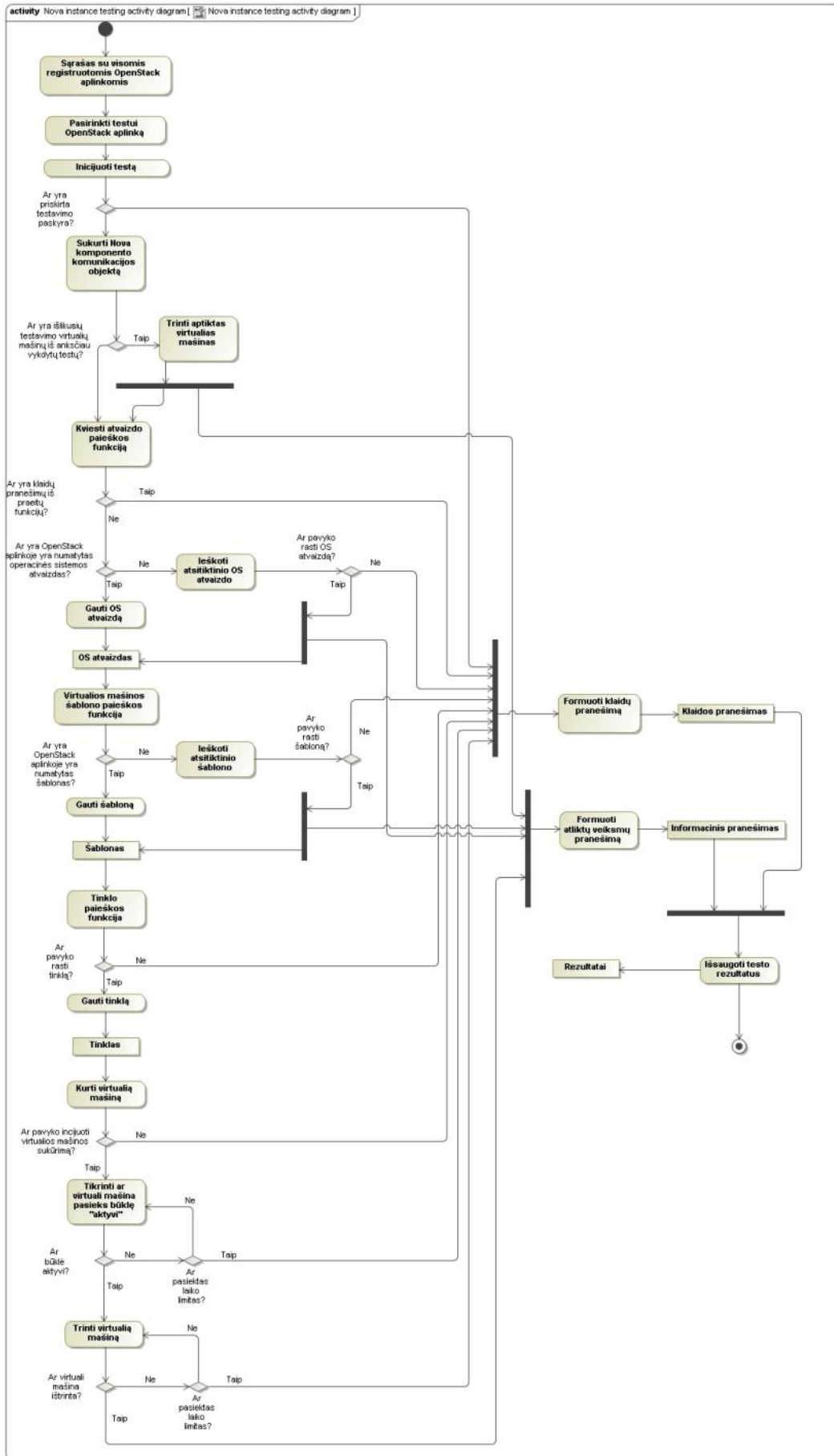
identifikuoti, kokiame etape OpenStack susidūrė su problemomis, taip stipriai palengvinant debesų kompiuterijos sprendimo palaikymo komandos darbą.

Kadangi per visus duomenų centrus bus tie patys operacinių sistemų atvaizdai ir virtualių mašinų šablonai – „flavors“, yra galimybė numatyti standartines parinktis, pagal kurias būtų kuriama virtuali mašina. Neradus jų – būtų bandoma imti bet kurią kitą elementą iš turimų resursų ir nenutraukti testo, tačiau būtų informuojamas sistemos administratorius. Nepaisant to, jog nebuvo rasta standartinių parinkčių, testo galutinė būseną būtų nusakoma tada, kai pavyksta sukurti mašiną, ji pasiekia *Active* būseną, o tada ji sėkmingai ištrinama.

Taip pat dėl didelio kiekio įvairių funkcijų, kurios kviečiamos šios veiklos metu – būtina turėti detalų atliktų veiksmų ir papildomos informacijos rinkimo ir atvaizdavimo būdą. Tai atliekama kuriant sąrašą iš visų atliktų veiksmų:

- Patikrinama, ar yra priskirta testavimo paskyra.
- Patikrinama, ar iš ankstesnių testų yra virtualių mašinų ir pamėginti jas ištrinti.
- Surasti operacinės sistemos atvaizdą.
- Surasti virtualios mašinos šabloną.
- Surasti tinklą.
- Sukurti virtualią mašiną ir patikrinti, ar ji pasiekė *Active* būseną per nustatytą laiką.
- Ištrinti mašiną ir patikrinti, ar pavyko ją ištrinti per nustatytą laiką.
- Išsaugoti testo rezultatus.
- Išvesti į ekraną testo rezultatus naudotojui.

Žemiau pateikiama šios funkcijos veiklos diagrama:



37. pav. Virtualios mašinos kūrimo ir ištrynimo testas

4.1.2. CloudMap gautos JSON tipo informacijos konversijos į *Dict* tipą pavyzdys

Toliau pateikiame duomenų pavyzdį, gaunamą iš SiteManager sistemos, kai jis registruojasi į CloudMap sistemą. Šiame pavyzdyje pakeisti IP adresai ir pavadinimai, tačiau puikiai atspindi struktūra. Duomenis galima sudėlioti į tam tikrą hierarchiją:

- SiteManager informacija
 - OpenStack aplinkų informacija (raktas „envs“)
 - OpenStack resursų informacija (raktas „hypervisors“)
 - OpenStack servisų būklė (raktas „health“)

Pavyzdys pateiktas žemiau:

```
{'comp_count': 8,
'cont_count': 7,
'dc': 'GBR',
'envs': [{'hypervisors': {'instances': 15,
'memory_free': 2113681752064,
'memory_total': 2164684488704,
'memory_used': 51002736640,
'storage_free': -126701535232,
'storage_total': 236223201280,
'storage_used': 362924736512,
'shares_total': 50000,
'shares_used': 26000,
'vcpu_free': 170,
'vcpu_total': 192,
'vcpu_used': 22},
'name': u'testenv1',
'state': u'Active',
'vip': u'1.1.1.1',
'public_vip': u'1.1.1.1'},
{'hypervisors': {'instances': 31,
'memory_free': 439978295296,
'memory_total': 540373155840,
'memory_used': 100394860544,
'storage_free': 6920266055680,
'storage_total': 7874822537216,
'storage_used': 954556481536,
'shares_total': 50000,
'shares_used': 26000,
'vcpu_free': 27,
'vcpu_total': 80,
'vcpu_used': 53},
'name': u'testenv2',
'state': u'Active',
'vip': u'1.1.1.2',
'public_vip': u'1.1.1.2',
'health': [
{'status': 'DOWN', 'type': 'metering',
'enabled': True}, {'status': 'UP',
'type':
'nova-conductor:LDTCLD010000010',
'enabled': True},
{'status': 'UP',
'type': 'nova-scheduler:LDTCLD010000010',
'enabled': True}, {'status': 'UP',
'type':
'nova-scheduler:LDTCLD010000013',
'enabled': True},
{'status': 'UP',
'type': 'nova-conductor:LDTCLD010000013',
'enabled': True}, {'status': 'UP',
'type':
'nova-conductor:LDTCLD010000014,
```

```

        'enabled': True},
    {'status': 'UP',
     'type': 'nova-scheduler:LDTCLD010000014',
     'enabled': True}, {'status': 'UP',
     'type':
       'nova-consoleauth:LDTCLD010000010',
     'enabled': True},
    {'status': 'UP',
     'type': 'nova-compute:LDTCLD010000015',
     'enabled': True}, {'status': 'UP',
     'type':
       'nova-compute:LDTCLD010000018',
     'enabled': False},
    {'status': 'UP',
     'type': 'nova-compute:LDTCLD010000017',
     'enabled': False}, {'status': 'UP',
     'type':
       'nova-consoleauth:LDTCLD010000013',
     'enabled': True},
    {'status': 'UP',
     'type':
       'nova-consoleauth:LDTCLD010000014',
     'enabled': True},
    {'status': 'UP', 'type': 'image',
     'enabled': True},
    {'status': 'UP', 'type': 'network',
     'enabled': True},
    {'status': 'UP', 'type': 'identity',
     'enabled': True},
    {'status': 'UP', 'type': 'volume',
     'enabled': True}}],
's_count': 21,
'strg_count': 3}

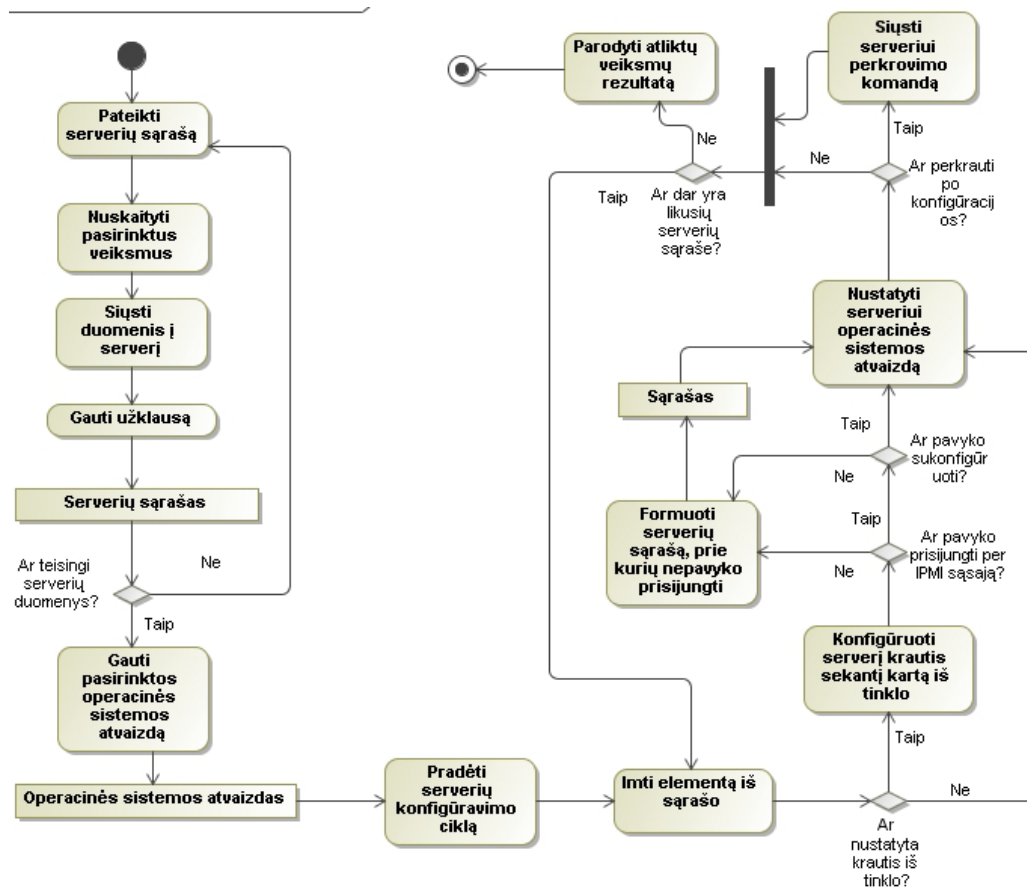
```

38. pav. SiteManager registracijos į CloudMap duomenų pavyzdys

Iš šio pavyzdžio galime matyti, jog servisų tipai nusakomi ne tik jų tipais, tačiau priskiriamas ir serverio vardas. To reikia, kad būtų galima turėti daugelį unikalių servisų debesų kompiuterijos sprendime. Dėl šios sistemos prigimties – daugelis servisų yra dubliuojami, pavyzdžiui skaičiavimo tipo serverių vienoje OpenStack aplinkoje gali būti nuo nulio (kai patys valdymo serveriai naudojami kurti virtualias mašinas) iki kelių tūkstančių.

4.1.3. SiteManager serverių importavimo iš CSV failo veiklos diagrama

Ši funkcija yra naudinga turint naujai įdiegtą SiteManager sistemą, kurios duomenų bazėje dar nėra serverių. Ši funkcija ne tik gali pridėti serverius, tačiau vartotojo nurodymu gali atlikti papildomus veiksmus, tokius kaip visos galimos informacijos atnaujinimas per IPMI sąsają arba pasirinktinai tik serverio vietos nustatymas. Šios veiklos diagrama pateikta žemiau:



39. pav. Informacijos apie serverius importavimas iš CSV failo

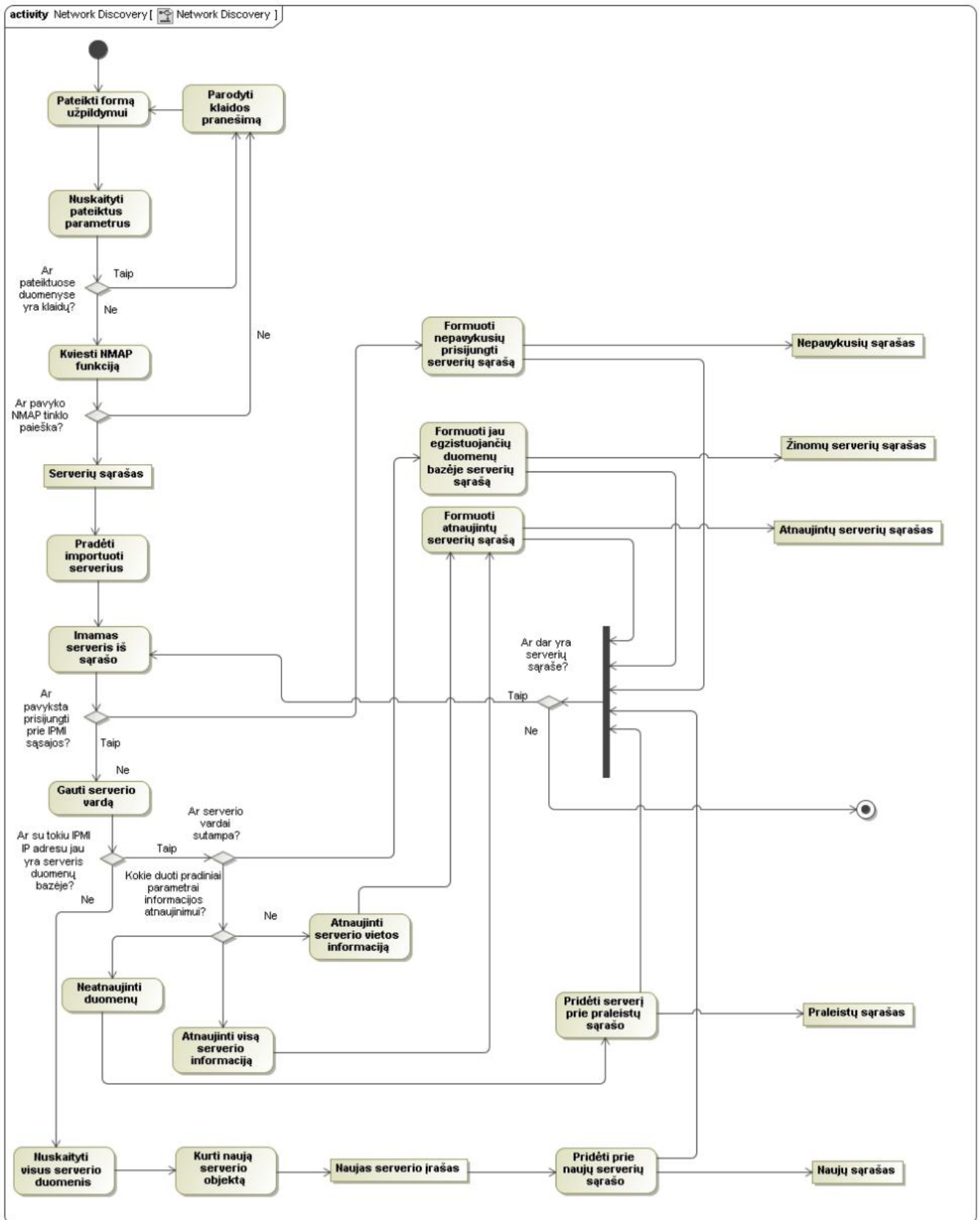
Iš aukščiau pateiktos veiklos diagramos matome, jog atliekami papildomi veiksmai, kai SiteManager sistema jungiasi į fizinius serverius per IPMI sąsają. Tam reikalingi IPMI prisijungimo duomenys. Viena iš problemų, kilusių eksportuojant ir importuojant duomenis buvo ta, jog eksportuojant serverių sąrašą negalima įtraukti IPMI prisijungimo vardo ir slaptažodžio dėl saugumo priežasčių. Taigi negalima to paties failo (be papildomų pakeitimų) eksportavus iš SiteManager ikelti į kitą SiteManager sistemą.

Taip pat importuojant serverių sąrašą nėra numatyta atkurti arba sukurti naujus ryšius tarp serverio ir OpenStack aplinkos. Šis metodas apsiriboja tik statinių duomenų (StaticData) taikomosios programos ribose.

4.1.4. SiteManager serverių paieškos tinkle veiklos diagrama

Automatinė serverių paieška pirmuose sistemos planavimo ir kūrimo etapuose buvo įsivaizduojamas tik kaip įrankis, kuris galėtų surasti serverį, jei nežinomas jo konkretus IP adresas, o tik potinklis ar IP adresų režis. Tačiau, įdiegus šią funkciją tai pasirodė kaip puikus būdas greitai ir efektyviai surasti serverius ir įrašyti jų informaciją į SiteManager duomenų bazę. Vienintelis šio serverių paieškos būdo trūkumas – tinklo skenavimas NMAP programa, kurios veikimas gali būti traktuojamas kaip atakos požymis, dėl to turi būti perspėti atitinkami įmonės skyriai. Taip pat serverių paieška vykdoma tik tikrinant 17988 sąsaja, kuri dar žinoma kaip „Virtual Media Port“ ir, dažniausiai, yra atvira tik serveriuose, kuriuose aktyvus IPMI modulis.

Naudojant šią funkciją ir žinant serverių IPMI prisijungimo duomenis, bei potinklį – galima nesunkiai jos pagalba pridėti net kelis šimtus serverių. Šios funkcijos veiklos diagrama pateikta žemiau:



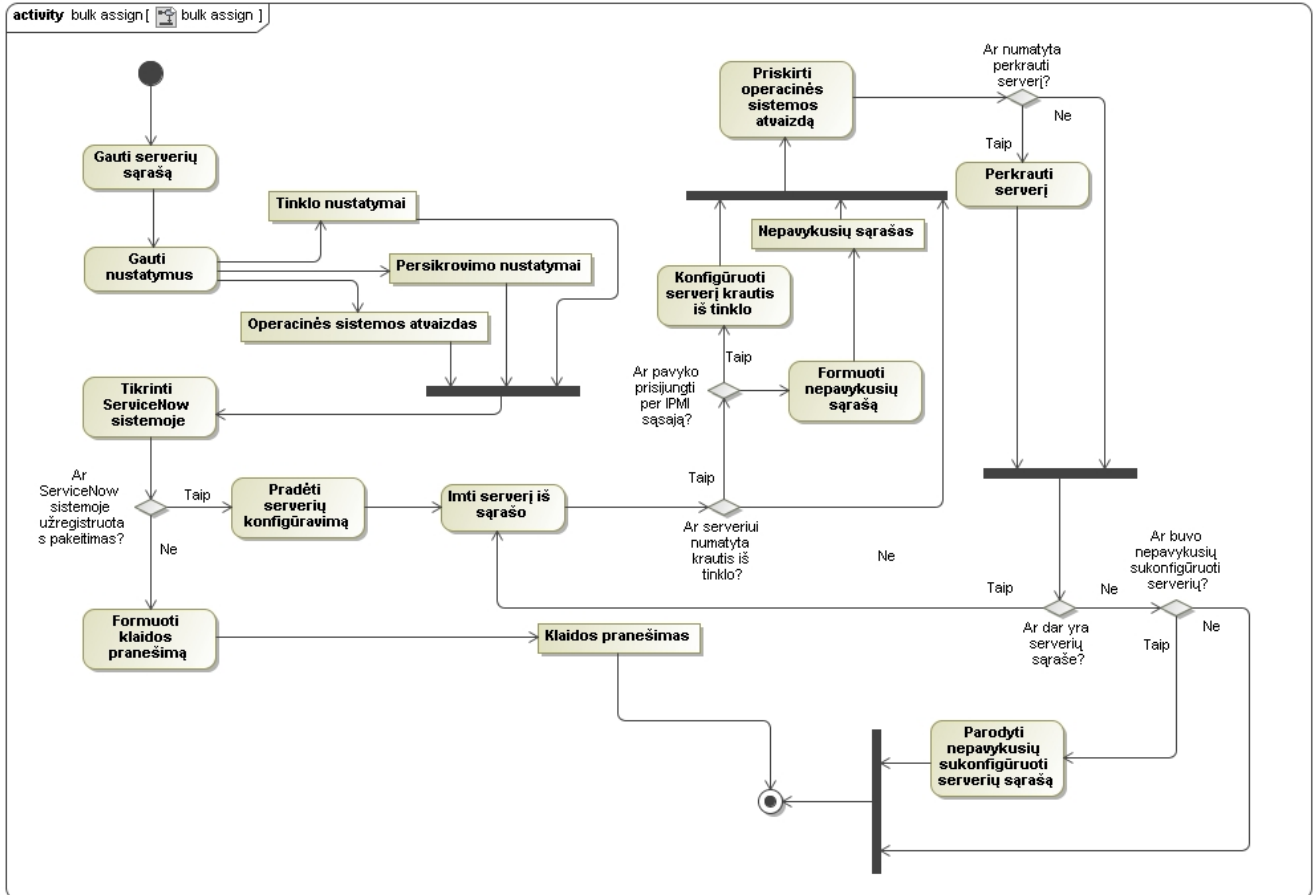
40. pav. Automatinės serverių paieškos tinkle veiklos diagrama

Iš veiklos diagramos matome, jog rezultate suformuojami keli sąrašai, kurie vėliau yra parodomi vartotojui. Vienas iš jų identifikuoja serverius, kurių vardai nesutampa, tačiau IPMI IP adresai vienodi, taip galima gana nesudėtingai surasti klaidas duomenyse. Tokiems atvejams gali būti kelios priežastys, o viena iš jų – serveryje buvo perrašyta operacinė sistema, o SiteManager esantis

įrašas nebuvo atnaujintas. Tokie neatitikimai gali vėliau sukelti problemų, kai pridėjus serverį prie OpenStack aplinkos ir bandant jį surasti ir pridėti prie serverių grupių nepavyks jo surasti.

4.1.5. SiteManager daugelio serverių operacinės sistemos atvaizdo priskyrimo veiklos diagrama

Daugeliui serverių operacinės sistemos atvaizdą priskirti per vieną funkciją naudinga tuomet, kai reikia OpenStack įrašymui paruošti iškart dešimtis arba šimtus serverių, nes kitu atveju – tai reiktų daryti po vieną, įeinant į kiekvieno serverio parinktį ir nustatant operacinės sistemos atvaizdą, nustatant sekantį kartą krautis iš tinklo ir, galiausiai, perkraunant serverį. Visus šiuos veiksmus iškart daugeliui serverių automatizuoja žemiau pateikta veiklos diagrama:



41. pav. Daugelio serverių operacinės sistemos atvaizdo konfigūravimo veiklos diagrama

Iš aukščiau pateiktos veiklos matome, jog perkonfigūravimo metu yra užtikrinama, kad jokie veiksmai nebus pradėti, jeigu pakeitimas nėra patvirtintas ServiceNow sistemoje.

4.2. Duomenų bazės modelis

Nors šios sistemos skirtos atlikti bendrą užduotį, tačiau natūralu, jog turi atskiras duomenų bazines, kuriose tam tikri modeliai persidengia ir atlieka tą pačią funkciją. Duomenų bazei realizuoti pasirinkta MariaDB – tai MySQL duomenų bazės kūrėjų darbas, kuris įgavo didelį pagreitį, kai MySQL buvo nupirktas Oracle kompanijos. Ši duomenų bazė yra gerokai pranašesnė už savo pirmtakę, dėl to ją renkasi naudoti vis didesniame kiekyje naujų sistemų. Be to, ji yra visiškai be problemų pakeičianti jau esamą sprendimą, kuris naudoja MySQL duomenų bazę, nes net serviso pavadinimas yra tas pats – „mysql“ arba „mysqld“. Taip pat tinka tos pačios sistemos tvarkyklės, pavyzdžiui Django naudojama jungtis į MySQL duomenų bazę „MySQL-python“ veikia tiek su MySQL, tiek ir su MariaDB. Duomenų bazių detalesnį palyginimą galima surasti čia: <https://mariadb.com/kb/en/mariadb/optimizer-feature-comparison-matrix/>. Viena iš stipriausių MariaDB savybių yra *Galera Cluster*, tai sinchroninis duomenų bazės klasteris, kur kiekvienas narys

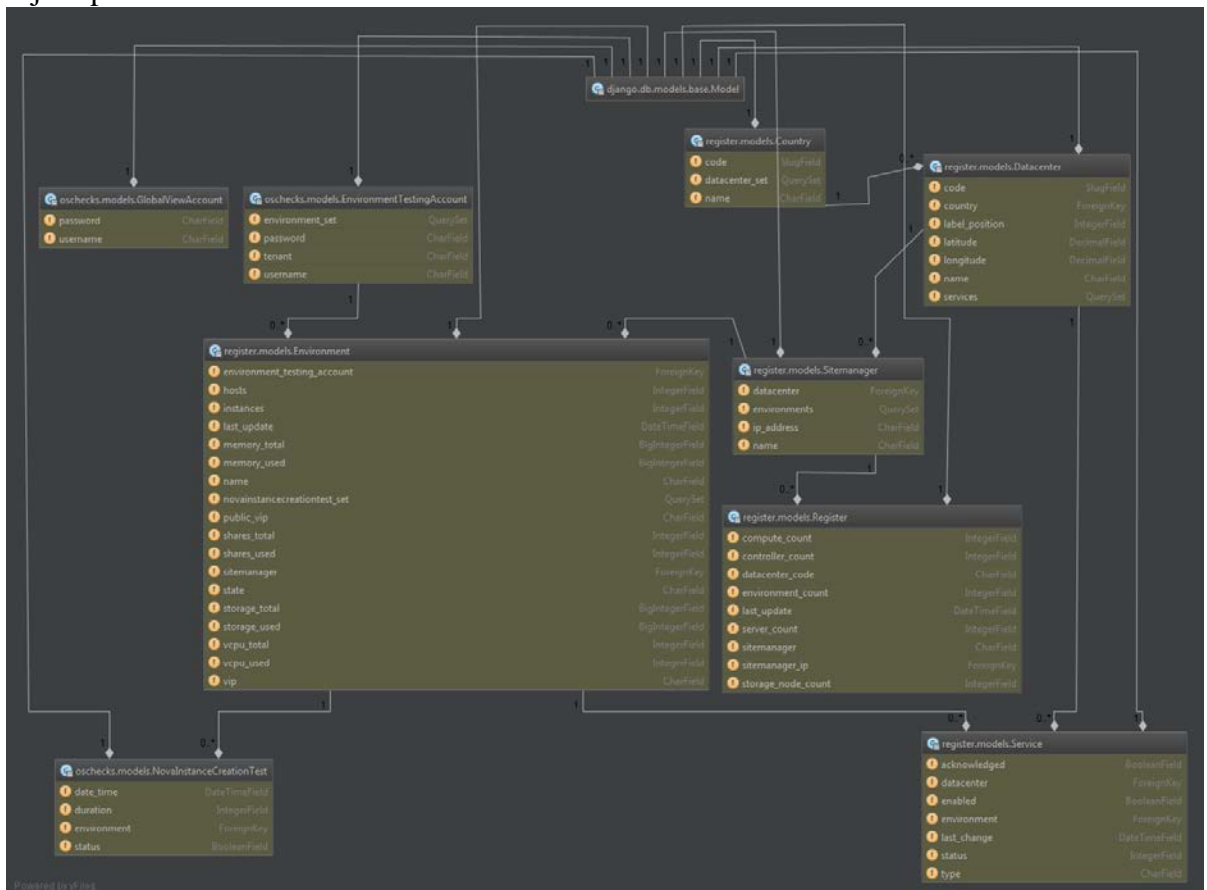
yra šeimininkas ir nėra žemesnio lygio duomenų bazių, kurios tik imtų informaciją. Ši sistema palaiko:

- sinchroninę replikaciją
- visos bazės visada aktyvios
- galima rašyti ir skaityti iš bet kurios duomenų bazės
- sutrikusios duomenų bazės automatiškai išmetamos iš klasterio
- automatiškai pridedamos naujos duomenų bazės į klasterį
- paralelinė replikacija duomenų bazės eilučių lygyje

Sujungus šias savybes galime turėti sistemą, kuri turi bendrą duomenų bazę, veikiančią iškart keliuose virtualiose mašinose, taip sukuriant didelį patikimumą (sistema veiktų net sutrikus dvejoms virtualioms mašinoms). Ateities planuose numatome galimybę taikomajai programai pačiai atkurti savo neveikiančias šakas, taip padarant ją praktiškai nesunaikinamą.

4.2.1. CloudMap duomenų bazės modelis

Žemiau pateikiamas CloudMap duomenų bazės modelis. Šiame modelyje matome, kad pagrindinis objektas šioje sistemoje yra OpenStack aplinka, kurios parametrai atsispindi Environment duomenų bazės lentelėje. Visi kiti duomenys vienaip ar kitaip papildo šio debesų kompiuterijos sprendimo duomenis.



42. pav. Pav. CloudMap duomenų bazės modelis, sugeneruotas per IntelliJ IDE

Toliau pateikiami duomenų bazės lentelių duomenų laukų aprašymai.

66. lentelė. „Environment“

Laukas	Tipas	Aprašymas
name	CharField	OpenStack aplinkos pavadinimas
sitemanager	ForeignKey	Išorinis raktas į SiteManager lentelę
environment_testing_account	ForeignKey	Išorinis raktas į EnvironmentTestingAccount lentelę
state	CharField	Būsena (Aktyvi, kuriama, nepavykusi)
vip	CharField	Vidinis virtualus IP, naudojamas krūvio paskirstymui

public_vip	CharField	Viešas virtualus IP, per šį IP galima pasiekti OpenStack aplinkos API.
instances	Integer	Virtualių mašinų kiekis aplinkoje.
hosts	Integer	Fizinių serverių kiekis, priskirtas OpenStack aplinkai.
memory_total	BigInteger	Visas RAM kiekis priskirtas OpenStack aplinkai.
memory_used	BigInteger	Išnaudotas RAM kiekis.
storage_total	BigInteger	Lokali saugojimui skirta vieta (yra ir ne lokali, kurios negalime nustatyti per OpenStack API).
storage_used	BigInteger	Išnaudota saugojimo vieta.
vcpu_total	Integer	Visas virtualių CPU kiekis.
vcpu_used	Integer	Išnaudotas virtualių CPU kiekis aplinkoje.
shares_total	Integer	Papildoma metrika nustatyti OpenStack resursams, bendras jos kiekis.
shares_used	Integer	Išnaudotas „shares“ kiekis.
last_update	DateTimeField	Paskutinė OpenStack aplinkos duomenų atnaujinimo data.

67. lentelė. „Sitemanager“

Laukas	Tipas	Aprašymas
ip_address	CharField	SiteManager sistemos IP adresas, per kurį būtų galima jį pasiekti. Nustatomas automatiškai, kai SiteManager prisiregistruoja į CloudMap sistemą.
name	CharField	Pavadinimas, automatiškai nepriskiriamas, tačiau sistemos administratorius gali duoti jiems vardus, kad būtų lengviau atskirti.
datacenter	ForeignKey	Duomenų centras, nustatomas iš OpenStack aplinkos parametrų. Šiuo metu yra 7 duomenų centrai, kuriuose diegiamas debesų kompiuterijos sprendimas.
environments	QuerySet	Automatiškai Django sukuriama laukas, kuris nustato visas OpenStack aplinkas, kurių išorinis raktas rodo į šį SiteManager objektą.

68. lentelė. „Register“

Laukas	Tipas	Aprašymas
compute_count	Integer	Skaičiavimo serverių kiekis SiteManager sistemoje.
controller_count	Integer	Valdymo serverių kiekis SiteManager sistemoje.
datacenter_code	CharField	Duomenų centro kodas.
environment_count	Integer	OpenStack aplinkų kiekis SiteManager sistemoje.
last_update	DateTimeField	Automatiškai priskiriama reikšmė įvykus registracijai.
server_count	Integer	Bendras serverių kiekis SiteManager, kurie yra užregistruoti sistemoje.
sitemanager	CharField	SiteManager pavadinimas, galima jį nustatyti, kad siųstų kiekvieną kartą.
sitemanager_ip	ForeignKey	IP adresas, nustatomas automatiškai.
storage_node_count	Integer	Informacijos saugojimo serverių kiekis SiteManager sistemoje.

69. lentelė. „Datacenter“

Laukas	Tipas	Aprašymas
code	SlugField	Duomenų centro kodas, unikalus.
country	ForeignKey	Išorinis raktas į Country lentelę, kurioje yra šalys.
label_position	IntegerField	Naudojant kartu su integruotu Django ChoiceField per šį lauką

		nustatoma pozicija duomenų centro papildomos informacijos laukelio pozicija žemėlapyje.
latitude	DecimalField	Duomenų centro koordinacijų platuma.
longitude	DecimalField	Duomenų centro koordinacijų ilguma.
name	CharField	Duomenų centro pavadinimas, ši informacija automatiškai sudedama į duomenų bazę kuriant lenteles.

70. lentelė. „Country“

Laukas	Tipas	Aprašymas
code	SlugField	Šalies kodas.
name	CharField	Šalies pavadinimas.

71. lentelė. „EnvironmentTestingAccount“

Laukas	Tipas	Aprašymas
username	CharField	Prisijungimo vardas.
password	CharField	Prisijungimo slaptažodis.
tenant	CharField	Tenant – OpenStack aplinkoje esantis projektas, turintis savo vartotojų grupę, priskirtų resursų kiekį, tinklų konfigūracijas. Testavimo tikslui kiekviena OpenStack aplinka turės numatytą tenant, kurio viduje bus galima kurti ir trinti virtualias mašinas.

72. lentelė. „GlobalViewAccount“

Laukas	Tipas	Aprašymas
username	CharField	Prisijungimo vardas.
password	CharField	Prisijungimo slaptažodis.

Šioje lentelėje bus saugomas prisijungimas į OpenStack paskyrą, iš kurios galima matyti absoliučiai visus resursus, taip sudarant galimybę iš CloudMap sistemos akimirksniu nustatyti virtualios mašinos vietą bet kuriame duomenų centre visame pasaulyje.

73. lentelė. „NovaInstanceCreationTest“

Laukas	Tipas	Aprašymas
date_time	DateTimeField	Data ir laikas, kai buvo atliktas testas
duration	Integer	Testo trukmė sekundėmis.
environment	ForeignKey	Išorinis raktas į OpenStack aplinką, kurioje buvo atliekamas testas.
status	BooleanField	Testo baigtis, jei sėkmingas – True, jei nesėkmingas – False.

74. lentelė. „Service“

Laukas	Tipas	Aprašymas
acknowledged	BooleanField	Jeigu servisas nebeveikia, šis laukas gali laikyti reikšmę, jog sistemos administratorius problemą žino ir taip deaktivuoti aliarmą.
datacenter	ForeignKey	Išorinis raktas į duomenų centrą, kuriam priklauso servisas.
enabled	BooleanField	Ar servisas įgalintas OpenStack aplinkoje.
environment	ForeignKey	OpenStack aplinka, kurioje veikia šis servisas.
last_change	DateTimeField	Paskutinis pasikeitimas statuse.
status	Integer	Statusas, gali būti kelių tipų, pvz. „UP“, „DOWN“, „UNKNOWN“ – kai SiteManager sistema pagal tipą negalėjo pritaikyti reikalingos tvarkyklės, prisijungti per API ir atlikti testavimo.
type	CharField	Serviso tipas, jeigu tai skaičiavimo taške, kur gali susidaryti keli šimtai vienodų servisų, tuomet prie serviso tipo dar pridedamas serverio vardas, pvz. „nova-conductor:LDTCLD010000010“.

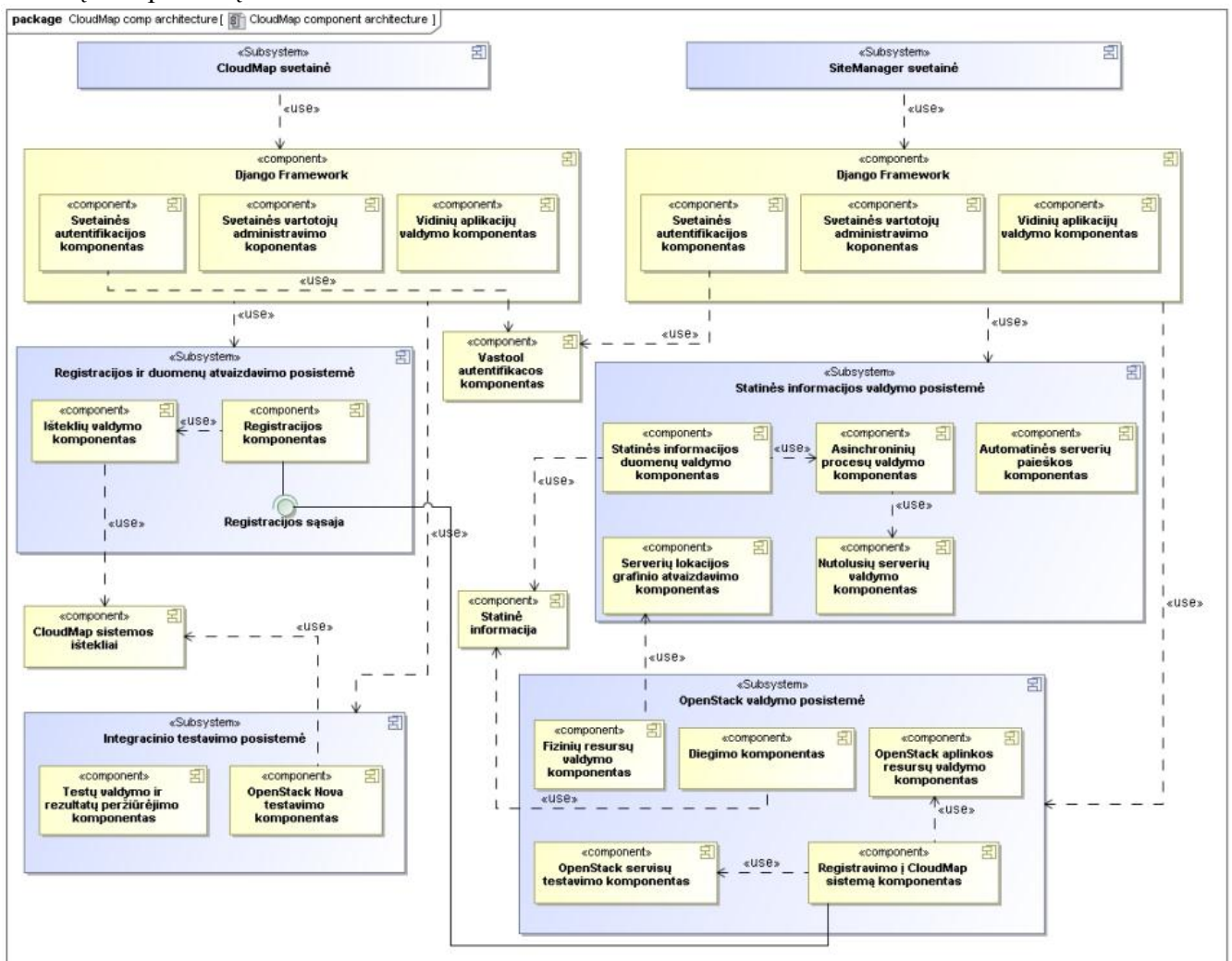
4.3. Detalus projektas

Šių dviejų sistemų didžiausias privalumas – dinamiškas apsijungimas, kai viena sistema gali papildyti kitą. Pavyzdžiui jau yra kitų sistemų, skirtų diegti OpenStack duomenų centre, tačiau iškyla problemų, jei įmonė turi kelis ar daugiau duomenų centrų ir tampa sunku suvaldyti ir visada žinoti prisijungimo duomenis, per kuriuos galima pasiekti visas OpenStack aplinkas bei matyti agreguotą informaciją iš visų duomenų centrų. Šioje vietoje atsiranda CloudMap sistema, kuri ir yra skirta dirbti su daugeliu OpenStack aplinkų vienu metu, taip pat turi galimybę kaupti informaciją apie SiteManager tipo sistemas – ją galėtų pakeisti bet koks kitas produktas, pvz. RedHat Tuskar, Mirantis Fuel ar bet kuris kitas produktas, jis vis tiek išlaikytų tuos pačius minimalius, tačiau pagrindinius parametrus, kuriuos dabar turi SiteManager.

Taip pat CloudMap sistema gali paleisti OpenStack integracinius testus – bandyti sukurti virtualią mašiną, o tada ją ištrinti.

4.3.1. Komponentų architektūra

Sistemos CloudMap ir SiteManager turi užtikrinti resursų administravimą (serverių paiešką, valdymą), OpenStack aplinkos diegimo automatizavimą, OpenStack aplinkos resursų valdymą realiu laiku, šių duomenų agregavimą ir persiuntimą į centrinę sistemą, kur būtų galimybė stebėti visų sistemų servisų būklę bei testų rezultatus globaliu mastu. Žemiau pateikta sistemos architektūrai svarbių komponentų architektūra.



43. pav. Sistemos komponentų architektūra

Viena iš svarbiausių šios sistemos savybių, kuri įneša į debesų kompiuteriją naujumą – automatizuotas informacijos rinkimas iš daugelio taškų. CloudMap sistemoje nereikia įvedinėti SiteManager adresų – jie patys susiranda CloudMap sistemą ir užsiregistruoja, taigi visada galima

žinoti, kiek yra įdiegtų OpenStack aplinkų, kurioje jos geografinėje vietoje ir jos būklė. Tokia sistemos architektūra aktualiausia, kai vienai serverių spintai yra skirtas vienas SiteManager (tada OpenStack sprendimas tampa visiškai autonomiškai veikiantis) ir net turint kelis šimtus SiteManager – turėtume vieną informacijos šaltinį apie juos visus (IP adresus, OpenStack aplinkas ir fizinių resursų duomenis).

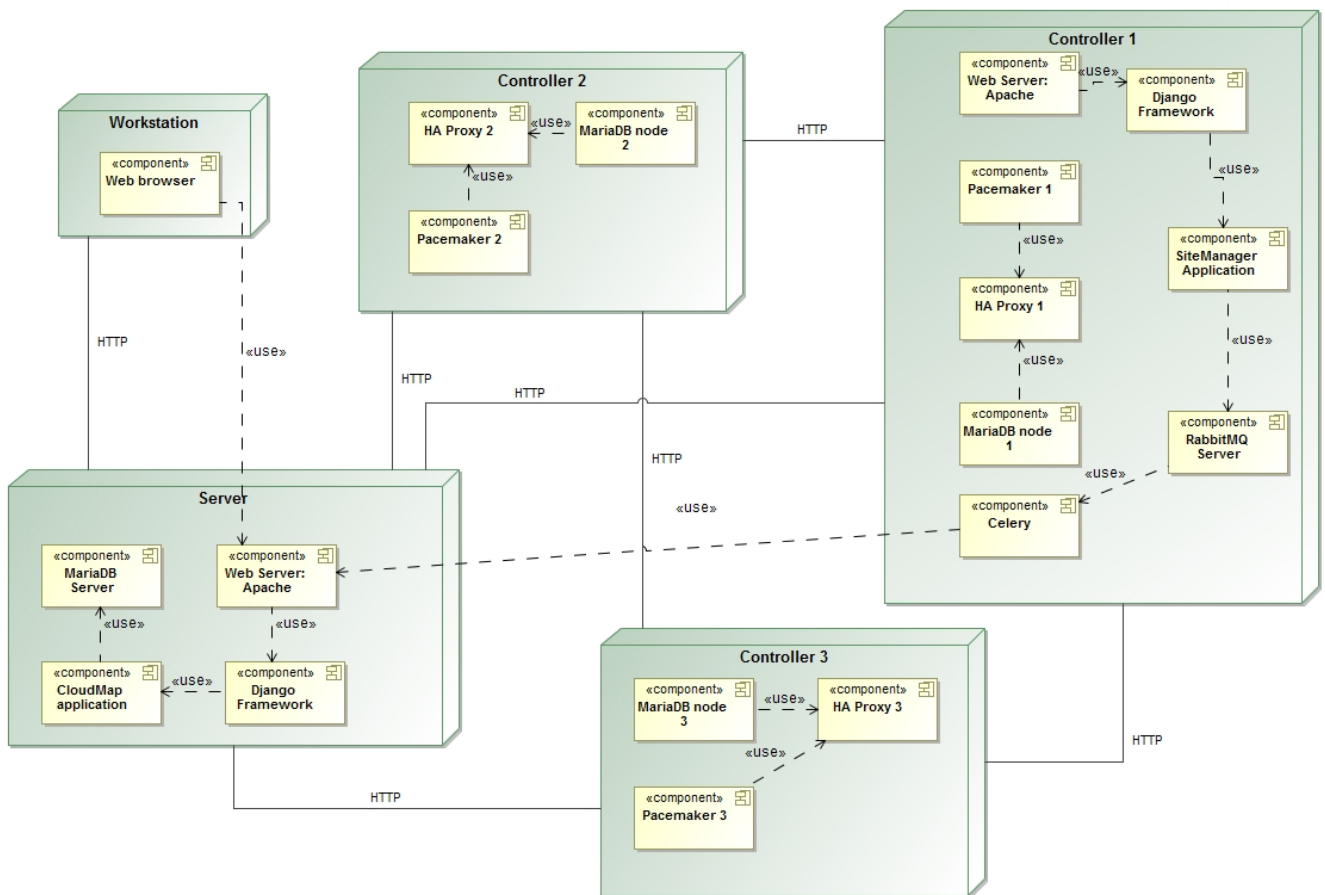
Buvo svarstyti ir kiti sprendimai:

- SiteManager duomenų bazę replikuoti per visas jo kopijas visuose duomenų centruose, tačiau tai būtų itin sudėtingas sprendimas, kuris sukurtų daug painiavos tiem, kas naudojami šia sistema, nes matytu, ne tik tuos resursus, kurie yra priskirti šioje kopijoje, bet ir visose kitose. Taigi, šis sprendimas buvo atmestas.
- Naudoti vieną SiteManager kuriant visas OpenStack aplinkas duomenų centruose. Šis sprendimas buvo pradinė idėja, kaip mes automatizuotai diegsime debesų kompiuterijos sprendimą duomenų centruose, tačiau vėliau buvo sukritikuotas ir atmestas, nes praradus tinklo ryšį tarp skirtingų duomenų centrų, kurie išsidėstę gana toli vienas nuo kito pasaulyje – nebūtų galimybės plėsti atskiras OpenStack aplinkas, nebūtų galimybės stebėti aplinkos parametrus. Taip pat išvis nepavyktų įdiegti OpenStack vietoje, kurioje nesukonfigūruotas tinklo ryšys (pvz. Nutolusioje Afrikoje, kur retai galima pasikliauti ryšiu).
- Sukurti atskirą modulį SiteManager sistemoje, kuris būtų atsakingas už kitų SiteManager sistemų paiešką ir agreguotų iš jų gaunamą informaciją. Šio sprendimo trūkumas – nepavyktų automatiškai rinkti informacijos, nes nebūtų vieno, centrinio taško, kurį būtų nesudėtinga sukonfigūruoti. Taip pat, žinant, jog šią sistemą ruošiamasi pakeisti kitais įrankiais – nėra protinga į ją įdėti perteklinio funkcionalumo.

Taigi, įvertinus kitus sprendimus, buvo pasirinkta kurti atskirą sistemą, skirtą aukštesnio lygio OpenStack ir SiteManager sistemų stebėjimui.

4.3.2. Diegimo modelis

CloudMap sistemos diegimo modelis yra paprastas, palyginus su SiteManager sistema, nes nereikia derintis jau prie serveryje egzistuojančių paketų, jų konfigūracijų ir duomenų bazių. Taip pat nereikia diegti Celery bei RabbitMQ serverio. Kita vertus, kadangi SiteManager diegiamas jau egzistuojančiuose OpenStack valdikliuose – didžioji dalis reikalavimų jau būna patenkinti (RabbitMQ, MariaDB) bei susidaro galimybė išnaudoti Galera Cluster.



44. pav. Sistemos elementų diegimo modelis

Vartotojo pusėje naudotis sistema bus reikalinga tik naršyklė (palaikomos visos, išskyrus IE8 ir žemesnės versijos).

Sistemai funkcionuoti reikalingi mažiausiai du serveriai, o produkcijai skirta konfigūracija susidaro iš mažiausiai keturių serverių bei krūvio paskirstymo įrenginio:

- **Web Server: Apache.** Šis komponentas skirtas apdoroti HTTP užklausoms. Su juo kartu naudojamas Python WSGI paketas, skirtas nukreipti užklausas į Django taikomąją programą.
- **Django Framework.** Tai vienas geriausių Python kalboje sukurtų projektų, turintis geriausią internetinių karkasų dokumentaciją ir didžiausią naudotojų bazę. Ši taikomoji programa savyje gali turėti daugelį kitų Python kalboje parašytų modulių, kurios taip pat vadinamos taikomosiomis programomis. Taip galima išskaidyti vieną didelį projektą į mažesnius, o tada juos naudoti pakartotinai kituose Django projektuose. Šis karkasas pasirūpina duomenų bazės modeliais, pradinių duomenų įrašytų, administraciniu moduliu (vos sukūrus naują projektą galima valdyti duomenų bazėje esančius duomenis). Taip pat šis karkasas suteikia galimybę nesudėtingai įdiegti autentifikaciją ir autorizaciją. [8]
- **CloudMap.** Tai pagrindinis komponentas, kuris atsakingas už SiteManager sistemų registraciją, duomenų atvaizdavimą ir integracinius OpenStack testus.
- **MariaDB.** Šis komponentas atsakingas už ilgalaikį duomenų saugojimą.
- **Galera Cluster.** Tai sinchroninė duomenų bazės replikacija skirta MariaDB duomenų bazei.
- **HAProxy.** Šis komponentas atsakingas už krūvio paskirstymą į serverius ir naudojamas tik didelio patikimumo OpenStack konfigūracijoje, kai sukuriama bent trys valdymo serveriai. Per jį vykdoma duomenų bazės replikacija bei nukreipiamas HTTP srautas, skirtas SiteManager sistemai į serverį, kuriame ji įdiegta.

- **Pacemaker.** Šis komponentas užtikrina, jog nustojus veikti tam valdymo serveriui, kuriame tuo metu yra sukonfigūruotas HAProxy – resursas bus perkonfigūruotas kitame serveryje. Taigi, jeigu vienu metu HAProxy yra sukonfigūruotas pirmajame valdymo serveryje ir per jį vyksta visas maršrutizavimas, tai jam nustojus veikti tai aptiks Pacemaker ir sukonfigūruos virtualų IP adresą kitame valdymo serveryje, pavyzdžiui antrajame arba trečiajame. Jeigu po tam tikro laiko pirmasis vėl pradės veikti – HAProxy bus perkonfigūruotas atgal ant jo.
- **RabbitMQ.** Tai servisas, atsakingas už žinučių pernešimą. Per jį vyksta bendravimas tarp SiteManager sistemos ir Celery komponento.
- **Celery.** Servisas, atsakingas už asinchroninių užduočių vykdymą. Paprastai būna sukonfigūruoti bent trys agentai, kurie klausosi per RabbitMQ servisą ateinančias jiems skirtas žinutes. Celery vykdo:
 - Fizinių serverių paiešką tinkle.
 - Serverių informacijos atnaujinimą.
 - OpenStack sistemos tikrinimą ir šios informacijos siuntimą į CloudMap.

Abejoms sistemoms (CloudMap ir SiteManager) taip pat reikia įdiegti papildomus Python paketus (bibliotekas) bei reikalingus operacinių sistemų paketus.

4.4. Vartotojo sąsajų projektai

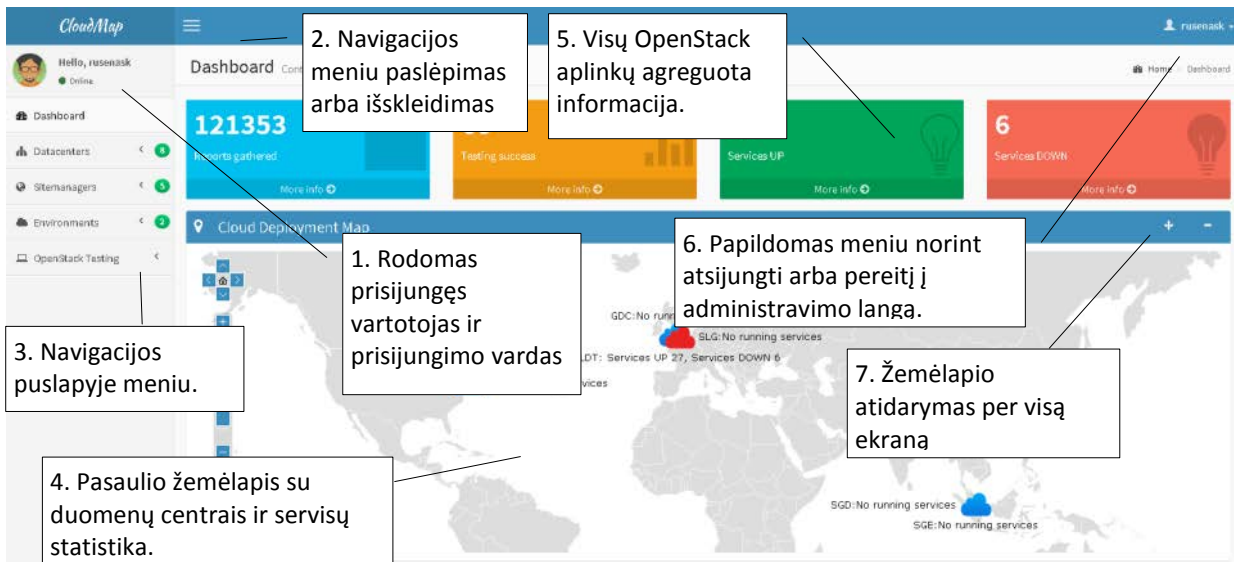
Vienas iš sudėtingiausių uždavinių kuriant informacines sistemas – nuspręsti, kur ir kaip bus atvaizduojama informacija, siekiant ją padaryti suprantamą bet kuriam sistemos naudotojui. Projektuojant SiteManager esantį OpenStack duomenų atvaizdavimą iškilo problemų, kai turint itin didelį kiekį informacijos – reikia išlaikyti praktiškai visus duomenis, tačiau juos pateikti naudotojui tvarkingai ir perdaug jo neapkraunant. Panašios problemos kartojosi ir CloudMap sistemoje, kai reikėjo suskirstyti duomenis į kategorijas pagal tam tikrus tipus, tačiau išlaikyti duomenų hierarchiją.

4.4.1. CloudMap pagrindinio puslapio sąsaja

Kad sistemos naudotojui užtektų būti atsidarius vieną puslapio langą siekiant gauti informaciją apie praktiškai visą debesų kompiuterijos sprendimo būklę – po daugelio iteracijų buvo prieita prie šio dizaino. Pagrindinis reikalavimas šiai sąsajai – ji turi gerai atrodyti dideliame ekrane, taigi iš to kyla keli kiti dizainui skirti reikalavimai:

- Žemėlapis turi būti vektorinės grafikos prigimties, kad būtų neribotai didinamas ir nepradėtų matytis pavieniai pikseliai.
- Neturėtų būti daug smulkiai pateiktos informacijos.

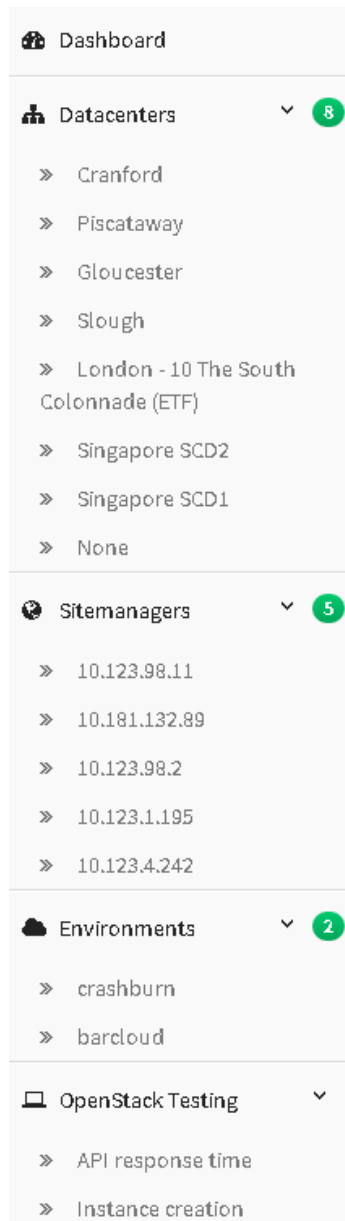
Žemiau paveikslėlyje pateiktas sistemos langas:



45. pav. CloudMap pagrindinio lango vartotojo sąsaja

Iš šio paveikslėlio matome, kad reikia papildomai sufokusuoti žemėlapi, nes nereikalingas viso pasaulio vaizdas. Taip pat reikėjo papildomai suderinti duomenų centro informacinės eilutės poziciją, kad jos nepersidengtų, dėl to, kad praktiškai visi duomenų centrai yra pastatyti poromis (dubliuoja vienas kitą avarijos atveju). Toliau pateikiame papildomą kiekvieno komponento aprašymą:

1. Šioje vietoje atvaizduojamas vartotojo prisijungimo vardas.
2. Šis mygtukas skirtas suskleisti arba išskleisti navigacijos meniu, esantį kairėje. Šis funkcionalumas naudingas, kai norima stebėti tik pagrindinę informaciją.
3. Navigacijos meniu projektavime stengiamasi taip pat sutalpinti kuo didesnę informaciją, tačiau vaizduojant ją kompaktiškai, išskleisto navigacijos meniu paveikslėlis pateiktas žemiau:

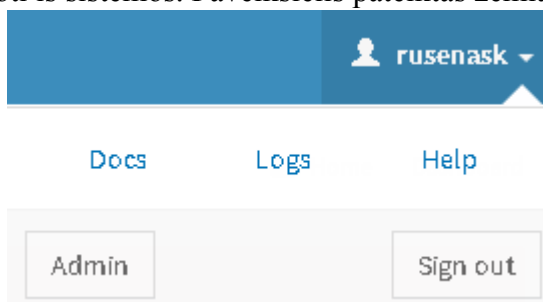


46. pav. Išskleistas navigacijos meniu

Iš šio meniu pavyzdžio galime matyti, kiek daug informacijos galima sutraukti į sąlyginai mažą plotą ir sistemos naudotojui pačiam leisti rinktis, kurią informaciją jis nori matyt.

4. Žemėlapių komponento sukūrimas buvo viena iš sudėtingiausių užduočių stiliaus atžvilgiu. Buvo išbandytos kelios JavaScript bibliotekos vektorinei grafikai kurti. Viena iš galimų alternatyvų *amCharts* produktui buvo *JvectorMap*, tačiau trūko funkcionalumo, kuris leistų rodyti statinį vaizdą žemėlapyje. Šis funkcionalumas buvo reikalingas norint atvaizduoti OpenStack servisų būklę duomenų centruose. Pačiai žemėlapių informacijai užpildyti (duomenų centram ir informacijai apie juos) naudojama JSON tipo informacija, kuri paduodama specialioje struktūroje.
5. Viršutiniai elementai skirti bendrai statistikai. Ateityje planuojama sukurti ir įdiegti automatinį OpenStack aplinkų testavimą, taigi kiekvieną dieną būtų galima pateikti testavimo statistiką. Bendrą raportų kiekį turint daugiau SiteManager taikomųjų programų duomenų centruose būtų galima pakeisti į raportų per valandą skaičių ir jo delta. Kiti parametrai, tokie kaip veikiantys/neveikiantys servais turėtų išlikti. Paspaudus ant bet kurio iš šių elementų išskviečiamas detalesnis puslapis. Ši informacija atnaujinama automatiškai kas 10 sekundžių.
6. Papildomas meniu norint nueiti į puslapio administravimo panelę reikalingas, kai yra poreikis redaguoti bet kurią informaciją, pavyzdžiui suteikti SiteManager

pavadinimą (standartiškai jie identifikuojami pagal IP adresą) arba duomenų centro detales (pridėti daugiau duomenų centrų, redaguoti koordinatės). Taip pat šis meniu skirtas ir išsiregistruoti iš sistemos. Paveikslėlis pateiktas žemiau:



47. pav. Papildomas meniu

Šis meniu taip pat turi nuorodas į dokumentaciją, atliktų veiksmų įrašus bei pagalbą.

7. Žemėlapiu atidarymas per visą ekraną leidžia naujame naršyklės lange atidaryti tik žemėlapiu atvaizdą, kuris irgi atnaujinamas kas 10 sekundžių, dėl vektorinės prigimties – jis teisingai išsitempia bet kokio dydžio ekrane.

4.4.2. CloudMap duomenų centrų puslapio sąsaja

Duomenų centrai CloudMap sistemoje užima svarbią vietą. Kitaip nei SiteManager sistemoje – čia jie turi papildomus koordinacių atributus ir naudojami kaip aukštesnio lygmens objektai, prie kurių per išorinius raktus priskiriami kiti objektai – servais ir SiteManager sistemos.

Žemiau pateikta duomenų centrų atvaizdavimo lentelė:

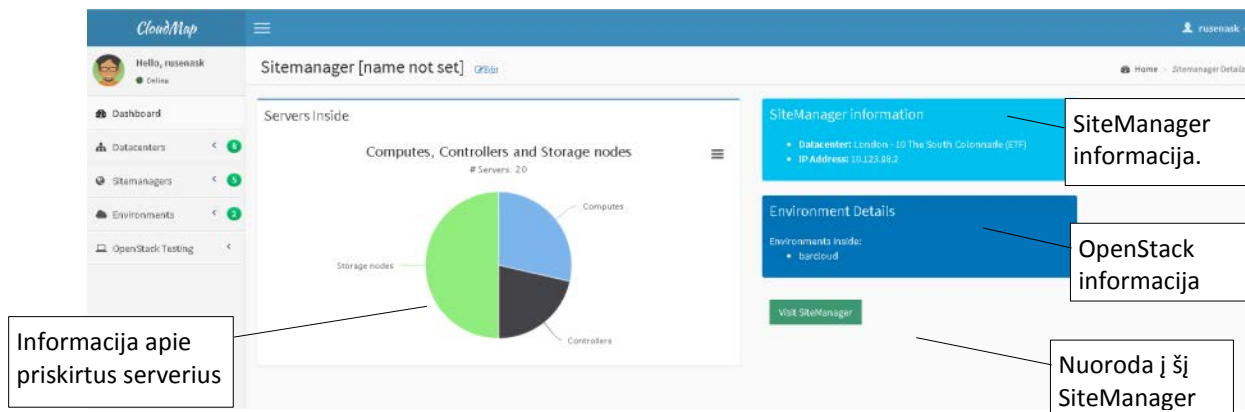
ID	Name	Code	Coords	SiteManagers inside
1	Cranford	CRC	40.85, -74.29	No SiteManagers found in this DC
2	Piscataway	PIC	40.24, -74.86	No SiteManagers found in this DC
3	Gloucester	GDC	51.86, -2.24	No SiteManagers found in this DC
4	Slough	SLG	51.51, -0.59	No SiteManagers found in this DC
	London - 10 The South Colonnade (ETF)	LDT	51.30, 0.07	<ul style="list-style-type: none"> [name.net.set] 10.123.98.11 [name.net.set] 10.123.98.2
	Singapore SCD2	SQE	1.29, 103.78	No SiteManagers found in this DC
	Singapore SCD1	SQD	1.32, 103.89	No SiteManagers found in this DC
8	None	NJS	None, None	[name.net.set] 10.181.132.89

48. pav. Duomenų centrų vartotojo sąsaja

Čia galima matyti naujus užregistruotus duomenų centrus (jei SiteManager registruodamasis pateikia naują, dar nežinomą duomenų centrą – jis automatiškai sukuriamas, tačiau nevaizduojamas žemėlapyje, nes nėra žinomos jo koordinatės).

4.4.3. CloudMap detalios informacijos apie SiteManager puslapio sąsaja

Kiekviena SiteManager, registruodamasis, siunčia savo turimų resursų statistiką. Šio puslapio pagrindinė užduotis – vaizdžiai ir aiškiai parodyti šią informaciją. Puslapio vaizdas pateiktas paveikslėlyje žemiau:



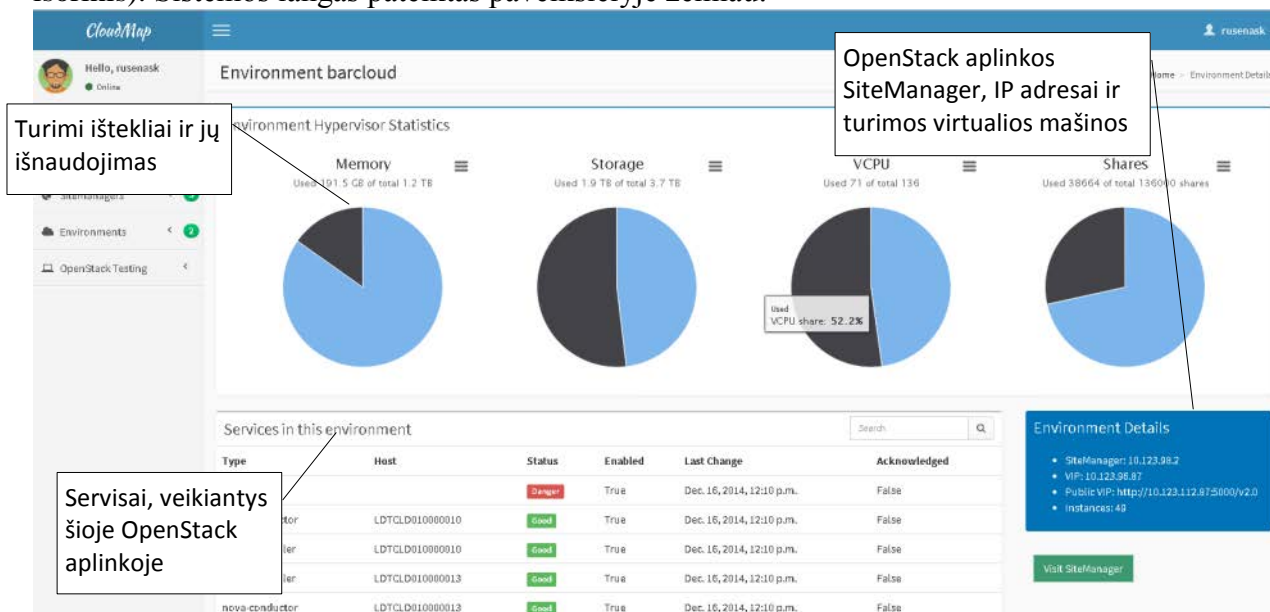
49. pav. CloudMap sistemoje registruoto SiteManager detalios informacijos langas

Iš šio paveikslėlio matome, jog daug informacijos apie SiteManager nekaupiami. Pagrindinė informacija, dominanti sistemos naudotoją:

- Nuoroda į SiteManager
- Serverių tipai ir kiekiai
- Turimos OpenStack aplinkos

4.4.4. CloudMap detalios informacijos apie OpenStack aplinką puslapio sąsaja

Šiame skyriuje pateikiama detali OpenStack informacija CloudMap sistemoje. Skirtingai nuo SiteManager detalios OpenStack aplinkos informacijos – čia nėra detalizuotos virtualios mašinos, operacinių sistemų atvaizdai ir t.t. Šioje sistemoje saugoma tik bendra informacija – turimi išteklių ir jų išnaudojimas, sukurtų virtualių mašinų kiekis, OpenStack aplinkos IP adresai (vidinis ir išorinis). Sistemos langas pateiktas paveikslėlyje žemiau:

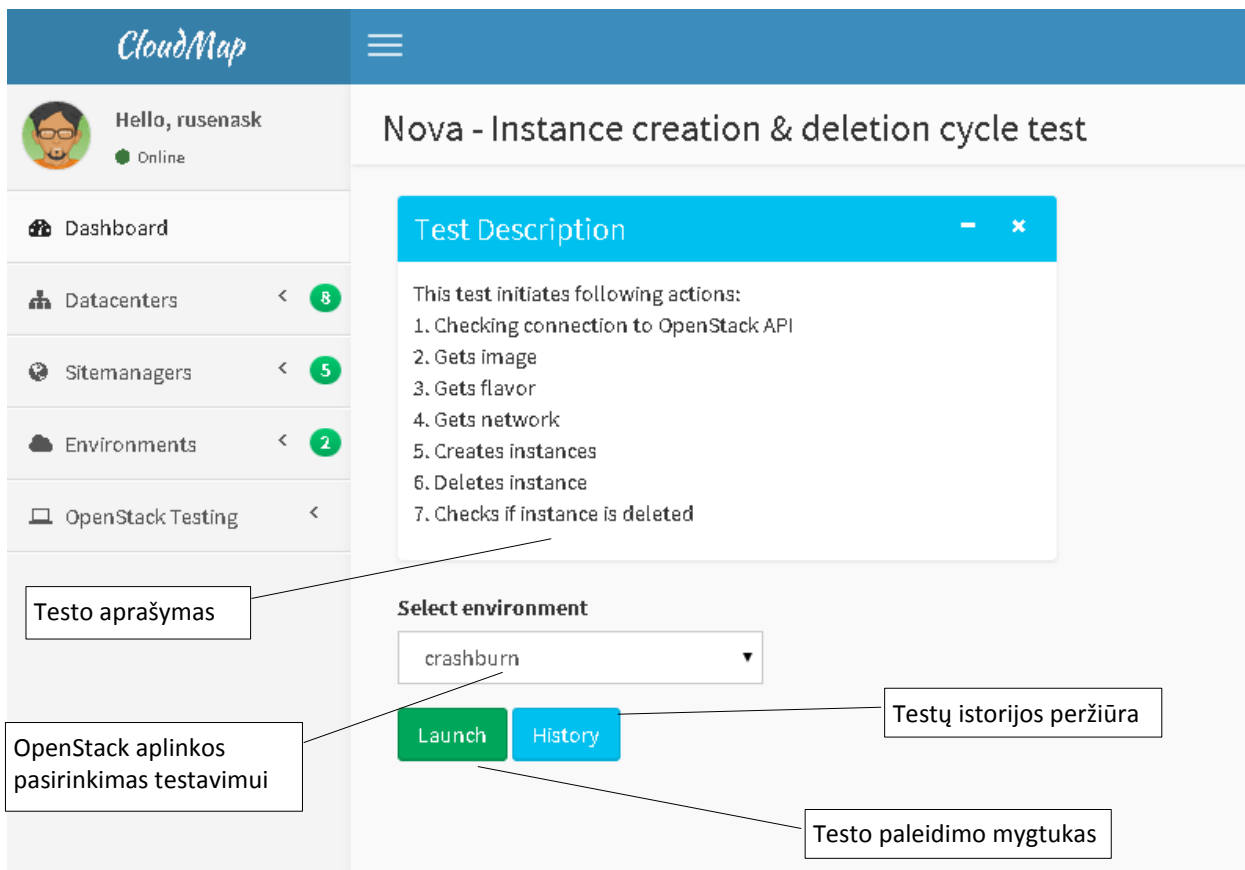


50. pav. CloudMap sistemoje OpenStack aplinkos detalios informacijos langas

Iš aukščiau esančio paveikslėlio matome, kad prie OpenStack aplinkos informacijos taip pat yra pateiktas ir veikiančių servisų sąrašas. Prie kiekvieno elemento yra pateiktas jo tipas, jei tai Nova komponentui priklausantis servisas – jis dar turi priskirtą serverio vardą, kuriame jis veikia. Taip pat prie servisų pateikiama jų būklė bei paskutinis būklės pasikeitimo laikas.

4.4.5. CloudMap testų paleidimo vartotojo sąsaja

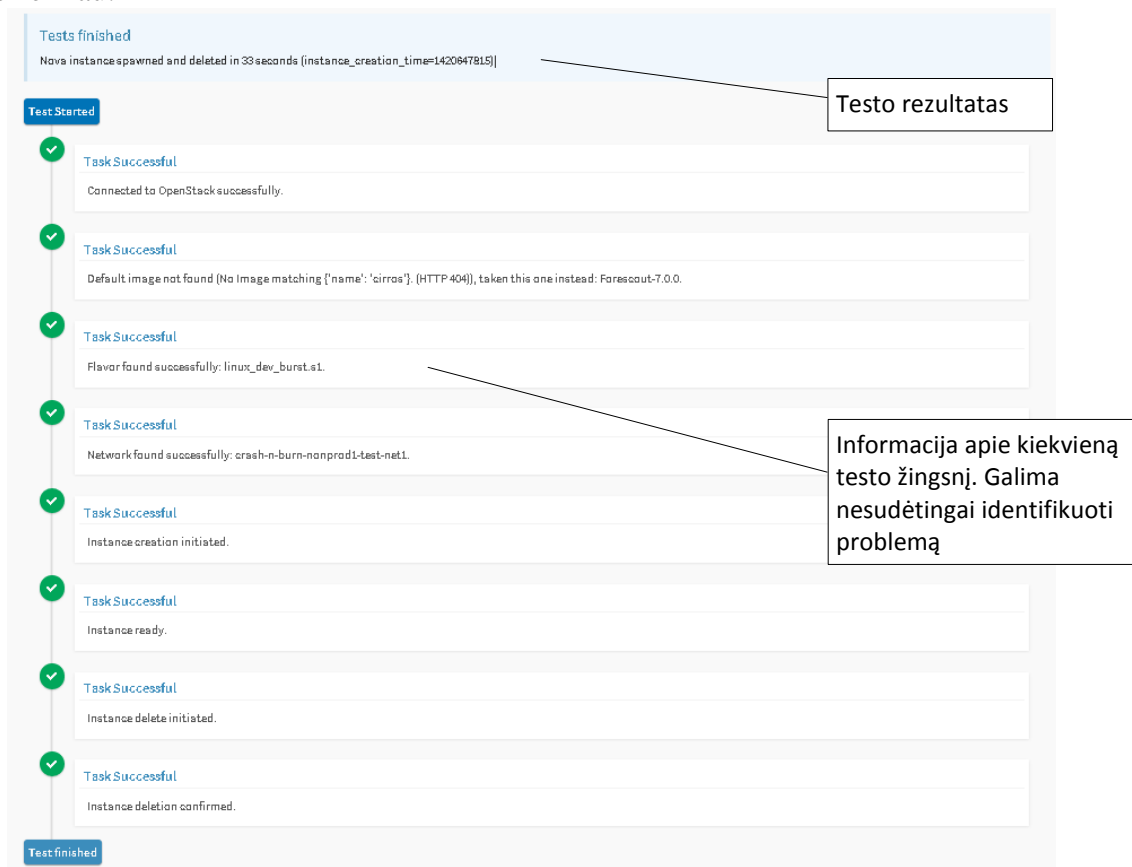
Testavimo sąsaja yra vienoda tiek API tikrinimo testui, tiek ir virtualios mašinos kūrimo testui, skiriasi tik aprašymas. Sąsajos paveikslėlis pateiktas žemiau:



51. pav. OpenStack testavimo sąsajos langas

Norint paleisti testą – pasirenkama OpenStack aplinka ir spaudžiamas mygtukas „Launch“, kadangi testai užtrunka tam tikrą laiką, lango dešinėje vartotojui pranešama, kad užduotis vykdoma.

Šio testo detalų veiksmų aprašymą galima rasti prie veiklos diagramų. Testų rezultatas pateiktas žemiau:

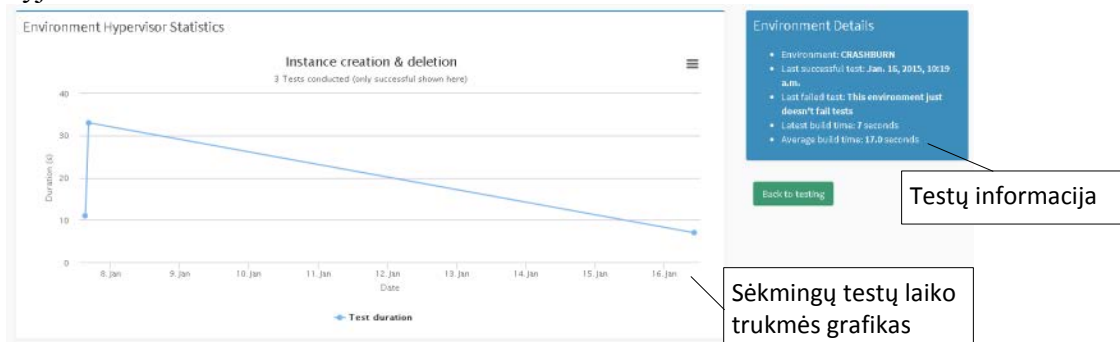


52. pav. OpenStack virtualios mašinos kūrimo testo pavyzdys

Turint tokį testo įvykių sekimą ir atvaizdavimą tampa labai paprasta identifikuoti problemą ir ją pašalinti. Pirminėse testo versijose, nebuvo įdiegtas funkcionalumas, kai CloudMap ieško alternatyvių išteklių, jeigu numatytieji operacinių sistemų atvaizdai arba virtualių mašinų šablonai neegzistuoja – testas pasibaigė net neprasidėjęs virtualios mašinos kūrimui. Tuomet buvo sukurta ir įdiegta nauja funkcija, kuri ieško alternatyvų. Viso testo metu naudojama strategija, kai stengiamasi nueiti kuo toliau, nepaisant įvykusių neatitikimų.

4.4.6. OpenStack virtualios mašinos kūrimo ir ištrynimo testavimo istorijos sąsajos langas

Testų istorija yra svarbi, kai norima išvelgti tendencijas, tam tikrame veiksmo, kurio atlikimo metu vykdomi keli kiti veiksmai, kurie gali daryti įtaką rezultato gavimui. Puikus pavyzdys yra virtualios mašinos kūrimas ir ištrynimas. Šio testo metu (daugiau apie jį galima sužinoti iš jo veiklos diagramos) veikia informacijos tikrinimo ir duomenų registravimo mechanizmai, kurie praneša apie naujos virtualios mašinos sukūrimą kitoms įmonės sistemoms. Dėl daugelio veikiančių dalių – būtina sekti testavimo duomenis ir užkirsti kelią galimai sistemos avarijai (jeigu pastebimai ilgėja mašinos kūrimo laikas nuo 7 sekundžių iki kelių valandų arba dienos). Ši sąsaja pateikta paveikslėlyje žemiau:



53. pav. OpenStack virtualios mašinos kūrimo ir ištrynimo testavimo istorijos sąsaja

4.4.7. SiteManager serverių sąrašo langas

Šis langas skirtas sistemos naudotojams dirbti su fiziniiais serveriais (teoriškai galima prisidėti ir virtualių mašinų, tačiau gali kilti problemų priskiriant operacinių sistemų atvaizdus ir nebus galima išnaudoti IPMI funkcionalumo). Pagrindinės funkcijos pasiekiamos vos per keturis mygtukus (vienas iš jų savyje turi vidines funkcijas, skirtas dirbti iškart su daugeliu serverių). Lentelė interaktyvi tampa su *JavaScript Data Tables* įskiepiu, tampa galimos tokios funkcijos kaip duomenų eksportas į CSV, PDF, Excel formatus bei filtravimas realiu laiku.

Pirmosiose šio puslapio versijose buvo naudojamas *Django pagination* funkcionalumas, derinamas kartu su *complex query* biblioteka, taip optimizuojant ir apribojant rodomų duomenų kiekį. Tačiau perėjus prie strategijos, kai SiteManager sistema yra naudojama aptarnaujant vieną duomenų centrą (o ne viso pasaulio korporacijos duomenų centrus), nebebuvo prasmės optimizuoti šioje vietoje ir buvo pasirinkta atiduoti visą turimą serverių sąrašą iškart, o filtravimą atlikti kliento pusėje.

Vartotojo sąsaja pateikta žemiau:

1. Serverių pridėjimo ir valdymo mygtukai

Hostname	MAC	IP	Serial	Model	IPMI IP	Current image	Last update	Status	IPMI details	TAGS
34383137-3630-544...	38:aa:7:8b:a8:20		CZ341232J9	ProLiant SL210K Gen8	10.123.108.20	None	Sept. 26, 2014, 3:01 p.m.	Preconfigured	✓	
34383137-3630-544...	38:aa:7:93:a8:b0		None	None	10.123.108.4	None	Sept. 29, 2014, 9:43 a.m.	Preconfigured	✓	
34383137-3630-544...	38:aa:7:8b:a5:94		CZ341232JF	ProLiant SL210K Gen8	10.123.108.42	None	Nov. 21, 2014, 2:28 p.m.	Preconfigured	✓	
34383137-3630-544...	38:aa:7:93:a7:58		CZ341232JJ	ProLiant SL210K Gen8	10.123.108.7	None	Nov. 26, 2014, 2:18 p.m.	Preconfigured	✓	
34383137-3630-544...	38:aa:7:8b:8a:94		None	None	10.123.108.6	None	Sept. 29, 2014, 9:40 a.m.	Preconfigured	✓	
34383137-3730-544...	38:aa:7:8b:86:00		None	None	10.123.108.53	None	Sept. 29, 2014, 9:42 a.m.	Preconfigured	✓	
labostmp11	38:aa:7:93:a8:bc		CZ341232KE	ProLiant SL210K Gen8	10.123.108.32	None	Sept. 29, 2014, 9:41 a.m.	Preconfigured	✓	
labostmp12	38:aa:7:93:a7:78		CZ341232KM	ProLiant SL210K Gen8	10.123.108.40	None	Sept. 29, 2014, 9:42 a.m.	Preconfigured	✓	
labostmp13	38:aa:7:93:a8:b4		None	None	10.123.108.44	None	Sept. 29, 2014, 9:43 a.m.	Preconfigured	✓	
labostmp15	38:aa:7:93:a8:a0		CZ341232KB	ProLiant SL210K Gen8	10.123.108.46	None	Sept. 29, 2014, 9:42 a.m.	Preconfigured	✓	

2. Informacija apie kiekvieną serverį.

54. pav. SiteManager serverių sąrašo langas

3. Duomenų eksportavimo mygtukai ir filtravimo laukas

Detalesni vartotojo sąsajos paaiškinimai:

4.4.7.1. Serverių valdymo mygtukai (1)

Šie mygtukai skirti pridėti ir valdyti serverius:



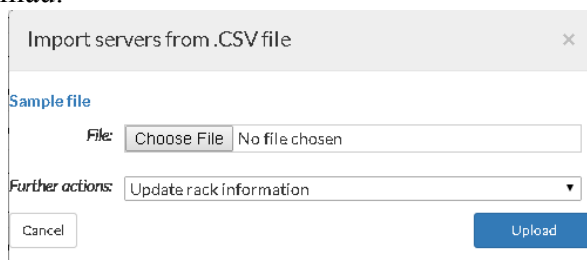
55. pav. Serverių valdymo mygtukai

Mygtukų funkcijos ir paaiškinimai:

- **Create.** Skirtas rankiniu būdu pridėti egzistuojantį serverį į SiteManager duomenų bazę. Paspaudus šį mygtuką iššoka naujas langas, į kurį reikia suvesti serverio duomenis. Teisingai suvedus IPMI duomenis SiteManager sistema daugelį kitų laukelių užpildys automatiškai. Sąsajos langas paveikslėlyje žemiau:

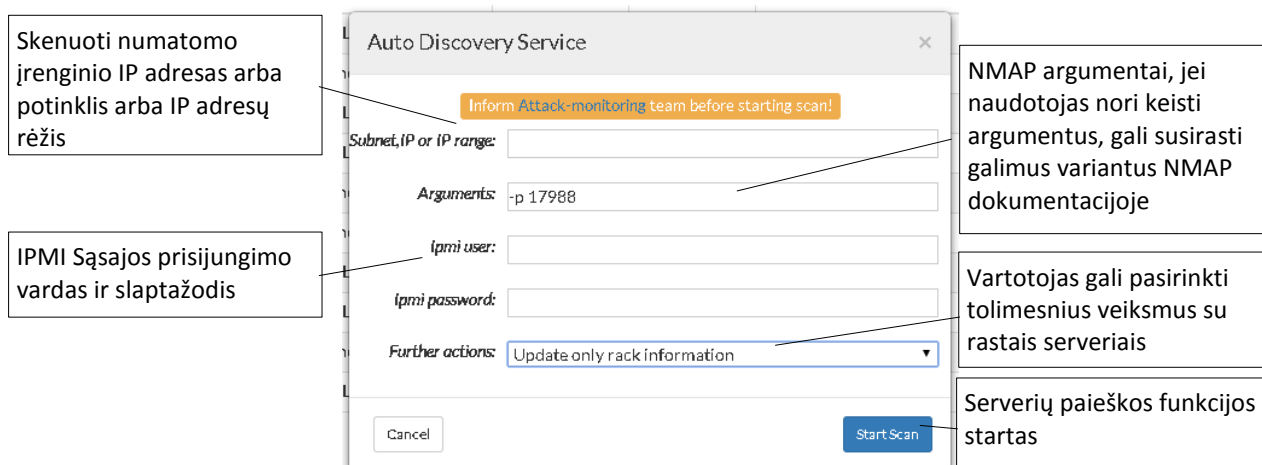
56. pav. Serverio pridėjimo į SiteManager sistemą langas

- **Import.** Šis mygtukas skirtas pridėti neribotą kiekį serverių pateikiant duomenis iš CSV failo. Šiame lange galima atsisiųsti pavyzdinį Excel formato failą, nes stulpeliai turi būti sudėlioti tam tikra tvarka. Taip pat galima pasirinkti sekančius veiksmus, kuriuos SiteManager turi atlikti po serverių pridėjimo. Šios funkcijos detalų aprašymą galima surasti prie veiklos diagramų. Sąsajos vaizdas paveikslėlyje žemiau:



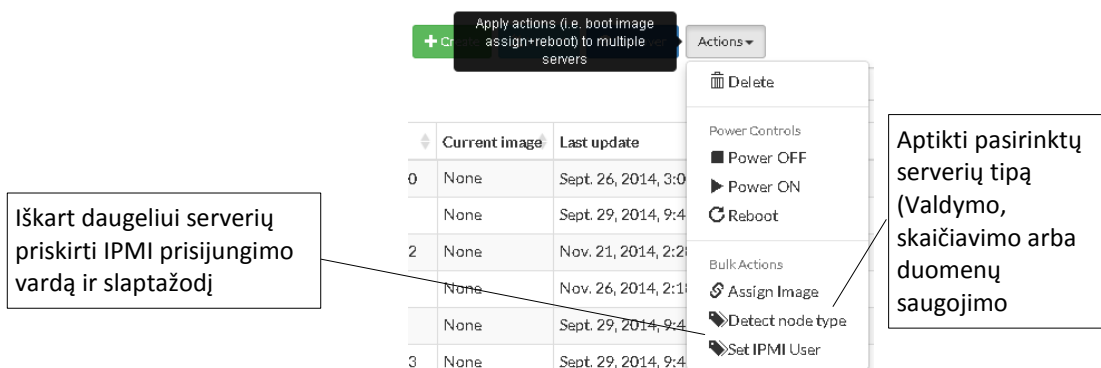
57. pav. Serverių pridėjimo iš CSV failo sąsajos langas

- **Discover.** Tai – invazinė funkcija ir prieš ją naudojant reikia perspėti už tinklo saugumą atsakingą komandą. Naudojama sistemoje įdiegta programa – NMAP. Kaip ir anksčiau minėjote importavimo iš CSV failo funkcijoje – čia irgi galima pasirinkti, ką norima toliau daryti su serveriais. Detalų funkcijos paaiškinimą galite rasti veiklos diagramose. Sąsajos langas pateiktas paveikslėlyje žemiau:



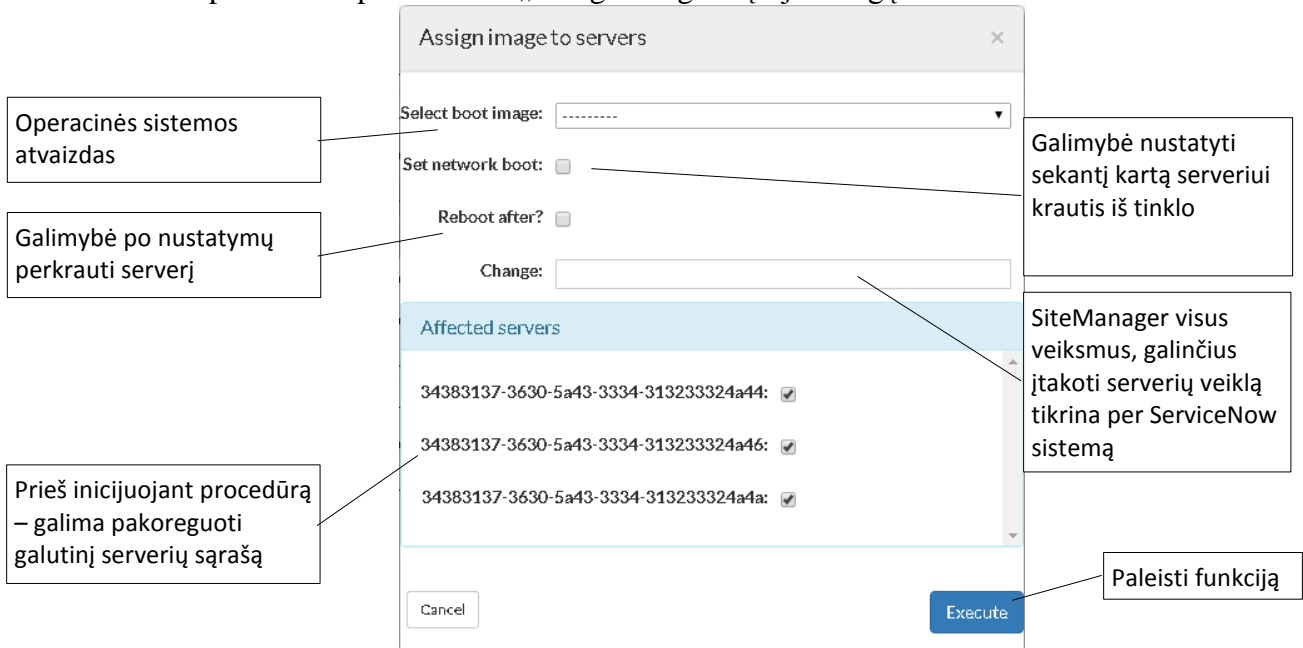
58. pav. Serverių pridėjimo skenuojant tinklą sąsajos langas

- **Actions.** Šis mygtukas savyje turi kelias funkcijas, skirtas vienu metu valdyti vieną ar daugiau serverių. Meniu dalinamas į kelias pagrindines skiltis:
 - „Power Controls“ – valdo serverio būseną per IPMI sąsają.
 - „Bulk Actions“ – atlieka sudėtingas funkcijas, kurios susideda iš kelių žingsnių, todėl sutaupo laiko ir palengvina sistemos naudotojų darbą.
 Meniu pavaizduotas paveikslėlyje žemiau:



59. pav. Serverių veiksmų mygtuko meniu

Toliau pateikiame pasirinkimo „Assign image“ sąsajos langą:



60. pav. „bulk image assign“ funkcijos sąsajos langas

Šiame lange galime matyti, jog pasirinktiems serveriams galima pritaikyti iškart kelis veiksmus. Tai reikalinga, kai norima iškart į OpenStack sistemą prijungti daugelį skaičiavimo serverių.

4.4.7.2. Serverių informacija (2)

Serverių sąsaja buvo sukurta dirbant kartu su inžinieriais, kurie ketina ja naudojasi. Vienas iš sudėtingesnių uždavinių – išrinkti duomenų laukus, kurie svarbūs sistemos naudotojui. Taip pat, derinant su realaus laiko serverių sąrašo filtravimu – svarbi paieška per bet kurį serverio parametą.

Serverių sąrašas vartotojo sąsajoje pateiktas paveikslėlyje žemiau:

Hostname	MAC	IP	Serial	Model	IPMI IP	Current image	Last update	Status	IPMI details	TAGS
34383137-3630-5a4...	38:ea:a7:8b:a8:00		CZ341232J9	ProLiant SL210t Gen8	10.123.108.20	None	Sept. 26, 2014, 3:01 p.m.	Preconfigured	✓	
34383137-3630-5a4...	38:ea:a7:93:e8:b0		None	None	10.123.108.4	None	Sept. 29, 2014, 9:43 a.m.	Preconfigured	✓	
34383137-3630-5a4...	38:ea:a7:8b:a5:84		CZ341232JF	ProLiant SL210t Gen8	10.123.108.42	None	Nov. 21, 2014, 2:28 p.m.	Preconfigured	✓	
34383137-3630-5a4...	38:ea:a7:93:e7:58		CZ341232JJ	ProLiant SL210t Gen8	10.123.108.7	None	Nov. 26, 2014, 2:18 p.m.	Preconfigured	✓	
34383137-3630-5a4...	38:ea:a7:8b:a8:94		None	None	10.123.108.6	None	Sept. 29, 2014, 9:43 a.m.	Preconfigured	✓	
34383137-3730-5a4...	38:ea:a7:8b:88:00		None	None	10.123.108.33	None	Sept. 29, 2014, 9:42 a.m.	Preconfigured	✓	
labostcmp11	38:ea:a7:93:e8:bc		CZ341232KE	ProLiant SL210t Gen8	10.123.108.32	None	Sept. 29, 2014, 9:41 a.m.	Preconfigured	✓	
labostcmp12	38:ea:a7:93:e7:78		CZ341232KM	ProLiant SL210t Gen8	10.123.108.40	None	Sept. 29, 2014, 9:42 a.m.	Preconfigured	✓	
labostcmp13	38:ea:a7:93:e8:b4		None	None	10.123.108.44	None	Sept. 29, 2014, 9:43 a.m.	Preconfigured	✓	
labostcmp15	38:ea:a7:93:e8:e0		CZ341232K8	ProLiant SL210t Gen8	10.123.108.46	None	Sept. 29, 2014, 9:42 a.m.	Preconfigured	✓	

Showing 1 to 10 of 22 entries

Previous 1 2 3 Next

61. pav. Serverių detali informacija SiteManager sistemoje

Stulpelyje „TAGS“ atsiranda žymės, jei serveris būna priskirtas OpenStack aplinkai. Tokios konfigūracijos pavyzdys pateiktas paveikslėlyje žemiau:

Hostname	MAC	IP	Serial	Model	IPMI IP	Current image	Last update	Status	IPMI details	TAGS
LDTCLD010100002	38:ea:a7:93:e8:b0	10.123.96.29	CZ341232JD	ProLiant SL210t Gen8	10.123.108.4	template:Openstac...	Jan. 6, 2015, 9:04 a.m.	Active	✓	HA, crashburn, controller

62. pav. Serverio, priskirto OpenStack aplinkai atvaizdavimas

Naudojantis šiomis žymėmis galima greitai nustatyti, kuriai OpenStack aplinkai priskirtas serveris, jo rolė ir kokia konfigūracija naudojama (šiuo atveju HA – „High Availability“, naudojami trys arba daugiau valdymo serveriai)

4.4.7.3. Duomenų eksportavimas ir filtravimas

Šios funkcijos reikalingos turimus duomenis išvesti į kitą formatą, pavyzdžiui CSV, tuomet galima informaciją perkelti į kitą SiteManager arba naudoti kitose sistemose. Tai pat tai pravartu perinstaliuojant SiteManager (vėliau naudojant importavimo funkciją).

„Search“ laukas naudojamas filtravimui realiu laiku pagal bet kurį duomenų stulpelį, nes sistemos naudotojai dažnai ieško pagal įvairius parametrus, tokius kaip IPMI IP adresas, serverio IP adresas, modelis, serverio vardas ir t.t. Sąsajos fragmentas pavaizduotas paveikslėlyje žemiau:

Search:

63. pav. SiteManager serverių duomenų eksportavimo ir filtravimo sąsaja

4.4.8. Serverių valdymo vartotojo sąsaja

Ši vartotojo sąsaja yra vienas iš didžiausių SiteManager privalumų prieš rinkoje esančius panašius produktus, skirtus OpenStack diegimui. Ši sąsaja gali atlikti daugelį veiksmų, kurie kitu atveju pasiekiami tik per HP ILO sąsają, jungiantis tiesiai į serverį. Sąsajos paveikslėlis pateiktas žemiau:

The screenshot shows the 'laboscmp11 LOM Details' page in SiteManager. It is divided into two main sections: 'Host Data' and 'Location Services'. The 'Host Data' section lists server specifications such as Model (ProLiant SL210t Gen8), CPU count (40), CPU type (64-bit Capable), CPU speed (2.8Ghz), RAM (256 GB), and Node type (Controller node). Below this are buttons for 'Check health', 'Get more details', and 'Boot controls'. The 'Location Services' section shows rack information: Rack description (HP 42U Intelligent Series Rack), Rack ID (CN8347P021), Rack height (42 U), Server position in rack (35 U), Server height (2 U), and Server bay (1). It also displays the HP iLO interface (10.123.108.32), UID status (OFF), and buttons for 'Turn ON', 'Restart Server', and 'Refresh'. The page is annotated with several callouts: 'Serverio rolė (jeigu priskirta)' points to the 'Server role' field; 'Serverio parametrai' points to the 'Host Data' table; '1. Serverio komponentų būklės patikrinimas' points to the 'Check health' button; '2. Serverio detali informacija' points to the 'Get more details' button; 'Informacijos atnaujinimas' points to the 'Refresh' button; 'Serverio perkrovimas' points to the 'Restart Server' button; 'LED lemputės įjungimas ant serverio' points to the 'Turn ON' button; and 'Serverio vieta, nustatoma iš HP išmaniųjų serverių spintų' points to the 'Rack height' field.

64. pav. Serverio valdymo vartotojo sąsaja

Sistemos naudotojas gali greitai gauti visą reikiamą informaciją per SiteManager, jam nereikia jungtis į patį serverių, o to jis net negalėtų padaryti dėl saugumo reikalavimų. Šiai sąsajai įgyvendinti naudojamas HP-ILO paketas, skirtas Python kalbai. Naujausia sąsajos versija turi RestAPI, tačiau ji yra dar eksperimentinėje stadijoje, todėl jos negalima naudoti produkcijoje. Toliau pateikiami 1 ir 2 funkcijų pavyzdžiai:

1. Serverio komponentų būklės patikrinimas iškviečia vidinę patikros funkciją, kuri tikrina serverio komponentus ir praneša apie jų būklę, rezultatas pateiktas žemiau:

laboscmp11 Server Health	
Parameter	Status
Bios status	OK ✓
Network status	OK ✓
Fan status	OK ✓
Memory status	OK ✓
Power status	OK ✓
Processor status	OK ✓
Temperature status	OK ✓

Close

65. pav. Serverio būklės sąsaja

2. Ši funkcija leidžia realiu laiku gauti detalią serverio informaciją. Dalis šios informacijos dubliuojasi su jau turima informacija „LOM“ skiltyje, tačiau yra svarbių elementų, kuriuos kartais reikia žinoti sistemos naudotojui. Sąsajos paveikslėlis žemiau:

Host Details	
Parameter	Status
Product name	ProLiant SL210t Gen8
Product system serial	CZ341232KE
Hardware bios version	P83
Hardware bios date	08/02/2014
Total CPU count	20
Total vCPU count	40
CPU Speed	2.8Ghz
CPU technology	64-bit Capable
Total memory	256 GB
Product UUID	34383137-3730-5A43-3334-313233324B45
UUID	718407CZ341232KE

Close

66. pav. Serverių detalios informacijos vartotojo sąsaja

Šiame lange matomas serverio serijinis numeris, BIOS programinės įrangos atnaujinimo data, CPU ir vCPU skaičius bei produkto unikalus identifikacinis numeris.

4.4.9. Serverių spintų vartotojo sąsaja

SiteManager esantis serverių spintų modelis taip pat yra unikalus elementas ir kitos panašios sistemos nepasižymi tokiomis savybėmis. Serverių spintų modeliai kuriami dinamiškai, nuskaitant turimų serverių IPMI sąsajas ir pagal vietos nustatymo posistemį sugeneruojant serverių spintos vaizdą. Sistemos naudotojui nereikia jo formuoti, užtenka, kad būtų teisingi IPMI duomenys.

CN8347P021 details

Properties History Rack View

Datacenter - None [see more](#)

Rack ID - CN8347P021, 42U

Serverių spintos vaizdas

Serverių informacija

Galimybė valdyti serverius tam tikroje spintoje

Servers in rack

#	Server	Model	Height	Position	Bay	Role	Env.	Action
+	ldtobjstg05	ProLiant SL4540 Gen8	4 U	15	0			⏏ ⏪ ⏩
+	ldtobjstg06	ProLiant SL4540 Gen8	4 U	20	3			⏏ ⏪ ⏩
+	34383137-3630-5a43-3334-313233324a4a	ProLiant SL210t Gen8	2 U	29	3			⏏ ⏪ ⏩
+	34383137-3630-5a43-3334-313233324a46	ProLiant SL210t Gen8	2 U	31	1			⏏ ⏪ ⏩
+	34383137-3630-5a43-3334-313233324a39	ProLiant SL210t Gen8	2 U	31	2			⏏ ⏪ ⏩
+	labostcmp12	ProLiant SL210t Gen8	2 U	33	3			⏏ ⏪ ⏩
+	labostcmp11	ProLiant SL210t Gen8	2 U	35	1			⏏ ⏪ ⏩
+	labostcmp15	ProLiant SL210t Gen8	2 U	37	2			⏏ ⏪ ⏩

67. pav. Serverių spintos vaizdas

Poreikis turėti tokį vartotojo sąsajos elementą atsiranda tuomet, kai serverių spintos yra komplektuojamos gamykloje tam tikra konfigūracija. Vienos spintos skirtos įgalinti debesų kompiuterijos sprendimą (turi valdymo serverius, skaičiavimo serverius ir kelis duomenų saugojimo serverius), kitos spintos skirtus tik skaičiavimo serveriams arba tik duomenų saugojimui. Taip pat valdymo serveriai dažniausiai turi būti iš tos pačios spintos, siekiant turėti maksimalią greitaveiką ir paprastesnį tinklo konfigūravimą. Detalus serverių spintos formavimo algoritmas pateiktas veiklos diagramose.

4.4.10. SiteManager naujos OpenStack aplinkos kūrimo vartotojo sąsaja

Aplinkos kūrimo sąsajoje galima nurodyti keletą pradinių parametų OpenStack aplinkai. Detalesnė informacija pateikta paveikslėlyje žemiau:

Create New Cloud Environment

Pavadinimas. Tai kosmetinis atributas, neturintis didelės reikšmės.

Name:

Build:

Diegimo tipas, pavyzdžiui HA – didelio patikimumo, arba „stand-alone“ – vienas serveris veikiantis kaip visas OpenStack

Deployment mode: HA

Datacenter:

OpenStack versijos pasirinkimas. Kadangi tai vienas iš sparčiausiai plėtojamų projektų pasaulyje – tikėtina, kad turėsime vienu metu net kelias produkcinės versijas

Duomenų centras. Reikalingas parametras, siekiant vėliau jį perduoti į CloudMap sistemą

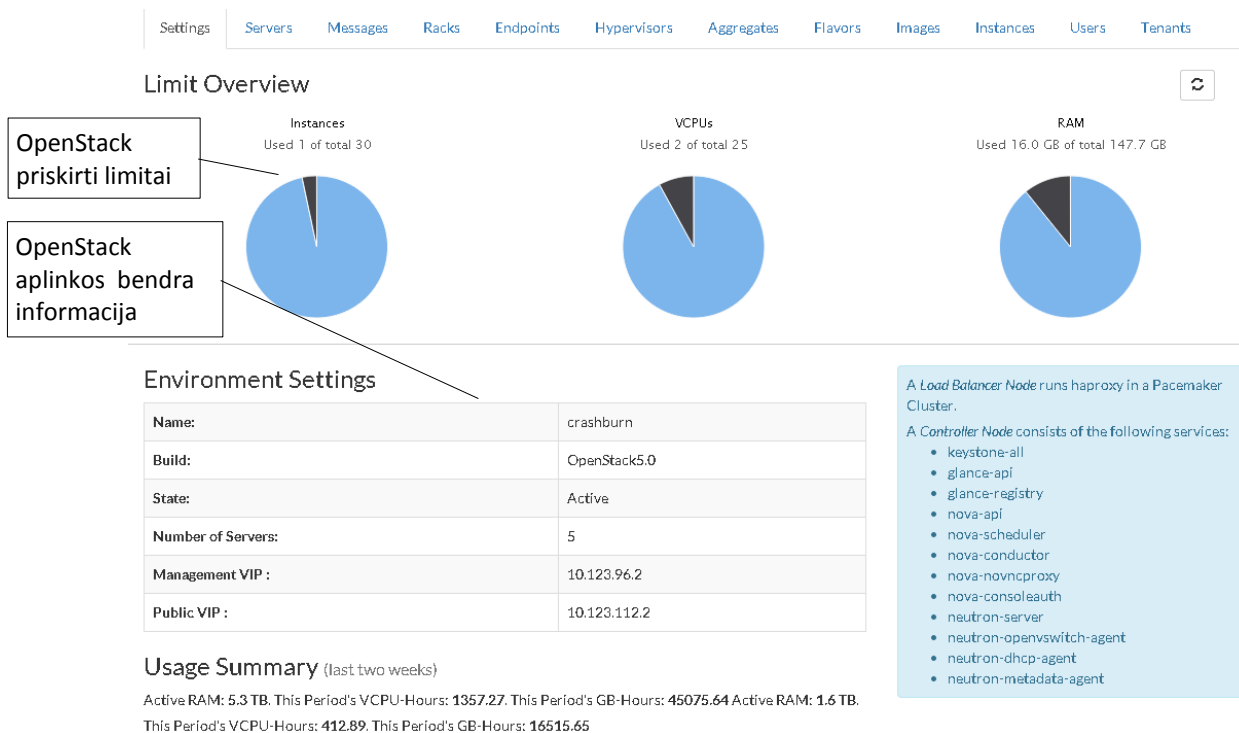
Cancel Create

68. pav. OpenStack aplinkos kūrimo lentelė

Planuojama turėti tik vieną OpenStack aplinką viename SiteManager, taigi šis langas nebus dažnai naudojamas.

4.4.11. SiteManager OpenStack nustatymų vartotojo sąsaja

Toliau pateikiamas aktyvios OpenStack aplinkos nustatymų langas. Jame matome bendrą informaciją apie aplinką, jos IP adresus, būklę, vardą, versiją bei priskirtų fizinių resursų kiekį. Taip pat viršuje esančiuose grafikuose matome OpenStack aplinkos limitus bei jų išnaudojimą

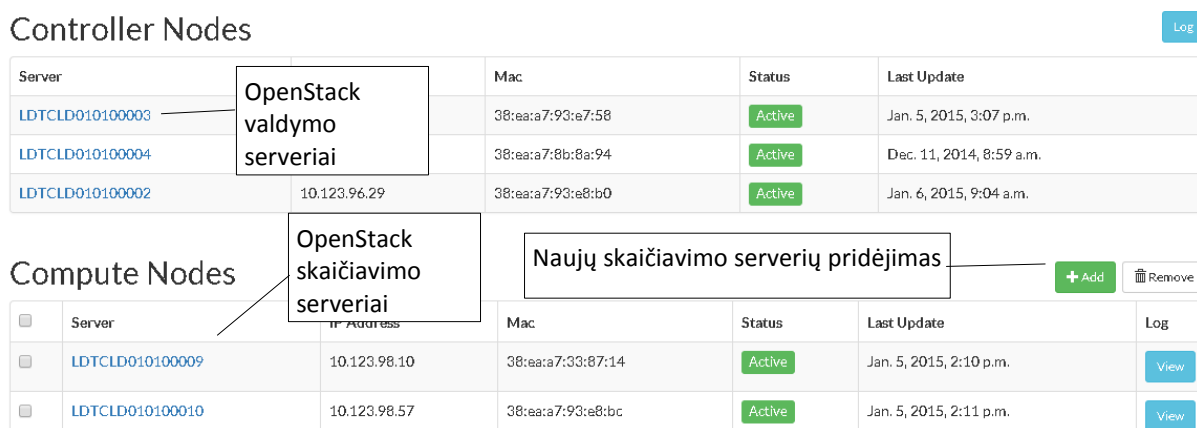


69. pav. OpenStack aplinkos pagrindinių nustatymų langas

Šiame lange svarbiausi elementai – aplinkos IP adresai. Taip pat, esant poreikiui, sąsaja galėtų būti modifikuota, kad leistų keisti pagrindinius parametrus, tačiau tai nėra šios kūrimo fazės reikalavimuose.

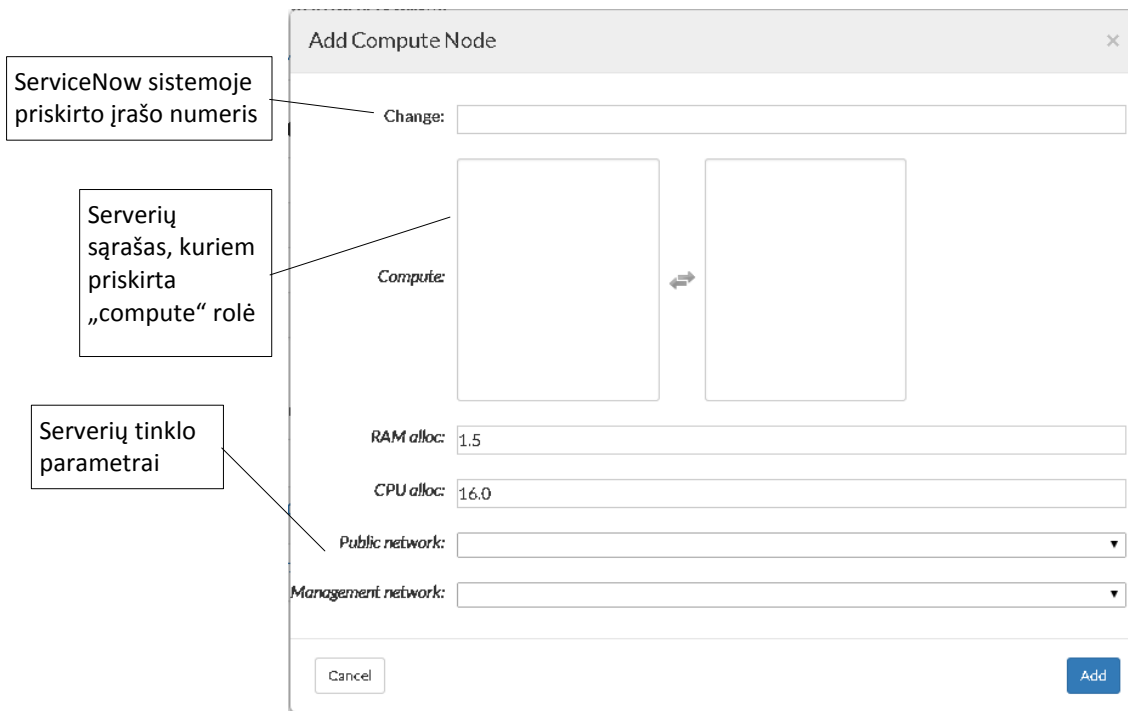
4.4.12. SiteManager OpenStack serverių vartotojo sąsaja

Ši sąsaja skirta pridėti naujus skaičiavimo serverius prie OpenStack aplinkos. Sąsaja neturi sudėtingų konfigūracijos pasirinkimų, nes kuriant OpenStack aplinką visa reikalinga konfigūracija jau atlikta. Taip pat per šią sąsają galima pasiekti diegimo metu išsaugotą informaciją apie atliktus veiksmus (paspaudus „Log“ mygtuką). Sąsaja pateikta paveikslėlyje žemiau:



70. pav. Naujų serverių į OpenStack aplinką pridėjimo sąsaja

Žemiau pateiktas vartotojui pasirodantis langas, kai nuspaudžiamas „Add“ mygtukas:



71. pav. Naujų skaičiavimo serverių pridėjimo langas

Šiame lange reikia pasirinkti norimus pridėti skaičiavimo serverius, dažniausiai tenka pridėti dideliais kiekiais, todėl būtinas šis funkcionalumas. Taip pat pasirenkami išoriniai ir vidiniai tinklai.

4.4.13. SiteManager OpenStack hipervizorių vartotojo sąsaja

Hipervizorius – programa, valdanti virtualiąsias mašinas ir operacines sistemas vienoje kompiuterinėje sistemoje. Šiame lange matomi hipervizorių apkrovimai grafiškai ir lentelėje.



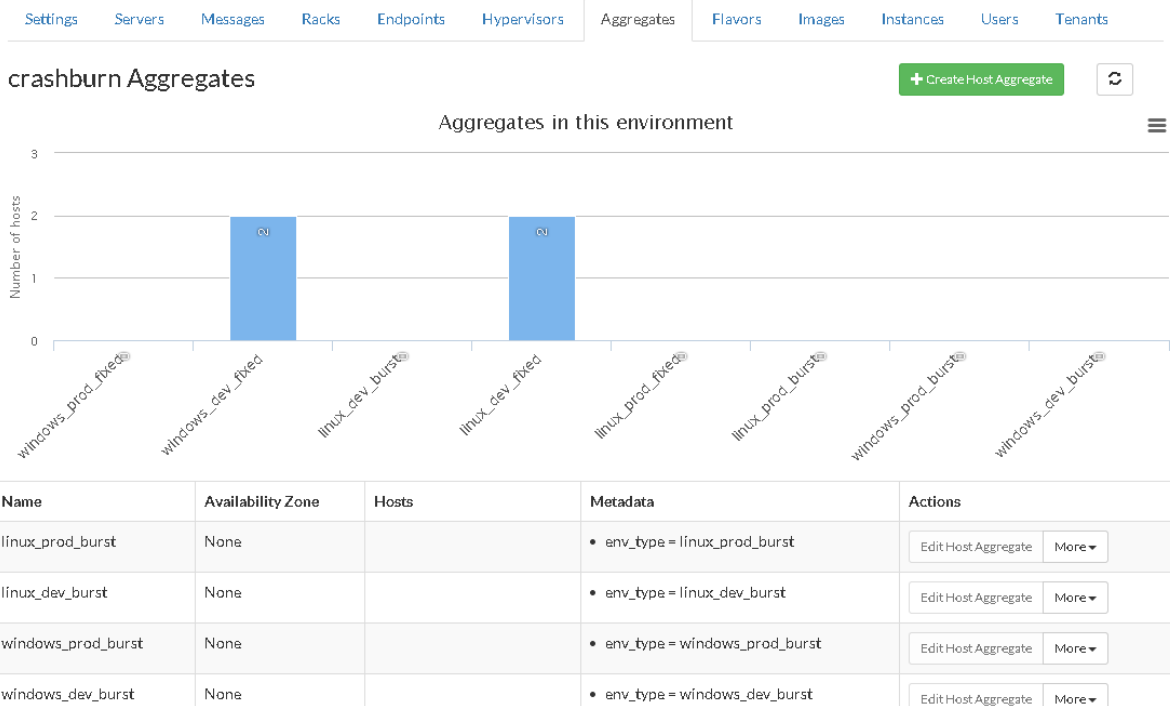
72. pav. Hipervizorių informacijos langas

Iš aukščiau esančio paveikslėlio matome, jog OpenStack talpa matuojama pagal kelis parametrus. Realybėje, praktiškai niekada nebus išnaudojami visiškai visi resursai, o bus pasiektas maksimumas pagal kažkurį parametą ir nebus leidžiama kurti daugiau virtualių mašinų. Parametru paaškinimai:

- **Memory** – sparčiosios atminties ištekliai (RAM).
- **Storage (local)** – tai lokaliai esančios duomenų saugojimo atminties ištekliai. Į šį kiekį neįeina talpa, kurią suteikia OpenStack Swift komponentas.
- **VCPU** – tai virtualių CPU kiekis.
- **Shares** – investinis dydis, kuris buvo sukurtas įmonės viduje ir įdiegtas kaip papildomas programinis modulis į OpenStack sistemą.

4.4.14. SiteManager OpenStack serverių grupavimo vartotojo sąsaja

Serverių grupavimo vartotojo sąsaja reikalinga tam, kad būtų įmanoma paskirstyti virtualių mašinų kūrimo srautus. Pavyzdžiui, leisti ant vieno serverio kurti tik Linux virtualias mašinas, o ant kito – tik Windows virtualias mašinas. Taip pat serverių grupavimas užtikrina galimybes atskirti ir virtualius kompiuterių tinklus, nes produkcijai skirtos taikomosios programos neturėtų veikti viename serveryje ir viename tinkle kartu su taikomosiomis programomis, kurios dar yra tik kuriamos. Ši sąsaja naudojama tada, kai pridedami skaičiavimo serveriai ir norima jiems paskirti serverių grupes. Taip pat buvo įdiegta galimybė ir keisti jau sugrupuotų serverių grupes, kurti naujas serverių grupes bei trinti anksčiau sukurtas. Resursų valdymui buvo sukurtas grafikas, kuriame matome serverių kiekius kiekvienoje grupėje. Sąsajos paveikslėlis pateiktas žemiau:



73. pav. OpenStack serverių grupavimo sąsaja

Aukščiau pateiktame paveikslėlyje matome ne tik serverių pavadinimus, valdymo mygtukus, tačiau ir metaduomenis, kurie nurodo, jog tik serveriai su tam tikrais ekstra parametrais gali patekti į šią serverių grupę.

Norint, kad šis filtravimo būdas būtų aktyvus – OpenStack Nova-Scheduler komponente turi būti įgalintas AggregateExtraSpecs filtras.

4.4.15. SiteManager OpenStack virtualių mašinų šablonų vartotojo sąsaja

Kadangi debesų kompiuterijos sprendimo naudotojai negalės patys nusistatyti virtualios mašinos parametrų – iš anksto yra numatyti šablonai, pagal kuriuos sistemos naudotojas galės pasirinkti norimus pajėgumus. Šiems šablonams administruoti buvo sukurta žemiau pateikta vartotojo sąsaja:

Name	vCPUs	Memory	Disk	Public	Shares per vCPU	Total shares
linux_dev_bursts1	1	2.0 GB	40 GB	Yes	312	312
linux_dev_bursts3	2	4.0 GB	40 GB	Yes	312	624
linux_dev_fixed.s3	2	8.0 GB	40 GB	Yes	1000	2000
linux_prod_bursts1	1	2.0 GB	40 GB	Yes	312	312
linux_prod_bursts3	2	4.0 GB	40 GB	Yes	312	624
linux_prod_fixed.s3	2	8.0 GB	40 GB	Yes	1000	2000
windows_dev_bursts1	1	2.0 GB	40 GB	Yes	312	312
windows_dev_bursts3	2	4.0 GB	40 GB	Yes	312	624
windows_dev_fixed.s3	2	8.0 GB	40 GB	Yes	1000	2000
windows_prod_bursts1	1	2.0 GB	40 GB	Yes	312	312

74. pav. Virtualių mašinų šablonų peržiūrėjimo sąsaja

Šie šablonai kuriami automatiškai skripto pagalba, kai diegiama OpenStack aplinka. Standartinis OpenStack komponentas Horizon leidžia jais manipuliuoti, tačiau mūsų įmonėje nebuvo reikalingas šis funkcionalumas, todėl SiteManager neleidžia šablonų nei keisti, nei redaguoti. Ši informacija reikalinga raportų kūrimams, kai susiejamos virtualios mašinos su šablonais.

4.4.16. SiteManager OpenStack operacinių sistemų atvaizdų vartotojo sąsaja

Ši sąsaja, panašiai kaip ir virtualių mašinų sąsaja, skirta atvaizduoti OpenStack vidinius išteklius, o ne jais manipuliuoti. Šioje sąsajoje atvaizduojami visi OpenStack žinomi operacinių atvaizdai, kurie bus leidžiami rinktis, kuriant virtualią mašiną. Sąsaja parodyta paveikslėlyje žemiau:

Name	Size	Status	Updated	Created	Min Disk	Min Ram	Format	Public	Protected
IBB0901-1006-Win2008R2-App-Dev	13.2 GB	Active	2 weeks, 2 days ago	1 month, 1 week ago	40GB	-	QCOW2	Yes	No
IBB1001-1002-Win2012R2-App-Dev	18.1 GB	Active	2 weeks, 2 days ago	1 month, 2 weeks ago	-	-	QCOW2	Yes	No
rhel6-650-dev-1000	5.4 GB	Active	1 month, 1 week ago	1 month, 1 week ago	40GB	-	QCOW2	Yes	No
rhel6-650-dev-servicemash	5.8 GB	Active	1 day, 21 hours ago	2 weeks, 1 day ago	40GB	-	QCOW2	Yes	No
rhel6-650-dev-tastad	5.4 GB	Active	2 weeks, 3 days ago	2 weeks, 5 days ago	40GB	-	QCOW2	Yes	No
rhel6-650-prod-1000	5.9 GB	Active	1 month, 1 week ago	1 month, 2 weeks ago	40GB	-	QCOW2	Yes	No
rhel6-650-prod-1001	6.1 GB	Active	3 weeks, 1 day ago	3 weeks, 2 days ago	40GB	-	QCOW2	Yes	No
rhel6-650-prod-tastad	6.4 GB	Active	2 weeks, 3 days ago	2 weeks, 5 days ago	40GB	-	QCOW2	Yes	No

75. pav. OpenStack operacinių sistemų atvaizdų peržiūrėjimo sąsaja

Šie atvaizdai nenumatyta, kad būtų dažnai keičiami, nes yra ilga procedūra, kol tam tikras operacinės sistemos atvaizdas būna tinkamai paruoštas naudojimui įmonės viduje.

4.4.17. SiteManager OpenStack virtualių mašinų vartotojo sąsaja

Ši sąsaja yra viena iš svarbiausių SiteManager komponentų įdiegus OpenStack aplinką. Čia rodomos visos virtualios mašinos ir jų parametrai. Sąsajos langas ir detalesnė informacija pateikti žemiau:

Fizinio serverio, kuriame sukurta virtuali mašina pavadinimas

Virtualios mašinos būklė

Projekto pavadinimas

Project	Host	Instance ID	Name	Image Name	IP Address	Flavor	Status	User	Created
100004	LDTCLD010000015	568dabe0-485c-4703-8adf-92452b9ac43f	LDTDSR000100114	rhel6-650-dev-1000	10.123.42.17	linux_dev_bursts1	Active	Noack, Brad: LEH (NYL)	Jan. 6, 2015, 6:04 p.m.
100004	LDTCLD010000015	2a7a0d48-5560-42a1-9857-08cfba9ff1a06	IBB1001-1002-Win2012R2-App-Dev-20150106-180241	IBB1001-1002-Win2012R2-App-Dev	10.123.42.16	windows_dev_bursts1	Active	Noack, Brad: LEH (NYL)	Jan. 6, 2015, 6:04 p.m.
100004	LDTCLD010000015	6b5073ec-61a2-401b-b1f9-772ec45c1a60	IBB1001-1002-Win2012R2-App-Dev-20141222-154330	IBB1001-1002-Win2012R2-App-Dev	10.123.42.15	linux_dev_bursts3	Active	Noack, Brad: LEH (NYL)	Dec. 22, 2014, 3:45 p.m.
100004	None	d4b7b243-bc8a-489f-8b52-65437fadcd44	Win2012-APP-Stable-20141219-135655	IBB1001-1002-Win2012R2-App-Dev		linux_dev_bursts1	Error	Went, David: IT (LDN)	Dec. 19, 2014, 1:58 p.m.
100004	LDTCLD010000015	2f95d5b5-e595-4f90-946d-28a1a084141a	Jamie_API_Win2008	IBB0901-1006-Win2008R2-App-Dev	10.123.42.14	windows_dev_bursts1	Active	Went, David: IT (LDN)	Dec. 16, 2014, 10:15 a.m.
000015		59c37567-1e8f-4c29-b839-5cc93f666180	DW_API_Win2008	IBB0901-1006-Win2008R2-App-Dev	10.123.42.13	windows_dev_bursts1	Active	Went, David: IT (LDN)	Dec. 16, 2014, 9:26 a.m.
000015		ecb400e2-fcaa-41ec-b204-7605f91679b2	Win2012-APP-Stable-20141215-141330	IBB1001-1002-Win2012R2-App-Dev	10.123.42.12	windows_dev_bursts1	Active	Went, David: IT (LDN)	Dec. 15, 2014, 2:15 p.m.
100004	LDTCLD010000015	a909ba6d-dc7a-4a51-8b45-03d9082d50f7	Win2012-APP-Stable-20141215-141244	IBB1001-1002-Win2012R2-App-Dev	10.123.42.11	windows_dev_bursts1	Active	Went, David: IT (LDN)	Dec. 15, 2014, 2:14 p.m.

76. pav. OpenStack aplinkoje sukurtų virtualių mašinų informacijos langas

Iš aukščiau pateikto paveikslėlio matome, jog yra papildomų mygtukų:

- **Meta** – tai metaduomenų peržiūra. Metaduomenys generuojami atliekant įvairius veiksmus su virtualia mašina kitose sistemose, taigi tai tampa naudinga informacija palaikančiajam personalui. Žemiau pateiktas metaduomenų lentelės pavyzdys:

L4 VNC meta rhel6-650-dev-10X

Instance Metadata	
build_status	: Completed
build_hostname	: LDTDSR000100114
build_phase	: 10 of 10
EMPTask	: http://empdev.barcapint.com/... TaskID=10771

77. pav. Metaduomenų langas

- **VNC** – tiesioginė jungtis per hipervizorių į virtualią mašiną. Ši funkcija reikalinga, jei virtualiai mašinai nepavyksta startuoti arba iškyla bet kokių kitų problemų ir nėra galimybės prisijungti per SSH.

Šiai informacijai gauti reikia 5 užklausų į OpenStack sistemą:

- Virtualių mašinų sąrašui gauti
- Sistemos vartotojų sąrašui gauti
- Projektų sąrašui (Tenants)
- Virtualių mašinų šablonų sąrašui gauti
- Operacinių sistemų atvaizdų sąrašui

Turint šią informaciją galima pagal vidinius OpenStack identifikacinius numerius apjungti visus sąrašus (itin praverčia Python galimybės efektyviai dirbti su sąrašais) į naują objektų sąrašą ir jį perduoti sistemos naudotojui.

4.4.18. SiteManager OpenStack sistemos vidinių projektų langas

Ši sąsaja buvo kurta analitikams, kuriuos domina atskirų projektų resursų išnaudojimas. Taip pat pasinaudojus eksportavimo į Excel funkcija – galima apjungti šią informaciją kartu su virtualių mašinų ir jų šablonų sąrašais taip suformuojant raportą apie OpenStack sistemoje esančius projektus, jiems priskirtus resursus ir išnaudotus resursus. Šie raportai bus naudojami iki tol, kol bus sukurtas ilgalaikis sprendimas matuoti debesų kompiuterijos sprendimo išnaudojamus ir turimus resursus. Projektų sąrašo sąsajos paveikslėlis žemiau:

Tenants Usage for past 24 hours

10 records per page

Search: [] Copy CSV Excel PDF Print

Resursų išnaudojimo statistika

Projekto pavadinimas

Enabled	Tenant description	Total Hours	VCPU hours	Storage usage (GB) hours	Memory usage (MB) hours
Yes	BCloudProject created on 02-Dec-2014, do not rename	207.762	383.682	11510.467	1060449.184
Yes	BCloudProject created on 03-Dec-2014, do not rename				
Yes	BCloudProject created on 04-Dec-2014, do not rename	446.271	590.152	17850.857	1306854.485
Yes	BCloudProject created on 04-Dec-2014, do not rename	100.660	172.620	4027.189	549971.339
Yes	BCloudProject created on 06-Dec-2014, do not rename	95.921	191.841	3836.822	687558.572
Yes	BCloudProject created on 06-Dec-2014, do not rename				
Yes	BCloudProject created on 06-Dec-2014, do not rename	119.901	239.801	4796.028	785781.225
Yes	BCloudProject created on 09-Dec-2014, do not rename				
Yes	BCloudProject created on 10-Dec-2014, do not rename	95.921	143.881	3836.822	294667.959
Yes	BCloudProject created on 10-Dec-2014, do not rename				

Showing 1 to 10 of 20 entries

Previous 1 2 Next

78. pav. Vidiniai projektai ir jų išnaudojami resursai

Kaip matome iš paveikslėlio aukščiau – kai kurie projektai neturi išnaudojamų resursų, iš to galima spręsti, kad jie dar nebuvo sukūrę virtualių mašinų. Taip pat galima nesunkiai palyginti skirtingus projektus pagal resursų išnaudojimą.


```

6.         'memory_total': 2164684488704,
7.         'memory_used': 51002736640,
8.         'storage_free': -126701535232,
9.         'storage_total': 236223201280,
10.        'storage_used': 362924736512,
11.        'shares_total': 50000,
12.        'shares_used': 26000,
13.        'vcpu_free': 170,
14.        'vcpu_total': 192,
15.        'vcpu_used': 22},
16.    'name': u'testenv1',
17.    'state': u'Active',
18.    'vip': u'1.1.1.1',
19.    'public_vip': u'1.1.1.1'},
20.    {'hypervisors': {'instances': 31,
21.        'memory_free': 439978295296,
22.        'memory_total': 540373155840,
23.        'memory_used': 100394860544,
24.        'storage_free': 6920266055680,
25.        'storage_total': 7874822537216,
26.        'storage_used': 954556481536,
27.        'shares_total': 50000,
28.        'shares_used': 26000,
29.        'vcpu_free': 27,
30.        'vcpu_total': 80,
31.        'vcpu_used': 53},
32.    'name': u'testenv2',
33.    'state': u'Active',
34.    'vip': u'1.1.1.2',
35.    'public_vip': u'1.1.1.2',
36.    'health': [
37.        {'status': 'DOWN', 'type': 'metering',
38.         'enabled': True}, {'status': 'UP',
39.         'type':
40.         'nova-conductor:LDTCLD010000010',
41.         'enabled': True},
42.        {'status': 'UP',
43.         'type': 'nova-scheduler:LDTCLD010000010',
44.         'enabled': True}, {'status': 'UP',
45.         'type':
46.         'nova-scheduler:LDTCLD010000013',
47.         'enabled': True},
48.        {'status': 'UP',
49.         'type': 'nova-conductor:LDTCLD010000013',
50.         'enabled': True}, {'status': 'UP',
51.         'type':
52.         'nova-conductor:LDTCLD010000014',
53.         'enabled': True},
54.        {'status': 'UP',
55.         'type': 'nova-scheduler:LDTCLD010000014',
56.         'enabled': True}, {'status': 'UP',
57.         'type':
58.         'nova-consoleauth:LDTCLD010000010',
59.         'enabled': True},
60.        {'status': 'UP',
61.         'type': 'nova-compute:LDTCLD010000015',
62.         'enabled': True}, {'status': 'UP',
63.         'type':
64.         'nova-compute:LDTCLD010000018',
65.         'enabled': False},
66.        {'status': 'UP',
67.         'type': 'nova-compute:LDTCLD010000017',
68.         'enabled': False}, {'status': 'UP',

```



```

69.         'type':
70.             'nova-consoleauth:LDTCLD010000013',
71.         'enabled': True},
72.     {'status': 'UP',
73.      'type':
74.          'nova-consoleauth:LDTCLD010000014',
75.      'enabled': True},
76.     {'status': 'UP', 'type': 'image',
77.      'enabled': True},
78.     {'status': 'UP', 'type': 'network',
79.      'enabled': True},
80.     {'status': 'UP', 'type': 'identity',
81.      'enabled': True},
82.     {'status': 'UP', 'type': 'volume',
83.      'enabled': True}}]],
84.     's_count': 21,
85.     'strg_count': 3}
86. request_ip = '192.168.0.0'

```

79. pav. Teisingų duomenų testavimui pavyzdys

Eilutėse 1 ir 86 matome, jog sukuriame du kintamieji *request_json* ir *request_ip*, kurie toliau gali būti naudojami funkcijose. Eilutėje nr. 4 formuojamas naujas sąrašas, kuriame aprašytos OpenStack aplinkos, esančios SiteManager sistemoje. CloudMap registracijos funkcija gali priimti neribotą skaičių OpenStack aplinkų kiekvienoje SiteManager sistemoje. Šiuose testiniuose duomenyse aprašytos dvi aplinkos: *testenv1* ir *testenv2*. Viena iš aplinkų neturi hipervizorių ir servisų informacijos, taigi CloudMap turi išsaugoti aplinką, tačiau apdoroti klaidas dėl trūkstamos informacijos ir vykdyti informacijos nuskaitymą toliau – registruoti antra OpenStack aplinką.

Toliau pateiktas testinių duomenų pavyzdys, kai SiteManager registruodamasis nepateikia jokių duomenų apie OpenStack aplinkas jame. Šis pavyzdys puikiai iliustruoja dažną atvejį, kai įdiegiama SiteManager sistema duomenų centre, tačiau nėra informacijos apie aplinkas, nes dar diegėjai nespėjo pradėti formuoti debesų kompiuterijos sprendimo.

```

1. request_json_wo_envs = {'comp_count': 8,
2.                        'cont_count': 7,
3.                        'dc': 'GBR',
4.                        'envs': [],
5.                        's_count': 21,
6.                        'strg_count': 3}
7. request_ip_wo_envs = '192.168.0.1'

```

80. pav. Duomenų testavimui pavyzdys, kai nėra informacijos apie OpenStack aplinkas

Iš šių, JSON formato, duomenų matome, jog *envs* sąrašas yra tuščias (`[]` – tai *list* tipo objektas, kuriame nėra elementų).

Kitas testavimo atvejis, kuris testuoja dar neįgyvendintą funkcionalumą – SiteManager vardo priskyrimas registracijos metu. Dabartinėje sistemų stadijoje SiteManager vardą galima priskirti tik rankiniu būdu, CloudMap sistemoje prisijungus į administracinę panelę (daugiau informacijos pateikiama prie CloudMap sistemos panaudos atvejų). Taigi, šiam testui skirti duomenys pateikti žemiau:

```

1. request_json_w_name = {'comp_count': 8,
2.                       'name': 'SM1',
3.                       'cont_count': 7,
4.                       'dc': 'GBR',
5.                       'envs': [],
6.                       's_count': 21,
7.                       'strg_count': 3}
8. request_ip_w_name = '192.168.0.2'

```

81. pav. Duomenų testavimui pavyzdys, kai perduodamas SiteManager sistemos pavadinimas

Eilutėje nr. 2 matome, jog naudojamas kintamasis *name*, kurį aptikusi CloudMap registracijos funkcija – priskiria šio kintamojo reikšmę SiteManager sistemai

Sekantys testavimo duomenų atvejai simuliuoja būsenas, kai perduodamoje informacijoje yra trūkstami elementai:

1. request_json_missing_inputs = {'comp_count': 8,
2. 'name': 'broken',
3. 'cont_count': 7,
4. 's_count': 21,
5. 'dc': 'ppp',
6. 'strg_count': 3}
7. request_ip_missing = '192.168.0.3'

82. pav. Duomenų testavimui pavyzdys, kai trūksta „envs“ elemento

Šis atvejis skiriasi nuo buvusio anksčiau, kai perduodamas tuščias OpenStack aplinkų sąrašas tuo, jog čia išvis nėra šio kintamojo. Programa turi nenulūžti ir veikti toliau, nepaisant nesėkmių renkant duomenis.

Kitas atvejis, kai pateikiami tik keli duomenų laukeliai:

1. request_json_no_counts = {'name': 'no_counts',
2. 'dc': 'ppp'}
3. request_ip_no_counts = '192.168.0.4'

83. pav. Duomenų testavimui pavyzdys, kai pateikiami tik du elementai

Šiuo atveju CloudMap vistiek turi užregistruoti SiteManager sistemą – turint tik IP adresą, jo pavadinimą ir duomenų centrą.

Prieš testų paleidimą taip pat yra užkraunami standartiniai pradiniai duomenys, kurie yra automatiškai įrašomi į duomenų bazę, kai sistema paleidžiama pirmą kartą. Šie duomenys pagal standartus yra aprašomi JSON formatu ir patalpinami kataloge „*fixtures*“, esančiame programos modulyje. Daugiau apie Django fixtures galima paskaityti dokumentacijoje [9]. Duomenų failo fragmentas:

1. [
2. {
3. "pk": 1,
4. "model": "register.datacenter",
5. "fields": {
6. "country": 2,
7. "code": "CRC",
8. "name": "Cranford",
9. "latitude": "40.65",
10. "longitude": "-74.29",
11. "label_position": 2
12. }
13. }
14.]

84. pav. Pradinio duomenų failo fragmentas

Iš šio fragmento matome, jog įrašant pradinis duomenis į duomenų bazę – galima nustatyti indekso numerį (trečioji eilutė), taip pat nustatyti į kurią duomenų bazės lentelę reikia įtraukti įrašą (ketvirtoji eilutė) bei lentelės laukus. Šeštoje eilutėje nustatomas išorinis raktas šalies, kurioje yra duomenų centras, tada įrašomas jo kodas, pavadinimas, koordinatės bei užrašo žemėlapyje pozicija. Paskutinis argumentas reikalingas tam, jog dubliuojantis duomenų centram (jie dažnai būna gana arti vienas kito), tampa sudėtinga juos atskirti dideliame žemėlapyje.

5.2.1.2. CloudMap automatiniai testai

Šiame skyriuje pateikti automatiniai testai ir jų aprašymai. Testuojami du sistemos moduliai:

- **register** – tai modulis, į kurį automatiškai nukreipiamos su statiniu duomenų tvarkymu susijusios užklausos (duomenų centrai, OpenStack aplinkos, SiteManager sąrašai bei jų registracijos į CloudMap sistemą).

- **common** – šis modulis skirtas registracijos logikai (realizuota registracijos grandinė, pradedant nuo SiteManager iki galutinio OpenStack aplinkos serviso), IP adreso gavimo iš užklauso, duomenų paruošimo atvaizdavimui (JSON formavimo žemėlapiui).

Pradiniai duomenys daugeliui testų yra imami iš prieš tai buvusiame skyriuje aprašytų JSON duomenų struktūrų, kadangi SiteManager registracijos metu, iš HTTP užklauso yra ištraukiamas ir atkoduojamas JSON formato duomenų objektas, kuris vėliau, naudojantis standartinėmis Python bibliotekomis perverčiamas į vidines duomenų struktūras.

Sistemos modulis „oschecks“, skirtas OpenStack testavimui nėra aprašytas testais, nes reikėtų papildomai simuliuoti OpenStack sistemos atsakymus, kas galėtų padidinti klaidų tikimybę. Be to, keičiantis OpenStack sistemos API versijoms – reiktų įdėti daug darbo, siekiant ištestuoti galimus variantus. Vietoj automatinųjų testų buvo nuspręsta naudoti itin platų informacijos kaupimo ir klaidų aptikimo mechanizmą, kurio pagalba būtų galima greitai identifikuoti, kodėl nepavyksta atlikti tam tikrų veiksmų OpenStack sistemoje.

Lentelėse pateikiami testai, jų aprašymai bei naudojami duomenys (jeigu naudojami) ir tikrinimo sąlygos:

75. lentelė. SiteManager sukūrimo su pilnais duomenimis testo aprašymas

Testo pavadinimas	test_sitemanager_creation
Aprašymas	Kviečiama pagrindinė SiteManager registravimo funkcija su pilnais duomenimis (dvi OpenStack aplinkos, hipervizorių ir fizinių serverių informacija, OpenStack servisų sąrašas).
Testo duomenys	request_json, request_ip
Tikrinamos sąlygos	Tikrinama, ar naujas SiteManager atsirado duomenų bazėje. Kadangi duomenys yra tvarkingi, turėjo susikurti sėkmingai.

76. lentelė. SiteManager kūrimo be OpenStack aplinkų testo aprašymas

Testo pavadinimas	test_sitemanager_creation_wo_ens
Aprašymas	Kviečiama pagrindinė SiteManager registravimo funkcija, tačiau vietoj OpenStack aplinkų – duodamas tuščias sąrašas. Programa neturėtų nulūžti, ši problema pirmą kartą pastebėta pradėjus testuoti sistemą, kai būdavo įdiegiama OpenStack sistema, tačiau nebuvo sukurta OpenStack aplinkų.
Testo duomenys	request_ip_wo_envs, request_ip_wo_envs
Tikrinamos sąlygos	Tikrinama, ar naujas SiteManager atsirado duomenų bazėje. Nors aplinkų informacija nebuvo perduota, tačiau pats SiteManager turėjo būti užregistruotas.

77. lentelė. SiteManager kūrimo perduodant sistemos vardą testo aprašymas

Testo pavadinimas	_test_sitemanager_creation_w_name
Aprašymas	Kviečiama pagrindinė SiteManager registravimo funkcija pradiniuose duomenyse nurodant ir SiteManager sistemos vardą. Vardas turėtų būti priskirtas šiam SiteManager.
Testo duomenys	request_json_w_name, request_ip_w_name
Tikrinamos sąlygos	Ieškomas SiteManager įrašas duomenų bazėje pagal vardą, jeigu rastas – testas sėkmingas.

78. lentelė. OpenStack aplinkos sukūrimo testo aprašymas

Testo pavadinimas	_test_environment_creation
Aprašymas	Tiesiogiai kviečiama OpenStack aplinkų kūrimo funkcija (ne per visą įvykių grandinę, kurią galite rasti prie veiklos diagramų). Imamas SiteManager objektas ir kartu perduodamas pradinių duomenų iš

	„request_json” kintamojo fragmentas „envs“, laikantis savyje tik OpenStack aplinkų sąrašą.
Testo duomenys	request_json['envs'], request_ip_w_name
Tikrinamos sąlygos	Tikrinama, ar buvo sukurtos duomenų bazėje dvi OpenStack aplinkos.

79. lentelė. OpenStack aplinkos sukūrimo perduodant nepilnus duomenis testo aprašymas

Testo pavadinimas	_test_environment_creation_wo_parameters
Aprašymas	Testas skirtas perduoti nepilnus duomenis į OpenStack aplinkos registravimo funkciją, taigi imami pradiniai duomenys iš request_json['envs'], tada pašalinami keli elementai („state“ ir „name“) ir tuomet kviečiama aplinkos registravimo funkcija.
Testo duomenys	request_json['envs']
Tikrinamos sąlygos	Tikrinama, ar pavyko registruoti OpenStack aplinkas.

80. lentelė. Duomenų kairėje esančiam menu perdavimo funkcijos testo aprašymas

Testo pavadinimas	_test_data_for_navigation_panel
Aprašymas	Šis testas skirtas funkcijai, kuri visuomet teikia duomenis kairėje esančiam menu (duomenų centrų, SiteManager sistemų ir OpenStack aplinkų sąrašas). Testas tikrina, ar po prieš tai buvusių funkcijų, kurios manipuliavo duomenų bazės duomenimis, dabar yra teisingas elementų skaičius duomenų bazės lentelėse.
Testo duomenys	Testas inicijuojamas po kitų funkcijų
Tikrinamos sąlygos	Patikrinamas SiteManager sistemų skaičius, OpenStack aplinkų skaičius, duomenų centrų skaičius

81. lentelė. Suformuoto JSON objekto, skirto į pagrindinį informacinį puslapį testo aprašymas

Testo pavadinimas	_test_get_am_world_data_statistics
Aprašymas	Šis testas tikrina, ar duomenyse yra visi elementai, kurių tikisi naudotojo naršyklėje esantis JavaScript kodas.
Testo duomenys	world_data
Tikrinamos sąlygos	Tikrinama, ar yra šie kintamųjų vardai: <ul style="list-style-type: none"> • datacenters • reports_count • success_rate • services_up • services_down

82. lentelė. Servisų kūrimo funkcijos testo aprašymas

Testo pavadinimas	_test_services_up_count
Aprašymas	Šiame teste tikrinama, ar visi servisai buvo sėkmingai įrašyti.
Testo duomenys	Testas leidžiamas po kitų testų, kurie įterpia testus.
Tikrinamos sąlygos	Tikrinamas OpenStack servisų kiekis (turi atitikti įterptų į duomenų bazę kiekį, jeigu kiekiai nesutampa – kažkurio serviso įterpimo metu kodas buvo nulūžęs).

83. lentelė. Duomenų centrų informacijos perdavimo į žemėlapių testo aprašymas

Testo pavadinimas	_test_get_am_world_data_mapping
--------------------------	---------------------------------

Aprašymas	Testas skirtas patikrinti, ar duomenų centrų informacija yra JSON objekte, kuris perduodamas į naudotojo naršyklę. Duomenys nėra tikrinami dėl jų reikšmės, tiesiog suskaičiuojami visi duomenų centrai pagal pradinis duomenis.
Testo duomenys	world_data
Tikrinamos sąlygos	Ar atitinka duomenų centrų skaičius galutiniame JSON objekte.

84. lentelė. Kelių testų grupės paleidimo kartu testo aprašymas

Testo pavadinimas	test_sitemanager_n_env_creation
Aprašymas	Šis testas naudoja kitus, anksčiau aprašytus testus. Veiksmai, atliekami šio testo metu: <ol style="list-style-type: none"> 1. Sukuria SiteManager 2. Registruoja OpenStack aplinkas sukurtame SiteManager 3. Testuoja OpenStack aplinkos registraciją su nepilnais parametrais 4. Patikrina, ar navigacijos meniu duomenys atitinka pateiktus duomenis 5. Patikrina, ar duomenys persiunčiami į žemėlapi 6. Patikrina, ar teisingas duomenų centrų skaičius perduodamas į žemėlapi (šio testo metu duomenų centrai, neturintys koordinatų – neturi būti perduodami į žemėlapi)
Testo duomenys	Kiekvienas testas turi sau priskirtus duomenis, šis testas tik kviečia anksčiau aprašytus testus
Tikrinamos sąlygos	Nėra, visi iškviešti testai turi savo tikrinimo funkcijas

85. lentelė. Nepilnų duomenų testo aprašymas

Testo pavadinimas	test_not_full_input
Aprašymas	Kviečiama SiteManager registracijos funkcija su nepilnais duomenimis. Ieškamos klaidos, kai gali nulūžti registracija paduodant nepilnus duomenis.
Testo duomenys	request_json_missing_inputs, request_ip_missing
Tikrinamos sąlygos	Tikrinama, ar SiteManager sistemoje nėra aplinkų ir ar sutampa SiteManager sistemų vardai.

86. lentelė. SiteManager registracijos, kai nepateikiami fizinių serverių kiekiai testo aprašymas

Testo pavadinimas	test_sitemanager_creation_wo_counts
Aprašymas	Šio testo metu registruojama SiteManager sistema, bet paduodami duomenys, kuriuose nėra fizinių serverių skaičiaus. Registracija turėtų būti sėkminga, nepaisant nepilnų duomenų.
Testo duomenys	request_json_no_counts, request_ip_no_counts
Tikrinamos sąlygos	Surandamas SiteManager pagal IP adresą ir patikrinama, ar serverių skaičius šiame SiteManager yra lygus nuliui.

87. lentelė. Duomenų centro patikrinimo funkcijos neperduodant kodo testo aprašymas

Testo pavadinimas	test_datacenter_creation_wo_code
Aprašymas	Testo metu kviečiama funkcija, kuri iš pradinio JSON objekto paima „dc“ reikšmę ir iš jos nustato, kokiam duomenų centrui turėtų būti priskirta SiteManager sistema. Jeigu „dc“ nėra – SiteManager turėtų būti priskirtas „LOST“ duomenų centrui. Ši situacija gali nutikti tuomet, kai SiteManager būna įdiegtas, tačiau nesukurta OpenStack aplinka, kurios kūrimo metu yra pasirenkamas duomenų centras.
Testo duomenys	request_json su pašalinta „dc“ reikšme

Tikrinamos sąlygos	Tikrinama, ar gražintas duomenų centro kodas yra “LOST”.
---------------------------	--

88. lentelė. Duomenų centro funkcijos patikrinimo perduodant kodą testo aprašymas

Testo pavadinimas	test_datacenter_creation_w_code
Aprašymas	Kviečiama funkcija, kuri iš pradinio JSON objekto paima „dc“ reikšmę ir iš jos nustato duomenų centrą. Kadangi duomenų centrai jau būna įvesti į duomenų bazę vos tik įdiegus CloudMap – dažniausiai ši funkcija turi tik identifikuoti duomenų centrą, tačiau jei kodas nežinomas – sukuriamas naujas duomenų centras.
Testo duomenys	request_json.
Tikrinamos sąlygos	Tikrinama, ar gražinto duomenų centro objekto kodas sutampa su pradiniais duomenimis.

89. lentelė. SiteManager sistemos būklės tikrinimo testo aprašymas

Testo pavadinimas	_test_get_sm_status
Aprašymas	Šis testas skirtas patikrinti, ar SiteManager būklės (būklė nusakoma skaičiais nuo 1 iki 3, kai 1 reiškia, jog viskas gerai ir SiteManager sistema siunčia užklausas į CloudMap sistemą dažniau nei kas 20 minučių) informacija apskaičiuojama korektiškai. Kadangi duomenys testo metu įterpiami greitai, dėl to būklė turėtų būti lygi vienetui. Ateityje, šis testas galėtų būti praplėstas ir tikrinti kitas būkles, manipuliuojant įrašų senumą duomenų bazėje.
Testo duomenys	Pirmas pasitaikęs SiteManager elementas duomenų bazėje.
Tikrinamos sąlygos	Tikrinama ar būklė lygi vienetui.

90. lentelė. Duomenų skaičiaus pradiniam faile testo aprašymas

Testo pavadinimas	test_datacenter_count
Aprašymas	Duomenų skaičiaus pradiniam faile testas skirtas patikrinti, ne kiek yra duomenų centrų pradiniam faile, o ar Django pavyksta juos užkrauti teisingai. Klaida galėtų įsivelti dėl blogos sintaksės JSON faile, tam užkirsti reikalingas šis testas.
Testo duomenys	initial_data.json
Tikrinamos sąlygos	Ar duomenų centrų skaičius duomenų bazėje yra lygus faile įrašytiems duomenų centrams.

91. lentelė. Pasaulio žemėlapiu duomenų formavimo testo aprašymas

Testo pavadinimas	test_world_map_data
Aprašymas	Testas kviečia duomenų formavimo pasaulio žemėlapiui funkciją. Tada, iš gauto JSON objekto paimamas duomenų centrus aprašantis sąrašas, o tada patikrinamas jo ilgis (elementų skaičius).
Testo duomenys	initial_data.json.
Tikrinamos sąlygos	Kadangi žinomas pradinis duomenų centrų skaičius – galima patikrinti, ar jie visi pateko į JSON objektą.

92. lentelė. Duomenų centro be koordinatų testo aprašymas

Testo pavadinimas	test_adding_dc_without_coords
Aprašymas	Testas sukuria duomenų centro objektą, tačiau jam nepriskiria koordinatų. Tuomet, kviečiama funkcija, kuri formuoja JSON objektą, skirtą žemėlapiui. Ši funkcija turėtų praleisti naujai sukurtą duomenų centrą, nes be koordinatų perduotas duomenų centras sukeltų problemų

	pasaulio žemėlapiu atvaizdavimui (JavaScript klaida, elementas neužsikrauna).
Testo duomenys	initial_data.json bei papildomas duomenų centras be koordinatų.
Tikrinamos sąlygos	Tikrinama, ar duomenų bazėje yra aštuoni duomenų centrai. Po to, tikrinama, ar į galutinį JSON objektą buvo perduoti septyni (atmetant duomenų centrą be koordinatų).

5.2.1.3. CloudMap automatinio testavimo rezultatai

Automatinis testavimas vykdomas RedHat operacinėje sistemoje, pasitelkiant virtualias Python aplinkas. Projekto kūrimo metu automatiškai arba rankiniu būdu leidžiant iš per IntelliJ IDE. Kartu buvo pasitelktas testų kodo padengiamumui nustatyti coverage.py paketas. Testavimo su kodo padengimu rezultatai pateikti paveikslėlyje žemiau:

```

.....
Name          Stmts  Miss  Cover   Missing
-----
common                0     0   100%
common.utils        204     40    80%   33, 81-82, 103-106, 115-121, 161-162, 187-194, 242-245, 255-271, 281, 283, 285, 301-305, 323, 339-340
-----
TOTAL                204     40    80%
-----
Ran 10 tests in 0.914s
OK
Destroying test database for alias 'default'...
Process finished with exit code 0

```

85. pav. CloudMap automatinių testų programinio kodo padengimo rezultatas

Iš aukščiau esančio paveikslėlio matome, jog net 80% programos kodo (iš „common“ modulio, kuris atsakingas už beveik visą loginį darbą) yra padengta testais. Tačiau lieka ir atvejų, kai testus parašyti yra sudėtinga, pavyzdžiui, imituoti netikrą kreipimąsi iš SiteManager sistemos ir nustatyti to kreipimosi IP adresą.

Sekančiame paveikslėlyje pateikti testo rezultatai, kuriuose matoma kiekvieno testo baigtis ir sugaištas laikas:

Test ▲	Time elapsed	Results
test_adding_dc_without_coords	27 ms	Passed
test_datacenter_count	0 s	Passed
test_datacenter_creation_w_code	3 ms	Passed
test_datacenter_creation_wo_code	3 ms	Passed
test_not_full_input	12 ms	Passed
test_sitemanager_creation	61 ms	Passed
test_sitemanager_creation_wo_counts	12 ms	Passed
test_sitemanager_creation_wo_ens	16 ms	Passed
test_sitemanager_n_env_creation	0.146 s	Passed
test_world_map_data	20 ms	Passed

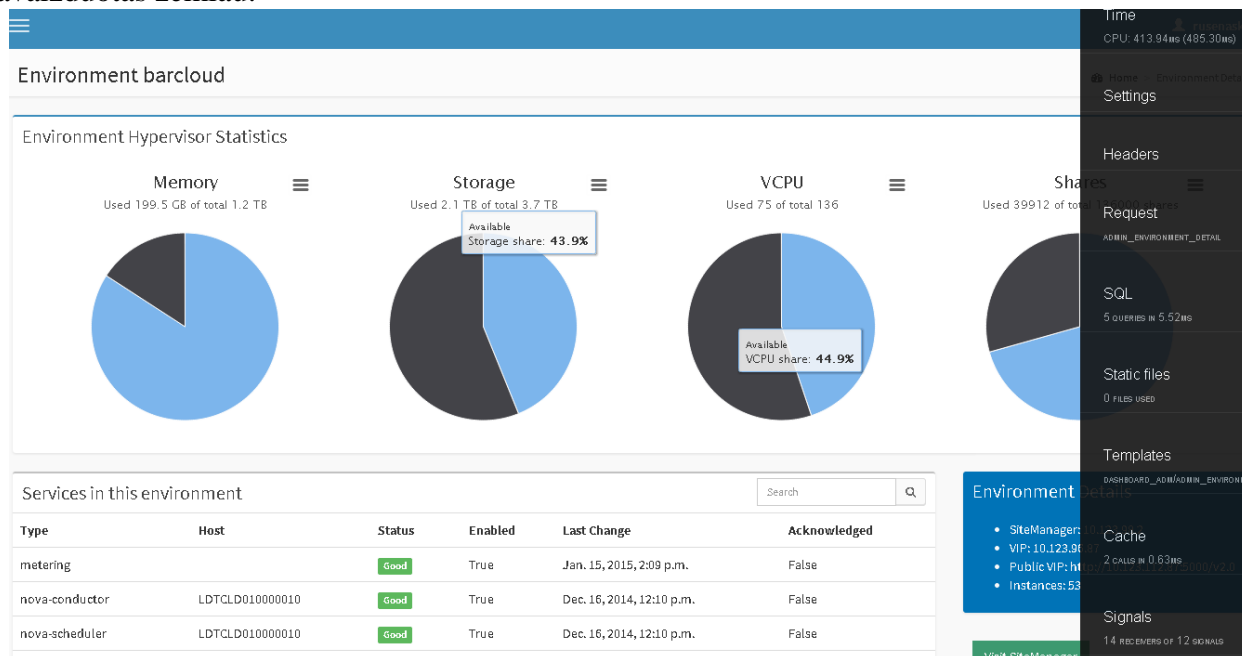
86. pav. Specifinių CloudMap automatinių testų trukmė ir rezultatai

Iš paveikslėlio matome, jog čia rodomų testų yra mažiau, nei buvo aprašyta. Taip yra dėl Django testų paleidimo mechanizmo – jis ieško tik testų, kurių funkcijų vardai prasideda „test...“, taigi visi testai, aprašyti „_test...“ – praleidžiami. Šiuos testus kviečia kiti testai. Tokios tvarkos reikia laikytis tuomet, kai norima nustatyti, kokia eilės tvarka bus paleidžiami testai, nes Django testų paleidimo metu stengiasi visus turimus testus leisti atsitiktine tvarka.

5.2.1.4. Django Debug Toolbar naudojimas testavimo metu

Tai įrankis su daug konfigūruojamų parametru, kurie gali rodyti įvairią informaciją apie kiekvieną užklausą. Taip pat, paspaudus ant parametru – parodoma jo papildoma informacija. Šiame skyriuje bus parodyti keli jo funkcionalumai, padėję aptikti spragas CloudMap sistemos programiniame kode, kurios galėtų sukelti nepatogumus (sulėtinti sistemos darbą) ateityje.

Pirmasis pavyzdys yra iš detalios OpenStack aplinkos informacijos lango. Šiame lange atvaizduojama tiek OpenStack aplinkos informacija, tiek ir jai priklausantys servisai. Šąsajos langas pavaizduotas žemiau:



87. pav. CloudMap sąsajos langas su Django Debug Toolbar

Iš aukščiau pateikto paveikslėlio matome, jog Django Debug Toolbar taip pat atvaizduoja serverio CPU skirtą laiką šiai užklausiai bei registruoja kreipimąsi į spartinančiąją atmintinę. Analizuojant kreipinius į duomenų bazę buvo nustatyta, jog pakartotiniai užklausos į duomenų bazę buvo vykdomos, kai pagal elemento (pavyzdžiui OpenStack aplinkos) išorinį raktą buvo ieškomi visi jai priklausantys servisai ir kuriamas jų sąrašas. Tai reiškia, jog jei OpenStack aplinka turėtų kelis šimtus servisų – kuriant jų sąrašą būtų kreiptasi į duomenų bazę irgi kelis šimtus kartų. Tokie atvejai gali susidaryti, kai klaidingai tikimasi iš Django karkaso perkelti informaciją į spartinančiąją atmintinę, tačiau to jis nepadaro. Pati problema, dėl servisų sąrašo kilo tuomet, kai reikėjo išskaidyti unikalų serviso identifikatorių (tipo ir serverio vardo junginį) į atskiras dalis. Kadangi, didinant serverių kiekį, skaičiavimai taps vis sudėtingesni, buvo pereita prie sudėtingesnių Django QuerySet komponento užklausų formavimo principo, kai per vieną užklausą gaunami reikalingi išoriniai raktai, pagal kuriuos šis elementas turi gauti kitą informaciją („select_related“ funkcija) bei visi elementai, kurie renkami šį elementą kaip savo išorinį raktą („prefetch_related“ funkcija). Naudojant šias dvi papildomas funkcijas buvo sumažintas užklausų skaičius tam tikruose sistemos puslapiuose nuo ~20 ir daugiau iki ~4-5 stabilus skaičiaus.

Toliau pateikta detali SQL užklausų informacija, kurią galima gauti paspaudus ant laukelio SQL:



88. pav. Detali Django Debug Toolbar SQL užklausų informacija

Iš šio paveikslėlio matome, jog šis įrankis išverčia Django duomenų bazės užklausų formavimo mechanizmo QuerySet funkcijas į SQL kalbą bei nustato laiką, kiek duomenų bazei trunka apdoroti kiekvieną kreipinį. Tokiu būdu galima greitai ir nesudėtingai identifikuoti, jei programa kreipiasi į duomenų bazę daugybę kartų (pavyzdžiui, ciklo viduje į objekto išorinius raktus kreipiamasi tiek kartų, kiek cikle yra elementų).

5.2.1.5. CloudMap ir SiteManager duomenų surinkimo ir perdavimo modulių testavimas

Šie moduliai, juos kuriant, buvo testuojami automatiniais testais ir taip pat tiesiogiai siunčiant duomenis iš SiteManager sistemos. Kadangi, dažniausiai būdavo pravartu sustabdyti CloudMap sistemos veikimą su Python Debugger (pdb moduliu) – reikėjo surasti būdą, kaip duomenis surinkti ir išsiųsti tik vieną kartą, o ne per Celery servisą, kuris juos siųstų kas kelias minutes. Šiam tikslui pasiekti buvo išnaudotas Django karkaso funkcionalumas, kuris leidžia visą sistemą paleisti per konsolę (shell). Taip pat buvo naudotas ne standartinis Python interpretatorius, o IPython modifikacija, kuri paprasčiau leidžia importuoti kitas bibliotekas, be to, IPython turi daug kitų privalumų (tarp jų galima automatiškai atnaujinti kodą, nereikia perkrauti programos). Sistemos paleidimas ir funkcijos įvykdymas parodytas paveikslėlyje žemiau:

```

-----
c-bash-4.1$ cd SM/Deploy/sitemanager/
-bash-4.1$ ls
celerybeat-schedule  cloud  fixtures  manage.pyc  pxe  static  templates
celerybeat.pid      common manage.py  puppet  sitemanager  staticdata  trigger_scripts
-bash-4.1$ python manage.py shell

Python 2.6.6 (r266:84292, Nov 21 2013, 10:50:32)
Type "copyright", "credits" or "license" for more information.

IPython 1.2.1 -- An enhanced Interactive Python.
?                -> Introduction and overview of IPython's features.
?quickref       -> Quick reference.
help            -> Python's own help system.
object?        -> Details about 'object', use 'object??' for extra details.

In [1]:

In [1]: from common.tasks.register import register_sitemanager_task

In [2]: register_sitemanager_task()

```

89. pav. SiteManager duomenų surinkimo ir persiuntimo funkcijos paleidimas per IPython konsolę

Iš šio paveikslėlio matome, jog paleidžiant sistemą per konsolę – tampa pasiekiami visi resursai, visos bibliotekos ir moduliai, kurie yra pasiekiami, kai sistema leidžiama per Apache arba per *runserver* komandą. Kadangi ši funkcija yra Celery užduotis, ji nieko negrąžina, nors jos paleidimo metu (detalų veikimą galima matyti šios funkcijos veiklos diagramoje) suveikia daug funkcijų, kurios duomenų bazėje aptinka visas OpenStack aplinkas ir sukuria jų sąrašą, tuomet iš sąrašo imamos aplinkos ir pagal jų prisijungimo duomenis į OpenStack kviečiamos atitinkamos funkcijos, kurios testuoja įvairius OpenStack komponentus.

Norint matyti rezultatus buvo modifikuotas CloudMap IP adresas SiteManager nustatymu faile ir nurodyta duomenis siųsti į 10.123.4.8:8001 IP adresu esantį serverį. Be to, CloudMap sistemoje, prieš paleidžiant registracijos funkciją, buvo įterptos duomenų išvedimo į konsolę komandos. Rezultatas pateiktas paveikslėlyje žemiau:

```

January 19, 2015 - 19:53:33
Django version 1.6.8, using settings 'smp.settings.settings'
Starting development server at http://0.0.0.0:8001/
Quit the server with CONTROL-C.
({'Sitemanager IP: ', '10.123.4.77')
({'strg_count': 2, 'comp_count': 8, 'envs': [{'name': '10.123.2.6 stable', 'state': 'Active', 'public_vip': 'http://:5000/v2.0', 'hypervisors': {'memory_used': 2147483648, 'shares_free': 4688, 'vcpu_used': 4, 'vcpu_total': 8, 'storage_free': 674309865472, 'vcpu_free': 4, 'memory_total': 28408020992, 'storage_total': 678604832768, 'shares_total': 8000, 'storage_used': 4294967296, 'shares_used': 3312, 'instances': 4, 'memory_free': 26260537344}, 'vip': '10.123.2.6', 'health': [{'status': 'UP', 'type': 'nova-consoleauth:ldtcsr000000aio13', 'enabled': True}, {'status': 'UP', 'type': 'nova-scheduler:ldtcsr000000aio13', 'enabled': True}, {'status': 'UP', 'type': 'nova-conductor:ldtcsr000000aio13', 'enabled': True}, {'status': 'UP', 'type': 'nova-compute:ldtcsr000000aio13', 'enabled': True}, {'status': 'UP', 'type': 'nova-compute:ldtcsr0000001450.etf.barcapetf.com', 'enabled': True}, {'status': 'UNKNOWN', 'type': 'cloudformation', 'enabled': True}, {'status': 'UNKNOWN', 'type': 'orchestration', 'enabled': True}, {'status': 'UP', 'type': 'image', 'enabled': True}, {'status': 'UP', 'type': 'volume', 'enabled': True}, {'status': 'UP', 'type': 'metering', 'enabled': True}, {'status': 'UP', 'type': 'network', 'enabled': True}, {'status': 'UP', 'type': 'identity', 'enabled': True}], {'name': 'barcloud_stable', 'state': 'Active', 'public_vip': 'http://10.123.112.87:5000/v2.0', 'hypervisors': {'memory_used': 209916526592, 'shares_free': 95960, 'vcpu_used': 71, 'vcpu_total': 136, 'storage_free': 1865089548288, 'vcpu_free': 65, 'memory_total': 1352528822272, 'storage_total': 4055522869248, 'shares_total': 139043320960, 'instances': 51, 'memory_free': 1142329687}, 'health': [{'status': 'UP', 'type': 'metering', 'enabled': True}, {'status': 'UP', 'type': 'nova-conductor:LDTCCLD010000010', 'enabled': True}, {'status': 'UP', 'type': 'nova-scheduler:LDTCCLD010000010', 'enabled': True}, {'status': 'UP', 'type': 'nova-conductor:LDTCCLD010000013', 'enabled': True}, {'status': 'UP', 'type': 'nova-conductor:LDTCCLD010000014', 'enabled': True}, {'status': 'UP', 'type': 'nova-scheduler:LDTCCLD010000014', 'enabled': True}, {'status': 'UP', 'type': 'nova-consoleauth:LDTCCLD010000010', 'enabled': True}, {'status': 'UP', 'type': 'nova-compute:LDTCCLD010000015', 'enabled': True}, {'status': 'UP', 'type': 'nova-compute:LDTCCLD010000018', 'enabled': True}, {'status': 'UP', 'type': 'nova-compute:LDTCCLD010000017', 'enabled': True}, {'status': 'DOWN', 'type': 'nova-consoleauth:LDTCCLD010000013', 'enabled': True}, {'status': 'DOWN', 'type': 'nova-consoleauth:LDTCCLD010000014', 'enabled': True}, {'status': 'UP', 'type': 'image', 'enabled': True}, {'status': 'UP', 'type': 'network', 'enabled': True}, {'status': 'UP', 'type': 'identity', 'enabled': True}, {'status': 'UP', 'type': 'volume', 'enabled': True}], 'dc': 'ldt', 'cont_count': 8, 's_count': 22}

```

SiteManager IP adresas

OpenStack aplinkos duomenys

90. pav. CloudMap gauti duomenys iš SiteManager sistemos

Iš šių gautų duomenų apie OpenStack aplinką matome, jog žmogui gana sudėtinga perskaityti šiuos duomenis (nors raktus ir reikšmes rasti gana lengva, tačiau suvokti, kur yra sąrašai, o kur žodynai – sudėtinga). Šią problemą galima išspręsti pasitelkiant *pprint* modulį, kuris šifruojasi kaip „pretty-print“, ir yra skirtas formatuoti duomenų struktūras. Panaudojus šį modulį, gaunamos informacijos fragmentas atrodo taip:

```

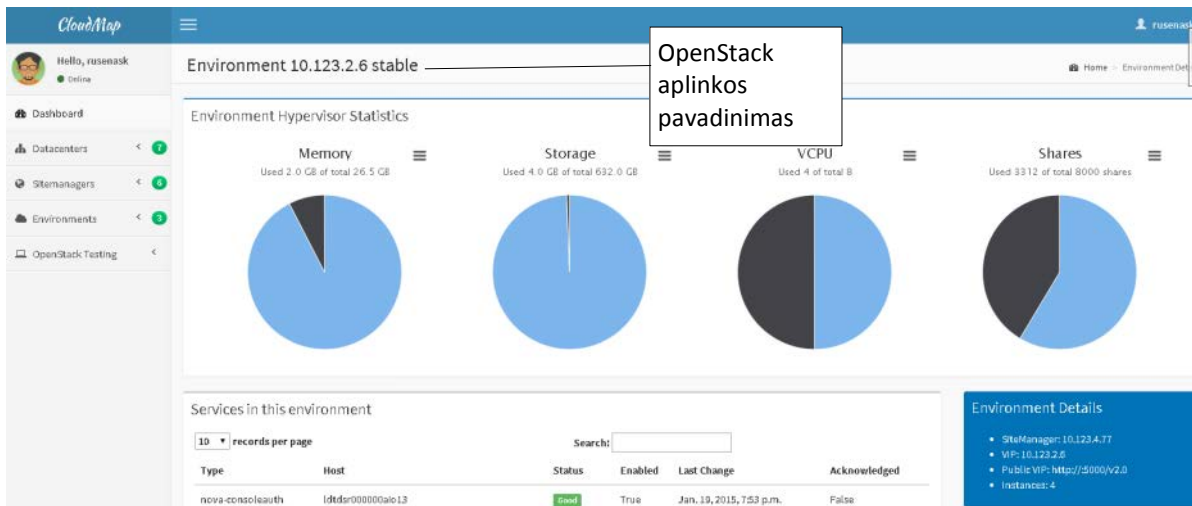
({'Sitemanager IP: ', '10.123.4.77')
{'comp_count': 8,
 'cont_count': 8,
 'dc': 'ldt',
 'envs': [{'health': [{'enabled': True,
                       'status': 'UP',
                       'type': 'nova-consoleauth:ldtcsr000000aio13'},
                    {'enabled': True,
                       'status': 'UP',
                       'type': 'nova-scheduler:ldtcsr000000aio13'},
                    {'enabled': True,
                       'status': 'UP',
                       'type': 'nova-conductor:ldtcsr000000aio13'},
                    {'enabled': True,
                       'status': 'UP',
                       'type': 'nova-compute:ldtcsr000000aio13'},
                    {'enabled': True,
                       'status': 'UP',
                       'type': 'nova-compute:ldtcsr0000001450.etf.barcapetf.com'},
                    {'status': 'UNKNOWN',
                       'type': 'cloudformation'},
                    {'enabled': True,
                       'status': 'UNKNOWN',
                       'type': 'orchestration'},
                    ...

```

OpenStack aplinkų sąrašas prasideda čia

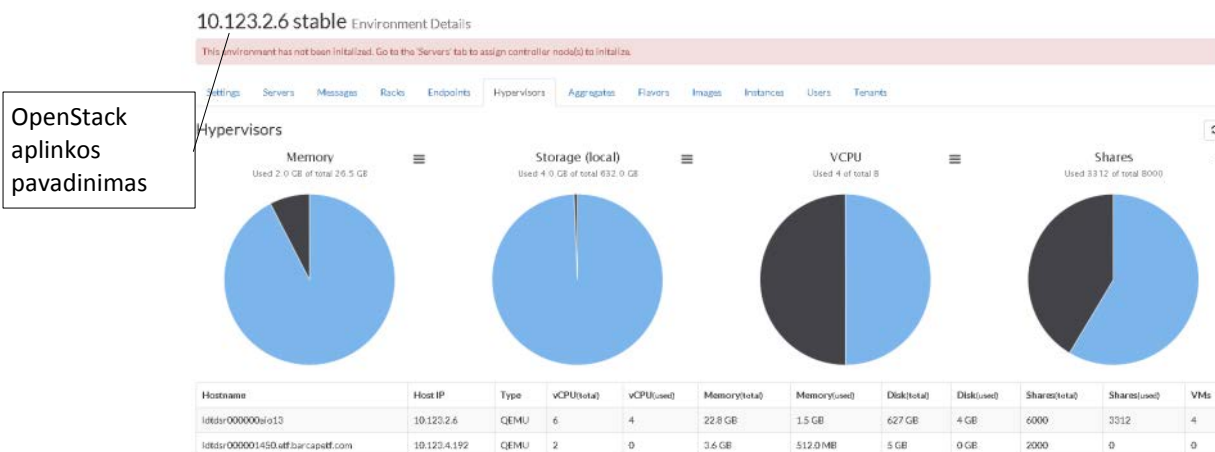
91. pav. Struktūrizuotų duomenų fragmentas iš SiteManager registracijos

Kadangi, registracija buvo sėkminga (CloudMap registracijos funkcija grąžino 200 kodą) – OpenStack aplinkos duomenys buvo įrašyti sistemoje. Atsidarius naršyklėje CloudMap ir įsijungus naujais sukurtos aplinkos duomenis matome, jog hipervizorių informacija sutampa (paveikslėlis žemiau):



92. pav. CloudMap OpenStack aplinkos langas po registracijos

Šiame lange matome, jog yra išnaudojami 2GB RAM, 4GB vietos diske, 4VCPU bei 3312 Shares. Tuomet, naršyklėje įvedame serverio adresą 10.123.4.77, atsidarome OpenStack aplinkos informacinį langą ir atkreipiame dėmesį į hipervizorių informaciją, kuri pateikta žemiau:



93. pav. SiteManager OpenStack aplinkos hipervizorių informacija

SiteManager ir CloudMap sistemų turima informacija apie OpenStack sutampa – registracija sėkminga. Turėti veikiantį duomenų šaltinį buvo pagrindinė ir svarbiausia pirminė užduotis, siekiant sukurti stabilią sistemą, nes šie duomenys puikiai pasitarnauja automatiniams testams.

5.2.2. Sistemoje vykstančių procesų įrašymas

Vien rezultatų pateikimas vartotojui nėra viskas, ką norėtų matyti sistemą administruojantis personalas. Norint stebėti sistemoje vykstančius procesus testavimo metu arba kai sistema yra tik išleidžiama į produkciją ir yra poreikis stebėti jos veikimą – renkami papildomi įrašai, kokios funkcijos ir kada paleidžiamos, kada užbaigiamos ir įrašomi visi veiksmai, kurie galėtų dominti sistemos kūrėjus arba administratorius. Taip pat šie duomenys naudingi sistemą palaikančiam personalui, jei kyla nenumatytų problemų. Papildomus duomenų rinkimas nėra rekomenduojamas būdas produkcijoje veikiančiai sistemai, nes tai gali sugeneruoti didelius duomenų kiekius, kurių nėra naudingi, jeigu sistema veikia be problemų.

Sistemoje vykstančių procesų įrašymą galima išjungti arba įjungti nustatant „DEBUG“ kintamojo reikšmę (True arba False) /settings/base.py faile. Dėl to, dirbant prie sistemos kodo ir testuojant ją – galima visuomet laikyti šią funkciją įjungtą. Žemiau pateiktas kodo pavyzdys, skirtas įrašinėti įvykiams sistemoje:

```

LOGGING = {
    'version': 1,
    'disable_existing_loggers': False,
    'formatters': {
        'verbose': {
            'format':
                """%(levelname)s %(asctime)s %(methodName)s %(process)d %(thread)d %(message)s"""
        },
        'simple': {
            'format': '%(levelname)s %(message)s'
        },
    },
    'filters': {
        'require_debug_false': {
            '()': 'django.utils.log.RequireDebugFalse'
        },
    },
    'handlers': {
        'mail_admins': {
            'level': 'ERROR',
            'filters': ['require_debug_false'],
            'class': 'django.utils.log.AdminEmailHandler'
        },
        'file': {
            'level': 'DEBUG',
            'class': 'logging.handlers.TimedRotatingFileHandler',
            'formatter': 'verbose',
            'filename': os.path.join('/var/log/cloudmap/', 'debug.log'),
        },
    },
    'loggers': {
        'django.request': {
            'handlers': ['mail_admins'],
            'level': 'ERROR',
            'propagate': True,
        },
        'cloudmap.register': {
            'handlers': ['file'],
            'level': 'DEBUG'
        },
        'cloudmap.oschecks': {
            'handlers': ['file'],
            'level': 'INFO'
        }
    }
}

```

formatters – tai funkcijos, kurios paverčia įrašus į tekstą

filters – papildomas įrankis įrašų filtravimui, prieš jiem patenkant į *handlers*

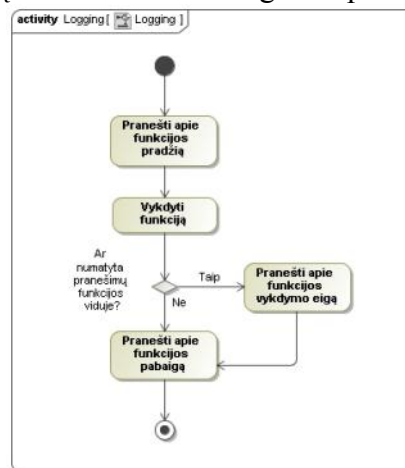
handlers – tai varikliai, nustatantys, kaip elgtis su ateinančiais įrašais

loggers – aprašomi duomenų rinkimo taškai, per kuriuos galima siųsti duomenis

94. pav. Sistemoje vykstančių procesų įrašymo konfigūracija

Iš šios konfigūracijos matome, jog išjungus klaidų ieškojimo režimą ir gavus klaidą, kurios numeris 500 – išsiunčiamas elektroninis laiškas sistemos administratoriams, kurie taip pat aprašomi nustatymų faile. Taip pat matome, į kurį failą įrašinėjami sistemoje vykstantys procesai.

Tačiau aukščiau pateikta yra tik žinučių apdorojimo konfigūracija, o pačios žinutės generuojamos kode kiekvienoje funkcijoje. Norint stebėti naujų SiteManager sistemų registraciją, kuri vykdoma inicijuojant kelias, viena paskui kitą, veikiančias funkcijas – jos visos praneša apie pradžią ir pabaigą bei įvykusius veiksmus jose (pavyzdžiui užregistruojama nauja SiteManager sistema arba OpenStack aplinka). Įrašų kūrimo veiklos diagrama pateikta paveikslėlyje žemiau:



95. pav. Pranešimų apie sistemos procesus veiklos diagrama

Iš šios veiklos diagramos matome, jog taip įrašant funkcijų veiksmus – sistemos įrašuose bus matomos, kurios funkcijas ir kokia tvarka yra paleidžiamos ir vykdomos. Žemiau pateiktas paveikslėlis su tokių įrašų failo fragmentu, kai vykdoma SiteManager sistemos registracija į CloudMap:

```
DEBUG 2015-01-22 12:26:42,272 common.utils.register_creator 3210.140030923859712 Entering
DEBUG 2015-01-22 12:26:42,273 common.utils.check_datacenter 3210.140030923859712 Entering
DEBUG 2015-01-22 12:26:42,413 common.utils.check_datacenter 3210.140030923859712 Leaving
DEBUG 2015-01-22 12:26:42,580 common.utils.environment_creator 3210.140030923859712 Entering
DEBUG 2015-01-22 12:26:42,593 common.utils.environment_creator 3210.140030923859712 Environme
DEBUG 2015-01-22 12:26:42,684 common.utils.service_creation 3210.140030923859712 Entering
DEBUG 2015-01-22 12:26:42,782 common.utils.service_creation 3210.140030923859712 Leaving
DEBUG 2015-01-22 12:26:42,789 common.utils.environment_creator 3210.140030923859712 Environme
DEBUG 2015-01-22 12:26:42,815 common.utils.service_creation 3210.140030923859712 Entering
DEBUG 2015-01-22 12:26:42,943 common.utils.service_creation 3210.140030923859712 Leaving
DEBUG 2015-01-22 12:26:42,943 common.utils.environment_creator 3210.140030923859712 Leaving
```

96. pav. Sistemos įrašai vykdant SiteManager registraciją į CloudMap

Šie įrašai tampa labai vertingi, kai reikia identifikuoti problemas testavimo metu. Net jeigu funkcijos veikia be klaidų – dar nereiškia, kad jos visos buvo iškviestos. Iš pateikto paveikslėlio matome, jog tampa nesudėtinga stebėti visą veikimo procesą, kai per vieną užklausą kreipiamasi į kelias funkcijas.

5.2.3. Naudotojų atliktas visų sistemų integracijos testas

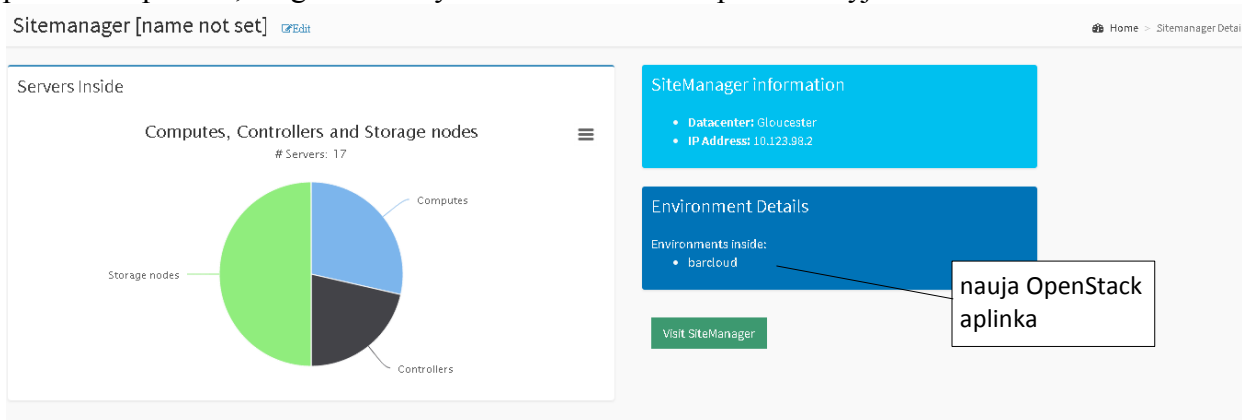
Sistemų integracijos testas, kitaip dar žinomas SIT vardu, buvo atliekamas tris mėnesius. Jo metu, buvo integruojamos visų komandų sukurtos sistemos bei sprendimai. SiteManager šio testo metu turėjo didelę svarbą, nes tai buvo pagrindinis įrankis, kurio pagalba buvo kuriama OpenStack aplinka bei pagrindinis įrankis, norint stebėti kuriamas virtualias mašinas ir jų status OpenStack sistemoje.

Nepaisant to, kad per SiteManager buvo sėkmingai įdiegta OpenStack aplinka – vis dėl to kilo tam tikrų problemų, kai OpenStack palaikymo komanda turėjo spręsti iškilusias bėdas su naudotojų virtualiomis mašinomis. Pagrindinės problemos:

- Trūko papildomų duomenų laukų virtualių mašinų sąrašė, tokių kaip virtualios mašinos identifikacinis numeris, metaduomenys.
- Operacinių sistemų atvaizdų identifikaciniai numeriai, formatai
- Projektų identifikaciniai numeriai, aprašymai bei resursų išnaudojimas

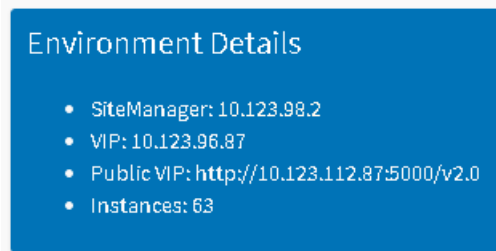
Nors kai kurios problemos atrodė kaip naujų funkcijų įvedimas (tam tikrais atvejais taip ir buvo), tačiau buvo nuspręsta šiuos pakeitimus įdiegti į dabartinę SiteManager versiją.

CloudMap testavimui buvo modifikuotas SiteManager programinis kodas ir nurodyta informaciją siųsti į 10.123.4.8 IP adresą, kuris buvo laboratorijoje esančiame serveryje. Įdiegus SiteManager – jis sėkmingai pradėjo siųsti duomenis apie savo fizinius serverius ir jame esančias OpenStack aplinkas, tai galima matyti žemiau esančiame paveikslėlyje:



97. pav. Sistemų integracijos testo metu sukurtas SiteManager užsiregistruoja CloudMap sistemoje

Šis testas buvo ypač svarbus OpenStack palaikymo komandai – prieš išleidimą į produkciją, ji turėjo reikėti išbandyti savo galimybes problemų aptikime ir sprendime, taigi buvo sukurta daug virtualių mašinų su įvairiausiais operacinių sistemų atvaizdais. Detali aplinkos informacija:



98. pav. Detali OpenStack aplinkos informacija

Viena iš indikacijų, kad viskas gerai – virtualių mašinų kiekis. Iš jo galime spręsti, kad OpenStack komponentai tarpusavyje komunikuoja sėkmingai. Nors ir negalime iš CloudMap sistemos tiesiogiai stebėti virtualių mašinų sąrašo ir jų kūrimo fazių, tačiau pagal hipervizorių informaciją galima spręsti apie pačios aplinkos apkrovimą. Taip pat, yra galimybė prisijungt į betkurią SiteManager sistemą ir iš ten stebėti kompiuterių debesies parametrus, nebent naudotojas nėra įdėtas į reikalingas Active Directory grupes – tuomet jam gali būti prieinamos tik kai kurios sistemos arba jų fragmentai.

5.2.4. Testavimo rezultatai

Sistemų kūrimo metu yra sudėtinga apsibrėžti, kokie yra testavimo rezultatai. Įvertinant tai, jog abiejų sistemų kūrimo metu jos abi buvo nuolat testuojamos (SiteManager buvo testuojama diegėjų, o CloudMap nuo pat pradžių turėjo automatinius testus), galima teigti, kad sistemų kokybė yra gera, nes jos tenkina iškeltas sąlygas:

- SiteManager serverių valdymo per IPMI funkcijos veikia be sutrikimų (jeigu būtų problemų – diegėjams nepavyktų įrašyti OpenStack)
- SiteManager atvaizduoja OpenStack aplinkos vidinių resursų išnaudojimą – taip pat ištestuota tiek diegėjų komandos, tiek palaikymo komandos.
- SiteManager sėkmingai perduoda duomenis į CloudMap – laboratorijoje sėkmingai užfiksuojami visi SiteManager iškart, po jų įdiegimo.
- CloudMap teisingai registruoja gautus duomenis iš SiteManager – tai patvirtina automatiniai testai bei realūs testai, kurie buvo atliekami laboratorijoje.

Galutinio testavimo (sistemų integracijos testo) metu nebuvo nustatyta jokių sistemos veikimo sutrikimų – tiek SiteManager, tiek CloudMap sistemos veikė taip, kaip ir buvo laukta. Vieninteliai nusiskundimai buvo dėl trūkstamos informacijos apie virtualias mašinas (identifikacinių numerių, metaduomenų bei kitų palaikymo komandai reikalingų parametrų), tačiau šios problemos buvo traktuojamos ne kaip defektai, o kaip naujų funkcijų įdiegimo prašymas.

Taip pat testavimo metu tapo aišku, jog visų kuriamos sistemos funkcijų praktiškai neįmanoma aprašyti automatiniais testais dėl šių priežasčių:

- Automatinis testas rašyti funkcijoms, kurios komunikuoja su fizinių serverių IPMI sąsajom, yra labai sudėtinga, nes reikia imituoti išorinės bibliotekos veiklą, kuri turi kintančias versijas bei įranga taip pat keičiasi (sistemos kūrimo pradžioje HP ILO versija buvo 2.0, o pradėjus diegti sistemas į produkcinę aplinką – HP ILO buvo su 4.03 versija ir turėjo pasikeitusią duomenų hierarchiją).
- Automatinis testas, kurie skirti komunikuojančioms su OpenStack aplinka funkcijoms, rašyti yra sudėtinga ir nepraktiška dėl to, jog tos funkcijos dažniausiai atvaizduoja tik informaciją, gautą per RestAPI iš OpenStack, taigi kilusios bėdos bus dažniausiai iš tos sistemos, o ne pačiame SiteManager.

Kita problema, kuri buvo pastebėta testavimo metu – sistemos naudotojai, atliekantys svarbią rolę testuodami sistemą (SiteManager), dažnai pasirinkdavo sau mėgstamą metodą serverių paieškai, o tada tik jį ir naudodavo. Taip buvo susidariusi problema su importavimo iš CSV funkcija,

kai po serverių lentelės duomenų bazėje pakeitimo ji nustojo veikti ir tai nebuvo pastebėta ilgą laiką, nes naudotojai rinkosi automatinę serverių paiešką tinkle.

6. EKSPERIMENTINIS CLOUDMAP IR SITEMANAGER SISTEMOS TYRIMAS

Norint išsiaiškinti sistemos realias galimybes ir iššūkius, kurie laukia tolimesnėse kūrimo ir tobulinimo fazėse – yra būtina patalpinti taikomąją programą į produkcinę aplinką, pilnai apsaugotus (ir apribotus) serverius ir palikti sistemą naudotojams tokią, kokią ji yra, daugiau prie jos nedirbant ir tiesiog laukiant atsiliepimų bei fiksuojant jos trūkumus.

6.1. Eksperimento planas

Šiam eksperimentui įgyvendinti viena iš būtinų sąlygų yra turėti serverį produkcinėje aplinkoje. Tada, kita sąlyga buvo gauti praregistruotą serverio CNAME – nuorodą į fizinį serverį, per kurią būtų galima jį pasiekti iš bet kurio įmonės tinkle esančio duomenų centro. Po pakeitimų įmonės DNS serveryje, CloudMap sistema (vidiniame tinkle) buvo pasiekama šiais adresais:

- cloudmap.intranet.barcapint.com
- cloudmap

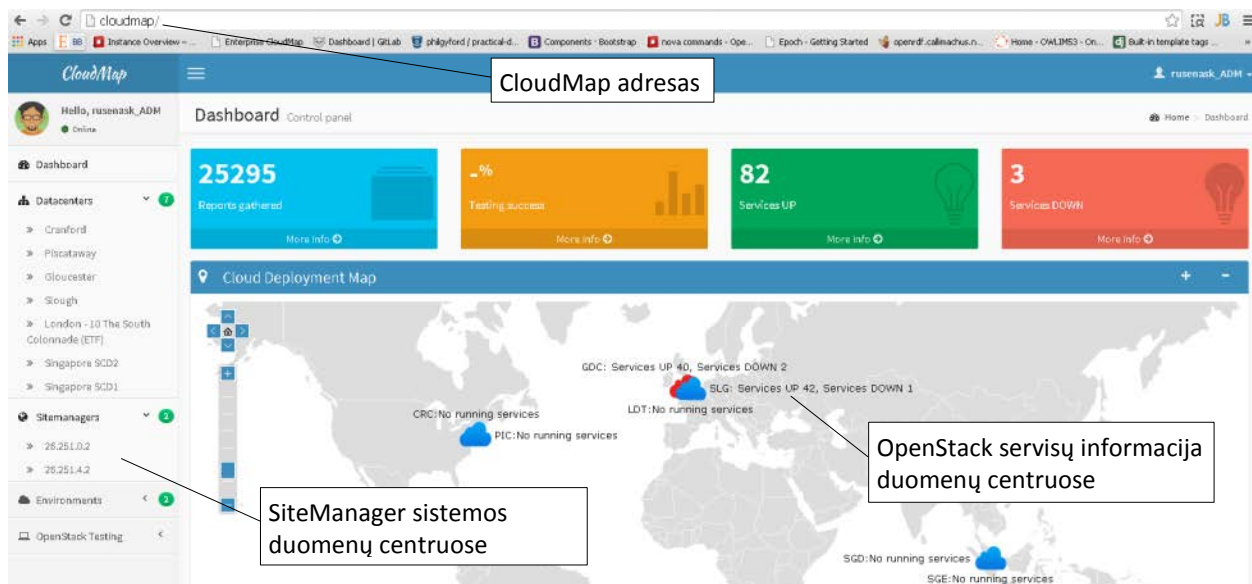
Taigi, visų SiteManager sistemų programiniame kode nurodžius siųsti duomenis į šiuos adresus – galima užtikrinti, kad duomenys CloudMap sistemą pasieks automatiškai.

Sekantis žingsnis buvo laukti, kol produkcinį sprendimą palaikantis personalas pradės diegti SiteManager sistemas ir kurti OpenStack aplinkas.

6.2. Eksperimento rezultatai

Eksperimentui prasidėjus paaiškėjo, jog nebetinka Active Directory grupės (nebegalima prisijungti tiek į SiteManager, tiek į CloudMap sistemas), kurios buvo naudojamos anksčiau. Pridėjus naujas grupes, kurias galima naudoti produkciniame tinkle – buvo galima prisijungti prie CloudMap sistemos.

Pradžioje, SiteManager sistemos buvo įdiegtos Slough ir Gloucester duomenų centruose. Paveikslėlis su gauta informacija žemiau:

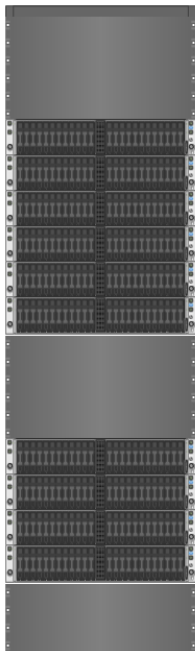


99. pav. Produkcinės CloudMap realizacijos programos langas


























































Kaip matome iš aukščiau pateikto paveikslėlio - per CloudMap sistemą buvo galima greitai identifikuoti SiteManager sistemų adresus, taigi buvo galima į juos prisijungti.

Toliau pateikiame produkcinės SiteManager sistemos vaizdus. Jie pateikti tik iš vienos sistemos, nes matome, jog sėkmingai veikia abiejuose duomenų centruose. Sekančiame paveikslėlyje parodytas serverių spintos vaizdas:

Rack ID - CN8352P1F4, 42U



Servers in rack

#	Server	Model	Height	Position	Bay	Role	Env.	Action
1	slgclid010000018	ProLiant SL210t Gen8	2 U	11	1	compute	slgclid01	   
2	slgclid010000038	ProLiant SL210t Gen8	2 U	11	3	compute	slgclid01	   
3	slgclid010000020	ProLiant SL210t Gen8	2 U	13	1	compute	slgclid01	   
4	slgclid010000021	ProLiant SL210t Gen8	2 U	13	3	compute	slgclid01	   
5	slgclid010000022	ProLiant SL210t Gen8	2 U	15	1	compute	slgclid01	   
6	slgclid010000023	ProLiant SL210t Gen8	2 U	15	3	compute	slgclid01	   
7	slgclid010000024	ProLiant SL210t Gen8	2 U	17	1	compute	slgclid01	   
8	slgclid010000025	ProLiant SL210t Gen8	2 U	17	3	compute	slgclid01	   
9	slgclid010000026	ProLiant SL210t Gen8	2 U	25	1	compute	slgclid01	   
10	slgclid010000027	ProLiant SL210t Gen8	2 U	25	3	compute	slgclid01	   
11	slgclid010000028	ProLiant SL210t Gen8	2 U	27	1	compute	slgclid01	   
12	slgclid010000029	ProLiant SL210t Gen8	2 U	27	3	compute	slgclid01	   
13	slgclid010000030	ProLiant SL210t Gen8	2 U	29	1	compute	slgclid01	   
14	slgclid010000031	ProLiant SL210t Gen8	2 U	29	3	compute	slgclid01	   
15	slgclid010000032	ProLiant SL210t Gen8	2 U	31	1	compute	slgclid01	   

100. pav. SiteManager serverių spintos vaizdas

Iš serverių spintos lango matome, jog informacija nuskaityta sėkmingai:

- Serverių modeliai gauti per IPMI
- Serverių aukščiai, pozicijos spintoje gautos per IPMI
- Serverio rolė sėkmingai nuskaityta
- Sėkmingai priskirta OpenStack aplinka serveriui

Pasirinkus vieną iš serverių – testavimui buvo prisijungta į jo LOM sąsajos langą, jį pavaizduota paveikslėlyje žemiau:

details

[Properties](#)
[History](#)
[LOM](#)
[Storage](#)

slgclid010000018 LOM Details

Host Data

Server role:	compute
Model:	ProLiant SL210t Gen8
CPU count:	40
CPU type:	64-bit Capable
CPU speed:	2.8Ghz
RAM:	256 GB
Node type:	Compute node

♥ Check health
🕒 Get more details

⚙️ Boot controls

Location Services

Rack description	HP 42U Intelligent Series Rack
Rack ID	CN8352P1F4
Rack height	42 U
Server position in rack	11 U
Server height	2 U
Server bay	1
HP iLO interface	10.103.156.40
UID status: OFF	★ Turn ON
Force power cycle:	🔄 Restart Server

Updated: Jan. 19, 2015, 9:35 p.m.
 🔄 Refresh

101. pav. SiteManager detalios serverio informacijos langas

Iš šio lango matome, jog buvo sėkmingai paimta visa informacija per IPMI sąsają:

- Sėkmingai nuskaityta serverio rolė

- Sėkmingai nuskaityta serverio vietos informacija (serverių spintoje), taigi buvo automatiškai sukurti serverių spintų objektai duomenų bazėje.
- Taip pat buvo įrašyti serverio parametrai į SiteManager duomenų bazę apie jo RAM, vCPU, CPU tipą, dažnį.

6.2.1. Išvados

Iš aukščiau pateiktų rezultatų matome, jog tikslas (kam buvo kurtos šios sistemos) pasiektas:

- **Diegimas įvykdytas automatiškai**, perdavus tik pagrindinius parametrus - diegimo metu nebuvo reikalingos gilios OpenStack žinios.
- **SiteManager sistemos automatiškai prisiregistravo į CloudMap sistemą**, su jom priklausančioms OpenStack aplinkomis.
- **Buvo suteikta prieiga prie OpenStack resursų stebėjimo** – buvo realiu laiku renkama informacija per OpenStack API apie aplinkos resursų išnaudojimą, virtualias mašinas (jų būsenas), virtualių mašinų šablonus, operacinių sistemų atvaizdus ir t.t.

Sukurta sistema, palyginus su panašiomis sistemomis, veikia geriau dėl šių pagrindinių funkcijų:

- **Gali diegti nestandartinius OpenStack komponentus** (modifikuotas Keystone, Neutron pakeistas į Nuage).
- Diegimo metu taip įdedamas papildomas komponentas pakeičiantis virtualių mašinų talpinimo serveriuose logiką.
- **SiteManager sistemoje automatiškai prisiregistravo visi serveriai**, jiems tuomet buvo per IPMI sąsają nustatyta krautis iš tinklo ir perduoti operacinių sistemų atvaizdai – panašios sistemos neturi tokio lygio automatizavimo.
- **OpenStack aplinkų informacija gaunama per Rest API**, taip pat įvesta veiksmų autorizacija (pavyzdžiui VM perkrovimo) pagal vartotojų grupes, o tokio funkcionalumo Horizon komponentas neturi.
- **CloudMap galėjo aktyviai automatiškai testuoti OpenStack aplinkas** – tokių galimybių analogiškos sistemos neturi, nes šis funkcionalumas nereikalauja sistemos naudotojui turėti privilegijų pačioje OpenStack aplinkoje. Jam užtenka turėti teises prisijungti į CloudMap, taigi jis negali padaryti jokios žalos paleisdamas testą.

Taigi, diegimo procesas buvo kokybiškai patobulintas – diegėjams nebereikėjo jungtis į dešimtis ar šimtus virtualių mašinų ir diegti bei konfigūruoti serverių rankiniu būdu. Reikalingas laikas, norint įdiegti vieną OpenStack aplinką (sudarytą iš 3 valdymo serverių) buvo sutrumpintas nuo **2-3 dienų iki 2-4 valandų**. Laiką, norint įdiegti OpenStack aplinką rankiniu būdu iš kelių šimtų serverių net neįmanoma nustatyti (tokį eksperimentą atlikti būtų sudėtinga ir beprasmiška), taigi galima įvertinti tik faktą, jog pridėdant papildomus skaičiavimo serverius – diegimo laikas ilgėja ne tiesiškai, nes su kiekvienu serveriu – sudėtingėja konfigūracija.

OpenStack aplinkos resursų priežiūros proceso patobulinimą galima įvertinti tuo, jog anksčiau nebuvo galimybės stebėti iškart daugelio OpenStack aplinkų. CloudMap sistema suteikia šį funkcionalumą, taip pat sistemos priežiūros komandai supaprastina problemų paiešką, nes pateikia informaciją apie visus veikiančius ir neveikiančius vidinius servisus.

6.3. Sprendimo veikimo ir savybių analizė, kokybės kriterijų įvertinimas

Šiame skyriuje kategorijomis suskirstomos savybės, jos analizuojamos. Taip pat įvertinami kokybės kriterijai įvairiais aspektais (tiek sistemos kūrimo kokybė, tiek ir naudojimas)

6.3.1. Sprendimo veikimo analizė

Atliekant eksperimentą buvo matoma, jog kaip ir tikėtasi – dėl padidėjusio apkrovimo (sistemomis vienu metu naudojosi kelios dešimtys darbuotojų) nesusidarė jokių problemų. Produkcijai skirta Apache konfigūracija ir tai, jog SiteManager sistema buvo įdiegta į vieną iš valdymo serverių, kurie dažniausiai turi po 48 vCPU ir kelis šimtus MB RAM atminties, galėtų

atlaikyti tūkstančius kartų didesnę krūvį. SiteManager sistemos taip pat sėkmingai registravosi į CloudMap sistemą.

Palyginus su anksčiau įmonėje naudota sistema „QuataViewer“ – SiteManager veikė daug greičiau, nes buvo nesikreipiama tiesiai į duomenų bazę (tai nėra patikimas būdas pateikti informaciją, nes OpenStack duomenų bazė nėra sukurta efektyviai atvaizduoti duomenis tiesiai iš lentelių, papildomai jų neapdorojant). Taip pat, naudojant *Django Cache Framework* buvo galima išsaugoti informaciją serverio sparčiojoje atmintyje, taip ženkliai sutrumpinant užklauskos laiką (nuo ~500ms užklauskos atsakymo laikas sumažėjo iki ~50ms).

6.3.2. Sprendimo savybių analizė

Naudotojams abi sistemos pasirodė intuityvios, lengvai suprantamos, nes nebuvo suteiktas joks papildomas apmokymas, o tiesiog duotos teisės prisijungti. Kita svarbi savybė – sistemų dizainas, Bootstrap CSS karkasas leido nesudėtingai sukurti naudotojui patrauklų dizainą, aiškiai atrodančias lenteles, mygtukus.

6.3.3. Sprendimo kokybės kriterijų įvertinimas

Pagrindiniai kokybės kriterijai yra nustatomi iš funkcinių reikalavimų. Kadangi šis sprendimas yra kuriamas inžinieriam, kurie vykdys konkrečius veiksmus, diegiant OpenStack sistemą – nesukurtos funkcijos galėtų sukelti problemų visai instaliacijos grandinei. Kokybės kriterijai šiuo atveju gali būti skirstomos į šias grupes:

Sistemos kūrimo kokybė – buvo naudojami vienetų testai, integraciniai testai, nenutrūkstantis RPM paketų kūrimas ir testavimas su Jenkins platforma. Taip pat didelę reikšmę sistemos kokybei turėjo nuolatinis naudojimas ja, kuriant diegimo skriptus.

Sistemos funkcijų kokybė – testavimo metu buvo sėkmingai įdiegta OpenStack aplinka duomenų centruose. Vienintelis naudotojų pageidavimas – papildomi informaciniai laukai prie virtualių mašinų sąrašo. Kadangi tai nebuvo numatyta pradiniuose sistemos reikalavimuose, tai buvo traktuojama, kaip naujo funkcionalumo prašymas, o ne kaip defektas.

6.4. Sprendimo taikymo rekomendacijos

Sprendimą taikyti reiktų atsižvelgus į OpenStack versiją bei skirti laiko susipažinimui su komponentais, kurie sudaro debesų kompiuterijos sprendimą. SiteManager sistema nediegia Swift komponento, atsakingo iš duomenų saugojimą, nes jis yra vienas ir naudojamas globaliai per visą pasaulį, todėl jį reiktų įsidiesti atskirai.

Norint automatinio SiteManager sistemų registravimosi į CloudMap – tarp jų turi būti galimybė susisiekti per 80 sąsają. Komunikacija vykdoma šifruotu protokolu, todėl dėl saugumo rūpintis nereiktų (net ir pati informacija nėra slapta, jokie slaptažodžiai neperduodami).

Serveriai debesų kompiuterijos sprendimui turėtų būti HP, jei norima pilno SiteManager funkcionalumo (automatiškai atvaizduojamos serverių spintos panaudojant „HP Intelligent Series Rack“ galimybes nustatant serverių poziciją), tačiau SiteManager gali valdyti ir Dell serverius per IPMI sąsają, tačiau nebus galimybės matyti serverių spintų diagramas. Kitų gamintojų serveriai nepalaikomi, tačiau yra sukurta „VendorManager“ klasė, kurią galima papildyti moduliais, kurie leistų valdyti kitų gamintojų serverius.

7. REZULTATŲ APIBENDRINIMAS IR IŠVADOS

Šiame skyriuje pateikiamas rezultatų apibendrinimas iš prieš tai buvusių skyrių tai pastebėjimai. Taip pat pateikiamos pagrindinės tiriamojo darbo išvados bei pasiūlymai tolimesniai darbui.

7.1. Rezultatų apibendrinimas

Šio tiriamojo darbo metu nustatyta, jog norint išnaudoti visus OpenStack privalumus ir turėti stabilią diegimo ir prižiūrėjimo platformą – reikia nemažai funkcionalumo sukurti įmonės viduje arba kombinuoti skirtingas sistemas. Taigi, pagrindiniai OpenStack diegimo į fizinius serverius trūkumai:

- Didelis reikalingų paketų kiekis
- Konfigūracijos sudėtingumas
- OpenStack konfigūracijų tikrinimas (sudėtinga įsitikinti, ar visi komponentai tarpusavyje veikia)
- Fizinių serverių valdymo funkcionalumo nebuvimas (Horizon arba joks kitas komponentas negali diegti OpenStack į naujus serverius)
- Nėra galimybės agreguoti daugiau nei vienos OpenStack aplinkos informacijos.

Rinkoje esančių projektų, kurie analizuoja ir sprendžia šias problemas, dėka, pavyko sukurti specifikacijas naujai sistemai, kuri tiesiogiai sprendžia aukščiau išvardintas problemas. Suprojektuota ir sukurta sistema visus trūkumus išsprendžia tam tikrais etapais:

- Didelis reikalingų paketų kiekis tampa visiškai paprasta užduotimi, kai diegimą atlieka instaliacinis skriptas.
- Konfigūracijos sudėtingumas taip pat tampa ne problema, nes naudojami kintamieji ir konstantos, taigi IP adresai, duomenų bazių konfigūracijos yra automatizuotos ir visa informacija išsaugoma.
- OpenStack konfigūracijų tikrinimas yra atliekamas nuolat, vos tik pradėjus diegti sistemą. Taigi matome, kaip po truputį pradeda veikti vis daugiau servisų.
- Fizinių serverių valdymas išsprendžiamas per SiteManager sistemą, kai sukuriamos serverių spintas atvaizduojančios diagramos bei sukuriamas funkcionalumas, leidžiantis valdyti serverius per IPMI sąsają (priskirti operacinės sistemos atvaizdą, perkrauti ir t.t.).
- Daugelio OpenStack aplinkų informacijos agregacija yra pasiekama per CloudMap sistemą.

Galime teigti, jog tikslas (išsaugoti žinias sukuriant automatizuotą serverių valdymo ir OpenStack diegimo karkasą) buvo pilnai įgyvendintas. Naudotojų atsiliepimai apie sistemą buvo itin teigiami, sistemos sėkmingai susitvarkė su savo užduotimis. OpenStack diegimą atliko darbuotojai iš kitų komandų, kurios nebuvo susijusios su instaliacijos skripto kūrimu. Taip pat pademonstruotos galimybės kaupti informaciją iš daugelio aplinkų ir nuolat stebėti OpenStack sistemos būklę visuose duomenų centruose.

Tiriamojo darbo metu ne tik pademonstruotos didžiulės OpenStack API galimybės, tačiau ir parodoma, kad yra gana nesudėtinga, naudojantis Python programavimo kalba bei Django karkasu, sukurti dideles ir patikimas (sistemos taip pat buvo tikrintos prasiskverbimo atakomis) informacines sistemas. Django karkasas – vienas iš geriausiai dokumentuotų Python projektų yra geriausias įrankis, norint greitai ir efektyviai kurti naujas sistemas.

7.2. Išvados

1. Atlikus analizę nustatyta, kad:

- Yra aiškiai apibrėžtos dažniausios problemos diegiant paskirstytą programinę įrangą (tokią kaip OpenStack).
2. Toliau, sistemų kūrimo etape:
 - Suprojektuotos sistemos, galinčios automatizuoti OpenStack diegimą bei automatiškai rinkti debesų kompiuterijos sprendimo duomenis ir perduoti į centrinę sistemą.
 - Sukurtos dvi sistemos (SiteManager ir CloudMap), skirtos automatizuotam fizinių serverių aptikimui ir operacinių sistemų, OpenStack diegimui bei galinčios realiu laiku stebėti ir kaupti šių OpenStack aplinkų informaciją.
 3. Atlikus OpenStack diegimo ir priežiūros eksperimentą nustatyta, kad sukurtos sistemos yra tinkamos debesų kompiuterijos sprendimo diegimui, resursų peržiūrai realiu laiku, vidinių procesų informacijos agregavimui ir persiuntimui į centrinę sistemą. Ši centrinė sistema, dar žinoma kaip CloudMap, be naudotojų įsikišimo surinko visus SiteManager duomenis.

7.3. Pasiūlymai tolimesniam sistemos tobulinimui

Paprasčiausi sistemos patobulinimai, kurie galėtų būti sekančioje sistemos kūrimo fazėje:

- **Automatinis aptiktų resursų priskyrimas į veikiančią OpenStack aplinką** - šis funkcionalumas pasinaudotų jau egzistuojančiomis funkcijomis, tokiomis kaip serverių registravimo funkcijas (kai serveriai patys registruojasi į SiteManager sistemą arba aptinkami skenuojant potinklius), serverių rolės nustatymas (ši funkcija būtų panaudojama norint nustatyti, į kokią resursų grupę reikia priskirti šį serverį: valdymo serverius, skaičiavimo serverius arba duomenų saugojimo serverius). Turint šią informaciją, SiteManager sistema galėtų per IPMI sąsają savarankiškai priskirti operacinės sistemos atvaizdą, tuomet jį perkrauti ir pradėti OpenStack instaliaciją. Galimi pavojai – gali būti įdiegta OpenStack sistema į serverius, kurie būtų atsitiktinai aptikti tinkle, nors juos į debesį prijungti nenumatyta. Kitas svarbus aspektas, kuris galėtų sutrikdyti šį procesą – organizacijoje naudojama ServiceNow sistema, kurioje turėtų būti iškeltas įrašas, jog serveryje ketinama diegti naują operacinę sistemą.
- **Debesų sprendimo išnaudojamų resursų istorijos stebėjimas ir ateities prognozės** – šis papildomas funkcionalumas leistų matyti, kaip įsisavinami resursai ir kaip plečiasi arba traukiasi infrastruktūra. Ateities prognozėm reiktų pasirinkti tinkamą modelį, nes linijinis modelis nebūtų realistiškas. Šiam uždaviniui reiktų ne tik įvertinti, kaip keičiasi resursų išnaudojimas pastarosiom dienom arba savaitėm, tačiau įvertinti ir metų laikus, nustatant ryšius su prieš tai keliais metais buvusiais duomenimis. Būtų patartina išnaudoti dirbtinio intelekto ir kompiuterio mokymosi (*angl. k. „machine learning“*) algoritmus. Atlikus nedidelį tyrimą Python trečiųjų šalių bibliotekose rasta nemažai paketų, kurie galėtų padėti išspręsti šį uždavinį. Produktai, kurie bando nustatyti resursų išnaudojimą ir nuspėti, kada nebepavyks paleisti naujų virtualių mašinų – *BMC TrueSight*. Šio darbo rašymo metu, įrankį kurianti kompanija surengė prezentaciją apie jo galimybes ir buvo matoma, jog jis gana ankstyvoje kūrimo stadijoje, nes prognozėm nustatyti buvo naudojamas linijinis modelis, kuris visiškai neatitiko realybės. Kitas įrankis, kuris taip pat buvo įvertintas – *VMTurbo*, šiame įrankyje resursų išnaudojimo prognozių funkcionalumo šiuo metu nėra ir numatoma tik tolimoje versijoje, taigi įdiegti šią funkciją į SiteManager arba CloudMap būtų tikrai naudingas žingsnis, nes kol kas tai planuojama daryti tik su MS Excel pagalba ir nesinaudojant matematiniais modeliais.
- **Docker LXC konteinerių panaudojimas automatizuotame diegime** – tai viena iš populiariausių technologijų šiuo metu. Nors LXC konteineriai jau egzistuoja daugelį

metų – jie nebuvo plačiai naudojami dėl sudėtingo konteinerių mechanizmo. *Docker* taikomoji programa išsprendė šią problemą ir dabar vos keliomis komandomis galima per sekundę paleisti konteinerį. Pritaikomumas – CloudMap galėtų būti paruošta konteineryje, kurį reikėtų tiesiog paleisti. Taip būtų galima gana sudėtingą instaliaciją pakeisti į visiškai paprastą konteinerio paleidimą. Tačiau ši technologija dar šiek tiek per jauna, nes *Docker daemon*, galintis veikti tik „root“ teisėmis gali būti sukompromituotas ir tapti nesaugiu komponentu, tuomet iš konteinerio galima užgrobti visą serverį. Produkciniame tinkle šis sprendimas galėtų būti naudojamas tik išsprendus saugumo problemas.

- **Automatizuota sistemų instaliacija Chef pagalba** – kitas, itin populiarus sprendimas automatizuoti diegimus. Šiuo metu SiteManager naudoja *bash* skriptus, kurie diegia ir konfigūruoja komponentus. Jie galėtų būti pakeisti *Chef* receptais, kuriuos paleistų SiteManager sistema. Sistemos naudotojas neturėtų matyti ar pajusti skirtumo, kokia technologija naudojama žemesniajame lygyje. *Chef* sistema padėtų tvarkingiau tobulinti OpenStack diegimą, išskirstyti viską mažesniais komponentais. Taip pat jis galėtų užtikrinti atnaujinimus.

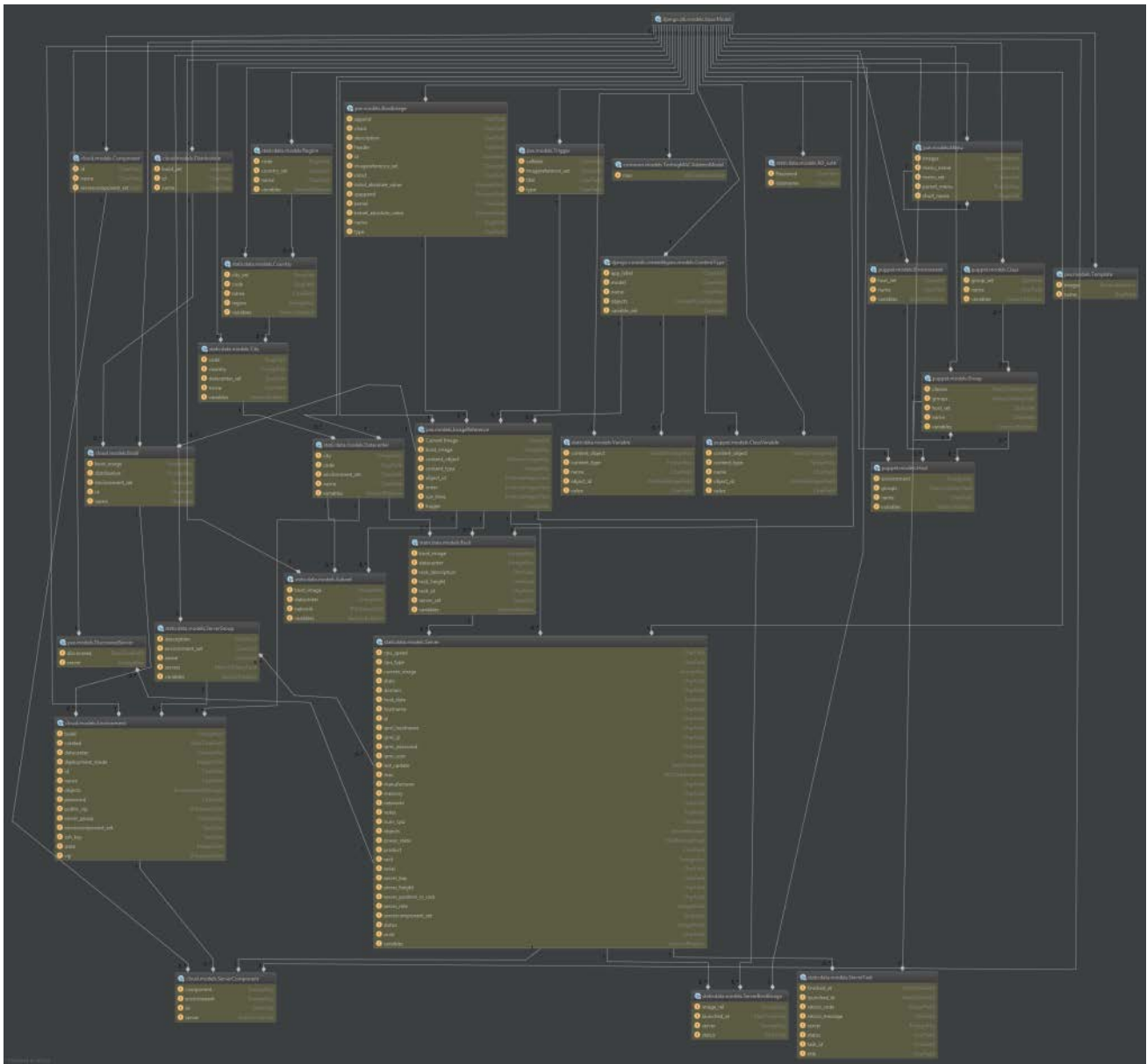
8. LITERATŪRA

- [1] P. Mell, T. Grance, „The NIST Definition of Cloud Computing, REF 800-145.,“ 2011.
- [2] B. Violino, „Bloomberg For Enterprise,“ 24 11 2014. [Tinkle]. Available: <http://bloomberg-datacap.cms.newscred.com/article/81-of-companies-shifting-mission-critical-apps-to-cloud/0c9fe1fed725496353e6d20bb042c91d>. [Kreiptasi 14 12 2014].
- [3] „Why banks are wary of public clouds,“ [Tinkle]. Available: <http://www.computerweekly.com/opinion/Why-banks-are-wary-of-public-clouds>. [Kreiptasi 02 05 2015].
- [4] „OpenStack Open Source Cloud Computing Software,“ 2009. [Tinkle]. Available: <https://www.openstack.org/>. [Kreiptasi 02 05 2015].
- [5] Jean-Paul Arcangeli, Raja Boujbel, Sébastien Leriche, „Automatic deployment of distributed software systems - Definitions and state of the art,“ 118 Route de Narbonne, F-31062 Toulouse, France, 2015.
- [6] O. community, „OpenSack Logical Architecture,“ [Tinkle]. Available: <http://docs.openstack.org/admin-guide-cloud/content/logical-architecture.html>. [Kreiptasi 03 05 2015].
- [7] Giuseppe Aceto, Alessio Botta, Walter de Donato, Antonio Pescapè, „Cloud monitoring: A survey,“ Napoli, Italy, 2013.
- [8] Fifield, Tom; Fleming, Diane; Gentle, Anne; Hochstein, Lorin; Proulx, Jonathan; Toews, Everett; Topjian, Joe, „Automated Deployment,“ įtraukta *OpenStack Operations Guide*, O'Reilly Media, Inc., 2014, p. 330.
- [9] „<http://pythonhackers.com/open-source/>,“ [Tinkle]. Available: <http://pythonhackers.com/open-source/>. [Kreiptasi 06 01 2015].
- [10] „Django code,“ Django Software Foundation, [Tinkle]. Available: <https://code.djangoproject.com/wiki/Fixtures>. [Kreiptasi 17 01 2015].

9. PRIEDAI

9.1. SiteManager sistemos duomenų bazės schema

Šiame skyriuje pateikiama SiteManager duomenų bazės schema. Ji yra automatiškai sugeneruojama per IntelliJ programavimo aplinką, nes naudojamas standartinės Django karkaso bibliotekos, skirtos darbui su duomenų bazėmis.



102. pav. Pav. SiteManager duomenų bazės modelis, sugeneruotas per IntelliJ IDE