

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
PROGRAMŲ INŽINERIJOS KATEDRA

Justinas Greibus

Duomenų bazių našumo tyrimo įrankis

Magistro darbas

Darbo vadovas
prof. dr. L. Nemuraitė

Kaunas, 2010

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
PROGRAMŲ INŽINERIJOS KATEDRA

Justinas Greibus

Duomenų bazių našumo tyrimo įrankis

Magistro darbas

Recenzentas

dr. Andrius Armonas
2010-05-25

Vadovas

prof. dr. L. Nemuraitė
2010-05-25

Atliko

IFM-4/2 gr. stud. Justinas Greibus
2010-05-25

Kaunas, 2010

Database Performance Audit Tool

Summary

The analysis of the database performance is the common challenge in the nowadays software testing. There are several methodologies of the analysis of the database performance in the market. However, tools, which are based on these methodologies, are available for the narrow circle of the privileged persons. According to the results of the analysis, this master thesis investigates a new methodology, which is based on several other methodologies.

The methodology of the database performance audit, which is discussed in this project, is based on the principles of the software load and stress testing. In order to identify the issues of database performance, values of the performance parameters are registered. These values are counted during the execution of the scenario of the automatic scenarios. The user has the possibility to re-execute historical scenarios and to compare the results of the separate executions. Generated reports with the deep data level facilitate the analysis of the database performance

The developed tool may be used by software architect or quality engineer of any software development team.

Key words: database performance, testing, audit, scenarios.

Turinys

1.	Įvadas	9
2.	Programinės įrangos testavimo metodų analizė	11
2.1.	Duomenų bazių testavimo svarba	14
2.2.	Išskylančios duomenų bazių testavimo problemos	16
2.3.	Testavimo įrankių analizė ir palyginimas	19
2.4.	Duomenų bazių našumo tyrimo įrankio kūrimo prielaidos	21
2.4.1.	Sistemos vartotojų analizė	21
2.4.2.	Architektūros ir įgyvendinimo priemonių analizė	22
2.4.3.	Siekiamos sistemos apibrėžimas	23
2.4.4.	Darbo tikslai	24
2.4.5.	Kompiuterizuojamos sistemos funkcijos	25
2.5.	Analizės išvados	25
3.	Duomenų bazių našumo tyrimo metodika	27
4.	Reikalavimai tyrimo įrankiui sukurti	30
4.1.	Duomenų bazių našumo tyrimo įrankio funkcinių reikalavimų specifikacija	30
4.2.	Dalykinės srities modelis	42
4.3.	Nefunkciniai reikalavimai ir apribojimai	43
4.3.1.	Reikalavimai sistemos išvaizdai	43
4.3.2.	Reikalavimai panaudojamumui	43
4.3.3.	Reikalavimai vykdymo charakteristikoms	43
4.3.4.	Reikalavimai veikimo sąlygoms	43
4.3.5.	Reikalavimai sistemos priežiūrai	43
4.3.6.	Reikalavimai saugumui	43
4.4.	Reikalavimų analizė	43
5.	Duomenų bazių našumo tyrimo įrankio projektas	47
5.1.	Sistemos architektūra	47
5.1.1.	Loginė sistemos architektūra	47
5.1.2.	Vartotojo paslaugos	48
5.1.3.	Veiklos paslaugos	49
5.1.4.	Duomenų paslaugos	50
5.2.	Detalus projektas	50
5.3.	Sistemos elgsenos modelis	53
5.4.	Duomenų bazės schema	61
6.	Tyrimo įrankio realizacija	64
6.1.	Realizacijos modelis	64
6.2.	Sistemos diegimas	65
6.2.1.	Reikalavimai techninei įrangai	65
6.3.	Vartotojo sąsaja	66
6.3.1.	IS tikslai	66
6.3.2.	Administravimas	67
6.3.3.	Veiksmų scenarijai	72
6.4.	Testavimo rezultatai	83
7.	Pavyzdinis duomenų bazės našumo tyrimas	88
7.1.	Tyrimo aprašymas	88
7.2.	Tyrimo rezultatai	91
8.	Išvados	95
9.	Literatūros sąrašas	96
	Priedai	98
	Priedas A. Pavyzdinio duomenų bazių našumo tyrimo rezultatai	98

Priedas B Įrankio diegimo aktas103

Paveikslėlių sąrašas

3.1 pav. Programinės įrangos testavimo kryptys	12
3.2 pav. Duomenų bazės testavimo sistemos struktūra	15
3.3 pav. Interneto taikomosios programos realizacinės architektūros schema	22
3.4 pav. „Windows“ tipo programos realizacinė schema	22
3.5 pav. Sistemos kontekstinė diagrama	24
3.6 pav. Kompiuterizuojami duomenų bazių našumo įrankio panaudojimo atvejai	25
4.1 pav. Duomenų bazės našumo tyrimo procesas	28
5.1 pav. PA „Prisijungti“ veiklos diagrama	30
5.2 pav. PA „sukurti projektą“	31
5.3 pav. PA „Redaguoti projekto informaciją“ veiklos diagrama	32
5.4 pav. PA „Sukurti sistemos vartotoją“ veiklos diagrama	33
5.5 pav. PA „Priskirti vartotojui teises“ veiklos diagrama	33
5.6 pav. PA „Redaguoti sistemos vartotojo informaciją“ veiklos diagrama	34
5.7 pav. PA „Nustatyti prisijungimo informaciją prie išorinių sistemų“ veiklos diagrama	35
5.8 pav. PA „sukurti vartotojo veiksmų scenarijų“ veiklos diagrama	36
5.9 pav. PA „sukurti atliekamus veiksmus“ veiklos diagrama	37
5.10 pav. PA „Peržiūrėti vartotojo veiksmų scenarijų sąrašą“ veiklos diagrama	37
5.11 pav. PA „Atlikti vartotojo veiksmų scenarijų“ veiklos diagrama	38
5.12 pav. PA „Sukurti automatinį veiksmų scenarijų“ veiklos diagrama	39
5.13 pav. PA „Registruoti klaidą“ veiklos diagrama	39
5.14 pav. PA „Peržiūrėti automatinį veiksmų scenarijų sąrašą“ veiklos diagrama	40
5.15 pav. PA „Vykdėti pasirinktą automatinį veiksmų scenarijų“ veiklos diagrama	40
5.16 pav. PA „Registruoti automatinio veiksmų scenarijaus metrikas“ veiklos diagrama	41
5.17 pav. PA „Formuoti ataskaitas“ veiklos diagrama	42
5.18 pav. Dalykinės srities modelis	42
5.19 pav. Prisijungimo analizės diagrama	43
5.20 pav. Projekto sukūrimo analizės diagrama	44
5.21 pav. Vartotojo sukūrimo analizės diagrama	44
5.22 pav. Vartotojo veiksmų sukūrimo analizės diagrama	44
5.23 pav. Vartotojo veiksmų scenarijaus vykdymo analizės diagrama	45
5.24 pav. Automatinio veiksmų scenarijaus sukūrimo analizės diagrama	46
5.25 pav. Automatinio veiksmų scenarijaus vykdymo analizės diagrama	46
6.1 pav. Loginė sistemos architektūra	48
6.2 pav. Vartotojo sąsajos modelis	48
6.3 pav. Sistemos veiklos logikos modelis	49
6.4 pav. Duomenų paslaugų modelis	50
6.5 pav. PA „Prisijungti“ realizacijos diagrama	50
6.6 pav. PA „Sukurti projektą“ realizacijos diagrama	51
6.7 pav. PA „Sukurti vartotoją“ realizacijos diagrama	51
6.8 pav. PA „Sukurti vartotojo veiksmų scenarijų“ realizacijos diagrama	51
6.9 pav. PA „Vartotojo veiksmų scenarijaus vykdymas“ realizacijos diagrama	52
6.10 pav. PA „Sukurti automatinį veiksmų scenarijų“ realizacijos diagramą	52
6.11 pav. PA „Vykdėti automatinį veiksmų scenarijų“ realizacijos diagrama	52
6.12 pav. PA „Prisijungti“ sekų diagrama	54
6.13 pav. PA „Sukurti projektą“ sekų diagrama	55
6.14 pav. PA „Sukurti vartotoją“ sekų diagrama	56
6.15 pav. PA „Vartotojo veiksmų scenarijaus sukūrimas“ sekų diagrama	57
6.16 pav. PA „Vartotojo veiksmų scenarijaus vykdymas“ sekų diagrama	58
6.17 pav. PA „Automatinio veiksmų scenarijaus sukūrimas“ sekų diagrama	59
6.18 pav. PA „Automatinio veiksmų scenarijaus vykdymas“ sekų diagrama	60

6.19 pav. Duomenų bazės schema	61
7.1 pav. Sistemos komponentų diagrama	64
7.2 pav. Sistemos komponentų ir artefaktų sąryšių diagrama	65
7.3 pav. Sistemos diegimo diagrama	65
7.4 pav. Sistemos vartotojų sąrašas	68
7.5 pav. Naujo sistemos vartotojo sukūrimo forma	69
7.6 pav. Projektų administravimo langas.....	71
7.7 pav. TFS nustatymų administravimo langas.....	72
7.8 pav. Vartotojo veiksmų scenarijaus sąrašas.....	73
7.9 pav. Naujo vartotojo veiksmų scenarijaus sukūrimo langas.....	75
7.10 pav. Veiksmų scenarijų vykdymo langas	79
7.11 pav. Automatinių veiksmų scenarijų pasirinkimo ir vykdymo langas.....	81
7.12 pav. Klaidos registravimo langas.....	82
8.1 pav. Duomenų modelis	89
8.2 pav. Eksperimentų rezultatai (automatinių veiksmų scenarijaus vykdymo laikas, min)....	91
8.3 pav. Eksperimentų rezultatai (procedūros vykdymo trukmė, s)	91
8.4 pav. Eksperimentų rezultatai (CPU užimtumas, %)	92

Lentelių sąrašas

3.1 lentelė. Duomenų bazės elementų testavimas pagal skirtingas metodikas	15
3.2 lentelė. Lyginamoji įrankių analizė	21
5.1 lentelė. Panaudojimo atvejis „Prisijungti“	30
5.2 lentelė. Panaudojimo atvejis „Sukurti projektą“	30
5.3 lentelė. Panaudojimo atvejis „Redaguoti projekto informaciją“	31
5.4 lentelė. Panaudojimo atvejis „sukurti sistemos vartotoją“	32
5.5 lentelė. Panaudojimo atvejis „Priskirti vartotojui teises“	33
5.6 lentelė. Panaudojimo atvejis „Redaguoti sistemos vartotojo informaciją“	34
5.7 lentelė. Panaudojimo atvejis „Nustatyti prisijungimo informaciją prie išorinių sistemų“	34
5.8 lentelė. Panaudojimo atvejis „Sukurti vartotojo veiksmų scenarijų“	35
5.9 lentelė. Panaudojimo atvejis „Sukurti atliekamus veiksmus“	36
5.10 lentelė. Panaudojimo atvejis „Peržiūrėti vartotojo veiksmų scenarijų sąrašą“	37
5.11 lentelė. Panaudojimo atvejis „Atlikti vartotojo veiksmų scenarijų“	38
5.12 lentelė. Panaudojimo atvejis „Sukurti automatinį veiksmų scenarijų“	38
5.13 lentelė. Panaudojimo atvejis „Registruoti klaidą“	39
5.14 lentelė. Panaudojimo atvejis „Peržiūrėti automatinio veiksmų scenarijų sąrašą“	40
5.15 lentelė. Panaudojimo atvejis „Vykdėti pasirinktą automatinį veiksmų scenarijų“	40
5.16 lentelė. Panaudojimo atvejis „Registruoti automatinio scenarijaus metrikas“	41
5.17 lentelė. Panaudojimo atvejis „Formuoti ataskaitas“	41
6.1 lentelė. Duomenų bazės lentelė „Vartotojas“	61
6.2 lentelė. Duomenų bazės lentelės „Vartotojo teise“	61
6.3 lentelė. Duomenų bazės lentelė „Projektas“	62
6.4 lentelė. Duomenų bazės lentelė „Projekto nustatymai“	62
6.5 lentelė. Duomenų bazių lentelė „Vartotojo veiksmų scenarijus“	62
6.6 lentelė. Duomenų bazės lentelė „Scenarijaus veiksmas“	62
6.7 lentelė. Duomenų bazės lentelė „Klaida“	62
6.8 lentelė. Duomenų bazės lentelė „Prioritetas“	62
6.9 lentelė. Duomenų bazės lentelė „Scenarijaus rezultatas“	63
6.10 lentelė. Duomenų bazės lentelė „Rezultatas“	63
6.11 lentelė. Duomenų bazės lentelė „Automatinio veiksmų scenarijus“	63
6.12 lentelė. Duomenų bazės lentelė „Automatinio vykdymo metrikos“	63
7.1 lentelė. Reikalavimai interneto tarnybinei stočiai (sistemos serveriui)	65
7.2 lentelė. Reikalavimai duomenų bazių serveriui	66
7.3 lentelė. Rolėms pasiekiamos funkcijos	68
7.4 lentelė. Naujo projekto sukūrimo detalus testavimas	83
7.5 lentelė. Sistemos vartotojo sukūrimo detalus testavimas	84
7.6 lentelė. Rolės suteikimas pasirinktam sistemos vartotojui	85
7.7 lentelė. Vartotojo veiksmų scenarijaus sukūrimo detalus testavimas	86
7.8 lentelė. Automatinio veiksmų scenarijaus sukūrimo detalus testavimas	87
8.1 lentelė. Eksperimento vykdymo tvarka	90
8.2 lentelė. Eksperimentų apibendrinti rezultatai	93
10.1 lentelė. E ₁ rezultatai	98
10.2 lentelė. E ₂ rezultatai	98
10.3 lentelė. E ₃ rezultatai	99
10.4 lentelė. E ₄ rezultatai	100
10.5 lentelė. E ₅ rezultatai	101

1. Įvadas

Programinės įrangos kūrimo organizacijose dažniausiai yra palaikomi visi programinės įrangos kūrimo etapai. Vienas iš šių etapų yra kuriamos programinės įrangos testavimas. Programinės įrangos testavimo etapo metu yra ieškoma ne tik kritinių sistemos klaidų, bet ir tikrinamas sukurtos programinės įrangos atitikimas užsakovų suformuluotiems reikalavimams. Viena aktualiausių programinės įrangos kūrimo įmonių problemų yra užtikrinti kuriamos programinės įrangos stabilų veikimą stresinėse situacijose ir tinkamai suvaldyti išskylančias našumo problemas. Šiai problemai spręsti yra taikomi programinės įrangos stresiniai testai bei programinės įrangos vykdomų užklausų analizė.

Nemažiau nei programinės įrangos našumo problemos aktualios yra ir duomenų bazių našumo problemos. Duomenų bazių tyrimo srityje vyksta vis daugiau tyrimų, siekiančių nustatyti efektyvius būdus nustatyti bei pašalinti duomenų bazių našumo problemas. Viena didžiausių problemų, su kuria yra susiduriama atliekant duomenų bazių tyrimus, yra duomenų bazių užklausų semantika.

Šio darbo **tyrimo sritis** yra duomenų bazių našumo problemų nustatymo metodika ir jos praktinis taikymas, o **tyrimo objektas** – duomenų bazių našumo tyrimo įrankis.

Šio darbo **tikslas** palengvinti duomenų bazių kūrėjų veiklą, sukuriant įrankį, kuris padėtų nustatyti duomenų bazių našumo problemines vietas naudojantis kompiuterizuotu duomenų bazių našumo tyrimo įrankiu.

Darbo uždaviniai:

- Nustatyti teisingą duomenų bazių našumo problemų tyrimo metodikos koncepciją;
- Nustatyti reikiamą įrankio funkcionalumą, kokybines charakteristikas, pasirinkti tinkamas realizavimo priemones;
- Suprojektuoti įrankį pagal apibrėžtus reikalavimus;
- Realizuoti ir ištestuoti įvertinimo įrankį;
- Atlikti bandomąjį duomenų bazės našumo tyrimą;
- Nustatyti įrankio kokybines charakteristikas ir tinkamumą.

Šiame darbe naudojama duomenų bazių našumo tyrimo metodika yra pagrįsta programinės įrangos stresinio bei apkrovos testavimo principais. Šio darbo metu kuriamas įrankis yra pritaikytas atlikti *MS SQL* tipo duomenų bazių našumo tyrimus. Pagrindiniai kintamieji, skirti nustatyti duomenų bazių našumo problemas, yra šie: duomenų bazės

procedūros vykdymo laikas, kompiuterio, kuriame įdiegta *MS SQL* duomenų bazė, procesoriaus apkrova bei loginių nuskaitymų ir loginių įrašymų skaičius. Pateikiant registruotas šių kintamųjų reikšmes skirtingais pjūviais, yra nustatomos tiriamos duomenų bazės našumo probleminės vietos. Ši metodika bei pagal šią metodiką sukurtas įrankis skirtas ne nustatyti klaidingą duomenų bazės veikimą, bet nustatyti bei pateikti vartotojui duomenų bazės vietas, kurios gali būti identifikuojamos kaip probleminės vykdymo laiko požiūriu.

Sukurta sistema atlieka automatinį registruotų vartotojo veiksmų scenarijų vykdymą bei registruoja nustatytus duomenų bazių našumo tyrimo kintamuosius. Pagal gaunamus rezultatus vartotojui yra pateikiamos išsamios ataskaitos. Šiuo įrankiu vartotojas gali atlikti jau vykdytus scenarijus tiriamos sistemos tolimesnės evoliucijos eigoje. Ši galimybė vartotojui leidžia palyginti duomenų bazės našumo tyrimo kintamuosius sistemos kūrimo pradžioje su duomenų bazės našumo tyrimo komponentais sistemos kūrimo pabaigoje.

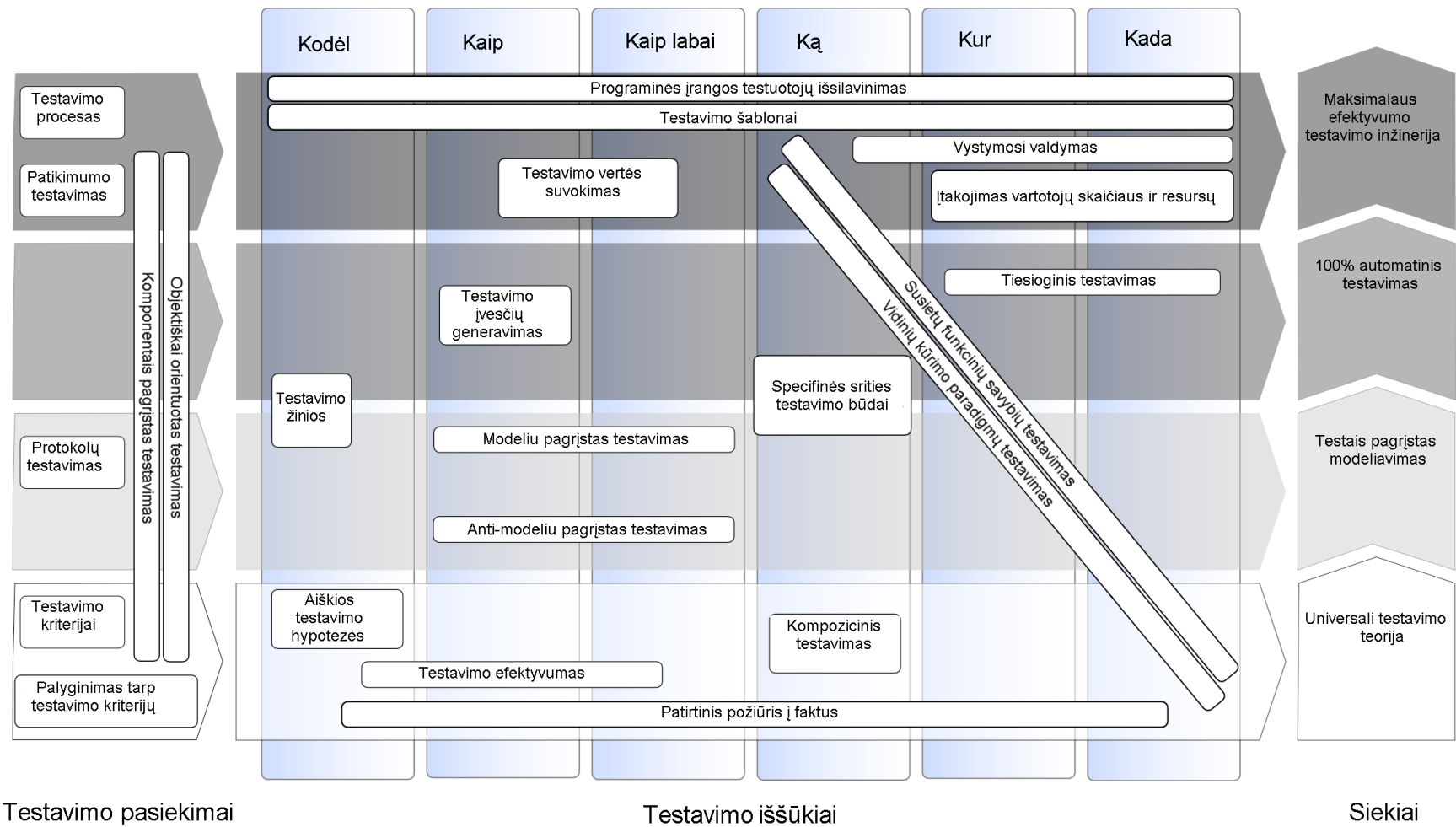
Sukurta įrankis buvo bandomas atliekant Kauno technologijos universiteto Informacijos sistemų katedros vidinių publikacijų svetainės tyrimą. Taip pat šis darbas yra įdiegtas IN-VOLV bendrovės vidinėje sistemoje. Pagal pateiktus vertinimus ir pastabas tolimesnėje šio darbo vystymo perspektyvoje yra numatoma plėsti duomenų bazių našumo tyrimo kintamųjų sąrašą įterpiančią fizinių skaitymų bei įrašymų skaičių bei pritaikant sistemą atlikti *Oracle* tipo duomenų bazių našumo tyrimą.

Darbo struktūra.

- Antrajame šio darbo skyriuje yra pateikiama programinės įrangos testavimo tendencijų bei duomenų bazių testavimo problemų analizė, lyginami egzistuojantys sprendimai, nagrinėjamos galimos įgyvendinimo priemonės bei nustatomos kompiuterizuojamos įrankio funkcijos.
- Trečiajame skyriuje pateikiama sudaryta duomenų bazių našumo tyrimo metodika.
- Ketvirtajame skyriuje yra pateikiami detalizuoti duomenų bazių našumo tyrimo įrankio funkciniai bei nefunkciniai reikalavimai, nustatomi atskirų įrankio komponentų tarpusavio ryšiai.
- Penktajame skyriuje pateikiamas duomenų bazių našumo tyrimo įrankio projektas.
- Šeštajame skyriuje aprašoma duomenų bazių našumo tyrimo įrankio realizacija, jo veikimas bei testavimo rezultatai.
- Septintajame šio darbo skyriuje pateikiamas duomenų bazių našumo tyrimo įrankio eksperimento aprašymas bei rezultatai.
- Aštuntajame skyriuje pateikiamos suformuluotos darbo išvados.

2. Programinės įrangos testavimo metodų analizė

Programinės įrangos testavimas yra labai plati sąvoka, apimanti įvairaus spektro veiklas, nuo mažo kodo segmento testavimo iki didelių informacinių sistemų testavimo. Kiekvieno programinės įrangos kūrimo etapo metu yra atliekamas sistemos ar atskirų jos dalių testavimas, pradedant nuo reikalavimų testavimo, atskirtų modulių tarpusavio sąveikos testavimo iki vartotojo sąsajos ir integracijų su kitomis sistemomis testavimo. Nors sistemos testavimas yra vykdomas visuose sistemos kūrimo etapuose, tačiau taikomi metodai ir siekiami tikslai skiriasi. Todėl atsižvelgiant į programinės įrangos testavimo sričių įvairovę, testavimo veiklos procesų tyrimai yra vykdomi daugelį skirtingų kryptių. Italijos nacionalinės mokslo tarybos mokslininkė Antonia Bertolino programinės įrangos testavimo veiklą skirsto į tris pagrindines grupes: testavimo pasiekimai, egzistuojantys iššūkiai ir tolimesni siekiai [10].



2.1 pav. Programinės įrangos testavimo kryptys

Detaliau nagrinėjant programinės įrangos testavimo kryptis, galima išskirti keturis pagrindinius siekius: universali testavimo teorija, testais pagrįstas modeliavimas, 100% automatinis testavimas bei maksimalaus efektyvumo testavimo inžinerija. Šiuo metu aktualiausias programinės įrangos testavimo siekis yra testais pagrįstas modeliavimas. Šio siekio pagrindinė idėja yra pagrįsta modelių apibrėžtų sistemose pagrindu testavimo scenarijų ir testavimo atvejų formavimu ir generavimu. Šiems tikslams pasiekti yra bandoma pritaikyti UML ar Z modeliavimo notacijas.

Modeliu pagrįstas testavimas (*angl.* model-based testing) yra vienas iš modeliavimo metodu skirtų suformuoti loginius segmentus (*angl.* artifacts) skirtus vykdyti programinės įrangos testavimą. Šiam tikslui yra naudojamas modelis, kuris apibrėžia visus testinių duomenų aspektus, bet daugiausia skirta apibrėžti testavimo atvejus ir testavimo aplinką. Dažniausiai testinis modelis yra visiškai arba dalinai išskirtas iš modelio, kuris apibūdina testuojamos sistemos funkcinius reikalavimus. Modelis apibūdinantis testuojamą sistemą dažniausiai yra abstraktus, apibūdinantis sistemos elgesį. Suformuoti testavimo atvejai iš šio modelio yra priskiriami funkciniam testams ir yra to paties abstrakcijos lygio kaip ir modelis [10].

Remiantis šiuo testavimo iššūkiu yra atliekami moksliniai darbai Kauno technologijos universitete siekiant išplėsti sistemų integracinius testus pagrįstus UML „*Communication*“ ir „*State Machine*“ modeliais ar nustatant sistemos testų priklausomybę nuo sistemos baigtumo lygio [11, 13].

Vertinant sistemas, kurios yra kuriamos remiantis egzistuojančiais komponentais, kuriems yra neprieinamos specifikacijos ar komponentų išeities kodai, tokių sistemų kūrimui yra sudėtinga pritaikyti modeliu pagrįstą testavimą arba šis metodas būtų labai brangus. Todėl tokio tipo sistemoms galima pritaikyti anti modeliu pagrįstą testavimo metodą (*angl.* anti model-based testing). Vadovaujantis šiuo metodu, yra atliekamas pavyzdinis darbas su sistema ir fiksuojami visi atlikti veiksmai bei veikslių rezultatai. Analizuojant atliktus veiksmus ir gautas reakcijas į šiuos veiksmus, yra sukuriamas pavyzdinis sistemos modelis, kuris pritaikomas tolimesniems sistemos testavimams. Bet kuriuo atveju tiek modeliu pagrįstas testavimas, tiek anti modeliu pagrįstas testavimas remiasi pagrindiniu principu: vienoje testavimo sąveikos pusėje yra sistemos modelis, kitoje sąveikos pusėje yra sistema [13].

Kitas svarbus iššūkis priskiriamas testais pagrįsto modeliavimo siekiui yra specifinių sričių testavimo būdų tyrimas ir analizė. Dabartiniu metu specifinės srities kalba (*angl.*

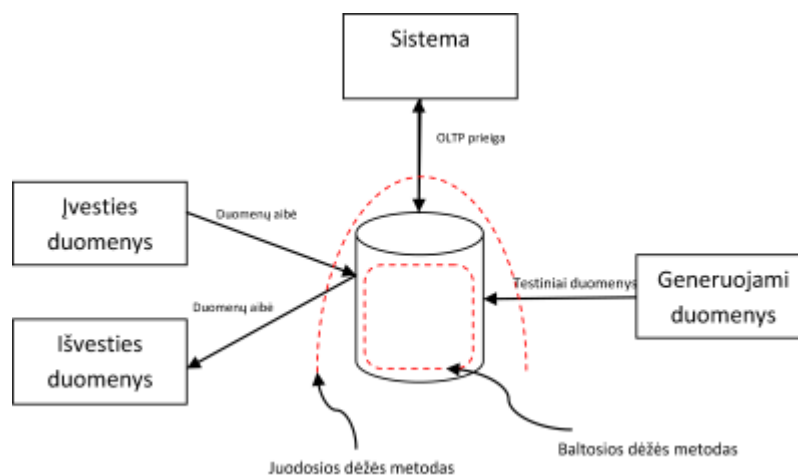
domain specific language) leidžia ekspertams išreikšti specifinės srities abstrakčias specifikacijas glaudžiau jas susiejant su kritinėmis situacijomis ir kurios gali būti automatinių būdų perverstos į funkcionuojančias sistemas. Specifinių sričių testavimo metodai yra taikomi atliekant duomenų bazių, vartotojo sąsajos, internetinių sistemų ar telekomunikacinių sistemų testavimus.

2.1. Duomenų bazių testavimo svarba

Iš paminėtų sričių tyrimų gausa išsiskiria duomenų bazių testavimas. Duomenų bazių testavimas yra kritinė sritis siekiant užtikrinti aukštą programinės įrangos kokybę ir apsaugant sistemas nuo neatitaisomo duomenų praradimo. Tradiciškai yra nusistovėję, kad duomenų bazių testavimo metu yra tikrinama ar nustatytų grupių funkcijos gali būti įvykdytos. Todėl testuoti duomenų bazę yra labai svarbu galima paminėti kelis aspektus:

- Duomenys yra vienas vertingiausių kuriamos sistemos kertinių akmenų. Kuriamos sistemos sėkmė priklauso nuo to, kaip sistemoje naudojami duomenys bus greitai apdorojami, kaip greitai bus gaunami atsakymai į pateiktas užklausas.
- Didžioji dalis kuriamų sistemų turi sąryšius su duomenų bazėmis, kuriose realizuojama daugelis sistemos logikos elementų. Siekiant užtikrinti aukštą sistemos logikos lygį yra būtina vykdyti duomenų bazių testus lygiai taip pat kaip ir pačios sistemos.
- Dabartiniai metodai, skirti užtikrinti duomenų kokybę, nėra efektyvūs. Daugelio įmonių praktika rodo, kad duomenų bazių schemų atnaujinimai yra atliekami duomenų bazių administratorių. Programuotojai prižiūrintys sistemą tik vizualiai patikrina pakeitimus. Tokie darbo metodai neužtikrina nepriekaištingo sistemos darbo.
- Testavimas padeda nustatyti problemų šaltinius. Neatliekant duomenų bazių testavimo, neįmanoma pasakyti, ar suprojektuota duomenų bazė yra efektyvi ir be klaidų.

Atsakius į klausimą kodėl yra svarbus duomenų bazių testavimas, kitas svarbus klausimas yra, ką tiksliai mes turėtume testuoti.



2.2 pav. Duomenų bazės testavimo sistemos struktūra

Pagal pateiktą sistemos struktūrą duomenų bazės testavimą galima išskaidyti pagal dvi testavimo metodikas: juodosios dėžės testavimo metodika ir baltosios dėžės testavimo metodika [6].

Juodosios dėžės testavimo metodika.

Juodosios dėžės testavimo metodika arba dar kitaip vadinamas funkcinis testavimas skirtas tikrinti sistemos funkcijas, kurios skirtos sistemos bendravimui su išore (duomenų sąrašų formavimas, duomenų perdavimas, duomenų saugojimas) [7]. Ši metodika yra pagrįsta sistemos išeities duomenų tikrinimu pagal pateiktus įvesties duomenis, neatsižvelgiant dėmesio, kokie procesai yra vykdomi sistemos viduje. Ši metodika gali būti taikoma įvairiuose testavimo lygiuose: vienetų testavimas, integracinis testavimas, sisteminis testavimas ar priėmimo testavimas.

Baltosios dėžės testavimo metodika

Baltosios dėžės testavimo metodika arba dar kitaip vadinamas struktūrinis testavimas skirtas tikrinti vidinių sistemos vienetų (sistemos sąlygų, duomenų struktūrų, duomenų blokų) vientisumą ir teisingumą [7]. Baltos dėžės testavimo metodai, tokie kaip sakinių testavimas, būsenų testavimas, prieigų testavimas gali būti pritaikyti testuoti dalį arba visą sistemos duomenų bazę. Tačiau duomenų bazių testavimas yra komplikotas dėl vidinių apribojimų bei SQL sakinių semantikos. Dėl šios priežasties vidiniai SQL sakiniai yra traktuojami kaip juodosios dėžės.

2.1 lentelė. Duomenų bazės elementų testavimas pagal skirtingas metodikas

Sąsajos elementai testuojami juodosios dėžės metodu	Duomenų bazės elementai testuojami baltosios dėžės metodu
---	---

<ul style="list-style-type: none"> • Įeinančių duomenų reikšmės; • Išeinančių duomenų reikšmės (užklausų, procedūrų rezultatai) 	<ul style="list-style-type: none"> • Procedūros (angl. Stored procedures); • Duomenų bazės vaizdai (angl. Views); • Apribojimai (angl. Constraints); • Pradinės stulpelių reikšmės; • Duomenų reikšmių aibės.
---	--

2.2. Išskylančios duomenų bazių testavimo problemos

Į duomenų bazių sistemą kaip ir į bet kokią kitą programą galima žiūrėti iš paprastos funkcijos perspektyvos, kuri turi įėjimą bei išėjimą. Žvelgiant iš tokios perspektyvos duomenų bazių sistemos testavimo metu būtų svarbu apibrėžti įėjimo ir išėjimo reikšmių specifikacijas. Pažymėjus įėjimo reikšmę ir įvykdžius reikalingus sistemos veiksmus, būtų lyginama išėjimo reikšmė su reikšme, nurodyta išėjimo specifikacijoje. Tačiau priešingai nei tradicinės programos, kurios yra analizuojamos literatūroje, duomenų bazių įėjimo ir išėjimo reikšmių struktūra yra komplikuoja, kuri duomenų bazių sistemų testavimą padaro sudėtingą. Daugumos programų atžvilgių įėjimo ir išėjimo reikšmių tikrinimo su šių reikšmių specifikacijomis rezultatas yra netrivialus. Tačiau duomenų bazių sistemos testavimo metu yra susiduriama su problema, kai specifikacijoje nurodyta ir po sistemos vykdymo gauta reikšmė turi būti panaudota siekiant nustatyti, ar sistema elgiasi teisingai.

Yra keletas galimų būdų, kaip elgtis su duomenų bazės būseną. Vienas iš šių būdų yra tiesiog ignoruoti būseną ir atkreipti dėmesį tik į vartotojo įvedamus įėjimo duomenis ir sistemos pateikiamus išėjimo duomenis. Tačiau šis būdas pasižymi ir neigiamomis savybėmis. Naudojant šį būdą, yra neįmanoma validuoti testavimo rezultatus ar pakartotinai atlikti testavimo atvejus. Kitas būdas yra laikyti duomenų bazės būseną kaip sistemos aplinkos nustatytą savybę ir tiksliai apibrėžti, kaip ši būseną įtakoja specifikacijoje nurodytas bei realiai gaunamas išėjimo reikšmės. Tokiu atveju išskyla kontroliavimo ir pastebėjimo problemos. Kontroliavimo problema išskyla siekiant užtikrinti tinkamą sistemos būseną prieš vykdant testavimo atvejus, o pastebėjimo problema išskyla nustatant sistemos būseną po testavimo atvejų įvykdymo.

Kita problema, kuri išskyla duomenų bazių sistemų testavimo metu, yra naudojami testiniai duomenys. Šiuo atveju gali būti naudojami realūs, testavimo metu duomenų bazėje esantys duomenys. Tačiau tokių duomenų naudojimas turi keletą minusų. Kaip alternatyva realiems duomenims gali būti specialiai testiniais tikslais sugeneruoti duomenys, kurie būtų vykdomi uždaroje testinėje aplinkoje. Kai duomenų bazės būseną yra sąryšių būsenų grupė, kuri kiekviena yra domeno poaibis, iš to išplaukia, kad viskas, ko reikia, tai sugeneruoti duomenis

kiekvienam duotam domeniui ir susieti juos sudarant lenteles. Tačiau tokiu atveju yra ignoruojamas pagrindinis duomenų bazių aspektas – vientisumo apribojimai.

Programinės įrangos su prieiga prie duomenų bazės naudojant SQL užklausas testavimo procesas yra keblus dėl keleto išskylančių problemų.

- Pirma užduotis yra sukurti pradinius egzempliorius, skirtus testinėm duomenų bazėms. Šis pirmas žingsnis yra labai svarbus siekiant užtikrinti gerą testų grupę, kurie bus įėjimo duomenys bet kuriame SQL sakinyje (*angl.* SQL statement). Yra svarbu nuspręsti, kokie duomenys yra aktualūs, ir kurie duomenys bei kiek kartų bus reikalingi. Dar daugiau, įvairios testų kombinacijos turi būti numatytos siekiant padengti kuo daugiau esančių SQL užklausių atvejų. Jeigu yra tik keletas egzempliorių, tokiu atveju duomenų bazių naudojimas tampa daug lengvesnis, o tai įtakoja paprastesnį ir lengvesnį būdą atrasti kuo daugiau klaidų.
- Kitas testinės duomenų bazės struktūros aspektas yra tai, kad sistema neturi tik vienos formuluotės. Todėl duomenys turi būti tinkami kuo didesniai galimų operatorių skaičiui, o tai labai ženkliai padidina resursų kaštus.
- SQL sakiniai gali būti parametrizuojami naudojant kintamuosius ar konstantas. Dėl šios priežasties ruošiant testavimo atvejus turi būti numatyti įėjimo duomenys ir šiems parametrams.

Šios duomenų bazių testavimo problemos yra bendros daugeliui testavimo metodų. Tačiau yra viena tyrimo sritis, kuri mažiau imli šioms problemoms t.y. duomenų bazių našumo problemų tyrimas. Kaip ir programinės įrangos taip ir duomenų bazių našumas matuojamas iš skirtingų perspektyvų siekiant pagerinti kuriamos sistemos sugebėjimą išlaikyti aukštą kokybę ir sistemos našumą keičiantis apimčiai (*angl.* Scalability). Stresinis testavimas (*angl.* Stress testing) ir apkrovos testavimas (*angl.* Load testing) yra testavimo strategijos, kurios taikomos sistemos kūrimo metu siekiant užtikrinti norimą kuriamo produkto lygį.

Apkrovos testavimas. Ši testavimo strategija taikoma modeliuojant tikėtiną programinės įrangos naudojimą ir imituojant tikėtiną vartotojų grupės darbą su sistema. Sistemos reakcija esant šioms darbo sąlygoms yra įtakoje įvairių faktorių: sistemos atminties panaudojimas, sistemos pralaidumo (*angl.* Throughput) panaudojimas ir kitų. Visos šios sąlygos suteikia galimybę nustatyti sistemos trūkumus, susijusius su sistemos apkrova ir ištaisyti juos iki sistemos sukūrimo pabaigos [4].

Stresinis testavimas. Šios strategijos tikslas yra nustatyti sistemos trūkumus sistemai reaguojant į apkrovos testavimą viršijantį nustatytus maksimalius apribojimus (vartotojų skaičius, atliekamų operacijų skaičius, sistemos veikimo periodas) [4].

Kuriamos sistemos našumo probleminės sritys yra išskiriamos analizuojant kuriamos programinės įrangos architektūrą bei atlikus programinės įrangos būsimų vartotojų apklausą ir išsiaiškinus jų lūkesčius. Kuriamos programinės įrangos našumo probleminių sričių nustatymo metu yra atliekamos šios pagrindinės veiklos:

- Kritinių ir nekritinių sistemos funkcijų išskyrimas. Šio etapo metu turi būti tiksliai išskirtos kuriamos sistemos funkcijos, kurios yra pagrindinės sistemos veikloje, o kurios yra šalutinės (atsiradusi klaida nekritinėje sistemos funkcijoje nesumažina sistemos kokybės lygio bei neapriboja vartotojų veiksmų).
- Tikėtino sistemos atsako laiko į vykdomas funkcijas nustatymas. Atliekant sistemos vartotojų lūkesčių analizę turi būti nustatyta kokių reakcijos laikų tikimasi iš sistemos vykdant kritines funkcijas (užklausų vykdymas, duomenų saugojimas, sąrašų formavimas, ataskaitų formavimas, duomenų perdavimas trečiosios šalies programinei įrangai).
- Sistemos maksimalaus apkrovimo ribų nustatymas. Atliekant reikalavimų analizę, sistemos vartotojai turi atsakyti į klausimus: *Koks yra galimas maksimalus vartotojų, dirbančių su sistema vienu metu, skaičius? Koks yra galimas maksimalus įrašų skaičius per nustatytą laiko tarpą? Koks yra galimas maksimalus perduodamų duomenų kiekis trečiosios šalies programinei įrangai?*[4] Remiantis gautais atsakymais, turi būti suformuotos kuriamos sistemos maksimalios apkrovos ribos.
- Sistemos našumo kriterijų, esant maksimaliam sistemos apkrovimui, nustatymas. Remiantis analizės metu surinkta informacija, turi būti nustatyti kuriamos sistemos našumo kriterijai (pvz. reakcijos laikas, kuris turi būti pasiektas vykdant sistemos kritines funkcijas vienu metu maksimaliam vartotojų skaičiui).

Remiantis surinkta informaciją bei nustatytais apribojimais, turi būti sukurti testavimo scenarijai, kurie padengtų visas sistemos problemines sritis (sistemos kritinės funkcijos, sistemos reakcijos laikas vykdant kritines funkcijas, sistemos reakciją imituojant maksimalią sistemos apkrovą).

2.3. Testavimo įrankių analizė ir palyginimas

Kalbant apie testavimo metodikas, yra labai svarbu apžvelgti testavimo įrankius ir įvertinti jų teigiamas bei neigiamas savybes. Testavimo įrankių analizei yra pasirinkti du įrankiai: „*Microsoft Visual Studio for Testers*“ [8] ir „*QaTrack*“ [9]. Šie įrankiai testavimo įrankių analizei buvo pasirinkti dėl funkcijų gausos, daugelio testavimo metodikų pritaikymo atliekant sistemų testavimus, bei dėl paplitimo.

Pirmasis įrankis („*Microsoft Visual Studio for Testers*“) yra „*Microsoft*“ kompanijos produktas ir priklauso programinės įrangos kūrimo paketui „*Visual Studio*“. Šis produktas turi integracijas su kitais to paties paketo produktais, todėl visas programinės įrangos kūrimas (nuo analizės, projektavimo, programavimo iki testavimo) gali būti valdomas vieno paketo produktais.

Testavimo įrankis „*QaTrack*“ vartotojams suteikia tvirtą pagrindą kuriant naujus bei redaguojant jau esamus testavimo atvejus. Šis įrankis yra orientuotas į internetinių sistemų testavimą. Pateiktas testavimo atvejų valdymo sprendimas yra patogus vartotojui, rezultatai pasiekiami suinteresuotiems asmenims. „*QaTrack*“ pasižymi galimybe formuoti grafines ataskaitas. Šios ataskaitos padengia viską nuo testų sukūrimo iki vykdymo.

Testavimo sistemų analizės metu buvo vertinamos šios savybės:

- Vartotojo identifikavimas – šis kriterijus yra svarbus siekiant užtikrinti saugų darbą komandoje bei įmonės viduje. Siekiant, kad kuriama programinė įranga atitiktų įmonėje įdiegtus kokybės standartus svarbu užtikrinti, kad programinė įranga būtų kuriama saugiomis priemonėmis.
- Testavimo atvejų kūrimas ir valdymas – tai yra pagrindinis testavimo sistemos kriterijus. Yra svarbu nustatyti, kaip realizuotas testavimo atvejų kūrimas ir valdymas. Testavimo atvejų valdymas turi apimti ne tik jo sukūrimą, redagavimą, susiejimą su reprezentuojamais projektais, bet testavimo valdymas turi apimti ir testavimo atvejų valdymą, susiejimą su atsakingais testuotojais, su testavimo uždaviniais.
- Testavimo atvejo išplėtimas – šis testavimo įrankių vertinimo kriterijus turi įvertinti testavimo įrankio turimą galimybę išplėsti sukurtą testavimo atvejį papildoma informacija.
- Testavimo statistika – kaip ir visame projekto vykdyme, taip ir testavimo etape yra svarbu turėti galimybę peržiūrėti projekto testavimo statistiką. Testavimo statistikos peržiūra įgalina lengviau identifikuoti projekto testavimo progresą.

- Projektų statistika – kitas svarbus testavimo įrankių vertinimo kriterijus. Testavimo įrankiams yra svarbu turėti galimybę peržiūrėti bei palyginti skirtingų projektų testavimo procedūras. Toks funkcionalumas leidžia projektų vadovams tiksliau įvertinti reikalingą testavimo laiką naujiems projektams.
- Testavimo rezultatų ataskaitos – kaip ir testavimo statistikos peržiūra, taip ir galimybę formuoti testavimo rezultatų ataskaitas yra svarbus funkcionalumas testavimo įrankiuose. Testavimo rezultatų ataskaitų formavimas suteikia galimybę pristatyti testavimo rezultatus, testavimo rezultatų statistiką ne tik vidiniams projekto komandos nariams, bet ir programinės įrangos užsakovams.
- Testavimo sistema pasiekama interneto prieiga – testavimo įrankio kriterijus, kuris leidžia įvertinti ar testavimo įrankis yra laisvai prieinamas programinės įrangos kūrimo komandai. Testavimo įrankio galimybė būti pasiekiamam interneto ryšio pagalba sumažina jo diegimo bei naudojimo įmonėje kaštus;
- Sistemos našumo testavimas – svarbus kriterijus siekiant išskirti pagrindines kuriamos sistemos našumo problemas. Sistemos našumo testavimas padeda nustatyti kritines sistemos vietas, kurios gali būti pažeidžiamos dėl didelio duomenų kiekio ar lygiagretaus daugelio vartotojų darbo su sistema.
- Duomenų bazių testavimas – galimybė testuoti duomenų bazes taip pat yra svarbus kriterijus vertinant bendras testavimo įrankio galimybes. Kaip ir sistemos testavimui taip ir duomenų bazių testavimui yra svarbu turėti galimybę sudaryti vienetų testus, kurie leistų patikrinti duomenų bazėse realizuotas procedūras ar funkcijas.
- Duomenų bazių našumo testavimas – kaip ir sistemos našumo testavimas, taip ir duomenų bazių našumo testavimas yra svarbus kriterijus. Yra svarbu įvertinti duomenų bazėse realizuotas procedūras ar funkcijas kurios gali būti kritinės esant didelei duomenų apkrovai.

Testavimo įrankiai buvo vertinami pagal anksčiau išskirtus ir aprašytus vertinimo kriterijus. Kiekvienas kriterijus yra vertinamas balais nuo 1 iki 3. Balo reikšmė nustatoma pagal:

- 1 – kriterijus visiškai neįgyvendintas vertinamame testavimo įrankyje.
- 2 – siekiant, kad vertinimo kriterijus būtų įvertintas reikalinga atlikti vertinamo testavimo įrankio praplėtimą;
- 3 – testavimo įrankio vertinimo kriterijus yra pilnai įgyvendintas.

2.2 lentelė. Lyginamoji įrankių analizė

Kriterijus	„Microsoft Visual Studio For Testers“	„QaTraQ“	Komentariai
Vartotojo identifikavimas	3	3	-
Testavimo atvejų kūrimas ir valdymas	3	3	-
Testavimo atvejų išplėtimas	3	3	-
Testavimo statistika	2	3	Siekiant peržiūrėti testavimo statistiką atliekant sistemos testavimą su „Microsoft Visual Studio for Studio“ yra reikalinga papildomų ataskaitų programavimas
Projektų statistika	2	1	Siekiant peržiūrėti projektų statistiką atliekant sistemos testavimą su „Microsoft Visual Studio for Studio“ yra reikalinga papildomų ataskaitų programavimas. Atliekant sistemos testavimą su „QaTrack“ šis funkcionalumas negalimas
Testavimo rezultatų ataskaitos	2	3	Siekiant peržiūrėti testavimo rezultatų ataskaitas atliekant sistemos testavimą su „Microsoft Visual Studio for Studio“ yra reikalinga papildomų ataskaitų programavimas
Pasiekiamas interneto prieigos pagalba	1	3	Testavimo įrankis „Microsoft Visual Studio for Studio“ neturi realizuotos šios galimybės
Sistemos našumo testavimas	3	1	Testavimo įrankis „QaTrack“ neturi realizuotos šios galimybės
Duomenų bazių testavimas	3	1	Testavimo įrankis „QaTrack“ neturi realizuotos šios galimybės
Duomenų bazių našumo testavimas	1	1	Įrankiai „Microsoft Visual Studio for Testers“ ir „QaTrack“ neturi realizuotos šios galimybės

Pagal atliktą programinės įrangos testavimo įrankių analizę yra pastebima, kad pagrindiniai testavimo įrankių trūkumai yra mažos galimybės atlikti duomenų bazių testavimą bei atlikti testuojamų duomenų bazių našumo analizę. Pagal pateiktas programinės įrangos testavimo metodikas bei pateiktus testavimo metodikų taikymo sprendimus galima spręsti, kad duomenų bazių testavimą galima atlikti remiantis tomis pačiomis metodikomis ir jų taikymo strategijomis.

2.4. Duomenų bazių našumo tyrimo įrankio kūrimo prielaidos

Šiame skyriuje yra pateikiama kuriamo įrankio, kuriamo įrankio architektūros, galimų kūrimo technologijų analizė. Taip pat pateikiamas suformuluotas duomenų bazių našumo tyrimo įrankio funkcijos be jų aprašymai.

2.4.1. Sistemos vartotojų analizė

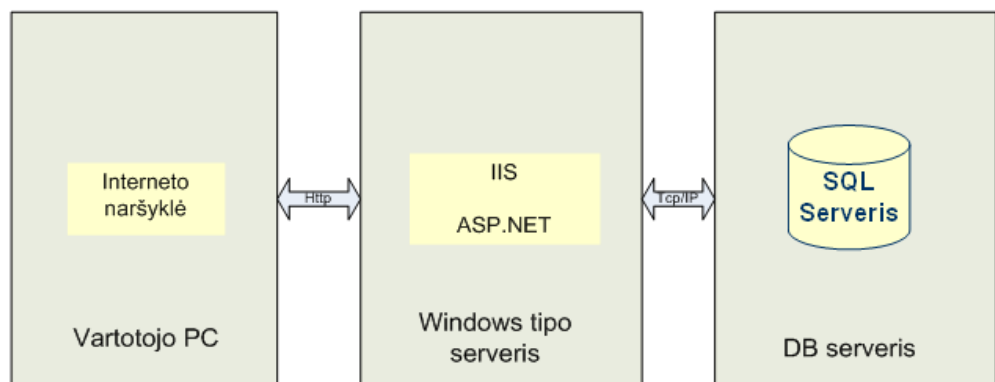
Duomenų bazių našumo tyrimo įrankio vartotojų aibę sudaro dvi vartotojų grupės:

- Administratorių grupė;
- Testuotojų grupė.

Kuriamas įrankis turi būti suskaidytas į atskiras posistemes pagal vartotojų grupes (administratoriaus posistemė ir testuotojo posistemė) Įrankio techniniai (prisijungimas prie nutolusių sistemų) bei sisteminiai (projektų, vartotojų kūrimas, teisių suteikimas) nustatymai turi būti atliekami administratoriaus posistemėje. Testuotojo posistemėje turi būti realizuotos duomenų bazių našumo tyrimo funkcijos.

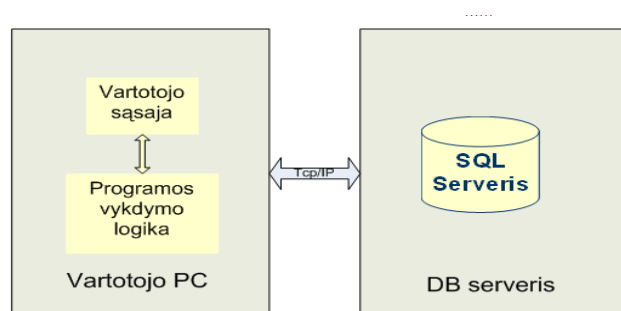
2.4.2. Architektūros ir įgyvendinimo priemonių analizė

Pagal pateiktus testavimo įrankių lyginamosios analizės rezultatus, duomenų bazių našumo tyrimo įrankis gali būti realizuojamas pagal dviejų tipų architektūras. Viena iš šių architektūrų yra internetinės taikomosios programos architektūra. Pasirinkus realizuoti įrankį kaip internetinę taikomąją programą, šios realizacijos architektūra turėtų būti trijų lygių pagal (2.3) pateikiamą schemą.



2.3 pav. Interneto taikomosios programos realizacinės architektūros schema

Kita duomenų bazių našumo tyrimo įrankio realizavimo architektūra gali būti „Windows“ tipo programa. Pasirinkus tokio tipo architektūrą, kuriama sistema būtų realizuojama pagal pavyzdyje (2.4) pateikiamą schemą.



2.4 pav. „Windows“ tipo programos realizacinė schema

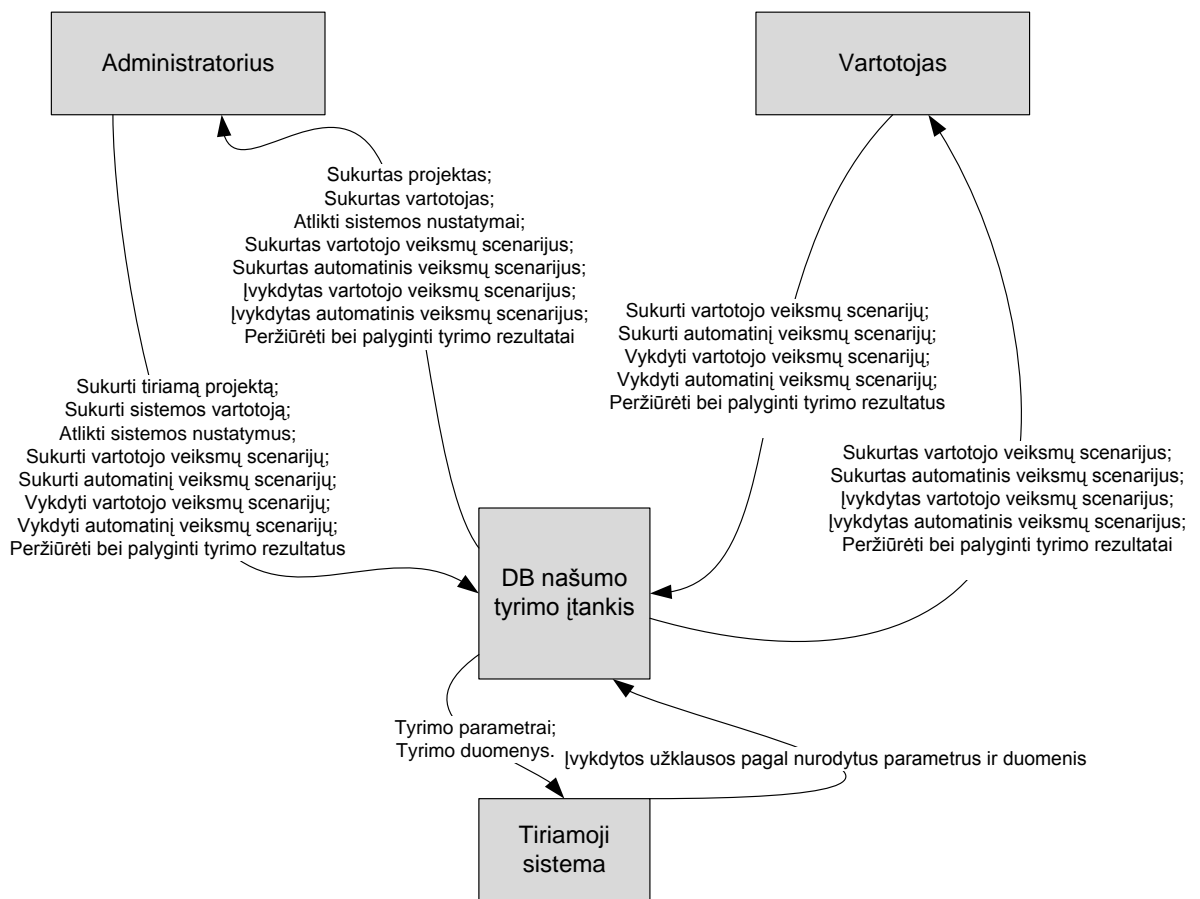
Kitas svarbus kuriamo įrankio aspektas - įrankio realizavimo technologija. Duomenų bazių našumo tyrimo įrankiui realizuoti galimų technologijų analizei yra pateikiama .NET technologija ir PHP programavimo kalba. .NET technologija priešingai nei visų internetinių sistemų (tame tarpe ir internetinių svetainių), kurių programavime naudojamos priemonės (JAVA, PHP, ColdFusion ir kita) yra tik programavimo kalbos, nuo pat pradžių buvo sumąstyta ir vystoma kaip pažangi programavimo technologija, orientuota į šiuolaikinio korporatyvinio verslo poreikius ir tų poreikių greitą augimą. .NET sukurta kaip nepriklausoma nuo programavimo kalbos technologija, todėl ja paremtus sprendimus, atsižvelgiant į jų specifiką, galima kurti naudojant įvairias programavimo kalbas: ASP.NET, C#, C++, VB, Delphi, netgi JAVA (J#) galima naudoti kuriant .NET sprendimus. Microsoft .NET užtikrina sprendimų, sukurtų skirtingomis programinėmis priemonėmis .NET terpėje, suderinamumą tarpusavyje.

Kaip pagrindinius .NET technologijos ir technologinių priemonių esminius privalumus galima paminėti [20]:

- Pilna, greita ir patogi integracija ir suderinamumas su visais Microsoft produktais.
- .NET yra daug saugiau. Šios technologijos saugumo politika paremti sprendimai yra gerokai saugesni vien dėl to, kad jie naudoja bendrą integruotą saugumo užtikrinimą.
- .NET technologija yra ir bus vystoma bei palaikoma Microsoft kompanijos, todėl šia technologija remiantis bus galima kurti naujus sprendimus integruojant juos į veikiančias, seniau sukurtas sistemas nekuriant jų iš naujo.
- .NET technologija paremtuose sprendimuose naudojant ASP.NET ir C# programavimo kalbą visas programinis kodas sistemoje yra išskirtas į du lygius: C# kalba parašytas biznio logikos/skaičiavimų lygis ir ASP atvaizdavimo lygis, jie yra vienas nuo kito logiškai atskirti. Be to, naudojamas objektinis programavimas. Norint sistemą praplėsti tam tikru papildomu funkcionalumu, integruoti tam tikrus komponentus iš kitos sistemos, surasti ir pakoreguoti problemines kodo vietas yra daug kartų greičiau ir pigiau nei esant vienalyčiam kodui kaip PHP ar ColdFusion. Daugeliu atvejų PHP sprendimuose naujo funkcionalumo pridėjimas ar didesnė integracija su kitais sprendimais susiveda į viso sprendimo ar didžiosios dalies perprogramavimą, o tai atima gerokai daugiau laiko, tuo pačiu ir išlaidų.

2.4.3. Siekiamos sistemos apibrėžimas

Kuriama sistema sąveikauja su sistemos vartotojais ir tiriamąja sistema. Sistemos ribos pateikiamos 2.5 paveikslėlyje



2.5 pav. Sistemos kontekstinė diagrama

2.4.4. Darbo tikslai

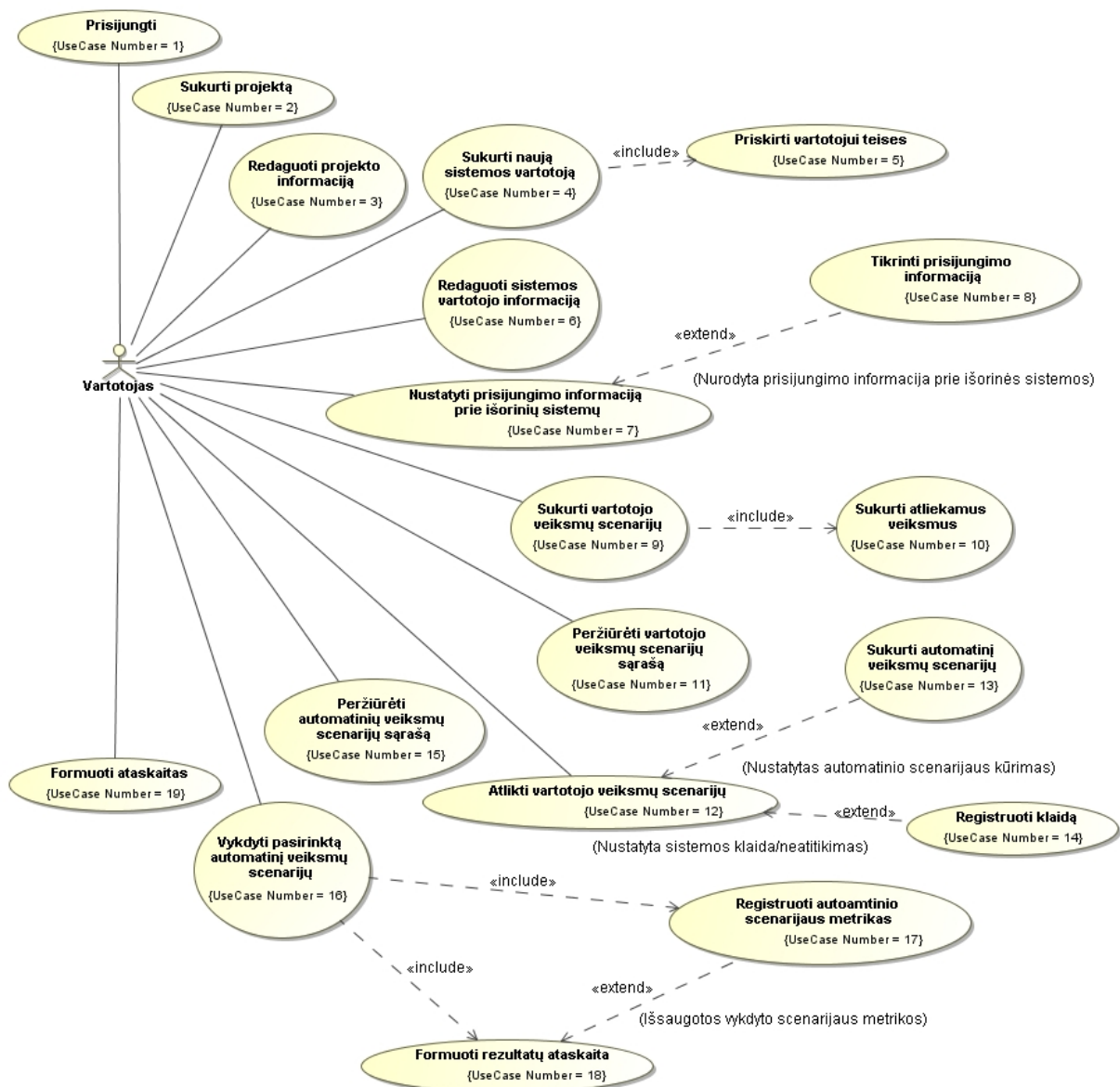
Šio darbo **tikslas** yra palengvinti duomenų bazių kūrėjų veiklą, sukuriant įrankį, kuris padėtų nustatyti duomenų bazių našumo problemines vietas naudojantis kompiuterizuotu duomenų bazių našumo tyrimo įrankiu. Tai reiškia, kad vartotojui reikia pateikti metodiką bei sukurti sistemą, kuri leistų paprasčiau (vieno įrankio pagalba) nustatyti tiriamos sistemos duomenų bazėje vykdomų užklausų našumo problemas sukeliančias vietas.

Darbo uždaviniai:

- Nustatyti teisingą duomenų bazių našumo problemų tyrimo metodikos koncepciją;
- Nustatyti reikiamą įrankio funkcionalumą, kokybines charakteristikas, pasirinkti tinkamas realizavimo priemones;
- Suprojektuoti įrankį pagal apibrėžtus reikalavimus;
- Realizuoti ir ištestuoti įvertinimo įrankį;
- Atlikti bandomąjį duomenų bazės našumo tyrimą;

Nustatyti įrankio kokybines charakteristikas ir tinkamumą.

2.4.5. Kompiuterizuojamos sistemos funkcijos



2.6 pav. Kompiuterizuojami duomenų bazių našumo įrankio panaudojimo atvejai

2.5. Analizės išvados

1. Literatūros šaltinių analizė parodė, kad pastaruoju metu sustiprėjo tyrimai duomenų bazių testavimo srityje siekiant atlikti duomenų bazės testavimą kuo žemesniame lygyje.
2. Galimų duomenų bazių testavimo analizės rezultate paaiškėjo, kad duomenų bazių testavimas yra skiriamas į dvi kryptis: užklausų testavimą ir duomenų bazių našumo testavimą.
3. Atlikta sistemų našumo testavimo metodų analizė parodė, kad duomenų bazių našumo tyrimui yra naudojamos tos pačios metodikos kaip ir programinės įrangos našumo testavimui.

4. Tiriamos sistemos duomenų bazės našumo vertinimas padėtų objektyviai atrasti duomenų bazės vietas, kurios ateities perspektyvoje gali įtakoti sistemos darbo sutrikimus ar duomenų praradimą.
5. Programinės įrangos testavimo įrankių analizė parodė, kad šiuo metu plačiai žinomų duomenų bazių našumo tyrimo įrankių nėra arba jie pasižymi panaudojimo apribojimais.
6. Galimų sistemos realizavimo variantų analizės metu nustatyta, kad tinkamiausias variantas yra interneto taikomosios programos (ASP.NET pagrindu), nereikalaujančios papildomos programinės įrangos vartotojo kompiuteryje.
7. Reikalavimų analizės metu nustatyta, kad pagrindiniai funkciniai reikalavimai – pasiekiamumas internetu, tiriamos sistemos našumo parametrų peržiūra įvairiais pjūviais.

3. Duomenų bazių našumo tyrimo metodika

Duomenų bazių našumo tyrimo įrankis yra kuriamas atsižvelgiant į duomenų bazių testavimo bei programinės įrangos našumo tyrimo principus. Pagrindinės duomenų bazių našumo problemos siejamos su duomenų pasirinkimo (*angl.* select), įvedimo (*angl.* insert) ir atnaujinimo (*angl.* update) užklausų vykdymo laikais esant dideliame apkrovimui. Šie trys užklausų tipai išskiriami todėl, kad būtent šių užklausų vykdymo laikai tiesiogiai įtakoja vartotoją ir jo darbo laiką [16].

Prieš atliekant tiriamos sistemos duomenų bazės našumo tyrimą yra svarbu sukurti bazinius veiksmų scenarijus bei nustatyti pagrindinius tyrimo parametrus (vartotojų skaičių bei iteracijų skaičių).

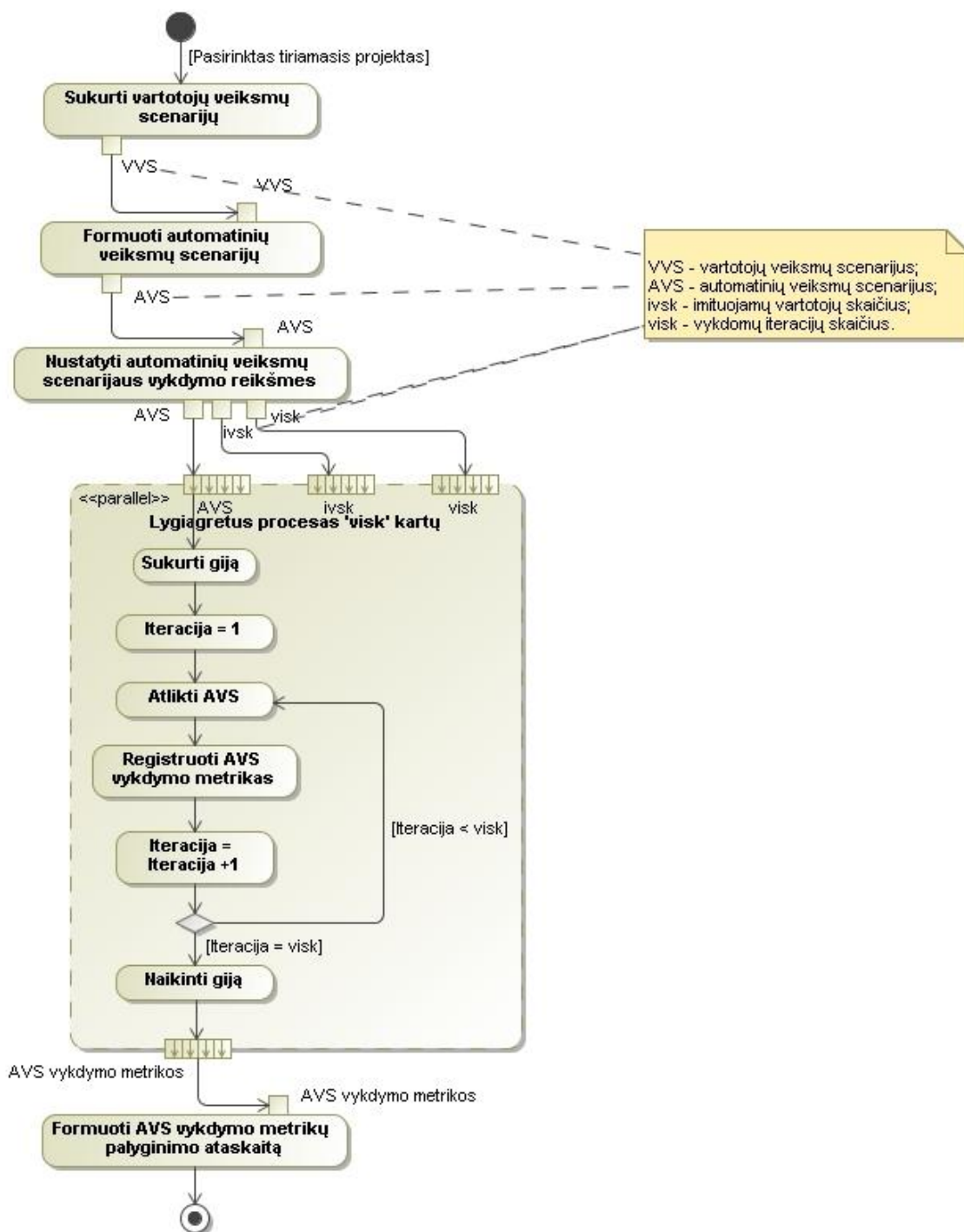
Siekiant padengti visą tiriamos sistemos funkcionalumą, yra reikalinga sukurti kuo platesnius bazinius vartotojo veiksmų scenarijus, kurie turi padengti kuo didesnę tiriamos sistemos funkcionalumo dalį. Sukurtų bazinių vartotojo veiksmų scenarijų pagrindu turi būti suformuojami automatinių veiksmų scenarijai, kurių pagrindinis tikslas yra vartotojo atliekamų veiksmų imitavimas. Šiam tikslui pasiekti turi būti sekamos vykdomos procedūros bei pagal šiuos stebėjimo rezultatus turi būti formuojamas automatinių veiksmų scenarijus.

Kelių vartotojų darbas turi būti imituojamas lygiagrečiai vykdant automatinių veiksmų scenarijų darbą. Kiekvieno automatinių veiksmų scenarijaus vykdymo metu turi būti registruojami scenarijų sudarančių procedūrų bei užklausų vykdymo parametrai. Kaip pagrindinius stebėjimo parametrus galima išskirti kelis:

- CPU – ši reikšmė nurodo CPU naudojimo užklausiai įvykdyti laikas milisekundėmis;
- Duration – ši reikšmė nurodo tikrąjį SQL paketo vykdymo trukmę. SQL Server 2005 šis laikas yra pateikiamas mikrosekundėmis;
- Reads – ši reikšmė nurodo loginių disko skaitymų skaičių 8KB puslapiuose. Pavyzdžiui, jei šio stulpelio reikšmė yra 85, tokiu atveju reiškia, kad SQL Server nuskaitytė 5570560 baitų duomenų. Fizinį skaitymų skaičius paprastai būna daug mažesnis negu loginių skaitymų. Nustatyti tikslų fizinį skaitymų skaičių nėra galimybių, todėl yra vadovaujamasi mintimi, kad jeigu yra sumažinamas loginių nuskaitymų skaičius, tokiu atveju sumažėja ir fizinį nuskaitymų skaičius. Loginiai nuskaitymai padidina atminties naudojimą, dėl šios priežasties užklausų vykdymas gali sulėtėti.

- Writes – ši reikšmė nurodo loginių įrašų skaičių 8KB puslapyje. Loginių įrašų skaičius yra daug artimesnis skaičius fizinių įrašų skaičiui.

Vykdam automatinių veiksmų scenarijus yra svarbu užtikrinti patikimą duomenų pateikimą. Kiekvienam scenarijui duomenys yra generuojami atsižvelgiant į naudotus duomenys vykdam vartotojų veiksmų scenarijus. Duomenų našumo tyrimo metodo vizualus vaizdas pateikiamas 3.1 paveikslėlyje.



3.1 pav. Duomenų bazės našumo tyrimo procesas

Siekiant užtikrinti platesnį duomenų spektrą našumo tyrimui yra patartina registruoti taip pat tokius kriterijus kaip:

- Execution Warnings (*liet.* vykdymo įspėjimai) – kriterijus, kuris parodo, kad yra tikėtina galimybė, kad labai apkrautiems serveriams reikės truputi palaukti, kol atsiras laisvų resursų atlikti numatytoms užklausoms. Šis kriterijus stebi ar įvykdyti numatytas užklausas serveris turi laukti atsilaisvinančių resursų ir kiek ilgai jis turi laukti. Kriterijus gali būti dviejų tipų: „Query Wait“ ir „Query Time-Out“. Kriterijui nurodžius pirmąjį tipą, yra stebima kaip dažnai užklausa turi laukti atsilaisvinančių resursų. Kriterijui nurodžius antrąjį tipą, yra stebima kaip dažnai užklausa laukia atsilaisvinančių resursų iki paskutinės laiko ribos (*angl.* time-out).
- Hash Warning (*liet.* maišos įspėjimas) – šis kriterijus yra naudojamas nustatyti maišos rekursiją. Maišos rekursija (event 0) nutinka kai užklaustos įėjimo duomenis nepakanka atminties, tokiu atveju SQL Serveris yra verčiamas skaidyti įėjimo duomenis į dalis, kurios vykdomos atskirai. Šis reiškinys gali ženkliai sulėtinti užklausių vykdymą.
- Missing Column statistics (*liet.* nerastų stulpelių statistika) – šis kriterijus parodo, kurie užklaustos statistiniai stulpeliai yra nerasti. Neradus dalies užklaustos statistinių stulpelių, tokiai užklausiai nebegali būti taikomas geriausias jos vykdymo planas, kuris nustatomas naudojant „Query Optimizer“. Dėl šios priežasties užklausių vykdymo laikas padidėja.
- Sort Warnings (*liet.* rikiavimo įspėjimai) – šis kriterijus parodo ar duomenų rikiavimo operacija galės būti pilnai atlikta atmintyje, ar turės būti skaidoma į atskiras dalis ir atliekama atskirai. Bet koks operacijų skaidymas į dalis ir jų vykdymas atskirai prailgina operacijos vykdymo laiką.

4. Reikalavimai tyrimo įrankiui sukurti

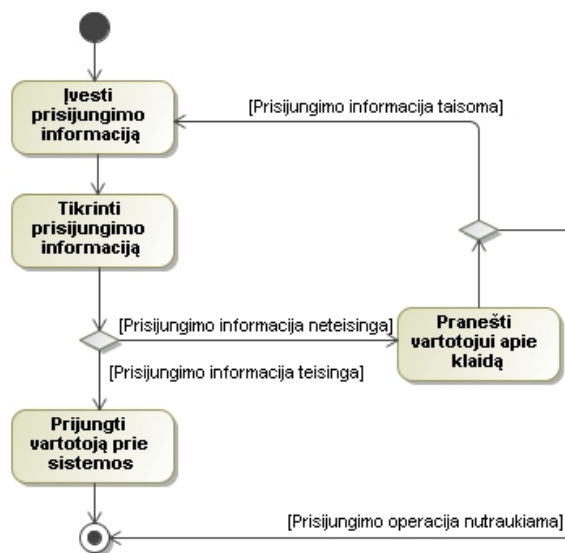
Tyrimo įrankio funkcijos yra pateiktos 2.6 paveikslėlyje pavaizduotuose sistemos panaudojimo atvejuose. Šiame skyriuje yra pateikiamos panaudojimo atvejų specifikacijos, įrankiui keliami nefunkciniai reikalavimai ir apribojimai. Taip pat pateikiama visa reikalavimų analizės informacija.

4.1. Duomenų bazių našumo tyrimo įrankio funkcinių reikalavimų specifikacija

Šiame skyriuje yra pateikiamos kompiuterizuojamų panaudojimo atvejų specifikacijos (4.1 - 4.17) ir panaudojimo atvejų veiklos diagramos (4.1 - 4.17).

4.1 lentelė. Panaudojimo atvejis „Prisijungti“

PA: Prisijungti		
Prieš sąlyga:	Sistemoje yra sukurtas vartotojas. Vartotojui suteiktos prisijungimo teisės	
Sužadinimo sąlyga:	Vartotojas nori prisijungti prie sistemos.	
Susiję PA	Apima PA:	Nėra
	Išplečia PA:	Nėra
	Specializuoja PA:	Nėra
Pagrindinis įvykių srautas:	1. Atidaromas sistemos prisijungimo langas; 2. Įvedama vartotojo prisijungimo informacija;	
Po sąlyga:	Vartotojas sėkmingai prisijungia prie sistemos. Prisijungusiam vartotojui sistemos funkcijos pasiekiamos pagal jam suteiktas teises.	
Pastabos:	Nėra	
Alternatyvus scenarijus:	Nėra	

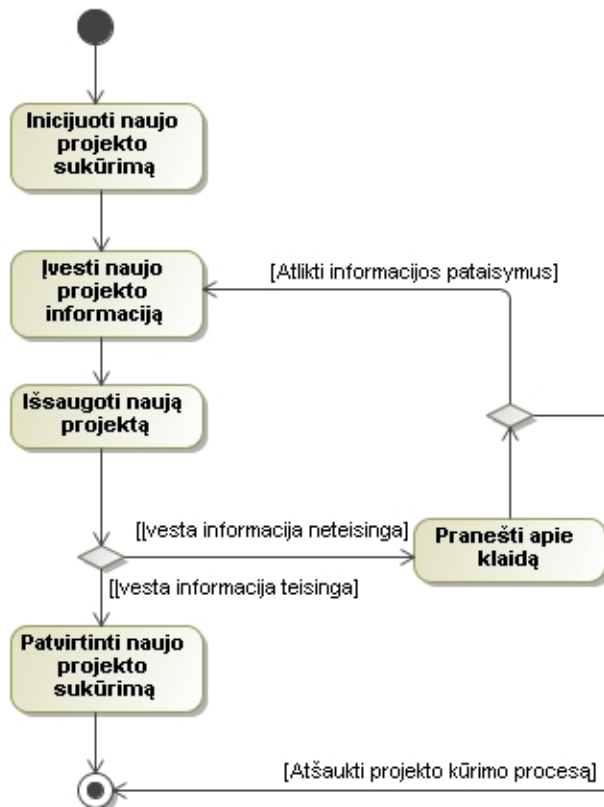


4.1 pav. PA „Prisijungti“ veiklos diagrama

4.2 lentelė. Panaudojimo atvejis „Sukurti projektą“

PA: Sukurti projektą	
Prieš sąlyga:	Vartotojas prisijungęs prie sistemos. Prisijungęs vartotojas turi administratoriaus teises.
Sužadinimo sąlyga:	Vartotojas nori sukurti naują projektą.

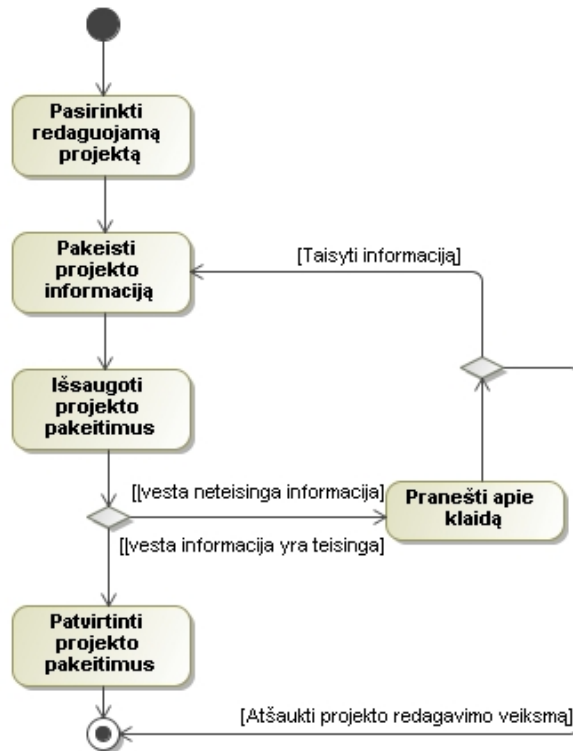
Susiję PA	Apima PA:	Nėra
	Išplečia PA:	Nėra
	Specializuoja PA:	Nėra
Pagrindinis įvykių srautas:	1. Atidaromas projektų sąrašas; 2. Inicijuojamas naujo projekto sukūrimas; 3. Įvedama naujo projekto informacija;	
Po sąlyga:	Vartotojas sėkmingai sistemoje sukuria naują projektą	
Pastabos:	Nėra	
Alternatyvus scenarijus:	Nėra	



4.2 pav. PA „sukurti projektą“

4.3 lentelė. Panaudojimo atvejis „Redaguoti projekto informaciją“

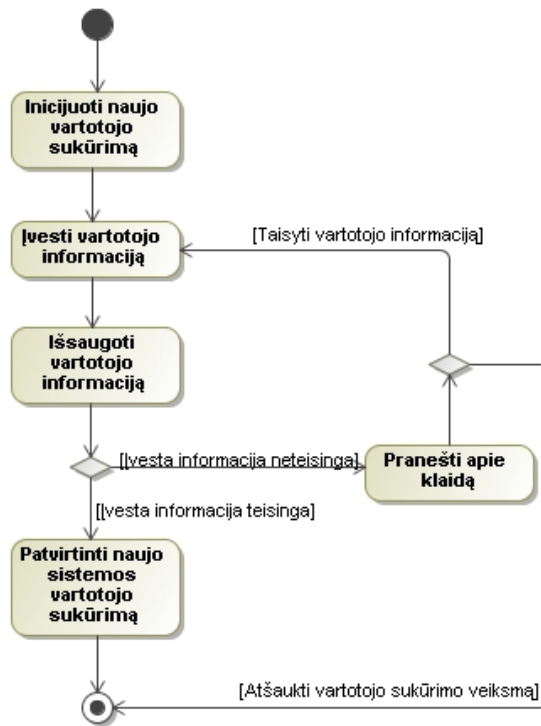
PA: Redaguoti projekto informaciją		
Prieš sąlyga:	Vartotojas prisijungęs prie sistemos. Prisijungęs vartotojas turi administratoriaus teises.	
Sužadinimo sąlyga:	Vartotojas nori atlikti projekto informacijos pakeitimus.	
Susiję PA	Apima PA:	Nėra
	Išplečia PA:	Nėra
	Specializuoja PA:	Nėra
Pagrindinis įvykių srautas:	1. Vartotojas atidaro pasirinkto projekto redagavimo formą; 2. Vartotojas atlieka projekto informacijos redagavimą; 3. Vartotojas išsaugo atliktus projekto informacijos pakeitimus sistemoje;	
Po sąlyga:	Sistemoje išsaugoti pasirinkto projekto pakeitimai.	
Pastabos:	Nėra	
Alternatyvus scenarijus:	Nėra	



4.3 pav. PA „Redaguoti projekto informaciją“ veiklos diagrama

4.4 lentelė. Panaudojimo atvejis „sukurti sistemos vartotoją“

PA: Sukurti naują sistemos vartotoją		
Prieš sąlyga:	Vartotojas prisijungęs prie sistemos. Prisijungęs vartotojas turi administratoriaus teises	
Sužadinimo sąlyga:	Vartotojas nori sukurti naują vartotoją.	
Susiję PA	Apima PA:	• Priskirti vartotojui teises
	Išplečia PA:	Nėra
	Specializuoja PA:	Nėra
Pagrindinis įvykių srautas:	1. Atidaro naujo vartotojo sukūrimo formą; 2. Įveda naujo vartotojo informaciją; 3. Išsaugo naujo vartotojo informaciją sistemoje;	
Po sąlyga:	Sistemoje išsaugotas naujas vartotojas.	
Pastabos:	Nėra	
Alternatyvus scenarijus:	Nėra	



4.4 pav. PA „Sukurti sistemos vartotoją“ veiklos diagrama

4.5 lentelė. Panaudojimo atvejis „Priskirti vartotojui teises“

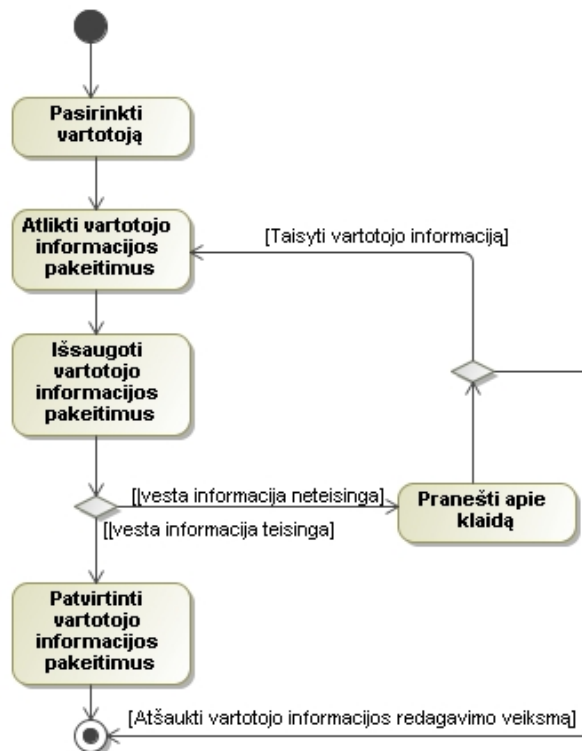
PA: Priskirti vartotojui teises		
Prieš sąlyga:	Vartotojas prisijungęs prie sistemos. Prisijungęs vartotojas turi administratoriaus teises.	
Sužadinimo sąlyga:	Vartotojas nori suteikti teises pasirinktam sistemos vartotojui.	
Susiję PA	Apima PA:	Nėra
	Išplečia PA:	Nėra
	Specializuoja PA:	Nėra
Pagrindinis įvykių srautas:	<ol style="list-style-type: none"> 1. Pasirenka vartotoją, kuriam nori suteikti teises; 2. Pasirinkto vartotojo informacijos redagavimo režime pasirenka vartotojui priskiriamas teises; 3. Išsaugomi pasirinkto vartotojo pakeitimai; 	
Po sąlyga:	Pasirinktam sistemos vartotojui suteiktos teisės.	
Pastabos:	Nėra	
Alternatyvus scenarijus:	Nėra	



4.5 pav. PA „Priskirti vartotojui teises“ veiklos diagrama

4.6 lentelė. Panaudojimo atvejis „Redaguoti sistemos vartotojo informaciją“

PA: Redaguoti sistemos vartotojo informaciją		
Prieš sąlyga:	Vartotojas prisijungęs prie sistemos. Prisijungęs vartotojas turi administratoriaus teises.	
Sužadinimo sąlyga:	Vartotojas nori atlikti pasirinkto vartotojo informacijos redagavimą.	
Susiję PA	Apima PA:	Nėra
	Išplečia PA:	Nėra
	Specializuoja PA:	Nėra
Pagrindinis įvykių srautas:	1. Vartotojas iš vartotojų sąrašo pasirenką vartotoją, kurio informaciją nori redaguoti; 2. Pasirinkto vartotojo informacijos redagavimo lange atlieka pakeitimus; 3. Atliktus pasirinkto vartotojo informacijos pakeitimus išsaugo sistemoje;	
Po sąlyga:	Pasirinkto vartotojo informacijos pakeitimai išsaugoti sistemoje.	
Pastabos:	Nėra	
Alternatyvus scenarijus:	Nėra	

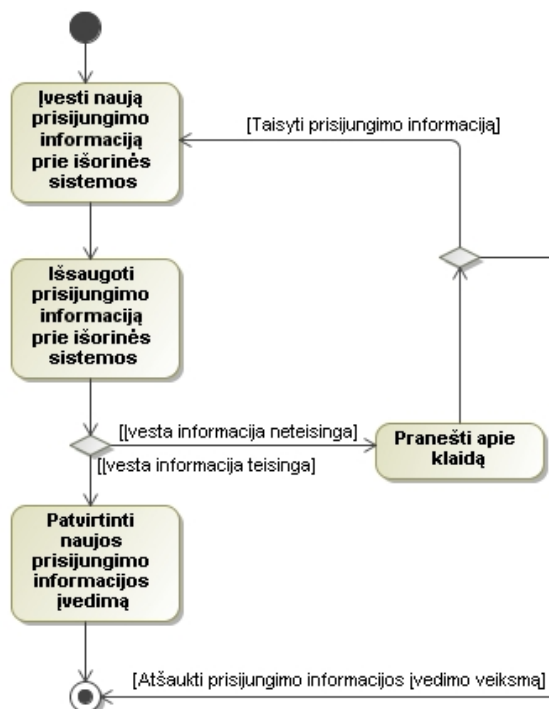


4.6 pav. PA „Redaguoti sistemos vartotojo informaciją“ veiklos diagrama

4.7 lentelė. Panaudojimo atvejis „Nustatyti prisijungimo informaciją prie išorinių sistemų“

PA: Nustatyti prisijungimo informaciją prie išorinių sistemų		
Prieš sąlyga:	Vartotojas prisijungęs prie sistemos. Prisijungęs vartotojas turi administratoriaus teises.	
Sužadinimo sąlyga:	Vartotojas nori įvesti naują prisijungimo prie išorinės sistemos informaciją.	
Susiję PA	Apima PA:	Nėra
	Išplečia PA:	Nėra
	Specializuoja PA:	Nėra
Pagrindinis įvykių srautas:	1. Vartotojas inicijuoja naujos prisijungimo prie išorinės sistemos informacijos įvedimą; 2. Įvedama nauja prisijungimo informacija; 3. Įvesta prisijungimo informacija išsaugoma sistemoje;	
Po sąlyga:	Prisijungimo informacija prie išorinės sistemos išsaugota sistemoje.	

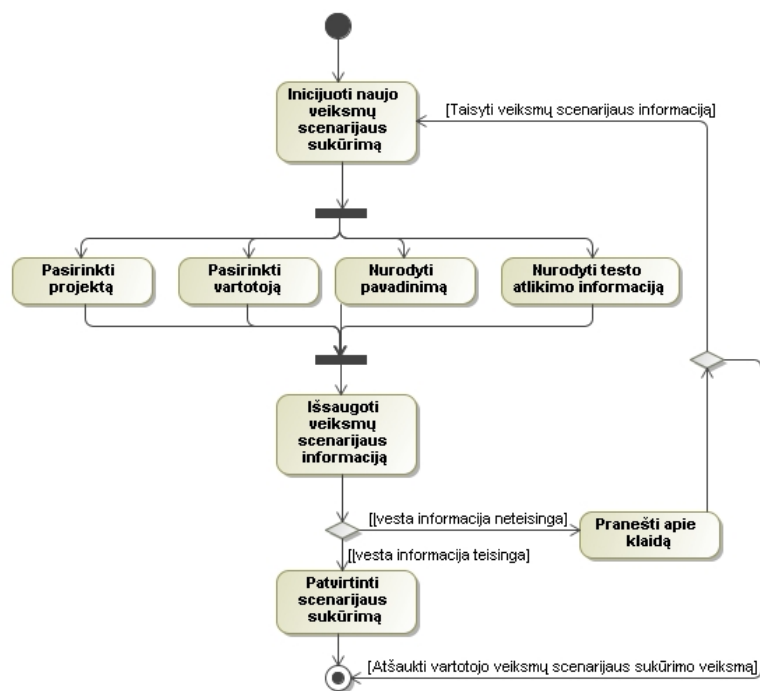
Pastabos:	Nėra
Alternatyvus scenarijus:	Nėra



4.7 pav. PA „Nustatyti prisijungimo informaciją prie išorinių sistemų“ veiklos diagrama

4.8 lentelė. Panaudojimo atvejis „Sukurti vartotojo veiksmų scenarijų“

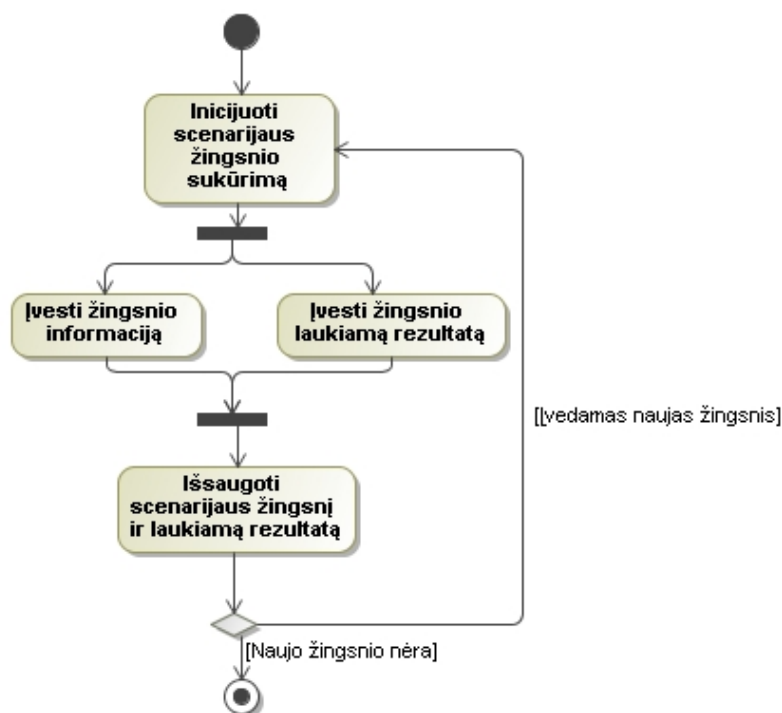
PA: Sukurti vartotojo veiksmų scenarijų		
Prieš sąlyga:	Vartotojas prisijungęs prie sistemos. Prisijungęs vartotojas turi administratoriaus arba testuotojo teises. Sistemoje yra sukurtas projektas.	
Sužadinimo sąlyga:	Vartotojas nori sukurti naują vartotojo vykdomą testavimo atvejį.	
Susiję PA	Apima PA:	• Sukurti atliekamus veiksmus
	Išplečia PA:	Nėra
	Specializuoja PA:	Nėra
Pagrindinis įvykių srautas:	<ol style="list-style-type: none"> Vartotojas inicijuoja naujo vartotojo vykdomo testavimo atvejo sukūrimą; Įvedama vartotojo vykdomo testavimo atvejo informacija: <ol style="list-style-type: none"> 2.1 Pasirenkamas projektas; 2.2 Pasirenkamas testuotojas; 2.3 Nurodomas testo pavadinimas; 2.4 Nurodomas testo atlikimo sąlygos; Vartotojo vykdomas testavimo atvejis išsaugomas sistemoje; 	
Po sąlyga:	Sukurtas ir sistemoje išsaugotas vartotojo vykdomas testavimo atvejis.	
Pastabos:	Nėra	
Alternatyvus scenarijus:	Nėra	



4.8 pav. PA „sukurti vartotojo veiksmų scenarijų“ veiklos diagrama

4.9 lentelė. Panaudojimo atvejis „Sukurti atliekamus veiksmus“

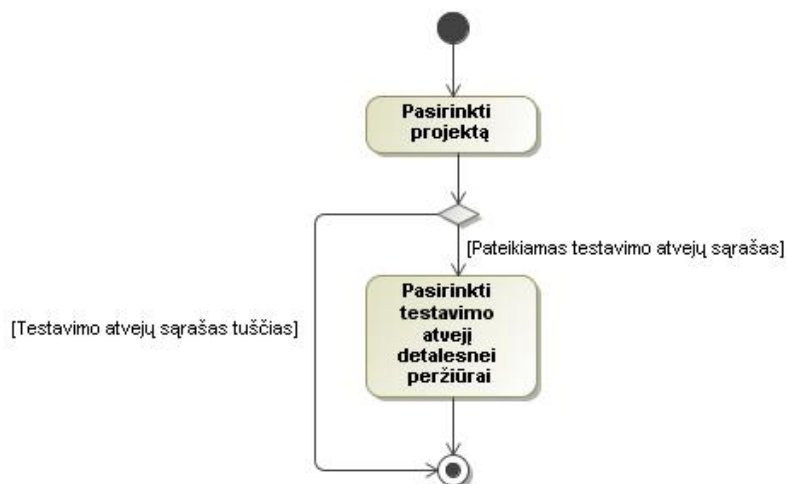
PA: Sukurti atliekamus veiksmus		
Prieš sąlyga:	Vartotojas yra prisijungęs prie sistemos. Prisijungęs vartotojas turi administratoriaus arba testuotojo tesies. Sistemoje yra sukurtas vartotojo vykdomas testavimo atvejis.	
Sužadinimo sąlyga:	Vartotojas nori sukurti veiksmų scenarijų pasirinktam vartotojo vykdomam testavimo atvejui.	
Susiję PA	Apima PA:	Nėra
	Išplečia PA:	Nėra
	Specializuoja PA:	Nėra
Pagrindinis įvykių srautas:	<ol style="list-style-type: none"> 1. Vartotojas pasirenka vartotojo vykdomą testavimo atvejį; 2. Vartotojo vykdomo testavimo atvejo informacijos redagavimo režime inicijuoja veiksmų scenarijaus sukūrimą; 3. Įvedama veiksmų scenarijaus informacija: <ol style="list-style-type: none"> 3.1 Nurodomas atlikti reikalingas veiksmas; 3.2 Nurodomas laukiamas veiksmo rezultatas; 4. Veiksmų scenarijus išsaugomas sistemoje; 	
Po sąlyga:	Pasirinktam testavimo atvejui yra sukurtas veiksmų scenarijus.	
Pastabos:	Nėra	
Alternatyvus scenarijus:	Nėra	



4.9 pav. PA „sukurti atliekamus veiksmus“ veiklos diagrama

4.10 lentelė. Panaudojimo atvejis „Peržiūrėti vartotojo veiksmų scenarijų sąrašą“

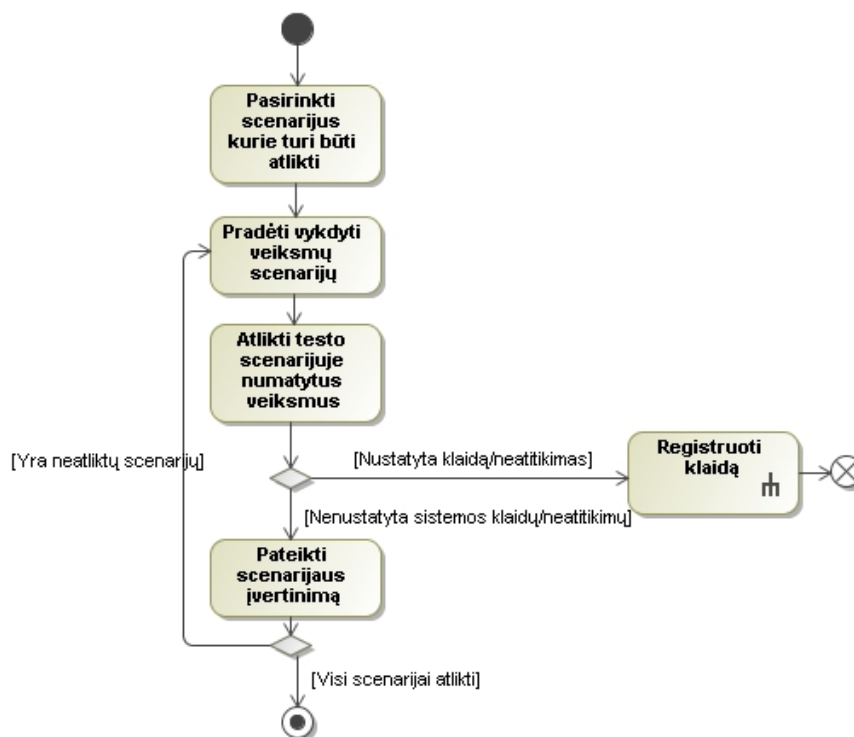
PA: Peržiūrėti vartotojo veiksmų scenarijų sąrašą		
Prieš sąlyga:		Vartotojas prisijungęs prie sistemos. Prisijungęs vartotojas turi administratoriaus arba testuotojo teises.
Sužadinimo sąlyga:		Vartotojas nori peržiūrėti sistemoje išsaugoti vartotojo vykdomų testavimo atvejų sąrašą.
Susiję PA	Apima PA:	Nėra
	Išplečia PA:	Nėra
	Specializuoja PA:	Nėra
Pagrindinis įvykių srautas:		1. Vartotojas pasirenką projektą, kurio vartotojų vykdomus testavimo atvejus nori peržiūrėti; 2. Peržiūri pateiktą testavimo atvejų sąrašą;
Po sąlyga:		Vartotojui pateikiamas sistemoje išsaugotų vartotojo vykdomų testavimo atvejų sąrašas.
Pastabos:		Nėra
Alternatyvus scenarijus:		Nėra



4.10 pav. PA „Peržiūrėti vartotojo veiksmų scenarijų sąrašą“ veiklos diagrama

4.11 lentelė. Panaudojimo atvejis „Atlikti vartotojo veiksmų scenarijų“

PA: Atlikti vartotojo veiksmų scenarijų		
Prieš sąlyga:	Vartotojas prisijungęs prie sistemos. Prisijungęs vartotojas turi administratoriaus arba testuotojo teises. Sistemoje išsaugotas vartotojo vykdomas testavimo atvejis	
Sužadinimo sąlyga:	Vartotojas nori atlikti pasirinktą vartotojo vykdomą testavimo atvejį.	
Susiję PA	Apima PA:	Nėra
	Išplečia PA:	Nėra
	Specializuoja PA:	Nėra
Pagrindinis įvykių srautas:	<ol style="list-style-type: none"> 1. Vartotojas pasirinktą vartotojo vykdomą testavimo atvejį iš sąrašo; 2. Inicijuoja pasirinkto testo pradžią; 3. Vartotojas testuojamoje sistemoje atlieka testavimo atvejo veiksmų scenarijuje nurodytus veiksmus; 4. Testuojamoje sistemoje atlikus visus nurodytus veiksmus, išsaugo vykdyto testavimo atvejo rezultata; 	
Po sąlyga:	Pasirinktas ir atliktas vartotojo vykdomas testavimo atvejis.	
Pastabos:	Nėra	
Alternatyvus scenarijus:	Nėra	



4.11 pav. PA „Atlikti vartotojo veiksmų scenarijų“ veiklos diagrama

4.12 lentelė. Panaudojimo atvejis „Sukurti automatinį veiksmų scenarijų“

PA: Sukurti automatinį veiksmų scenarijų		
Prieš sąlyga:	Vartotojas prisijungęs prie sistemos. Prisijungęs vartotojas turi administratoriaus arba testuotojo teises. Sistemoje išsaugotas vartotojo vykdomas testavimo atvejis.	
Sužadinimo sąlyga:	Vartotojas nori sukurti automatinį veiksmų scenarijų.	
Susiję PA	Apima PA:	Nėra
	Išplečia PA:	• Atlikti vartotojo veiksmų scenarijų
	Specializuoja PA:	Nėra
Pagrindinis įvykių srautas:	<ol style="list-style-type: none"> 1. Vartotojas pasirenką vartotojo vykdomą testavimo atvejį, kuriam nori sukurti automatinį veiksmų scenarijų; 2. Pasirenka automatinio veiksmų scenarijaus sukūrimo nustatymą; 3. Atlieka pasirinktą vartotojo vykdomą testavimo atvejį; 4. Išsaugo automatinį veiksmų scenarijų sistemoje; 	

Po sąlyga:	Sistemoje išsaugotas vartotojo vykdomo testavimo atvejo automatinių veiksmų scenarijus.
Pastabos:	Nėra
Alternatyvus scenarijus:	Nėra



4.12 pav. PA „Sukurti automatinių veiksmų scenarijų“ veiklos diagrama

4.13 lentelė. Panaudojimo atvejis „Registruoti klaidą“

PA: Registruoti klaidą		
Prieš sąlyga:	Vartotojas prisijungęs prie sistemos. Prisijungęs vartotojas turi administratoriaus arba testuotojo teises. Sistemoje išsaugotas vartotojo vykdomas testavimo atvejis.	
Sužadinimo sąlyga:	Nustatyta testuojamos sistemos klaida/neatitikimas.	
Susiję PA	Apima PA:	Nėra
	Išplečia PA:	<ul style="list-style-type: none"> Atlikti vartotojo veiksmų scenarijų
	Specializuoja PA:	Nėra
Pagrindinis įvykių srautas:	<ol style="list-style-type: none"> Vartotojas inicijuoja klaidos registravimą; Nurodoma klaidos informacija; Išsaugojama klaidos informacija; 	
Po sąlyga:	Sistemoje registruota nustatyta testuojamos sistemos klaida /neatitikimas.	
Pastabos:	Nėra	
Alternatyvus scenarijus:	Nėra	



4.13 pav. PA „Registruoti klaidą“ veiklos diagrama

4.14 lentelė. Panaudojimo atvejis „Peržiūrėti automatinių veiksmų scenarijų sąrašą“

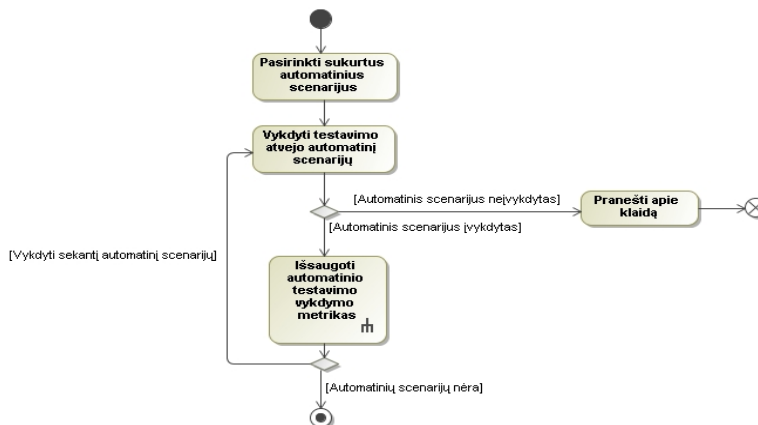
PA: Peržiūrėti automatinių veiksmų scenarijų sąrašą		
Prieš sąlyga:	Vartotojas yra prisijungęs prie sistemos. Prisijungęs vartotojas turi administratoriaus arba testuotojo teises.	
Sužadinimo sąlyga:	Vartotojas nori peržiūrėti automatinių veiksmų scenarijų sąrašą.	
Susiję PA	Apima PA:	Nėra
	Išplečia PA:	Nėra
	Specializuoja PA:	Nėra
Pagrindinis įvykių srautas:	1. Vartotojas pasirenką projektą, kurio automatinius veiksmų scenarijus nori peržiūrėti; 2. Vartotojas peržiūri pateiktą sąrašą;	
Po sąlyga:	Vartotojui pateiktas automatinių veiksmų scenarijų sąrašas.	
Pastabos:	Nėra	
Alternatyvus scenarijus:	Nėra	



4.14 pav. PA „Peržiūrėti automatinių veiksmų scenarijų sąrašą“ veiklos diagrama

4.15 lentelė. Panaudojimo atvejis „Vykdėti pasirinktą automatinių veiksmų scenarijų“

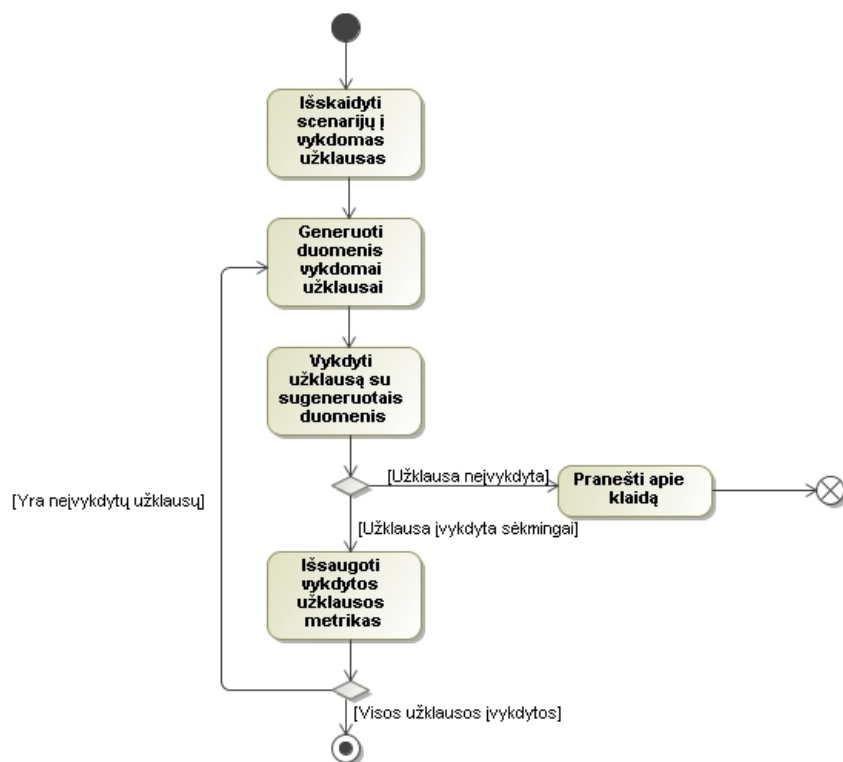
PA: Vykdėti pasirinktą automatinių veiksmų scenarijų		
Prieš sąlyga:	Vartotojas yra prisijungęs prie sistemos. Prisijungęs vartotojas turi administratoriaus arba testuotojo teises. Sistemoje išsaugotas automatinių veiksmų scenarijus.	
Sužadinimo sąlyga:	Vartotojas nori įvykdyti automatinių veiksmų scenarijų.	
Susiję PA	Apima PA:	<ul style="list-style-type: none"> • Formuoti rezultatų ataskaita • Registruoti automatinio scenarijaus metrikas
	Išplečia PA:	Nėra
	Specializuoja PA:	Nėra
Pagrindinis įvykių srautas:	1. Vartotojas pasirenka automatinių veiksmų scenarijų kurį nori įvykdyti; 2. Vykdomas pasirinktas automatinių veiksmų scenarijus;	
Po sąlyga:	Įvykdytas pasirinktas automatinių veiksmų scenarijus.	
Pastabos:	Nėra	
Alternatyvus scenarijus:	Nėra	



4.15 pav. PA „Vykdėti pasirinktą automatinių veiksmų scenarijų“ veiklos diagrama

4.16 lentelė. Panaudojimo atvejis „Registruoti automatinio scenarijaus metrikas“

PA: Registruoti automatinio scenarijaus metrikas		
Prieš sąlyga:	Vartotojas prisijungęs prie sistemos. Prisijungęs vartotojas turi administratoriaus arba testuotojo teises.	
Sužadinimo sąlyga:	Vykdomas automatinis veiksmų scenarijus.	
Susiję PA	Apima PA:	Nėra
	Išplečia PA:	• Formuoti rezultatų ataskaita
	Specializuoja PA:	Nėra
Pagrindinis įvykių srautas:	1. Vykdomas pasirinktas automatinis veiksmų scenarijus;	
Po sąlyga:	Sistemoje išsaugotos vykdyto automatinio veiksmų scenarijaus vykdymo metrikos.	
Pastabos:	Nėra	
Alternatyvus scenarijus:	Nėra	



4.16 pav. PA „Registruoti automatinio veiksmų scenarijaus metrikas“ veiklos diagrama

4.17 lentelė. Panaudojimo atvejis „Formuoti ataskaitas“

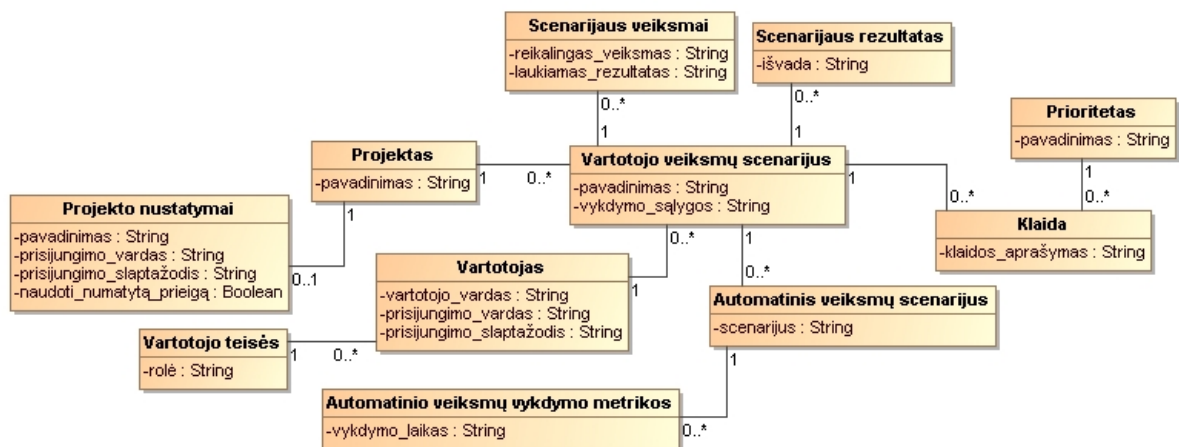
PA: Formuoti ataskaitas		
Prieš sąlyga:	Vartotojas prisijungęs prie sistemos. Prisijungęs vartotojas turi administratoriaus arba testuotojo teises. Įvykdytas automatinis veiksmų scenarijus.	
Sužadinimo sąlyga:	Nėra	
Susiję PA	Apima PA:	Nėra
	Išplečia PA:	Nėra
	Specializuoja PA:	Nėra
Pagrindinis įvykių srautas:	1. Vartotojas nurodo ataskaitos formavimo parametrus; 2. Pagal pateiktus parametrus suformuojama ataskaita.	
Po sąlyga:	Suformuota ir pateikta peržiūrai rezultatų ataskaita.	
Pastabos:	Nėra	
Alternatyvus scenarijus:	Nėra	



4.17 pav. PA „Formuoti ataskaitas“ veiklos diagrama

4.2. Dalykinės srities modelis

Sistemoje yra numatytas vartotojų teisių mechanizmas. Kiekvienam sistemos vartotojui yra suteikiamos priėjimo prie sistemos teisės. Vartotojas galės atlikti veiksmus tik pagal jam numatytas teises. Sistemoje pagal nutylėjimą yra numatytos dviejų rūšių teisės: „Administratorius“ ir „Vartotojas“. Administratoriaus teises turintys vartotojai gali atlikti visus sistemoje numatytus veiksmus (sistemos administravimo bei duomenų bazių tyrimo). Vartotojai turintys paprasto vartotojo teises galės atlikti tik duomenų bazių tyrimo veiksmus. Esant poreikiui sistemos teisių sąrašas gali būti išplečiamas naudojant Microsoft ASP.NET teisių mechanizmą.



4.18 pav. Dalykinės srities modelis

4.3. Nefunkciniai reikalavimai ir apribojimai

4.3.1. Reikalavimai sistemos išvaizdai

- Lengvai suprantama navigacija;
- Sistema sukurta naudojant užsakovo pateiktą dizaino šabloną.

4.3.2. Reikalavimai panaudojamumui

- Naudotojo sąsaja turi būti lietuvių kalba
- Paprasta naudotis sistema, nereikalauja apmokymų

4.3.3. Reikalavimai vykdymo charakteristikoms

- Sistema turi veikti stabiliai, sąrašai vartotojui turi pateikiami per 1-2 sekundes nuo užklauso pateikimo. Ataskaitos turi būti generuojamos iki 10 sekundžių nepriklausomai nuo įrašų kiekio.

4.3.4. Reikalavimai veikimo sąlygoms

- Sistema turi veikti bet kokiame kompiuteryje, turinčiame interneto ryšį ir interneto naršyklę.

4.3.5. Reikalavimai sistemos priežiūrai

- Turi būti lengvai pritaikoma prie pasikeitusių įmonės veiklos taisyklių.

4.3.6. Reikalavimai saugumui

- Sistema turi būti uždara;
- Realizuotos dvi vartotojų grupės, testuotojas – specialistas ir paprastas vartotojas;
- Sistemoje pateikiama konfidenciali informacija, todėl svarbu užtikrinti, kad prisijungtų tik registruoti vartotojai.

4.4. Reikalavimų analizė

Šiame skyriuje yra pateikiamos įrankio analizės diagramos, kurios parodo, kokios klasės turi būti sukurtos, kad sistemos panaudojimo atvejai būtų realizuoti. Diagramos sukurtos tik pagrindiniams sistemos panaudojimo atvejams.



4.19 pav. Prisijungimo analizės diagrama

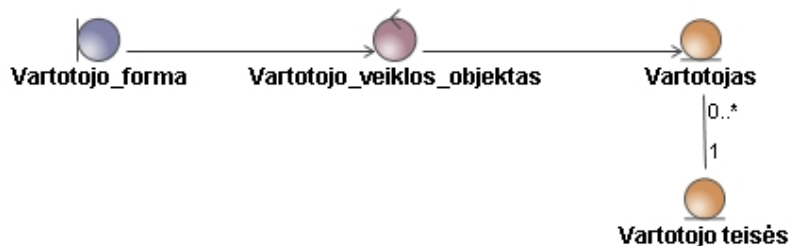
Vartotojas norėdamas prisijungti prie sistemos turi įvesti jam suteiktus prisijungimo duomenis. Menu valdiklis, yra atsakingas už bandančio prisijungti prie sistemos vartotojo autentifikaciją. Menu valdiklis, vartotojo įvestus prisijungimo duomenis sulygina su

prisijungimo duomenimis esančiais vartotojo esybėje. Sėkmės atveju vartotojas yra prijungiamas prie sistemos.



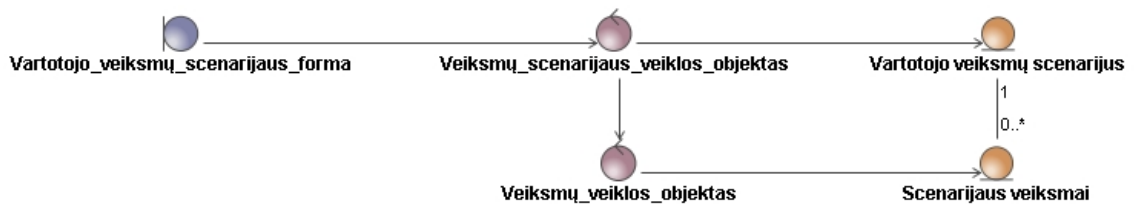
4.20 pav. Projekto sukūrimo analizės diagrama

Naujas tyrimo projektas yra sukuriamas projekto formoje. Šioje formoje yra pateikiamas visų sistemoje registruotų projektų sąrašas. Naujo tyrimo projekto kūrimo metu vartotojo įvesta naujo projekto informacija yra tikrinama. Informacijos unikalumo tikrinimo veiksmus atlieka projekto veiklos objektas. Esant unikaliems duomenims, jie yra išsaugomi projekto esybėje. Taip pat yra išsaugomi ir projekto nustatymų informacija. Ši informacija yra saugoma atskiroje projektų nustatymų esybėje.



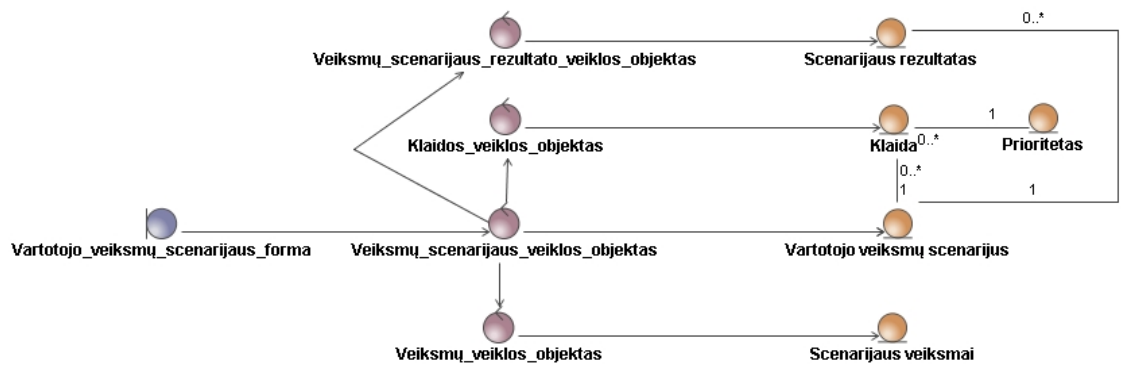
4.21 pav. Vartotojo sukūrimo analizės diagrama

Naujas sistemos vartotojas yra sukuriamas vartotojo formoje. Šioje formoje yra pateikiamas sistemoje registruotų vartotojų sąrašas bei jiems nurodytos teisės. Naujo vartotojo įvesta informacija yra tikrinama. Už informacijos tikrinimą bei išsaugojimą yra atsakingas vartotojo veiklos objektas. Vartotojo informacija yra išsaugoma vartotojo esybėje susiejant šią informaciją su sistemos teisėmis, kurios yra saugomos vartotojo teisių esybėje.



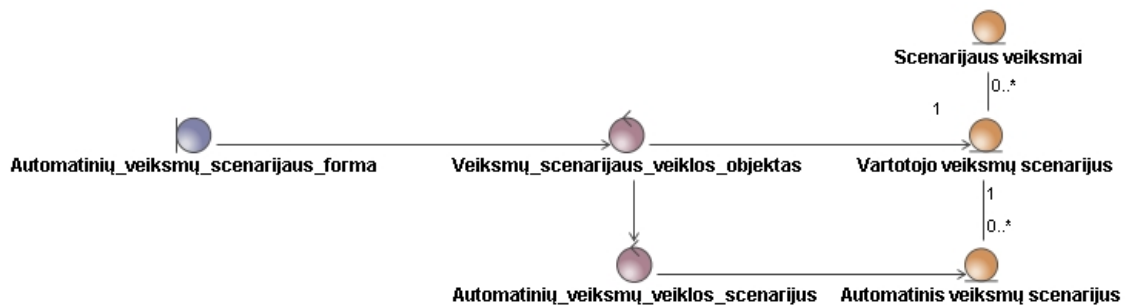
4.22 pav. Vartotojo veiksmų sukūrimo analizės diagrama

Vartotojo veiksmų scenarijus yra sukuriamas vartotojo veiksmų scenarijaus formoje. Šioje formoje vartotojas nurodo naujo veiksmų scenarijaus pavadinimą, datą, jo vykdymo sąlygas. Kiekvienam vartotojo veiksmų scenarijui yra suformuojamas turimų atlikti veiksmų sąrašas. Už pilną vartotojo veiksmų scenarijaus suformavimą yra atsakingi du valdikliai: veiksmų scenarijaus veiklos objektas ir veiksmų veiklos objektas. Dėl ypatybės, kad kuriamas vartotojo veiksmų scenarijus gali turėti keletą reikalingų atlikti veiksmų, dėl šios priežasties informacija yra saugoma dvejose esybėse: vartotojo veiksmų scenarijaus esybėje ir scenarijaus veiksmų esybėje.



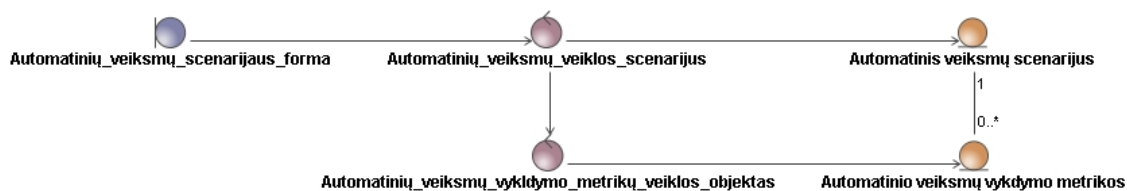
4.23 pav. Vartotojo veiksmų scenarijaus vykdymo analizės diagrama

Vartotojo veiksmų scenarijų sąrašas yra pateikiamas vartotojo veiksmų scenarijų formoje. Pasirinktas vartotojo veiksmų scenarijus yra atliekamas peržiūrint jame pateikiama informaciją bei atliekant jame nurodytus atliekamas veiksmus. Reikalingi atlikti veiksmai yra atliekami susietoje sistemoje. Registruotų sistemoje vartotojo veiksmų scenarijų sąrašą suformuoja ir pateikia veiksmų scenarijų veiklos objektas kartu su veiksmų veiklos objektu. Vartotojui atliekant nurodytus veiksmus susietoje sistemoje ir pastebėjus neatitikimus, klaidos veiklos objekto pagalba yra registruojama klaida/neatitikimas. Klaidos/neatitikimo duomenys išsaugomi klaidos esybėje. Atlikus visus nurodytus veiksmus yra registruojamas rezultatas. Rezultato duomenys yra valdomi veiksmų scenarijaus rezultato veiklos objekto. Šie duomenys yra išsaugomi scenarijaus rezultato esybėje.



4.24 pav. Automatinio veiksmų scenarijaus sukūrimo analizės diagrama

Automatinis veiksmų scenarijus yra sukuriamas vartotojo veiksmų scenarijaus vykdymo metu. Vartotojui pasirinkus vykdyti vartotojo veiksmų scenarijų yra nustatoma automatinio veiksmų scenarijaus kūrimo pradžia. Nuo šios akimirkos vartotojo atliekami reikalingi veiksmai susietoje sistemoje yra stebimi automatinių veiksmų veiklos objekto. Šis veiklos objektas analizuoja atliekamas procedūras susietoje sistemoje ir išsaugo šiuos duomenis automatinio veiksmų scenarijaus esybėje.



4.25 pav. Automatinio veiksmų scenarijaus vykdymo analizės diagrama

Automatinių veiksmų scenarijų sąrašas yra pateikiamas automatinių veiksmų scenarijų formoje. Vartotojui nurodžius automatinius veiksmų scenarijus kurie turi būti atlikti, automatinių veiksmų veiklos objektas pakartoja vartotojo atliktus veiksmus (vykdytas procedūras) vartotojo veiksmų scenarijaus vykdymo metu. Automatinių veiksmų veiklos objekto atliekamų veiksmų metu, kitas veiklos objektas (automatinių veiksmų vykdymo metrikų veiklos objektas) stebi atliekamus veiksmus, bei registruoja veiksmų atlikimo metrikas (atlikimo laiką). Šios metrikos yra išsaugomos automatinio veiksmų vykdymo metrikų esybėje.

5. Duomenų bazių našumo tyrimo įrankio projektas

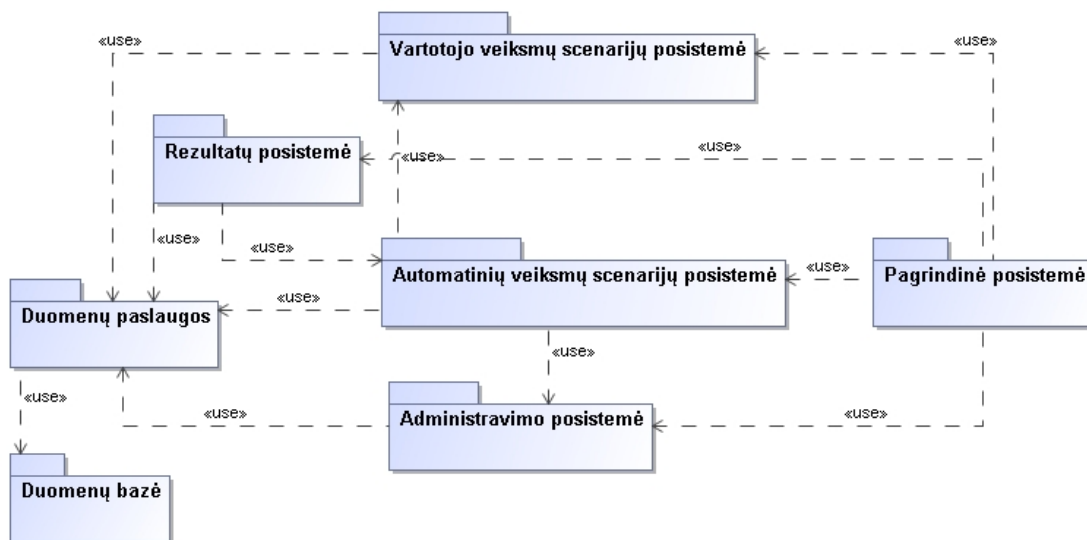
5.1. Sistemos architektūra

Skyriuje pateikiama informacija projektuojamos sistemos loginės architektūros vizija. Taip pat pateikiama projektuojamos sistemos klasių diagrama bei pateikiama projektuojamos sistemos veiklos logika realizuota valdymo klasėmis.

5.1.1. Loginė sistemos architektūra

Sistemos architektūra pateikiama išskaidyta į posistemas. Sistemoje numatyta realizuoti šias posistemas: administravimo posistemę, vartotojo veiksmų scenarijų posistemę, automatinių veiksmų scenarijų posistemę, rezultatų posistemę. Visas šias posistemas susiejamas realizuojant pagrindinę posistemę. Duomenų saugojimas atliekamas realizuojant duomenų paslaugas bei sąryšį su duomenų baze. Kiekvienos posistemės atsakomybės aprašomos atskirai:

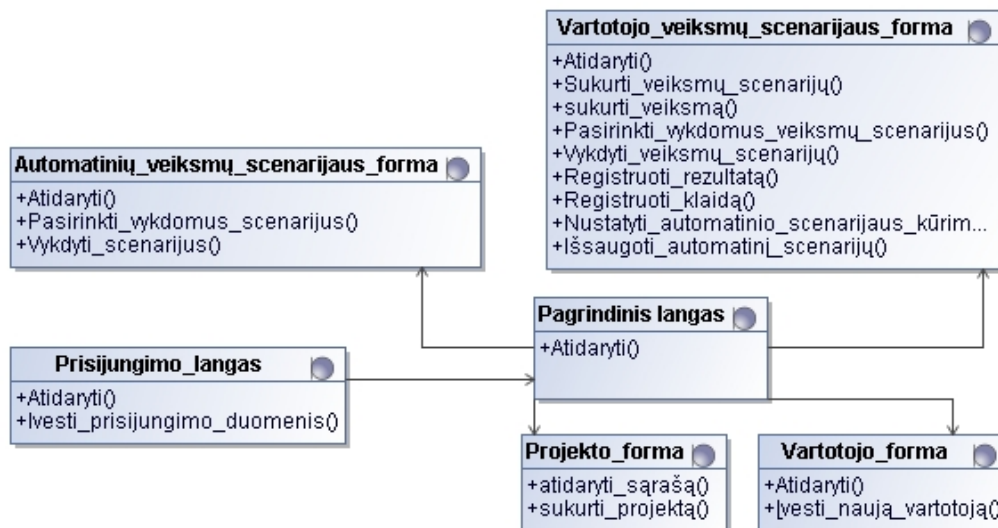
- Pagrindinė posistemė atsakinga už visų realizuojamų posistemių bendradarbiavimą, duomenų mainus;
- Administravimo posistemė – atsakinga už tyrimo projektų administravimą (kūrimą, peržiūrą, šalinimą), už sistemos vartotojų administravimą (kūrimą, peržiūrą, šalinimą, teisių suteikimą), už sistemos nustatymų administravimą (kūrimą, peržiūrą, šalinimą);
- Vartotojų veiksmų scenarijų posistemė – atsakinga už naujų vartotojo veiksmų scenarijų kūrimą, jau egzistuojančių scenarijų peržiūrą ir redagavimą. Taip pat atsakinga už sukurtų vartotojų veiksmų scenarijų vykdymą;
- Automatinių veiksmų scenarijų posistemė – atsakinga už automatinių scenarijų kūrimą, sukurtų scenarijų peržiūrą bei jų vykdymą. Taip pat atsakinga už vykdomų automatinių scenarijų metrikų sekimą bei išsaugojimą;
- Rezultatų posistemė – atsakinga už ataskaitų formavimą, scenarijų metrikų lyginimą tarpusavyje, užklausų vykdymą.



5.1 pav. Loginė sistemos architektūra

5.1.2. Vartotojo paslaugos

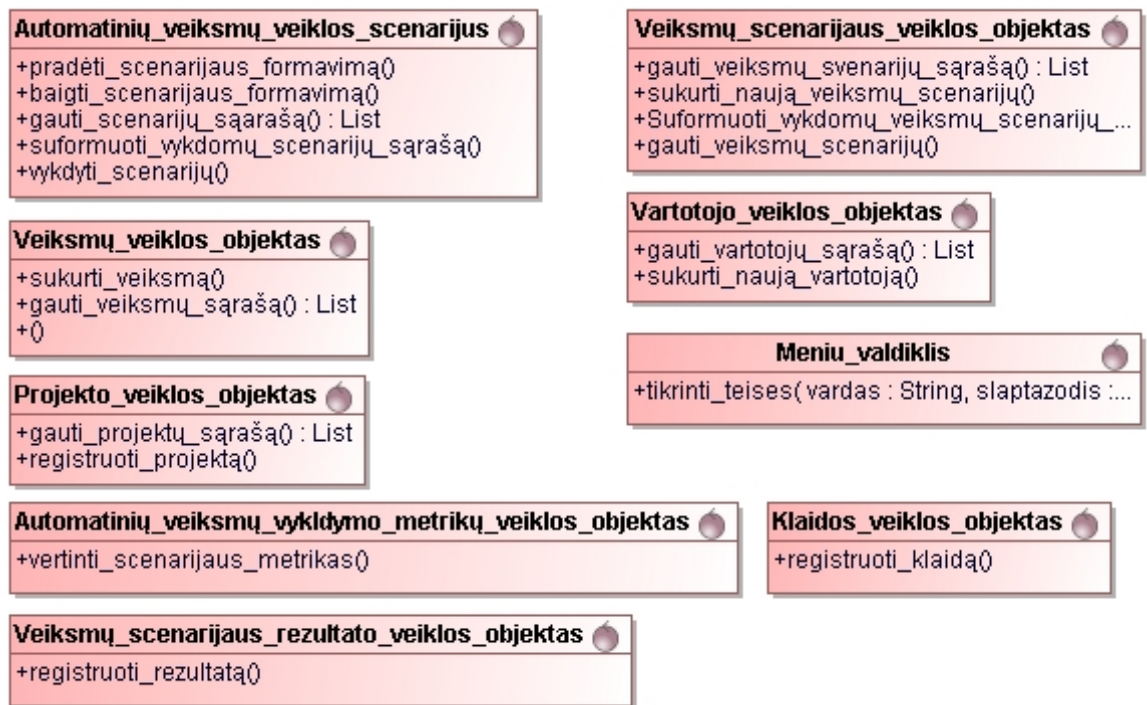
Vartotojo sąsajos modelis realizuotas pateikiant ribines klases. Vartotojas nurodęs teisingą prisijungimo informaciją iš prisijungimo lango patenka į pagrindinį langą, iš kurio vartotojas gali naviguoti sistemoje pagal jam nurodytas vartotojo teises. Iš pagrindinio lango yra pasiekiamos šios formos: projekto forma, vartotojų formą, vartotojų veiksmų scenarijaus forma bei automatinių veiksmų scenarijaus forma (čia yra pateikiamos tik aukščiausio lygio formos).



5.2 pav. Vartotojo sąsajos modelis

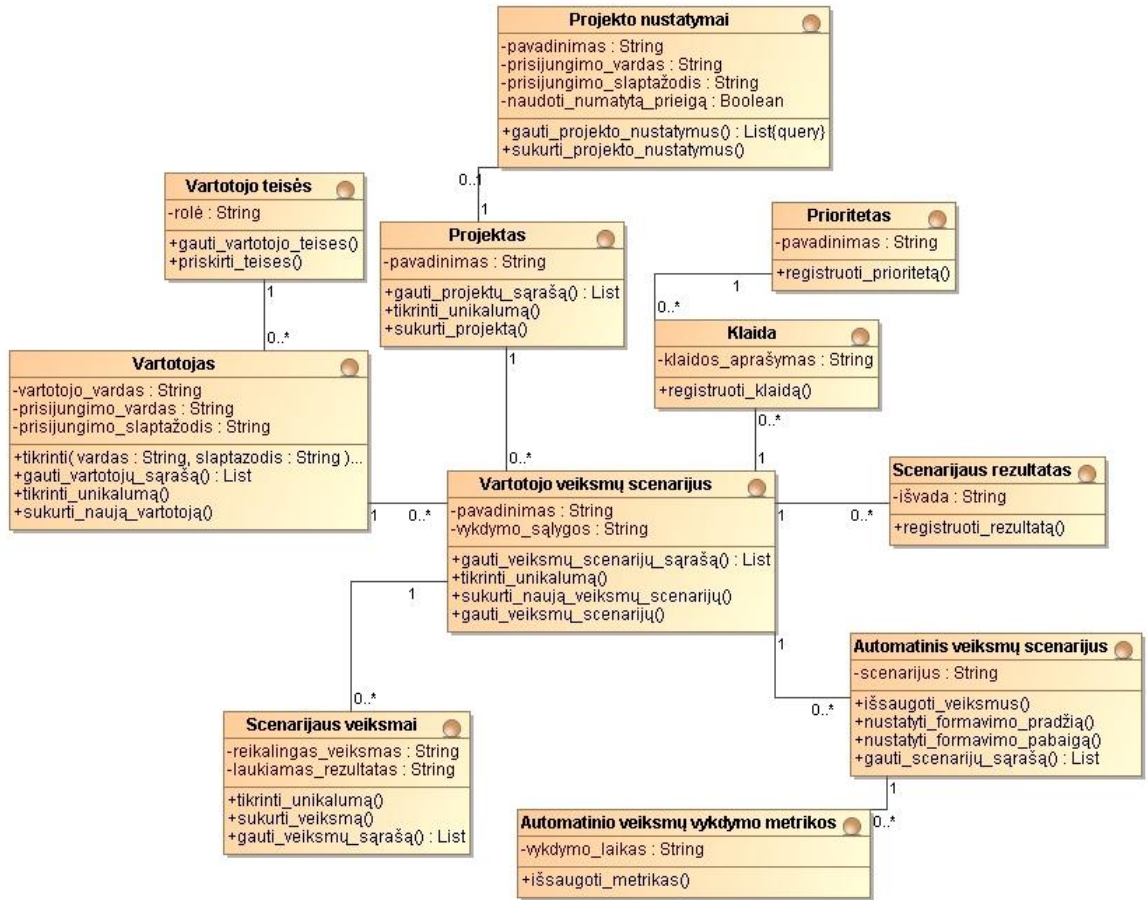
5.1.3. Veiklos paslaugos

Sistemos veiklos logika yra realizuojama valdymo klasėmis. Veiklos logika atspindi sistemos veiksmus tarp vartotojo sąsajos ir duomenų bazės.



5.3 pav. Sistemos veiklos logikos modelis

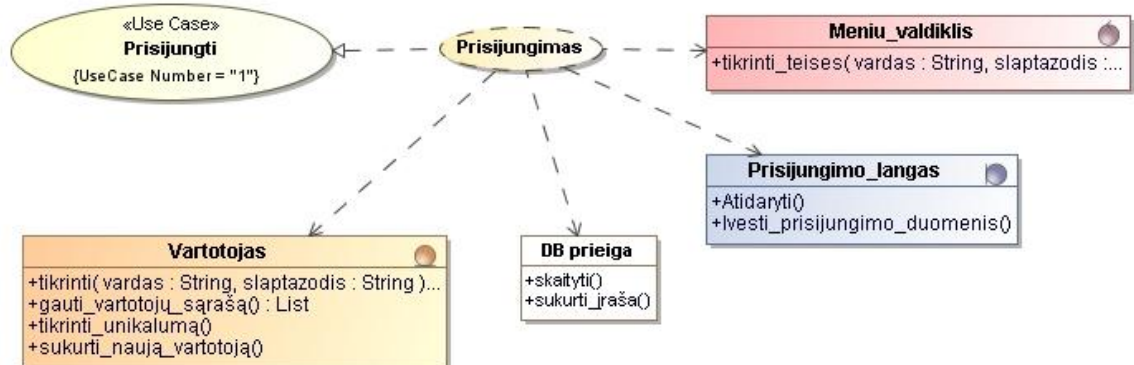
5.1.4. Duomenų paslaugos



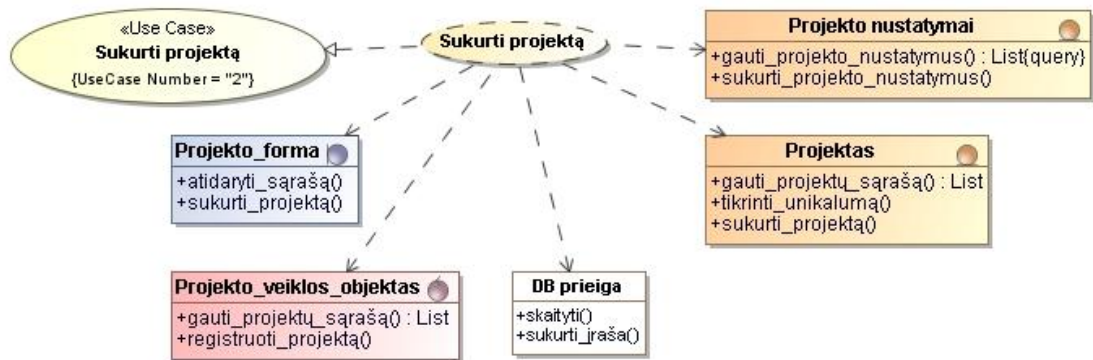
5.4 pav. Duomenų paslaugų modelis

5.2. Detalus projektas

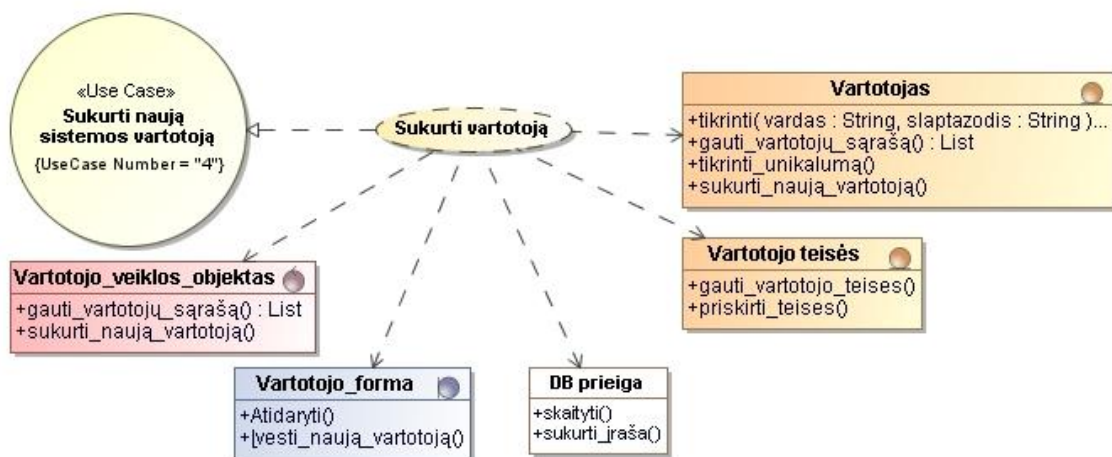
Šiame skyriuje yra pateikiamos pagrindinių sistemos panaudojimo atvejų suskirstytų pagal posistemes realizacijos diagramos.



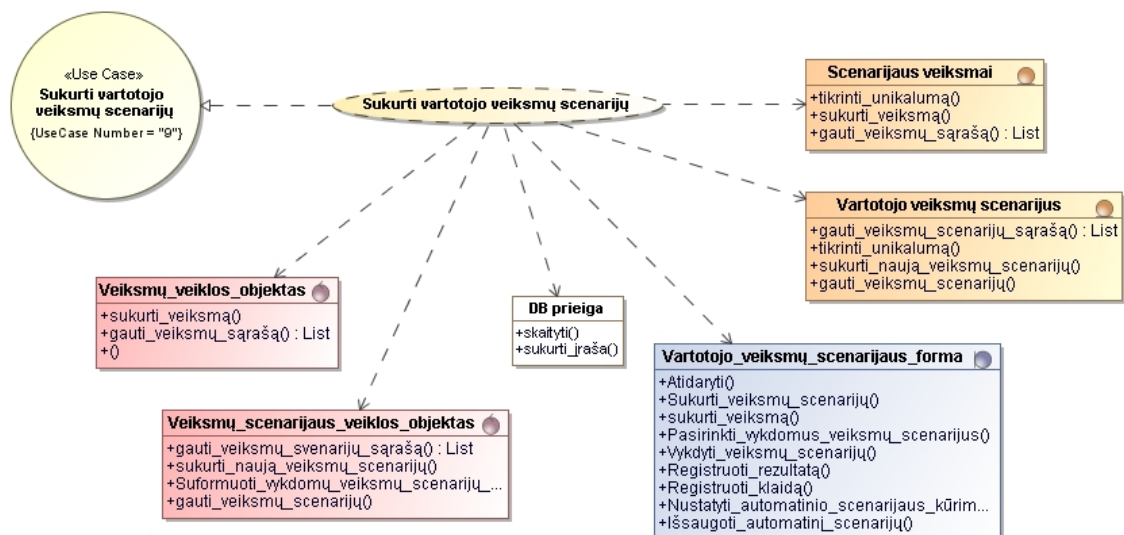
5.5 pav. PA „Prisijungti“ realizacijos diagrama



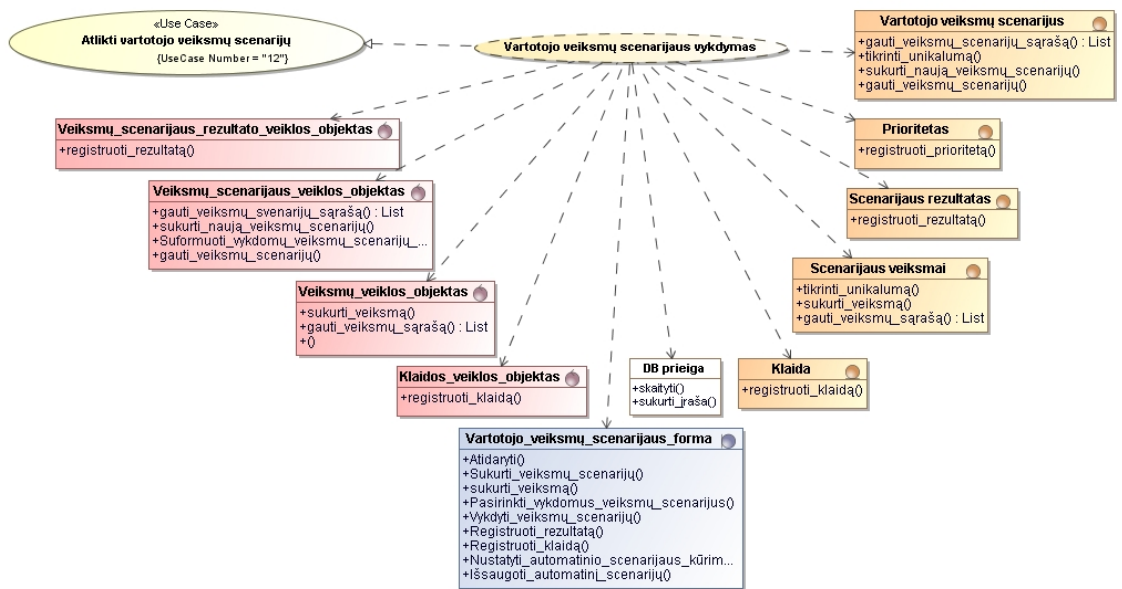
5.6 pav. PA „Sukurti projektą“ realizacijos diagrama



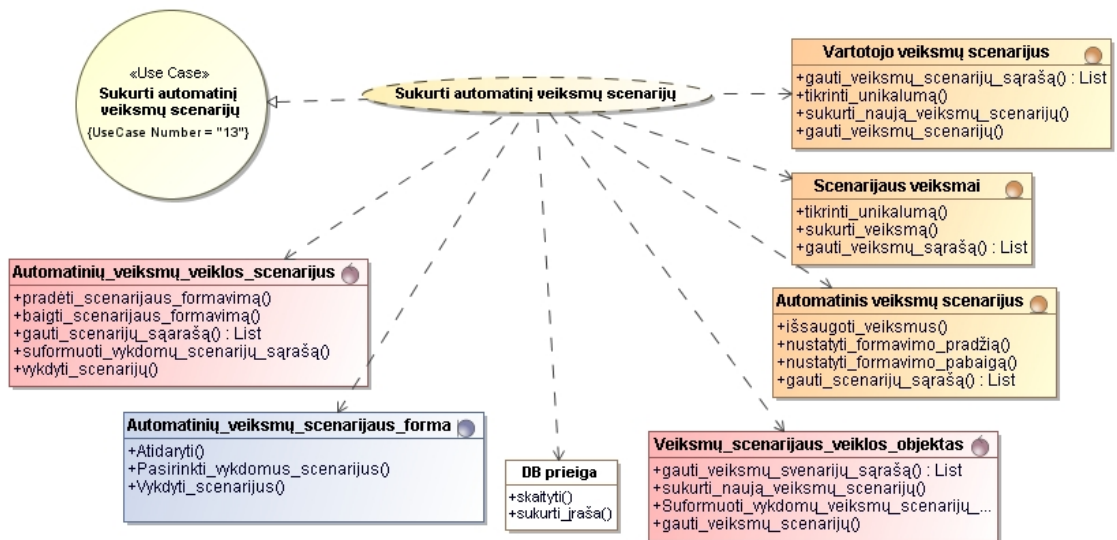
5.7 pav. PA „Sukurti vartotoją“ realizacijos diagrama



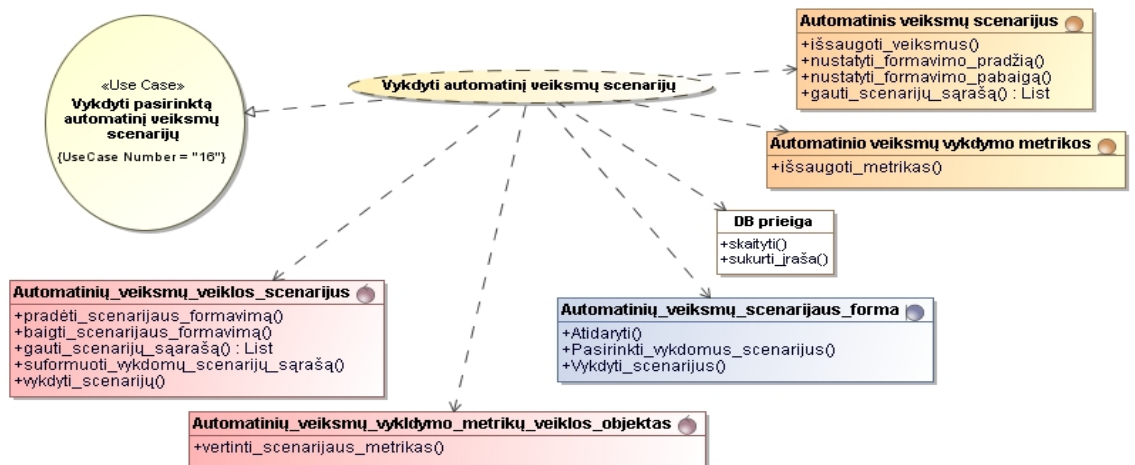
5.8 pav. PA „Sukurti vartotojo veiksmų scenarijų“ realizacijos diagrama



5.9 pav. PA „Vartotojo veiksmų scenarijaus vykdymas“ realizacijos diagrama



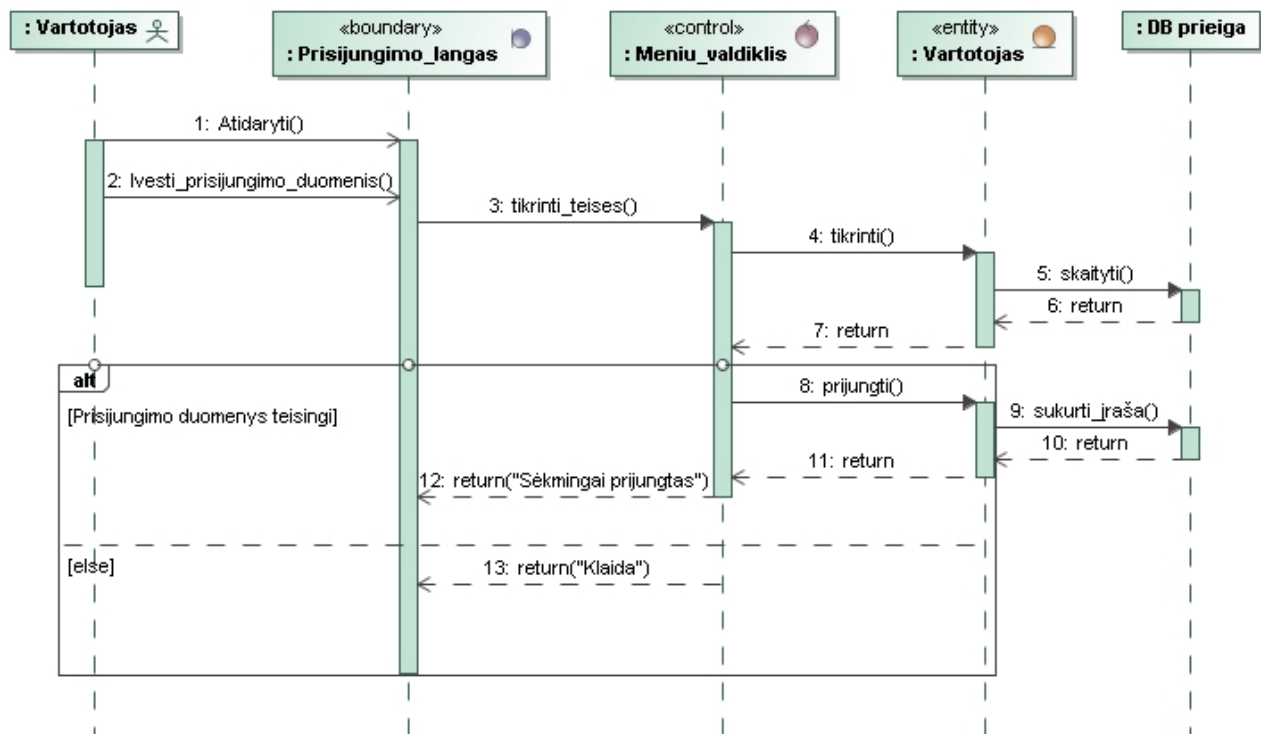
5.10 pav. PA „Sukurti automatinį veiksmų scenarijų“ realizacijos diagrama



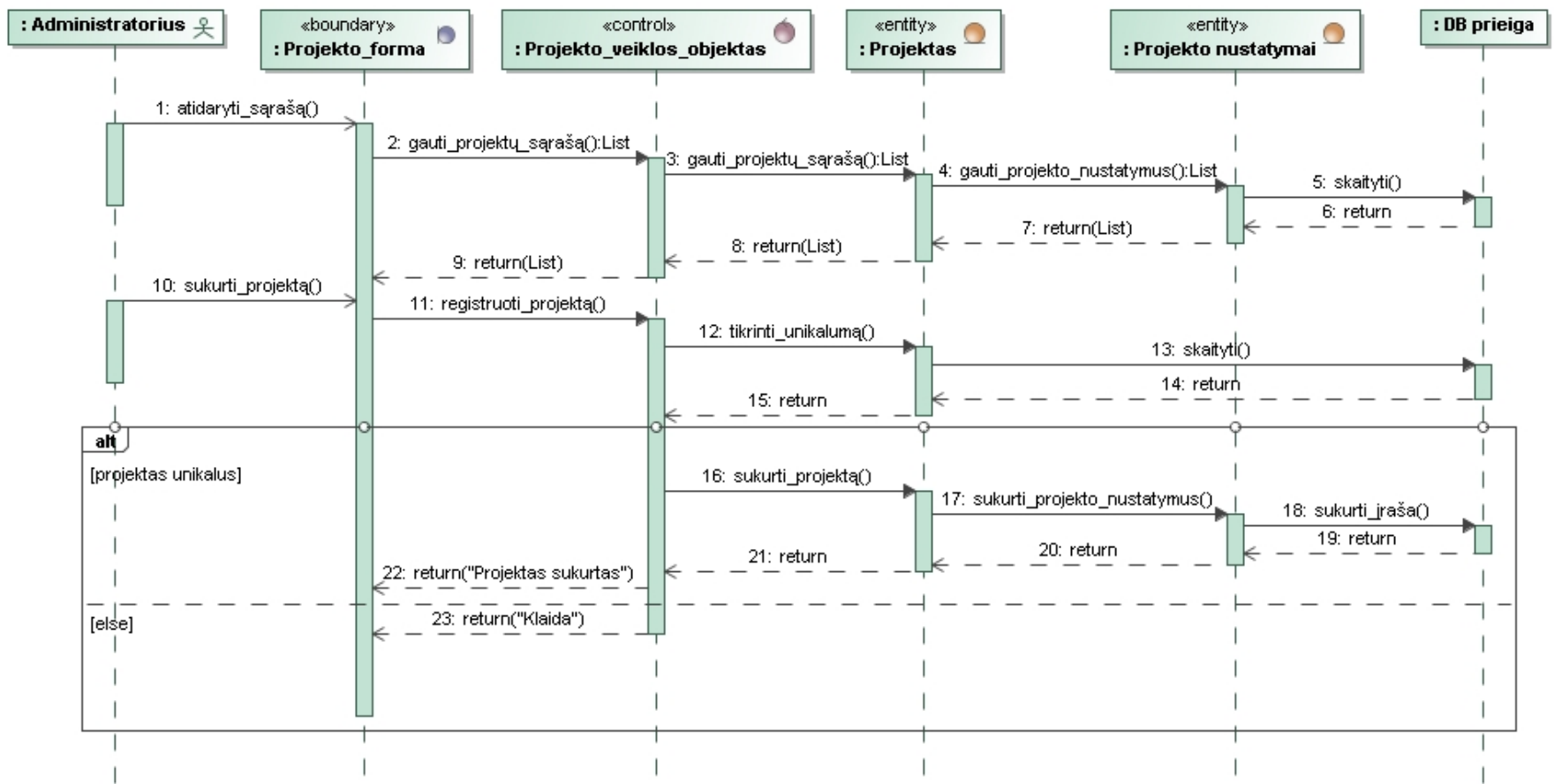
5.11 pav. PA „Vykdyti automatinį veiksmų scenarijų“ realizacijos diagrama

5.3.Sistemos elgsenos modelis

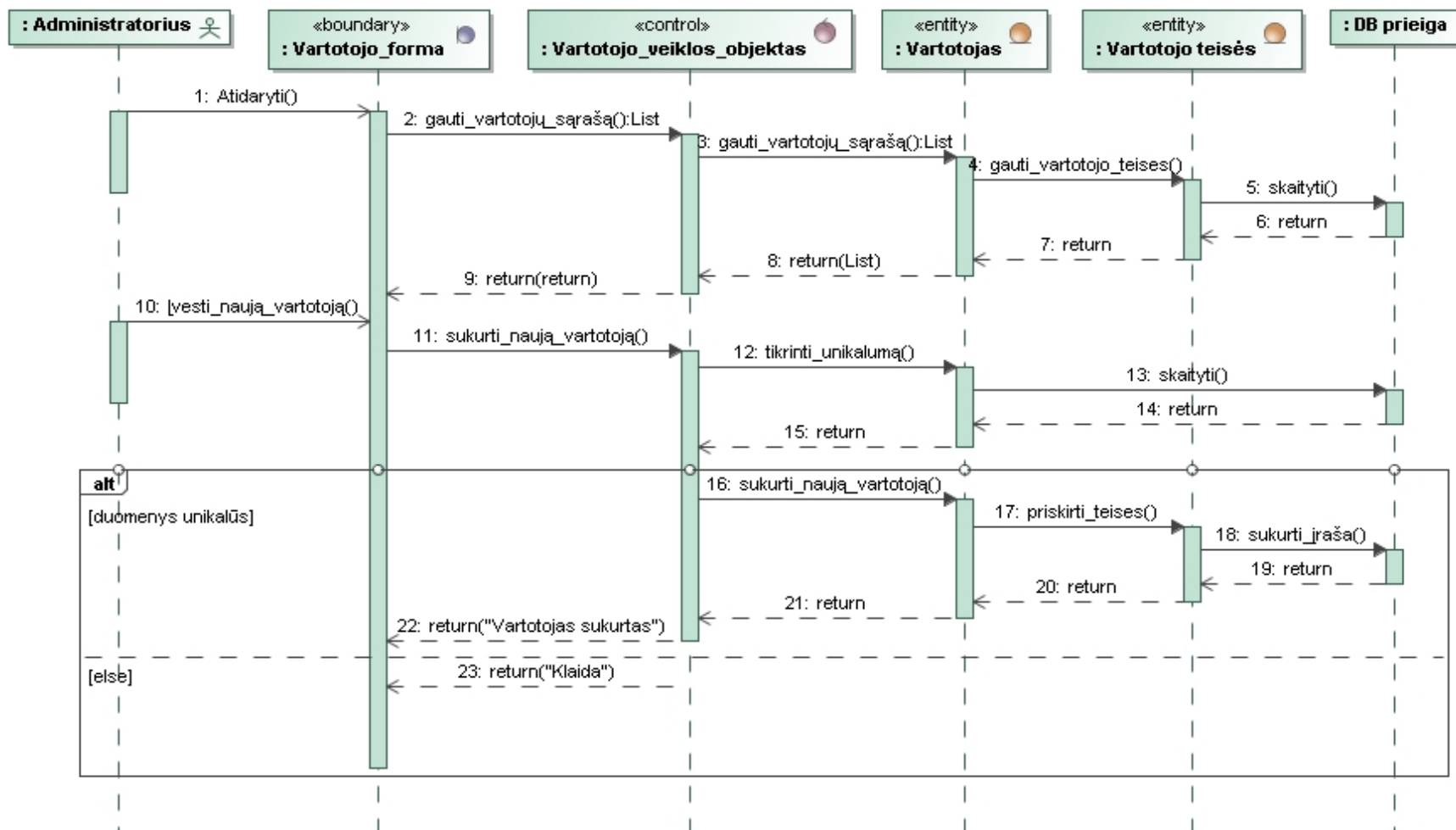
Šiame skyriuje pateikiamos sistemos elgsenos diagramos, kurios atspindi kokie veiksmai yra vykdomi sistemoje vartotojui pasirinkus vieną ar kitą sistemos funkciją.



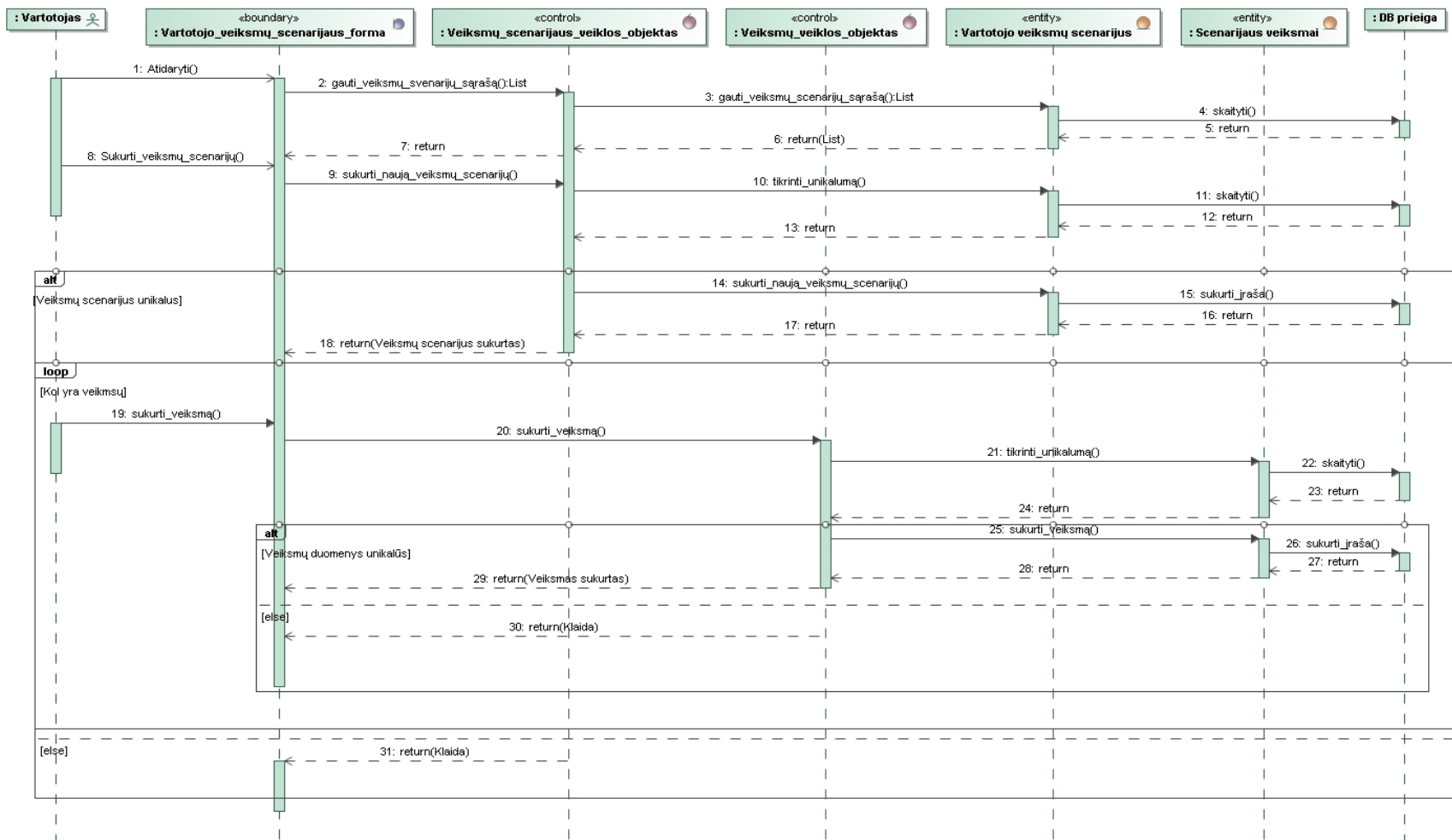
5.12 pav. PA „Prisijungti“ sekų diagrama



5.13 pav. PA „Sukurti projektą“ sekų diagrama



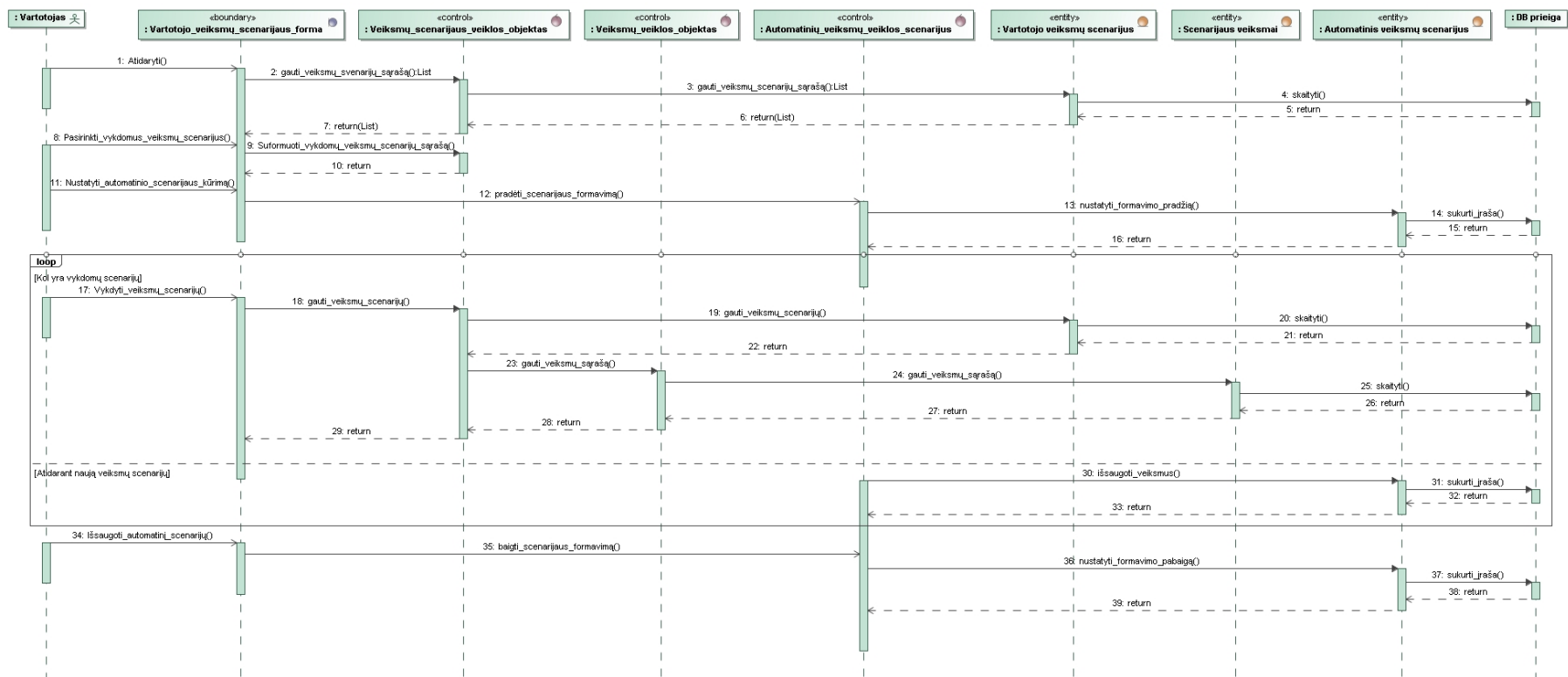
5.14 pav. PA „Sukurti vartotoją“ sekų diagrama



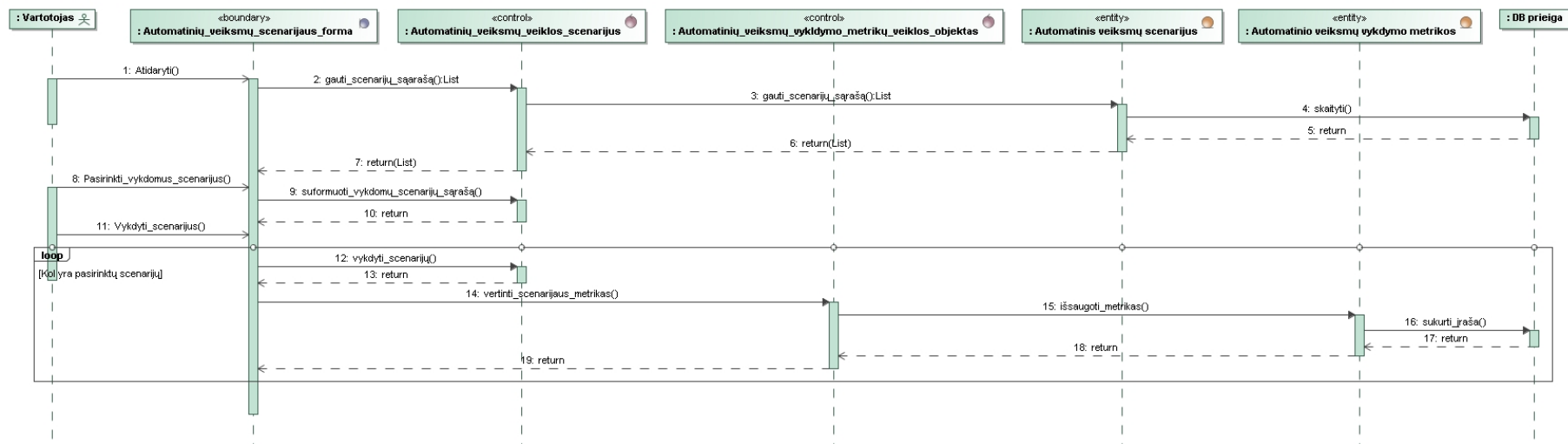
5.15 pav. PA „Vartotojo veiksmų scenarijaus sukūrimas“ sekų diagrama



5.16 pav. PA „Vartotojo veiksmų scenarijaus vykdymas“ sekų diagrama



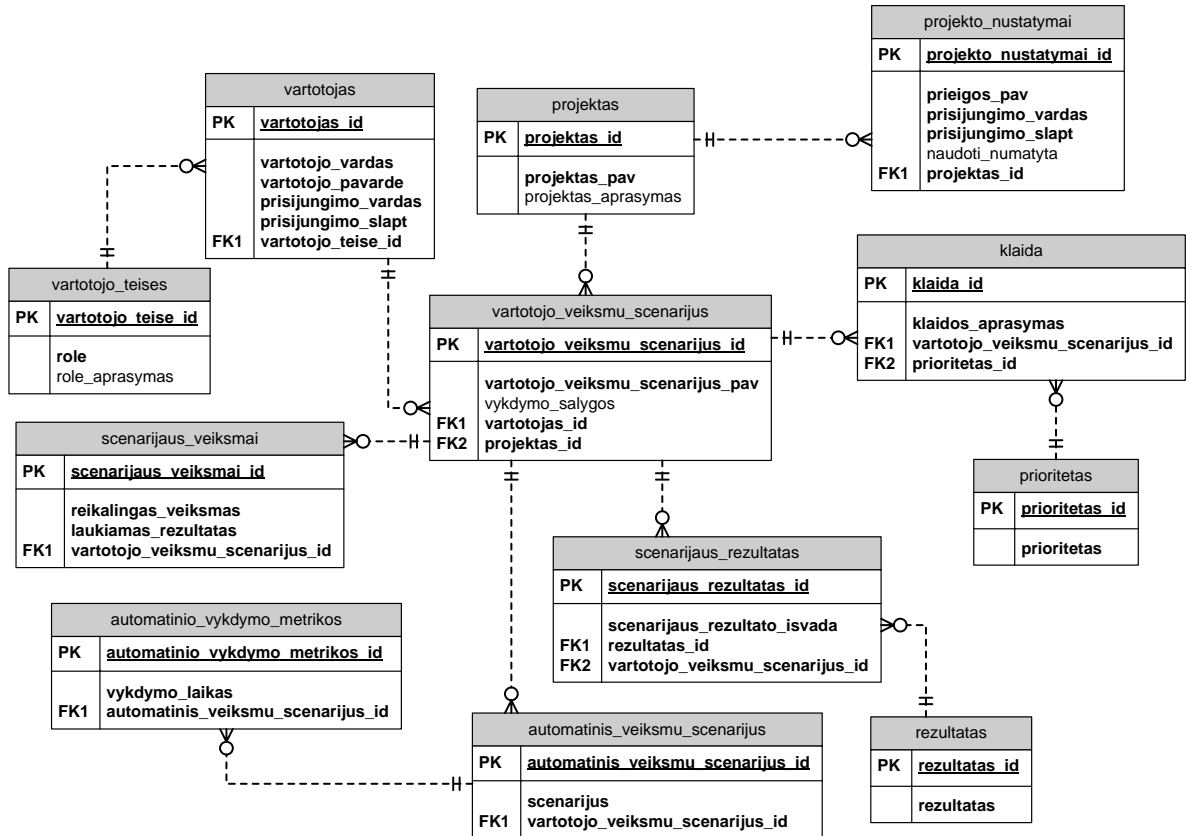
5.17 pav. PA „Automatinio veiksmų scenarijaus sukūrimas“ sekų diagrama



5.18 pav. PA „Automatinio veiksmų scenarijaus vykdymas“ sekų diagrama

5.4. Duomenų bazės schema

Šiame skyriuje yra pateikiama tyrimo įrankio duomenų bazės schema. Kiekvienos duomenų bazių lentelės aprašymas (lentelėje esantys laukai, laukų tipai bei laukų aprašymas) pateikiamas (5.1 - 5.12) lentelėse. Duomenų bazės schema pateikiama 5.19 paveikslėlyje.



5.19 pav. Duomenų bazės schema

5.1 lentelė. Duomenų bazės lentelė „Vartotojas“

Lauko pavadinimas	Lauko tipas	Lauko aprašymas
vartotojas_id	integer	Lentelės identifikacinis raktas
vartotojo_vardas	varchar(20)	Sistemos vartotojo vardas
vartotojo_pavardė	varchar(20)	Sistemos vartotojo pavardė
prisijungimo_vardas	varchar(20)	Sistemos vartotojo prisijungimo vardas
prisijungimo_slapt	varchar(10)	Sistemos vartotojo prisijungimo prie sistemos slaptažodis
vartotojo_teise_id	integer	Vartotojo teisių identifikacinis laukas

5.2 lentelė. Duomenų bazės lentelės „Vartotojo teise“

Lauko pavadinimas	Lauko tipas	Lauko aprašymas
vartotojo_teise_id	integer	Lentelės identifikacinis raktas
role	varchar(10)	Vartotojo teisės pavadinimas

roles_aprasymas	varchar(50)	Vartotojo teisės aprašymas
-----------------	-------------	----------------------------

5.3 lentelė. Duomenų bazės lentelė „Projektas“

Lauko pavadinimas	Lauko tipas	Lauko aprašymas
projektas_id	integer	Lentelės identifikacinis raktas
projektas_pav	varchar(20)	Tyriamojo projekto pavadinimas
projektas_aprasymas	varchar(50)	Tyriamojo projekto aprašymas

5.4 lentelė. Duomenų bazės lentelė „Projekto nustatymai“

Lauko pavadinimas	Lauko tipas	Lauko aprašymas
projekto_nustatymai_id	Integer	Lentelės identifikacinis raktas
prieigos_pav	varchar(10)	Projekto prieigos pavadinimas
prisijungimo vardas	varchar(20)	Prisijungimo prie prieigos vartotojo vardas
prisijungimo_slapt	varchar(20)	Prisijungimo prie prieigos vartotojo slaptažodis
naudoti_numatyta	Bit	Požymis ar naudojamas numatytasis sistemos prisijungimo prie prieigos nustatymas
projektas_id	Integer	Projekto identifikacinis raktas

5.5 lentelė. Duomenų bazių lentelė „Vartotojo veiksmų scenarijus“

Lauko pavadinimas	Lauko tipas	Lauko aprašymas
vartotojo_veiksmu_scenarijus_id	Integer	Vartotojų veiksmų scenarijų lentelės identifikacinis raktas
vartotojo_veiksmu_scenarijus_pav	varchar(20)	Vartotojo veiksmų scenarijus pavadinimas
vykdymo_salygos	varchar(200)	Vartotojo veiksmų scenarijus vykdymo sąlygos
vartotojas_id	Integer	Susieto sistemos vartotojo identifikacinis raktas
projektas_id	Integer	Projekto identifikacinis raktas

5.6 lentelė. Duomenų bazės lentelė „Scenarijaus veiksmi“

Lauko pavadinimas	Lauko tipas	Lauko aprašymas
scenarijaus_veiksmi_id	Integer	Lentelės identifikacinis raktas
reikalingas_veiksmas	varchar(50)	Numatytas atlikti veiksmas
laukiamas_rezultatas	varchar(50)	Laukiamas numatytojo veiksmo rezultatas
vartotojo_veiksmu_scenarijus_id	Integer	Susieto vartotojo veiksmų scenarijaus identifikacinis raktas

5.7 lentelė. Duomenų bazės lentelė „Klaida“

Lauko pavadinimas	Lauko tipas	Lauko aprašymas
klaida_id	Integer	Lentelės identifikacinis raktas
klaidos_aprasymas	varchar(50)	Registruojamas klaidos aprašymas
vartotojo_veiksmu_scenarijus_id	Integer	Susieto vartotojo veiksmų scenarijaus identifikacinis raktas
prioritetas_id	Integer	Susieto prioriteto identifikacinis raktas

5.8 lentelė. Duomenų bazės lentelė „Prioritetas“

Lauko pavadinimas	Lauko tipas	Lauko aprašymas
prioritetas_id	integer	Lentelės identifikacinis raktas
Prioritetas	varchar(10)	Prioriteto pavadinimas

5.9 lentelė. Duomenų bazės lentelė „Scenarijaus rezultatas“

Lauko pavadinimas	Lauko tipas	Lauko aprašymas
scenarijaus_rezultatas_id	integer	Lentelės identifikacinis raktas
scenarijaus_rezultato_aprasymas	varchar(50)	Nurodomas scenarijaus aprašymas
rezultatas_id	integer	Susieto rezultato identifikacinis raktas
virtotojo_veiksmu_scenarijus_id	integer	Susieto virtotojo veiksmų scenarijaus identifikacinis raktas

5.10 lentelė. Duomenų bazės lentelė „Rezultatas“

Lauko pavadinimas	Lauko tipas	Lauko aprašymas
rezultatas_id	integer	Lentelės identifikacinis raktas
Rezultatas	varchar(10)	Rezultato pavadinimas

5.11 lentelė. Duomenų bazės lentelė „Automatinio veiksmų scenarijus“

Lauko pavadinimas	Lauko tipas	Lauko aprašymas
automatinis_veiksmu_scenarijus_id	integer	Lentelės identifikacinis raktas
Scenarijus	varchar(200)	Automatinių veiksmų scenarijaus sakiniai
virtotojo_veiksmu_scenarijaus_id	integer	Susieto virtotojo veiksmų scenarijaus identifikacinis raktas

5.12 lentelė. Duomenų bazės lentelė „Automatinio vykdymo metrikos“

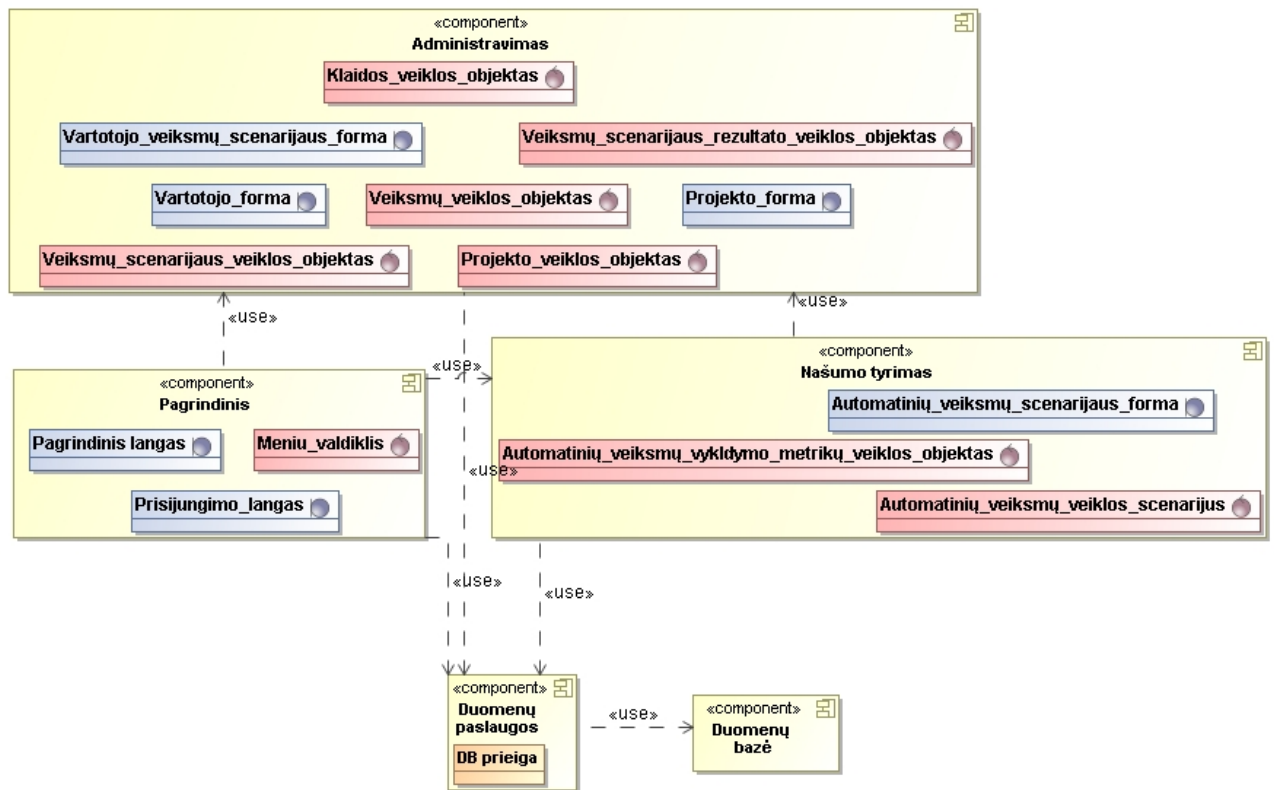
Lauko pavadinimas	Lauko tipas	Lauko aprašymas
automatinio_vykdymo_metrikos_id	integer	Lentelės identifikacinis raktas
vykdymo_laikas	double	Automatinio veiksmų scenarijaus atlikimo laikas
automatinis_veiksmu_scenarijus_id	integer	Susieto automatinio veiksmų scenarijaus identifikacinis raktas

6. Tyrimo įrankio realizacija

Šiame skyriuje yra pateikiama tyrimo įrankio realizacijos modelis, sistemos diegimo nurodymai bei vartotojo sąsajos ir testavimo aprašymai.

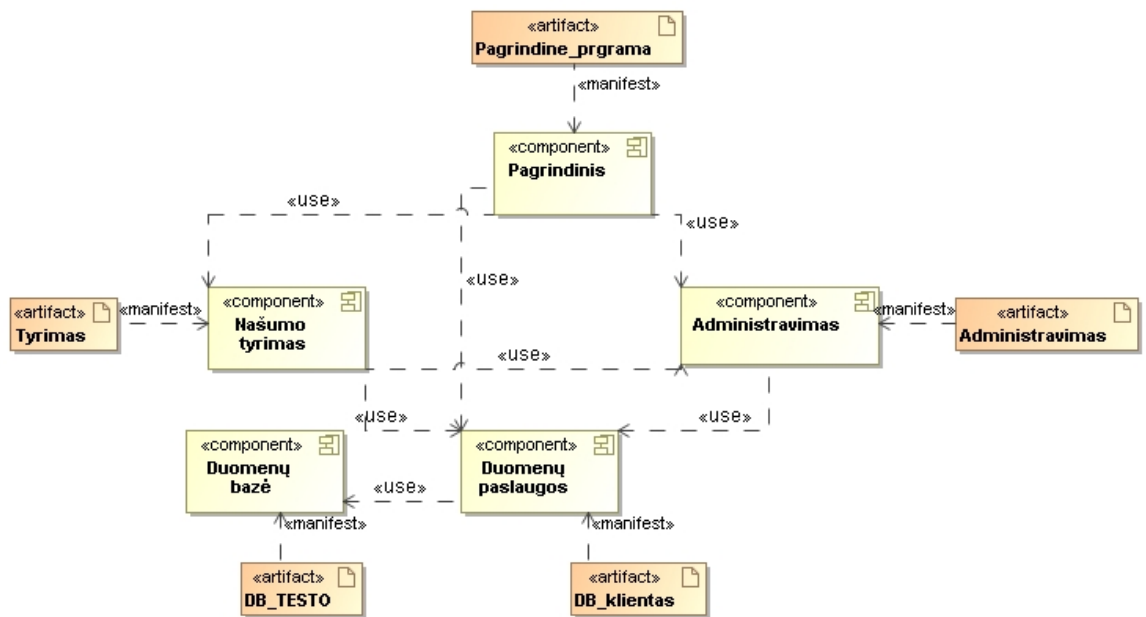
6.1. Realizacijos modelis

Realizacijos modelyje nurodomi sistemos komponentai bei juos atitinkantys artefaktai. Atskiroje komponentų diagramoje pateikiami komponentų tarpusavio sąryšiai.



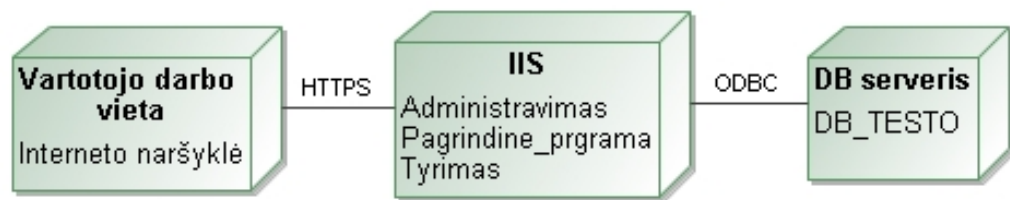
6.1 pav. Sistemos komponentų diagrama

Kiekvieno sistemos komponento sąryšiai su juos realizuojančiais artefaktais yra pateikiami sistemos komponentų ir artefaktų diagramoje (6.2 paveikslėlis).



6.2 pav. Sistemos komponentų ir artefaktų sąryšių diagrama

Sistemos komponentus realizuojantys artefaktai yra pateikiami fiziniuose įrenginiuose. Sistemos artefaktų diegimo fiziniuose įrenginiuose diagrama pateikiama (6.3).



6.3 pav. Sistemos diegimo diagrama

6.2. Sistemos diegimas

Tyrimo įrankis sistema yra diegiama kaip ir internetinis puslapis IIS („*Internet Information Service*“). Visi IIS nustatymai yra atliekami pagal įmonės, kurioje diegiama sistema vidaus taisyklės. 6.2.1 skyrelyje pateikiama informacija apie reikalavimus techninei įrangai.

6.2.1. Reikalavimai techninei įrangai

Tyrimo įrankio techninės priemonės turi susidėti iš:

- 1 sistemos serverio;
- 1 duomenų bazių serverio.

6.1 lentelė. Reikalavimai interneto tarnybinei stočiai (sistemos serveriui)

Charakteristikos pavadinimas	Parametro reikšmė
Procesorius	Pentium III tipo procesorius arba galingesnis
Operatyvioji atmintis	1GB arba daugiau (iki maksimalaus palaikomo kiekio)
Operacinė sistema	Windows server 2003
Taikomieji paketai	IIS 6.0 ar naujesnis

6.2 lentelė. Reikalavimai duomenų bazių serveriui

Charakteristikos pavadinimas	Parametro reikšmė
Procesorius	Pentium III tipo procesorius arba galingesnis
Operatyvioji atmintis	1GB arba daugiau (iki maksimalaus palaikomo kiekio)
Operacinė sistema	Windows server 2003
Taikomieji paketai	SQL server 2005 Express

6.3. Vartotojo sąsaja

TESTO - tai duomenų bazių našumo tyrimo informacinė sistema, kuri vartotojui leidžia patogiai administruoti vartotojo veiksmų scenarijus, sukurti automatinius veiksmų scenarijus bei sistemos duomenų bazės našumo tyrimą. Sistemos TESTO vartotojui pateikiamos išsamios tiriamos sistemos darbo charakteristikos padeda priimti svarbius sistemos optimizavimo sprendimus.

6.3.1. IS tikslai

Sistema TESTO tenkina šiuos pagrindinius tikslus:

Administravimas:

- Sistemos vartotojų administravimas;
- Sistemos nustatymų administravimas;
- Tiriamų projektų informacijos administravimas.

Veiksmų scenarijai:

- Vartotojo veiksmų scenarijaus sukūrimas;
- Automatinių veiksmų scenarijaus sukūrimas;
- Paprastas veiksmų scenarijų administravimas;
- Paprastas veiksmų scenarijų vykdymas.

Rezultatų analizė:

- Patogi gautų našumo tyrimo rezultatų peržiūra;
- Išsami našumo tyrimo rezultatų analize pagal pasirinktą našumo charakteristiką.

Sistemos integracijos:

- TESTO integracija su Team Foundation Server.

6.3.2. Administravimas

Administravimo dalyje vartotojas atlieka pradinis sistemos nustatymus. Šioje dalyje yra pateikiamas sistemos vartotojų administravimas, kurioje vartotojas gali sukurti naujus TESTO sistemos vartotojus bei administruoti jų duomenis (vardas, pavardė, rolė). Taip pat vartotojui administravimo dalyje patikėtas ir tiriamų projektų valdymas. Vartotojas tiriamų projektų dalyje sukuria tiriamų projektų įrašus ir pateikia jų detalius aprašymus bei jų prijungimo informaciją. Tiems, kurie kuriamų sistemų valdymui naudoja Team Foundation Server yra pateikiamas integracijos su Team Foundation Server administravimo funkcionalumas.

6.3.2.1 Sistemos vartotojų administravimas

Sistemos vartotojų administravimo dalyje yra pateikiamas sistemos vartotojų sąrašas. Šiame sąrašė esantys vartotojų įrašai gali būti redaguojami pakeičiant jų rolę ar prisijungimo vardą.

6.3.2.1.1 Sistemos vartotojų sąrašas

Lango aprašymas

Sistemos vartotojų sąrašas yra pasiekiamas sistemos meniu pasirinkus punktą [Administravimas -> Vartotojų administravimas]. Sistemos vartotojų sąrašė yra pateikiami visi sukurti sistemos vartotojai nurodant jiems priskirtą rolę. Sistemos vartotojų sąrašo peržiūros langas pateikiamas 6.4 paveikslėlyje.

Pastaba: šis langas pasiekiamas administratoriaus rolę turintiems vartotojams.

Laukų aprašymas

- Vartotojo vardas – tipas: tekstinis, privalomas. Šiame stulpelyje pateikiamas sukurtos sistemos vartotojo vardas;
- Rolė - tipas: sąrašas, privalomas. Šiame stulpelyje yra nurodomas sukurtam sistemos vartotojui priskirta rolė. Rolė yra pasirenkama iš pasirenkamojo sąrašo. Sąrašė pateikiamos reikšmės: „Testuotojas“ ir „Administratorius“.

Veikimo logika

- Redaguoti – vartotojui paspaudus nuorodą [Redaguoti] pasirinkto sistemos vartotojo įrašo eilutė yra atidaroma redagavimo režime;
- Išsaugoti – vartotojui paspaudus nuorodą [Išsaugoti] atlikti pasirinkto sistemos vartotojo įrašo pakeitimai išsaugomi sistemoje;
- Naujas – vartotojui paspaudus mygtuką [Naujas] yra atidaroma naujo sistemos vartotojo sukūrimo forma, kurioje vartotojas nurodo naujojo sistemos vartotojo vardą bei prisijungimo prie sistemos slaptažodį. Naujo sistemos vartotojo sukūrimas aprašomas 6.3.2.1.2 skyriuje.

Vartotojų administravimas		
		<input type="button" value="Naujas"/>
Vartotojo vardas	Rolė	
ADMIN	Administratorius	Išsaugoti
jusgre	Testuotojas	Redaguoti

6.4 pav. Sistemos vartotojų sąrašas

Pateikiamas funkcijų sąrašas, kurias gali atlikti testuotojas ir administratorius

6.3 lentelė. Rolėms pasiekiamos funkcijos

Administratoriaus rolės atliekamos funkcijos	Testuotojo rolės atliekamos funkcijos
Peržiūrėti veiksmų scenarijų sąrašą	Peržiūrėti veiksmų scenarijų sąrašą
Sukurti naują vartotojo veiksmų scenarijų	Sukurti naują vartotojo veiksmų scenarijų
Vykdyti pasirinktus vartotojo veiksmų scenarijus	Vykdyti pasirinktus vartotojo veiksmų scenarijus
Sukurti automatinių veiksmų scenarijų	Sukurti automatinių veiksmų scenarijų
Registruoti klaidą	Registruoti klaidą
Peržiūrėti/sukurti vartotojų sąrašą	-
Peržiūrėti/sukurti projektų sąrašą	-
Sukurti/redaguoti TFS nustatymus	-
Peržiūrėti/sukurti sistemos klasifikatorių įrašus	-

6.3.2.1.2 Naujo sistemos vartotojo sukūrimas

Lango aprašymas

Naujo vartotojo sukūrimo forma yra atidaroma sistemos vartotojų sąrašo lange paspaudus mygtuką [Naujas]. Naujo vartotojo sukūrimo metu yra nurodomas naujojo sistemos vartotojo

vardas bei prisijungimo prie sistemos slaptažodis. Pagal nutylėjimą naujam vartotojui suteikiama „Administratorius“ rolė. Ši rolė gali būti pakeista sistemos vartotojų sąraše, redaguojant naujo vartotojo įrašą (sistemos vartotojų sąrašas aprašomas 6.3.2.1.1 skyriuje). Naujo sistemos vartotojo sukūrimo formos pavyzdys pateikiamas 6.5 paveikslėlyje.

Pastaba: šis langas pasiekiamas administratoriaus rolę turintiems vartotojams.

Laukų aprašymas

- Vartotojo vardas – tipas: skaitinis, privalomas. Šiame lauke nurodomas naujo sistemos vartotojo vardas;
- Slaptažodis – tipas: tekstinis, privalomas. Šiame lauke nurodomas naujo sistemos vartotojo prisijungimo slaptažodis;
- Pakartoti slaptažodis – tipas: skaitinis, privalomas. Pakartotinai nurodomas naujo sistemos vartotojo prisijungimo slaptažodis.

Pastaba: lauko [Slaptažodis] ir lauko [Pakartoti slaptažodį] reikšmės turi sutapti.

Veikimo logika

- Registruoti – vartotojui paspaudus mygtuką [Registruoti] sistemoje yra išsaugomas naujas sistemos vartotojas;
- Atšaukti – vartotojui paspaudus mygtuką [Atšaukti] yra panaikinama vartotojo įvesta informacija ir uždaroma naujo sistemos vartotojo registracijos forma.

Naujo vartotojo registracija

Vartotojo vardas:

Slaptažodis:

Pakartoti slaptažodį:

6.5 pav. Naujo sistemos vartotojo sukūrimo forma

6.3.2.2 Tiriamų projektų administravimas

Lango aprašymas

Langas „Projektų administravimas“ skirtas peržiūrėti sistemoje registruotus projektus bei sukurti naujus projektus, kurie bus tiriami naudojant TESTO. Projektų administravimo lange

yra pateikiamas projektų sąrašas, kuriame nurodomas projekto pavadinimas, projekto trumpas aprašymas bei prisijungimo informacija prie projekto duomenų bazės (prieigos pavadinimas, duomenų bazės pavadinimas, naudojamas vartotojo prisijungimo vardas, naudojamas vartotojo prisijungimo slaptažodis bei požymis ar naudojama „Windows“ vartotojo autentifikacija). Projektų administravimo lango pavyzdys pateikiamas 6.6 paveikslėlyje.

Pastaba: šis langas pasiekiamas administratoriaus rolę turintiems vartotojams.

Laukų aprašymas

- Projekto pavadinimas – tipas: tekstinis, privalomas. Vartotojas nurodo naujo projekto pavadinimą;
- Projekto aprašymas – tipas: tekstinis, neprivalomas. Vartotojas nurodo trumpą naujojo projekto aprašymą;
- Prieigos pavadinimas – tipas: tekstinis, privalomas. Vartotojas nurodo naujo projekto duomenų bazės prieigos pavadinimą (instance name);
- DB pavadinimas – tipas: tekstinis, privalomas. Vartotojas nurodo naujojo projekto duomenų bazės pavadinimą;
- Vartotojo vardas – tipas: tekstinis, privalomas. Vartotojas nurodo prie naujojo projekto duomenų bazės naudojamą prisijungimo vardą;
- Slaptažodis – tipas: tekstinis, privalomas. Vartotojas nurodo prie naujo projekto duomenų bazės naudojamą prisijungimo slaptažodį;
- Windows Authentication – tipas: požymis, neprivalomas. Vartotojas pažymi šį požymį jeigu prisijungimui prie naujojo projekto duomenų bazės yra naudojamas „Windows“ vartotojo prisijungimas.

Veikimo logika

- Įterpti – vartotojui paspaudus nuorodą [Įterpti] aktyvuojama eilutė naujo įrašo (projekto) įvedimui į projektų sąrašą;
- Redaguoti – vartotojui paspaudus nuorodą [Redaguoti] aktyvuojama pasirinkta projektų administravimo sąrašo eilutė redagavimo režime;
- Išsaugoti – vartotojui paspaudus nuorodą [Išsaugoti] išsaugomi pakeitimai atlikti pasirinktame projektų administravimo sąrašo įrašė.

Projektų administravimas							
Projekto pavadinimas	Projeto aprašymas	Prieigos pavadinimas	DB pavadinimas	Vartotojo vardas	Slaptažodis	Windows authentication	
TESTO	Testavimo atvejų vykdymo informacinė sistema	SQL2005	TESTO	sa	sa	<input checked="" type="checkbox"/>	Redaguoti
						<input type="checkbox"/>	Išterpti

6.6 pav. Projektų administravimo langas

6.3.2.3 Sistemos nustatymų administravimas

Lango aprašymas

Langas „TFS nustatymai“ skirtas administruoti TESTO sistemos ir „Team Foundation Server“ integracijos nustatymus. Šiame lange yra nurodomi visi reikalingi prisijungimui prie „Team Foundation Server“ duomenys. TFS nustatymų administravimo lango pavyzdys pateikiamas 6.7 paveikslėlyje.

Pastaba: šis langas pasiekiamas administratoriaus rolę turintiems vartotojams.

Laukų aprašymas

- TFS pavadinimas – tipas: tekstinis, privalomas. Vartotojas nurodo TFS nustatymų pavadinimą;
- TFS prisijungimas – tipas: tekstinis, privalomas. Vartotojas nurodo prisijungimo prie „Team Foundation Server“ duomenys („connection string“);
- TFS aprašymas – tipas: tekstinis, neprivalomas. Vartotojas nurodo svarbias pastabas apie nurodytus duomenis.

Veikimo logika

- Išsaugoti - vartotojui paspaudus mygtuką [Išsaugoti] sistemoje išsaugomi visi vartotojo nurodyti prisijungimo prie „Team Foundation Server“ duomenys;
- Atšaukti – vartotojui paspaudus mygtuką [Atšaukti] yra panaikinama vartotojo nurodyta informacija ir uždaromas „TFS nustatymai“ langas.

TFS nustatymai

TFS pavadinimas:

TFS prisijungimas: ▲▼

TFS aprašymas: ▲▼

Išsaugoti
Atšaukti

6.7 pav. TFS nustatymų administravimo langas

6.3.3. Veiksmų scenarijai

Veiksmų scenarijų dalis – tai pagrindinė TESTO sistemos dalis. Šioje dalyje vartotojas sukuria veiksmų scenarijus tiriamoms sistemoms, kurių pagrindu yra formuojami automatinių veiksmų scenarijai. Taip pat šioje dalyje yra vykdomas pagrindinis veiksmų scenarijų administravimas – vykdymas. Veiksmų scenarijų dalį sudaro šios funkcijos:

- Vartotojo veiksmų scenarijų sąrašo peržiūra;
- Naujo vartotojo veiksmų scenarijaus sukūrimas;
- Vartotojo veiksmų scenarijaus pasirinkimas ir vykdymas;
- Automatinio veiksmų scenarijaus sukūrimas;
- Automatinių veiksmų scenarijaus sąrašo peržiūra;
- Automatinio veiksmų scenarijaus pasirinkimas ir vykdymas;
- Nustatytų tiriamos sistemos klaidų registravimas.

Detaliau apie kiekvieną iš šių funkcijų informacija pateikiama 6.3.3.1 - 6.3.3.5 skyriuose.

6.3.3.1 Vartotojo veiksmų scenarijų sąrašas

Lango aprašymas

Langas „Vartotojo veiksmų scenarijų sąrašas“ skirtas peržiūrėti TESTO sistemoje registruotus veiksmų scenarijus. Veiksmų scenarijų sąrašas yra filtruojamas pagal pasirinktą projektą. Taip pat šalia kiekvieno veiksmų scenarijaus esančio vartotojo veiksmų scenarijų sąraše yra pateikiama informacija apie sistemos vartotoją, kuriam yra priskirtas veiksmų scenarijus bei pateikiama veiksmų scenarijaus sukūrimo data. Vartotojo veiksmų scenarijų sąrašo lango pavyzdys pateikiamas 6.8 paveikslėlyje.

Pastaba: šis langas pasiekiamas administratoriaus ir testuotojo rolę turintiems vartotojams.

Laukų aprašymas

- Projektas – tipas: sąrašas, privalomas. Vartotojas iš pasirenkamojo sąrašo pasirenka projektą, kurio veiksmų scenarijai yra pateikiami suformuotame sąrašė. Pasirenkamajame sąrašė pateikiami visi TESTO sistemoje registruoti projektai, kuriem yra sukurti veiksmų scenarijai;
- Veiksmų scenarijaus pavadinimas – pateikiamas pasirinktam projektui sukurto veiksmų scenarijaus pavadinimas;
- Vartotojas – pateikiamas sistemos vartotojo informacija, kuriam yra priskirtas pasirinkto projekto veiksmų scenarijus;
- Sukūrimo data – pateikiama pasirinkto projekto sukūrimo data.

Veikimo logika

- Naujas scenarijus – vartotojui paspaudus mygtuką [Naujas scenarijus] yra atidaromas naujo veiksmų scenarijaus sukūrimo langas. Naujo veiksmų scenarijaus sukūrimo lango aprašymas pateikiamas 6.3.3.2 skyriuje;
- Peržiūrėti – vartotojui paspaudus nuorodą [Peržiūrėti] yra atidaromas veiksmų scenarijaus sukūrimo langas su užpildyta pasirinkto veiksmų scenarijaus informacija. Veiksmų scenarijaus peržiūros metu vartotojas gali atlikti informacijos redagavimo veiksmus;
- Šalinti – vartotojui paspaudus nuorodą [Šalinti] yra pašalinamas pasirinktas veiksmų scenarijus iš TESTO sistemos.

Vartotojo veiksmų scenarijų sąrašas

Projektas:

Veiksmų scenarijaus pavadinimas	Vartotojas	Sukūrimo data	
TESTO_Login	jusgre	2009-10-15	Peržiūrėti šalinti
...	
TESTO_Administration	jusgre	2009-10-15	

6.8 pav. Vartotojo veiksmų scenarijaus sąrašas

6.3.3.2 Naujo vartotojo veiksmų scenarijaus sukūrimas

Lango aprašymas

Langas „Naujas veiksmų scenarijus“ skirtas sukurti naują veiksmų scenarijų. Naujo veiksmų scenarijaus sukūrimo metu yra nurodomas projektas, su kuriuo yra susiejamas veiksmų scenarijus, priskiriamas vartotojas, nurodoma veiksmų scenarijaus pavadinimo, sukūrimo bei vykdymo sąlygų informacija. Taip pat suformuojamas kuriamo veiksmų scenarijaus vykdymo užduočių sąrašas (nurodant atliekamus žingsnius ir laukiamus šių žingsnių rezultatus). Naujo veiksmų scenarijaus sukūrimo lango pavyzdys pateikimas 6.9 paveikslėlyje.

Pastaba: šis langas pasiekiamas administratoriaus ir testuotojo rolę turintiems vartotojams.

Laukų aprašymas

- Projektas – tipas: sąrašas, privalomas. Vartotojas iš pasirenkamojo projektų sąrašo projektą, su kuriuo yra susiejamas naujasis veiksmų scenarijus. Pagal nutylėjimą nustatomas projektas, kuris buvo pasirinktas veiksmų scenarijų sąrašė;
- Atliekamas vartotojo – tipas: sąrašas, privalomas. Vartotojas iš pasirenkamojo sąrašo pasirenka sistemos vartotoją, kuriam yra priskiriamas atlikti naujasis veiksmų scenarijus. Šiame sąrašė pateikiami visi sistemoje registruoti vartotojai, kuriems yra priskirta rolė „Testuotojas“;
- Veiksmų scenarijaus pavadinimas – tipas: tekstinis, privalomas. Vartotojas nurodo naujojo veiksmų scenarijaus pavadinimą;
- Scenarijaus sukūrimo data – tipas: data, privalomas. Nurodoma veiksmų scenarijaus sukūrimo data, pagal nutylėjimą pateikiama einamosios dienos data;
- Scenarijaus sąlygos – tipas: tekstinis, neprivalomas. Vartotojas nurodo reikalingas sąlygas naujojo veiksmų scenarijaus atlikimui;

Scenarijaus veiksmi:

- Scenarijaus žingsnis – tipas: tekstinis, privalomas. Vartotojas nurodo naujojo veiksmų scenarijaus atlikimo žingsnius;
- Laukiamas rezultatas – tipas: tekstinis, privalomas. Vartotojas nurodo kokie yra laukiami rezultatai kiekvieno atliekamo žingsnio.

Veikimo logika

- Naujas žingsnis – vartotojui paspaudus mygtuką [Naujas žingsnis] yra sukuriama nauja eilutė informacijos įvedimui scenarijaus veiksmų dalyje;
- Išsaugoti – vartotojui paspaudus nuorodą [Išsaugoti], esančią scenarijaus veiksmo eilutėje, pasirinktos scenarijaus eilutės informacija išsaugoma sistemoje;
- Atšaukti – vartotojui paspaudus nuorodą [Atšaukti], esančią scenarijaus eilutėje, pasirinktos scenarijaus eilutės informacija pašalina ir nauja eilutė scenarijaus dalyje panaikinama;
- Išsaugoti veiksmų scenarijų – vartotojui paspaudus mygtuką [Veiksmų scenarijų] sistemoje išsaugomas naujas veiksmų scenarijaus įrašas bei su juo susietas veiksmų sąrašas.

Naujas veiksmų scenarijus

Projektas:

Atliekamas vartotojo:

Scenarijaus pavadinimas:

Scenarijaus sukūrimo data:

Scenarijaus sąlygos:

Scenarijaus veiksmai:

Scenarijaus žingsnis	Laukiamas rezultatas	
Prisijungti prie sistemos	Prisijungta prie sistemos	Išsaugoti Atšaukti

6.9 pav. Naujo vartotojo veiksmų scenarijaus sukūrimo langas

6.3.3.3 Vartotojo veiksmų scenarijų pasirinkimas ir vykdymas

Lango aprašymas

Langas „Veiksmų scenarijų vykdymas“ skirtas peržiūrėti sistemoje registruotus pasirinkto projekto veiksmų scenarijus. Vartotojui pasirinkus projektą, veiksmų scenarijų sąrašas yra pateikiami veiksmų scenarijai ir nurodoma jų informacija (veiksmų scenarijaus apvaadinimas,

veiksmų scenarijaus sukūrimo data, paskutinio vykdymo data bei paskutinio vykdymo rezultatas). Vartotojas pažymi pasirinktus veiksmų scenarijus, kurie bus vykdomi. Vartotojui paspaudus mygtuką [Vykdyti pažymėtus scenarijus], apatinėje lango dalyje pateikiama pasirinkto (-ų) veiksmų scenarijaus informacija. Vartotojui įvertinus pateiktą veiksmų scenarijų (nustačius rezultata) ir paspaudus mygtuką [Išsaugoti] sistemoje išsaugomas pateiktos veiksmų scenarijaus rezultatas bei pateikiama sekančio pažymėto veiksmų scenarijaus informacija. Jeigu vartotojas prieš pasirenkant veiksmų scenarijus planuojamam vykdymui buvo pažymėjęs požymį „Sukurti automatinį veiksmų scenarijų“, tokiu atveju po veiksmų scenarijaus atlikimo (paspaudus mygtuką [Išsaugoti]) išsaugomas ir vykdyto veiksmų scenarijaus automatinis veiksmų scenarijus. Veiksmų scenarijų vykdymo lango pavyzdys pateikiamas 6.10 paveikslėlyje.

Pastaba: šis langas pasiekiamas administratoriaus ir testuotojo rolę turintiems vartotojams.

Laukų aprašymas

- Projektas – tipas: sąrašas, privalomas. Vartotojas iš pasirinktųjų projektų sąrašo pasirenka projektą, kurio veiksmų scenarijai bus pateikiami formuojamame veiksmų scenarijų sąrašė;
- Sukurti automatinį veiksmų scenarijų – tipas: požymis, neprivalomas. Vartotojas pažymi šį požymį, jeigu pasirinktųjų veiksmų scenarijų vykdymo metu bus kuriamas automatinis veiksmų scenarijus. Pagal nutylėjimą šis požymis nepažymėtas.

Veiksmų scenarijų sąrašas

- Eil. Nr. – tipas: skaitinis, neredaguojamas. Pateikiamas į suformuotą veiksmų scenarijų sąrašą įtraukto veiksmų scenarijaus eilės numeris
- Scenarijaus pavadinimas – tipas: tekstinis, neredaguojamas. Pateikiamas į suformuotą veiksmų scenarijų sąrašą įtraukto veiksmų scenarijaus pavadinimas;
- Sukūrimo data – tipas: data, neredaguojamas. Pateikiama į suformuotą veiksmų scenarijų sąrašą įtraukto veiksmų scenarijaus sukūrimo data;
- Paskutinio vykdymo data – tipas: data, neredaguojamas. Pateikiama į suformuotą veiksmų scenarijų sąrašą įtraukto veiksmų scenarijaus paskutinio vykdymo data;

- Paskutinio vykdymo rezultatas – tipas: tekstinis, neredaguojamas. Pateikiamas į suformuotą veiksmų scenarijų sąrašą įtraukto veiksmų scenarijaus paskutinio vykdymo rezultatas;
- Ar vykdyti – tipas: požymis, neprivalomas. Vartotojas pažymi šį požymį, jeigu pasirinktas veiksmų scenarijus bus vykdomas;

Vykdomas testas

- Sukūrimo data – tipas: data, neredaguojamas. Pateikiama vykdomo scenarijaus sukūrimo data;
- Paskutinio vykdymo data – tipas: data, neredaguojamas. Pateikiama vykdomo veiksmų scenarijaus paskutinio vykdymo data;
- Paskutinio vykdymo rezultatas – tipas: tekstinis, neredaguojamas. Pateikiamas vykdomo veiksmų scenarijaus paskutinio vykdymo rezultatas;
- Paskutinio vertinimo komentaras – tipas: tekstinis, neredaguojamas. Pateikiamas vykdomo veiksmų scenarijaus paskutinio vykdymo vertinimo komentaras;
- Scenarijaus vykdymo sąlygos – tipas: tekstinis, neredaguojamas. Pateikiamas vykdomo veiksmų scenarijaus sąlygų aprašymas;
- Papildoma informacija – tipas: nuoroda, neredaguojamas. Pateikiama nuoroda į vykdomo veiksmų scenarijaus papildomos informacijos dokumentą;
- Scenarijaus veiksmas (Eil. Nr.) – tipas: skaitinis, neredaguojamas. Pateikiamas scenarijuje nurodyti veiksmo eilės numeris;
- Scenarijaus veiksmas (Atliekamas veiksmas) – tipas: tekstinis, neredaguojamas. Pateikiamas vykdomo scenarijuje nurodytas veiksmas;
- Scenarijaus veiksmas (Laukiamas rezultatas) – tipas: tekstinis, neredaguojamas. Pateikiamas vykdomo scenarijuje nurodytas laukiamas rezultatas;
- Scenarijaus įvertinimas – tipas: sąrašas, privalomas. Vartotojas iš pasirenkamojo sąrašo pasirenka vykdomo scenarijaus įvertinimą. Scenarijaus įvertinimų reikšmės pasirenkamajame sąraše pateikiamos iš scenarijaus įvertinimo reikšmių klasifikatoriaus;
- Scenarijaus įvertinimo komentaras – tipas: tekstinis, neprivalomas. Vartotojas nurodo vykdyto scenarijaus pastabas/komentarus.

Veikimo logika

- Vykdyti pažymėtus scenarijus – vartotojui paspaudus mygtuką [Vykdyti pažymėtus scenarijus] pateikiama pasirinkto veiksmų scenarijaus informacija lango dalyje „Vykdomas scenarijus“;
- Registruoti klaidą – vartotojui paspaudus mygtuką [Registruoti klaidą] atidaroma klaidos registravimo forma, kurioje vartotojas nurodo su klaidos registravimu susijusią informaciją ir išsaugo klaidą vieningoje klaidų registravimo sistemoje;
- Išsaugoti – vartotojui paspaudus mygtuką [Išsaugoti] yra išsaugomas vykdyto veiksmų scenarijaus rezultatas ir nurodyti vykdyto veiksmų scenarijaus komentarai. Jeigu prieš pradėdant pasirinkto veiksmų scenarijaus vykdymą buvo pažymėtas požymis „Sukurti automatinį veiksmų scenarijų“, tokiu atveju išsaugomas ir automatinių veiksmų scenarijus. Išsaugotus automatinių veiksmų scenarijus vartotojas gali peržiūrėti scenarijų vykdymo posistemėje, automatinių veiksmų scenarijų dalyje;
- Atšaukti – vartotojui paspaudus mygtuką [Atšaukti] nutraukiamas pasirinkto scenarijaus vykdymas. Jeigu buvo pažymėtas požymis „Sukurti automatinių veiksmų scenarijų“, tokiu atveju vykdyto veiksmų scenarijaus automatinių veiksmų scenarijus neišsaugomas. Jeigu testų vykdymui buvo pažymėti keli scenarijai, tokiu atveju vartotojui paspaudus mygtuką [Atšaukti], vykdymui yra pateikiamas sekantis veiksmų scenarijus.

Scenarijų vykdymas

Projektas: TESTO Sukurti automatinių veiksmų scenarijų

Veiksmų scenarijų sąrašas

Eil. Nr.	Veiksmų scenarijų pavadinimas	Sukūrimo data	Paskutinio vykdymo data	Paskutinio vykdymo rezultatas	Ar vykdyti?
1.	TESTO_Login	2009-09-25	2009-10-15	Neišlaikytas	<input checked="" type="checkbox"/>
...	<input type="checkbox"/>
17.	TESTO_Administration	2009-09-25	-	-	<input checked="" type="checkbox"/>

Vykdyti pažymėtus scenarijus

Vykdomas scenarijus: TESTO_Login

Sukūrimo data: **2009-09-25**

Paskutinio vykdymo data: **2009-10-15**

Paskutinio vykdymo rezultatas: **Neišlaikytas**

Paskutinio vykdymo komentaras:

Scenarijaus vykdomo sąlygos:

Papildoma informacija: [prisegtas dokumentas.pdf](#)

Scenarijaus veiksmi:

Eil. Nr.	Atliekamas veiksmas	Laukiamas rezultatas
1.	Įvedamas sistemos prisijungimo sistemos adresas	Atidaromas sistemos prisijungimo langas
2.	Įvedamas prisijungimo vardas ir slaptažodis, inicijuojamas prisijungimas	Vartotojas prisijungia prie sistemos

Scenarijaus įvertinimas: Pasirinkti

Scenarijaus įvertinimo pastabos:

Registruoti klaidą

Išsaugoti

Atšaukti

6.10 pav. Veiksmų scenarijų vykdymo langas

6.3.3.3.1 Testavimo eigos scenarijaus sukūrimas

Automatinių veiksmų scenarijus yra kuriamas veiksmų scenarijų pasirinkimo ir jų vykdymo metu. Vartotojui pažymėjus požymį „Sukurti automatinių veiksmų scenarijų“. Vartotojui šį požymį ir vykdant pasirinktą veiksmų scenarijų, sistemoje bus išsaugotas ir testuojamam projektui priskirtas automatinių veiksmų scenarijus. Automatinių veiksmų scenarijaus pavadinimas sutaps su veiksmų scenarijaus pavadinimu.

6.3.3.4 Automatinių veiksmų scenarijų pasirinkimas ir vykdymas

Lango aprašymas

Langas „Automatinių veiksmų scenarijai“ skirtas peržiūrėti sukurtus automatinių veiksmų scenarijų sąrašą bei pasirinkti reikalingus scenarijus vykdymui. Vartotojui pasirinkus automatinių veiksmų scenarijus yra nurodoma informacija kelių vartotojų darbas bus imituojamas ir kiek skirtingų vykdymo iteracijų turi vykdyti sistema. Automatinių veiksmų scenarijų lango pavyzdys pateikiamas 6.11 paveikslėlyje.

Pastaba: šis langas pasiekiamas administratoriaus ir testuotojo rolę turintiems vartotojams.

Laukų aprašymas

- Projektas – tipas: sąrašas, privalomas. Vartotojas iš pasirenkamojo projektų sąrašo pasirenka projekto pavadinimą, kurio automatinių veiksmų scenarijai bus pateikiami formuojamame sąrašė;
- Imituojamas vartotojų sk. – tipas: skaitinis, privalomas. Vartotojas nurodo vartotojų skaičių, kelių vartotojų darbas bus imituojamas vykdant pasirinktą automatinių veiksmų scenarijų;
- Imituojamų iteracijų sk. – tipas: skaitinis, privalomas. Vartotojas nurodo iteracijų skaičių, kiek iteracijų bus vykdomi pasirinkti automatinių veiksmų scenarijai;
- Automatinių veiksmų scenarijaus pavadinimas – pateikiamas pasirinkto projekto automatinių veiksmų scenarijaus pavadinimas;
- Sukūrimo data – pateikiama data, kada buvo sukurtas pasirinkto projekto automatinių veiksmų scenarijus;
- Paskutinio vykdymo data – pateikiama data, kad paskutinį kartą buvo vykdytas pasirinktas automatinių veiksmų scenarijus;
- Paskutinio vykdymo vartotojų skaičius – pateikiamas vartotojų skaičius, kuris buvo imituojant paskutinį kartą leidžiant pasirinkta automatinių veiksmų scenarijų.

Veikimo logika

- Ar vykdyti – vartotojui pažymėjus požymį [Ar vykdyti], esantį šalia pasirinkto automatinių veiksmų scenarijaus, šis scenarijus yra įtraukiamas į vykdomų automatinių veiksmų scenarijų sąrašą;
- Vykdyti pažymėtus scenarijus – vartotojui paspaudus mygtuką [Vykdyti pasirinktus scenarijus] yra pradamas pasirinktų automatinių veiksmų scenarijų vykdymas pagal nurodytus parametrus („Imituojamų vartotojų sk.“ Ir „Imituojamų iteracijų sk.“).

Automatinių veiksmų scenarijai

Projektas: ▼

Imituojamas vartotojų sk.:

Imituojamų iteracijų sk.:

Automatinių veiksmų scenarijų sąrašas

Eil. Nr.	Automatinio veiksmų scenarijaus pavadinimas	Sukūrimo data	Paskutinio vykdymo data	Paskutinio vykdymo vartotojų sk.	Ar vykdyti?
1.	TESTO_Login	2009-10-15	-	-	<input checked="" type="checkbox"/>
...	<input type="checkbox"/>
17.	TESTO_Administration	2009-10-15	-	-	<input checked="" type="checkbox"/>

6.11 pav. Automatinių veiksmų scenarijų pasirinkimo ir vykdymo langas

6.3.3.5 Klaidos registravimas

Lango aprašymas

Langas „Naujos klaidos registravimas“ skirtas registruoti aptiktą klaidą „Team Foundation Server“ klaidų valdymo sistema. „Team Foundation Server“ nustatymai yra nurodomi sistemos administravimo dalyje (plačiau apie TFS nustatymus skaitykite 6.3.2.3 skyriuje). Naujos klaidos registravimo metu vartotojas nurodo projekto pavadinimą, testavimo atvejo pavadinimą, klaidos prioritetą bei pateikiama klaidos aprašymą. Naujos klaidos registravimo lango pavyzdys pateikiamas 6.12 paveikslėlyje.

Pastaba: šis langas pasiekiamas administratoriaus ir testuotojo rolę turintiems vartotojams.

Laukų aprašymas

- Projektas - tipas: tekstinis, neredaguojamas. Sistema nurodo projekto, kuris buvo testuojamas, pavadinimą;
- scenarijus – tipas: tekstinis, neredaguojamas. Sistema nurodo scenarijaus, kuris buvo vykdomas, pavadinimą;
- Klaidos prioritetas – tipas: sąrašas, privalomas. Vartotojas iš pasirinkamojo klaidų prioritetų žinyno sąrašo pasirenka klaidos prioriteto reikšmę. Klaidų prioritetų žinyno reikšmės pateikiamos 6.3.3.5.1 skyriuje;
- Klaidos aprašymas – tipas: tekstinis, privalomas. Vartotojas detalai pateikia aptiktos klaidos aprašymą.

Veikimo logika

- Išsaugoti – vartotojui paspaudus mygtuką [Išsaugoti] TFS klaidų valdymo dalyje išsaugoma naujos klaidos informacija;
- Atšaukti – vartotojui paspaudus mygtuką [Atšaukti] panaikinama visa vartotojo įvesta informacija, uždaromas klaidos registravimo langas ir tęsiamas pasirinktų testavimo atvejų testavimas.



6.12 pav. Klaidos registravimo langas

6.3.3.5.1 Klaidos prioriteto žinynas

Žinyno aprašymas

Klaidos prioriteto žinynas skirtas nustatyti registruojamos klaidos prioritetą.

Žinyne pateikiamos reikšmės

- Aukštas;
- Vidutinis;
- Žemas.

6.4. Testavimo rezultatai

Sistemos testavimo metu yra testuojamos atskiros sistemos funkcijos. Sistemos funkcijos yra testuojamos atsižvelgiant į reikalavimų dalyje nurodytus pagrindinius šių funkcijų reikalavimus bei pagal panaudojimo atvejų modelyje pateikiamus veiksmų scenarijus.

Pateikiami šių sistemos funkcijų testavimo aprašymas:

- Sukurti naują projektą;
- Sukurti naują vartotoją;
- Priskirti vartotojui rolę;
- Sukurti naują vartotojo veiksmų scenarijų;
- Sukurti automatinių veiksmų scenarijų

6.4 lentelė. Naujo projekto sukūrimo detalus testavimas

Eil. Nr.	Vartotojo atliekamas veiksmas	Sistemos reakcija	Laukiamas veiksmų rezultatas	Rezultatas
1.	Projektų sąraše spaudžiama nuoroda „Įterpti“	Aktyvuojama sąrašo eilutė naujo projekto informacijos įvedimui	Aktyvuojama sąrašo eilutė naujo projekto įvedimui	Teisingas
2.	Neįvedus projekto informacijos paspausti nuorodą „Išsaugoti“	Pateikiama klaida apie neįvestus privalomus laukus	Vartotojui parodomas klaidos: Neįvestas projekto pavadinimas; Neįvesta s prieigos pavadinimas; Neįvestas DB pavadinimas; Neįvestas vartotojo vardas; Neįvestas vartotojo slaptažodis.	Teisingas
3.	Įvedus tik naujo projekto pavadinimą, spaudžiama nuoroda „Išsaugoti“	Pateikiama klaida apie neįvestus privalomus laukus	Vartotojui parodomas klaidos: Neįvesta s prieigos pavadinimas; Neįvestas DB pavadinimas; Neįvestas vartotojo vardas; Neįvestas vartotojo slaptažodis.	Teisingas
4.	Įvedus naujo projekto pavadinimą, ir prieigos pavadinimą, spaudžiama nuoroda „Išsaugoti“	Pateikiama klaida apie neįvestus privalomus laukus	Vartotojui parodomas klaidos: Neįvestas DB pavadinimas; Neįvestas vartotojo vardas; Neįvestas vartotojo slaptažodis.	Teisingai

5.	Įvedus naujo projekto pavadinimą, prieigos pavadinimą ir DB pavadinimas	Pateikiama klaida apie neįvestus privalomus laukus	Vartotojui parodomos klaidos: Neįvestas vartotojo vardas; Neįvestas vartotojo slaptažodis.	Teisingai
6.	Įvedus naujo projekto pavadinimą, prieigos pavadinimą, DB pavadinimą ir vartotojo vardą yra paspaudžiama nuoroda „Išsaugoti“	Pateikiama klaida apie neįvestus privalomus laukus	Vartotojui parodomos klaidos: Neįvestas vartotojo vardas.	Teisingai
7.	Įvedus naujo projekto pavadinimą, prieigos pavadinimą, DB pavadinimą, vartotojo vardą ir vartotojo slaptažodį yra paspaudžiama nuoroda „Atšaukti“	Panaikinama naujo projekto įvedimo eilutė projektų sąrašė	Panaikinamas naujo projekto sukūrimas, grįžtama atgal į projektų sąrašą	Teisingai
8.	Įvedus naujo projekto pavadinimą (egzistuojantį), prieigos pavadinimą, DB pavadinimą, vartotojo vardą ir vartotojo slaptažodį yra paspaudžiama nuoroda „Išsaugoti“	Pateikiama klaida apie neunikalų projekto pavadinimą.	Vartotojui pateikiama klaida: Sistemoje egzistuoja projektas nurodytu pavadinimu	Teisingai
9.	Įvedus naujo projekto pavadinimą, prieigos pavadinimą, DB pavadinimą, vartotojo vardą ir vartotojo slaptažodį yra paspaudžiama nuoroda „Išsaugoti“	Projektas išsaugomas sistemoje	Panaikinama projekto įvedimo eilutė, pateikiamas atnaujintas projektų sąrašas su nauju projekto įrašu	Teisingai

6.5 lentelė. Sistemos vartotojo sukūrimo detalus testavimas

Eil. Nr.	Vartotojo atliekamas veiksmas	Sistemos reakcija	Laukiamas veiksmų rezultatas	Rezultatas
1.	Vartotojų sąrašo lange paspaudžiamas mygtukas „Naujas“	Atidaroma naujo sistemos vartotojo sukūrimo forma	Atidaroma naujo sistemos vartotojo sukūrimo forma	Teisingai
2.	Naujo vartotojo sukūrimo formoje neįvedus informacijos yra paspaudžiamas mygtukas „Išsaugoti“	Pateikiama vartotojui klaida apie neįvestą privalomą informaciją	Pateikiama klaida apie neįvestą informaciją: Neįvestas prisijungimo vardas; Neįvestas prisijungimo slaptažodis; Neįvestas pakartotinis	Teisingai

			prisijungimo slaptažodis.	
3.	Naujo vartotojo sukūrimo formoje įvedus tik naujo vartotojo prisijungimo vardą yra spaudžiamas mygtukas „Išsaugoti“	Pateikiama vartotojui klaida apie neįvestą privalomą informaciją	Pateikiama klaida apie neįvestą informaciją: Neįvestas prisijungimo slaptažodis; Neįvestas pakartotinis prisijungimo slaptažodis.	Teisingai
4.	Naujo vartotojo sukūrimo formoje įvedus naujo vartotojo prisijungimo vardą (jau egzistuojantis), vartotojo prisijungimo slaptažodį, bei pakartotinį slaptažodį ir paspaudus mygtuką „Išsaugoti“	Pateikiama klaida apie sistemoje egzistuojantį vartotoją	Pateikiama klaida apie įvestą neunikalų vartotojo vardą	Teisingai
5.	Naujo vartotojo sukūrimo formoje įvedus naujo vartotojo prisijungimo vardą, vartotojo prisijungimo slaptažodį, bei pakartotinį slaptažodį ir paspaudus mygtuką „Atšaukti“	Panaikinama įvesta informacija ir uždaroma naujo sistemos vartotojo sukūrimo forma	Panaikinama įvesta informacija ir uždaroma naujo sistemos vartotojo sukūrimo forma	Teisingai
6.	Naujo vartotojo sukūrimo formoje įvedus naujo vartotojo prisijungimo vardą, vartotojo prisijungimo slaptažodį, bei pakartotinį slaptažodį ir paspaudus mygtuką „Išsaugoti“	Įvesta informacija išsaugoma sistemoje ir pateikiamas sistemos vartotojų sąrašas	Uždaroma naujo vartotojo sukūrimo forma ir pateikiamas sistemos vartotojų sąrašas su naujo vartotojo įrašu	Teisingai

6.6 lentelė. Rolės suteikimas pasirinktam sistemos vartotojui

Eil. Nr.	Vartotojo atliekamas veiksmas	Sistemos reakcija	Laukiamas veiksmų rezultatas	Rezultatas
1.	Vartotojų sąrašo lange, pasirinktam sistemos vartotojui paspaudžiama nuoroda „Redaguoti“	Atidaroma pasirinkto sistemos vartotojo redagavimo forma	Atidaroma pasirinkto sistemos vartotojo redagavimo forma	Teisingai
2.	Pakeitus pasirinkto sistemos vartotojo rolę, paspausti mygtuką „Atšaukti“	Sistemoje neišsaugomi pakeitimai ir vartotojui pateikiamas sistemos vartotojų sąrašas su nepakitusia informacija	Uždaroma pasirinkto sistemos vartotojo informacijos redagavimo forma ir pateikiamas sistemos vartotojų sąrašas su nepakitusia	Teisingai

			informacija	
3.	Pakeitus pasirinkto sistemos vartotojo rolę, paspausti mygtuką „Išsaugoti“	Sistemoje išsaugomi pakeitimai ir vartotojui pateikiamas sistemos vartotojų sąrašas su atliktais pakeitimais	Uždaroma pasirinkto sistemos vartotojo informacijos redagavimo forma ir pateikiamas sistemos vartotojų sąrašas su atliktai pakeitimais	Teisingai

6.7 lentelė. Vartotojo veiksmų scenarijaus sukūrimo detalus testavimas

Eil. Nr.	Vartotojo atliekamas veiksmas	Sistemos reakcija	Laukiamas veiksmų rezultatas	Rezultatas
1.	Vartotojo veiksmų scenarijų sąrašė paspaudžiamas mygtukas „Naujas“	Atidaromas naujo vartotojo veiksmų scenarijaus sukūrimo forma	Atidaromas naujo vartotojo veiksmų scenarijaus sukūrimo forma	Teisingai
2.	Atsidariusiame naujo vartotojo veiksmų scenarijaus sukūrimo formoje neįvedus informacijos spaudžiamas mygtukas „Išsaugoti“	Pateikiama klaida apie neįvestą informaciją	Pateikiama klaida apie neįvestą informaciją: Nepasirinktas tiriamas projektas; Nepasirinktas vartotojas; Neįvestas scenarijaus pavadinimas; Neįvesti scenarijaus veiksmi;	Teisingai
3.	Naujo vartotojo veiksmų scenarijaus formoje pasirenkamas tiriamasis projektas ir spaudžiamas mygtukas „Išsaugoti“	Pateikiama klaida apie neįvestą informaciją	Pateikiama klaida apie neįvestą informaciją: Nepasirinktas vartotojas; Neįvestas scenarijaus pavadinimas; Neįvesti scenarijaus veiksmi;	Teisingai
4.	Naujo vartotojo veiksmų scenarijaus formoje pasirenkamas tiriamasis projektas ir atsakingas vartotojas ir spaudžiamas mygtukas „Išsaugoti“	Pateikiama klaida apie neįvestą informaciją	Pateikiama klaida apie neįvestą informaciją: Neįvestas scenarijaus pavadinimas; Neįvesti scenarijaus veiksmi;	Teisingai
5.	Naujo vartotojo veiksmų scenarijaus formoje pasirenkamas tiriamasis projektas, atsakingas vartotojas ir nurodomas veiksmų scenarijaus pavadinimas ir spaudžiamas mygtukas „Išsaugoti“	Pateikiama klaida apie neįvestą informaciją	Pateikiama klaida apie neįvestą informaciją: Neįvesti scenarijaus veiksmi;	Teisingai
6.	Naujo vartotojo veiksmų scenarijaus formoje pasirenkamas tiriamasis projektas, atsakingas vartotojas	Pateikiama klaida apie neįvestą informaciją ir neunikalų scenarijaus pavadinimą	Pateikiama klaida apie neįvestą informaciją ir neunikalų scenarijaus pavadinimą: • Neįvesti	Teisingai

	ir nurodomas veiksmų scenarijaus pavadinimas (egzistuojantis pasirinktam tiriamajam projektui) ir spaudžiamas mygtukas „Išsaugoti“		scenarijaus veiksmi; • Neunikalus veiksmų scenarijaus pavadinimas;	
7.	Naujo vartotojo veiksmų scenarijaus formoje pasirenkamas tiriamasis projektas, atsakingas vartotojas ir nurodomas veiksmų scenarijaus pavadinimas bei veiksmų sąrašas ir spaudžiamas mygtukas „Išsaugoti“	Išsaugomas naujas vartotojo veiksmų scenarijus sistemoje ir pateikiamas veiksmų scenarijų sąrašas su nauju įrašu	Uždaromas naujo veiksmų scenarijaus sukūrimo forma ir pateikiamas vartotojo veiksmų scenarijaus sąrašas su nauju įrašu	Teisingai

6.8 lentelė. Automatinio veiksmų scenarijaus sukūrimo detalus testavimas

Eil. Nr.	Vartotojo atliekamas veiksmas	Sistemos reakcija	Laukiamas veiksmų rezultatas	Rezultatas
1.	Vartotojo veiksmų scenarijų sąraše pažymėti požymį „Sukurti automatinių veiksmų scenarijų“	Pažymimas požymis „Sukurti automatinių veiksmų scenarijų“	Pažymimas požymis „Sukurti automatinių veiksmų scenarijų“	Teisingai
2.	Pasirinkti vartotojų veiksmų scenarijus, kurių pagrindu bus kuriami automatinių veiksmų scenarijai	Pažymi pasirinkti vartotojo veiksmų scenarijai	Pažymi pasirinkti vartotojo veiksmų scenarijai	Teisingai
3.	Vykdomi pasirinkti vartotojo veiksmų scenarijai	Išsaugojus vykdomo vartotojo veiksmų scenarijaus rezultatą (teigiamą) sukuriamas automatinių veiksmų scenarijus	Išsaugojus vykdomo vartotojo veiksmų scenarijaus rezultatą (teigiamą) sukuriamas automatinių veiksmų scenarijus	Teisingai
4.	Atidaromas automatinių veiksmų scenarijų sąrašas	Pateikiamas automatinių veiksmų scenarijų, sukurtų vykdytų vartotojo veiksmų scenarijų pagrindu, sąrašas	Pateikiamas automatinių veiksmų scenarijų, sukurtų vykdytų vartotojo veiksmų scenarijų pagrindu, sąrašas	Teisingai

Visos sistemos testavimas atliktas pagal pateiktą scenarijų:

Testo pradžioje yra sukuriamas naujas tiriamojo projekto įrašas administravimo posistemėje, projektų dalyje. Pasirinktai tiriamajai sistemai yra sukuriama vartotojo veiksmų scenarijai, kurie susiejami su naujuoju tiriamuoju projektu. Šių vartotojo veiksmų scenarijų

vykdymo metu yra registruojamos klaidos, klaidų registravimo posistemėje (Team Foundation Server). Atliekamas pakartotinis vartotojo veiksmų scenarijų vykdymas (šiuo metu inicijuotas automatinių veiksmų scenarijų sukūrimas). Peržiūrimas automatinių veiksmų scenarijų sąrašas. Tikrinama ar šis sąrašas atitinka vykdytų vartotojo veiksmų scenarijų sąrašą. Pasirenkami automatinių veiksmų scenarijai ir inicijuojamas jų vykdymas. Pagal gaunamus rezultatus formuojamos ataskaitos.

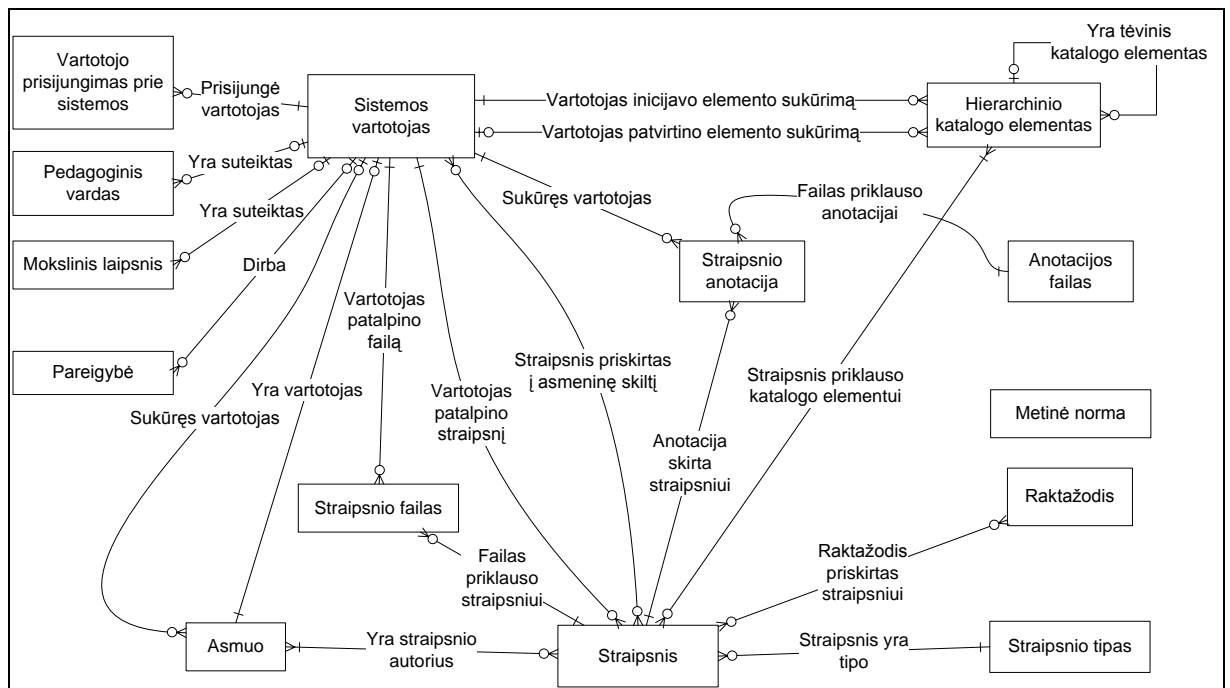
Testavimo rezultatas: Vykdydamas testavimą pagal sudarytą scenarijų sistema veikė teisingai, gauti rezultatai sutapo su numatytais rezultatais. Testavimo rezultatas teigiamas.

7. Pavyzdinis duomenų bazės našumo tyrimas

7.1. Tyrimo aprašymas

Eksperto metu yra analizuojamas ir tikrinamas duomenų bazių našumo tyrimo įrankis, siekiant įvertinti šio įrankio tikslumą, patikimumą, išskirti šio įrankio naudojimo sąlygas bei nustatyti veikimo apribojimus efektyvumo bei paprastumo atžvilgiu. Įrankis vertinamas iš vartotojo perspektyvos atliekant bandomąjį tyrimą analizuojant Informacijos Sistemų katedros vidinio portalo sistemą bei jos duomenų bazę.

Tiriamoji sistema yra Informacijos sistemų katedros vidinis publikacijų portalas su ataskaitų ir analizės priemonėmis. Ši sistema yra vertikalaus tipo (t.y. sistema pritaikyta konkrečiai veiklai - katedros mastu organizuotam mokslinės literatūros talpinimui, skaitymui ir analizavimui.). Prieš atliekant eksperimentą yra svarbu gerai suprasti tiriamąją sistemą, jos duomenų struktūrą bei duomenų srautus. Informacijos sistemų katedros vidinio publikacijų portalo duomenų modelis yra pateikiamas (7.1) paveikslėlyje.



7.1 pav. Duomenų modelis

Informacijos sistemų katedros vidinio publikacijų portalo tyrimo metu bus sukurti vartotojų veiksmų scenarijai, kurių metu bus

Šio eksperimento metu yra tikrinamos dvi hipotezės:

- Nulinė hipotezė (H_0) teigia, kad duomenų bazių našumo tyrimo įrankis netinkamas tirti dideles duomenų bazes;
- Alternatyvi hipotezė (H_a) teigia, kad duomenų bazių našumo tyrimo įrankis tinkamas tirti dideles duomenų bazes, turinčias daugiau nei 500 vartotojų.

Eksperimento metu yra siekiama atlikti Kauno Technologijos Universiteto Informacijos sistemų katedros vidinio publikacijų portalo našumo tyrimą. Šio eksperimento metu yra siekiama ne tik išskirti publikacijų portalo problemines vietas, bet ir nustatyti tyrimo įrankio galimybių ribas, patikrinti ar šis įrankis yra tinkamas tirti dideles (daugiau kaip 500 vartotojų) sistemas.

Eksperimento metu publikacijų portalui tirti yra sukuriami du vartotojo veiksmų scenarijai (naujo autoriaus sukūrimas bei naujo publikacijos tipo sukūrimas). Šiems vartotojo veiksmų scenarijams yra sukuriami automatinių veiksmų scenarijai.

Eksperimento metu yra siekiama gauti šiuos rezultatus:

- Tyrimo įrankio darbo laikas vykdant automatinų veiksmų scenarijų (šie rezultatai turi atskleisti tyrimo įrankio veikimo ribas (maksimalus vartotojų skaičius, kuris gali būti imituojamas));
- Kiekvieno bandymo metu ilgiausiai vykdyta procedūra ir jos vykdymo, CPU apkrova, loginių skaitymų ir įrašymų skaičius (šie rezultatai turi parodyti kaip kinta tiriamosios sistemos našumas didinant jos apkrovą).

Atliekamas eksperimentas yra skaidomas į 12 dalių. Eksperimentui bus keičiami du nepriklausomi kintamieji – imituojamų vartotojų skaičius ir automatinų veiksmų scenarijus. Šio eksperimento metu priklausomi kintamieji yra šie: automatinų veiksmų scenarijaus vykdymo laikas, ilgiausiai vykdytos procedūra, jos vykdymo laikas, CPU užimtumas, loginių nuskaitymų ir įrašymų skaičius.

Eksperimentas atliekamas pagal (7.1) lentelėje pateikiamą tvarką. Siekiant tikslumo, kiekvienas eksperimentas yra atliekamas 30 kartų.

7.1 lentelė. Eksperimento vykdymo tvarka

Eksperimento numeris	Imituojamas vartotojų skaičius	Eksperimento žymėjimas
1	100	E ₁
2	250	E ₂
3	500	E ₃
4	1000	E ₄
5	2500	E ₅

Eksperimento rezultatų (priklausomų kintamųjų) aritmetinis vidurkis skaičiuojamas pagal formulę:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$$

7.1 formulė. Aritmetinio vidurkio skaičiavimo formulė

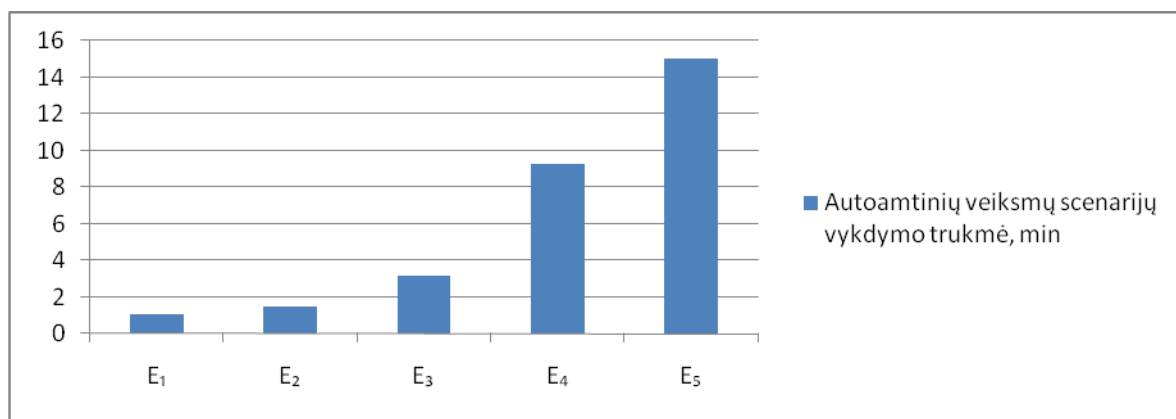
Eksperimento rezultatų (priklausomų kintamųjų) standartinis nuokrypis skaičiuojamas pagal formulę:

$$\sigma = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2}$$

7.2 formulė. Standartinio nuokrypio skaičiavimo formulė

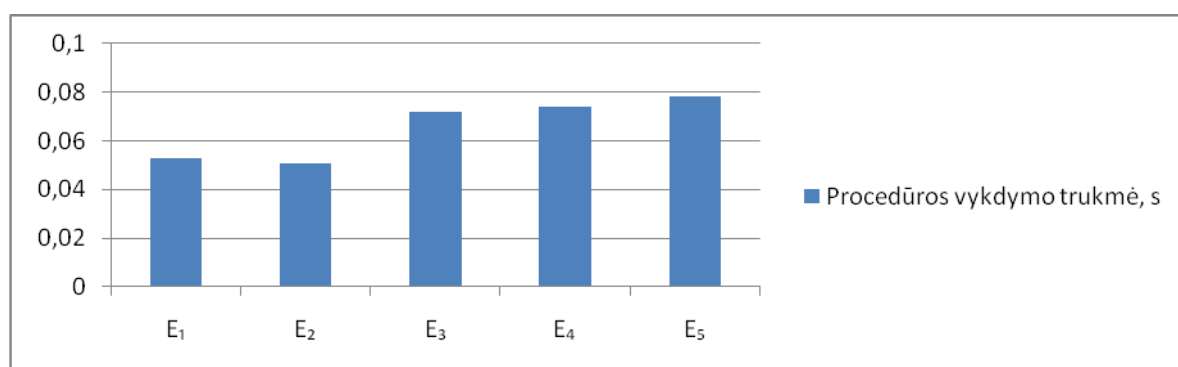
7.2. Tyrimo rezultatai

Atliekant eksperimentus buvo registruojami kiekvieno eksperimento bei jo bandymo priklausomų kintamųjų (automatinių veiksmų scenarijaus, ilgiausios procedūros, CPU užimtumo, loginių nuskaitymų bei įrašymų) reikšmės.



7.2 pav. Eksperimentų rezultatai (automatinių veiksmų scenarijaus vykdymo laikas, min)

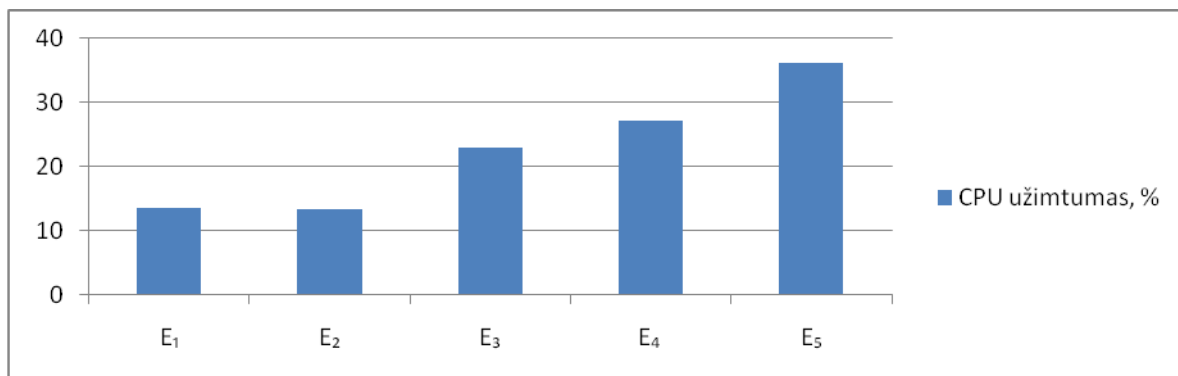
Pagal (7.2) paveikslėlyje pateiktą diagramą matyti, kad didėjant vartotojų skaičiui (E₁ – E₅) didėja ir automatinių veiksmų scenarijaus vykdymo laikas. Tačiau pagrindinis vykdymo laiko pokytis fiksuojamas po E₃ eksperimento, t.y. imituojamų vartotojų skaičiui viršijus 500 vartotojų.



7.3 pav. Eksperimentų rezultatai (procedūros vykdymo trukmė, s)

Pateiktame (7.3) paveikslėlyje yra pateikiamas eksperimentų metu užfiksuotų ilgiausiai vykdytų procedūrų vykdymo laikų pokyčių grafikas. Kaip ir automatinių veiksmų scenarijų vykdymo laikų atveju taip ir procedūrų vykdymo trukmės grafike didesnis laiko pokytis fiksuojamas viršijus 500 imituojamų vartotojų skaičių (eksperimentas E₃).

Pateiktame (7.4) paveikslėlyje matyti kompiuterio, kuriame buvo vykdomos procedūros CPU užimtumas procentais. Kaip ir ankstesniuose priklausomų kintamųjų rezultatų grafikuose, taip ir šiame žymesnis reikšmių pokytis yra fiksuojamas po (E_3) eksperimento, t.y. viršijus 500 imituojamų vartotojų skaičių.



7.4 pav. Eksperimentų rezultatai (CPU užimtumas, %)

Kiekvieno eksperimento visų bandymų rezultatai yra pateikiami šio darbo prieduose (0.1 - 0.5) lentelėse.

Kiekvieno eksperimento priklausomų kintamųjų aritmetiniai vidurkiai bei standartiniai nuokrypai yra pateikiami (7.2) lentelėje.

7.2 lentelė. Eksperimentų apibendrinti rezultatai

Eilės nr.	Nepriklausomi kintamieji		Priklausomi kintamieji									
	Imituojamų vartotojų sk.	Automatinių veiksmų scenarijus	Automatinio veiksmų scenarijaus vykdymo trukmė	Procedūros pavadinimas	Ilgiausiai trukusi procedūra							
					Vykdymo trukmė		CPU užimtumas		Loginių nuskaitymų skaičius		Loginių įrašymų skaičius	
					\bar{x}^1	σ^2	\bar{x}^1	σ^2	\bar{x}^1	σ^2	\bar{x}^1	σ^2
1	100	1	1,033	exec [persons_insert]	0,053	0,035	13,533	2,374	4,467	0,860	1,000	0,000
2	250	1	1,451	exec [persons_insert]	0,051	0,029	13,400	2,094	4,867	1,252	1,000	0,000
3	500	1	3,148	exec [persons_insert]	0,072	0,185	23,033	2,442	5,467	1,737	1,000	0,000
4	1000	1	9,238	exec [persons_insert]	0,074	0,010	27,100	2,139	6,067	1,780	1,000	0,000
5	2500	1	14,978	exec [persons_insert]	0,078	0,014	36,100	2,123	6,600	1,499	1,000	0,000

¹ Aritmetinis vidurkis;

² Standartinis nuokrypis;

Apibendrinus pateiktas eksperimentų rezultatų aritmetinių vidurkių bei standartinių nuokrypių reikšmes matyti, kad vykdytų eksperimentu priklausomi kintamieji kito netolygiai atsižvelgiant į keičiamą nepriklausomą kintamąjį – imituojamų vartotojų skaičių.

Pagal pateiktus apibendrintus eksperimentų rezultatus matyti, kad automatinių veiksmų scenarijaus vykdymo trukmė stipriai didėja didinant vartotojų skaičių nuo 500. Kadangi ilgiausiai vykdytos procedūros vykdymo trukmė kito neženkiai, galima teigti rekomendacijas, kad našumo tyrimo įrankis yra tinkamas naudoti imituojant vartotojų skaičių iki 500. Pagal atrinktą ilgiausiai vykdytą procedūrą matyti, kad visais atvejais kaip probleminė vieta yra nurodoma ta pati procedūra, dėl šios priežasties galima teigti, kad sėkmingai yra nustatoma tiriamos sistemos duomenų bazės probleminė vieta.

Pagal apskaičiuotus priklausomų kintamųjų standartinius nuokrypius matyti, kad vykdant atskirus kiekvieno eksperimento bandymus buvo gauti skirtingi rezultatai. Dėl šios priežasties turi būti rekomenduojama automatinių veiksmų scenarijus vykdyti nurodžius ne mažesnę kaip 30 iteracijų skaičių ir kiekvieno tyrimo rezultatuose pateikti apskaičiuotus aritmetinius vidurkius. Taip pat rekomenduojama apskaičiuoti ir ataskaitose pateikti kiekvieno tyrimo bei kiekvienos reikšmės standartinio nuokrypio reikšmes, kurios padėtų rezultatus peržiūrinčiam asmeniui geriau įvertinti galimas paklaidas bei sisteminius nukrypimus atliekant tyrimą.

8. Išvados

1. Literatūros šaltinių analizė parodė, kad viena labiausiai nagrinėjamų programinės įrangos testavimo sričių yra duomenų bazės.
2. Pastaruoju metu sustiprėjo tendencijos ir poreikiai įvertinti kuriamos programinės įrangos duomenų bazių našumą, bet atlikti duomenų bazių našumo tyrimą paprastomis priemonėmis yra labai sudėtinga, o specializuotų priemonių laisvai prieinamų nėra.
3. Pagal pateiktus sistemos reikalavimus, duomenų bazių našumo įrankiui realizuoti buvo pasirinkta ASP.NET technologija, kadangi ja realizuota sistema yra nereikli resursams, prieinama per internetą, nereikalauja papildomos programinės įrangos.
4. Atliktas sukurto duomenų bazių našumo tyrimo įrankio testavimas parodė, kad įrankis yra sukurtas pagal pateiktus reikalavimus ir visi testai patvirtino, kad programinė ranga veikia gerai.
5. Atliktas eksperimentinis tyrimas atskleidė šias duomenų bazių našumo tyrimo įrankio savybes:
 - Našumo tyrimas gali būti atliekamas imituojant iki tam tikro skaičiaus vartotojų (vykdytame eksperimente toks skaičius buvo 500). Jei našumo rodikliai ženkliai krenta, problema gali būti ne tik dėl tiriamos duomenų bazės problemų, bet ir dėl tyrimo įrankio bei aplinkos ribojimų;
 - Patartina vykdyti automatinių veiksmų scenarijus ne per vieną iteraciją, nes skirtingų iteracijų rezultatai skiriasi (pavyzdžiui, atlikti 30 iteracijų ar daugiau);
 - Įrankis patogus vartotojui, nes formuoja ataskaitas pagal vartotojo nurodytus pjūvius (išskiriant ilgiausiai vykdytą procedūrą ar procedūrų grupę arba pateikiant visus rezultatus pagal pasirinktą priklausomą kriterijų).
6. Tyrimo įrankį išplėtus *ODP.NET* biblioteka, duomenų bazių našumo tyrimo įrankis būtų suderinamas su *Oracle* duomenų bazėmis.
7. Siekiant pateikti detalesnius analizės rezultatus yra svarbu pateikti įvykdytų fizinių nuskaitymų parametras bei užklausų vykdymo planą.

9. Literatūros sąrašas

1. Lu Luo, „Software testing techniques“, Carnegie Mellon University
2. R.Venkat Rajendran, „White paper on Unit Testing“, Deccanet Designs Ltd.
3. Integration testing. [Interaktyvus] [žiūrėta 2008-11-22] Prieiga per Internetą: <http://msdn.microsoft.com/en-us/library/aa292128%28VS.71%29.aspx>
4. Lewis, W. E. (2009). Software testing and continuous quality improvement. Taylor & Francis Group.
5. W. T. Tsai, A. Saimi, L. Yu. (2003). Scenario-based Object-Oriented Testing Framework. Tempe: Department of Computer Science and Engineering, Arizona State University.
6. AMBLER, W. S.; SADALAGE, J. P. Refactoring Databases: Evolutionary Database Design Addison Wesley Professional [žiūrėta 2009-10-11] Prieiga per Internetą: <http://www.agiledata.org/essays/databaseTesting.html>
7. TIAN, J. Software Quality Engineering Testing, Quality Assurance, and Quantifiable Improvement Dallas, TX, 2005-28p.
8. Microsoft, visual studio produkto svetainė. [Interaktyvus] [žiūrėta 2008-11-23] Prieiga per Internetą: [http://msdn.microsoft.com/en-us/library/ms364079\(VS.80\).aspx](http://msdn.microsoft.com/en-us/library/ms364079(VS.80).aspx)
9. Qatraq produkto svetainė. [Interaktyvus] [žiūrėta 2008-11-23] Prieiga per internetą: <http://www.testmanagement.com/about.html>
10. BERTOLINO, A. Software Testing Research: Achievements, Challenges, Dreams // International Conference on Software Engineering. ISBN:0-7695-2829-5. 2007-18p.
11. BARISAS, D.; BAREIŠA, E. A Software Testing Approach Based on Behavioral UML Models. Information Technology And Control, Kaunas, Technologija, 2009, Vol. 38, No. 2, 119 – 124p.
12. BERTOLINO, A.; POLINI A. Towards Anti-Model-based Testing // International Conference on Software Engineering. 2007-124-125p.
13. BAREIŠA, E.; JUSAS. V.; MOTIEJŪNAS, K.; ŠEINAUSKAS, R. Realization-Independent Testing of Sequential Circuits. Experimental Results. Information technology and control, Kaunas, Technologija, 2004, No. 3(32), 7 – 13p.
14. SHELAR, S.; SAWAKAR, S.D. Database instances Generation Tool for White-box Testing // International conference of Signal Acquisition and Processing. ISBN: 978-0-7695-3594-4. 2009 112-116p.

15. DAVI, C.; YUETANG, D.; PHULLIS, G. F.; SAIKAT, D.; FILIPPOS, I. V.; ELAINE, J. W. An AGENDA for testing relational database applications. Department of Mathematics and Computer Science, Adelphi University. Gaden City, New York, 2004-30p.
16. Diagnose Server Performance Problems with SQL Server's SQL Profiler. [žiūrėta 2010-03-27]. Prieiga per Internetą: <http://www.devx.com/getHelpOn/Article/8202>
17. Reading a Trace File using C# in SQL Server 2005. [žiūrėta 2009-10-21]. Prieiga per Internetą: <http://www.sqlteam.com/article/reading-a-trace-file-using-charp-in-sql-server-2005>
18. Using the SQL Server Profiler. [žiūrėta 2010-02-10]. Prieiga per Internetą: http://www.sql-server-performance.com/tips/sql_server_profiler_tips_p1.aspx
19. Examining SQL Server Trace Files. [žiūrėta 2009-10-11]. Prieiga per Internetą: <http://www.sqlteam.com/article/examining-sql-server-trace-files>
20. Kompanijos IT technologinė strategija. Arba kodėl nėra alternatyvos Microsoft .NET?. [žiūrėta 2009-04-26] Prieiga per Internetą: <http://www.netvision.lt/lt/Kompanijos-IT-technologin%C4%97-strategija-Arba-kod%C4%97-n%C4%97ra-alternatyvos-Microsoft-NET-.aspx>
21. JAITEG, S.; KAWALJEET, S. Designing a Customized Test Data Generator for effective Testing of a Large Database // International Conference on Advanced Computer Theory and Engineering. India, ISBN: 978-0-7695-3489-3, 2008, p. 84-88
22. BUTKIENĖ, R.; ČEPONIENĖ, L.; NEMURAITĖ, L. Informacinių sistemų inžinerijos magistrų darbų metodiniai nurodymai: mokomoji knyga - Kauno Technologijos Universitetas, KAUNAS, 2008-311p.

Priedai

Priedas A Pavyzdinio duomenų bazių našumo tyrimo rezultatai

Kiekvieno bandymo pradžioje vykdomas operatyviosios atminties išvalymas. Kiekvieno vykdyto eksperimento bei jo bandymų rezultatai yra pateikiami (0.1- 0.5) lentelėse.

0.1 lentelė. E₁ rezultatai

Bandymo nr.	E ₁				
	Laikas,min	SP laikas, s	CPU, %	Skaitymas	Rašymas
1	1.04	0.09	11	4	1
2	1.03	0.07	13	4	1
3	1.03	0.09	17	6	1
4	1.04	0.01	11	4	1
5	1.04	0.09	12	6	1
6	1.03	0.09	15	4	1
7	1.04	0.09	17	4	1
8	1.06	0.01	13	4	1
9	1.03	0.07	11	4	1
10	1.03	0.01	18	4	1
11	1.03	0.09	12	4	1
12	1.02	0.07	11	6	1
13	1.04	0.09	12	4	1
14	1.03	0.07	11	4	1
15	1.03	0.01	15	6	1
16	1.03	0.09	12	4	1
17	1.03	0.01	17	4	1
18	1.03	0.09	13	4	1
19	1.03	0.07	17	6	1
20	1.03	0.09	15	4	1
21	1.04	0.07	13	4	1
22	1.04	0.09	12	4	1
23	1.03	0.01	13	4	1
24	1.03	0.01	11	4	1
25	1.04	0.09	17	6	1
26	1.03	0.07	15	4	1
27	1.02	0.09	12	4	1
28	1.03	0.01	11	6	1
29	1.03	0.01	17	4	1
30	1.04	0.07	12	4	1

0.2 lentelė. E₂ rezultatai

E ₂					
Bandymo nr.	Laikas, min	SP laikas, s	CPU, %	Skaitymas	Įrašymas
1	1.45	0.07	11	4	1
2	1.44	0.01	15	6	1
3	1.46	0.09	14	4	1
4	1.46	0.08	17	4	1
5	1.45	0.03	12	4	1
6	1.45	0.06	14	6	1
7	1.46	0.04	13	4	1
8	1.46	0.07	11	6	1
9	1.46	0.03	18	6	1
10	1.45	0.07	11	4	1
11	1.45	0.08	12	4	1
12	1.45	0.09	9	4	1
13	1.44	0.03	12	4	1
14	1.46	0.08	14	6	1
15	1.45	0.09	16	4	1
16	1.45	0.07	14	4	1
17	1.45	0.09	12	6	1
18	1.46	0.03	13	4	1
19	1.45	0.08	14	8	1
20	1.45	0.09	15	4	1
21	1.44	0.01	16	4	1
22	1.45	0.03	12	6	1
23	1.45	0.07	11	4	1
24	1.45	0.03	15	8	1
25	1.44	0.08	11	4	1
26	1.45	0.01	16	4	1
27	1.44	0.03	13	6	1
28	1.45	0.01	14	4	1
29	1.45	0.03	15	4	1
30	1.45	0.07	12	6	1

0.3 lentelė. E₃ rezultatai

E ₃					
Bandymo nr.	Laikas, min	SP laikas, s	CPU, %	Skaitymas	Įrašymas
1	2.57	0.03	23	4	1
2	3.18	0.04	24	6	1
3	3.18	0.01	26	4	1
4	3.12	0.01	27	8	1
5	3.18	0.05	23	4	1
6	3.18	0.02	24	6	1
7	3.17	0.06	26	4	1

8	3.18	0.07	21	8	1
9	3.15	0.03	17	6	1
10	3.18	0.01	23	4	1
11	3.18	0.06	21	4	1
12	3.17	0.01	20	8	1
13	3.18	0.05	28	4	1
14	3.18	0.05	23	6	1
15	3.15	0.03	24	4	1
16	3.16	0.09	26	4	1
17	3.19	0.06	21	8	1
18	3.19	0.09	23	4	1
19	3.16	0.05	25	8	1
20	3.18	1.05	21	4	1
21	3.17	0.03	20	4	1
22	3.15	0.09	23	8	1
23	3.18	0.06	24	4	1
24	3.19	0.03	21	8	1
25	3.18	0.05	23	4	1
26	3.18	0.01	20	4	1
27	3.18	0.06	26	6	1
28	3.13	0.01	21	4	1
29	3.12	0.03	23	8	1
30	3.18	0.06	24	6	1

0.4 lentelė. E₄ rezultatai

Bandymo nr.	E ₄				
	Laikas, min	SP laikas, s	CPU, %	Skaitymas	Įrašymas
1	9.24	0.04	28	4	1
2	9.22	0.05	27	6	1
3	9.25	0.07	29	4	1
4	9.28	0.06	25	8	1
5	9.22	0.05	26	4	1
6	9.21	0.07	27	6	1
7	9.25	0.04	25	8	1
8	9.23	0.06	26	8	1
9	9.24	0.07	25	6	1
10	9.23	0.06	27	4	1
11	9.26	0.05	27	4	1
12	9.26	0.04	26	8	1
13	9.24	0.06	29	8	1
14	9.23	0.07	31	6	1
15	9.22	0.05	28	4	1
16	9.25	0.06	26	4	1

17	9.24	0.07	24	8	1
18	9.23	0.04	28	4	1
19	9.23	0.05	29	8	1
20	9.24	0.06	34	8	1
21	9.24	0.05	26	6	1
22	9.24	0.06	27	8	1
23	9.22	0.04	28	4	1
24	9.25	0.06	26	8	1
25	9.23	0.05	24	4	1
26	9.24	0.05	26	6	1
27	9.24	0.04	28	8	1
28	9.22	0.06	28	4	1
29	9.23	0.06	29	8	1
30	9.26	0.04	24	6	1

0.5 lentelė. E₅ rezultatai

E ₅					
Bandymo nr.	Laikas, min	SP laikas, s	CPU, %	Skaitymas	Įrašymas
1	15.03	0.05	34	4	1
2	15.03	0.04	35	6	1
3	15.04	0.06	32	4	1
4	15.01	0.07	36	8	1
5	15.01	0.08	34	8	1
6	15.03	0.07	35	6	1
7	15.06	0.06	32	8	1
8	15.01	0.05	36	8	1
9	14.59	0.07	37	6	1
10	15.01	0.08	39	8	1
11	15.02	0.06	34	4	1
12	15.03	0.08	36	8	1
13	15.02	0.07	35	8	1
14	15.01	0.05	37	6	1
15	15.02	0.08	36	6	1
16	15.01	0.07	38	4	1
17	15.02	0.05	34	8	1
18	14.59	0.06	36	6	1
19	15.03	0.07	34	8	1
20	15.01	0.08	39	8	1
21	15.01	0.06	37	6	1
22	15.01	0.08	38	8	1
23	15.03	0.07	36	6	1
24	15.01	0.08	34	8	1
25	15.03	0.07	36	4	1

26	15.01	0.06	38	6	1
27	15.04	0.03	39	8	1
28	15.01	0.04	41	6	1
29	14.59	0.05	37	8	1
30	15.03	0.07	38	6	1

Priedas B Įrankio diegimo aktas



Kauno Technologijos Universitetui

DĖL JUSTINO GREIBAUS PRAKTIKOS METU SUKURTOS PROGRAMINĖS ĮRANGOS

Studentas Justinas Greibus praktikos metu (nuo 2009-09-07 iki 2009-12-18) įmonėje „iTreeFinance“ atliko šias užduotis:

- Sukūrė ir įdiegė tyrimo įrankį „Duomenų bazių našumo tyrimo įrankis“;
- Atliko tyrimo įrankio „Duomenų bazių našumo tyrimo įrankis“ integraciją su vidine įmonės projektų valdymo sistema

Įdiegtas tyrimo įrankis šiuo metu yra naudojamas įmonės vidiniams tikslams.

Projektų vadovas



Evaldas Bružė

UAB iTree Finance
Konstitucijos pr. 7, LT-09308 Vilnius, Lietuva
Tel. +370 5 248 75 15
Faks. +370 5 248 75 18

Įmonės kodas 302044589
PVM mok. kodas LT100004351917

IBAN: LT20 7044 0600 0655 8490
Bankas: AB SEB bankas
Banko kodas: 70440
SWIFT: CBVI LT 2X