

KAUNO TECHNOLOGIJOS UNIVERSITETAS  
INFORMATIKOS FAKULTETAS  
INFORMACIJOS SISTEMŲ KATEDRA

Egidijus Zinkevičius

**Reikalavimų specifikavimo šablonų pagrindu analizė  
ir programinis prototipas**

Magistro darbas

Darbo vadovas  
prof. dr. R. Butleris

Kaunas, 2009

KAUNO TECHNOLOGIJOS UNIVERSITETAS  
INFORMATIKOS FAKULTETAS  
INFORMACIJOS SISTEMŲ KATEDRA

Egidijus Zinkevičius

**Reikalavimų specifikavimo šablonų pagrindu analizė  
ir programinis prototipas**

Magistro darbas

Recenzentas

doc. dr. V. Pilkauskas

2009-01-12

Vadovas

prof. dr. R. Butleris

2009-01-12

Atliko

IFM-3/4 gr. stud.  
Egidijus Zinkevičius

2009-01-12

Kaunas, 2009

# Turinys

SUMMARY .....	4
1. Įvadas.....	5
2. Reikalavimų specifیکavimo šablonų pagrindu analizė .....	5
2.1. Reikalavimų specifیکavimo analizės tikslas .....	5
2.2. Reikalavimų specifیکavimo proceso tyrimo aspektai .....	6
2.3. Vartotojų, kurie naudosis programiniu prototipu, analizė .....	7
2.3.1. Vartotojų grupės ir savybės .....	7
2.3.2. Vartotojų tikslai ir problemos.....	8
2.4. Reikalavimų specifیکavimo proceso analizė .....	9
2.4.1. Reikalavimų analizė .....	9
2.4.2. Reikalavimų specifیکavimo šablonų analizė .....	13
2.4.2.1. Volere šablonas .....	13
2.4.2.2. SRS šablonas .....	15
2.4.2.3. URD šablonas .....	17
2.4.2.4. CPE4004 šablonas .....	18
2.4.2.5. Reikalavimų specifیکavimo šablonų palyginimas .....	20
2.4.3. Reikalavimų valdymo įrankių analizė .....	21
2.4.3.1. Reikalavimų valdymo įrankio vaidmuo .....	21
2.4.3.2. RequisitePro įrankis.....	22
2.4.3.3. Optimal Trace įrankis .....	23
2.4.3.4. Doors įrankis .....	24
2.4.3.5. Reikalavimų valdymo įrankių palyginimas.....	25
2.5. Atliktos analizės išvados .....	28
3. Reikalavimų specifیکavimo šablonų pagrindu koncepcinis modelis.....	29
3.1. Reikalavimų specifیکavimo modelio pagrindimas ir esmės išdėstymas .....	29
3.2. Projektuojamo prototipo architektūros pasirinkimas.....	29
3.3. Reikalavimų specifیکavimo prototipo apibrėžimas .....	30
3.4. Programinio prototipo pritaikymo privalumai.....	31
3.5. Reikalavimų specifیکavimo sudėties koncepcinis modelis .....	31
3.6. Specifیکavimo procesų modelis.....	33
3.7. Reikalavimų specifیکavimo koncepcinės dalies apibendrinimas .....	45
4. Reikalavimų specifیکavimo programinio prototipo realizacija .....	46
4.1. Reikalavimų specifیکacijos statinės struktūros modelis .....	46
4.1.1. Loginė prototipo architektūra .....	46
4.1.2. Reikalavimų objektų saugyklos schema.....	47
4.2. Reikalavimų specifیکavimo prototipo realizacija .....	49
4.3. Reikalavimų specifیکavimo prototipo realizavimo apibendrinimas.....	50
5. Reikalavimų specifیکavimo programinio prototipo tyrimas.....	51
5.1. Pasirinktos dalykinės srities aprašas.....	51
5.2. Programinio prototipo veikimo tyrimo eiga .....	51
5.3. Programinio prototipo langai.....	52
5.4. Reikalavimų specifیکavimo programinio prototipo įvertinimas.....	66
6. Išvados .....	69
7. Literatūros sąrašas .....	71
8. Priedai .....	73

# **Analysis of Requirements Specification Using Templates and Programme Prototype**

## **SUMMARY**

The usage of computers takes more important place in our modern life. It was noticed that more and more enterprises use requirement management tools and specification templates to increase quality of requirement specification and projects.

Very often ineffective requirements-gathering processes are the source of project failures and software defects. In addition, the cost of fixing defects rises exponentially as their discovery occurs later in the development process. However, a solid requirements process is fundamental to an effective overall process, and companies seeking increased quality and productivity should start with the requirements process.

The intention of this work is to make the experimental research of requirements specification using templates and evaluate it.

# 1. Įvadas

Šiuolaikiniame sparčiai besivystančiame pasaulyje vis rečiau sutinkama kompiuterizacijos proceso nepaliesta sritis. Kompiuterizacijos procesui didelę įtaką turi ne tik pati kompiuterinė technika, bet ir tinkamos programinės įrangos kūrimas bei naudojimas.

Informacinės sistemos (IS) sąvoka taip pat slepiasi po programinės įrangos apibrėžimu. Taigi reikalavimų inžinerija yra bene svarbiausias etapas informacinės sistemos kūrime. Tik tikslus reikalavimų specifikavimo proceso valdymas leis sėkmingai projektuoti, realizuoti ir įdiegti informacines sistemas kompiuterizuojamose srityse.

Dažnai neefektyvus reikalavimų rinkimo procesas tampa projekto nesėkmės ar programinės įrangos defekto pagrindu. Be to, aptiktų netikslumų pašalinimo kaina kyla eksponentiškai, kai jie atrandami vėlyvosiose projekto stadijose. Todėl įmonės ar asmenys, siekiantys pagerinti projekto kokybę ir produktyvumą, turėtų ypatingai atkreipti dėmesį į reikalavimų valdymo procesą [1].

Šis magistrinis darbas yra skirtas reikalavimų specifikavimo šablonų pagrindu eksperimentiniam tyrimui ir jo vertinimui.

## 2. Reikalavimų specifikavimo šablonų pagrindu analizė

### 2.1. Reikalavimų specifikavimo analizės tikslas

Visų pirma atliekama reikalavimų specifikavimo šablonų pagrindu analizė. Analizės tikslas yra išsiaiškinti, ko tikisi vartotojai, kokie vyksta procesai ir kokie modeliai yra jau sukurti ar netgi realizuoti.

Tuomet reikia išsiaiškinti vartotojus, t.y. kokie vartotojų tipai, kokia vartotojų atsakomybė bei kvalifikacija, kiek yra vartotojų ir kokia rinka egzistuoja. Taip pat būtina išanalizuoti vartotojų tikslus bei atkreipti dėmesį į problemas, su kuriomis susiduriama.

Toliau pereinama prie egzistuojančių sprendimų peržiūros ir analizavimo. Analizuojama reikalavimų specifikavimo metodika, šablonai bei reikalavimų valdymo įrankiai. Taip siekiama išsiaiškinti, ar yra sukurtas mus tenkinantis sprendimas (modelis ir/arba programinis jo realizavimas), bei programinio prototipo sudarymas atsižvelgiant į egzistuojančių analogų sprendimus.

## 2.2. Reikalavimų specifikuavimo proceso tyrimo aspektai

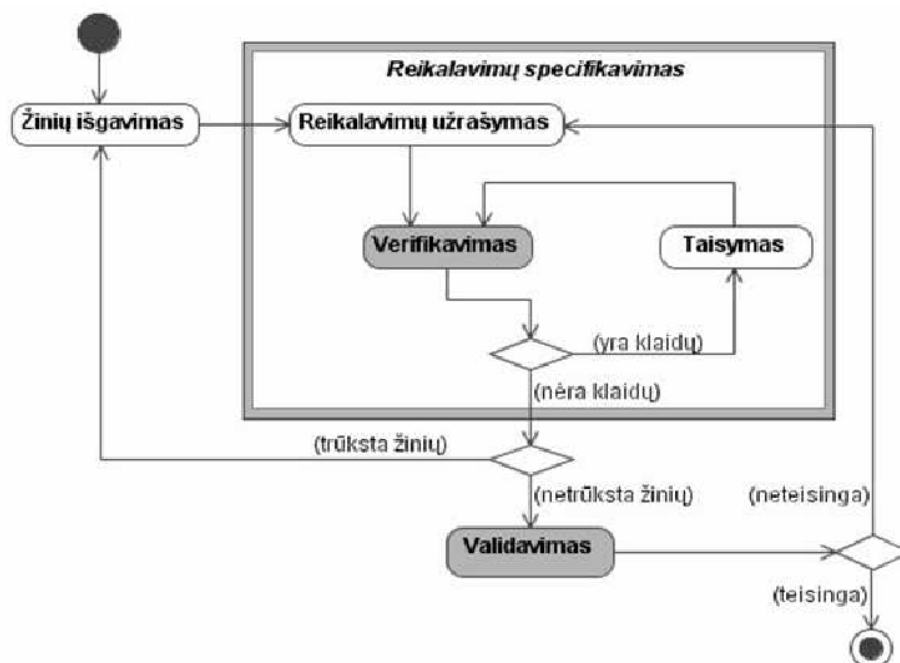
Tyrimo sritis apima reikalavimų valdymo įrankius bei reikalavimų specifikuojamą šablonų pagrindu. Tiriama panašūs modeliai, jų pritaikymas praktikoje. Informacija apžvelgiama tarptautiniu mastu.

Tyrimo objektas - reikalavimų specifikuavimo remiantis šablonais procesas bei specifikuavimo sudėtis.

Tyrimo problemą sudaro reikalavimų valdymas, nes tai vienas svarbiausių informacinės sistemos kūrimo etapų, nuo kurio priklauso sėkminga projekto evoliucija. Todėl bus kuriamas programinis modelis paremtas šablonais, gerinantis reikalavimų valdymą, ir tokiu būdu mažinantis projekto riziką bei gerinantis jo kokybę.

Taigi tyrimo objektą sudaro reikalavimų specifikuavimo remiantis šablonais procesas bei specifikuavimo sudėtis. Visų pirma apžvelgiami universalūs šablonai (pavyzdžiui VOLERE, SRS), kurie gali būti panaudojami patogesniai reikalavimų specifikuavimo proceso įgyvendinimui.

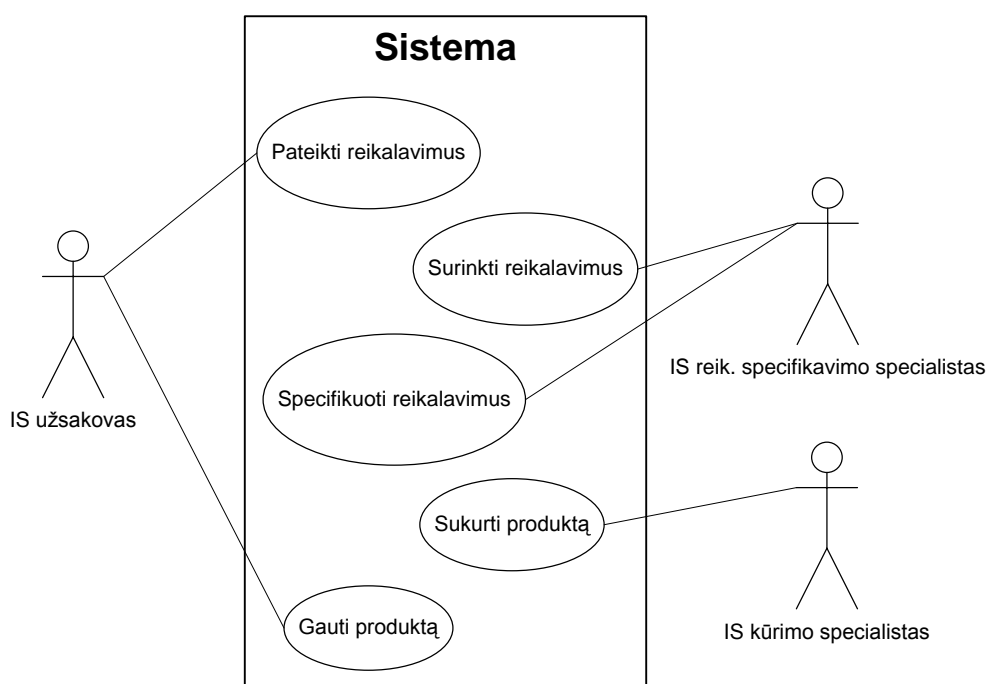
Toliau pereinama prie pačios specifikuavimo sudėties analizės. 2.1 paveiksle pateikta standartinė reikalavimų specifikuavimo proceso diagrama [2].



2.1 pav. Reikalavimų specifikuavimo proceso diagrama[2]

## 2.3. Vartotojų, kurie naudosis programiniu prototipu, analizė

Projektuojamu programiniu prototipu galės naudotis keletas vartotojų tipų. Kiekvieno tipo asmenys gali atlikti tik tam tikras funkcijas. Dėl vartotojų kompetencijos skirtumų nerekomenduojama bandyti atlikti kitų grupių funkcijas. Vartotojų atliekamos funkcijos pavaizduotos panaudojimo atvejų modelyje 2.2 paveiksle.



2.2 pav. Vartotojų panaudojimo atvejų modelis

### 2.3.1. Vartotojų grupės ir savybės

Egzistuoja trys vartotojų grupės, turinčios tik joms būdingas savybes ir specifines funkcijas:

- ✓ Informacinės sistemos užsakovai – tai vartotojų grupė, vienu ar kitu būdu pateikianti ir išdėstanti savo reikalavimus kuriamai informacinei sistemai. Ši vartotojų grupė bendrauja su IS reikalavimų specifikuojimo ir kūrimo specialistais.
- ✓ Informacinės sistemos reikalavimų specifikuojimo specialistai – tai pagrindinė vartotojų grupė, kuriai priklausantys asmenys tiesiogiai susiję su kuriamos informacinės sistemos reikalavimų specifikuojimu šablonų pagrindu.

- ✓ Informacinės sistemos kūrimo specialistai – tai vartotojų grupė, kuriai priklausantys asmenys naudoja pagrindinės vartotojų grupės specialistų sudarytas reikalavimų specifikacijas tolesniam informacinės sistemos kūrimui ar tobulinimui.

### 2.3.2. Vartotojų tikslai ir problemos

Visų trijų vartotojų grupių atstovai suinteresuoti operatyviai ir tinkamai įgyvendinti savo tikslus bei siekius. IS užsakovas siekia gauti produktą, atitinkantį jo keliamus reikalavimus. IS reikalavimų specifikavimo specialistai yra atsakingi už profesionalų reikalavimų surinkimą, jų specifikavimą bei pateikimą IS kūrimo specialistams, kurie, savo ruožtu, siekia sukurti užsakovą tenkinantį produktą. Todėl reikalingas visų vartotojų grupių aktyvumas ir bendravimas tarpusavyje atliekant iškeltas užduotis. Vartotojų siekiami tikslai ir problemos, su kuriomis susiduriama, pateikiami 2.1 lentelėje.

2.1 lentelė. Vartotojų tikslai ir problemos

Vartotojas	Tikslai	Problemos
IS užsakovas	Kuo tiksliau perteikti reikalavimus IS reikalavimų specifikavimo specialistams. IS kūrimo specialistų dėka gauti kuo labiau jo reikalavimus atitinkančią sistemą.	Reikalavimų surinkimo procesas vyksta lėtai, užsakovo reikalavimai nuolat keičiasi. Netiksliai sudarytas darbo planas.
IS reikalavimų specifikavimo specialistas	Greitai, efektyviai ir tiksliai surinkti reikalavimus ir juos specifikuoti.	Dviprasmiški ir nuolat besikeičiantys užsakovų reikalavimai.
IS kūrimo specialistas	Gauti aiškiai specifikuotus ir nusakytus reikalavimus. Sukurti užsakovą pilnai tenkinantį produktą.	Neaiški ir dviprasmiška reikalavimų specifikacija.

Projektuojant sistemos modelį atsižvelgiama į vartotojų problemų sprendimus bei tikslų įgyvendinimą.



## **2.4. Reikalavimų specifikavimo proceso analizė**

### **2.4.1. Reikalavimų analizė**

Reikalavimą galima apibrėžti kaip objekto ar proceso savybę, numatytą specifikacijoje, sutartyje, standarte ar kitokio pobūdžio dokumente.

Gera reikalavimų aibė yra tiesiog būtina bet kuriam projektui, tačiau ypatingai kompiuterinių sistemų projektams, norint, jog jie būtų sėkmingai įgyvendinti. Čia daugelis projektų žlunga, nes nėra teisingai specifikuojama, ką sistema turėtų daryti. Tiesą sakant didžiajai daliai kuriamų sistemų yra nustatomi galutiniai terminai, biudžetas ir neaiškios užgaidos, ką sistema turi daryti. Šios problemos priežastys labai paprastos:

- kompiuterinių sistemų kūrėjai dažnai nenutuokia, kaip tam tikra veikla vyksta ar turėtų vykti, lyginant su tos veiklos vartotojais;
- veiklos vartotojai sunkiai suvokia, ką kompiuterinė sistema jiems gali suteikti;

Todėl visas veiklos valdymo laikas orientuojamas į terminus bei biudžetą, o ne tam, kas turi būti atlikta ir pristatyta [3].

Tik pradėjus reikalavimų, keliamų kuriamai sistemai, specifikavimą, susiduriama su trimis pagrindiniais reikalavimų tipais:

- funkciniai reikalavimai (Functional requirements). Šio tipo reikalavimai nusako vidinį sistemos ar komponento darbą (funkciją) t.y. skaičiavimus, techninius elementus, duomenų manipuliaciją bei apdorojimą ir kitą specifinį funkcionalumą, kuris parodo panaudos atvejo išpildymą. Be to, funkcinis reikalavimas gali būti susietas su vienu ar keliais nefunkciniais reikalavimais.
- nefunkciniai reikalavimai (Nonfunctional requirements). Tai reikalavimai, nusakantys sistemos veikimą, bet ne specifinį elgesį. Kitaip tariant čia apibrėžiama ne ką sistema darys, bet kaip darys. Nefunkciniai reikalavimai gali būti išskirti į dvi pagrindines kategorijas:
  - 1) Vykdyto savybės (saugumas, tinkamumas naudoti, matomumas vykdymo metu);
  - 2) Plėtojimo savybės (tikrinamumas, palaikomumas, praplečiamumas, programinės įrangos sistemos statinės struktūros apėmimas);
- projektavimo tikslai (Design objectives). Šio tipo reikalavimai atlieka pagrindinį vaidmenį pasirenkant sprendimą iš siūlomų. Čia atsižvelgiama į svarbiausias naujos sistemos savybes: ar ji turi būti greita, ar turi pasižymėti didelėmis kaupimo

galimybėmis, ar lengvai naudojama ir pan. Tačiau dažniausiai ne viską galima įgyvendinti taip, kaip pageidaujama.

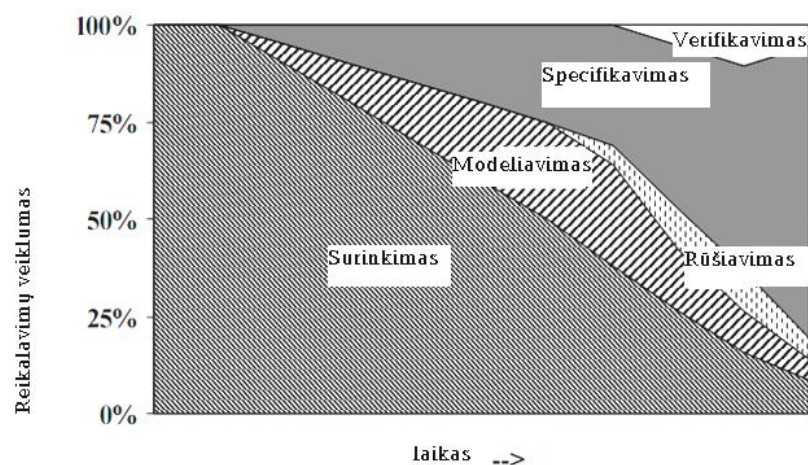
Šiuolaikiniame pasaulyje reikalavimų specifikavimo procesui skiriama vis daugiau dėmesio ir išlaidų. Reikalavimų specifikavimas yra reikalavimų inžinerijos uždavinys, kurio metu išanalizuoti reikalavimai yra tinkamai dokumentuojami tolesniam naudojimui [4]. Reikalavimų inžineriją sudaro procesai, naudojami sistemos reikalavimų išgavimui, analizei ir atestavimui. Čia susiduriama su svarbiomis problemomis [5]:

- Klientai nežino to, kas įmanoma techniškai;
- Klientas dažnai pakeičia reikalavimus, pamatęs sistemos galimybes;
- Klientas ne visada ir ne pilnai sugeba išdėstyti, ko jis norėtų;
- Vėliau iškilę reikalavimai gali konfliktuoti su jau įvykdytais reikalavimais;
- Prieš projektuojant sistemą labai sunku apibrėžti visus reikalavimus ir jų korektiškumą;

Modernizuojant šią sritį vis dažniau pereinama nuo tradicinio krioklio metodo prie modernių ir efektyvių vystymosi ciklų:

- iteratyvinė reikalavimų inžinerija (Iterative Requirements Engineering);
- inkrementinė reikalavimų inžinerija (Incremental Requirements Engineering);
- lygiagreti reikalavimų inžinerija (Parallel Requirements Engineering);
- laiko rėmų planavimo reikalavimų inžinerija (Timeboxed Requirements Engineering);

Reikalavimų inžinerija su kiekviena diena vis labiau tobulėja. Pripažįstama daug naudingų reikalavimų uždavinių tokių, kaip veiklos analizė, reikalavimų rinkimas, reikalavimų analizė, reikalavimų specifikavimas ir reikalavimų valdymas. 2.3 paveiksle pavaizduotas reikalavimų inžinerijos etapų pasiskirstymas laike [5].



2.3 pav. Reikalavimų inžinerijos etapų pasiskirstymas laike [5]

Vartotojo įtraukimas į reikalavimų inžinerijos procesą yra labai svarbus. Tokiu atveju vartotojas lengviau priims kuriamą sistemą, ja lengviau naudosis, žinodamas, kad dalyvavo kūrimo procese, prisidėjo prie jos reikalavimų formavimo, ir kad buvo atsižvelgta į jo, kaip sistemos naudotojo, keliamus reikalavimus [5].

Egzistuoja daugybė reikalavimų analizavimo metodų, kurių dalis (pvz. panaudojimo atvejų analizė) yra plačiai pripažįstama kaip svarbus senesnių metodų patobulinimas. Tokie reikalavimų analizės metodai naudoja sudėtinius abstrakcijos tipus ir lygius, kurie pateikia skirtingas reikalavimų peržiūras skirtingoms auditorijoms [4].

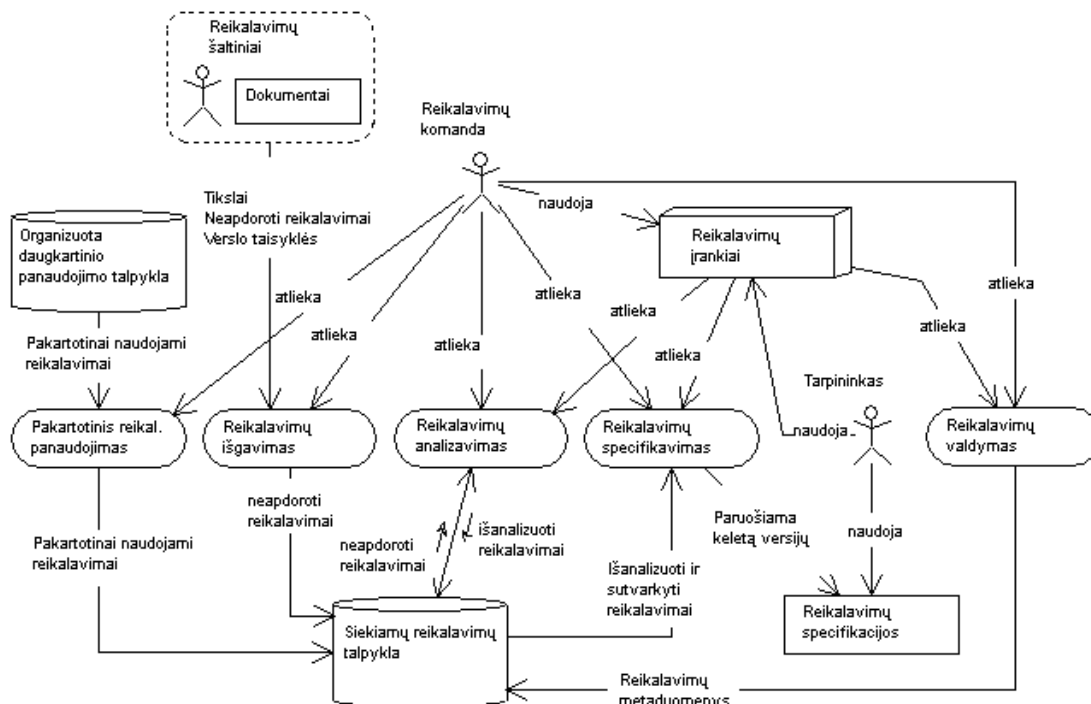
Visi šie reikalavimų analizės metodai daro įtaką reikalavimų specifikavimui paveikdami specifikuojamos informacijos tipus, turinį, formatą bei patį reikalavimų specifikavimo organizavimą. Galimas reikalavimų nusakymas:

- natūralios kalbos tekstiniai reikalavimai;
- reikalavimų modeliai grafinio modeliavimo kalba;
- sprendimų lentelės;
- formaliai specifikuoti reikalavimai specifikacijos kalba;
- kiti;

Tobulinant reikalavimų specifikavimo procesą didėja ir reikalavimų valdymo įrankių paklausa. Atsiradus reikalavimų talpykloms (Requirements Repository), prireikia įrankių, kurių dėka būtų galima automatiškai sugeneruoti reikalavimų specifikacijas bei ataskaitas. Tokie įrankiai turi žemiau išvardintas savybes:

- vartotojui priimtina grafinė sąsaja;
- reikalavimų inžinerijos palaikymas;
- reikalavimų inžinerijai giminingų veiklų palaikymas;
- kolektyvinio darbo galimybė;
- saugumas, išbaigtumas, išplečiamumas;
- paskirstyta sistema;
- pakartotinis reikalavimų panaudojimas;
- ne tik CASE įrankis;

2.4 paveiksle pateikiamas reikalavimų specifikacijos būdas, kuris padeda išspręsti problemas, atsirandančias naudojant modernius reikalavimų valdymo įrankius paremtus reikalavimų talpyklomis [4]. Naudojant tam tikrą įrankį, paremtą šiuo reikalavimų specifikacijos būdu, galima automatiškai, lengvai ir nebrangiai generuoti skirtingų tipų aukštos kokybės reikalavimų specifikacijas, kurios atitinka individualius skirtingų auditorijų reikalavimus.



2.4 pav. Talpykla paremta reikalavimų specifikacija[4]

Atliekant reikalavimų specifikavimą būtina užtikrinti, kad specifikacija būtų neprieštaringa, o tuo pačiu būtų pasiekta maksimali jos kokybė. Norint sukurti kokybišką specifikaciją, reikalingas metodas, atitinkantis šiuos reikalavimus [6]:

- Reikalavimų informacijos sistemai sudarymo eiga turi būti natūrali, t.y. turi atitikti tradicinį šio proceso eiliškumą;
- Metodas turi sumažinti atotrūkį tarp vartotojo ir analitiko;
- Modeliavimo procesas turi užtikrinti organizacijos veiklos ar jos išskirto fragmento modelio pilnumą;
- Turi būti galimybė atlikti specifikacijos kokybės kontrolę;

Reikalavimų specifikavimo procese vis didesnis vaidmuo tenka šablonams. Šablonas nusakomas kaip iš anksto parengtas kokio nors rašto projektas, naudojamas konkreitiems struktūrizuotiems dokumentams kurti. Šablonų pritaikymas reikalavimų specifikavime leidžia greičiau ir tiksliau išsiaiškinti užsakovo pateiktus reikalavimus ir juos struktūrizuoti bei grupuoti tolesniam apdorojimui.

Apibendrinant galima drąsiai teigti, kad reikalavimų valdymas yra kiekvieno sėkmingo projekto esmė. Juk reikalavimai atkeliauja iš žmonių, kurie nėra tiksliai apsisprendę, ko

nori iš produkto, nuolat keičia reikalavimus ir kitais būdais apsunkina tikslių reikalavimų išpildymą. Tačiau inžinieriai, dirbantys su projektu, turi tiksliai žinoti, ką turi padaryti. Todėl reikalavimai privalo būti tikslūs ir pateikiami tokia forma, kad būtų įmanomas jų sekimas bei susietumas su kitais reikalingais dokumentais. Taigi reikalavimų valdyme reikalingas ne tik tekstas, bet ir struktūra, turinti hierarchiją, informaciją lentelių ir diagramų pavidale bei ryšius tarp elementų.

Reikalavimų valdymo įrankis privalo išpildyti vartotojų grupių, tokių kaip projekto vadybininkai, kūrimo inžinieriai, keitimų vadybininkai, testavimo inžinieriai ir kitų, poreikius. Jis turi pateikti informaciją aiškiai, laikyti ją saugiai, išlaikyti versijas, fiksuoti kelius tarp įrašų, suteikti patogią importavimo ir eksportavimo galimybę, skaičiuoti metriką ir būti nesudėtingu vartotojo atžvilgiu. Todėl egzistuoja begalė reikalavimų, keliamų reikalavimų valdymo įrankiui [7].

## **2.4.2. Reikalavimų specifikavimo šablonų analizė**

Apžvelgtoje literatūroje rasta keletas šablonų, naudojamų reikalavimų rinkimo ir specifikavimo procese. Panagrinėsime labiausiai paplitusius reikalavimų specifikavimo šablonus, sutinkamus reikalavimų inžinerijoje.

### **2.4.2.1. Volere šablonas**

Volere reikalavimų specifikavimo šablonas. Pirmoji Volere šablono versija buvo išleista 1995 metais. Volere šablonas naudojamas pradiniame sistemos kūrimo etape ir yra kaip pagrindas užregistruoti vartotojų reikalavimus. Šablonas suskirstytas į skyrius pagal reikalavimų tipą. Šablonas padeda sukaupti reikalavimus, kuriuos pateikia vartotojai per interviu arba kurie užregistruoti analizuojamo objekto veiklą reglamentuojančioje dokumentacijoje. Tai atviras šablonas, kurį galima pritaikyti konkrečiam atvejui. Šablono skyrių, kuris netinka nagrinėjamam objektui, galima išmesti arba sukurti naują skyrių, kuris leidžia specifiuoti specifines dalykinės srities charakteristikas [8]. Volere šablonas apima šiuos reikalavimų tipus [9]:

- funkciniai reikalavimai (Functional requirements);
- nefunkciniai reikalavimai (Nonfunctional requirements);
- projekto apribojimai (Project constraints);
- projekto vykdytojai (Project drivers);
- projekto išeiga (Project issues);
- testavimo reikalavimai (Testing requirements);

Visi reikalavimų tipai yra suskaidyti į smulkesnius ir nepersidengiančius potipius. Tokiu būdu užtikrinamas reikalavimų konkretumas, pilnumas, nepersidengiamumas ir vienareikšmiškumas.

Volere reikalavimų specifikavimo šablone reikalavimai yra susieti su panaudojimo atvejais. Čia kiekvienas panaudojimo atvejis parodo tai, ką produktas atliks, todėl jis turi daug susijusių funkcinį reikalavimų. Be to, panaudojimo atvejis taip pat gali turėti ir nefunkcinių reikalavimų bei apribojimų.

Kiekvienas surinktas reikalavimas, nepriklausomai nuo jo tipo (funkcinis, nefunkcinis, projekto apribojimas), turi daug atributų. Vienas atributas yra unikalus reikalavimo identifikatorius, o kitas – ryšys su visais panaudojimo atvejais, kuriuose dalyvauja šis reikalavimas. Aprašymas, pagrindimas bei tikimo kriterijus kartu nusako reikalavimo reikšmę ir tuo pačiu padaro jį matuojamu bei testuojamu. 2.5 paveiksle pavaizduota Volere kortelė, skirta atominių reikalavimų atributų rinkimui [7].

<u>Reikalavimas #:</u>	<b>45</b>	<u>Reikalavimo tipas:</u>	<b>10</b>	<u>Ivykis/panaudojimo atvejis #:</u>	<b>6</b>
<u>Aprašymas:</u>	<i>Sistema turi signalizuoti, jei oro stotis nepertnuoda duomenų</i>				
<u>Pagrindimas:</u>	<i>Trikdžiai pertnuodant duomenis gali nurodyti, kad oro duomenų registravimo stotis sugedusi ir plikšalai prognozuoti naudojami duomenys yra nepilni</i>				
<u>Šaltinis:</u>	<i>Keitų priežiūros inžinierius</i>				
<u>Tikimo kriterijus:</u>	<i>Sistema informuoja vartotoją, jei kurios nors oro stoties pertnuotų kiekvieno tipo parametrų kiekis per valandą neatitinka numatyto parametrų registravimo dažnumo</i>				
<u>Užsakovo tenkinimas:</u>	<b>3</b>	<u>Užsakovo netenkinimas:</u>	<b>5</b>		
<u>Priklausomybės:</u>	<i>Nėra</i>	<u>Konfliktai:</u>	<i>Nėra</i>		
<u>Papildoma medžiaga:</u>	<i>Philips oro parametrų matavimo stoties specifikacija</i>				
<u>Istoriija:</u>	<i>Užregistruotas 2001 spalio 15 d.</i>				

**2.5 pav.** Volere kortelė (ang. Volere Snow Card)[7]

Naudojant Volere šabloną reikalavimams aprašyti, vadovaujamasi susisteminto proceso apribojimais, nusakančiais, kokio etapo metu ir kokie reikalavimai turi būti aprašyti. Kai kuriuose proceso etapuose naudojamas specializuotas apklausos lapas, padedantis surinkti reikalavimus, juos analizuoti ir apibendrinti [8].

Volere šablono autoriai pateikia reikalavimų specifikacijos užpildymo proceso eiga, kuri naudinga ne tik pradedantiems, bet ir pažengusiems [7]. Ji pateikiama „Mastering the Requirements Process by James and Suzanne Robertson. Addison-Wesley, 1999” knygoje. Volere šablono naudojimas mokymo tikslams yra nemokamas. Kitu atveju reikalaujama vienkartinės paramos už kiekvieną projektą. Caliber, DOORS ir Requisite yra populiariausi automatizuoti reikalavimų specifikavimo įrankiai, kurie gali būti naudojami su Volere reikalavimų specifikavimo šablonu.

#### 2.4.2.2. SRS šablonas

Antrasis reikalavimų specifikavimo šablonas yra SRS (Software Requirements Specification). Jį naudoja nemažai programinės įrangos kūrėjų visame pasaulyje. Programinės įrangos reikalavimų specifikacija yra dokumentas, naudojamas bendravimui tarp kliento ir kūrėjo. Šis šablonas gali būti panaudotas įvairiems tikslams bei daugybėje kontekstų, pradedant kompanijomis, viešai publikuojančiomis reikalavimų specifikaciją ir baigiant konkurencingomis paraiškomis [10].

Pagal SRS šabloną paruošta specifikacija nustato susitarimą tarp kliento ir teikėjo, ką programinės įrangos produktas darys, sumažina kūrimo išlaidas, teikia apytikrių išlaidų bei tvarkaraščių bazę, nustato validavimo bei verifikavimo pradinį tašką ir pateikia galimų patobulinimų bazę.

Dažniausiai SRS šablono pagalba reikalavimai specifikuojami kaip paprasti deklaratyvūs teiginiai, užrašyti tekstu paremta natūralia kalba. Surinkti reikalavimai turi būti sugretinami, analizuojami ir išskirstomi į atskiras kategorijas. Šablonas apima sekančias reikalavimų kategorijas [10]:

- funkciniai reikalavimai (Functional Requirements);
- našumo reikalavimai (Performance Requirements);
- sąsajos reikalavimai (Interface Requirements);
- veikimo reikalavimai (Operational Requirements);
- išteklių reikalavimai (Resource Requirements);
- tikrinimo reikalavimai (Verification Requirements);
- dokumentacijos reikalavimai (Documentation Requirements);
- kokybės reikalavimai (Quality Requirements);
- saugumo reikalavimai (Safety Requirements);
- patikimumo ir palaikomumo reikal. (Reliability & Maintainability Requirements);

Reikalavimų specifikuojant, naudojant SRS šabloną, reikia atkreipti dėmesį į du faktorius [10]. Pirmasis – notacija, naudojama reikalavimams apibrėžti. Naudojamos modeliavimo technikos, priešingos natūraliai kalbai, apima aukštesnį specifikuojimo tikrinimo lygį, kuris nėra pasiekiamas naudojant natūralią kalbą, stipriai sumažina dviprasmybių galimybę, kas suteikia specifikuojimui aukštesnio lygio integralumą, nuoseklumo tikrinimas yra supaprastintas ir netgi automatizuotas, kuomet modeliavimo technika yra palaikoma programinės įrangos įrankio.

Antrasis – būdas, kuriuo parašyta notacija pateikiama reikalavimų specifikuojimo dokumento skaitytojams. Metodai, paremti grafiniais elementais, tokiais kaip duomenų srautų diagrama, natūraliai talpina iš anksto apibrėžtą pateikimo metodą.

SRS šablono pagrindu sudaryta reikalavimų specifikuojimo specifikacija pasižymi šiomis charakteristikomis:

- pilnumu (pagrindinių reikalavimų apibrėžimai siejami su funkcionalumu, našumu, projektavimo ribomis, išorinėmis sąsajomis; visų programinės įrangos standartų atitikimas; visų lentelių pavadinimų bei nuorodų susietumo pilnumas);
- nuoseklumu (nei vienas reikalavimas neturi konfliktų su kitais reikalavimais; sudėtinųjų deskriptorių naudojimas);
- sekamumu (reikalavimų ištakos bei nuorodos yra pasiekiamos; atliktų modifikacijų sekimas iki dabartinės būsenos);
- nedviprasmiškumu (kiekvienas reikalavimas turi vieną ir tik vieną interpretaciją; formalios reikalavimų specifikuojimo kalbos vartojimas; dviprasmiška sąvoka galima prie jos pateikus skaidrų tekstą, paaiškinantį sąvokos kontekstą);
- patikrinamumu (egzistuoja išbaigtas rentabilus metodas, kurio pagalba asmuo ar mašina tikrina produkto atitikimą reikalavimams);

SRS paketą sudaro du atskiri šablonai [11]: 1) SRS šablonas naudojamas projektuose, pagrįstuose panaudojimo atvejų modeliavimu 2) SRS šablonas, naudojamas projektams tik su tradicinėmis reikalavimų specifikuojimo metodikomis. Tarpusavyje derinant SRS šabloną ir panaudojimo atvejų metodikas, galima veiksmingiau specifikuoti reikalavimų aibę, keliamų sistemai ar taikomajai programai. SRS šablono naudojimas yra nemokamas.

Elektros ir elektronikos inžinierių instituto (ang. IEEE) standartas suteikė detalizuotą turinio struktūrą, kuri atitinka gerą SRS reikalavimų specifikuojimo šabloną. Tai suteikia šablonui lankstumo bei leidžia vartotojui pasirinkti labiausiai tinkamą šablono tipą, orientuotą į specifikuojamą sistemą.



### 2.4.2.3. URD šablonas

Trečiasis reikalavimų specifikuojimo šablonas yra URD (User Requirement Document) [12]. Šis dokumentas buvo sukurtas remiantis SRS reikalavimų specifikuojimo šablonu, taigi iš dalies paveldėjo pastarojo privalumus ir trūkumus.

URD šablone kiekvienam reikalavimų skyriui pateikiamos gairės. Be to, kiekvienas skyrius gali būti pertvarkytas į tokius poskyrius, kurie aktualūs specifikuojimo sričiai. Įžanginėje dokumento srityje randame apibrėžimus, akronimus, santrumpas bei nuorodas į išorinius dokumentus.

Antroje šablono dalyje išvardinami visi specifiniai reikalavimai kartu su atributais. Vartotojo reikalavimai patenka į dvi kategorijas: funkcinis ir nefunkcinis reikalavimas. Kiekvienas vartotojo reikalavimas turi turėti šiuos atributus:

- identifikatorių (vėlesnių etapų sekimo palengvinimui);
- prioritetą (jo dėka kūrėjas gali sudaryti produkto gamybos tvarkaraštį);

Funkcinio reikalavimo apibrėžimas niekuo nesiskiria nuo naudojamo SRS reikalavimų specifikuojimo šablone. Tačiau URD šablone pastarųjų reikalavimų žymėjimui reikalinga speciali forma [12], pateikta 2.2 lentelėje.

**2.2 lentelė.** *Reikalavimų specifikuojimo forma*

<b>[Vartotojo reikalavimo identifikatorius]</b>	[ kiekvienam reikalavimui suteikiama ID reikšmė norint palengvinti susietumą su nuorodomis ]
<b>Reikalavimo apibrėžimas</b>	[ trumpas reikalavimo apibrėžimas ]
<b>Prioritetas</b>	Suteikia prioriteto lygį (pvz. aukštas, vidutinis, žemas) – kai kurie reikalavimai gali turėti vienodus prioritetus ]
<b>Smulkesnės detalės</b>	[ išsamus reikalavimo apibrėžimas ]

Nefunkciniai reikalavimai URD reikalavimų specifikuojimo šablone yra išskirstyti į keturis stambius poskyrius:

- aparatūrinės įrangos reikalavimai (tam tikra aparatūrinė įranga, kuri reikalinga produkto veikimui);
- programinės įrangos reikalavimai (tam tikra įranga, kurią naudos produktas);
- testavimo reikalavimai;
- kiti nefunkciniai reikalavimai (diegimas ir palaikymas);

Kaip ir funkcinių, taip ir nefunkcinių reikalavimų žymėjimui URD šablone reikalinga ta pati forma, pateikta 2.2 lentelėje. Kaip matome šio šablono detalumo lygis nėra didelis. Taigi specifikuojamų reikalavimų detalizavimą ir išskyrimą į smulkesnius poskyrius sąlygoja reikalavimų specifikuojančio specialisto kompetencija. URD reikalavimų specifikuojančio šablono naudojimas nėra autorių apmokestintas.

#### 2.4.2.4. CPE4004 šablonas

Ketvirtasis dokumentas, naudojamas reikalavimų specifikuojimui – CPE4004 šablonas [13]. Čia reikalavimai traktuojami kaip duomenų, įvykių seku, veiksmų, laiko planavimo, praktiškumo, ryšių tarp objektų ar žmonių rinkinys. CPE4004 reikalavimų specifikuojančio šablono turinį sudaro net penkiolika reikalavimų tipų [14]:

- projekto aprašymas ir tikslas (The project description and Purpose);
- klientas, pirkėjas ir tarpininkas (The Client, the Customer, and Other Stakeholders);
- produkto vartotojai (Users of the Product);
- apribojimai (Constraints);
- sistemos diegimo aplinka (Implementation Environment of the Current System);
- faktai ir veiklos taisyklės (Facts and business rules);
- prielaidos (Assumptions);
- funkcionalumo ir duomenų reikalavimai (Functional and Data Requirements);
- veiklos panaudojimo atvejai (Business Use cases);
- produkto sritis (The Scope of the Product);
- našumo reikalavimai (Performance Requirements);
- išvaizdos bei naudojimo reikalavimai (Look and Feel Requirements);
- vykdymo bei aplinkos reikalavimai (Operational and Environmental Requirements);
- saugumo reikalavimai (Security Requirements);
- reikalavimų specifikuojančio formato (Requirement specification format)

Toks detalus reikalavimų išskyrimas užtikrina jų teisingumą, tikslumą, išbaigtumą ir svarbiausia atskleidimą. Nežiūrint į tai prie kiekvieno reikalavimo tipo pateikiama motyvacija (kodėl tai yra svarbu) ir pavyzdys kaip atrodys specifikuotas reikalavimas.

Funkcinių reikalavimų specifikavimui naudojama speciali forma, pateikta 2.3 lentelėje [13]. Kiekvienam reikalavimui reikia nurodyti numerį, tipą, prioritetą, trumpą antraštę ir apibrėžimą, rodyklę į panaudojimo atvejį, racionalumą (reikalavimo pateisinimą), atitikimo kriterijų (nustatymas, ar priimtas sprendimas atitinka originalų reikalavimą) bei pilną ir išsamų reikalavimo aprašymą.

**2.3 lentelė. Reikalavimų specifikavimo forma[13]**

<i>Reikalavimas #</i>	<i>Reikalavimo tipas = F</i>	<i>Panaudojimo atvejai</i>	<i>Užrašai</i>	<i>Vartotojo sąsaja</i>
Aprašymas Racionalumas Atitikimo kriterijus Prioritetas			Reikalavimo apibrėžimas Kodėl tai yra reikalavimas Testavimo atvejis	
<b>Pagalbinė medžiaga</b>				

CPE4004 reikalavimų specifikavimo šablonas pagrįstas panaudojimo atvejų diagramomis. Kiekvienas panaudojimo atvejis specifikuojamas specialios formos pagalba, kuri pateikta 2.4 lentelėje [13]. Tokiu būdu užtikrinamas panaudojimo atvejo ir reikalavimų surišimas.

**2.4 lentelė. Panaudojimo atvejų specifikavimo forma[13]**

<i>Panaudojimo atvejis #</i>	<i>Reikalavimas #</i>	<i>Panaudojimo atvejo vardas</i>	<i>Papildomų srautų sąrašas</i>
Aprašymas Pagrindinis srautas Atitikimo kriterijus Prioritetas		Panaudojimo atvejo apibrėžimas Trumpas scenarijaus aprašymas  Testavimo kriterijus	
<b>Pagalbinė medžiaga</b>			

Naudojant CPE4004 šabloną reikalavimams specifikuoti, vadovaujamosi susisteminto proceso apibrėžimais, priklausomai nuo reikalavimų, kurie turi būti aprašyti, bei esamo etapo. Prie kiekvieno reikalavimo tipo pateikiamos gairės, užtikrinančios specifikuojamo reikalavimo išpildymą. Gaila, bet CPE4004 šablono autoriai nepateikia reikalavimų specifikacijos užpildymo proceso eigos, kuri atskleistų pilną specifikuojamų reikalavimų susietumą.

## 2.4.2.5. Reikalavimų specifikavimo šablonų palyginimas

Toliau pateikiama apžvelgtų reikalavimų specifikavimo šablonų palyginimo lentelė. Šablonai tarpusavyje lyginami atsižvelgiant į svarbius reikalavimų specifikavimo kriterijus: specifikavimo procesas, reikalavimų tipai bei detalumas, susietumas, panaudojimo atvejai, suderinamumas su įrankiu, modifikavimas ir išlaidos. Kriterijų paaiškinimai:

- specifikavimo procesas – reikalavimų surinkimo bei specifikavimo proceso pateikimas;
- reikalavimų tipai - šablono išskirstymas į skyrius pagal reikalavimų tipus;
- reikalavimų detalumas – reikalavimų detalizavimo ir tikslumo lygis;
- reikalavimų susietumas – specifikuojamų reikalavimų susietumas tarpusavyje;
- panaudojimo atvejai – panaudojimo atvejų naudojimas bei reikalavimų surišimas su pastaraisiais;
- suderinamumas su įrankiu – šablono panaudojimas reikalavimų valdymo įrankiuose;
- modifikavimas – šablono struktūros, turinio keitimas bei pritaikymas asmeniniams poreikiams;
- kaštai – šablono apmokestinimo nebuvimas;

Išanalizuotų reikalavimų specifikavimo šablonų privalumų bei trūkumų palyginimas pagal išvardintus kriterijus pateiktas 2.5 lentelėje.

**2.5 lentelė. Šablonų palyginimas**

Savybė	Šablonas			
	Volere	SRS	URD	CPE4004
Specifikavimo procesas	+/-	-	-	-
Reikalavimų tipai	+	+	-	+
Reikalavimų detalumas	+	+/-	-	+/-
Reikalavimų susietumas	+	+/-	+/-	+/-
Panaudojimo atvejai	+	+	-	+
Suderinamumas su įrankiais	+/-	-	-	-
Modifikavimas	+	+	+	+
Kaštai (nemokamas)	+/-	+	+	+

Iš 2.5 lentelės matome, jog tinkamiausias sprendimas reikalavimų specifikavimui yra Volere šablonas. Reikalavimų tipų pasirinkimo galimybė, detalumas bei specifikavimo procesas yra tikrai svarbūs specifikuojant reikalavimus. Be to, reikia atkreipti dėmesį ir į vartotojus, dirbsiančius su šablonais. Volere šablono dalinis suderinamumas su reikalavimų valdymo įrankiais yra itin svarbus. Pagrindinis jo minusas yra tame, jog šablonas, naudojamas nemokymo tikslams, yra mokamas.

### **2.4.3. Reikalavimų valdymo įrankių analizė**

#### **2.4.3.1. Reikalavimų valdymo įrankio vaidmuo**

Reikalavimų valdymo įrankiai atlieka svarbų vaidmenį reikalavimų specifikavimo procese. Naudojant tik tekstinius dokumentus ir skaičiuokles, galima sukurti daugkartinio panaudojimo standartizuotus dokumentus ir formas. Tačiau šie dokumentai suteikia mažai naudos projekto plėtojimo ir valdymo metu ir yra neefektyvūs bendradarbiaujant su projekto varovais. Tuo tarpu daug efektyvesnis būdas yra naudoti reikalavimų valdymo įrankį, kurio pagalba būtų kaupiami teisingi duomenys ir valdomas sekamumas. Fizinis projekto narių atskyrimas sąlygoja neefektyvų komunikavimą tarpusavyje, ko pasekoje didėja iteracijų skaičius. Tai stabdo projekto plėtojimo procesą ir tuo pačiu mažina užsakovo pasitenkinimą kuriamu produktu. Reikalavimai pateikimas gali turėti keletą formų: tekstiniai aprašai, diagramos ir prototipai, nustatytos taisyklės.

Minimaliai reikalavimų valdymo įrankis turi apimti:

- individualių reikalavimų identifikaciją;
- reikalavimų rūšiavimą ir perskirstymą;
- reikalavimų grupės pakartotinį identifikavimą/išeities taško keitimą;
- elementarią duomenų sąsają;

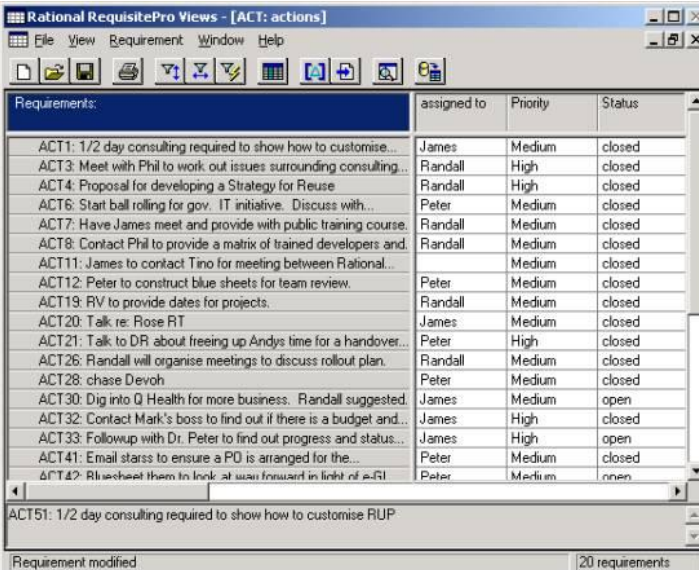
Rimtesni įrankiai teikia sekamumą, bendradarbiavimo palaikymą, modeliavimo bei testavimo integraciją. Viena iš svarbiausių savybių yra sekamumas. Ypač naudingi tie įrankiai, kurie įgalina sekamumą tarp artefaktų, t.y. ne tik tarp reikalavimų, bet taip pat ir modelių, programinio kodo ir testų [1].

Reikalavimų specifikavimo proceso organizavimui rasti trys įrankiai. Šios sistemos realizuotos skirtingai ir kiekviena iš jų atlieka tam tikras funkcijas, orientuotas į reikalavimų valdymo gerinimą.

### 2.4.3.2. RequisitePro įrankis

Pirmasis įrankis Rational RequisitePro [15], skirtas projektavimo grupėms ar atskiriems vartotojams, norintiems sėkmingai valdyti pastovius ar nuolat besikeičiančius reikalavimus, kurti panaudojimo atvejų diagramas, pagerinti sekamumą, sumažinti projekto riziką ir pagerinti jo kokybę. Įrankis palengvina netekstinių reikalavimų elementų (pvz. modeliai, diagramos, grafikai) įtraukimą į reikalavimų dokumentą, kuomet kiti įrankiai pateikia tik nuorodas tarp reikalavimų ir išorinių elementų. Reikalavimus turi sudaryti tik svarbi informacija.

Organizacijos, siekiančios pagerinti reikalavimų proceso valdymą, pirma turėtų pagerinti procesą bei apmokymus, ir tik tada naudoti įrankį reikalavimų valdymo automatizavimui ir sekamumui. Šis įrankis nepalaiko projekto valdymo integracijos [1]. Įrankis talpina galingą reikalavimų variklį, kurio pagalba nesudėtinga paruošti reikalavimus, atributus ir dokumentų tipus. Užklausų ir filtrų pagalba nesunku rasti reikalingą informaciją. Grafinis abejotinių reikalavimų, kaip reikalavimų modifikavimo rezultato, pateikimas bet kuriuo projekto metu yra įmanomas. Ypatingai patogiu yra tai, jog įrankis turi priėjimą internetu. Vartotojas, turintis interneto naršyklę bei priėjimą prie interneto, nepriklausomai nuo naudojamos platformos, gali greitai ir efektyviai peržiūrėti, keisti ir valdyti reikalavimus netgi neturint įdiegto įrankio [15]. Tai ypač aktualu tarp paskirstytų darbo komandų. 2.6 paveiksle pavaizduota Requisite Pro įrankio forma, talpinanti apibrėžtus reikalavimus, priskyrimus asmenims, prioritetus bei būsenas.



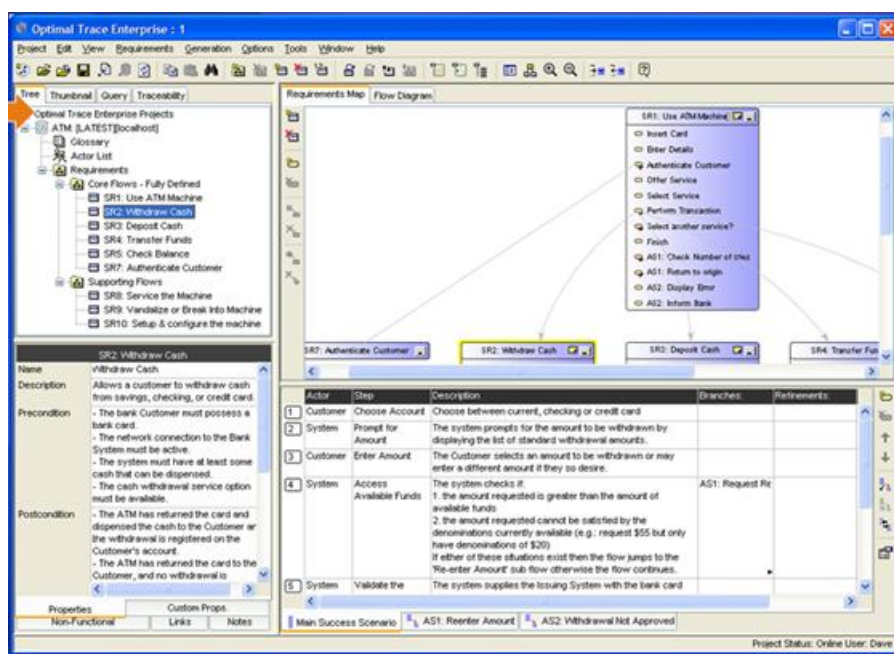
Requirements:	assigned to	Priority	Status
ACT1: 1/2 day consulting required to show how to customise...	James	Medium	closed
ACT3: Meet with Phil to work out issues surrounding consulting...	Randall	High	closed
ACT4: Proposal for developing a Strategy for Reuse	Randall	High	closed
ACT6: Start ball rolling for gov. IT initiative. Discuss with...	Peter	Medium	closed
ACT7: Have James meet and provide with public training course.	Randall	Medium	closed
ACT8: Contact Phil to provide a matrix of trained developers and...	Randall	Medium	closed
ACT11: James to contact Tino for meeting between Rational...		Medium	closed
ACT12: Peter to construct blue sheets for team review.	Peter	Medium	closed
ACT19: RV to provide dates for projects.	Randall	Medium	closed
ACT20: Talk re: Rose RT	James	Medium	closed
ACT21: Talk to DR about freeing up Andys time for a handover...	Peter	High	closed
ACT26: Randall will organise meetings to discuss rollout plan.	Randall	Medium	closed
ACT28: chase Devoh	Peter	Medium	closed
ACT30: Dig into Q Health for more business. Randall suggested.	James	Medium	open
ACT32: Contact Mark's boss to find out if there is a budget and...	James	High	closed
ACT33: Followup with Dr. Peter to find out progress and status...	James	High	open
ACT41: Email stars to ensure a PO is arranged for the...	Peter	Medium	closed
ACT42: Rubsheet them to link at way forward in light of a RI	Peter	Medium	open
ACT51: 1/2 day consulting required to show how to customise RUP			

2.6 pav. Requisite Pro įrankio pagrindinė reikalavimų forma

Įrankis puikiai suderinamas su IBM UDB DB2, Microsoft SQL Server, Oracle, ir Microsoft Access duomenų bazėmis. Be to reikalavimų informacija pateikiama tokia forma, kuri prieinama ir modifikuojama kitų IBM produktų pagalba (pvz. Rational Software Architect, Rational Software Modeler, Rational Systems Developer, WebSphere Business Modeler, WebSphere Integration Developer, Rational SoDA®, Rational Unified Process and Rational TestManager®, ir daug kitų).

### 2.4.3.3. Optimal Trace įrankis

Antrasis įrankis yra Optimal Trace [16], skirtas reikalavimų surinkimui, dokumentavimui ir modeliavimui. Jis talpina intuityvią vartotojo sąsają reikalavimų kaupimui, peržiūrai ir valdymui bei bendradarbiavimo mechanizmą, kuris įgalina keleto analitikų darbą su vienu ar keliais projektais tuo pačiu metu. Jo sąsaja suprojektuota greitam ir aiškiam reikalavimų fiksavimui. Automatinės srautų diagramos įgalina lengvą tarpusavio priklausomybių ir aplinkos poveikio vizualizaciją reikalavimų specifikavimo procese. Reikalavimų specifikavimas įvairių formų pagalba suteikia galimybę tarpininkams bendrauti taip, kaip jiems yra patogiau (pvz. Word, Excel, html). Užbaigti projektai gali būti eksportuojami į MSProject, CSV ar kitus formatus. 2.7 paveiksle pavaizduota Optimal Trace įrankio pagrindinė forma, apimanti reikalavimų žemėlapi, srautų diagramas ir pačius reikalavimus.



2.7 pav. OptimalTrace įrankio pagrindinė forma

Tai yra labiau bendradarbiavimui skirtas įrankis, nes naudoja centrinę talpyklą visiems projekto reikalavimams ir artefaktams. Tokiu būdu projekte naudojama informacija yra nuolat atnaujinama ir prieinama nuotoliniam vartojimui. Automatizuotų testavimo atvejų generavimas tikrina originalią reikalavimų specifikaciją ir užtikrina, kad netikslumai ar defektai būtų rasti ankstyvojoje projekto vykdymo stadijoje taip sumažinant išlaidas ir keliant kokybę. Reikalavimų sekamumas pilnai apima funkcinių ir nefunkcinių reikalavimų testavimą. Visi tarpininkai gali valdyti ir sekti projekto pokyčius realiu laiku. Atliktų pakeitimų istorija nuolat saugoma. Kiekvienam vartotojui pritaikytas vaizdavimas, modernūs bei pilnai paruošti pranešimai ir sudėtinės užklausos leidžia pilnai panaudoti projekto duomenis ir sudaryti užsakovą tenkinančią reikalavimų specifikaciją [1].

#### 2.4.3.4. Doors įrankis

Trečiasis įrankis yra Doors [17], talpinantis sekamumo ataskaitas apie reikalavimus viso kuriamos programinės įrangos gyvavimo ciklo metu. Nauji šablonai buvo įtraukti į Doors šablonų biblioteką norint palengvinti reikalavimų dokumentų generavimą. Nauji šablonai apima ISO 12207, ISO 6592 ir IEEE programinės įrangos standartus. 2.8 paveiksle pavaizduota Doors įrankio pagrindinė forma, apimanti specifikuojamus reikalavimus.

The screenshot shows the Doors software interface with a table of requirements. The table has columns for ID, Requirements in Volere Template, Use Cases to Link by Name or ID, Conflicting Reqs, and Dependee Reqs. The requirements are organized into sections: 1 ATM Requirements, 1.1 Customer Services, 1.2 Utilities, and 1.3 Customer NFRs.

ID	Requirements in Volere Template	Use Cases to Link by Name or ID	Conflicting Reqs	Dependee Reqs
R-12	<b>1 ATM Requirements</b>			
R-11	<b>1.1 Customer Services</b>			
R-3	Customer can withdraw cash	Withdraw some Cash		R-39
R-2	Customer can obtain statements	Obtain a Statement		R-39
R-14	Customer can order a chequebook	Order a Chequebook		R-39
R-15	Customer can change password	Change Password		R-39 R-43
R-18	<b>1.2 Utilities</b>			
R-39	ATM can authenticate the customer	UC-19 UC-16 UC-32 UC-13		
R-41	ATM can obtain current balance	UC-36		
R-45	ATM can count out cash	UC-37		
R-47	ATM can ensure cash includes small notes	UC-37		
R-46	ATM can dispense cash	UC-38		
R-42	ATM can debit customer's account	UC-39		
R-43	ATM can update customer's password	UC-46		
R-40	<b>1.3 Customer NFRs</b>			
R-44	All transactions must be quick and simple			
R-16	Customer accounts must be extra secure	Change Password Withdraw some Cash Obtain a Statement	R-44	R-39
R-17	Cash withdrawal must be very reliable	Withdraw some Cash		R-3

2.8 pav. Doors įrankio pagrindinė forma



Telelogic Doors yra reikalavimų valdymo produktų rinkos lyderė ir turi labiausiai išbaigtą gyvavimo ciklo integraciją, įtraukiant ir integruojant su kitomis Telelogic produktų versijomis bei projektavimo įrankiais (pvz. Continuous, Cool) ir netgi Microsoft Project įrankiu. Ši integracija su MSProject įgalina galimybę matyti reikalavimų pakeitimų įtaką darbų planui ir apskritai užduočių valdymui. Doors įrankis naudoja į duomenis orientuotą metodą reikalavimų rinkimui ir valdymui [1]. Telelogic Doors įrankis pasižymi šiomis savybėmis:

- Intuityvi sąsaja, skatinanti reikalavimų valdymo pasirinkimą.
- Projekto išplečiamumas
- Lanksti, moderni, lengvai naudojama reikalavimų sekamumo matrica.
- Išsami pagalba rašant, struktūrizuojant, valdant ir analizuojant reikalavimus ir jų sekamumą.
- Neturinti sau lygių integracija su kitais Telelogic sprendimais ir trečiųjų šalių įrankiais, norint pagerinti reikalavimų pateikiamumą bei užtikrinti sekamumą viso kūrimo ciklo metu.

#### 2.4.3.5. Reikalavimų valdymo įrankių palyginimas

Toliau pateikiama apžvelgtų įrankių savybių palyginimo lentelė. Reikalavimų valdymo įrankiai lyginami tarpusavyje atsižvelgiant į kuriamam prototipui svarbius kriterijus: operatyvus informavimas, intuityvi sąsaja, saugumas sekamumas, šablonų panaudojimas, papildomo funkcionalumo suteikimas, integracija su kitais produktais, daugiavartotojiškumas ir kiti. Įrankių palyginimo rezultatai pateikti 2.6 lentelėje [18, 19].

**2.6 lentelė. Analogiškų reikalavimų specifikavimo įrankių palyginimas**

Savybė	RequisitePro	OptimalTrace	Doors
Ms Word dokumento ataskaita	•	•	
Kurti ir keisti struktūrizuotus reikalavimus įskaitant ir pagrindinį veiksmų planą	•	•	
Pilnas dvikryptis atnaujinimas į/iš MS Word. Leidžiami keitimai įtraukiant sugeneruotą dokumentą atgal į originalų projektą	•	•	
Keletas vartotojų gali dirbti prie to paties projekto smulkiame reikalavimų lygyje	•	•	
Integracija su IBM/Rational Rose 2002, automatiškai sugeneruojama panaudojimo atvejų diagrama	•	•	
Detalus lygis išlaikyti atliktų pakeitimų istoriją ir sugeneruotą ataskaitą		•	
Dalinis šablonų taikymas	•		•
Intuityvi vartotojo sąsaja	•	•	•
Papildomų funkcijų gausa	•	•	•
Laisvai platinama versija (nemokama)			
Sekamumas	•	•	•
Prieinamumas	•		
Integracija su MS Project paketu		•	•
Grafinių reikalavimų elementų įtraukimas	•		

Iš 2.6 lentelės matome, jog visi trys reikalavimų valdymo įrankiai daugiau ar mažiau tarpusavyje panašūs savo privalumais bei minusais. Iš apžvelgtų reikalavimų valdymo įrankių tinkamiausias reikalavimų specifikavime šablonų pagrindu būtų IBM firmos RequisitePro produktas dėl tokių savo teikiamų galimybių, kaip dalinis reikalavimų specifikavimo šablonų panaudojimas, grafinių reikalavimų elementų įtraukimas, sekamumas ir kiti. Deja, šio IBM firmos RequisitePro produkto esminis trūkumas tame, jog įrankis yra komercinis (mokamas), o ne laisvai platinamas. Todėl nenuostabu, kad papildomo funkcionalumo suteikimas įrankiui yra labai keblus ir

sudėtingas procesas. Svarbu yra tai, jog įrankis neturi galimybės pilnai palaikyti Volere šablona, o tai reiškia, kad nėra ir metodikos, kurios dėka būtų galima tiksliai panaudoti reikalavimų specifikavimo šablonus.

Taigi atidžiai įvertinus visų apžvelgtų reikalavimų valdymo įrankių gerąsias ir blogąsias savybes, nuspręsta nesirinkti nei vieno komercinio produkto, o sudaryti originalų programinį prototipą, pasižymintį specializuotu reikalavimų specifikavimo procesu. Visų pirma būtina teisingai suprojektuoti reikalavimų specifikavimo pasirinkto šablono pagrindu koncepcinį modelį, kurio pagrindu kursime programinį prototipą atsižvelgdami į analogiškų įrankių turimas teigiamas ir mus tenkinančias savybes. Taipogi naujasis prototipas pasižymės ir tokiu funkcionalumu, kurio daugiau ar mažiau pasigendama viename ar kitame analogiškame įrankyje (pvz. atitinkamų reikalavimų objektų susietumas, pilnas šablono pritaikymas). Nežiūrint į tai programinis prototipas bus projektuojamas taip, jog pasižymėtų išplečiamumu. Todėl papildomo funkcionalumo suteikimas, esant reikalui, bus nesudėtingai ir greitai įgyvendinamas iš esmės nekeičiant pačio programinio prototipo struktūros.

## **2.5. Atliktos analizės išvados**

1. Reikalavimų specifikavimo procesas vis labiau kompiuterizuojamas. Efektyvus reikalavimų surinkimas ir specifikavimas labai svarbus kiekvienos informacinės sistemos kūrimo. Klaidų ir netikslumų palikimas reikalavimų specifikavime sąlygoja itin didelius kaštus vėlyvosiose projekto stadijose. Pastebėta, jog naudojant reikalavimų valdymo įrankius gerinama ne tik reikalavimų specifikacijos, bet ir pačio galutinio produkto kokybė.

2. Atlikus išsamią panašaus konteksto reikalavimų specifikavimo šablonų analizę nuspręsta naudoti Volere šabloną reikalavimų specifikavimui. Tokį apsisprendimą lėmė gana ryški šablono teikiamų privalumų gausa lyginant su kitais šablonais.

3. Atlikus išsamią analogiškų reikalavimų valdymo įrankių analizę pastebėta, jog laisvai platinamo (nemokamo) įrankio, atitinkančio mūsų keliamus reikalavimus, nerasta. Be to papildomo funkcionalumo suteikimas komerciniam reikalavimų valdymo įrankiui yra sudėtingas ir labai problematiškas procesas. Todėl nuspręsta, jog reikalavimų specifikavimui Volere šablono pagrindu bus kuriamas naujas programinis prototipas.

4. Sudarytas programinis prototipas padidins išplečiamumą ir lankstumą reikalavimų specifikavime šablonų pagrindu. Dėka prototipo internetinės prieigos, vartotojai nebebus susaistyti erdvės atžvilgiu.

5. Norint išlaikyti projekto kokybę ir produktyvumą, reikia susidaryti efektyvius reikalavimų specifikavimo procesus ir tik tada naudoti reikalavimų valdymo įrankius sekamumo supaprastinimui ir palengvinimui.

### **3. Reikalavimų specifikavimo šablonų pagrindu koncepcinis modelis**

Baigus analizės dalį, pereiname prie reikalavimų specifikavimo Volere šablono pagrindu koncepcinio modelio projektavimo bei sudarymo remiantis medžiaga, gauta pagal analitinėje dalyje iškeltus reikalavimus. Šiame skyriuje pateiksime projektuojamo prototipo architektūros, išsamų klasių diagramos su naudojamomis operacijomis bei specifikavimo procesų modelius.

#### **3.1. Reikalavimų specifikavimo modelio pagrindimas ir esmės išdėstymas**

Labai svarbu tinkamai sudaryti reikalavimų specifikavimo šablonų pagrindu koncepcinį modelį, nes juo remiantis bus kuriamas programinis prototipas. Visi vartotojų iškelti reikalavimai turi atsispindėti modelyje, o vėliau perkelti į prototipą. Taigi nuo projekto modelio teisingumo ir išbaigtumo priklauso galutinio programinio prototipo funkcionalumas bei vartotojų poreikių išpildymas. Svarbu ir tai, jog sukurtas programinis prototipas būtų naudojamas praktikoje.

#### **3.2 Projektuojamo prototipo architektūros pasirinkimas**

Reikalavimų specifikavimo šablonų pagrindu prototipo realizavimui egzistuoja keletas variantų. Reikėtų nepamiršti, jog pirmenybė turėtų būti teikiama tam sprendimui, kuris užtikrintų lengvą programinio prototipo priėjimą iš įvairių vietų, o ne tik iš tam tikro kompiuterio. Tokiu atveju vartotojai nebus susaistyti erdvės atžvilgiu, o prototipas bus visad pasiekiamas iš bet kurio kompiuterio, turinčio internetinę prieigą.

Pirmasis realizacijos būdas yra labiau inžinerinės pakraipos. Pradžioje būtina saugykla, užtikrinanti korektišką duomenų patalpimą. Tam tikslui galima rinktis MY SQL (darbą atliekant su PHP programavimo kalba) duomenų bazę. Duomenų saugyklos užpildymui, reikalinga vartotojo sąsaja, talpinanti specialias formas reikalavimų specifikacijos užpildymo modelio panaudojimui bei atvaizdavimui.

Antrasis realizacijos būdas yra labiau adaptacinio pobūdžio. Šiuo atveju pasirenkamas jau egzistuojantis reikalavimų specifikavimo įrankis. Pasirinktas įrankis pritaikomas prie reikalavimų specifikavimo šablonų pagrindu modelio. Čia reikalinga speciali metodika, užtikrinanti suprojektuoto modelio įliejimą į jau egzistuojančių reikalavimų specifikavimo procesų aibę.

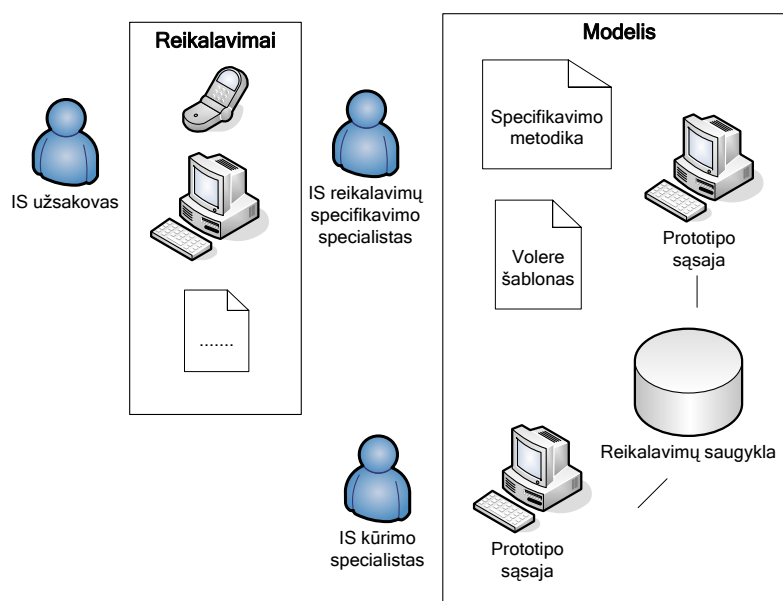
Įvertinus abiejų architektūrų teikiamus privalumus bei susidarytų reikalavimų išpildymą, buvo pasirinktas inžinerinės pakraipos sprendimas. Taigi reikalavimų specifikavimo šablonų pagrindu programinis prototipas bus realizuotas naudojant PHP programavimo kalbą bei MY SQL duomenų bazę reikalavimams kaupti.

Pasirinkus architektūrą reikalingas tikslus bei išbaigtas reikalavimų specifikavimo šablonų pagrindu proceso modelis. Tik teisingai suprojektavus pastarąjį modelį galima pereiti prie tolimesnio prototipo realizacijos etapo.

### 3.3 Reikalavimų specifikavimo prototipo apibrėžimas

Reikalavimų specifikavimo šablonų pagrindu prototipas palengvina reikalavimų inžinerijos proceso valdymą. Jis skirtas surinktų reikalavimų specifikavimui bei pateikimui pagal nustatytą metodiką. Šio prototipo pagalba gerinama reikalavimų specifikavimo kokybė, ko pasekoje mažinamos išlaidos vėlyvosiose projekto stadijose (mažiau besikeičiančių reikalavimų), o užsakovas gauna jį pilnai tenkinantį produktą.

Prototipą galima apibrėžti kaip tam tikrą suplanuotą veiksmų seką. Visų pirma IS užsakovas pateikia reikalavimus kuriamam produktui. Tuomet IS reikalavimų specifikavimo specialistas, naudodamas Volere reikalavimų specifikavimo šabloną, reikalavimų specifikacijos užpildymo modelį bei duomenų saugyklą, apdoroja, specifikuoja ir išsaugo gautus reikalavimus. Užbaigus šį etapą, IS kūrimo specialistas, naudodamasis sudaryta specifikacija, kuria užsakovą tenkinantį produktą. Prototipo struktūrinė diagrama pavaizduota 3.1 paveikslėlyje.



3.1 pav. Reikalavimų specifikavimo prototipo struktūrinė diagrama

### **3.4 Programinio prototipo pritaikymo privalumai**

Darbo tikslas - sukurti vartotojui priimtina reikalavimų specifikavimo šablonų pagrindu programinį prototipą, kuris gerintų reikalavimų specifikavimo proceso kokybę. Analizės metu parinktas sprendimas turėtų būti pritaikytas programinio prototipo sudarymui. Tik tokiu būdu vartotojas galės optimaliai išnaudoti programinio prototipo teikiamas galimybes ir įgyvendinti užsibrėžtus tikslus.

Reikalavimų specifikavime šablonų pagrindu bus orientuojamasi į Volere šabloną, nes šis reikalavimų specifikavimo šablonas savo išbaigtumu ir specifikavimo savybėmis lenkia kitus šablonus. Pastarąjį šabloną galima dalinai panaudoti reikalavimų valdymo įrankiuose, o pati šablono struktūra gali būti keičiama pagal poreikius. Taigi struktūros nustatymas šablonų pagrindu ir perkėlimas į prototipą yra vienas svarbiausių uždavinių.

Toliau seka tam tikro funkcionalumo suteikimas programiniam prototipui tokiu būdu užtikrinant tikslų jo panaudojamumą reikalavimų proceso valdyme. Taigi bus sudarytas reikalavimų specifikacijos statinis struktūros modelis bei specifikavimo procesų modelis ir iširtas jų funkcionalumas.

### **3.5. Reikalavimų specifikavimo sudėties koncepcinis modelis**

Detalųjį koncepcinį modelį sudaro klasių diagrama su naudojamomis operacijomis. Sudaryta tokia klasių diagrama, kuri suteikia dinamiškumo ir išplečiamumo reikalavimų specifikavimo modifikuoto Volere šablono pagrindu prototipui.

Detalaus modelio klasių diagramoje pavaizduoti ryšiai tarp klasių, kurios čia yra su atributais ir operacijomis. Ryšiai tarp klasių yra supaprastinti iki asociacijų. Tokiu būdu šį modelį bus galima panaudoti duomenų saugyklai gauti. Reikalavimų specifikavimo klasių diagramos su operacijomis modelis pavaizduotas 3.2 paveiksle.

Suprojektuotas klasių modelis perteikia reikalavimų specifikavimo sudėties koncepcinį modelį. Remiantis modeliu, programinis prototipas turėtų padidinti išplečiamumą ir lankstumą reikalavimų specifikavime šablonų pagrindu. Įvertinimai bus pateikti eksperimentinio modelio tyrimo dalyje.

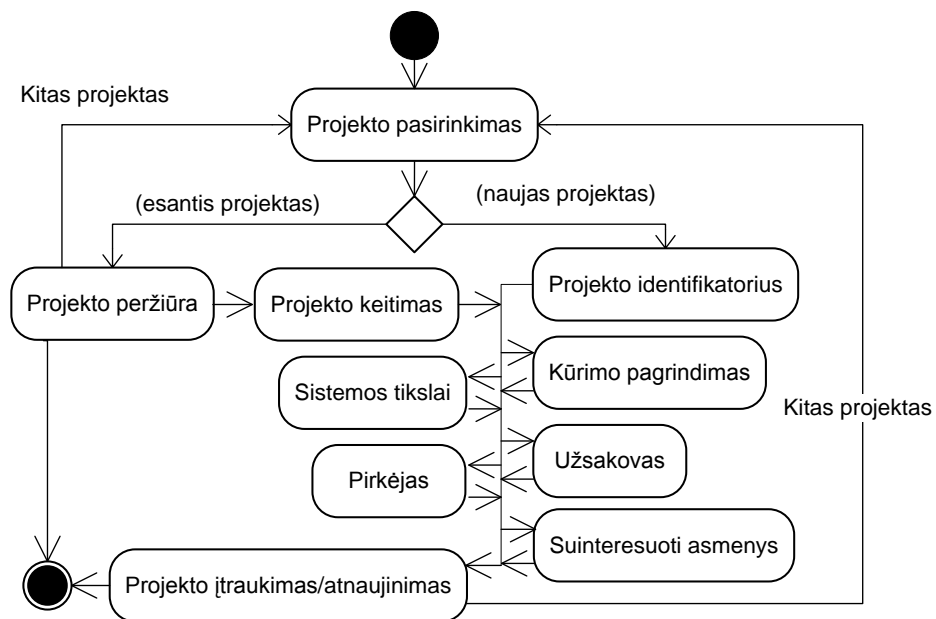




### 3.6. Specifikavimo procesų modelis

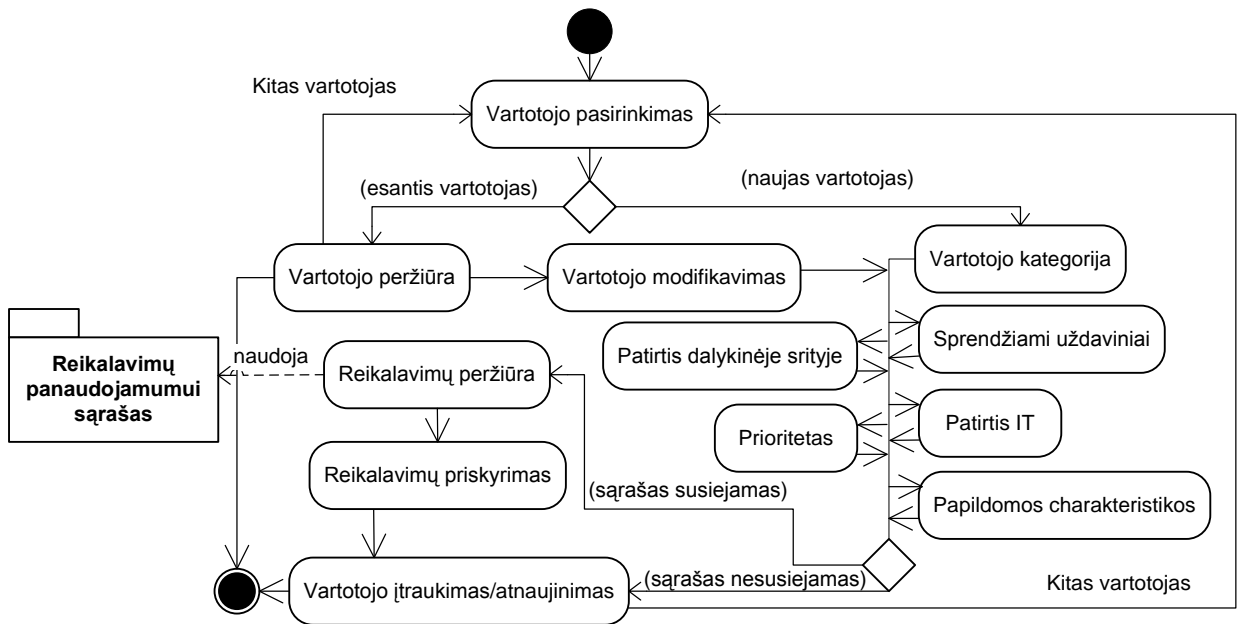
Reikalavimų specifikavimo Volere šablono pagrindu elgsenos modeliui atvaizduoti pasirinktos veiklos diagramos. Jų pagalba atvaizduojami reikalavimų specifikacijos užpildymo modelio vykdomi procesai. Specifikacijos procesų modelis būtinas teisingam duomenų saugyklos užpildymui, priežiūrai bei naudojimui. Todėl kiekvieną procesą aprašysime detaliau.

Pirmasis veiklos modelis, pavaizduotas 3.3 paveiksle, atlieka pagrindinį vaidmenį reikalavimų specifikavimo Volere šablono pagrindu procese. Visų pirma registruojamas projektas, kurio reikalavimų specifikacija turi būti sudaryta. Kol projektas neužregistruotas, jokie kiti procesai negali būti pradėti. Registruojant projektą įvardinamas projekto identifikatorius, kūrimo pagrindimas, sistemos tikslai, užsakovai, pirkėjai bei kiti suinteresuoti asmenys. Įtrauktus duomenis galima bet kada peržiūrėti bei atnaujinti.



3.3 pav. Projekto registravimo veiklos modelis

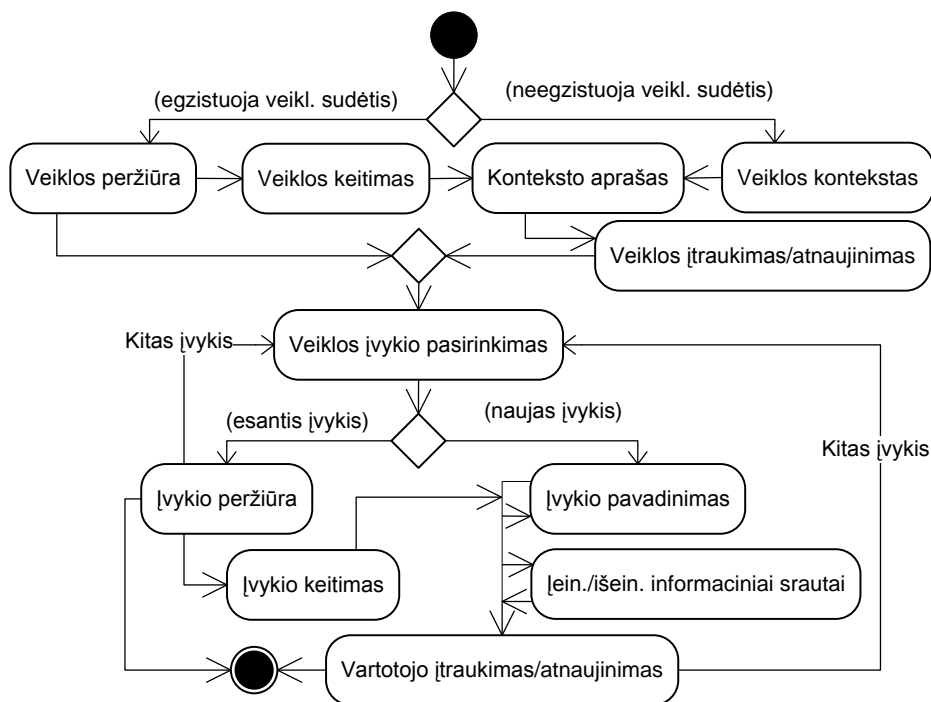
Antrasis veiklos modelis, pavaizduotas 3.4 paveiksle, yra atsakingas už asmenų, kurie betarpiškai naudosis sistema, registravimą. Įtraukiant naują vartotoją reikia nurodyti tokią informaciją: vartotojo kategoriją, sprendžiamus uždavinius, patirtį dalykinėje srityje bei informacinėse technologijose, prioritetą ir papildomas charakteristikas. Pažymėtina, jog vartotojo charakteristika turi būti įvertinta ir pildant reikalavimus panaudojamumui. Jei vėliau bus užregistruota pastarojo tipo reikalavimų, jie turi būti priskiriami atitinkamiems vartotojams.



3.4 pav. Vartotojų registravimo veiklos modelis

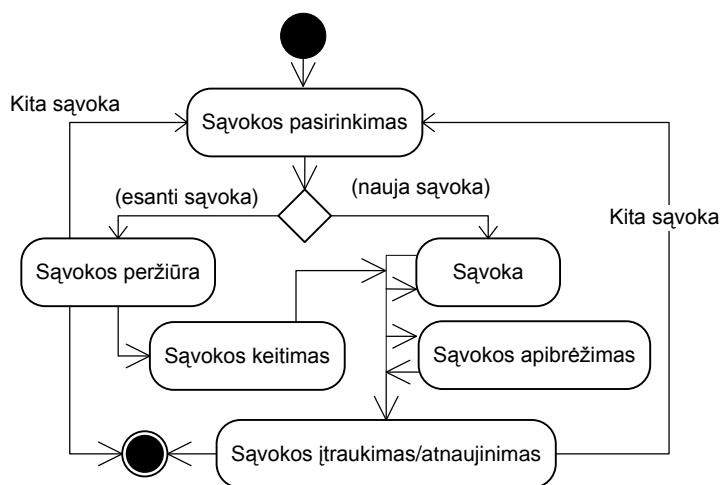
Trečiasis veiklos modelis, pavaizduotas 3.5 paveiksle, yra neatskiriama bendro specifikacijos užpildymo modelio procesų dalis. Visų pirma apibrėžiamas nagrinėjamos veiklos kontekstas bei išsamus konteksto aprašas. Veiklos kontekstas apima plačiau, nei kuriamos sistemos atliekamos funkcijos. Veiklos kontekstas apibrėžia dominančią veiklą ir jos naudojamus bei formuojamus informacijos srautus. Sėkmingai užregistravus veiklos kontekstą galima pereiti prie veiklos įvykių išskyrimo ir sąvokų registravimo terminų žodyne.

Suplanavus veiklos kontekstą, pereinama prie veiklos įvykių sąrašo sudarymo, kuris apima visus veiklos įvykius, už kuriuos yra atsakinga nagrinėjama veikla. Veiklos įvykių galima apibrėžti kaip vartotojo išskiriamus veiksmus, atliekamus veiklos metu. Įtraukiant veiklos įvykių reikalingas įvykio pavadinimas bei įeinantys ir išeinantys informacijos srautai. Tokio veiklos padalinimo tikslas – identifikuoti atskiras veiklos dalis, kurių pagrindu būtų galima nustatyti reikalavimus.



3.5 pav. Veiklos konteksto sudarymo bei įvykių fiksavimo veiklos modelis

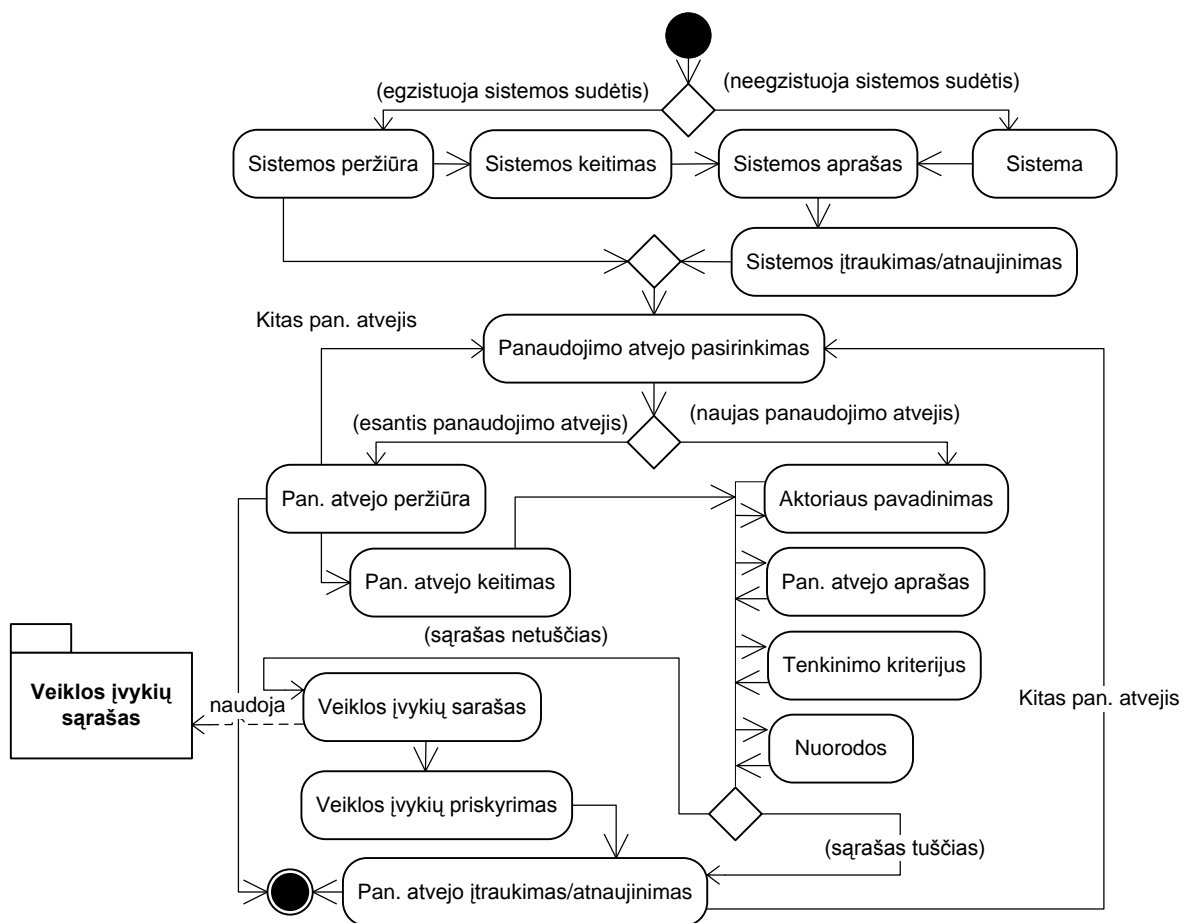
Nei vienas projektas negali apsieiti be terminų žodyno. Jame talpinamos sąvokos, kurios bus naudojamos specifikuojant reikalavimus. Įtraukiant sąvokas kiekvienai jų pateikiamas glaustas paaiškinimas. Be to, sąvokos turi būti priimtinos dalykinės srities atstovams. Žodynas naudojamas bei papildomas sistemos projektavimo metu. Terminų žodyno užpildymo sąvokomis veiklos modelis pavaizduotas 3.6 paveiksle.



3.6 pav. Terminų žodyno užpildymo bei peržiūros veiklos modelis

Penktasis veiklos modelis, pavaizduotas 3.7 paveiksle, atsakingas už sistemos sudėties įvardinimą bei panaudojimo atvejų registravimą. Visų pirma apibrėžiamos sistemos ribos bei registruojamas pilnas sistemos aprašas. Sėkmingai užregistravus sistemą galima pereiti prie panaudojimo atvejų išskyrimo ir apribojimų reikalavimams registravimui.

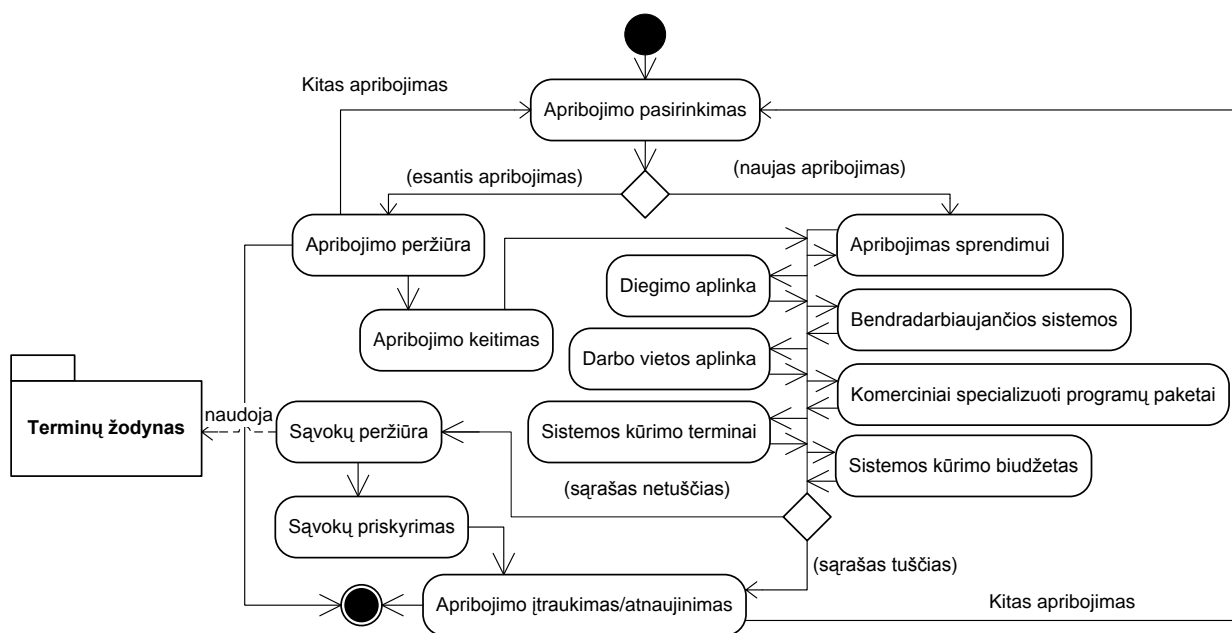
Taigi pereiname prie panaudojimo atvejų sąrašo sudarymo. Ribas tarp sistemos ir vartotojo nusako panaudojimo atvejų sąrašas. Panaudojimo atvejų sąrašas sudaromas įvertinant kiekvieną išskirtą veiklos įvykį ir kuriamos sistemos indėlių šio įvykio atžvilgiu. Įtraukiant panaudojimo atvejį pateikiamas panaudojimo atvejo numeris, vartotojo / aktoriaus pavadinimas, nuorodos, panaudojimo atvejo aprašas bei tenkinimo kriterijus. Reikia nepamiršti, jog kiekvieną panaudojimo atvejį būtina susieti su bent vienu veiklos įvykiu. Sistemos sudėties registravimo bei panaudojimo atvejų sąrašo sudarymo veiklos modelis pateiktas 3.7 paveiksle.



**3.7 pav.** Sistemos sudėties registravimo bei panaudojimo atvejų sąrašo sudarymo veiklos modelis

Šeštasis veiklos modelis, pavaizduotas 3.8 paveiksle, reikalingas apribojimų reikalavimams įtraukimui į projektą. Tai apribojimai, kurie įtakoja reikalavimų specifikaciją bei sistemos kūrimo eigą bei charakteristikas. Projekto apribojimai išskaidyti į atskiras, bet tarpusavyje

susijusias dalis. Kiekviena iš jų gali turėti savo struktūrą ir talpinti tik jai būdingus reikalavimus. Norint užregistruoti pastarojo tipo apribojimus projektui, reikia atsižvelgti į apribojimus sprendimui, diegimo aplinką, bendradarbiaujančias sistemas, darbo vietos aplinką, komercinius specializuotus programų paketus, sistemos kūrimo terminus bei biudžetą. Jei kuriame nors apribojime sutinkama terminų žodyne esanti sąvoka, ji turi būti susieta su apribojimu.

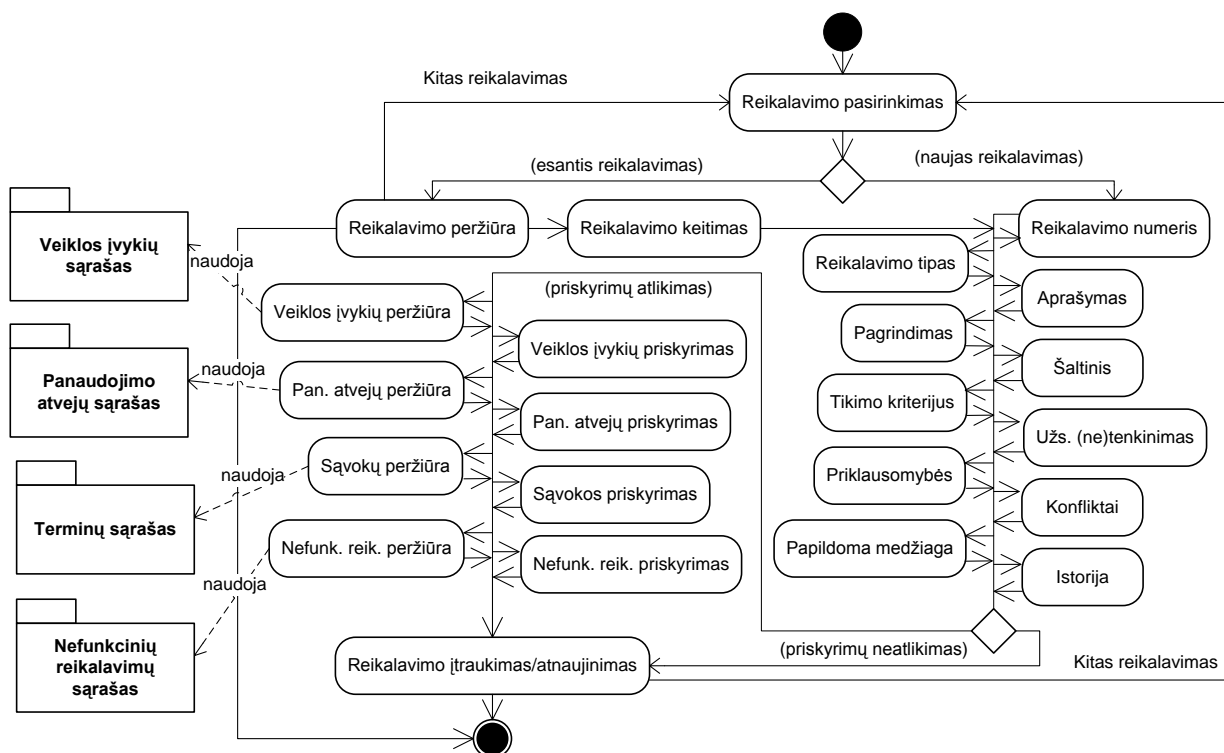


**3.8 pav.** *Apribojimų reikalavimams nustatymo veiklos modelis*

Toliau pereiname prie funkcinų ir nefunkcinų reikalavimų specifavimo etapo. Šiame etape turi būti užfiksuoti visi vartotojų iškelti reikalavimai sistemos funkcionalumui. Reikalavimų aprašas turi būti suprantamas vartotojui ir užsakovui, turi būti aiškiai suformuluotas, kad galėtų būti patikrintas. Detalumo laipsnis turi būti toks, kad užsakovas sugebėtų verifikuoti numatomą sistemos funkcionalumą. Taigi kiekvieną panaudojimo atvejį reikia peržiūrėti, įvertinti ir iš jo išskirti funkcinį reikalavimą. Kiekvienas išskirtas funkcinis reikalavimas yra fiksuojamas.

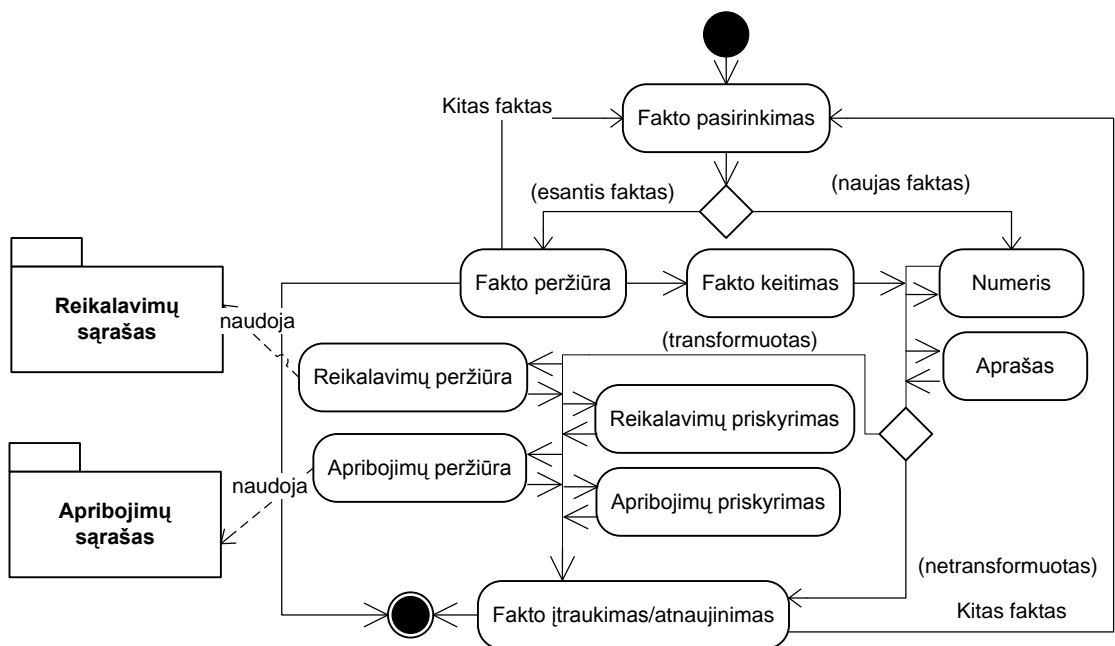
Funkciniai ir nefunkciniai reikalavimai pateikiami prisilaikant vieningos formos. Reikalavimą nusako reikalavimo numeris, tipas, aprašymas, pagrindimas, šaltinis, tinkamumo kriterijus, užsakovo tenkinimas ir netenkinimas, priklausomybės, konfliktai, papildoma medžiaga bei istorija. Svarbu tai, jog formuojant funkcinis reikalavimus panaudojamas įvykių/panaudojimo atvejų sąrašas. Be to yra galimybė funkcinis reikalavimus susieti su vienu ar keliais nefunkciniais reikalavimais. Jei kuriame nors reikalavime (funkciniame ar nefunkciniame) sutinkama terminų žodyne esanti sąvoka, ji turi būti priskiriama pastarajam reikalavimui.

Nefunkcinius sistemos reikalavimus registruojame naudodami tą pačią formą kaip ir funkciniais reikalavimams . Nefunkciniai reikalavimai nusako sistemos savybes, kuriomis ji turi pasižymėti. Kitaip tariant, tai kokybinės funkcinuose reikalavimuose numatytų funkcijų vykdymo charakteristikos, kurios dažnai atspindi vartotojo patirtį atliekant tam tikrą veiklą. Nefunkciniai reikalavimai apima šiuos tipus: reikalavimai sistemos išvaizdai, panaudojamumui, vykdymo charakteristikoms, veikimo sąlygoms, sistemos priežiūrai, saugumui, kultūriniai – politiniai reikalavimai bei teisiniai reikalavimai. Funkcinių bei nefunkcinių reikalavimų identifikavimo veiklos modelis pateiktas 3.9 paveiksle.



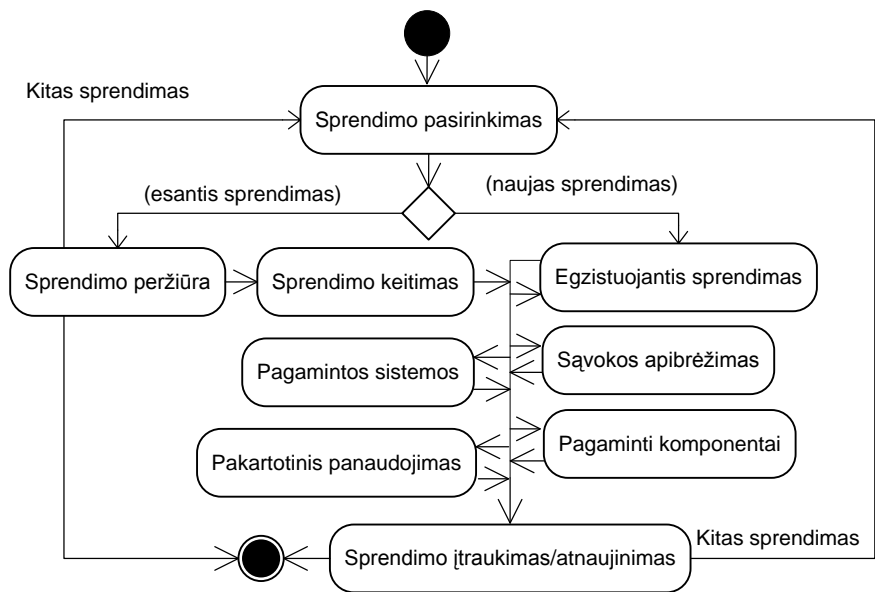
3.9 pav. Funkcinių bei nefunkcinių reikalavimų identifikavimo veiklos modelis

Baigus registruoti reikalavimus, pereinama prie svarbių faktų įtraukimo veiklos modelio, pavaizduoto 3.10 paveiksle. Tai išoriniai faktoriai, įtakojantys kuriamą sistemą, tačiau nepriklausantys iš anksto nustatomiems apribojimams, kurie įtakoja reikalavimų specifikaciją. Svarbūs faktai ne visuomet transformuojami į reikalavimus ar apribojimus, tačiau toks atvejis galimas. Jei įvyksta pastarasis atvejis, faktui reikia priskirti transformuotą reikalavimą/apribojimą. Kitaip tariant čia įtraukiami tie faktai, kurie nepatenka į kitus šablono punktus, tačiau numanoma, kad jie įtakoja reikalavimų specifikaciją.



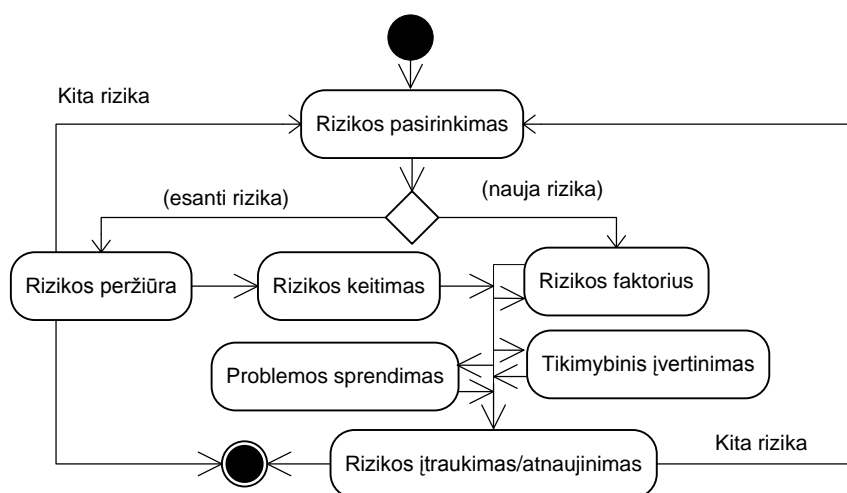
**3.10 pav.** Svarbių faktų įtraukimo veiklos modelis

Veiklos modelio, pavaizduoto 3.11 paveiksle, pagalba atliekamas egzistuojančių sprendimų įvertinimas. Egzistuojantys sprendimai gali būti įvairių tipų. Reiktų atsižvelgti į tai, ar jau yra pagamintų sistemų, kurios gali būti nupirktos, ar yra pagamintų komponentų, kuriuos galima būtų panaudoti, ar galima ką nors nukopijuoti. Egzistuojančių sprendimų esmė paremta tuo, jog geriau pakartotinas panaudojimas, nei naujo sukūrimas.



**3.11 pav.** Egzistuojančių sprendimų įvertinimo veiklos modelis

Toliau atliekama rizikos analizė. Ji reikalinga rizikos laipsnio įvertinimui. Rizika nėra visai neigiamas faktorius, nes be rizikos nebūtų vystymosi. Svarbiausia yra tai, jog rizika turi būti valdoma. Projekto valdymo metu būtina numatyti būdus, kaip kuo anksčiau identifikuoti momentus, kuomet rizika pradeda kelti problemas. Todėl sudaromas rizikingų faktorių sąrašas, šalia pateikiant tikimybinį įvertinimą, kad rizika sukels problemas bei numatytą planą sukeltoms problemoms spręsti. Sudarytas rizikos faktorių sąrašas bus panaudojamas registruojant sistemos uždavinius. Rizikos numatymo ir įvertinimo veiklos modelis pavaizduotas 3.12 paveiksle.

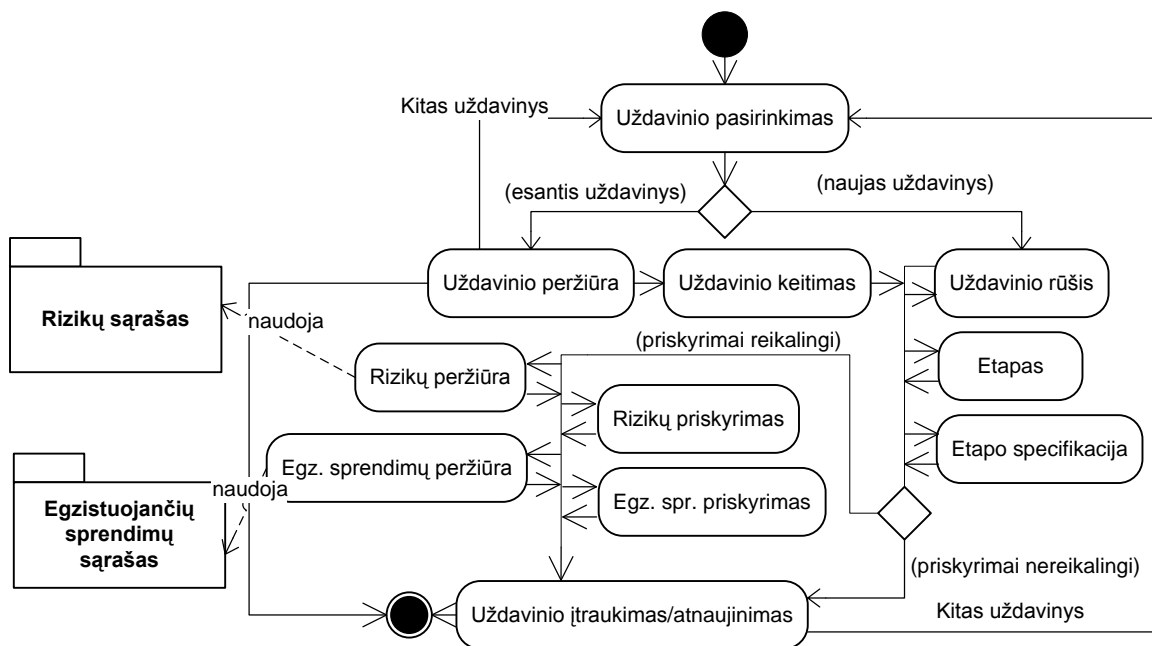


**3.12 pav.** Rizikos numatymo ir įvertinimo veiklos modelis

Sistemos uždavinių iškėlimo bei aprašymo veiklos modelis pavaizduotas 3.13 paveiksle. Pirmiausia nurodoma, kokio tipo bus uždavinys. Jei tai sistemos pateikimo uždavinys, tuomet aprašomos sistemos gyvavimo ciklo detalės bei sistemos pateikimo būdas. Tai skirta numatyti būdus sistemai pateikti, kad kiekvienas turėtų vienodą supratimą apie kuriamą sistemą. Jei uždavinio tipas yra sistemos vystymas, tuomet pateikiama sistemos kiekvienos vystymo fazės (etapo) specifikacija bei veikimo aplinkos komponentų charakterizavimas. Taip galima numatyti etapus naujos sistemos veikimo aplinkai sukurti, kad kūrimo procesą būtų galima sėkmingai valdyti.

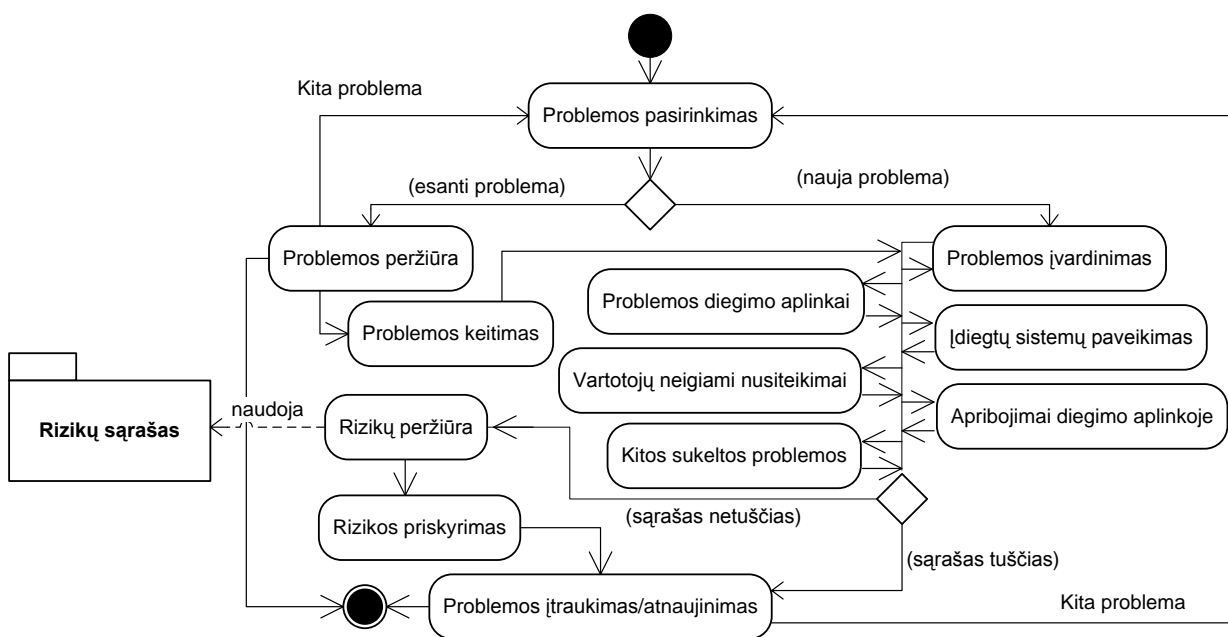
Jei kuriame nors uždavinio etape susiduriame su anksčiau identifikuota rizika, būtina etapą susieti su pastarąja rizika naudojant sudarytą rizikų sąrašą. Ta pati sąlyga galioja ir egzistuojantiems sprendimams juos susiejant su uždaviniais pasitelkus sudarytą egzistuojančių sprendimų sąrašą.





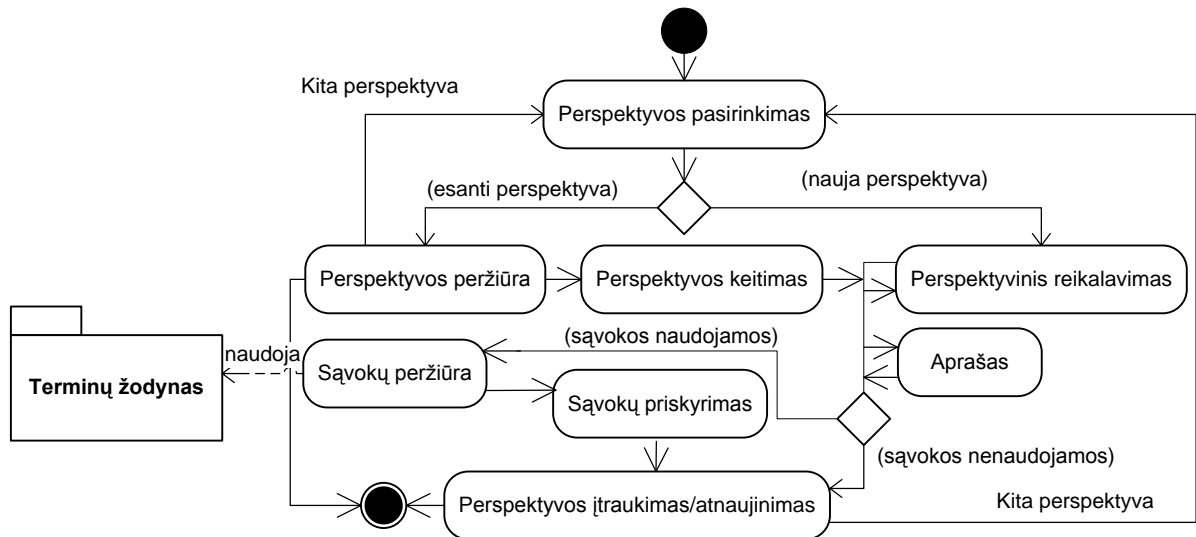
3.13 pav. Sistemos uždavinių iškėlimo bei aprašymo veiklos modelis

Baigus aprašyti sistemos uždavinius, pereinama prie naujų problemų, kurias gali sukelti sistema, įtraukimo veiklos modelio, pavaizduoto 3.14 paveiksle. Šiame žingsnyje įvertinamos ir įvardinamos sistemos diegimo aplinkai sukeltos problemos, poveikis jau įdiegtoms sistemoms, vartotojų neigiamas nusiteikimas prieš sistemą, kokie apribojimai egzistuoja numatomoje diegimo aplinkoje, galintys kliudyti naujai sistemai bei kito pobūdžio problemos. Jei kurią nors problemą vienu ar kitu būdu įtakoja rizika, būtina problemą susieti su pastarąja rizika naudojant suformuotą rizikų sąrašą.



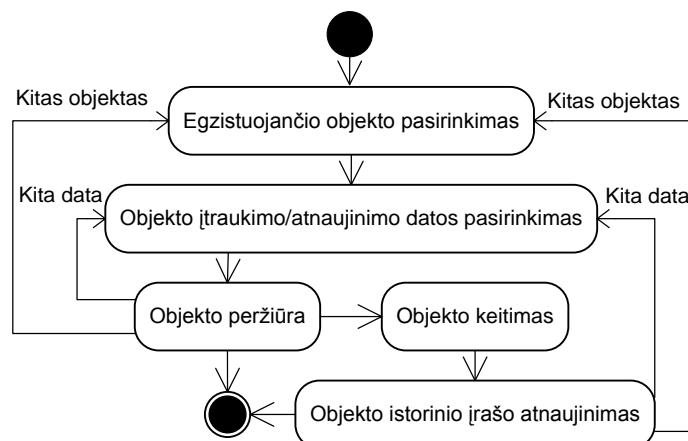
3.14 pav. Naujų problemų įtraukimo veiklos modelis

Toliau fiksuojami šioje sistemoje neįvertinami reikalavimai, kurie gali būti įvertinti perspektyvinėje sistemos versijoje ar kitoje sistemoje. Tai gali būti bet kokio tipo reikalavimai. Perspektyvinių reikalavimų nustatymo veiklos modelio, pavaizduoto 3.15 paveiksle, tikslas - užregistruoti svarbius, bet nerealizuotus reikalavimus, kurie "laukia savo eilės". Jei kuriame nors perspektyviniame reikalavime sutinkama terminų žodyne esanti sąvoka, ji turi būti susieta su pastaruoju reikalavimu.



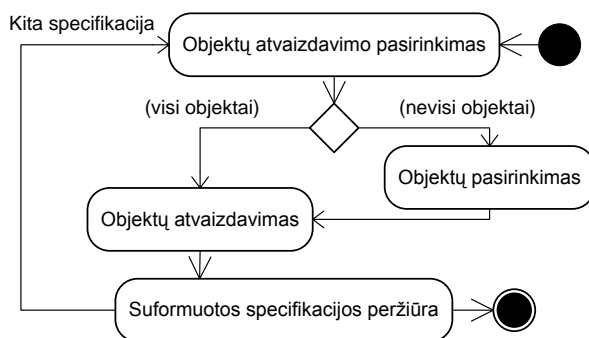
**3.15 pav.** *Perspektyvinių reikalavimų nustatymo veiklos modelis*

Reikalavimų sekamumas yra labai svarbi reikalavimų specifikuojamo proceso dalis. Jo dėka galima sekti reikalavimų kitimo istoriją. Šiame programiniame prototipe bus suteikiama galimybė sekti visų tipų objektus, atitinkančius modifikuoto Volere šablono punktus. Tik į projektą įtraukus tam tikrą objektą, pradedamas jo atnaujinimų fiksavimas. Kiekvieną kartą atnaujinus objektą, senoji jo versija išlieka saugykloje ir esant reikalui bet kada gali būti peržiūrėta ir netgi modifikuota. Projekto objektų sekamumo veiklos modelis pavaizduotas 3.16 paveiksle.



**3.16 pav.** *Projekto objektų sekamumo veiklos modelis*

Atlikus visų projekto objektų identifikavimą ir suregistravimą į saugyklą, galima pereiti prie sukaupytų objektų galutinio atvaizdavimo. Tai leidžia atvaizduoti viso projekto arba atskirų projekto objektų specifikaciją. Atvaizduotą specifikaciją gali naudoti informacinių sistemų kūrimo specialistas. Projekto dalinės arba pilnos specifikacijos atvaizdavimo veiklos modelis, pateiktas 3.17 paveiksle.

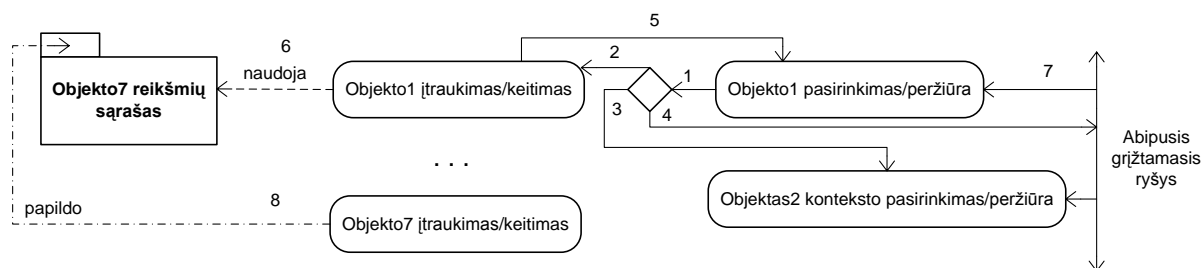


**3.17 pav.** Projekto dalinės arba pilnos specifikacijos atvaizdavimo veiklos modelis

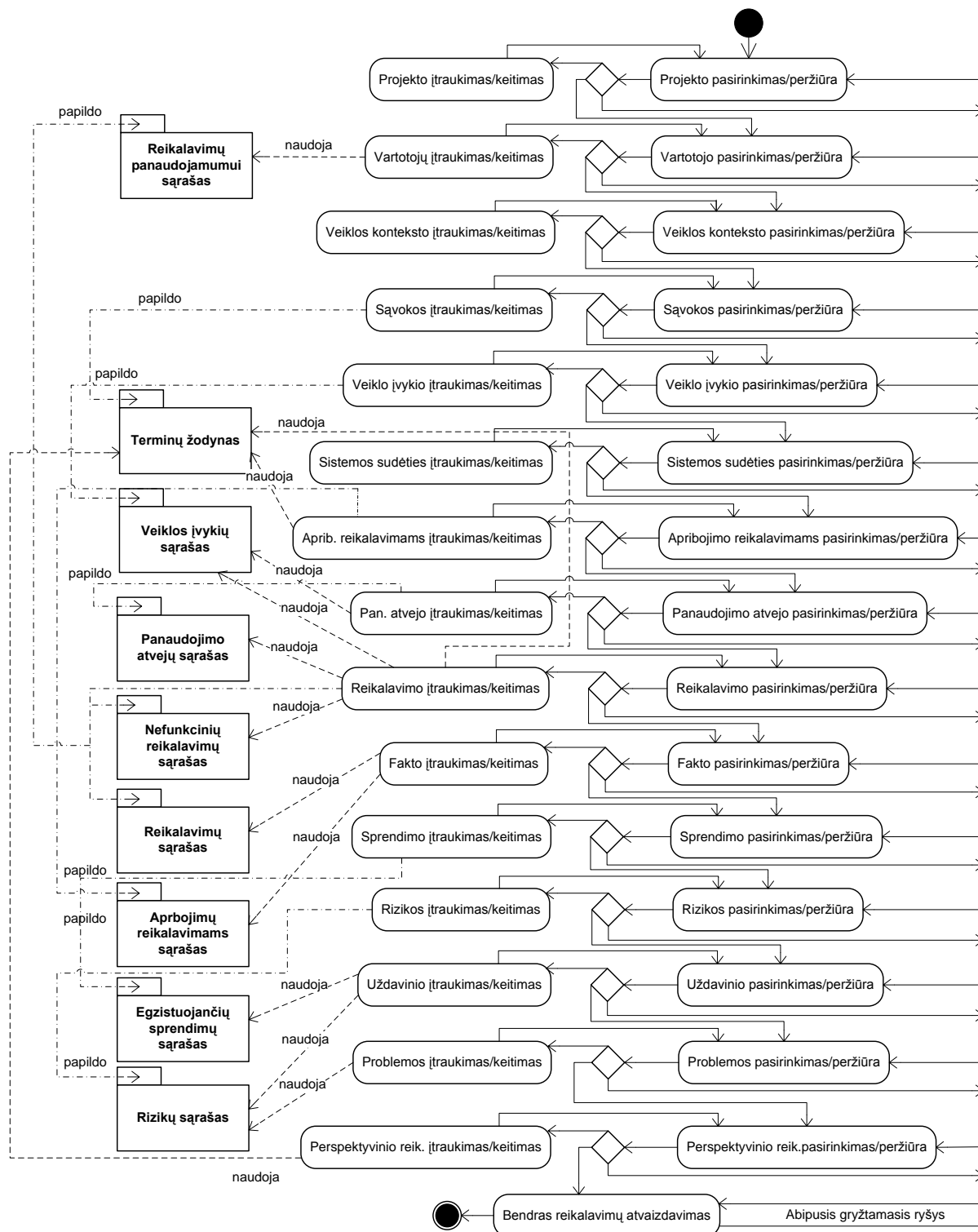
Taigi 3.19 paveiksle pateikiamas reikalavimų specifikavimo modifikuoto Volere šablono pagrindu apibendrintas veiklos modelis, apimantis ir apjungiantis visus anksčiau šiame skyriuje aptartus atskirus veiklos modelius (3.3 – 3.17 paveiksiai).

Reikalavimų specifikavimo modifikuoto Volere šablono pagrindu veiklos modelis prasideda projekto įtraukimu ir baigiasi bendru reikalavimų atvaizdavimu pakeliui įtraukiant visus tarpinius veiklos objektus. Abipusis grįžtamasis ryšys užtikrina visų šablono punktų susietumą. Tiesa, sugrįžimas galimas tik į tuos objektus, kurie bent kartą registruoti projekte.

Veiklos modelyje naudojami aštuonių tipų ryšiai: (1) objekto pasirinkimas/peržiūra, (2) objekto įtraukimas/keitimas, (3) perėjimas prie kito tipo objekto, (4) patekimas į abipusį grįžtamąjį ryšį, (5) įtraukto/pakeisto objekto pasirinkimas/peržiūra, (6) objekto įtraukime/keitime naudojamas kito objekto reikšmių sąrašas, (7) objekto pasirinkimas/peržiūra atėjus abipusiu grįžtamoju ryšiu, (8) objekto įtraukimas/keitimas papildo jo reikšmių sąrašą. Pagrindinio veiklos modelio sutartinis žymėjimas pateiktas 3.18 paveiksle.



**3.18 pav.** Sutartinis žymėjimas



3.19 pav. Reikalavimų specifikavimo Volere šablono pagrindu apibendrintas veiklos modelis

Pagal sudarytą reikalavimų specifikavimo Volere šablono pagrindu proceso modelį buvo sėkmingai surinkti testiniai informavimo sistemos mokymo įstaigai reikalavimai.

### **3.7. Reikalavimų specifikuojamo koncepcinio modelio apibendrinimas**

Koncepcinėje dalyje sudarytas reikalavimų specifikuojamo modifikuoto Volere šablono pagrindu koncepcinis modelis, apimantis projektuojamo prototipo architektūrą, išsamia klasių su naudojamomis operacijomis diagramą bei specifikuojamo procesų modelį.

Visų pirma suprojektuota prototipo struktūrinė diagrama, nusakanti jo vaidmenį reikalavimų inžinerijos valdymo procese. Diagramoje pateikiamas vartotojų grupių ir projektuojamo prototipo tarpusavio susietumas ir priklausomybės.

Reikalavimų specifikuojamo sudėties koncepcinio modelio atvaizdavimui pasirinkta detali klasių su naudojamomis operacijomis diagrama. Sudaryta klasių diagrama bus panaudota duomenų saugyklai gauti. Tai turėtų suteikti dinamiškumo ir išplečiamumo reikalavimų specifikuojamo modifikuoto Volere šablono pagrindu prototipui.

Reikalavimų specifikuojamos užpildymo modelio vykdomų procesų atvaizdavimui pasirinktos veiklos diagramos. Kiekvienas pastarųjų procesų pavaizduotas atskira veiklos diagrama, o visų procesų aibė perteikta apibendrinta veiklos diagrama. Specifikuojamos procesų modelis būtinas teisingam duomenų saugyklos užpildymui, priežiūrai bei naudojimui.

## 4. Reikalavimų specifikavimo programinio prototipo realizacija

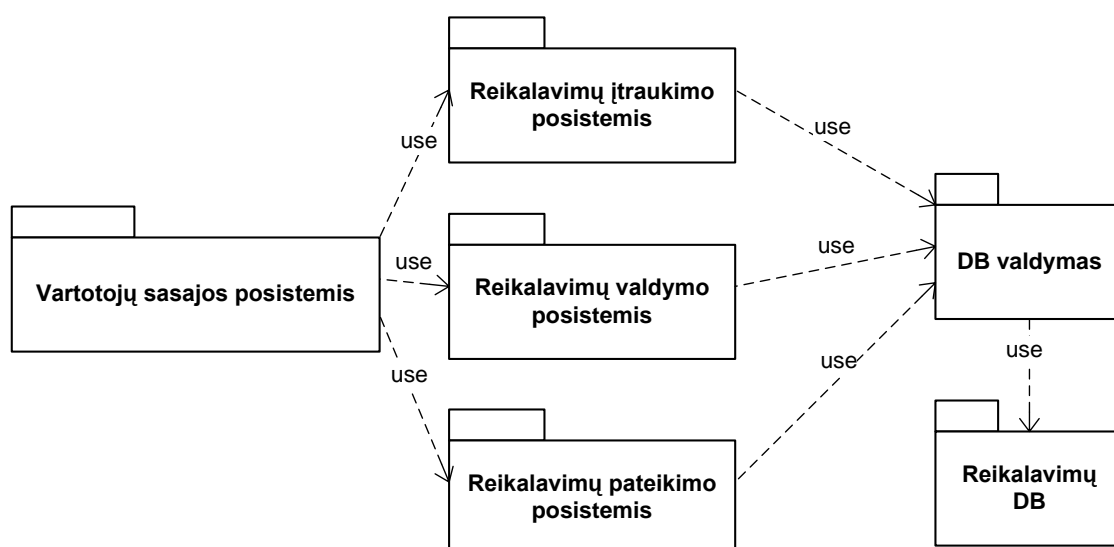
Reikalavimų specifikavimo šablonų pagrindu modelio projektas pateikia realų programinio prototipo projektą pagal koncepcinėje dalyje suprojektuotus modelius. Programinio prototipo projektas pateikiamas sistemos architektūra, duomenų bazės schema bei realizacijos modeliais.

### 4.1. Reikalavimų specifikacijos statinės struktūros modelis

Visų pirma suprojektuojama reikalavimų specifikavimo šablonų pagrindu programinio prototipo architektūra. Ji pateikiama per loginės prototipo architektūros diagramą bei modifikuoto Volere šablono sudėties duomenų saugyklos schemą. Pastarieji modeliai vaizduoja nekintančią prototipo architektūrą.

#### 4.1.1. Loginė prototipo architektūra

Prototipas padalintas į penkias dalis. Jį sudaro vartotojų sąsajos posistemis, reikalavimų įtraukimo posistemis, reikalavimų valdymo posistemis bei reikalavimų pateikimo posistemis. Pastarieji posistemiai privalo sąveikauti su duomenų baze. Todėl sudarytas dar vienas posistemis – tai duomenų bazės valdymo posistemis, kuris jungia reikalavimų apdorojimo posistemius su duomenų baze. Reikalavimų specifikavimo šablonų pagrindu prototipo loginės architektūros modelis pateiktas 4.1 paveiksle.



4.1 pav. Reikalavimų specifikavimo prototipo loginės architektūros modelis

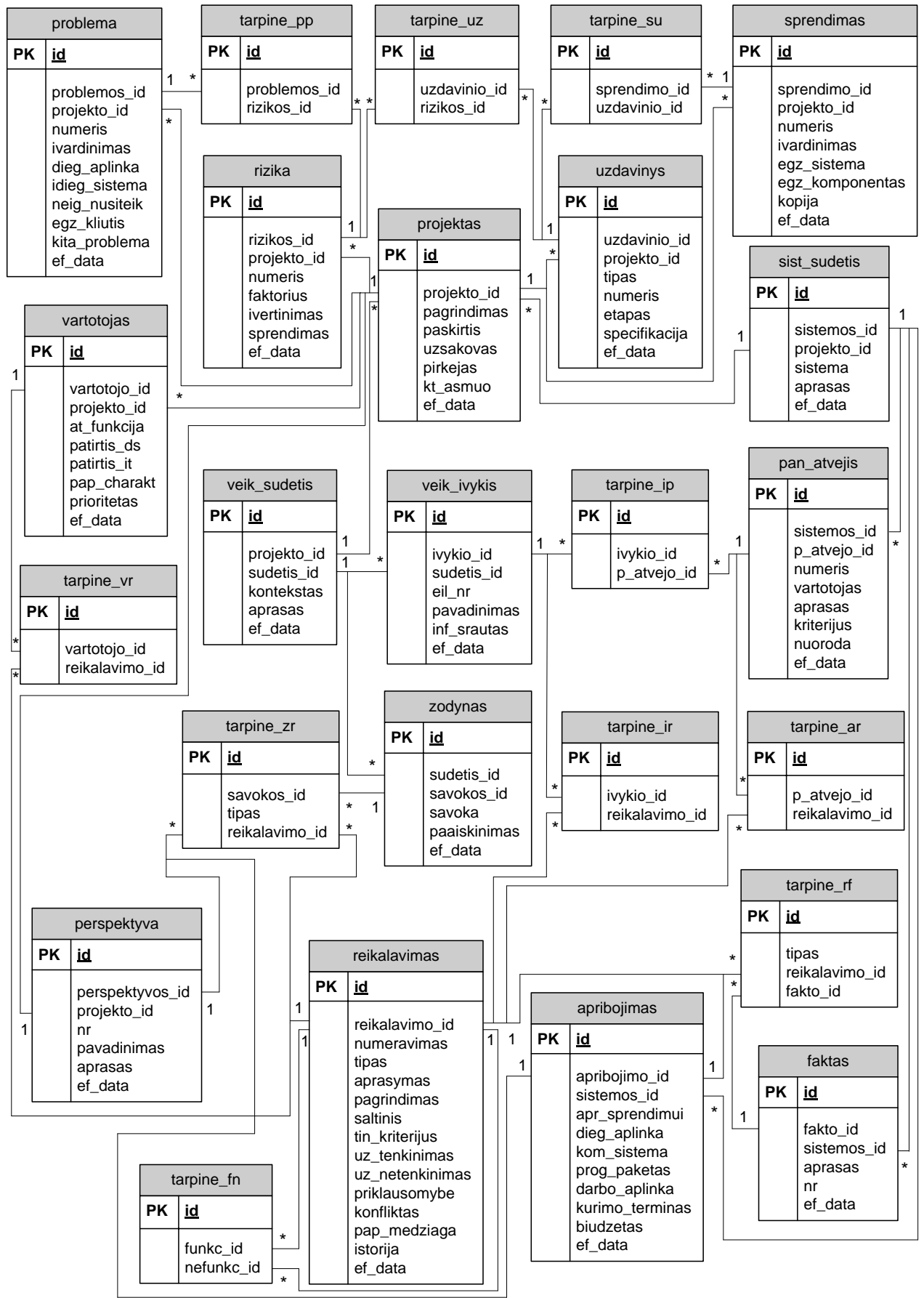
Vartotojų sąsajos posistemis nusprendžia, kuris kitas posistemis bus panaudotas iš trijų reikalavimų apdorojimo posistemų. Pasirinkus vieną iš pastarųjų posistemų naudojamas duomenų bazės valdymo posistemis, kuris bendrauja su duomenų baze.

#### **4.1.2. Reikalavimų objektų saugyklos schema**

Duomenų bazės schema (saugyklos schema) suprojektuota remiantis atidžiai išanalizuotu Volere reikalavimų specifikavimo šablonu bei jo struktūra. Šablonas buvo modifikuotas atsižvelgiant į analitinėje dalyje iškeltus tikslus bei vartotojų reikalavimus. Ši schema talpina visus pagrindinius reikalavimų tipus, kuriuos apima Volere reikalavimų specifikavimo šablonas. Suprojektuota duomenų bazės schema (saugyklos schema) pateikta 4.2 paveiksle.

Saugyklos schema būtina reikalavimų specifikacijos užpildymo proceso modelio naudojimui. Schema sudarančios lentelės talpina atitinkamus Volere reikalavimų specifikavimo šablono punktų duomenis. Ryšių tarp lentelių dėka galima išvelgti šablono punktų susietumą tarpusavyje. Visa tai reikalinga norint nuosekliai ir tiksliai specifikuoti reikalavimus.

Prireikus nesudėtingai galima papildyti saugyklos modelį lentelėmis ir ryšiais, atvaizduojančiais pakitusius ar naujus modifikuoto Volere reikalavimų specifikavimo šablono punktus. Reikėtų atkreipti dėmesį į tai, jog schemoje nuimti pavadinimai nuo ryšių. Ryšiai sudėti prie tų lentelės atributų, su kuriais susijungiama.



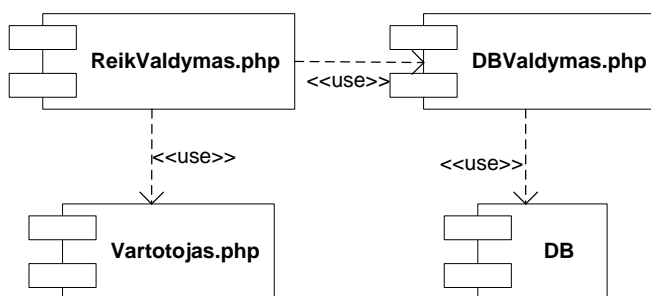
4.2 pav. Reikalavimų specifikavimo Volere šablono pagrindu duomenų bazės schema



## 4.2. Reikalavimų specifیکavimo prototipo realizacija

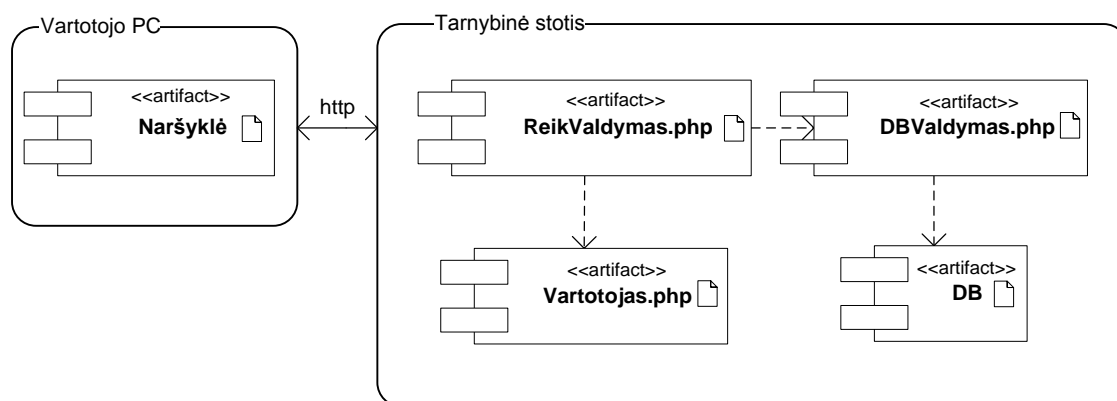
Prototipo realizacijos modelis pateikia ne tik programinių komponentų architektūrą, bet ir įdiegimo modelį. Prototipo išskaidymas komponentais suteikia jam daugiau nepriklausomybės bei padidina išplečiamumą. Kiekvienas komponentas programuojamas ir bandomas atskirai. Tik tuomet jis įtraukiamas į programinį prototipą. Prireikus galima atlikti pakeitimus reikiamuose komponentuose nepaveikiant kitų pavienių komponentų ar jų sudarytos visumos.

Komponentai, pateikti diagramoje, realizuoti „php“ išplėtimo internetiniais puslapiais. Suprojektuoti vartotojų sąsajos, duomenų bazės, reikalavimų valdymo bei duomenų bazės valdymo komponentai. Reikalavimų specifیکavimo prototipo komponentų diagrama pavaizduota 4.3 paveiksle.



4.3 pav. Reikalavimų specifیکavimo komponentų diagrama

Prototipo projektavimo metu paaiškėjo, jog reikalingi du mazgai norint užtikrinti jo veikimą. Pirmasis mazgas yra vartotojo kompiuteris. Tuo pačiu metu gali dirbti keletas vartotojų iš nutolusių darbo vietų. Tam tereikia turėti kompiuterį su interneto prieiga bei internetine naršykle, o už visą kitą atsakingas HTTP protokolas, bendraujantis su antruoju mazgu – tarnybine stotimi. Tarnybiniėje stotyje yra patalpintas programinis prototipas su visais jam reikalingais sisteminiais failais bei duomenų baze. Reikalavimų specifیکavimo prototipo diegimo diagrama pavaizduota 4.4 paveiksle.



**4.4 pav.** Reikalavimų specifikuojimo diegimo diagrama

### **4.3. Reikalavimų specifikuojimo prototipo realizavimo apibendrinimas**

Reikalavimų specifikuojimo šablonų pagrindu realizacijos metu sudarytas programinio prototipo statinės struktūros modelis, apimantis jo loginę architektūrą. Taip pat suprojektuota ir realizuota reikalavimų saugykla (duomenų bazė), remiantis atidžiai išanalizuotu Volere reikalavimų specifikuojimo šablonu bei jo struktūra. Reikalavimų saugykla gauta iš koncepcinėje dalyje sudaryto detalaus klasių modelio su naudojamomis operacijomis.

Programinio prototipo realizacija pateikiama ne tik komponentų, bet ir įdiegimo diagramomis. Programinių komponentų diagrama parodo prototipo išskaidymą nepriklausomomis dalimis, kurios didina jo išplečiamumą bei atskirų komponentų daugkartinį panaudojimą. Įdiegimo diagramos tikslas - perteikti prototipo naudojimui būtinų mazgų išskyrimą, sandarą ir tarpusavio sąveiką.

Šio etapo metu pasiektas vienas pagrindinių uždavinių - koncepcinėje dalyje sudaryto reikalavimų specifikuojimo užpildymo modelio procesų perkėlimas į programinį prototipą. Reikia nepamiršti, jog specifikuojimo procesų modelis būtinas teisingam duomenų saugyklos užpildymui, priežiūrai bei naudojimui.

## **5. Reikalavimų specifikuojamo programinio prototipo tyrimas**

Šiame skyriuje aprašysime sukurto reikalavimų specifikuojamo Volere šablono pagrindu programinio prototipo veikimą. Visų pirma reikalinga dalykinė sritis, kuriai bus renkami testiniai reikalavimai bei sudaryta reikalavimų specifikuojama naudojant programinį prototipą. Vėliau bus atliekamas prototipo įvertinimas bei pateikiamas testavimo apibendrinimas.

### **5.1. Pasirinktos dalykinės srities aprašas**

Greitai besivystančiame šiuolaikiniame pasaulyje naudojamos įvairios komunikacijos priemonės. Mobiliosios komunikacijos - tai didžiulis civilizacijos žingsnis, atveriantis dar platesnes komunikavimo galimybes. Nuolat besiplečiantis interneto tinklas ir vis mažėjančios mobiliųjų paslaugų kainos sudaro puikų pagrindą SMS žinučių vystymuisi ir gyvavimui Lietuvos mobiliojoje rinkoje. SMS panaudojimo galimybės įgauna vis didesnę svarbą ne tik asmeninėje komunikacijoje, bet ir organizacinėje veikloje.

Informavimo procesas tampa ypatingai svarbus šiuolaikinėje ugdymo įstaigoje. Analizuojami ugdymo įstaigoje vykstantys procesai bei informavimo trumposiomis žinutėmis galimybės įgyvendinimo perspektyva šioje aplinkoje. Informavimas elektroninėmis priemonėmis atitinka rekomenduojamą taikyti „pirmos dienos pranešimo modelį“ bei nurodomas kaip viena iš efektyvių moksleivių lankomumo, pažangumo monitoringo, mokyklos personalo komunikacijos su tėvais, formų.

Informavimo trumposiomis žinutėmis sistemos, orientuotos į ugdymo įstaigos poreikius, realizavimas pasitarnautų moksleivių ugdymo proceso tobulinimui bei efektyvinimui bei būtų vienu iš galimų greito mokyklos bendruomenės (mokytojų, tėvų bei auklėtinių) informavimo problemos sprendimų.

Taigi nuspręsta surinkti reikalavimus informavimo trumposiomis žinutėmis sistemai, orientuotai į ugdymo įstaigos poreikius.

### **5.2. Programinio prototipo veikimo tyrimo eiga**

Reikalavimų specifikuojamo modifikuoto Volere šablono pagrindu programinio prototipo atskiri komponentai buvo testuojami jų kūrimo metu naudojant iš anksto susidarytus testavimo atvejus, talpinančius specifinius duomenis. Pastarasis testavimas nepateikiamas tolimesniuose skyriuose.

Ištestavus visus komponentus atskirai, pradedamas viso realizuoto programinio prototipo veikimo tyrimas. Tyrimas atliekamas pasirinkus dalykinės srities problematiką. Visų pirma vykdomas reikalavimų surinkimo procesas. Pasibaigus pastarajam etapui pereinama prie surinktų reikalavimų apdorojimo bei reikalavimų specifikacijos sudarymo, naudojant koncepcinėje dalyje susidarytą reikalavimų specifikacijos užpildymo metodiką. Tuomet sudarytą reikalavimų specifikaciją modifikuoto Volere šablono pagrindu galima peržiūrėti, atspausdinti bei pateikti IS kūrimo specialistui.

### 5.3. Programinio prototipo langai

Šiame skyriuje bus pateikiami sukurto reikalavimų specifikavimo modifikuoto Volere šablono pagrindu programinio prototipo langai su testiniais duomenimis. Tai yra programinis prototipas, o ne komercinis įrankis, todėl tam tikri grafiniai, programiniai ar duomenų tikrinimo elementai, kurie nedaro įtakos prototipo funkcionalumo veikimui, gali būti nepilnai išbaigti. Toliau pateiksime pagrindinius reikalavimų specifikacijos užpildymo proceso langus su duomenimis.

Pirmasis žingsnis prototipo naudojime yra prisijungimas. Jo metu identifikuojama kuriai vartotojų grupei priklauso prisijungęs specialistas. Jei tai IS reikalavimų specifikavimo specialistas, tuomet jam suteikiamos visos reikalavimų kūrimui bei peržiūrai reikalingos teisės. Jei prisijungė IS kūrimo specialistas, tuomet jis gali atlikti tik reikalavimų peržiūrą. Prisijungimo forma pateikta 5.1 paveiksle

**PROGRAMINIS PROTOTIPAS**

---



**Prisijungimo vardas:**

**Slaptažodis:**

[Pamiršai slaptažodį?](#)

---

Autorius: IFM-3/4 Egidijus Zinkevičius  
©2008 Visos teisės saugomos

5.1 pav. Prisijungimo prie programinio prototipo forma

Pirmasis prototipo langas, kurį sutinkame tik prisijungę prie sistemos, pavaizduotas 5.2 paveiksle, yra pagrindinis reikalavimų specifikavime. Prisijungęs IS reikalavimų specifikavimo specialistas visų pirma registruoja projektą, kurio reikalavimų specifikacija bus toliau daroma. Jei projektas neužregistruotas - kiti specifikavimo procesai yra draudžiami. Registruojant projektą būtinai įvardiname projekto identifikatorių, kūrimo pagrindimą bei sistemos tikslus kaip tai pateikta 5.2 paveiksle. Likę trys užsakovų, pirkėjų bei kitų suinteresuotų asmenų laukai nėra privalomi, tačiau be jų specifikacija bus nepilna, todėl užpildome ir juos. Čia pat talpinama ir besikeičianti projekto įrašo istorija, kurią galima ne tik peržiūrėti, bet ir keisti.

5.2 pav. Reikalavimų specifikavimo diegimo diagrama

Antrajame prototipo lange, pavaizduotame 5.3 paveiksle, užregistruojame asmenis, kurie betarpiškai naudosis specifikuojama sistema. Pirmiausia įtraukiame aukščiausią prioritetą turinčius vartotojus, nes jų nuomonė ir pageidavimai yra svarbiausi kuriamoje sistemoje. Vartotojo kategorija, sprendžiami uždaviniai ir prioritetas yra privalomi, o likusieji laukai įtraukiami pagal poreikį. Papildomas vartotojo charakteristikas būtinai įvertiname ir pildant reikalavimus panaudojamumui. Jei vėliau bus užregistruota pastarojo tipo reikalavimų, tuomet būtina vartotoją susieti su jais naudojant reikalavimų panaudojamumui sąrašą.

**Vartotojai**

**Projektas: Informavimas\_SMS**

Pasirinkite vartotoją:

---

**Vartotojo kategorija**

**Vartotojo sprendžiami uždaviniai**

**Patirtis dalykinėje srityje**

**Patirtis informacinėse technologijose**

**Papildomos vartotojo charakteristikos**

**Prioritetas**

**Vartotojo charakteristikos atsižvelgtos šiuose reikalavimuose:**

**Reikalavimai panaudojamumui:**

**5.3 pav.** *Vartotojų, kurie naudosis sistema, registravimo forma*

Trečiame programinio prototipo lange, pavaizduotame 5.4 paveiksle, registruojame būsimos sistemos pagrindus. Pirmiausia apibrėžiame nagrinėjamos veiklos kontekstą bei pateikiame išsamų konteksto aprašą. Veiklos kontekstas apibrėžia dominančią veiklą ir jos naudojamus bei formuojamus informacijos srautus. Pastarieji srautai bus reikalingi formuojant veiklos įvykius. Abu informaciniai laukai yra būtini, todėl juos užpildome. Baigę registruoti veiklos kontekstą, pereiname prie veiklos įvykių išskyrimo etapo.

Taigi tolimesnis etapas apima visų veiklos įvykių išskyrimą bei įtraukimą į specifikaciją, už kuriuos atsakinga nagrinėjama veikla. Įtraukiant išskirtą veiklos įvykį reikalingas numeravimas, įvykio pavadinimas, atspindintis vartotojo išskiriamus veiksmus, bei įeinantys ir išeinantys informacijos srautai. Beje visi projekte talpinami veiklos įvykiai keliauja į specialų sąrašą, kurio elementus panaudosime specifikuojant funkcinius reikalavimus. Taigi veiklos įvykiai yra pirmoji funkcinių reikalavimų sudedamoji dalis.

Šiame etape sutinkame ir pirmuosius veiklos kontekste naudojamus terminus. Visi pastarieji terminai turi patekti į terminų žodyną. Kaip tai padaryti, paaiškinta kitoje pastraipoje.

**Veiklos sudėtis ir veiklos įvykiai**

**Projektas:** Informavimas\_SMS

**Kontekstas:** Informavimo infrastruktūra Atnaujinti

**Konteksto aprašas**

Infrastruktūra apjungia mokyklos bendruomenę, švietimo ministeriją ir administraciją bei nurodo

**Pasirinkite įvykį:** 1. Vartotojas atlieka SMS žinu Tęsti

---

**Įvykio numeris**

1

**Įvykio pavadinimas**

Vartotojas atlieka SMS žinutės siuntimą.

**Įeinantys/išeinantys informacijos srautai**

Visa SMS žinutės siuntimui būtina informacija (in)

Įtraukti Atnaujinti

---

**Veiklos įvykių istorija:** istorinis įrašas Gera! Keisti

**5.4 pav.** Veiklos konteksto sudarymo bei įvykių fiksavimo langas

Programinio prototipo lange, pavaizduotame 5.5 paveiksle, matome terminų žodyną. Jame talpiname sąvokas, kurios yra arba bus naudojamos specifikuojant reikalavimus. Įtraukiant kiekvieną sąvoką pateikiame glaustą paaiškinimą, kuris būtų priimtinas dalykinės srities atstovams. Žodyną naudojame bei papildome viso sistemos projektavimo metu. Veiklos konteksto apraše, veiklos įvykiuose, sistemos panaudojimo atvejuose, apribojimuose reikalavimams bei pačiuose funkcinuose/nefunkcinuose/perspektyviniuose reikalavimuose sutinkamos neaiškios sąvokos turi būti įtrauktos į žodyną ir, naudojant terminų žodyno sąrašą, priskirtos atitinkamiems reikalavimų objektams.

**Terminų žodynas**

**Projektas:** Informavimas\_SMS

**Pasirinkite sąvoką:** SMS Tęsti

---

**Sąvoka**

SMS

**Sąvokos apibrėžimas**

Trumpoji žinutė, siunčiama pagrinde mobiliųjų telefonų pagalba. Standartinis ilgis 160 simbolių.

Įtraukti Atnaujinti

---

**Sąvokų istorija:** istorinis įrašas Gera! Keisti

**5.5 pav.** Terminų žodyno užpildymo langas

Kitas programinio prototipo langas, pavaizduotas 5.6 paveiksle, apima sistemos ribų nustatymą bei panaudojimo atvejų registravimą. Taigi pirmiausia nustatome būsimos sistemos ribas bei pateikiame pilną sistemos aprašą. Kitaip tariant, šio žingsnio tikslas – iš anksčiau sudaryto veiklos konteksto išskirti būsimos sistemos ribas. Tuomet seka panaudojimo atvejų išskyrimas ir jų sąrašo sudarymas, kuris bus naudojamas funkcinių reikalavimų specifikavime.

Taigi pereiname prie panaudojimo atvejų, nustatančių ribas tarp sistemos ir vartotojo, sąrašo sudarymo. Svarbu nepamiršti, jog panaudojimo atvejų sąrašą sudarome įvertinant kiekvieną išskirtą veiklos įvykį ir kuriamos sistemos indėlį šio įvykio atžvilgiu. Šiame žingsnyje dominuoja vartotojo/aktoriaus pavadinimas bei panaudojimo atvejo aprašas. Kiekvienas su panaudojimo atveju susijęs reikalavimas turi atitikti panaudojimo reikalavimo tenkinimo kriterijų. Veiklos įvykis ar įvykiai, kurių pagrindu yra užregistruotas panaudojimo atvejis, turi būti priskiriami pastarajam atvejui naudojant ankstesniuose etapuose sudarytą veiklos įvykių sąrašą.

**Sistemos sudėtis ir pan. atvejai**

**Projektas:** Informavimas\_SMS

**Sistema:** Informavimo sistema

**Sistemos aprašas**

Informavimo sistema apima už numatytas informavimo funkcijas, vartotojų administravimą ir SMS limitus.

**Panaudojimo atvejis:** 1. Siųsti SMS žinutę

**Panaudojimo atvejo numeris**

1

**Vartotojo/aktoriaus pavadinimas**

Mokinys, Mokytojas, Administratorius

**Panaudojimo atvejo aprašas**

Siųsti SMS žinutę

**Panaudojimo atvejo tenkinimo kriterijus**

Siunčiant SMS žinutę bus naudojamas automatinis parašas.

**Nuorodos**

Nepateikiama.

**Priskirti veiklos įvykiai**

1. Vartotojas atlieka SMS žinutės siuntimą.

**Galimi veiklos įvykiai**

1. Vartotojas atlieka SMS žinutės siuntimą.  
2. Vartotojas atlieka paprastos žinutės siuntimą.  
3. Kontaktų administravimas.

**Panaud. atvejų istorija:** istorinis įrašas

**5.6 pav.** Sistemos sudėties registravimo bei panaudojimo atvejų sąrašo sudarymo langas



Šeštasis prototipo langas, pavaizduotas 5.7 paveiksle, suteikia apribojimų reikalavimams, kurie įtakoja reikalavimų specifikaciją, sistemos kūrimo eigą bei charakteristikas, įtraukimo į projektą galimybę. Prototipe realizuoti septyni apribojimų reikalavimams tipai: apribojimai sprendimui, diegimo aplinka, bendradarbiaujančios sistemos, darbo vietos aplinka, komerciniai specializuoti programų paketai, sistemos kūrimo terminai bei biudžetas. Vadovaudamiesi šiuo principu užregistruojame reikalavimų apribojimams rinkinį. Visas apribojimuose sutinkamas neaiškias sąvoką, kaip ir anksčiau užregistruojame terminų žodyne ir priskiriame pastarajam apribojimui naudojant terminų žodyno sąrašą.

Apribojimai reikalavimams

**Projektas:** Informavimas\_SMS

**Apribojimas:** 19. Sistema turi būti pritaikyta į ▼ Tęsti

---

**Apribojimas sprendimui**

Sistema turi būti pritaikyta ir įdiegta į jau egzistuojančią mokymo įstaigos infrastruktūrą. ▲▼

**Diegimo aplinka**

Sistema bus patalpinta Web serveryje. Reikalinga integracija su LDAP duomenų baze. ▲▼

**Bendradarbiaujančios sistemos**

Sistemos vartotojų autentifikavimui bus panaudota LDAP duomenų bazės integracija. ▲▼

**Komerciniai specializuoti programų paketai**

SMS žinučių siuntimui realizuoti bus panaudotas komercinis NowSMS programinis paketas. ▲▼

**Darbo vietos aplinka**

"Dauguma vartotojų neturės spausdintuvų" - sistema nepateiks jokių spausdintinių rezultatų. ▲▼

**Sistemos kūrimo terminai**

Sistemos kūrimo pradžia 2009 06 30. ▲▼

Sistemos kūrimo pabaiga 2009 08 30. ▲▼

**Sistemos kūrimo biudžetas**

Sistemos kūrimo darbams ir palaikymui skirta XX tūkstančių litų biudžetas. ▲▼

**Apribojime naudojami terminai:**

SMS - Trumpoji žinutė, siunčiama pagrinde mobiliųjų telefonų pagalba. S

**Sąvokos, esančios terminų žodyne:**

SMS - Trumpoji žinutė, siunčiama pagrinde mobiliųjų telefonų pagalba. S  
Paprasta žinutė - Pranešimas, siunčiamas sistemos ribose naudojant tar  
PHP - Dinaminė interpretuojama programavimo kalba.

Įtraukti Atnaujinti

---

**Panaud. atvejų istorija:** istorinis įrašas ▼ Gerai Keisti

**5.7 pav.** Apribojimų reikalavimams nustatymo ir pateikimo langas

Baigus specifikuoti apribojimus reikalavimams, pereiname prie funkcinių ir nefunkcinių reikalavimų specifikavimo etapo. Pastarieji reikalavimai yra svarbiausi išpildant projektuojamos sistemos funkcionalumą, todėl juos specifikuosime prisilaikant vieningos formos, pavaizduotos 5.8 ir 5.9 paveiksluose, bei naudodami ankstesniuose etapuose sudarytus sudėtinius sąrašus. Funkcinio reikalavimo detalumo laipsnis turi leisti užsakovui tiksliai verifikuoti numatomą sistemos funkcionalumą. Kitu atveju sistemos funkcionalumas gali netenkinti užsakovo pageidavimų.

Funkcinius ir nefunkcinius reikalavimus identifikuojame trimis parametrais: (1) reikalavimo numeriu, (2) tipu bei (3) įvykiu ar panaudojimo atveju. Reikalavimas bus stebimas per visą sistemos kūrimo laiką, todėl numeravimas turi išlikti unikalus. Funkcinio reikalavimo tipą atitinka skaitmuo 9, o nefunkcinio - skaitmenys nuo 10 iki 17, kurių kiekvienas pagal Volere šabloną atitinka tam tikrą nefunkcinio reikalavimo punktą. Trečiajam reikalavimo identifikavimo parametro priskyrimui naudojame įvykių/panaudojimo atvejų sąrašą. Reikalavimo aprašymas, pagrindimas bei tikimo kriterijus yra privalomi formos laukai, nes perteikia visą reikalavimo esmę. Likę laukai yra neprivalomi, tačiau kiekvieno pastarųjų laukų užpildymas specifikacijoje padeda tiksliau perteikti vartotojų poreikius kuriamai sistemai. Be to, esant poreikiui, funkcinius reikalavimus galime susieti su vienu ar keliais nefunkciniais reikalavimais naudojant nefunkcinių reikalavimų sąrašą.

**Funkciniai ir nefunkciniai reikalavimai**

**Projektas: Informavimas\_SMS**

Reikalavimas: 1. (9) Informavimo sistema turi siųsti

---

Reikalavimas #:  Reikalavimo tipas:

**Įvykis/panaudojimo atvejis #:**

1. Vartotojas atlieka SMS žinutės siuntimą  
1. Siųsti SMS žinutę

**Aprašymas**

Informavimo sistema turi siųsti SMS žinutes

**Pagrindimas**

Informavimas SMS žinutėmis yra vienas efektyviausių ir greičiausių bendravimo būdų.

**Šaltinis**

Mokymo įstaigos informacinio skyriaus vadovas.

**Tikimo kriterijus**

Sistema tiesiogiai perduoda SMS gavėjui, o nesėkmės atveju apie tai informuojamas siuntėjas.

Užsakovo tenkinimas:  Užsakovo netenkinimas:

**Priklausomybės**

Nėra.

**Konfliktas**

Nėra.

**Papildoma medžiaga**

NowSMS programinio paketo specifikacija pasiekama adresu [www.nowsms.com/specification](http://www.nowsms.com/specification).

**Istorija**

Reikalavimas pirmą kartą buvo iškeltas apklausos su atsakingu mokyklos asmeniu metu.

**Reikalavimo aprašyme naudojami terminai:**

SMS - Trumpoji žinutė, siunčiama pagrįdė mobiliųjų telefonų pagalba Standartinis il

**Reikalavimas susijęs su šiais nefunkciniais reikalavimais:**

3. Informavimo sistema turi subegėti išsiųsti ne mažiau 10 sms žinučių per minutę.

---

Reikalavimų istorija: istorinis įrašas

5.8 pav. Funkcinių reik. pateikimo langas

**Funkciniai ir nefunkciniai reikalavimai**

**Projektas: Informavimas\_SMS**

Reikalavimas: 3. (12) Informavimo sistema turi su

---

Reikalavimas #:  Reikalavimo tipas:

**Įvykis/panaudojimo atvejis #:**

1. Vartotojas atlieka SMS žinutės siuntimą  
1. Siųsti SMS žinutę

**Aprašymas**

Informavimo sistema turi subegėti išsiųsti ne mažiau 10 sms žinučių per minutę.

**Pagrindimas**

Esant dideliame vartotojų skaičiui informavimo sistema turi atlaikyti didesnes apkrovas.

**Šaltinis**

Mokymo įstaigos informacinio skyriaus vadovas.

**Tikimo kriterijus**

Siunčiamų SMS žinučių kiekis laukimo režime neviršija 200.

Užsakovo tenkinimas:  Užsakovo netenkinimas:

**Priklausomybės**

Jei keičiasi išsiunčiamų SMS kiekis per minutę, tai keičiasi ir laukiančiųjų SMS limitas.

**Konfliktas**

Nėra.

**Papildoma medžiaga**

Nuoroda į dažniausiai užduodamus klausimus [www.nowsms.com/faq](http://www.nowsms.com/faq)

**Istorija**

Nepateikiama.

**Reikalavimo aprašyme naudojami terminai:**

SMS - Trumpoji žinutė, siunčiama pagrįdė mobiliųjų telefonų pagalba Standartinis il

**Sąvokos, esančios terminų žodyne:**

SMS - Trumpoji žinutė, siunčiama pagrįdė mobiliųjų telefonų pagalba Standartinis il  
Paprasta žinutė - Pranešimas, siunčiamas sistemos ribose naudojant tam pritaiky

---

Reikalavimų istorija: istorinis įrašas

5.9 pav. Nefuncinių reik. pateikimo langas

Įvertinus projektuojamos sistemos būsimą funkcionalumą bei kokybines charakteristikas, pereiname prie svarbių faktų įtraukimo prototipo lango, pavaizduoto 5.10 paveiksle. Šio žingsnio metu įtraukiame visus išorinius faktorius, įtakojančius kuriamą sistemą, tačiau nepriklausančius iš anksto nustatomiems apribojimams, kurie įtakoja reikalavimų specifikaciją. Čia užregistruojame ir tokius reikalavimus, kurie bus realizuoti vėlesnėse sistemos versijose. Jei įvyksta pastarasis atvejis, faktui reikia priskirti transformuotą reikalavimą/apribojimą naudojant atitinkamus duomenų sąrašus.

**Faktai**

Projektas: Informavimas\_SMS

Faktas: 1. Esant nepakankamam NowSMS pa Tęsti

Numeris  
1

Aprašas  
Esant nepakankamam NowSMS paketo funkcionalumui, pereiti prie kito alternatyvaus paketo naudojimo.

Egzistuojantys reikalavimai:  
1. (9) Informavimo sistema turi siųsti SMS žinutes  
3. (12) Informavimo sistema turi subegėti išsiųsti ne mažiau 10 sms žinu

Egzistuojantys apribojimai:  
19. Sistema turi būti pritaikyta ir įdiegta į jau egzistuojančią mokymo įstaig

Įtraukti Atnaujinti

Fakto istorija: istorinis įrašas Gerai Keisti

**5.10 pav.** Svarbių faktų įtraukimo langas

Prototipo langą, pavaizduotą 5.11 paveiksle, naudojame egzistuojančių sprendimų įtraukimui ir pateikimui. Egzistuojantys sprendimai būna įvairių tipų, todėl rekomenduojama kiekvieną sprendimą registruoti atskirai prie jo pateikiant tikslią informaciją vienu ar keliais toliau išvardintais aspektais: ar jau yra pagamintų sistemų, kurios gali būti nupirktos, ar yra pagamintų komponentų, kuriuos galima būtų panaudoti, ar galima ką nors nukopijuoti. Šiame žingsnyje vadovaujamės principu – geriau pakartotinas panaudojimas nei naujo sukūrimas.

**Esantys sprendimai**

**Projektas:** Informavimas\_SMS

**Egzistuojantis sprendimas** 1. SMS žinučių siuntimo sp

---

**Numeris**

1

**Egzistuojantis sprendimas**

SMS žinučių siuntimo sprendimas

**Pagamintos sistemos, kurios gali būti nupirktos**

Komercinis SendSMS programinis paketas, skirtas SMS žinučių siuntimui.

**Pagaminti komponentai, kuriuos galima būtų panaudoti**

Nėra.

**Pakartotinis panaudojimas**

Nėra.

**5.11 pav.** Egzistuojančių sprendimų pateikimo langas

Kitas programinio prototipo langas, pavaizduotas 5.12 paveiksle, reikalingas rizikos laipsnio įvertinimui. Rizika ne visada yra neigiamas faktorius, tačiau svarbiausia yra tai, jog mes turime ją valdyti. Tą atliekame numatant būdus, kaip kuo anksčiau identifikuoti momentus, kuomet rizika pradeda kelti problemas. Rizikas registruojame tam tikru formatu pateikiant rizikos faktorių, tikimybinį įvertinimą, kad rizika sukels problemas bei numatytą planą sukeltoms problemoms spręsti. Tik tokiu būdu aprašę rizikas, galime sumažinti išlaidas rizikos padariniams šalinti. Be to sudarytą rizikos faktorių sąrašą naudosime registruojant sistemos uždavinius bei naujas problemas, kurias gali sukelti nauja sistema.

**Rizikos**

**Projektas:** Informavimas\_SMS

**Rizika** 1. SMS įkainių padidėjimas

---

**Numeris**

1

**Rizikos faktorius**

SMS įkainių padidėjimas

**Tikimybinis įvertinimas**

5

**Problemos sprendimas**

Rasti alternatyvų variantą, užtikrinantį palankius SMS siuntimo įkainius.

---

**Rizikos istorija:** istorinis įrašas

**5.12 pav.** Rizikos numatymo ir įvertinimo langas

Sistemos uždavinių prototipo langas pavaizduotas 5.13 paveiksle. Čia galimi dviejų tipų uždaviniai. Kai registruojame sistemos pateikimo uždavinį, tuomet būtinai aprašome sistemos gyvavimo ciklo detales bei sistemos pateikimo būdus. Kai registruojame sistemos vystymo uždavinį, tuomet būtinai pateikiame sistemos kiekvienos vystymo fazės specifikaciją bei veikimo aplinkos komponentų charakterizavimą. Jei kuriame nors uždavinio etape susiduriame su anksčiau identifikuota ar nauja rizika, būtina etapą susieti su pastarąja rizika naudojant atitinkamą sąrašą. Analogiški veiksmai taikomi ir egzistuojančių sprendimų priskyrimui.

**Uždaviniai**

**Projektas:** Informavimas\_SMS

**Uždavinys:** 1. [P] Duomenų perkėlimo etapas (kor) Tęsti

---

**Uždavinio rūšis:** Sistemos pateikimas

**Numeris:** 1

**Etapas:** Duomenų perkėlimo etapas (konversija).

**Etapo specifikacija:**  
 Dabartiniai vartotojų duomenys saugomi LDAP duomenų bazėje. Dalis pastarųjų vartotojų duomenų turi būti perkelti į mysql duomenų bazę.

**Priskirtos rizikos:**  
 2. [10] Reikalavimų specifikacijos pasikeitimai realizavimo fazėje

**Galimos rizikos:**  
 1. [5] SMS įkainių padidėjimas  
 2. [10] Reikalavimų specifikacijos pasikeitimai realizavimo fazėje

**Egzistuojantys sprendimai:**  
 1. SMS žinučių siuntimo sprendimas  
 2. Paprastų žinučių siuntėjo alternatyva

Įtraukti Atnaujinti

**5.13 pav.** *Sistemos uždavinių iškėlimo bei aprašymo langas*

Prieš paskutinis žingsnis reikalavimų specifikavime modifikuoto Volere šablono pagrindu vykdomas 5.14 paveiksle pavaizduotame prototipo lange. Čia pateikiame naujas problemas, kurias gali sukelti sistema. Tai sistemos diegimo aplinkai sukeltos problemos, poveikis jau įdiegtoms sistemoms, vartotojų neigiamas nusiteikimas prieš sistemą, apribojimų egzistavimas numatomoje diegimo aplinkoje bei kitos problemos. Jei kurią nors problemą vienu ar kitu būdu įtakoja rizika, būtinai ją susiejame su problema naudojant anksčiau sudarytą rizikų sąrašą.

**Naujos problemos**

**Projektas:** Informavimas\_SMS

**Problema:** 1. Su SMS siuntimu susijusios problemr

---

**Numeris**

1

**Problemos įvardinimas**

Su SMS siuntimu susijusios problemos.

**Kokias problemas nauja sistema sukels jos diegimo aplinkai?**

Bet koks SMS limitų padidinimas vartotojams sąlygoja mokymo įstaigos biudžetą.

**Ar nauja sistema paveiks jau įdiegtas sistemas?**

Padidėjusi LDAP vartotojų duomenų bazės apkrova.

**Ar kas nors iš esamų vartotojų gali neigiamai nusiteikti prieš naują sistemą?**

Mokymo įstaigos mokiniai gali būti nepatenkinti tėvų informavimo atveju.

**Kokie apribojimai, galintys kliudyti naujai sistemai, egzistuoja numatomoje diegimo aplinkoje?**

Techninė įranga, skirta SMS siuntimui, nėra pakankamai greita apdoroti didelį SMS kiekį.

**Ar nauja sistema sukels kokių nors kitų problemų?**

Mokymo įstaigos biudžetas gali nukentėti dėl didelio išsiunčiamų SMS žinučių kiekio.

**Priskirtos rizikos**

1. (5) SMS įkainių padidėjimas

---

**Galimos rizikos:**

1. [5] SMS įkainių padidėjimas  
2. [10] Reikalavimų specifikacijos pasikeitimai realizavimo fazėje

**5.14 pav.** *Naujų problemų pateikimo langas*

Paskutinis žingsnis, registruojant reikalavimų objektus, pavaizduotas 5.15 paveiksle. Čia pateikiame šioje sistemoje neįvertinamus reikalavimus, kurie gali būti įvertinti perspektyvinėje sistemos versijoje ar kitoje sistemoje. Tai gali būti bet kokio tipo reikalavimai, todėl juos registruojame natūralia kalba netaikant jokios specialios formos. Kaip ir ankstesniuose reikalavimų specifikavimo etapuose, taip ir šiame priskiriame terminų žodyno sąvokas, jei jos įeina į pastarojo reikalavimo apibrėžimą ar paaiškinimą.

**Perspektyviniai reikalavimai**

**Projektas:** Informavimas\_SMS

**Perspektyvinis reik.** 2. MMS žinučių siuntimas ir ga

---

**Numeris**

2

**Perspektyvinis reikalavimas**

MMS žinučių siuntimas ir gavimas

**Aprašas**

Informavimo sistema turėtų galimybę ne tik siųsti, bet ir gauti MMS žinutes. Tai itin svarbu

**Perspektyviniame reikalavime naudojami terminai:**

MMS - Vaizdo žinutė, siunčiama mobiliųjų telefonų pagalba.

**Sąvokos, esančios terminų žodyne:**

SMS - Trumpoji žinutė, siunčiama pagrinde mobiliųjų telefonų pagalba  
 Paprasta žinutė - Pranešimas, siunčiamas sistemos ribose naudojant  
 PHP - Dinaminė interpretuojama programavimo kalba.

---

**Perspek. reik. istorija:** istorinis įrašas

**5.15 pav.** *Perspektyvinių reikalavimų pateikimo langas*

Kaip jau buvo minėta, reikalavimų sekamumas yra itin svarbus reikalavimų specifikavimo procese, nes atsiranda galimybė sekti reikalavimų kitimo istoriją viso projekto metu. Kiekvieno šiame poskyryje aprašyto reikalavimų objekto istorija randame pasinaudojus tam skirta forma, pavaizduota 5.16 paveiksle. Pastaroji forma sutinkamas kiekviename lange, kuriame registruojami vieno ar kito tipo reikalavimų objektai. Tik į projektą įtraukus tam tikrą objektą, pradedamas jo kitimų fiksavimas. Kiekvieną kartą, kai atnaujiname objektą, senoji jo versija išlieka saugykloje ir esant reikalui bet kada galime peržiūrėti ir netgi modifikuoti išlaikant senąją datą ar priskiriant naują.

**Projekto istorija:** istorinis įrašas

istorinis įrašas  
 2008-12-05 17:34:52  
 2008-12-03 18:20:26

**5.16 pav.** *Projekto objektų sekamumo langas*



Bet kuriuo reikalavimų specifikavimo metu galime peržiūrėti jau užregistruotų reikalavimų specifikacijas tokioje formoje, kokia ji bus baigus registruoti visus reikalavimų objektus. Specifikacijos vaizdavimui naudojame specialią formą, pateiktą 5.17 paveiksle. Čia galime sugeneruoti viso projekto arba atskirų projekto reikalavimų objektų specifikaciją. Be to yra galimybė nurodyti į specifikaciją pateksiančių reikalavimų pradžios datą ir pabaigos datą. Kitaip tariant sugeneruotoje specifikacijoje bus tik pasirinktam laikotarpiui galiojantys reikalavimai.

**5.17 pav.** Projekto dalinės arba pilnos specifikacijos pateikimo langas

Sudaryta reikalavimų specifikacija modifikuoto Volere šablono pagrindu gali būti pateikiama ir XML formatu. Tai suteikia galimybę prototipo pagalba sugeneruotos specifikacijos panaudojimui kituose įrankiuose, kurių veikimas paremtas pastaruoju formatu. XML formatu sugeneruotos reikalavimų specifikacijos dalis pateikiama 5.18 paveiksle.

```
<?xml version="1.0" encoding="UTF-8" ?>
<root>
  <PROJEKTAS>
    <id>15</id>
    <projekto_id>Informavimas_SMS</projekto_id>
    <pagrindimas>Naudojamos priemonės neišsprendžia greito mokyklos bendruomenės informavimo problemas.</pagrindimas>
    <paskirtis>Informavimo sistema užtikrins greitą mokyklos bendruomenės informavimą SMS žinutėmis.</paskirtis>
    <uzsakovas>Kauno "XX" mokyklos informacinių sistemų skyriaus vadovas Jonas Jonaitis, mob. 867712345.</uzsakovas>
    <pirkejas>Kauno "XX" mokyklos direktorius Petras Petraitis, mob. 868698765, Kepenų g. 28 Kaunas.</pirkejas>
    <kt_asmuo>Nepriklausomas veiklos srities ekspertas Antanas Antanaitis, mob. 864512123.</kt_asmuo>
    <ef_data>2008-12-05 17:34:52</ef_data>
  </PROJEKTAS>
  <VARTOTOJAI>
</root>
```

**5.18 pav.** Projekto specifikacijos atvaizdavimas XML formatu

## 5.4. Reikalavimų specifikuojimo programinio prototipo įvertinimas

Sukurtą reikalavimų specifikuojimo modifikuoto Volere šablono pagrindu programinį prototipą įvertinsime atsižvelgdami į analizės metu ištirtus reikalavimų valdymo įrankius. Tačiau reiktų nepamiršti, jog darbo tikslas buvo sukurti programinį prototipą, o ne komercinis reikalavimų valdymo įrankis, todėl, savaime aišku, funkcionalumo gausos atžvilgiu jis neprilygs nei vienam reikalavimų valdymo gigantui.

Sudaryta palyginamoji programinio prototipo ir reikalavimų valdymo įrankių vertinimo lentelė. Lyginimas atliekamas remiantis analizės metu išryškėjusiais reikalavimų specifikuojimo aspektais:

- šablonų taikymas – konkrečiu atveju pilnas Volere reikalavimų specifikuojimo šablono pritaikymas;
- metodika – reikalavimų specifikuojimo Volere šablono pagrindu užpildymo proceso darna;
- susietumas - reikalavimų objektų tarpusavio susietumas;
- sekamumas – reikalavimų objektų registravimo ir atliktų modifikacijų sekimas iki dabartinės būsenos;
- dokumentavimo galimybės – ataskaitų formavimo įvairovė: reikalavimų objektų, jų galiojimo laikotarpio datos bei ataskaitos formato pasirinkimas;
- prieinamumas – galimybė pasiekti įrankį iš įvairių vietų, o ne tik iš tam tikro kompiuterio;
- išplečiamumo galimybės – papildomo funkcionalumo suteikimas, naujų reikalavimų objektų įtraukimas;
- intuityvi sąsaja – langų, formų bei nuorodų (mygtukų) išdėstymas. Analitiko specialisto sėkmingo pasinaudojimo įrankiu be specialaus apmokymo tikybė;
- integracija – įrankio pagalba sukurto produkto panaudojimas kitose sistemose;

Reikalavimų valdymo įrankių ir programinio prototipo palyginimas pateiktas 5.1 vertinimo lentelėje. Vertinimui naudojama skalė, kurią sudaro balų sistema nuo vieno iki penkių. Balai išreiškiami žvaigždutėmis, kur ★ reiškia vieną balą, o ☆ - nulį balų.

Reikėtų atkreipti dėmesį ir į tai, jog kriterijai, pagal kuriuos vertinsime įrankius, nėra vienareikšmiškos lygio, ir kiekvieno jų indėlis reikalavimų specifikuojime šablonų pagrindu yra atitinkamos svarbos.

5.1 lentelė. Sukurto programinio prototipo ir analogiškų įrankių palyginimas

Įrankis Kriterijus	RequisitePro	OptimalTrace	Doors	Progr. prot. (kūrėjas)	Progr. prot. (ekspertai)
Šablonų taikymas	★★★★☆	★★★☆☆	★★★★☆	★★★★★	★★★★★
Metodika	★★★☆☆	★★★☆☆	★★★★☆	★★★★☆	★★★★☆
Susietumas	★★★★☆	★★★★☆	★★★★☆	★★★★☆	★★★☆☆
Sekamumas	★★★★☆	★★★★☆	★★★★★	★★★★☆	★★★★★
Dokumentavimo galimybės	★★★★☆	★★★★☆	★★★★☆	★★★★☆	★★★★☆
Prieinamumas	★★★★★	★★★☆☆	★★★★☆	★★★★★	★★★★★
Išplečiamumo galimybės	★★★☆☆	★★★☆☆	★★★★☆	★★★★☆	★★★☆☆
Intuityvi sąsaja	★★★☆☆	★★★★☆	★★★★☆	★★★★☆	★★★★★
Integracija	★★★★★	★★★★☆	★★★★★	★★★☆☆	★★★☆☆

Iš 5.1 vertinimo lentelės matome, jog sukurtas reikalavimų specifikavimo Volere šablono pagrindu programinis prototipas daugiau ar mažiau tenkina visus analizės dalyje išskeltus reikalavimų valdymo įrankių vertinimo kriterijus. Tam tikrų savybių atžvilgiu programinis prototipas gan smarkiai lenkia netgi komercinius įrankius. Taigi realizuotas programinis prototipas išpildė suprojektuotą reikalavimų specifikavimo šablonų pagrindu koncepcinį modelį. Išskelti modelio tikslai įgyvendinti sėkmingai. Pagal sudarytą koncepcinį modelį sukurtas programinis prototipas padidina išplečiamumą ir lankstumą reikalavimų specifikavime Volere šablono pagrindu.

Taip pat buvo atlikta informacinių technologijų diegimo centro darbuotojų apklausa, kurios metu penki reikalavimų analizės ekspertai turėjo įvertinti programinį prototipą pagal 5.1 lentelėje pateiktus kriterijus. Remiantis apibendrintais tyrimo rezultatais 5.1 lentelėje pateikiamas ekspertų vertinimas, taip pat išreikštas balais.

Ekspertai beveik vienbalsiu nusprendė, jog programinis prototipas pilnai atitinka Volere reikalavimų specifikavimo šablona, o pastarojo šablono užpildymo proceso darnos, reikalavimų sekamumui naudojamo metodo, paremto efektyvios datos režimu, bei itin intuityvios sąsajos dėka, galima paspartinti reikalavimų specifikavimo procesą. Be to, ataskaitose pateikiamų reikalavimų objektų, jų galiojimo laikotarpio datos bei ataskaitos formato pasirinkimo galimybių pilnai užtenka visapusiškam specifikacijos atvaizdavimui. Programinio prototipo išplečiamumo

galimybės, pasak ekspertų, tėra vidutiniškos, o reikalavimų objektų susietumas ir integracija su kitais įrankiais yra gana silpnos prototipo vietos. Taigi, apibendrinus ekspertų vertinimą, galima teigti, jog naudojant programinį prototipą reikalavimų specifikavimui Volere šablono pagrindu, galima sutaupyti iki 4-5% laiko resursų reikalavimų specifikavimo procese.

## 6. Išvados

1. Atlikta išsami reikalavimų specifikavimo šablonų pagrindu analizė. Jos metu pastebėta, kad reikalavimų specifikavimo procesas vis labiau kompiuterizuojamas, o reikalavimų valdymo įrankių bei šablonų naudojimas gerina reikalavimų specifikacijos ir būsimo produkto kokybę. Analizės rezultatų pagrindu sudarytas reikalavimų specifikavimo šablonų pagrindu programinio prototipo kokybės vertinimo kriterijų sąrašas.

2. Sudarytas reikalavimų specifikavimo sudėties koncepcinis modelis, perteiktas detalia klasių su naudojamomis operacijomis diagrama. Sudaryta klasių diagrama parodo reikalavimų specifikacijos statinę struktūrą ir yra naudojama reikalavimų objektų saugyklos gavime. Pastarasis modelis suteikia dinamiškumo ir išplečiamumo reikalavimų specifikavimo šablonų pagrindu.

3. Sudarytas reikalavimų specifikavimo Volere šablono pagrindu procesų modelis, perteiktas apibendrinta veiklos diagrama. Jo pagalba atvaizduojami reikalavimų specifikacijos užpildymo metu vykdomi procesai. Specifikavimo procesų modelis būtinas teisingam duomenų saugyklos užpildymui bei programinio prototipo pagalba rengiamos reikalavimų specifikacijos užpildymo darnumui.

4. Įvertinus programiniam prototipui iškeltus reikalavimus bei būsimus duomenų srautus, pasirinktas labiau inžinerinės pakraipos prototipo realizacijos būdas, kuris paremtas šiomis technologijomis: PHP programavimo kalba skirta specialių reikalavimų specifikacijos formų kūrimui, o My SQL duomenų bazių valdymo sistema - reikalavimų kaupimui. Pastarosios technologijos užtikrina glaudų tarpusavio bendradarbiavimą, prototipo pasiekiamumą iš nutolusių darbo vietų bei nesudėtingą prototipo išplečiamumą ir palaikymą.

5. Realizuoto programinio prototipo veikimo tyrimui buvo pasirinkta konkreti dalykinė sritis. Remiantis srities problematika, buvo surinkti ir specifikuoti reikalavimai. Panaudojus reikalavimų specifikavimo proceso metodiką, tyrimo metu buvo sėkmingai išbandyti visi programiniame prototipe esantys reikalavimų objektai, ištirtas jų tarpusavio susietumas, o sekamumui naudojamo metodo dėka buvo fiksuojama visa besikeičiančių reikalavimų istorija. Sudaryta dalykinės srities reikalavimų specifikacija gali būti pateikiama IS kūrėjui.

6. Programinis prototipas buvo įvertintas reikalavimų analizės ekspertų komandos reikalavimų specifikuojimo įrankiams svarbių kriterijų atžvilgiu. Vertintojų nuomone, sukurtas programinis prototipas daugiau ar mažiau tenkina visus reikalavimų specifikuojimo vertinimo kriterijus ir gali būti sėkmingai naudojamas reikalavimų specifikuojimo procese. Be to pastebėta, jog programinio prototipo vartotojai laimi keletu aspektų:

- sutaupoma dalis laiko resursų reikalavimų specifikuojimo procese;
- padidinamas išplečiamumas ir lankstumas reikalavimų specifikuojime Volere šablono pagrindu;

## 7. Literatūros sąrašas

- [1] Murphy, T. Mastering the Requirements of Requirements Management. Application Delivery Strategies, Integration & Development Strategies, 2003.
- [2] Butkiene, R. [interaktyvus]. Tyrimo objekto analizė [žiūrėta 2007.12.02]. Prieiga per internetą: <[ftp://isd.ktu.lt/Isd/Butkiene/T000M109/T000M109\\_5.pdf](ftp://isd.ktu.lt/Isd/Butkiene/T000M109/T000M109_5.pdf)>
- [3] Coley Consulting [interaktyvus]. Reducing Your Acceptance Testing Risk [žiūrėta 2008.09.15]. Prieiga per internetą: <<http://www.coleyconsulting.co.uk/require.htm>>
- [4] Firesmith, D.G. Journal of Object Technology. Modern Requirements Specification, 2003, p. 53 – 63.
- [5] Platužienė, J. [interaktyvus]. Reikalavimų inžinerija [žiūrėta 2008.09.10]. Prieiga per internetą: <[kopustas.elen.ktu.lt/studentai/\\_media/reikalavimu\\_inzinerija.ppt](http://kopustas.elen.ktu.lt/studentai/_media/reikalavimu_inzinerija.ppt)>
- [6] Butleris, R.; Danikauskas, T. Veiklos procesų ir informacinių poreikių analizė. Veiklos modeliavimo metodų analizė reikalavimų specififikavimo aspektu, 2002, p. 395 – 400.
- [7] Volere Requirements Resources [interaktyvus]. How to Get Started [žiūrėta 2008.09.17]. Prieiga per internetą: <<http://www.volere.co.uk/gettingstarted.htm>>
- [8] Butleris, R.; Danikauskas, T. Informacijos mokslas, 19 tomas. Reikalavimų specififikavimo ORACLE CASE terpėje plėtra, 2001, p. 51 – 62.
- [9] Requirements Resources [interaktyvus]. Volere Requirements Specification Template [žiūrėta 2008.03.12]. Prieiga per internetą: <<http://www.volere.co.uk/template.htm>>
- [10] Requirements template Chapter 3 [interaktyvus]. Software Specification Template [žiūrėta 2008.09.19]. Prieiga per internetą: <[www4.informatik.tu-muenchen.de/proj/va/SRS.pdf](http://www4.informatik.tu-muenchen.de/proj/va/SRS.pdf)>

- [11] Probasco, L.; Leffingwell, D. Software Requirements Specifications and Use-Case Modeling [interaktyvus]. [žiūrėta 2008.04.02]. Prieiga per internetą: <[www.spc.ca/downloads/srs\\_usecase.doc](http://www.spc.ca/downloads/srs_usecase.doc) >
- [12] Requirements template [interaktyvus]. User Requirement Document [žiūrėta 2008.09.22]. Prieiga per internetą: <[http://hopelive.hope.ac.uk/computing/Level\\_I/NOS/PBLCasesAssess/CW2/URDTemplate.doc](http://hopelive.hope.ac.uk/computing/Level_I/NOS/PBLCasesAssess/CW2/URDTemplate.doc)>
- [13] Monash University [interaktyvus]. CPE4004 - Requirements Specification Template [žiūrėta 2008.09.23]. Prieiga per internetą: <[neptune.netcomp.monash.edu.au/cpe4004/doc/TuteWeek1-3/CPE4004Requirements%20Specification%20template.doc](http://neptune.netcomp.monash.edu.au/cpe4004/doc/TuteWeek1-3/CPE4004Requirements%20Specification%20template.doc)>
- [14] Polgar, J. [interaktyvus]. CPE4004 Lecture2 - Requirements Explained [žiūrėta 2008.09.25]. Prieiga per internetą: <<http://neptune.netcomp.monash.edu.au/cpe4004/doc/Lectures2005/CPE4004-01-2007-%20Requirements%20spec.pdf>>
- [15] IBM Rational Software [interaktyvus]. Rational RequisitePro [žiūrėta 2008.04.03]. Prieiga per internetą: <<http://www-306.ibm.com/software/awdtools/reqpro>>.
- [16] Compuware [interaktyvus]. Optimal Trace [žiūrėta 2008.04.12]. Prieiga per internetą: <<http://www.compuware.com/products/optimaltrace/>>
- [17] Telelogic Requirements - Driven Innovation [interaktyvus]. Telelogic Doors [žiūrėta 2008.04.15]. Prieiga per internetą: <[www.telelogic.com/products/doors/index.cfm](http://www.telelogic.com/products/doors/index.cfm)>.
- [18] Torkzadeh, K. A Comparison between Requirement tools [interaktyvus]. [žiūrėta 2008.04.20]. Prieiga per internetą: <[http://users.encs.concordia.ca/~k\\_torkza/A%20Comparision%20between%20Requirement%20tools.pdf](http://users.encs.concordia.ca/~k_torkza/A%20Comparision%20between%20Requirement%20tools.pdf)>
- [19] Requirements Resources [interaktyvus]. Requirements Tools [žiūrėta 2008.04.20]. Prieiga per internetą: <<http://www.volere.co.uk/tools.htm>>



## **8. Priedai**

### ***Priedas nr. 1.***

#### ***Rekomendacijos ir padėkos***

Nuoširdžiai dėkoju baigiamojo magistrinio darbo vadovui prof. dr. Rimantui Butleriui už pagalbą analizuojant ir projektuojant reikalavimų specifikavimo šablonų pagrindu programinį prototipą.

## **Priedas nr. 2.**

### **Sutrumpinimų žodynėlis**

**DB** – duomenų bazė, naudojama informacijai saugoti.

**MySQL** – viena iš reliacinių duomenų bazių apdorojimo sistemų.

**PHP (PHP Hypertext Preprocessor)** – tai plačiai naudojama atviro kodo bendros paskirties interpretuojama programavimo kalba, skirta internetinių puslapių kūrimui.

**XML (eXtensible Markup Language)** – W3C rekomenduojama bendros paskirties duomenų struktūrų bei jų turinio aprašomoji kalba. Pagrindinė XML kalbos paskirtis yra užtikrinti lengvesnį duomenų keitimąsi tarp skirtingo tipo sistemų, dažniausiai sujungtų internetu. Pagrindinis XML kalbos vienetas yra elementas.

**UML (Unified Modeling Language)** – vieninga modeliavimo kalba, modeliavimo ir specifikacijų kūrimo kalba.

**Priedas nr. 3.*****Paveikslėlių ir lentelių sąrašas***

<b>Eil. nr.</b>	<b>Numeris tekste</b>	<b>Pavadinimas</b>	<b>Psl. nr.</b>
<b>Paveikslėliai</b>			
1.	2.1 pav.	Reikalavimų specifikavimo proceso diagrama	6
2.	2.2 pav.	Vartotojų panaudojimo atvejų modelis	7
3.	2.3 pav.	Reikalavimų inžinerijos etapų pasiskirstymas laike	10
4.	2.4 pav.	Talpykla paremta reikalavimų specifikacija	12
5.	2.5 pav.	Volere kortelė (ang. Volere Snow Card)	14
6.	2.6 pav.	Requisite Pro įrankio pagrindinė reikalavimų forma	22
7.	2.7 pav.	OptimalTrace įrankio pagrindinė forma	23
8.	2.8 pav.	Doors įrankio pagrindinė forma	24
9.	3.1. pav.	Reikalavimų specifikavimo prototipo struktūrinė diagrama	30
10.	3.2 pav.	Reikalavimų specifikavimo klasių diagramos su operacijomis modelis	32
11.	3.3 pav.	Projekto registravimo veiklos modelis	33
12.	3.4 pav.	Vartotojų registravimo veiklos modelis	34
13.	3.5 pav.	Veiklos konteksto sudarymo bei įvykių fiksavimo veiklos modelis	35
14.	3.6 pav.	Terminų žodyno užpildymo bei peržiūros veiklos modelis	35
15.	3.7 pav.	Sistemos sudėties registravimo bei panaudojimo atvejų sąrašo sudarymo veiklos modelis	36
16.	3.8 pav.	Apribojimų reikalavimams nustatymo veiklos modelis	37
17.	3.9 pav.	Funkcinių bei nefunkcinių reikalavimų identifikavimo veiklos modelis	38
18.	3.10 pav.	Svarbių faktų įtraukimo veiklos modelis	39
19.	3.11 pav.	Egzistuojančių sprendimų įvertinimo veiklos modelis	39
20.	3.12 pav.	Rizikos numatymo ir įvertinimo veiklos modelis	40
21.	3.13 pav.	Sistemos uždavinių iškėlimo bei aprašymo veiklos modelis	41
22.	3.14 pav.	Naujų problemų įtraukimo veiklos modelis	41
23.	3.15 pav.	Perspektyvinių reikalavimų nustatymo veiklos modelis	42
24.	3.16 pav.	Projekto objektų sekamumo veiklos modelis	42
25.	3.17 pav.	Projekto dalinės arba pilnos specifikacijos atvaizdavimo veiklos modelis	43
26.	3.18 pav.	Sutartinis žymėjimas	43
27.	3.19 pav.	Reikalavimų specifikavimo Volere šablono pagrindu apibendrintas veiklos modelis	44
28.	4.1 pav.	Reikalavimų specifikavimo prototipo loginės architektūros modelis	46
29.	4.2 pav.	Reikalavimų specifikavimo Volere šablono pagrindu duomenų bazės schema	48
30.	4.3 pav.	Reikalavimų specifikavimo komponentų diagrama	49
31.	4.4 pav.	Reikalavimų specifikavimo diegimo diagrama	50
32.	5.1 pav.	Prisijungimo prie programinio prototipo forma	52
33.	5.2 pav.	Reikalavimų specifikavimo diegimo diagrama	53
34.	5.3 pav.	Vartotojų, kurie naudosis sistema, registravimo forma	54
35.	5.4 pav.	Veiklos konteksto sudarymo bei įvykių fiksavimo langas	55
36.	5.5 pav.	Terminų žodyno užpildymo langas	55

<b>Eil. nr.</b>	<b>Numeris tekste</b>	<b>Pavadinimas</b>	<b>Psl. nr.</b>
37.	5.6 pav.	Sistemos sudėties registravimo bei panaudojimo atvejų sąrašo sudarymo langas	56
38.	5.7 pav.	Apribojimų reikalavimams nustatymo ir pateikimo langas	57
39.	5.8 pav.	Funkcinių reik. pateikimo langas	59
40.	5.9 pav.	Nefuncinių reik. pateikimo langas	59
41.	5.10 pav.	Svarbių faktų įtraukimo langas	60
42.	5.11 pav.	Egzistuojančių sprendimų pateikimo langas	61
43.	5.12 pav.	Rizikos numatymo ir įvertinimo langas	61
44.	5.13 pav.	Sistemos uždavinių iškėlimo bei aprašymo langas	62
45.	5.14 pav.	Naujų problemų pateikimo langas	63
46.	5.15 pav.	Perspektyvinių reikalavimų pateikimo langas	64
47.	5.16 pav.	Projekto objektų sekamumo langas	64
48.	5.17 pav.	Projekto dalinės arba pilnos specifikacijos pateikimo langas	65
49.	5.18 pav.	Projekto specifikacijos atvaizdavimas XML formatu	65
<b>Lentelės</b>			
1.	2.1 lentelė.	Vartotojų tikslai ir problemos	8
2.	2.2 lentelė.	Reikalavimų specifikavimo forma	17
3.	2.3 lentelė.	Reikalavimų specifikavimo forma	19
4.	2.4 lentelė.	Panaudojimo atvejų specifikavimo forma	19
5.	2.5 lentelė.	Šablonų palyginimas	20
6.	2.6 lentelė.	Analogiškų reikalavimų specifikavimo įrankių palyginimas	26
7.	5.1 lentelė.	Sukurto programinio prototipo ir analogiškų įrankių palyginimas	67

## Priedas nr. 4.

### Atliktos apklausos metu užpildytos anketos

#### REIKALAVIMŲ SPECIFIKAVIMO VOLERE ŠABLONO PAGRINDU PROGRAMINIO PROTOTIPO ĮVERTINIMO ANKETA

Anketoje nereikia įrašyti savo pavardės, todėl prašau gerai apgalvoti ir nuoširdžiai atsakyti į anketos klausimus. Užpildę anketą atsiųskite elektroniniu paštu, adresu: [egidijus.zinkevicius@ktu.lt](mailto:egidijus.zinkevicius@ktu.lt)

---

1. Specialybė: Analitikas

2. Jūsų lytis:   X                      Vyras                                            Moteris

3. Įvertinkite programinį prototipą remiantis šiais reikalavimų specifikavimo įrankiu svarbiais kriterijais (1 – silpnai, 5 – puikiai):

- šablonų taikymas – konkrečiu atveju pilnas Volere reikalavimų specifikavimo šablono pritaikymas;
- metodika – reikalavimų specifikacijos Volere šablono pagrindu užpildymo proceso darna;
- susietumas - reikalavimų objektų tarpusavio susietumas;
- sekamumas – reikalavimų objektų registravimo ir atliktų modifikacijų sekimas iki dabartinės būsenos;
- dokumentavimo galimybės – ataskaitų formavimo įvairovė: reikalavimų objektų, jų galiojimo laikotarpio datos bei ataskaitos formato pasirinkimas;
- prieinamumas – galimybė pasiekti įrankį iš įvairių vietų, o ne tik iš tam tikro kompiuterio;
- išplečiamumo galimybės – papildomo funkcionalumo suteikimas, naujų reikalavimų objektų įtraukimas;
- intuityvi sąsaja – langų, formų bei nuorodų (mygtukų) išdėstymas. Analitiko specialisto sėkmingo pasinaudojimo įrankiu be specialaus apmokymo tikybė;
- integracija – įrankio pagalba sukurto produkto panaudojimas kitose sistemose;

Kriterijus	Įvertinimas	1	2	3	4	5
		silpnai	vidutiniškai	gerai	labai gerai	puikiai
šablonų taikymas						X
metodika						X
susietumas			X			
sekamumas					X	
dokumentavimo galimybės						X
prieinamumas						X
išplečiamumo galimybės					X	
intuityvi sąsaja					X	
integracija			X			

*Ačiū už atsakymus!*

## REIKALAVIMŲ SPECIFIKAVIMO VOLERE ŠABLONO PAGRINDU PROGRAMINIO PROTOTIPO ĮVERTINIMO ANKETA

Anketoje nereikia įrašyti savo pavardės, todėl prašau gerai apgalvoti ir nuoširdžiai atsakyti į anketos klausimus. Užpildę anketą atsiųskite elektroniniu paštu, adresu: [egidijus.zinkevicius@ktu.lt](mailto:egidijus.zinkevicius@ktu.lt)

---

1. Specialybė: Analitikas

2. Jūsų lytis:   X                      Vyras                                            Moteris

3. Įvertinkite programinį prototipą remiantis šiais reikalavimų specifیکavimo įrankiu svarbiais kriterijais (1 – silpnai, 5 – puikiai):

- šablonų taikymas – konkrečiu atveju pilnas Volere reikalavimų specifیکavimo šablono pritaikymas;
- metodika – reikalavimų specifیکacijos Volere šablono pagrindu užpildymo proceso darna;
- susietumas - reikalavimų objektų tarpusavio susietumas;
- sekamumas – reikalavimų objektų registravimo ir atliktų modifikacijų sekimas iki dabartinės būsenos;
- dokumentavimo galimybės – ataskaitų formavimo įvairovė: reikalavimų objektų, jų galiojimo laikotarpio datos bei ataskaitos formato pasirinkimas;
- prieinamumas – galimybė pasiekti įrankį iš įvairių vietų, o ne tik iš tam tikro kompiuterio;
- išplečiamumo galimybės – papildomo funkcionalumo suteikimas, naujų reikalavimų objektų įtraukimas;
- intuityvi sąsaja – langų, formų bei nuorodų (mygtukų) išdėstymas. Analitiko specialisto sėkmingo pasinaudojimo įrankiu be specialaus apmokymo tikybė;
- integracija – įrankio pagalba sukurto produkto panaudojimas kitose sistemose;

Kriterijus	Įvertinimas	1 silpnai	2 vidutiniškai	3 gerai	4 labai gerai	5 puikiai
šablonų taikymas						X
metodika				X		
susietumas			X			
sekamumas						X
dokumentavimo galimybės						X
prieinamumas						X
išplečiamumo galimybės			X			
intuityvi sąsaja						X
integracija				X		

*Ačiū už atsakymus!*

## REIKALAVIMŲ SPECIFIKAVIMO VOLERE ŠABLONO PAGRINDU PROGRAMINIO PROTOTIPO ĮVERTINIMO ANKETA

Anketoje nereikia įrašyti savo pavardės, todėl prašau gerai apgalvoti ir nuoširdžiai atsakyti į anketos klausimus. Užpildę anketą atsiųskite elektroniniu paštu, adresu: [egidijus.zinkevicius@ktu.lt](mailto:egidijus.zinkevicius@ktu.lt)

---

1. Specialybė: Vyr. analitikas
2. Jūsų lytis:   X                      Vyras                                            Moteris
3. Įvertinkite programinį prototipą remiantis šiais reikalavimų specifikavimo įrankiu svarbiais kriterijais (1 – silpnai, 5 – puikiai):

- šablonų taikymas – konkrečiu atveju pilnas Volere reikalavimų specifikavimo šablono pritaikymas;
- metodika – reikalavimų specifikacijos Volere šablono pagrindu užpildymo proceso darna;
- susietumas - reikalavimų objektų tarpusavio susietumas;
- sekamumas – reikalavimų objektų registravimo ir atliktų modifikacijų sekimas iki dabartinės būsenos;
- dokumentavimo galimybės – ataskaitų formavimo įvairovė: reikalavimų objektų, jų galiojimo laikotarpio datos bei ataskaitos formato pasirinkimas;
- prieinamumas – galimybė pasiekti įrankį iš įvairių vietų, o ne tik iš tam tikro kompiuterio;
- išplečiamumo galimybės – papildomo funkcionalumo suteikimas, naujų reikalavimų objektų įtraukimas;
- intuityvi sąsaja – langų, formų bei nuorodų (mygtukų) išdėstymas. Analitiko specialisto sėkmingo pasinaudojimo įrankiu be specialaus apmokymo tikybė;
- integracija – įrankio pagalba sukurto produkto panaudojimas kitose sistemose;

Kriterijus	Įvertinimas	1 silpnai	2 vidutiniškai	3 gerai	4 labai gerai	5 puikiai
šablonų taikymas					x	
metodika						x
susietumas				x		
sekamumas						x
dokumentavimo galimybės					x	
prieinamumas						x
išplečiamumo galimybės				x		
intuityvi sąsaja						x
integracija			x			

*Ačiū už atsakymus!*

## REIKALAVIMŲ SPECIFIKAVIMO VOLERE ŠABLONO PAGRINDU PROGRAMINIO PROTOTIPO ĮVERTINIMO ANKETA

Anketoje nereikia įrašyti savo pavardės, todėl prašau gerai apgalvoti ir nuoširdžiai atsakyti į anketos klausimus. Užpildę anketą atsiųskite elektroniniu paštu, adresu: [egidijus.zinkevicius@ktu.lt](mailto:egidijus.zinkevicius@ktu.lt)

---

1. Specialybė: Programuotojas / analitikas
2. Jūsų lytis:   X                      Vyras                                            Moteris
3. Įvertinkite programinį prototipą remiantis šiais reikalavimų specifikavimo įrankiu svarbiais kriterijais (1 – silpnai, 5 – puikiai):

- šablonų taikymas – konkrečiu atveju pilnas Volere reikalavimų specifikavimo šablono pritaikymas;
- metodika – reikalavimų specifikacijos Volere šablono pagrindu užpildymo proceso darna;
- susietumas - reikalavimų objektų tarpusavio susietumas;
- sekamumas – reikalavimų objektų registravimo ir atliktų modifikacijų sekimas iki dabartinės būsenos;
- dokumentavimo galimybės – ataskaitų formavimo įvairovė: reikalavimų objektų, jų galiojimo laikotarpio datos bei ataskaitos formato pasirinkimas;
- prieinamumas – galimybė pasiekti įrankį iš įvairių vietų, o ne tik iš tam tikro kompiuterio;
- išplečiamumo galimybės – papildomo funkcionalumo suteikimas, naujų reikalavimų objektų įtraukimas;
- intuityvi sąsaja – langų, formų bei nuorodų (mygtukų) išdėstymas. Analitiko specialisto sėkmingo pasinaudojimo įrankiu be specialaus apmokymo tikybė;
- integracija – įrankio pagalba sukurto produkto panaudojimas kitose sistemose;

Kriterijus	Įvertinimas	1 silpnai	2 vidutiniškai	3 gerai	4 labai gerai	5 puikiai
šablonų taikymas						X
metodika					X	
susietumas			X			
sekamumas					X	
dokumentavimo galimybės						X
prieinamumas						X
išplečiamumo galimybės					X	
intuityvi sąsaja						X
integracija			X			

*Ačiū už atsakymus!*



## REIKALAVIMŲ SPECIFIKAVIMO VOLERE ŠABLONO PAGRINDU PROGRAMINIO PROTOTIPO ĮVERTINIMO ANKETA

Anketoje nereikia įrašyti savo pavardės, todėl prašau gerai apgalvoti ir nuoširdžiai atsakyti į anketos klausimus. Užpildę anketą atsiųskite elektroniniu paštu, adresu: [egidijus.zinkevicius@ktu.lt](mailto:egidijus.zinkevicius@ktu.lt)

---

1. Specialybė: Programuotojas / analitikas
2. Jūsų lytis:   X                      Vyras                                            Moteris
3. Įvertinkite programinį prototipą remiantis šiais reikalavimų specifikavimo įrankiu svarbiais kriterijais (1 – silpnai, 5 – puikiai):

- šablonų taikymas – konkrečiu atveju pilnas Volere reikalavimų specifikavimo šablono pritaikymas;
- metodika – reikalavimų specifikacijos Volere šablono pagrindu užpildymo proceso darna;
- susietumas - reikalavimų objektų tarpusavio susietumas;
- sekamumas – reikalavimų objektų registravimo ir atliktų modifikacijų sekimas iki dabartinės būsenos;
- dokumentavimo galimybės – ataskaitų formavimo įvairovė: reikalavimų objektų, jų galiojimo laikotarpio datos bei ataskaitos formato pasirinkimas;
- prieinamumas – galimybė pasiekti įrankį iš įvairių vietų, o ne tik iš tam tikro kompiuterio;
- išplečiamumo galimybės – papildomo funkcionalumo suteikimas, naujų reikalavimų objektų įtraukimas;
- intuityvi sąsaja – langų, formų bei nuorodų (mygtukų) išdėstymas. Analitiko specialisto sėkmingo pasinaudojimo įrankiu be specialaus apmokymo tikybė;
- integracija – įrankio pagalba sukurto produkto panaudojimas kitose sistemose;

Kriterijus	Įvertinimas	1 silpnai	2 vidutiniškai	3 gerai	4 labai gerai	5 puikiai
šablonų taikymas						X
metodika					X	
susietumas				X		
sekamumas						X
dokumentavimo galimybės					X	
prieinamumas						X
išplečiamumo galimybės				X		
intuityvi sąsaja						X
integracija			X			

*Ačiū už atsakymus!*