



**KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS**

Diana Kuznecova

**ĮRANKIS PRELIMINARIEMS *UML* MODELIAMS KURTI IŠ
TEKSTO**

Baigiamasis magistro projektas

Vadovas
prof. L. Nemuraitė

KAUNAS, 2015

**KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS**

**ĮRANKIS PRELIMINARIEMS *UML* MODELIAMS KURTI IŠ
TEKSTO**

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

Vadovas

prof. dr. L. Nemuraitė
2015-05-

Recenzentas

doc. dr. V. Šakys
2015-05-

Projektą atliko

IFM-3/4 grupės studentė
Diana Kuznecova
2015-05-



KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS

(Fakultetas)

Diana Kuznecova

(Studento vardas, pavardė)

Informacinių sistemų inžinerijos studijų programa, 621E15001

(Studijų programos pavadinimas, kodas)

Baigiamojo projekto „Įrankis preliminariems *UML* modeliams kurti iš teksto“

AKADEMINIO SAŽINGUMO DEKLARACIJA

20 15 m. gegužės 22 d.
Kaunas

Patvirtinu, kad mano, **Dianos Kuznecovos**, baigiamasis projektas tema „Įrankis preliminariems *UML* modeliams kurti iš teksto“ yra parašytas visiškai savarankiškai ir visi pateikti duomenys ar tyrimų rezultatai yra teisingi ir gauti sąžiningai. Šiame darbe nei viena dalis nėra plagijuota nuo jokių spausdintinių ar internetinių šaltinių, visos kitų šaltinių tiesioginės ir netiesioginės citatos nurodytos literatūros nuorodose. Įstatymų nenumatytų piniginių sumų už šį darbą niekam nesu mokėjęs.

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

(vardą ir pavardę įrašyti ranka)

(parašas)

Kuznecova, D. Tool for Developing Preliminary UML models from Texts. *Final Degree Project of Master of Information Systems Engineering* / Supervisor Prof. Lina Nemuraitė; Kaunas University of Technology, Faculty of Informatics.
Kaunas, 2015. 59 p.

SUMMARY

Software development starts with business analysis. Business clients usually formulate their requirements using natural language. There is a need to automate, as far as possible, creation of preliminary *UML* models.

The aim of this project is to analyze semi-automated and automated methods and tools of developing preliminary *UML* models from natural language. The tool would help to analyze primary requirements, given in the form of natural language and create preliminary *UML* models.

This work consists of several main stages:

- Preparation
- Design
- Realization
- Testing
- Experiment

This paper describes a tool that helps to create preliminary *UML* models from requirements written in natural language. The main advantage of this tool is that it has web interface, and there is no need to install any additional software. Analyst, using his network browser, enters online tool, where he can paste his requirements. After pasting, requirements are being saved to database for further processing. Analyst can now analyze requirements and assign elements, such as Actors and Use cases, to words or phrases. If use case has included or extended use cases, analyst can assign them too. Finally, as a result, analyst can export *XMI* model, which can later be imported to *CASE* tool.

After implementation, an experiment was conducted. The tool was evaluated by precision and recall criteria. The results show that tool for creating preliminary *UML* models from text meets precision and recall criteria by 100% when requirements are written in highly structured text.

Key words: UML, CASE tool, Text to Model, Use CASE diagram, Requirement engineering, Natural Language Processing

TURINYS

Lentelių sąrašas.....	7
Paveikslų sąrašas.....	8
Terminų ir santrumpų žodynas	9
Įvadas	10
1. Probleminės srities analizė.....	12
1.1. Analizės tikslai.....	12
1.2. Tyrimo objektas, sritis ir problema	12
1.3. Tyrimo objekto analizė	12
1.4. Tyrimo objekto naudotojų analizė	13
1.4.1. Vartotojų aibė, tipai ir savybės	13
1.4.2. Vartotojų tikslai ir problemos	13
1.5. Esamų problemos sprendimo metodų analizė (Lietuvos ir tarptautiniu mastu)	13
1.6. Darbo tikslas, uždaviniai ir siekiami privalumai	16
1.7. Siekiamo sprendimo apibrėžimas	17
1.8. Analizės išvados.....	18
2. Įrankio preliminariems <i>UML</i> modeliams kurti iš teksto reikalavimų specifikacija ir projektas, formalus aprašas.....	19
2.1. Reikalavimų specifikacija.....	19
2.2. Dalykinės srities modelis	23
2.3. Naudotojų sąsajos modelis.....	24
2.4. Įrankio preliminariems <i>UML</i> modeliams kurti iš teksto algoritmas	24
2.5. Reikalavimų apibendrinimas	25
3. Įrankio preliminariems <i>UML</i> modeliams kurti iš teksto projektas.....	26
3.1. Sistemos architektūra	26
3.1.1. Reikalavimų analizė.....	26
3.1.2. Loginė visos sistemos architektūra	26
3.1.3. Veiklos logikos (valdymo ir esybių klasių) modelis	27
3.2. Sistemos elgsenos modelis.....	27
3.3. Duomenų bazės schema.....	29
3.4. Detalus projektas.....	30
3.5. Realizacijos modelis	31
3.5.1. Programinių komponentų architektūra	31
3.5.2. Diegimo modelis.....	31
4. Sprendimo realizacija ir testavimas	32
4.1. Sprendimo realizacijos ir veikimo aprašas	32
4.1.1. Realizacijos priemonės	32
4.1.2. Įrankio preliminariems <i>UML</i> modeliams kurti iš teksto veikimas	32
4.2. Testavimo modelis, duomenys, rezultatai.....	35
5. Eksperimentinis įrankio preliminariems <i>UML</i> modeliams kurti iš teksto tyrimas	36
5.1. Eksperimento planas	36
5.1.1. Eksperimentas su griežtos struktūros tekstu	36
5.1.2. Eksperimentas su laisvos struktūros tekstu.....	40
5.1.3. Planuojami gauti PA modeliai	43
5.2. Eksperimento rezultatai	49
5.2.1. Eksperimento su griežtos struktūros reikalavimais rezultatai.....	49
5.2.2. Eksperimento su laisvos struktūros reikalavimais rezultatai	50
5.3. Sprendimo veikimo ir savybių analizė, kokybės kriterijų įvertinimas, taikymo rekomendacijos	50
6. Rezultatų apibendrinimas ir išvados	52
7. Literatūra.....	53
8. Priedai	54

8.1. priedas. Testavimo duomenys.....	54
8.2. priedas. Gauto modelio <i>XMI</i> formatu failo ištrauka.	58

LENTELIŲ SĄRAŠAS

1.1 lentelė. Esamų sprendimų palyginimas.....	16
2.1 lentelė. Panaudojimo atvejo „Įvesti pradinį reikalavimų tekstą“ specifikacija.....	19
2.2 lentelė. Panaudojimo atvejo „Taisyti pradinį tekstą“ specifikacija.....	20
2.3 lentelė. Panaudojimo atvejo „Pašalinti tekstą“ specifikacija	20
2.4 lentelė. Panaudojimo atvejo „Žymėti žodžius/žodžių junginius“ specifikacija.....	20
2.5 lentelė. Panaudojimo atvejo „Pašalinti žodžio žymą“ specifikacija	21
2.6 lentelė. Panaudojimo atvejo „Priskirti žodžiams elementus“ specifikacija	21
2.7 lentelė. Panaudojimo atvejo „Keisti priskirtus elementus“ specifikacija	22
2.8 lentelė. Panaudojimo atvejo „Išgauti reikalavimus iš teksto“ specifikacija.....	22
2.9 lentelė. Panaudojimo atvejo „Eksportuoti rezultatų modelį XMI formatu“ specifikacija	22
5.1 lentelė. Pirmojo bandymo su griežtos struktūros tekstu suvestinė.....	36
5.2 lentelė. Antrojo bandymo su griežtos struktūros tekstu suvestinė	37
5.3 lentelė. Trečiojo bandymo su griežtos struktūros tekstu suvestinė.....	37
5.4 lentelė. Ketvirtojo bandymo su griežtos struktūros tekstu suvestinė	38
5.5 lentelė. Penktojo bandymo su griežtos struktūros tekstu suvestinė	38
5.6 lentelė. Šeštojo bandymo su griežtos struktūros tekstu suvestinė.....	38
5.7 lentelė. Septintojo bandymo su griežtos struktūros tekstu suvestinė	39
5.8 lentelė. Aštuntojo bandymo su griežtos struktūros tekstu suvestinė.....	39
5.9 lentelė. Devintojo bandymo su griežtos struktūros tekstu suvestinė	39
5.10 lentelė. Dešimtojo bandymo su griežtos struktūros tekstu suvestinė.....	40
5.11 lentelė. Pirmojo bandymo su laisvos struktūros tekstu suvestinė	40
5.12 lentelė. Antrojo bandymo su laisvos struktūros tekstu suvestinė.....	40
5.13 lentelė. Trečiojo bandymo su laisvos struktūros tekstu suvestinė	41
5.14 lentelė. Ketvirtojo bandymo su laisvos struktūros tekstu suvestinė.....	41
5.15 lentelė. Penktojo bandymo su laisvos struktūros tekstu suvestinė.....	41
5.16 lentelė. Šeštojo bandymo su laisvos struktūros tekstu suvestinė	42
5.17 lentelė. Septintojo bandymo su laisvos struktūros tekstu suvestinė.....	42
5.18 lentelė. Aštuntojo bandymo su laisvos struktūros tekstu suvestinė	43
5.19 lentelė. Devintojo bandymo su laisvos struktūros tekstu suvestinė.....	43
5.20 lentelė. Dešimtojo bandymo su laisvos struktūros tekstu suvestinė	43
5.21 lentelė. Eksperimento su griežtos struktūros reikalavimais vertinimas tikslumo bei išsamumo kriterijais	49
5.23 lentelė. Eksperimento su laisvos struktūros reikalavimais vertinimas tikslumo ir išsamumo kriterijais	50

PAVEIKSLŲ SĄRAŠAS

1.1 pav. Projekto tikslas	12
1.2 pav. <i>UMLG</i> sistemos veikimo principas	13
1.3 pav. <i>KPSpotter</i> sistemos architektūra	14
1.4 pav. žodžių junginių priskyrimas atributams	15
1.5 pav. žodžių reikšmių tipų bei aprašymų lentelė	15
1.6 pav. panaudojimo atvejų diagramos elementai	16
1.7 pav. Veiklos kontekstinė diagrama	17
1.8 pav. Veiklos panaudojimo atvejų diagrama	17
1.9 pav. Veiklos sąveikų modelis.....	18
1.10 pav. Siekiamo sprendimo veiklos proceso modelis	18
2.1 pav. Detalizuotas panaudojimo atvejų modelis.....	19
2.2 pav. Esybių klasių modelis.....	23
2.3 pav. Planuotos vartotojo sąsajos prototipas	24
2.4 pav. Įrankio algoritmas	25
3.1 pav. Analizės modelis	26
3.2 pav. Loginė sistemos architektūra.....	27
3.3 pav. Projektavimo klasių modelis	27
3.4 pav. Įrankio būsenų modelis	28
3.5 pav. <i>UML SQL</i> diagrama	29
3.6 pav. Prisijungimo panaudojimo atvejo realizacija	30
3.7 pav. Prisijungimo panaudojimo atvejo sekų diagrama	30
3.8 pav. Elementų priskyrimo panaudojimo atvejo realizacija	30
3.9 pav. Elementų priskyrimo panaudojimo atvejo sekų diagrama	31
3.10 pav. Komponentų diagrama	31
3.11 pav. Sistemos diegimo modelis.....	31
4.1 pav. Naujo vartotojo registracija	32
4.2 pav. Meniu juosta.....	33
4.3 pav. Teksto įvedimo laukas.....	33
4.4 pav. Elementų priskyrimo langas.....	34
4.5 pav. Elemento priskyrimas aktoriui	34
4.6 pav. „ <i>include</i> “ ar „ <i>extend</i> “ priskyrimas.....	35
5.1 pav. Pirmojo bandymo planuojama panaudojimo atvejų diagrama	44
5.2 pav. Antrojo bandymo planuojama panaudojimo atvejų diagrama	45
5.3 pav. Trečiojo bandymo planuojama panaudojimo atvejų diagrama	45
5.4 pav. Ketvirtojo bandymo planuojama panaudojimo atvejų diagrama	46
5.5 pav. Penktojo bandymo planuojama panaudojimo atvejų diagrama.....	46
5.6 pav. Šeštojo bandymo planuojama panaudojimo atvejų diagrama	47
5.7 pav. Septintojo bandymo planuojama panaudojimo atvejų diagrama	47
5.8 pav. Aštuntojo bandymo planuojama panaudojimo atvejų diagrama	48
5.9 pav. Devintojo bandymo planuojama panaudojimo atvejų diagrama.....	48
5.10 pav. Dešimtojo bandymo planuojama panaudojimo atvejų diagrama	49

TERMINŲ IR SANTRUMPŲ ŽODYNAS

SQL – (angl. *Structured Query Language*) struktūrizuota užklausų kalba;

CASE – (angl. *Computer-Aided Software Engineering*) automatizuotas kompiuterinis programinės įrangos projektavimas;

UML – (angl. *Unified Modeling Language*) vieninga modeliavimo kalba;

HTML – (angl. *Hyper text Markup Language*) hiperteksto žymėjimo kalba;

XML – (angl. *Extensible Markup Language*) bendros paskirties duomenų struktūrų bei jų turinio aprašomoji kalba;

HTTP – (angl. *HyperText Transfer Protocol*) pagrindinis metodas pasiekti informaciją pasauliniame tinkle;

ODBC – (angl. *Open Database Connectivity*) standartizuota taikomosios programinės įrangos programavimo sąsaja prisijungimui prie duomenų bazių;

XMI – (angl. *XML Metadata Interchange*) OMG standartas metaduomenų keitimui XML pagalba;

OMG – (angl. *Object Management Group*) tarptautinis technologinių standartų konsorciūmas;

NLP – (angl. *Natural Language Processing*) natūralios kalbos apdorojimas.

ĮVADAS

Darbo problematika ir aktualumas

Pasauliui vis labiau pereinant prie kompiuterizuotų verslo valdymo sistemų iškyla poreikis automatizuoti kuo įmanoma daugiau sistemų projektavimo etapų. Pirmasis žingsnis programinės įrangos gamyboje yra verslo analizė. Užsakovai savo vizijas formuoja paprasto teksto pavidalu. Rankiniu būdu perkelti reikalavimus iš teksto į veiklos modelius trunka daug laiko. Taip pat, analitikai gali nepastebėti trūkumų sudėtinguose natūralia kalba suformuluotuose reikalavimuose, kas įtakoja dviprasmybes bei sunkumus, atkuriant konkrečius reikalavimus. Tokios problemos kyla, kai analitikas neturi pakankamai tos srities žinių.[11,12]

Pagrindinis programinės įrangos sėkmės matas yra lygmuo, kuriame programinė įranga atitinka jai iškeltą tikslą. Reikalavimų inžinerija – tai procesas, kuris nustato būtent šį tikslą, identifikodamas būsimos programinės įrangos naudotojus bei jų poreikius, visa tai dokumentuodamas.[9]

Nors D. Kumar [11] teigia, jog pilnai automatizuoti reikalavimų, pateiktų natūralia kalba, analizę yra neįmanoma, tačiau reikia automatizuoti kaip įmanoma daugiau analizės etapų. Jo aprašomas UMGAR įrankis analizuoja reikalavimus naudodamas natūralios kalbos apdorojimo (NLP) įrankius.[12]

Šiame darbe analizuojamos galimybės pagreitinti informacijos perkėlimo į modelį procesą, sukuriant įrankį, kuris padėtų atlikti pirminį reikalavimų inžinerijos etapą - reikalavimų teksto analizę ir sukurtų *CASE* įrankiui pradinį veiklos modelį iš duoto teksto.

Darbo tikslas ir uždaviniai

Pagrindinis darbo tikslas yra sukurti įrankį, kuris palengvintų pirminį sistemų projektavimo etapą. Įrankis padės kurti preliminarius modelius, analizuojant turimus reikalavimus, suformuluotus paprasta natūralia kalba.

Darbo uždaviniai:

1. Išanalizuoti esamus modelių kūrimo iš teksto algoritmus, jų privalumus bei trūkumus;
2. Įvertinti galimybes, suprojektuoti ir realizuoti naują tikslinį preliminarių modelių generavimo įrankį;
3. Sukurtą įrankį ištestuoti;
4. Atlikti eksperimentą, įvertinsiantį sukurto įrankio veiksmingumą.

Darbo rezultatai ir jų svarba

Šio darbo pagrindinis rezultatas – įrankis, padedantis kurti preliminarius *UML* modelius iš natūralia kalba pateiktų reikalavimų bei generuojantis modelių *XMI* formatu failus.

Įrankio veikimas patikrintas automatiniais testais, o nauda įvertinta atlikus du eksperimentus. Eksperimentus vykdė studentas, panaudodamas kiekvienam eksperimentui 10 specialiai sukurtų reikalavimus aprašančių tekstų pavyzdžių. Pirmojo eksperimento metu studentas dirbo su griežtos struktūros tekstais, o antrojo eksperimento metu buvo naudojami laisvos struktūros tekstai aprašantys tuos pačius reikalavimus.

Eksperimento vykdymas – kiekvienam bandymui pateikiama aktorių ir panaudojimo atvejų lentelė ir numatomas gauti rezultatas – rankiniu būdu nubraižyta panaudojimo atvejų diagrama. Reikalavimai sukeliama į sukurtą įrankį, juose sužymimi aktoriai bei panaudojimo atvejai ir transformuojami į panaudojimo atvejų modelį.

Eksperimento rezultatai parodė, jog sudarant modelį iš griežtos struktūros reikalavimų, tikslumo ir išsamumo kriterijai lygūs vienetui. Sudarant modelį iš laisvos struktūros reikalavimų, tikslumo ir išsamumo kriterijų maksimumo pasiekti nepavyko.

Atlikus eksperimentų rezultatų analizę, nustatyta, jog įrankis preliminariems modeliams kurti iš teksto yra naudingas ir sukuriama modeliai yra tikslūs bei išsamūs dirbant su griežtos struktūros tekstais. Įrankį galima taikyti realiame programinės įrangos kūrimo procese. Ateityje įrankį tobulinant (pridedant naujas funkcijas) galimas didelis pasisiekimas rinkoje.

Darbo struktūra

Dokumentą sudaro aštuoni pagrindiniai skyriai su poskyriais:

1. Probleminės srities analizė – šiame skyriuje pateikiama su darbo problematika susijusios informacijos analizė;
2. Įrankio preliminariems modeliams kurti iš teksto reikalavimų specifikacija ir projektas, formalus aprašas – šiame skyriuje aprašomi visi įrankiui taikomi reikalavimai;
3. Įrankio preliminariems modeliams kurti iš teksto projektas – skyriuje pateikiamas įrankio projektas;
4. Įrankio realizacija ir testavimas – skyriuje aprašomas įrankio veikimas, testavimo modelis bei testavimo rezultatai;
5. Eksperimentinis tyrimas – šiame skyriuje aptariamas bei aprašomas įvykdytas eksperimentas, aprašomi eksperimento rezultatai bei rekomendacijos tolimesniam įrankio taikymui;
6. Rezultatų skyriuje aprašomas sukurtas įrankis ir pateikiamos darbo išvados;
7. Literatūros skyriuje išvardijama visa naudota literatūra;
8. Priedai – šiame skyriuje pateikiami visi netiesiogiai su darbu susiję bet reikšmingi duomenys.

1. PROBLEMINĖS SRITIES ANALIZĖ

Šiame skyriuje pateikiama informacija susijusi su darbo problematika. Analizuojamos į darbe aprašomą įrankį panašios sistemos, programos, įrankiai bei algoritmai, jų veikimas, pagrindinės savybės. Taip pat informacija, susijusi su galimais skirtingais įrankio įgyvendinimo būdais, programinė įranga, kuri bus naudojama kuriant sistemą.

1.1. Analizės tikslai

- Identifikuoti esminius veiklos modelių kūrimo metodus ir jų problemas;
- Susipažinti su esamais teksto analizės metodais bei įrankiais;
- Susipažinti su galimomis veiklos modelių kūrimo priemonių realizavimo technologijomis.

1.2. Tyrimo objektas, sritis ir problema

Tyrimo objektas:

Teksto analizės ir preliminarus reikalavimų objektinio modelio sudarymo procesas.

Tyrimo sritis:

Tekstų analizė ir veiklos objektinių modelių kūrimas iš teksto.

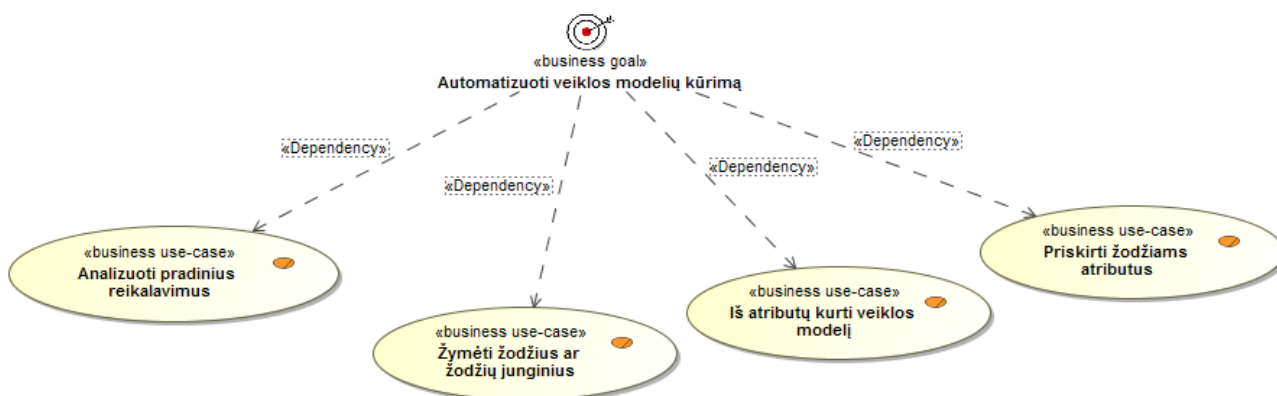
Tyrimo problema:

Siekiant sukurti informacinę sistemą, pirmiausia sukuriama jos vizijos aprašymas, kuriuo remiantis kuriamas sistemos projektas. Tačiau nėra sklandaus perėjimo nuo vizijos prie pradinių veiklos modelių, iš kurių toliau sudaromas sistemos projektas. Šiame tyrime siekiama užpildyti šią spragą, sukuriant teksto analizės metodiką ir priemones, kurios leistų CASE įrankiui sukurti kompiuterizuojamas veiklos modelius.

1.3. Tyrimo objekto analizė

Teksto analizės ir veiklos objektinio modelio sudarymo procesas aprašomas projekto tikslų diagrama (1.1 pav.). Pagrindinis tikslas yra automatizuoti veiklos modelių kūrimą, kuris priklauso nuo:

- Pradinių reikalavimų analizės;
- Žodžių ar žodžių junginių žymėjimo reikalavimuose;
- Elementų žodžiams priskyrimo;
- Veiklos modelių kūrimo iš pažymėtų elementų.



1.1 pav. Projekto tikslas

1.4. Tyrimo objekto naudotojų analizė

1.4.1. Vartotojų aibė, tipai ir savybės

Preliminarių *UML* modelių kūrimo iš teksto įrankio vartotojai – tai visi analitikai bei projektuotojai, dirbantys su *MagicDraw* modeliavimo įrankiu. Taip pat ir besimokantys modeliuoti šiuo įrankiu.

1.4.2. Vartotojų tikslai ir problemos

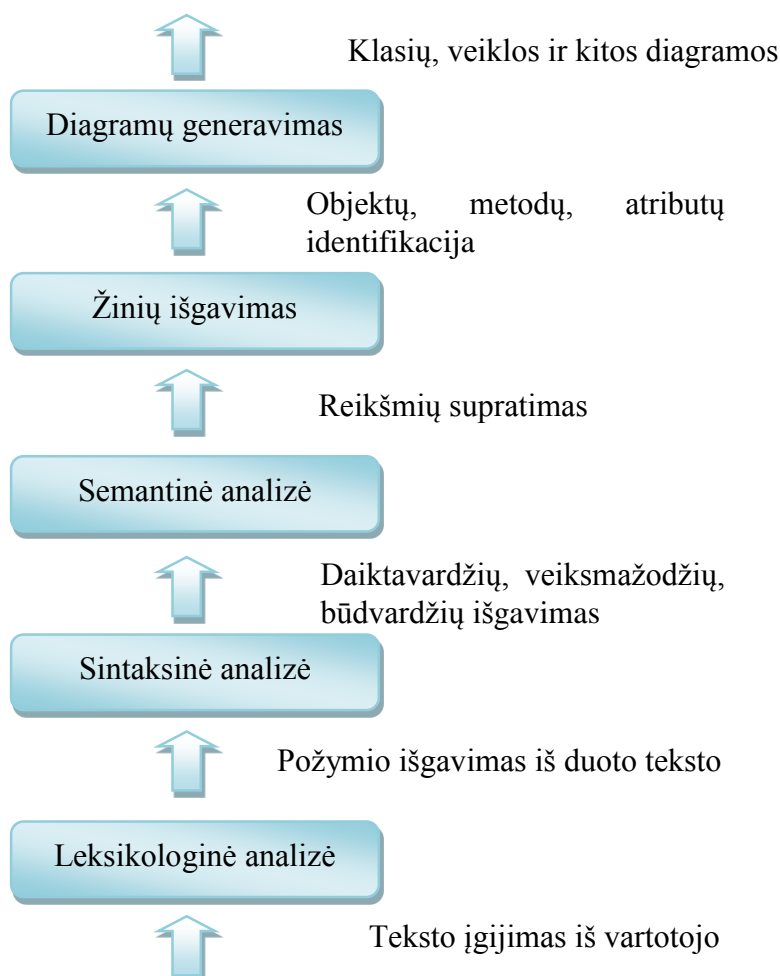
Projektuotojų ir analitikų pagrindinis tikslas – sutaupyti kuo daugiau laiko projektuojant programinę įrangą, bei sumažinti galimas klaidas. Pagrindinis sėkmingo programinės įrangos proceso veiksnys – tinkamas natūralia kalba suformuluotų reikalavimų įvertinimas. [7]

Problema – rankinis reikalavimų iš teksto perkėlimas į modelį ir auganti klaidų atsiradimo tikimybė.

1.5. Esamų problemos sprendimo metodų analizė (Lietuvos ir tarptautiniu mastu)

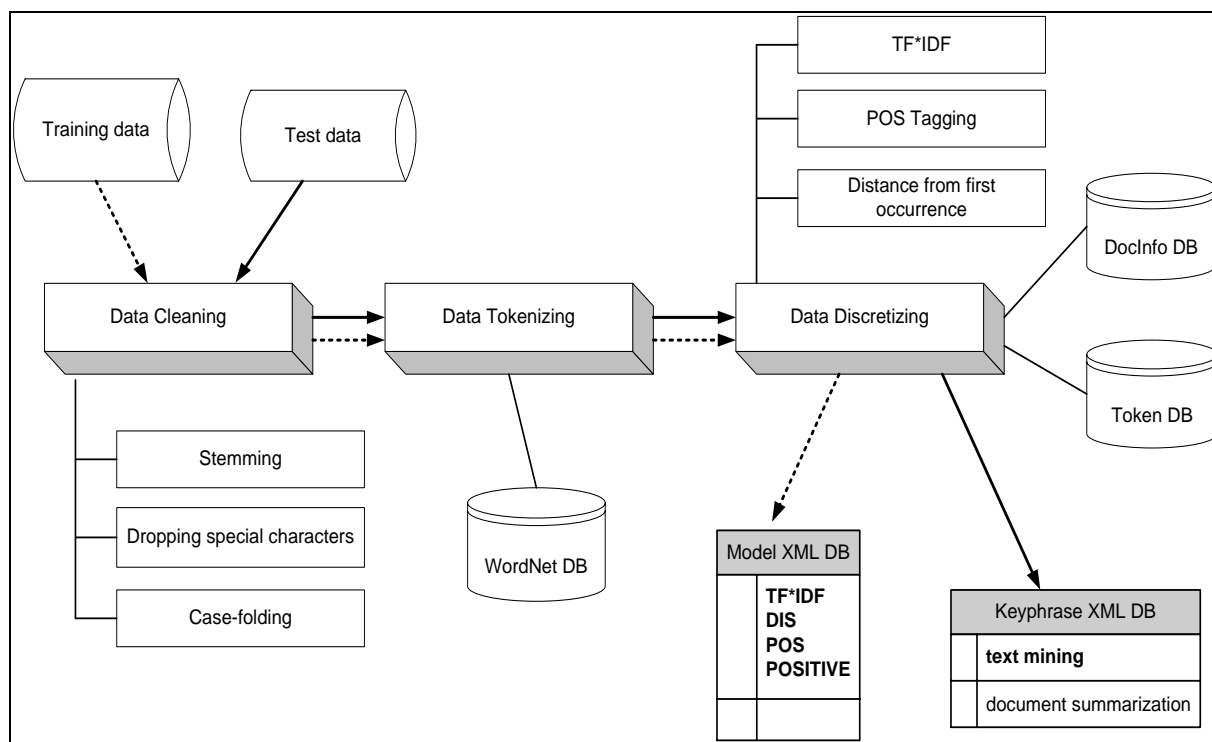
UMLG sistema [2] analizuoja galimybę braižyti *UML* diagramas, perskaičius pradinį struktūrizuotą vizijos dokumentą. Braižymas vyksta penkiais moduliais (1.2 pav.):

1. Teksto įgijimas iš vartotojo: modulis skaito įvestą tekstą ir generuoja žodžius arba žodyną jungdamas įvedamus ženklus;
2. Sintaksinė analizė: remdamasis nustatytomis kategorizavimo taisyklėmis, modulis kategorizuoja žodžius į įvairias klases, tokias kaip veiksmažodis, daiktavardis ar būdvardis;
3. Teksto supratimas: teksto žodžiams yra suteikiama reikšmė pagal semantines taisykles;
4. Žinių ištraukimas: modulis ištraukia objektus ir klases bei priskiria jiems atributus;
5. *UML* diagramų generavimas: naudodamas *UML* simbolius braižo įvairias *UML* diagramas.



1.2 pav. *UMLG* sistemos veikimo principas [2]

KPSpotter sistema [3] – lanksti raktažodžių ištraukimo sistema (1.3 pav.), palaikanti daugumą failų formatų, tokių kaip *HTML*, *XML*.



1.3 pav. *KPSpotter* sistemos architektūra

Šaltinis: <http://onlinelibrary.wiley.com/doi/10.1002/meet.1450410143/full>

KPSpotter sistemoje yra dvi fazės: [10]

- Ištraukiamas modelis;
- Ištraukiamos raktinės frazės.

Viena iš panašaus principo sistemų yra *Visual Paradigm for UML* [1].

Šis įrankis atlieka teksto analizę sekančiu principu:

- Sukuriamas naujas projektas;
- Pasirenkama, kokia analizė ar diagramos bus kuriamos;
- Įkeliamas pradinis reikalavimų dokumentas;
- Dokumento tekste pažymimi žodžiai ar žodžių junginiai ir jiems yra priskiriami atributai (1.4 pav.);
- Sudaroma lentelė, kurioje matomi visi pažymėti žodžiai, jiems priskirti tipai bei galimi jų aprašai (1.5 pav.);
- Žodžiai pagal kuriuos yra kuriami elementai pažymimi skirtingomis spalvomis;
- Pasirinkus braižyti naują panaudojimo atvejų diagramą, atvaizduojami visi elementai, belieka jiems priskirti ryšius. (1.6 pav.).

OTV (Online Television) is a company which delivers both paid and free online television broadcasting services to all TV fans. Members can watch both live and archived TV programs on OTV's website, anytime and anywhere.

There are two kinds of memberships - general and premium. It is free of charge for visitors to register as a **general member** to **watch any archived TV programs**. On the other hand, they can register as a **premium member** to watch **live TV programs**. **general member** can upgrade himself to **premium member** and **premium member** is not allowed to change his/her account permanently by **remove his/her account permanently** by **posting their opinion**. **premium member** can receive **monthly newsletter**, which lists the recommended **live TV programs**.

Besides watching TV programs, **premium member** can **post their opinion** on **live TV programs** with each other, by posting their opinion on **live TV programs**. **premium member** who raises the most active discussion on **live TV programs** can receive **monthly newsletter**, which lists the recommended **live TV programs**.

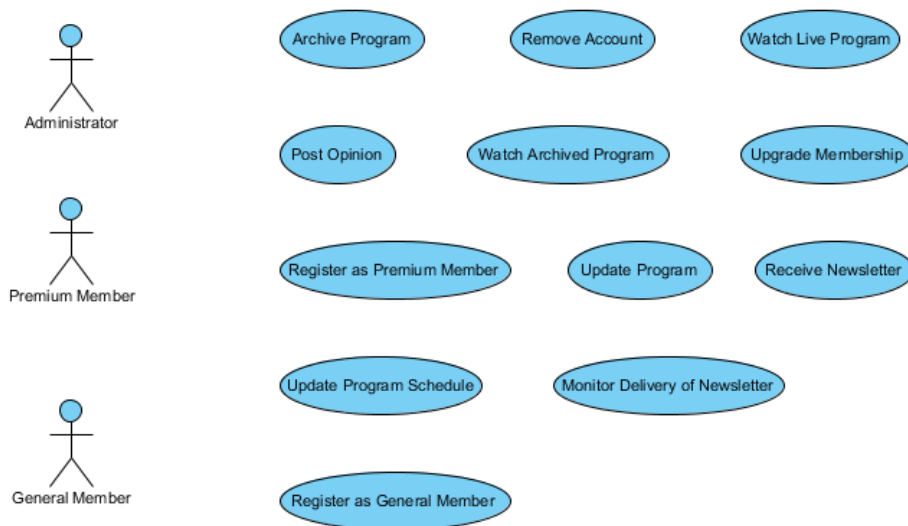
1.4 pav. žodžių junginių priskyrimas atributams

Šaltinis: <http://www.visual-paradigm.com/product/vpUML/tutorials/textualanalysis.jsp>

No.	Candidate Class	Extracted Text	Type	Description	Occurrence	Highli...
1	premium member	premium member	Actor		1	Yellow
2	general member	general member	Actor		2	Orange
3	administrators	administrators	Actor		1	Yellow
4	watch any archived	watch any archived	Use Case		1	Green
5	register as a premi	register as a premium	Use Case		1	Green
6	register as a gener	register as a general	Use Case		1	Green
7	upgrade himself to	upgrade himself to a	Use Case		1	Blue
8	remove his/her acc	remove his/her accou	Use Case		1	Purple
9	posting their opinio	posting their opinion	Use Case		1	Pink
10	receive monthly ne	receive monthly new	Use Case		1	Yellow
11	update the program	update the program	Use Case		1	Yellow
12	update the program	update the program	Use Case		1	Yellow
13	archive programs	archive programs	Use Case		1	Yellow
14	monitor the deliver	monitor the delivery	Use Case		1	Yellow

1.5 pav. žodžių reikšmių tipų bei aprašymų lentelė.

Šaltinis: <http://www.visual-paradigm.com/product/vpUML/tutorials/textualanalysis.jsp>



1.6 pav. panaudojimo atvejų diagramos elementai

Šaltinis: <http://www.visual-paradigm.com/product/vpUML/tutorials/textualanalisys.jsp>

Visual Paradigm for UML trūkumai:

- Žodžiai ar jų junginiai vis tiek yra pažymimi rankiniu būdu;
- Braižant panaudojimo atvejų diagramą ryšius tarp elementų reikia nubrėžti rankiniu būdu;
- Veikia kaip atskira programa.

1.1 lentelė. Esamų sprendimų palyginimas

Sąvybė	<i>UMLG</i>	<i>KPSpotter</i>	<i>Visual Paradigm for UML</i>
Naudoja kalbos apdorojimo technologijas	+	-	-
Veiklos diagramų susiejimas su tekstu	-	-	+
Raktažodžių išskyrimas	-	+	+
Automatinis	+	+	-
Taiko kalbos analizę	+	-	-

1.6. Darbo tikslas, uždaviniai ir siekiami privalumai

Darbo tikslas – padidinti atitikimą tarp kuriamos sistemos vizijos ir veiklos modelių, sukuriant kompiuterizuotos tekstų analizės ir veiklos modelių kūrimo metodiką bei priemones pasirinktam *CASE* įrankiui.

Uždaviniai:

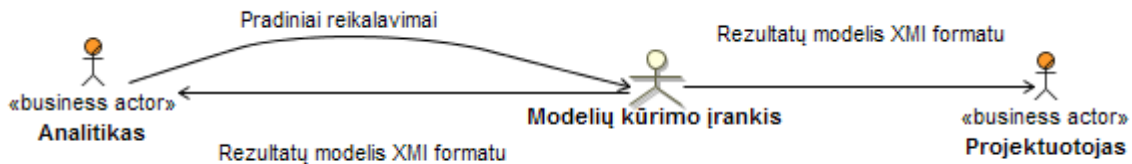
1. Išanalizuoti:

- 1.1. Veiklos modeliavimo metodus;
- 1.2. Esamus tekstų analizės metodus ir algoritmus;
- 1.3. Esamus tekstų analizės įrankius;
- 1.4. Galimas veiklos modelių kūrimo priemonių realizavimo technologijas.

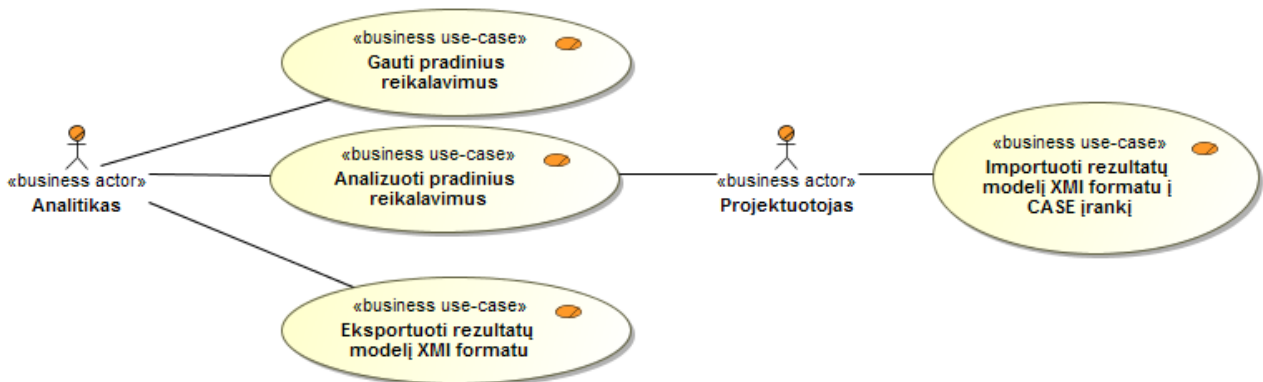
2. Sukurti kompiuterizuotos tekstų analizės ir veiklos modelių kūrimo iš teksto metodiką;
3. Suprojektuoti įrankio priemones, kurios leistų kurti veiklos modelius iš teksto;
4. Realizuoti ir ištestuoti sukurtą programinį prototipą;
5. Atlikti eksperimentą, kuris leistų įvertinti sukurtos metodikos ir priemonių funkcionalumą.

1.7. Siekiamo sprendimo apibrėžimas

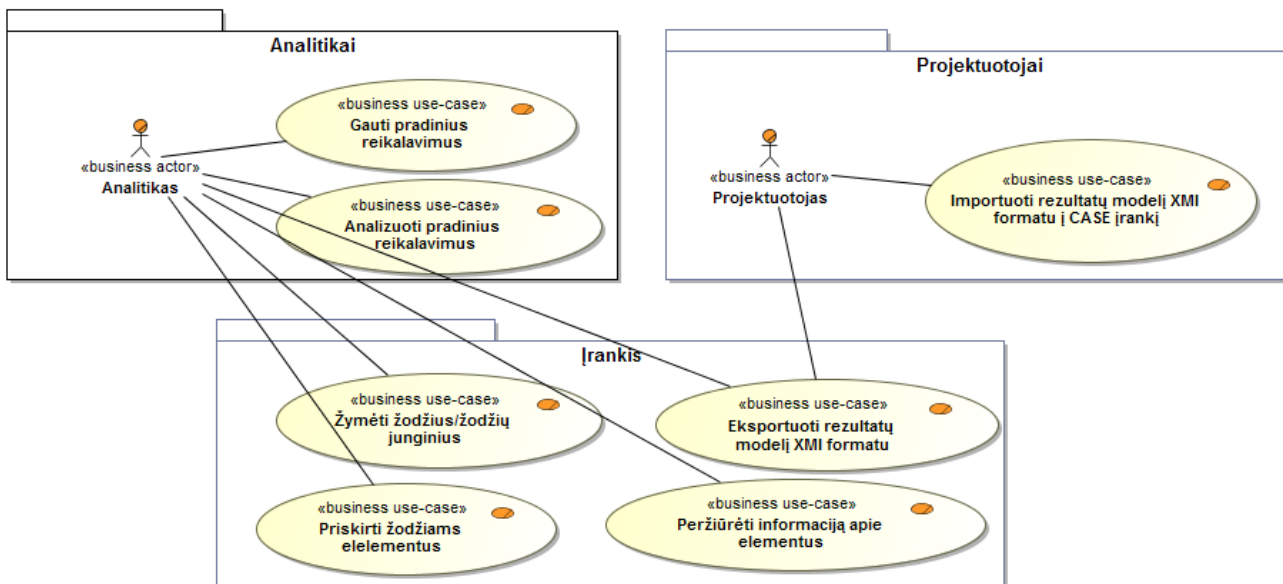
Pradedant kurti naują programinę įrangą, organizacijoje dirbantys analitikai analizuoja užsakovų suformuluotus pradinio reikalavimus. Dalinai automatizuojamas preliminarių *UML* modelių sudarymas iš natūralia kalba suformuluotų reikalavimų. Dalinis automatizavimas palengvina pirminį projektavimo etapą. Analitikai turi galimybę į įrankį įvesti pradinio reikalavimus, atlikti jame teksto analizę (žymėti žodžius bei žodžių junginius, priskirti jiems elementus), bei gauti rezultatų modelio *XMI* formatu failą, kurį projektuotojas gali importuoti į *CASE* įrankį tolimesniai projektavimo etapui įgyvendinti (1.7 – 1.9 pav.). 1.10 paveiksle pateikiamas siekiamo sprendimo veiklos proceso modelis, kuriame pavaizduotas preliminarių *UML* modelių kūrimo iš teksto procesas. Jame dalyvauja du aktoriai – projektuotojas ir analitikas. (1.10 pav.)



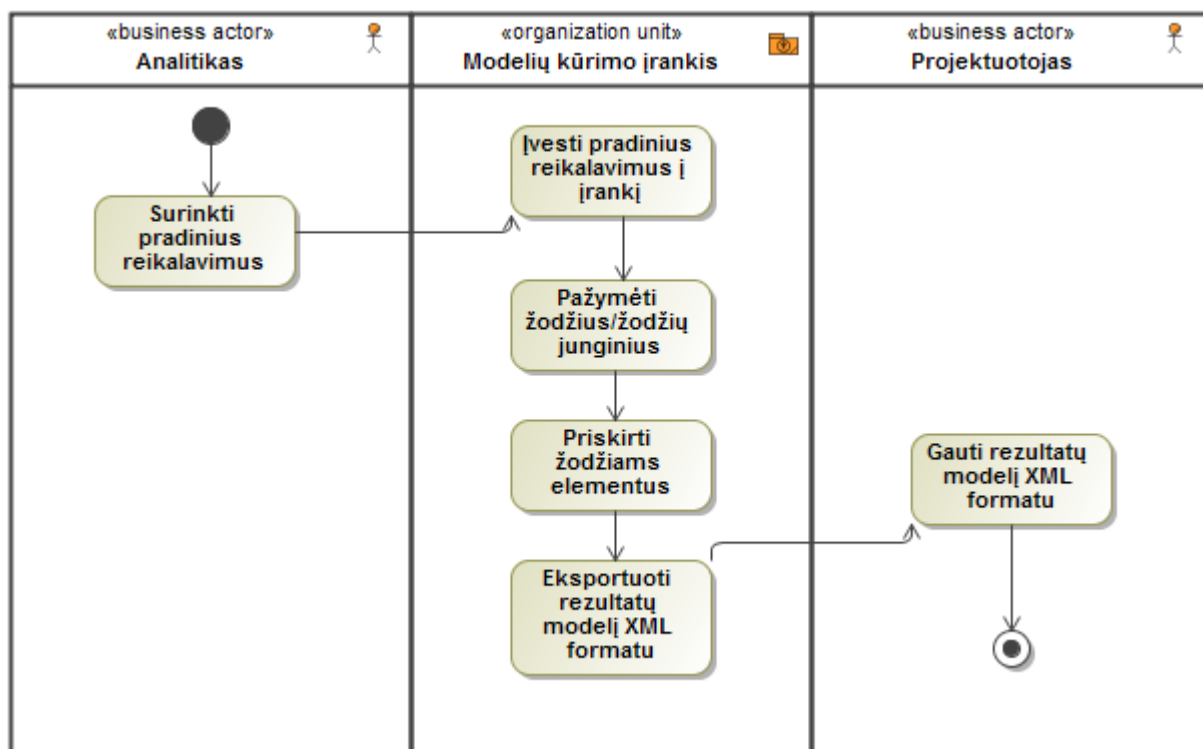
1.7 pav. Veiklos kontekstinė diagrama



1.8 pav. Veiklos panaudojimo atvejų diagrama



1.9 pav. Veiklos sąveikų modelis



1.10 pav. Siekiamo sprendimo veiklos proceso modelis

1.8. Analizės išvados

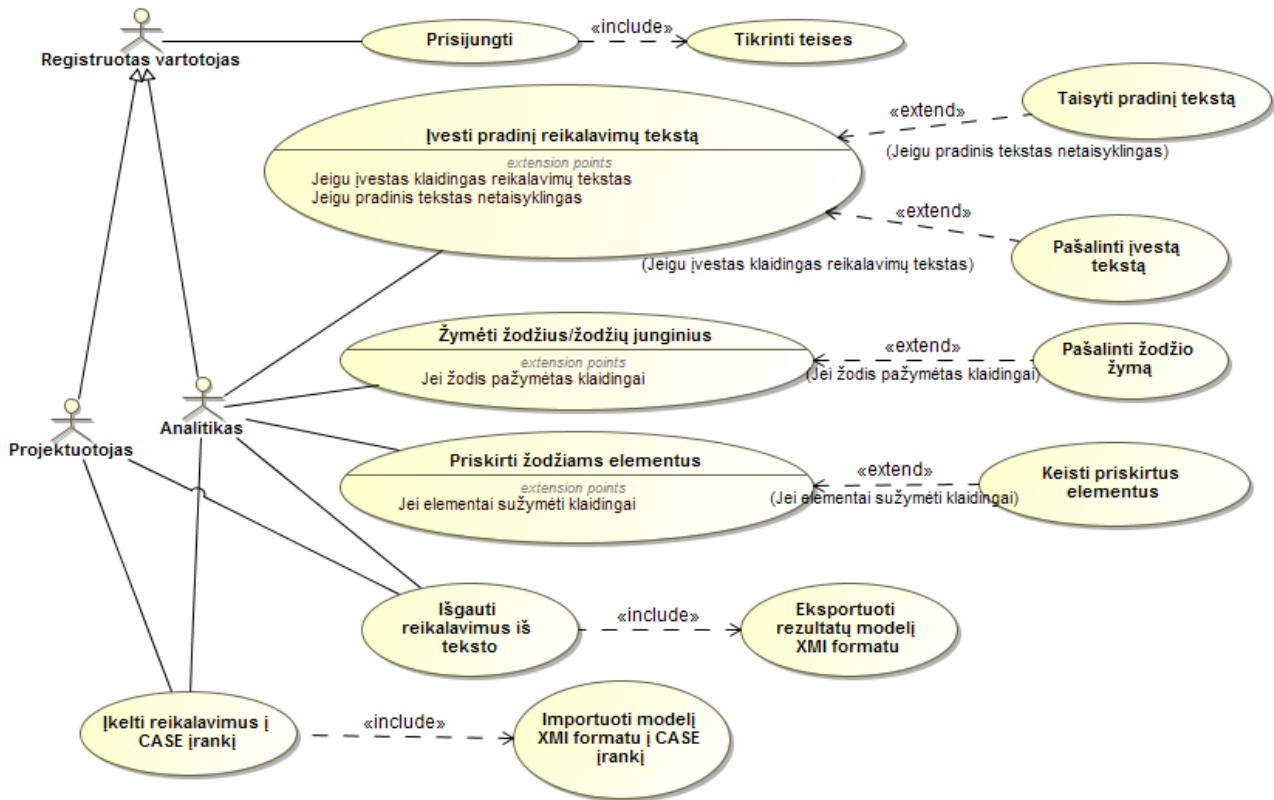
Atlikus analizę buvo nustatyta:

1. Problemos svarba bei galimas sprendimo būdas. Itin svarbu iš užsakovo gautų reikalavimų, teisingai suformuoti pradinus veiklos modelius. Reikalavimų perkėlimo į modelį etape padarius klaidų, jos gali įtakoti viso projekto žlugimą;
2. Tikslinės vartotojų grupės poreikiai. Pagrindiniai vartotojai – tai įmonėse dirbantys analitikai bei projektuotojai;
3. Panašaus principo sistemų, sprendimų veikimo trūkumai bei privalumai. Pagrindinis įrankių trūkumas yra tas, kad pilnai automatizuoto modelių kūrimo neturi nei vienas įrankis;

2. ĮRANKIO PRELIMINARIEMS UML MODELIMS KURTI IŠ TEKSTO REIKALAVIMŲ SPECIFIKACIJA IR PROJEKTAS, FORMALUS APRAŠAS

2.1. Reikalavimų specifikacija

Įrankio preliminariniams UML modeliams kurti iš teksto pagrindinės funkcijos vaizduojamos panaudojimo atvejų modelyje (2.1 pav.). Modelis susideda iš dviejų vartotojų grupių ir 13 (trylikos) panaudojimo atvejų. Lentelėse pateikiamos pagrindinių panaudojimo atvejų specifikacijos (2.1 – 2.9 lentelės).



2.1 pav. Detalizuotas panaudojimo atvejų modelis

2.1 lentelė. Panaudojimo atvejo „Įvesti pradinį reikalavimų tekstą“ specifikacija

PA „Įvesti pradinį reikalavimų tekstą“	
Tikslas. Įvesti naujus, analizei skirtus, pradinis reikalavimus.	
Aprašymas. Šis panaudojimo atvejis skirtas naujų reikalavimų įvedimui, taip pat jis apibendrintai vaizduoja teksto taisymo ir šalinimo PA.	
Prieš sąlyga	Vartotojas prisijungęs
Aktorius	Analitikas
Sužadinimo sąlyga	Vartotojas nori įvesti naują analizei skirtą tekstą
Susiję panaudojimo atvejai	Išplečia PA
	Apima PA
	Specializuoja PA
Pagrindinis įvykių srautas	Įrankio reakcija ir sprendimai
Vartotojas nori įvesti naują tekstą	Įrankis išsaugo tekstą duomenų bazėje ir atvaizduoja jį tolimesniam apdorojimui
Po sąlyga	Duomenų bazėje išsaugotas naujas reikalavimų tekstas
Alternatyvūs scenarijai	
Pastabos	

2.2 lentelė. Panaudojimo atvejo „Taisyti pradinį tekstą“ specifikacija

PA „Taisyti pradinį tekstą“		
Tikslas. Pataisyti klaidingai įvestą reikalavimų tekstą.		
Aprašymas. Šis panaudojimo atvejis skirtas įvesto teksto redagavimui.		
Prieš sąlyga	Vartotojas prisijungęs. Įvestas pradinis reikalavimų tekstas.	
Aktorius	Analitikas	
Sužadinimo sąlyga	Vartotojas nori redaguoti analizei skirtą tekstą	
Susiję panaudojimo atvejai	Išplečia PA	„Įvesti pradinį reikalavimų tekstą“
	Apima PA	
	Specializuoja PA	
Pagrindinis įvykių srautas	Įrankio reakcija ir sprendimai	
Vartotojas nori redaguoti naują tekstą	Įrankis išsaugo redaguotą tekstą duomenų bazėje ir atvaizduoja jį tolimesniam apdorojimui	
Po sąlyga	Duomenų bazėje išsaugotas redaguotas reikalavimų tekstas	
Alternatyvūs scenarijai		
Pastabos		

2.3 lentelė. Panaudojimo atvejo „Pašalinti tekstą“ specifikacija

PA „Pašalinti tekstą“		
Tikslas. Pašalinti nereikalingus reikalavimus.		
Aprašymas. Šis panaudojimo atvejis skirtas reikalavimų šalinimui.		
Prieš sąlyga	Vartotojas prisijungęs Įvestas pradinis reikalavimų tekstas	
Aktorius	Analitikas	
Sužadinimo sąlyga	Vartotojas nori pašalinti analizei skirtą tekstą	
Susiję panaudojimo atvejai	Išplečia PA	„Įvesti pradinį reikalavimų tekstą“
	Apima PA	
	Specializuoja PA	
Pagrindinis įvykių srautas	Įrankio reakcija ir sprendimai	
1. Vartotojas nori pašalinti tekstą	Įrankis klausia, ar vartotojas tikras	
2. Vartotojas patvirtina	Įrankis pašalina įrašą duomenų bazėje	
Po sąlyga	Duomenų bazėje ištrintas reikalavimų tekstas	
Alternatyvūs scenarijai		
2a. Vartotojas atšaukia šalinimą	Įrankis grįžta į pradinę būseną	
Pastabos		

2.4 lentelė. Panaudojimo atvejo „Žymėti žodžius/žodžių junginius“ specifikacija

PA „Žymėti žodžius/žodžių junginius“	
Tikslas. Pažymėti reikalavimuose esančius žodžius ar jų junginius.	
Aprašymas. Šis panaudojimo atvejis skirtas žodžių žymėjimui, taip pat jis apibendrintai vaizduoja žodžio žymos šalinimo PA.	
Prieš sąlyga	Vartotojas prisijungęs Įvestas pradinis reikalavimų tekstas
Aktorius	Analitikas
Sužadinimo sąlyga	Vartotojas nori pradėti teksto analizę
	Išplečia PA

Susiję panaudojimo atvejai	Apima PA	
	Specializuoja PA	
Pagrindinis įvykių srautas		Įrankio reakcija ir sprendimai
1. Vartotojas nori pažymėti žodį ar žodžių junginį		Įrankis pažymi tekstą jį nuspalvindamas
Po sąlyga		Išlieka teksto žymės
Alternatyvūs scenarijai		
1. Vartotojas pažymi klaidingą žodį, ir nori pašalinti žymą		Įrankis vykdo žodžio žymos pašalinimo PA
Pastabos		

2.5 lentelė. Panaudojimo atvejo „Pašalinti žodžio žymą“ specifikacija

PA „Pašalinti žodžio žymą“		
Tikslas. Pašalinti klaidingą žodžio žymą.		
Aprašymas. Šis panaudojimo atvejis skirtas klaidingai pažymėtų žymų šalinimui		
Prieš sąlyga		Vartotojas prisijungęs Įvestas pradinių reikalavimų tekstas Pažymėtas žodis
Aktorius		Analitikas
Sužadinimo sąlyga		Vartotojas nori pašalinti klaidingai pažymėtą žymą
Susiję panaudojimo atvejai	Išplečia PA	„Žymėti žodžius/žodžių junginius“
	Apima PA	
	Specializuoja PA	
Pagrindinis įvykių srautas		Įrankio reakcija ir sprendimai
1. Vartotojas nori pašalinti žodžio žymą		Įrankis pasiteirauja ar vartotojas tikras
2. Vartotojas patvirtina		Įrankis pašalina blogą žymą
Po sąlyga		Blogos žymos pašalintos
Alternatyvūs scenarijai		
1. Vartotojas nusprendžia nešalinti žymos		Įrankis grįžta į pradinę būseną
Pastabos		

2.6 lentelė. Panaudojimo atvejo „Priskirti žodžiams elementus“ specifikacija

PA „Priskirti žodžiams elementus“		
Tikslas. Priskirti pažymėtiems žodžiams elementus		
Aprašymas. Šis panaudojimo atvejis skirtas elementų žodžiams priskyrimui, taip pat jis apibendrintai vaizduoja priskirto elemento keitimo PA.		
Prieš sąlyga		Vartotojas prisijungęs Įvestas pradinių reikalavimų tekstas Pažymėtas žodis
Aktorius		Analitikas
Sužadinimo sąlyga		Vartotojas nori priskirti žodžiui elementą
Susiję panaudojimo atvejai	Išplečia PA	
	Apima PA	
	Specializuoja PA	
Pagrindinis įvykių srautas		Įrankio reakcija ir sprendimai
1. Vartotojas nori priskirti žodžiui elementą		Įrankis pažymėtam tekstui priskiria elementą palikdamas įrašą duomenų bazėje

Po sąlyga	Duomenų bazėje žodžiui ar žodžių junginiui priskiriamas elementas
Alternatyvūs scenarijai	
Pastabos	

2.7 lentelė. Panaudojimo atvejo „Keisti priskirtus elementus“ specifikacija

PA „Keisti priskirtus elementus“	
Tikslas. Pakeisti klaidingai priskirtą elementą.	
Aprašymas. Šis panaudojimo atvejis skirtas klaidingai priskirtų elementų taisymui	
Prieš sąlyga	Vartotojas prisijungęs Įvestas pradinių reikalavimų tekstas Pažymėtas žodis Žodžiui priskirtas elementas
Aktorius	Analitikas
Sužadinimo sąlyga	Vartotojas nori pakeisti priskirtą elementą
Susiję panaudojimo atvejai	Išplečia PA
	Apima PA
	Specializuoja PA
Pagrindinis įvykių srautas	Įrankio reakcija ir sprendimai
1. Vartotojas nori pakeisti priskirtą elementą	Įrankis pasiteirauja ar vartotojas tikras
2. Vartotojas patvirtina	Įrankis pašalina priskirtą elementą
Po sąlyga	Įrankis pakeičia priskirtą elementą
Alternatyvūs scenarijai	
1. Vartotojas nusprendžia nekeisti elemento	Įrankis grįžta į pradinę būseną
Pastabos	

2.8 lentelė. Panaudojimo atvejo „Išgauti reikalavimus iš teksto“ specifikacija

PA „Išgauti reikalavimus iš teksto“	
Tikslas. Išgauti reikalavimus iš teksto	
Aprašymas. Apibendrintai pavaizduoti reikalavimų išgavimą iš teksto.	
Prieš sąlyga	Vartotojas prisijungęs Įvestas pradinių reikalavimų tekstas Pažymėti žodžiai ir žodžių junginiai Žodžiams ar jų junginiams priskirti elementai
Aktorius	Analitikas
Sužadinimo sąlyga	Vartotojas nori išgauti reikalavimus
Susiję panaudojimo atvejai	Išplečia PA
	Apima PA
	Specializuoja PA
Pagrindinis įvykių srautas	Įrankio reakcija ir sprendimai
1. Vartotojas nori išgauti reikalavimus iš teksto	Įrankis įvykdo PA „Eksportuoti rezultatų modelį XMI formatu“
Po sąlyga	Vartotojas turi rezultatų failą.
Alternatyvūs scenarijai	
Pastabos	

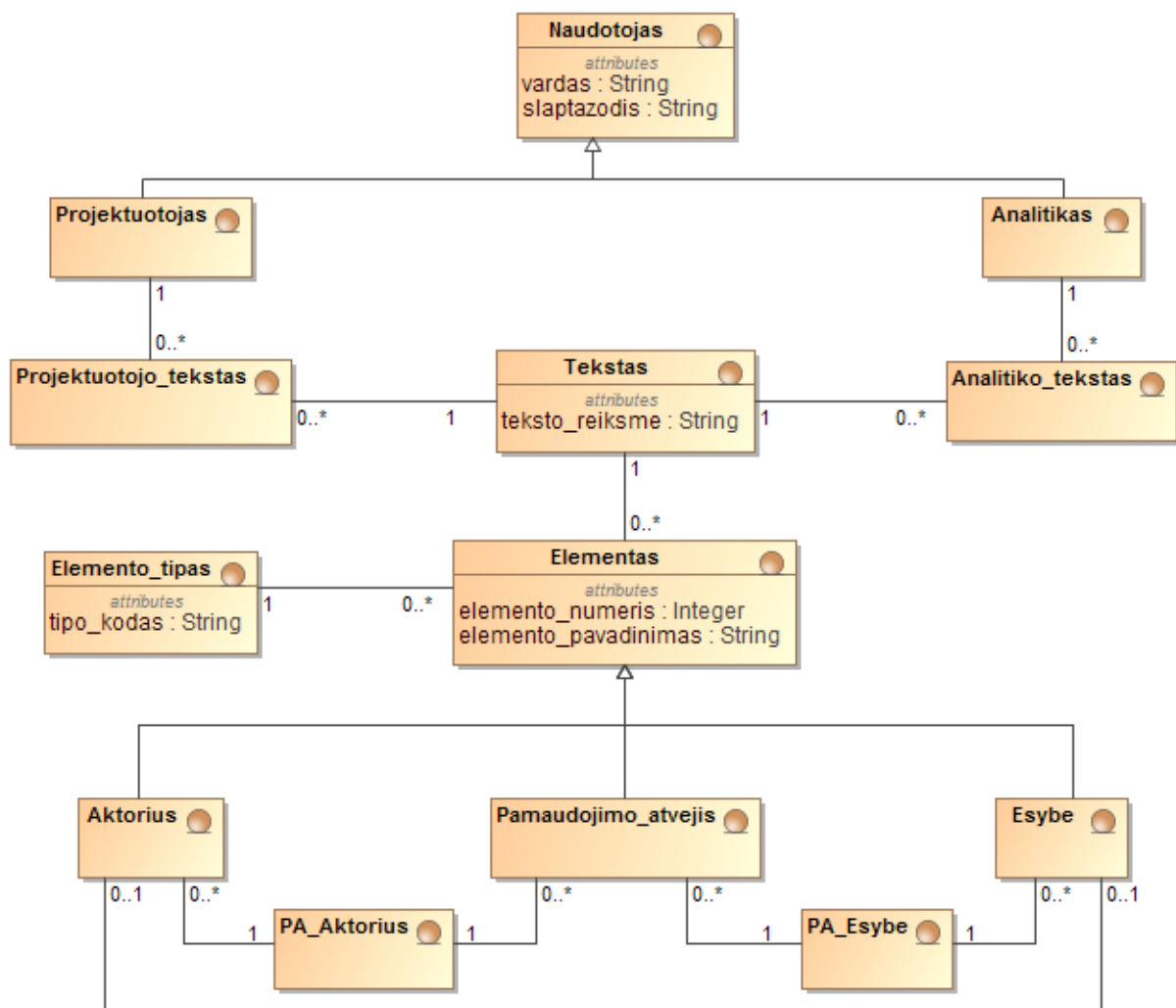
2.9 lentelė. Panaudojimo atvejo „Eksportuoti rezultatų modelį XMI formatu“ specifikacija

PA „Eksportuoti rezultatų modelį XMI formatu“	
Tikslas. Išgauti rezultatų failą	

Aprašymas. Šis PA skirtas rezultatų failo eksportavimui	
Prieš sąlyga	Vartotojas prisijungęs Įvestas pradinis reikalavimų tekstas Pažymėti žodžiai ir žodžių junginiai Žodžiams ar jų junginiams priskirti elementai
Aktorius	Analitikas
Sužadinimo sąlyga	Vartotojas nori gauti rezultatų failą
Susiję panaudojimo atvejai	Išplečia PA
	Apima PA
	Specializuoja PA
Pagrindinis įvykių srautas	Įrankio reakcija ir sprendimai
1. Vartotojas nori gauti rezultatų failą	Įrankis eksportuoja rezultatų modelio XMI formatu failą
Po sąlyga	Vartotojas turi rezultatų failą.
Alternatyvūs scenarijai	
Pastabos	

2.2. Dalykinės srities modelis

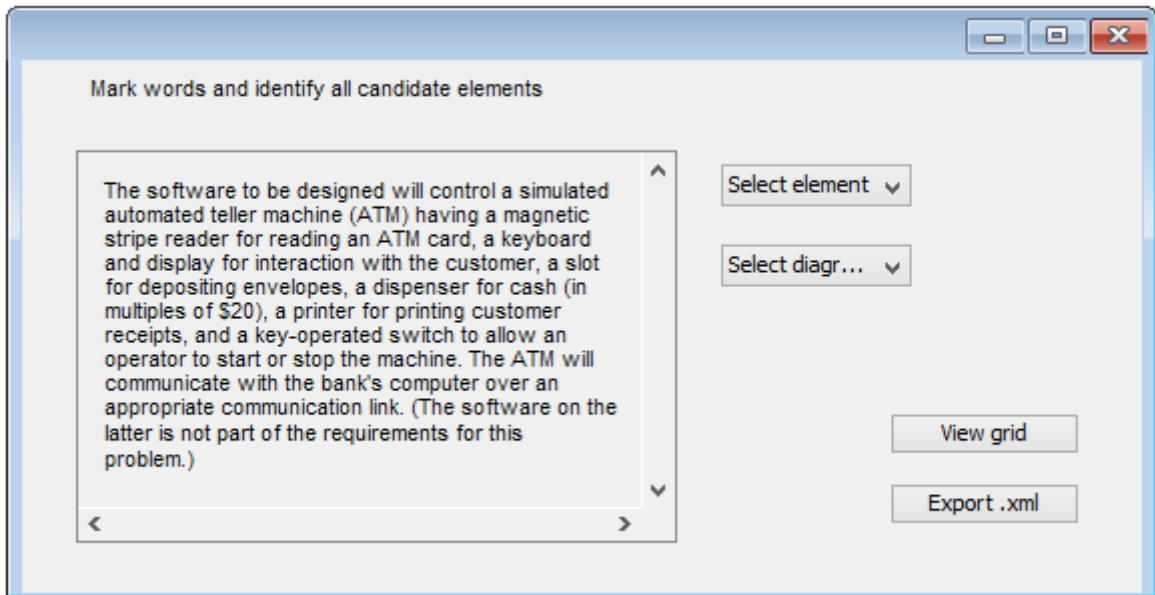
Įrankio preliminariniams *UML* modeliams kurti iš teksto esybių klasių diagrama atvaizduoja pagrindines įrankio esybes, klases bei ryšius, kurių pagrindu vėliau bus kuriama duomenų bazė (2.2 pav.)



2.2 pav. Esybių klasių modelis

2.3. Naudotojų sąsajos modelis

Žemiau pavaizduotas planuojamos vartotojo sąsajos prototipas (2.3 pav.). Pagrindiniai elementai – teksto įvedimo langas, į kurį įkeliami pirminiai natūralia kalba suformuluoti reikalavimai. Šiame lange taip pat vyksta žodžių ir žodžių junginių žymėjimas. Mygtukų pagalba priskiriami elementai bei eksportuojami rezultatai.



2.3 pav. Planuotos vartotojo sąsajos prototipas

2.4. Įrankio preliminariems *UML* modeliams kurti iš teksto algoritmas

Įrankio preliminariems *UML* modeliams kurti iš teksto esmė yra dalinai automatizuoti, palengvinti ir pagreitinti modelių sudarymo procesą. Taip pat, šio proceso dalinis automatizavimas sumažina galimų klaidų tikimybę.

Programinės įrangos gamyba prasideda nuo verslo analizės, tačiau verslo užsakovai savo poreikius suformuoja paprasto teksto pavidalu. Tokie veiksmai skatina artėjimą prie automatizuotos tekstinės analizės, kuri palengvintų programinių įrangų projektavimo procesą, sugeneruodama pradinį modelį iš duoto teksto.

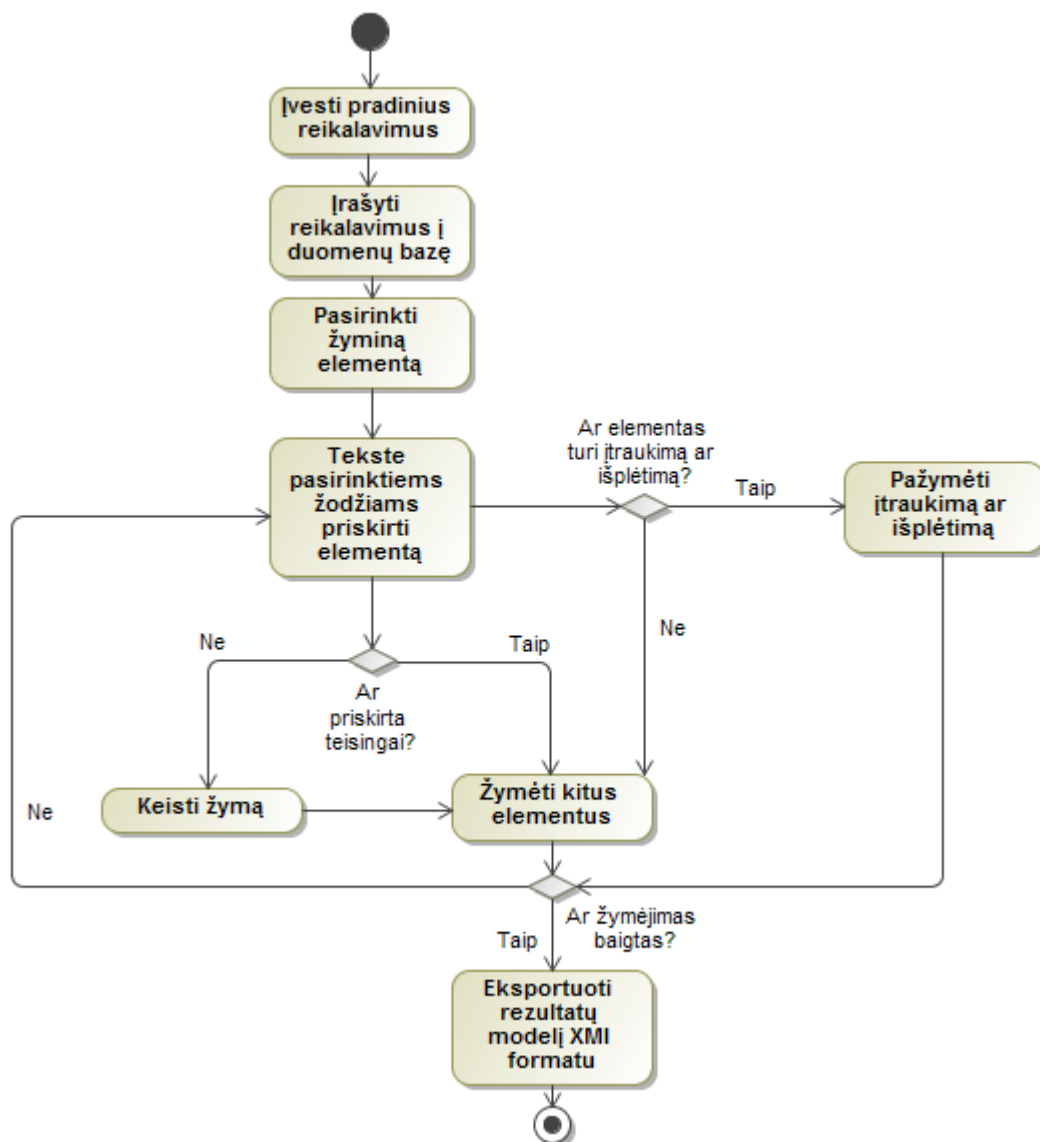
Tokiu būdu būtų pagreitinamas pirmasis etapas - informacijos perkėlimas į modelį ir klaidų tikimybės sumažinimas.

Tikslas yra suprojektuoti ir realizuoti įrankį pradiniam projektavimo etapui automatizuoti t. y. modelio sukūrimui iš tekstinės informacijos. Įrankis turėtų leisti vartotojui įkopi juoti tekstą į dialogą; vartotojas turėtų galėti pasirinkti iš kokių žodžių kokie elementai bus sukurti; žodžiai pagal kuriuos buvo sukurti elementais turėtų būti pažymėti tekste.

Taip pat yra aktualu matyti visų žodžių bei jiems priskirtų elementų suvestinę, taip pat skaičiuoti žodžių pasikartojimo dažnumą. To pasekoje galima būtų daryti išvadas, kokie žodžiai pradinuose reikalavimuose pasitaiko dažniausiai, ir kokie elementai jiems dažniausiai yra priskiriami.

Kuriamo įrankio naujumas pasireiškia savo tikslingumu bei novatoriška vartotojo sąsaja. Įrankis realizuotas tinkle, todėl jo nereikia siųsti ir diegti į lokalų kompiuterį. Įrankis kuriamas konkrečiai įmonei, bus susiejamas su konkrečiu produktu. Įrankio eksportuojamas rezultatų failas yra specifinės struktūros, pritaikytos būtent tam įrankiui ir bus importuojamas į *CASE* įrankį, kuriame toliau vyks projektavimo etapas.

Žemiau pateikiamas įrankio veikimo algoritmas.(2.4 pav.)



2.4 pav. Įrankio algoritmas

2.5. Reikalavimų apibendrinimas

Apibendrinant visus įrankiui keliamus reikalavimus, galima glaustai teigti, kad įrankis turėtų leisti vartotojui įkopi juoti tekstą į dialogą; vartotojas turėtų galėti pasirinkti iš kokių žodžių kokie elementai bus sukurti; žodžiai pagal kuriuos buvo sukurti elementais turėtų būti pažymėti tekste.

Galutinis rezultatas – įrankyje eksportuojamas modelio *XMI* formatu failas, kuris vėliau importuojamas į *MagicDraw* įrankį ir taip gaunamas pradinis panaudojimo atvejų modelis sudarytas iš teksto.

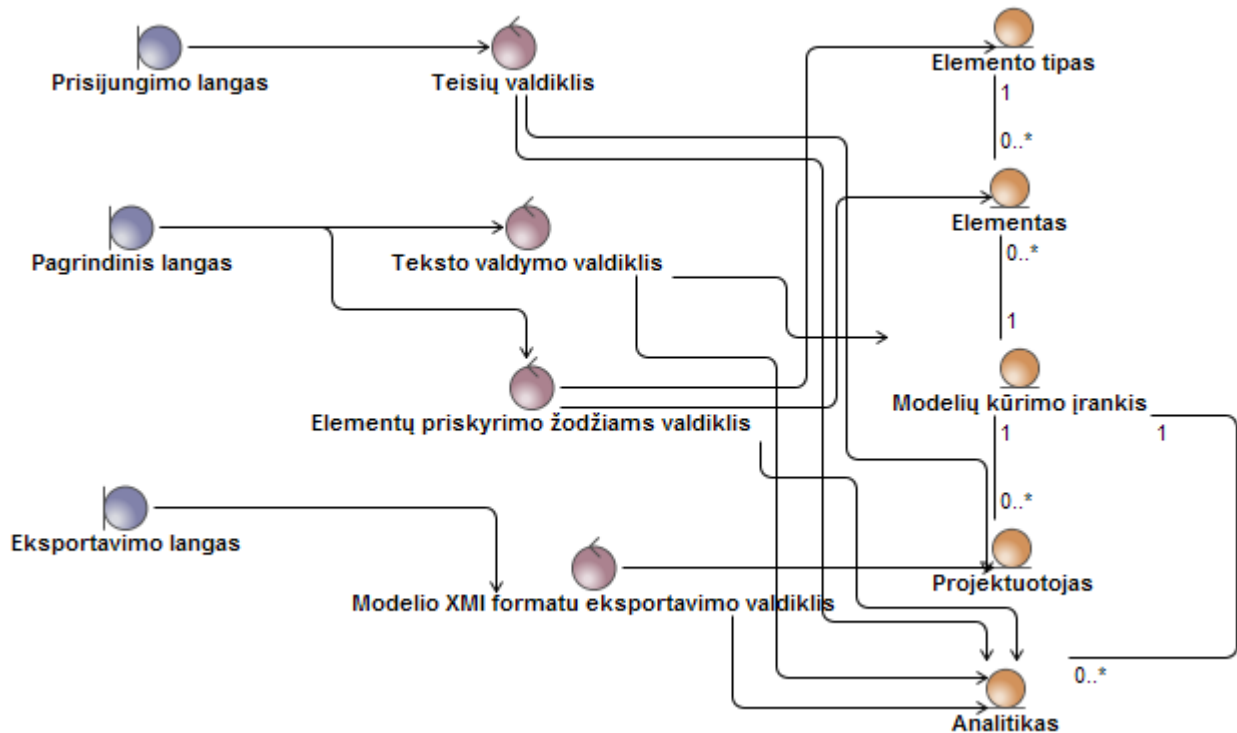
3. ĮRANKIO PRELIMINARIEMS *UML* MODELiams KURTI IŠ TEKSTO PROJEKTAS

3.1. Sistemos architektūra

3.1.1. Reikalavimų analizė

Analizės modeliu (3.1 pav.) parodoma įrankio preliminariems *UML* modeliams kurti iš teksto panaudojimo atvejų realizacija.

Analizės modelyje pavaizduoti valdikliai, komunikuojantys su duomenų baze, tai teisių valdymo, teksto valdymo, elementų priskyrimo žodžiams bei modelio *XMI* formatu eksportavimo valdikliai.

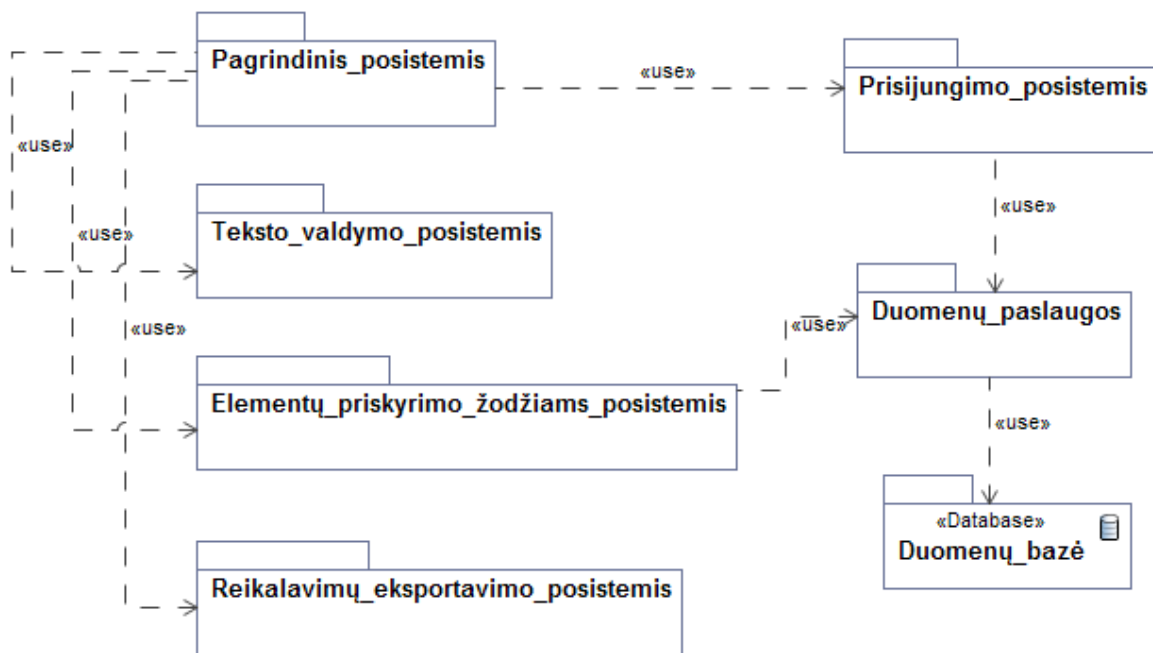


3.1 pav. Analizės modelis

3.1.2. Loginė visos sistemos architektūra

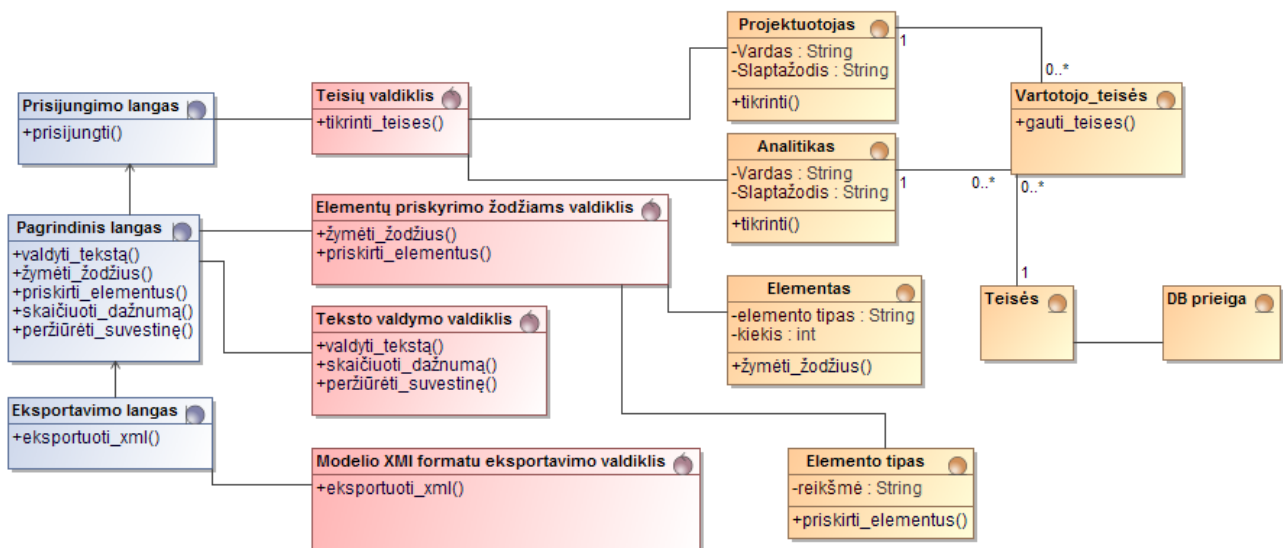
Loginėje įrankio preliminariems *UML* modeliams kurti iš teksto architektūroje atsispindi visi posistemiai dalyvaujantys modelių kūrimo iš teksto procese (3.2 pav.).

Pagrindinis posistemis – tėvinis posistemis teksto valdymo posistemiiui, prisijungimo posistemiiui, elementų priskyrimo žodžiams posistemiiui bei reikalavimų eksportavimo posistemiiui. Posistemiai komunikuoja su duomenų baze naudodamiesi duomenų paslaugomis.



3.2 pav. Loginė sistemos architektūra

3.1.3. Veiklos logikos (valdymo ir esybių klasių) modelis



3.3 pav. Projektavimo klasių modelis

3.2. Sistemos elgsenos modelis

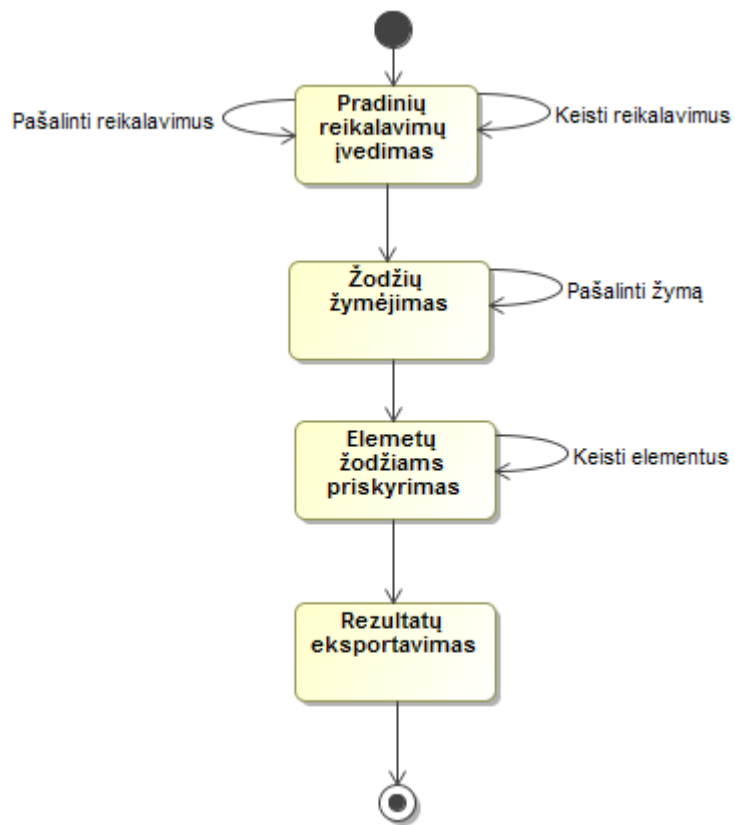
Sistemos elgsena pavaizduota įrankio preliminariniams *UML* modeliams kurti iš teksto būsenų modelyje (3.4 pav.).

Pirmoji būseną – pradinių reikalavimų įvedimas. Yra galimybė keisti reikalavimus, taip pat juos šalinti.

Antroji būseną – žodžių žymėjimas. Šioje būsenoje reikalavimuose pasirenkamas ir pažymimas žodis. Jei pažymėtas ne tas žodis, žymą galima pašalinti.

Trečioji būseną – elementų žodžiams priskyrimas. Šioje būsenoje pažymėtiems žodžiams priskiriami elementai. Jei priskiriamas neteisingas elementas, žymą galima pakeisti.

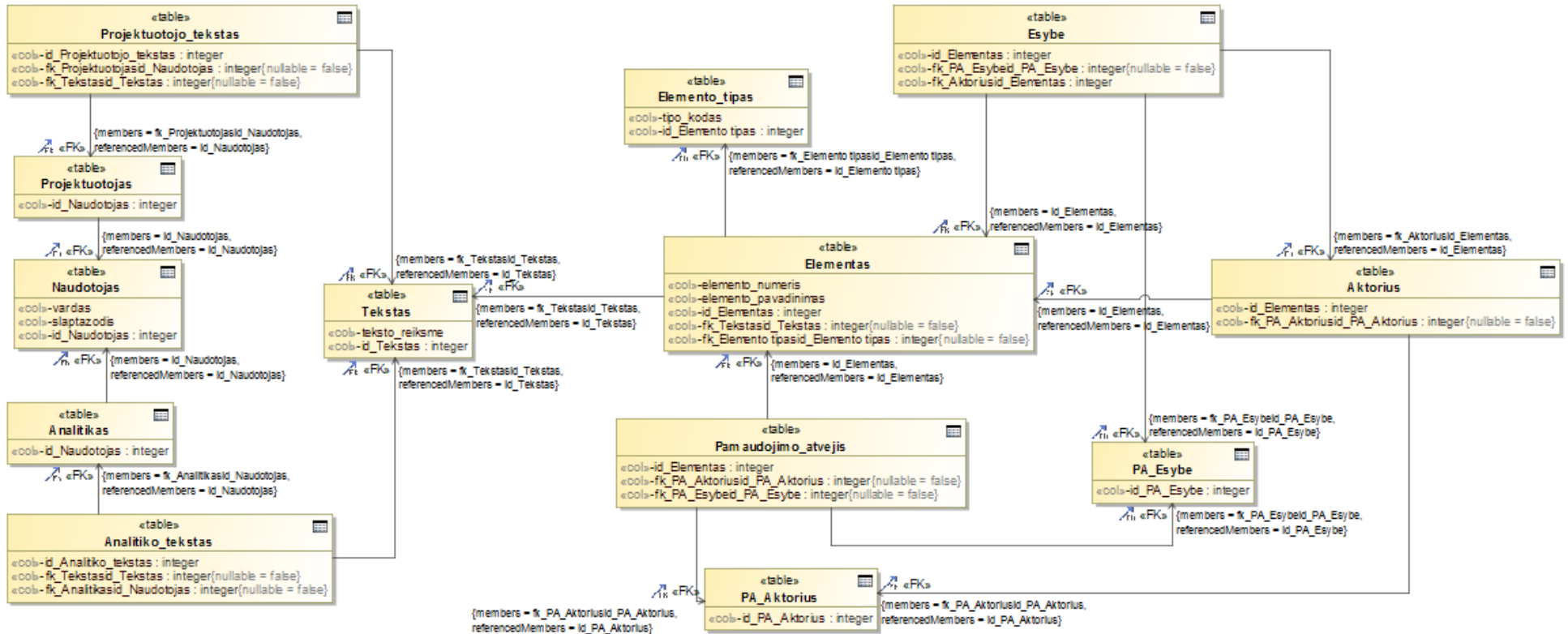
Ketvirtoji būseną – paskutinė būseną, kurios metu eksportuojama rezultatų failas, kurią vėliau galima importuoti į *CASE* įrankį.



3.4 pav. Įrankio būsenų modelis

3.3. Duomenų bazės schema

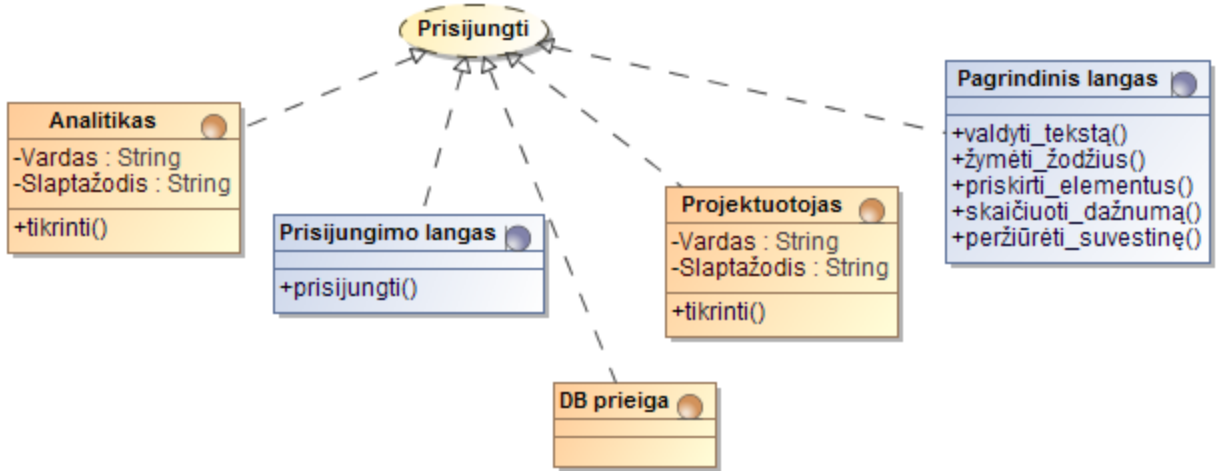
Įrankio preliminariems *UML* modeliams kurti iš teksto duomenų bazės schemoje atvaizduotos visos modelių kūrimo procese dalyvaujančios lentelės. Visi duomenys (duomenys apie vartotojus, tekstai, elementai, ryšiai ir kt.), dalyvaujantys procese, saugomi duomenų bazėje (3.5 pav.).



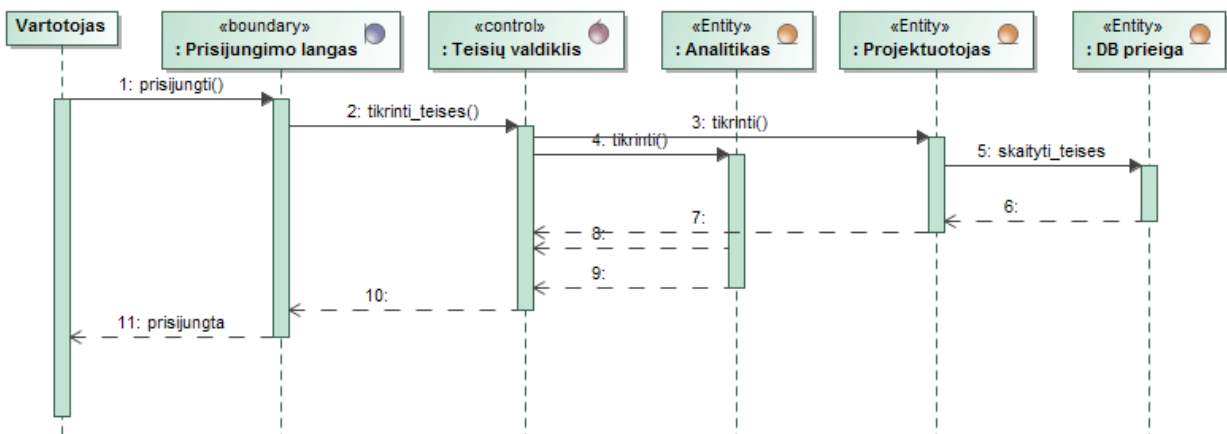
3.5 pav. *UML* SQL diagrama

3.4. Detalus projektas

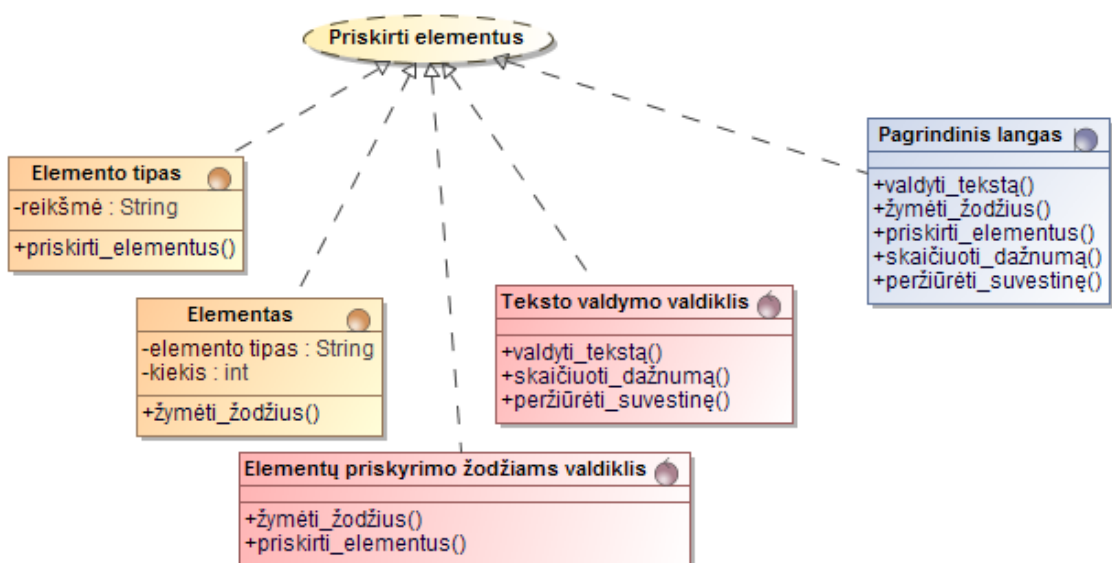
Žemiau pateikta prisijungimo panaudojimo atvejo bei pagrindinio, elementų priskyrimo žodžiams, panaudojimo atvejo realizacijos bei sekų diagramos. (3.6 – 3.9 pav.)



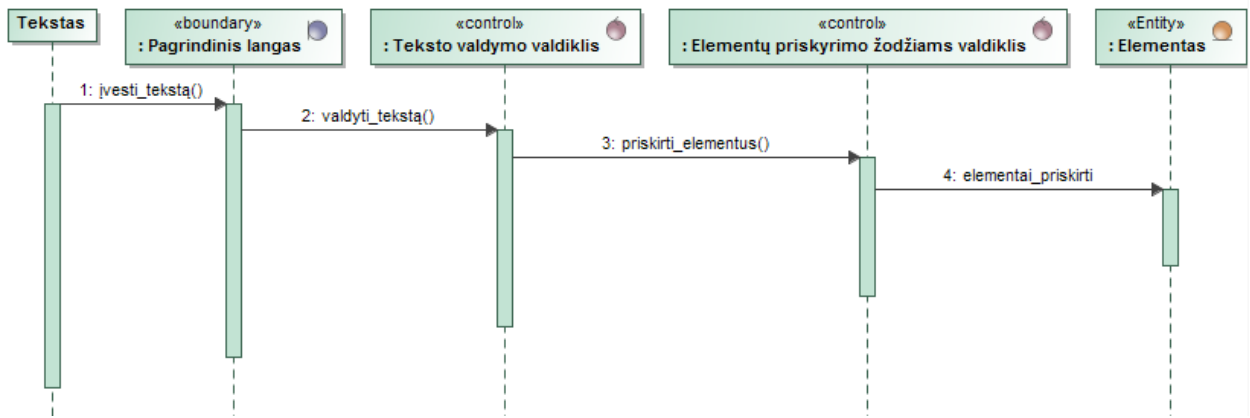
3.6 pav. Prisijungimo panaudojimo atvejo realizacija



3.7 pav. Prisijungimo panaudojimo atvejo sekų diagrama



3.8 pav. Elementų priskyrimo panaudojimo atvejo realizacija

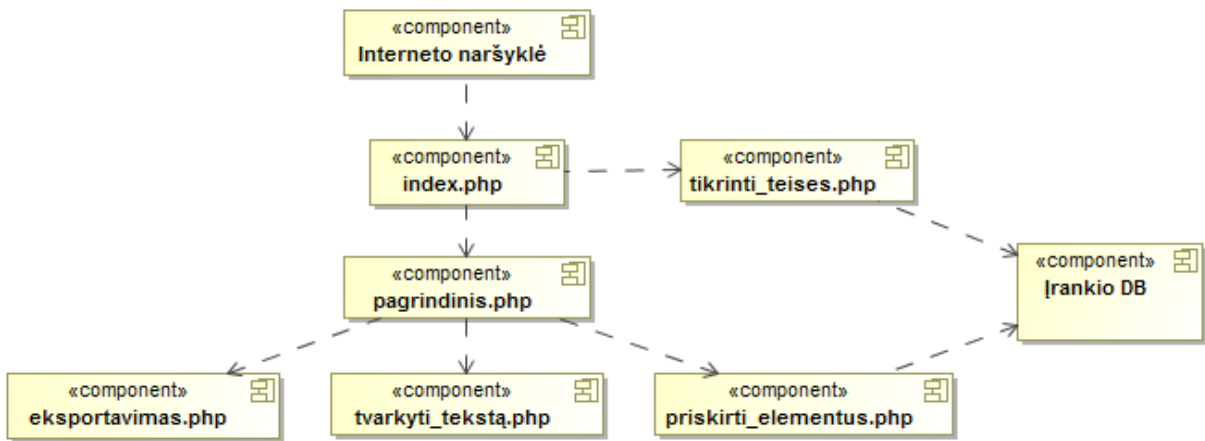


3.9 pav. Elementų priskyrimo panaudojimo atvejo sekų diagrama

3.5. Realizacijos modelis

Realizacijos modelis vaizduojamas programinių komponentų bei sistemos diegimo modeliais.

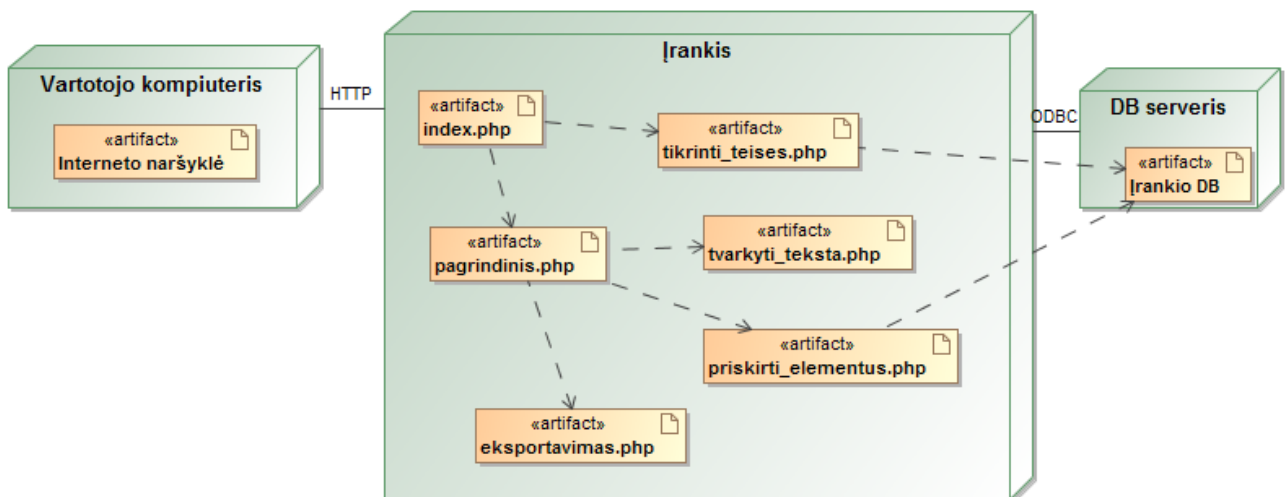
3.5.1. Programinių komponentų architektūra



3.10 pav. Komponentų diagrama

3.5.2. Diegimo modelis

Įrankio diegimo modelis rodo, jog įrankis veikia nutolusiame serveryje (3.11 pav.). Įrankiu galima naudotis interneto naršyklės pagalba. *HTTP* protokolo pagalba, vartotojo kompiuteris komunikuoja su įrankiu, o įrankis, *ODBC* sąsajos pagalba komunikuoja su duomenų baze.



3.11 pav. Sistemos diegimo modelis

4. SPRENDIMO REALIZACIJA IR TESTAVIMAS

4.1. Sprendimo realizacijos ir veikimo aprašas

4.1.1. Realizacijos priemonės

Atlikus galimų realizacijos įrankių analizę, prieita išvados, jog įrankį preliminariems *UML* modeliams kurti iš teksto patogiau realizuoti *PHP* programavimo kalba. Tai padidina sistemos naujumą, kadangi atsiranda papildomų privalumų, kurių panašaus profilio programos neturi. Pvz.:

- Sistemos nereikia diegti;
- Ji yra prieinama iš bet kurio įrenginio, turinčio interneto naršyklę;
- Sistema patogiu naudotis, įvertinant naudotojų šiuolaikinį pripratimą naudotis programomis esančiomis tinkle, neapkraunant savęs nereikalingais veiksmais.

Sistema realizuota naudojantis *YII* karkasu.

YII karkasas yra paremtas *MVC* (angl. *Model – View - Controller*) šablonu.

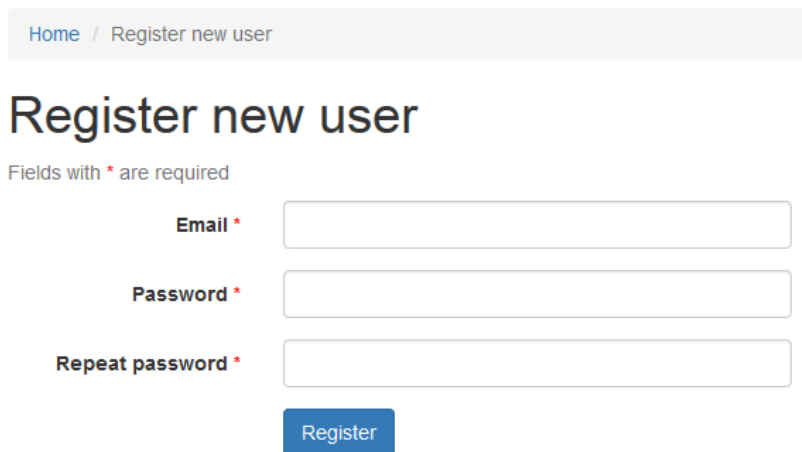
MVC šabloną sudaro trys elementai:

1. Valdiklis (angl. *Controller*) – siunčia komandas vaizdui (angl. *View*) ir modeliui (angl. *Model*). Valdiklyje nėra atliekama jokia verslo logika, o parametrai gaunami *HTTP* užklausos pagalba.
2. Vaizdas – Gauna norimą atvaizduoti informaciją iš valdiklio ir atvaizduoja suformuotą *HTML* dokumentą. Parametrai nėra gaunami per *HTTP* užklausą, taip pat nėra vykdoma verslo logika.
3. Modelis – atlieka visą verslo logiką, modelyje parametrai negali būti gaunami per *HTTP* užklausą. [5]

YII karkasas pasirinktas išanalizavus Manto Vaitkūno magistro darbą „*PHP* karkasų architektūrinis tyrimas“ [6]. Darbe atlikti tyrimai rodo, jog *YII* karkasas imlumo tyrime buvo nustatytas kaip labiausiai imliausias karkasas, t.y. *YII* yra labiausiai suprantamas ir juo kurti sistemos yra paprasta ir greitai. Karkasų našumo tyrime buvo nustatyta jog *YII* karkasas yra greičiausias. Pasikliauta darbe pateiktomis rekomendacijomis naudoti *YII* karkasą dėl jo greičio ir darbo našumo, tačiau tik nedideliuose arba vidutiniškuose projektuose. Kadangi aprašomas įrankis nėra sudėtingas, jis realizuotas *YII* karkaso pagalba.

4.1.2. Įrankio preliminariems *UML* modeliams kurti iš teksto veikimas

Įrankis pasiekiamas interneto naršyklės pagalba. Prisijungus prie sistemos matomas pradinis jos langas. Administratorius gali pasirinkti registruoti naują vartotoją (4.1 pav.).



Home / Register new user

Register new user

Fields with * are required

Email *

Password *

Repeat password *

Register

4.1 pav. Naujo vartotojo registracija

Prisijungę ir meniu juostoje (4.2 pav.) pasirinkę meniu punktą „Programa“ vartotojai gali naudotis pačia sistema. Realizuotas teksto įvedimo laukas, įvedamas tekstas yra išsaugomas ir atsiranda kiek žemiau (4.3 pav.).

4.2 pav. Meniu juosta

Home / Program list / Program

Please enter text title

Antras atvejis

Please enter your text below to begin creating a new model

Portalo pagrindinis tikslas yra suburti kuo daugiau įvairių profesijų ir pomėgių žmonių į vieną renginių organizavimo erdvę, kurioje vartotojas galės lengvai susiplanuoti savo rengiamą šventę. Neregistruotas vartotojas (svečias) gali apsilankyti portale, susipažinti su portalo struktūra, paskirtimi bei tikslais. Gali užsiregistruoti bei tapti portalo nariu. Užsiregistravęs vartotojas tampa paslaugų ieškotoju. Paslaugų ieškotojas gali peržiūrėti portalo paslaugų siūlytojus, peržiūrėti informaciją apie juos bei jų teikiamas paslaugas. Gali pradėti organizuoti savo renginį, siųsti žinutes paslaugų siūlytojams.

Please save your text to database

Save text

Portalo pagrindinis tikslas yra suburti kuo daugiau įvairių profesijų ir pomėgių žmonių į vieną renginių organizavimo erdvę, kurioje vartotojas galės lengvai susiplanuoti savo rengiamą šventę. Neregistruotas vartotojas (svečias) gali apsilankyti portale, susipažinti su portalo struktūra, paskirtimi bei tikslais. Gali užsiregistruoti bei tapti portalo nariu. Užsiregistravęs vartotojas tampa paslaugų ieškotoju. Paslaugų ieškotojas gali peržiūrėti portalo paslaugų siūlytojus, peržiūrėti informaciją apie juos bei jų teikiamas paslaugas. Gali pradėti organizuoti savo renginį, siųsti žinutes paslaugų siūlytojams. Prisiregistravęs vartotojas gali pasipildyti virtualią sąskaitą, įdėti paslaugos siūlymo skelbimą ir tapti paslaugos siūlytoju. Gali siuntinėti žinutes kitiems registruotiems vartotojams, paslaugų siūlytojams. Administratorius: išskirtines teises turintis asmuo portale. Gali administruoti vartotojus, suteikti teises.

4.3 pav. Teksto įvedimo laukas

Įvedus naujus, laisvu tekstu suformuotus reikalavimus, vartotojas gali pradėti teksto analizę. Kaip pavyzdys yra paimta teksto ištrauka iš reikalavimų švenčių bei renginių organizavimo portalui:

„Portalo pagrindinis tikslas yra suburti kuo daugiau įvairių profesijų ir pomėgių žmonių į vieną renginių organizavimo erdvę, kurioje vartotojas galės lengvai susiplanuoti savo rengiamą šventę.

Neregistruotas vartotojas (svečias) gali apsilankyti portale, susipažinti su portalo struktūra, paskirtimi bei tikslais. Gali užsiregistruoti bei tapti portalo nariu. Užsiregistravęs vartotojas tampa paslaugų ieškotoju.

Paslaugų ieškotojas gali peržiūrėti portalo paslaugų siūlytojus, peržiūrėti informaciją apie juos bei jų teikiamas paslaugas. Gali pradėti organizuoti savo renginį, siųsti žinutes paslaugų siūlytojams.





Prisiregistravęs vartotojas gali pasipildyti virtualią sąskaitą, įdėti paslaugos siūlymo skelbimą ir tapti paslaugos siūlytoju. Gali siuntinėti žinutes kitiems registruotiems vartotojams, paslaugų siūlytojams.






Administratorius – išskirtines teises turintis asmuo portale. Gali administruoti vartotojus, suteikti teises.“

Iš teksto galime spręsti, jog vyrauja keturi aktoriai ir keturiolika panaudojimo atvejų. Vartotojas aktorius bei panaudojimo atvejus pasižymi atlikdamas veiksmus (4.4 pav.):

1. Paspaudus ant spalvos lauko pasirenkama, ką norima žymėti – aktorių (mėlyna spalva) ar panaudojimo atvejį (gelsva spalva)
2. Teksto įvedimo lauke pelės kursoriumi pažymimas norimas žodis ar žodžių junginys.
3. Lentelėse po elementų pavadinimais atsiranda pasirinkti elementai.

Please click on the element to select it and then mark your text in the text area above

Actor	
1.	Neregistruotas vartotojas 
2.	Paslaugų ieškotojas 
3.	Prisiregistravęs vartotojas 
4.	Administratorius 

Use case	
apsilankyti portale	Selections 
Selected actors: Neregistruotas vartotojas	
susipažinti su portalo struktūra	Selections 
Selected actors: Neregistruotas vartotojas	
užsiregistruoti	Selections 
Selected actors: Neregistruotas vartotojas	
tapti portalo nariu	Selections 
Selected actors: Neregistruotas vartotojas	
peržiūrėti portalo paslaugų siūlytojus	Selections 
Selected actors: Paslaugų ieškotojas	

4.4 pav. Elementų priskyrimo langas

Aktorių stulpelyje kiekvienas aktorius yra automatiškai sunumeruojamas. Panaudojimo atvejų stulpelyje prie kiekvieno priskirto panaudojimo atvejo yra iškrentantis pasirinkimo sąrašas, kuriame pasirinkus „*Select Actor*“ bei pažymėjus norimą aktorių, panaudojimo atvejui priskiriamas aktorius, kuriam priklauso tas panaudojimo atvejis. (4.5 pav.)

Please select actors

Neregistruotas vartotojas

Paslaugų ieškotojas

Prisiregistravęs vartotojas

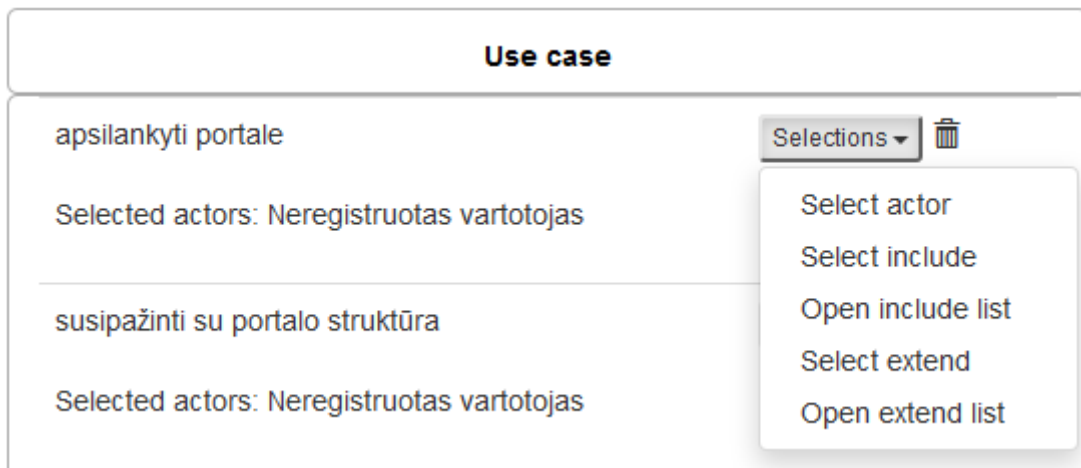
Administratorius

Confirm
Cancel

4.5 pav. Elemento priskyrimas aktoriui

Panaudojimo atvejams, priskyrus aktorių, taip pat galima priskirti „*include*“ ar „*extend*“ panaudojimo atvejus tokiu pat principu kaip žymimi ir pagrindiniai panaudojimo atvejai (4.6 pav.).

Išanalizavus turimą tekstą, eksportuojamas rezultatų modelio *XMI* formatu failas, kuris vėliau įkeliamas į *MagicDraw* įrankį ir tęsiamas programinės įrangos modeliavimo procesas. (Modelio *XMI* formatu failo turinys pridedamas Priede Nr. 2).



4.6 pav. „include“ ar „extend“ priskyrimas

4.2. Testavimo modelis, duomenys, rezultatai

Testavimas buvo atliktas Selenium testavimo karkasu. Sugeneruoti testai ir jų rezultatai pateikiami Priede Nr. 1.

5. EKSPERIMENTINIS ĮRANKIO PRELIMINARIEMS UML MODELIAMS KURTI IŠ TEKSTO TYRIMAS

5.1. Eksperimento planas

Eksperimentas vykdomas eksperimentinėje aplinkoje pasitelkiant eksperimentinius duomenis. Eksperimento metu bus atliekama 10 bandymų. Jo tikslas yra išanalizuoti įrankio preliminariems UML modeliams kurti iš teksto prototipą, siekiant patvirtinti, kad prototipas veikia tinkamai ir leidžia sudaryti preliminarius modelius iš natūralios kalbos teksto.

Prototipo tinkamumas bus vertinamas tikslumo P (angl. *Precision*) ir išsamumo R (angl. *Recal*) kriterijais:

$$P_i = \frac{TE_i}{GE_i}$$
$$R_i = \frac{TE_i}{VE_i}$$
$$P = \sum_{i=1}^n P_i$$
$$R = \sum_{i=1}^n R_i$$

Čia P_i , R_i – i -tojo bandymo tikslumas ir išsamumas, $i = 1, \dots, n$;

TE_i – teisingai transformuotų elementų (aktorių, panaudojimo atvejų, asociacijų) i -tajame bandyme skaičius;

GE_i – visų transformuotų elementų i -tajame bandyme skaičius;

VE_i – visų elementų, kurie turėjo būti transformuoti i -tajame bandyme, skaičius.

Eksperimentas susideda iš dviejų dalių – eksperimentas su griežtos struktūros tekstu ir eksperimentas su laisvos struktūros tekstu.

Eksperimento kontekstas – eksperimentinė aplinka (mokymo laboratorija), kur eksperimentą vykdo studentas, kiekvienai eksperimento daliai panaudodamas 10 specialiai sukurtų reikalavimus aprašančių tekstų pavyzdžių.

Eksperimento vykdymas – kiekvienam bandymui pateikiama aktorių ir panaudojimo atvejų lentelė ir numatomas gauti rezultatas – rankiniu būdu nubraižyta panaudojimo atvejų diagrama. Reikalavimai sukeliama į sukurtą įrankį, juose sužymimi aktoriai bei panaudojimo atvejai ir transformuojami į panaudojimo atvejų modelį.

5.1.1. Eksperimentas su griežtos struktūros tekstu

Pirmasis bandymas. Pateikiamas reikalavimų aprašas:

„CRM sistemos (toliau: sistema) pagrindinis tikslas – kaupti informaciją apie darbuotojus, klientus, prekes bei jų grupes, darbuotojų susitikimus su klientais bei pardavimus.

Vadybininkas – sistemos vartotojas, galintis kaupti informaciją apie klientus, paskirti klientams susitikimus bei registruoti įvykdytus pardavimus.

Padalinio vadovas gali matyti visą padalinio informaciją: vadybininkus, jų veiksmus su klientais.

Administratorius – išskirtines teises turintis asmuo portale. Jis gali administruoti vartotojus (vartotojo registracija, vartotojo šalinimas, vartotojo redagavimas), suteikti vartotojams teises, priskirti vartotojams regionus.“

5.1 lentelė. Pirmojo bandymo su griežtos struktūros tekstu suvestinė

Aktoriai	Panaudojimo atvejai
Vadybininkas	Kaupti informaciją apie klientus; paskirti klientams susitikimus; registruoti įvykdytus pardavimus
Padalinio vadovas	Matyti visą padalinio informaciją

Administratorius	Administruoti vartotojus - vartotojo registracija, vartotojo šalinimas, vartotojo redagavimas ; suteikti vartotojams teises; priskirti vartotojams regionus.
-------------------------	--

Antrasis bandymas. Pateikta ištrauka iš reikalavimų švenčių organizavimo portalui.

„Portalo pagrindinis tikslas yra suburti kuo daugiau įvairių profesijų ir pomėgių žmonių į vieną renginių organizavimo erdvę, kurioje vartotojas galės lengvai susiplanuoti savo rengiamą šventę.

Neregistruotas vartotojas (svečias) gali apsilankyti portale, susipažinti su portalo struktūra, paskirtimi bei tikslais. Gali užsiregistruoti bei tapti portalo nariu. Užsiregistravęs vartotojas tampa paslaugų ieškotoju.

Paslaugų ieškotojas gali peržiūrėti portalo paslaugų siūlytojus, peržiūrėti informaciją apie juos bei jų teikiamas paslaugas. Gali pradėti organizuoti savo renginį, siųsti žinutes paslaugų siūlytojams.

Prisiregistravęs vartotojas gali papildyti virtualią sąskaitą, įdėti paslaugos siūlymo skelbimą ir tapti paslaugos siūlytoju. Gali siuntinėti žinutes kitiems registruotiems vartotojams, paslaugų siūlytojams.

Administratorius – išskirtines teises turintis asmuo portale. Gali administruoti vartotojus, suteikti teises.“

5.2 lentelė. Antrojo bandymo su griežtos struktūros tekstu suvestinė

Aktoriai	Panaudojimo atvejai
Neregistruotas vartotojas	apsilankyti portale; susipažinti su portalo struktūra; užsiregistruoti bei tapti portalo nariu
Paslaugų ieškotojas	peržiūrėti portalo paslaugų siūlytojus; peržiūrėti informaciją apie juos bei jų teikiamas paslaugas; pradėti organizuoti savo renginį; siųsti žinutes paslaugų siūlytojams
Prisiregistravęs vartotojas	papildyti virtualią sąskaitą; įdėti paslaugos siūlymo skelbimą; tapti paslaugos siūlytoju; siuntinėti žinutes
Administratorius	administruoti vartotojus; suteikti teises

Trečiasis bandymas. Aprašyti elementarūs studento, atėjusio pasiimti stipendijos iš bankomato veiksmai.

„Studentas nori pasinaudoti bankomatu. Jis turi prisijungti, peržiūrėti sąskaitos likutį, pasiimti pinigus ir atsijungti.“

5.3 lentelė. Trečiojo bandymo su griežtos struktūros tekstu suvestinė

Aktoriai	Panaudojimo atvejai
Studentas	Pasinaudoti bankomatu; prisijungti; peržiūrėti sąskaitos likutį; pasiimti pinigus; atsijungti

Ketvirtasis bandymas. Aprašoma internetinės prekybos sistema su dviejų tipų vartotojais.

„Klientas, apsilankęs internetinėje parduotuvėje gali pateikti užsakymą (įtraukti prekę į užsakymą bei nurodyti kontaktinius duomenis). Paprastas klientas gali pateikti paprastą užsakymą, o auksinis klientas gali pateikti auksinį užsakymą. Pateikdamas užsakymą, auksinis klientas gali ieškoti prekės išplėstiniame kataloge bei pasirinkti pristatymo būdą. Paprastas klientas, pateikdamas užsakymą gali ieškoti prekės įmonės kataloge.“

5.4 lentelė. Ketvirtojo bandymo su griežtos struktūros tekstu suvestinė

Aktoriai	Panaudojimo atvejai
Paprastas klientas	pateikti užsakymą (įtraukti prekę į užsakymą bei nurodyti kontaktinius duomenis); pateikti paprastą užsakymą; ieškoti prekės įmonės kataloge
Auksinis klientas	pateikti užsakymą (įtraukti prekę į užsakymą bei nurodyti kontaktinius duomenis); pateikti auksinį užsakymą; ieškoti prekės išplėstiniame kataloge; pasirinkti pristatymo būdą

Penktasis bandymas. Ištrauka iš reikalavimų bibliotekos sistemai. Trumpai aprašyti reikalavimai knygų užsakymo moduliui.

„Bibliotekos sistema skirta internetu užsisakyti norimas knygas. Skaitytojas, norėdamas pasinaudoti sistema ir turėdamas savo prisijungimo duomenis gali prisijungti prie sistemos, peržiūrėti knygų sąrašą bei pateikti knygų užsakymą. Jei skaitytojas pamato, kad kažkurios knygos jis visgi nenori, gali atšaukti užsakymą.“

5.5 lentelė. Penktojo bandymo su griežtos struktūros tekstu suvestinė

Aktoriai	Panaudojimo atvejai
Skaitytojas	Pasinaudoti sistema; prisijungti prie sistemos; peržiūrėti knygų sąrašą; pateikti knygų užsakymą; atšaukti užsakymą

Šeštasis bandymas. Aptariami reikalavimai profesionalaus krepšinio žaidėjų ir laisvųjų agentų rinkos valdymo informacinei sistemai.

„Pagrindinė problema, kurią turės išspręsti profesionalaus krepšinio žaidėjų ir laisvųjų agentų rinkos valdymo informacinė sistema, yra ta, kad klubai galėtų greičiau susirasti jiems norimus žaidėjus. Iškilę poreikis krepšininkus ir laisvuosius agentus suvesti į informacinę sistemą. Yra dviejų tipų vartotojai: žaidėjas ir klubo direktorius. Abu vartotojai gali registruotis informaciniame sistemoje, įvesti duomenis apie save, padaryti savo profilį konfidencialiu. Žaidėjas gali peržiūrėti klubų statistiką, peržiūrėti informaciją apie klubų mokumą bei priimti ar atmesti klubo pasiūlymą. Klubo direktorius gali peržiūrėti žaidėjo statistiką bei išsiųsti pasiūlymą žaidėjui.“

5.6 lentelė. Šeštojo bandymo su griežtos struktūros tekstu suvestinė

Aktoriai	Panaudojimo atvejai
Žaidėjas	registruotis informaciniame sistemoje, įvesti duomenis apie save, padaryti savo profilį konfidencialiu; peržiūrėti klubų statistiką; peržiūrėti informaciją apie klubų mokumą; priimti ar atmesti klubo pasiūlymą
Klubo direktorius	registruotis informaciniame sistemoje, įvesti duomenis apie save, padaryti savo profilį konfidencialiu; peržiūrėti žaidėjo statistiką; išsiųsti pasiūlymą žaidėjui

Septintasis bandymas. Pateikiama ištrauka iš reikalavimų ryšio padengiamumo sistemos.

„Ryšio padengiamumo sistemos tikslas yra optimaliai išdėstyti bevielio ryšio stoteles patalpose. Inžinierius gali importuoti patalpų planą, interaktyviai dėlioti stoteles, sekti ryšio padengiamumą ir taip matyti, kuriose vietose pastatytos bevielio ryšio stotelės padengia visą patalpą tolygiai, kas leidžia inžinieriui parengti bevielio ryšio stotelių statymo planą.“

5.7 lentelė. Septintojo bandymo su griežtos struktūros tekstu suvestinė

Aktoriai	Panaudojimo atvejai
Inžinierius	importuoti patalpų planą; interaktyviai dėlioti stoteles; sekti ryšio padengiamumą; parengti bevielio ryšio stotelių statymo planą

Aštuntasis bandymas. Banko sistemos indėlių tvarkymo posistemio veikimas. Duoti trumpi reikalavimai.

„Prisijungęs prie elektroninės bankininkystės sistemos bei pasirinkęs indėlių tvarkymo posistemę, naudotojas gali matyti turimus indėlius, sekti indėlių palūkanas. Jei naudotojas neturi indėlio, turi būti realizuota galimybė pačiam padėti indėlį, sudaryti sutartį su banku. Vadybininkas, gavęs pranešimą apie norimą pasirašyti sutartį, gali peržiūrėti sutartį, patvirtinti arba atmesti sutartį.“

5.8 lentelė. Aštuntojo bandymo su griežtos struktūros tekstu suvestinė

Aktoriai	Panaudojimo atvejai
Naudotojas	matyti turimus indėlius; sekti indėlių palūkanas; padėti indėlį; sudaryti sutartį su banku;
Vadybininkas	peržiūrėti sutartį; (patvirtinti, atmesti)

Devintasis bandymas. Aprašoma sistema skirta darbuotojų darbo laiko apskaitai.

„Sistemos tikslas – sekti informaciją apie darbuotojo išdirbtą laiką. Darbuotojas privalo kiekvieną dieną žymėti savo išdirbtą laiką. Atskirai nuo privalomo išdirbti laiko, darbuotojas turi pateikti išdirbtų viršvalandžių skaičių, jei tokių buvo. Darbuotojo vadovas sistemoje turi matyti darbuotojų dirbtas valandas, gali priimti sprendimą dėl viršvalandžių (patvirtinti arba atmesti.)“

5.9 lentelė. Devintojo bandymo su griežtos struktūros tekstu suvestinė

Aktorius	Panaudojimo atvejis
Darbuotojas	žymėti savo išdirbtą laiką; pateikti išdirbtų viršvalandžių skaičių
Vadovas	matyti darbuotojų dirbtas valandas; priimti sprendimą dėl viršvalandžių (patvirtinti, atmesti)

Dešimtas bandymas. Laivų kontrolės jūroje sistema. Ištrauka iš reikalavimų.

„Pagrindinis sistemos tikslas – stebėti laivus esančius jūroje, reguliuoti jų kursą, padėti kritinėse situacijose. Operatorius, budintis krante, gali matyti laivus, esančius jam priskirtame regione jūroje. Gali kontroliuoti kursą – jei reikia, nukreipti į šoną, patikslinti kursą. Operatorius taip pat gali perspėti apie pavojus (pranešti apie ledkalnius, pranešti apie pavojingai artėjančius laivus, perspėti apie artėjančią pasienio ribą).“

5.10 lentelė. Dešimtojo bandymo su griežtos struktūros tekstu suvestinė

Aktorius	Panaudojimo atvejis
Operatorius	matyti laivus; kontroliuoti kursą; nukreipti į šoną, patikslinti kursą; perspėti apie pavojus (pranešti apie ledkalnius, pranešti apie pavojingai artėjančius laivus, perspėti apie artėjančią pasienio ribą)

5.1.2. Eksperimentas su laisvos struktūros tekstu

Pirmasis bandymas. Pateikiamas reikalavimų aprašas:

„CRM sistemos (toliau: sistema) pagrindinis tikslas – kaupti informaciją apie darbuotojus, klientus, prekes bei jų grupes, darbuotojų susitikimus su klientais bei pardavimus.

Vadybininkas – sistemos vartotojas, galintis kaupti informaciją apie klientus. Vadybininkas taip pat paskiria klientams susitikimus bei registruoja įvykdytus pardavimus.

Padalinio vadovas mato visą padalinio informaciją: vadybininkus, jų veiksmus su klientais.

Administratorius – išskirtines teises turintis asmuo portale. Jis gali administruoti vartotojus (vartotojo registracija, vartotojo šalinimas, vartotojo redagavimas), suteikti vartotojams teises, priskirti vartotojams regionus.“

5.11 lentelė. Pirmojo bandymo su laisvos struktūros tekstu suvestinė

Aktoriai	Panaudojimo atvejai
Vadybininkas	Kaupti informaciją apie klientus; paskirti klientams susitikimus; registruoti įvykdytus pardavimus
Padalinio vadovas	Matyti visą padalinio informaciją
Administratorius	Administruoti vartotojus - vartotojo registracija, vartotojo šalinimas, vartotojo redagavimas ; suteikti vartotojams teises; priskirti vartotojams regionus.

Antrasis bandymas. Pateikta ištrauka iš reikalavimų švenčių organizavimo portalui.

„Portalo pagrindinis tikslas yra suburti kuo daugiau įvairių profesijų ir pomėgių žmonių į vieną renginių organizavimo erdvę, kurioje vartotojas galės lengvai susiplanuoti savo rengiamą šventę.

Neregistruotas vartotojas (svečias) gali apsilankyti portale, susipažinti su portalo struktūra, paskirtimi bei tikslais. Neregistruotas vartotojas registruojasi portale taip tapdamas portalo nariu. Užsiregistravęs vartotojas tampa paslaugų ieškotoju.

Paslaugų ieškotojas gali peržiūrėti portalo paslaugų siūlytojus, peržiūrėti informaciją apie juos bei jų teikiamas paslaugas. Gali pradėti organizuoti savo renginį, siųsti žinutes paslaugų siūlytojams.

Prisiregistravęs vartotojas gali papildyti virtualią sąskaitą, įdėti paslaugos siūlymo skelbimą ir tapti paslaugos siūlytoju. Gali siuntinėti žinutes kitiems registruotiems vartotojams, paslaugų siūlytojams.

Administratorius – išskirtines teises turintis asmuo portale. Administruoja vartotojus, suteikdamas jiems teises.“

5.12 lentelė. Antrojo bandymo su laisvos struktūros tekstu suvestinė

Aktoriai	Planuojami panaudojimo atvejai
Neregistruotas vartotojas	apsilankyti portale; susipažinti su portalo struktūra; užsiregistruoti bei tapti portalo nariu

Paslaugų ieškotojas	peržiūrėti portalo paslaugų siūlytojus; peržiūrėti informaciją apie juos bei jų teikiamas paslaugas; pradėti organizuoti savo renginį; siųsti žinutes paslaugų siūlytojams
Prisiregistravęs vartotojas	pasipildyti virtualią sąskaitą; įdėti paslaugos siūlymo skelbimą; tapti paslaugos siūlytoju; siuntinėti žinutes
Administratorius	administruoti vartotojus; suteikti teises

Trečiasis bandymas. Aprašyti elementarūs studento, atėjusio pasiimti stipendijos iš bankomato veiksmai.

„Studentas nori pasinaudoti bankomatu. Jis prisijungia, peržiūri sąskaitos likutį, pasiima pinigus ir atsijungia.“

5.13 lentelė. Trečiojo bandymo su laisvos struktūros tekstu suvestinė

Aktorai	Panaudojimo atvejai
Studentas	Pasinaudoti bankomatu; prisijungti; peržiūrėti sąskaitos likutį; pasiimti pinigus; atsijungti

Ketvirtasis bandymas. Aprašoma internetinės prekybos sistema su dviejų tipų vartotojais.

„Klientas, apsilankęs internetinėje parduotuvėje gali pateikti užsakymą (įtraukti prekę į užsakymą bei nurodyti kontaktinius duomenis). Paprastas klientas gali pateikti paprastą užsakymą, o auksinis klientas gali pateikti auksinį užsakymą. Pateikdamas užsakymą, auksinis klientas gali ieško prekės išplėstiniame kataloge bei pasirinka pristatymo būdą. Paprastas klientas, pateikdamas užsakymą ieškoti prekės standartiniame įmonės kataloge.“

5.14 lentelė. Ketvirtojo bandymo su laisvos struktūros tekstu suvestinė

Aktorai	Panaudojimo atvejai
Paprastas klientas	pateikti užsakymą (įtraukti prekę į užsakymą bei nurodyti kontaktinius duomenis); pateikti paprastą užsakymą; ieškoti prekės įmonės kataloge
Auksinis klientas	pateikti užsakymą (įtraukti prekę į užsakymą bei nurodyti kontaktinius duomenis); pateikti auksinį užsakymą; ieškoti prekės išplėstiniame kataloge; pasirinkti pristatymo būdą

Penktasis bandymas. Ištrauka iš reikalavimų bibliotekos sistemai. Trumpai aprašyti reikalavimai knygų užsakymo moduliui.

„Bibliotekos sistema skirta internetu užsisakyti norimas knygas. Skaitytojas, norėdamas pasinaudoti sistema ir turėdamas savo prisijungimo duomenis prisijungia prie sistemos, peržiūri knygų sąrašą bei pateikia knygų užsakymą. Jei skaitytojas pamato, kad kažkurios knygos jis visgi nenori – atšaukia užsakymą.“

5.15 lentelė. Penktojo bandymo su laisvos struktūros tekstu suvestinė

Aktorai	Panaudojimo atvejai
----------------	----------------------------

Skaitytojas	Pasinaudoti sistema; prisijungti prie sistemos; peržiūrėti knygų sąrašą; pateikti knygų užsakymą; atšaukti užsakymą
--------------------	---

Šeštasis bandymas. Aptariami reikalavimai profesionalaus krepšinio žaidėjų ir laisvųjų agentų rinkos valdymo informacinei sistemai.

„Pagrindinė problema, kurią turės išspręsti profesionalaus krepšinio žaidėjų ir laisvųjų agentų rinkos valdymo informacinė sistema, yra ta, kad klubai galėtų greičiau susirasti jiems norimus žaidėjus. Iškilę poreikis krepšininkus ir laisvuosius agentus suvesti į informacinę sistemą. Yra dviejų tipų vartotojai: žaidėjas ir klubo direktorius. Abu vartotojai gali registruotis informaciniame sistemoje, įvesti duomenis apie save, padaryti savo profilį konfidencialiu. Žaidėjas peržiūri klubų statistiką, informaciją apie klubų mokumą bei priima arba atmeta klubo pasiūlymą. Klubo direktorius, peržiūrėjęs žaidėjo statistiką, gali išsiųsti pasiūlymą žaidėjui.“

5.16 lentelė. Šeštojo bandymo su laisvos struktūros tekstu suvestinė

Aktoriai	Panaudojimo atvejai
Žaidėjas	registruotis informaciniame sistemoje, įvesti duomenis apie save, padaryti savo profilį konfidencialiu; peržiūrėti klubų statistiką; peržiūrėti informaciją apie klubų mokumą; priimti ar atmesti klubo pasiūlymą
Klubo direktorius	registruotis informaciniame sistemoje, įvesti duomenis apie save, padaryti savo profilį konfidencialiu; peržiūrėti žaidėjo statistiką; išsiųsti pasiūlymą žaidėjui

Septintasis bandymas. Pateikiama ištrauka iš reikalavimų ryšio padengiamumo sistemos.

„Ryšio padengiamumo sistemos tikslas yra optimaliai išdėstyti bevielio ryšio stoteles patalpose. Inžinierius importuoja patalpų planą, interaktyviai dėlioja stoteles, seka ryšio padengiamumą ir taip mato, kuriose vietose pastatytos bevielio ryšio stotelės padengia visą patalpą tolygiai, kas leidžia inžinieriui parengti bevielio ryšio stotelių statymo planą.“

5.17 lentelė. Septintojo bandymo su laisvos struktūros tekstu suvestinė

Aktoriai	Panaudojimo atvejai
Inžinierius	importuoti patalpų planą; interaktyviai dėlioti stoteles; sekti ryšio padengiamumą; parengti bevielio ryšio stotelių statymo planą

Aštuntasis bandymas. Banko sistemos indėlių tvarkymo posistemio veikimas. Duoti trumpi reikalavimai.

„Prisijungęs prie elektroninės bankininkystės sistemos bei pasirinkęs indėlių tvarkymo posistemę, naudotojas mato turimus indėlius, seka indėlių palūkanas. Jei naudotojas neturi indėlio, gali pats jį padėti, taip sudarydamas sutartį su banku. Vadybininkas, gavęs pranešimą apie norimą pasirašyti sutartį, peržiūri sutartį, ją patvirtina arba atmeta.“

5.18 lentelė. Aštuntojo bandymo su laisvos struktūros tekstu suvestinė

Aktoriai	Panaudojimo atvejai
Naudotojas	matyti turimus indėlius; sekti indėlių palūkanas; padėti indėlį; sudaryti sutartį su banku;
Vadybininkas	peržiūrėti sutartį; (patvirtinti, atmesti)

Devintasis bandymas. Aprašoma sistema skirta darbuotojų darbo laiko apskaitai.

„Sistemos tikslas – sekti informaciją apie darbuotojo išdirbtą laiką. Darbuotojas kiekvieną dieną žymisi savo išdirbtą laiką. Atskirai nuo privalomo išdirbti laiko, darbuotojas pateikia išdirbtų viršvalandžių skaičių, jei tokių buvo. Darbuotojo vadovas sistemoje mato darbuotojų dirbtas valandas, priima sprendimą dėl viršvalandžių (patvirtinti arba atmesti).“

5.19 lentelė. Devintojo bandymo su laisvos struktūros tekstu suvestinė

Aktorius	Panaudojimo atvejis
Darbuotojas	žymėtis savo išdirbtą laiką; pateikti išdirbtų viršvalandžių skaičių
Vadovas	matyti darbuotojų dirbtas valandas; priimti sprendimą dėl viršvalandžių (patvirtinti, atmesti)

Dešimtas bandymas. Laivų kontrolės jūroje sistema. Ištrauka iš reikalavimų.

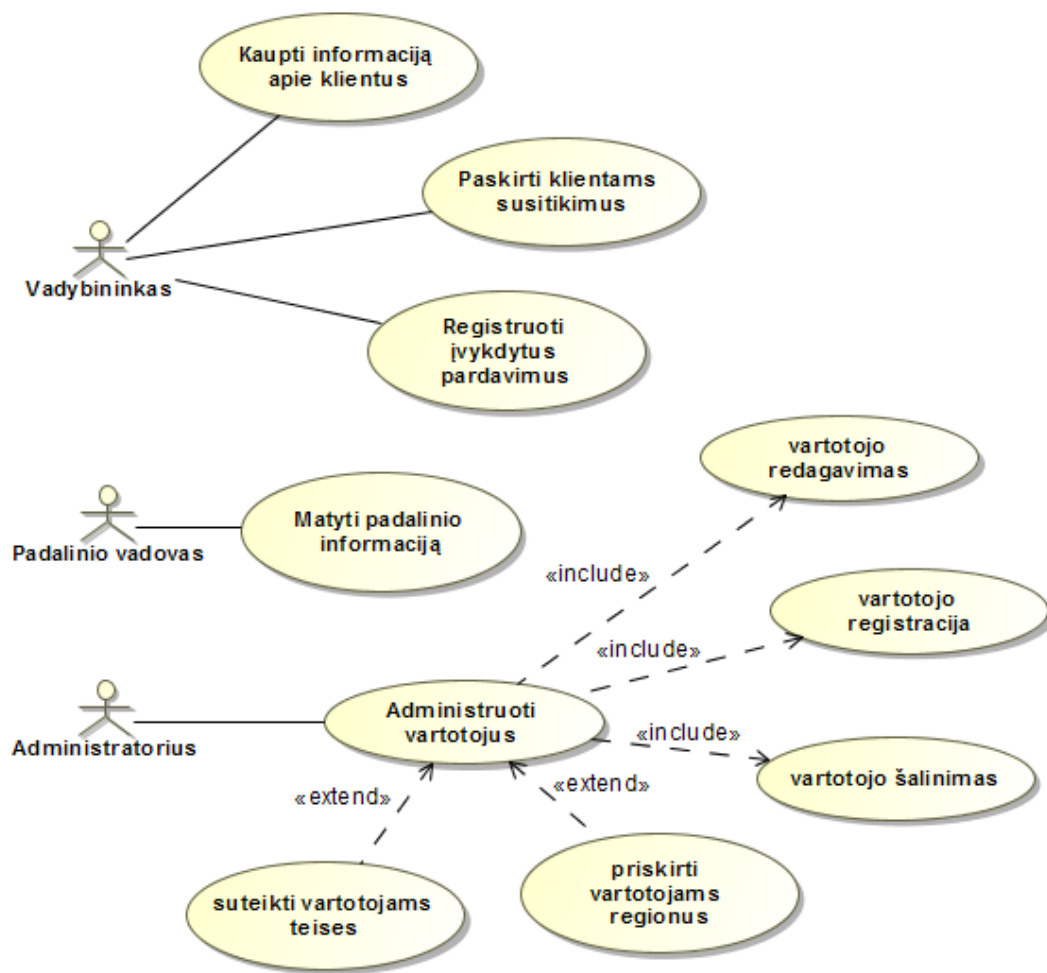
„Pagrindinis sistemos tikslas – stebėti laivus esančius jūroje, reguliuoti jų kursą, padėti kritinėse situacijose. Operatorius, budintis krante, mato laivus, esančius jam priskirtame regione jūroje. Kontroliuoja jų kursą – jei reikia, nukreipia į šoną, patikslina kursą. Operatorius taip pat gali perspėti apie pavojus (pranešti apie ledkalnius, pranešti apie pavojingai artėjančius laivus, perspėti apie artėjančią pasienio ribą).“

5.20 lentelė. Dešimtojo bandymo su laisvos struktūros tekstu suvestinė

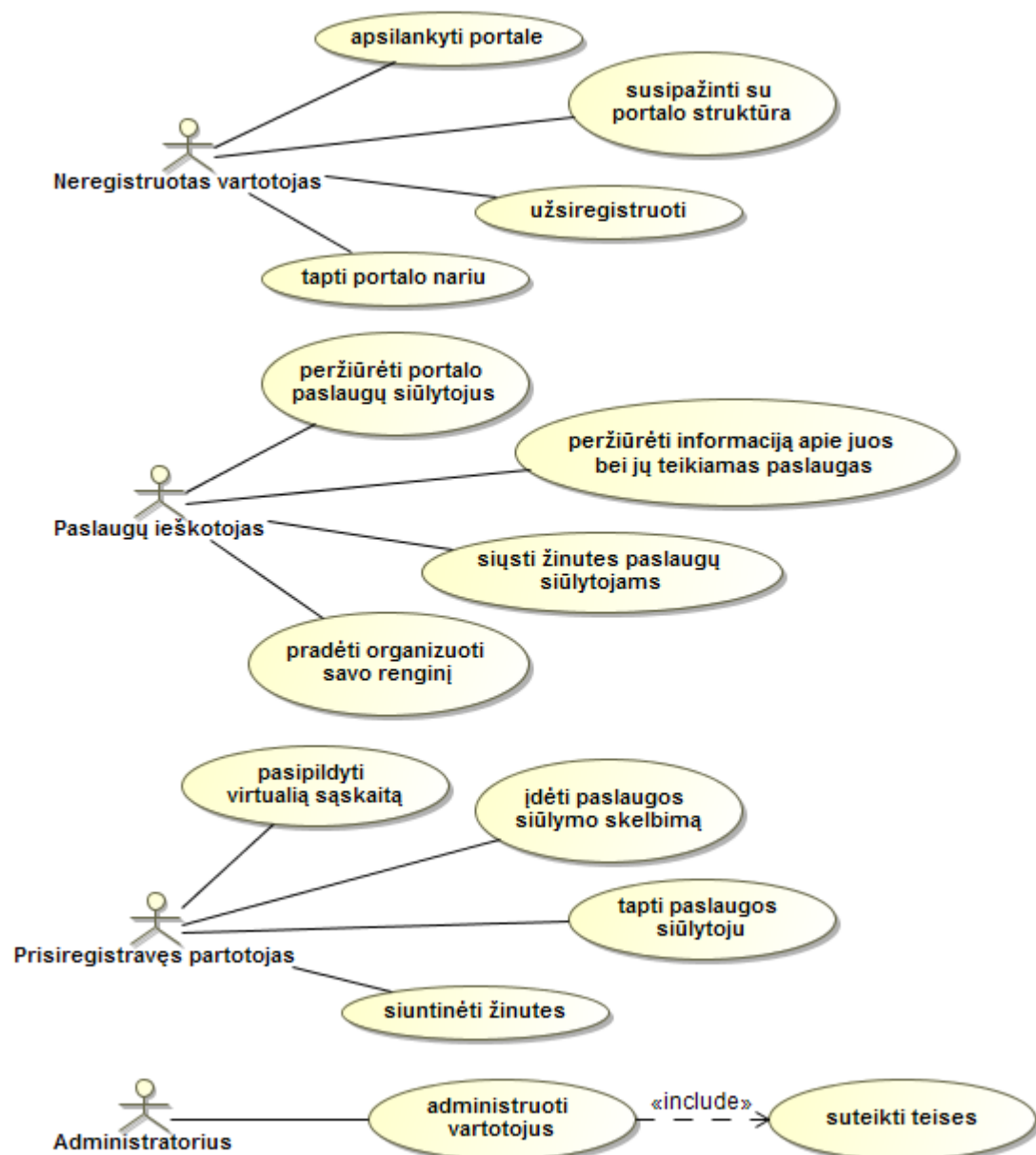
Aktorius	Panaudojimo atvejis
Operatorius	matyti laivus; kontroliuoti kursą: nukreipti į šoną, patikslinti kursą; perspėti apie pavojus (pranešti apie ledkalnius, pranešti apie pavojingai artėjančius laivus, perspėti apie artėjančią pasienio ribą)

5.1.3. Planuojami gauti PA modeliai

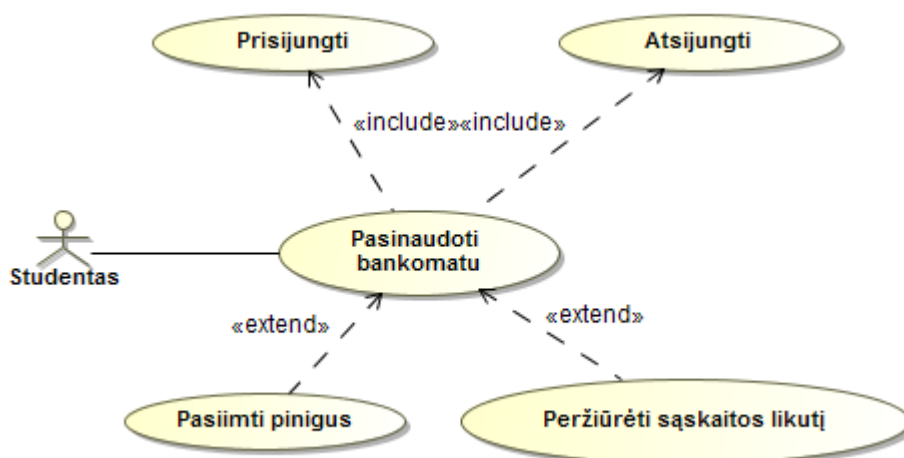
Atliekant eksperimentą tiek su griežtos struktūros tekstu, tiek su laisvos struktūros tekstu planuojama gauti vienodus ir 100 procentų tikslus bei išsamius modelius. Planuojami gauti panaudojimo atvejų modeliai pateikti paveikslėliuose žemiau (5.1 – 5.10 pav.)



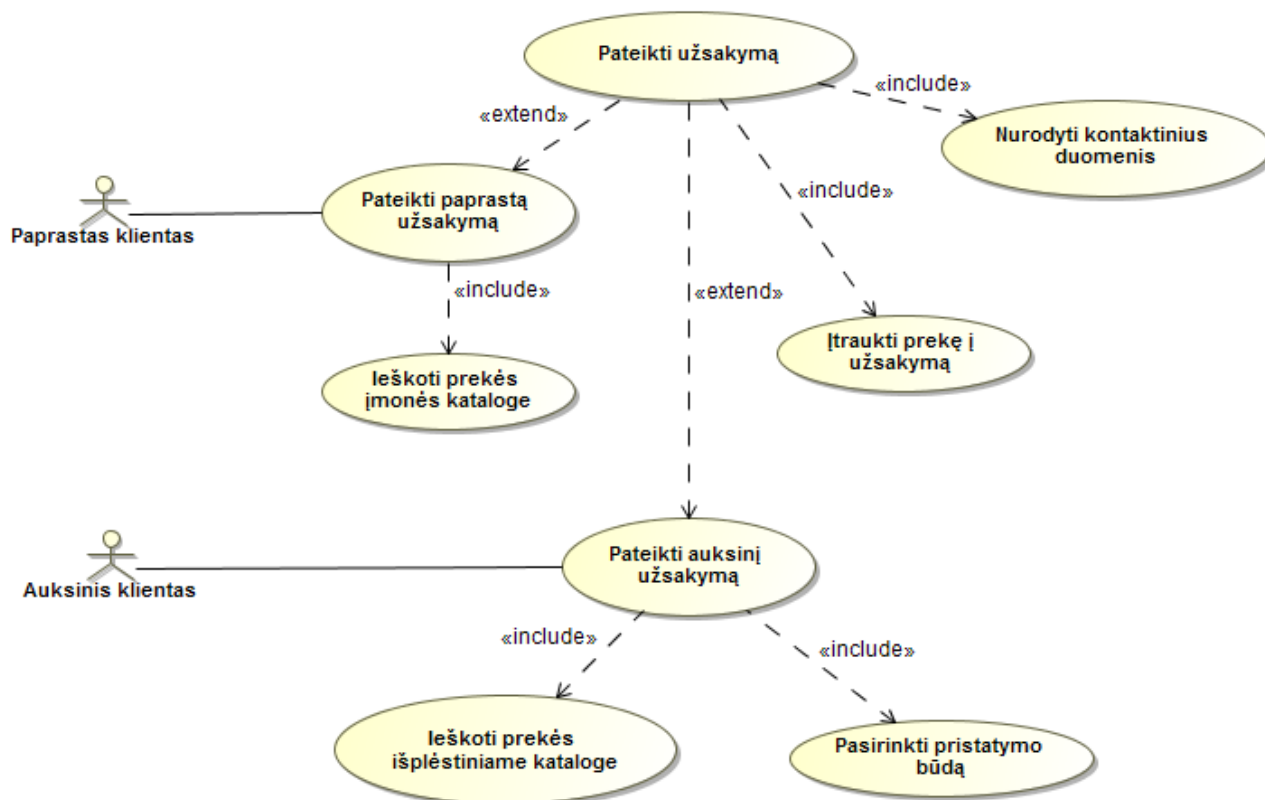
5.1 pav. Pirmojo bandymo planuojama panaudojimo atvejų diagrama



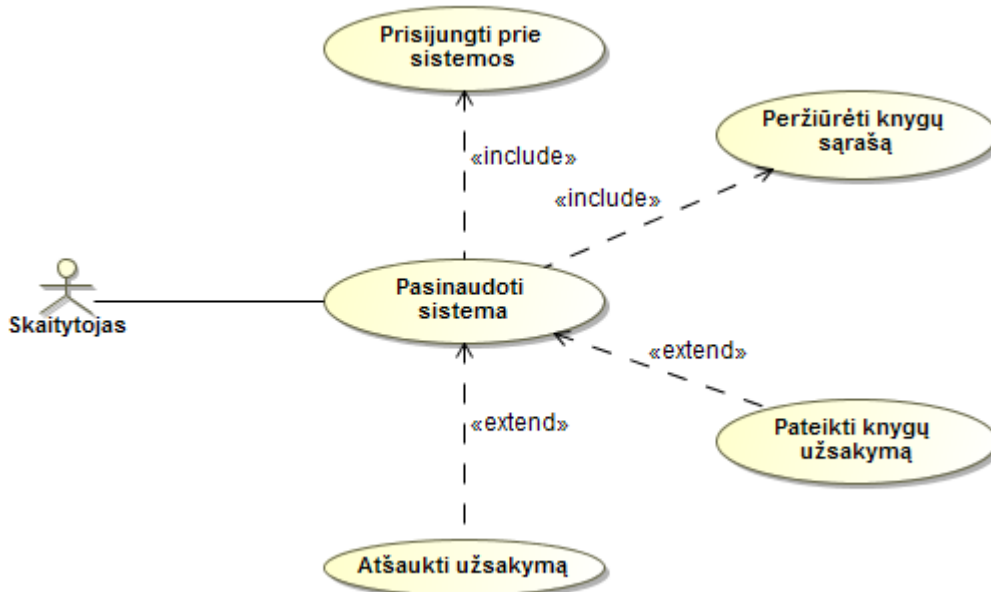
5.2 pav. Antrojo bandymo planuojama panaudojimo atvejų diagrama



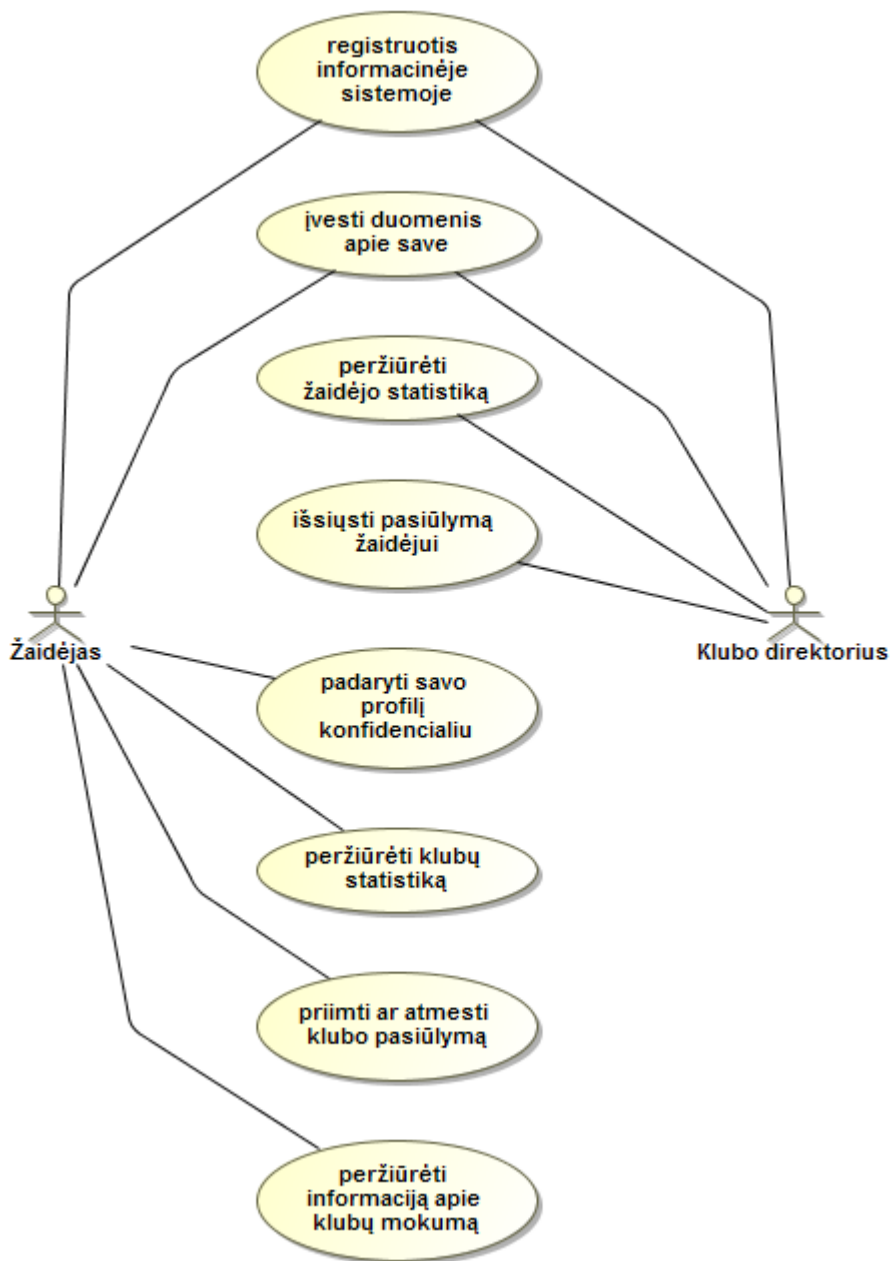
5.3 pav. Trečiojo bandymo planuojama panaudojimo atvejų diagrama



5.4 pav. Ketvirtojo bandymo planuojama panaudojimo atvejų diagrama



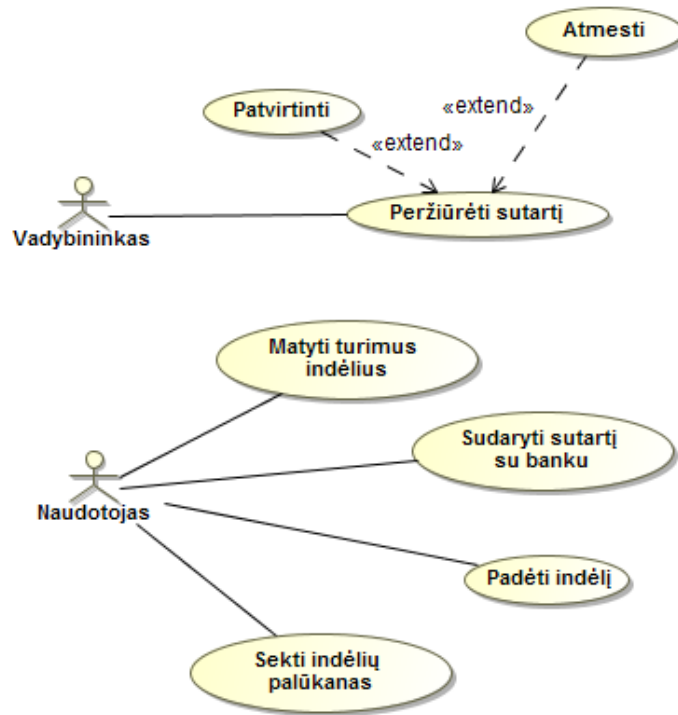
5.5 pav. Penktojo bandymo planuojama panaudojimo atvejų diagrama



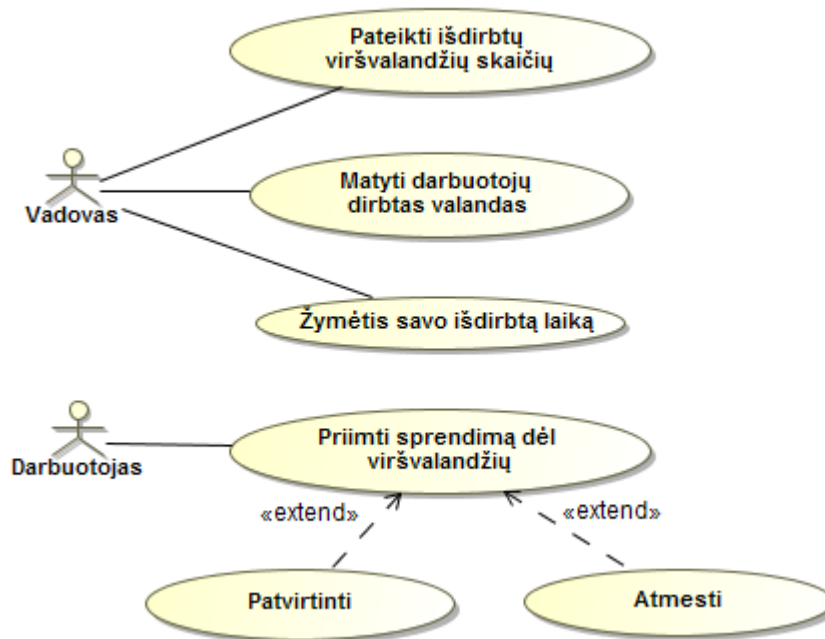
5.6 pav. Šeštojo bandymo planuojama panaudojimo atvejų diagrama



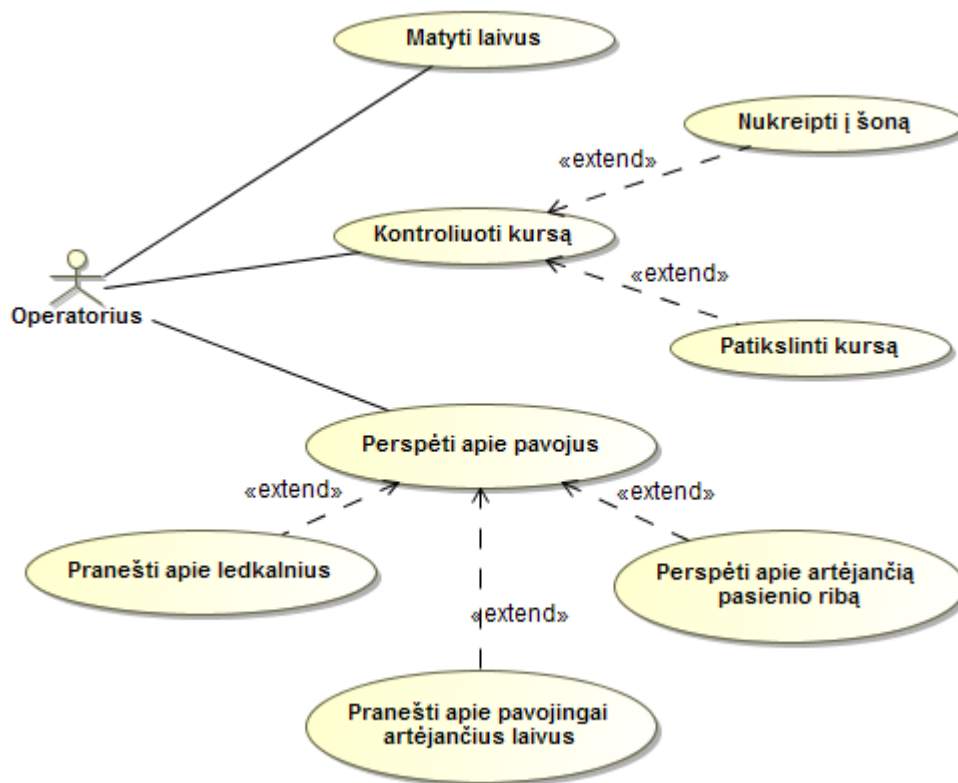
5.7 pav. Septintojo bandymo planuojama panaudojimo atvejų diagrama



5.8 pav. Aštuntojo bandymo planuojama panaudojimo atvejų diagrama



5.9 pav. Devintojo bandymo planuojama panaudojimo atvejų diagrama



5.10 pav. Dešimtojo bandymo planuojama panaudojimo atvejų diagrama

5.2. Eksperimento rezultatai

Eksperimentas vertinamas tikslumo P (angl. *Precision*) ir išsamumo R (angl. *Recal*) kriterijais. Žemiau patiktos lentelės, vaizduojančios eksperimento rezultatus.

5.2.1. Eksperimento su griežtos struktūros reikalavimais rezultatai

Tikslumo kriterijumi pagrįsto vertinimo lentelė atskleidžia, jog visų transformuotų elementų skaičius atitinka teisingai transformuotų elementų skaičiui, kas leidžia daryti išvadą, jog įrankio preliminariems *UML* modeliams kurti iš teksto tikslumo kriterijus yra lygus 1. Tai reiškia, kad įrankis veikia tiksliai.

5.21 lentelė. Eksperimento su griežtos struktūros reikalavimais vertinimas tikslumo bei išsamumo kriterijais

Bandymo numeris i	Teisingai transformuotų elementų skaičius TE_i	Visų transformuotų elementų skaičius GE_i	Visų elementų, kurie turėjo būti transformuoti skaičius VE_i	Bandymo tikslumas P_i	Bandymo išsamumas R_i
1	23	23	23	1	1
2	32	32	32	1	1
3	11	11	11	1	1
4	19	19	19	1	1
5	11	11	11	1	1
6	20	20	20	1	1
7	9	9	9	1	1
8	16	16	16	1	1
9	14	14	14	1	1

10	17	17	17	1	1
----	----	----	----	---	---

Išsamumo kriterijumi pagrįsto vertinimo lentelė atskleidžia, jog visų teisingai transformuotų elementų skaičius atitinka visų elementų, kurie turėjo būti transformuoti skaičiui, kas leidžia daryti išvadą, jog įrankio preliminariems *UML* modeliams kurti iš teksto išsamumo kriterijus yra lygus 1. Tai reiškia, kad įrankiu sukurti preliminarūs *UML* modeliai yra išsamūs.

5.2.2. Eksperimento su laisvos struktūros reikalavimais rezultatai

Tikslumo kriterijumi pagrįsto vertinimo lentelė atskleidžia, jog laisvos struktūros reikalavimuose visų transformuotų elementų skaičius mažesnis už teisingai transformuotų elementų skaičių. Netikslūs buvo visi panaudojimo atvejai, kurie buvo suformuluoti laisvu tekstu. Ryšiai bei aktoriai nuo to nenukentėjo. Įrankio preliminariems *UML* modeliams kurti iš teksto tikslumo kriterijus yra tarp 0,5 ir 1. Tai reiškia, kad įrankio veikimo tikslumas ne mažesnis kaip 50 procentų.

5.23 lentelė. Eksperimento su laisvos struktūros reikalavimais vertinimas tikslumo ir išsamumo kriterijais

Bandymo numeris i	Teisingai transformuotų elementų skaičius TE_i	Visų elementų, kurie turėjo būti transformuoti skaičius VE_i	Bandymo išsamumas R_i	Visų transformuotų elementų skaičius GE_i	Bandymo tikslumas P_i
1	20	23	0,87	23	0,87
2	28	32	0,88	32	0,88
3	7	11	0,64	11	0,64
4	15	19	0,79	19	0,79
5	7	11	0,64	11	0,64
6	16	20	0,8	20	0,8
7	5	9	0,56	9	0,56
8	9	16	0,56	16	0,56
9	8	14	0,57	14	0,57
10	13	17	0,76	17	0,76

Išsamumo kriterijumi pagrįsto vertinimo lentelė atskleidžia, jog visų teisingai transformuotų elementų skaičius yra mažesnis už visų elementų, kurie turėjo būti transformuoti skaičių, kas leidžia daryti išvadą, jog įrankio preliminariems *UML* modeliams kurti iš teksto išsamumo kriterijus yra tarp 0,5 ir 1. Neišsamūs buvo visi panaudojimo atvejai, kurie buvo suformuluoti laisvu tekstu. Ryšiai bei aktoriai nuo to nenukentėjo. Tai reiškia, kad įrankiu sukurti preliminarūs *UML* modeliai yra ne mažiau kaip 50 procentų išsamūs.

5.3. Sprendimo veikimo ir savybių analizė, kokybės kriterijų įvertinimas, taikymo rekomendacijos

Įrankio preliminariems *UML* modeliams kurti iš teksto veikimas tiesiogiai proporcingas analitiko, dirbančio šiuo įrankiu kvalifikacijai ir patirčiai. Išlieka žmogiškasis faktorius, jog analitikas gali suklysti ir savaip interpretuoti pateiktus natūralia kalba suformuluotus reikalavimus. Nuo šio faktoriaus apsaugoti negali joks įrankis. Taip pat modelių kūrimas priklauso nuo to, kaip analitikas interpretuos gautą tekstą. Būna reikalavimų, kurie suformuluojami dviprasmiškai, taigi ir modelis gali turėti kelis variantus. Analitikai gali nepastebėti defektų natūralia kalba pateiktuose reikalavimuose, kas sukelia dviprasmybes. [4]

Dirbant su griežtos struktūros tekstu, sukurti modeliai tikslūs ir išsamūs, tačiau braižant tokius pat modelius ir laisvos struktūros teksto tikslumas ir išsamumas nukentčia ne daugiau kaip 50 procentų.

Atsižvelgiant į aukščiau pateiktas tikslumo ir išsamumo kriterijų lenteles, galima teigti, jog įrankis dalinai atitinka jam užduotus kriterijus.

Įrankio preliminariniams *UML* modeliams kurti iš teksto esmė yra dalinai automatizuoti, palengvinti ir pagreitinti modelių sudarymo procesą. Taip pat, šio proceso dalinis automatizavimas sumažina galimų klaidų tikimybę.

Šį įrankį siūloma naudoti analitikams, projektuotojams bei studentams, besimokantiems Informacinių sistemų inžinerijos ar panašiose studijų programose. Įrankį patogiausia naudoti sudarant nedidelius bei vidutinio dydžio pradinius panaudojimo atvejų modelius pagal natūralia kalba pateiktus rišliai suformuluotus pradinius reikalavimus.

Norint pasiekti maksimalų įrankio efektyvumą, jį reikėtų naudoti analizuojant griežtos struktūros tekstą.

Įrankio eksportuotus rezultatus galima importuoti tik į *MagicDraw* įrankį, nes eksportuoto modelio *XMI* formatu failo sintaksė pritaikyta tik šiam įrankiui. Įkelti eksportuotą modelio *XMI* formatu failą į kitą *CASE* įrankį gali nepavykti arba rezultatai gali būti stipriai iškreipti ir neatitikti realybės.

6. REZULTATŲ APIBENDRINIMAS IR IŠVADOS

1. Kadangi ryškėja vis ankstesnių programinės įrangos kūrimo etapų automatizavimo poreikiai, buvo išanalizuotos Lietuvoje bei kitose pasaulio šalyse naudojamos modelių kūrimo iš teksto metodikos bei įrankiai (*UMLG, KPSpotter ir kt.*);
2. Atlikta modelių kūrimo iš teksto metodų bei įrankių analizė parodė, kad pagrindinis įrankių trūkumas yra tas, kad pilnai automatizuoto modelių kūrimo iš teksto neturi nei vienas įrankis, todėl buvo nuspręsta iširti tokio kūrimo galimybes;
3. Vartotojų aibės bei jų poreikių analizė parodė, kad pagrindiniai šio sprendimo vartotojai būtų įmonėse dirbantys projektuotojai bei analitikai, taip pat studentai, studijuojantys Informacinių sistemų inžineriją ar kitas panašios krypties specialybes;
4. Vartotojų poreikių ir jų įgyvendinimo galimybių analizė parodė, kad tikslinga būtų pradėti nuo pradinių panaudojimo atvejų modelių sudarymo pradiniam projektavimo etapui, šiam tikslui įgyvendinant teksto analizavimo, pažymint žodžius bei jų junginius ir priskiriant jiems modelių elementus.
5. Realizuoto įrankio testavimas parodė, kad toks sprendimas įgyvendinamas ir veikia sėkmingai, be klaidų;
6. Atlikus eksperimentą, nustatyta, kad įrankis preliminariems *UML* modeliams kurti iš teksto atitinka jam iškeltus kokybinius tikslumo ir išsamumo kriterijus, jeigu dirbama su griežtos struktūros reikalavimų tekstais. Analizuojant laisvos struktūros reikalavimus, gauto modelio tikslumas ir išsamumas yra ne mažesnis kaip 50 procentų.

7. LITERATŪRA

- [1] Visual Paradigm available: <http://www.visual-paradigm.com/product/vpUML/tutorials/textualanalisys.jsp> [kreiptasi 2014-06-15]
- [2] Imran Sarwar Bajwa, M. Abbas Choudhary. *Natural language processing based automated system for UML diagrams generation*. Pakistan
- [3] Min Song, Xiaohua Hu. *Designing and Developing an Automatic Interactive Keyphrase Extraction System with Unified Modeling Language (UML)*. Drexel University, Philadelphia, USA. Available: <http://onlinelibrary.wiley.com/doi/10.1002/meet.1450410143/full> [kreiptasi 2014-06-15]
- [4] Nik Boyd, *Using Natural Language in Software Development*, 1999, available: <http://www.educery.com/papers/rhetoric/road/> [kreiptasi 2014-03-24]
- [5] Steve Burbeck, *Applications Programming in Smalltalk-80(TM): How to use Model-View-Controller (MVC)*, 1992, available: <http://st-www.cs.illinois.edu/users/smarch/st-docs/mvc.html> [kreiptasi 2015-01-29]
- [6] Vaitkūnas, M. *PHP karkasų architektūrinis tyrimas*. Kaunas, Kauno technologijos universitetas, 2013. Available: http://vddb.library.lt/fedora/get/LT-eLABa-0001:E.02~2013~D_20130826_102846-07165/DS.005.0.01.ETD [kreiptasi 2014-06-15]
- [7] Vincenzo Ambriola, Vincenzo Gervasi. *Processing Natural Language Requirements*. 1997. Available: <http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=B025F7DB4065BD5C3D837D595879B352?doi=10.1.1.28.8142&rep=rep1&type=pdf> [kreiptasi 2015-03-17]
- [8] Michael A. Jackson, *Problems and requirements [software development]*, 1995. Available: http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=512540&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxppls%2Fabs_all.jsp%3Farnumber%3D512540 [kreiptasi 2015-03-17]
- [9] Bashar Nuseibeh, Steve Easterbrook, *Requirements Engineering: A Roadmap*, 2000. Available: <http://mcs.open.ac.uk/ban25/papers/sotar.re.pdf> [kreiptasi 2015-04-20]
- [10] Min Song, Il-Yeol Song, Xiaohua Hu, *KSPotter: A Flexible Information Gain-based Keyphrase Extraction System*. 2003. Available: http://www.researchgate.net/publication/220759161_KSPotter_a_flexible_information_gain_based_keyphrase_extraction_system [kreiptasi 2014-06-20]
- [11] Deva Kumar Deeptimahanti, Ratna Sanyal, *Semi-automatic Generation of UML Models from Natural Language Requirements*. 2011. Available: <http://ulir.ul.ie/bitstream/handle/10344/2348/Deeptimahanti.pdf?sequence=2> [kreiptasi 2015-03-17]
- [12] Deva Kumar Deeptimahanti, Muhammad Ali Babar, *An Automated Tool for Generating UML Models from Natural Language Requirements*. 2009. Available: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5431706> [kreiptasi 2014-03-17]

8. PRIEDAI

8.1. priedas. Testavimo duomenys

1 testavimo atvejis:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en"
lang="en">
<head profile="http://selenium-ide.openqa.org/profiles/test-
CASE">
<meta http-equiv="Content-Type" content="text/html;
charset=UTF-8" />
<link rel="selenium.base"
href="http://diakuzn.stud.if.ktu.lt/" />
<title>New Test</title>
</head>
<body>
<table cellpadding="1" cellspacing="1" border="1">
<thead>
<tr><td rowspan="1" colspan="3">New Test</td></tr>
</thead><tbody>
<tr>
<td>open</td>
<td>/basic/web/index.php</td>
<td></td>
</tr>
<tr>
<td>clickAndWait</td>
<td>link=Prisijungimas</td>
<td></td>
</tr>
<tr>
<td>type</td>
<td>id=loginform-username</td>
<td>diakuzn@ktu.lt</td>
</tr>
<tr>
<td>type</td>
<td>id=loginform-username</td>
<td>diakuzn@ktu.lt</td>
</tr>
<tr>
<td>type</td>
<td>id=loginform-password</td>
<td>testtest</td>
</tr>
<tr>
<td>type</td>
<td>id=loginform-password</td>
<td>testtest</td>
</tr>
```

```

</tr>
<tr>
  <td>click</td>
  <td>name=login-button</td>
  <td></td>
</tr>
<tr>
  <td>click</td>
  <td>name=login-button</td>
  <td></td>
</tr>
<tr>
  <td>click</td>
  <td>name=login-button</td>
  <td></td>
</tr>
<tr>
  <td>clickAndWait</td>
  <td>name=login-button</td>
  <td></td>
</tr>

</tbody></table>
</body>
</html>

```

2 testavimo atvejais:

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en"
lang="en">
  <head profile="http://selenium-ide.openqa.org/profiles/test-
CASE">
    <meta http-equiv="Content-Type" content="text/html;
charset=UTF-8" />
    <link rel="selenium.base"
href="http://diakuzn.stud.if.ktu.lt/" />
    <title>New Test</title>
  </head>
  <body>
    <table cellpadding="1" cellspacing="1" border="1">
      <thead>
        <tr><td rowspan="1" colspan="3">New Test</td></tr>
      </thead><tbody>
        <tr>
          <td>open</td>
          <td>/basic/web/index.php</td>
          <td></td>
        </tr>
        <tr>
          <td>click</td>
          <td>xpath=(//a[contains(text(),'Programa')])[2]</td>

```

```

        <td></td>
</tr>
<tr>
    <td>clickAndWait</td>
    <td>xpath=(//a[contains(text(),'Programa')])[2]</td>
    <td></td>
</tr>
<tr>
    <td>click</td>
    <td>link=Pridėti naują įrašą</td>
    <td></td>
</tr>
<tr>
    <td>clickAndWait</td>
    <td>link=Pridėti naują įrašą</td>
    <td></td>
</tr>
<tr>
    <td>type</td>
    <td>id=textarea-textarea</td>
    <td>If you would like YCGA to prepare samples for
sequencing, please see appropriate sample submission requirements
below.</td>
</tr>
<tr>
    <td>type</td>
    <td>id=textarea-textarea</td>
    <td>If you would like YCGA to prepare samples for
sequencing, please see appropriate sample submission requirements
below.</td>
</tr>
<tr>
    <td>click</td>
    <td>name=program-button</td>
    <td></td>
</tr>
<tr>
    <td>click</td>
    <td>name=program-button</td>
    <td></td>
</tr>
<tr>
    <td>click</td>
    <td>name=program-button</td>
    <td></td>
</tr>
<tr>
    <td>clickAndWait</td>
    <td>name=program-button</td>
    <td></td>
</tr>
<tr>
    <td>click</td>
    <td>//th/div</td>

```



```

        <td></td>
</tr>
<tr>
    <td>click</td>
    <td>//th[2]/div</td>
    <td></td>
</tr>

</tbody></table>
</body>
</html>

```

3 testavimo atvejais:

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en"
lang="en">
    <head profile="http://selenium-ide.openqa.org/profiles/test-
CASE">
        <meta http-equiv="Content-Type" content="text/html;
charset=UTF-8" />
        <link rel="selenium.base"
href="http://diakuzn.stud.if.ktu.lt/" />
        <title>New Test</title>
    </head>
    <body>
        <table cellpadding="1" cellspacing="1" border="1">
            <thead>
                <tr><td rowspan="1" colspan="3">New Test</td></tr>
            </thead><tbody>
                <tr>
                    <td>open</td>
                    <td>/basic/web/index.php?r=site%2Fprogram&amp;programId=31
</td>
                    <td></td>
                </tr>
                <tr>
                    <td>click</td>
                    <td>link=Atsijungti (diakuzn@ktu.lt)</td>
                    <td></td>
                </tr>
                <tr>
                    <td>clickAndWait</td>
                    <td>link=Atsijungti (diakuzn@ktu.lt)</td>
                    <td></td>
                </tr>
            </tbody></table>
        </body>
    </html>

```

8.2. priedas. Gauto modelio XMI formatu failo ištrauka.

```
<packagedElement XMI:type='UML:Actor' XMI:id='239-51-1'
name='Neregistruotas vartotojas' /><packagedElement
XMI:type='UML:Actor' XMI:id='240-51-1' name='Paslaugu
ieškotojas' /><packagedElement XMI:type='UML:Actor' XMI:id='241-51-1'
name='Prisiregistravęs vartotojas' /><packagedElement
XMI:type='UML:Actor' XMI:id='242-51-1'
name='Administratorius' /><packagedElement XMI:type='UML:UseCASE'
XMI:id='243-51-2' name='apsilankyti
portale' /></packagedElement><packagedElement XMI:type='UML:UseCASE'
XMI:id='244-51-2' name='susipažinti su
portalo
struktūra' /></packagedElement><packagedElement
XMI:type='UML:UseCASE' XMI:id='245-51-2'
name='užsiregistruoti' /></packagedElement><packagedElement
XMI:type='UML:UseCASE' XMI:id='246-51-2' name='tapti portalo
nariu' /></packagedElement><packagedElement XMI:type='UML:UseCASE'
XMI:id='247-51-2' name='peržiūrėti portalo
paslaugu
siūlytojus' /></packagedElement><packagedElement
XMI:type='UML:UseCASE' XMI:id='248-51-2' name='peržiūrėti
informacija apie juos bei jų teikiamas
paslaugas' /></packagedElement><packagedElement
XMI:type='UML:UseCASE' XMI:id='249-51-2' name='siūsti žinutes
paslaugu siūlytojams' /></packagedElement><packagedElement
XMI:type='UML:UseCASE' XMI:id='250-51-2' name='pradėti organizuoti
savo renginį' /></packagedElement><packagedElement
XMI:type='UML:UseCASE' XMI:id='251-51-2' name='pasipildyti virtualia
sąskaita' /></packagedElement><packagedElement XMI:type='UML:UseCASE'
XMI:id='252-51-2' name='įdėti paslaugos siūlymo
skelbimą' /></packagedElement><packagedElement XMI:type='UML:UseCASE'
XMI:id='253-51-2' name='tapti paslaugos
siūlytoju' /></packagedElement><packagedElement
XMI:type='UML:UseCASE' XMI:id='254-51-2' name='siuntinėti
žinutes' /></packagedElement><packagedElement XMI:type='UML:UseCASE'
XMI:id='255-51-2' name='administruoti vartotojus' /><include
XMI:type='UML:Include' XMI:id='77-51-255-3-include_ID'
visibility='public' addition='77-51-255-
3' /></packagedElement><packagedElement XMI:type='UML:Association'
XMI:id='32-51-243-239'>
    <memberEnd XMI:idref='32-51-243-239-pirmas' />
    <memberEnd XMI:idref='32-51-243-239-antras' />
    <navigableOwnedEnd XMI:idref='32-51-243-239-
pirmas' />
    <navigableOwnedEnd XMI:idref='32-51-243-239-
antras' />
    <ownedEnd XMI:type='UML:Property' XMI:id='32-
51-243-239-pirmas' visibility='private' type='239-51-1'
association='32-51-243-239' />
    <ownedEnd XMI:type='UML:Property'
XMI:id='32-51-243-239-antras' visibility='private' type='243-51-2'
association='32-51-243-239' />
</packagedElement><packagedElement
XMI:type='UML:Association' XMI:id='33-51-244-239'>
    <memberEnd XMI:idref='33-51-244-239-pirmas' />
```

```

        <memberEnd XMI:idref='33-51-244-239-antras' />
        <navigableOwnedEnd XMI:idref='33-51-244-239-
pirmas' />
        <navigableOwnedEnd XMI:idref='33-51-244-239-
antras' />
        <ownedEnd XMI:type='UML:Property' XMI:id='33-
51-244-239-pirmas' visibility='private' type='239-51-1'
association='33-51-244-239' />
        <ownedEnd XMI:type='UML:Property'
XMI:id='33-51-244-239-antras' visibility='private' type='244-51-2'
association='33-51-244-239' />
        </packagedElement><packagedElement
XMI:type='UML:Association' XMI:id='34-51-245-239'>
        <memberEnd XMI:idref='34-51-245-239-pirmas' />
        <memberEnd XMI:idref='34-51-245-239-antras' />
        <navigableOwnedEnd XMI:idref='34-51-245-239-
pirmas' />
        <navigableOwnedEnd XMI:idref='34-51-245-239-
antras' />
        <ownedEnd XMI:type='UML:Property' XMI:id='34-
51-245-239-pirmas' visibility='private' type='239-51-1'
association='34-51-245-239' />
        <ownedEnd XMI:type='UML:Property'
XMI:id='34-51-245-239-antras' visibility='private' type='245-51-2'
association='34-51-245-239' />
        </packagedElement><packagedElement
XMI:type='UML:Association' XMI:id='35-51-246-239'>

```