

**KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIOS FAKULTETAS
INFORMACIJOS SISTEMŲ KATEDRA**

Vilius Lukošius

**VEIKLOS TAISYKLĖMIS PAGRĮSTAS IS
PROJEKTAVIMO PROCESO MODELIS**

Magistro darbas

**Vadovas
doc. dr. R. Butleris**

KAUNAS, 2005

**KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIOS FAKULTETAS
INFORMACIJOS SISTEMŲ KATEDRA**

**TVIRTINU
Katedros vedėjas
doc. R. Butleris
2005-05-23**

**VEIKLOS TAISYKLĖMIS PAGRĪSTAS IS
PROJEKTAVIMO PROCESO MODELIS**

Informatikos magistro baigiamasis darbas

**Kalbos konsultantė
Lietuvių k. katedros lekt.
dr. J. Mikelionienė
2005-05-23**

**Recenzentas
doc. dr. E. Karčiauskas
2005-05-23**

**Vadovas
doc. dr. R. Butleris
2005-05-23**

**Atliko
IFM 9/4 gr. stud.
V. Lukošius
2005-05-23**

KAUNAS, 2005

Integrating Business Rules into Software Development Process Model

Summary

Today's business's more and more depend on information systems to give them edge on competition and keep double digit growth rates needed to satisfy shareholders. And information systems, serving their needs, have to keep up with changing business requirements. Business Rules Approach [10] seems to be the answer to their demands, but not many service and solution providers can satisfy their needs, because there is no standard based software engineering process to efficiently provide solution to customers.

This works provides methodology of integrating business rules approach into existing and time proven software development process as well as providing needed infrastructure to creating solution specification and integrating externalized business rules with created system specification.

This work concentrates on tasks:

1. Creating business rules management model;
2. Selecting software development process, adaptable to business rules approach;
3. Integrating business rules management model into selected software development process;
4. Relating externalized business rules with specification entities;
5. Providing comparison of unmodified software process with provided proposal.

Turinys

IVADAS	6
Tyrimo Objektas ir problema.....	6
Tikslas ir uždaviniai	7
1. VEIKLOS TAISYKLIŲ VALDYMO MODELIO PARINKIMAS.....	8
1.1. Veiklos taisyklės apibrėžimas	8
1.2. Veiklos taisyklių kategorizavimo modeliai.....	9
1.3. Veiklos taisyklių klasifikacija pagal pritaikymą	10
1.3.1. Orientuotos į duomenų bazę.....	11
1.3.2. Orientuotos į taikomąją programą.....	11
1.4. Veiklos taisyklių modelio parinkimas.....	13
2. PROJEKTAVIMO METODIKOS PARINKIMAS.....	16
2.1. Agile požiūris: neformalios, iteratyvios projektavimo metodikos.....	17
2.2. SEI CMM, SEI CMMI, ISO/IEC standartais įvertinamos metodikos	18
2.2.1. SEI CMM - proceso kokybės įvertinimo metodas	18
2.2.2. SEI CMMI - modernizuotas proceso kokybės įvertinimo metodas.....	19
2.3. RUP IS kūrimo procesas	19
2.4. MSF projektavimo metodika.....	20
2.5. OSS – atviro kodo licenzija paremtu IS kūrimo procesas.....	21
2.6. HALLE – veiklos taisyklėmis paremta IS kūrimo metodika	22
2.7. Projektavimo metodikų palyginimas.....	23
3. VEIKLOS TAISYKLĖMIS PAREMTAS IS KŪRIMO PROCESAS.....	26
3.1. Veiklos taisyklių integracija į MSF proceso modelį	26
3.2. VT požiūrio įtaka MSF rolėms.....	28
3.3. Detalizuotas veiklos taisyklių klasifikacijos modelis.....	29
3.4. Detalizuotas veiklos taisyklių formalizavimo modelis	31
3.5. VT įtaka vizijos formavimo fazei.....	32
3.6. Veiklos taisyklių įtaka IS konceptualiosios architektūros specifikacijai	36
3.7. Veiklos taisyklių įtaka IS loginės architektūros specifikacijai.....	39
3.8. Veiklos taisyklių įtaka IS fizinės architektūros specifikacijai	41
3.9. Veiklos taisyklių valdymo modeliai.....	42
3.10. Projektavimo aplinkos pritaikymas veiklos taisyklėms	44

4.	EKSPERIMENTINĖ DALIS	49
4.1.	Tikslas	49
4.2.	Realizacija	49
4.3.	Teikiamos naudos, trūkumai	53
5.	IŠVADOS.....	54
	LITERATŪRA	55
	TERMINŲ IR SUTRUMPINIMŲ ŽODYNAS	57
	PRIEDAI.....	Error! Bookmark not defined.
	PRIEDAS A: MSF vizijos/apimties dokumentas	Error! Bookmark not defined.
	PRIEDAS B: MSF loginis panaudojimo atvejų modelis.....	Error! Bookmark not defined.
	PRIEDAS C: Detalizuoti MSF panaudojimo atvejų aprašai	Error! Bookmark not defined.
	PRIEDAS D: Detalizuoti siūlomo metodo panaudojimo atvejų aprašai	Error! Bookmark not defined.

defined.

Lentelių sąrašas

1.1 lentelė Veiklos taisyklių struktūrizavimo modeliai.....	9
1.2 lentelė VT tipai pagal jų pritaikymą.....	13
1.3 lentelė Pateiktos ir išanalizuotų veiklos taisyklių metodikų sąsaja.....	15
2.1 lentelė Projektavimo metodikų palyginimas.....	23
3.1 lentelė MSf proceso fazių aprašas.....	27
3.2 lentelė Modifikuotų MSF rolių suderinamumo matrica.....	29
3.3 lentelė Detalizuotų veiklos taisyklių tipų paaiškinimai.....	30
3.4 lentelė Veiklos taisyklių formalizavimo šablonai pagal tipus.....	31
3.5 lentelė Veiklos įvykio aprašas.....	35
3.6 lentelė Veiklos panaudojimo atvejų ir sprendimų sąsaja.....	38
3.7 lentelė Veiklos panaudojimo atvejų ir sprendimų sąsaja.....	40
3.8 lentelė VT valdymo modelių palyginimas.....	43

Paveikslų sąrašas

1.1 pav. Aukštesnio lygio veiklos taisyklių klasifikacijos schema	14
2.1 pav. Projektavimo metodikų palyginimo koordinačių plokštuma.	16
2.2 pav. Agile požiūris projektavimo metodikų vertinimo plokštumoje.....	17
2.3 pav. CMM ir CMMI projektavimo metodikų brandos vertinimo metodikos.	19
2.4 pav. RUP leidžia pasiekti lankstumą priklausomai nuo projekto dydžio ir vartotojo poreikių.	20
2.5 pav. MSF pozicija projektavimo metodikų įvertinimo skalėje.	21
2.6 pav. OSS – neformavus ir lankstus kūrimo procesas.	22
2.7 pav. HALLE – veiklos taisyklėmis paremta IS kūrimo metodika.	23
3.1 pav. MSF ir HALLE proceso modelių sąsajos.....	26
3.2 pav. Detalizuotas „veiksmo teiginys“	30
3.3 pav. Veiklos taisyklių išreiškimo tipai	31
3.4 pav. VT įtaka pasiruošimo fazės veiklos panaudojimo atvejų sudarymo metodikai	33
3.5 pav. VT įtakotas pasiruošimo fazės veiklos panaudojimo atvejų sudarymo procesas.....	34
3.6 pav. Veiklos įvykių susiejimo su veiklos panaudojimo atvejais konceptuali diagrama	35
3.7 pav. Siūlomas konceptualios specifikacijos sudarymo proceso modelis	37
3.8 pav. Siūlomas konceptualios architektūros sudarymo proceso modelis	38
3.9 pav. Siūlomas loginės specifikacijos sudarymo proceso modelis.....	39
3.10 pav. Pasiruošimo ir planavimo fazėse atliekamų veiksmų nauda ir reikšmė VT modelio sudarymui.	40
3.11 pav. Veiklos taisyklės sąsajos su specifikacijos artefaktu adaptavimo pavyzdys Nr. 1.....	41
3.12 pav. Veiklos taisyklės sąsajos su specifikacijos artefaktu adaptavimo pavyzdys Nr. 2.....	42
3.13 pav. Eksperimentinė veiklos taisyklių valdymo IS	45
3.13 pav. Eksperimentinės VT valdymo IS duomenų modelis	45
3.14 pav. Panaudojimo atvejo išplėstinė reikšmė	47
3.15 pav. IS specifikacijos ir VT saugyklos integracijos schema	48
4.1 Sugeneruotas detalus MSF šablono panaudojimo atvejo aprašas	50
4.2 Sugeneruotas detalus teikiamo siūlymo panaudojimo atvejo aprašas.....	51
4.3 Detalus panaudojimo atvejo aprašas papildytas VT turiniu.....	52

IVADAS

Šiandienos sparčiai besivystantis verslas reikalauja lanksčių verslo valdymo sistemų, kurios sugebėtų vystyti sparta, atitinkančia veiklos pokyčius. Organizacija, kurdama specializuotą verslo valdymo sistemą, siekia įgyti konkurencinį pranašumą, sumažinti veiklos kaštus, pagerinti procesų valdymą [2]. Vienas iš esminių produkto naudingumo rodiklių galime laikyti investicijų grąžą (ROI – *return of investment*) [25]. Šis rodiklis dažniausiai nusako projekto sukūrimo išlaidų ir naudos, gautos įdiegus sukurtą produktą, santykį. Tačiau ne mažiau svarbu suprasti, kad nauja informacinė sistema (IS) savaime esamų valdymo problemų neišspręs. Reikalinga ir pačių procesų reinžinerija. Veiklos vadovai, prieš nusprenddami sukurti naują sistemą, privalo žinoti esmines savo veiklos taisykles (VT) [11]. Kartu subrangovai, turėdami pakankamai patirties esamoje dalykinėje srityje, turi padėti nuspręsti, kuriuos procesus galima ir reikia patobulinti. Kadangi proceso tobulinimas yra tęstinis procesas, kuris niekada nesustoja, tai nauja IS turi būti sukurta ne juodos dėžės principu, o su galimybe žinoti, kurios veiklos taisyklės, kuriame sistemos modulyje, mazge, procese atlieka sprendimus. Labai svarbus sistemos faktorius yra galimybė atlikti analizę ir gauti statistinius rezultatus įvairiais pjūviais. Tik žinant, kaip organizacijos procesai realizuojami ir realiai veikia, galima nuspręsti, kaip juos patobulinti.

Šioms problemoms spręsti buvo sukurtas iš esmės naujas požiūris į sistemos kūrimą – Veiklos taisyklių (VT) požiūris [10]. VT požiūris teigia, kad veiklos vadovas ar atstovas, priimančias strateginius sprendimus ir žinantis bendrą organizacijos veiklos koncepciją, turi pateikti ir turėti galimybę keisti išskirtas veiklos taisykles sukurtoje sistemoje. Esamu momentu kuriamos sistemos dažnai yra neadekvačiai dokumentuotos veiklos prasme. T. y. veiklos taisyklės nėra aiškiai ir vienareikšmiškai susietos su realizacija konkrečioje platformoje, kas apsunkina galimybę atsekti norimų pakeitimų įtaką bendram realizuotam procesui [11]. Taigi veikla, naudodama esamą sistemą gali netekti galimybės dinamiškai reaguoti į pasikeitusią aplinką ir būti priversta savo procesus taikyti prie esamos IS, vietoj to, kad esamą IS pritaikyti prie kintančių procesų.

Tyrimo Objektas ir problema

Tyrimo Objektas – pasirinkto veiklos taisyklių valdymo modelio integravimas į egzistuojančios projektavimo metodikos proceso modelį, bei infrastruktūros, būtinos sėkmingai pasinaudoti veiklos taisyklėmis teikiamais privalumais pateikimas. Pagrindinė sprendžiama problema – optimalaus VT

valdymo modelio sudarymas, sudaryto VT modelio integravimo į projektavimo metodikos proceso modelį pateikimas ir projektavimo įrankių pritaikymas pasikeitusiems reikalavimams.

Tikslas ir uždaviniai

Integruoti veiklos taisyklių požiūrį į egzistuojančios projektavimo metodikos proceso modelį, pritaikyti egzistuojančius projektavimo įrankius veiklos taisyklių integracijai su specifikacija bei pateikti palyginamąjį eksperimentinį pateikto metodo tyrimą.

Sprendžiami uždaviniai:

1. veiklos taisyklių modelio sudarymas;
2. projektavimo metodikos, tinkamos veiklos taisyklių požiūrio integracijai parinkimas;
3. veiklos taisyklių požiūrio integracija į pasirinktos projektavimo metodikos proceso modelį;
4. veiklos taisyklių susiejimo su kuriamos IS specifikacija metodikos pateikimas;
5. projektavimo įrankio pritaikymas kuriamos IS specifikacijos susiejimui su veiklos taisyklių saugykla;
6. standartinės proceso specifikacijos ir siūlomo proceso modelio specifikacijos palyginimas ir siūlymo naudų/trūkumų įvertinimas.

1. VEIKLOS TAISYKLIŲ VALDYMO MODELIO PARINKIMAS

1.1. Veiklos taisyklės apibrėžimas

Veiklos taisyklė yra teiginys, kuris apibrėžia ar apriboja tam tikrus veiklos aspektus. Veiklos taisykle nusakoma tam tikros veiklos struktūra, kontroliuojama arba įtakojama tam tikra veikla [9]. VT gali būti išreikšta tiesiog šnekamąja kalba, tačiau siekiant ją panaudoti kuriant IS, veiklos taisyklę reikia perversi į deklaratyvią formą. Dirbant prie didesnio produkto ir veiklos taisyklių skaičiui didėjant kyla poreikis veiklos taisykles kategorizuoti, verifikuoti jų atitikimą veiklos atstovo išreikštai formai. Kadangi veiklos taisyklės dalyvaus visuose trijuose architektūriniuose lygiuose (GVS, VL, DBVS), tai analizėje bus naudojama viena iš veiklos taisyklių kategorizavimo metodikų: GUIDE, ROSS, HALLE C. J Date, Versta Inc. ir USoft Inc.

Veiklos taisyklė yra nepriklausoma nuo projektavimo modelio ar techninės platformos. Taigi veiklos taisykle galima laikyti veiklos aplinkybę, kuri atitinka sekančius parametrus [5]:

- deklaratyvios;
- elementarios;
- išreikštos natūralia kalba;
- savarankiškos, nepriklausomos loginės struktūros;
- orientuotos į veiklą, ne į technologiją;
- priklausančios veiklai, ne technologijai.

Veiklos taisyklių integravimas į kuriamą IS teikia tris unikalias naudas[11]:

1. bendras organizacijos mastu veiklos taisyklių variklis;
2. objektinės orientacijos, informacijos inžinerijos ir taisyklių formalumo integracija;
3. veiklos taisyklių ir veiklos motyvacijos (strategijos, tikslų) koreliacija.

Svarbiausiu iš nurodytų aspektų galima laikyti veiklos taisyklių saugyklą. Tai yra taisyklių rinkinys, užtikrinantis duomenų ir vartotojo sąveiką ir sistemoje egzistuojantis kaip nepriklausomai veikiantis ir valdomas komponentas[11].

Yra keli galimi šio modelio realizacijos variantai. Tai ir komercinis veiklos taisyklių variklis, ar specializuotas modulis, skirtas taisyklėms valdyti. Tačiau siekiant veiklos taisykles panaudoti IS kūrime, pirmiausia yra reikalingas veiklos taisyklių valdymo modelis.

1.2. Veiklos taisyklių kategorizavimo modeliai

Nuo veiklos taisyklių požiūrio atsiradimo, įvairios mokslinės grupės ir komercinės organizacijos yra pateikę įvairių VT formalizavimo ir kategorizavimo metodikų (1.1 lentelė). Vienos metodikos sudarytos kaip mokslinių tyrimų bazė (GUIDE, ROSS, HALLE), kitos – komerciniuose produktuose naudojami VT valdymo modeliai (Versta Inc., USoft Inc.). 1.1 lentelėje pateikiami detaliausiai išnagrinėti ir labiausiai paplitę veiklos taisyklių struktūrizavimo modeliai.

1.1 lentelė

Veiklos taisyklių struktūrizavimo modeliai

Autorius	Klasifikacija
Business Rules Group - GUIDE (2000)	<p>Išvestis - sakinyss, išreiškiantis žinias, kurios savo ruožtu išvedamos iš kitų veiklos žinių [5]. Skirstoma:</p> <ul style="list-style-type: none"> • matematinis skaičiavimas; • loginė išvada. <p>Struktūrinis teiginys - apibrėžta mintis arba fakto konstatavimas, išreiškiantis tam tikrą organizacijos sandaros aspektą. Skirstoma:</p> <ul style="list-style-type: none"> • sąvoka; • faktas. <p>Veiksmo teiginys - yra apribojimą ar sąlygą reiškiantis sakinyss, kuris nustato arba valdo organizacijos veiksmus. Gali būti:</p> <ul style="list-style-type: none"> • sąlyga; • apribojimas; • leidimas; • įgaliotojas; • laikmatis; • vykdymo; • veiksmą valdantis; • veiksmą veikiantis.
Ross (2001)	<p>Pagrindinė šio metodo idėja – suskirstyti taisykles į standartinius, aukšto lygio taisyklių tipus, kurie nurodo, kokį patikrinimą turi atlikti taisyklės. Taisyklės gali būti nedalomos ir išvestinės. Nedalomas taisykles galima suskirstyti į 32 tipus, kurie papildomai grupuojami į 7 tipų grupes [19, 20]:</p> <ul style="list-style-type: none"> • egzempliorių patvirtinimo; • tipo patvirtinimo; • pozicijos patvirtinimo; • funkcinio patvirtinimo; • palyginamojo įvertinimo; • matematinio įvertinimo; • projekcijos valdikliai. <p>Išvesties taisyklės - tai taisyklės, kurios sudaromos iš kitų taisyklių. Jos gali būti sudarytos iš nedalomų arba kitų išvesties taisyklių. Išskiriama 58 išvestinių taisyklių tipai, kurie pagal panaudojimo sritį yra grupuojami į 12 tipų grupių.</p>
General Data Analysis Rule Types	<p>Atributai [11, 20]:</p> <ul style="list-style-type: none"> • unikalumo užtikrinimo; • privalomumo požymis; • reikšmės patvirtinimo.

Autorius	Klasifikacija
	Matematinis skaičiavimas. Loginė išvada. Esybės-atributo kiekybinis apribojimas. Sąsajų taisyklės: <ul style="list-style-type: none"> • kardinalumo; • būtinumo; • integralumo; • kardinalumo apribojimo.
C. J. Date (2000) [7]	Apribojimo; Būsenos apribojimo; Reikšmės kitimo apribojimo; Įvykio-reakcijos; Išvesties; Matematinio skaičiavimo; Loginės išvesties.
C. J. Date (2000) [11]	Autorius siūlo alternatyvią klasifikacijos schemą, paremtą duomenų modeliu: <ul style="list-style-type: none"> • dalykinės srities apribojimai; • stulpelių (duomenų esybės) apribojimai; • esybės apribojimai (apribojimai galiojantys konkrečiai esybei - lentelei); • duomenų modelio apribojimai (apribojimai, galiojantys tarp kelių modelio esybių).
Versata Inc. [11]	Integralumo užtikrinimo; Išvesties taisyklė; Tikrinimo (atributo reikšmė privaloma, maksimali riba, minimali reikšmės riba...); Ryšių apribojimo; Įvykio/veiksmo; Atvaizdavimo taisyklės.
USoft Inc. [11]	Apribojimo taisyklės: veiklos apribojimai saugomai informacijai; Elgsenos taisyklės: <ul style="list-style-type: none"> • sistemos elgsena tam tikrose numatytose situacijose; • sistemos automatiniai veiksmai nenumatytose situacijose. Išvesties taisyklės: <ul style="list-style-type: none"> • sistemos logikos valdymas; • sistemos skaičiavimų valdymas. Atvaizdavimo taisyklės - kaip sistema sąveikauja su vartotoju, veiksmų ir užduočių organizavimo tvarka; Nurodymų taisyklės: kaip vartotojas turi valdyti sistemą tam tikrose situacijose.

Veiklos taisyklių valdymo modelio parinkimas pirmiausia priklauso nuo tikslo, kuriam tarnaus. Tačiau, nepriklausomai nuo tikslo, veiklos taisyklių klasifikacijos modelis yra naudingas veiklos taisyklių aptikimui, jų analizei ir struktūrizavimui.

1.3. Veiklos taisyklių klasifikacija pagal pritaikymą

Veiklos taisyklės pagal tipus taip pat galim skirstyti į orientuotas į duomenų bazę, bei orientuotas į taikomąją programą. Toks taisyklių suskirstymas galėtų pagerinti duomenų modelio bei taikomosios programos kūrimą. Abu veiklos taisyklių tipai naudoja tam tikrus apribojimus, kurie padeda įgyvendinti ir palaikyti bendrą duomenų integralumą. Skiriasi jie tuo, kur ir kaip jie yra laikomi.

1.3.1. Orientuotos į duomenų bazę

Orientuotos į duomenų bazę veiklos taisyklės yra tos, kurios gali būti nustatytos duomenų bazės viduje. Duomenų bazę gali būti sumodeliuota kaip leidžiamų duomenų bazės būsenų aibė, kur [18]:

- kiekviena duomenų bazės būseną yra leidžiamų lentelės reikšmių būsenų aibė;
- kiekviena lentelės reikšmė yra leidžiamų įrašų reikšmių aibė;
- kiekviena įrašo reikšmė yra leidžiamų atributų (stulpelių) reikšmių aibė;
- kiekviena atributo reikšmė yra leidžiama reikšmė iš šio atributo srities.

Patogiausia yra leisti tik 'leidžiamas' reikšmių aibes. Yra keturi apribojimų tipai (pateikti didėjimo tvarka):

1. atributų apribojimai: sąlygos, kurias turi atitikti atributo reikšmės;
2. įrašų apribojimai: sąlygos, kurias turi atitikti atributų reikšmių kombinacijos tame pačiame įrašė;
3. lentelių apribojimai: sąlygos, kurias turi atitikti įrašų kombinacijos toje pačioje lentelėje;
4. duomenų bazės apribojimai: sąlygos, kurias turi atitikti lentelių kombinacijos toje pačioje duomenų bazės būsenoje.

Šio tipo taisyklės naudojamos projektuojant veiklos srities duomenų modelį. Jos palengvina duomenų analitikų bei projektuotojų darbą. Kai kurios taisyklės nebūtinai turi būti realizuotos, jos gali būti naudojamos kaip papildomas duomenų šaltinis.

1.3.2. Orientuotos į taikomąją programą

Orientuotos į taikomąją programą veiklos taisyklės yra pareiškimai, kurie naudoja apribojimus, kurių negalima įgyvendinti modifikuojant atributų specifikaciją ar ER diagramą [13].

Orientuotos į taikomąją programą veiklos taisyklės pagrindinai yra dviejų tipų:

- *veiklos* taisyklės (valdymo, veiklos, priėjimo teisių, žinių ir t.t.) *Veiklos* taisyklės įtraukia organizacijos taisykles, politikas, taip pat įstatymų nuostatas, reguliavimo taisykles ar vidinius organizacijos įstatymus, standartus, praktikas, politiką, kontraktus;
- *sistemos* taisyklės (stebėjimas ir kontrolė, transakcijos, resursų utilizacija, patikimumas/prieinamumas, atstatymas, priėjimo kontrolė).

Valdymo taisyklės

Valdymas įtraukia planavimą, vadovavimą, organizavimą, stebėjimą ir kontrolę.

- planavimas, vadovavimas, organizavimas - resursų paskirstymas užduotims;
- stebėjimas ir kontrolė. Stebėjimas įtraukia aktyvumo ir pasyvumo nustatymą, palygindamas esamą elgesį su pageidaujamu. Kontrolė naudojasi stebėjimo aptiktais įvykiais arba 'ne-įvykiais'. 'Pvz.: Vartotojas privalo atsijungti, jei būna neaktyvus 30 min'.

Priėjimo kontrolės taisyklės

Priėjimo kontrolės taisyklės aprašo kas gali prieiti prie kokių duomenų ir kas gali paskirti leidimus. Pvz. 'Tik prisiregistravęs vartotojas gali prisijungti prie tinklapio'.

Veiklos praktikos taisyklės

Veiklos praktikos taisyklės įtraukia:

- veiklos rizikos ir kontrolės taisyklės. Pvz. 'Užsakymas privalo būti patvirtintas vadybininko'.
- įkainių taisyklės. Jos apima įvairias nuolaidas klientams, specifines sąlygas ir pan.;
- paslaugų sąlygų taisyklės. Pvz. 'Produktai privalo būti pristatyti per 24 valandas po užsakymo patvirtinimo';
- formules, algoritmus, matavimo kriterijus bei standartus. Formulės ir algoritmai yra bendri apskaičiavimai, pvz. 'Metinis atlyginimas = savaitinis atlyginimas * 52 savaitės'.

Duomenų integralumo taisyklės

Duomenų integralumo taisyklės apima:

- lygiagretaus priėjimo taisyklės. Pvz. 'Darbuotojas negali pasiekti pirkėjo įrašo, jei jį modifikuoja kitas darbuotojas';
- įvedimo validavimo taisyklės. Validavimo taisyklės aprašo įvairias sąlygas, kurios turi būti patenkintos, kad surinkta informacija būtų galima įvesti į duomenų bazę;
- transakcijų taisyklės. Pvz. 'Prieš patvirtinat transakciją reikia užtikrinti duomenų kopiją'.

Išvedimo taisyklės

Išvedimo taisyklės gali priklausyti nuo:

- puslapių. Pvz. puslapio formatas, standartiniai puslapio ar dokumento dalis virš teksto (*header*) ir dalis po tekstu (*footer*), standartinis puslapio fono paveikslukas ir spalva ir pan.;
- laukų. Pvz. reikalaujamas lauko formatas, spalva, šriftas, rūšiavimo tvarka, sąlygos, prie kurių atitinkamas laukas privalo ar neprivalo būti rodomas.

Ontologinės taisyklės

Jos siejasi su:

- apibrėžimais. ‘Darbo savaitė prasideda pirmadienį ir baigiasi penktadienį’;
- kvalifikacijos kriterijais. Referentai turi atitikti kriterijų, pvz. ‘Norint gauti darbą, reikia 2 metų stažo’.

Filtravimo taisyklės

Filtravimo taisyklės pateikia kriterijus, pagal kuriuos galima atrinkti informaciją ar informacija iš vienos būsenos pereina į kitą. Pvz. ‘Parodyti užsakymus, kurie buvo pateikti per paskutines 10 dienų’, ‘Produktas nebetinkamas vartoti, jai buvo sandėlyje daugiau nei 30 dienų’.

Patikimumo, vykdymo taisyklės

Taisyklės dažnai naudojamos patikimumo ar vykdymo pagerinimui, pvz. ‘Jie serveris ‘nulūžta’ ar yra perkrautas, vartotoją nukreipti į kita prieinamą serverį

Šio tipo taisyklės naudojamos taikomosios programos kūrėjų bei veiklos taisyklių analitikų. Kaip ir orientuotos į duomenų bazę taisyklės, jos taip pat nebūtinai visos privalo būti realizuotos, kai kurios iš jų gali būti naudojamos tam, kad pagerintų taikomosios srities bei projektuojamos programos veikimą.

Visi taisyklių tipai pateikiami 1.2 lentelėje.

1.2 lentelė

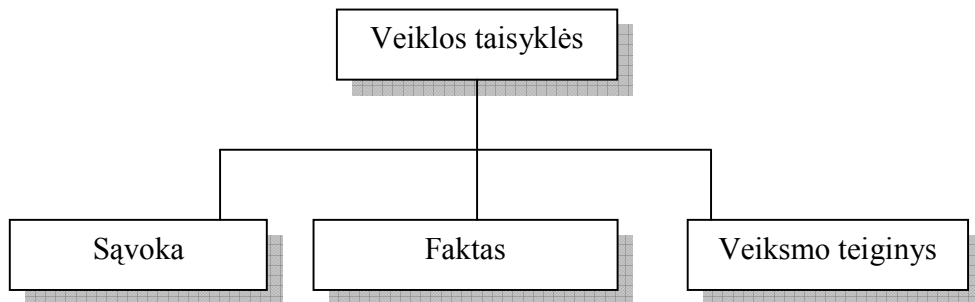
VT tipai pagal jų pritaikymą

Orientuotos į duomenų bazę VT	Orientuotos į taikomąją programą VT
Atributų apribojimai	Valdymo taisyklės
Įrašų apribojimai	Priėjimo kontrolės taisyklės
Lentelių apribojimai	Veiklos praktikos taisyklės
Duomenų bazės apribojimai	Duomenų integralumo taisyklės
	Išvedimo taisyklės
	Ontologinės taisyklės
	Filtravimo taisyklės
	Patikimumo, vykdymo taisyklės

1.4. Veiklos taisyklių modelio parinkimas

Išanalizavus pateiktas veiklos taisyklių klasifikacijos metodikas ir įvertinus darbo specifiką, nuspręsta sukauptos informacijos pagrindu sudaryti naują veiklos taisyklių klasifikaciją, kuri remiasi GUIDE projekto pateiktu modeliu [5], bei Halle veiklos taisyklių klasifikacijos modeliu [10]. Pateiktas VT klasifikacijos modelis pavaizduotas 1.1 paveiksle ir tenkina sekančius reikalavimus:

- Leisti aprašyti pilną rinkinį veiklos taisyklių;
- Veiklos taisyklių pateikimo ir klasifikacijos forma turi būti lengvai suprantama tiek veiklos žinovui, tiek IS kūrimo komandos atstovui;
- Atitikti pagrindinių pateiktų veiklos taisyklių bendrus klasifikacijos principus.



1.1 pav. Aukštesnio lygio veiklos taisyklių klasifikacijos schema

Pateiktas veiklos taisyklių klasifikacijos modelis suskirsto veiklos taisykles į tris pagrindines grupes:

1. *Sąvokos* – tai žodis ar frazė, kuris turi specifinę reikšmę veiklai [5]. Sąvokos gali būti:
 - veiklos sąvokos: žodis ar frazė, turintis specifinę reikšmę veiklai kokiame nors nurodytame kontekste;
 - bendros sąvokos: kasdieninės kalbos žodžiai, naudojant įprastinę jų reikšmę. Bendros sąvokos yra bendrinio žodyno dalis, pvz. ‘mašina’, ‘miestas’ ir pan.
2. *Faktai* parodo asociaciją tarp dviejų ar daugiau sąvokų, kitaip tariant, išreiškia ryšį [5]. Faktas įtraukia dvi ar daugiau sąvokų, o sąvoka gali būti viename ar daugiau faktų. Faktai gali būti:
 - pirminis faktas;
 - išvestinis faktas: tai teiginys, kuris susideda iš kitų teiginių.
 - atributas išreiškia faktą, kuriame viena sąvoka aprašo kitos sąvokos aspektą. Pvz. ‘vardas yra pirkėjo atributas’;
 - dalyvis išreiškia faktus, kuriuose sąvokų aibės yra susietos tam tikra, bizniui reikšminga, prasme, pvz. ‘automobilis gali būti išnuomotas daugeliui klientų’(bet tik vienam atitinkamu laiko momentu;

- apibendrinimas nustato, kad konkreti sąvoka (supertipas) yra vienos ar daugiau kitų sąvokų (subtipų arba potipių) apibendrinimas. Šio tipo fakte viena sąvoka-tipas aprašo kitos sąvokos-tipo variantų poaibį.

3. *Veiksmo teiginiai* – tai sakiniai, kurie turi ryšį su koku nors dinamiu veiklos aspektu [10]. Jis apibrėžia rezultato, kurį duoda veiksmas, apribojimus. Apribojimai aprašomi neprocedūriškai, kitų nedalomų veiklos taisyklių terminais. Veiksmo teiginių tipai:

- veiksmą valdantis – aprašo, kas privalo ar neprivalo būti ar įvykti;
- veiksmą veikiantis – aprašo, kas turėtų ar neturėtų būti ar įvykti;
- leidimas – teiginys, kuris aprašo prerogatyvą ar privilegiją vienos ar daugiau konstrukcijų atžvilgiu. Šį teiginį galima atvaizduoti predikatu '(Tiktai) X gali daryti Y', kur X paprastai yra vartotojas, o Y yra veiksmas, kuris gali būti vykdomas.
- matematinis skaičiavimas pateikia išvestinius faktus, priklausomai nuo aprašyto matematinio algoritmo.
- loginė išvada pateikia išvestinius faktus, naudojantis logine indukcija (iš konkrečių elementų) ar dedukcija (iš bendrų elementų).

Nurodyto veiklos taisyklių klasifikacijos elementų atitikmuo kitos projektavimo metodikose atvaizduotos 1.3 lentelėje.

1.3 lentelė

Pateiktos ir išanalizuotų veiklos taisyklių metodikų sąsaja

Pateikta metodika	GUIDE	GDART	C.J. Date	C.J. Date - DB	Versata Inc.	USoft Inc.
Sąvoka	Sąvoka	-	-	-	-	-
Faktas	Faktas	-	-	-	-	-
Veiksmą valdantis	Veiksmą valdantis	Sąsajų taisyklės	-	-	Iš dalies tikrinimo	Elgsenos taisyklės
Veiksmą veikiantis	Veiksmą veikiantis	Sąsajų taisyklės	Apribojimo	-	Iš dalies tikrinimo	Elgsenos taisyklės
Leidimas	Leidimas	-	Būsenos/reikšmės apribojimo	-	-	Apribojimo taisyklės
Matematinis skaičiavimas	Matematinis skaičiavimas	Matematinis skaičiavimas	Matematinio skaičiavimo	-	Iš dalies tikrinimo	Sistemos skaičiavimų valdymas
Loginė išvada	Loginė išvada	Loginė išvada	Loginės išvesties	-	Iš dalies tikrinimo	Sistemos logikos valdymas

Kaip galima pastebėti iš pateiktos metodikų palyginimo matricos, GUIDE projekto siūlomas modelis pilnai padengia darbe pasiūlytą modelį. Tačiau GUIDE projekto modelis skiriasi pilnumu ir

struktūrizacija, o darbe yra siūlomas modifikuotas modelis, pagal kurį struktūrizuotas taisykles galima būtų realizuoti bet kuria iš pateiktų metodikų. Ross ir C. J. Date į duomenų bazes orientuoti modeliai fundamentaliai skiriasi nuo pateiktos metodikos, dėl ko palyginime nefigūruoja.

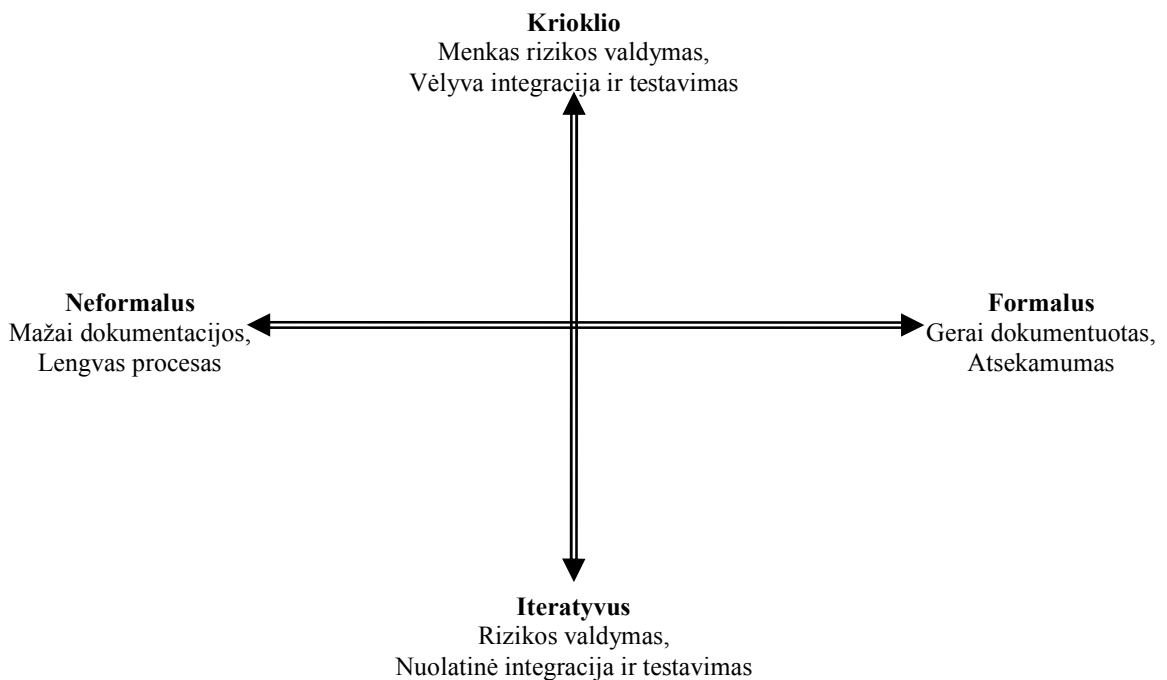
2. PROJEKTAVIMO METODIKOS PARINKIMAS

Šiuo darbu siekiama ne sukurti iš esmės naują metodiką, o „įlieti“ VT požiūrį į jau egzistuojančią ir praktikoje patikrintą metodiką. Nagrinėjamos metodikos: RUP, MSF, Agile, OSS, HALLE.

Projektavimo metodikų palyginimui naudosime dviejų matmenų sistemą [1], kur:

- horizontalioje ašyje – proceso dokumentacijos, atsekamumo, formalizacijos lygis;
- vertikaliojoje ašyje – krioklio principas: nuoseklus, integravimas ir testavimas vėlyvoje projekto stadijoje, iteratyvus principas: rizikos valdymas, integracija bei testavimas atliekamas ankstyvose projekto stadijose.

Projektavimo metodikų palyginimo koordinačių plokštuma pavaizduota 2.1 paveiksle.

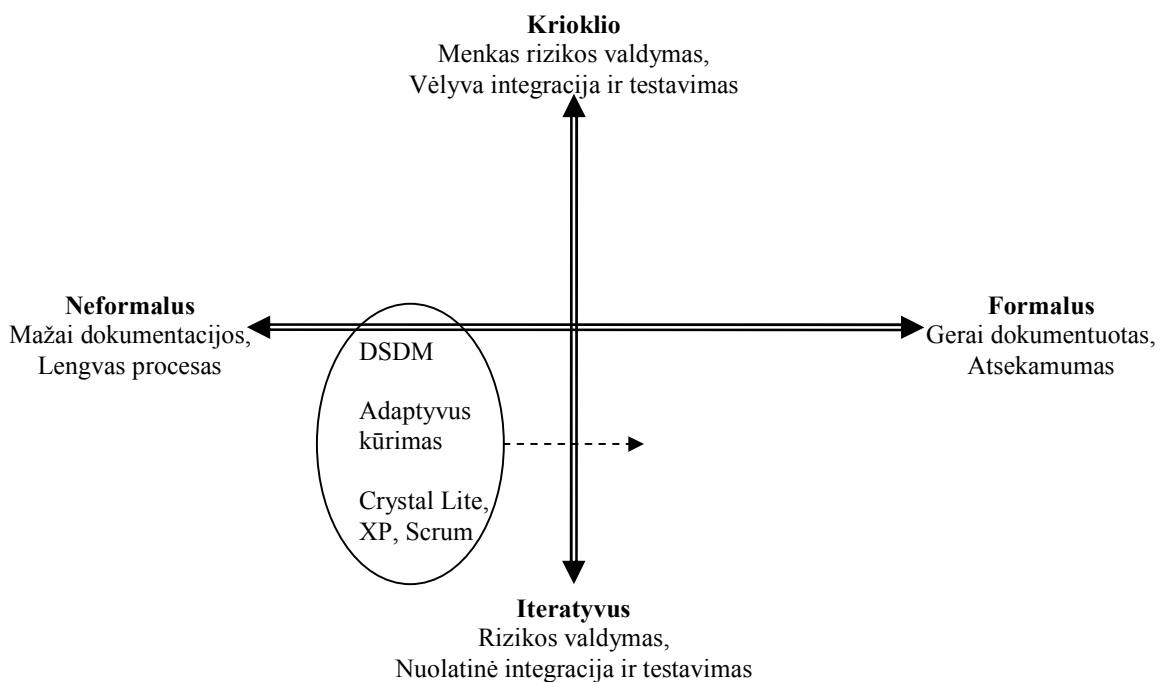


2.1 pav. Projektavimo metodikų palyginimo koordinačių plokštuma.

2.1. Agile požiūris: neformalios, iteratyvios projektavimo metodikos

Agile, kaip projektavimo metodikos yra ganėtinai naujas, tačiau greitai plintantis reiškinys [24]. Ypač nedideliuose projektuose ir mažose komandose. Kelios iš Agile metodikų yra: XP, Scrum, bei adaptyvus kūrimas.

Agile požiūris aukoja dokumentacijos pilnumą, bei procesą vardan lankstumo ir prisitaikymo prie besikeičiančių veiklos reikalavimų [1]. Daugiau dėmesio yra skiriama spartesniam veikiančio produkto pateikimui, nei dokumentacijos nuoseklumui ir pilnumui. Vietoj darbo pagal sudarytą konkretų planą Agile procesas siekia reaguoti į poreikių pasikeitimus, atsirandančius kūrimo procese. Nors tai nereiškia, kad dokumentacija ir planavimas nėra svarbu, tiesiog veikiančio produkto pateikimas laiku yra svarbiau. Taipogi verta pastebėti, kad Agile požiūris išpopuliarėjo apie 1990-uosius metus [14]. Agile požiūrio atsiradimas yra paremtas geriausiomis praktikomis/analogais, sukautais per daugelį metų: iteratyvus kūrimas, tęstinis integravimas, sutelkimas į veikiančią produktą... Taigi vertinimo koordinatinių plokštumoje Agile požiūris patenka į žemesnįjį kairinį ketvirtį, kadangi visos šiame požiūryje esančios metodikos yra iteratyvios ir neformalios – pateikiančios minimalią dokumentaciją (2.2 pav.).



2.2 pav. Agile požiūris projektavimo metodikų vertinimo plokštumoje.

2.2. SEI CMM, SEI CMMI, ISO/IEC standartais įvertinamos metodikos

Šalia Agile požiūrio yra poreikis ir tokioms formalioms projektavimo metodikos kokybę nusakančioms metodikoms, kaip SEI CMM, SEI CMMI, ISO/IEC [1]. Įmonės pradeda į IS žiūrėti kaip į strateginę investiciją ir kurdamos IS nori turėti garantijų dėl biudžeto pakankamumo ir kokybės. To pasekoje įmonės pradeda suvokti kokybiško proceso ir dokumentacijos kokybės svarbą.

Formalios projektavimo metodikos charakteristikos:

- Planai yra smulkiai dokumentuoti;
- Dokumentacija apima valdymo, poreikių, architektūros bei testavimo sritis;
- Valdymo, poreikių, architektūros bei testavimas yra kruopščiai sudokumentuoti ir saugomi versijų kontrolės sistemoje;
- Susekami artefaktų ryšiai tarp poreikių, architektūros, testavimo elementų yra sukuriami ir palaikomi.
- Pasikeitimams reikalingas pokyčių kontrolės grupės patvirtinimas.
- Peržiūros rezultatai yra kruopščiai įrašomi ir sekami.

Formalios metodikos geriausiai tinka sudėtingoms sistemoms, prie kurių dirba didelės komandos, tačiau kūrimo laikas yra ilgesnis, o kaina didesnė. Kita vertus tokia sistema yra lengviau palaikoma.

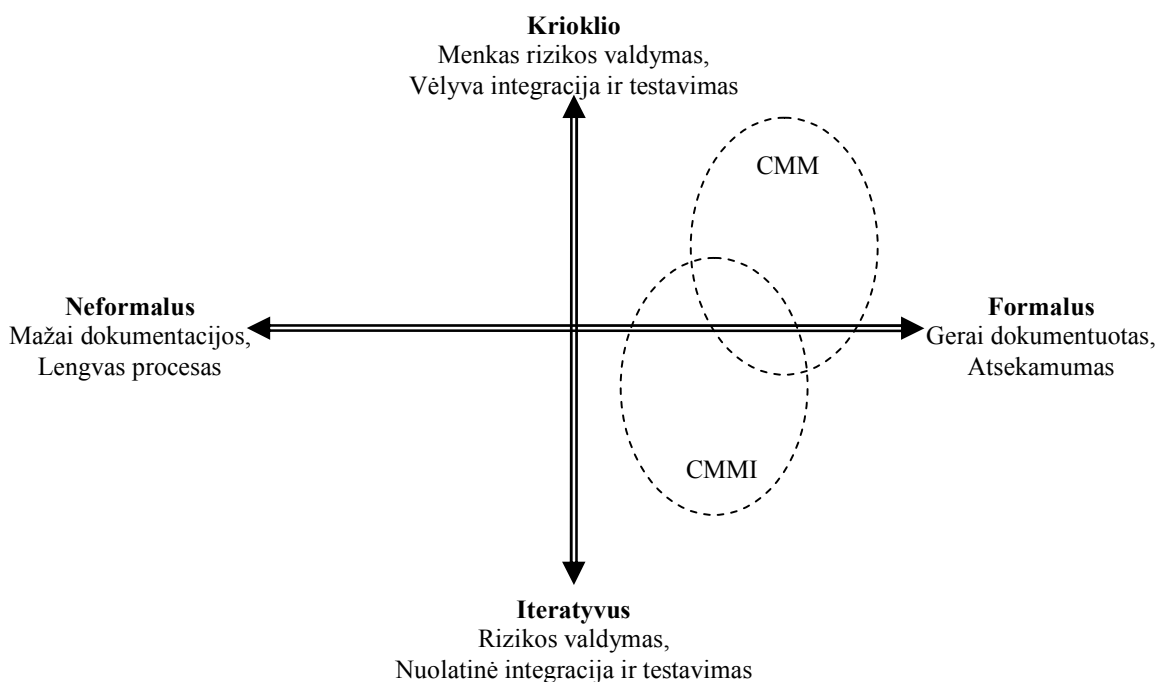
2.2.1. SEI CMM - proceso kokybės įvertinimo metodas

Siekiant geresnio nuspėjamumo ir aukštesnės kokybės, organizacijos pirmiausia turi žinoti savo proceso trūkumus ir kaip juos galima eliminuoti ateityje. CMM(*Capability Maturity Model*) sukurtas Software Engineering Institute kaip tik tam ir skirtas [1]. CMM nėra projektavimo metodika, tai greičiau projektavimo metodikos kokybės/brandos įvertinimo metodika, kuri procesą graduoja 1-6 balų skalėje. CMM nenurodo, kaip kurti programinę įrangą. Tam yra procesai(XP, RUP, MSF).

CMM pilnai apibrėžia rodiklius, kurie nusakytų produkto kūrimo proceso brandą, tačiau jei įmonė stengiasi aklai patenkinti CMM keliamus reikalavimus nesiekdama proceso optimalumo, tai rezultatas gali gautis atvirkštinis ir procesas per daug formalizuotas. Taipogi CMM proceso reikalavimai natūraliai veda prie krioklio proceso (2.3 pav.).

2.2.2. SEI CMMI - modernizuotas proceso kokybės įvertinimo metodas

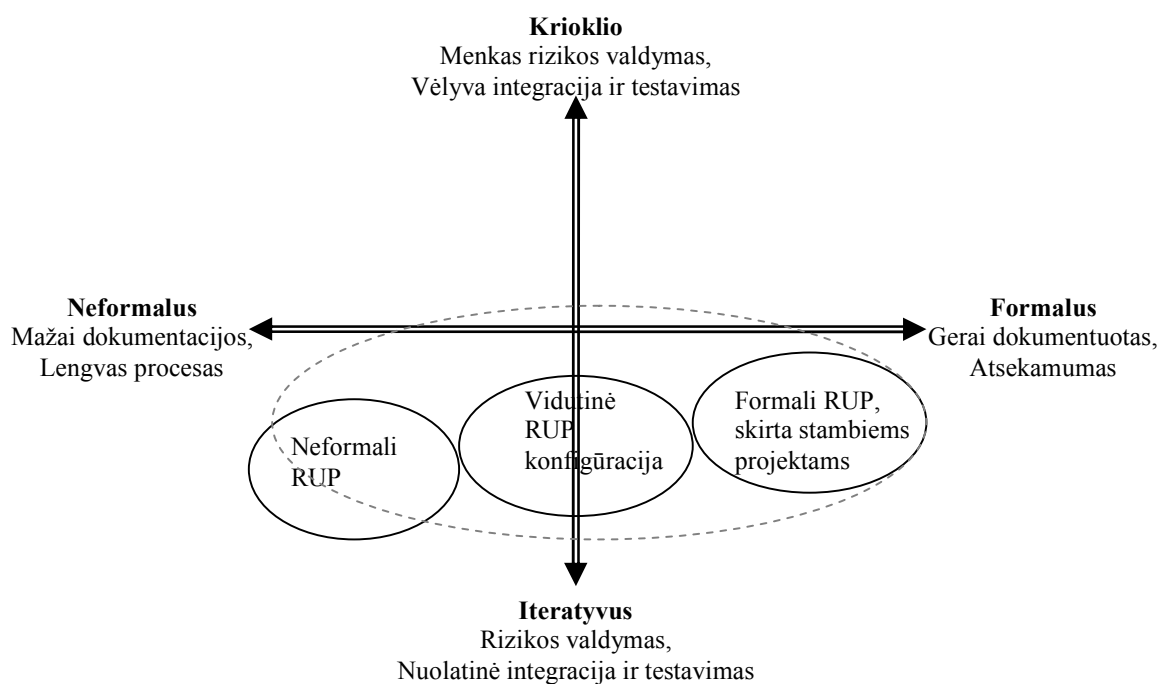
SEI CMMI yra atnaujintas CMM variantas, siekiant įvertinti modernius procesus – tokius, kaip į riziką reaguojančius ir iteratyvius [1]. Vietoj to, kad formalizuoti procesą, kaip to reikalauja CMM, CMMI susikoncentruoja į tam tikras korekcijų reikalaujančias sritis, kurios yra svarbiausios organizacijos tikslams pasiekti ir sumažina galimą projektų riziką. 2.3 paveiksle pavaizduoti CMM ir CMMI įgyvendinančių projektavimo metodikų įvertinimai [1].



2.3 pav. CMM ir CMMI projektavimo metodikų brandos vertinimo metodikos.

2.3. RUP IS kūrimo procesas

RUP (*Rational Unified Process*) yra lanksti IS kūrimo metodika, kurios formalumas priklauso didžiaja dalimi nuo projekto dydžio ir kuriamos sistemos tipo ar vartotojo poreikių [3] (2.4 pav.). RUP propaguoja rizikos valdymu paremtą, iteratyvų, nepertraukiamu testavimu ir integracija paremtą kūrimo procesą. Be kita ko yra lankstumo krioklio/iteratyvaus ašyje. Tam tikros organizacijos naudoja RUP ir Krioklio procesą, nors tai yra nerekomenduojama.

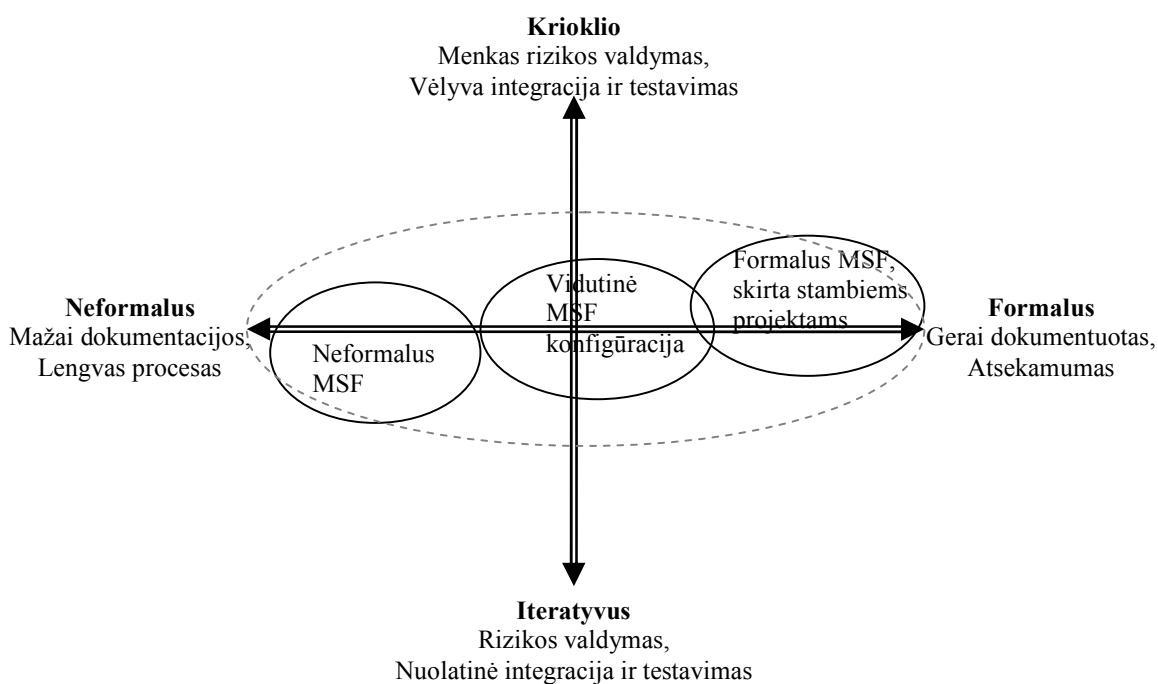


2.4 pav. RUP leidžia pasiekti lankstumą priklausomai nuo projekto dydžio ir vartotojo poreikių.

2.4. MSF projektavimo metodika

MSF (*Microsoft Solutions Framework*) pasiūlyta projektavimo metodika yra labai panaši į RUP. MSF sujungia krioklio ir iteratyvų IS kūrimo gyvavimo ciklus: krioklio versijomis pagrįstą planavimą ir iš to sekantį nuspėjamumą, bei iteratyvaus metodo atgalinį ryšį ir kūrybingumą [2]. MSF kaip ir RUP siūlo lankstumą priklausantį nuo vartotojo poreikių bei projekto dydžio.

MSF pozicija projektavimo metodikų įvertinimo skalėje pavaizduota 2.5 paveiksle [1].

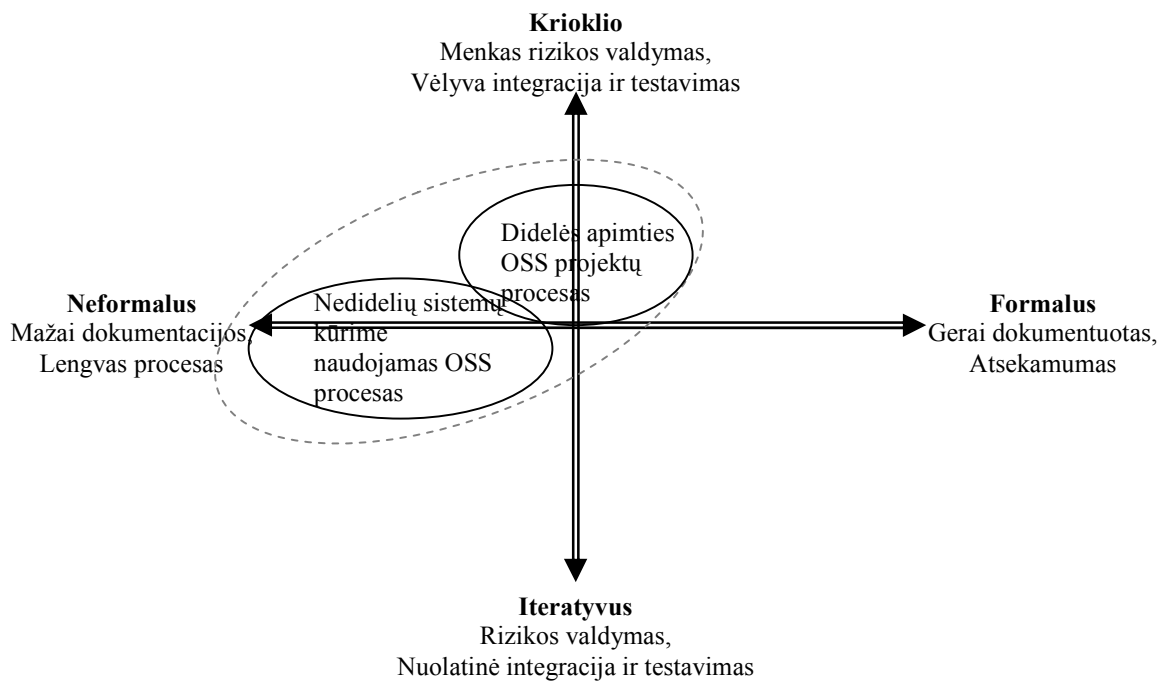


2.5 pav. MSF pozicija projektavimo metodikų įvertinimo skalėje.

2.5. OSS – atviro kodo licenzija paremtu IS kūrimo procesas

OSS (*Open Source Software*) – atviro kodo licencijomis paremtos programinės įrangos kūrimo procesas. Kuriama programinė įranga yra nemokama, viešai platinama ir vartotojų skaičius priklauso tik nuo kuriamos programinės įrangos aktualumo ir kokybės [17].

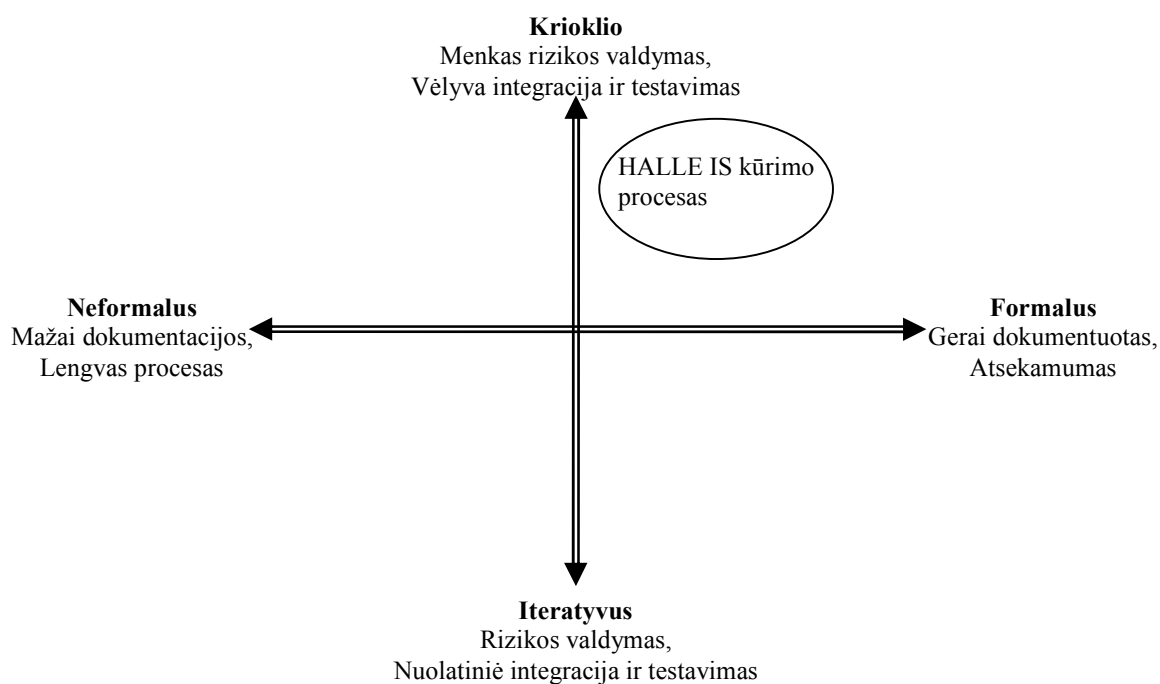
Atviro kodo licencijomis paremtos programinės įrangos kūrimo procese programuotojai nevaržomai gali modifikuoti esamą sistemos išėties kodą ar įtraukti naują funkcionalumą. Projektas, priklausomai nuo apimties ir licencijos versijos gali turėti daugiau ar mažiau struktūrizuotą komandos struktūrą. Procesas susideda iš komandos branduolio sudarymo, rolių paskirstymo, tikslų bei koncepcijų nustatymo. Komandos struktūra yra dinamiška ir dažnai kintanti. Proceso įvertinimas pateiktas 2.6 paveiksle.



2.6 pav. OSS – neformalus ir lankstus kūrimo procesas.

2.6. HALLE – veiklos taisyklėmis paremta IS kūrimo metodika

HALLE – veiklos taisyklėmis paremtų IS kūrimo procesas [10]. Proceso esmė yra kuo ankstesnėje kūrimo stadijoje stabilizuoti duomenų modelį ir pagal sudarytą modelį surinkti, klasifikuoti bei verifikuoti veiklos taisyklių aibę, kuri bus pagrindu kuriamoje sistemoje. HALLE metodika pateikia veiklos taisyklių struktūrizavimo modelį surinkimo, valdymo bei saugojimo metodą. HALLE metodikos geriausios savybės yra atskleidžiamos krioklio procese (2.7 pav.).



2.7 pav. HALLE – veiklos taisyklėmis paremta IS kūrimo metodika.

2.7. Projektavimo metodikų palyginimas

Detalus projektavimo metodikų palyginimas pateiktas 2.1 lentelėje.

2.1 lentelė

Projektavimo metodikų palyginimas

Charakteristika	RUP	MSF	OSS	HALLE	XP
Projekto dydis	Didelės apimties projektai, didelė komanda su teise prieiti prie išėjimo kodo valdymo sistemos.	Įvairios apimties projektai, išskirtos rolių atsakomybės ribos.	Didelės apimties projektai. Daug programuotojų. Daug vartotojų-testuotojų.	Įvairios apimties, su centrine veiklos taisyklių saugykla.	Tinka mažos apimties projektams arba didelė sistema skaidoma dalimis.
Kuriamos IS tipas	Neribojama.	Neribojama.	Infrastruktūrinės, daugiavartotojiškos.	Verslo valdymo sistemos.	Nedidelės sistemos, arba didelių sistemų atskiros dalys.
IS kūrimo principai	Iteratyvus procesas, reikalavimų valdymas, komponentais paremta architektūra, CASE priemonių panaudojimas,	Iteratyvus procesas, reikalavimų valdymas, kompromisų valdymas, komponentais paremta architektūra, CASE	Pakeitimų ir atnaujinimų laisvė. Galimybė teikti atnaujinimus trečioms šalims nemokamai. Greitas reagavimas į vartotojų pageidavimus.	Krioklio principas, proceso ir kuriamos sistemos šerdis – veiklos taisyklės. Veiklos taisyklių valdymas.	Darbas su klientais; Metaforų naudojimas sudėtingiems procesams aprašyti; Planavimas; Trumpi susirinkimai; Prioritetas testavimui; Programavimo standartų laikymasis;

Charakteristika	RUP	MSF	OSS	HALLE	XP
	kokybės ir pasikeitimų valdymas.	priemonių panaudojimas, kokybės ir pasikeitimų valdymas, lanksti komandos struktūra.			Bendra kodo nuosavybė; Nepertraukiamas integravimas; Dažnas atnaujinimų pateikimas; Reagavimas į reikalavimų pasikeitimus.
Rolių ir atsakomybės struktūra	Didelė komanda su teise priėti prie išėities kodo valdymo sistemos.	Rolės: projekto valdymas, produkto valdymas, programavimas, testavimas, vartojamumo valdymas, pateikimo valdymas. Dinamiška rolių priskyrimo komandos nariams metodika.	Daug savanoriškais pagrindais dirbančių programuotojų.	Veiklos taisyklių valdymo. Kitos projekto rolės nenusakytos. Klientas-veiklos žinovas įtrauktas į projektą.	Programavimas poromis; Rolės: priežiūros/valdymo, pasikeitimų valdymo, programuotojas, architektas.
Kūrimo procesas	Turi būti sudarytas detalus kūrimo planas. Iteratyvus ir rizikas įvertinantis kūrimo procesas.	Projektas vykdomas konkrečiomis fazėmis ir iteratyviai. Rizikos ir versijų valdymo procesas.	Nėra detalaus sistemos kūrimo plano. Nėra projekto plano, grafiko.	Veiklos taisyklėms suteikiamas prioritetas. Aiškios veiklos taisyklių valdymo fazės ir metodika.	Nėra pilno proceso, bet susitelkiama į efektyvų reikalavimų modeliavimą ir dokumentavimą.
Kūrimo fazės	Pasiruošimas (<i>inception</i>), Plėtojimo (<i>elaboration</i>), Realizavimas (<i>construction</i>), Pateikimas (<i>transition</i>).	Pasiruošimo, planavimo, kūrimo, stabilizavimo, diegimo.	Nustatyti roles ir atsakomybes. Nustatyti darbo apimtis ir projekto tikslus. Suskirstyti ir vykdyti kūrimo procesą. Testavimas prieš pateikiant.	Konceptualizavimas, VT atradimas, VT analizė, veiklos taisyklių modelio formavimas, realizavimas, testavimas.	Sistemos konceptualizavimas. Sistemos planavimas. Sistemos kūrimas. Sistemos pateikimas.
Testavimas	Pirma vykdomas atskirų komponentų testavimas, kurį seka vis stambesnių IS dalių integravimas ir testavimas. Proceso ir kuriamos IS testavimas pastoviai sekamas.	Atitinka RUP principus. Dažnų vidinių versijų pateikimas, įvairių faktorių stebėjimas.	Nėra konkretaus testavimo plano. Testuojama pastoviai kūrimo metu.	Veiklos taisyklių teisingumo ir jų valdymo modelio pilnumo užtikrinimas bei konfliktų paieška bei šalinimas.	Pirma testuojama, po to realizuojama. Testuojama viskas, ką galima testuoti. Našumo bei kokybės testavimo procesas pastovus.

Charakteristika	RUP	MSF	OSS	HALLE	XP
Realizacijos ir produkto pateikimo valdymas	Dažnas tarpinių versijų pateikimas, tačiau pateikimo versijoje pateikiamos „beta“ versijos ir galutinis produktas.	Dažnas tarpinių versijų pateikimas, iteratyvus kūrimo procesas.	Dažnas atnaujintų produkto komponentų pateikimas.	Klientui pateikiamas pilnas, galutinis produktas.	Dažnų nedidelių versijų pateikimas klientams.
Projektavimas ir naudojami artefaktai	Veiklos, reikalavimų, architektūros modeliavimas. Projekto valdymo, testavimo planai.	Atitinka RUP. Pagal poreikius gali būti modifikuotas.	Proceso modelius reikia sukurti.	Veiklos taisyklių rinkinys.	Panaudojimo atvejų, klasių, duomenų bei vartotojo sąsajos modeliai. Skirtingi modeliai kuriami lygiagrečiai.
Proceso modeliavimo ir kūrimo priemonės	Visa programinė įranga pateikiama.	Visa programinė įranga pateikiama.	Reikalingi nemokami servisai. Naujienų grupės. Kitos nesudėtingos bendradarbiavimo priemonės. Remias nemokamais produktais.	Teikiami komerciniai veiklos taisyklių valdymo įrankiai.	Panaudoja kitų procesų įrankiais.
UML modeliavimas	UML paremtas kūrimas.	UML panaudojimas pagal projekto sudėtingumą ir poreikius.	Pagal poreikius.	Kaip pagalbinė priemonė.	Naudoja 3 UML diagramas: klasių, sekų ir būsenų.

Iš pateiktos išanalizuotų metodikų palyginimo matricos, nuspręsta panaudoti MSF projektavimo metoką, kadangi MSF [2]:

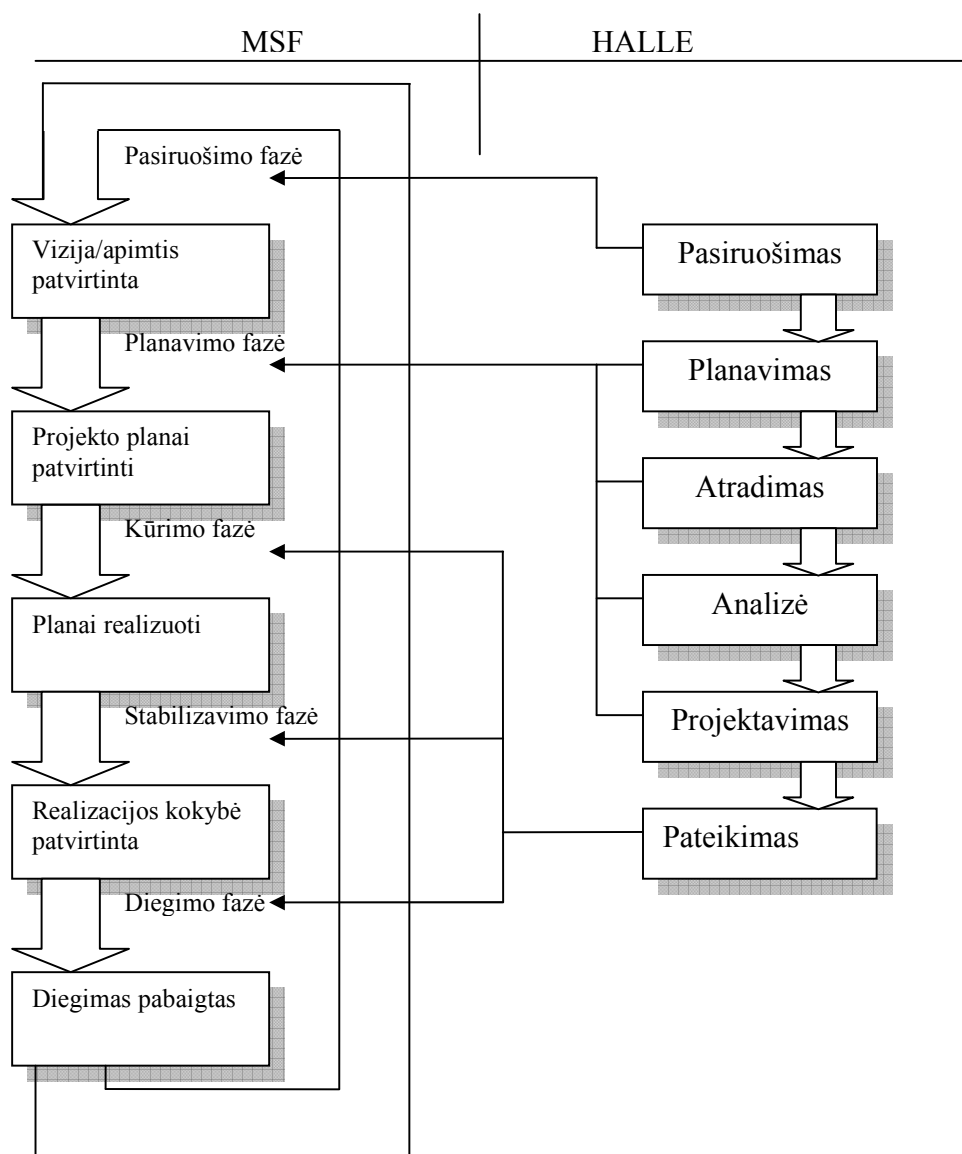
- tinkama bet kokios apimties IS kurti;
- detalizuota, tačiau lanksti rolių ir atsakomybių struktūra – palanki integruoti veiklos taisyklių valdymo rolę;
- pakankamai formalizuotas procesas;
- kūrimo fazės atitinka HALLE veiklos taisyklėmis paremtos metodikos fazes;
- kontrolės ir rizikos valdymo procesas leidžia integruoti veiklos taisyklių modelio valdymą;
- pateikiama integruota ir lanksti kūrimo platforma, leidžianti integruoti veiklos taisyklių valdymo modelį;
- naudojamos pasitvirtinusios UML modeliavimo priemonės;
- egzistuoja bazinis veiklos taisyklių valdymo modelis.

Pateiktus kriterijus tenkina ir RUP procesas, tačiau IBM korporacijos siūlomas procesas yra formalėsniis ir padengia siauresnį projektų dydžio spektrą. Taipogi RUP yra ne toks palankus veiklos taisyklių požiūrio integracijai.

3. VEIKLOS TAISYKLĖMIS PAREMTAS IS KŪRIMO PROCESAS

3.1. Veiklos taisyklių integracija į MSF proceso modelį

Veiklos taisyklės yra integruojamos jau į egzistuojanti ir realiai veikiančią (3.1 pav.) MSF proceso modelį [5].



3.1 pav. MSF ir HALLE proceso modelių sąsajos

Pateikiamas proceso modelis apima geriausias krioklio ir iteratyvaus IS gyvavimo ciklo sudedamąsias dalis, o taipogi integruoja HALLE [10] metodikos fazių fragmentus.

Standartinis MSF gyvavimo ciklo fazės, jų aprašai ir fazės pabaigos rezultatai yra pateikiami 3.1 lentelėje. Pateiktas proceso aprašas yra papildytas ir dėl veiklos taisyklių integracijos atsiradusiais papildomais artefaktais, kaip pvz.: veiklos taisyklių valdymo strategijos parinkimas pasiruošimo fazėje, VT modelio sudarymas proceso planavimo fazėje, bei pasirinkto veiklos taisyklių variklio realizacija kūrimo fazėje.

3.1 lentelė

MSf proceso fazių aprašas

Pavadinimas	Aprašas	Rezultatas
Pasiruošimas	Šioje fazėje komanda, vartotojas bei užsakovas nustato aukštesnio lygio veiklos reikalavimus, bei bendrus projekto tikslus. Svarbiausias pasiruošimo fazės tikslas yra suprasti bendrą kuriamos sistemos tikslą, bei sutarti, kad projektas yra svarbus organizacijai ir kad projekto laukia sėkmė;	Vizijos-apimties dokumentas, kuris apima: <ul style="list-style-type: none"> ▪ sprendžiama problema; ▪ kuriamos IS vizija; ▪ veiklos tikslai; ▪ architektūriniai tikslai; ▪ sprendimo tikslai; ▪ galimos rizikos; ▪ projekto komanda; ▪ kūrimo strategija; ▪ vartotojų profiliai; ▪ planas ir biudžetas; <i>VT požiūrio integracijos rezultatas:</i> <ul style="list-style-type: none"> ▪ <i>veiklos taisyklių strategija.</i>
Planavimas	Šioje fazėje komanda suprojektuoja sprendimą: ką sukurti, kaip sukurti ir kas tai sukurs. Planavimo fazėje yra sukuriamą funkcinė specifikacija, sudaromi darbo planai, kainos įvertinimas bei projekto subproduktai;	Funkcinės specifikacija, kurią sudaro: <ul style="list-style-type: none"> ▪ konceptuali architektūra; ▪ loginė architektūra; ▪ fizinė architektūra; ▪ projekto planas; <i>Dėl VT integracijos atsiradę papildomi artefaktai:</i> <ul style="list-style-type: none"> ▪ <i>veiklos taisyklių modelis;</i> ▪ <i>veiklos taisyklių rinkinys;</i>
Kūrimas	Kūrimo fazėje pagal planavimo fazėje sukurtas specifikacijas programuotojai kuria sistemą;	Kūrimo fazė pabaigiama, kai pagal pateiktą specifikaciją yra sukuriamą funkcinius reikalavimus atitinkanti sistema. <i>Veiklos taisyklių integracija sąlygoja vienokio ar kitokio VT valdymo modelio realizacijos.</i>
Stabilizavimas	Šioje fazėje yra testuojami pilnai realizuoti sistemos moduliai. Nors testavimas yra vykdomas viso kūrimo metu pagal planavimo fazėje nustatytus kriterijus, tačiau stabilizavimo fazėje testavimas apima visos sistemos funkcionavimą, savybių išbaigtumą.	Stabilumo ir vartojamumo reikalavimus atitinkanti IS.
Diegimas	Diegimo fazė yra pasiekiamą, kada sprendimas yra patvirtintas kaip užbaigtas ir projektavimo stadijoje nustatyti kriterijai yra patenkinti. Produktas įdiegiamas	Pas vartotoją įdiegta sistema. Vartotojai apmokyti naudotis pateikta IS.

Pavadinimas	Aprašas	Rezultatas
	pas vartotoją, sistemos vartotojai yra apmokomi dirbti. Ši fazė yra užbaigiama, kai sistemos užsakovas patvirtina sistemos atitikimą pradiniam reikalavimams.	

Integruojant veiklos taisyklių valdymą į MSF gyvavimo ciklą daugiau ar mažiau pakinta visos MSF fazės ir informacija, perteikiama sekančioms fazėms. Iš HALLE VT paremto proceso modelio [10] yra integruojami sekantys artefaktai:

- pasiruošimo fazėje – veiklos įvykių atradimas ir jų susiejimas su analizuojamos veiklos panaudojimo atvejais;
- planavimo fazėje – veiklos taisyklių modelio pasirinkimas;
- atradimo, analizės bei projektavimo fazėse – veiklos taisyklių išskyrimas, formalizavimas ir susiejimas su specifikacijos artefaktais, o taipogi įvedama papildoma veiklos taisyklių valdymo rolė (3.2 skyrelis);
- pateikimo fazėje – pasiruošimo fazėje pasirinkto veiklos taisyklių valdymo modelio integracija su kuriama IS.

3.2. VT požiūrio įtaka MSF rolėms

IS kūrimo procese yra eilė veiksmų, kurie turi būti atlikti ir sprendimas turi būti stebimas iš kelių perspektyvų. To siekiant yra apibrėžiamos 7 skirtingos rolės. Kiekviena rolė turi tiksliai apibrėžtą kompetencijos ir atsakomybės sritį [5]:

- Projekto valdymas: atsakingas už bendravimą su vartotoju ir jo poreikių tenkinimą analizės fazėje. Projektavimo fazėje projekto valdymo rolė atsakinga už reikalavimų surinkimą ir veiklos poreikių patenkinimą. Taipogi projekto valdymo rolė atsakinga už visas kitas komunikacijas su vartotoju kitose IS kūrimo stadijose.
- Produkto valdymas: atsakingas už kūrimo procesą ir produkto pateikimą laiku.
- Programavimas: atsakingas už konkretaus modulio realizaciją pagal programos vadovo pateiktą specifikaciją.
- Testavimas: atsakingas, už produkto kokybės problemų identifikavimą ir sprendimą. Ši rolė nustato, kada sprendimą galima pateikti vartotojui. Taipogi ši rolė patikrina projekto suderinamumą, pilnumą su realizacija.
- Vartojamumo valdymas: analizuoja našumo poreikius ir palaikymo nesklaidumus bei įvertina ar pateiktas sprendimas patenkina vartotojo keliamus reikalavimus.

- Pateikimo valdymas: atsakinga už sklandų sukurtos IS įdiegimą pas klientą. Užtikrina, kad kliento IT infrastruktūra yra paruošta ir tinkama sėkmingam sukurtos IS darbui.
- VT valdymas [10]: veiklos taisyklių atradimas ir išskyrimas analizės fazėje, VT modelio sudarymas, galimų modelio pasikeitimų valdymas, jų susiejimas su kuriama IS ir priežiūra.

Nedidelės apimties projekte, vienas komandos narys gali realizuoti kelias roles, tačiau ne visos rolės yra suderinamos tarpusavyje (3.2 lentelė). Priskiriant konkreitiems projekto vykdytojams roles reikia atsižvelgti, kad tai gali sukelti papildomas rizikas, kurias reikia atidžiai pasverti ir įvertinti.

3.2 lentelė

Modifikuotų MSF rolių suderinamumo matrica

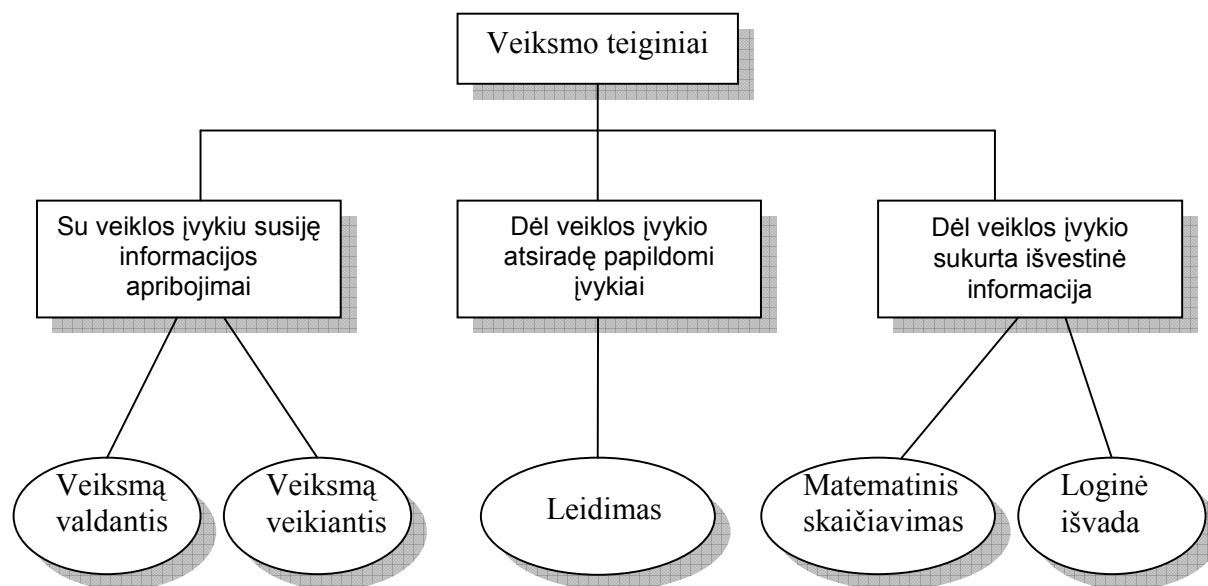
Rolė	Projekto valdymas	Produkto valdymas	Programavimas	Testavimas	Vartojamumo valdymas	Pateikimo valdymas	VT valdymas
Projekto valdymas		GN	GN	G	G	N	GN
Produkto valdymas	GN		GN	N	N	G	N
Programavimas	GN	GN		GN	GN	GN	G
Testavimas	G	N	GN		G	G	G
Vartojamumo valdymas	G	N	GN	G		N	N
Pateikimo valdymas	N	G	GN	G	N		G
VT valdymas	GN	N	G	G	N	G	

Lentelės notacija:

- GN: Griežtai nerekomenduojama – rolės yra nesuderinamos;
- N: Nepatartina – rolės yra suderinamos, tačiau reikia įvertinti rizikas;
- G: Galima – rolės yra suderinamos.

3.3. Detalizuotas veiklos taisyklių klasifikacijos modelis

Analizės dalyje pateiktas veiklos taisyklių klasifikacijos modelis išskiria 3-is VT grupes: sąvoka, faktas ir veiksmo teiginys. Siekiant panaudoti veiklos taisyklių valdymo modelį IS kūrimo procese, pateiktas modelis turi būti detalizuojamas. Sąvokos ir fakto veiklos taisyklių tipai yra jau nedalomos veiklos taisyklės. Detalizavimo reikalauja tik veiksmo teiginys(3.2 pav.), kuris yra padalinamas į 5 žemesnio lygio VT tipus [5, 10]: veiksmą valdančios, veiksmą veikiančios, leidimo, matematinio skaičiavimo ir loginės išvados. Pateiktų potipių aprašai pateikti 3.3 lentelėje.



3.2 pav. Detalizuotas „veiksmo teiginys“

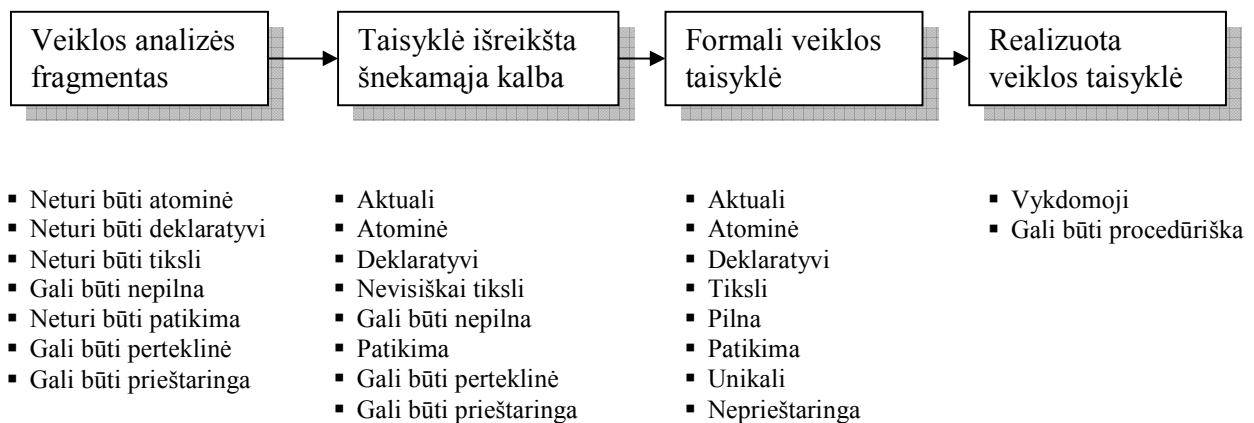
3.3 lentelė

Detalizuotų veiklos taisyklių tipų paaiškinimai

Taisyklės tipas	Taisyklės tipo aprašas	Taisyklės tipo pavyzdys
Sąvoka	Žodis ar frazė, kuris turi specifinę reikšmę veiklai.	Klientas. Kliento amžius. Darbo savaitė.
Faktas	Parodo asociaciją tarp dviejų ar daugiau sąvokų - išreiškia ryšį. Faktas įtraukia dvi ar daugiau sąvokų, o sąvoka gali būti viename ar daugiau faktų.	Klientas gali pateikti užsakymą. Klientas gali pateikti užsakymą darbo dienomis.
Veiksmą valdanti	Aprašo, kas privalo ar neprivalo būti ar įvykti.	Užsakymo vertė <i>negali</i> būti didesnė už kliento kreditą. Išsiuntus prekes klientas <i>negali</i> atšaukti užsakymo.
Veiksmą veikianti	Nusako, kas turėtų ar neturėtų būti ar įvykti.	Užsakymo vertė <i>neturėtų</i> būti didesnė už kredito limitą.
Leidimas	Teiginys, kuris aprašo prerogatyvą ar privilegiją vienos ar daugiau konstrukcijų atžvilgiu. Šį teiginį galima atvaizduoti predikatu '(Tiktai) X gali daryti Y', kur X paprastai yra vartotojas, o Y yra veiksmas, kuris gali būti vykdomas.	Jei kliento kreditas viršytas, tai užsakymų vykdymą sustabdyti. Jei klientas nepatvirtina užsakymo per 10 dienų, tai pranešti apie užsakymo užklausos panaikinimą.
Matematinis skaičiavimas	Matematinis skaičiavimas pateikia išvestinius faktus, priklausomai nuo aprašyto matematinio algoritmo.	Užsakymo kaina apskaičiuojama susumavus užsakymo vienetų kainas ir pridėjus transportavimo išlaidas.
Loginė išvada	Loginė išvada pateikia išvestinius faktus, naudojantis logine indukcija (iš konkrečių elementų) ar dedukcija (iš bendrų elementų).	Jei klientas yra užsakęs daugiau nei 20 prekių, tai jam taikoma 5% nuolaida.

3.4. Detalizuotas veiklos taisyklių formalizavimo modelis

Veiklos taisyklės gali būti atvaizduotos keturiais būdais (3.3 pav.). Kiekvienas atvaizdavimo būdas yra naudojamas skirtingose proceso stadijose ir skirtas skirtingai projekto dalyvių auditorijai. Veiklos taisyklė pradiniam etape yra išskiriama iš pateikiamos informacijos srauto – tai yra neformali veiklos taisyklės forma. Antrame etape neformalizuotai veiklos taisyklė yra išreikšta vis dar natūralia kalba, tačiau ji yra formalizuojama. Trečioji veiklos taisyklės forma yra formali ir skirta tiek veiklos atstovams, tiek projekto komandos nariams. Na, o ketvirtoji forma jau yra realizuotos pasirinktoje veiklos taisyklių saugyklos koncepcijoje forma.



3.3 pav. Veiklos taisyklių išreiškimo tipai

Surinktų veiklos taisyklių formalizavimui yra pateikiami jų išraiškai saugoti naudojami šablonai(3.4 lentelė).

3.4 lentelė

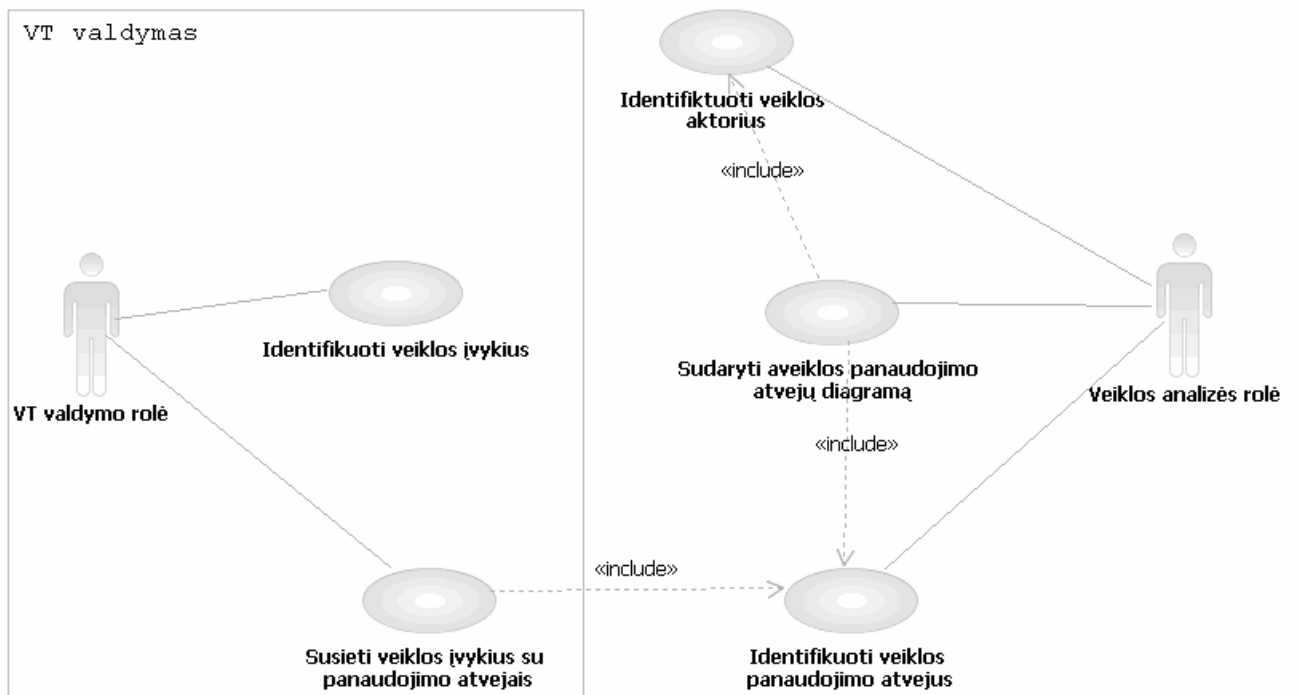
Veiklos taisyklių formalizavimo šablonai pagal tipus

Veiklos taisyklės klasifikacijos egzempliorius	Detalus aprašas	Taisyklės šablonas
Sąvoka	Žodis ar frazė, kuris turi specifinę reikšmę veiklai. Sąvokos gali būti: <ul style="list-style-type: none"> ▪ klasė, esybė; ▪ atributas, savybė; ▪ reikšmė; ▪ reikšmių rinkinys. 	< sąvoka > REIŠKIA < tekstą >;
Faktas	Parodo asociaciją tarp dviejų ar daugiau sąvokų - išreiškia ryšį. Faktas įtraukia dvi ar	< sąvoka1 > YRA < sąvoka2 >; < sąvoka1 > YRA SUDARYTA IŠ < sąvoka2 >;

Veiklos taisyklės klasifikacijos egzempliorius	Detalus aprašas	Taisyklės šablonas
	daugiau sąvokų, o sąvoka gali būti viename ar daugiau faktų. Faktai gali būti: <ul style="list-style-type: none"> ▪ ryšys tarp esybių; ▪ atributas-esybė ryšys; ▪ supertipas-supertipas ryšys. 	<sąvoka3>... ; <sąvoka1> ATLIEKA <sąvoka2> ROLE; <sąvoka2> YRA <sąvoka1>. ATRIBUTAS;
Veiksmą valdanti	Aprašo, kas privalo ar neprivalo būti ar įvykti.	<sąvoka1> TURI TURĖTI DAUGIAUSIA/MAŽIAUSIA BENT N <sąvoka2> EGZEMPLIORIŲ; <sąvoka1> YRA <a, b, c> SĄRAŠE; JEI <taisyklė> TAI <bet kuri veiksmą valdanti taisyklė>.
Veiksmą įtakojanti	Nusako, kas turėtų ar neturėtų būti ar įvykti.	<sąvoka1> GALI TURĖTI DAUGIAUSIA/MAŽIAUSIA BENT N <sąvoka2> EGZEMPLIORIŲ; <sąvoka1> GALI BŪTI <a, b, c> SĄRAŠE; JEI <taisyklė> TAI <bet kuri veiksmą įtakojanti taisyklė>.
Leidimas	Teiginys, kuris aprašo prerogatyvą ar privilegiją vienos ar daugiau konstrukcijų atžvilgiu.	JEI <sąvoka 1> <operatorius> <sąvoka 2> VYKDOMAS <veiksmas>.
Matematinis skaičiavimas	Matematinis skaičiavimas pateikia išvestinius faktus, priklausomai nuo aprašyto matematinio algoritmo.	<sąvoka1> yra apskaičiuojama <matematinė formulė>;
Loginė išvada	Loginė išvada pateikia išvestinius faktus, naudojantis logine indukcija (iš konkrečių elementų) ar dedukcija (iš bendrų elementų).	JEI <sąvoka 1> <operatorius> <sąvoka 2, reikšmė, reikšmių aibė> TAI <sąvoka 3> <operatorius> <sąvoka 4>, kur operatorius gali būti: =, <, <=, >=, priklauso, nepriklauso, ir t. t.

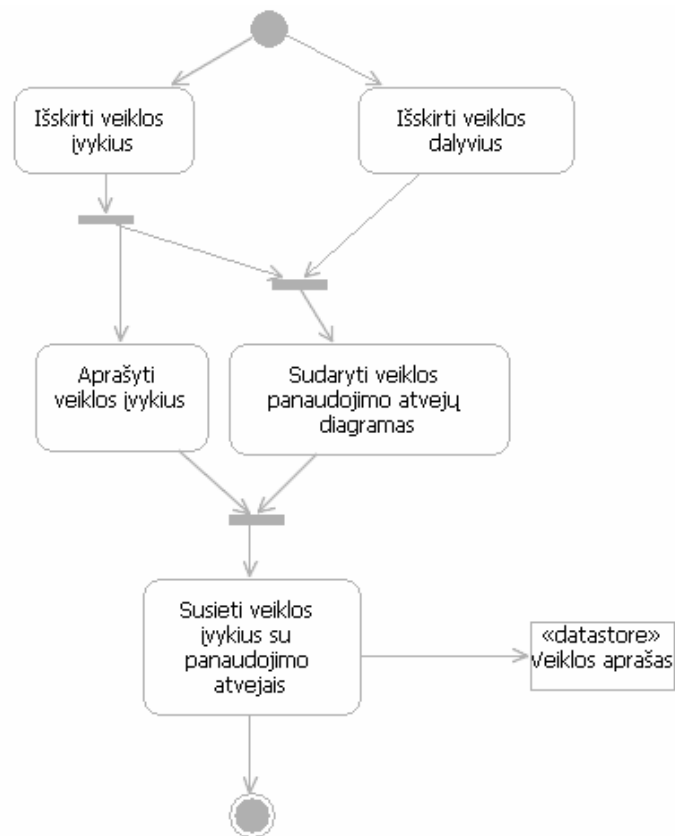
3.5. VT įtaka vizijos formavimo fazei

Informacijos išgavimo ir analizės fazėje yra surenkama, analizuojama ir tikslinama iš įvairių su veikla susijusių šaltinių gaunama informacija, kol nusprendžiama, kad surinktos informacijos kiekis yra pakankamas pereiti prie kitų projekto fazių [2, 23]. Pagrindiniai šios fazės rezultatai yra veiklos panaudojimo atvejai. Veiklos analizės rolė išskiria kuriamos IS vartotojus, bei identifikuoja veiklas, kurias reikia kompiuterizuoti. Veiklos valdymo rolė savo ruožtu pagal išskirtus panaudojimo atvejus identifikuoja įvykius [10], kurie sukuria panaudojimo atvejų kompiuterizavimo poreikį (3.4 pav.) bei susieja išskirtus veiklos įvykius su konkrečiais panaudojimo atvejų egzemplioriais (3.5 pav.).



3.4 pav. VT įtaka pasiruošimo fazės veiklos panaudojimo atvejų sudarymo metodikai

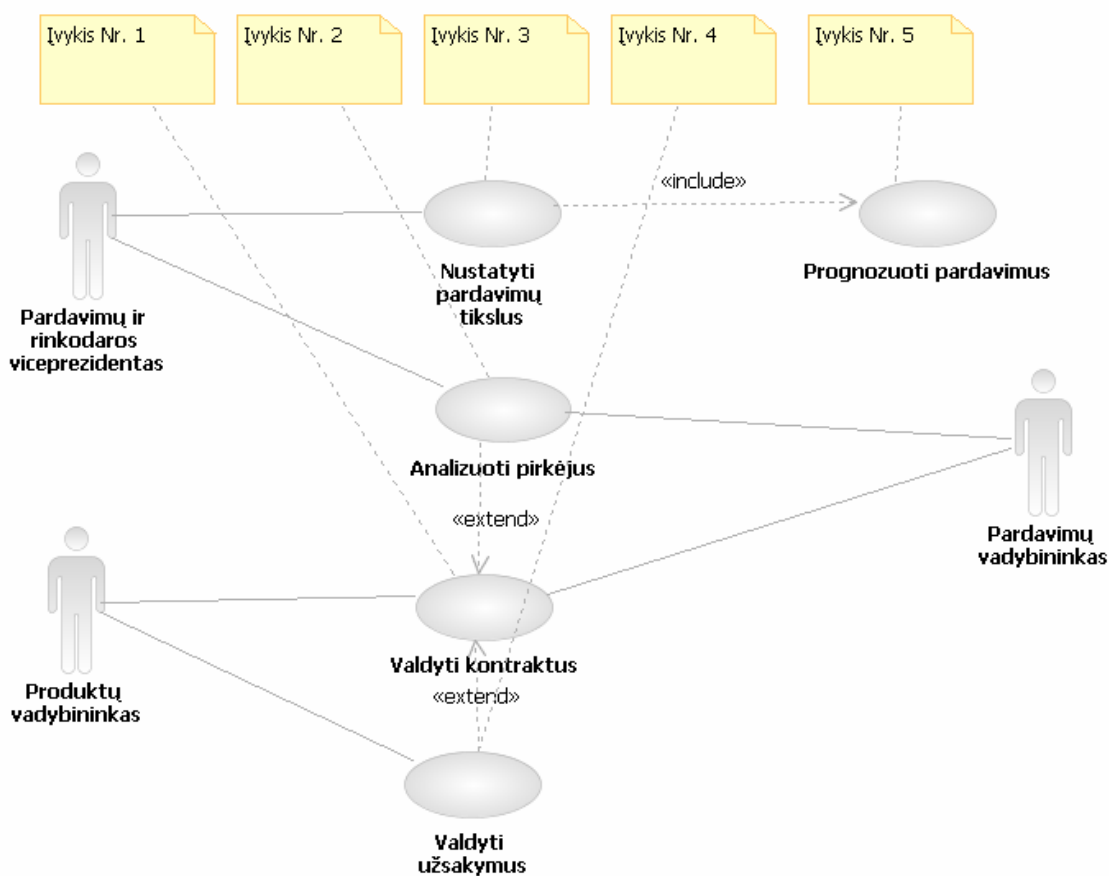
Iš 3.4 paveikslėlio galima matyti, kad veiklos taisyklių požiūris tik papildo standartinį MSF panaudojimo atvejų atradimo procesą. VT požiūrio integracija reikalauja, kad panaudojimo atvejai būtų išskiriami atrandant veiklos įvykius, arba jei yra pasirenkamas kitoks kelias, tai veiklos įvykiai turi būti atrasti atskirai nuo panaudojimo atvejų ir susieti specifikacijos sudarymo galutinėje stadijoje (3.5 pav.).



3.5 pav. VT įtakotas pasiruošimo fazės veiklos panaudojimo atvejų sudarymo procesas

Pateikto modifikuoto panaudojimo atvejų sudarymo proceso veiklų priskirti vienai ar kitai rolei neįmanoma, kadangi nurodytas veiklas atlikti gali tiek VT valdymo rolė, tiek veiklos analizės rolė, arba abi kartu. Taipogi rolių suderinamumo matrica (3.2 lentelė) nedraudžia pateiktų rolių realizuoti vienam komandos nariui.

3.5 paveikslėlyje yra iliustruojamas veiklos taisyklių sujungimo su išskirtais panaudojimo atvejais loginis vaizdas. Sudarytame kuriamos sistemos apimties dokumente veiklos įvykiai yra aprašomi (3.5 lentelė) ir saugomi siekiant sekančiose projekto fazėse išskirti iškeliamas veiklos taisykles [10].



3.6 pav. Veiklos įvykių susiejimo su veiklos panaudojimo atvejais konceptuali diagrama

3.5 lentelė

Veiklos įvykio aprašas

Veiklos įvykis	Aprašas
Pardavimų ir rinkodaros prezidentas nustato pardavimų tikslus	Remiantis ankstesniojo periodo pardavimų rezultatais bei prognozuojama rinkos situacija sekančiam periodui pardavimų ir rinkodaros viceprezidentas nustato pardavimų tikslus sekančiam periodui.
Įvykio tipas	Vidinis įvykis.
Įvykį iššaukiantys aktoriai	Pardavimų ir rinkodaros viceprezidentas; Pardavimų vadybininkas (pateikia individualias pardavimų prognozes).
Periodiškumas	4 kartus per metus.

Sėkmingai pabaigę vizijos formavimo fazę mes turime vizijos/apimties dokumentą, kuris apima:

- problemos apibūdinimą;
- kuriamos IS viziją;
- detalizuotus kuriamos IS vartotojų profilius;
- preliminarią projekto apimtį;

- kuriamos aplikacijos koncepcija;
- projekto tikslus:
 - veiklos tikslus;
 - architektūrinius tikslus;
- kritinius sėkmės faktorius;
- preliminarų projekto grafiką;

Galima pastebėti, kad veiklos panaudojimo atvejai nėra įtraukti į vizijos/apimties dokumentą, tačiau šiame etape panaudojimo atvejai atlieka tik apimties ir vartotojų profilių išsiaiškinimo ir formavimo vaidmenį. Panaudojimo atvejų svarba išryškės tik kituose IS kūrimo etapuose [5].

3.6. Veiklos taisyklių įtaka IS konceptualiosios architektūros specifikacijai

Konceptualiosios architektūros specifikacija yra planavimo fazės pirminis etapas [5, 10]. Šioje fazės dalyje yra sudaromi detalūs veiklos panaudojimo atvejai pasinaudojant pasiruošimo fazės konceptualiaisiais veiklos panaudojimo atvejais. Konceptualioji architektūra yra sudaroma eile žingsnių [5]:

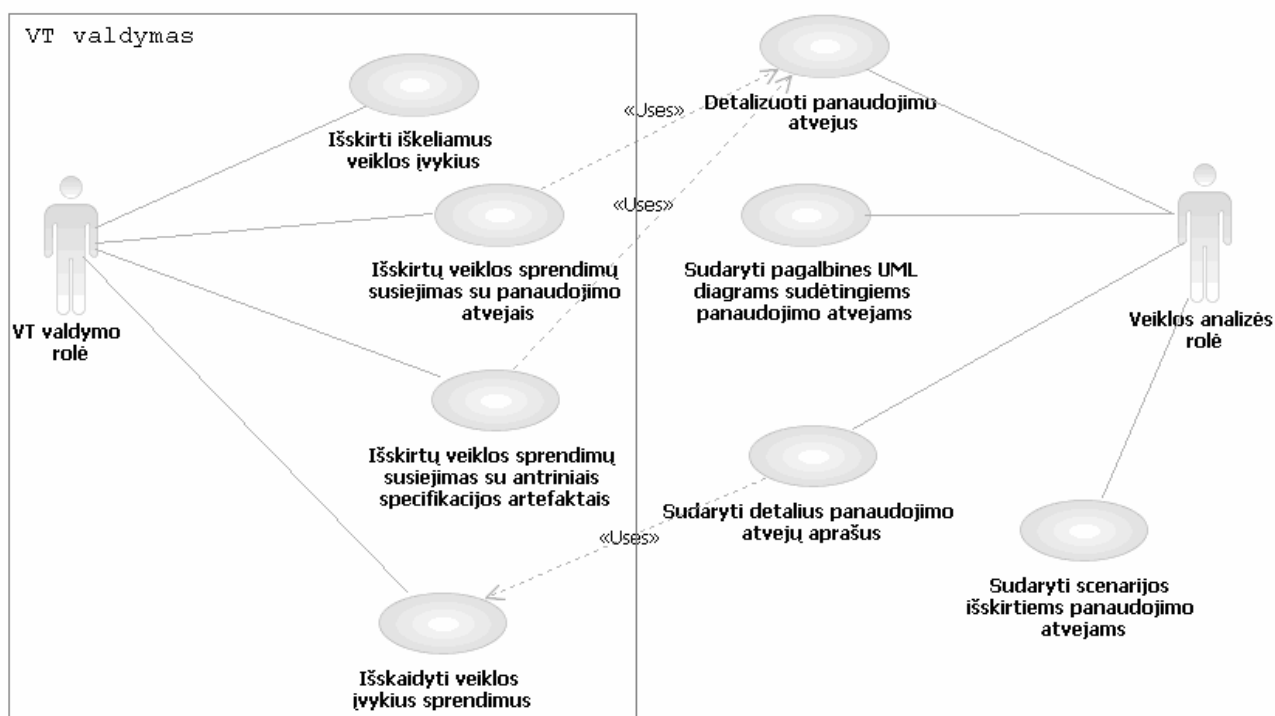
- panaudojimo atvejų detalizavimas;
- detalių panaudojimo atvejų aprašų sudarymas;
- konkrečių veiklos scenarijų sudarymas išskirtiems panaudojimo atvejams;
- pagalbinių UML diagramų (sekų, veiklų) sudarymas sudėtingiems panaudojimo atvejams.

Pateikta veiksmų seka yra būdinga daugumai objektiškai orientuotų projektavimo metodikų ir integruojant veiklos taisykles nekinta. Tačiau veiklos taisyklių požiūrio integracija reikalauja iš pateiktų panaudojimo atvejų išskirti panaudojimo atvejus, ar jų fragmentus, kuriuos yra siekiama iškelti (*externalize*). Taipogi, išskirtoms veiklos taisyklėms aprašyti ir paruošti panaudojimui sekančiose projekto fazėse yra reikalinga:

- veiklos taisyklių analizė detalizuojant veiklos įvykius į veiklos sprendimus – svarbu atkreipti dėmesį, kad tik įvykiai, kurie yra susiję su konkrečiais sprendimais yra tinkami išskirti;
- taisyklių analizavimas koreliuojant sprendimus į veiklos kontekstą (veiklos taisykles, tikslus, sprendimus ir t.t.) – svarbu užtikrinti, kad sprendimai yra priimami neatsiejamai nuo veiklos konteksto. Taipogi susiejimas padeda nustatyti konkrečios veiklos taisyklės tipą;

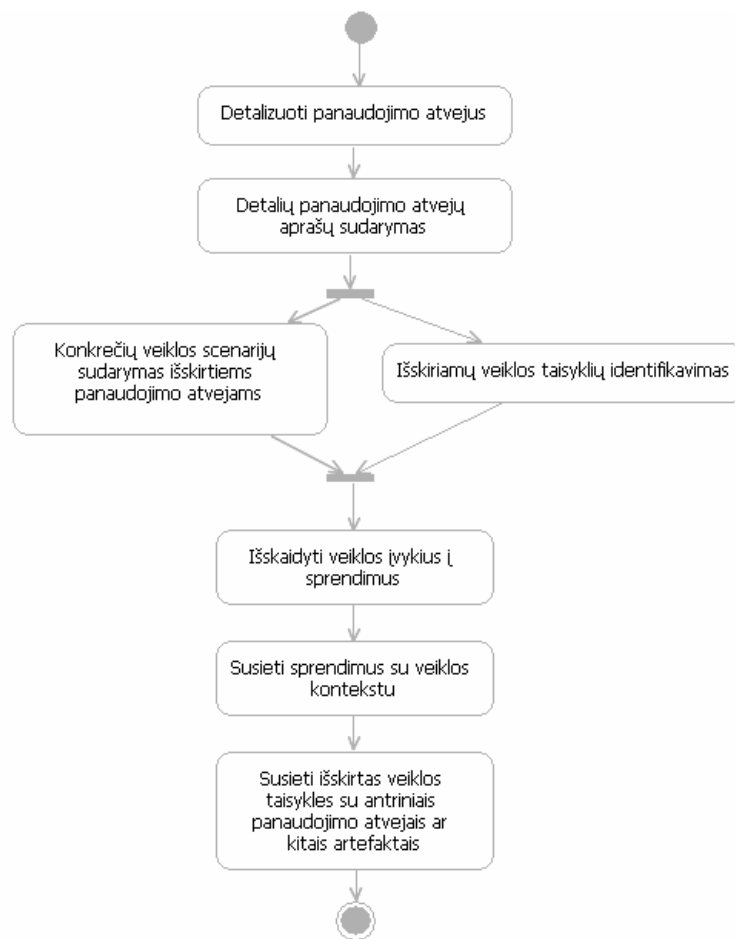
- sprendimų ir taisyklių susiejimas su konkrečiais panaudojimo atvejais – užtikrina specifikacijos vientisumą ir galimų desinchronizacijos atvejų atradimą ir pašalinimą;
- išskirtų veiklos taisyklių sąsajų su antriniais panaudojimo atvejais ar kitais artefaktais atradimas – kitose specifikacijos sudarymo fazėse garantuos veiklos taisyklių atsekamumą ir palengvins galimą poreikių pasikeitimo valdymo procesą;

Konceptualios specifikacijos fazėje analizės rolės(MSF projekto valdymo rolė) ir veiklos taisyklių valdymo rolės vykdomos veikos atvaizduotos 3.7 paveiksle.



3.7 pav. Siūlomas konceptualios specifikacijos sudarymo proceso modelis

Bendras veiklos taisyklių valdymo procesas konceptualios architektūros sudaryme pavaizduotas 3.8 diagramoje. Veiklų priskyrimas konkrečios rolėms yra negalimas, kadangi VT valdymo rolė ir Veiklos analizės rolė yra ne konkuruojančios, o bendradarbiaujančios ir kiekvienos veiklos vykdyme dalyvauja abi pateiktos rolės.



3.8 pav. Siūlomas konceptualios architektūros sudarymo proceso modelis

Išskirtų veiklos taisyklių nurodyti artefaktai yra detalizuojami ir aprašomi prie konkrečių veiklos panaudojimo atvejų, tačiau veiklos taisyklės dar nėra identifikuojamos kaip atskiros esybės. Taipogi pateikto modelio skirtumas palyginus su standartiniais veiklos taisyklių aprašais yra veiklos panaudojimo atvejų žingsnių susiejimas su sprendimais (3.6 lentelė).

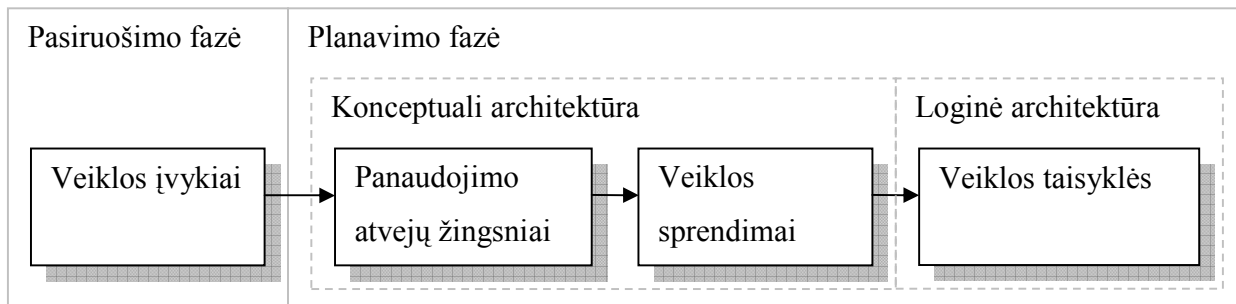
3.6 lentelė

Veiklos panaudojimo atvejų ir sprendimų sąsaja

Panaudojimo atvejo žingsnis	Sprendimas	Taisyklė
1. Klientas atidaro interneto svetainę.	-	
2. Klientas įveda autorizacijos duomenis.	-	
3. Sistema autorizuoja vartotoją.	Ar vartotojo duomenys priimti?	
4.	

Svarbu pastebėti, kad pateiktame panaudojimo atvejo sekos fragmente veiklos taisyklės dar nėra identifikuojamos, kadangi veiklos taisyklės apims tik tam tikrus panaudojimo atvejo fragmentus ir šiame

Šiame proceso etape galutinai išaiškėja pasiruošimo fazėje ir konceptualiosios architektūros sudarymo fazėje atliktų veiksmų nauda ir reikšmė veiklos taisyklių modelio sudarymui (xx paveikslas).



3.10 pav. Pasiruošimo ir planavimo fazėse atliekamų veiksmų nauda ir reikšmė VT modelio sudarymui

Išskiriamų sprendimų identifikavimas nėra naujas veiksmas, o tik konceptualiosios specifikacijos sudarymo fazėje išskirtų sprendimų verifikavimas su klientu ir jų aibės papildymas ar nustatytų veiklai nesvarbių sprendimų pašalinimas. Likusiems sprendimams yra sudaromos veiklos taisyklės ir įvedamos į pasirinktą veiklos taisyklių saugyklą, kur bendrinis veiklos taisyklių formalizuotas modelis pavaizduotas 3.7 lentelėje.

3.7 lentelė

Veiklos panaudojimo atvejų ir sprendimų sąsaja

Taisyklės identifikatorius	Taisyklės aprašas	Taisyklės kategorija	Susiję objektai
VT01	Vartotojo identifikatorius turi būti autorizuotų vartotojų sąraše.	Veiksmą valdanti	Atributas vartotojo identifikatorius, esybė vartotojai.
VT02	Vartotojo slaptažodis turi atitikti pateiktą identifikatorių	Veiksmą valdanti	Esybė vartotojas, VT01.

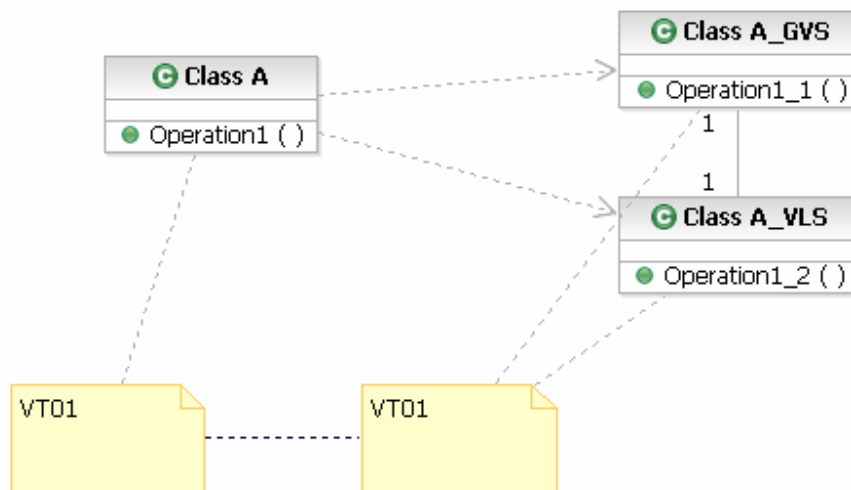
Sekantis veiklos taisyklių aprašas mums pateikia visą būtiną informaciją veiklos taisyklių valdymui:

- kiekviena veiklos taisyklė turi unikalų identifikatorių, kuris leidžia išskirtas veiklos taisykles susieti su konkrečiais specifikacijos artefaktais;
- veiklos taisyklės aprašas yra formalizuotas pagal 3.4 skyrelyje pateiktą šabloną. Toks aprašas leidžia lengvai transformuoti veiklos taisykles į konkretaus VT variklio metakalbą;
- taisyklė yra priskirta konkrečiai kategorijai pagal pasirinktą modelį. Tai palengvina taisyklių valdymą ir realizaciją;
- veiklos taisyklėms yra priskirti konkretūs objektai. Tai yra objektai, kurie yra reikalingi, kad veiklos taisyklė galėtų būti realizuota ir galėtų funkcionuoti kuriamoje sistemoje. Verta

pastebėti, kad tai nėra veiklos artefaktai, kuriuos veiklos taisyklė valdo, o sistemos objektai, kurie yra būtini, kad veiklos taisyklė teisingai funkcionuotų.

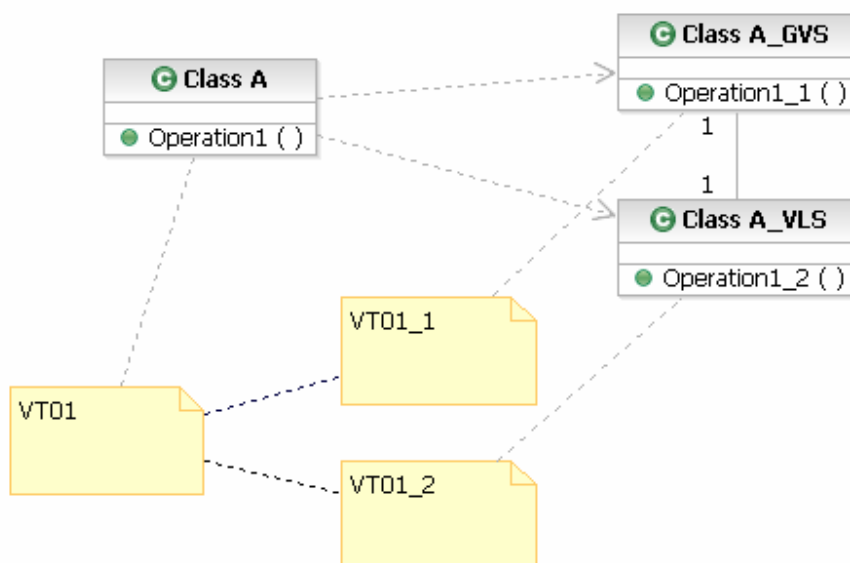
3.8. Veiklos taisyklių įtaka IS fizinės architektūros specifikacijai

Fizinės architektūros specifikacijos sudaryme yra transformuojami loginės architektūros artefaktai ir pritaikomi konkrečiai realizacijos platformai ir specifikai. Šioje fazėje veiklos taisyklių modelis nekinta. Gali kisti tik susiję objektai. Jei tam tikros esybės yra išskaidomos saugumo, našumo, lankstumo sumetimais, tai atitinkamai turi pakisti ir veiklos taisyklių susiejimas su išskaidyto objekto dalimis. Konkretūs sprendimo variantai priklauso nuo konkretaus atvejo, tačiau svarbu išlaikyti veiklos taisyklių modelio ir IS specifikacijos integralumą. Paveiksluose Nr. 3.11 ir 3.12 pateikti keli galimi pakeitimų variantai, tačiau gali būti įvairūs architektūriniai sprendimai, kuriuos visus numatyti yra praktiškai neįmanoma.



3.11 pav. Veiklos taisyklės sąsajos su specifikacijos artefaktu adaptavimo pavyzdys Nr. 1

Kaip iš pateikto pavyzdžio matome, loginės architektūros klasė „Class A“ buvo išskaidyta į dvi klases, kur kiekviena iš klasių realizuoja dalį klasės „Class A“ serviso „Operation1“ funkcionalumo. Atitinkamai ir veiklos taisyklė, kuri riboja ar kitaip veiks serviso „Operation1“ veikimą įvykdyto pakeitimo kontekste ribos ar kitaip veiks „Operation1_1“ ir „Operation1_2“ funkcionalumo. Pateiktam pavyzdžiui galimi ir kiti sąsajos pakitimo variantai, kaip pvz. veiklos taisyklės „VT01“ įtakojamas funkcionalumas bus perkeltas tik į vieną iš transformuotos specifikacijos klasių ar kt.



3.12 pav. Veiklos taisyklės sąsajos su specifikacijos artefaktu adaptavimo pavyzdys Nr. 2

Kaip ir 3.11 paveiksle pateiktame pavyzdyje loginės architektūros klasė „Class A“ buvo išskaidyta į dvi klases, kur kiekviena iš klasių realizuoja dalį klasės „Class A“ serviso „Operation1“ funkcionalumo. Tačiau pagal funkcionalumą ir veiklos taisyklė buvo išskaidyta į dvi atskiras veiklos taisykles, kurios atitinkamai veikia skirtingus išskaidytos operacijos „Operation1“ funkcionalumo aspektus. Pateiktas variantas yra galimas, tačiau labai retais atvejais, kadangi išskirtos veiklos taisyklės turi būti atominės. Žinoma tam tikrais sumetimais (bereikalingo kompleksiško vengimo, dalykinės srities specifikos) pateikta situacija yra galima, tačiau vengtina ir gali pasitarnauti kaip veiklos taisyklių modelio verifikavimo bei tobulinimo mechanizmas.

3.9. Veiklos taisyklių valdymo modeliai

Veiklos taisykles realizuoti galima 3-mis būdais [10]:

- veiklos taisyklių saugojimas IS specifikacijoje;
- veiklos taisyklių variklis realizuotas kaip specializuotas aplikacijos modulis;
- veiklos taisyklės saugomos specializuotame VT valdymo produkte.

Šių veiklos taisyklių valdymo modelių palyginimas pagal pasirinktus kriterijus yra pateiktas 3.8 lentelėje.

VT valdymo modelių palyginimas

VT teikiami privalumai	IS specifikacija	Specializuotas aplikacijos modulis	Specializuotas VT produktas
Atsekamumas	Taip	Taip	Taip
Keičiamumas	Iš dalies	Taip	Taip
Versijų valdymas	Ne	Iš dalies	Taip
Realizacijos kompleksiskumas	Paprasta	Sudėtingesnė	Sudėtingiausia
Verifikavimas	Ne	Ne	Taip
Pasiekiamumas iš įvairių IS	Ne	Ne	Taip

Kai veiklos taisyklės yra pateikiamos tik kuriamos IS specifikacijoje:

- atsekamumo faktorius priklauso tik nuo sukurtos specifikacijos kokybės ir jei veiklos taisyklių modelio integralumas su pateikta specifikacija yra įvykdytas, tai atsekamumas yra garantuojamas;
- veiklos taisyklių keičiamumas yra pagerinamas dėl atsekamumo faktoriaus. T. y. siekiant pakeisti vieną ar kitą veiklos taisyklę, galima atsekti pakeitimo įtaką sukurtai IS ir realizuoti norimus pakeitimus trumpiausiu keliu ir užtikrinant visų susijusių komponentų adaptaciją pasikeitusiai situacijai;
- versijų valdymas nurodytu atveju yra įmanomas, tačiau sudėtingas ir mažai tikėtinas dėl atsirandančių papildomų sąnaudų;
- realizacijos kompleksiskumas yra paprasčiausias iš visų pateiktų variantų, kadangi realizacijos nesiskiria nuo bet kurio kito kuriamos IS modulio;
- VT verifikavimas yra negalimas;
- pasiekiamumas iš kitų veikloje naudojamų informacinių sistemų be specialių architektūrinių sprendimų yra negalimas.

Kai veiklos taisyklių variklis yra realizuotas kaip specializuotas IS modulis:

- atsekamumas priklauso tik nuo sukurtos specifikacijos kokybės;
- keičiamumas priklauso nuo kuriamo modulio architektūros, tačiau tai, kad kuriamas specializuotas modulis, savaime parodo, kad norima keisti veiklos taisyklių parametrus;
- versijų valdymas yra galimas tik atitinkamai suprojektavus modulį, tačiau tai prideda papildomos kompleksijos;
- realizacijos kompleksiskumas priklauso tik nuo kuriamo modulio galimybių, tačiau tai, kad kuriamas modulis yra realizuojamas tomis pačiomis priemonėmis kaip ir visa kuriama IS leidžia daryti prielaidą, kad realizacija yra nesudėtinga;

- verifikavimas yra galimas, tačiau mažai tikėtina, kad jis bus pilnas ir nuoseklus;
- pasiekiamumas iš kitų veikloje naudojamų IS yra galimas, tačiau šios savybės realizacija reikalauja papildomų sąnaudų.

Naudojant komercinį VT valdymo produktą:

- keičiamumą garantuoja VT valdymo produktas;
- versijų valdymą užtikrina dauguma rinkoje egzistuojančių VT valdymo komercinių produktų;
- realizacija daugeliu atvejų yra sudėtingesnė nei kituose dviejuose pateiktuose variantuose, kadangi veiklos taisyklės yra aprašomos specifine naudojamu VT valdymo produkto kalba ir yra reikalingi papildomi aplikacijos objektai (servisai, operacijos ir t.t.) kuriamos IS ir komercinio VT varikli sąveikai;
- VT verifikavimą užtikrina dauguma rinkoje esančių VT produktų;
- VT valdymo variklis yra pasiekiamas iš bet kurios modernios IS.

3.10. Projektavimo aplinkos pritaikymas veiklos taisyklėms

Siekiant integruoti veiklos taisykles į IS specifikaciją, reikalingi sekantys žingsniai:

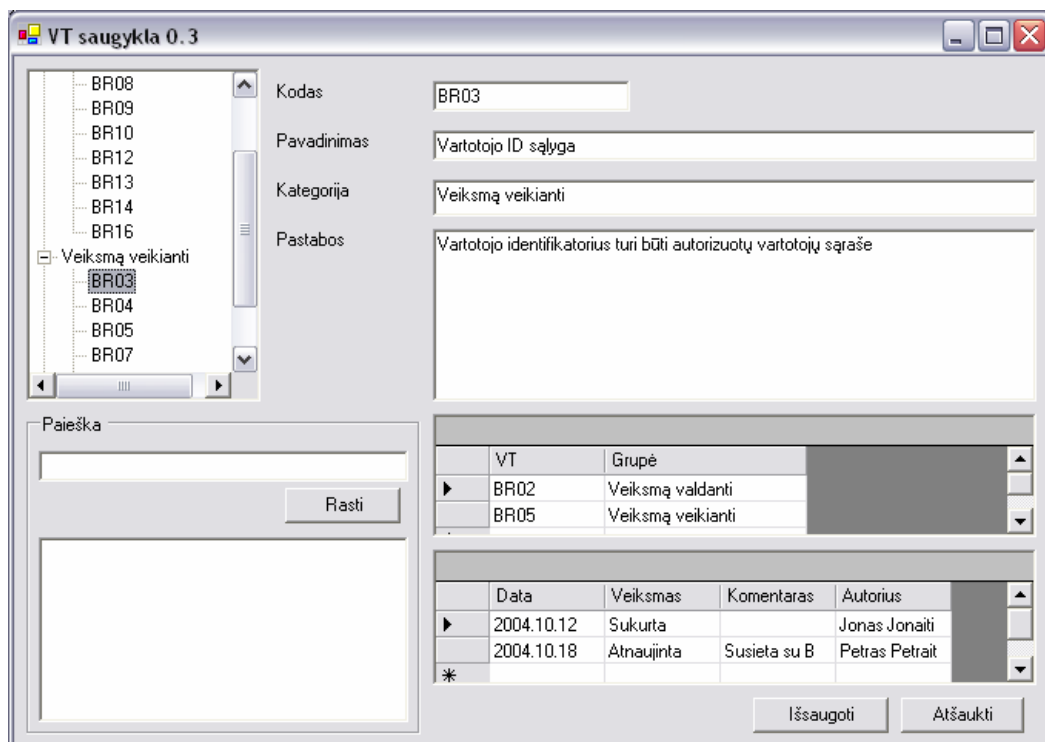
- veiklos taisyklių valdymo sistema;
- projektavimo įrankio adaptavimas veiklos taisyklėms;
- veiklos taisyklių ir sudarytos specifikacijos integracija;
- veiklos taisyklių ir išėties teksto integracija.

Siūlomos koncepcijos įvykdomumui parodyti buvo sukurta veiklos taisyklių valdymo aplikacija.

Aplikacija leidžia įvykdyti sekančius veiksmus:

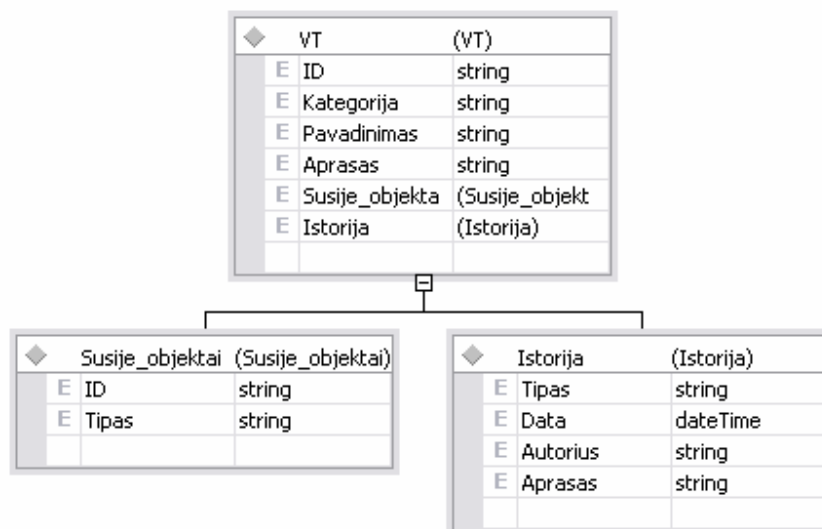
- įvesti, redaguoti bei šalinti veiklos taisykles;
- susieti veiklos taisykles tarpusavyje;
- automatiškai suteikti unikalius veiklos taisyklių identifikatorius;
- kategorizuoti ir grupuoti veiklos taisykles;
- ieškoti veiklos taisyklių pagal turinio fragmentą.

Pateiktiems tikslams pasiekti eksperimente panaudota .NET platformoje sukurta aplikacija, kuri pavaizduota 3.13 paveiksle.



3.13 pav. Eksperimentinė veiklos taisyklių valdymo IS

Veiklos taisyklės saugomos xml laikmenoje (3.14 pav.). Kaip galime pastebėti, pateiktas duomenų modelis yra pakankamai primityvus ir neleidžia saugoti išskirtų veiklos taisyklių pagal darbe pateiktą veiklos taisyklių kategorizavimo (3.3 skyrius) ir atvaizdavimo (3.4 skyrius) modelį. Tačiau pateiktas modelis yra sudarytas tik siekiant parodyti veiklos taisyklių susiejimo su IS specifikacija įvykdomumą.



3.13 pav. Eksperimentinės VT valdymo IS duomenų modelis

Sekantis žingsnis pasirinkus veiklos taisyklių valdymo modelį yra susieti jas su IS kūrimo procesu. Pagal esamą UML specifikaciją ir notaciją, veiklos taisyklės gali būti integruotos į norimas UML diagramas kaip komentarai, diagramų dokumentacija, arba OCL sintakse paremtas kodas. Tačiau nurodyti būdai (išskyrus OCL) yra tik kaip išvestiniai metodai, kuriuos galima panaudoti tik neformalios UML notacijos dėka. Šiuo atveju veiklos taisyklių valdymas yra ypač sudėtingas, kadangi pasikeitus specifikacijai, veiklos taisyklės gali tekti keisti ar atnaujinti skirtingose specifikacijos vietose ir jų valdymas savaime sukuria papildomų sunkumų ir darbo, kurį CASE priemonės gali ir turi automatizuoti.

Alternatyvus ir pasirinktas variantas yra pasinaudojant UML 2.0 specifikacijos savybe „išplėstinė reikšmė“ (*tagged value*) [8], veiklos taisyklių unikalius kodus susieti su konkrečiais UML diagramų artefaktais. Tai įgalina gauti formalų metodą veiklos taisyklių susiejimui su norimais specifikacijos elementais, kas leidžia atsieti veiklos taisyklių turinį nuo specifikacijos ir žymiai palengvina jų valdymą.

Eksperimentui buvo pasirinktas „Rational Rose XDE 2003“ CASE įrankis, kuris pasižymi sekančiomis eksperimentui būtinomis savybėmis:

- UML 2.0 standarto palaikymas;
- „Išplėstinių reikšmių“ kūrimo galimybės;
- Modelių analizės ir redagavimo programavimo sąsaja (RXE API);
- MDA (modeliu paremta architektūra) [15] metodikos palaikymas.

Siekiant esamą CASE įrankį pritaikyti VT požiūriui, artefaktams, su kuriais yra prasminga susieti veiklos taisyklės buvo sukurta papildoma „išplėstinė reikšmė“ (2 pav.).

Galimybė išplėstines reikšmes priskirti bet kuriam UML diagramų aibės artefaktui leidžia atsekti konkrečios veiklos taisyklės įtaką kuriamai sistemai. Pasinaudojant diagramų analizės priemonėmis galima atsekti, pavyzdžiui, tokius specifikacijos aspektus:

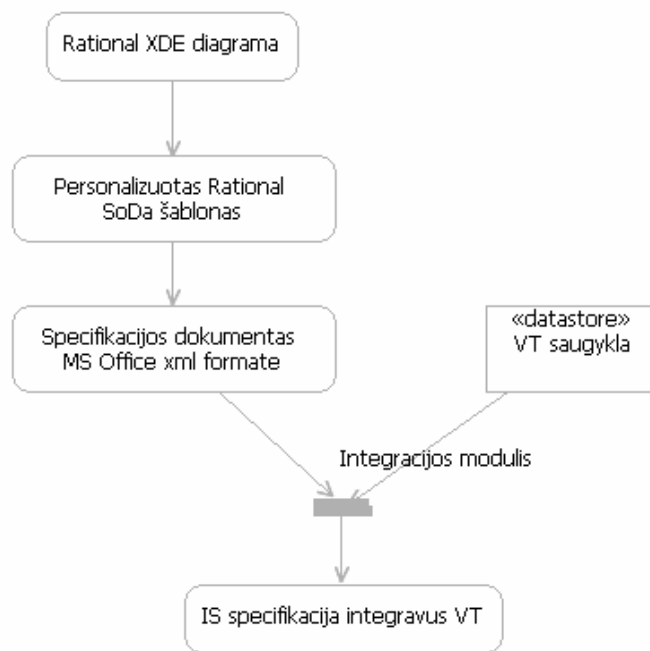
- Ar VT yra naudojama specifikacijoje;
- Kaip dažnai VT pasikartoja specifikacijoje;
- Kokiems artefaktams konkreiti VT yra priskirta.

Properties	
Uzsakymo tvirtinimas <UseCase>	
Property	Value
[-] UML	
(Name)	Uzsakymo tvirtinir
Alias	
Collections	
IsAbstract	False
IsLeaf	False
IsRoot	False
IsSpecification	False
Multiplicity	
Persistence	0 - PERSISTENT
Stereotype	
Visibility	0 - PUBLIC
Veiklos Taisyklės	BR01, BR02
[-] View	
Height	1100
Y	2000
SortAttributes	1 - Storage Order
SortOperations	1 - Storage Order
Width	5574
WordWrapTextWidth	6000
X	8000

3.14 pav. Panaudojimo atvejo išplėstinė reikšmė

Kadangi veiklos taisyklės yra integruojamos kaip UML diagramos artefaktų savybės, tai standartinėje specifikacijos dokumentacijoje šios savybės neregistruojamos. Integruoti veiklos taisykles su kuriamos IS specifikacija, galima keliais būdais:

- Rational SoDa – Rational paketo įrankis, leidžiantis kurti generuojamos specifikacijos šablonus pasinaudojant modelio struktūros elementais. Susikūrus personalizuotą šabloną, veiklos taisyklės figūruos specifikacijos dokumentuose kartu su susijusiais artefaktais. Taigi sugeneruotoje dokumentacijoje prie modelių artefaktų figūruotų veiklos taisyklių unikalūs identifikatoriai (BR01, BR02...).
- Pasinaudojant Rational SoDa dokumentacijos šablonų redagavimo galimybėmis, bei MS Office paketo galimybe kurti dokumentus XML formatu, integruoti veiklos taisyklių saugykloje sukauptą informaciją su sukurta sistemos specifikacija (3.15 pav.).



3.15 pav. IS specifikacijos ir VT saugyklos integracijos schema

Viena iš labiausiai deklaruojamų veiklos taisyklių naudų yra galimybė adaptuoti jas prie besikeičiančios verslo aplinkos ir poreikių. Veiklos taisyklėmis paremtuose produktuose tai yra pasiekama panaudojant veiklos taisyklių saugyklas, kuriose netgi veiklos atstovas gali atlikti reikiamas korekcijas ir stebėti IS veikimą. Realizuojant IS be veiklos taisyklių saugyklos šis pranašumas yra prarandamas, tačiau, pasinaudojant moderniomis IS kūrimo priemonėmis, šį trūkumą nors ir ne visada galima panaikinti, tačiau galima žymiai supaprastinti produkto pritaikymo prie besikeičiančios verslo aplinkos procesą, tuo sumažinant sąnaudas ir sutrumpinant atnaujintos versijos pateikimo vartotojui laiką. Tai galima pasiekti Rational XDE kodo generavimo priemonių lankstumo pagalba. Į Rational XDE paketą yra integruotas lankstus MDA (*model driven architecture*) modulis, kuris leidžia redaguoti kodo generavimo šablonus. Pritaikius šablonus savo poreikiams, su artefaktais saugomus veiklos taisyklių unikalius identifikatorius galima perkelti į IS išeities kodą:

```

/// <summary>
/// Method recalculates price according to users previous sales
/// </summary>
///
/// <TABLE>
/// Implemented Business Rules
/// -----
/// BR001
/// BR002
/// </TABLE>
  
```

```
/// <param name="price">price to apply discount to</param>
/// <param name="uid">system user id</param>
public void PritaikytiNuolaida(float price, long uid)
{}
```

Veiklos taisyklės logikos realizacija vis dar priklauso nuo programuotojų, tačiau vykdant veiklos taisyklių modifikacijas, netgi pasitelkus tokias standartines priemones kaip VT identifikatorių paiešką kode, galima garantuoti, kad pasikeitusios VT logika bus atnaujinta visame pakete.

4. EKSPERIMENTINĖ DALIS

4.1. Tikslas

Pademonstruoti veiklos taisyklių susiejimo su IS specifikacija realizacijos siūlymą ir išskirti teikiamo siūlymo teikiamas naudas ir trūkumus.

4.2. Realizacija

Eksperimentiniame tyrime yra pateikiamas standartinės MSF specifikacijos fragmentas ir specifikacijos, sudarytos pagal pateiktą siūlymą fragmentas. Visos specifikacijos pateikimas yra negalimas dėl sistemos specifikacijos apimties, kadangi analizuojamos dalykinės srities specifikacija pagal MSF proceso modelį sudaro 254 puslapius. Taigi yra pateikiamas vizijos/apimties dokumentas, sistemos panaudojimo atvejų diagramos ir 5 charakteringi detalizuoti panaudojimo atvejai. Panaudojimo atvejų diagramų dokumentas yra sugeneruotas Rational SoDa dokumentacijos generavimo priemone, o panaudojimo atvejų detalių aprašų šablonai yra sugeneruoti tik įtraukus diagramose esančius artefaktus:

- panaudojimo atvejo identifikatorius;
- scenarijaus aprašymas;
- aktoriai;
- naudoja/išplečia kvalifikatoriai.

Pagal MSF reikalavimus sugeneruotas detalizuotas panaudojimo atvejo aprašas pavaizduotas 4.1 paveiksle.

Use Case ID: UC 01.01.01

Title: UC 01.0.01 Prisijungti
Abbreviated Title: Prisijungti
Requirement ID:

Tikslas

Scenarijaus Aprašymas

Vartotojas nori pasinaudoti sistema. Tam jis turi prisijungti prie jos naudodamasis savo prisijungimo vardą ir slaptažodį.

Prielaidos/Pradinės sąlygos

Aktoriai

1. Vartotojas

Veiksmų Seka

Alternatyvios Veiksmų Sekos

Naudoja/Išplečia

1. Naudoja. UC 01.01.02 Gauti pamirštą slaptažodį.
2. Naudoja. UC 01.01.03 Registruotis.

Vartotojo Pageidavimai Realizacijai

Dažnumas

Atsakingi

- 4.1 Sugeneruotas detalus MSF šablono panaudojimo atvejo aprašas

Generuojant specifikaciją su modifikuotu „Rational SoDa“ šablonu yra pridedama papildoma „Veiklos Taisyklės“ kategorija, kur įrašomi veiklos taisyklių identifikatoriai, kurie buvo susieti su modeliu „išplėstinių reikšmių“ pagalba (4.2 pav.).

Use Case ID: UC 01.01.03

Title: UC 01.0.03 Registruotis

Abbreviated Title: Registruotis

Requirement ID:

Tikslas

Scenarijaus Aprašymas

Vartotojas prisijungia prie sistemos, kad galėtų pateikti naują užsakymą, peržiūrėti užsakymų istoriją. Tam reikia turėti savo prisijungimo duomenis, registravimosi metu vartotojas suveda duomenis apie save ir susikuria savo prisijungimo duomenis.

Prielaidos/Pradinės sąlygos

Aktoriai

1. Klientas

Veiksmų Seka

Alternatyvios Veiksmų Sekos

Naudoja/Išplečia

Vartotojo Pageidavimai Realizacijai

Veiklos Taisyklės

BR02
BR05

VT identifikatoriai

Dažnumas

Atsakingi

4.2 Sugeneruotas detalus teikiamo siūlymo panaudojimo atvejo aprašas

Pasinaudojant sukurtu Rational XDE priedu (*plugin*), xml formate išsaugotame MS Word specifikacijos dokumente yra atrandami veiklos taisyklių identifikatoriai ir papildomi veiklos taisyklių aprašais iš veiklos taisyklių saugyklos(4.3 pav.)

Use Case ID: UC 01.01.03

Title: UC 01.0.03 Registruotis
Abbreviated Title: Registruotis
Requirement ID:

Tikslas

Scenarijaus Aprašymas

Vartotojas prisijungia prie sistemos, kad galėtų pateikti naują užsakymą, peržiūrėti užsakymų istoriją. Tam reikia turėti savo prisijungimo duomenis, registravimosi metu vartotojas suveda duomenis apie save ir susikuria savo prisijungimo duomenis.

Prielaidos/Pradinės sąlygos

Aktoriai

1. Klientas

Veiksmų Seka

Alternatyvios Veiksmų Sekos

Naudoja/Išplečia

Vartotojo Pageidavimai Realizacijai

Veiklos Taisyklės

Kodas	Aprašas	Tipas	Susije objektai
BR02	Email TURI TURĖTI @ simbolį.	Veiksma valdantis	
BR05	Jei BR02 = true autorizuoti	Leidimas	BR02

Dažnumas

4.3 Detalus panaudojimo atvejo aprašas papildytas VT turiniu

Taigi pateiktas eksperimentas parodo veiklos taisyklių susiejimo su IS specifikacija procesą ir rezultata. Tai žinoma yra tik pavyzdinis realizacijos variantas ir jo veikimo įvairiems specifikacijos

šablonams garantuoti negalima, tačiau dauguma modernių projektavimo priemonių turi išplėtimo mechanizmus ir suteikia galimybę pritaikyti įrankius specializuotiems poreikiams.

4.3. Teikiamos naudos, trūkumai

Pateiktas eksperimentas išskiria sekančias naudas:

- veiklos taisyklių modelio integracija su IS specifikacija;
- esamų įrankių panaudojimas VT sąsajai su UML diagramomis;
- lankstus VT ir IS specifikacijos integracijos modelis;
- VT formalizavimo modelio atskyrimas nuo UML projektavimo įrankio;
- galimas VT modelio integralumo verifikavimas koreliuojant į specifikacijos artefaktus.

Pateiktas siūlymas neišsprendžia sekančių problemų:

- kadangi su UML artefaktais yra susiejami tik VT identifikatoriai, tai modelis savaime nesuteikia informacijos apie su artefaktu susijusios VT turinį;
- UML modelių integralumo tikrinimo įrankiai neidentifikuoja su VT susijusių problemų;
- reikalingi papildomi įrankiai VT ir programos išėties kodo sąsajai.

5. IŠVADOS

1. Išanalizavus labiausiai paplitusius veiklos taisyklių struktūrizavimo modelius, nuspręsta pasinaudoti dviejų VT struktūrizavimo modelių (GUIDE, HALLE) baze sudarant darbo specifiką tenkinantį modelį.
2. Išanalizavus labiausiai paplitusias objektiškai orientuotas projektavimo metodikas nuspręsta integruoti sudarytą VT modelį į MSF (*Microsoft Solution Framework*) procesą, kuris tenkina veiklos taisyklių integracijai apibrėžtus kriterijus, kurie pateikti 2.7 skyriuje.
3. Projektinėje dalyje pateikiamas Halle [10] siūlomo veiklos taisyklėmis paremto IS kūrimo proceso fazių integracijos į MSF projektavimo metodikos fazes variantas reikalauja minimalių MSF metodikos pakeitimų (rolės, vizijos/apimties dokumentas,)...
4. Atsižvelgiant į Halle [10] ir Ross [20] pateiktų VT paremtų projektavimo metodikų principus, yra pateikti MSF konceptualiosios, loginės bei fizinės specifikacijų pritaikymo veiklos taisyklėmis modeliai, kuriais siekiama minimaliai modifikuoti esamus modelius suteikiant maksimalią VT teikiamą naudą.
5. Darbe pateikiamas eksperimentinis veiklos taisyklių valdymo modelio ir projektavimo aplinkos integracijos tyrimas parodo, kad pateiktas proceso modelis gali būti įgyvendintas įgyvendinant realius veiklos taisyklėmis paremtus projektus.
6. Eksperimentinėje dalyje pateikta veiklos taisyklių modelio integracija į IS specifikaciją palengvina veiklos taisyklių valdymą ir naudojimą siūlomame IS kūrimo procese, tačiau eilę problemų palieka neišspręstų ir siūlomas eksperimentinis metodas komerciniams projektams realizuoti dar nėra tinkamas.

LITERATŪRA

1. Albertyn F. Comparing Possible E-Processes [žiūrėta 2005-01-12]. Prieiga per internetą: www.naccq.ac.nz/conference04/proceedings_03/pdf/189.pdf;
2. Analyzing Requirements and Defining Microsoft .NET Solution Architectures. Microsoft Press. 2003;
3. Boggs W., Boggs M. Mastering Rational XDE. Sybex Inc; 1 edition (May 23, 2003);
4. Cover Pages. Business Rules Markup Language (BRML) [žiūrėta 2003-12-10]. Prieiga per internetą: <http://xml.coverpages.org/brml.html>;
5. Business Rules Group. Defining Business Rules – What are they really? [žiūrėta 2003-12-11]. Prieiga per internetą: http://www.business-rules.com/en/White%20Papers/defining_business_rules/defining_business_rules.htm;
6. Business Rules. Powering Business and e-Business [žiūrėta 2004-03-12]. Prieiga per internetą: http://www.ilog.com/products/rules/wp_businessrules.pdf;
7. Date C. J. What Not How: The Business Rules Approach to Application Development. Addison-Wesley Professional; 1st edition (April 12, 2000);
8. Foundation package: Extension Mechanisms [žiūrėta 2005-03-28]. Prieiga per internetą: <http://etna.int-evry.fr/COURS/UML/semantics/semant6.html>;
9. Gottesdiener E., President, EBG Consulting, Inc. Business Rules Show Power, Promise [žiūrėta 2003-12-14]. Prieiga per internetą: http://www.ebgconsulting.com/powerpromise_article.htm;
10. Halle V. B., Business Rules Applied: Building Better Systems Using the Business Rules Approach, 2001. ISBN 0471412937;
11. Halle V. B. Building a Business Rules System [žiūrėta 2003-12-11]. Prieiga per internetą: <http://www.kpiusa.com/>;
12. Hansen J. E., Thomsen C. Enterprise Development with Visual Studio .NET, UML, and MSF. Apress; 1 edition (May 24, 2004);
13. Hernandez M. J. Database Design of Mere Mortals: A Hands-On Guide to Relational Database Design, Chapter 11, Business Rules [žiūrėta 2004-12-15]. Prieiga per internetą: <http://cseng.awl.com/book/related/0,3833,0201694719+12,00.html>;
14. Highsmith J. Agile Project Management: Creating Innovating Products (Agile Software Development Series). Addison-Wesley Professional (April 6, 2004);

15. IBM Rational Software – MDA Toolkit [žiūrēta 2004-07-12]. Prieiga per internetą: <http://www-306.ibm.com/software/rational/mda/toolkit.html>;
16. IBM Research. Overview of IBM CommonRules 1.0 Alpha Release [žiūrēta: 2004-12-12]. Prieiga per internetą: <http://www.research.ibm.com/rules/commonrules-overview.html>;
17. Kavanaqh P. Open Source Software: Implementation and Management. Digital Press (July 26, 2004);
18. Koppelaars T. Business rules. Classification and implementation (implementation through classification) [žiūrēta 2004-01-14]. Prieiga per internetą: <http://web.inter.nl.net/users/T.Koppelaars/maastricht94.doc>;
19. Ross R. G. The Business Rule Book: Classifying, Defining and Modeling Rules. Business Rule Solutions, 1997. ISBN 0-941049-03-5;
20. Ross R. G. Principles of the Business Rules Approach. Addison – Wesley Publ. Co., January 30, 2003;
21. Revolutionizing Application Design, Development and Maintenance [žiūrēta 2004-11-14]. Prieiga per internetą: http://www.blazesoft.com/brwhitepapers/advisor_revolution.pdf;
22. Rosenberg D. with Kendall S. Use Case Driven Object Modeling with UML: A Practical Approach. Addison-Wesley, 1999;
23. Shoemaker M. L. UML Applied: A .NET Perspective. Apress; 1 edition (April 5, 2004);
24. Schwaber K. Agile Project Management with Scrum (Microsoft Professional). Microsoft Press (March 10, 2004);
25. Eliyahu M. Goldratt, Schragenheim E. Tikslas III. Būtina, bet nepakankama. Goldratt Baltic Network, 2003.

TERMINŲ IR SUTRUMPINIMŲ ŽODYNAS

CASE (*Computer-Aided Software Engineering*) – automatizuoto programų kūrimo sistema;

ORM (*Object Role Modeling*) – duomenų modelio tipas;

DBVS – duomenų bazių valdymo sistema;

DB – duomenų bazė;

ER (*Entity Relationship*) – esybių ryšių (diagrama);

GVS – grafinė vartotojo sąsaja;

IS – informacijos sistema;

ISA – potipio sąryšio tipas, „yra potipis“;

SQL (*Structured Query Language*) – struktūrinė užklausų kalba duomenims apdoroti;

UML (*Unified Modeling Language*) – universali modeliavimo kalba;

VT – veiklos taisyklė;

VL – veiklos logika;

RUP (*Rational Unified Process*) – IBM Corp. pasiūlyta projektavimo metodika;

MSF (*Microsoft Solutions Framework*) – Microsoft Corp. pasiūlyta projektavimo metodika;

CMM (*Company Maturity Model*) – įmonės proceso brandos rodiklis;

XP (*eXtreme Programming*) – IS kūrimo metodika.

PRIEDAI

PRIEDAS A: MSF vizijos/apimties dokumentas

Kompanijos Foo Project

Vizija/Apimtis

Kompanija Foo

Web sistemos pardavimas kūrimas ir kompanijos veiklos pagerinimas

Author	Vilius Lukošius
Author	Principal Consultant, Program
Position	Manager
Date	2005-05-23

Version: 0.01

Revision and Sign-off Sheet
Change Record

Date	Author	Version	Change Reference
2005-05-23	Vilius Lukošius	1.1	Initial

Reviewers

Name	Version Approved	Position	Date

Distribution

Name	Position

Document Properties

Item	Details
Document Title	Vizija/Apimtis

Item	Details
Author	Vilius Lukošius
Creation Date	2005-05-23
Last Updated	2005-05-23

Vizijos formuluotė

Mūsų tikslas yra per sekančius metus padidinti savo rinkos dalį 50%, tuo pačiu padidinant 75% pardavimų kiekį, pardavimų spartą, tuo užtikrinant didėjančią klientų pasitenkinimą ir mūsų kompanijos kylantį prestižą.

Galimybių formuluotė

Didėjant klientų susidomėjimu kompanija ir jos teikiamais produktais ir paslaugomis, kompanija gauna vis daugiau užsakymų iš klientų, o tuo pačiu ir didėjančios darbų apimtys pardavimų ir gamybos departamentams. Kadangi dabar vienintelis būdas užsisakyti produktus yra tik telefonu tiesiogiai bendraujant su vadybininkais, tai sukelia problemų apdorojant padidėjusius klientų užsakymų kiekius. Taip pat, plintant Internet technologijoms, atsiranda naujos galimybės padidinti kompanijos užimamą rinkos dalį realizuojant automatinį užsakymų pateikimą ir vykdymą naudojant web elektroninę parduotuvę. Taip pat pasitelkus elektroninės bankininkystės sistemas automatizuotus užsakymų apmokėjimus, o tuo pačiu ir sumažinti užsakymų vykdymo laiką, kas padidintų klientų pasitenkinimą.

Apimtis

Iš vartotojo perspektyvos kuriamas sprendimas susideda iš dviejų dalių: elektroninės parduotuvės ir verslo valdymo sistemos padedančios automatizuoti užsakymų priėmimą ir apdorojimą.

Elektroninė parduotuvė yra skirta pritraukti didesnę klientų kiekį, tuo išplečiant kompanijos rinkos dalį, automatizuojant užsakymų pateikimą ir didinant kompanijos pelną. Elektroninė parduotuvė teiks tokias funkcijas:

- klientai galės peržiūrėti knygų katalogą, susidėti norimas knygas į krepšelį ir patvirtinti užsakymą;
- klientai galės peržiūrėti užsakymų istoriją;
- klientai galės apmokėti užsakymus naudojantis elektroninės bankininkystės priemones;
- klientai galės stebėti savo užsakymų vykdymą;
- klientai galės patys užsiregistruoti elektroninėje parduotuvėje;
- klientai galės pasiekti savo duomenis naudojantis savo susikurtus sistemos vartotojus;
- klientai galės automatiškai atsistatyti pamirštus slaptažodžius.

Užsakymų priėmimas ir vykdymas bus automatizuotas naudojant web programinę įrangą. Vadybininkai galės prisijungti prie sistemos per web naršyklę ir ten pildyti užsakymus, tvarkyti klientų duomenis, stebėti užsakymų vykdymą. Gamybos skyrius galės peržiūrėti naujus užsakymus, taip pat fiksuoti jų vykdymo būsenas.

Programinė įranga teiks tokias funkcijas:

- vadybininkai galės valdyti savo klientų duomenis (registruoti, redaguoti klientus);
- vadybininkai galės fiksuoti naujus užsakymus;
- vadybininkai galės modifikuoti užsakymus;
- vadybininkai galės stebėti užsakymų vykdymo būsenas;
- gamyba galės gauti užsakymų sąrašus;
- gamyba galės fiksuoti užsakymų vykdymo stadijas;
- buhalterija galės fiksuoti užsakymų apmokėjimus;

Visi aukščiau apteikti reikalavimai bus pateikti sprendimo pirmoje versijoje.

Ne šios versijos reikalavimai

Žemiau pateikti reikalavimai neįeina į versijos V1 apimtį ir jie bus realizuoti sekančiose versijose.

Funkcija	<i>Komentaras</i>	<i>Versija</i>
Automatinis pristatymo pateikimas kurjerių tarnybai.	Kurjerių tarnyba neteikia tokios paslaugos.	
Populiariausių knygų suradimas	-	V2

Tikslas

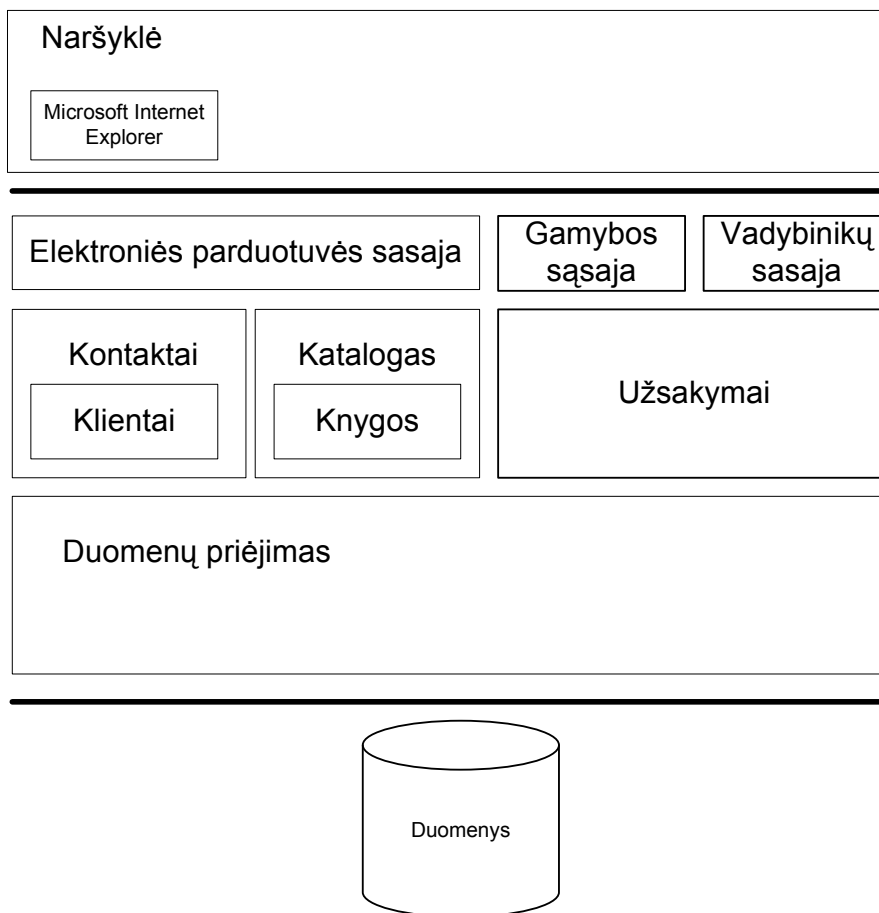
Šis skyrius pateikia koncepcinę sprendimo apžvalgą, ir supažindina su sprendimo reikalavimais.

Sprendimo bendras supratimas

Sprendimas susideda iš:

- **Elektroninė parduotuvė:**
pateikia knygų katalogą, iš kurio klientai gali susirinkti norimas knygas ir jų kiekius, užpildyti užsakymą, jį apmokėti ir sulaukti produkto pristatymo į namus. Pateikiama kaip web aplikacija.
- **Užsakymų priėmimo ir įvykdymo automatizavimas:**
Vadybininkai galės automatiškai užpildyti užsakymus, juos perduoti gamybos departamentui, stebėti užsakymų vykdymo būsenas, redaguoti užsakymus. Taipogi kaupti informaciją apie klientus. Gamybos departamentas galės automatiškai gauti naujus užsakymus vykdymui, žymėti jų vykdymo etapus. Sprendimas realizuojamas kaip web aplikacija.

Sekančioje diagramoje pateikiama kuriamo sprendimo architektūra aukštame lygyje.



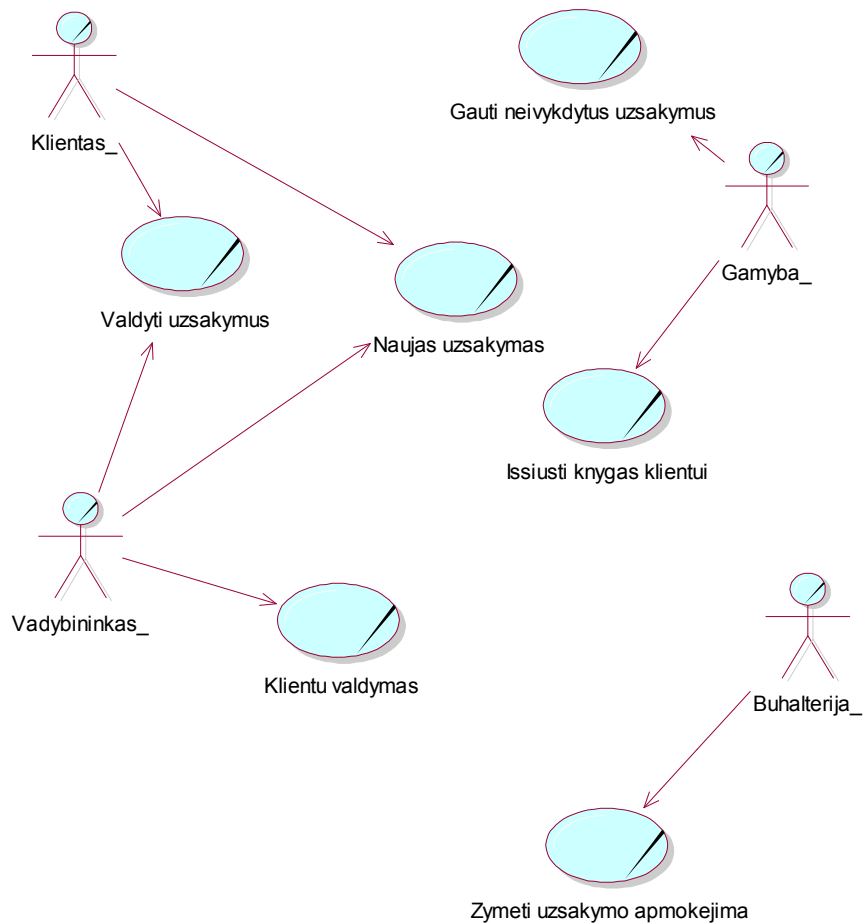
Vartotojų profiliai

Žemiau lentelėje pateikti sprendimo vartotojų tipai ir jų aprašymai:

Vartotojas	Trumpas vartotojo veiksmų aprašymas
Klientas	Peržiūri knygų katalogą, užpildo užsakymus, apmoka užsakymus, stebi užsakymų vykdymą.
Vadybininkas	Fiksuoja klientų pateiktus užsakymus, perduoda jus gamybai, stebi užsakymų būsenas.
Gamyba	Gauna naujų užsakymų sąrašus, žymi užsakymų vykdymo stadijas.

Sprendimo panaudojimo suvestinė

Sprendimo versijoje V1 bus realizuoti šie sistemos panaudojimo atvejai. Pilni panaudojimo atvejų scenarijai bus detalizuoti sekančiose projekto fazėse. Sistemos panaudojimo atvejai pateikti sekančioje diagramoje:



Reikalavimų santrauka

Žemiau pateikiami preliminarūs reikalavimai gauti pradinių interviu metu.

Veiklos reikalavimai

Veiklos tikslas sprendimui yra palaikyti didėjanti klientų kiekį ir pritraukti naujus klientus, pagreitinti užsakymų vykdymą. Veiklos reikalavimai yra aptarti Apimties skyriuje, o žemiau pateikiamos papildomos jų detalės:

- vadybininkai matytu tik savo klientų sąrašus;
- vadybininkai matytų visus užsakymus;
- klientai matytų tik savo užsakymus;
- klientai galėtų apmokėti užsakymus vėliau;
- gamyba vykdytų tik apmokėtus užsakymus;
-

Vartotojų reikalavimai

Vartotojų reikalavimai sugrupuoti pagal jų tipus.

Vadybininkai:

- peržiūrėti užsakymus įvairiais pjūviais:
 - pagal klientą;
 - pagal vykdymo stadiją;
 - pagal apmokėjimo požymį;
 - surasti neapmokėtus užsakymus;
- peržiūrėti knygų katalogus;
- matyti kainų pokyčius knygų kataloguose;
- užpildyti užsakymą;
- redaguoti užsakymą;
- atšaukti užsakymą;
- įtraukti naują klientą;
- redaguoti klientą.

Web klientas:

- peržiūrėti knygų katalogą;
- įtraukti knygas į prekių krepšelį;
- užsakyti knygas;
- apmokėti užsakymą;
- nurodyti užsakymo pristatymo adresą;
- atšaukti užsakymą;
- modifikuoti užsakymą;
- gauti užsakymo vykdymo būseną.

Gamyba:

- gauti naujus užsakymus;
- fiksuotu užsakymų vykdymą.

Veikimo reikalavimai

Sekantys reikalavimai aukštame lygyje pateikia sistemos veikimo apribojimus:

- puslapio užkrovimo laikas ne daugiau kaip 10 sek.;
- naudojama *incremental backup* strategija;
- sistema turi būti prieinama visoje kompanijoje, kur yra tinklas;
- vadybininko dalis turi veikti 1024x768 ir didesnės raškos spalvotuose ekranuose.

Sistemos reikalavimai

- Sprendimas turi veikti naudojant Internet Explorer 5.0 ar naujesnę versijas.
- Sistema turi veikti Windows NT 5.0 arba naujesniame serveryje.

5.1. Sėkmės kriterijus

Kad nustatyti projekto įgyvendino požymius senatys sėkmės matavimo faktoriai turi būti identifikuoti ir išmatuojami, ir naudojami nustatyti projekto sėkmei.

Elektroninė parduotuvė

- Klientai gali užsakyti bekuria knygą per tinklą, esančia sandėlyje;
- knygą galiam rasti pagal ISBN numerį, arba peržiūrint katalogą;
- klientai gali stebėti užsakymo vykdymą;
- klientai mato kataloge dabartines knygų kainas;
- klientai gali apmokėti užsakymą kredito kortele;
- klientai gali apmokėti užsakymą banko pavedimu;
- klientai gali apmokėti užsakymą hanza.net sistema.
-

Užsakymų priėmimo ir įvykdymo automatizavimas

- Vadybininkas gali įvesti naują užsakymą;
- vadybininkas gali įvesti nauja klientą;
- vadybininkas gali atsaukti neapmokėta užsakymą;
- vadybininkas gali peržiūrėti bet kurio užsakymo vykdymo būseną;
- gamyba gali gauti naujų užsakymų sąrašą;
- gamyba gali keisti apmokėtų užsakymo vykdymo būsenas;
- vadybininkas gali stebėti užsakymo būsenas;
- buhalterija gali fiksuoti užsakymo apmokėjimą.

PRIEDAS B: MSF loginis panaudojimo atvejų modelis

Kompanijos Foo Project

Panaudojimo atvejai

Kompanija Foo

Web sistemos pardavimas kūrimas ir kompanijos veiklos pagerinimas

Author	Vilius Lukošius
Author	Principal Consultant, Program
Position	Manager
Date	2005-05-23

Version: 0.01

Panaudojimo atvejai **Kompanijos Foo Project**

Actors

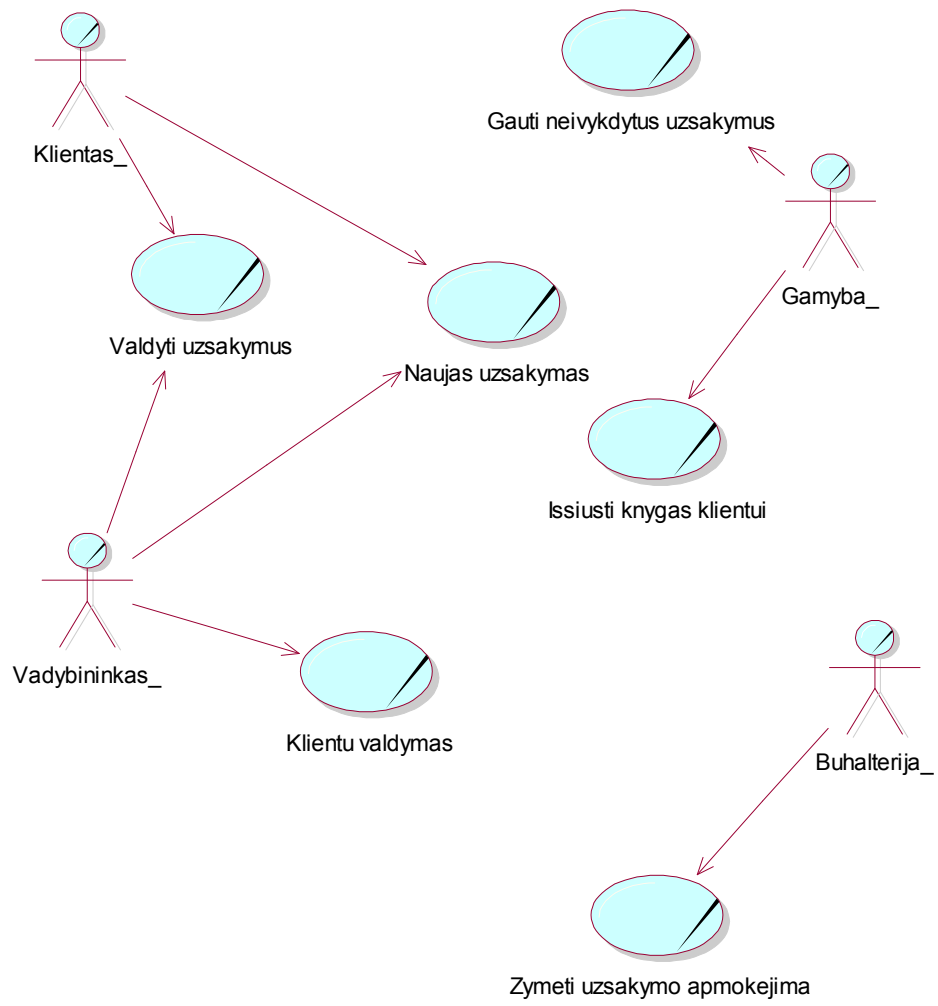
Klientas, Vadybininkas, Gamyba, Administratorius, Vartotojas, Buhalterija

Use Cases

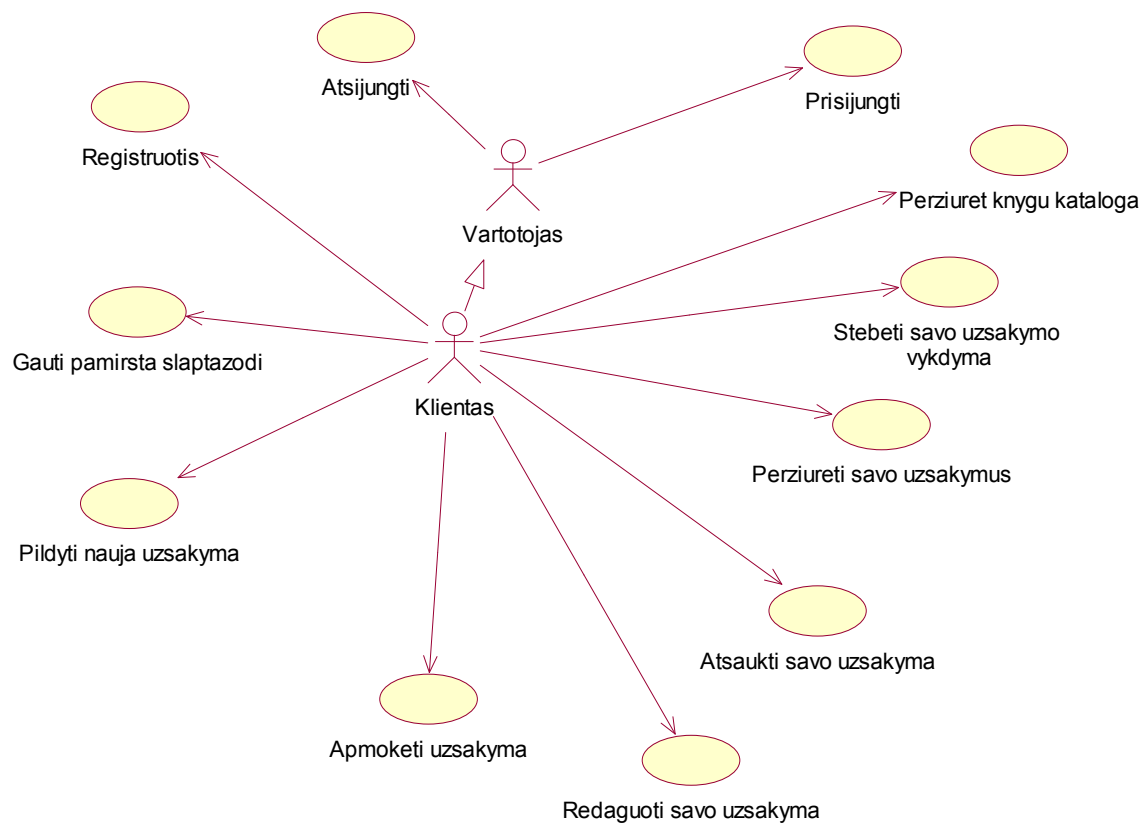
Apmoketi kredito kortele, Apmoketi pavedimu, Apmoketi per Hanza.NET, Apmoketi uzsakyma, Atsaukti savo uzsakyma, Atsaukti uzsakyma, Atsijungti, Fiksuoti uzsakyma, Fiksuoti uzsakyma isvezimui klientui, Fiksuoti uzsakymo apmokejima, Fiksuoti uzssakymo ivykdyma, Gauti apmoketus uzsakymus, Gauti neivykdytus uzsakymus, Gauti pamirsta slaptazodi, Gauti uzsakymo vykdymo busena, Issiusti knygas klientui, Itraukti knyga, Itraukti knyga i krepsele, Kesiti knygu kiekis, Kesiti knygu kiekius, Klientu Redagavimas, Klientu valdymas, Naujas uzsakymas, NewUseCase, Nurodyti pristatymo adresa, Pakeisti slaptazodi, Pasalinti klienta, Pasalinti knyga, Pasalinti knyga is krepsele, Pasalinti vartotoja, Pasirinkti klienta, Pasirinkti kliento pristatymo adresa, Patvirtinti uzsakyma, Perziureti knygu kataloga, Perziureti savo uzsakymus, Perziureti uzsakyma, Perziureti uzsakymus, Pildyti nauja uzsakyma, Prisijungti, Pristatymo adreso ivedimas, Redaguoti klienta, Redaguoti Klientus, Redaguoti savo uzsakyma, Redaguoti uzsakyma, Registruoti klienta, Registruoti uzsakyma, Registruotis, Sisuti kleintui pranesima, Skurti vartotoja, Stebeti savo uzsakymo vykdyma, Valdyti uzsakymus, ymeti uzsakymo apmokejima

Views

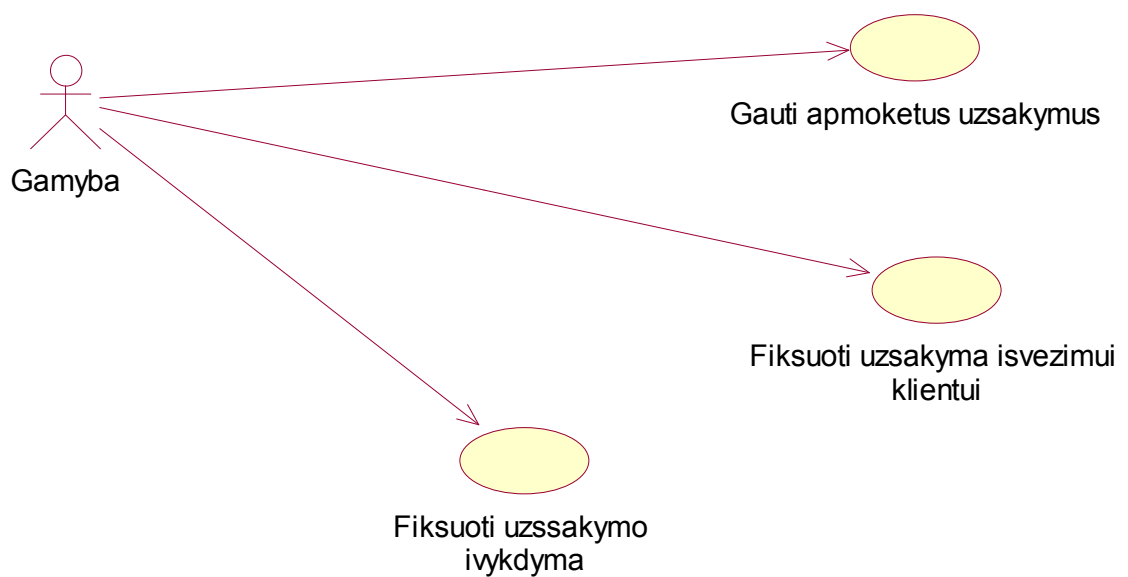
LogicalModel



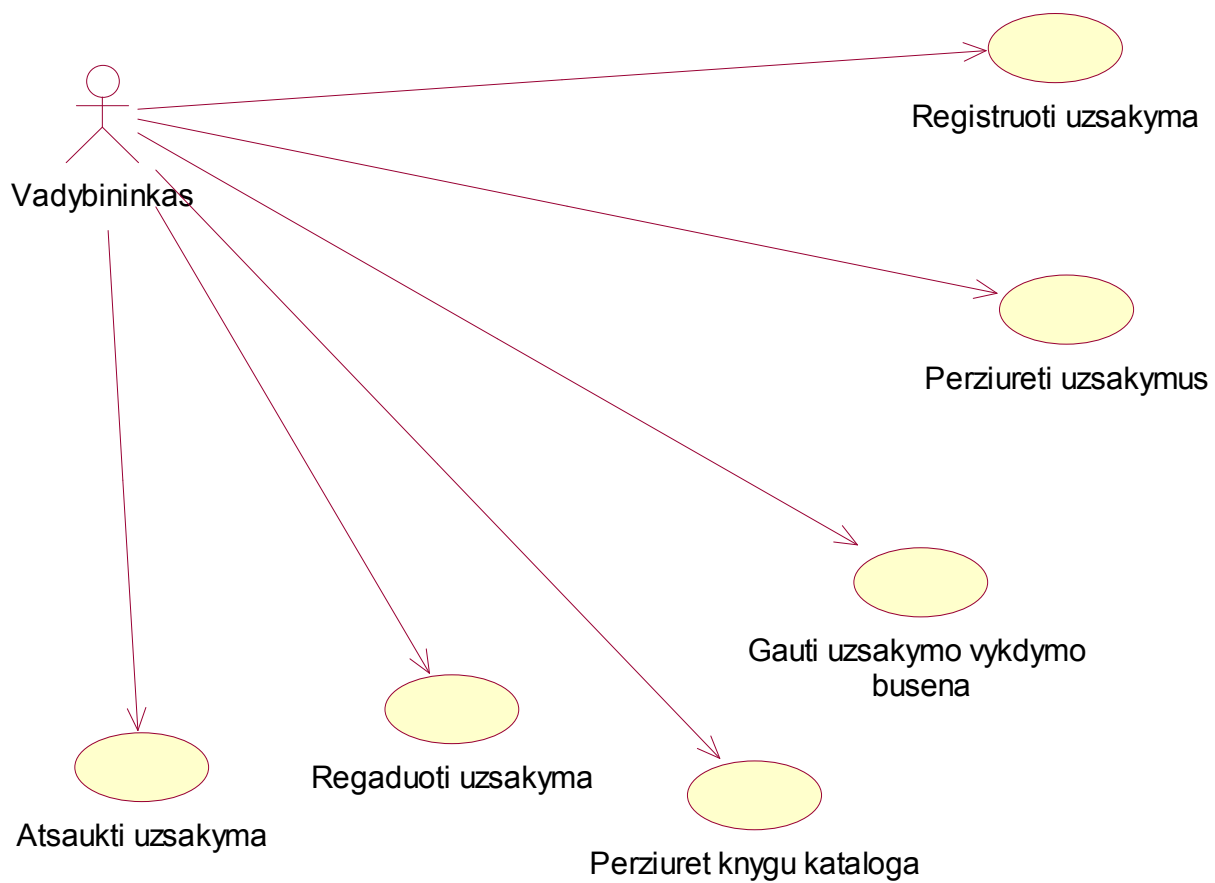
Main



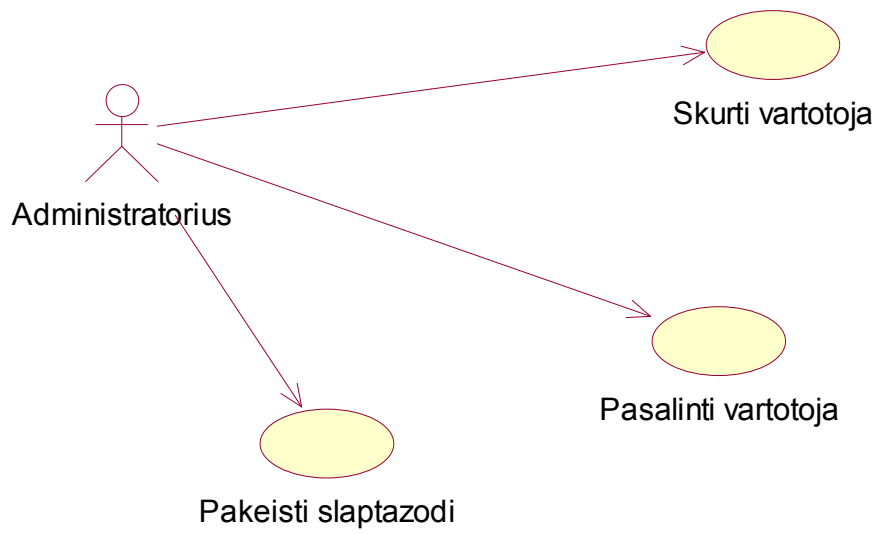
Klientas



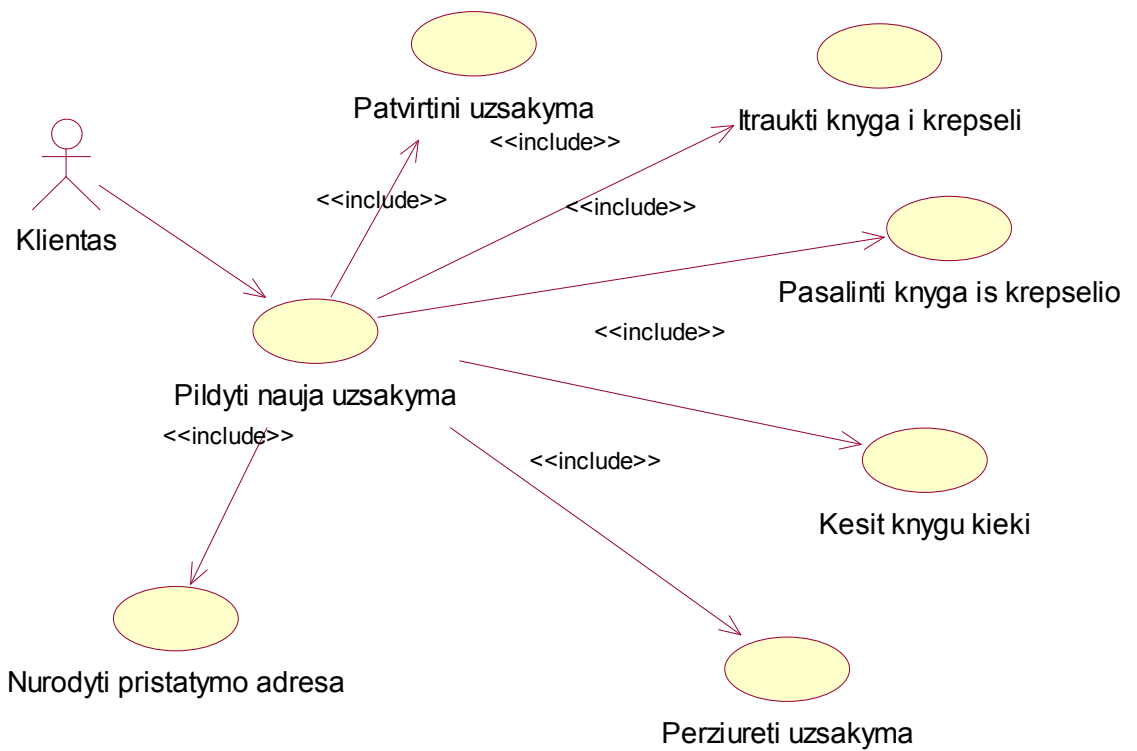
Gamyba



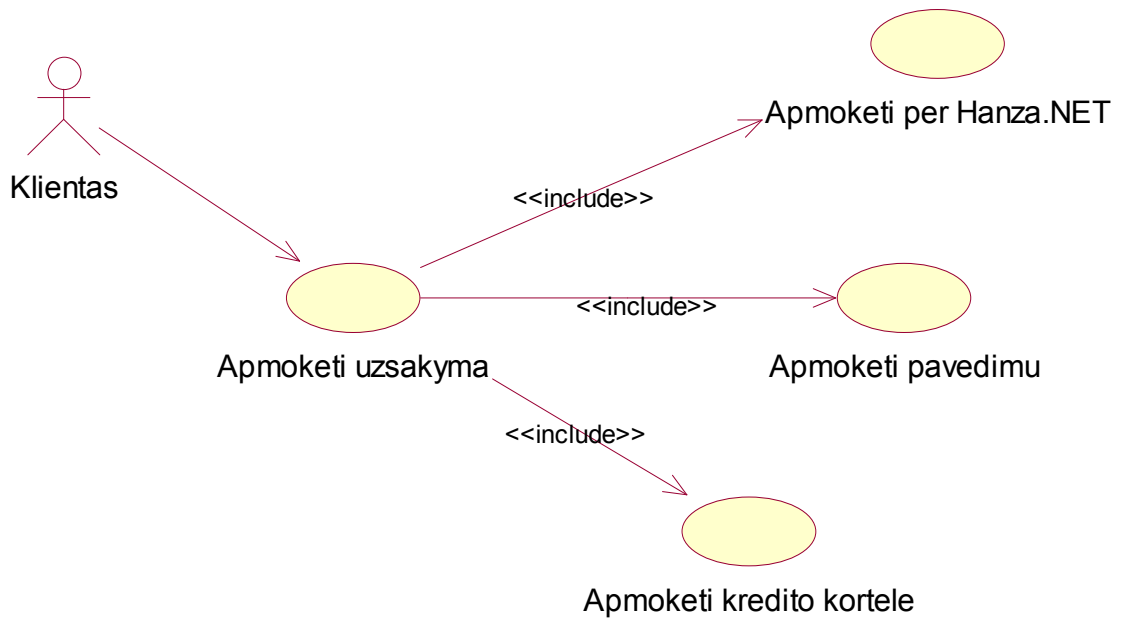
Vadybininkas



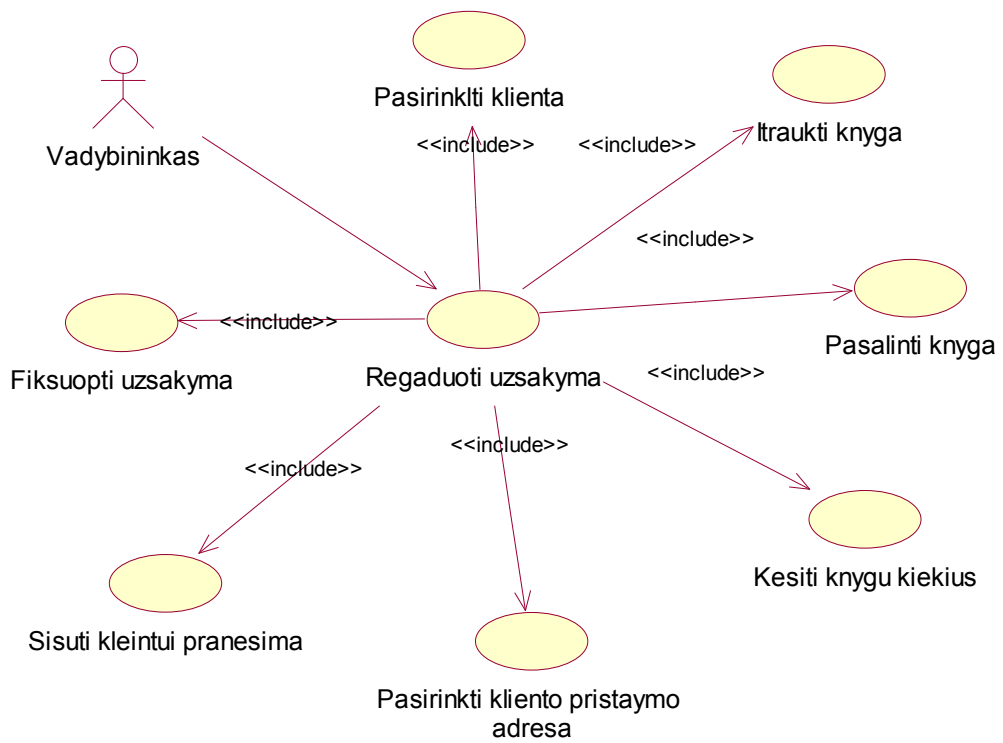
Administratorius



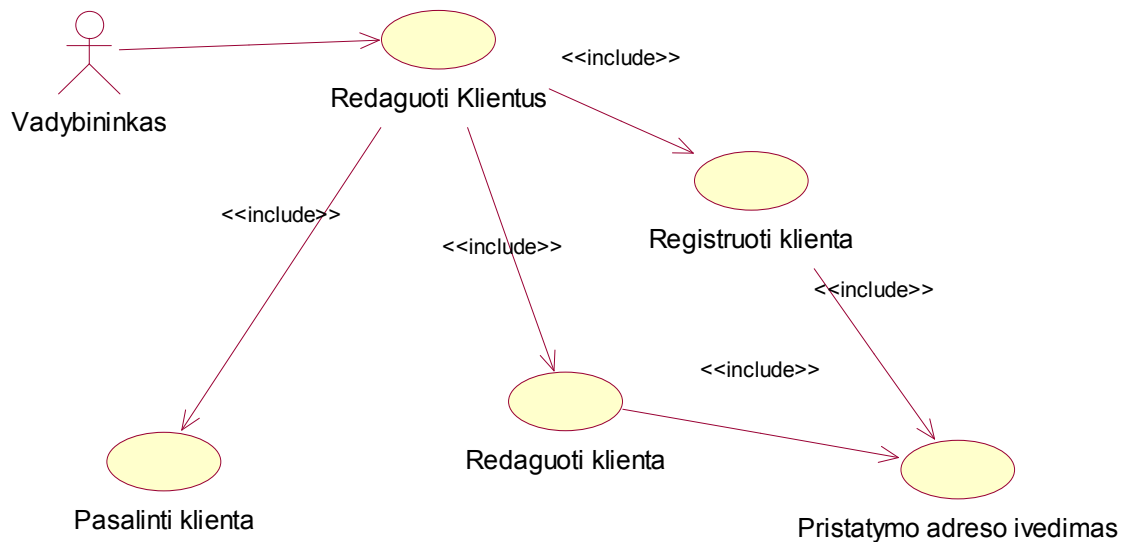
PildytiUzsakyma



ApmoketiUzsakyma



RegistruotiUzsakyma



Klientu Redagavimas

PRIEDAS C: Detalizuoti MSF panaudojimo atvejų aprašai

Use Case ID: UC 01.01.01

Title: UC 01.0.01 Prisijungti

Abbreviated Title: Prisijungti

Requirement ID: 1

Tikslas

Sprendimu gali naudotis tik autorizuoti vartotojai.

Scenarijaus Aprašymas

Vartotojas nori pasinaudoti sistema. Tam jis turi prisijungti prie jos naudodamasis savo prisijungimo vardu ir slaptažodį.

Prielaidos/Pradinės sąlygos

1. Vartotojas turi prisijungimo vardą ir slaptažodį.

Aktoriai

1. Vartotojas

Veiksmų Seka

1. Use Case prasideda kai vartotojas bando pasinaudoti sistema.
2. Sistema prašo įvesti prisijungimo vardą ir slaptažodį.
3. Vartotojas įveda prisijungimo vardą ir slaptažodį.
4. Sistema autentifikuoja vartotoją.
5. Use Case baigias kai sistema rodo vartotojui būdingą meniu tolimesnių veiksmų.

Alternatyvios Veiksmų Sekos

1. Jei blogai įvesti prisijungimo duomenys sistema apie tai praneša.
2. Jei vartotojas yra e parduotuvės klientas siūlo atstatyti pamirštą slaptažodį. Jei vartotojas yra vadybininkas siūlo kreiptis į sistemos administratorių.
3. Jei įvedamas neegzistuojantis vartotojas, tai siūloma vartotojui registruotis kaip klientui elektroninėje parduotuvėje.

Naudoja/Išplečia

1. Naudoja. UC 01.01.02 Gauti pamirštą slaptažodį.
2. Naudoja. UC 01.01.03 Registruotis.

Vartotojo Pageidavimai Realizacijai

1. Nėra

Dažnumas

Dažnai

Atsakingi

1. Nėra

Susijusios Specifikacijos

Nr.	Pavadinimas	Kategorija	Tipas

Problemos

1. Nėra

Problemų Sprendimo Idėjos

1. Nėra

Ateities Reikalavimai

1. Nėra

Modifikavimas

Data	Autorius	Aprašymas

Use Case ID: UC 01.01.02

Title: UC 01.0.02 Gauti pamirštą slaptažodį

Abbreviated Title: Gauti pamirštą slaptažodį

Requirement ID: 1

Tikslas

Elektroninės parduotuvės vartotojai gali pamiršti savo slaptažodis, tad reikalingas mechanizmas, kuries jiems leistų atsistatyti savo slaptažodį ir vėl naudotis sistema nesiregistruojantis joje iš naujo.

Scenarijaus Aprašymas

Klientas bando pasinaudoti sistema ir sistema praneša kad jis įvedė klaidingą slaptažodį ir pateikia galimybę atstatyti pamirštą slaptažodį, kuris išsiunčiamas klientui elektroniniu paštu.

Prielaidos/Pradinės sąlygos

2. Klientui nepavyko prisijunkti prie elektroninės parduotuvės, naudojantis prisijungti funkcija.

Aktoriai

2. Klientas

Veiksmų Seka

6. Use Case prasideda kai vartotojas pasirenka gauti pamirštą slaptažodį funkciją.
7. Sistema prašo įvesti vartotojo elektroninio pašto adresą.
8. Sistema išsiunčia elektroniniu paštu sugeneruota to vartotojo naują slaptažodį.
9. Use Case baigias kai klientas gauna elektroniniu paštu naują slaptažodį.

Alternatyvios Veiksmų Sekos

4. Vartotojs įvedė neteisingą savo elektroninio pašto adresą. Sistema apie tai praneša ir prašo pakartoti įvedimą.

Naudoja/Išplečia

3. Naudoja. UC 01.01.01 Prisijunkti

Vartotojo Pageidavimai Realizacijai

1. Nėra

Dažnumas

Dažnai

Atsakingi

2. Nėra
- 3.

Susijusios Specifikacijos

Nr.	Pavadinimas	Kategorija	Tipas

Problemos

2. Nėra

Problemų Sprendimo Idėjos

2. Nėra

Ateities Reikalavimai

2. Nėra

Modifikavimas

Data	Autorius	Aprašymas

Use Case ID: UC 01.01.03

Title: UC 01.0.03 Registruotis

Abbreviated Title: Registruotis

Requirement ID: 1

Tikslas

Elektroninės parduotuvės klientai turi būti registruoti vartotojai, t.y. jie turi įvesti apie save informaciją prieš tai kad galėtų pildyti užsakymus. Kad nereiktų pakartotinai įvedinėti informacijos apie save vartotojoms reikia prisiregistruoti sistemoje ir vėliau jie tik prisijunks naudodamiesi prisijungimo vardu ir slaptažodžiu.

Scenarijaus Aprašymas

Vartotojas prisijungia prie sistemos, kad galėtų pateikti naują užsakymą, peržiūrėti užsakymų istoriją. Tam reikia turėti savo prisijungimo duomenis, registravimosi metu vartotojas suveda duomenis apie save ir susikuria savo prisijungimo duomenis.

Prielaidos/Pradinės sąlygos

3. Klientas nori naudotis elektronine parduotuve.

Aktoriai

3. Klientas

Veiksmų Seka

10. Use Case prasideda kai vartotojas pasirenka „Registruotis“ funkciją.
11. Sistema prašo įvesti vartotojo vardą, pavardę, e-mail adresą, adresą, prisijungimo vardą ir slaptažodį.
12. Sistema sukuria vartotoją pagal nurodytus duomenis.
13. Use Case baigias kai klientas automatiškai prijungiamas prie sistemos su sukurtu vartotojo ir rodo pradini kliento langą.

Alternatyvios Veiksmų Sekos

5. Vartotojas įvedė jau egzistuojantį prisijungimo vardą, tada siūloma registruotis iš naujo ir pranešama apie tai, kad toks prisijungimo vardas jau egzistuoja.
6. Vartotojas įvedė jau užregistruotą sistemoje e-mail adresą, sistema pasiūlo atsisiųsti pamirštą slaptažodį, vietoje naujo vartotojo kūrimo.

Naudoja/Išplečia

- 4.

Vartotojo Pageidavimai Realizacijai

1. Nėra

Dažnumas

Retai

Atsakingi

4. Nėra

Susijusios Specifikacijos

Nr.	Pavadinimas	Kategorija	Tipas

Problemos

3. Nėra

Problemu Sprendimo Idėjos

3. Nėra

Ateities Reikalavimai

3. Nėra

Modifikavimas

Data	Autorius	Aprašymas

Use Case ID: UC 01.01.04

Title: UC 01.0.04 Atsijungti

Abbreviated Title: Atsijungti

Requirement ID: 1

Tikslas

Sprendimu gali naudotis tik autorizuoti vartotojai. Baigus darbą atsijungti, kad piktavaliai vartotojai negalētu pasinaudoti sistema jie pavyktų įsilaužti į vartotojo kompiuterį.

Scenarijaus Aprašymas

Vartotojas baigia darbą su sistema. Paspaudžia atsijungimo mygtuką. Papildomai vartotojas yra atjungiamas po nurodyto nieko nedarymo su sistema laiko tarpo.

Prielaidos/Pradinės sąlygos

4. Vartotojas prisijungęs prie sistemos.

Aktoriai

4. Vartotojas

Veiksmų Seka

14. Use Case prasideda kai vartotojas baigia darbą su sistema.
15. Vartotojas paspaudžia atsijungimo mygtuką ekrane.
16. Sistema atjungia vartotoją.
17. Uždaromas naršyklės langas.
18. Use Case baigias kai uždaromas naršyklės langas arba rodomo kad vartotojas atsijungė.

Alternatyvios Veiksmų Sekos

7. Vartotojas atjungiamas po nurodyto laiko nieko neveikimo.
8. Pradėjus atlikti koki kitą veiksmą rodomas pranešimas, kad vartotojas buvo automatiškai atjungtas ir siūloma jungtis prie sistemos iš naujo.

Naudoja/Išplečia

5. Naudoja. UC 01.01.01 Prisijunkti

Vartotojo Pageidavimai Realizacijai

1. Nėra

Dažnumas

Dažnai

Atsakingi

5. Nėra

Susijusios Specifikacijos

Nr.	Pavadinimas	Kategorija	Tipas

Problemos

4. Nėra

Problemu Sprendimo Idėjos

4. Nėra

Ateities Reikalavimai

4. Nėra

Modifikavimas

Data	Autorius	Aprašymas

PRIEDAS D: Detalizuoti siūlomo metodo panaudojimo atvejų aprašai

Use Case ID: UC 01.01.01

Title: UC 01.0.01 Prisijungti

Abbreviated Title: Prisijungti

Requirement ID: 1

Tikslas

Sprendimu gali naudotis tik autorizuoti vartotojai.

Scenarijaus Aprašymas

Vartotojas nori pasinaudoti sistema. Tam jis turi prisijungti prie jos naudodamasis savo prisijungimo vardą ir slaptažodį.

Prielaidos/Pradinės sąlygos

5. Vartotojas turi prisijungimo vardą ir slaptažodį.

Aktoriai

5. Vartotojas

Veiksmų Seka

19. Use Case prasideda kai vartotojas bando pasinaudoti sistema.
20. Sistema prašo įvesti prisijungimo vardą ir slaptažodį.
21. Vartotojas įveda prisijungimo vardą ir slaptažodį.
22. Sistema autentifikuoja vartotoją.
23. Use Case baigias kai sistema rodo vartotojui būdingą meniu tolimesnių veiksmų.

Alternatyvios Veiksmų Sekos

9. Jei blogai įvesti prisijungimo duomenys sistema apie tai praneša.
10. Jei vartotojas yra e parduotuvės klientas siūlo atstatyti apmirštą slaptažodį. Jei vartotojas yra vadybininkas siūlo kreiptis į sistemos administratorių.
11. Jei įvedamas neegzistuojantis vartotojas, tai siūloma vartotojui registruotis kaip klientui elektroninėje parduotuvėje.

Naudoja/Išplečia

6. Naudoja. UC 01.01.02 Gauti pamirštą slaptažodį.
7. Naudoja. UC 01.01.03 Registruotis.

Vartotojo Pageidavimai Realizacijai

2. Nėra

Veiklos Taisyklės

1. Nėra

Dažnumas

Dažnai

Atsakingi

6. Nėra

Susijusios Specifikacijos

Nr.	Pavadinimas	Kategorija	Tipas

Problemoms

5. Nėra

Problemu Sprendimo Idėjos

5. Nėra

Ateities Reikalavimai

5. Nėra

Modifikavimas

Data	Autorius	Aprašymas

Use Case ID: UC 01.01.02

Title: UC 01.0.02 Gauti pamirštą slaptažodį

Abbreviated Title: Gauti pamirštą slaptažodį

Requirement ID: 1

Tikslas

Elektroninės parduotuvės vartotojai gali pamiršti savo slaptažodis, tad reikalingas mechanizmas, kuries jiems leistų atsistatyti savo slaptažodį ir vėl naudotis sistema nesiregistruojantis joje iš naujo.

Scenarijaus Aprašymas

Klientas bando pasinaudoti sistema ir sistema praneša kad jis įvedė klaidingą slaptažodį ir pateikia galimybę atstatyti pamirštą slaptažodį, kuris išsiunčiamas klientui elektroniniu paštu.

Prielaidos/Pradinės sąlygos

6. Klientui nepavyko prisijunkti prie elektroninės parduotuvės, naudojantis prisijungti funkcija.

Aktoriai

6. Klientas

Veiksmų Seka

24. Use Case prasideda kai vartotojas pasirenka gauti pamirštą slaptažodį funkciją.

25. Sistema prašo įvesti vartotojo elektroninio pašto adresą.

26. Sistema išsiunčia elektroniniu paštu sugeneruota to vartotojo naują slaptažodį.

27. Use Case baigias kai klientas gauna elektroniniu paštu naują slaptažodį.

Alternatyvios Veiksmų Sekos

12. Vartotojas įvedė neteisingą savo elektroninio pašto adresą. Sistema apie tai praneša ir prašo pakartoti įvedimą.

Naudoja/Išplečia

8. Naudoja. UC 01.01.01 Prisijunkti

Vartotojo Pageidavimai Realizacijai

2. Nėra

Veiklos Taisyklės

1. Nėra

Dažnumas

Dažnai

Atsakingi

7. Nėra

Susijusios Specifikacijos

Nr.	Pavadinimas	Kategorija	Tipas

Problemos

6. Nėra

Problemų Sprendimo Idėjos

6. Nėra

Ateities Reikalavimai

6. Nėra

Modifikavimas

Data	Autorius	Aprašymas

Use Case ID: UC 01.01.03

Title: UC 01.0.03 Registruotis

Abbreviated Title: Registruotis

Requirement ID: 1

Tikslas

Elektroninės parduotuvės klientai turi būti registruoti vartotojai, t.y. jie turi įvesti apie save informaciją prieš tai kad galėtų pildyti užsakymus. Kad nereiktų pakartotinai įvedinėti informacijos apie save vartotojoms reikia prisiregistruoti sistemoje ir vėliau jie tik prisijunks naudodamiesi prisijungimo vardu ir slaptažodžiu.

Scenarijaus Aprašymas

Vartotojas prisijungia prie sistemos, kad galėtų pateikti naują užsakymą, peržiūrėti užsakymų istoriją. Tam reikia turėti savo prisijungimo duomenis, registravimosi metu vartotojas suveda duomenis apie save ir susikuria savo prisijungimo duomenis.

Prielaidos/Pradinės sąlygos

7. Klientas nori naudotis elektronine parduotuve.

Aktoriai

7. Klientas

Veiksmų Seka

28. Use Case prasideda kai vartotojas pasirenka „Registruotis“ funkciją.

29. Sistema prašo įvesti vartotojo vardą, pavardę, e-mail adresą, adresą, prisijungimo vardą ir slaptažodį.

30. Sistema sukuria vartotoją pagal nurodytus duomenis.

31. Use Case baigias kai klientas automatiškai prijungiamas prie sistemos su sukurtu vartotojo ir rodo pradini kliento langą.

Alternatyvios Veiksmų Sekos

13. Vartotojas įvedė jau egzistuojantį prisijungimo vardą, tada siūloma registruotis iš naujo ir pranešama apie tai, kad toks prisijungimo vardas jau egzistuoja.

14. Vartotojas įvedė jau užregistruotą sistemoje e-mail adresą, sistema pasiūlo atsisiųsti pamirštą slaptažodį, vietoje naujo vartotojo kūrimo.

Naudoja/Išplečia

9.

Vartotojo Pageidavimai Realizacijai

2. Nėra

Veiklos Taisyklės

Kodas	Aprašas	Tipas	Susiję objektai
BR02	Email TURI TURĖTI	Veiksmą valdantis	
BR05	@ simbolį. Jei BR02 = true autorizuoti	Leidimas	BR02

Dažnumas

Retai

Atsakingi

8. Nėra

Susijusios Specifikacijos

Nr.	Pavadinimas	Kategorija	Tipas

Problemos

7. Nėra

Problemu Sprendimo Idėjos

7. Nėra

Ateities Reikalavimai

7. Nėra

Modifikavimas

Data	Autorius	Aprašymas

Use Case ID: UC 01.01.04

Title: UC 01.0.04 Atsijungti

Abbreviated Title: Atsijungti

Requirement ID: 1

Tikslas

Sprendimu gali naudotis tik autorizuoti vartotojai. Baigus darbą atsijungti, kad piktavaliai vartotojai negalētu pasinaudoti sistema jie pavyktų įsilaužti į vartotojo kompiuterį.

Scenarijaus Aprašymas

Vartotojas baigia darbą su sistema. Paspaudžia atsijungimo mygtuką. Papildomai vartotojas yra atjungiamas po nurodyto nieko nedarymo su sistema laiko tarpo.

Prielaidos/Pradinės sąlygos

8. Vartotojas prisijungęs prie sistemos.

Aktoriai

8. Vartotojas

Veiksmų Seka

32. Use Case prasideda kai vartotojas baigia darbą sus sistema.
33. Vartotojas paspaudžia atsijungimo mygtuką ekrane.
34. Sistema atjungia vartotoją.
35. Uždaromas naršyklės langas.
36. Use Case baigias kai uždaromas naršyklės langas arba rodomo kad vartotojas atsijungė.

Alternatyvios Veiksmų Sekos

15. Vartotojas atjungiamas po nurodyto laiko nieko neveikimo.
16. Pradėjus atlikinėti koki kitą veikslę rodomas pranešimas, kad vartotojas buvo automatiškai atjunktas ir siūloma jungtis prie sistemos iš naujo.

Naudoja/Išplečia

10. Naudoja. UC 01.01.01 Prisijunkti

Vartotojo Pageidavimai Realizacijai

2. Nėra

Veiklos Taisyklės

Kodas	Aprašas	Tipas	Susiję objektai
	Jei Uždaryti = true Sesiją nutraukti	Leidimas	BR02

Dažnumas

Dažnai

Atsakingi

9. Nėra

Susijusios Specifikacijos

Nr.	Pavadinimas	Kategorija	Tipas

Problemos

8. Nėra

Problemų Sprendimo Idėjos

8. Nėra

Ateities Reikalavimai

8. Nėra

Modifikavimas

Data	Autorius	Aprašymas