

KAUNO TECHNOLOGIJOS UNIVERSITETAS

INFORMATIKOS FAKULTETAS

PROGRAMŲ INŽINERIJOS KATEDRA

Algirdas Varnagiris

**PAVEIKSLŲ APRAŠYMO ATVIRO KODO PROGRAMINĖS
ĮRANGOS TYRIMAS**

Magistro darbas

Vadovas: doc. dr. Tomas Blažauskas

KAUNAS, 2009

KAUNO TECHNOLOGIJOS UNIVERSITETAS

INFORMATIKOS FAKULTETAS

PROGRAMŲ INŽINERIJOS KATEDRA

Algirdas Varnagiris

**PAVEIKSLŲ APRAŠYMO ATVIRO KODO PROGRAMINĖS
ĮRANGOS TYRIMAS**

Magistro darbas

Recenzentas:.....

2009 05 25

Vadovas:.....

doc. dr. Tomas Blažauskas

2009 05 25

Autorius:.....

IFM-3/2 gr. Stud. Algirdas Varnagiris

2009 05 25

KAUNAS, 2009

TURINYS

Įvadas.....	8
1.Programinės įrangos įgyvendinimo ir esamų sistemų analizė.....	9
2.Tikslas.....	9
3.Egzistuojantys sprendimai.....	9
1.1.Steve.museum sprendimas.....	9
1.2.Flickr sprendimas.....	10
1.3.Wikipedia.....	11
4.Programų sistemų savybių kiekybinis ir/arba kokybinis palyginimas.....	12
5.Bendradarbiavimas (Collaboration).....	12
1.4.Atviras turinys.....	12
1.5.Atviro turinio pavyzdžiai.....	13
1.6.Atviro turinio modelis.....	14
1.7.Atviro turinio procesų modelis.....	14
1.8.Dvi atviro turinio kūrimo kraštutinybės.....	19
6.Įgyvendinimo problemos.....	20
1.9.Turinio redagavimo bendradarbiaujant problemos.....	20
1.9.1.Priešiška nusiteikę vartotojai.....	20
1.9.2.Reklamos robotai.....	21
1.10.Svetainės turinio formavimas, nemokant HTML, CSS.....	21
1.10.1.WYSIWYG redaktoriai.....	21
1.10.2.Supaprastinto žymėjimo kalbos.....	22
1.11.Daugiakalbiškumo problemos.....	22
7.Paveikslų aprašymo atviro kodo programinės įrangos sistemos projektas.....	23
8.Funkciniai reikalavimai.....	23
1.12.Veiklos sfera.....	23
1.13.Produkto veiklos sfera.....	24
9.Funkciniai reikalavimai ir reikalavimai duomenims.....	28
1.14.Funkciniai reikalavimai.....	28
1.15.Reikalavimai duomenims.....	31
10.Nefunkciniai reikalavimai.....	32
1.16.Reikalavimai sistemos išvaizdai.....	32
1.17.Reikalavimai panaudojamumui.....	32
1.18.Reikalavimai vykdymo charakteristikoms.....	32
1.19.Reikalavimai sistemos priežiūrai.....	34
11.Architektūros specifikacija.....	35
1.20.Apžvalga.....	35
1.21.Architektūros pateikimas.....	35
1.22.Architektūros tikslai ir apribojimai.....	35
1.23.Sistemos statinis vaizdas.....	36
1.23.1.Apžvalga.....	36
1.24.Sistemos dinaminis vaizdas.....	39
1.24.1.PA veiklos diagramos.....	39
1.24.2.Sekų (bendradarbiavimo) diagramos.....	41
1.25.Išdėstymo vaizdas.....	44
1.26.Duomenų vaizdas.....	45
1.27.Kokybė.....	45

12. Detalios architektūros specifikacija.....	45
1.28. Sistemos architektūra.....	45
1.28.1. Sistemos vaizdas.....	45
1.28.2. Sistemos paketų detalizavimas.....	49
13. Testavimo medžiaga.....	58
1.29. Testavimo tikslai.....	58
1.30. Testavimo apimtis.....	58
1.31. Testavimo planas.....	58
1.32. Testuojama programinė įranga.....	59
1.33. Testavimo strategija.....	59
1.33.1. Vienetų testavimas.....	60
1.33.2. Integravimo testavimas.....	60
1.33.3. Priėmimo testavimas.....	60
1.33.4. Aukšto lygio testavimas.....	61
1.34. Testavimo resursai.....	61
1.35. Testavimo procedūra.....	61
1.35.1. Vienetų testavimas.....	61
1.35.2. Integravimo testavimas.....	62
1.35.3. Vartotojo sąsajos testavimas.....	62
1.35.4. Priėmimo testavimas.....	63
1.35.5. Aukšto lygio testavimas.....	63
14. Paveikslų aprašymo atviro kodo programinės įrangos tyrimas.....	64
15. Sistemos funkcijų tyrimas.....	64
16. Programinės įrangos kokybės tyrimas pagal ISO-9126 standartą.....	66
17. DB objektų panaudojimo tyrimas.....	68
18. Išvados.....	71
19. Terminų santrumpų žodynas.....	72
20. Literatūra.....	73

Open source software for image tagging research

SUMMARY

The main objective of this project was to design and develop open source software for image tagging. During this process the analysis of alternative image tagging systems was performed, as well as gathering system requirements.

When the system was developed, there was made a research to determine software quality. There was made three types of researches: research of software functionality, research of software quality to fit ISO 9126 standard and research of using DB objects.

Finally it was determined that software for image tagging fits most functional requirements. Also It fits ISO 9126 quality standards. And using DB objects was justifiable and suitable for this kind of system.

Paveikslėlių sąrašas

- 1 Paveikslas.** Žmonės ir procesai įtraukti į atviro turinio kūrimą.
- 2 Paveikslas.** Sekų diagrama hipotetiniam atviro turinio projektui.
- 3 Paveikslas.** Bendravimo būdai.
- 4 Paveikslas.** Panaudos atvejų diagrama.
- 5 Paveikslas.** Preliminari sistemos objektų bei jų ryšių schema.
- 6 Paveikslas.** Paketų detalizavimas.
- 7 Paveikslas.** Sistemos detalizavimas.
- 8 Paveikslas.** Vartotojų valdymo paketas.
- 9 Paveikslas.** Paveikslų valdymo paketas.
- 10 Paveikslas.** Žymių valdymo paketas.
- 11 Paveikslas.** Komentarų valdymo paketas.
- 12 Paveikslas.** Vartotojo registracijos veiklos diagrama.
- 13 Paveikslas.** Vartotojo prisijungimas.
- 14 Paveikslas.** Vartotojo informacijos keitimas.
- 15 Paveikslas.** Valdyti paveikslus.
- 16 Paveikslas.** Valdyti komentarus.
- 17 Paveikslas.** Paieška.
- 18 Paveikslas.** Registracija sekų diagrama.
- 19 Paveikslas.** Prisijungimas sekų diagrama.
- 20 Paveikslas.** Vartotojo informacijos keitimas sekų diagrama.
- 21 Paveikslas.** Valdyti paveikslus sekų diagrama.
- 22 Paveikslas.** Valdyti komentarus sekų diagrama.
- 23 Paveikslas.** Paieška sekų diagrama.
- 24 Paveikslas.** Išdėstymo vaizdas.
- 25 Paveikslas.** Duomenų vaizdas.
- 26 Paveikslas.** Sistemos vaizdas.
- 27 Paveikslas.** „Sistema“ paketo struktūra.
- 28 Paveikslas.** „Paveikslų valdymo“ paketo struktūra.
- 29 Paveikslas.** „Komentarų valdymo“ paketo struktūra.
- 30 Paveikslas.** „Žymų valdymo“ paketo struktūra.
- 31 Paveikslas.** „Vartotojų valdymas“ paketo struktūra.

- 32 Paveikslas.** „Sistemos parametrai“ paketo struktūra.
- 33 Paveikslas.** „Aplinkos atributų keitimo“ paketo struktūra.
- 34 Paveikslas.** Sistemos funkcionalumo tyrimo rezultatų grafikas.
- 35 Paveikslas.** Sistemos kokybės atitikimo ISO 9126 standartui tyrimo rezultatų grafikas.
- 36 Paveikslas.** Užklausų trukmės naudojant DB objektus rezultatų grafikas.

Lentelių sąrašas

- 1 lentelė.** Programų savybių palyginimas.
- 2 lentelė.** Programų sistemų techninių savybių palyginimas.
- 3 lentelė.** Pamokos kurias galima išmokti iš atviro kodo programinės įrangos vystymo.
- 4 Lentelė.** Procesai, įrankiai ir žmonės dalyvaujantys atviro turinio kūrime.
- 5 Lentelė.** Dviejų atviro turinio kūrimo kraštutinybių charakteristikos.
- 6 Lentelė.** Veiklos padalinimas.
- 7 Lentelė.** Funkciniai reikalavimai, 1 reikalavimas.
- 8 Lentelė.** Funkciniai reikalavimai, 2 reikalavimas.
- 9 Lentelė.** Funkciniai reikalavimai, 3 reikalavimas.
- 10 Lentelė.** Funkciniai reikalavimai, 4 reikalavimas.
- 11 Lentelė.** Funkciniai reikalavimai, 5 reikalavimas.
- 12 Lentelė.** Funkciniai reikalavimai, 6 reikalavimas.
- 13 Lentelė.** Funkciniai reikalavimai, 7 reikalavimas.
- 14 Lentelė.** Funkciniai reikalavimai, 8 reikalavimas.
- 15 Lentelė.** Funkciniai reikalavimai, 9 reikalavimas.
- 16 Lentelė.** Funkciniai reikalavimai, 10 reikalavimas.
- 17 Lentelė.** Funkciniai reikalavimai, 11 reikalavimas.
- 18 Lentelė.** Reikalavimai sistemos išvaizdai.
- 19 Lentelė.** Reikalavimai panaudojamumui.
- 20 Lentelė.** Reikalavimai vykdymo charakteristikoms.
- 21 Lentelė.** Reikalavimai sistemos priežiūrai.
- 22 Lentelė.** Sistemos funkcionalumo tyrimo rezultatų lentelė.
- 23 Lentelė.** Sistemos kokybės atitikimo ISO 9126 standartui tyrimo rezultatų lentelė.
- 24 Lentelė.** Užklausų trukmės naudojant DB objektus rezultatai.

IVADAS

Šio projekto tikslas sukurti programinę įrangą, kuri padės muziejines vertybes padaryti prieinamas plačiajai visuomenei. Sistemos veikimo principas bus paremtas atviro turinio pildymu.

Didėjant interneto vartotojų skaičiui, augant informacijos kiekiams internete, tampa sunku ir netgi per brangu vienam žmogui ar organizacijai aprašyti, kategorizuoti ar kitaip apibūdinti informaciją. Todėl atsirado nauja metodologija pavadinta anglišku terminu „Folksonomy“, kurį lietuviškai būtų galima pavadinti žmonių taksonomija.

Tai yra informacijos internete viešas pateikimas, naudojant visiškai neapribotus ir nesuvaržytus apibūdinimus. Taip gali būti aprašytos internetinės svetainės, nuotraukos, nuorodos. „Folksonomy“ tikriausiai yra priešingybė taksonomijai, kur aprašymo sistemos autoriai dažniausiai yra turinio kurį aprašo autoriai. Aprašymai yra bendrai vadinami žymėmis (Tags), o aprašymo procesas – žymėjimas (tagging).

Šio žymėjimo tikslas yra padaryti informaciją lengviau randamą, lengviau išaiškinamą ir lengviau skleidžiamą internete.

Kadangi „folksonomija“ sukūrė internetinę socialinę aplinką, tai vartotojai gali atrasti kas sukūrė žymę ir pamatyti kitas to asmens sukurtas žymes. Tokiu būdu vartotojai atranda kitų asmenų žymių rinkinius, kurie turi prasmę būtent jiems. To rezultatas yra greitas ir naudingas reikalingos informacijos radimas.

Žmonių taksonomijos kūrimas ir paieškos įrankiai nėra dalis WWW protokolo. Žmonių taksonomija atsirado internetinių bendruomenių pagrindu, kur turinys yra kuriamas naudojant žymes. Šios bendruomenės sukūrė galimybę Interneto vartotojams aprašyti ir pasidalinti pačių sukurtu turiniu, pavyzdžiui: fotografijomis, interneto svetainių, knygų katalogais.

Šiame darbe nagrinėjama tyrimo sritis susijusi su paveikslų žymėjimo sistemų projektavimu ir galimų projektavimo metodų parinkimu. Analitinėje darbo dalyje pristatomos panašios sistemos, jos lyginamos. Projektinėje dalyje pateikiama suprojektuotos sistemos architektūra. Tiriamojoje darbo dalyje apžvelgiamas sistemos funkcionalumas, kokybė ir DB objektų panaudojimo tinkamumas šiai sistemai.

1. PROGRAMINĖS ĮRANGOS ĮGYVENDINIMO IR ESAMŲ SISTEMŲ ANALIZĖ

2. Tikslas

Projekto tikslas padaryti muziejines vertybes prieinamas plačiajai visuomenei. Įgyvendinant šiuos reikalavimus:

- Galimybė registruotis ir pasirinktai sudalyvauti demografinėje apklausoje.
- Peržiūrėti muziejaus paveikslų rinkinius arba ieškoti paveikslų pagal susietą meta informaciją.
- Priskirti žymes paveikslams
- Peržiūrėti žymes priskirtas paveikslams.
- Atvaizduoti paveikslus, žymes ir/arba meta duomenis
- Paveikslams pridėti arba redaguoti komentarus, naudojant Wikipedia modelį. Sistema ištraukia raktažodžius iš paveikslo aprašymo bei vartotojų komentarų ir automatiškai išverčia raktažodžius skirtingomis kalbomis. Vartotojai gali redaguoti šiuos raktažodžius, juos pridedant ar koreguojant.

3. Egzistuojantys sprendimai

Šiuo metu yra daugybė sprendimų, kurie paremti „Folksonomy“ principu, vieni populiariausių yra šie:

- Steve Museum;
- Flickr;
- Wikipedia;

1.1. *Steve.museum sprendimas*

„Steve“ yra bendras tiriamasis darbas, tiriantis vartotojų sukurtų meno darbų aprašymų galimybes pagerinti muziejų kolekcijų prieinamumą. Šį projektą vykdo grupė savanorių iš muziejų, kurių tikslas pagerinti prieinamumą prie jų kolekcijų. Dalyvavimas šiame projekte yra atviras kiekvienam norinčiam prisidėti prie projekto vystymo, nesvarbu ar jis atstovauja muziejų ar ne [9].

Kartu dirbdami, jie tikisi išmokti daugiau apie tai kaip pagerinti prieinamumą prie muziejų kolekcijų. Atlikdami tyrimus, suprato, jog žmonės dažniausiai prisimena apie kūrinius visiškai ne tai kas yra aprašyta oficialiuose kolekcijos dokumentuose. Todėl norima

surinkti vartotojų aprašymus, kurie galėtų pateikti trūkstamus aprašus ir šiuos internetinius resursus padaryti naudingesnius.

Taigi, jie bendradarbiaudami kuria atviro kodo įrankį, leisiantį vartotojams aprašyti muziejines vertybes. Vartotojai mato ir aprašo kūrinius visiškai skirtingais terminais nei tai padaro muziejų specialistai, todėl vartotojų aprašai bus daug naudingesni ir pagerins paieškos rezultatus ir vartotojas lengviau ras tai kas jį domina.

Rugsėjo mėnesį buvo išleista pirmoji Steve 1.0 versija. Šioje versijoje bus paprasta žymėjimo (tagging) aplikacija. Taip pat bus prieiga kuri palaikys SOAP WEB servisą, o tai įgalins kūrėjus dar daugiau išplėsti Steve funkcionalumą kuriant aplikacijas kitomis programavimo kalbomis.

Šiuo metu esantis prototipas leidžia aprašinėti esamus paveikslus, tai gali daryti tik registruoti vartotojai. Vėliau surinkti vartotojų aprašai yra panaudojami paieškos rezultatams gerinti.

Prisijungusiam vartotojui parodomas, dar nerodytas paveikslas ir leidžiama jį aprašyti savais žodžiais. Neregistruotas vartotojas, gali naršyti po katalogą, arba pasinaudoti paieška. Yra matomas sąrašas nuotraukų, kurias išdidinus galima pamatyti ir daugiau informacijos apie kūrinį.

Registracija vykdoma siunčiant sistemos sugeneruotą slaptažodį nurodytu el. pašto adresu, taip stengiamasi apsisaugoti nuo robotų automatiškai vedančių netinkamą informaciją.

Paveikslus gali įkelti administratorius, papildomai įvesdamas informaciją apie autorių, pavadinimą ir t.t.

Vartotojų įvesti nauji aprašymai, ieškomi raktažodžiai yra sekami ir apdorojami įvairiais valdymo algoritmais.

Sistema suprogramuota naudojant PHP – vieną populiariausių atviro kodo WEB aplikacijų kūrimo kalbų. Visa informacija saugoma SQL duomenų bazėje, šio metu tai yra atviro kodo DB MySQL, tačiau ji prieinama per ADODB, kuri suteikia galimybę duomenis saugoti daugelyje kitų duomenų bazių, tokių kaip Oracle ar MSSQL. Sistema naudoja šablonų sistemą Smarty todėl programavimo logika yra atskirta nuo puslapio išdėstymo šablonų.

1.2. Flickr sprendimas

Flickr – tai vienas didžiausių portalų, kuriame vartotojai gali dalintis nuotraukomis. Vartotojas gauna savo asmeninį puslapį, kuriame galės rodyti asmenines nuotraukas. Žmonės gali palikti komentarus prie bet kurios fotografijos. Sistema automatiškai

sukuria įvairių dydžių nuotraukas (Sąrašui, vidutines, dideles). Vartotojas turi galimybę nustatyti kas galės matyti jo nuotraukas, taip pat gali apriboti galimybę jas komentuoti.

Vartotojas, gali kurti vartotojų grupes, kviesti prisijungti draugus, šeimos narius, gimines, ir visiems jiems priskirti skirtingas teises matyti nuotraukas.

Esminė galimybė – tai žymėjimas (tagging) ir aprašymų pridėjimas, kurie vėliau bus panaudoti atvaizduojant paieškos rezultatus. Tuo būdu nuotraukos bus lengviau randamos žmonių kurie ieško būtent tai ką vartotojas aprašė. Žymės yra lengvai redaguojamos, jas galima pridėti, ištrinti ir jų kiekis nėra apribotas.

Papildomos sistemos galimybės [10]:

- Pilnas RSS ir Atom palaikymas
- Įrankiai leidžiantys lengvai įkelti nuotraukų rinkinius tiesiai iš Windows ar Macintosh OS iškart su žymėmis ir privatumo nustatymais.
- Mobilii svetainės versija leidžianti matyti nuotraukas mobiliajame telefone ar PDA kompiuteryje
- Bendras nuotraukų telkinys, į kurį nuotraukas gali sukelti keli žmonės, pavyzdžiui dalyvavę tame pačiame renginyje.

1.3. *Wikipedia*

Wikipedia – internetinė, enciklopedija, buvo sukurta 2001 metais ir greitai augant, tapo vienu daugiausiai informacijos turinčiu puslapiu Internete. Turinys esantis šioje svetainėje yra visiškai nemokamas ir parašytas bendradarbiaujant žmonėms iš viso pasaulio. Šio puslapio esmė yra tai kad bet kuris žmogus, turintis prieigą prie interneto, gali rašyti, taisyti, papildyti šios enciklopedijos informaciją, paprasčiausiai paspausdamas „redaguoti šį puslapį“ nuorodą [11].

Wikipedia naudoja Mediawiki programinę įrangą, kuri yra licencijuota pagal GLP licenziją. Ši programinė įranga yra suprojektuota taip kad galėtų veikti keliuose serveriuose vienu metu ir pritaikyta svetainėms, kurias aplanko milijonai vartotojų per dieną. MediaWiki yra nepaprastai galinga ir kintamo dydžio programinė įranga. Ji naudoja PHP programavimo kalbą, apdoroti ir atvaizduoti duomenims, kurie laikomi MySQL duomenų bazėje [12].

Puslapių atvaizdavimui yra naudojamas wikitext formatas, dėl ko vartotojams nėra būtina žinoti XHTML ar CSS ir jie gali laisvai ir paprastai juos redaguoti. Standartinė svetainės išvaizda yra lengvai keičiama, nes yra galimybė susikurti arba pasirinkti jau egzistuojančių stilių.

Viena iš pagrindinių MediaWiki savybių yra tai kad vartotojui poredagavus puslapį ir jį įkėlus į duomenų bazę, senoji versija nėra ištrinama, todėl tokių būdu yra galimybė atstatyti bet kurią senesnę svetainės versiją ir apsaugoti nuo vandalų ar reklamuotojų, norinčių piktybiškai pakeisti svetainės turinį. MediaWiki taip pat gali valdyti paveikslėlius ir Multimedia failus, kurie yra išsaugomi failų sistemoje. Didelės apimties svetainėms, kuriose apsilanko daugybė vartotojų, MediaWiki palaiko kešavimą ir gali būti lengvai susieta su kitu serveriu naudojant Squid proxy serverio programinę įrangą.

4. Programų sistemų savybių kiekybinis ir/arba kokybinis palyginimas

1 lentelė. Programų savybių palyginimas

	Steve	Flickr	Wikipedia
Žymių naudojimas	+	+	+
Bendras turinio pildymas	+	+	+
Kelių serverių palaikymas vienai svetainei	-	+	+
Galimybe laisvai platinti sukauptus duomenis	+	+	-

2 lentelė. Programų sistemų techninių savybių palyginimas

	Steve	Flickr	Wikipedia
Šablonų naudojimas	+	+	+
Naudojama programavimo kalba	PHP	PHP	PHP
Naudojama DB	MySQL	MySQL	MySQL
Galimybė naudoti kitą DB	+	?	?

5. Bendradarbiavimas (Collaboration)

Turinio valdymo bendradarbiaujant esmė – tai kad papildyti, keisti ar ištrinti svetainės informaciją gali bet kuris interneto vartotojas. Šiuo metu tokiu principu veikia daugybė portalų, o vienas populiariausių – internetinė enciklopedija WikiPedia.

1.4. Atviras turinys

Kuomet turinys yra pildomas bendradarbiaujant, jis vadinamas atviru turiniu. Atviro turinio idėjos šaknys yra kilusios iš atviro kodo programinės įrangos koncepcijos ir gali būti sukurtas bei išleistas laikantis atviro turinio licenzijos sutarties. Iš filosofinės pusės, atviro turinio principas yra tai, kad ši informacija turėtų būti laisvai ir nemokamai panaudota visų kuriems jos reikia. Esminis atviro turinio raktas yra tas kad bet kuri informacija yra laisvai prieinama redagavimui, panaudojimui, platinimui su tam tikrais apribojimais.

Originali OpenContent License (OPL), versija 1.0 sukurta 1998 Liepos 14 [16] nenustato kaip turinio pakeitimai turėtų būti žinomi. Dėl to buvo sukurta Open Publication License (OPvL). Ši licenzija palengvins autoriaus rūpesčius, apsaugant jį nuo to, kad kažkam panaudojus kūrinį ir padarius pakeitimus jis prarastų teises į kūrinį. Trumpai tariant turinys bus laisvai prieinamas naudojimui ar net keitimui bet tai nereikš, kad po to bus galima tvirtinti, kad tai yra jau tavo nuosavybė. Yra aiškiai nustatyta, kad turi būti pažymėta kas buvo pakeista, kada pakeista ir kas pakeitė.

Creative Commons skatina novatorišką bet kokios intelektualinės nuosavybės panaudojimą ir šiuo metu siūlo keturių tipų licenzijas [2]:

- Atribucija (Attribution) – leidžia kopijuoti, platinti, vaizduoti ar atlikti darbą tik jei yra nurodomas autoriaus vardas.
- Nekomercinė (Noncommercial) – leidžia kopijuoti, platinti, vaizduoti ar atlikti darbą tik nekomerciniams tikslams.
- Tik originalūs darbai (No Derivative Works) – leidžia kopijuoti, platinti, vaizduoti ar atlikti tik originalų darbą. Jokių perdarytų ar kitaip pakeistų darbų paremtu originalu.
- Share Alike – leidžia kitiems platinti pakeistus darbus tik pagal tokią licenziją, kuri buvo pritaikyta originaliam darbui.

1.5. Atviro turinio pavyzdžiai

Yra galybė organizacijų kurios veikla susijusi su atviru turiniu. OpenContentList (<http://www.opencontentlist.com/>) yra svetainė ir adresų sąrašas kuriame dalinamasi atviru turiniu iš WEB dienoraščių, enciklopedijų, media duomenų bazių, sukurtų bendradarbiaujant. Open Content Network tikslas yra būti didžiausiu pasaulyje turinio platinimo tinklu. Šis portalas tvirtina, kad greitai metu vartotojai galės atsisiųsti atviro kodo programinę įrangą, filmus, muziką. Open content network yra paremtas P2P technologija, vadinama „Content-Addressable Web“.

Šiuo metu yra pora atviro turinio internetinių enciklopedijų projektų – Nuperia ir Wikipedia. Abu projektai buvo inicijuoti Larry Sanger'io. Nuperia yra atidžiai prižiūrimas projektas, tuo tarpu Wikipedia yra daug labiau atviras ir mažiau formalus projektas. Abu šie projektai yra patalpinti Bomis – WebPortal kompanijos. Nuperia buvo kurį laiką kūrimo stadijoje ir šiuo metu turi ne itin didelį skaičių straipsnių. Šis projektas visiškai patikėtas savanoriams kuriantiems turinį. Straipsniai yra parašyti individualiai o ne kažkokios komandos. Vertinant Nuperia kaip portalą galima pasakyti kad jis dar turi sulaukti datos kai

bus pavadintas sėkmingu projektu. Tuo tarpu Wikipedia startavo 2001 sausį ir 2002 lapkritį turėjo 93245 straipsnių anglų kalba. Vartotojai anonimiškai gali keisti turinį, bet nepaisant to čia galima rasti stebėtinai gerų straipsnių. Taip pat kaip ir Nuperia, Wikipedia visiškai paremta savanorių, bet straipsniai yra sukurti daugiau bendruomenės pastangomis.

Andamooka ([http://www.andamooka.org/.](http://www.andamooka.org/)) talpina atviro turinio knygas, kurias galima skaityti, rašyti joms anotacijas, apie jas diskutuoti. Beveik visos šiuo metu turimos knygos yra apie kompiuterines temas, susijusias su Linux operacine sistema ir panašia programine įranga.

1.6. Atviro turinio modelis

Kuriant atviro turinio modelį, galime mokintis ir remtis iš jau sukurto ir egzistuojančio atviro kodo programinės įrangos modelio.

Atviro kodo idėja yra tai kad programinės įrangos kodas yra atviras ir prieinamas visiems. Programa yra laisvai platinama kartu su kodu, kurį gali bet kas laisvai jį modifikuoti. Taip pat pakeista programinė įranga gali būti toliau modifikuojama ir platinama. Vis dėl to atviro turinio, atviro kodo filosofinė idėja yra pritraukti žmones bendradarbiavimui.

Atviro kodo programinės įrangos metodai yra gerai išvystyti, todėl tai kas yra žinoma apie atviro kodo metodus, galima panaudoti kuriant atviro turinio procesų modelį.

3 lentelė. Pamokos kurias galima išmokti iš atviro kodo programinės įrangos vystymo.

Pamokos:

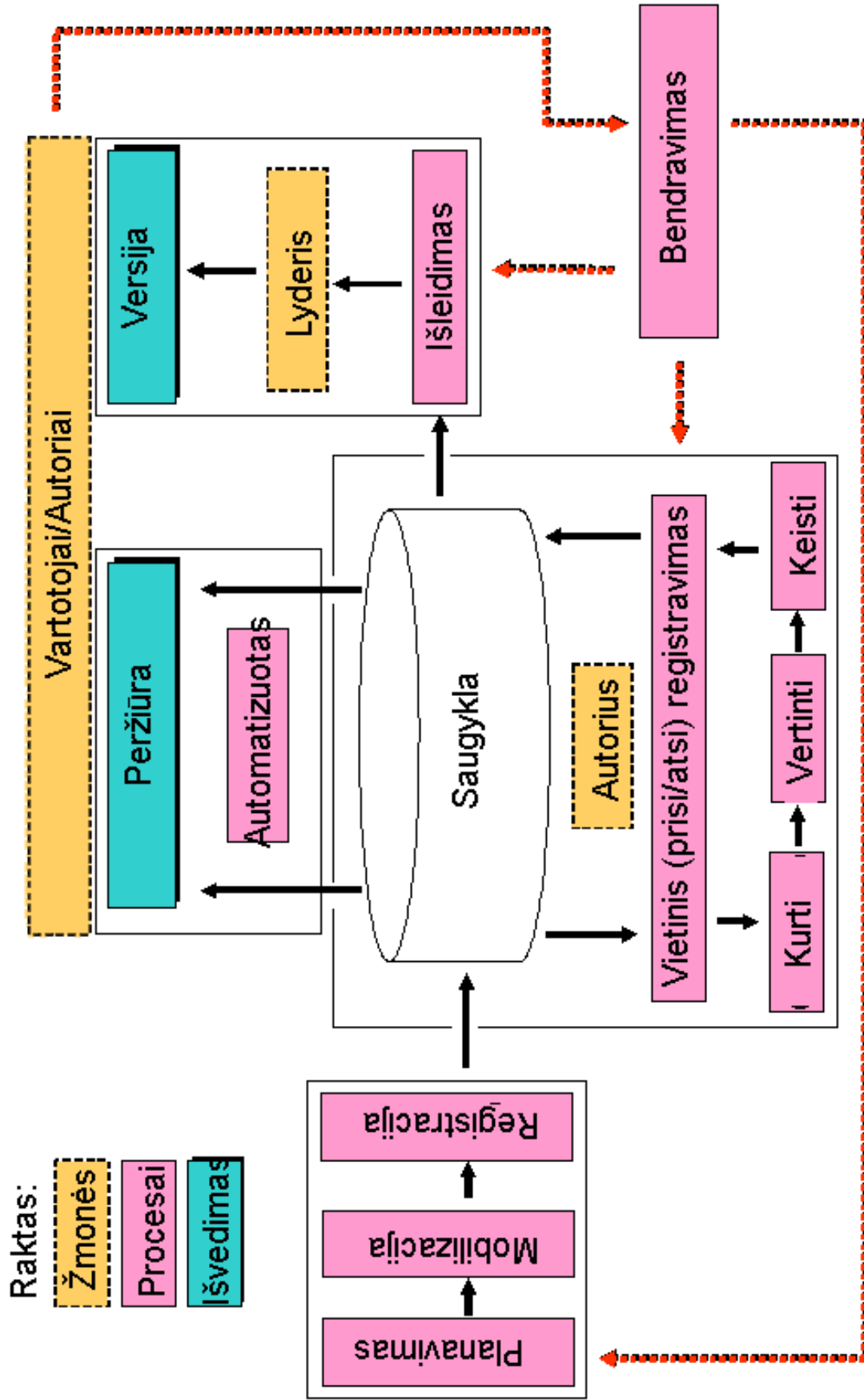
- Veiksmingas bendravimas yra būtinas.
- Versijų kontrolė yra privaloma.
- Formuoti bendras komandas su skirtingais įgūdžiais.
- Pasitikėjimas yra raktinis sėkmės elementas
- Lyderis vaidina esminę rolę kokybės valdyme.
- Vartotojų atsiliepimai yra svarbūs.
- Vystymas yra ciklinis procesas.

1.7. Atviro turinio procesų modelis

Procesų modelio raktiniai elementai yra:

- Procesai
- Įrankiai

- Žmonės



1 Pavėikslas. Žmonės ir procesai įtraukti į atviro turinio kūrimą.

4 Lentelė. Procesai, įrankiai ir žmonės dalyvaujantys atviro turinio kūrimo.

Procesai	Įrankiai	Žmonės
Planavimas	Asmeniniai susitikimai, el. paštu, diskusijų lentos, pokalbių kambariai. Internetinė svetainė kaip bendra sąsaja	Projekto iniciatorius arba grupelė
Mobilizacija	El. paštas, diskusijų lentos, pokalbių kambariai, interneto svetainė	Projekto iniciatorius arba grupelė, naujai prisijungę nariai
Registracija	Interneto svetainė	Projekto iniciatorius arba grupelė
Vietinis atsiregistravimas	Interneto svetainė	Bet kuris komandos narys
Kurti	Plati įvairovė įrankių skirtų redaguoti HTML, Multimedia...	Bet kuris komandos narys
Vertinti	Naršyklė	Autoriai, kompetentingi vartotojai
Keisti	Plati įvairovė įrankių skirtų redaguoti HTML, Multimedia...	Bet kuris komandos narys
Prisiregistravimas	Interneto svetainė	Bet kuris komandos narys
Automatizuota peržiūra	Interneto svetainė	Automatinis procesas
Išleistas turinys	Interneto svetainė, naršyklė	Susitarimas tarp autorių, lyderis, darbo grupė
Bendravimas	Internetinė svetainė su diskusijų lentomis, palaikančiomis el. pašto siuntimą, Pokalbių kambariai.	Vartotojai, kūrėjai

1 Būsena (Priengiamoji): Pirmoji atviro turinio projekto būsena įtraukia parengiamuosius procesus:

- Planavimas.
- Žmonių mobilizavimas.
- Registracija prie projekto.

Planavimas dažniausiai bus inicijuotas mažos grupelės žmonių, kurie turi tokį patį tikslą. Tai gali padaryti ir vienas asmuo. Po šio proceso vyks kitų suinteresuotų žmonių pritraukimas į turinio kūrimo procesą (Mobilizacija). Atėjus naujiems nariams, bus vėl

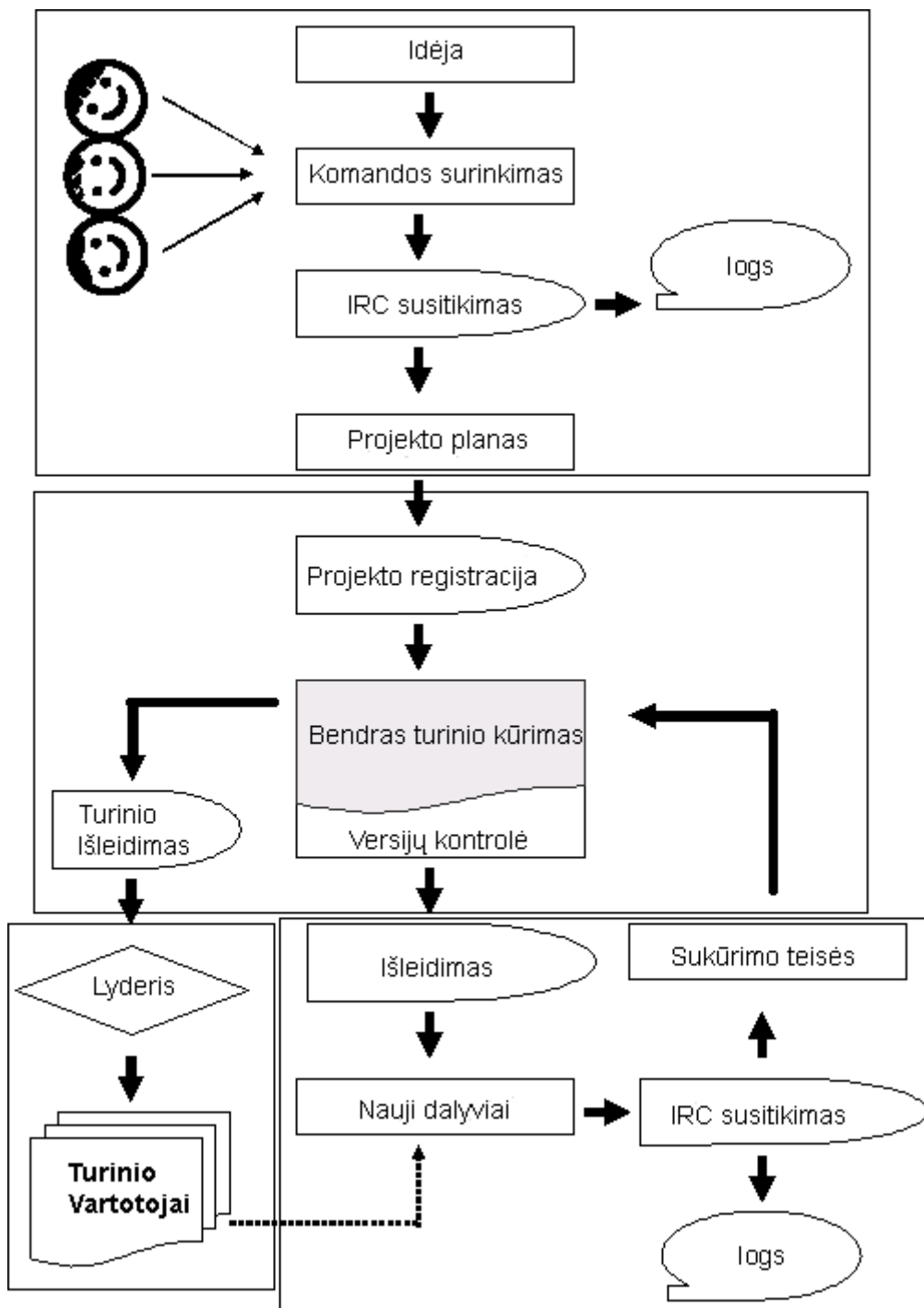
grįžtama prie derybų ir planavimo. Planavimas ir mobilizacija gali būti vykdoma asmeniniuose susitikimuose, tačiau dažniausiai tai atliekama panaudojant tokius įrankius kaip diskusijų lentos, el. pašto sąrašai, pokalbių kambariai. Svetainė pateikianti sąsają bendravimui taip pat pateikia ir prieigą prie projekto registracijos. Šios būsenos pagrindinė veikla yra žmonių mobilizacija.

2 Būsena (Turinio kūrimas): Antroji būsena įtraukia vykstančius procesus turinio kūrime. Kai tik projektas buvo paleistas, kūrimo komanda gauna priėjimą prie saugyklos, kuriame yra saugoma esamos ir ankstesnės turinio versijos. Tai yra labai panašu į programinės įrangos kūrime naudojamą CVS.

3 Būsena (Esamos versijos pateikimas): Trečioji būsena yra automatiškai pateikiama programinės įrangos kuri valdo saugyklą. Ji pateikia esama turinio versiją. Tai būtų panašus procesas vykdant suplanuotus darbus atnaujinant interneto svetainę iš CVS saugyklos. Šio proceso itin svarbi dalis yra gauti atsakymus iš vartotojų ir kitų autorių tam kad būtų gautos išleidimo versijos.

4 Būsena (Išleidimas): Ketvirtoji būsena yra turinio išleidimas viešam naudojimui. Šis procesas turėtų būti automatizuotas programinės įrangos, tačiau prižiūrimas vieno ar kelių lyderiu, kurie užtikrina kad informacija atitinka sutartus standartus viešam naudojimui.

Kitas pavyzdys pažvelgti į atvirą turinį būtų sekti įvykius hipotetiniame projekte. Projektas prasideda kai kažkam kyla idėja ir yra surenkama komanda dirbsianti prie projekto. Komanda susirinktų asmeniškai pabendraut arba tai galėtų daryt naudojant pokalbių kambarius tokius kaip IRC. Susirinkimo metu komanda sukurtų apytikrį planą ir užregistruotų projektą turinio valdymo serveryje („contentforge“) Turinio valdymo serveris pateiktų įrankius bendram turinio kūrimui, versijų kontrolei, tokius kuriuos pateikia SourceForge atviro kodo programinei įrangai.



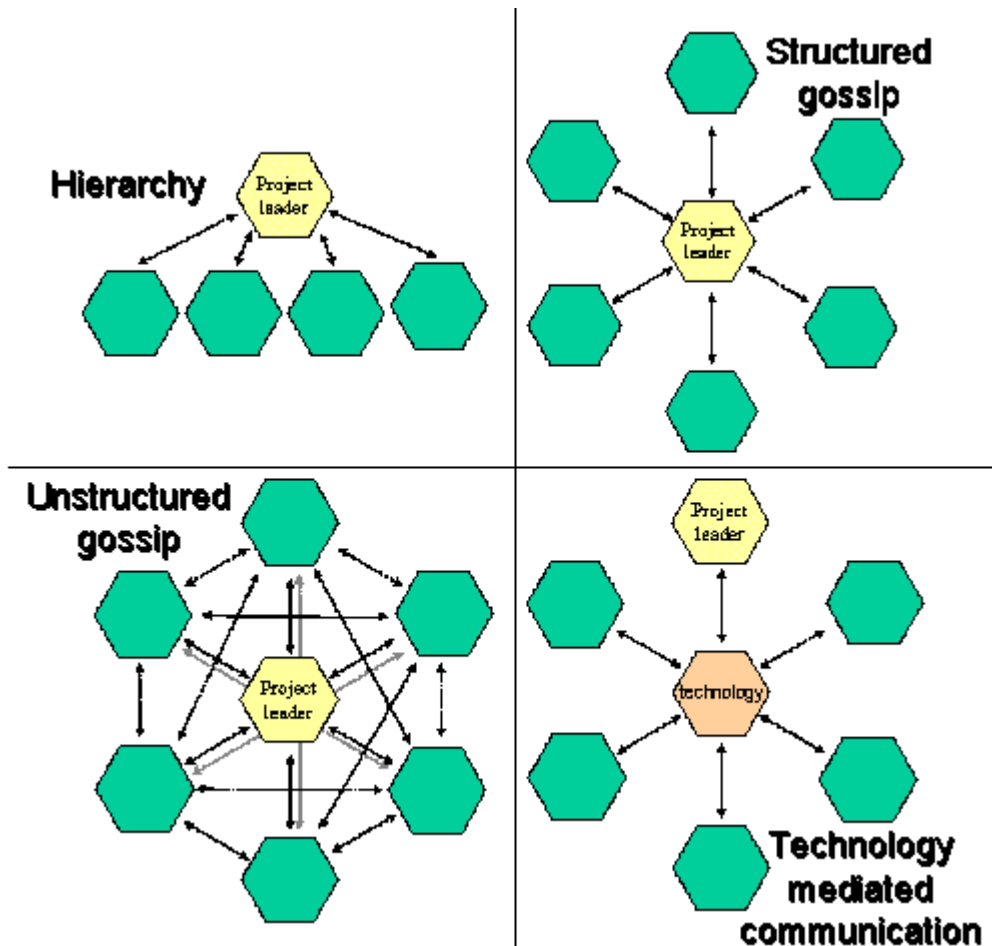
2 Paveikslas. Sekų diagrama hipotetiniam atviro turinio projektui.

Kai tik projektas yra užregistruotas, informacijos kūrėjai pradeda gaminimo procesą. Kai yra sukurta užtektinai informacijos, kad ji būtų naudinga, yra sudaromas tvarkaraštis išleidimui, į kurį yra įtraukiamas lyderis, atsakingas už informacijos kokybę ir standartus.

Bendravimas yra itin svarbus visose projekto stadijose. Svarbu yra kad įrankiai skirti bendravimui būtų visais atžvilgiais cikliškas. Bendravimas gali būti (3 Paveikslas):

- Hierarchinis
- Struktūriškas šnekėjimas

- Nestruktūriškas šnekėjimas
- Technologijoms tarpininkaujant



3 Paveikslas. Bendravimo būdai.

1.8. Dvi atviro turinio kūrimo kraštutinybės

Paprastai atviras turinys yra kuriamas „Kūrimo bendruomenių“, tačiau atviras turinys gali būti sukurtas centralizuotu leidybos procesu, kuriame praktiškai nėra arba yra tik labai mažai bendradarbiavimo už bendruomenės ribų.

5 Lentelė. Dviejų atviro turinio kūrimo kraštutinybių charakteristikos.

	Centralizuoto leidimo charakteristikos	Kūrimo bendruomenės charakteristika
Organizacijos struktūra	Vienas arba daugiau ekspertų, kurie kuria turinį vienai organizacijai	Vienas arba daugiau ekspertų, kurie kuria turinį kelioms organizacijai
Pagrindinių vartotojų įtraukimas	Vartotojai nėra įtraukti į konstravimą.	Vartotojai yra įtraukti į konstravimą.
Naujų kūrėjų priėmimas	Tik išimtiniais atvejais	Bet kuriuo atveju.
Pavyzdžiai	MIT Open Courseware	Webpedia, Wikipedia, AVOIR

6. Įgyvendinimo problemos

Bandant įgyvendinti projektą kuris paremtas bendru turinio pildymu su galimybe informaciją saugoti visomis Europos kalbomis, galime susidurti su šiomis problemomis:

- Turinio redagavimo bendradarbiaujant problemos.
- Svetainės turinio formavimas, nemokant HTML, CSS.
- Daugiakalbiškumo problemos.

1.9. Turinio redagavimo bendradarbiaujant problemos

Kuomet yra galimybė bet kuriam vartotojui pakeisti, sukurti ar ištrinti turinį, atsiranda tikimybė jog turinį bandys specialiai sugadinti priešiška nusiuteikę vartotojai, reklamos robotai.

1.9.1. Priešiška nusiuteikę vartotojai.

Visuomet atsiras tokių vartotojų, kurie lengvai gavę priėjimą prie puslapio redagavimo, stengsis pakeisti turinį į neatitinkantį tematikos, panaudos necenzūrinių žodžių arba tiesiog viską bandys ištrinti. Kaip galima nuo to apsisaugoti?

Pirmiausia norintiems keisti informaciją turėtų būti privaloma registracija. Padidėjus formalumams, dauguma nesistengs prisijungti prie bendruomenės vien tam kad galėtų gadinti turinį.

Dauguma vandalų panaudos necenzūrinius žodžius, todėl reiktų sukurti algoritmus, kurie filtruotu galimus vandališkus pakeitimus.

Visgi, 100% automatiškai atfiltruoti ar pakeitimai yra tikri neįmanoma, todėl reiktų saugoti visas ankstesnes duomenų kopijas, kad bet kuriuo metu būtų galimybė atstatyti senesnį puslapio turinį

1.9.2. Reklamos robotai

Populiarėjant internetinei prekybai, atsiranda reklaminiai robotai kurie reklamuoja produktus internete. Šiuo metu yra sukurti keli apsaugos būdai.

- Privaloma registracija ir prisijungimas siunčiant sistemos sugeneruotą slaptažodį.
- Prieš pateikiant informaciją yra reikalaujama įvesti tekstą iš automatiškai sugeneruoto paveiksluko, kas teoriškai turėtų būti įmanoma padaryti tik žmogui.

Visgi jei yra pageidaujama leisti talpinti informaciją ir neregistruotiems vartotojams tai reiktų naudoti aukščiau minėtą apsaugą su tekstu ant paveiksluko arba kurti algoritmus kurie bandytų filtruoti robotų žinutes. Tai neturėtų būti sudėtinga nes populiariausios robotų žinutės yra apie medikamentus arba programinę įrangą pridodant nuorodas į svetaines.

1.10. Svetainės turinio formavimas, nemokant HTML, CSS

Kadangi turinys bus laisvai redaguojamas eilinių interneto vartotojų, didžioji jų dalis visiškai nemokės HTML ar CSS, todėl atsiranda nauja problema. Kaip formuoti svetainės turinį.

Šiuo metu yra šios galimybės [13]:

- Naudoti WYSIWYG redaktoriaus
- Naudoti specialias žymes (BB code arba wikitext)

1.10.1.WYSIWYG redaktoriai

WYSIWYG HTML redaktoriai yra itin populiarūs ir šiuo metu jų yra gana plati įvairovė: htmlArea, TinyMCE, FCKeditor ir k.t. Tačiau kaip ir visi jie turi trūkumų (pagal [13]):

- Šie redaktoriai padidina puslapio dydį, dėl ko pailgėja krovimosi laikas.
- HTML kodas sugeneruotas šių redaktorių dažnai neatitinka esamų standartų.
- Kažkurie redaktoriai nepalaiko CSS, dėl ko tenka įvesti daugiau informacijos apie teksto spalvas, dydį ir k.t. visiems puslapiams.

- Naudojant šiuos redaktorius, tenka dažniau žiūrėti kaip puslapis atrodo skirtingose naršyklėse.
- Pagrindinė problema ta, kad šie redaktoriai naudoja JavaScript, tačiau ne visi vartotojai yra suteikę teises veikti JavaScript, todėl pas juos šie redaktoriai tiesiog neveiks.

1.10.2. Supaprastinto žymėjimo kalbos.

BBCode, WikiText – tai supaprastinta žymėjimo kalba, kuri leidžia pakeisti teksto atvaizdavimą nemokant HTML, kuri yra žymiai sudėtingesnė. Bene vienintelis privalumas naudojant šias supaprastintas žymėjimo kalbas yra tas, kad jos rašomos į formą ir nereikalauja jokių papildomų JavaScript kodų.

Bene vienintelis tačiau itin svarbus nepatogumas naudoti supaprastintas žymėjimo kalbas, tai kad jos yra ribotų galimybių, nėra tiesiogiai matomas būsimas tekstas, todėl vartotojui WYSIWYG redaktoriai yra priimtinesni.

1.11. Daugiakalbiškumo problemos

Norint, kad svetainė taptų dar populiareesnė tarp viso pasaulio vartotojų – yra diegiama daugiakalbystė kad vartotojai svetainę matytų jų gimtąja kalba. Tačiau neišvengiamai kūrėjai susiduria su šiomis problemomis (pagal [14]):

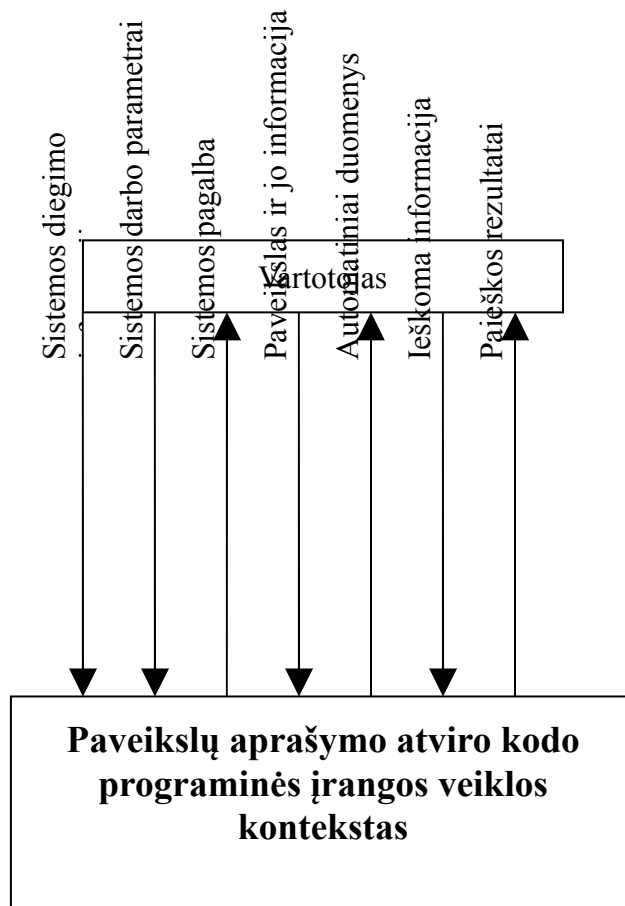
- Kaip apibrėžti ir susisteminti failus ir stilius
- Kaip pateikti vartotojui tinkamą kalbą
- Kaip elgtis su skirtingų kalbų simbolių rinkinių informacija.
- Užtikrinti kalbų logiškumą

7. PAVEIKSLŲ APRAŠYMO ATVIRO KODO PROGRAMINĖS ĮRANGOS SISTEMOS PROJEKTAS

8. Funkciniai reikalavimai

1.12. Veiklos sfera

Veiklos kontekstas (pateikiama konteksto diagrama)



4 Paveikslas. Veiklos konteksto diagrama

Veiklos padalinimas

6 Lentelė. Veiklos padalinimas.

Eil. Nr.	Įvykio pavadinimas	Įeinantys/išeinantys informacijos srautai
1	<i>Vartotojas diegia sistemą</i>	<i>Sistemos diegimui skirta informacija (in)</i>
2	<i>Vartotojas nustato parametrus</i>	<i>Sistemos darbo parametrai (in)</i>
3	<i>Vartotojas gauna informaciją apie sistemos darbą, jam rūpimus klausimus</i>	<i>Sistemos pagalba (out)</i>
4	<i>Vartotojas pateikia paveikslą ir informaciją apie jį</i>	<i>Paveikslas (in)</i>
5	<i>Vartotojas gauna sistemos automatinio būdu sugeneruotą papildomą informaciją apie paveikslą</i>	<i>Automatiniai duomenys (out)</i>
6	<i>Vartotojas ieško informacijos apie muziejines vertybes</i>	<i>Paieškos raktažodžiai (in)</i>
7	<i>Vartotojas gauna paieškos rezultatus iš sistemos</i>	<i>Paieškos rezultatai (out)</i>

1.13. Produkto veiklos sfera

Sistemos ribos – panaudos atvejų modelis

Panaudos atvejų ir aktorių sąrašai

Aktoriai:

Sistemos vartotojas

Panaudos atvejai:

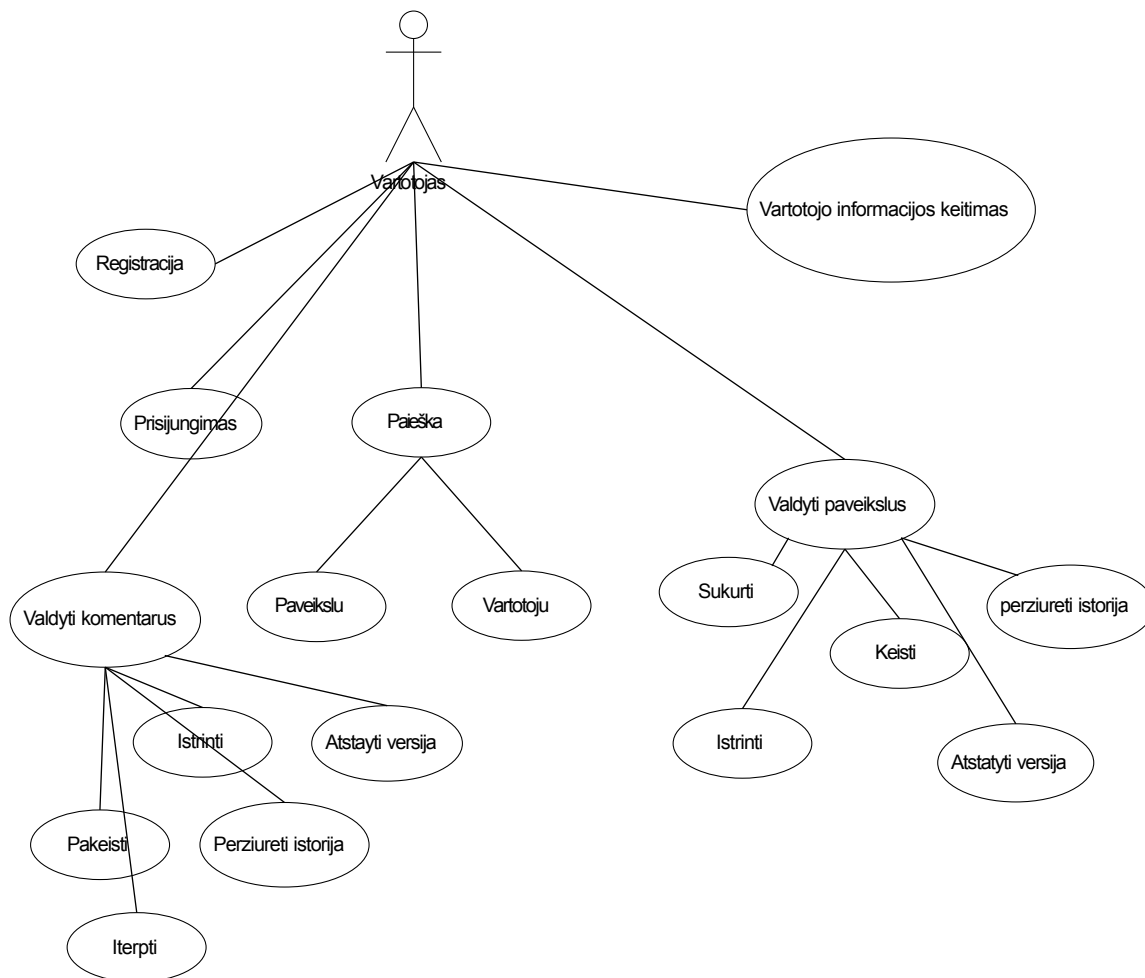
1. Registracija
2. Prisijungimas
3. Vartotojo informacijos keitimas
4. Valdyti paveikslus
 - a. Kurti
 - b. Redaguoti
 - c. Peržiūrėti keitimus
 - d. Atstatyti keitimus
5. Valdyti komentarus
 - a. Kurti
 - b. Redaguoti
 - c. Peržiūrėti keitimus
 - d. Atstatyti keitimus

6. Paieška

a. Vartotojų

b. Paveikslų

Panaudos atvejų diagrama



4 Paveikslas. Panaudos atvejų diagrama

Aktorių funkcijos

Sistemos vartotojas – asmuo, kuris naudojami programine įranga

Panaudos atvejų specifikacijos

1. Registracija

Tikslas: Užregistruoti į sistemą naują vartotoją

Aktoriai: Vartotojas

Ryšiai su kitais PA: nėra.

Nefunkciniai reikalavimai: Prisijungimo vardas turi būti unikalus ir negali egzistuoti sistemoje.

Sužadinimo sąlygą: Pasirenkamas meniu punktas „Registruotis“

Prieš sąlygos: nėra

Po sąlygos: Vartotojas įvedamas į DB

2. Prisijungti

Tikslas: Vartotojui prisijungti prie sistemos

Aktoriai: Vartotojas

Ryšiai su kitais PA: Šis PA yra paveldimas „Tvarkyti paveikslus“, Tvarkyti komentarus“ ir „Keisti vartotojo informaciją“ panaudos atvejų.

Nefunkciniai reikalavimai: Turi būti įvestas prisijungimo vardas ir slaptažodis.

Sužadinimo sąlygą: Pasirenkamas meniu punktas „Prisijungti“

Prieš sąlygos: Vartotojas turi būti užsiregistravęs.

Po sąlygos: Vartotojas suveda savo identifikavimo informaciją ir yra atpažįstamas kaip egzistuojantis sistemos vartotojas.

3. Vartotojo informacijos keitimas

Tikslas: Vartotojas pasikeičia savo asmeninę informaciją

Aktoriai: Vartotojas

Ryšiai su kitais PA: Šis PA paveldi „Prisijungti“ panaudos atvejį.

Nefunkciniai reikalavimai: Turi būti įvedami korektišką informacija.

Sužadinimo sąlygą: Pasirenkamas meniu punktas „Keisti informaciją“

Prieš sąlygos: Vartotojas turi būti prisijungęs.

Po sąlygos: Vartotojas atnaujina informaciją.

4. Valdyti paveikslus

Tikslas: Vartotojas sukuria/pakeičia paveikslo informaciją

Aktoriai: Vartotojas

Ryšiai su kitais PA: Šis PA paveldi „Prisijungti” panaudos atvejį.

Nefunkciniai reikalavimai: Turi būti įvedama korektiška informacija.

Sužadinimo sąlygą: Pasirenkamas meniu punktas „Įkelti naują paveikslą“ arba redaguojamas egzistuojantis paveikslas

Prieš sąlygos: Vartotojas turi būti prisijungęs.

Po sąlygos: Vartotojas sukuria arba atnaujina informaciją apie paveikslą.

5. Valdyti komentarus

Tikslas: Vartotojas sukuria/pakeičia paveikslo komentarą

Aktoriai: Vartotojas

Ryšiai su kitais PA: Šis PA paveldi „Prisijungti” panaudos atvejį.

Nefunkciniai reikalavimai: Turi būti įvedama korektiška informacija.

Sužadinimo sąlygą: Pasirenkamas meniu punktas „Komentuoti“ arba redaguojamas egzistuojantis komentaras

Prieš sąlygos: Vartotojas turi būti prisijungęs.

Po sąlygos: Vartotojas sukuria arba atnaujina paveikslo komentarą.

6. Paieška

Tikslas: Vartotojas ieško paveikslų

Aktoriai: Vartotojas

Ryšiai su kitais PA: nėra.

Nefunkciniai reikalavimai: Turi būti įvestas paieškos raktas.

Sužadinimo sąlygą: Panaudojama paieškos forma.

Prieš sąlygos: nėra

Po sąlygos: Surandami produktai atitinkantys paieškos raktą.

9. Funkciniai reikalavimai ir reikalavimai duomenims

1.14. Funkciniai reikalavimai

7 Lentelė. Funkciniai reikalavimai, 1 reikalavimas

Reikalavimas #1	Reikalavimo tipas: 10	Įvykis/panaudojimo atvejis #1
Aprašymas	Galimybė registruotis ir užpildyti demografinę apklausą	
Pagrindimas	Sistema turi turėti vartotojus kurie galės redaguoti paveikslus	
Šaltinis	Užsakovas	
Tinkamumo kriterijus	Sistema turi vartotojų valdymą	
Užsakovo patenkinimas	5	Užsakovo nepatenkinimas 5
Priklausomybė	Nėra	Konfliktai Nėra
Istorija	Užregistruotas 2006-10-10	

8 Lentelė. Funkciniai reikalavimai, 2 reikalavimas

Reikalavimas #2	Reikalavimo tipas: 10	Įvykis/panaudojimo atvejis #6
Aprašymas	Galimybė peržiūrėti paveikslus arba ieškoti pagal meta duomenis	
Pagrindimas	Sistemoje turi būti paveikslų grupavimas ir paieška	
Šaltinis	Užsakovas	
Tinkamumo kriterijus	Yra paveikslų grupės ir paieška	
Užsakovo patenkinimas	5	Užsakovo nepatenkinimas 5
Priklausomybė	Nėra	Konfliktai Nėra
Istorija	Užregistruotas 2006-10-10	

9 Lentelė. Funkciniai reikalavimai, 3 reikalavimas

Reikalavimas #3	Reikalavimo tipas: 10	Įvykis/panaudojimo atvejis #4
Aprašymas	Galimybė paveikslams priskirti žymes	
Pagrindimas	Paveikslai turi savo žymes pagal kurias bus vykdoma paieška	
Šaltinis	Užsakovas	
Tinkamumo kriterijus	Paveikslams galima priskirti žymes	
Užsakovo patenkinimas	5	Užsakovo nepatenkinimas 5
Priklausomybė	Nėra	Konfliktai Nėra
Istorija	Užregistruotas 2006-10-10	

10 Lentelė. Funkciniai reikalavimai, 4 reikalavimas

Reikalavimas #4	Reikalavimo tipas: 10	Įvykis/panaudojimo atvejis #4
Aprašymas	Peržiūrėti, redaguoti paveikslų žymes	
Pagrindimas	Paveikslų žymės gali būti redaguojamos	
Šaltinis	Užsakovas	
Tinkamumo kriterijus	Yra padarytas žymių valdymas	
Užsakovo patenkinimas	5	Užsakovo nepatenkinimas 5
Priklausomybė	Nėra	Konfliktai Nėra
Istorija	Užregistruotas 2006-10-10	

11 Lentelė. Funkciniai reikalavimai, 5 reikalavimas

Reikalavimas #5	Reikalavimo tipas: 10	Įvykis/panaudojimo atvejis #5
------------------------	------------------------------	--------------------------------------

Aprašymas	Paveikslų komentavimas		
Pagrindimas	Muziejinių vertybių paveikslai gali būti komentuojami		
Šaltinis	Užsakovas		
Tinkamumo kriterijus	Vartotojai turi galimybę komentuoti paveikslus		
Užsakovo patenkinimas	5	Užsakovo nepatenkinimas	5
Priklausomybė	Nėra	Konfliktai	Nėra
Istorija	Užregistruotas 2006-10-10		

12 Lentelė. Funkciniai reikalavimai, 6 reikalavimas

Reikalavimas #6	Reikalavimo tipas: 10	Įvykis/panaudojimo atvejis #5	
Aprašymas	Paveikslų komentarų valdymas		
Pagrindimas	Paveikslų komentarai gali būti redaguojami, trinami.		
Šaltinis	Užsakovas		
Tinkamumo kriterijus	Komentarai yra redaguojami.		
Užsakovo patenkinimas	5	Užsakovo nepatenkinimas	5
Priklausomybė	Nėra	Konfliktai	Nėra
Istorija	Užregistruotas 2006-10-10		

13 Lentelė. Funkciniai reikalavimai, 7 reikalavimas

Reikalavimas #7	Reikalavimo tipas: 10	Įvykis/panaudojimo atvejis #5	
Aprašymas	Žymių išskyrimas iš komentarų		
Pagrindimas	Paveikslų žymės yra automatiškai išskiriamos iš komentarų ir išverčiamos į kitas kalbas.		
Šaltinis	Užsakovas		
Tinkamumo kriterijus	Yra įgyvendinamas automatinis žymių išskyrimas iš paveikslų komentarų.		
Užsakovo patenkinimas	5	Užsakovo nepatenkinimas	5
Priklausomybė	Nėra	Konfliktai	Nėra
Istorija	Užregistruotas 2006-10-10		

14 Lentelė. Funkciniai reikalavimai, 8 reikalavimas

Reikalavimas #8	Reikalavimo tipas: 10	Įvykis/panaudojimo atvejis #4	
Aprašymas	Peržiūrėti paveikslų žymų redagavimo istoriją		
Pagrindimas	Paveikslų žymų atstatymas		
Šaltinis	Užsakovas		
Tinkamumo kriterijus	Yra padarytas žymių valdymas		
Užsakovo patenkinimas	5	Užsakovo nepatenkinimas	5
Priklausomybė	Nėra	Konfliktai	Nėra
Istorija	Užregistruotas 2006-10-10		

15 Lentelė. Funkciniai reikalavimai, 9 reikalavimas

Reikalavimas #9	Reikalavimo tipas: 10	Įvykis/panaudojimo atvejis #4	
Aprašymas	Atstatyti ankstesnę žymos versiją		
Pagrindimas	Paveikslų žymų atstatymas		
Šaltinis	Užsakovas		
Tinkamumo kriterijus	Yra padarytas žymių valdymas		
Užsakovo patenkinimas	5	Užsakovo nepatenkinimas	5
Priklausomybė	Nėra	Konfliktai	Nėra
Istorija	Užregistruotas 2006-10-10		

16 Lentelė. Funkciniai reikalavimai, 10 reikalavimas

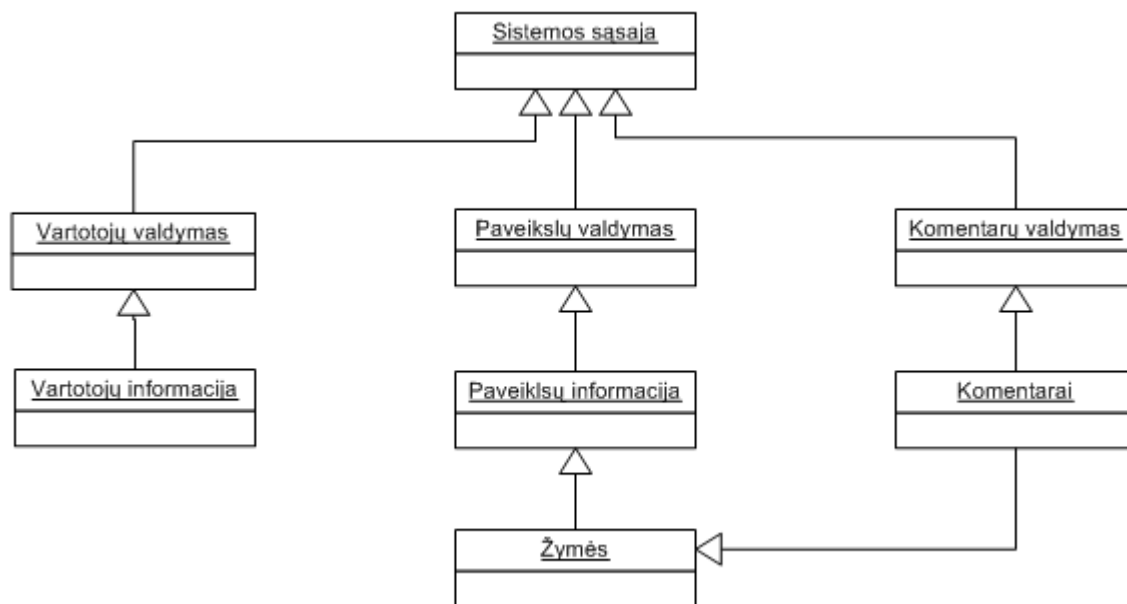
Reikalavimas #10	Reikalavimo tipas: 10	Įvykis/panaudojimo atvejis #4
Aprašymas	Peržiūrėti paveikslų komentaro redagavimo istoriją	
Pagrindimas	Paveikslų žymų atstatymas	
Šaltinis	Užsakovas	
Tinkamumo kriterijus	Yra padarytas žymių valdymas	
Užsakovo patenkinimas	5	Užsakovo nepatenkinimas 5
Priklausomybė	Nėra	Konfliktai Nėra
Istorija	Užregistruotas 2006-10-10	

17 Lentelė. Funkciniai reikalavimai, 11 reikalavimas

Reikalavimas #11	Reikalavimo tipas: 10	Įvykis/panaudojimo atvejis #5
Aprašymas	Atstatyti ankstesnę komentaro versiją	
Pagrindimas	Paveikslų žymų atstatymas	
Šaltinis	Užsakovas	
Tinkamumo kriterijus	Yra padarytas žymių valdymas	
Užsakovo patenkinimas	5	Užsakovo nepatenkinimas 5
Priklausomybė	Nėra	Konfliktai Nėra
Istorija	Užregistruotas 2006-10-10	

1.15. Reikalavimai duomenims

Preliminari sistemos objektų bei jų ryšių schema:



5 Paveikslas. Preliminari sistemos objektų bei jų ryšių schema

10. Nefunkciniai reikalavimai

1.16. Reikalavimai sistemos išvaizdai

18 Lentelė. Reikalavimai sistemos išvaizdai

Reikalavimas #8	Reikalavimo tipas: 11	Įvykis/panaudojimo atvejis #	
Aprašymas	Patogus valdymas		
Pagrindimas	Patogus valdymas užtikrina laiko sąnaudų sumažinimą bei efektyvesnę sistemos naudojimą; taip pat tai vienas iš svarbiausių sistemos panaudojamumo rodiklių		
Šaltinis	Užsakovas		
Tinkamumo kriterijus	Valdymas gali būti atliekamas pele, yra alternatyvos komandoms atlikti		
Užsakovo patenkinimas	5	Užsakovo nepatenkinimas	5
Priklausomybė	Nėra	Konfliktai	Nėra
Istorija	Užregistruotas 2006-10-10		

1.17. Reikalavimai panaudojamumui

19 Lentelė. Reikalavimai panaudojamumui

Reikalavimas #9	Reikalavimo tipas: 12	Įvykis/panaudojimo atvejis #	
Aprašymas	Paprastas panaudojimas be apsimokymo		
Pagrindimas	Sistemos vartotojai gali turintis tik pagrindines darbo kompiuteriu žinias, todėl sistemos paprastumas lems jų požiūrį į sistemą		
Šaltinis	Potencialus vartotojas		
Tinkamumo kriterijus	Programos sėkmingas panaudojimas be apsimokymo 80% atvejų		
Užsakovo patenkinimas	5	Užsakovo nepatenkinimas	5
Priklausomybė	Nėra	Konfliktai	Nėra
Istorija	Užregistruotas 2006-10-10		

1.18. Reikalavimai vykdymo charakteristikoms

20 Lentelė. Reikalavimai vykdymo charakteristikoms

Reikalavimas #10	Reikalavimo tipas: 13	Įvykis/panaudojimo atvejis #	
Aprašymas	Išplečiamumas		
Pagrindimas	Sistema gali būti vystoma toliau, todėl ji turi būti lengvai plečiama		
Šaltinis	Užsakovas		
Tinkamumo kriterijus	Programos kodas, failai yra tinkamai suprojektuoti tolimesniam sistemos vystymui (įdiegti objektiniai projektavimo principai)		
Užsakovo patenkinimas	5	Užsakovo nepatenkinimas	5
Priklausomybė	Nėra	Konfliktai	Nėra

1.19. Reikalavimai sistemos priežiūrai

21 Lentelė. Reikalavimai sistemos priežiūrai

Reikalavimas #11	Reikalavimo tipas: 14	Įvykis/panaudojimo atvejis #
Aprašymas	Lengva programos priežiūra	
Pagrindimas	Kadangi sistema naudojama daugelio vartotojų bei nėra numatytas aktyvus sistemos palaikymas bei atnaujinimas, sistema turi nereikalauti sudėtingų palaikymo veiksmų	
Šaltinis	Užsakovas	
Tinkamumo kriterijus	Sistemos pagalba galima išspręsti kilusias problemas; nereikia sistemos projektuotojų įsikišimo	
Užsakovo patenkinimas	5	Užsakovo nepatenkinimas 5
Priklausomybė	Nėra	Konfliktai Nėra
Istorija	Užregistruotas 2006-10-10	

11. Architektūros specifikacija

1.20. Apžvalga

Dokumentas aprašo paveikslų aprašymo atviro kodo programinės įrangos architektūrą. Sistemos pristatymo eiliškumas pateikiamas skyriuje „Architektūros pateikimas“. Sistemos nefunkciniai reikalavimai ir apribojimai pateikiami skyriuje „Architektūros tikslai ir apribojimai“. Panaudojimo atvejai pateikiami skyrelyje „Panaudojimo atvejų vaizdas“. Sistemos išskaidymas į paketus bei statinė sistemos struktūra pateikiama skyriuose „Sistemos statinis vaizdas“. Sistemos procesai aprašomi skyriuje „Sistemos dinaminis vaizdas“. Sistemos išdėstymas ir techninė įranga pateikiama „išdėstymo vaizdas“. Skyriuje „Išdėstymo vaizdas“ pateikiamas techninės įrangos vaizdas. „Duomenų vaizdas“ skyriuje atvaizduota sistemos duomenų struktūra. Skyriuje „Kokybė“ aprašoma, kaip architektūra įtakoja kokybės parametrus, tokius kaip išplečiamumas, pernešamumas, patikimumas.

1.21. Architektūros pateikimas

Sistemos architektūra pateikiama keliais aspektais: panaudojimo atvejų (PA), procesų, išdėstymo. Visa tai pateikiama diagramose su paaiškinimais, sukurtose Rational Rose įrankiu. Sistemos architektūra pateikiama remiantis RUP rekomendacijomis. Sistemos specifikacija pateikiama:

1. Panaudojimo atvejų vaizdu (panaudojimo atvejų UML diagrama)
2. Sistemos statiniu vaizdu (Klasių diagramos, skaidymas į paketus)
3. Procesų vaizdais (Veiklos, sekų diagramos)
4. Išdėstymo vaizdu (išdėstymo diagrama)
5. Duomenų vaizdu (duomenų modelis)

1.22. Architektūros tikslai ir apribojimai

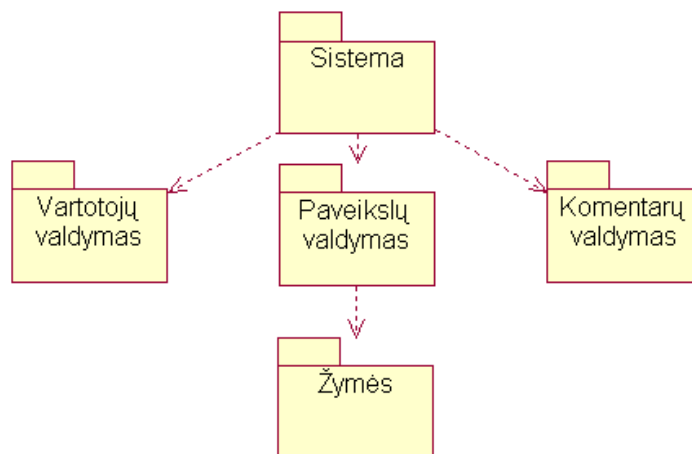
- Sistemos architektūra turi būti parinkta taip, kad ją galima būtų nesunku išplėsti ar prijungi naujus komponentus;
- Architektūra turi leisti pakartotinai panaudoti sistemos komponentus (analizės modulį);

- Architektūra turi būti tokia, kad nesudarytų sistemos patikimumo problemų bei užtikrintų sistemos vientisumą;
- Sudarant sistemos architektūrą turi būti atsižvelgta į būtinas programos vykdymo charakteristikas, apibrėžtas reikalavimų specifikacijoje;

1.23. *Sistemos statinis vaizdas*

1.23.1. Apžvalga

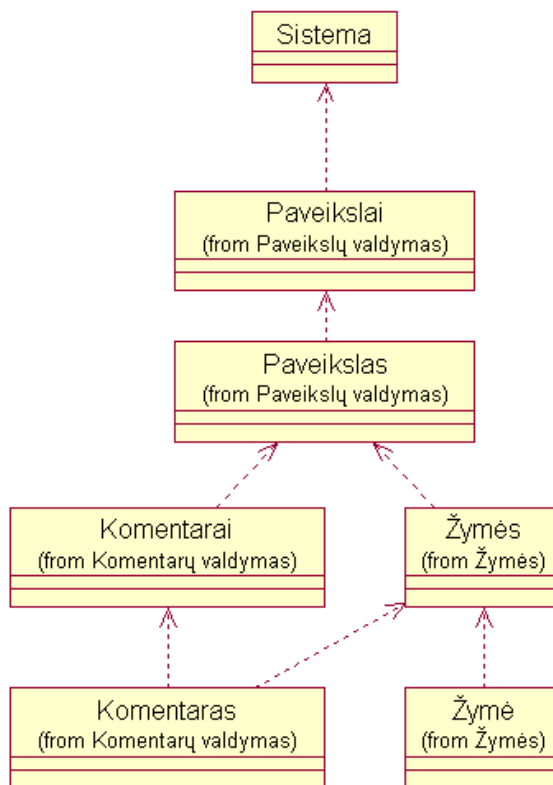
Šis skyrius aprašo sistemos statinę sistemos struktūrą. Pateikiamas sistemos išskaidymas į loginius paketus ir juos sudarančias klases. Išskaidymas į paketus pasirinktas pagal sistemos atliekamas funkcijas, kadangi bus reikalinga pakartotinai panaudoti tam tikras sistemos funkcijas.



6 Paveikslas. Paketų detalizavimas.

Sistema

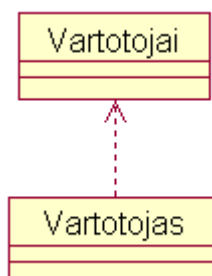
Bendrai valdo sistemą ir paveldi visas klases.



7 Paveikslas. Sistemos detalizavimas.

Vartotojų valdymas

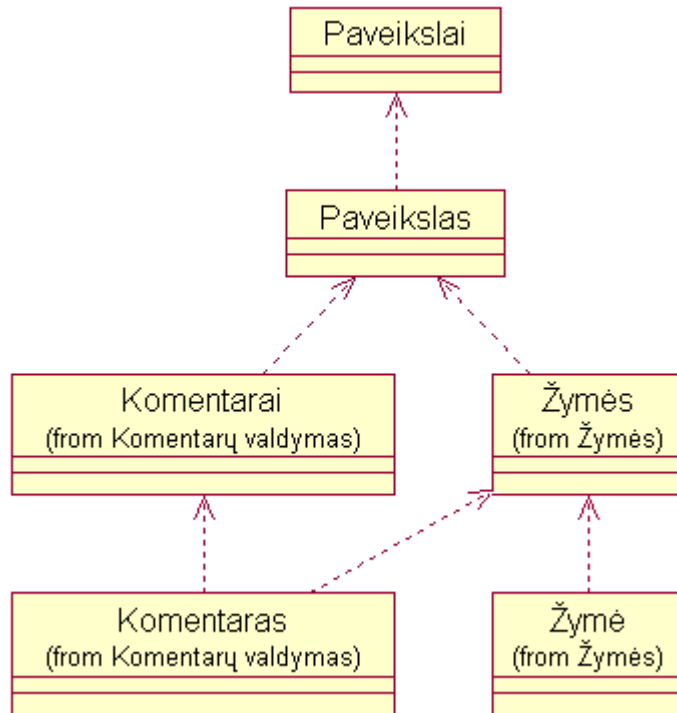
Paketą sudaro dvi klasės. Vartotojai – valdo vartotojų sąrašus ir Vartotojas – valdo atskirai vieną vartotoją.



8 Paveikslas. Vartotojų valdymo paketas.

Paveikslai

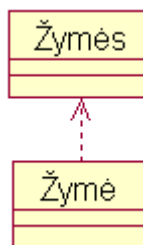
Paketą sudaro dvi klasės. Paveikslai – valdo paveikslų sąrašus ir Paveikslas – valdo atskirai vieną vartotoją. Papildomai pridėtos klasės iš Žymių ir komentarų valdymo paketų kad būtų matomi sąryšiai tarp klasių.



9 Paveikslas. Paveikslų valdymo paketas.

Žymės

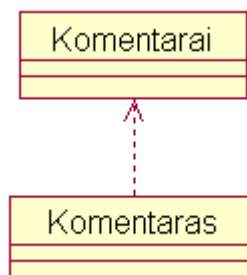
Paketą sudaro dvi klasės. Žymės – valdo paveikslų žymių sąrašus ir Žymė – valdo atskirai vieną žymę.



10 Paveikslas. Žymių valdymo paketas.

Komentarų valdymas

Paketą sudaro dvi klasės. Komentarai – valdo paveikslų komentarų sąrašus ir Komentaras – valdo atskirai vieną komentarą.

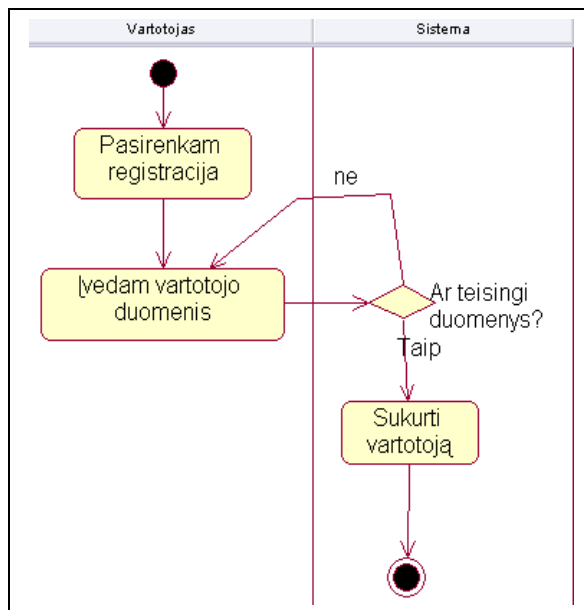


11 Paveikslas. Komentarų valdymo paketas.

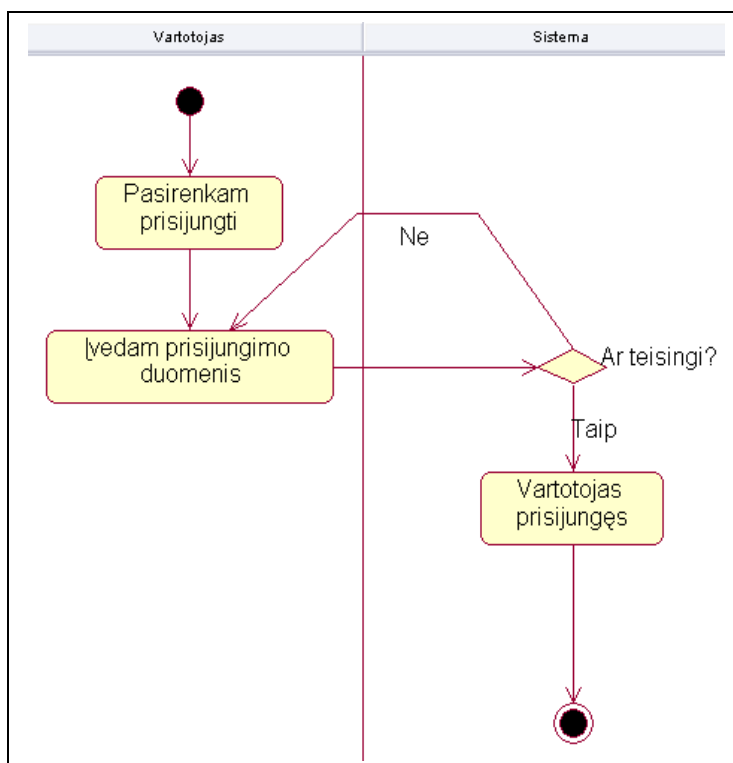
1.24. Sistemos dinaminis vaizdas

1.24.1.PA veiklos diagramos

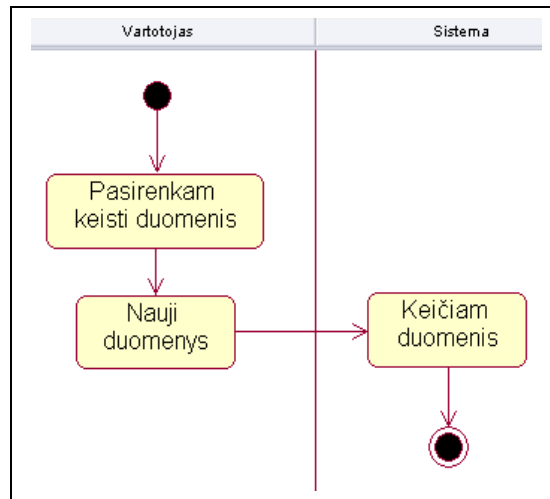
Veiklos diagramos parodo sistemos dinaminį vaizdą, kaip pagrindą naudojant panaudos atvejus. Diagramose veikla yra skirstoma į vartotojo bei sistemos atliekamą veiklą.



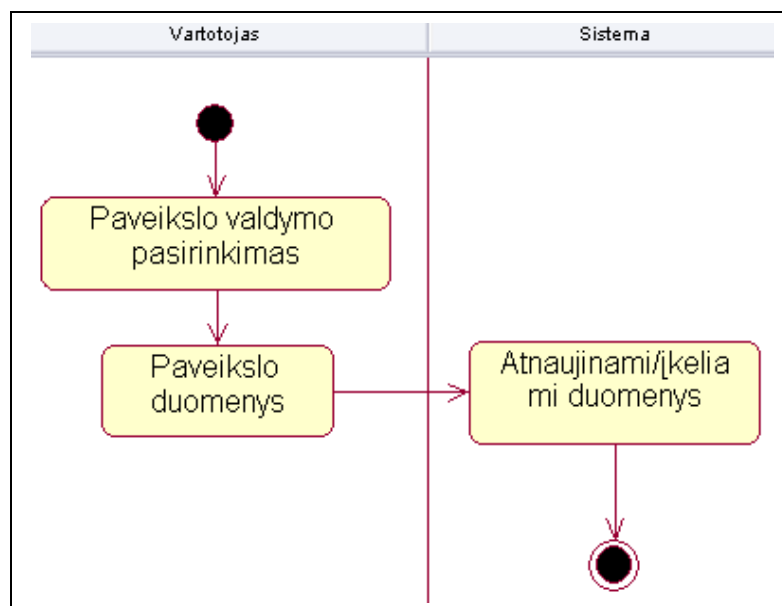
12 Paveikslas. Vartotojo registracijos veiklos diagrama



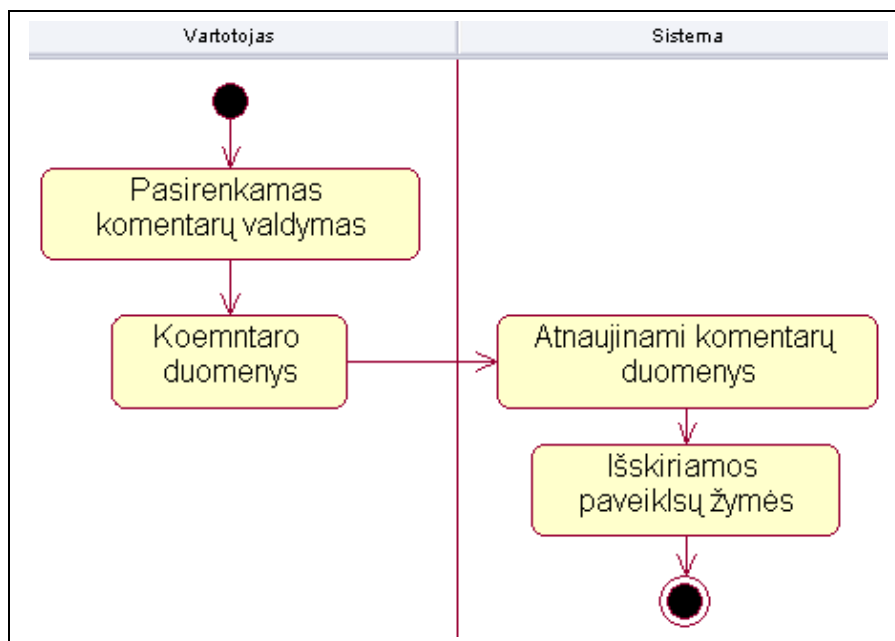
13 Paveikslas. Vartotojo prisijungimas



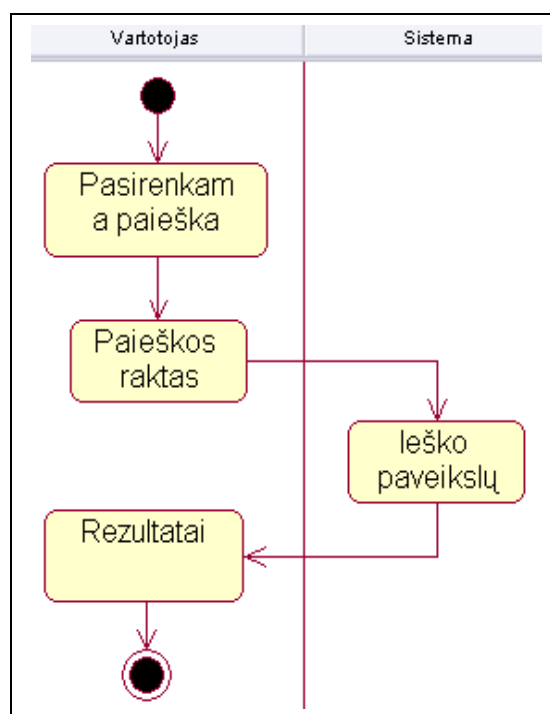
14 Paveikslas. Vartotojo informacijos keitimas



15 Paveikslas. Valdyti paveikslus



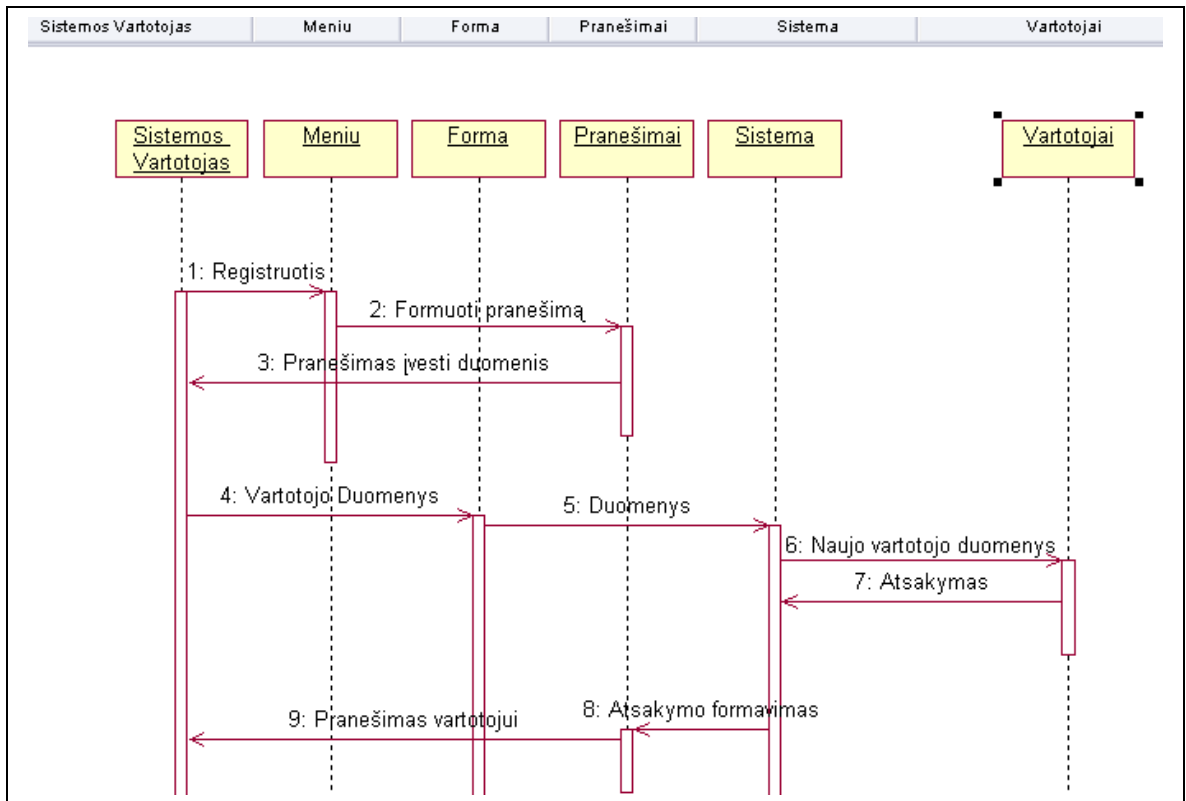
16 Paveikslas. Valdyti komentarus



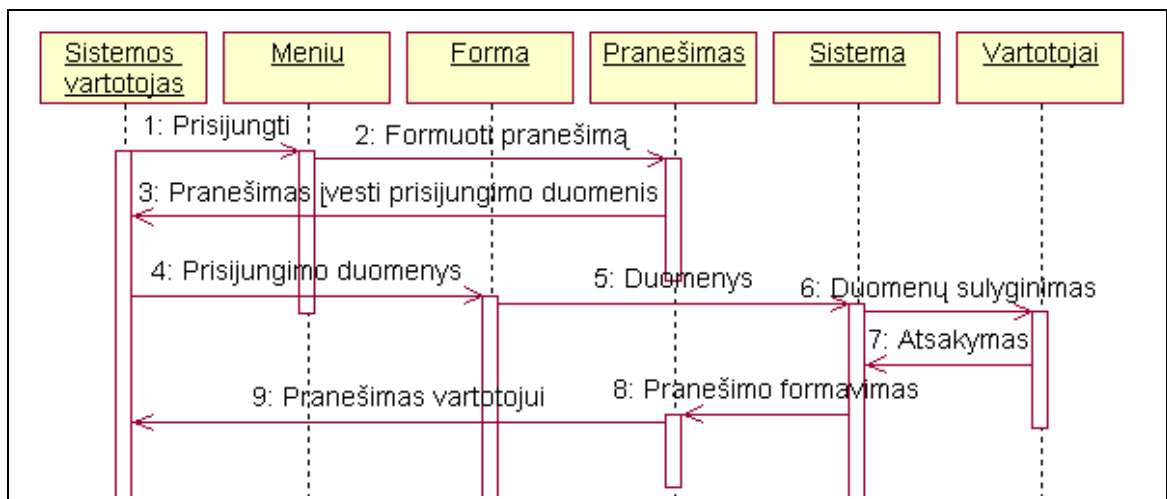
17 Paveikslas. Paieška

1.24.2. Sekų (bendradarbiavimo) diagramos

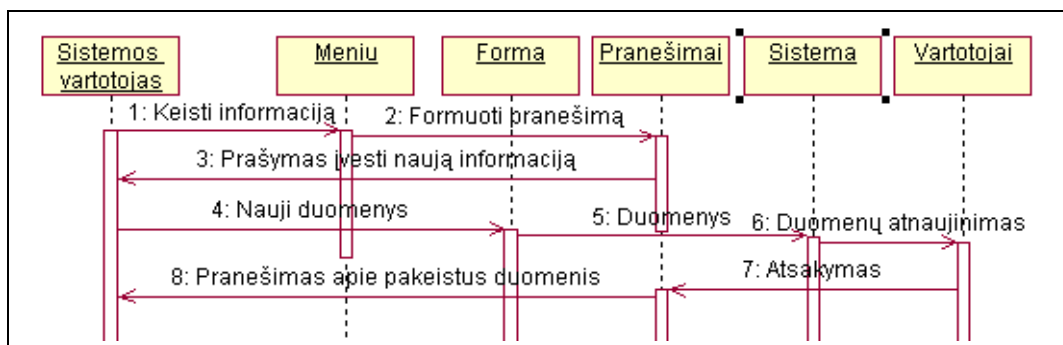
Sekų arba bendradarbiavimo diagramos vaizduoja atliekamų vartotojo bei sistemos veiksmų seką.



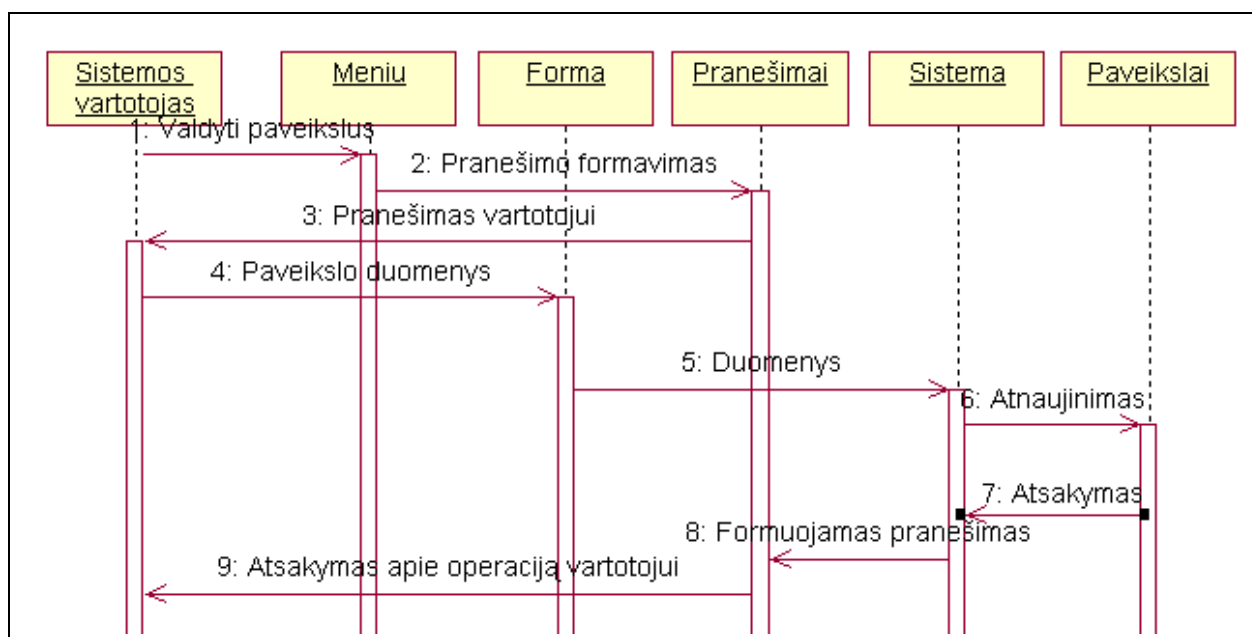
18 Paveikslas. Registracija sekų diagrama



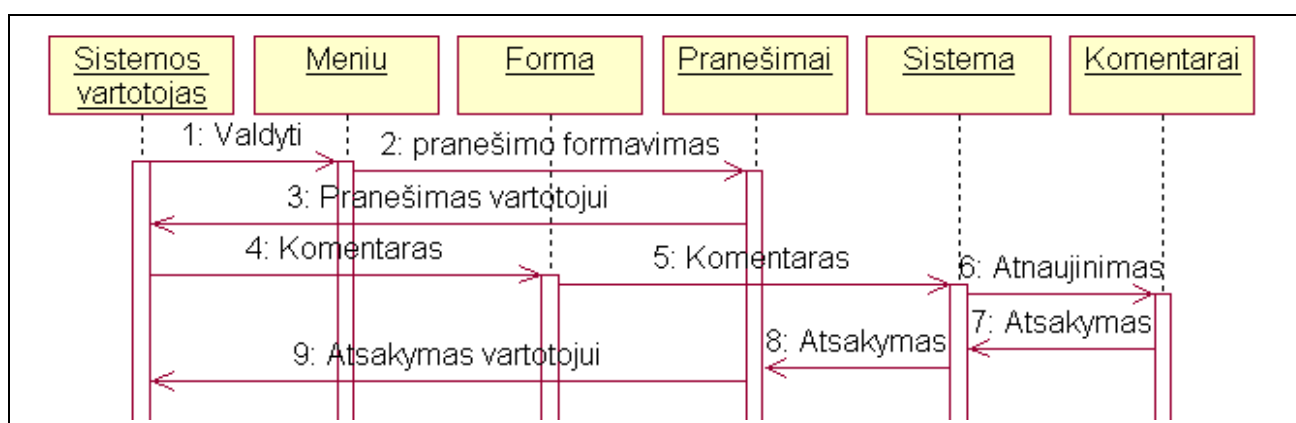
19 Paveikslas. Prisijungimas sekų diagrama



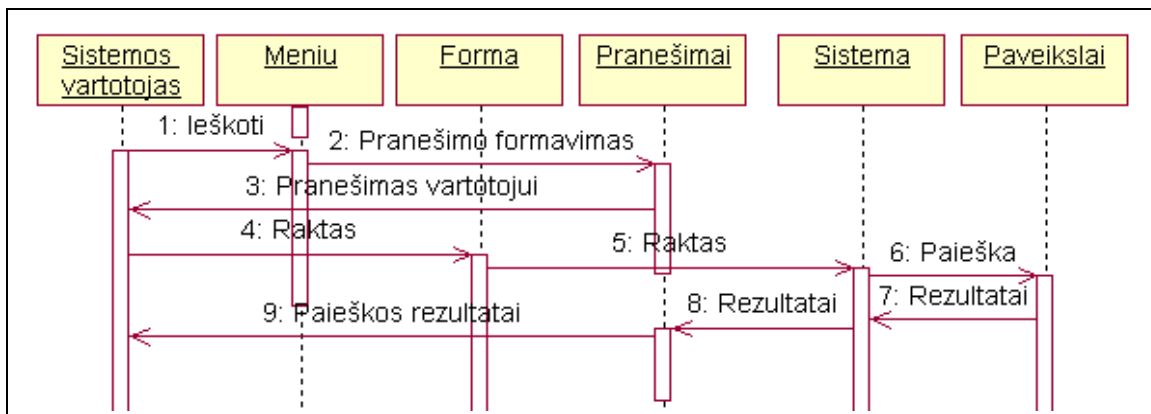
20 Paveikslas. Vartotojo informacijos keitimas sekų diagrama



21 Paveikslas. Valdyti paveikslus sekų diagrama



22 Paveikslas. Valdyti komentarus sekų diagrama



23 Paveikslas. Paieška sekų diagrama

1.25. Išdėstymo vaizdas

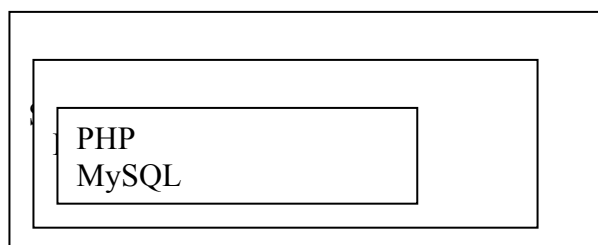
Sistemą sudaro serveris.

Serveriui keliami techninės įrangos reikalavimai:

- Operatyviosios atminties talpa 256 MB (su 256-512 MB virtualiosios atminties);
- Procesoriaus greitis 1 GHz;
- 1GB kietojo disko vietos programos įdiegimui bei papildoma vieta projektams saugoti/eksportuoti.

Reikalinga programinė įranga:

- Linux OS
- PHP
- MySQL

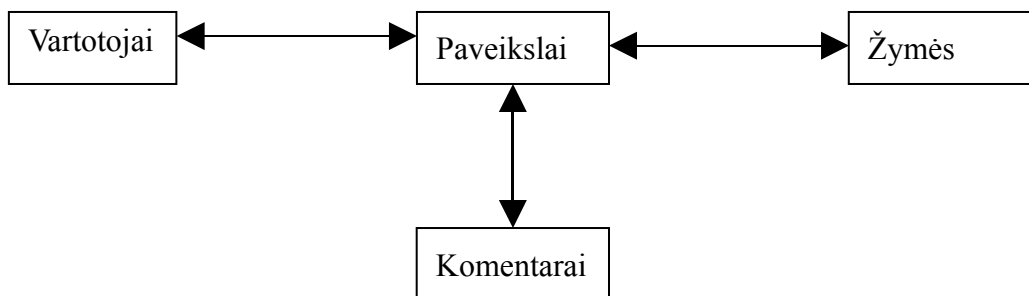


24 Paveikslas. Išdėstymo vaizdas

1.26. Duomenų vaizdas

Duomenų bazės esybės:

- Vartotojai
- Paveikslai
- Komentarai
- Žymės



25 Paveikslas. Duomenų vaizdas

1.27. Kokybė

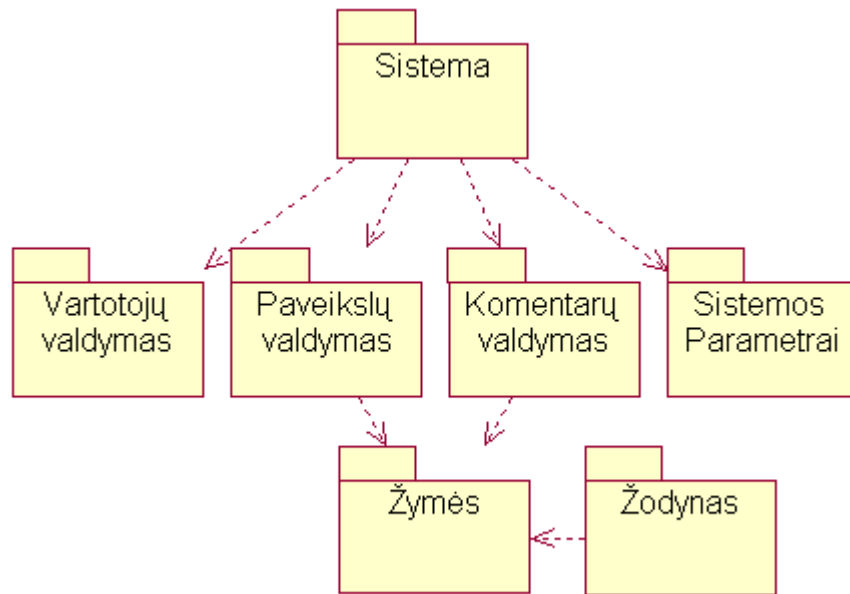
Pasirinkta sistemos architektūra leis nesunkiai praplėsti sistemą naujomis funkcijomis, nekeičiant visos sistemos architektūros bei esamų ryšių tarp paketų/klasijų. Funkcinis išskaidymas į paketus leis atskirus programos modulius pakartotinai panaudoti kitose sistemose.

Sistemos architektūra užtikrina sistemos nepriklausomumą nuo kitų.

12. Detalios architektūros specifikacija

1.28. Sistemos architektūra

1.28.1. Sistemos vaizdas



26 Paveikslas. Sistemos vaizdas

Sistema

Klasifikacija	Paketas
Apibrėžimas	Paketas, skirtas pagrindiniam programos valdymui.
Atsakomybės	Paketas valdo bendrus programos duomenis bei objektus. Šio paketo klasės valdo kitų paketų sąsajų kūrimą.
Apribojimai	Apribojimų paketui nėra
Struktūra	Detali paketo struktūra pateikiama skyriuje 2.2. <i>Paketų detalizavimas</i>
Sąveikavimas	Paketas sąveikauja kitais paketais: Vartotojų valdymas, Paveikslų valdymas, Komentarų valdymas, Sistemos parametrai.
Resursai	Serveris
Skaičiavimai	Paketo atliekami skaičiavimai detalizuojami aprašant jo klases.
Sąsaja	Paketo sąsaja yra pagrindinis programos langas.

Vartotojų valdymas

Klasifikacija	Paketas
Apibrėžimas	Paketas, skirtas vartotojų valdymui, jų kūrimui, redagavimui, grupavimui.
Atsakomybės	Paketas atsakingas už sistemos vartotojų valdymą, jų prisijungimus, teisių suteikimus.
Apribojimai	Apribojimų paketui nėra
Struktūra	Detali paketo struktūra pateikiama skyriuje 2.2. <i>Paketų detalizavimas</i>
Sąveikavimas	Šis paketas yra inicijuojamas Sistemos paketo bei sąveikauja su Paveikslo valdymo paketu
Resursai	Sistemos duomenų bazė
Skaičiavimai	Paketo atliekami skaičiavimai detalizuojami aprašant jo klases.

Sąsaja Paketo sąsaja – „Vartotojų valdymo langas“.

Paveikslų valdymas

Klasifikacija Paketas
Apibrėžimas Skirtas vartotojo pateikiamiems paveikslams apdoroti bei įkelti į sistemą.
Atsakomybės Paketas yra atsakingas už paveikslų valdymą: konvertavimą, tikrinimą, ar galima įkelti, bei pati įkėlimą.
Apribojimai Įkeliami paveiksai gali būti tik sistemoje numatyto formato.
Struktūra Detali paketo struktūra pateikiama skyriuje 2.2. *Paketų detalizavimas*
Sąveikavimas Paketas sąveikauja su paketu „Sistema“ ir „Žymių valdymas“
Resursai Darbui naudojami vartotojo pateikti paveiksai.
Paketo atliekami skaičiavimai detalizuojami aprašant jo klases.
Skaičiavimai
Sąsaja Paketo sąsaja – „Paveikslų valdymo langas“.

Komentarų valdymas

Klasifikacija Paketas
Apibrėžimas Paketas, skirtas paveikslų komentarų valdymui.
Atsakomybės Pakete valdomi paveikslų komentarai: kuriami ir redaguojami komentarai, išskiriamos žymės.
Apribojimai Komentuoti galima kai sistemoje yra paveikslų.
Struktūra Detali paketo struktūra pateikiama skyriuje 2.2. *Paketų detalizavimas*
Sąveikavimas Paketas sąveikauja su paketu „Sistema“ ir „Žymių valdymas“
Resursai Sistemoje esantys paveiksai
Skaičiavimai Paketo atliekami skaičiavimai detalizuojami aprašant jo klases.
Sąsaja Paketo sąsaja – „Komentarų valdymo langas“.

Sistemos parametrai

Klasifikacija Paketas
Apibrėžimas Paketas, skirtas sistemos parametrams nustatyti ir juos keisti.
Atsakomybės Keičia sistemos parametrus.
Apribojimai Apribojimų nėra.
Struktūra Detali paketo struktūra pateikiama skyriuje 2.2. *Paketų detalizavimas*
Sąveikavimas Paketas sąveikauja su paketu „Sistema“.
Resursai Sistemos nustatymų failas.
Skaičiavimai Paketo atliekami skaičiavimai detalizuojami aprašant jo klases.
Sąsaja Paketo sąsaja – „Sistemos parametrų langas“.

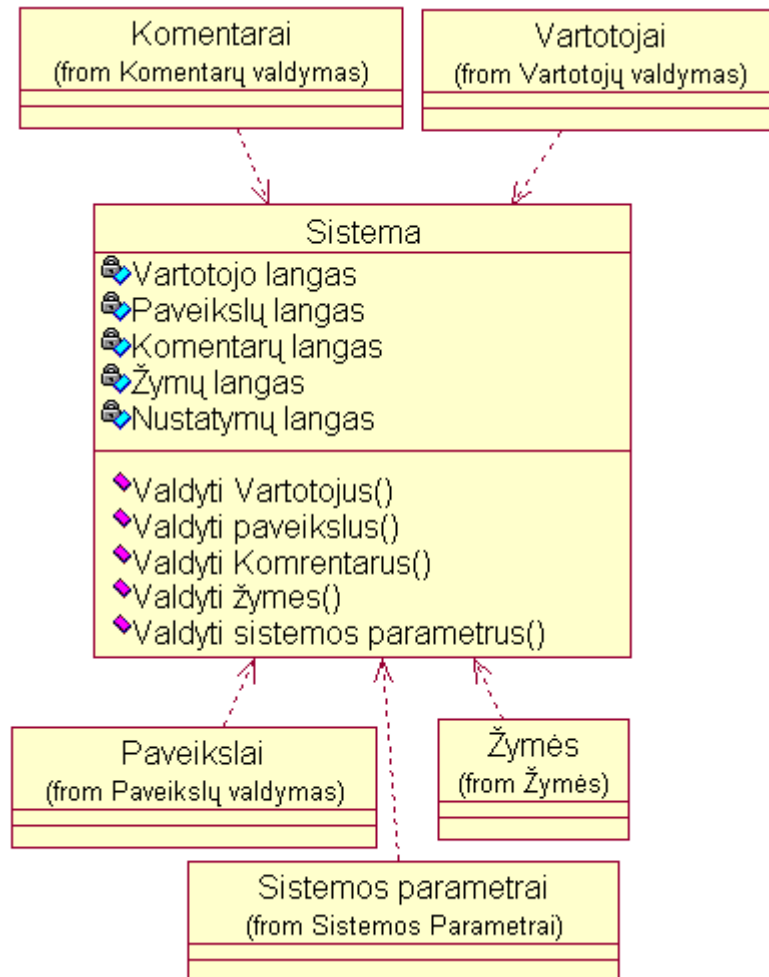
Žymių valdymas

Klasifikacija Paketas
Apibrėžimas Paketas, skirtas valdyti paveikslų žymes.
Atsakomybės Paketas atsakingas už automatinį žymių išskyrimą iš paveikslų

Apribojimai	aprašymų ir jų komentarų. Norint sugeneruoti žymes, paveikslai turi būti aprašyti ir/arba turėti komentarų.
Struktūra	Detali paketo struktūra pateikiama skyriuje 2.2. <i>Paketų detalizavimas</i>
Sąveikavimas	Muzikos eksporto paketas sąveikauja su paketais „Sistema“, „paveikslų valdymas“, „komentarų valdymas“, „žodynas“
Resursai	Paveikslai, komentarai
Skaičiavimai	Paketo atliekami skaičiavimai detalizuojami aprašant jo klases.
Sąsaja	Paketo sąsaja – „Žymų langas“
Žodynas	
Klasifikacija	Paketas
Apibrėžimas	Paketas, skirtas žymų vertimui į kitas kalbas
Atsakomybės	Paketas verčia pateiktas žymas.
Apribojimai	Galima išversti tik pavienes žymas ir tik į tas kalbas, kurios yra įdiegtos į sistemą.
Struktūra	Detali paketo struktūra pateikiama skyriuje 2.2. <i>Paketų detalizavimas</i>
Sąveikavimas	Paketas sąveikauja su paketu „Žymės“.
Resursai	Žodynai.
Skaičiavimai	Paketo atliekami skaičiavimai detalizuojami aprašant jo klases.
Sąsaja	Paketo sąsaja – „Žodyno langas“

1.28.2.Sistemos paketų detalizavimas

Sistema

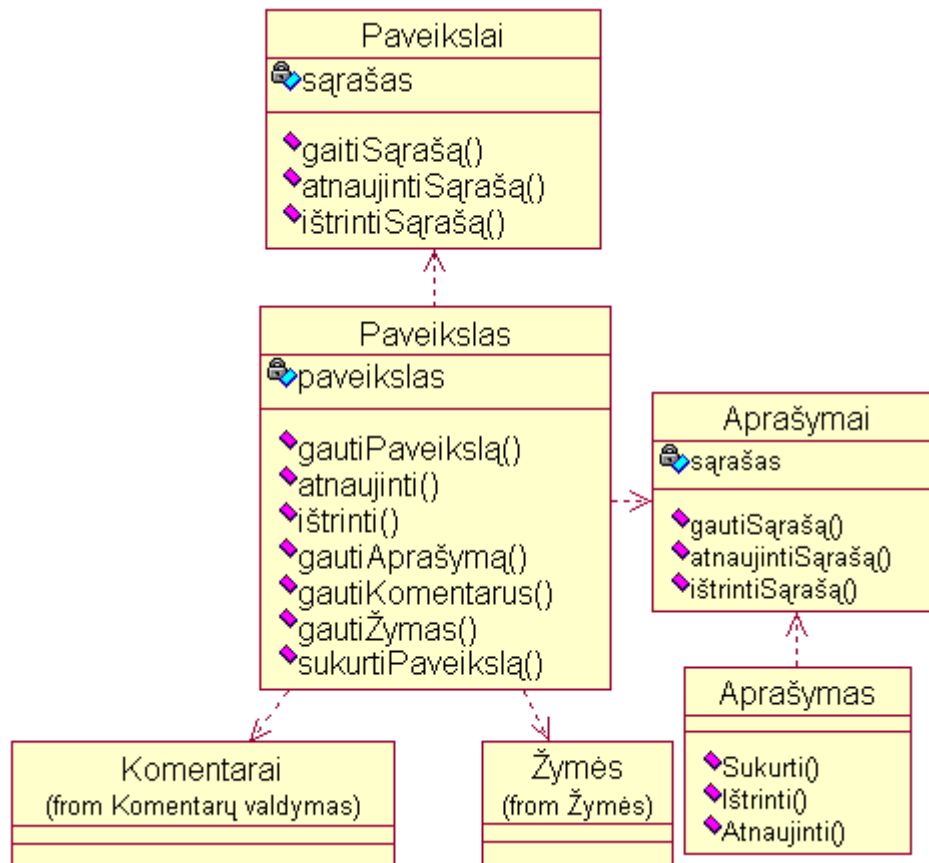


27 Paveikslas. „Sistema“ paketo struktūra

Pavadinimas	Sistema
Klasifikacija	Klasė
Apibrėžimas	Pagrindinė sistemos sąsaja
Atsakomybės	Skirta valdyti pagrindines vartotojo galimybes, dirbant su sistema
Apribojimai	Apribojimų nėra
Struktūra	Sudaro metodus, skirti apdoroti vartotojo veiksams: valdytiPaveikslus() valdytiKomentaru() valdytiŽymes() valdytiVartotojus() valdytiSistemosParametrus()
Sąveikavimas	Sukuria pagrindinio valdymo objektą, kuris apdoroja vartotojo atliekamus

Resursai	veiksmus. Gražina (vaizduoja) programos darbo rezultatus vartotojui.
Skaičiavimai	Nėra
Sąsaja/eksportas	Pati klasė skaičiavimų neatlieka. Ji kuria valdymų klasių objektus. Programos sąsaja – langas, kuriame yra vaizduojamos vartotojui galimos funkcijos.

Paveikslų valdymas



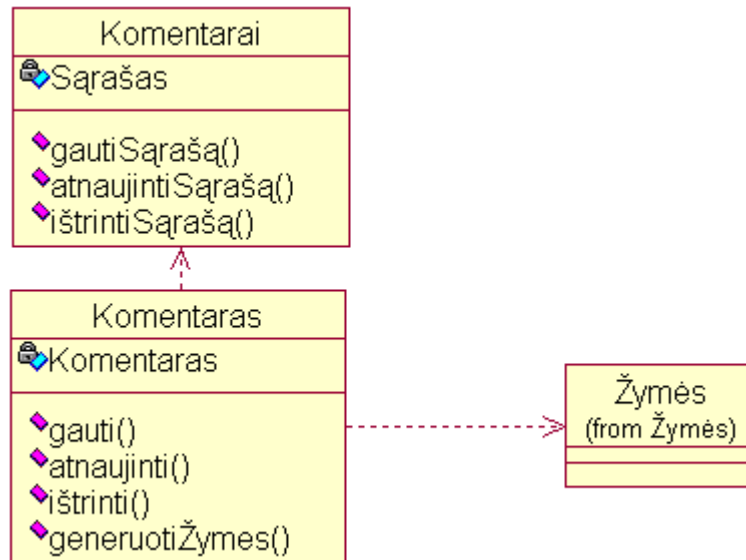
28 Paveikslas. „Paveikslų valdymo“ paketo struktūra

Pavadinimas	Paveikslas
Klasifikacija	Klasė
Apibrėžimas	Vieno paveikslų valdymo klasė
Atsakomybės	Valdyti vieną paveikslą
Apribojimai	Nėra.
Struktūra	Sudaro metodus, skirti apdoroti paveikslui: sukurtiPaveikslą() gautiPaveikslą() atnaujinti() ištrinti() gautiAprašymus() gautiKomentarus()

Sąveikavimas	gautiŽymas() Pagrindinė klasė kuri valdo vieną paveikslą, sąveikauja su Paveikslų,
Resursai	aprašų, Komentarų ir Žymų klasėmis
Skaičiavimai	Pateiktas paveikslas, DB
Sąsaja/eksportas	Pati klasė kuria ir padaroja pateiktą paveikslą Sąsaja – langas, kuriame apdorojamas vienas paveikslas
Pavadinimas	Paveikslai
Klasifikacija	Klasė
Apibrėžimas	Klasė, valdo paveikslų sąrašus.
Atsakomybės	Skirta apdoroti vartotojo atliekamus veiksmus su paveikslų sąrašais
Apribojimai	Norint apdoroti paveikslų sąrašus, reikia kad jie egzistuotų sistemoje.
Struktūra	gautiPaveikslus() atnaujintiPaveikslus() ištrintiPaveikslus()
Sąveikavimas	Naudoja paveikslo klasės objektus sąrašams formuoti.
Resursai	DB.
Skaičiavimai	Sukuria, ištrina arba atnaujina paveikslų sąrašus
Sąsaja/eksportas	Sąsaja – langas, kuriame apdorojami paveikslų sąrašai.
Pavadinimas	Aprašymas
Klasifikacija	Klasė
Apibrėžimas	Vieno paveikslo aprašymo valdymo klasė
Atsakomybės	Valdyti vieną paveikslo aprašymą
Apribojimai	Nėra.
Struktūra	Sudaro metodai, skirti apdoroti aprašymui: sukurti () atnaujinti() ištrinti()
Sąveikavimas	Pagrindinė klasė kuri valdo vieną paveikslo aprašą, sąveikauja su Paveikslų aprašų klase.
Resursai	DB
Skaičiavimai	Kuriamas ir apdorojamas aprašas
Sąsaja/eksportas	Sąsaja – langas, kuriame apdorojamas vienas paveikslo aprašas
Pavadinimas	Aprašymai
Klasifikacija	Klasė
Apibrėžimas	Klasė, valdo paveikslų aprašymus įvairiomis kalbomis.
Atsakomybės	Skirta apdoroti paveikslo aprašymus
Apribojimai	Aprašytas gali būti tik esantis sutemoje paveikslas.
Struktūra	gautiSąrašą() atnaujintiSąrašą() ištrintiSąrašą()
Sąveikavimas	Naudoja aprašo klasės objektus sąrašams formuoti.

Resursai	DB.
Skaičiavimai	Sukuria, ištrina arba atnaujina paveikslų aprašymus
Sąsaja/eksportas	Sąsaja – langas, kuriame apdorojami paveikslų aprašymo sąrašai.

Komentarų valdymas



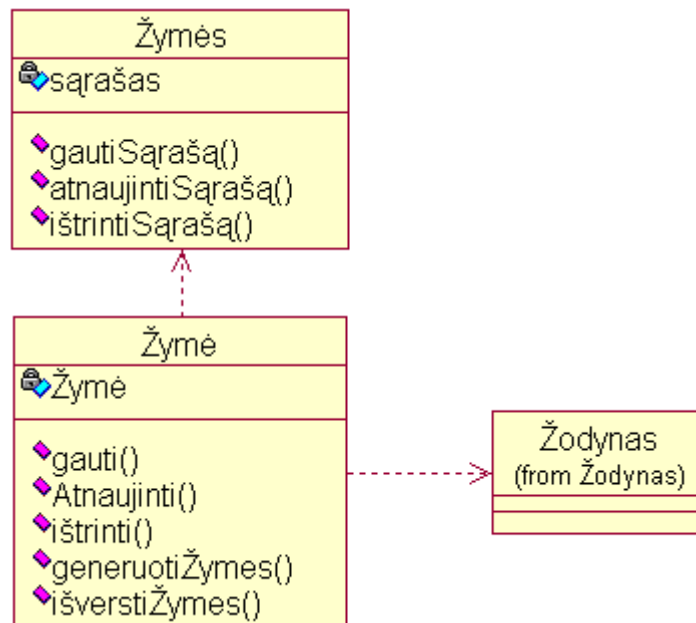
29 Paveikslas. „Komentarų valdymo“ paketo struktūra

Pavadinimas	Komentaras
Klasifikacija	Klasė
Apibrėžimas	Vieno paveikslų komentaro valdymo klasė
Atsakomybės	Valdyti vieną paveikslų komentarą
Apribojimai	Komentuojamas gali būti tik esantis sistemoje paveikslas.
Struktūra	Sudaro metodus, skirti apdoroti komentarui: sukurti () atnaujinti() ištrinti()
Sąveikavimas	Pagrindinė klasė kuri valdo vieną paveikslų aprašą, sąveikauja su Žymų ir Komentarų klasėmis.
Resursai	DB
Skaičiavimai	Kuriamas ir apdorojamas komentaras.
Sąsaja/eksportas	Sąsaja – langas, kuriame apdorojamas vienas paveikslų komentaras.

Pavadinimas	Komentarai
Klasifikacija	Klasė
Apibrėžimas	Klasė, valdo paveikslų komentarus įvairiomis kalbomis.
Atsakomybės	Skirta apdoroti paveikslų komentarus
Apribojimai	Komentuojamas gali būti tik esantis sistemoje paveikslas.
Struktūra	gautiSąrašą() atnaujintiSąrašą() ištrintiSąrašą()

Sąveikavimas	Naudoja Komentarų klasės objektus sąrašams formuoti.
Resursai	DB.
Skaičiavimai	Sukuria, ištrina arba atnaujina paveikslų komentarus
Sąsaja/eksportas	Sąsaja – langas, kuriame apdorojami paveikslų komentarų sąrašai.

Žymų valdymas

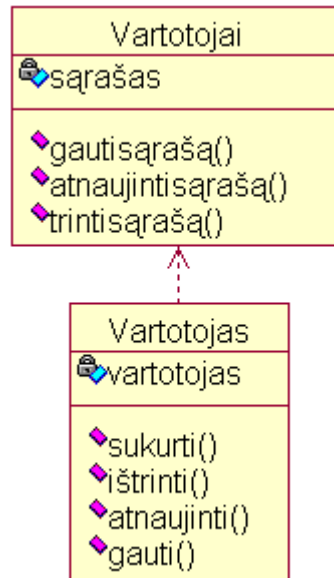


30 Paveikslas. „Žymų valdymo“ paketo struktūra

Pavadinimas	Žymė
Klasifikacija	Klasė
Apibrėžimas	Vieno paveikslų žymės valdymo klasė
Atsakomybės	Valdyti vieną paveikslų žymę
Apribojimai	Žymė gali būti susieta tik su esančiu paveikslu.
Struktūra	Sudaro metodus, skirti apdoroti žyme: gauti () atnaujinti() ištrinti() generuotiŽymes() išverstiŽymes()
Sąveikavimas	Pagrindinė klasė kuri valdo vieną paveikslų aprašą, sąveikauja su Žymų ir Žodynų klasėmis.
Resursai	DB
Skaičiavimai	Kuriama ir apdorojama žymė.
Sąsaja/eksportas	Sąsaja – langas, kuriame apdorojama viena paveikslų žymė.
Pavadinimas	Žymės
Klasifikacija	Klasė

Apibrėžimas	Klasė, valdo paveikslų žymes įvairiomis kalbomis.
Atsakomybės	Skirta apdoroti paveikslo žymes
Apribojimai	Žymė gali būti susieta tik su esančiu paveikslu.
Struktūra	gautiSąrašą() atnaujintiSąrašą() ištrintiSąrašą()
Sąveikavimas	Naudoja Žymės klasės objektus sąrašams formuoti.
Resursai	DB.
Skaičiavimai	Sukuria, ištrina arba atnaujina paveikslo žymes
Sąsaja/eksportas	Sąsaja – langas, kuriame apdorojami paveikslų žymų sąrašai.

Vartotojų valdymas

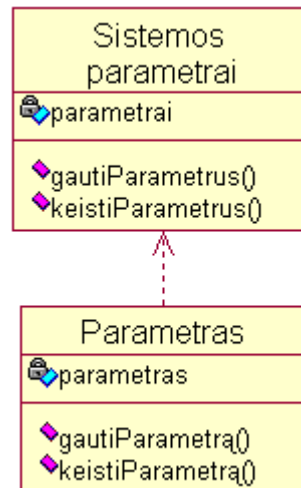


31 Paveikslas. „Vartotojų valdymas“ paketo struktūra

Pavadinimas	Vartotojas
Klasifikacija	Klasė
Apibrėžimas	Vieno sistemos vartotojo klasė
Atsakomybės	Valdyti vieną sistemos vartotoją
Apribojimai	Negali būti sukurtas egzistuojantis vartotojas.
Struktūra	Sudaro metodus, skirti apdoroti vartotojui: gauti () atnaujinti() ištrinti() sukurti()
Sąveikavimas	Pagrindinė klasė kuri valdo vieną sistemos vartotoją, sąveikauja su Vartotojų klase
Resursai	DB
Skaičiavimai	Kuriamas ir valdomas vartotojas.
Sąsaja/eksportas	Sąsaja – langas, kuriame valdomas vienas vartotojas.
Pavadinimas	Vartotojai
Klasifikacija	Klasė
Apibrėžimas	Klasė, valdo vartotojų sąrašus.
Atsakomybės	Skirta valdyti sistemos vartotojų sąrašams
Apribojimai	nėra
Struktūra	gautiSąrašą() atnaujintiSąrašą() ištrintiSąrašą()
Sąveikavimas	Naudoja Vartotojo klasės objektus sąrašams formuoti.

Resursai	DB.
Skaičiavimai	Sukuria, ištrina arba atnaujina sistemos vartotojus
Sąsaja/eksportas	Sąsaja – langas, kuriame apdorojami sistemos vartotojų sąrašai.

Sistemos Parametrai

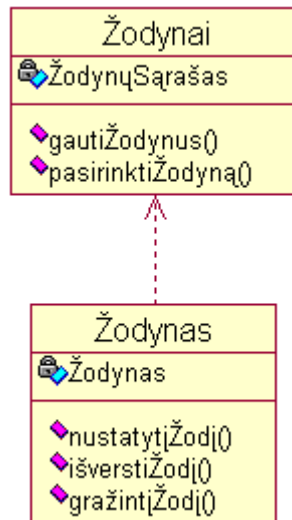


32 Paveikslas. „Sistemos parametrai“ paketo struktūra

Pavadinimas	Parametras
Klasifikacija	Klasė
Apibrėžimas	Sistemos parametro klasė
Atsakomybės	Valdyti vieną sistemos parametą
Apribojimai	nėra
Struktūra	Sudaro metodus, skirti apdoroti parametrai: gauti Parametrą() keistiParametrą()
Sąveikavimas	Pagrindinė klasė kuri valdo vieną sistemos parametą, sąveikauja su sistemos parametų klase
Resursai	Parametų failas
Skaičiavimai	Gaunami ir keičiami sistemos parametrai
Sąsaja/eksportas	nėra
Pavadinimas	Parametrai
Klasifikacija	Klasė
Apibrėžimas	Klasė, valdo sistemos parametrus.
Atsakomybės	Skirta valdyti sistemos parametrus
Apribojimai	nėra
Struktūra	gauti Parametrus() keistiParametrus()
Sąveikavimas	Naudoja Parametro klasės objektus sąrašams formuoti.

Resursai	DB.
Skaičiavimai	Sukuria, ištrina arba atnaujina sistemos parametrus
Sąsaja/eksportas	Sąsaja – langas, kuriame apdorojami sistemos parametrai.

Žodynai



33 Paveikslas. „Aplinkos atributų keitimo“ paketo struktūra

Pavadinimas	Žodynas
Klasifikacija	Klasė
Apibrėžimas	Vieno žodyno klasė
Atsakomybės	Valdyti vieną žodyną
Apribojimai	
Struktūra	Sudaro metodai, skirti versti žodžiams:

nustatytiŽodį ()

išverstiŽodį()

gražintiŽodį()

Sąveikavimas	Sąveikauja su žodynų klase
Resursai	DB
Skaičiavimai	Išverčiami žodžiai į kitą kalbą.
Sąsaja/eksportas	nėra

Pavadinimas	Žodynai
Klasifikacija	Klasė
Apibrėžimas	Klasė, valdo žodynus.
Atsakomybės	Skirta valdyti žodynams, gauti egzistuojančius žodynus, pasirinkti žodyną
Apribojimai	nėra
Struktūra	gautiŽodynus() pasirinktiŽodyną()
Sąveikavimas	Naudoja Žodyno klasės objektus sąrašams formuoti.
Resursai	DB.

Skaičiavimai Pasirenka pageidaujamus žodynus
Sąsaja/eksportas nėra

13. Testavimo medžiaga

Šis dokumentas – paveikslų aprašymo atviro kodo programinės įrangos testavimo specifikacija. Dokumente pateikiama programinės įrangos testavimo apžvalga, testavimo planas bei procedūra.

1.29. Testavimo tikslai

Testavimo tikslas – patikrinti bei užtikrinti, kad – paveikslų aprašymo atviro kodo programinė įranga atitinka vartotojo reikalavimus bei sistemos specifikaciją pagal ankstesniuose dokumentuose nustatytus kriterijus. Testavimas yra taip pat reikalingas sistemos tobulinimui bei defektų taisymui, jei yra randamos sistemos klaidos.

Testavimu siekiama užtikrinti, kad vartotojas gaus produktą su minimaliu klaidų kiekiu, pilnai paruoštą vartotojo darbui.

1.30. Testavimo apimtis

Planuojama sistemos testavimo apimtis:

- Sistemos dalių testavimas
 - Sistemos vienetų testavimas
 - Sistemos apjungimo (integracijos) testavimas
 - Sistemos sąsajos testavimas
- Sistemos priėmimo testavimas (pas užsakovą)
- Sistemos aukšto lygio testavimas (pas užsakovą).

1.31. Testavimo planas

Testavimo planas apibrėžia sistemos testavimo strategiją, nenurodant sistemos testavimo smulkmenų. Šiame skyriuje aprašomas testuojamas objektas, jo moduliai, būdai, kuriais testuojama. Taip pat aprašomi kiti su testavimu susiję klausimai (testavimo resursai, grafikas ir kt.).

Testavimas susideda iš kelių etapų:

- Bendras sistemos testavimas
 - Vienetų testavimas
 - Sistemos integracijos testavimas
 - Sistemos sąsajos testavimas
- Priėmimo testavimas
- Aukšto lygio testavimas

Sistemos dalių testavimas – programinės įrangos dalių, susietumo testavimas, baigus programuoti sistemos dalis bei visą sistemą. Jis atliekamas prieš bandomąjį testavimą pas užsakovą. Pagrindiniai bendrojo testavimo uždaviniai yra patikrinti/užtikrinti sistemos bei jos dalių tinkamumą naudoti bei toliau testuoti.

Sistemos dalių testavimas baigiasi sistemos priėmimo testavimu, dalyvaujant užsakovui. Tai – visos sistemos testai.

Priėmimo testavimas turi užtikrinti, kad sistema atitinka vartotojo poreikius, aprašytus sistemos reikalavimų specifikacijoje. Orientuojamasi į funkcinis bei nefunkcinis sistemos reikalavimus.

Aukšto lygio testavimas yra testavimas pas užsakovą, įdiegiant į užsakovo kompiuterius suprogramuotą sistemą. Čia yra tikrinami tokie nefunkciniai reikalavimai, kaip naudojamumas, greitas vartotojų apsimokymas ir kt.

1.32. Testuojama programinė įranga

Testuojama programinė įranga – paveikslų aprašymo atviro kodo programinė įranga. Tikslas: padaryti muziejines vertybes prieinamas plačiajai visuomenei. Analogas, www.steve.museum. Reikalavimai: Daugiakalbystė - turi būti galimybė aprašyti paveikslus bet kuria Europos kalba; Vartotojo tyrimai - turi būti galimybė analizuoti kokie vartotojai jungiasi, skirstyti juos į kategorijas. Galimybė peržiūrėti paveikslus arba ieškoti pagal meta duomenis. Priskirti žymes paveikslams. Peržiūrėti, redaguoti, ieškoti pagal žymes. Komentuoti paveikslus, redaguoti kitų komentarus. Automatiškai generuoti žymes iš vartotojų komentarų.

1.33. Testavimo strategija

Toliau yra aprašomos vienetų, sistemos integracijos, priėmimo bei aukšto lygio testavimo strategijos.

1.33.1.Vienetų testavimas

Atskirų vienetų (klasių ar modulių) testavimui bus naudojami struktūrinis ir funkcinis testavimo metodai.

Struktūrinio testavimo metu yra atsižvelgiama į algoritmo struktūrą („balta dėžė“) ir tikrinamas metodo veikimas kuomet kiekvienas funkcijos sakiny yra įvykdomas bent vieną kartą.

Funkcinio testavimo metu tikrinama ar pagal metodui perduotus pradinius duomenys gaunami laukiami rezultatai. Duomenys parenkami neatsižvelgiant į metodo struktūrą („juoda dėžė“). Testiniai atvejai sudaromi pagal sistemos specifikaciją.

Testuojami vienetai:

Vartotojų modulis – sistemos vartotojų kūrimo ir valdymo modulis.

Kategorijų modulis – paveikslų kategorijų kūrimo ir valdymo modulis.

Paveikslų modulis – paveikslų valdymo modulis.

Komentarų modulis – paveikslų komentarų modulis.

Meta duomenų modulis – paveikslo meta duomenų modulis.

1.33.2.Integravimo testavimas

Atlikus vienetų testavimą, vienetai yra apjungiami dalimis ir testuojama jų sąveika. Naudojamas „bottom-up“ integracinis testavimas. Ištestuoti žemiausio lygio komponentai (funkcijos, klasės) apjungiami į posistemas (modulius). Vėliau sąveikaujantys moduliai apjungiami, apjungimą užbaigiant „Pagrindinio valdymo“ moduliu bei testuojant vartotojo sąsajos veikimą („Pagrindinio valdymo“ modulis atsakingas už teisingą sąsajų veikimą).

1.33.3.Priėmimo testavimas

Priėmimo testavimas atliekamas pagal „juodos dėžės“ principą. Sistema bus nuodugnai ištestuota pagal reikalavimų specifikaciją ir patikrinta, ar sistema atitinka vartotojo poreikius. Radus neatitikimą tarp sistemos veikimo ir vartotojo poreikių, bus tikrinama reikalavimų specifikacija. Jei specifikacija atitiks vartotojo poreikius, reiškia sistema neatitinka specifikacijos ir registruojama klaida. Jei priešingu atveju, registruojamas vartotojo reikalavimas, tačiau jis neįgyvendinamas, paliekant galimybę tai padaryti ateityje.

1.33.4. Aukšto lygio testavimas

Užbaigto produkto testavimas, siekiant išsiaiškinti produkto kokybę bei parengti produkto kokybės dokumentą.

1.34. Testavimo resursai

Vienetų bei sistemos integracijos testavimui naudojamas kompiuteris su tokiomis charakteristikomis:

- Operatyviosios atminties talpa 1024 MB
- Procesoriaus greitis 1.75 MHz;
- 1280x800 ekrano raiška;
- 1000 MB laisvos kietojo disko vietos.

Programinė kompiuterio įranga:

Windows XP su SP2

Testavimo komanda:

Testavimo vadovas – atsakingas už testavimo planus ir sėkmingą jų vykdymą.

Testuotojai – atsakingi už testų vykdymą pagal sudarytą planą.

Sistemos dalių testavimo – profesionalūs testuotojai, tikrinantys ir fiksuojantys testo plane sistemos dalių testavimui numatytas situacijas.

Priėmimo bei aukšto lygio testavimo – neprofesionalūs vartotojai, kurie testuoja sistemą iš vartotojo perspektyvos.

Produkto vadovas – užtikrina, kad testai yra įvykdyti sėkmingai žiūrint iš vartotojo perspektyvos.

1.35. Testavimo procedūra

„Testavimo procedūroje“ detaliam aprašomi sistemos komponentų integracijos, priėmimo ir aukšto lygio testai. Detalizuojamas sistemos testavimui reikalingų resursų paskirstymas bei aprašoma rezultatų saugojimo struktūra.

1.35.1. Vienetų testavimas

Vienetų testavimas aprašo modulių, klasių bei jas sudarančių funkcijų testavimo veiksmus. Dėl laiko stokos ir kai kurių vienetų trivialumo, šie vienetai yra netestuojami. Testuojami vienetai, kurie yra esminiai sistemai. Pilnas sistemos vienetų (modulių, klasių, metodų) sąrašas yra detalios architektūros specifikacijoje.

1.35.2.Integravimo testavimas

Atskirai ištestuoti komponentai palaipsniui bus integruojami į programų sistemą, kuri bus testuojama toliau. Tokiu būdu bus atliekamas stambinantysis (“bottom-up”) testavimas. Programų sistemos integravimas vyks tokia tvarka:

1. Vartotojų valdymo modulis,
2. Kategorijų valdymo modulis,
3. Paveikslų valdymo modulis,
4. Komentarų valdymo modulis.
5. Meta duomenų valdymo modulis.

Integruojant programų sistemą šiuo būdu, reikalingi vykdykliai, iškviečiantys žemesnio lygio komponentus. Vykdykliai pakeičia nerealizuotus arba neintegruotus aukštesnio lygio komponentus.

1.35.3.Vartotojo sąsajos testavimas

Vartotojo sąsajos testavimas susideda iš sistemos langų bei jų komponentų veikimo testavimo.

Sąsajos testavimas atliekamas atskirai ir kartu su integracijos testavimu, kai yra tikrinamas sistemos funkcionalumas (kartu patikrinamas ir vartotojo sąsajos veikimo teisingumas).

Su integracijos testavimu atliekamo sąsajų testavimo metu yra patikrinamas tvarkingas sąsajų rodymas:

- Nėra galimybės iškviešti negalimo tuo metu aktyvuoti
- Vartotojo pagalbos žinynas pilnas (nuorodos veikia į visus žinyno punktus).
- Įkelti galima tik leidžiamo tipo failus.

Atskiros sąsajos testavimo metu atliekama visos sąsajos peržiūra:

- Tikrinama, ar korektiški meniu, mygtukų pavadinimai, ar matosi, naudojant skirtingą ekrano raišką
- Tikrinama sistemos pateikiama vartotojui informacija – sistemos pranešimai, informacija apie failus.
- Tikrinamas daugiakalbiškumo naudojimas visoje sistemoje.
- Tikrinami sistemos šriftai – pavadinimai, dydžiai bei spalvos.
- Tikrinamas sąsajos dydžio kitimas priklausomai nuo vartotojo atliekamų veiksmų. Kintant lango dydžiui, atitinkamai keičiasi, persiskirsto komponentai.

1.35.4.Priėmimo testavimas

Sistema įdiegiama pas užsakovą. Atliekamas visos sistemos testavimas. Patikrinami sistemai keliami funkciniai ir nefunkciniai reikalavimai. Registruojami vartotojo poreikiai, nesutampantys su reikalavimų specifikacija. Jei vartotojo poreikiai nėra aprašyti reikalavimų specifikacijos dokumente, jie išsaugomi tolimesniam sistemos tobulinimui. Priešingu atveju yra registruojama sistemos klaida, kurią reikia taisyti.

1.35.5.Aukšto lygio testavimas

Sistema duodama testuoti KTU studentams. Aukšto lygio testavimo dokumente registruojamas apsimokymo dirbti su sistema laikas (50% sėkmingas pasinaudojimas sistema be apsimokymo). Išsiaiškinama ir užrašoma vartotojų nuomonė apie sistemos paprastumą (standartiškumą) bei patogų valdymą (60% vartotojų laiko sistemą patogiai vartoti).

Registruojamos nenumatytos sistemos klaidos.

14. PAVEIKSLŲ APRAŠYMO ATVIRO KODO PROGRAMINĖS ĮRANGOS TYRIMAS

15. Sistemos funkcijų tyrimas.

Šiame tyrime bus apžvelgtos analogiškos arba panašiu principu veikiančios sistemos. Ir ištirtos bei palygintos sistemų funkcijos. Tyrimui parinkti šie analogai:

- Wikipedia – internetinė enciklopedija.
- Flickr – nuotraukų dalinimosi tarnyba.
- Steve museum – analogas tiriamai sistemai, kuri kaip ir tiriamą sistemą skirta muziejinių vertybių aprašymui ir žymėjimui.

Tyrimui buvo parinkti keturi pagrindiniai kriterijai:

- Paveikslų valdymas.
- Keitimų valdymas
- Vartotojų valdymas
- Bendrinės sistemos funkcijos

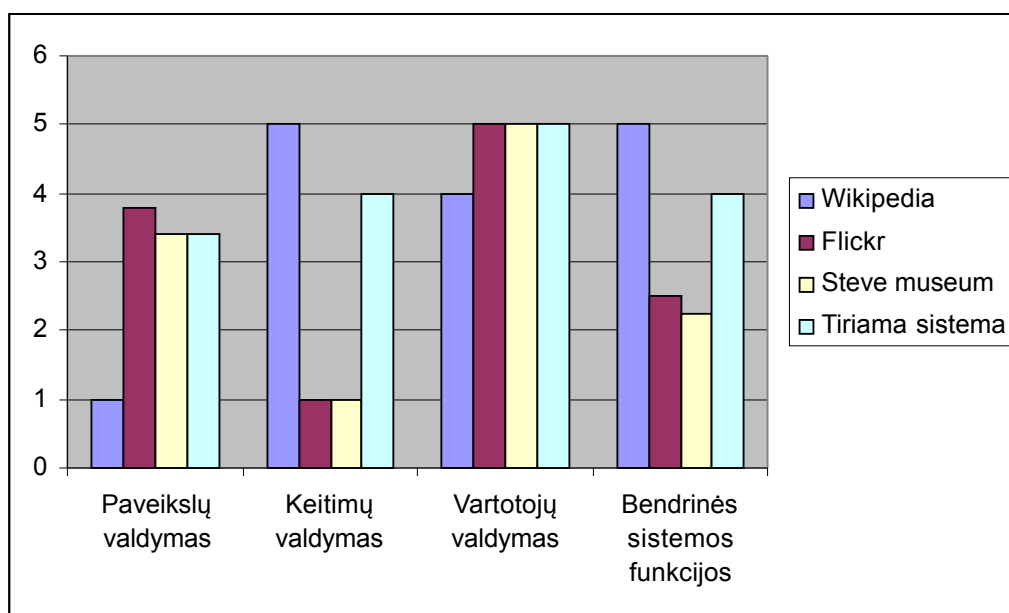
Pagrindiniai keturi kriterijai detalizuojami į smulkesnius kriterijus kurie ir bus vertinami. Pagrindinių kriterijų įvertinimas bus smulkesnių kriterijų įvertinimo aritmetinis vidurkis.

Tyrimo rezultatai pateikiami žemiau esančioje įvertinimo lentelėje, kur kiekvienas kriterijus vertinamas skalėje nuo 1 iki 5, kur 1 reiškia prastai įgyvendinta arba neįgyvendinta, o 5 – puikiai įgyvendintas funkcionalumas.

22 Lentelė. Sistemos funkcionalumo tyrimo rezultatų lentelė

	Wikipedia	Flickr	Steve museum	Tiriama sistema
Paveikslų valdymas	1	3,8	3,4	3,4
Paveikslų rinkinių įkėlimas	1	5	5	1
Automatizuotas informacijos priskyrimas	1	3	5	1
Galimybė priskirti žymes	1	5	5	5
Galimybė komentuoti	1	5	1	5
Automatizuotas žymių išrinkimas	1	1	1	5
Keitimų valdymas	5	1	1	4
Galimybė atstatyti pakeistą informaciją	5	1	1	5
Pakeitimų išryškėjimas/išskyrimas	5	1	1	3
Vartotojų valdymas	4	5	5	5
Registracija	5	5	5	5

Redagavimas	5	5	5	5
Slaptažodžio priminimas ir keitimas	5	5	5	5
Demografinių duomenų priskyrimas	1	5	5	5
Bendrinės sistemos funkcijos	5	2,5	2,25	4
Daugiakalbiškumas	5	3	1	5
Paieška	5	5	3	5
Informacijos keitimas neprisijungus	5	1	4	1
Statinių puslapių kūrimas	5	1	1	5
Bendras	3,75	3,075	2,91	4,1



34 Paveikslas. Sistemos funkcionalumo tyrimo rezultatų grafikas

Apibendrinus bendrą vidutinį įvertinimą, galima teigti, kad tiriamos sistemos funkcionalumas nėra kiek neatsilieka nuo esamų analogų. Gautas didžiausias bendras įvertinimas nerodo, kad esama sistema yra geriausia tarp visų tiriamų analogų, nes vertinimo kriterijai buvo išrinkti remiantis sistemos funkciniais reikalavimais. Reikia įvertinti tai, kad lyginti analogai greičiausiai neturėjo tokių pačių funkcinų reikalavimų, todėl analogiškos sistemos yra lygiai taip pat kokybiškos ir atitinkančios joms keltus reikalavimus.

16. Programinės įrangos kokybės tyrimas pagal ISO-9126 standartą.

ISO 9126 yra programinės įrangos kokybės tarptautinis standartas, kuris buvo sukurtas dar 1991 metais ir per dešimtmetį dar patobulintas, tam kad nustatytų karkasą programinės įrangos kokybės įvertinimui.

Šis standartas apibrėžia šešias charakteristikas, kurios smulkinamos į keletą sub. charakteristikų:

Funkcionalumas – ar visos reikalaujamos funkcijos yra programinėje įrangoje.

- Tinkamumas – ar programinė įranga vykdo reikiamas užduotis.
- Tikslumas – ar rezultatai yra tokie kurių tikėtasi.
- Funkcinis suderinamumas – ar programinė įranga gali sąveikauti su kitomis sistemomis.
- Saugumas – Ar sistema yra apsaugota nuo neleistinių vartotojų.

Patikimumas – kokio patikimumo yra programinė įranga.

- Užbaigtumas – ar dauguma klaidų yra ištaisytos.
- Klaidų toleravimas – ar programinė įranga sugeba valdyti klaidas.
- Atstatomumas – ar programinė įranga gali tęsti darbą bei atstatyti prarastus duomenis įvykus gedimui.

Naudojamumas – ar programinė įranga yra lengvai naudojama.

- Suprantamumas – ar vartotojai lengvai supranta kaip naudotis sistema.
- Išmokstamumas – ar vartotojai greitai išmoksta naudotis sistema.
- Patrauklumas – ar sistemos vartotojo sąsaja yra gera ir patraukli.

Efektyvumas – kokio efektyvumo yra programinė įranga.

- Laiko savybės – kaip greitai sistema atsako.
- Resursų naudojimas – kaip efektyviai sistema naudoja resursus.

Palaikomumas – kaip lengvai programinė įranga keičiama ir tobulinama.

- Analizuojamumas – ar klaidos lengvai diagnozuojamos.
- Keičiamumas – ar sistema lengvai keičiama.
- Stabilumas – ar sistema gali toliau veikti įvykužius pakeitimus.
- Testuojamumas – ar programinė įranga yra lengvai testuojama.

Portatyvumas – kaip lengvai galima perkelti į kitą aplinką.

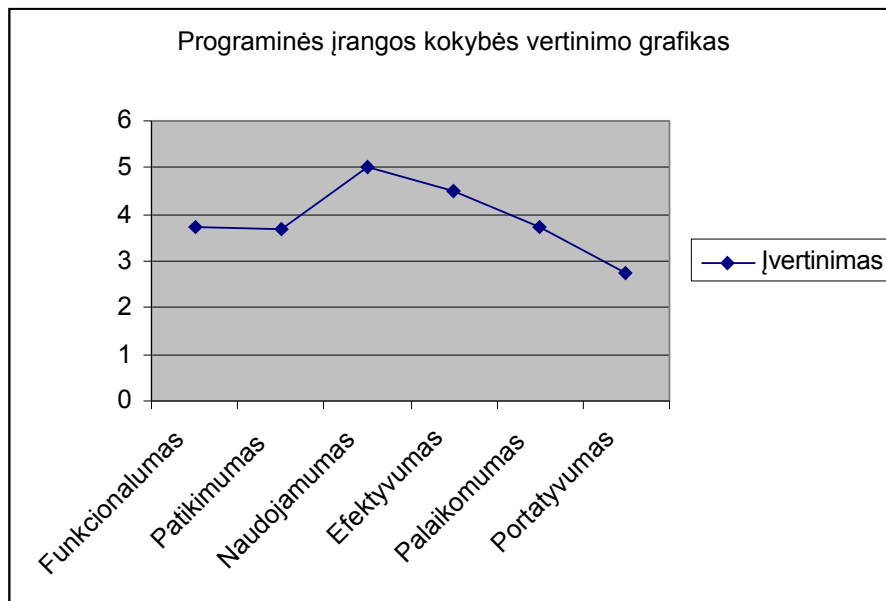
- Adaptyvumas – ar programinė įranga gali būti perkelta į kitą aplinką.
- Diegiamumas – ar programinė įranga yra lengvai diegiama.

- Atitikimas – ar programinė įranga atitinka portatyvumo standartus.
- Pakeičiamumas – ar programinė įranga gali lengvai pakeisti kitą PĮ.

Tyrimo rezultatai pateikiami žemiau esančioj įvertinimo lentelėje, kur kiekviena charakteristika vertinama skalėje nuo 1 iki 5, kur 1 reiškia prastai atitinka arba neatitinka, o 5 – puikiai atitinka charakteristiką. Pagrindinių charakteristikų vertė yra detalesnių charakteristikų įvertinimo aritmetinis vidurkis.

23 Lentelė. Sistemos kokybės atitikimo ISO 9126 standartui tyrimo rezultatų lentelė

Charakteristika	Įvertinimas
Funkcionalumas	3,75
Tinkamumas	4
Tikslumas	5
Funkcinis suderinamumas	1
Saugumas	5
Patikimumas	3,67
Užbaigtumas	4
Klaidų toleravimas	4
Atstatomumas	3
Naudojamumas	5
Suprantamumas	5
Išmokstamumas	5
Patrauklumas	5
Efektyvumas	4,5
Laiko savybės	5
Resursų naudojimas	4
Palaikomumas	3,75
Analizuojamumas	4
Keičiamumas	5
Stabilumas	4
Testuojamumas	2
Portatyvumas	2,75
Adaptyvumas	5
Diegiamumas	3
Atitikimas	1
Pakeičiamumas	2
Bendrai	3,9



35 Paveikslas. Sistemos kokybės atitikimo ISO 9126 standartui tyrimo rezultatų grafikas

Įvertinus visas charakteristikas ir gavus bendrą programinės įrangos įvertį, kuris yra 3.9 iš 5, darome išvadą, kad programinė įranga atitinka daugumą charakteristiką ir yra kokybiška.

17.DB objektų panaudojimo tyrimas

DB objektai yra daugiau žinomi kaip ActiveRecord pattern arba CRUD.

Programų sistemų inžinerijoje active record pattern yra architektūros šablonas dažniausiai randamas programinėje įrangoje, kuri saugo duomenis duomenų bazėse. ActiveRecord pattern apibrėžimas buvo sugalvotas Martino Fowlerio ir aprašytas jo knygoje „Patterns of Enterprise Application Architecture“ [17].

DB objektai tai priemonė lengvai ir paprastai valdyti įrašus duomenų bazėje. Duomenų bazės lentelė yra įvelkama į klasę ir tos klasės objektas yra sulyginamas su vienu lentelės įrašu. Sukūrus objektą ir išsaugojus, nauja eilutė yra įrašoma į lentelę. Bet koks užkrautas objektas gauna visą eilutės informaciją. Atnaujinus objekto informaciją, atitinkama eilutė atnaujinama ir lentelėje.

Standartinis objekto pavyzdys:

- - field1
- - field2
- Assign(\$data)
- Save()

- Insert()
- Delete()
- getObject()
- validate()
- getError(\$field_name)
- select(\$where)
- query()

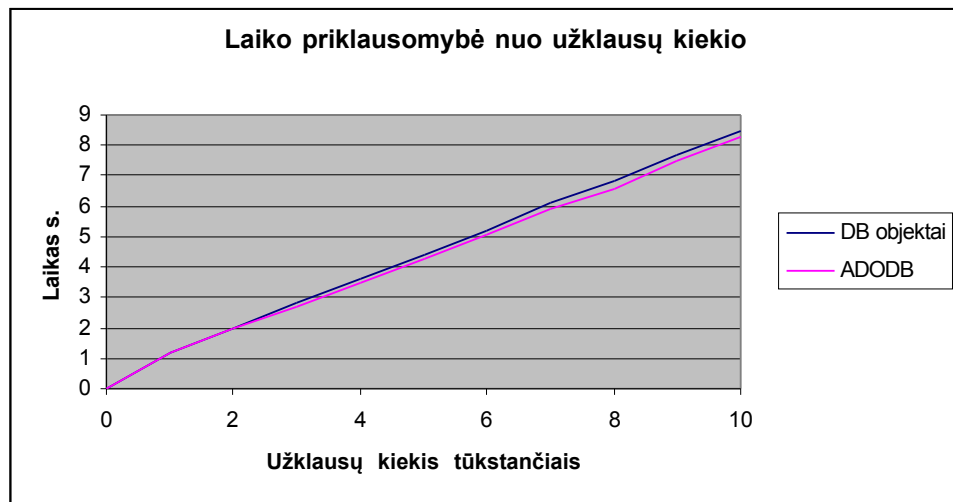
Kaip ir minėjau, lentelės įrašo laukai tampa objekto kintamaisiais, o pats objektas paveldi standartinius metodus skirtus valdyti įrašui.

Atlikus tyrimus nustatyta, kad DB objektai palengvina darbą programuotojui ir sumažina kodo kiekį daugiau nei 50%. Tuo pačiu mažėja laiko sąnaudos programuojant duomenų bazės įrašų valdymą.

Akivaizdu, kad naudojant DB objektus, kreipimasis į duomenų bazę bus vėlinamas, todėl buvo tiriama kaip DB objektai įtakoja sistemos veikimą laiko atžvilgiu esant skirtingam užklausų kiekiui.

24 Lentelė. Užklausų trukmės naudojant DB objektus rezultatai.

Užklausų kiekis (tūkst.)	DB objektai	ADODB
0	0 sek.	0 sek.
1	1,2 sek.	1,16 sek.
2	2 sek.	1,94 sek.
3	2,8 sek.	2,71 sek.
4	3,6 sek.	3,49 sek.
5	4,4 sek.	4,26 sek.
6	5,2 sek.	5,04 sek.
7	6,1 sek.	5,91 sek.
8	6,8 sek.	6,59 sek.
9	7,7 sek.	7,46 sek.
10	8,5 sek.	8,24 sek.



36 Paveikslas. Užklausų trukmės naudojant DB objektus rezultatų grafikas

Apibendrinant gautus tyrimo rezultatus, matome, kad laiko priklausomybė nuo užklausų skaičiaus yra tiesioginė, o ne eksponentiškai didėjanti. Skirtumas nuo tiesioginio kreipimosi į DB taip pat yra nedidelis – mažiau nei 3%. O tai nėra kritinis dydis.

Įvertinus gautus tyrimo plusus ir minusus daroma išvada, kad DB objektų naudojimas yra teigiamas sprendimas nes šios technologijos naudojimas atneša didesnę naudą. O šalutiniai efektai yra nedideli ir nedarantys didelės įtakos sistemos našumui.

18. IŠVADOS

Analitinėje dalyje buvo apibūdinta atviro turinio, arba turinio valdymo bendradarbiaujant sąvoka. Apžvelgti modeliai bei išskylančios problemos. Išnagrinėti trys analogai kuriamai sistemai bei nustatyti kuriamos sistemos reikalavimai.

Realizacijos dalyje pateikti funkciniai ir nefunkciniai reikalavimai, reikalavimai sistemos diegimo aplinkai bei pačios sistemos architektūra.

Tyrimo dalyje atliktas sukurtos sistemos tyrimas trim atžvilgiais. Ištirtas sistemos funkcionalumas ir nustatyta, kad sistema atitinka daugumą keltų reikalavimų, o taip pat funkcionalumu neatsilieka nuo esamų analogų. Taip pat ištirta kaip sistema atitinka ISO 9126 kokybės standartą. Trumpai apžvelgtos šio standarto charakteristikos. Nustatyta, kad sistemos atitikimas minėtam standartas yra geras ir vertinamas 3,9 balų iš 5 galimų. Realizuojant sistemą buvo pasirinkta naudoti DB objektus, kurie naudojami duomenims iš duomenų bazės, valdyti. Todėl buvo atliktas tyrimas ir įvertintas DB objektų naudojimas sistemoje tinkamumas. Atlikti tyrimai parodė, kad DB objektų naudojimas pasiteisino, nes buvo stipriai supaprastintas duomenų iš DB valdymas, o vėlinimas laiko atžvilgiu yra nedidelis ir nekritisinis.

19. TERMINŲ SANTRUMPŲ ŽODYNAS

WWW – World Wide Web – pasaulinis tinklas.

SOAP – Simple Object Access Protocol. Standartas leidžiantis keistis XML paremtomis žinutėmis kompiuteriniuose tinkluose.

PHP – Hypertext Preprocessor. Serverio pusės, daugiaplatformė programavimo kalba.

SQL – Structured query language. Programavimo kalba, skirta gauti ar atnaujinti informaciją duomenų bazėje.

ADODB – Duomenų bazės abstrakti biblioteka skirta PHP.

OS – Operacinė sistema

PDA – Nešiojamas įrenginys, kuris gali vykdyti kompiuterio, telefono, fakso funkcijas.

OPL – Atviro publikavimo licenzija.

XHTML – eXtensible HyperText Markup Language. Išplėsta žymėjimo kalba skirta interneto svetainėms kurti.

HTML – hyperText Markup Language. Žymėjimo kalba skirta interneto svetainėms kurti.

CSS – Cascading Style Sheet. Kalba skirta apibūdinti kaip HTML dokumentas turėtų būti atvaizduotas.

CVS – Concurrent Versioning System. Atviro kodo versijų kontrolės ir bendradarbiavimo sistema.

IRC Internet Really Chat. Pokalbių sistema.

WYSIWYG - Grafinė sąsaja kurti HTML puslapiams.

DB – duomenų bazė.

CRUD – trumpinys iš anglišku žodžių „Create read update delete“, kas reiškia sukurti, skaityti, atnaujinti, ištrinti.

ISO – International Organization for Standardization – tarptautinė standartų organizacija.

PA – Panaudos atvejis.

KTU – Kauno technologijos universitetas.

UML – unified modeling language – vieninga modeliavimo kalba - modeliavimo ir specifikacijų kūrimo kalba, skirta specifiuoti, atvaizduoti ir konstruoti objektiškai orientuotų programų dokumentus.

RUP – rational unified process - kartotinio programinės įrangos kūrimo metodika, sukurta įmonės Rational Software, nuo 2003 m. priklausančios IBM.

20.LITERATŪRA

- [1] U. Askland and L. Bendix, 2002. „A study of configuration management in opensource software projects” *IEE Proceedings — Software*, 40-46 psl.
- [2] K. Crowston and B. Scozzi, 2002. „Open source software projects as virtual organizations: Competency rallying for software development“ *IEE Proceedings — Software*, 3-17 psl.
- [3] J.G. Drummond, 2000. „Open source software and documents: A literature and online resource review“ at <http://www.omar.org/opensource/litreview/index.html>, žiūrēta 2006 12 02.
- [4] S.A. Hissam, D. Plakosh, and C.B. Weinstock, 2002. „Trust and vulnerability in open source software,” *IEE Proceedings — Software*, 47-51psl.
- [5] B. Katzy and K. Crowston, 2000. „A process theory of competency rallying in engineering projects“ at <http://crowston.syr.edu/papers/virtual-short.pdf>, žiūrēta 2006 12 05.
- [6] J. Newmarch, 2001. „Lessons from open source: Intellectual property and courseware“ at http://firstmonday.org/issues/issue6_6/newmarch/, žiūrēta 2006 12 05.
- [7] J. Sandred, 2001. *Managing open source projects: A Wiley tech brief*. New York: Wiley Computer Publishing.
- [8] F. Stalder and J. Hirsch, 2002. „Open source intelligence“ at http://firstmonday.org/issues/issue7_6/stalder/, žiūrēta 2006 12 05.
- [9] Steve museum sistema, [Žiūrēta 2006 12 03]. Prieiga per internetā http://www.steve.museum/index.php?option=com_content&task=view&id=35&Itemid=61
- [10] Flickr sistema, [Žiūrēta 2006 12 03]. Prieiga per internetā http://www.flickr.com/learn_more.gne
- [11] Wikipedia sistema, [Žiūrēta 2006 12 03]. Prieiga per internetā <http://en.wikipedia.org/wiki/Wikipedia>
- [12] Mediawiki sistema, [Žiūrēta 2006 12 03]. Prieiga per internetā http://www.mediawiki.org/wiki/How_does_MediaWiki_work%3F
- [13] Advantages and Disadvantages of Using Web Templates or WYSIWYG Editors for Web Design, [Žiūrēta 2006 12 03]. Prieiga per internetā <http://www.webknowhow.net/dir/HTML/Miscellaneous/061023WYSIWYGTemplates.html>
- [14] Creating Multilingual Web Sites, [Žiūrēta 2006 12 03]. Prieiga per internetā http://www.i18n.ca/workshops/MLW_overview.htm
- [15] Issues of content and structure for a multilingual web site, [Žiūrēta 2006 12 03]. Prieiga per internetā <http://delivery.acm.org/10.1145/510000/501537/p103-huang.pdf?>

[key1=501537&key2=7127716611&coll=&dl=ACM&CFID=15151515&CFTOKEN=6184618](http://www.acm.org/...key1=501537&key2=7127716611&coll=&dl=ACM&CFID=15151515&CFTOKEN=6184618)

[16] OpenContent License (OPL), [Žiūrėta 2006 12 03]. Prieiga per internetą <http://www.opencontent.org/opl.shtml>

[17] Martin Fowler 2003 „Patterns of Enterprise Application Architecture“