

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
PROGRAMŲ INŽINERIJOS KATEDRA

Jonas Abromaitis

**Programos vartojamumo tyrimas naudojant karkasą
vartotojo veiksmams registruoti ir analizuoti**

Magistro darbas

Darbo vadovas

dr. Tomas Blažauskas

Kaunas, 2012

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
PROGRAMŲ INŽINERIJOS KATEDRA

Jonas Abromaitis

**Programos vartojamumo tyrimas naudojant karkasą
vartotojo veiksmams registruoti ir analizuoti**

Magistro darbas

Vadovas

dr. Tomas Blažauskas
2012 - 05 -

Atliko

IFM-0/2 gr. stud.
Jonas Abromaitis
2012 - 05 - 17

Recenzentas

dr. Armantas Ostreika
2012 - 05 -

Kaunas, 2012

Summary

Usability is increasingly recognized as an important quality factor for interactive software systems, including traditional GUIs-style applications, Web sites, and the large variety of mobile and PDA interactive services. Unusable user interfaces are probably the single largest reasons why encompassing interactive systems – computers plus people, fail in actual use.

First of all we overview some main methods for usability testing and describe some metrics. There is no software that can collect these metrics and follow main methods. Main goal is to combine some different methods, extract metrics from them and create new method for usability testing. We will also create new software to collect some of these metrics and analyze them.

Later-on one web game will be tested so we could see if new method and framework works. All data according to metrics will be collected and analyzed. Some changes will be made to the games code and user interface and then we will see if usability was affected.

Santrauka

Vartojamumas tampa vienu iš pagrindinių programinės įrangos, tokios kaip internetiniai tinklalapiai, programų sistemos, RIA (angl. Rich Internet Application) aplikacijos bei mobiliųjų įrenginių programinė įranga, vertinimo faktoriumi. Blogas vartojamumas yra pagrindinė priežastis, kodėl kartais neįvyksta sėkminga žmogaus ir kompiuterio sąveika.

Darbe visų pirma peržvelgsime egzistuojančius metodus bei technologijas vartojamumo testavimui. Skirtingi metodai naudoja skirtingas metrikas, apibrėžiančias vartojamumo kokybę. Taip pat surasime keletą ekspertų rekomenduojamų būdų vartojamumo gerinimui. Vėliau visą šia informaciją panaudosime kuriant naują vartojamumo testavimo metodą, kurį taikant būtų galima pagerinti programų sistemos vartojamumą. Taip pat sukursime programinę įrangą – vartojamumo testavimo karkasą, kuris, integravus į testuojamą produktą, nuotoliniu būdu registruos vartotojų veiksmus ir leis juos analizuoti.

Parengus testavimo metodą ir karkasą bus bandoma efektyviai pagerinti vieno internetinio žaidimo „Nebula44“ vartojamumą. Proceso metu bus atliekami įvairūs tyrimai, bandant įrodyti metodo veiksmingumą bei pakoreguoti siūlomą metodą pridodant papildomus etapus ar metrikas arba pašalinant neveiksmingus.

Turinys

<u>Paveikslų sąrašas.....</u>	<u>7</u>
<u>Lentelių sąrašas.....</u>	<u>8</u>
<u>1.Įvadas.....</u>	<u>9</u>
<u>1.1.Kas yra vartojamumas?.....</u>	<u>9</u>
<u>1.2.Dokumento paskirtis.....</u>	<u>10</u>
<u>2.Analitinė dalis.....</u>	<u>11</u>
<u>2.1.QUIM modelis.....</u>	<u>11</u>
<u>2.1.1.Vartojamumo faktoriai.....</u>	<u>11</u>
<u>2.1.2.Išmatuojami vartojamumo kriterijai.....</u>	<u>13</u>
<u>2.2.„Keturių kampų“ technologija vartojamumui vertinti.....</u>	<u>16</u>
<u>2.2.1.Vartojimo testavimas.....</u>	<u>17</u>
<u>2.2.2. Peržiūros metodai.....</u>	<u>17</u>
<u>2.2.3.Pažintinis modeliavimas.....</u>	<u>18</u>
<u>2.2.4.Standartiniai klausimynai.....</u>	<u>18</u>
<u>2.3.Vartojamumo testavimo įrankiai.....</u>	<u>19</u>
<u>2.4.Siūlomas metodas bei jo pagrindimas.....</u>	<u>21</u>
<u>3.PROJEKGINĖ DALIS.....</u>	<u>24</u>
<u>3.1.Įvadas.....</u>	<u>24</u>
<u>3.1.1.Santrauka.....</u>	<u>24</u>
<u>3.1.2.Situacijos Lietuvoje įvertinimas.....</u>	<u>24</u>
<u>3.2.Produkto apibūdinimas.....</u>	<u>24</u>
<u>3.2.1.Programų sistemos funkcijos.....</u>	<u>24</u>
<u>3.2.2.Sistemos kontekstas.....</u>	<u>25</u>
<u>3.2.3.Vartotojo charakteristikos.....</u>	<u>25</u>
<u>3.2.4.Vartotojo problemos.....</u>	<u>25</u>
<u>3.2.5.Bendri apribojimai.....</u>	<u>25</u>
<u>3.3.Tikslas.....</u>	<u>26</u>
<u>3.4.Architektūros pasirinkimas.....</u>	<u>26</u>
<u>3.4.1.Sistemos paskirtis.....</u>	<u>26</u>
<u>3.4.2.Projekto kūrimo pagrindimas.....</u>	<u>27</u>
<u>3.4.3.Sistemos tikslai.....</u>	<u>27</u>
<u>3.5.Projekto išėja.....</u>	<u>28</u>
<u>3.5.1.Atviri klausimai.....</u>	<u>28</u>
<u>3.5.2.Galimos problemos.....</u>	<u>28</u>

3.5.3.Pritaikymas.....	29
3.5.4.Rizikos.....	29
3.5.5.Vartotojų dokumentacija ir apmokymas.....	29
3.5.6.Perspektyviniai reikalavimai.....	30
3.6.Projekto techninė specifikacija.....	30
3.6.1.Dokumento paskirtis.....	30
3.6.2.Apžvalga.....	30
3.6.3.Architektūros pateikimas.....	31
3.6.4.Architektūros tikslai ir apribojimai.....	31
3.6.5.Panaudojimo atvejų vaizdas.....	31
3.7.Sistemos statinis vaizdas.....	34
3.7.1.Apžvalga.....	34
3.7.2.Paketų detalizavimas.....	34
3.7.3.Bendradarbiavimo diagrama.....	36
3.7.4.Išdėstymo vaizdas.....	37
3.8.Kokybė.....	37
4.Tyrimo dalis.....	38
4.1.Atliekamo tyrimo metodologija.....	38
4.2.Pasirinktų tyrimo metodų paskirtis.....	38
4.3.Tyrimo parametrų aprašas.....	39
5.Eksperimentinė dalis.....	40
5.1.Tyrimas Nr. 1.....	40
5.2.Tyrimas Nr. 2.....	41
5.3.Tyrimas Nr. 3.....	42
5.4.Tyrimas Nr. 4.....	43
5.5.Tyrimas Nr. 5.....	45
6.Išvados.....	47
7.Rekomendacijos.....	48
8.Literatūros sąrašas.....	49
9.Terminų ir santrumpų žodynas.....	52
10.Priedai.....	53
10.1.Trumpa vartotojo dokumentacija.....	53
10.1.1.Sistemos funkcinis aprašymas.....	53
10.1.2.Sistemos galimybės.....	53
10.1.3.Vartotojo atmintinė.....	54

Paveikslų sąrašas

Paveikslas 1: Siūlomo metodo bendrinis vaizdas.....	21
Paveikslas 2: Karkaso panaudojimo atvejų bendrinis vaizdas.....	31
Paveikslas 3: Sistemos statinis vaizdas.....	34
Paveikslas 4: Integruotos dalies klasių diagrama.....	34
Paveikslas 5: Serverio dalies klasių diagrama.....	35
Paveikslas 6: Analizavimo įrankio klasių diagrama.....	35
Paveikslas 7: Visos sistemos bendradarbiavimo diagrama.....	36
Paveikslas 8: Visos sistemos išdėstymo diagrama.....	37
Paveikslas 9: Tyrimas 1, vidutinis laikas pasiūlymo turguje sukūrimo formos užpildymui.....	40
Paveikslas 10: Tyrimas 1, vidutinis laikas nustatymų keitimo formos užpildymui.....	41
Paveikslas 11: Tyrimas 2, vidutinis sesijos laikas.....	42
Paveikslas 12: Tyrimas 3, praneštų klaidų skaičius.....	43
Paveikslas 13: Tyrimas 4, pozityvių/negatyvių komentarų santykis.....	44
Paveikslas 14: Tyrimas 4, pozityvių/negatyvių komentarų skaičiai.....	44
Paveikslas 15: Tyrimas 5, bendra mokėjimų suma per keturis mėnesius.....	45
Paveikslas 16: Tyrimas 5, pirkusiųjų vartotojų kiekis ir vidutinis moka vartotojo mokėjimas. .	46

Lentelių sąrašas

Lentelė 1: QUIM modelyje siūlomi vartojamumo vertinimo kriterijai.....	13
Lentelė 2: Ryšiai tarp QUIM modelyje naudojamų kriterijų ir faktorių.....	15
Lentelė 3: Galimos sistemos ir kūrimo rizikos.....	29
Lentelė 4: Pasirinktų tyrimo metodų aprašas ir paskirtis.....	38

1. Įvadas

1.1. Kas yra vartojamumas?

Vartojamumas (angl. usability) – tai bendrinis terminas nusakyti vartotojų naudojimosi produktu kokybę. Į šią sąvoką įeina tokie faktoriai kaip vartotojo patiriamos emocijos, lengvai pasiekiamos funkcijos, patraukli akiai vartotojo sąsaja, greitai išmokstamas produkto naudojimas ir t.t.

Vartojamumo testavimas buvo naudojamas dar tada, kai ši sąvoka nebuvo žinoma. Pavyzdžiui Doug Engelbart, žmogus kuris išrado kompiuterinę pelę, atlikinėjo kompiuterio valdymo įrenginių efektyvumo tyrimus NASA organizacijoje dar 1965 metais [26]. Šiuos tyrimus irgi galima vadinti vartojamumo testavimu. 1981 metais Xerox PARC (angl. Palo Alto Research Center) tyrimų centras pradėjo vartoti sąvoką „vartojamumo testavimas“ kuomet norėjo pagerinti savo produktų efektyvumą. Vėliau ši sąvoka buvo minima vis daugiau kompanijų, o dar vėliau buvo sukurti ISO standartai vartojamumui apibrėžti, taip pat vartojamumo testavimas buvo pripažintas kaip programų kūrimo ir vystymo etapas.

Plečiantis internetinei rinkai tampa vis sunkiau konkuruoti su kitais produktais, todėl labai svarbu vartotojui pateikti kokybišką produktą. Vartojamumo gerinimas internetiniuose tinklalapiuose keičia tokius efektyvumo rodiklius kaip apsilankančių ir pasiliekančių vartotojų santykis (angl. bounce rate), aktyvių vartotojų skaičius, mokių vartotojų skaičius ir t.t. Dauguma kompanijose dirbančių vartojamumo ekspertų pasidalina savo sėkmės istorijomis, kaip vartojamumo testavimas jiems padėjo pagerinti produkto efektyvumą ar pelningumą, todėl nenuostabu, kad ir kitos kompanijos sužinojusios apie šios strategijos teikiamą naudą ieško būdų vartojamumui tobulinti.

Šiuo metu galima rasti gausybę įvairių metodų vartojamumo testavimui ir vertinimui, tačiau tampa sunku išsirinkti tinkamą strategiją savo produktui. Taip pat egzistuoja daug įvairių programinės įrangos produktų skirtų vartojamumo testavimui, tačiau šie produktai yra labiau susieti su testavimo praktika, o ne su įvairių tyrimų centrų apibrėžtais standartais ir metodais.

1.2. Dokumento paskirtis

Šis dokumentas – tai programų sistemų inžinerijos magistro studijų baigiamasis darbas. Jame pateikiama problema, egzistuojančių problemos sprendimų analizė, siūlomas sprendimas bei kuriama programinė įranga problemai spręsti. Taip pat pateikiami atliekami eksperimentiniai tyrimai norint įrodyti siūlomo problemos sprendimo veiksmingumą. Dokumento pabaigoje pateikiamos darbo metu sukauptos išvados.

2. Analitinė dalis

Visų pirma peržvelgsim jau egzistuojančius metodus vartojamumui testuoti ir vertinti. Mėginsim surasti egzistuojančių technologijų stipriąsias puses ir pasiūlysim atrinktų metrikų rinkinį.

2.1. QUIM modelis

QUIM (angl. Quality in Use Integrated Measurement) - tai bendrinis hierarchinis modelis, kuriame yra apjungti įvairūs vartojamumo vertinimo ir testavimo metodai. Šis modelis susideda iš 10 faktorių, kurių kiekvienas atstovauja atskirą vartojamumo aspektą. Visi aspektai yra paimti iš atskirų egzistuojančių standartų, metodų ar technologijų. Šie 10 faktorių yra skaidomi į 26 išmatuojamus kriterijus, kurie vėliau yra apibrėžiami 127 specifinėmis metrikomis. Savaime suprantama, kad tokį modelį panaudoti darosi sudėtinga dėl per didelio detalumo. Kyla klausimas: kaip teisingai pasirinkti modelyje apibrėžtas technologijas, kad būtų galima kaip įmanoma pilniau ir tiksliau įvertinti vartojamumą?

2.1.1. Vartojamumo faktoriai

Išanalizavus ISO 9241-11 (1998) standartus paaiškėjo, jog jis vartojamumo vertinimą apima plačiau nei ISO/IEC 9126 (2001) standartai. Senesniuose standartuose pagrindinis dėmesys skiriamas tokiems faktoriams kaip organizaciniai veiksniai, programinės įrangos išvaizda bei kokybė, pavyzdžiui 9126 ISO standartuose pasigendama funkcionalumo, tikslumo ir veiksmingumo. Pasak Bevan ir Schoeffel (2001), nei viena programų sistema neturi sau būdingo vartojamumo, o tik galimybę būti skirtingai panaudota įvairiuose kontekstuose. Pasinaudojus šiuo požiūriu galima rasti tam tikras naujesnių standartų dalis, papildančias senesni (1998 metų) standartų rinkinį. Beje, ISO 9241-11 yra plačiai pripažinti standartai Europoje. Būtent dėl šių priežasčių Ahmed Seffah, Mohammad Donyaee, Rex B. Kline ir Harkirat K. Padda (2006) pasirinko 1998 metų standartus kaip pagrindinį pamatą statant QUIM modelį.

Sudarant QUIM modelį buvo remtasi įvairiais standartais, technologijomis, metodais bei straipsniais ir galų gale buvo pateikta 10 pagrindinių faktorių apibrėžiančių vartojamumą:

1. **Efektyvumas**, arba kitaip programinės įrangos produkto galimybė, kad vartotojai, atlikdami vieną ar kitą veiksmą bei išnaudodami programinės įrangos funkcionalumą išseikvotų išteklius, tokius kaip laikas ar paspaudimų skaičius, atitinkančius veiksmų nauda tam tikram kontekste.
2. **Veiksmingumas**, arba kitaip galimybė vartotojams pilnai ir tiksliai vykdyti konkrečias užduotis.
3. **Našumas**, arba kitaip programinės įrangos teikiama nauda vartotojui investavus tam tikrus resursus (kad ir finansinius). Kartu su efektyvumu šis faktorius apibrėžia investuojamų resursų (efektyvumas) ir gaunamos naudos (našumas) balansą. Tiesa, ne visos užduotys turi gaunamą naudą, t. y. generuoja rezultatą, pavyzdžiui Macleod ir kt. (1997) pastebėjo, kad užduotis galime skirstyti į produktyvias ir neproduktyvias, tačiau šioje vietoje rezultatu galima laikyti užduoties atlikimo faktą.
4. **Pasitenkinimas**, kurį patiria vartotojai besinaudodami programine įranga. Ši informacija dažniausiai iš vartotojų surenkama jiems pildant iš anksto sudarytas anketas.
5. **Išmokstamumas** arba kaip lengvai vartotojai gali išmokti naudotis programų sistema. Tai vartotojų savijauta, kad jie gali produktyviai naudoti programinės įrangos produktą iš karto ir greitai išmokti naudotis kitomis naujomis funkcijomis.
6. **Sauga**, kuri pagal ISO/IEC 9126-4 (2001) standartą gali būti suprantama dviem aspektais: eksploatavimo sauga, kuri aprėpia programų sistemos būklę normalaus veikimo sąlygomis, bei nenumatytų ar neįprastų atvejų sauga, kuri galioja kritiniais momentais kai programų sistema patenka į neįprastą būseną (kritinės klaidos ir pan.).
7. **Patikimumas**, kurį apibrėžia vartotojų ištikimybė programų sistemai. Šis faktorius labiausiai galioja elektroninės prekybos tinklalapiams (Ahuja, 2000; Atif, 2002; Cheskin Research & Studio Archetype/Sapient, 1999; Friedman ir kt., 2000; Tilson et al., 1997) tačiau gali būti taikomas ir kitoms programų sistemoms.
8. **Prieinamumas** – tai programų sistemos galimybė būti naudojamai žmonių su negaliomis, pvz.: silpnaregių, kurčių ir pan. žmonių. “The World Wide Web Consortium” (Caldwell ir kt., 2004) pateikia gausybę naudingų patarimų kaip pritaikyti programinę įrangą vartotojams turintiems negalią.

9. **Universalumas** – ar programų sistema taip pat sėkmingai gali naudotis skirtingų kultūrų atstovai? Pvz.: Edukaciniai ar religiniai projektai.
10. **Naudingumas** – ar programų sistema sprendžia kažkokias realias bėdas; ar programų sistema palengvina užduotis ir pan.

2.1.2. Išmatuojami vartojamumo kriterijai

Praeitame skyrelyje pateikta 10 faktorių, kurie apsprendžia vartojamumą. Vadovaujantis šiais faktoriais yra sudaromi 26 išmatuojami kriterijai, kurių kiekvienas turi bent po 1 tiesiogiai pamatuojamą matmenį. Sekančioje lentelėje pateikiami 26 kriterijai kuriais QUIM modelyje yra siūloma vertinti vartojamumą:

Lentelė 1: QUIM modelyje siūlomi vartojamumo vertinimo kriterijai

Nr.	Kriterijus	Aprašymas
1	Sugaištamas laikas	Užduoties sugaištamas laikas kol ji atlieka savo funkciją.
2	Išteklių panaudojimas	Galimybė vartoti atitinkamą kiekį atitinkamos rūšies išteklių, kai programinė įranga atlieka savo funkcijas (ISO / IEC 9126-1, 2001)
3	Patrauklumas	Programinės įrangos produktas turi būti patrauklus vartotojui (pvz., spalva ar grafinis dizainas, naudojant ISO / IEC 9126-1, 2001 m.)
4	Patinkamumas	Vartotojo suvokimas, jausmai ir nuomonė apie produktą (Rubin, 1994)
5	Keičiamumas	Ar vartotojas gali nustatymu pagalba pakeisti programinės įrangos produkto vartotojo sąsaja taip, kad ji atitiktų vartotojo asmeninius pageidavimus?
6	Minimalus veiksmų skaičius	Reikia siekti, kad programinės įrangos produktas padėtų vartotojams atlikti savo užduotis mažiausiu etapų skaičiumi.
7	Minimali atminties apkrova	Vartotojas turi išlaikyti minimalų kiekį informacijos galvoje norėdamas įvykdyti tam tikrą užduotį (Lin ir kt., 1997)
8	Reikalaujamos pastangos	Kiek pastangų reikia valdyti ir kontroliuoti programinės įrangos produktą
9	Pagalba vartotojui	Ar programų sistema padeda vartotojui gaudyti kontekste ir suteikia reikiamą pagalbą jei įvyksta klaida ar nesusipratimas
10	Pastovumas, vienodumas	Ar vartotojo sąsajos elementai panašūs, dera vienas su kitu. Ar atskirtos konteksto grupės ir t.t.
11	Suprantamumas,	Ar programinės įrangos produktas aiškiai pateikia savo paskirtį ir

Nr.	Kriterijus	Aprašymas
	intuitivumas	vartotojo sąsaja yra intuityvi? Ar vartotojas žino kas bus kai paspaus atitinkamą mygtuką?
12	Grįžtamasis ryšys	Programos reagavimas į vartotojo atliekamus veiksmus bei programų sistemoje vykstančius įvykius.
13	Tikslumas	Galimybė pateikti teisingus ir tikslus rezultatus (ISO / IEC 9126-1, 2001)
14	Išsamumas	Ar vartotojas gali pilnai atlikti tam tikrą užduotį?
15	Atsparumas gedimui	Programinės įrangos produkto galimybė išlaikyti funkcionalumą klaidų ar gedimų metu (ISO/IEC 9126–1, 2001) .
16	Išteklių apsauga	Ar visi ištekliai (įskaitant žmones) yra tinkamai prižiūrimi ir saugomi?
17	Skaitomumas	Ar visas vizualinis turinys, toks kaip tekstiniai dialogai ir panašiai, gali būti lengvai skaitomas ir suprantamas?
18	Kontroliavimas	Ar vartotojai jaučia, kad jie pilnai valdo programinės įrangos produktą?
19	Navigacija	Ar vartotojai gali lengvai, veiksmingai ir greitai pereiti iš vienos programų sistemos dalies į kitą?
20	Paprastumas	Ar programų sistema nėra apkrauta nereikalingais elementais? Reikia pašalinti kaip įmanoma daugiau elementų, kurie nesuteikia naudingos informacijos ar funkcionalumo.
21	Privatumo politika	Ar vartotojų asmeninė informacija yra tinkamai apsaugota?
22	Saugumas	Programinės įrangos produkto gebėjimas apsaugoti informaciją ir duomenis taip, kad pašaliniai asmenys ar sistemos be įgaliojimų negalėtų jų skaityti ar modifikuoti (ISO/IEC 12207, 1995) .
23	Draudimas	Programinės įrangos produkto tiekėjų atsakomybė neteisėto vartotojų asmeninės informacijos naudojimo atveju.
24	Pažįstamumas	Ar vartotojo sąsajos elementai yra įprasti vartotojams? Ar naudojami standartiniai įvesties metodai ir pan.
25	Krovimosi laikas	Laikas reikalingas tinklalapiui užsikrauti (ty, kaip greitai jis reaguoja į vartotojo veiksmus)
26	Tinkamumas	Ar vizualinė medžiaga (paveiksliukai, diagramos ir t.t.) yra prasminga ir suprantama vartotojui?

Kadangi kiekvienas išmatuojamas kriterijus veikia vieną arba daugiau vartojamumo faktorių, galime sudaryti ryšių lentelę:

Lentelė 2: Ryšiai tarp QUIM modelyje naudojamų kriterijų ir faktorių

Kriterijus	Efektyvumas	Veiksmingumas	Našumas	Pasitenkinimas	Išmokstamumas	Sauga	Patikimumas	Prieinamumas	Universalumas	Naudingumas
Sugaištamasis laikas	+			+						
Išteklių panaudojimas	+			+						+
Patrauklumas			+						+	
Patinkamumas			+							
Keičiamumas		+	+					+	+	+
Minimalus veiksmų skaičius	+		+		+			+		
Minimali atminties apkrova	+		+		+			+	+	+
Reikalaujamos pastangos	+		+				+	+		+
Pagalba vartotojui			+		+			+	+	
Pastovumas, vienodumas		+			+	+		+	+	
Suprantamumas, intuitivumas					+		+	+	+	
Grižtamasis ryšys	+	+							+	+
Tikslumas		+				+				+
Išsamumas		+				+				
Atsparumas gedimui						+	+			+
Išteklių apsauga						+				
Skaitomumas								+	+	
Kontroliavimas							+	+	+	+
Navigacija	+	+					+	+	+	

Kriterijus	Efektyvumas	Veiksmingumas	Našumas	Pasitenkinimas	Išmokstamumas	Sauga	Patikimumas	Prieinamumas	Universalumas	Naudingumas
Paprastumas					+			+	+	
Privatumo politika							+		+	+
Saugumas						+	+			+
Draudimas						+	+			
Pažįstamumas					+		+			
Krovimosi laikas	+			+					+	+
Tinkamumas			+	+	+			+		

2.2. „Keturių kampų“ technologija vartojamumui vertinti

Nėra jokio prietaiso, metodo ar technologijos, kuri pasakytu kiek vartotina yra programų sistemos vartotojo sąsaja. Praktikoje dažniausiai yra stebima vartotojų sąveika su produktu, atkreipiamas dėmesys į nepageidaujamus veiksnius, ir tada, gerinama vartotojo sąsaja siekiant išvengti nepageidautinos vartotojų sąveikos su produktu.

Taip pat nėra jokios auksinės taisyklės apibrėžiančios kaip testuoti vartojamumą ir kokius metodus naudoti. Kaip tik, vartojamumo ekspertai ragina naudoti kuo daugiau įvairių technikų ir metodų žvelgiant į vartojamumą skirtingais aspektais. Paeitame skyrelyje aptartame QUIM modelyje į vartojamumą buvo žiūrima iš įvairių aspektų, tačiau modelis buvo gana sudėtingas ir jį taikyti ne visada pravartu.

Jeff Sauro (2011) dirbantis „Measuring Usability LLC“, kuri dirba su tokiomis kompanijomis kaip „PayPal“, „Autodesk“, „intuit“ ir t.t. siūlo naudoti „keturių kampų“ metodą. Tai tiesiog keletas vartojamumo testavimo technologijų, suskirstytų pagal tam tikrus bendrus požymius. „Keturių kampų“ metode aptariamos vartojamumo testavimo technologijos labiau orientuotos į programinę įrangą ir jos patogų naudojimą. Kitas populiarus požiūris yra vartotojų apibrėžimas, grupavimas ir jų tikslų išsiaiškinimas, stengiantis pritaikyti programinę įrangą vartotojams kuo labiau juos pažįstant.

2.2.1. Vartojimo testavimas

Elementarus vartojimo stebėjimas – tai pirmas šio virtualaus keturkampio kampas. Pagalvojus apie vartojamumą visų pirmą į galvą ateina mintis apie vartotoją, kuris naudojami programų sistema. Nors vartotojų stebėjimas gali atrodyti savaime suprantama produkto kūrimo dalis, tačiau vis dar egzistuoja tendencija galvoti, kad mes galim nuspėti, kaip vartotojas elgsis vartodamas produktą ir su kokiomis problemomis jis susidurs. Nors programų kūrėjai irgi vartotojai ir gali patys vartoti produktą, tačiau kiekvienas vartotojas yra skirtingas ir įvairios unikalių vartotojų bei jiems duodamų uždavinių kombinacijos gali sugeneruoti visiškai nenuspėjamus rezultatus.

Šioje vietoje galima išskirti du vartojamumo testavimo tipus:

1. **Problemų aptikimo testai** – jų metu stengiamasi aptikti problemas, kurias vartotojai patiria sąveikaudami su vartotojo sąsaja ir ištaisyti jas.
2. **Palyginamieji (angl. Benchmark) testai** – jų metu yra renkama statistika, kaip vartotojams sekasi įvykdyti pateiktas užduotis (nesėkmių skaičius, laikas užduočiai atlikti ir t.t.), įvertinimai lyginami su laukiamais ir atitinkamai daromos išvados.

Vartojimo testavimo metu vartojamumo ekspertas arba kitas atsakingas asmuo ar asmenų grupė sudaro užduočių ar scenarijų rinkinius atrinktai vartotojų grupei. Vartotojai gali būti lokaliai stebimi ekspertų arba atlikinėti užduotis per atstumą siunčiant naudingą informaciją internetu (vaizdo kamerų pagalba, specialios programinės įrangos pagalba ar pan.). Testavimo metu yra renkama įvairi informacija, tokia kaip vartojamumo problemos, sėkmingai užbaigtų užduočių santykis, užduoties atlikimo laikas ir t.t. Testavimas lokaliaje patalpoje kiekvieną vartojimo atvejį stebint ekspertui yra naudingas kai norima atrasti konkrečias vartojamumo problemas arba sužinoti ilgai užtrunkančių užduočių gaišinimo priežastis. Nuotolinis testavimas taikomas kai yra parinkta didelė testuojamų vartotojų grupė ir reikalingas didelis tikslumas.

2.2.2. Peržiūros metodai

Profesionalai, kurie yra patyrę dizaino bei žmogaus ir kompiuterio sąveikavimo srityse, gali peržiūrėti vartotojo sąsają ir pastebėti problemas su kuriomis gali susidurti vartotojai. Tai yra

ne tik ekspertų nuomonė, principai pagal kuriuos yra vertinamas programų sistemos vartojamumas yra vystomi jau daug metų ir ši technika pastoviai tobulinama.

Ekspertui yra duodama programų sistema bei įvairūs jos panaudojimo atvejai. Tada ekspertas peržiūri kiekvieną atvejį atskirai ir užrašo iškilusias arba galinčias iškilti problemas. Geriausia šį metodą taikyti su keliais ekspertais, kadangi kai kurie ekspertai gali rasti problemų ten kur jų nėra, o kiti gali tiesiog nepastebėti egzistuojančių problemų.

2.2.3. Pažintinis modeliavimas

Panašiai kaip peržiūros metuose, pažintinio modeliavimo metu irgi vartotojai nėra tiesiogiai apklausiami ar stebimi. Šio proceso metu yra sudaromi galimi vartojimo keliai, scenarijai, kokia tvarka vartotojai galėtų atlikti veiksmus. Tada kiekvienas veiksmas yra skaidomas į „operatorius“. Čia operatorius – tai minimalus neskaidomas veiksmas, kurį galima apibrėžti laiko trukme. Dažniausiai operatoriai, tai: pelės paspaudimas, pelės vedimas, galvojudimas, skaitymas, rašymas ir t.t. Tada yra bandoma nuspėti arba nustatyti kiek laiko tipinis vartotojas užtruks atlikti vieną operatorių. Teigiama, kad toks – dekompozicinis metodas, padeda aptikti daug vartojamumo problemų bei pakankamai tiksliai nuspėti užduočių atlikimo laiką.

2.2.4. Standartiniai klausimynai

Vartotojui pasinaudojus programų sistema jam galima duoti užpildyti standartinę anketą/klausimyną. Egzistuoja įvairios klausimynų formos, klausimus gali sudarinėti pats programų sistemų kūrėjas, testuotojas ar užsakovas, tačiau labiausiai paplitę yra jau sudaryti ir pasiteisinę klausimynai sudaryti iš 10 klausimų. Kiekvienas klausimas gali būti įvertintas balais nuo 1 iki 5. Šis standartinio klausimyno variantas dar vadinamas SUS (angl. System usability scale). Pastarasis klausimynas dažniau taikomas programų sistemoms, egzistuoja kitas standartinis klausimynas sudarytas iš 13 klausimų, kuris yra taikomas specialiai interneto tinklalapiams vertinti – tai SUPR-Q (angl. The Standardized Universal Percentile Rank Questionnaire). Žemiau pateikiami 10 SUS klausimų:

1. Manau, kad norėčiau naudotis šia sistema reguliariai.
2. Manau, kad sistema yra nereikalingai sudėtinga.
3. Manau sistema yra lengva naudotis.

4. Manau, kad man reikėtų techninio personalo pagalbos, kad galėčiau naudotis šia sistema.
5. Manau, kad šioje sistemoje buvo gerai integruota daug įvairių funkcijų.
6. Manau, kad sistema netolydi ir nederanti.
7. Manau, kad dauguma žmonių išmoktų naudotis šia sistema labai greitai.
8. Man buvo labai nepatogu naudotis šia sistema.
9. Naudojantis sistema jaučiausi patikimai.
10. Man reikėjo išmokti daug dalykų prieš normaliai naudojantis sistema.

Surinkus vartotojų atsakymus reikia įvertinti kiekvieną anketą. Anketos atsakymai (balais nuo 1 iki 5) yra perdirbami iš kiekvieno nelyginio atsakymo atimant 1, o kiekvieną lyginį atsakymą atimant iš 5. Taip atsakymai yra pertvarkomi, kad jų taškai svyruotų nuo 0 iki 4. Tada visi taškai yra sudedami ir padauginami iš 2.5, taip gaunant įvertinimą nuo 0 iki 100.

Šis metodas patogus, kai reikia palyginti dviejų vartotojo sąsajų ar produktų vartojamumo kokybę.

2.3. Vartojamumo testavimo įrankiai

Pasaulyje egzistuoja gausybė įvairių vartojamumo testavimo įrankių. Apžvelgsime kelis populiariausius įrankius palygindami jų funkcionalumą, praktinį pritaikymą ir kainą.

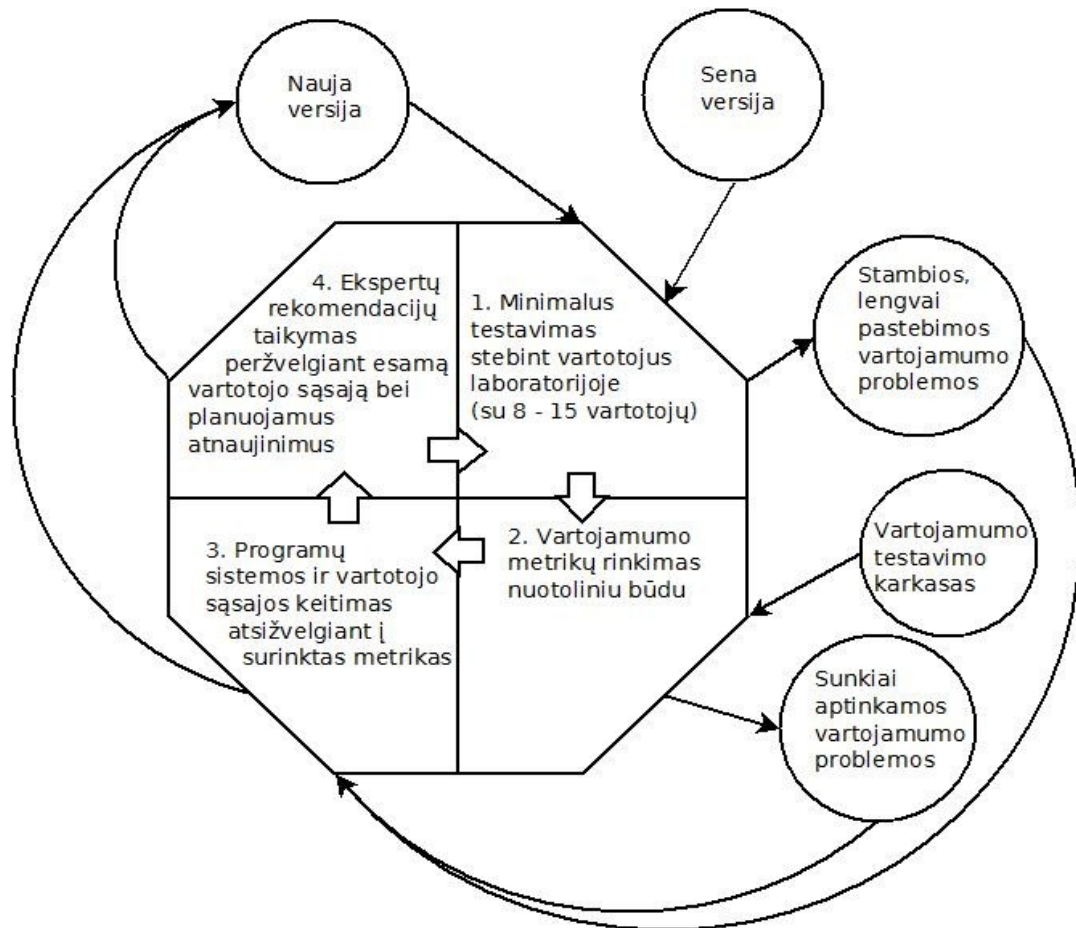
1. Concept Feedback – ekspertai praneša pastabas, patarimus ir savo nuomonę apie pateiktą produktą. Ekspertas gali būti įvertintas taškais, jei jo patarimas buvo naudingas. Kuo daugiau taškų turi ekspertas, tuo didesnę dėmesį reikėtų kreipti į jo patarimą. [11]
2. „Popierius ir pieštukas“ – pati pigiausia ir paprasčiausia testavimo priemonė, tačiau testuotojas turi būti šalia testuojamojo vartotojo ir rašyti pastabas. Beje ši priemonė turi vieną išskirtinį privalumą – testuojamųjų gali būti paklausama kodėl jie elgiasi vienaip ar kitaip. [12]
3. Chalkmark – ši priemonė yra labai panaši į kuriamą karkasą. Testuotojas pateikia ekrano su vartotojo sąsaja nuotrauką ir išsamius užduoties aprašymus. Priemonė sugeneruoja URL adresu pasiekiamą testavimo puslapį. Testuotojas nusiunčia adresą testuojamiems vartotojams ir šie gali pradėti testavimą. Testavimo metu priemonė registruoja visus pelės paspaudimus, ir bet kuriuo testavimo metu testuotojas gali parsisiųsti ataskaitą PDF

- formatu. Ataskaitoje pateikiamas šilumos žemėlapis (angl. „heatmap“) kur parodytos vietos, kurias bandė spausti testuojami žmonės. Taip pat yra pateikiami kiekvienam paspaudimui sugaišti laikai. Priemonė kainuoja 109\$ mėnesiui, arba 559\$ metams. [13]
4. ClickHeat – ši priemonė yra panaši į „Chalkmark“ (aprašytą aukščiau), tačiau yra visiškai nemokama, atviro kodo, ir taikoma realaus vartojimo metu. Vartotojams naudojantis puslapiu, jų veiksmai yra registruojami, vėliau galima peržiūrėti jų atliktus pelės paspaudimus šilumos žemėlapyje. Tačiau ši priemonė tinka tik HTML puslapiams. [14]
 5. ClickTale - ši priemonė registruoja vartotojo pelės judėjimą, klaviatūros bei pelės mygtukų paspaudimus, įrašydama viską vaizdo įrašo pavidalu. Ryškus apvadas, piešiamas aplink pelės žymeklį ir kitos panašios priemonės padeda lengviau sekti vartotojo veiksmus. Ši priemonė nesuteikia galimybės lengvai atlikti didelių vartotojų kiekių bendros vartojamumo analizės. Kaina įvairi, nuo nemokamos iki auksinės (angl. „Gold“) licencijos (0\$ - 790\$ per mėnesį) priklausomai nuo testavimo poreikių. [15]
 6. Clixpy - tai priemonė registruojanti visus vartotojo veiksmus (nuo pelės judėjimo, iki formų pildymo), veikianti tik ant HTML puslapių, tačiau kaina priklauso nuo vartojimo įrašų skaičiaus. 5\$ už 100 įrašų, 10\$ už 200, 20\$ už 600 ir 30\$ už 1,000. [16]
 7. Ethnio - ši priemonė leidžia vartojimo metu rasti vartotoją, tinkantį testavimui ir su juo bendrauti per internetą. Vartotojas turi sutikti būti testuojamu, ir turi užpildyti formą, kur yra nustatoma ar šis vartotojas atitinka norimo testuoti vartotojo reikalavimus. Pirmi 20 testuojamų vartotojų yra nemokami, vėliau 200 testuojamų vartotojų kainuoja 400\$, o 2000 – 800\$. [17]
 8. Feng-GUI - įdomi priemonė, atliekanti testavimą naudodama algoritmus, kurie (kaip jos kūrėjai sako) atitinka žmogaus akį traukiančių puslapio elementų, jų išdėstymo bei bendro vaizdo poveikį vartotojui. Testuotojui reikia nusiųsti tiriamos vartotojo sąsajos ekrano nuotraukas. Priemonė pateikia analizę su šilumos žemėlapiu, kur labiausiai „prikaustoma“ žmogaus akis, kokie numatomi žvilgsnio judesiai ant ekrano ir panašiai. Vienos ekrano nuotraukos analizė nemokama, 50 nuotraukų - 50\$, 250 - 225\$, 500 - 400\$ ir 1,000 – 700\$. [18]
 9. Five second test - priemonė, skirta nustatyti eilinio vartotojo atliktus veiksmus bei jų priežastis. Galimi du testavimo būdai: testuoti su atsitiktiniais vartotojais užsukančiais į „fivesecondtest“, arba testuoti su pakviestais testuotojais. Kiekvienas atlikęs testą pateikia kelias priežastis kodėl paspaudė vienoje ar kitoje vietoje. Pagrindinė testo savybė – testas atliekamas per 5 sekundes. Priemonė nemokama. [19]

10. Loop11 - priemonė leidžianti iširti internetinio puslapio vartojamumą, neatliekant jokių modifikacijų puslapyje. Yra sukuriamos užduotys bei klausimai vartotojams, atliekamas testavimas (galimas net ir konkurentų puslapiuose) ir užpildoma anketa su klausimais. Surinkus anketas pateikiama analizė su diagramomis, atsakymu skaičiais ir panašiai. [20]
11. Mechanical Turk – ši priemonė leidžia palikti informaciją vartojamumo testuotojams, ir gauti jų išvadas. Vartojamumo testuotojai registruojasi tam pačiam puslapyje ir jiems yra mokamas piniginis atlyginimas už kiekvieną atliktą vartojamumo tyrimą. [21]

2.4. Siūlomas metodas bei jo pagrindimas

Išanalizavus aptiktus metodus, technologijas bei priemones vartojamumo testavimui buvo nuspręsta kurti naują metodą pasitelkus tam tikras esamų metodų technikas bei rekomendacijas. Siūlomas metodas susideda iš 4 apibendrintų etapų, pateiktų paveikslėlyje:



Paveikslas 1: Siūlomo metodo bendrinis vaizdas

Metode siūloma iš pradžių seną testuojamą produkto versiją išbandyti su atrinktais testuotojais laboratorijoje, taip aptinkant lengvai pastebimas, dažnai pasitaikančias ir stambias vartojamumo problemas. Jei buvo aptikta stambių problemų, kurias buvo nutarta nedelsiant taisyti, savo ruožtu atliekant pakeitimus yra generuojama nauja versija.

Atlikus minimalų testavimą laboratorijoje galima pereiti prie antro etapo – nuotolinio metrikų rinkimo. Šiam etapui naudojamas specialiai sukurtas karkasas, renkantis ir grupuojantis metrikas, kurios yra generuojamos vartotojams naudojantis testuojama programų sistema.

Vėliau siūloma pirmuose dviejuose etapuose surinktas metrikas peržvelgti aptinkant įvairias vartojamumo problemas. Jei metrikų rinkimas buvo atliktas ir su senesne programų sistemos versija, rezultatus galima palyginti įvertinant buvusių pakeitimų įtaką vartojamumui. Aptiktas problemas reikia spręsti keičiant programų sistemos vartotojo sąsają ar tam tikras funkcijas. Taip pat keičiant programų sistemą siūloma atsižvelgti į ekspertų rekomendacijas, kurios pateikiamos sekančiame etape.

Atlikus visus pakeitimus yra sugeneruojama nauja programų sistemos versija, kurią vėl reikia testuoti pradėdant nuo pirmo modelyje siūlomo etapo. Vartojamumo testavimo modelį siūloma taikyti visą programų sistemos gyvavimo laikotarpį.

Detaliau aptarsime kiekvieną metode esantį etapą:

1. Minimalus testavimas stebint vartotojus laboratorijoje – tai specialiai atrinktų vartotojų bandomasis sistemos naudojimas. Vartotojui yra duodamos konkrečios užduotys, o vartojamumo ekspertai stebi bei įrašinėja sesiją, taip yra pastebimos stambios vartojamumo problemos.
2. Vartojamumo metrikų rinkimas nuotoliniu būdu – tai metodas, kuriam įgyvendinti reikalinga speciali programinė įranga. Dažnai vartojamumo testavimo metu metrikų rinkimui naudojama internetinė vaizdo kamera, ekrano vaizdo įrašą daranti programinė įranga ar kitas vienos sesijos duomenis kaupiantis produktas, tačiau norint pasiekti statistiškai patikimų rezultatų reikalinga programinė įranga renkanti, grupuojanti ir analizuojanti paprasčiausias vartojamumą nusakančias metrikas. Kartu su metodu pateiksime ir programinę įrangą – vartojamumo testavimo karkasą – kuris rinks dalį metode siūlomų metrikų.
3. Peržiūrint surinktas metrikas galima susidaryti bendrą vaizdą apie programų sistemos vartojamumą bei daryti išvadas apie efektyvumą. Vietos, kuriose efektyvumas netenkina

lūkesčių, galima bandyti gerinti vartojamumą supaprastinant vartotojo sąsają bei taikant ekspertų rekomendacijas, kurios pateikiamos kartu su metodu.

4. Metode siūloma kaupti visas ekspertų rekomendacijas vienam sąrašui, taip dalinantis sukaupia patirtimi. Internetu galima rasti daug įvairių rekomendacijų pateiktų mažais kiekiais, pavyzdžiui: „10 naudingų patarimų gerinant vartojamumą“ [25]. Jei metodas pasiteisintų ir rekomendacijų sąrašas išaugtų iki tokio dydžio, kad jį būtų nebeapatogu skaityti, būtų galima jį skaidyti į grupes ar skyrius, taip sudarant rekomendacijų žinyną.

Toliau paminėtos pagrindinės siūlomo metodo savybės bei pagrindimai:

1. Nielsen Jakob ir Landauer Thomas K [22] savo tyrime nustatė, kad vartotojams kylančioms emocijoms bei stambioms vartojamumo problemoms aptikti užtenka 8 – 15 stebimų vartotojų. Todėl metode rekomenduojama visų pirma atlikti testavimą laboratorijoje su minėtu kiekiu vartotojų. Mus domina ir problematiškesnė vartojamumo testavimo dalis – smulkios, sunkiai aptinkamos vartojamumo problemos stipriai įtakojančios vartojamumą, kurias lengviau aptikti renkant statistinius duomenis - metrikas. Taip pat ir kai kurie ekspertai teigia, jog atliekant nuotolinį testavimą galima aptikti žymiai daugiau problemų nei lokalaus testavimo metu [23].

2. Vienos iš svarbiausių metrikų, kurios įeina į siūlomą metodą yra sesijos laikas bei būsenose sugaištamasis laikas. Šios metrikos tiesiogiai parodo kiek laiko programų sistema sugeba išlaikyti vartotoją arba kiek laiko vartotojas sugaišta atlikdamas visas norimas užduotis.

3. Norint, kad vartotojų atliekami veiksmai būtų kuo autentiškesni, vartotojai net nežinos, kad jų veiksmai yra stebimi ir jie dalyvauja testavime. Visos metrikos ir informacija yra renkama be jokių trukdžių vartotojo ir programų sistemos sąveikai.

4. Dar viena, kiek rečiau naudojama metrika, kuri įeina į siūlomą metodą – tai pozityvių ir negatyvių komentarų atsiliepimų lange santykis. Jis parodo vartotojų nuomonę apie produktą, nepateikiant vartotojams jokių klausimynų ar anketų.

5. Siūloma naudoti ekspertų rekomendacijas vartojamumui gerinti. 10 naudingų rekomendacijų, kurias pravartu peržvelgti kuriant bet kokią programų sistemą yra pateikta Dmitry Fadeyev straipsnyje [25].

3. PROJEKTINĖ DALIS

3.1. Įvadas

3.1.1. Santrauka

Projekto tikslas yra sukurti karkasą Flex4 programavimo aplinkai, leidžiantį nuotoliniu būdu testuoti sistemos vartojamumą. Karkasas turi įgyvendinti siūlomą vartojamumo testavimo metodą.

Programų sistemų kūrėjas, taikydamas šį karkasą savo projektams, galėtų pašalinti vartojamumo spragas, kurių nebūtų pastebėjęs be tinkamo įrankio. Kartais tokios spragos gali kainuoti daugiau nei galime įsivaizduoti (Pvz.: Vieno mygtuko pakeitimas internetinei parduotuvei sutaupė 300 000 000\$ metinio pelno [27]). Taip pat, šį karkasą norėtų įsigyti ir kiti programinės įrangos kūrėjai, besinaudojantys Flex4 technologijomis, kadangi vartojamumo testavimas kasdien tampa vis svarbesniu projektavimo etapu.

3.1.2. Situacijos Lietuvoje įvertinimas

Šiuo metu Lietuvoje nėra kuriama daug projektų naudojantis Flex4, tačiau pasaulyje labai sparčiai populiarėja šios technologijos taikymas kuriant internetinius projektus. Pastebėti pirmi Flex4 technologijos naudojimo atvejai Lietuvoje leidžia daryti išvada apie artėjanti karkaso paklausos Lietuvoje augimo būmą.

3.2. Produkto apibūdinimas

3.2.1. Programų sistemos funkcijos

Karkasas registruoja vartotojų atliekamus veiksmus, tokius kaip mygtuko paspaudimai, navigacija į programos dalis, sugaištas laikas ar pelės judėjimas. Taip pat karkasas leidžia registruotus duomenis analizuoti, grupuoti, peržvelgti atskirus atvejus ir pan.

3.2.2. Sistemos kontekstas

Programų sistemą sudaro trys programos:

1. Pirma programa „įrišama“ į testuojamo projekto kliento programą. Ji registruoja konfigūraciniame faile nurodytus žingsnius ir siunčia informaciją apie vartojamumą serveriui.
2. Antra programa tai serveris, kuris priima visų kliento programų siunčiamus duomenis ir juos atitinkamai apdorojęs saugo.
3. Trečia programa skirta duomenų analizei. Ji sugrupuoja duomenis, parodo kritines vietas, atskleidžia vartojamumo spragas ir pateikia vizualiai aiškia ataskaitą.

3.2.3. Vartotojo charakteristikos

Vartotojai - tai Flex4 technologijomis besinaudojantys programų kūrėjai, kuriantys vidutinės apimties ar stambius projektus.

3.2.4. Vartotojo problemos

Karkasas leidžia aptikti vartotojo kuriamos programinės įrangos vartojamumo spragas, kurias be tinkamo įrankio aptikti yra sudėtinga. Tikslinis karkaso vartotojas turėtų naudotis Flex4 technologija ir siekti patogaus ir paprasto savo sistemos vartojamumo.

3.2.5. Bendri apribojimai

Karkasas negali pastebimai kenkti testuojamo produkto greitaeigai bei turi būti lengvai pritaikomas individualiems projektams.

3.3. Tikslas

Projekto tikslas yra sukurti karkasą, kuris leistu nuotoliniu būdu registruoti vartotojų atliekamus veiksmus flex4 technologija sukurtuose projektuose, juos analizuoti ir aptikti vartotojo sąsajos spragas įgyvendinant naujai siūlomą vartojamumo testavimo bei vertinimo metodą.

3.4. Architektūros pasirinkimas

Bus kuriamos trys atskiros programos:

1. Kliento karkasas, sukurtas naudojant flex4 technologijas, integruojamas pridedant swc biblioteką bei prirašant reikiamus kreipinius.
2. Serveris, priimantis kliento karkaso siunčiamus pranešimus apie jo atliekamus veiksmus, Parašytas su ruby, nes nereikalauja didelės greیتaveikos ir sudėtingų užduočių sprendimo. Serveris skirtas tik veiksmų registravimui bei eksportavimui.
3. Analizės programa, grupuojanti rezultatus ir atvaizduojanti surinktą informaciją pagal pasirinktus kriterijus, išryškinant testuojamos sistemos trūkumus, kuriama naudojant flex4 technologiją.

3.4.1. Sistemos paskirtis

Vartojamumo tyrimas nėra pati sunkiausia testavimo dalis, bet labai dažnai nepelnytai nuvertinama. Kaip vėliau suprasite, neatliktas arba neteisingai atliktas vartojamumo tyrimas gali reikšti dvigubai ar net daugiau kartų mažesnę pelną. Šiai sričiai analizuoti yra įkurta speciali asociacija, sukurta daugybė įrankių, parašyta begalės straipsnių, bet sėkmės receptas lieka nežinomas. Trumpai tariant, atlikus tobulą vartojamumo tyrimą, bei pataisius sistemą, ji turėtų būti tokia: Vartotojui malonu žiūrėti į visus sistemos langus; Vartotojas akimirksniu randa tą sistemos dalį, kurios ieško; Pasinaudojus sistema, išlieka tik teigiamos emocijos, bei norisi naudotis ir kitą kartą; Įsigyti prekes ar paslaugas yra paprasta (jokių nereikalingų lentelių, mygtukų, laukų, kuriuos reikia užpildyti). Pasiiekti tokį sistemos lygį yra ganėtinai sunku dėl skirtingų vartotojų savybių (pvz.: Europiečiai rašo iš kairės į dešinę ir pagrindinio mygtuko

ieškos kairėje pusėje, arabai rašo iš dešinės į kairę ir mygtuko ieškos dešinėje.) Taigi, teisingai atliktas vartojamumo tyrimas gali reikšti įspūdingą metinio pelno prieaugį. [27]

Dažnai testuotojai tiesiog randa pirmą nemokamą arba pigų įrankį, pabando maksimaliai jį pritaikyti savo testuojamam produktui ir galvoja, kad to užteks, tačiau kiekvieno testuotojo poreikiai yra individualūs ir jiems patenkinti gali reikėti labai specifinio įrankio ar metodo. Todėl šis naujas karkasas ir yra kuriamas.

3.4.2. Projekto kūrimo pagrindimas

Norint pilnai iširti programos, skirtos tarptautinei rinkai, vartojamumą, reikia išbandyti ją su skirtingų kultūrų atstovais. Kadangi dažniausiai nėra galimybių tiesiogiai bendrauti su vartotojais iš viso pasaulio, reikalingas karkasas, nuotoliniam vartotojų veiksmų registravimui. Tuo labiau siūlomas naujas vartojamumo testavimo metodas neturi jokio įrankio vartojamumo testavimui, todėl buvo sukurtas prototipas, kuriuo bus galima patikrinti siūlomo metodo veiksmingumą.

Sukurtas karkasas ne tik registruoja vartotojo veiksmus, bet ir leidžia juos analizuoti. Tokiu būdu, galima aptikti tam tikrojo pasaulio vietoje būdingų vartotojo sąsajos trūkumų pagal siūlomą metodą, juos pataisyti, ir iširti naujos versijos efektyvumą. Karkasas veiks flex4 technologijomis sukurtose programose.

3.4.3. Sistemos tikslai

Sukurtas karkasas turi veikti Flex4 programavimo aplinkoje bei leisti nuotoliniu būdu testuoti sistemos vartojamumą registruojant vartotojo atliekamus veiksmus bei juos analizuojant. Karkasas registruos vartotojų atliekamus veiksmus, tokius kaip mygtuko paspaudimai, navigacija į programos dalis, sugaištas laikas ar pelės judėjimas. Taip pat karkasas leis registruotus duomenis analizuoti, grupuoti, peržvelgti atskirus atvejus ir pan. Programų sistemą sudaro trys programos:

- Pirmą programą „išrašoma“ į testuojamo projekto kliento programą. Ji registruoja konfigūraciniame faile nurodytus žingsnius ir siunčia informaciją apie vartojamumą serveriui.
- Antra programa tai serveris, kuris priima visų kliento programų siunčiamus duomenis ir juos atitinkamai apdorojęs saugo.
- Trečia programa skirta duomenų analizei. Ji sugrupuoja duomenis, parodo kritines vietas,

atskleidžia vartojamumo spragas ir pateikia vizualiai aiškia ataskaitą.

3.5. Projekto išeiga

3.5.1. Atviri klausimai

Veikiant daug atskirų sistemos dalių gali tapti sudėtinga įgyvendinti tarpusavio sąsajų realizavimą bei komponentų integraciją. Nenumatytoms problemoms turi būti skiriamas didelis dėmesys.

3.5.2. Galimos problemos

3.5.2.a. Problemos diegimo aplinkai

Testavimo sistemos įdiegimas daryt labai nežymius pakeitimus diegimo aplinkai, tokius kaip nedidelis sugeneruoto *.swf failo apimties padidėjimas, nežymus greitaveikos sumažėjimas bei interneto srauto užimtumo pakilimas, tačiau šie pokyčiai yra tokie maži, jog vartotojas jų turėtų nepastebėti.

3.5.2.b. Įtaka jau instaliuotoms sistemoms

Sistema nedarys jokios išskirtinės įtakos jau instaliuotoms sistemoms.

3.5.2.c. Neigiamas vartotojų nusiteikimas

Vartotojai net nepastebės jog yra testuojami, o testuotojas turi būti suinteresuotas galimybe surinkti informaciją apie vartotojų atliekamus veiksmus.

3.5.2.d. Kliudantys diegimo aplinkos apribojimai

Kiekvienam atskiram projektui reikės atskiros integracijos, tačiau tai tik darbo jėgos ir laiko kaštai.

3.5.2.e. Galimos naujos sistemos sukeltos problemos

Pastebėjus, jog testavimo integracija sukėlė problemų, galima ją išjungti ir taisyti problemas, kol sistema, kuri buvo testuojama, funkcionuoja įprastai.

3.5.3. Pritaikymas

3.5.3.a. Reikalavimai esamų duomenų perkėlimui

Esamų duomenų nėra, todėl ir perkėlimas nereikalingas. Tačiau prireikus reikėtų tiesiog nukopijuoti duomenų bazės įrašus.

3.5.3.b. Reikalingas duomenų transformavimas perkeliant į naują sistemą

Esamų duomenų nėra, todėl ir perkėlimas nereikalingas. Norint perkelti duomenis į kitą sistemą, jokių transformacijų vykdyti nereikėtų. Pakoregavus testavimo sistemą ir sukūrus kelias jos versijas galima daryti migracijas, transformuojančias duomenis iš vienos sistemos versijos į kitą. Taip pat karkase yra versijos kontrolės dalis, kuri užtikrina, kad siunčiami duomenys yra generuojami teisingos versijos produkto.

3.5.4. Rizikos

3.5.4.a. Galimos sistemos ir kūrimo rizikos

Lentelė 3: Galimos sistemos ir kūrimo rizikos

Nr.	Rizikos faktorius	Tikimybė/Įtaka
1	Reikalavimų pasikeitimas	Vidutinė/Didelė
2	Bendradarbiaujančių sistemų sąsajos pasikeitimai	Vidutinė/Didelė
3	Pakartotinai naudojami komponentai su klaidom, prasta dokumentacija	Vidutinė/Vidutinė
4	Serga projekto dalyviai	Vidutinė/Maža

3.5.5. Vartotojų dokumentacija ir apmokymas

Kartu su sistema pateikiama dokumentacija dokumentų pavidalu:

- Integravimo vadovas (instrukcijos testavimo sistemai integruoti į testuojamą sistemą)
- Testavimo vadovas (instrukcijos konfigūracinio failo nustatymui, duomenų eksportavimui bei duomenų analizės priemonės valdymui)

3.5.6. Perspektyviniai reikalavimai

- Konfigūracinio failo generavimo priemonė
- Testavimo sistemos integravimo priemonė pagal nustatymus
- Atgalinio atsako galimybės įvedimas (vartotojų atsiliepimai)

3.6. Projekto techninė specifikacija

3.6.1. Dokumento paskirtis

Skyriuje pateikiamas išsamus architektūrinis kuriamos sistemos vaizdas. Jam pateikti naudojami keletas skirtingų architektūrinių vaizdų, kurie parodo skirtingus kuriamos sistemos architektūrinius aspektus. Šio skyriaus tikslas surinkti ir pateikti svarbius architektūrinius sprendimus, kurie buvo atlikti, projektuojant sistemą. Šis dokumentas tarnauja kaip bendravimo medžiaga tarp programinės įrangos architekto ir kitu komandos narių dėl architektūrinių sistemos kūrimo sprendimų.

Šis skyrius bus pagrindas sudarant sistemos detalią architektūrą, bei bus naudingas rašant sistemos kodą.

3.6.2. Apžvalga

Skyriuje aprašoma UT karkaso architektūra. Sistemos nefunkciniai reikalavimai ir apribojimai pateikiami poskyryje „Architektūros tikslai ir apribojimai“. Sistemos panaudojimo atvejai pateikiami poskyryje „Panaudojimo atvejų vaizdas“. Sistemos išskaidymas ir statinė struktūra pateikta poskyryje „Statinis vaizdas“. Sistemos procesai ir jų aprašymai pateikiami poskyryje „Dinaminis vaizdas“. Sistemos išdėstymas, ir techninė įranga, kurioje bus realizuota sistema, pateikiama poskyryje „Išdėstymo vaizdas“. Poskyryje „Kokybė“ aprašoma kaip architektūra įtakoja sistemos plečiamumą, pernešamumą, patikimumą ir pan.

3.6.3. Architektūros pateikimas

Dokumente sistemos architektūra pateikiama keliais vaizdais: panaudojimo atvejų (PA), statinis, dinaminis ir išdėstymo. Sistemos architektūra pateikta remiantis RUP (Rational Unified Process) rekomendacijomis. Sistemos specifikacija pateikta šiais vaizdais kuriems įgyvendinti reikia UML diagramų:

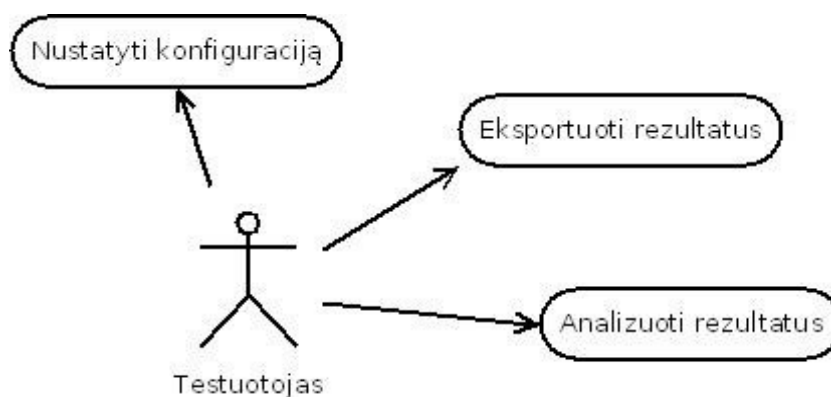
- Panaudojimo atvejų vaizdas (panaudojimo atvejų diagrama)
- Sistemos statinis vaizdas (paketai ir klasių diagramos)
- Sistemos dinaminis vaizdas (būsenų, veiklos, sekų, bendradarbiavimo diagramos)
- Išdėstymo vaizdas (išdėstymo diagrama)

3.6.4. Architektūros tikslai ir apribojimai

Architektūrinius sprendimus įtakoiantys reikalavimai:

- Sistema turi būti suprojektuota taip, kad ją galima būtų lengva išplėsti ar prijungi naujus modulius.
- Sistema neturi leisti neautorizuotiems vartotojams prie jos prisijungti.
- Sistema turi būti lengvai konfigūruojama.
- Sudarant sistemos architektūra, turi būti atsižvelgta į būtinas programos vykdymo charakteristikas, apibrėžtas reikalavimų specifikacijoje (skyrius „4. Reikalavimų specifikacija“).

3.6.5. Panaudojimo atvejų vaizdas



Paveikslas 2: Karkaso panaudojimo atvejų bendrinis vaizdas

Panaudojimo atvejis 1: Nustatyti konfigūraciją

Aktorius: Testuotojas.

Tikslas: Pasirinkti testuojamus modulius, bei testavimo kriterijus

Ryšiai su kitais PA: Nėra

Prieš-sąlygos: Nėra

Sužadinimo sąlyga: Vartotojas atsidaro konfigūracinį failą redaktoriuje ir išsaugo arba naudojami tam skirta

priemone

Po-sąlyga: konfigūracinis failas išsaugotas ir sistema paruošta testavimui

Pagrindinis scenarijus:

1. Vartotojas atsidaro konfigūracinį failą redaktoriuje arba tam skirtoje priemonėje.

Paveikslas 1: Sistemos panaudojimo atvejai

2. Baigus redaguoti išsaugojami pakeitimai.

3. Atnaujinama sistema ir įsigalioja nauji testavimo kriterijai.

Alternatyvūs scenarijai:

- Redaktorius uždaromas neišsaugojus pakeitimų
- Į formos laukus įvesti klaidingi duomenys (jei redaguota su priemone)

Panaudojimo atvejis 2: Eksportuoti rezultatus

Aktorius: Testuotojas.

Tikslas: Sugeneruoti tam tikro laikotarpio testavimo rezultatų duomenų ruošinį.

Ryšiai su kitais PA: Nėra

Prieš-sąlygos: Paruoštas konfigūracinis failas bei sukaupti testavimo rezultatai

Sužadinimo sąlyga: Vartotojas atsidaro konsolę ir įveda specialią komandą

Po-sąlyga: Sugeneruotas duomenų failas išsaugomas tam skirtame kataloge

Pagrindinis scenarijus:

1. Vartotojas atsidaro konsolę.
2. Įvedama komanda skirta duomenų eksportavimui.
3. Sistema sugeneruoja failą ir išsaugo jį tam skirtame kataloge.

Alternatyvūs scenarijai:

- Klaidingai įvesta komanda

Panaudojimo atvejis 3: Analizuoti rezultatus

Aktorius: Testuotojas.

Tikslas: Aptikti silpnąsias vartotojo sąsajos vietas.

Ryšiai su kitais PA: Nėra

Prieš-sąlygos: Testuotojas turi vieną ar kelis sugeneruotus duomenų ruošinius

Sužadinimo sąlyga: Testuotojas aktyvuoja analizavimo programą.

Po-sąlyga: Testuotojas mato norimas analizės vietas arba išsaugoja analizės ataskaitą.

Pagrindinis scenarijus:

1. Testuotojas atsidaro analizės programą.
2. Testuotojas pasirenka analizės kriterijus
3. Ekrane pateikiama analizė.
4. Vartotojas išsaugo analizės ataskaitą.

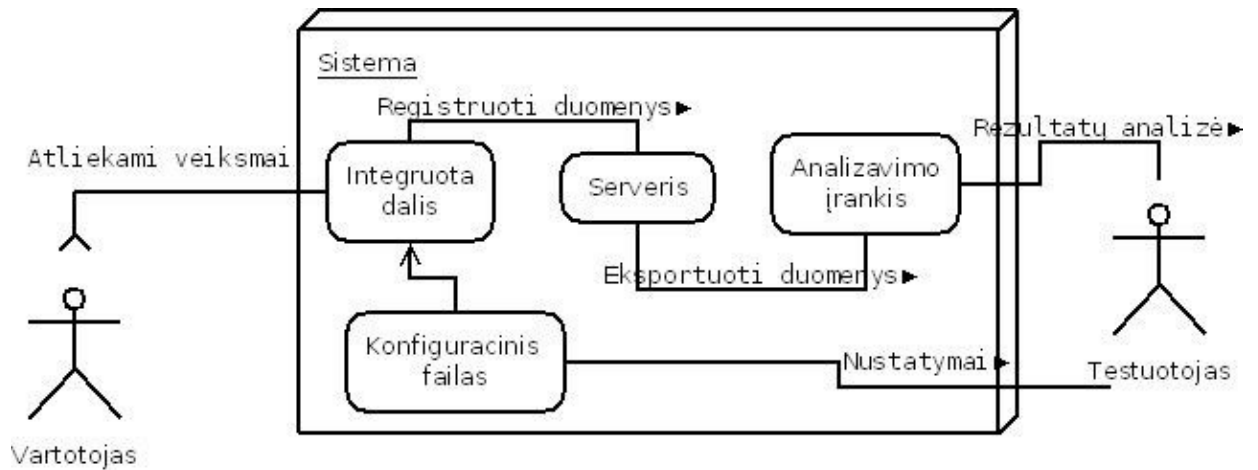
Alternatyvūs scenarijai:

- Vartotojas išjungia analizės programą prieš išsaugodamas ataskaitą.

3.7. Sistemos statinis vaizdas

Šis skyrius aprašo sistemos loginę sistemos struktūrą. Pateikia sistemos išskaidymą į paketus ir juos sudarančias klases.

3.7.1. Apžvalga

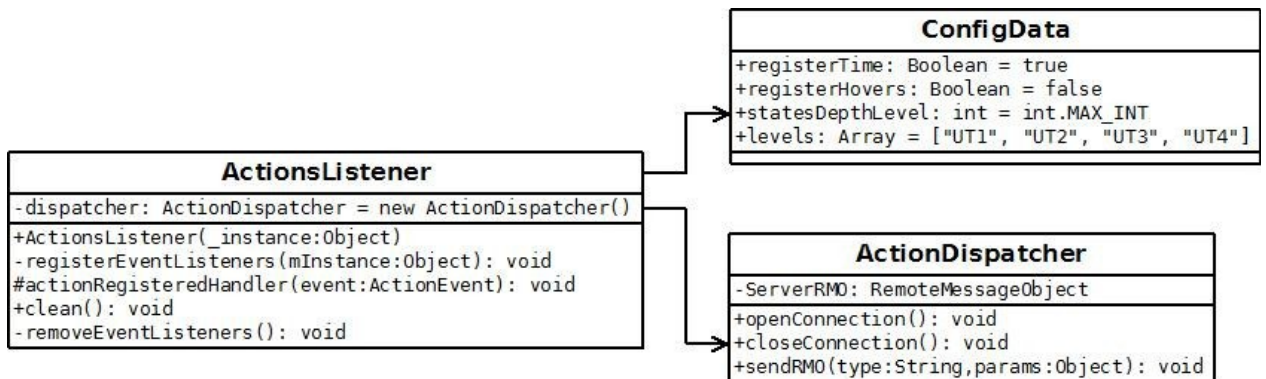


Paveikslas 3: Sistemos statinis vaizdas

3.7.2. Paketų detalizavimas

3.7.2.a. Integruota dalis

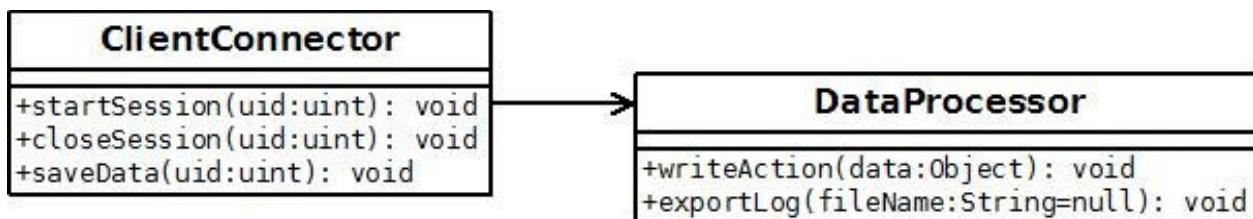
Šiame pakete pateikiamos į testuojamą produktą integruotos klasės.



Paveikslas 4: Integruotos dalies klasių diagrama

3.7.2.b. Serveris

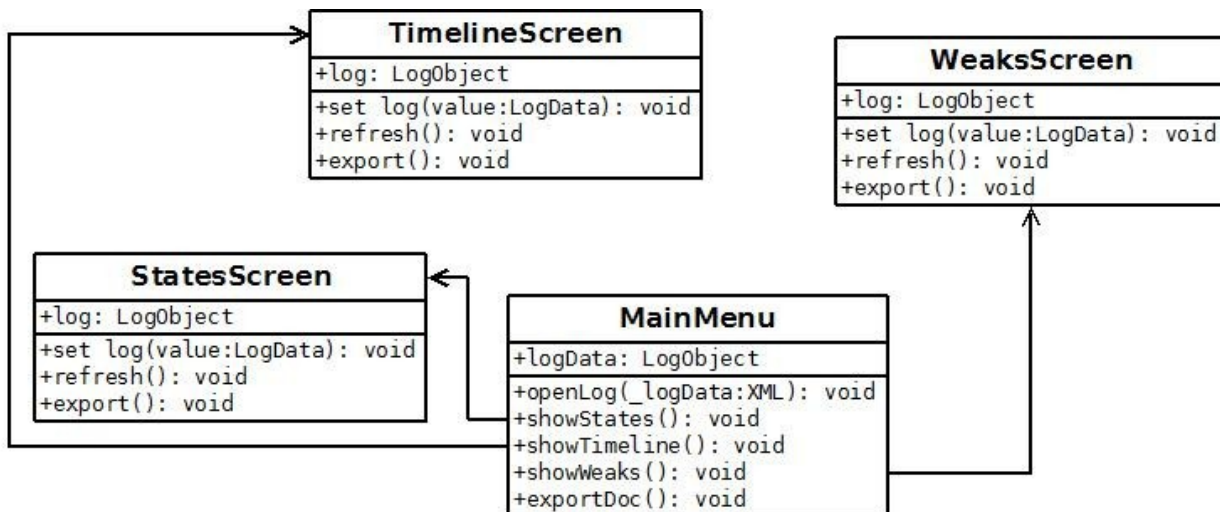
Šiame pakete yra serverio klasės skirtos surinktų analizės duomenų saugojimui.



Paveikslas 5: Serverio dalies klasių diagrama

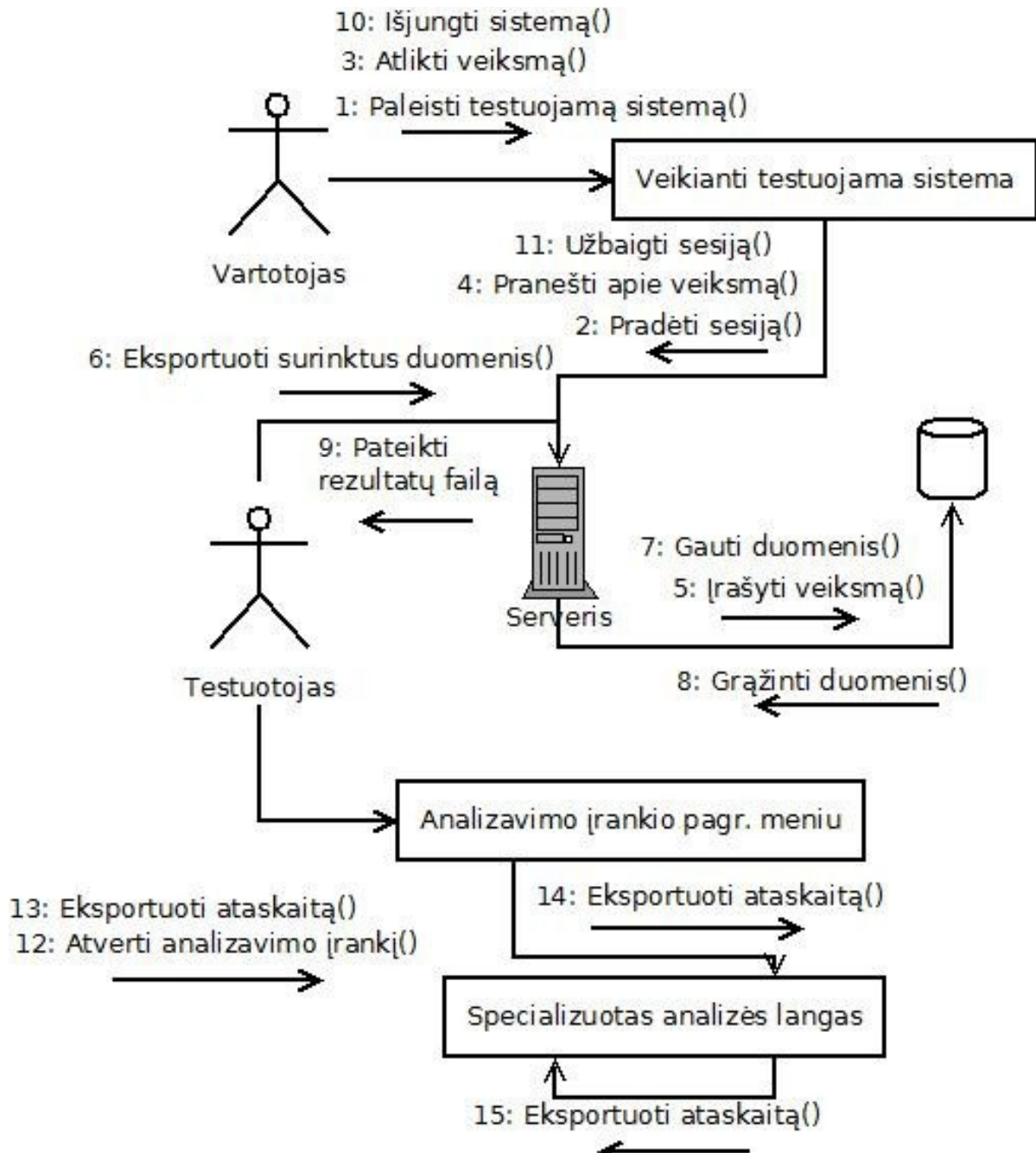
3.7.2.c. Analizavimo įrankis

Šiame pakete pateikiamos klasės esančios analizavimo įrankio pakete.



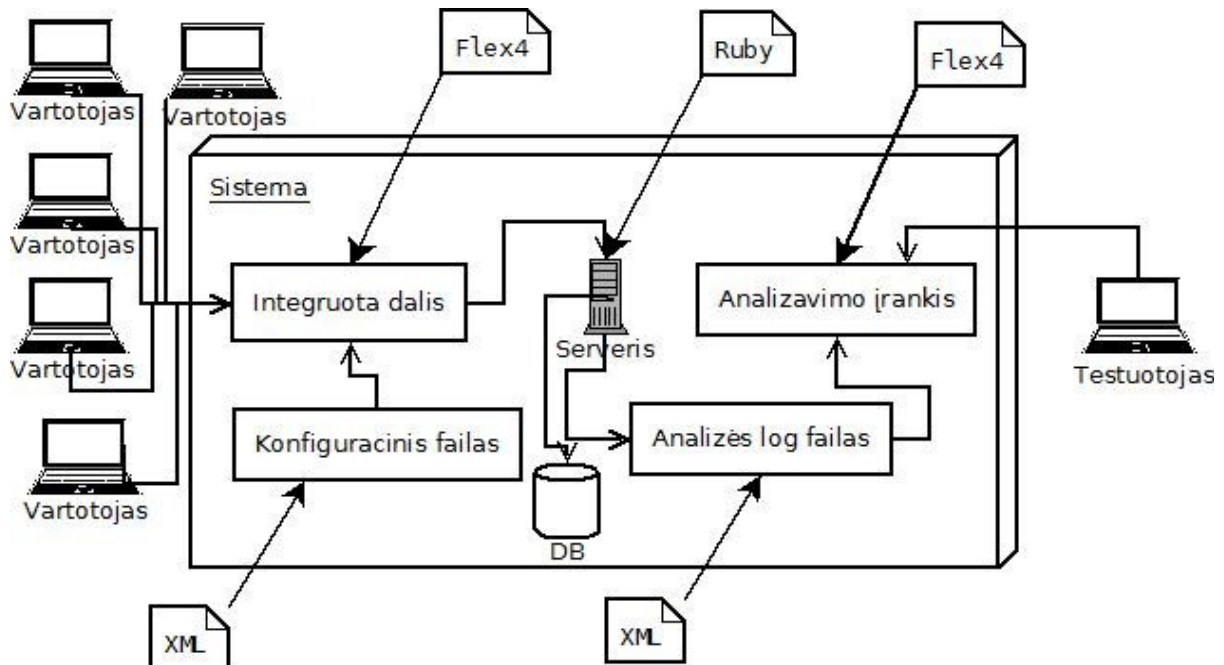
Paveikslas 6: Analizavimo įrankio klasių diagrama

3.7.3. Bendradarbiavimo diagrama



Paveikslas 7: Visos sistemos bendradarbiavimo diagrama

3.7.4. Išdėstymo vaizdas



Paveikslas 8: Visos sistemos išdėstymo diagrama

3.8. Kokybė

- Konfigūracinio failo pagalba bus galima pritaikyti karkasą skirtingų detalumo lygių testavimui.
- Testavimo galimybės lengvai plečiamos pridėdant naują registravimo dalį, konfigūracijos pasirinkimą ir analizės metodus.
- Galima tobulinti ataskaitos rengimo dalį profesionalioms ar specializuotoms ataskaitoms teikti.

4. Tyrimo dalis

4.1. Atliekamo tyrimo metodologija

Siūlomas vartojamumo testavimo metodas bei karkasas apima kai kurias metrikas, parinktas pagal įvairias ekspertų rekomendacijas, aptartas analitinėje dalyje. Vykiant tyrimus bus bandoma surinkti įvairias metrikas apie programų sistemos vartojamumą, pakeisti vartotojo sąsają ir surinkti naujas metrikas, taip bandant įrodyti metodo naudą bei karkaso renkamų metrikų teisingumą.

Šiame darbe yra nutarta vykdyti tokius tyrimus:

1. Formų pildymui sugaištamo laiko priklausomybė nuo užrašų ir įvedamo teksto laukų išdėstymo (horizontalus išdėstymas dviem stulpeliais ar vertikalus).
2. Vidutinio sesijos laiko priklausomybės nuo greitaveikos bei galimų atlikti programų sistemoje funkcijų skaičiaus tyrimas.
3. Programų sistemoje atsirandančių klaidų įtaka vartojamumui.
4. Teigiamų ir neigiamų vartotojų atsiliepimų santykio kaita po siūlomo vartojamumo testavimo metodo taikymo.
5. Vartojamumo gerinimo strategijos įtaka vartotojų išigyjamoms mokamoms paslaugoms.

4.2. Pasirinktų tyrimo metodų paskirtis

Lentelė 4: Pasirinktų tyrimo metodų aprašas ir paskirtis

Tyrimo numeris	Paskirtis
1	Ištirti formų pildymui sugaištamo laiko priklausomybę nuo užrašų ir įvedamo teksto laukų išdėstymo. UX Matters [Matteo Penzo 2006] tyrimų duomenimis formose paaiškinamuosius užrašus efektyviau yra dėti virš įvedamo teksto laukų negu kairėje pusėje. Norėdami įsitikinti šios rekomendacijos korektiškumu bei sukurto karkaso veiksmingumu ištirsime bent vieną formos pildymo langą prieš pakeitimus ir po

Tyrimo numeris	Paskirtis
	pakeitimų (su horizontaliu išdėstymu ir su vertikaliu).
2	Vidutinio sesijos laiko priklausomybės nuo greitaveikos bei galimų atlikti programų sistemoje funkcijų skaičiaus tyrimas turėtų parodyti bendrą poveikį vidutinės sesijos trukmei atliekant pakeitimus programų sistemai. Sesijos trukmė turėtų nesikeisti, nebent pakeitimai padidintų greitaveiką arba pridėtų papildomo funkcionalumo. Tyrimo metu stebėsime 4 mėnesių vartotojų sesijų trukmes ir pakeitimus programų sistemoje, taip sudarant ryšį tarp šių dviejų veiksnių.
3	Ištirti programų sistemoje atsirandančių klaidų įtaka vartojamumui. Šiuo tyrimu bus tikrinamas vartotojų savanoriškas klaidų pranešinėjimas programų sistemų kūrėjams bei kokia dalis vartotojų išjungia programų sistemą po klaidos ir rezultatai lyginami su automatiniu klaidų apdorojimu.
4	Ištirti teigiamų ir neigiamų vartotojų atsiliepimų santykio kaita po siūlomo vartojamumo testavimo metodo taikymo. Šio tyrimo metu nustatysime kokie siūlomo metodo aspektai teigiamai ar neigiamai paveikė vartotojų atsiliepimus. Taip pat bus nustatyta kokie veiksniai vartotojus labiausiai skatina reikšti savo nuomonę.
5	Nustatyti vartojamumo gerinimo strategijos įtaka vartotojų įsigyjamos mokamoms paslaugoms. Kadangi pagrindinė komercinės programų sistemos užduotis yra generuoti pelną, ištirti siūlomos strategijos įtaką mokėjimams būtina. Nustatysime ar taikydami pasiūlytą metodą galime pakelti programų sistemos pelningumą.

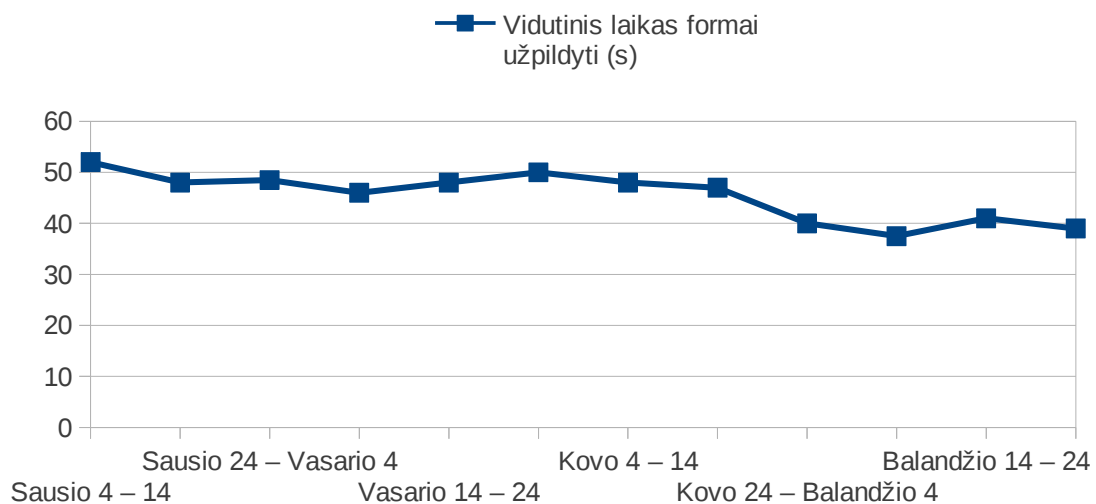
4.3. Tyrimo parametrų aprašas

- Tyrimas vyksta 4 mėnesius;
- Tyrimo metu siūlomas metodas yra taikomas internetinio strateginio žaidimo „Nebula44” vartojamumui testuoti;
- Tyrimo metu buvo stebimi nuo 200 (pirmą mėnesį) iki 500 (paskutinį mėnesį) aktyvių vartotojų;
- Tyrimo metu buvo pastoviai keičiamos įvairios programų sistemos funkcijos ir vartotojo sąsaja;

5. Eksperimentinė dalis

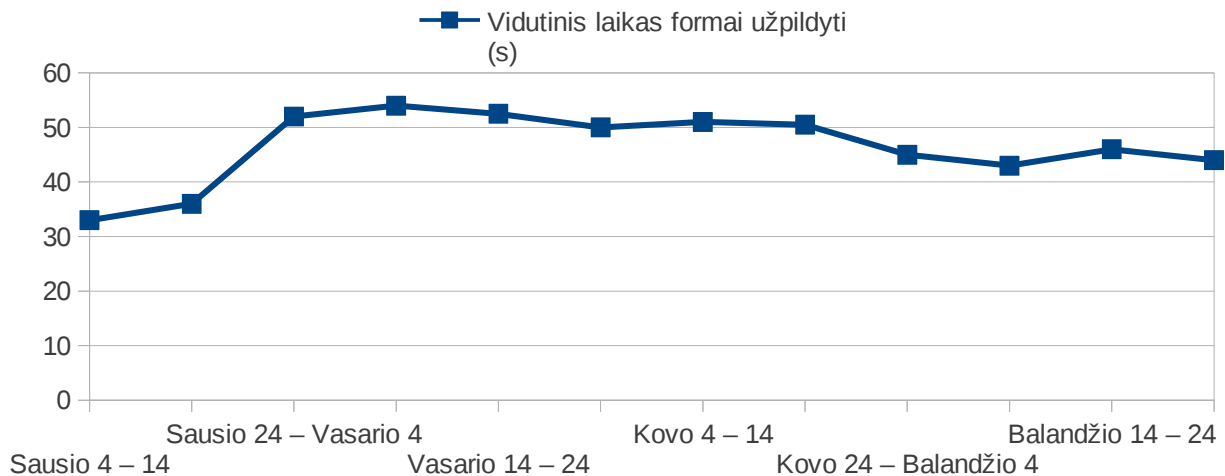
5.1. Tyrimas Nr. 1

Tirsime formų pildymui sugaištamo laiko priklausomybę nuo užrašų ir įvedamo teksto laukų išdėstymo. Testuojamame žaidime egzistavo dvi formos, kuriose buvo naudojamas horizontalus užrašų pozicionavimas šalia įvedamų laukų. Po pakeitimų formos buvo pertvarkytos į vertikalų vieno stulpelio išdėstymą. Pakeitimai formoms buvo įvykdyti kovo 23 dieną. Pateiksime vidutines vartotojo sugaištamo laiko formai užpildyti trukmes išskirstytas 10 dienų segmentais.



Paveikslas 9: Tyrimas 1, vidutinis laikas pasiūlymo turguje sukūrimo formos užpildymui

Pastebime nežymų sugaištamo laiko sumažėjimą nuo kovo 23 dienos pakeitimų. Pastaroji diagrama žymi sugaištamą laiką turgaus pasiūlymo kūrimo formoje testuojamame internetiniame žaidime. Ši forma yra pakankamai dažnai pildoma, tačiau turi palyginus nedaug elementų. Pakartosime tyrimą su daugiau elementų turinčia forma: vartotojo nustatymų keitimo langu.



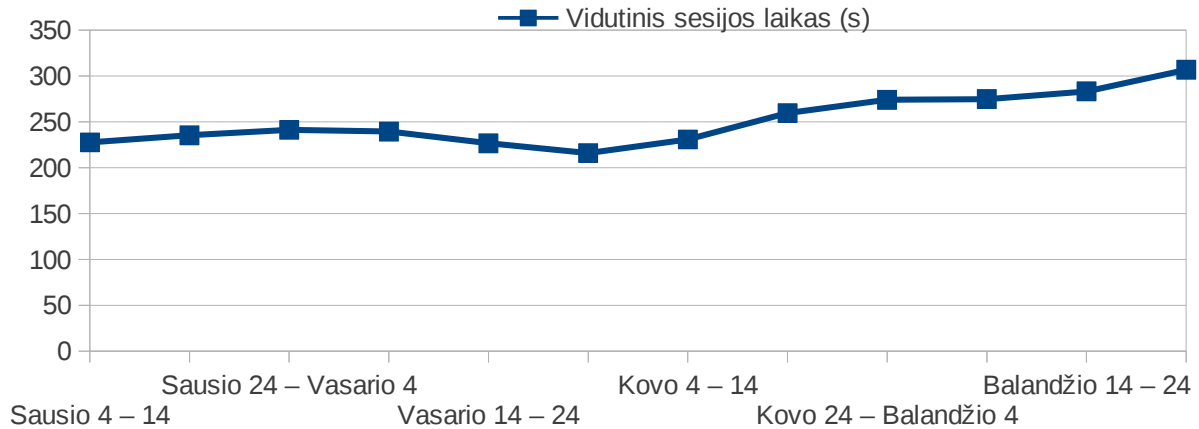
Paveikslas 10: Tyrimas 1, vidutinis laikas nustatymų keitimo formos užpildymui

Kaip ir praeito tyrimo metu, čia pastebime sugaištamo laiko sumažėjimą nuo kovo 24 dienos. Kyla klausimas: kodėl ir sausio pradžioje formos pildymui buvo sugaištama pastebimai mažiau laiko? Peržiūrėjus sausio pradžioje žaidimo versijos kontrolės programoje įrašytus atliktus darbus buvo aptikta, kad ši forma buvo praplėsta naujais elementais, kas ir lėmė didesnę sugaištamą laiką formos pildymui nuo sausio vidurio.

Atlikus tyrimą nustatėme, kad viena iš ekspertų rekomendacijų (dėti užrašus virš įvedamo teksto laukų) yra veiksminga. Stebint sugaištamo laiko tam tikroje būsenoje kaitą galime pastebėti atliktų pakeitimų programų sistemose įtaką atskirų būsenų vartojamumui sugaištamo laiko aspektu.

5.2. Tyrimas Nr. 2

Nustatysime kaip įvairūs veiksniai taikant siūlomą metodą veikia vidutinę sesijos trukmę. Visų pirmą reikėtų atkreipti dėmesį, kad tikslas nėra pakeisti (sumažinti ar padidinti) sesijos laiką, išskyrus tuos atvejus, kai siekiama padidinti greitaveiką. Sesija – tai vartotojo praleidžiamas laikas programų sistemoje vieno pasinaudojimo metu. Vartotojai atlieka veiksmus, kuriuos ir turi atlikti. Stebėsime vidutinės sesijos laiko kaitą šiuos 4 mėnesius ir bandysime išsiaiškinti įvairių pasikeitimų priežastis. Vėliau, norint pakeisti sesijos trukmę, bus galima orientuotis į aptiktas priežastis jas atkartojant ir taip redaguojant sesijos trukmę. Taip pat, viduryje tyrimo, bus atlikti testuojamo žaidimo greitaveikos optimizavimo darbai, tai turėtų atsispindėti ir vidutinėje sesijos trukmėje.



Paveikslas 11: Tyrimas 2, vidutinis sesijos laikas

Kaip pastebime sesijos laikas visus 4 mėnesius pastoviai didėjo, tai lėmė trys esminiai dalykai:

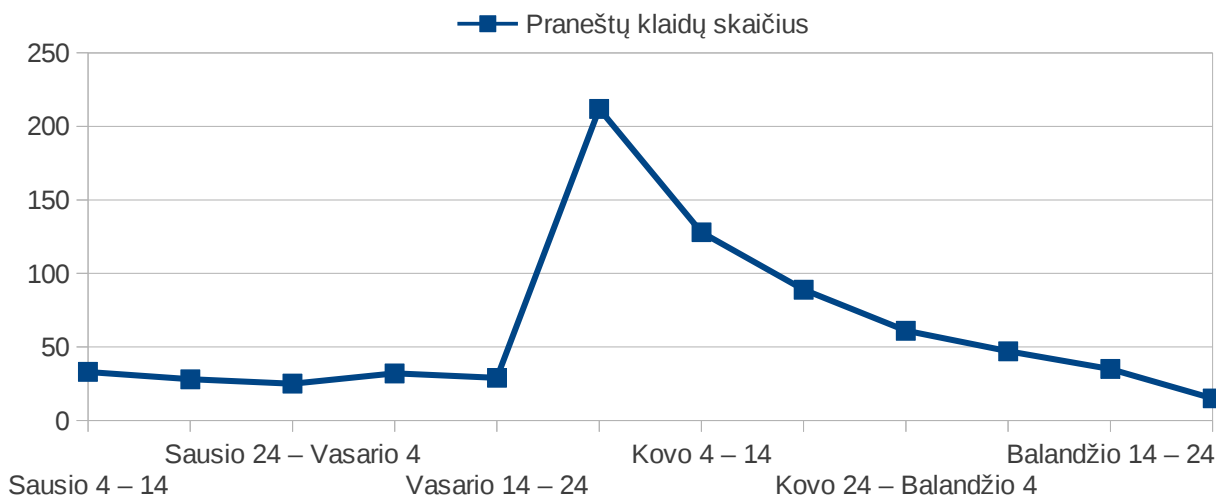
1. Žaidimo funkcionalumas yra pastoviai plečiamas pridedant naujų galimybių, todėl žaidėjai randa vis daugiau įdomesnių užsiėmimų vienos sesijos metu.
2. Kadangi žaidimo partija trunka ~6 mėnesius, o sausį buvo pradėta nauja partija, žaidėjai tolygiai plėtė savo virtualų turtą, o didesnę žaidimo kapitalą valdyti užtrunka daugiau laiko.
3. Nuo sausio pradžios iki balandžio pabaigos žaidėjų skaičius tolygiai augo 2,5 karto, kas lėmė didesnes žaidimo serverio apkrovas ir ilgesnį atsako laiką. Tai taip pat savo ruožtu didino vidutinį sesijos laiką.

Taip pat pastebime, kad diagramoje aiškiai matomas greitaveikos optimizacijos poveikis vidutinei sesijos trukmei. Kaip ir tikėtasi vasario viduryje atlikta optimizacija sutrumpino vidutinę sesiją, nes vartotojai sugaišdavo mažiau laiko atlikti toms pačioms užduotims.

5.3. Tyrimas Nr. 3

Ištirsime programų sistemoje atsirandančių klaidų įtaka vartojamumui. Nebuvo rasta jokių statistinių duomenų, kiek procentų vartotojų paprastai praneša apie klaidą kai su ja susiduria. Svarbu padaryti gerą klaidų pranešimų sistemą, kad klaidos būtų efektyviai taisomos. Iki vasario pabaigos testuojamame žaidime nebuvo automatinio klaidų pranešimo, vartotojai susidūrę su klaidomis buvo prašomi parašyti informaciją apie klaidą specialioje atsiliepimų puslapio dalyje. Nuo vasario pabaigos į žaidimą buvo integruota automatine klaidų pranešimo sistema, kaip

rekomenduojama siūlomame metode. Žemiau pateikiami gautų klaidų prieš integraciją ir po kiekiai:

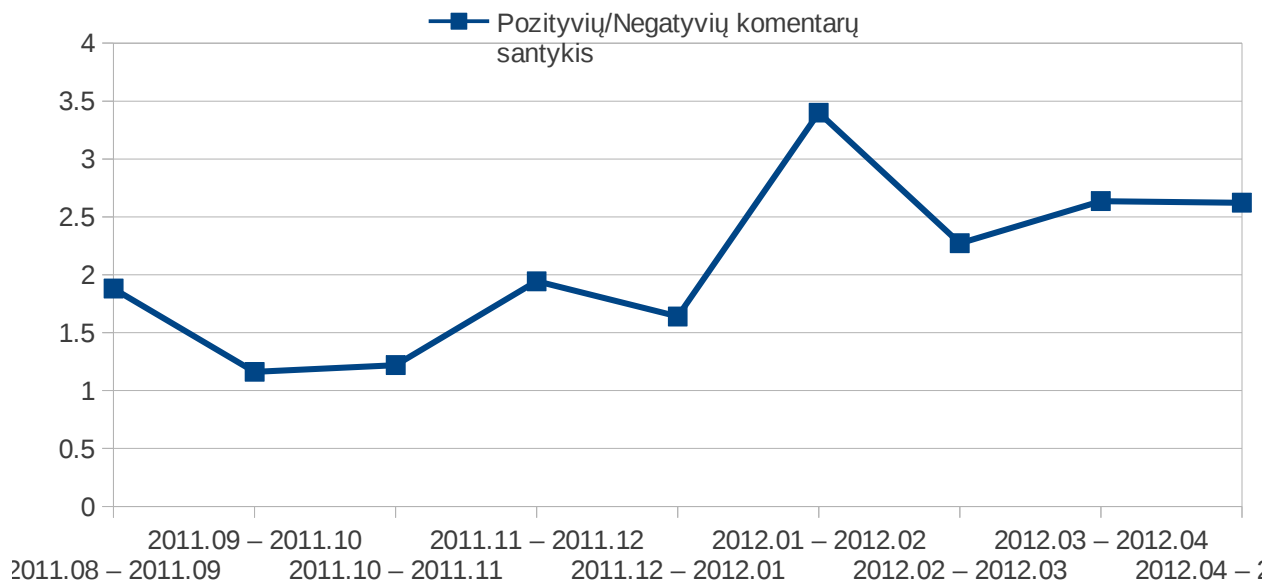


Paveikslas 12: Tyrimas 3, praneštų klaidų skaičius

Kaip pastebime vartotojai pranešdavo tik ~15% klaidų, nors klaidų pranešimo sistemą projektuotojai ir stengėsi padaryti kiek įmanoma „draugiškesnę“ vartotojui. Integravus automatinę klaidų pranešimo sistemą buvo aptikta gausybė anksčiau nepastebėtų ir nepraneštų klaidų, o su papildoma pateikta medžiaga klaidos buvo nesunkiai pašalintos. Mažesnis klaidų kiekis ne tik gerina vartojamumo kokybę, bet ir didina vartotojų pasitikėjimą programų sistema, kas savo ruožtu didina aktyvių vartotojų skaičių kuris dažnai yra tiesiogiai proporcingas uždirbamoms pajamoms.

5.4. Tyrimas Nr. 4

Tirsime teigiamų ir neigiamų vartotojų atsiliepimų santykio kaitą po siūlomo vartojamumo testavimo metodo taikymo. Komentarai vertinami pagal žmogiškąją emocinį faktorių. Jei komentare išreiškiamas nepasitenkinimas arba pranešama nepageidaujama funkcija – jis laikomas neigiamu. Jei komentare padėkojama už gerą funkcionalumą, maloniai pranešamas pasiūlymas ar jaučiamos geros emocijos – komentaras laikomas pozityviu. Komentarai pradėti registruoti nuo 2011 metų rugpjūčio vidurio.



Paveikslas 13: Tyrimas 4, pozityvių/negatyvių komentarų santykis

Ištyrus pozityvių/negatyvių komentarų santyki pastebėjome, jog santykinai pozityvių komentarų buvo daugiau prieš taikant siūlomą vartojamumo testavimo metodą. Kad būtų lengviau paaiškinti kodėl taip yra, šalia pateikiame stulpelinę pozityvių ir negatyvių komentarų skaičiaus palyginimo diagramą:



Paveikslas 14: Tyrimas 4, pozityvių/negatyvių komentarų skaičiai

Peržvelgus komentarų skaičių ir santyki susidaro įspūdis, kad vartotojai yra linkę labiau komentuoti tada, kai nori išreikšti nepasitenkinimą. Iki 2012 metų pradžios (kol veikė testinė versija su įvairiomis klaidomis ir blogu funkcionavimu) vartotojų komentarų buvo žymiai daugiau: tiek negatyvių, tiek pozityvių. Po 2012 buvo išleista stabili versija, kurioje buvo

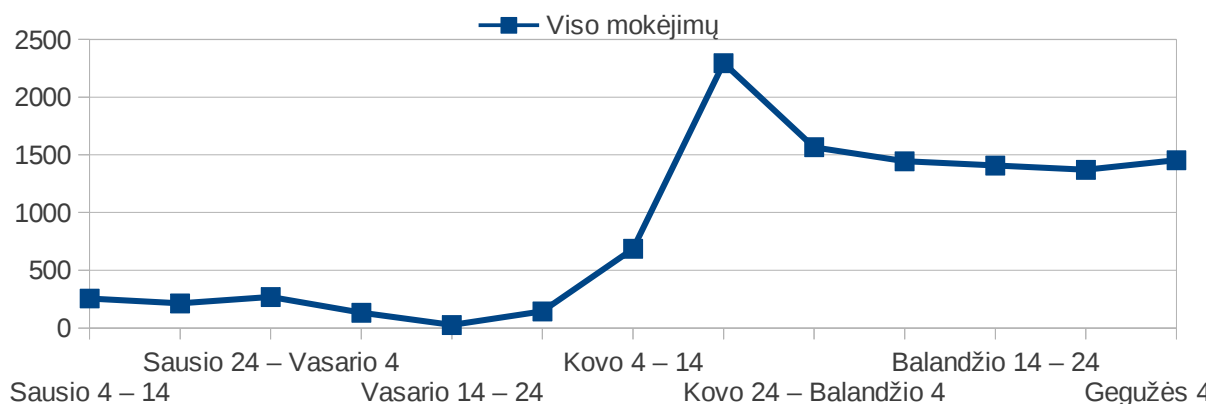
pataisyta dauguma testinėje versijoje rastų problemų. Vartotojai reiškė nuomonę dėl šio pokyčio ir savaime suprantama, kad didžioji dalis vartotojų nuomonių buvo pozityvios.

Pradėjus taikyti siūlomą metodą aktyvių vartotojų kiekis pradėjo augti, tačiau bendras komentarų skaičius pradėjo mažėti. Šį paradoksą lėmė tai, kad vartotojai, kurie sėkmingai naudojami produktui, nenori užsukti į atsiliepimų kambarį ir išreikšti savo nuomonę. Neskaitant bendro komentarų skaičiaus sumažėjimo, pozityvių/negatyvių komentarų santykis stabilizavosi. Norėdami suteikti vartotojams šiek tiek aktyvesnės veiklos bei pajavairinti žaidimo eigą, balandžio pabaigoje žaidimo kūrėjai suorganizavo konkursą. Konkursas į atsiliepimų puslapį atvedė pakankamai didelį kiekį žmonių, norinčių išreikšti savo nuomonę. Įdomiausia tai, kad net ir keturgubai padidėjus bendram komentarų skaičiui, jų santykis išliko toks pat, kas reiškia, jog buvo pasiektas stabilus vartojamumo lygis.

Siūloma šio rodiklio gerinimo strategija visų pirma būtų serverio stabilumo užtikrinimas, kadangi ištyrus neigiamų komentarų priežastis paaiškėjo, kad net daugiau nei 50% neigiamų komentarų kyla dėl serverio veikimo problemų (tokių kaip nenumatyta profilaktika, tam tikrų funkcijų sugedimas ir pan.). Taip pat būtų galima raginti vartotojus reikšti savo nuomonę atsiliepimų puslapyje, suteikiant jiems galimybę pasiekti geresnį statusą priklausomai nuo parašytų atsiliepimų skaičiaus (pvz.: iki 10 žinučių – naujokas, iki 50 žinučių – pažengęs, vėliau – aktyvus, ekspertas ir t.t.).

5.5. Tyrimas Nr. 5

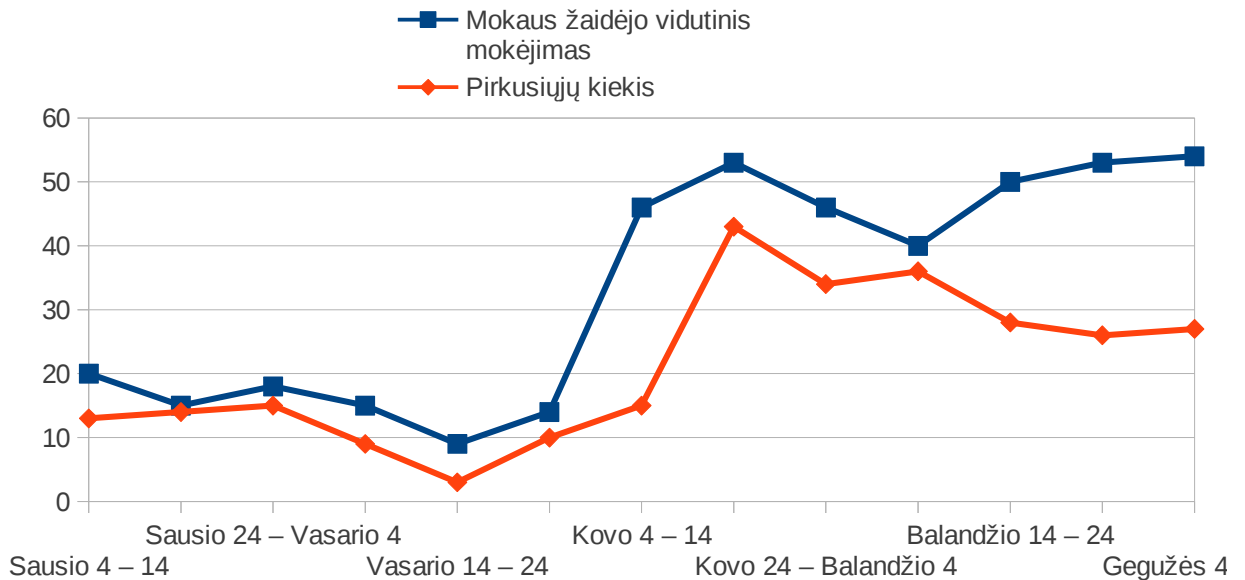
Tirsime vartojamumo gerinimo strategijos įtaka vartotojų įsigyjamos mokamos paslaugoms. Kadangi dažniausiai vartojamumo testavimo tikslas galų gale atsiremia į pelningumo didinimą, ši dalis yra itin svarbi įrodant siūlomo metodo veiksmingumą. Žemiau pateikiama mokamų paslaugų statistika metodo taikymo laikotarpyje.



Paveikslas 15: Tyrimas 5, bendra mokėjimų suma per keturis mėnesius

Pastebime, kad kovo pabaigoje stipriai išaugo mokėjimai, taip yra todėl, nes buvo atidaryta nauja (ne testinė) žaidimo partija, kurioje buvo pritaikyti visi iki šiol įgyvendinti siūlomo vartojamumo testavimo metodo etapai. Matomas staigus mokėjimų šuolis kovo mėnesį, o vėliau mokėjimai stabilizavosi. Iš viso, pritaikius metodą, pavyko pakelti pelningumą ~5 kartus.

Žemiau pateikiama šiek tiek išskaidyta statistika, kur bus galima peržvelgti metodo taikymo įtaką individualiam mokėjimui.



Paveikslas 16: Tyrimas 5, pirkusiųjų vartotojų kiekis ir vidutinis mokaus vartotojo mokėjimas

Matome, jog panašiai augo tiek mokių žaidėjų skaičius, tiek ir mokaus žaidėjo išleidžiama suma. Tai reiškia, kad sugebėjome padidinti ne tik aktyvių žaidėjų skaičių, bet ir žaidėjų pasitikėjimą žaidimu bei malonumą, kurį žaidėjas patiria žaisdamas. Surinkti duomenys yra pasiskirstę pakankamai netolygiai dėl šiek tiek mažo žaidėjų skaičiaus. Pritraukus daugiau žaidėjų duomenys turėtų tapti pastovesni, tačiau pagrindinė tendencija turėtų išlikti tokia pati.

6. Išvados

1. Atlikus analizę nebuvo aptikta įrankių skirtų Flex 4 technologijomis kurtų programų sistemų vartojamumo duomenų surinkimui ir analizei.
2. Išnagrinėjus literatūrą bei ekspertų rekomendacijas nustatyta, kad reikia sukurti apibendrintą vartojamumo analizės metodą. Darbe pasiūlytas naujas metodas, kuriame apjungiamos QUIM modelyje esančios metrikos, ilgametė ekspertų praktika bei įvairios kituose modeliuose naudojamos technologijos.
3. Sukurtas karkaso prototipas, kuris registruoja dalį metrikų įeinančių į pasiūlytą metodą. Atlikus eksperimentinius tyrimus nustatyta, kad:
 - 3.1. Įvertinus prototipo teikiamus analizės duomenis ir pritaikius ekspertų rekomendacijas buvo pagerinta vartojamumo kokybė - padidėjo vartotojų pozityvių atsiliepimų dalis, vartotojai sugaišta mažiau laiko elementarioms užduotims atlikti bei didesnė dalis vartotojų išlieka aktyvūs. Paaiškėjo, kad vartojamumo kokybė turi tiesioginį ryšį su bendru komentarų skaičiumi atsiliepimų puslapyje.
 - 3.2. Įvertinus prototipo teikiamus analizės duomenis ir pritaikius ekspertų rekomendacijas iki 5 kartų padidėjo testuojamos programų sistemos pelningumas, be to, 2,5 karto padidėjo aktyvių vartotojų skaičius.
 - 3.3. Metodo pritaikymas neturi pastebimos įtakos vidutinei sesijos trukmei, išskyrus tuos atvejus kai optimizuojama greitaveika arba elementarių užduočių atlikimo procesas.
 - 3.4. Pritaikius automatinį klaidų pranešimą gaunama apytiksliai 7 kartus daugiau bei detalesnių klaidų pranešimų nei prašant vartotojų pranešti apie klaidas atsiliepimų puslapyje.
4. Nors eksperimentai parodė sukurto karkaso naudą analizuojant programos vartojamumą, norint pilnai taikyti produktą praktikoje (metodą kartu su analizės programine įranga) karkasą reikėtų tobulinti bei plėsti įtraukiant naujas metrikas, tokias kaip paspaudžiami klaviatūros klavišai, pelės judėjimo kelias, šiluminiai pelės judėjimo žemėlapiai (angl. heat map) ir panašiai.

7. Rekomendacijos

Norintiems atlikti vartojamumo tyrimą savo projektuose, rekomenduojame:

- Išsiaiškinti siekiamus tikslus: ką norima atskleisti vartojamumo testavimo metu? Gal būt norima padidinti pelningumą, gal sutrumpinti sugaištamą laiką, gal kaip tik ilgiau išlaikyti vartotojus?
- Pasirinkti tinkamą vartojamumo testavimo modelį arba atskirus metodus priklausomai nuo išsikeltų tikslų bei testuojamo produkto.
- Neapsiriboti vienu metodu ir taikyti kaip įmanoma daugiau ir įvairesnių testavimo technologijų.
- Klaidų registravimui naudoti automatinį klaidų pranešimo metodą. Kaip paaiškėjo vartotojai nėra nusiteikę pranešinėti programų sistemoje įvykusias klaidas.
- Po kiekvieno didesnio pakeitimo produkte įvertinti vartojamumą ir palyginti su buvusiais įvertinimais.
- Vadovautis ekspertų pateikiamomis rekomendacijomis ir strategijomis vartojamumui gerinti. Kartais užtenka paskaityti vieną atliktą mokslinį tyrimą apie paprastą vartotojo sąsajos elementą (pvz.: vartotojo pildomą formą), kad suprastum kaip galima lengvai padidinti vartojamumo kokybę.

8. Literatūros sąrašas

- [1] International Organization for Standardization, 1998. ISO 9241-11, Ergonomic Requirements for Office Work with Visual Display Terminals (VDTs), Part 11: Guidance on Usability, Geneva;
- [2] International Organization for Standardization/International Electrotechnical Commission, 2001. ISO/IEC 9126-1 Standard, Software Engineering, Product Quality, Part 1: Quality Model, Geneva;
- [3] International Organization for Standardization/International Electrotechnical Commission, 2001. ISO/IEC 9126-4, Software Engineering, Product Quality, Part 4: Quality in Use Metrics, Geneva;
- [4] Bevan, N. and Schoeffel, R. 2001. A proposed standard for consumer product usability, Proceedings of 1st International Conference on Universal Access in Human Computer Interaction, New Orleans, LA, pp. 557–561.
- [5] Ahmed Seffah, Mohammad Donyaee, Rex B. Kline, Harkirat K. Padda , 2006. Usability measurement and metrics: A consolidated model , Springer Science + Business Media, Inc.
- [6] Macleod, M., Bowden, R., Bevan, N. and Curson, I., 1997. The MUSiC performance method, Behaviour and Information Technology 16: 279-293.
- [7] Ahuja, V. 2000. Building trust in electronic commerce, IT Professional 2: 61–63.
- [8] Atif, Y. 2002. Building trust in e-commerce, IEEE Internet Computing 6: 18–24.
- [9] Caldwell, B., Chisholm, W., Vanderheiden, G., and White, J. (Eds.), 2004. Web Content Accessibility Guidelines 2.0, W3C Working Draft 30 July 2004, World Wide Web Consortium. Prieiga internete: <http://www.w3.org/TR/2004/WD-WCAG20-20040730/>;
- [10] Jeff Sauro, 2011. The four corners of usability measurement. Prieiga internete: <http://www.measuringusability.com/blog/four-corners.php>;
- [11] „Concept feedback” įvertinimų tinklapis, prieiga internete <http://www.conceptfeedback.com>;

- [12] Dr. Charles B. Kreitzberg vartojamumo tyrimo apžvalga, prieiga internete <http://msdn.microsoft.com/en-us/magazine/dd920305.aspx>;
- [13] „Chalkmark” testų kūrimo tinklapis, prieiga internete <http://www.optimalworkshop.com/chalkmark.htm>;
- [14] Paspaudimų vietas fiksuojančios priemonės tinklapis, prieiga internete <http://www.labsmedia.com/clickheat/index.html>;
- [15] Vartojamumo filmavimo įrankio tinklapis, prieiga internete http://www.clicktale.com/default_d.aspx;
- [16] Vartotojų veiksmų registravimo priemonės tinklapis, prieiga internete <http://www.clixpy.com>;
- [17] Testuotojų ir testuojamųjų komunikavimo priemonės tinklapis, prieiga internete <http://www.ethnio.com>;
- [18] Automatinio vartojamumo testavimo priemonės tinklapis, prieiga internete <http://www.feng-gui.com>;
- [19] Vartojamumo testas per penkias sekundes, prieiga internete <http://fivesecondtest.com>;
- [20] Internetinių puslapių vartojamumo testavimo priemonės tinklapis, prieiga internete <http://www.loop11.com>;
- [21] Testuotojų ir norinčių išsitestuoti vartojamumą bendradarbiavimo tinklapis, prieiga internete <https://www.mturk.com/mturk/welcome>;
- [22] Nielsen, Jakob, and Landauer, Thomas K.: "A mathematical model of the finding of usability problems," *Proceedings of ACM INTERCHI'93 Conference* (Amsterdam, The Netherlands, 24-29 April 1993), pp. 206-213.
- [23] Thompson, K.E., Rozanski, E.P. and Haake, A.R. (2004). "Here, There, Anywhere: Remote Usability Testing That Works", in *Conference on Information Technology Education Proceedings*, pp. 132-137, prieiga internete <http://portal.acm.org/citation.cfm?id=1029567&dl=&coll=&CFID=15151515&CFTOKEN=618> 4618 ;
- [24] Bolt, N. (2006). *Guide to Remote Usability Testing, OK/Cancel*, prieiga internete <http://www.ok-cancel.com/archives/article/2006/07/guide-to-remote-usability-testing.html> ;

- [25] Dmitry Fadeyev, 2009. [10 Useful Usability Findings and Guidelines](http://uxdesign.smashingmagazine.com/2009/09/24/10-useful-usability-findings-and-guidelines/), prieiga internete <http://uxdesign.smashingmagazine.com/2009/09/24/10-useful-usability-findings-and-guidelines/>;
- [26] Stanford Research Institute, 1965. Computer-aided display control, prieiga internete http://www.archive.org/stream/nasa_techdoc_19660020914/19660020914#page/n0/mode/2up
- [27] W Craig Tomlin asmeninis tinklapis, prieiga internete <http://www.wcraigtomlin.com>;

9. Terminų ir santrumpų žodynas

- [1] RIA (Rich Internet Application) – tai yra internetinė aplikacija, kuri turi labai daug panašių savybių kaip ir normali „darbastalio“ aplikacija, tačiau veikianti tiesiai iš internetinės naršyklės.
- [2] Nebula44 – Tai internetinis strateginis naršyklinis žaidimas, kurio vartojamumą buvo bandoma gerinti eksperimentinio tyrimo metu.
- [3] QUIM (angl. Quality in Use Integrated Measurement) - tai bendrinis hierarchinis modelis, kuriame yra apjungti įvairūs vartojamumo vertinimo ir testavimo metodai.
- [4] SUS (angl. System usability scale) – standartinis 10 klausimų klausimynas programinės įrangos vartojamumui vertinti.
- [5] SUPR-Q (angl. The Standardized Universal Percentile Rank Questionnaire) – standartinis 13 klausimų klausimynas tinklalapių vartojamumui vertinti.
- [6] Flex4 - tai programinės įrangos kūrimo komplektas (SDK), kurį išleido „Adobe Systems“, skirtas kurti turtinga internetines aplikacijas (Rich Internet Applications), pagrįsta „Adobe Flash“ pagrindu.
- [7] UI – (User Interface) Vartotojo sąsaja
- [8] UT – (Usability testing) Vartojamumo testavimas
- [9] SDK (software development kit) – tai paprasčiausiai įvairių konkrečios kuriamos programinės įrangos kūrimo įrankių rinkinys.
- [10] Adobe Flash Builder – tai yra integruota programinės įrangos kūrimo aplinka (IDE), sukurta „Eclipse“ platformos pagrindu pagreitinanti turtingų internetinių aplikacijų kūrimą.
IntelliJ IDEA – tai yra integruota programinės įrangos kūrimo aplinka (IDE).

10. Priedai

10.1. Trumpa vartotojo dokumentacija

10.1.1. Sistemos funkcinis aprašymas

10.1.1.a. Sistemos paskirtis

Karkasas registruoja vartotojų atliekamus veiksmus, tokius kaip mygtuko paspaudimai, navigacija į programos dalis, sugaištas laikas ar pelės judėjimas. Taip pat karkasas suteikia galimybę registruotus duomenis analizuoti, grupuoti, peržvelgti atskirus atvejus ir pan. Sistemos paskirtis yra panaudojant užregistruotus duomenis suteikti galimybes pastebėti vartojamumo spragas ir jas taisyti.

10.1.2. Sistemos galimybės

Karkaso skirto vartotojų veiksams registruoti ir analizuoti galimybės:

- Sistema veikia Windows, Linux, macOS operacinėse sistemose.
- Sistema išsaugo vartotojų atliktus veiksmus duomenų bazėje.
- Sistema veikia naudodama kliento - serverio architektūrą.
- Sistema yra integruojama į kitus projektus.
- Sistema suteikia galimybes analizuoti vartotojų veiksmus.

10.1.3. Vartotojo atmintinė

10.1.3.a. Kas yra karkasas skirtas vartotojų veiksams registruoti ir analizuoti?

Karkasas registruoja vartotojų atliekamus veiksmus, tokius kaip mygtuko paspaudimai, navigacija į programos dalis, sugaištas laikas ir t.t.. Taip pat karkasas leidžia registruotus duomenis analizuoti, grupuoti, peržvelgti atskirus atvejus ir taip aptikti vartojamumo spragas.

10.1.3.b. Ko man reikia, norint naudoti šią sistemą?

Serveriui reikia Linux arba Windows XP (arba vėlesnės versijos) operacinės sistemos, analizavimo programai reikia Flash Player naršyklės įskiepio. Sėkmingam sistemos naudojimui visų pirma reikia integruoti registravimo dalį į testuojamą projektą. Karkasas gali testuoti tik Flex 4 projektus.

10.1.3.c. Kaip naudotis šia sistema?

Reikia integruoti veiksmų registravimo dalį į testuojamą projektą, serveryje paleisti serverio programinę dalį, tada leisti testuojamo projekto vartotojams naudotis programa. Testuojama dalis automatiškai siunčia duomenis serveriui ir saugo vartotojų atliktus veiksmus duomenų bazėje.

Nusprendus, kad jau yra sukauptas pakankamas duomenų kiekis reikia eksportuoti surinktus duomenis į analizavimui skirtą formatą ir atidaryti jį su analizavimo įrankiu. Analizavimo įrankio pagalba galima sužinoti kiekvienos dalies vartojamumo savybes, reikėtų atkreipti dėmesį į tas dalis, kur vartotojai sugaišta daugiausiai laiko bei neatlieka numatytu veiksmų.