

KAUNO TECHNOLOGIJOS UNIVERSITETAS  
INFORMATIKOS FAKULTETAS  
MULTIMEDIJOS INŽINERIJOS KATEDRA

**Aistė Ivonytė**

**COSMIC METODO TAIKYMAS DARBO IMLUMO  
ĮVERTINIMUI AGILE PROJEKTUOSE**

Magistro darbas

Darbo vadovas:

dr. Arūnas Tomkevičius

**KAUNAS, 2014**

**KAUNO TECHNOLOGIJOS UNIVERSITETAS**  
**INFORMATIKOS FAKULTETAS**  
**MULTIMEDIJOS INŽINERIJOS KATEDRA**

**Aistė Ivonytė**

**COSMIC METODO TAIKYMAS DARBO ĮMĖLUMO**  
**ĮVERTINIMUI AGILE PROJEKTUOSE**

Magistro darbas

Darbo vadovas:

dr. Arūnas Tomkevičius

2014-05-23

Recenzentas:

prof. K. Motiejūnas

2014-05-23

Atliko:

IFM-2/1 gr. studentė

Aistė Ivonytė

2014-05-23

**KAUNAS, 2014**

# AUTORIŲ GARANTINIS RAŠTAS

## DĖL PATEIKIAMO KŪRINIO

20.. - ..... - ..... d.  
Kaunas

**Autoriai,** \_\_\_\_\_ **Aistė Ivonytė** \_\_\_\_\_  
(vardas, pavardė)

\_\_\_\_\_ ,  
patvirtina, kad Kauno technologijos universitetui pateiktas baigiamasis magistro darbas (toliau vadinama – Kūrinys) \_\_\_\_\_ COSMIC metodo taikymas darbo imlumo įvertinimui Agile projektuose \_\_\_\_\_  
(kūrinio pavadinimas)

pagal Lietuvos Respublikos autorių ir gretutinių teisių įstatymą yra originalus ir užtikrina, kad

- 1) jį sukūrė ir parašė Kūrinyje įvardyti autoriai;
- 2) Kūrinys nėra ir nebus įteiktas kitoms institucijoms (universitetams) (tiek lietuvių, tiek užsienio kalba);
- 3) Kūrinyje nėra teiginių, neatitinkančių tikrovės, ar medžiagos, kuri galėtų pažeisti kito fizinio ar juridinio asmens intelektualinės nuosavybės teises, leidėjų bei finansuotojų reikalavimus ir sąlygas;
- 4) visi Kūrinyje naudojami šaltiniai yra cituojami (su nuoroda į pirminį šaltinį ir autorių);
- 5) neprieštarauja dėl Kūrinio platinimo visomis oficialiomis sklaidos priemonėmis.
- 6) atlygins Kauno technologijos universitetui ir tretiesiems asmenims žalą ir nuostolius, atsiradusius dėl pažeidimų, susijusių su aukščiau išvardintų Autorių garantijų nesilaikymu;
- 7) Autoriai už šiame rašte pateiktos informacijos teisingumą atsako Lietuvos Respublikos įstatymų nustatyta tvarka.

**Autoriai**

_____	Aistė Ivonytė _____	_____
	(vardas, pavardė)	(parašas)
_____		_____
	(vardas, pavardė)	(parašas)
_____		_____
	(vardas, pavardė)	(parašas)
_____		_____
	(vardas, pavardė)	(parašas)

## SANTRAUKA

COSMIC metodas apibrėžia principus, taisykles ir standartinio funkcinio dydžio matavimo procesą. Šio metodo realizacija Agile projektuose, padės prognozuoti konkrečios užduoties atlikimo laiką ankstyvoje planavimo stadijoje. Netikslus darbo imlumo įvertinimas planavimo stadijoje gali tapti rimta problema, dėl ko gali nepavykti įgyvendinti projekto, arba jo dalies, laiku. Vartotojų scenarijai yra vertinami SP, kurie nėra verčiami į laikinę skalę, kas dar labiau apsunkina tikslų darbo imlumo įvertinimą pradinėje projekto ir užduočių planavimo stadijoje. Šiame darbe atliekama detali COSMIC metodo analizė, apžvelgiami esami įrankiai, skirti Agile projektams. Įsitikinama, kad nėra sukurto įrankio, leidžiančio prognozuoti vartotojų scenarijų atlikimo laiką, sujungus COSMIC su istoriniais darbų atlikimo duomenimis.

Darbe iškeltų tikslų pasiekimui buvo atliktas trijų dalių eksperimentinis tyrimas kurio metu paaiškėjo, kad COSMIC metodo apjungimas su istoriniais darbų atlikimo duomenimis gali pateikti panašią ar net geresnę prognozę, lyginant su programuotojų komandos įvertinimu. Taip pat pateikiamos idėjos įrankio korekcijai, jos realizuojamos ir kartojamas eksperimentas. Aptariami gauti rezultatai ir pateikiamos išvados.

### *Raktiniai žodžiai*

COSMIC metodas, Agile projektai, darbo imlumo įvertinimas, laiko įvertis, sudėtingumo įvertinimas, funkciniai taškai, istoriniai duomenys, produkto šeimininkas, faktinis vartotojų scenarijų atlikimo laikas, Visual Studio 2013, C# programavimo kalba (WPF modelis), SQLite, TFS 2013 ir TFS Scrum šablonas.

## SUMMARY

COSMIC method defines principles, rules and the process of standard functional size measurement. Method implementation in Agile projects will help forecasting the completion time for a specific task in the early planning stages. Inaccurate task capacity estimation in the planning stage can become a serious problem which in turn may lead to not being able to complete parts of the assignment or even the whole project. User stories are being evaluated as SP which are not converted into a timescale which further complicates accurate job receptivity in the early project and task planning stage. Within this paper a detailed COSMIC method analysis is being performed as well as a review of existing Agile project tools. Also making sure that there are no other tools which allow the forecast of user story completion time by merging COSMIC with task completion time history log. To complete the goals of this paper a three part experimental survey took place which determined that the consolidation of the COSMIC method with the task completion time history may result in a similar or even more accurate forecast as compared to the forecast of a team of software developers. Also new ideas for tool adjustments are included, their implementation and repeating of the experiment, review of the obtained results and findings.

### *Keywords*

COSMIC method, Agile projects, work capacity estimation, time estimate, complexity rating, functional points, history log, product owner, actual user story completion time, Visual Studio 2013, C# programming language (WPF), SQLite, TFS 2013 and TFS Scrum template.

## TURINYS

LENTELIŲ SĄRAŠAS .....	8
PAVEIKSLĖLIŲ SĄRAŠAS .....	9
TERMINŲ IR SANTRUMPŲ ŽODYNAS .....	10
ĮVADAS .....	11
1    PROGNOZAVIMO METODŲ IR PROGRAMINĖS ĮRANGOS ANALIZĖ .....	13
1.1    Analizės tikslas .....	13
1.2    Tyrimo objekto analizė .....	13
1.2.1    Agile metodai .....	13
1.2.2    Užduočių planavimas .....	14
1.3    COSMIC metodo analizė.....	15
1.3.1    Metodo pritaikomumas.....	15
1.3.2    Funkcinio dydžio matavimo principai.....	15
1.4    Egzistuojančių sprendimų analizė.....	17
1.4.1    Metodai taikomi darbo imlumo įvertinimui .....	17
1.4.1.1    Maksimalaus laiko įvertis.....	17
1.4.1.2    Ketrios reikšmės.....	18
1.4.1.3    Nepriklausomas balsavimas .....	18
1.4.2    Programinė įranga naudojama darbo imlumo įvertinimui ir vėlesnei kontrolei.....	18
1.5    Pirminiai reikalavimai prototipui.....	19
1.6    Analizės išvados.....	20
2    REIKALAVIMŲ SPECIFIKACIJA IR SISTEMOS REALIZACIJA .....	21
2.1    Funkciniai reikalavimai .....	21
2.1.1    Bendras funkcionalumas.....	21
2.1.2    Algoritmo apžvalga .....	24
2.2    Nefunkciniai reikalavimai.....	25
2.3    Reikalavimų analizės apibendrinimas.....	26

2.4	Sistemos projektas.....	26
2.5	Realizacijos modelis .....	27
2.5.1	Loginė sistemos architektūra .....	27
2.5.2	Sistemos komponentų struktūra.....	28
2.5.2.1	Atvaizdavimas ( <i>View</i> ) .....	29
2.5.2.2	Atvaizdavimo modelis ( <i>ViewsModel</i> ) .....	29
2.6	Sistemos veikimo aprašymas .....	30
2.7	Testavimo modelis .....	30
2.8	Testavimo duomenys ir rezultatai .....	31
3	TYRIMO IR EKSPERIMENTINĖ DALIS .....	32
3.1	Eksperimento planas .....	32
3.2	Eksperimento rezultatai.....	32
3.2.1	Etapas Nr. 1 .....	33
3.2.2	Etapas Nr. 2 .....	35
3.2.3	Etapas Nr. 3 .....	38
4	IŠVADOS.....	42
5	LITERATŪROS SĄRAŠAS.....	44
6	PRIEDAI .....	46
6.1	Pirmo tyrimo etapo rezultatai.....	46
6.2	Antro tyrimo etapo rezultatai .....	48
6.3	Trečiojo tyrimo etapo rezultatai.....	51

## LENTELIŲ SĄRAŠAS

<b>1 lentelė.</b> Programinės įrangos palyginimas .....	19
<b>2 lentelė.</b> Testavimo rezultatų analizė.....	31



## PAVEIKSLĖLIŲ SĄRAŠAS

<b>1 pav.</b> COSMIC metodo pritaikymo pavyzdys .....	16
<b>2 pav.</b> Kuriamo įrankio panaudojimo diagrama.....	23
<b>3 pav.</b> Įprasto scenarijaus veiklos diagrama.....	23
<b>4 pav.</b> Duomenų gavimo veiklos diagrama.....	24
<b>5 pav.</b> Prognozės skaičiavimo diagrama .....	25
<b>6 pav.</b> Įrankio klasių diagrama .....	27
<b>7 pav.</b> Loginė sistemos architektūra.....	27
<b>8 pav.</b> Įrankio komponentų struktūra .....	28
<b>9 pav.</b> Atvaizdavimo struktūra .....	29
<b>10 pav.</b> Atvaizdavimo modelio struktūra .....	30
<b>11 pav.</b> Komandos įverčio ir dviejų prognozių palyginimas su faktiniu US atlikimo laiku.....	34
<b>12 pav.</b> Komandos įverčio ir prognozių nuokrypio palyginimas.....	34
<b>13 pav.</b> Komandos įverčio ir dviejų prognozių palyginimas su faktiniu US atlikimo laiku.....	36
<b>14 pav.</b> Komandos įverčio ir prognozių nuokrypio palyginimas I ir II etapuose .....	36
<b>15 pav.</b> Komandos įverčio ir dviejų prognozių palyginimas su faktiniu US atlikimo laiku.....	39
<b>16 pav.</b> Komandos įverčio ir prognozių nuokrypio palyginimas visuose etapuose.....	39
<b>17 pav.</b> Prognozių palyginimas visuose tyrimo etapuose .....	40

## TERMINŲ IR SANTRUMPŲ ŽODYNAS

<b>AGILE</b>	iteracinio pobūdžio programų kūrimo metodų rinkinys.
<b>CFP</b>	(angl. <i>COSMIC Function Point</i> ) – dydis apibrėžiantis vieną funkcinį tašką.
<b>COSMIC</b>	(angl. <i>Common Software Measurement International Consortium</i> ) – standartizuotas metodas naudojamas įvertinant programinės įrangos darbo imlumą.
<b>MVVM</b>	(angl. <i>Model View ViewModel</i> ) – architektūrinis modelis, kuriame išskiriami Model, View ir ViewModel komponentai, taip užtikrinant tinkamą bendradarbiavimą tarp sistemos dalių.
<b>PO</b>	(angl. <i>Product Owner</i> ) – asmuo atsakingas už naujų vartotojų scenarijų sukūrimą. Grandis tarp programuotojų komandos ir užsakovo.
<b>SCRUM</b>	(angl. <i>SCRUM</i> ) – Agile struktūra paremtas programų kūrimo metodų rinkinys.
<b>SM</b>	(angl. <i>Scrum Master</i> ) – programuotojų komandos vadovas.
<b>SP</b>	(angl. <i>Story Points</i> ) – matavimo vienetas, kuriuo įvertinamas vienos vartotojo scenarijaus realizacijos laikas.
<b>TFS</b>	(angl. <i>Team Foundation Server</i> ) – Microsoft produktas skirtas duomenų rinkimui, ataskaitų ir projektų stebėjimui.
<b>US</b>	(angl. <i>User Story</i> ) – vartotojo scenarijus. Agile projektuose vartotojo scenarijus – tai trumpas ir paprastas funkcionalumo aprašymas, kuris yra pateikiamas iš suinteresuoto asmens perspektyvos.
<b>WPF</b>	(angl. <i>Windows Presentation Foundation</i> ) – Microsoft produktas skirtas Windows klientinės dalies programavimui.

## **ĮVADAS**

Šiame skyriuje pristatoma tyrimų sritis bei problemos aktualumas. Suformuluojamas tiriamojo darbo tikslas ir uždaviniai. Aptariamas darbo naujumas ir apibrėžiama darbo struktūra.

### **Tyrimo srities objektas ir problema**

Darbo objektas – darbo imlumo įvertinimas Agile projektuose.

Problema – netikslus darbo imlumo įvertinimas planavimo stadijoje, dėl ko gali nepavykti įgyvendinti projekto, arba jo dalies, laiku. Vartotojo scenarijai – tai trumpi ir paprasti funkcionalumo aprašymai, pateikiami iš suinteresuoto asmens perspektyvos. Jų sudėtingumas yra vertinamas taškais SP, kurie nėra verčiami į laikinę skalę, kas dar labiau apsunkina tikslų darbo imlumo įvertinimą pradinėje projekto ir užduočių planavimo stadijoje.

### **Tyrimo tikslas ir uždaviniai**

Pagrindinis tyrimo tikslas - pritaikyti COSMIC metodą realizacijai Agile projektuose, padėsiantį prognozuoti konkrečios užduoties atlikimo laiką ankstyvoje planavimo stadijoje. Tikslui pasiekti buvo suformuluoti šie uždaviniai:

1. Ištirti planavimo ir likusio užduotims skirto laiko metodus naudojamus Agile projektuose.
2. Apžvelgti kuo daugiau įvairių metodų skirtų darbo imlumo įvertinimui, bei padaryti bendras išvadas apie jų patikimumą ir efektyvumą.
3. Susipažinti su jau sukurta programine įranga, taikoma vertinant vartotojo scenarijus pirminio planavimo metu, vėliau sekant užduočių atlikimą ir joms skirtą likusį laiko banką įvertinti jos privalumus bei trūkumus.
4. Išsiaiškinti COSMIC metodo taikymą bei jo ypatumus ir sukurti viziją, pagal kurią būtų galima nustatyti būsimos programinės įrangos prototipo pirminius reikalavimus.
5. Realizuoti prototipą ir atlikti eksperimentinį tyrimą, siekiant išsiaiškinti ar COSMIC metodo integracija Agile projektuose padeda tiksliai prognozuoti užduočių atlikimo laiką.

### **Mokslinis naujumas**

Siūlomas COSMIC metodo pritaikymas Agile projektuose gali padėti išgauti kaip galima tikslesnį laiko įvertį apie planuojamą atlikti vartotojo scenarijų ankstyvos planavimo stadijos metu. Tai yra svarbu dėl kelių priežasčių:

- žinant tikslų vartotojo scenarijaus atlikimo laiką galima atitinkamai planuoti išteklius;

- programuotojų komandos nuomonės dėl potencialių vartotojo scenarijų nebereikia, remiamasi istoriniais duomenimis. Programuotojai gali netrukdomai dirbti prie einamųjų užduočių.

## **Darbo struktūra**

Darbą sudaro keturi pagrindiniai skyriai:

1 Skyrius. Įvadas į tiriamąjį darbą. Šiame skyriuje pristatoma tyrimų sritis bei problemos aktualumas. Suformuluojamas tiriamojo darbo tikslas ir uždaviniai. Aptariamas darbo naujumas ir apibrėžiama darbo struktūra.

2 Skyrius. Apžvalginė dalis. Šiame skyriuje atliekama detali COSMIC metodo analizė, apžvelgiami esami įrankiai, skirti Agile projektams. Įsitikinama, kad nėra sukurto įrankio, leidžiančio prognozuoti vartotojų scenarijų atlikimo laiką, sujungus COSMIC su istoriniais darbų atlikimo duomenimis. Taip pat sudaromi pirminiai reikalavimai kuriamam įrankiui.

3 Skyrius. Reikalavimų specifikacijos ir realizacijos dalis. Skyriuje apžvelgiama ko tikimasi iš kuriamo įrankio ir pateikiamos panaudos atvejų diagramos. Detalizuojamas algoritmas, pristatoma kuriamo įrankio vizija bei aprašomi funkciniai ir nefunkciniai reikalavimai. Taip pat pateikiama sistemos realizacija.

4 Skyrius. Šiame skyriuje trimis etapais atliekamas eksperimentinis tyrimas, kurio metu bandoma išsiaiškinti, ar COSMIC metodo apjungimas su istoriniais darbų atlikimo duomenimis gali pateikti panašią ar net geresnę prognozę, lyginant su programuotojų komandos įvertinimu. Taip pat pateikiamos idėjos įrankio tobulinimui, jos realizuojamos ir kartojamas eksperimentas, siekiant gauti rezultatą kuo panašesnį į faktinį vartotojų scenarijų atlikimo laiką.

# 1 PROGNOZAVIMO METODŲ IR PROGRAMINĖS ĮRANGOS ANALIZĖ

## 1.1 Analizės tikslas

Analizės metu bus nustatytos sukurtos programinės įrangos, skirtos darbumo imlumo įvertinimui bei jo pokyčio sekimui Agile projektuose, privalumai ir trūkumai. Atliekama programinės įrangos analizė. Apžvelgiama galimybė apjungti COSMIC metodą, skirtą darbo imlumo įvertinimui, su Agile projektais (vartotojų scenarijų planavimo stadijoje), bei peržiūrėti realūs taikymo pavyzdžiai. Remiantis atliktos analizės rezultatais vėliau bus išskiriami pagrindiniai kuriamos programinės įrangos aspektai bei komponentai.

## 1.2 Tyrimo objekto analizė

### 1.2.1 Agile metodai

Agile - tai tam tikra programinės įrangos kūrimo ideologija [1]. Remiantis Agile manifestu [2], šis projektų valdymo būdas yra labiausiai orientuotas į rezultatyvumo pasiekimą mažose iteracijose, kurių trukmė yra nuo vienos iki keturių savaitių. Kiekviena iteracija - tai lyg mažas projektas (dar žinomas, kaip vartotojo scenarijus (*angl.* User Story)), turintis visas įprastines dalis: planavimą, reikalavimų surinkimą, dizainą, kodavimą, testavimą ir dokumentacijos parengimą [3].

Įprastinę Agile komandą sudaro:

- Grupės vadovas (Scrum projektuose vadinamas Scrum vadovu) – atsakingas už laiku atliktus ir pristatytus komandos darbus;
- Programuotojų komanda – dažniausiai tai 5-9 darbuotojai;
- Produkto šeimininkas – asmuo atsakingas už vartotojo scenarijų parengimą ir pateikimą komandai, užtikrinimą jog informacija yra teisingai suprasta, komandos narių konsultavimą bei vartotojo reikalavimų išaiškinimą ir jau realizuoto funkcionalumo pristatymą jam.

Prieš prasidedant iteracijai, produkto šeimininkas suorganizuoja susirinkimą (*angl.* grooming), per kurį yra pristatomi vartotojo scenarijai, kuriuos planuojama įgyvendinti artimiausiu metu. Šio susirinkimo metu yra apžvelgiami jau egzistuojantys sprendimai, supažindinama su konkrečia sritimi, esant poreikiui peržiūrimas turimas programinis kodas. Prasidėjus iteracijai vyksta užduočių planavimas, kurio metu vartotojo scenarijai išskaidomi į atskiras užduotis (detaliau apžvelgta kitame skyriuje (1.2.2 *Užduočių planavimas*)). Iteracijos metu vyksta užduočių realizavimas, o jos pabaigoje pasirošama pristatomajai sesijai (*angl.* demo), kurios metu yra pristatomas realizuotas funkcionalumas produkto šeimininkui [4].

## 1.2.2 Užduočių planavimas

Kodėl taip svarbu išskirti Agile metodiką iš kitų projektų valdymų tipų? Todėl, kad visos komandos nuomonė yra vienintelė ir lemiamą, kai siekiama suplanuoti pasirinktą vartotojo scenarijaus realizaciją [4]. Jeigu lygintume su įprastiniu krioklio metodu, būtų galima pastebėti, jog jokiai programuotojų komandai nėra suteikiama pilna laisvė įsivertinti atliekamų užduočių imlumą [5]. Agile projektuose produkto šeimininkas sutaria su vartotoju dėl vartotojo scenarijaus, kurį pateikia komandai. Vartotojo scenarijus susideda iš trijų pagrindinių dalių:

- koks yra pageidaujamas funkcionalumas?
- kokiam asmeniui jis reikalingas?
- kodėl to reikia?

Turint atsakymus į aukščiau išvardintus pagrindinius klausimus, komanda gali pradėti planuoti užduotis. Išskirsčius vartotojo scenarijų į pačias mažiausias individualias užduotis, kurios idealiai atveju tarpusavyje nepersidengia, yra prognozuojama, kiek laiko užtruks kiekvienos atskiros užduoties atlikimas. Bendra visų užduočių prognozių suma ir yra laikoma vartotojo scenarijaus įgyvendinimo laiku. Nors užduotys suplanuojamos valandomis, tačiau pats vartotojo scenarijus yra įvertinamas kitu matu, kuris neturi tiesioginio atitikmens nei vienai iš laiko sistemų. Tai nebūtų didelė problema, jei komanda visuomet tiksliai nustatytų planuojamų užduočių darbo imlumą.

Įvairiuose šaltiniuose teigiama, jog vartotojo scenarijų sudėtingumo įvertinimas panaudojant SP matą palengvina versijų išleidimo prognozę, duoda tikslesnius užduočių imlumo įvertinimus bei drastiškai sumažina patį planavimo laiką [6]. Tačiau problema, kad sunku pasakyti tikslų laiką, kada bus baigtas vartotojo scenarijus – niekur nedingsta. Ypač tai yra aktualu, kol programuotojų komanda nepradėjo planavimo, o produkto šeimininkui reikia žinoti bent jau preliminarų laiko intervalą, per kurį vartotojo scenarijus galėtų būti įgyvendintas. Agile projektai ne išimtis. Užduočių planavimo stadijoje Agile projektai yra tokie patys, kaip ir kiti projektai, kurių valdyme kyla sunkumų. „Pagrindinė problema laikoma nepakankamas lengvų užduočių įvertinimas, o sudėtingų užduočių – pervertinimas“ [5, p. 82]. Dažnai pasitaikanti tarp programuotojų problema - pervertinti sudėtingą užduotį ir nepakankamai įvertinti lengvesnę, kurios įvykdymui atrodė užteks keleto valandų. Galima išmokyti teisingai įvertinti darbo imlumą, tačiau ši patirtis įgyjama sėkmingai ir laiku įgyvendinant užduotis.

### 1.3 COSMIC metodo analizė

COSMIC metodas apibrėžia principus, taisykles ir standartinio funkcinio dydžio matavimo procesą. Funkcinis dydis – tai matavimo vienetas nurodantis funkcionalumo kiekį, kurį suteikia įrankis arba atskira jo dalis, visiškai neatsižvelgiant į technines arba kokybines įrankio specifikacijas [7].

#### 1.3.1 Metodo pritaikomumas

COSMIC metodas gali būti naudojamas dydžio įvertinimui toliau pateiktuose programinės įrangos tipuose:

- verslui orientuotos informacinės sistemos;
- realaus laiko programinė įranga;
- infrastruktūros programinė įranga (pvz.: operacinės sistemos);
- aukščiau išvardintų programinės įrangos tipų hibridai.

Visus šiuos programinės įrangos tipus apjungia funkcijos, kurios yra susijusios su įvesties, saugojimo ir atkūrimo bei išvesties duomenimis. COSMIC metodas gali būti taikomas norint išmatuoti programinės įrangos funkcinius vartotojo reikalavimus:

- bet kokiame išskaidymo lygmenyje – galima vertinti visą programinę įrangą arba tik jos dalį;
- bet kuriame iš daugiasluoksnės architektūros sluoksnių;
- bet kuriame programinės įrangos gyvavimo ciklo etape.

#### 1.3.2 Funkcinio dydžio matavimo principai

Metodas yra skirtas įvertinti programinės įrangos dydį žiūrint iš vartotojo perspektyvos ir naudoja „bendrą COSMIC programinės įrangos modelį“ (*angl. „COSMIC Generic Software Model“*), kuris yra paremtas esminiais programinės įrangos kūrimo principais:

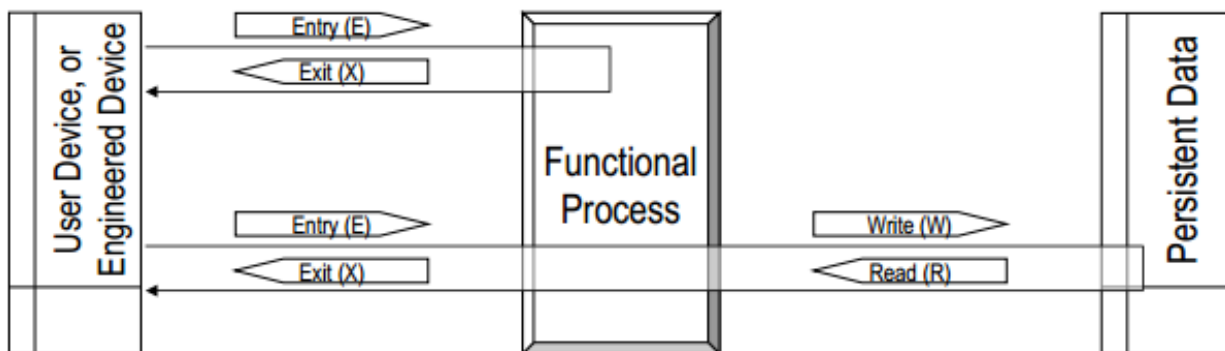
- funkciniai vartotojo reikalavimai programinei įrangai gali būti suskaidyti į unikalius funkcinius procesus, kurie sudaryti iš sub-procesų. Sub-procesas gali būti arba informacijos judėjimas arba informacijos apdorojimas;
- kiekvienas funkcinis procesas prasideda nuo vartotojo pradinės informacijos perdavimo programinei įrangai;
- vienas informacijos grupės atributas, apibūdinantis domėjimosi objektą, yra perduodamas informacijos judėjimu.

Kiekvienas informacijos judėjimas (operacija) gali būti aprašomas vienu iš keturių raktinių žodžių:

- įėjimas – transakcija, kuri keičia vidinius programos duomenis;
- išėjimas – transakcija, kai duomenys yra perduodami už sistemos ribų;
- rašymas – informacijos įrašymas į duomenų saugyklą;
- skaitymas – informacijos nuskaitymas iš duomenų saugyklos.

Sub-procesai nėra atskirai matuojami dėl matavimų apvalinimo. Programinės įrangos dydis nurodomas kaip informacijos judėjimo kartų (procesų - „įėjimas“, „išėjimas“, „rašymas“ bei „skaitymas“) suma sudedant visus funkcinis procesus. Kiekvienas informacijos judėjimas yra skaičiuojamas kaip vienas CFP. Funkcinio proceso bei pačios programinės įrangos dydis turi būti ne mažesnis kaip 2 CFP, be jokių viršutinių limitų [7].

Naudojantis COSMIC metodu, galima paprastai suskaičiuoti funkcinis taškus. 1 paveiksle pateiktas pavyzdys, kurio bendra funkcinis taškų suma yra 6FT, nes kiekviena operacija yra įvertinama vienu funkcinis tašku [8].



1 pav. COSMIC metodo pritaikymo pavyzdys [8]

Jau turint konkretų taškų skaičių galima juos paversti į laikinę išraišką, tačiau tam reikia papildomo pasiruošimo. Skirtingos programuotojų komandos dirba skirtingu greičiu, tad ir toki pat vartotojo scenarijų, įvertintą vienodu funkcinis taškų skaičiumi jos realizuos per skirtingą laiką. Akivaizdu, kad būtina pasitelkti istorinius duomenis siekiant išgauti laikinę išraišką iš funkcinis taškų, skirtą konkrečiai programuotojų komandai [9].

Atlikus šį veiksma ir peržiūrėjus istorinę informaciją (kiek vidutiniškai užtruko atlikti vartotojo scenarijus įvertintus funkcinis taškų skaičiumi) galima išvesti vidurkį, kuris parodytų, kiek laiko užtrunka realizuoti vieną funkcinis tašką. Tai būtų labai naudinga produkto šeiminkui, kuris galėtų greitai atlikti



dar net nepradėto realizuoti funkcionalumo preliminarų laiko įvertinimą ir nuspręsti ar komanda spės jį įgyvendinti.

#### **1.4 Egzistuojančių sprendimų analizė**

Siekiant tiksliai įvertinti darbo imlumą reikia ne tik naudotis tinkama programine įranga, bet ir išmanyti pagrindinius darbo imlumo įvertinimo metodus.

##### **1.4.1 Metodai taikomi darbo imlumo įvertinimui**

Yra daug skirtingų darbo imlumo įvertinimo metodų. Kai kurie autoriai pateikia net šešiolika skirtingų variantų [10]. Trumpai apžvelgsime kelis iš jų:

###### **1.4.1.1 Maksimalaus laiko įvertis**

Vienas iš laiko planavimo metodų yra maksimalaus laiko įvertis [10]. Jei programuotojas mano, jog užduotį gali įvykdyti per vieną dieną, jis atverčia kortelę su skaičiumi 1. Deja, vėliau šią užduotį gali tekti daryti kitam programuotojui, kuris iškart manė, kad ši užduotis verta dviejų ar net keturių darbo dienų, tokiu būdu užtrukdamas prie užduoties gerokai ilgiau nei tikėtasi. Būtent dėl šios priežasties galima pasinaudoti racionalia idėja - kiekvienai užduočiai rinktis didžiausią iš pateiktų laiko apskaičiavimų. Gali atrodyti, kad tai yra tik laiko švaistymas, tačiau, skirtingi komandos nariai apsvarsto skirtingus galimus sunkumus, atliekant specifinę užduotį, ir tas žmogus, kuris apskaičiavo ilgiausią laiką, greičiausiai apmąstė daugiausiai klausimų ir nesklandumų, galinčių atsirasti pradėjus ją įgyvendinti.

Maksimalaus laiko įverčio pasirinkimas turi ir daugiau privalumų. Renkantis pateiktą trumpesnę laiko apskaičiavimą komandos narys, kuris būtų sakęs didesnę skaičių gali būti priverstas paaiškinti, kodėl jo planuojamas užduoties laikas yra ilgesnis. Dažnai tokioje situacijoje programuotojas gali pateikti didesnę laiką vien dėl patirties ar reikiamų įrankių valdymo meistriškumo stokos. Laiko apskaičiavimas dažniausiai remiasi savo patirties lygiu ir būtų nesąžininga, jei komandos narys jaustųsi nepatogiai pateikdamas savo tikrą apskaičiuotą laiką vien dėl to, kad bijotų aptarinėti savo patirties, susijusios su tam tikrais įrankiais arba darbo sritimi, trūkumą.

Laiko apskaičiavimas dažniausiai remiasi darbuotojo patirtimi, jei komandos narys pateikdamas apskaičiuotą laiką suklystų, tai turėtų įtakos laiko planavimui. Paskutiniame etape renkantis ilgiausią apskaičiuotą laiką, galima sutrumpinti susirinkimus, skirtus laiko įvertinimui. Jei programuotojas mano, jog konkreti užduotis turėtų užtrukti, tarkim, aštuonias dienas, jis gali tą pasakyti aptariant užduotį pradžioje. Jei niekas iš kitų komandos narių neprieštarauja tokiam laiko apskaičiavimui tada užkertamas kelias tolimesnei bereikšmei diskusijai šia tema ir galima pereiti prie kitos užduoties.

### 1.4.1.2 Keturios reikšmės

Viename projekte [10] buvo naudojamos keturios reikšmės - 1, 2, 4 ir 8 nurodant laiko, reikalingo atlikti užduotį, apskaičiavimui. Po pirmų dviejų planavimo susirinkimų buvo pastebėta, kad vos mažiau nei 5% užduočių buvo priskirta vieneto reikšmė ir daugiau nei 30% - dvejeta. Projekto vadovai nusprendė pašalinti vieneto reikšmę iš planavimo sistemos, nes tai pasirodė protingas žingsnis. Po kelių susitikimų paaiškėjo, dabar vos 5% užduočių buvo įvertintos dvejetu ir daugiau nei 30% - ketvertu. Manoma, kad taip yra dėl to, kad programuotojai yra iš prigimties gana skeptiški žmonės ir retas programuotojas yra pasiryžęs teigti, kad užduotis gali būti atlikta taip greitai, kaip tik įmanoma tai įvertinti. Šie atradimai parodo, kad vertinant užduotis geriausia naudoti mažiausiai keturias reikšmes. Taip pat geriausia naudoti maksimaliai 4 reikšmes, nes tai tėra apytikslis planavimas.

### 1.4.1.3 Nepriklausomas balsavimas

Kitų žmonių įtaka gali turėti esminės reikšmės sprendimui. Jei labiausiai patyręs ir gerbiamas komandos narys sako, jos ši užduotis verta dviejų dienų, tai yra labai tikėtina, jog ir kiti komandos nariai jam pritaris. Dėl šios priežasties yra svarbu, kad balsavimas vyktų taip, kad komandos nariai negalėtų vienas kito įtakoti. Tam puikiai tinka kiekvieno nario taškų rašymas popieriaus lapuos, kurie peržiūrimi tik susirinkimo pabaigoje [10].

Kitas, labai panašus metodas, tai planavimo pokeris (*angl. Planning Poker*), kuris tarytum apjungia nepriklausomo balsavimo ir keturių reikšmių metodus ir yra vienas populiariausių Agile projektuose [3].

## 1.4.2 Programinė įranga naudojama darbo imlumo įvertinimui ir vėlesnei kontrolei

Apžvelgsime penkis skirtingus nemokamos programinės įrangos pavyzdžius:

**IceScrum** – interneto įrankis, galintis pasiūlyti daugiausiai funkcionalumo [11]. Pradžioje gali atrodyti, kad jo naudojimas sudėtingas, tačiau užtenka prie jo įprasti, kad būtų galima įvertinti siūlomas galimybes. Vis dar aktyviai tobulinamas. Šis įrankis susideda iš elektroninės lentos, skirtingų užduočių statusų bei galimybės balsuoti įvertinant užduoties darbo imlumą planavimo pokerio metodu.

**Agilefant** – greitas internetinis įrankis skirtas kurti produktus, projektus, iteracijas, ir vartotojo scenarijus [12]. Nors šis įrankis ir yra sukurtas Agile projektams, tačiau jame nėra tokios sąvokos kaip SP.

**XPlanner** – vienas pirmųjų internetinių įrankių skirtų Agile projektų valdymui, turintis daug teigiamų ypatybių, kuriomis negali pasigirti kitų programinių įrangų kūrėjai. Deja, jau kurį laiką (nuo 2006 m.) nebepasirodo atnaujinimų [13].

*Agilo* – įrankis vis dar aktyviai tobulinamas. Nors ir turi keletą gerų ypatybių (diagramos bei skelbimų lenta), tačiau pati įrankio koncepcija turi trūkumų – nėra realizuotų projektų (tai reiškia, jog dirbant su daugiau nei vienu projektu, reikės ieškoti būdų, kaip apeiti šią spragą), sudėtinga sukurti iteracijų planą, nors tai dažnai naudojama užduotis [14].

*XPStoryStudio* – nors ir atrodo gražiai ir patogiai, tačiau šio įrankio atnaujinimai buvo sustabdyti dar 2004 m., tad šis įrankis nebebus tobulinamas [15].

Aukščiau aptartų skirtingų įrankių palyginimas pateiktas pirmoje lentelėje.

**1 lentelė.** Programinės įrangos palyginimas

Įrankis	Plusai	Minusai
IceScrum	<ul style="list-style-type: none"> <li>• Patogi vartotojo sąsaja;</li> <li>• Tinkamos diagramos;</li> <li>• Realizuotas Drag&amp;Drop.</li> </ul>	<ul style="list-style-type: none"> <li>• Tinkamas tik mažoms komandoms;</li> <li>• Tinkamas tik vienam projektui.</li> </ul>
Agilefant	<ul style="list-style-type: none"> <li>• Lengvas naudojimas;</li> <li>• Patogus laiko sekimas.</li> </ul>	<ul style="list-style-type: none"> <li>• Trūksta SP koncepcijos;</li> <li>• Per mažai ataskaitų.</li> </ul>
XPlanner	<ul style="list-style-type: none"> <li>• E-mail pranešimai;</li> <li>• Tinkamos ataskaitos;</li> <li>• Didelė funkcijų pasiūla.</li> </ul>	<ul style="list-style-type: none"> <li>• Sudėtingas diegimas (nedirbantiems su Java);</li> <li>• Nėra išleidimo versijų (<i>angl.</i> release date), tik iteracijos.</li> </ul>
Agilo	<ul style="list-style-type: none"> <li>• Integruotas Wiki;</li> <li>• Išsaugoma užklausų istorija;</li> <li>• Vartotojo scenarijus gali būti išskaidytas į užduotis.</li> </ul>	<ul style="list-style-type: none"> <li>• Nėra projektų;</li> <li>• Sudėtinga naudoti vartotojo sąsaja;</li> <li>• Nėra planavimo aplinkos.</li> </ul>
XPStoryStudio	<ul style="list-style-type: none"> <li>• Automatinė instaliavimo sistema.</li> </ul>	<ul style="list-style-type: none"> <li>• Neveikiantis funkcionalumas su FireFox naršykle;</li> <li>• Nėra išleidimo versijų, tik iteracijos.</li> </ul>

## 1.5 Pirminiai reikalavimai prototipui

Susipažinus su įvairiais planavimo metodais bei apžvelgus pagrindinius Agile projektuose naudojamus įrankius, galima apibrėžti pirminius reikalavimus prototipui. Kuriama programinė įranga turi sugebėti:

- leisti įvesti funkcinių taškų skaičių, aprašant vartotojo scenarijus;
- užfiksuoti skirtingų komandos narių kiekvienos individualios užduoties prognozuojamą darbo imlumo įvertinimą;
- kaupti istorinę informaciją apie vartotojo scenarijus ir tikrąjį jų atlikimo laiką;
- pateikti prognozę, kiek laiko užtruks įgyvendinti vartotojo scenarijų, remiantis funkcinių taškų skaičiumi ir sukaupta istorine informacija.

## **1.6 Analizės išvados**

Atlikus analizę paaiškėjo, kad yra įvairių darbo imlumo planavimo ir matavimo metodų, bei įrankių. Tačiau nėra nei vieno įrankio, kuris būtų pritaikytas konkrečiai Agile projektams ir remtųsi programuotojų komandos darbų istoriniais atlikimo duomenimis bei turėtų integruotą COSMIC metodą, skirtą gauti greitą ir maksimaliai tikslų darbo imlumo įvertinimą.

## **2 REIKALAVIMŲ SPECIFIKACIJA IR SISTEMOS REALIZACIJA**

### **2.1 Funkciniai reikalavimai**

Šiame skyriuje yra sudaromi funkciniai reikalavimai, kurie gaunami detalizuojant pirminius reikalavimus prototipui.

#### **2.1.1 Bendras funkcionalumas**

Produkto šeimininkui sukūrus naują vartotojo scenarijų, jis pats galės nuspręsti ir pažymėti atitinkamas kuriamo komponento savybes: skaitymo ir rašymo transakcijas. Ankstyvos planavimo stadijos metu kiekvienas iš komandos narių turės pateikti skaitinį įvertinimą, reprezentuojantį vartotojo scenarijaus atlikimo laiką SP vienetais. Vėlyvoje planavimo stadijoje, kurios metu yra kreipiamas didelis dėmesys į vartotojo scenarijaus detales bei specifinius reikalavimus, komandos nariai išanalizavę reikalavimus nustato, kiek laiko užtruks įgyvendinti kiekvieną iš individualių užduočių, kurių suma yra prilyginama anksčiau nustatytam SP įverčiui (esant poreikiui, jis gali būti keičiamas). Pasitaiko atveju, kai vartotojo scenarijus būna panašus į anksčiau realizuotą. Tokiu atveju, reikia pažymėti vartotojo scenarijų, kaip panašų į prieš tai buvusį ir remtis jo atlikimo greičiu. Pasibaigus iteracijai, turimi tikslūs duomenys, kiek laiko užtruko kiekviena iš užduočių. Remiantis šiais duomenimis galima apskaičiuoti komandos darbo greitį ir surasti valandinį atitikmenį vienam SP įverčiui.

Detalizuoti funkciniai reikalavimai:

1. aprašant vartotojo scenarijų, įvedamos skaitinės reikšmės, nurodančios kiek yra įėjimų, išėjimų, skaitymų ir rašymų operacijų;
2. turi būti realizuota galimybė pažymėti vartotojo scenarijų panašiu į prieš tai buvusius, nurodant jų identifikacinius numerius (bus remiamasi ankstesnių vartotojo scenarijų atlikimo vidutiniu greičiu);
3. į sistema turi būti įvedamas komandos narių sąrašas, kurį bus galima koreguoti;
4. tuo atveju, kai komandos narys nedalyvauja ankstyvoje planavimo stadijoje – jo įvertinimas paliekamas tuščias ir neįtraukiamas į skaičiavimus;
5. komandos narių įvertinimai pateikiami skaitine išraiška reprezentuojančia darbo valandas;
6. esant situacijai, kai komanda pakeitė nuomonę dėl SP įverčio, šiuos pakeitimus būtina įvesti kuriamu įrankiu. Visa SP reikšmės istorija yra išsaugoma;
7. pasibaigus iteracijai iš TFS modulio yra paimami duomenys, apie konkrečių užduočių faktinį atlikimo laiką;

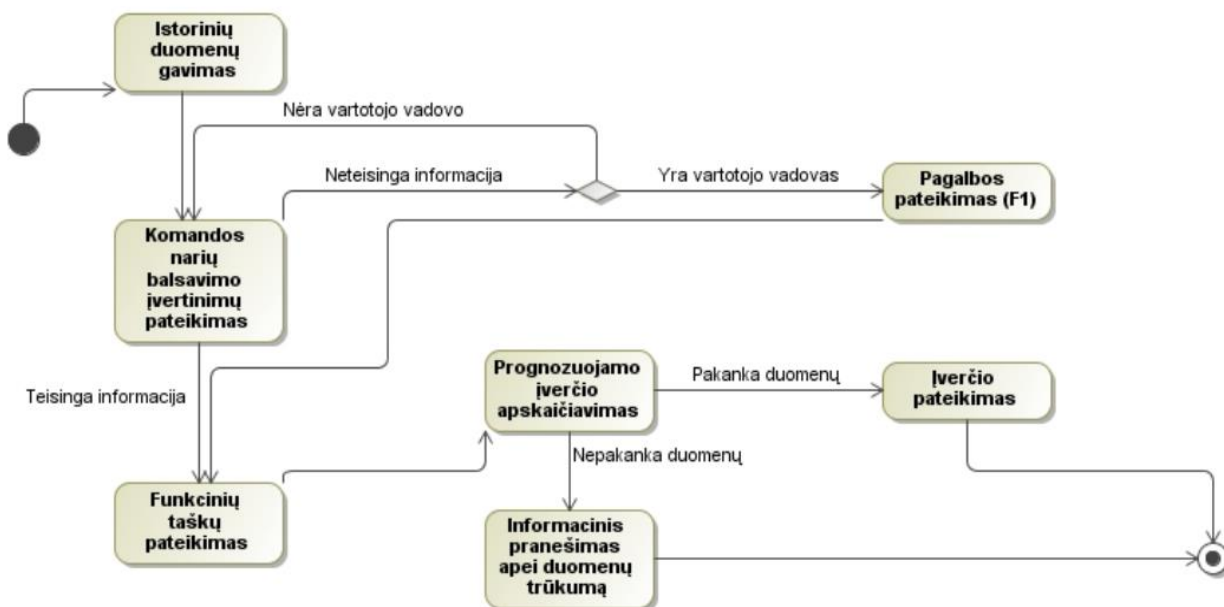
8. tolimesnėse iteracijose įvedus funkcinių taškų skaičių yra pateikiama valandinė išraiška, kiek laiko užtruks įgyvendinti vartotojo scenarijų;
9. po vartotojo scenarijaus įgyvendinimo, kai jau bus žinomas tikslus sugaištas laikas, atsidarius jį, bus pateiktas prognozės tikslumo procentas;
10. skirtingi vartotojai galės atlikti skirtingus veiksmus: produkto šeimininkas neturės galimybės įvesti komandos narių balsavimo įvertinimą, tai galės atlikti tik grupės vadovas. Šis atitinkamai neturės teisės priskirti funkcinių taškų skaičiaus vartotojų scenarijams;
11. turi būti realizuota galimybė eksportuoti istorinius duomenis kartu su prognozuotomis reikšmėmis *Excel* formatu.

Pagrindinis kuriamos sistemos panaudojimo atvejis pateikiamas 2 paveiksle . Išskiriami trys veikėjai: produkto šeimininkas, grupės vadovas ir administratorius, nurodoma ir TFS sistema. Prieš pradėdant darbą su kuriu įrankiu administratoriaus teises turintis asmuo turės nurodyti, kokiems asmenims kokios rolės yra priskirtos, taip bus užtikrintas tikslingas ir korektiškas įrankio naudojimas. Produkto šeimininkui kuriant naują vartotojo scenarijų, jis turės nurodyti aprašymą, bei įėjimų, išėjimų, skaitymų ir rašymų kieki. Programa turėdama šį įvertinimą remsis turimais istoriniais duomenimis ir pateiks prognozę, kiek laiko užtruks įgyvendinti konkretų vartotojo scenarijų. Bus išskirtas atvejis, kai dėl istorinių duomenų trūkumo prognozės pateikti nebus įmanoma. Taip pat, kitas ne mažiau svarbus atvejis, kai kuriamas vartotojo scenarijus bus panašus į jau anksčiau buvusius. Tokiu atveju, bus realizuota galimybė pažymėti tokį vartotojo scenarijų panašiu į ankstesnius, tuomet bus remiamasi ankstesnių vartotojų scenarijų atlikimo vidutiniu greičiu.



2 pav. Kuriamo įrankio panaudojimo diagrama

Turint istorinius duomenis, kiek laiko buvo planuota, jog užtruks įgyvendinti vartotojo scenarijų, ir kiek faktiškai užtruko jį realizuoti, galima prognozuoti, kiek laiko prireiks ateityje. Įvedus funkcinių taškų skaičių, bus galima nustatyti, kiek laiko užtruks įgyvendinti vieną funkcinį tašką ir tokiu būdu apskaičiuoti bendrą vartotojo scenarijaus atlikimo laiką.



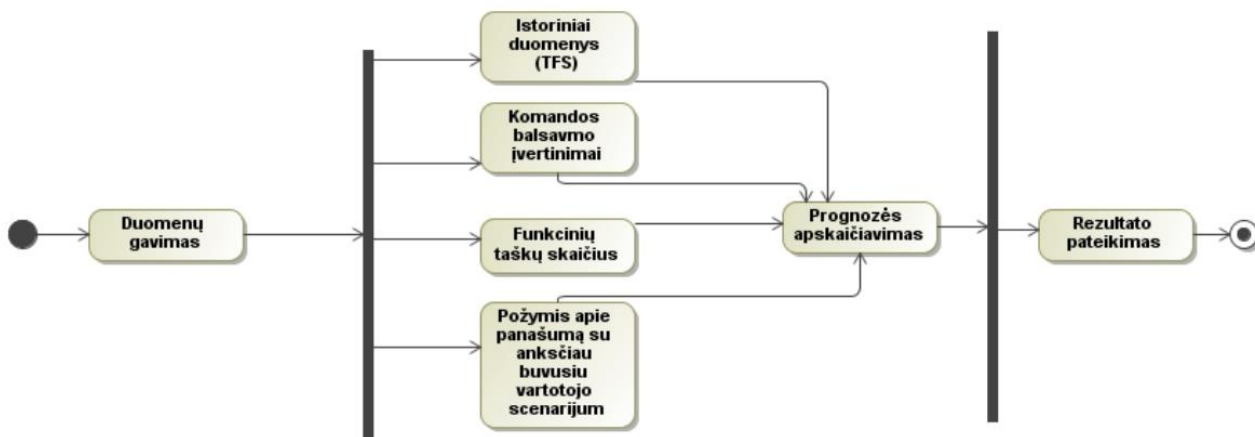
3 pav. Įprasto scenarijaus veiklos diagrama

3 paveiksle pateiktoje diagramoje detalizuojamas įprastinis scenarijus, kurio metu pirmiausia paimami istoriniai duomenys iš TFS sistemos. Vėliau gaunami komandos balsavimo rezultatai dėl

konkreto vartotojo scenarijus: tuo atveju, kai duomenys yra užpildomi neteisingai, siūloma naudotis pateiktu vartotojo vadovu. Tada produkto šeimininkas pateikia savo įverčius funkcinų taškų skaičiumi ir nurodo, ar aktualus vartotojo scenarijus yra panašus į prieš tai buvusį. Turint visą reikalingą informaciją yra įvertinama ar pakanka duomenų prognozei apskaičiuoti. Tuo atveju, kai istorinių duomenų trūksta, yra pateikiamas informacinis pranešimas. Esant pakankamam kiekiui istorinių duomenų yra atliekami skaičiavimai ir pateikiamas prognozuojamo vartotojo scenarijus valandinis įvertis.

## 2.1.2 Algoritmo apžvalga

Prognozė pateikiama tik tuo atveju, jei sistemoje bus pateikta pakankamai istorinių duomenų. Veiklos diagrama pateikta 4 paveiksle, atvaizduoja lygiagretų reikiamos informacijos gavimą, kuri vėliau bus naudojama vartotojo scenarijus atlikimo laiko prognozavimui.



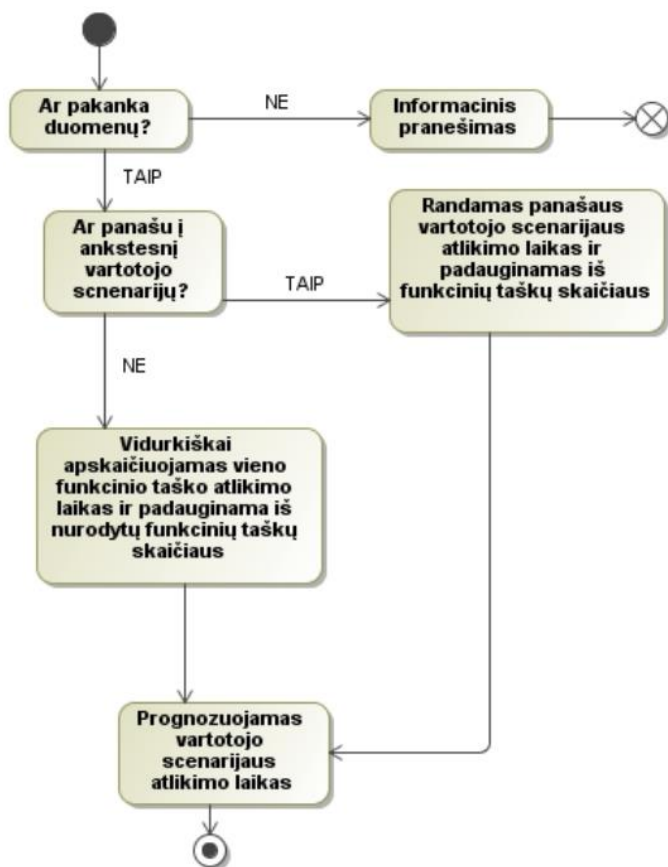
4 pav. Duomenų gavimo veiklos diagrama

Pagrindine kuriamo įrankio dalimi galima išskirti prognozės skaičiavimo modulį. 5 paveiksle detalios išnagrinėtas aukščiau paminėtas modulis: patikrinus ar pakanka istorinių duomenų (jei jų nepakanka, yra parodomas informacinis pranešimas), yra atliekami skaičiavimai. Priklausomai nuo to, ar produkto šeimininkas pažymėjo šį vartotojo scenarijų panašų į anksčiau buvusį, yra apskaičiuojama, kiek laiko užtruks įgyvendinti vartotojo scenarijų:

- jei vartotojo scenarijus yra panašus į anksčiau buvusį, yra surandamas ankstesnio scenarijus atlikimo laikas, apskaičiuojama kiek laiko užtruks vieno funkcinio taško atlikimas ir įvertis padauginamas iš bendro funkcinų taškų skaičiaus priskirto aktualiam vartotojo scenarijui;
- tuo atveju, kai vartotojo scenarijus nėra panašus į anksčiau buvusį, remiantis istoriniais duomenis apskaičiuojama, kiek laiko užtruks atlikti vartotojo scenarijus. Suradus šį atitikmenį, apskaičiuojama kiek laiko užtrunka vieno funkcinio taško atlikimas ir šis įvertis



yra taip pat padauginamas iš produkto šeimininko nurodyto funkcinio taškų skaičiaus. Tokiu būdu gaunamas įvertis, parodantis laiką, reikalingą šiuo metu planuojamo vartotojo scenarijaus atlikimui.



5 pav. Prognozės skaičiavimo diagrama

Priklausomai nuo istorinių duomenų kiekio ir prieinamumo, galima vis tiksliau ir tiksliau prognozuoti vartotojo scenarijų atlikimo laiką.

## 2.2 Nefunkciniai reikalavimai

Šiame skyriuje apžvelgiami nefunkciniai sistemos reikalavimai, kurie padės dar geriau išanalizuoti kuriamos sistemos detales.

Kaip ir kiekviena programinė įranga, taip ir kuriamas įrankis, turi būti patogus naudoti. Apžvelgus sukurtos programinės įrangos pavyzdžius, nustatyta, kokie vartotojo sąsajos elementai ir minimalūs kokybės kriterijai yra būtini kuriamam įrankiui:

1. F1 mygtuko panaudojimas vartotojo pagalbos pateikimui;

2. Bandant įvesti ne skaitinę reikšmę (nurodant komandos narių įverčius arba įėjimus, išėjimus, skaitymą ir rašymą) programa pateiks informacinį pranešimą apie nekorektiškus duomenis;
3. TFS sistemos nepasiekiamumo atveju (esant ryšio sutrikimams) sistema informuos vartotoją apie tai ir bandys prisijungti kas 10 sekundžių, kol pavyks tai padaryti;
4. Vartotojas galės nenutrūkstamai naviguoti per vieno lango duomenų laukus mygtuko „Tab“ pagalba;
5. Išsaugojus rezultatus, sistema pateiks pranešimą, apie sėkmingai įvykdytą veiksmą;

### **2.3 Reikalavimų analizės apibendrinimas**

Reikalavimų analizės metu buvo detalizuoti pirminiai reikalavimai išskiriant juos į pagrindinius funkcinius ir nefunkcinius reikalavimus, kurių įgyvendinimas užtikrins kuriamo įrankio korektišką ir nenutrūkstamą veikimą. Nustatytos pagrindinės rolės, kurias įgaus sistema besinaudojantys asmenys. Pateiktas bendras algoritmo aprašas, kuriame išskirti pagrindiniai sudedamieji elementai ir būsimi skaičiavimai. Svarbiausia kuriamo įrankio dalis – prognozės skaičiavimo modulis, išnagrinėtas ir detalizuotas, siekiant užtikrinti teisingą ir kuo tikslesnį prognozuojamo laiko įverčio nustatymą. Numatyti kokybės kriterijai atsarginių kopijų periodiškumui bei ryšio sutrikimo atvejų metu – laukimo laikui.

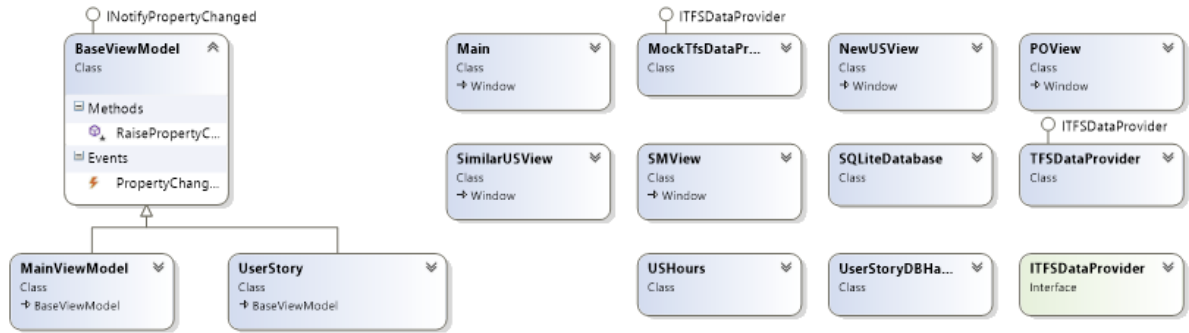
### **2.4 Sistemos projektas**

Apžvelgus detalizuotus reikalavimus kuriamam įrankiui, buvo pasirinkta:

- Visual Studio 2013;
- C# programavimo kalba (WPF modelis);
- SQLite biblioteka prisijungimui ir duomenų gavimui iš duomenų bazės;
- TFS 2013 sistema ir TFS Scrum šablonas.

## 2.5 Realizacijos modelis

6 paveiksle pateikta bendra klasių diagrama.

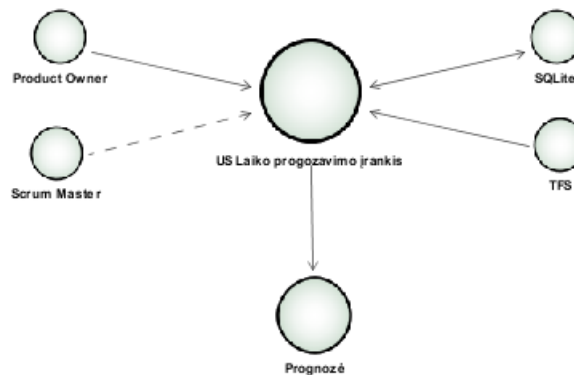


6 pav. Įrankio klasių diagrama

Kiekvienai atskirai grafinei vartotojo sąsajai yra sukuriama nauja klasė, kurios pavadinimo pabaigoje pridamas žodis atvaizdavimas (angl. *View*). Išskiriamos atskiros klasės prisijungimui prie duomenų bazės ir TFS sistemos. Taip pat, pagrindinė atvaizdavimo modelio klasė, paveldinti informaciją iš bazinės klasės ir turinti priėjimą prie informacijos apie vartotojo scenarijų.

### 2.5.1 Loginė sistemos architektūra

Pagal 2 paveiksle pateiktą panaudos diagramą buvo kuriama sistemos architektūra. Pirmiausia išskiriami du skirtingi duomenų atvaizdavimai produkto šeiminkui (PO) ir komandos vadovui (SM). Taip pat integruojamas prisijungimas prie duomenų bazės, kurioje saugoma informacija apie vartotojo scenarijus. Galiausiai realizuojamas prisijungimas prie TFS sistemos, iš kurios gaunami istoriniai duomenys apie vartotojo scenarijus, prognozuojamą ir faktinį užduočių atlikimo laiką. Turint visus šiuos duomenis, galima prognozuoti, kiek laiko turėtų užtrukti vartotojo scenarijaus realizavimas. Loginė sistemos architektūra pateikta 7 paveiksle.

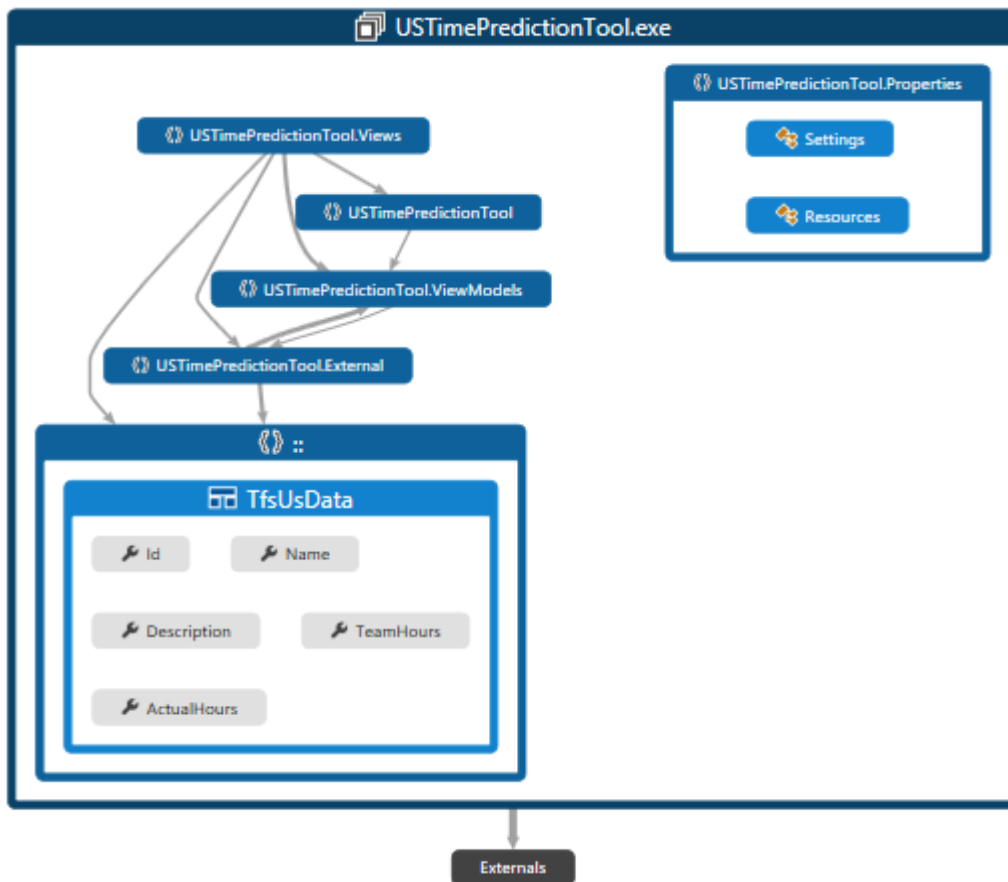


7 pav. Loginė sistemos architektūra

Svarbu paminėti, kad SM rolė nėra būtina (pažymėta punktyrine linija) – įrankiu galima pilnai naudotis ir gauti prognozuojamą valandinę išraišką remiantis tik istoriniais duomenimis ir PO įvestu funkinių taškų skaičiumi. Komunikacija su duomenų baze reikalinga dėl keleto dalykų: įrašymas skirtas išsaugoti informaciją apie naują vartotojo scenarijų kartu su PO nurodytu funkinių taškų skaičiumi, nuskaitymas – duomenų atvaizdavimui. Prisijungimas ir bendradarbiavimas su TFS sistema reikalingas siekiant gauti istorinius duomenis apie jau atliktus vartotojo scenarijus. Tiesa, TFS sistemoje saugoma informacija apie vartotojo scenarijus, kurie nėra vertinami laikine išraiška. Kiekvienas vartotojo scenarijus turi įvertinimą pateiktą SP. Norint surasti laiką, kuris buvo sugaištas atliekant vartotojo scenarijų, reikėjo rasti susietų užduočių prognozuojamą ir realų atlikimo laikus.

### 2.5.2 Sistemos komponentų struktūra

Įrankis buvo kurtas remiantis MVVM architektūra. Ši architektūra leidžia patogiai išskirti sistemos komponentus, taip užtikrinant, kad atvaizduojama informacija būtų korektiška ir pasiektų tik tas sistemos dalis, kurioms ji skirta. 8 pav. Įrankio pavaizduotas šių komponentų išskyrimas.

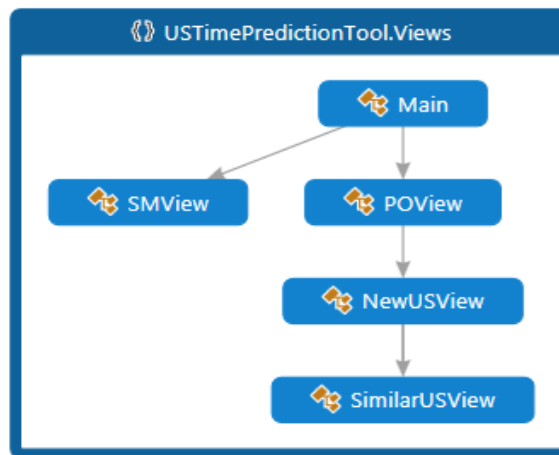


8 pav. Įrankio komponentų struktūra

Tolimesniuose skyriuose (2.5.2.1 ir 2.5.2.2) yra išsamiau apžvelgiami atvaizdavimo ir atvaizdavimo modelio komponentai bei pateikiamas jų detalus paaiškinimas.

### 2.5.2.1 Atvaizdavimas (View)

Šiame skyriuje apžvelgiamos įrankio grafinės vartotojo sąsajos. Atsidarius programą pirmiausia reikia pasirinkti rolę (SM arba PO), su kuria jungiamasi prie įrankio. Prisijungus kaip SM – galima suvesti informaciją apie egzistuojančio (turi būti įvesta informacija TFS sistemoje) vartotojo scenarijaus užduočių balsavimo rezultatus. TFS sistemoje yra saugoma informacija tik apie konkrečių užduočių planuojamą atlikimo laiką, tačiau nėra galimybės įvesti individualių programuotojų balsavimo rezultatų. Prisijungus kaip PO – galima peržiūrėti egzistuojančius vartotojų scenarijus, pažiūrėti planuotą atlikimo laiką ir faktinį. Taip pat realizuota galimybė sukurti naują vartotojo scenarijų (jis taip pat turi egzistuoti TFS sistemoje) papildomai įvedant funkcinį taškų skaičių. Pasirinkus žymę, jog vartotojo scenarijus yra panašus į anksčiau buvusius – atsiveria langas, kuriame galima pažymėti net keletą vartotojo scenarijų. Grafinės vartotojo sąsajos yra pateiktos 9 paveiksle.

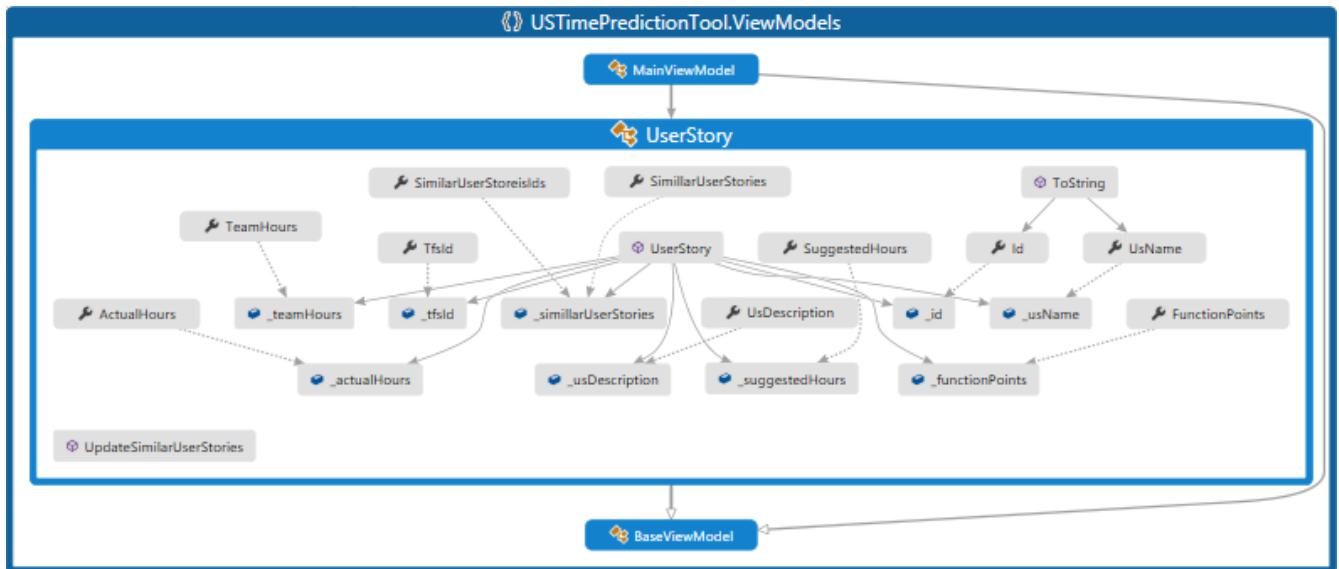


9 pav. Atvaizdavimo struktūra

SM rolė yra reikalinga tik tam, kad būtų galima pažymėti individualius programuotojų balsavimo rezultatus, nes visą kitą informaciją galima gauti iš TFS sistemos. Siekiant išsiaiškinti individualių programuotojų balsavimų rezultatus ir jų vėlesnį palyginimą su realiais duomenimis, galima įvesti rezultatus pasirinkus SM rolę ir sekti nuokrypį.

### 2.5.2.2 Atvaizdavimo modelis (ViewsModel)

Šiame skyriuje apžvelgiami pagrindiniai atvaizdavimo modelio komponentai, kurių struktūra pateikta 10 paveiksle.



10 pav. Atvaizdavimo modelio struktūra

Bazinis atvaizdavimo modelio komponentas apima informaciją apie pavienius atvaizdavimo modelius ir konkrečius vartotojo scenarijus. Šis komponentas yra atsakingas už tai, kad aktuali informacija pasiektų reikalingas įrankio dalis. Vartotojo scenarijaus komponentas užtikrina, kad programos paleidimo metu visa naujausia informacija susijusi su vartotojo scenarijais būtų išsaugota atmintyje ir atlikus pakeitimus apie tai informuotų atvaizdavimo modelius. Naudojant MVVM architektūrą pavyko sėkmingai atskirti dalis, kurios yra atsakingos už logiką ir kurios skirtos pasikeitusios informacijos atvaizdavimui.

## 2.6 Sistemos veikimo aprašymas

Paleidus programą visų pirma reikia pasirinkti rolę (SM arba PO) ir paspausti atitinkamą mygtuką. SM grafinė vartotojo sąsaja suteikia galimybę įvesti individualius programuotojų komandos balsavimo rezultatus, kurie yra išsaugomi duomenų bazėje. Vėliau, kai vartotojo scenarijus jau yra įgyvendintas, galima pažiūrėti, kaip skiriasi individualių programuotojų balsavimo rezultatai palyginus su esama situacija. PO rolė turi daugiau galimybių: peržiūrėti atliktus vartotojo scenarijus, įvesti naują ir susieti jį su jau atliktais vartotojo scenarijais. Pažymėjus, jog vartotojo scenarijus yra panašus į anksčiau buvusį, funkcinio taško įvertis yra apskaičiuojamas išvedant vidurkį iš istorinių vartotojo scenarijaus atlikimo duomenų. Siekiant, kad prognozė būtų kuo tikslesnė, pirmas tris iteracijas yra kaupiami duomenys apie planuojamų atlikti užduočių laiką ir jų realią trukmę. Praėjus trims iteracijoms, įrankis jau gali pateikti prognozę, kiek laiko užtruks atlikti vartotojo scenarijų, įvedus tik funkcinio taškų skaičių.

## 2.7 Testavimo modelis

Įgyvendinus iškeltus reikalavimus ir pabaigus įrankio kūrimą buvo paruoštas testavimo planas:

- programinio kodo padengimas testais (*angl. Unit Test*);
- ne mažiau kaip penkių vartotojų scenarijų realių duomenų sukaupimas;
- duomenų įvedimas į sistemą;
- naujo vartotojo scenarijaus kūrimas ir palyginimas su realiais duomenimis;
- naujo, panašaus į ankstesnį, vartotojo scenarijaus kūrimas ir palyginimas su realiais duomenimis;
- testavimo rezultatų apžvalga ir įrankio įvertinimas.

## 2.8 Testavimo duomenys ir rezultatai

Šiame skyriuje apžvelgiami testavimo rezultatai ir įvertinamas įrankio tinkamumas naudoti.

Pirmų trijų iteracijų istoriniai duomenys buvo išsaugoti duomenų bazėje. Vedant likusiųjų iteracijų duomenis nebuvo nurodytas realus užduočių atlikimo laikas (nors jis ir buvo žinomas). 2 lentelėje pateikiama informacija apie vartotojo scenarijus, programuotojų komandos pateiktus laikinius įverčius, faktinį atlikimo laiką ir įrankio pateiktą prognozę.

**2 lentelė.** Testavimo rezultatų analizė

Vartotojo scenarijaus Nr.	Bendras programuotojų įvertis (val.)	Faktinis atlikimo laikas (val.)	Ar scenarijus panašus į ankstesnį?	Įrankio pateikta prognozė (val.)
15827	85	128	Taip	94
15828	47	65	Ne	72
15829	78	70	Ne	85
15830	96	134	Ne	123
15831	112	148	Ne	137

Įrankis buvo testuojamas su penkių vartotojo scenarijų duomenų rinkiniu. Apžvelgus testavimo duomenis, paaiškėjo, kad pateikiama prognozė yra artimesnė faktiniam atlikimo laikui, nei bendras programuotojų įvertis.

### **3 TYRIMO IR EKSPERIMENTINĖ DALIS**

#### **3.1 Eksperimento planas**

Eksperimentui buvo pasirinkta viena programuotojų komanda, kurią sudaro 5 programuotojai. Surinkti pusės metų istoriniai duomenys apie vartotojų scenarijus: pateiktas bendras komandos narių darbo imlumo įvertis pagrindinės planavimo stadijos metu, bei faktinis vartotojų scenarijų atlikimo laikas. Taip pat paprašyta dviejų produktų šeimininkų naudojantis įrankiu priskirti funkcinių taškų skaičių kiekvienam iš vartotojų scenarijų.

Per pusės metų laikotarpį pavyko sukaupti 60 vartotojų scenarijų istorinius duomenis, kurie naudojami trijuose skirtinguose tyrimo etapuose:

- 1 etapas – gauti informaciją apie funkcinius taškus iš produktų šeimininkų apie vartotojų scenarijus. Numatomi veiksmai: išanalizuoti duomenis, pateikti rezultatus ir atlikti įrankio korekciją, jeigu pateikiama įrankio prognozė yra mažiau teisinga nei bendras programuotojų komandos narių darbo imlumo įvertis;
- 2 etapas – pakartoti eksperimentą eliminuojant tokius istorinius duomenis, kur programuotojų komandos narių balsavimo įvertis turėjo didesnę nei 50 procentų paklaidą. Esant poreikiui pakeisti įrankio veikimą taip, kad prognozė būtų kuo artimesnė faktiniam vartotojų scenarijų atlikimo laikui;
- 3 etapas – pakartotinai surinkti informaciją apie vartotojų scenarijų funkcinių taškų įverčius iš produktų šeimininkų. Apžvelgti rezultatus ir pateikti idėjas tolimesniam įrankio tobulinimui.

Šiame skyriuje apžvelgiami eksperimentinio tyrimo rezultatai po kiekvieno iš trijų etapų. Siekiamas rezultatas – įrankio pateikiama prognozė apie vartotojo scenarijaus atlikimo laiką ankstyvoje planavimo stadijoje, įvedus tik funkcinių taškų skaičių pagal pateiktą vartotojo scenarijaus aprašymą, kuri būtų tikslesnė už bendrą programuotojų komandos darbo imlumo įvertį ir kuo artimesnė faktiniam vartotojų scenarijų atlikimo laikui. Eksperimento metu įrankis gali būti koreguojamas, atsižvelgiant į skirtingų tyrimo etapų rezultatus.

#### **3.2 Eksperimento rezultatai**

Šioje darbo dalyje atliekami trys eksperimentinio tyrimo etapai ir pateikiami jų rezultatai. Atlikus kiekvieno tyrimo etapo rezultatų analizę buvo nuspręsta kokie pakeitimai reikalingi įrankiui, kad pateikiama prognozė būtų kuo artimesnė faktiniam vartotojų scenarijų atlikimo laikui.



### 3.2.1 Etapas Nr. 1

Pirmojo etapo metu du produktų šeimininkai įvertino vartotojų scenarijus, kiekvienam iš jų priskirdami funkcinių taškų įvertį. Vienas funkcinio taško įvertis buvo apskaičiuotas pagal žemiau pateiktą formulę, kur *val* – faktinis vartotojo scenarijaus atlikimo laikas.

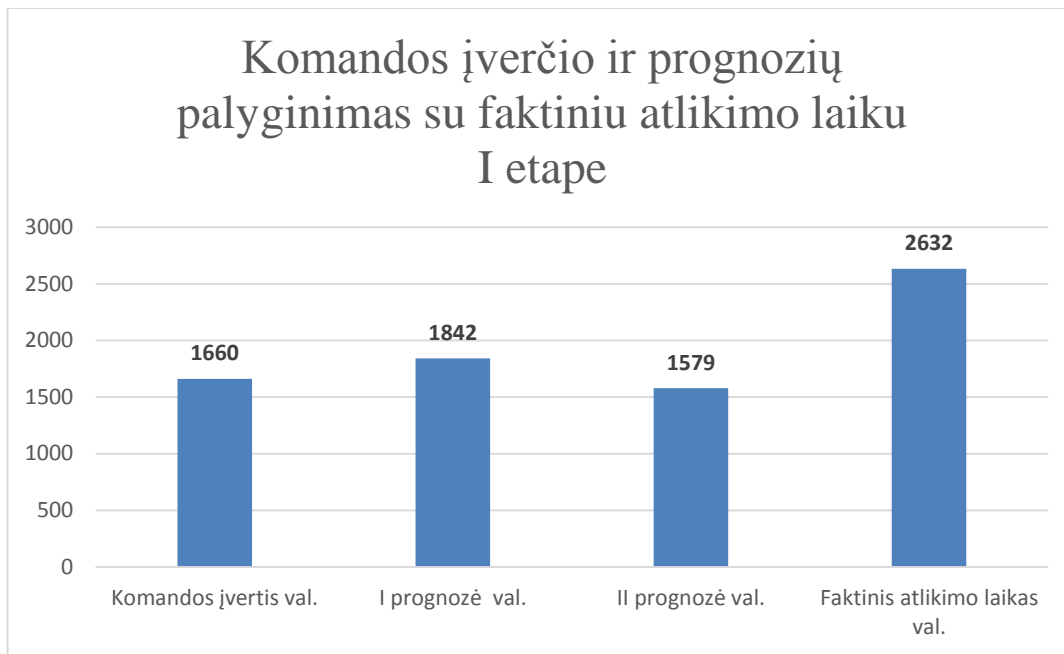
$$1FT \text{ įvertis} = \sum val / \sum FT$$

Svarbu pastebėti, kad tik vienas iš produktų šeimininkų yra susipažinęs su COSMIC metodu bei funkcinių taškų skaičiavimo principais, todėl produktų šeimininkų nuomonė apie funkcinių taškų skaičių priklausantį individualiems vartotojų scenarijams gali stipriai skirtis.

Svarbiausi klausimai, į kuriuos buvo siekiama atsakyti pirmojo eksperimentinio tyrimo etapo metu:

- Ar COSMIC metodo naudojimas Agile projektuose padeda tiksliau suplanuoti darbo laiką? T.y. ar pateikta prognozė yra artimesnė faktiniam atlikimo laikui, nei bendras programuotojų komandos narių laikinis įvertis?
- Ar pateiktos prognozės bendras nuokrypis yra mažesnis, nei programuotojų komandos laikinis įvertis, lyginant su faktiniu vartotojų scenarijų atlikimo laiku?
- Kokie pakeitimai reikalingi įrankiui, siekiant gauti geresnius rezultatus (prognozė kuo artimesnė faktiniam atlikimo laikui)?

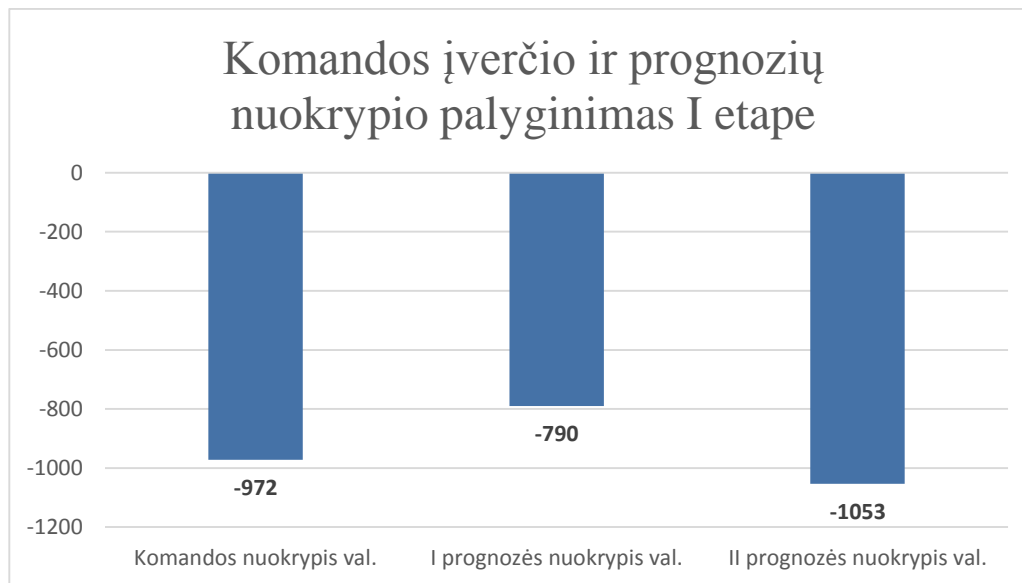
11 paveiksle pateikiama informacija apie visų vartotojų scenarijų bendrą atlikimo laiką (faktinis atlikimo laikas *val.*), įrankio pateikiamas prognozes, funkcinis taškus priskiriant skirtingiems produktų šeimininkams, bei programuotojų komandos prognozuojamą laikinį įvertį. Akivaizdu, kad niekam iš eksperimento dalyvių nepavyko tiksliai nuspėti, kiek laiko užtruks atlikti vartotojų scenarijus, tačiau pirmajam produkto šeimininkui priskiriant funkcinis taškus pavyko gauti prognozę, kuri yra artimesnė faktiniam vartotojų scenarijų atlikimo laikui.



**11 pav.** Komandos įverčio ir dviejų prognozių palyginimas su faktiniu US atlikimo laiku

I ir II prognozės yra pateikiamos gavus produktų šeiminių įverčius apie funkcinių taškų skaičių kiekvienam iš vartotojo scenarijų.

12 paveiksle pateikta informacija apie kiekvieno iš eksperimento dalyvių gautą bendrą nuokrypį lyginant su faktiniu vartotojų scenarijų atlikimo laiku.



**12 pav.** Komandos įverčio ir prognozių nuokrypio palyginimas

Priešasčių, kodėl prognozės taip stipriai skiriasi yra ne viena, tačiau pagrindinė problema išryškėjusi studijuojant eksperimento rezultatus – istoriniuose duomenyse gali būti didelių paklaidų. Pvz., vartotojo

scenarijus išties buvo sudėtingesnis, nei pradžioje atrodė programuotojų komandai, todėl ji skyrė ženkliai per mažą įvertinimą. Vartotojo scenarijus turi tik aprašymą, pagal kurį yra skiriami programuotojų komandos įvertinimai. Pagal tą patį aprašymą yra skiriami ir funkciniai taškai. Jeigu vartotojo scenarijus buvo aprašytas netinkamai ar nepilnai, tai gali sutrukdyti produkto šeimininkui įvesti teisingą funkcinių taškų skaičių. Esant didesnei nei 50 proc. paklaidai, tokius istorinius duomenis būtų galima eliminuoti, siekiant turėti tikslesnį funkcinio taško įvertį.

6.1 priede pateikiama pilna informacija apie pirmojo tyrimo etapo rezultatus. Net 21,67 proc. vartotojų scenarijų (pažymėti raudona spalva) buvo pervertinti (kai paklaida viršijo 50 proc. ribą) programuotojų komandos narių, todėl galima daryti prielaidą, jog pašalinus nekorektiškus istorinius duomenis, prognozuojamas vartotojo scenarijaus atlikimo laikas taps artimesnis faktiniam atlikimo laikui.

Analizuojant atskirus vartotojų scenarijus pastebėta, kad:

- pirmam produkto šeimininkui skiriant funkcinius taškus net 70 proc. vartotojų scenarijų buvo įvertinti geriau (arčiau faktinio atlikimo laiko) nei programuotojų komandos įvertis;
- antrojo produkto šeimininko atveju šis įvertinimas sumažėjo iki 43,34 proc.;
- 26,92 proc. blogiau prognozuotų vartotojų scenarijų buvo tiesiogiai susiję su nekorektiškais aktualaus vartotojo scenarijaus duomenimis (pvz. netikslus vartotojo scenarijaus aprašas).

### **3.2.2 Etapas Nr. 2**

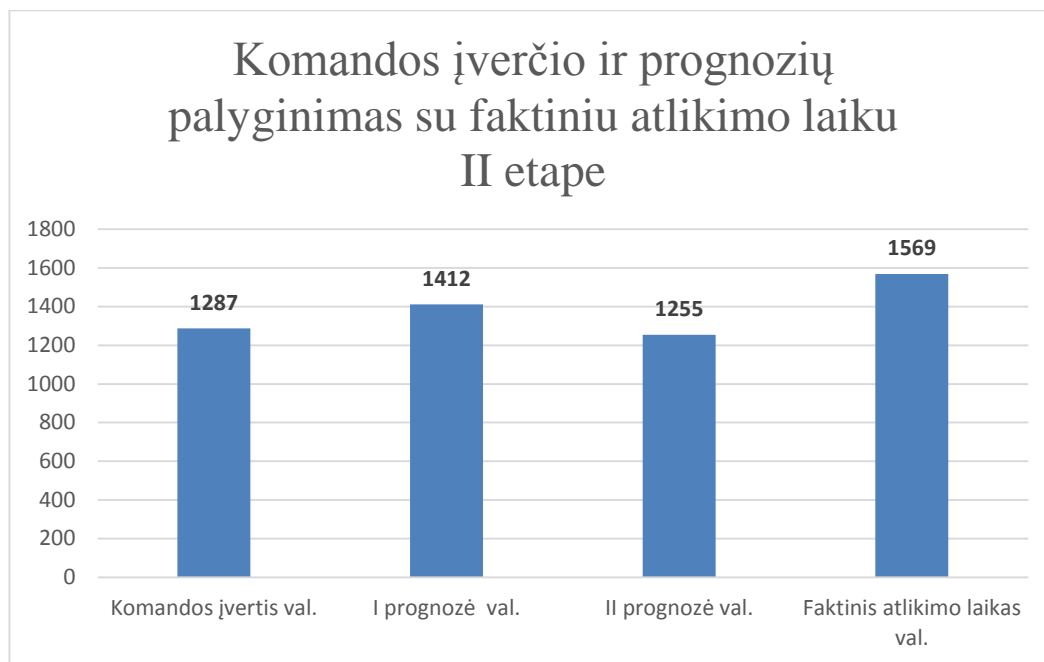
Atsižvelgiant į pirmojo tyrimo etapo eigą ir gautas analizės išvadas buvo nuspręsta pakoreguoti įrankį. Kai vartotojo scenarijaus faktinis atlikimo laikas skiriasi nuo bendro komandos narių įverčio daugiau nei 50 proc., istoriniai duomenys bus eliminuojami. Peržiūrėjus vartotojų scenarijų duomenis, paaiškėjo, kad eksperimento apimtis sumažėjo nuo 60 iki 47. Antrojo etapo metu eksperimentas bus pakartojamas su visais likusiais vartotojų scenarijais.

Svarbiausi klausimai, į kuriuos bus siekiama atsakyti antrojo etapo metu:

- Ar ekstremumų pašalinimas padeda gauti teisingesnę prognozę nei bendras programuotojų komandos narių laikinis įvertis?
- Ar pateikta prognozė yra artimesnė faktiniam vartotojų scenarijų atlikimo laikui?
- Kaip galima patobulinti įrankį, siekiant gauti geresnius rezultatus?

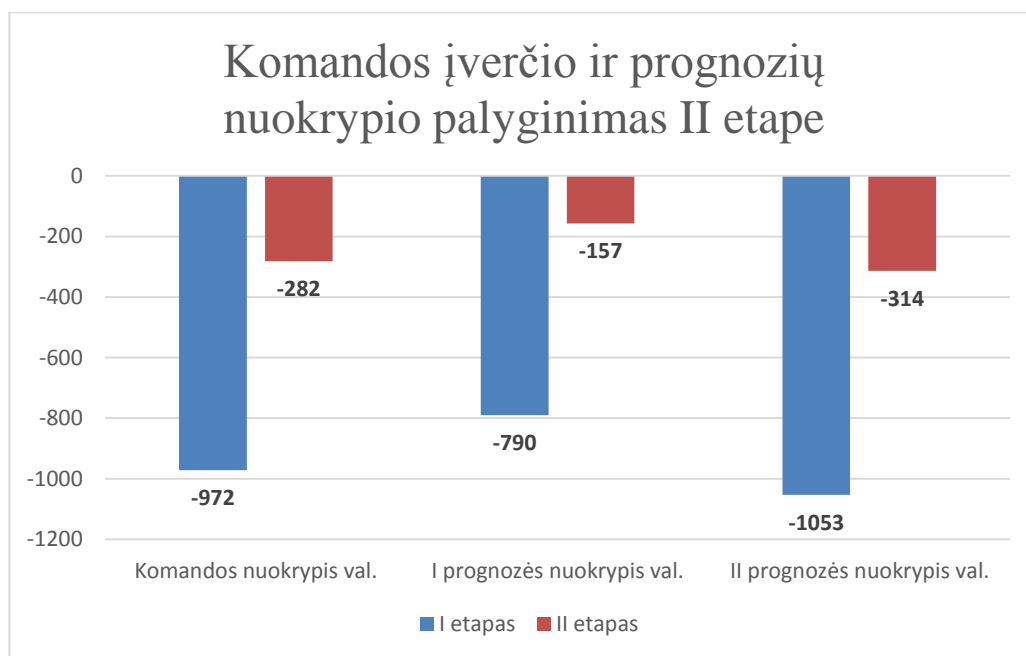
13 paveiksle pateikta informacija apie visų vartotojų scenarijų bendrą atlikimo laiką (faktinis atlikimo laikas val.), įrankio pateikiamas prognozes, funkcinius taškus priskiriant skirtingiems produktų

šeimininkams, bei programuotojų komandos prognozuojamą laikinį įvertį. Galima pastebėti, kad niekam nepavyko tiksliai nuspėti kiek laiko užtruks atlikti vartotojų scenarijus, tačiau šio etapo metu pastebėtas ženklus visų eksperimento dalyvių tikslesnis (artimesnis faktiniam atlikimo laikui) laiko prognozavimas.



**13 pav.** Komandos įverčio ir dviejų prognozių palyginimas su faktiniu US atlikimo laiku

14 paveiksle pateikta informacija apie pirmojo ir antrojo etapo eksperimento dalyvių nuokrypį nuo faktinio vartotojų scenarijų atlikimo laiko, išreikštą valandomis.



**14 pav.** Komandos įverčio ir prognozių nuokrypio palyginimas I ir II etapuose

Analizuojant antrojo etapo rezultatus buvo pastebėta, kad:

- programuotojų komandai pavyko 29,01 proc. tiksliau nustatyti vartotojų scenarijų faktinį atlikimo laiką, nei I etape;
- I prognozės rezultatas pagerėjo net 80,12 proc., tačiau įvertinus vartotojų scenarijus individualiai, pateiktas darbo imlumo įvertis valandomis, tapo 30,95 proc. tolimesnis faktiniam vartotojų scenarijų atlikimo laikui (lyginant su programuotojų komandos nuomone);
- II prognozės rezultatas pagerėjo net 70,18 proc.. Šis ženklus tiek I, tiek II prognozės pagerėjimas gali būti susijęs su ekstremumų pašalinimu, nes kitos eksperimentinio tyrimo sąlygos išliko nepakitusios;
- Vertinant abiejų prognozių tikslumą I ir II tyrimo etapuose, paaiškėjo, kad įrankis pateikia 2,75 proc. tikslesnę prognozę, nei programuotojų komanda.

Tyrimo metu buvo nustatyta, kad abiejuose etapuose tiek programuotojų komandos nariai, tiek produktų šeimininkai skirdami funkcinius taškus, neįvertino vartotojų scenarijų t.y. skyrė mažesnę valandų ar funkcinių taškų skaičių nei faktinis atlikimo laikas. Šis faktas yra visiškai natūralus, nes žmonės nežinodami konkrečių užduoties sąlygų, yra linkę manyti, kad sugebės ją atlikti greičiau [16].

6.2 priede pateikiama išsami informacija apie antrojo tyrimo etapo rezultatus.

Analizuojant atskirus vartotojų scenarijus buvo pastebėta, kad:

- I produkto šeimininkas skirdamas funkcinius taškus kiekvienam iš vartotojų scenarijų sugebėjo 44,32 proc. pagerinti prognozę, t.y. prognozė gauta naudojantis įrankiu buvo tikslesnė, nei programuotojų komandos nuomonė;
- II produkto šeimininkas šį rodiklį pablogino 10,19 proc.;
- pašalinus ekstremumus buvo gauti geresni rezultatai nei pirmajame etape, t.y. prognozė tapo artimesnė faktiniam atlikimo laikui.

Aptarus abiejų etapų rezultatus su produktų šeimininkais, paaiškėjo, kad didžiausias sunkumas, su kuriuo buvo susiduriama – tai nevisiškai aiškus COSMIC metodo funkcinių taškų priskyrimas. Buvo pasiūlyta idėja pakoreguoti įrankį taip, kad vietoj vieno skaičiaus, atitinkančio funkcinius taškus, būtų galima įvesti raidžių S, M ir L atitikmenis kiekvienai iš operacijų (įėjimas, išėjimas, rašymas ir skaitymas). Kiekviena iš raidžių identifikuoja operacijas pagal sudėtingumą (plačiau apžvelgta 3.2.3 skyriuje).

### 3.2.3 Etapas Nr. 3

Atsižvelgiant į antrojo etapo rezultatus, buvo atlikta įrankio korekcija pagal produktų šeimininkų pateiktus pasiūlymus. Rezultatų analizės metu paaiškėjo, kad produktų šeimininkams yra sudėtinga priskirti funkcinių taškų kiekį skaitine išraiška. Siekiant ne tik palengvinti jų darbą, bet ir gauti tikslesnę prognozę, buvo nuspręsta pakeisti funkcinių taškų įvedimą juos skirstant į tris sudėtingumo lygius (S, M ir L). Kiekviena iš operacijų (įėjimas, išėjimas, skaitymas, rašymas) nuo šiol gali būti įvertinama naudojant S, M ir L raides nurodant kiek lengvų, vidutinių ir sudėtingų operacijų turi kiekvienas iš vartotojų scenarijų. Toks funkcinių taškų priskyrimas turėtų padėti gauti tikslesnę prognozę.

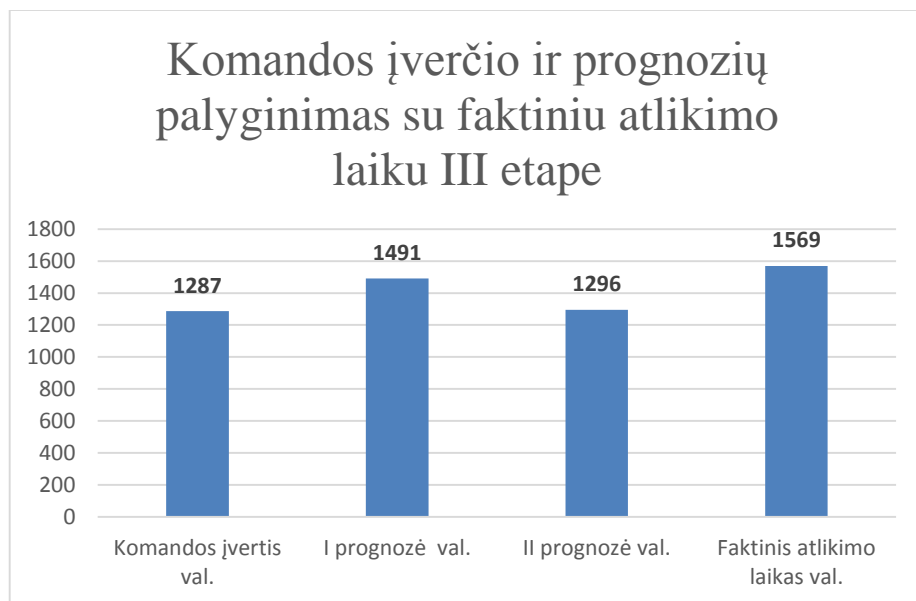
Atlikus įrankio korekciją funkcinius taškus galima įvesti naudojant raides, o vienas funkcinis taškas yra apskaičiuojamas pagal formulę, pateiktą žemiau, kur  $Op$  – viena iš operacijų, o S, M ir L reikšmės, kurios yra padaugintos iš koeficientų ir susumuojamos.

$$FT = \sum Op (1S + 2M + 3L)$$

Svarbiausi klausimai, į kuriuos bus siekiama atsakyti paskutiniojo tyrimo etapo metu:

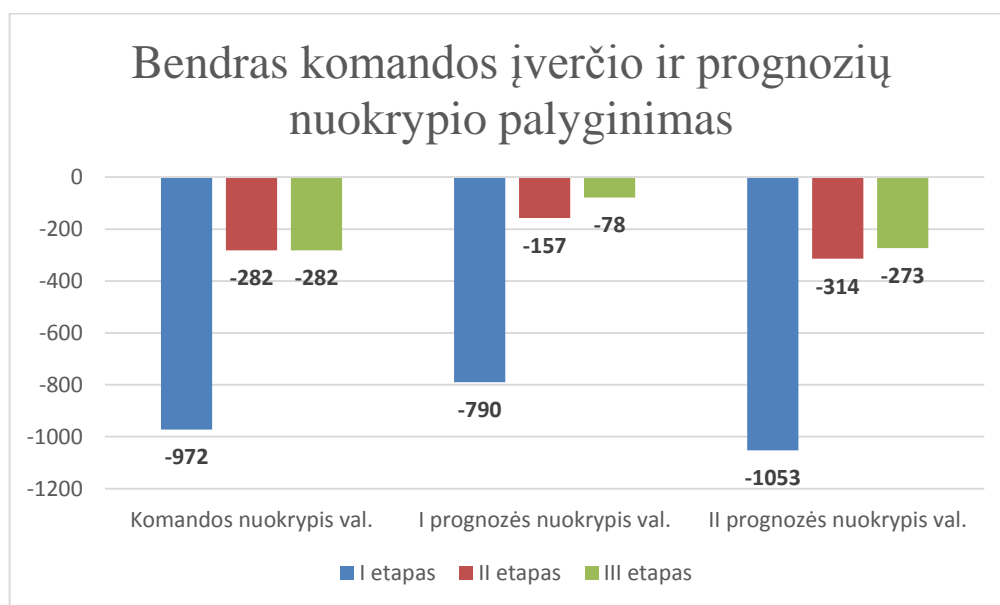
- Ar funkcinių taškų priskyrimas naudojant simbolius padeda gauti teisingesnę prognozę nei bendras programuotojų komandos narių laikinis įvertis?
- Ar pateikta prognozė yra artimesnė faktiniam vartotojų scenarijų atlikimo laikui?
- Kaip dar galima patobulinti įrankį, siekiant gauti geresnius rezultatus?

Pristačius įrankio pakeitimus produktų šeimininkams, buvo paprašyta įvertinti 47 vartotojų scenarijus iš naujo, šiems priskiriant raides S, M ir L.



**15 pav.** Komandos įverčio ir dviejų prognozių palyginimas su faktiniu US atlikimo laiku

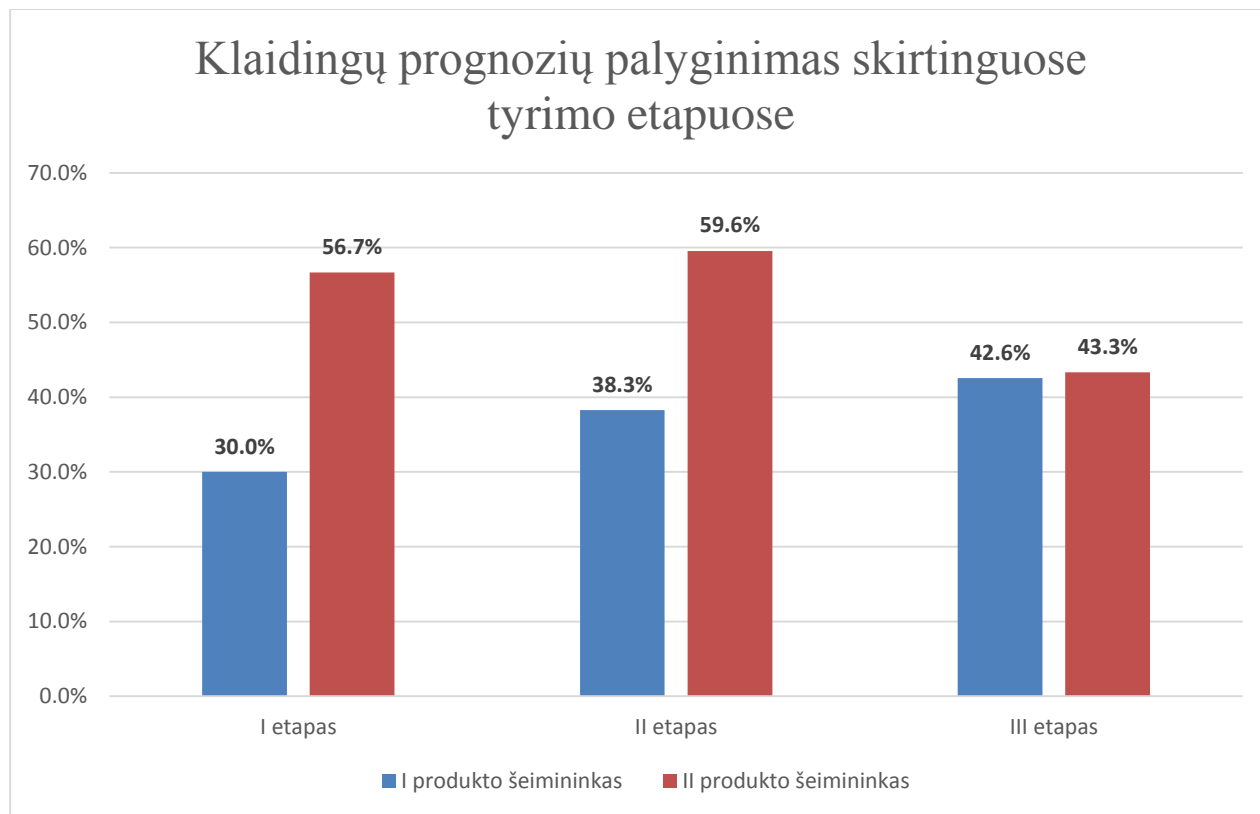
15 pav. Komandos įverčio ir dviejų prognozių palyginimas su faktiniu US atlikimo laikueiksle pavaizduotas programuotojų komandos ir prognozių palyginimas su faktiniu vartotojų scenarijų atlikimo laiku. Šio etapo metu abiejų produktų šeiminių pateikti funkciniai taškų įverčiai, buvo tikslesni už programuotojų komandos pateiktą įvertinimą. Visų trijų etapų bendro komandos įverčio ir prognozių nuokrypio palyginimas pateiktas 16 paveiksle.



**16 pav.** Komandos įverčio ir prognozių nuokrypio palyginimas visuose etapuose

Galima pastebėti, kad paskutiniojo etapo metu, įrankio pateikiama prognozė tapo ženkliai (37,59 proc.) artimesnė faktiniam vartotojų scenarijų atlikimo laikui.

17 pav. Prognozių palyginimas visuose tyrimo etapuose pavaizduota informacija apie klaidingų (tokių, kur įrankio pateikiama prognozė buvo mažiau teisinga lyginant su programuotojų komandos nuomone) vartotojų scenarijų įvertinimus skirtinguose tyrimo etapuose. Galima pastebėti, kad pirmajam produkto šeimininkui priskiriant funkcinis taškus individualiems vartotojų scenarijams, klaidingų prognozių daugėjo, tuo tarpu funkcinis taškus priskiriant antrajam produkto šeimininkui – trečiajame etape pavyko šį rodiklį šiek tiek sumažinti. Svarbu paminėti, kad bendras prognozės tikslumas (vertinant įrankio pateikiamą prognozę su faktiniu atlikimo laiku) gerėjo visuose trijuose tyrimo etapuose.



**17 pav.** Prognozių palyginimas visuose tyrimo etapuose

Aptarus visų trijų eksperimentinio tyrimo rezultatus su produktų šeimininkais buvo pasiūlyta įdomi idėja tolimesniam įrankio tobulinimui: automatizuotas funkcinis taškų priskyrimas pagal raktinius vartotojų scenarijų aprašymo žodžius. Apžvelgus visus eksperimento metu įvertintus vartotojų scenarijus, paaiškėjo, kad yra ganėtinai sudėtinga atskirti individualius aprašymo žodžius ir suteikti jiems svorinius koeficientus, nes:

- aprašymai yra nestructūrizuoti;
- sudaromi naudojant šnekamąją kalbą;
- žodžiai, kurie kartojasi, nėra raktiniai.



Žinoma, turint daugiau informacijos apie vartotojų scenarijus, šią idėją būtų galima realizuoti.

0 priede pateikiama išsami informacija apie trečiojo tyrimo etapo rezultatus. Analizuojant atskirus vartotojų scenarijus buvo pastebėta, kad:

- vartotojų scenarijai buvo įvertinti nepakankamai (t.y. nuokrypis nuo faktinio atlikimo laiko yra neigiamas), kaip ir pirmuosiuose tyrimo etapuose;
- naudojant sukurtą įrankį trečiojo etapo metu, buvo gauta 37,59 proc. tikslesnė prognozė, nei programuotojų komandos nuomonė, lyginant su faktiniu atlikimo laiku.

## 4 IŠVADOS

1. Yra įvairiausių darbo imlumo planavimo metodų bei įrankių skirtų planuoti ir sekti vėlesnį užduočių statusą vartotojų scenarijų realizavimo metu, tačiau nėra nei vieno įrankio, kuris būtų pritaikytas Agile projektams. Taip pat remtūsi konkrečios programuotojų komandos istoriniais užduočių atlikimo duomenimis, bei turėtų integruotą COSMIC metodą, skirtą gauti greitą ir tikslų darbo imlumo įvertinimą ankstyvoje planavimo stadijoje.
2. COSMIC metodo pritaikymas Agile projektuose svarbus dėl kelių priežasčių: žinant tikslų vartotojo scenarijaus atlikimo laiką galima atitinkamai planuoti resursus; programuotojų komandos nuomonės dėl potencialių vartotojo scenarijų nebereikia, remiamasi istoriniais duomenimis. Programuotojai gali netrukdomai dirbti prie einamųjų užduočių.
3. Darbo metu sukurtas įrankis, skirtas produktų šeimininkams įvertinti vartotojų scenarijus funkciniais taškais ir gauti prognozę pagal istorinius duomenis. Įrankis buvo testuotas su penkių vartotojo scenarijų duomenų rinkiniu.
4. Buvo atliktas eksperimentas, kurio metu realizuotos skirtingos idėjos siekiant gauti kuo tikslesnę prognozę:
  - 4.1. Pirmojo etapo metu du produktų šeimininkai įvertino 60 vartotojų scenarijų. Pastebėta, kad:
    - 4.1.1. pirmam produkto šeimininkui skiriant funkcinis taškus net 70 proc. vartotojų scenarijų buvo įvertinti geriau (arčiau faktinio atlikimo laiko) nei programuotojų komandos įvertis;
    - 4.1.2. antrojo produkto šeimininko atveju šis įvertinimas sumažėjo iki 43,34 proc.;
    - 4.1.3. 26,92 proc. blogiau prognozuotų vartotojų scenarijų buvo tiesiogiai susiję su nekorektiškais aktualaus vartotojo scenarijaus duomenimis.
  - 4.2. Antrojo etapo metu įrankio veikimas buvo pakoreguotas ir pašalinti tokie istoriniai duomenys, kur programuotojų komandos įvertinimas turėjo didesnę nei 50 proc. paklaidą. Vartotojų scenarijų skaičius sumažėjo iki 47 ir apžvelgus antrojo tyrimo etapo duomenis, paaiškėjo, kad:
    - 4.2.1. I prognozės rezultatas pagerėjo net 80,12 proc., tačiau įvertinus vartotojų scenarijus individualiai, pateiktas darbo imlumo įvertis valandomis, tapo 30,95 proc. tolimesnis faktiniam vartotojų scenarijų atlikimo laikui (lyginant su programuotojų komandos nuomone);
    - 4.2.2. II prognozės rezultatas pagerėjo net 70,18 proc. Šis ženklus tiek I, tiek II prognozės pagerėjimas gali būti susijęs su ekstremumų pašalinimu, nes kitos eksperimentinio tyrimo sąlygos išliko nepakitusios;

- 4.2.3. I produkto šeimininkas skirdamas funkcinius taškus kiekvienam iš vartotojų scenarijų sugebėjo 44,32 proc. pagerinti prognozę, t.y. prognozė gauta naudojantis įrankiu buvo tikslesnė, nei programuotojų komandos nuomonė;
- 4.2.4. II produkto šeimininkas šį rodiklį pablogino 10,19 proc.
- 4.3. Trečiojo etapo metu įrankis buvo pakoreguotas taip, kad funkcinių taškų įvedimui būtų galima naudoti S, M ir L raidžių atitikmenis, identifikuojančius operacijos sudėtingumą. Produktų šeimininkams iš naujo įvertinus 47 vartotojo scenarijus paaiškėjo, kad:
  - 4.3.1. įrankio pateikiama prognozė tapo 37,59 proc. artimesnė faktiniam vartotojų scenarijų atlikimo laikui, lyginant su programuotojų komandos įvertinimu;
  - 4.3.2. vartotojų scenarijai buvo įvertinti nepakankamai (t.y. nuokrypis nuo faktinio atlikimo laiko yra neigiamas), kaip ir pirmuosiuose tyrimo etapuose.
- 5. Buvo pasiūlyta idėja tolimesniam įrankio tobulinimui – automatizuotas funkcinių taškų priskyrimas pagal raktinius žodžius. Apžvelgus eksperimento duomenis, paaiškėjo, kad ją būtų galima įgyvendinti nebent turint didesnę duomenų imtį.

## 5 LITERATŪROS SĄRAŠAS

- [1] K. Waters, „All about Agile,“ 10 vasario 2007. [Tinkle]. Available: <http://www.allaboutagile.com/what-is-agile-10-key-principles/>. [Kreiptasi 19 sausio 2013].
- [2] „Agile Alliance,“ 2001. [Tinkle]. Available: <http://www.agilealliance.org/the-alliance/the-agile-manifesto/>. [Kreiptasi 5 rugsėjo 2013].
- [3] A. L. „Coaching Agile Teams,“ įtraukta *Coaching Agile Teams*, Addison-Wesley, 2010, p. 315.
- [4] W. S. Shore J., *The Art of Agile Development*, O'REILLY, 2008.
- [5] Hughes B., Cotterell M., *Software Project Management*, Mc Graw Hill, 1999, p. 384.
- [6] J. Sutherland, „Scruminc,“ 30 September 2012. [Tinkle]. Available: <http://scrum.jeffsutherland.com/2010/04/story-points-why-are-they-better-than.html>. [Kreiptasi 17 sausio 2013].
- [7] „Overview of the COSMIC Functional Size Measurement Method,“ [Tinkle]. Available: <http://www.cosmicon.com/methodV3.asp>. [Kreiptasi 30 kovo 2014].
- [8] Thomas Fehlmann, Luca Santillo, „From Story Points to COSMIC Function Points in Agile Software Development – A Six Sigma perspective,“ 2010. [Tinkle]. Available: <http://www.cosmicon.com/portal/public/FromStoryPointsToCOSMIC.pdf>. [Kreiptasi 17 sausio 2013].
- [9] Jean-Marc Desharnais, Buğra Kocatürk, Luigi Buglione, „Improving Agile Software Project Planning using the COSMIC Method,“ [Tinkle]. Available: [https://cours.etsmtl.ca/mgl801/private/R%C3%A9f%C3%A9rences%20et%20articles/Articles/INFSOF-COSMIC-Agile\\_v12lb.pdf](https://cours.etsmtl.ca/mgl801/private/R%C3%A9f%C3%A9rences%20et%20articles/Articles/INFSOF-COSMIC-Agile_v12lb.pdf). [Kreiptasi 17 sausio 2013].
- [10] J. Fields, „User Story Estimation Techniques,“ 30 June 2008. [Tinkle]. Available: <http://www.infoq.com/articles/agile-estimation-techniques>. [Kreiptasi 17 sausio 2013].
- [11] „iceScrum,“ 2011. [Tinkle]. Available: <http://www.icescrum.org/en/>. [Kreiptasi 25 sausio 2013].

- [12] „Simple and powerful AGILEFLANT,“ 2013. [Tinkle]. Available: <http://agilefant.com/>. [Kreiptasi 29 sausio 2013].
- [13] „XPlanner,“ 2006. [Tinkle]. Available: <http://xplanner.codehaus.org/>. [Kreiptasi 29 sausio 2013].
- [14] „Agilo software,“ 2013. [Tinkle]. Available: <http://agilosoftware.com/>. [Kreiptasi 29 sausio 2013].
- [15] „XPStoryStudio,“ 2004. [Tinkle]. Available: <http://www.agile-tools.net/agileprojectmanagement/XPStoryStudio.aspx>. [Kreiptasi 29 sausio 2013].
- [16] Kahneman, Daniel; Tversky, Amos, „Intuitive prediction: biases and corrective procedures,“ 1979, p. 313–327.

## 6 PRIEDAI

### 6.1 Pirmo tyrimo etapo rezultatai

Nr.	US Nr.	Balsu otų val. sk.	Fakti nis val. sk.	Nuok rypis val.	Paklaida %	I produkto šeimininkas					II produkto šeimininkas				
						FT	Prog nozė val.	Nuok rypis val.	Paklaida %	Ar geresnė prognozė?	FT	Prognozė val.	Nuokryp is val.	Paklaida %	Ar geresnė prognozė?
1	81022	10	33	-23	70	5	19	-14	42	T	10	24	-9	26	T
2	90974	29	30	-1	3	8	31	1	3	T	9	22	-8	27	N
3	90916	15	16	-1	6	3	12	-4	28	N	7	17	1	6	T
4	90921	51	97	-46	47	15	58	-39	40	T	28	68	-29	30	T
5	88180	16	80	-64	80	10	39	-41	52	T	5	12	-68	85	N
6	88949	19	30	-11	37	5	19	-11	36	T	15	36	6	21	T
7	88948	17	27	-10	37	6	23	-4	14	T	13	32	5	17	T
8	88950	29	32	-3	9	4	15	-17	52	N	12	29	-3	9	T
9	88951	13	40	-27	68	4	15	-25	61	T	4	10	-30	76	N
10	88952	12	8	4	50	2	8	0	4	T	3	7	-1	9	T
11	88953	22	18	4	22	5	19	1	7	T	6	15	-3	19	T
12	87158	26	26	0	0	4	15	-11	41	N	5	12	-14	53	N
13	87159	8	8	0	0	2	8	0	4	N	3	7	-1	9	N
14	87157	10	12	-2	17	3	12	0	4	T	4	10	-2	19	N
15	83440	37	26	11	42	4	15	-11	41	T	8	19	-7	25	T
16	83442	31	17	14	82	4	15	-2	9	T	15	36	19	114	N
17	83449	11	10	1	10	2	8	-2	23	N	4	10	0	3	T
18	83448	21	17	4	24	4	15	-2	9	T	5	12	-5	29	N
19	83450	77	66	11	17	15	58	-8	12	T	8	19	-47	71	N
20	83443	13	15	-2	13	4	15	0	3	T	5	12	-3	19	N
21	83113	55	63	-8	12.70	18	69	6	10	T	16	39	-24	38	N
22	82378	16	9	7	77.78	2	8	-1	14	T	2	5	-4	46	T
23	82379	18	18	0	0.00	7	27	9	50	N	11	27	9	48	N
24	82380	10	11	-1	9.09	3	12	1	5	T	3	7	-4	34	N

Nr.	US Nr.	Balsu otų val. sk.	Fakti nis val. sk.	Nuok rypis val.	Paklaida %	I produkto šeimininkas					II produkto šeimininkas				
						FT	Prog nozė val.	Nuok rypis val.	Paklaida %	Ar geresnė prognozė?	FT	Prognozė val.	Nuokryp is val.	Paklaida %	Ar geresnė prognozė?
25	82381	13	20	-7	35.00	4	15	-5	23	T	7	17	-3	15	T
26	82382	11	12	-1	8.33	4	15	3	28	N	3	7	-5	39	N
27	82383	25	23	2	8.70	7	27	4	17	N	7	17	-6	26	N
28	82384	17	14	3	21.43	5	19	5	38	N	4	10	-4	31	N
29	82385	18	15	3	20.00	4	15	0	3	T	5	12	-3	19	T
30	82386	30	38	-8	21.05	15	58	20	52	N	11	27	-11	30	N
31	81020	32	70	-38	54.29	10	39	-31	45	T	8	19	-51	72	N
32	81024	38	46	-8	17.39	15	58	12	26	N	13	32	-14	31	N
33	81022	16	33	-17	51.52	5	19	-14	42	T	5	12	-21	63	N
34	81023	12	6	6	100.00	2	8	2	28	T	2	5	-1	19	T
35	81026	10	10	0	0.00	3	12	2	16	N	4	10	0	3	N
36	80489	22	14	8	57.14	4	15	1	10	T	2	5	-9	65	N
37	80490	25	28	-3	10.71	8	31	3	10	T	8	19	-9	31	N
38	79818	10	9	1	11.11	3	12	3	28	N	3	7	-2	19	N
39	79819	48	52	-4	7.69	17	65	13	26	N	15	36	-16	30	N
40	78671	42	80	-38	47.50	11	42	-38	47	T	15	36	-44	55	N
41	78685	45	75	-30	40.00	21	81	6	8	T	13	32	-43	58	N
42	74392	56	140	-84	60.00	30	116	-24	17	T	16	39	-101	72	N
43	76885	55	249	-194	77.91	20	77	-172	69	T	32	78	-171	69	T
44	77039	62	231	-169	73.16	22	85	-146	63	T	26	63	-168	73	T
45	76881	69	93	-24	25.81	18	69	-24	25	T	33	80	-13	14	T
46	76882	26	48	-22	45.83	16	62	14	28	T	15	36	-12	24	T
47	77055	25	36	-11	30.56	10	39	3	7	T	15	36	0	1	T
48	74392	32	141	-109	77.30	10	39	-102	73	T	30	73	-68	48	T
49	76315	29	29	0	0.00	5	19	-10	34	N	12	29	0	0	N
50	76317	8	8	0	0.00	2	8	0	4	T	4	10	2	21	N
51	76316	16	17	-1	5.88	5	19	2	13	N	8	19	2	14	N

Nr.	US Nr.	Balsu otų val. sk.	Fakti nis val. sk.	Nuok rypis val.	Paklaida %	I produkto šeimininkas					II produkto šeimininkas				
						FT	Prog nozė val.	Nuok rypis val.	Paklaida %	Ar geresnė prognozė?	FT	Prognozė val.	Nuokryp is val.	Paklaida %	Ar geresnė prognozė?
52	75187	22	16	6	37.50	3	12	-4	28	T	5	12	-4	24	T
53	75188	48	74	-26	35.14	13	50	-24	32	T	30	73	-1	2	T
54	75189	15	15	0	0.00	4	15	0	3	T	7	17	2	13	N
55	75190	49	68	-19	27.94	13	50	-18	26	T	29	70	2	3	T
56	75191	23	23	0	0.00	7	25	2	9	N	11	27	4	16	N
57	75192	18	18	0	0.00	5	19	1	7	N	8	19	1	8	T
58	74357	58	79	-21	26.58	16	62	-17	22	T	32	78	-1	2	T
59	74360	61	83	-22	26.51	16	62	-21	26	T	35	85	2	2	T
60	74354	8	10	-2	20.00	3	12	2	16	T	3	7	-3	27	N
Viso:		1660	2632	-972			1842	-790				1579	-1053		

## 6.2 Antro tyrimo etapo rezultatai

Nr.	US Nr.	Balsu otų val. sk.	Fakti nis val. sk.	Nuok rypis val.	Paklaida %	I produkto šeimininkas					II produkto šeimininkas				
						FT	Prog nozė val.	Nuok rypis val.	Paklaida %	Ar geresnė prognozė?	FT	Prognozė val.	Nuokryp is val.	Paklaida %	Ar geresnė prognozė?
1	81022	10	33	-23	70										
2	90974	29	30	-1	3	8	31	1	3	T	9	22	-8	28	N
3	90916	15	16	-1	6	3	12	-4	28	N	7	17	1	6	T
4	90921	51	97	-46	47	15	58	-39	40	T	28	68	-29	30	T
5	88180	16	80	-64	80										
6	88949	19	30	-11	37	5	19	-11	36	T	15	36	6	21	T
7	88948	17	27	-10	37	6	23	-4	14	T	13	31	4	16	T
8	88950	29	32	-3	9	4	15	-17	52	N	12	29	-3	9	N



Nr.	US Nr.	Balsu otų val. sk.	Fakti nis val. sk.	Nuok rypis val.	Paklaida %	I produkto šeimininkas					II produkto šeimininkas				
						FT	Prog nozė val.	Nuok rypis val.	Paklaida %	Ar geresnė prognozė?	FT	Prognozė val.	Nuokryp is val.	Paklaida %	Ar geresnė prognozė?
9	88951	13	40	-27	68										
10	88952	12	8	4	50	2	8	0	4	T	3	7	-1	9	T
11	88953	22	18	4	22	5	19	1	7	T	6	14	-4	20	T
12	87158	26	26	0	0	4	15	-11	41	N	5	12	-14	54	N
13	87159	8	8	0	0	2	8	0	4	N	3	7	-1	9	N
14	87157	10	12	-2	17	3	12	0	4	T	4	10	-2	20	N
15	83440	37	26	11	42	4	15	-11	41	T	8	19	-7	26	T
16	83442	31	17	14	82										
17	83449	11	10	1	10	2	8	-2	23	N	4	10	0	3	T
18	83448	21	17	4	24	4	15	-2	9	T	5	12	-5	29	N
19	83450	77	66	11	17	15	58	-8	12	T	8	19	-47	71	N
20	83443	13	15	-2	13	4	15	0	3	T	5	12	-3	20	N
21	83113	55	63	-8	12.70	18	69	6	10	T	16	39	-24	39	N
22	82378	16	9	7	77.78										
23	82379	18	18	0	0.00	7	27	9	50	N	11	27	9	48	N
24	82380	10	11	-1	9.09	3	12	1	5	T	3	7	-4	34	N
25	82381	13	20	-7	35.00	4	15	-5	23	T	7	17	-3	16	T
26	82382	11	12	-1	8.33	4	15	3	28	N	3	7	-5	40	N
27	82383	25	23	2	8.70	7	27	4	17	N	7	17	-6	27	N
28	82384	17	14	3	21.43	5	19	5	38	N	4	10	-4	31	N
29	82385	18	15	3	20.00	4	15	0	3	T	5	12	-3	20	T
30	82386	30	38	-8	21.05	15	58	20	52	N	11	27	-11	30	N
31	81020	32	70	-38	54.29										
32	81024	38	46	-8	17.39	15	58	12	26	N	13	31	-15	32	N
33	81022	16	33	-17	51.52										
34	81023	12	6	6	100.00										
35	81026	10	10	0	0.00	3	12	2	16	N	4	10	0	3	N

Nr.	US Nr.	Balsu otų val. sk.	Fakti nis val. sk.	Nuok rypis val.	Paklaida %	I produkto šeiminkas					II produkto šeiminkas				
						FT	Prog nozė val.	Nuok rypis val.	Paklaida %	Ar geresnė prognozė?	FT	Prognozė val.	Nuokryp is val.	Paklaida %	Ar geresnė prognozė?
36	80489	22	14	8	57.14										
37	80490	25	28	-3	10.71	8	31	3	10	T	8	19	-9	31	N
38	79818	10	9	1	11.11	3	12	3	28	N	3	7	-2	20	N
39	79819	48	52	-4	7.69	17	65	13	26	N	15	36	-16	30	N
40	78671	42	80	-38	47.50	11	42	-38	47	T	15	36	-44	55	N
41	78685	45	75	-30	40.00	21	81	6	8	T	13	31	-44	58	N
42	74392	56	140	-84	60.00										
43	76885	55	249	-194	77.91										
44	77039	62	231	-169	73.16										
45	76881	69	93	-24	25.81	18	69	-24	25	T	33	80	-13	14	T
46	76882	26	48	-22	45.83	16	62	14	28	T	15	36	-12	25	T
47	77055	25	36	-11	30.56	10	39	3	7	T	15	36	0	1	T
48	74392	32	141	-109	77.30										
49	76315	29	29	0	0.00	5	19	-10	34	N	12	29	0	0	N
50	76317	8	8	0	0.00	2	8	0	4	T	4	10	2	21	N
51	76316	16	17	-1	5.88	5	19	2	13	N	8	19	2	14	N
52	75187	22	16	6	37.50	3	12	-4	28	T	5	12	-4	25	T
53	75188	48	74	-26	35.14	13	50	-24	32	T	30	72	-2	2	T
54	75189	15	15	0	0.00	4	15	0	3	T	7	17	2	13	N
55	75190	49	68	-19	27.94	13	50	-18	26	T	29	70	2	3	T
56	75191	23	23	0	0.00	7	25	2	9	N	11	27	4	15	N
57	75192	18	18	0	0.00	5	19	1	7	N	8	19	1	7	T
58	74357	58	79	-21	26.58	16	62	-17	22	T	32	77	-2	2	T
59	74360	61	83	-22	26.51	16	62	-21	26	T	35	84	1	2	T
60	74354	8	10	-2	20.00	3	12	2	16	T	3	7	-3	28	N
Viso:		1287	1569	-282			1412	-157				1255	-314		

### 6.3 Trečiojo tyrimo etapo rezultatai

Nr.	US Nr.	Balsu otų val. sk.	Faktin is val. sk.	Nu okr ypi s val.	Paklaid a %	I produkto šeimininkas					II produkto šeimininkas				
						FT	Prog nozė val.	Nuok rypis val.	Paklaida %	Ar geresnė prognozė?	FT	Prognozė val.	Nuokryp is val.	Paklaida %	Ar geresnė prognozė?
1	90974	29	30	-1	3	5	20	-10	35	N	9	22	-8	28	N
2	90916	15	16	-1	6	3	12	-4	27	N	7	17	1	5	T
3	90921	51	97	-46	47	15	59	-38	39	T	28	67	-30	31	T
4	88949	19	30	-11	37	5	20	-10	35	T	15	36	6	20	T
5	88948	17	27	-10	37	6	24	-3	13	T	13	31	4	15	T
6	88950	29	32	-3	9	8	31	-1	2	T	12	29	-3	10	N
7	88952	12	8	4	50	2	8	0	2	T	3	7	-1	10	T
8	88953	22	18	4	22	5	20	2	9	T	6	14	-4	20	T
9	87158	26	26	0	0	4	16	-10	40	N	11	26	0	1	N
10	87159	8	8	0	0	2	8	0	2	N	3	7	-1	10	N
11	87157	10	12	-2	17	3	12	0	2	T	4	10	-2	20	N
12	83440	37	26	11	42	7	27	1	5	T	11	26	0	1	T
13	83449	11	10	1	10	2	8	-2	22	N	4	10	0	4	T
14	83448	21	17	4	24	4	16	-1	8	T	5	12	-5	30	N
15	83450	77	66	11	17	15	59	-7	11	T	8	19	-47	71	N
16	83443	13	15	-2	13	4	16	1	4	T	5	12	-3	20	N
17	83113	55	63	-8	12.70	22	86	23	37	N	24	57	-6	9	T
18	82379	18	18	0	0.00	7	27	9	52	N	11	26	8	46	N
19	82380	10	11	-1	9.09	3	12	1	7	T	3	7	-4	35	N
20	82381	13	20	-7	35.00	6	24	4	18	T	7	17	-3	16	T
21	82382	11	12	-1	8.33	4	16	4	31	N	3	7	-5	40	N
22	82383	25	23	2	8.70	7	27	4	19	N	7	17	-6	27	N
23	82384	17	14	3	21.43	5	20	6	40	N	4	10	-4	32	N
24	82385	18	15	3	20.00	4	16	1	4	T	5	12	-3	20	N
25	82386	30	38	-8	21.05	15	59	21	55	N	13	31	-7	18	T

Nr.	US Nr.	Balsu otų val. sk.	Faktin is val. sk.	Nu okr ypis s val.	Paklaid a %	I produkto šeimininkas					II produkto šeimininkas				
						FT	Prog nozė val.	Nuok rypis val.	Paklaida %	Ar geresnė progozė?	FT	Prognozė val.	Nuokryp is val.	Paklaida %	Ar geresnė progozė?
26	81024	38	46	-8	17.39	15	59	13	28	N	13	31	-15	32	N
27	81026	10	10	0	0.00	3	12	2	18	N	4	10	0	4	N
28	80490	25	28	-3	10.71	8	31	3	12	N	11	26	-2	6	T
29	79818	10	9	1	11.11	3	12	3	31	N	3	7	-2	20	N
30	79819	48	52	-4	7.69	17	67	15	28	N	14	34	-18	36	N
31	78671	42	80	-38	47.50	15	59	-21	27	T	15	36	-44	55	N
32	78685	45	75	-30	40.00	21	82	7	10	T	13	31	-44	58	N
33	76881	69	93	-24	25.81	18	71	-22	24	T	33	79	-14	15	T
34	76882	26	48	-22	45.83	16	63	15	31	T	15	36	-12	25	T
35	77055	25	36	-11	30.56	10	39	3	9	T	15	36	0	0	T
36	76315	29	29	0	0.00	5	20	-9	32	N	12	29	0	1	N
37	76317	8	8	0	0.00	2	8	0	2	T	4	10	2	20	N
38	76316	16	17	-1	5.88	5	20	3	15	N	8	19	2	13	N
39	75187	22	16	6	37.50	3	12	-4	27	T	5	12	-4	25	T
40	75188	48	74	-26	35.14	13	51	-23	31	T	30	72	-2	3	T
41	75189	15	15	0	0.00	4	16	1	4	T	7	17	2	12	N
42	75190	49	68	-19	27.94	13	51	-17	25	T	29	69	1	2	T
43	75191	23	23	0	0.00	7	25	2	11	N	11	26	3	15	N
44	75192	18	18	0	0.00	5	20	2	9	N	8	19	1	6	T
45	74357	58	79	-21	26.58	16	63	-16	21	T	32	77	-2	3	T
46	74360	61	83	-22	26.51	16	63	-20	24	T	35	84	1	1	T
47	74354	8	10	-2	20.00	3	12	2	18	T	3	7	-3	28	N
Viso:		1287	1569	-			1491	-78				1296	-273		