

Scrum su Kanban naudojančios organizacijos programų sistemų kūrimo proceso vertinimas

Vaidotas Pėkis

Vilniaus universiteto Matematikos ir informatikos fakulteto I kurso magistrantas
Vilnius University, Faculty of Mathematics and Informatics, I year postgraduate student
Saulės g. 10-36, LT-26123 Elektrėnai
El. paštas: pekis.vaidotas@gmail.com

Stasys Peldžius

Vilniaus universiteto Matematikos ir informatikos fakulteto asistentas, daktaras
Vilnius University, Faculty of Mathematics and Informatics, Assistant, PhD
Naugarduko g. 24, LT-03225 Vilnius
El. paštas: stasys.peldzius@mif.vu.lt

Organizacijos, siekdamos aukštesnės kuriamo produkto kokybės, ieško būdų, kaip pagerinti programų sistemų kūrimo procesą, todėl daugėja galimybių laimėti skelbiamus konkursus, nes projektus gali atlikti greičiau ir laikytis nustatyto biudžeto bei terminų. Savo programų sistemų kūrimo procesą įmonės vertina pagal pasirinktus programų sistemų kūrimo proceso vertinimo modelius. Pagal poreikius ir galimybes organizacijos pasirenka dirbti vienu ar kitu programų sistemų kūrimo metodu. Vėliau pirmojo pasirinkto metodo neužtenka norint turėti didesnę proceso brandą, todėl reikia rinktis papildomus metodus. Svarbu iš anksto žinoti, kokią įtaką įmonės brandai darys pasirinktas metodas. Šiame straipsnyje nagrinėjama, kaip pasikeičia įmonės, kuri savo procesą vertina pagal CMMI-DEV, branda, kai dirbdama pagal „Scrum“ metodą papildomai pradeda naudoti „Kanban“ metodą. Nustatyta, kad įmonė, derindama „Scrum“ su „Kanban“, visiškai įgyvendintų specifinių praktikų skaičių padidina 75 procentais.

Reikšminiai žodžiai: programų sistemų kūrimo metodai, Scrum, Kanban, CMMI-DEV, organizacijos brandos lygis.

Kuriant programų sistemas vis daugiau dėmesio skiriama produkto kokybei, kad būtų galima sėkmingai konkuruoti dėl užsakymų, nes tik kokybiškas produktas gali ilgai išsilaikyti rinkoje. Įprasta, kad kokybiško produkto kūrimas dažnai reikalauja didelių išlaidų, todėl programų sistemų kūrimo industrijoje nuolatos ieškoma išlaidų ma-

žinimo ir efektyvumo didinimo galimybių. Šiam tikslui yra apibrėžta dešimtys programų sistemų kūrimo metodų, kurie skirti komandų efektyvumui didinti, resursams tausoti, produktui sistemingai pristatyti. Kiekvienas metodas įgyvendina tam tikrą gyvavimo ciklo modelį. Šiame straipsnyje yra tiriama du gerai žinomi *Agile* programų

sistemų kūrimo metodai: *Scrum* ir *Kanban*. Pagrindiniai šių metodų analizės šaltiniai yra J. Sutherlando ir K. Schwaberio išleistas *Scrum* gidas (Schwaber, Sutherland, 2013) ir D. J. Andersono knyga apie *Kanban* (Anderson, 2010a).

Egzistuoja daug *Agile* metodų vertinimų pagal įvairius standartus, tačiau *Scrum* metodas daugiausia įvertinamas pagal proceso vertinimo modelį *CMMI-DEV*. Egzistuojančiuose įvertinimuose yra tik vienas iš šių metodų ir visi nagrinėti vertinimai kuo nors skiriasi vienas nuo kito, o tai rodo, kad vertinimas yra reikalingas, bet kartu ir subjektyvus procesas. Todėl buvo nuspręsta kritiškai peržiūrėti atliktus *Scrum* vertinimus ir pateikti naują apibendrinamąjį įvertinimą.

Taikant *Kanban* metodą pasiekiami geresni rezultatai (Anderson ir kt, 2012; Sjøberg, Johnsen, Solberg, 2012), todėl daroma prielaida, kad *Scrum* metodą praplėtus *Kanban* metodu (to ne tik nedraudžia, bet netgi skatina vienas iš pagrindinių *Kanban* principų) turėtų pagerėti ne tik programų sistemų kūrimo efektyvumas, bet ir įvertinimas pagal *CMMI-DEV*.

Šio straipsnio tikslas – nustatyti, kiek pagerėja įmonės, kuri papildė *Scrum* metodą *Kanban* metodu, įvertinimas pagal *CMMI-DEV*.

Programų sistemų kūrimo metodų tendencijos

Scrum metodas yra vienas iš labiausiai žinomų programų sistemų kūrimo metodų, įgyvendinančių judrųjį programų sistemų kūrimo gyvavimo ciklo modelį. Apie tokį metodą pirmą kartą rašyta dar 1986 metais „Harvard Business Review“ žurnale (Takeuchi, Nonaka, 1986), tačiau programų sistemų kūrimui jis pradėtas taikyti tik įpusėjant dešimtajam dešimtmečiui (Sut-

herland, 2004). *Scrum* yra inkrementinis (prieaugio) produkto kūrimo metodas, jo lanksčią strategiją įgyvendina holistinė komanda, kuri bendromis pastangomis siekia nuolatos atnaujinamo tikslo. Metodo esmė yra žinojimas, kad užsakovas dažnai keičia savo nuomonę ir poreikius projekto metu, todėl svarbiausia yra maksimizuoti komandos gebėjimą greitai pristatyti produktą bei reaguoti į pasikeitusius reikalavimus ar išorinius technologinius ir rinkos pokyčius.

Nors *Toyota* produktų gamybos procese *Kanban* metodas buvo naudojamas jau nuo penktojo dešimtmečio (Ohno, 1988), tačiau programų sistemų kūrimo jis pritaikytas daug vėliau nei *Scrum*. Pirmą kartą *Kanban* sistema programų sistemų kūrimo paminėta tik 2003 metais, o pirmieji pranešimai apie šį metodą pasirodė tik 2007 metais. *Kanban* metodas skirtas darbų atlikimui valdyti ir optimizuoti, jis paremtas „pačiu laiku“ procesų planavimo sistema.

Produkto darbų sąrašo įrašą naudojant *Kanban* metodą atitinka *Kanban* kortelė, ji taip pat užpildoma komandai svarbia informacija, susijusia su konkrečiu darbu. Visos kortelės atvaizduojamos *Kanban* lentoje, kurioje yra mažiausiai trys grafos: į pirmą grafą įtraukiamos visos naujos kortelės, į paskutinę grafą patenka atliktų darbų kortelės, o kiekviena tarpinė grafa atvaizduoja konkretų kūrimo procesą. Šiuos procesus atvaizduojančios grafos gali turėti tik ribotą *Kanban* kortelių skaičių, kuris užrašomas prie grafos pavadinimo. Kortelės perkelioms tik tada, kai proceso darbas yra baigiamas ir kitame procese yra vietos kortelei priimti, priešingu atveju kortelė nėra perkeliama ir tai pažymi kortelių spūstis bei galimas procesų pralaidumo problemas.

D. J. Anderson yra nustatęs tris esminius *Kanban* principus (Anderson,

2010b): „pradėk su tuo, ką turi“, „sutik siekti inkrementinių, evoliucinių pokyčių“ bei „gerbk esamus procesus, vaidmenis ir atsakomybes“. Šie principai nurodo, kad *Kanban* neapibrėžia konkretaus darbo proceso, tačiau leidžia lengvai tobulinti esamą, pavyzdžiui, apibrėžtą *Scrum* metodo. Netrukus po *Kanban* pritaikymo programų sistemų kūrimui buvo pradėta publikuoti ir rezultatus, įmonėms perėjus nuo *Scrum* metodo prie *Kanban*.

Efektyvumo ir našumo tyrimai

2012 metais Oslo universiteto profesorius D. I. K. Sjøbergas kartu su „Software Innovation“ įmonės vadovais A. Johnseniu ir J. Solbergu atliko tyrimą, kurio tikslas – įvertinti *Scrum* ir *Kanban* metodų sėkmingumą vidutinio dydžio programinės įrangos kūrimo įmonėje (Sjøberg, Johnsen, Solberg, 2012). Nuo 2001 iki 2006 metų „Software Innovation“ programuotojai dirbo pagal „krioklio“ metodą, o 2007 metais perėjo prie *Scrum* metodo. Po kelerių metų buvo nuspręsta, kad *Scrum* metodas yra per mažai lankstus, todėl 2010 metų rudenį įmonė perėjo prie *Kanban* metodo. Tyrimui buvo pasirinkti duomenys nuo 2009 metų pirmo ketvirčio iki 2011 metų ketvirčio ketvirčio, taip užtikrinant, kad įmonė yra išmananti *Agile* programų sistemų kūrimo gyvavimo ciklo modelį ir perėjimas nuo „krioklio“ prie *Scrum* 2007 metais nedarys įtakos tyrimo rezultatams. O rezultatai buvo stulbinantys: nustatyta, kad įmonė, pradėjusi dirbti pagal *Kanban* metodą, beveik perpus sumažino laiką, per kurį atliekamas darbas (produkto darbų sąrašo įrašas arba *Kanban* kortelė). Dešimtadaliu sumažėjo pastangos defektų taisymui, dėl to 20 procentų pagerėjo produktyvumas. Apibendrinus šį tyrimą galima teigti, kad *Kanban* smarkiai lenkia

Scrum metodą, tačiau vieno tyrimo šiam teiginiui įrodyti nepakanka.

Kad *Kanban* yra efektyvesnis metodas nei *Scrum*, rodo ir kitas tyrimas. 2012 metais D. J. Andersonas atliko simuliacijomis pagrįstą tyrimą, kurio tikslas buvo palyginti *Scrum* ir *Kanban* metodų efektyvumą perėjus nuo *PSP/TSP* (Anderson ir kt., 2012). Simuliacijų įvesties duomenims buvo panaudoti tikri „Microsoft“ duomenys, programų priežiūros komandai pereinant nuo *PSP/TSP* prie *Kanban*. Pagrindinis tyrimo metu stebėtas rodiklis buvo proceso pralaidumas – vidutinis laikas, per kurį užbaigiamas vienas darbų sąrašo įrašas. Iš pradžių simuliacijos buvo suderintos, kad kuo tiksliau atitiktų duomenis komandai dirbant pagal *PSP/TSP*. Vėliau buvo paleistos simuliacijos pagal *Kanban* nustatymus ir rezultatas palygintas su tikrais duomenimis. Įvertinus, kad rezultatas yra tinkamas, simuliacija buvo laikyta tikslia ir sėkminga. Tokios simuliacijos buvo atliktos ir pagal *Scrum* nustatymus, o rezultatai išsamiai analizuojami. Apibendrinami tyrimą autoriai nurodė, kad, simuliacijų duomenimis, šešių asmenų komandai dirbant pagal *PSP/TSP* proceso pralaidumas buvo šeši darbų sąrašo įrašai per mėnesį, dirbant pagal *Scrum* – 11,5 darbų sąrašo įrašo per mėnesį, o pagal *Kanban* – 12 darbų sąrašo įrašų. Taigi, nors *Kanban* metodo pranašumas atrodo mažas, tačiau per ilgą laikotarpį arba didesnėse komandose rezultatų skirtumas tampa akivaizdesnis.

CMMI-DEV proceso vertinimo modelis

Kadangi praktiniais tyrimais įrodyta, kad *Kanban* metodas yra efektyvesnis už *Scrum* metodą, mūsų straipsnyje *Kanban* pranašumas įrodomas įvertinus metodus pagal *CMMI-DEV* (*CMMI-DEV*, 2010). Verti-

nama, kaip *Scrum* ir *Kanban* metodai įgyvendina antram *CMMI-DEV* brandos lygiui priskirtas proceso sritis. Svarbu pabrėžti, kad atliekant vertinimą buvo atsižvelgiama tik į metodų apibrėžtus įvykius, veiklas ar kitus faktus, tačiau nebuvo išgalvota jokia bendrinė organizacija ar organizaciniai procesai bei gebėjimai.

Iš pradžių buvo atlikti kokybinis ir kiekybinis įvertinimai *Scrum* metodui, vėliau taip pat atlikti įvertinimai *Kanban* metodui, naudojamam kartu su *Scrum*, ir galiausiai abu įvertinimai buvo palyginti.

Kokybinis įvertinimas buvo atliekamas remiantis „*Standard CMMI® Appraisal Method for Process Improvement (SCAMPISM) A, Version 1.3: Method Definition Document*“ metodo nurodymais (SCAMPI A, 2011). Kiekvienos proceso srities specifinių tikslų specifinės praktikos buvo vertinamos atskirai. Kiekvienos specifinės praktikos įgyvendinimas pakomentuojamas ir įvertinamas keturiais vertinimo lygiais: „F“ – visiškai įgyvendinama specifinė praktika, „L“ – daugiausia įgyvendinama, „P“ – iš dalies įgyvendinama ir „N“ – nėra įgyvendinama. „*SCAMPI A*“ dokumente nurodoma, kad specifinis tikslas laikomas patenkintu tada ir tik tada, kai visos jam priskirtos specifinės praktikos yra įvertintos „F“ arba „L“, o silpnybės, siejamos su specifiniu tikslu, neturi didelės neigiamos įtakos tikslui patenkinti. Tačiau šiame darbe pagrindinis dėmesys skiriamas ne specifinių tikslų patenkimui, o kiekvienos specifinės praktikos įgyvendinimui.

Kiekybinis įvertinimas buvo atliekamas po kokybinio įvertinimo. Buvo suskaičiuota, kiek specifinių praktikų yra įvertinta kiekvienu iš keturių įvertinimo lygių, bei analizuojama, ties kuriomis proceso sritimis telkiasi aukštesnieji įvertinimai („L“ ir „F“).

Šiems įvertinimams pateikiamos diagramos su tiksliais reikšmėmis ir lentelės su procentinėmis reikšmėmis. Pabaigoje buvo atliktas lyginamasis įvertinimas – palyginti abu kiekybiniai įvertinimai.

***Scrum* vertinimas pagal CMMI-DEV**

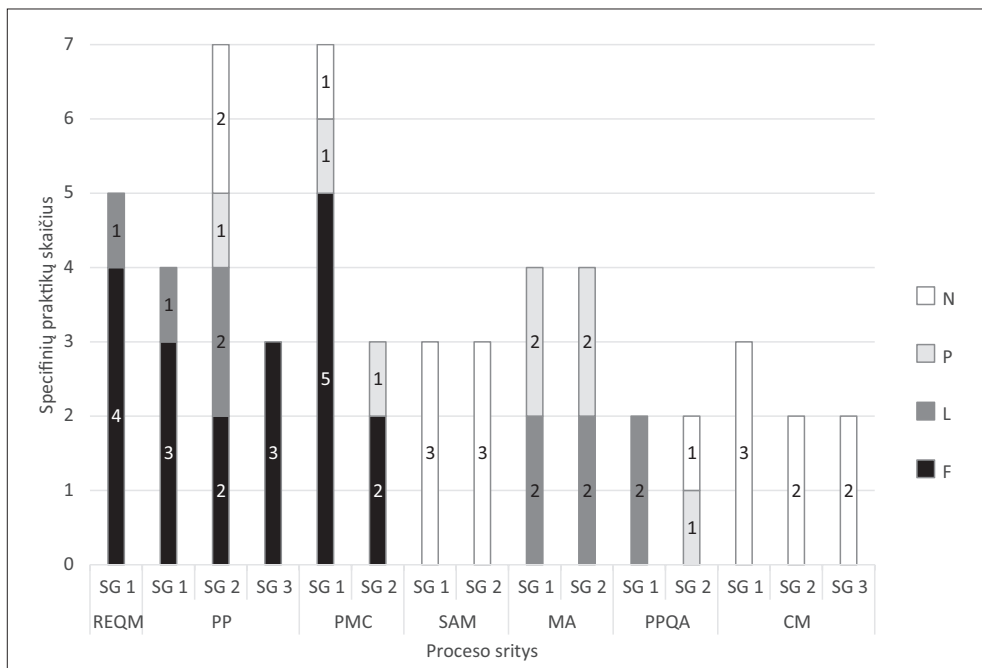
Šiame skyriuje pateikiamas apibendrintas *Scrum* metodo įvertinimas pagal pakopinės architektūros *CMMI-DEV* modelį. Buvo atliktas išsamus įvertinimas analizuojant penkis straipsnius (Diaz, Garbajosa, Calvo-Manzano, 2009; Foegen, Croome, 2011; Marçal ir kt., 2007; Potter, Sakry, 2009; Varthanan Moorthy, 2012), kuriuose *Scrum* siejamas su antru pakopinės architektūros *CMMI-DEV* modelio brandos lygiu. Gautas įvertinimas pateikiamas matrica (1 lentelė).

Siekiant aiškiau atvaizduoti įvertinimo lygių pasiskirstymą proceso srityse, duomenys buvo atvaizduoti diagrama (1 pav.). Joje išryškėja gana aukštas projekto valdymo proceso sričių įgyvendinimas – trys iš keturių proceso sričių (REQM, PP ir PMC) turi didelį santykį visiškai įgyvendintų specifinių praktikų, net 19 iš 29 specifinių praktikų įvertintos kaip visiškai įgyvendinamos *Scrum* metodo. Deja, paskutinė projekto valdymo proceso sritis (SAM) iš viso nėra aprašoma *Scrum*. Dvi iš trijų palaikymo proceso sričių (MA ir PPQA) yra iš dalies įgyvendintos, lygiai pusė (6 iš 12) jų specifinių praktikų yra daugiausia įgyvendintos, tačiau trečioji palaikymo proceso sritis (CM) nėra įgyvendinta *Scrum* metodo.

Specifinių praktikų įvertinimo statistikoje (2 lentelė) matome, kiek iš viso kiekvienu įgyvendinimo lygiu yra įvertinta praktikų. 29 iš 54, arba 53,70 %, specifinių praktikų yra įvertintos kaip visiškai arba daugiausia įgyvendinamos *Scrum* metodo.

1 lentelė. Scrum įvertinimo pagal CMMI-DEV matrica

Specifinis tikslas	Specifinė praktika	Proceso sritis						
		REQM	PP	PMC	SAM	MA	PPQA	CM
SG 1	SP 1.1	F	F	F	N	P	L	N
	SP 1.2	F	F	F	N	L	L	N
	SP 1.3	F	F	P	N	P		N
	SP 1.4	L	L	N		L		
	SP 1.5	F		F				
	SP 1.6			F				
	SP 1.7			F				
SG 2	SP 2.1		L	F	N	P	P	N
	SP 2.2		P	P	N	L	N	N
	SP 2.3		N	F	N	P		
	SP 2.4		F			L		
	SP 2.5		N					
	SP 2.6		F					
	SP 2.7		L					
SG 3	SP 3.1		F					N
	SP 3.2		F					N
	SP 3.3		F					



1 pav. Scrum įvertinimo pagal CMMI-DEV diagrama

2 lentelė. *Specifinių praktikų įvertinimo statistika*

<i>Įgyvendinimo lygis</i>	<i>Praktikų skaičius</i>	<i>Dalis nuo visų praktikų</i>
F	19	35,19 %
L	10	18,52 %
P	8	14,81 %
N	17	31,48 %
Iš viso:	54	100,00 %

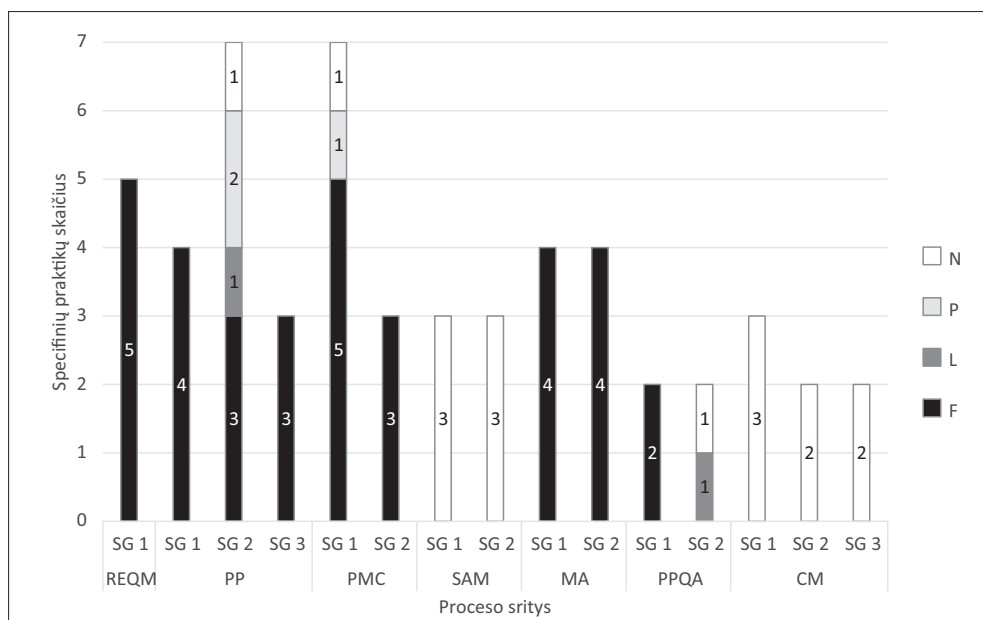
Scrum, praplėsto Kanban, įvertinimas pagal CMMI-DEV

Šiame skyriuje pateikiama, kaip pasikeičia įvertinimas pagal pakopinės architektūros CMMI-DEV modelį, jei su Scrum metodu kartu naudojamas ir Kanban. Kadangi straipsnių, aprašančių Kanban įvertinimą pagal CMMI-DEV, nėra, įvertinimas atliekamas empiriniu tyrimu, remiantis visa analizuota literatūra apie Kanban metodą, bei patirtimi. Kanban metodas skatina neat-sisakyti organizacijos darbo įpročių, o juos tobulinti, todėl įvertinant praplėtimą buvo daroma prielaida, kad praplečiant Scrum

metodą neatsisakoma jokių Scrum artefaktų, įvykių ar vaidmenų, tačiau pridedamos visos Kanban taisyklės. Pavyzdžiui, PP proceso srities „SP 1.4. Įvertinti darbų kiekį ir kainą“ įvertinimas pagal Scrum yra „L“, nes Scrum metodas nurodo darbus skaidyti į mažesnes dalis ir įvertinti jas sudėtingumo taškais (kas dažnai yra netikslu), o pagal Kanban – „F“, nes naudojant šį metodą darbus stengiamasi skaidyti į vienodos apimties Kanban korteles. Tai leidžia nustatyti projekto užduočių pastangų kiekį ir kainą. Gautas įvertinimas pateikiamas matrica (3 lentelė).

3 lentelė. *Scrum, praplėsto Kanban, įvertinimo pagal CMMI-DEV matrica*

<i>Specifinis tikslas</i>	<i>Specifinė praktika</i>	<i>Proceso sritis</i>						
		REQM	PP	PMC	SAM	MA	PPQA	CM
SG 1	SP 1.1	F	F	F	N	F	F	N
	SP 1.2	F	F	F	N	F	F	N
	SP 1.3	F	F	P	N	F		N
	SP 1.4	F	F	N		F		
	SP 1.5	F		F				
	SP 1.6			F				
	SP 1.7			F				
SG 2	SP 2.1		F	F	N	F	L	N
	SP 2.2		P	F	N	F	N	N
	SP 2.3		N	F	N	F		
	SP 2.4		F			F		
	SP 2.5		P					
	SP 2.6		F					
	SP 2.7		L					
SG 3	SP 3.1		F					N
	SP 3.2		F					N
	SP 3.3		F					



2 pav. Scrum, praplėsto Kanban, įvertinimas pagal CMMI-DEV

Įvertinus antro pakopinės architektūros CMMI-DEV lygio proceso sritis, galutinis visų sričių įvertinimas suvestas į bendrą diagramą (2 pav.). Joje išryškėja labai aukštas projekto valdymo proceso sričių įgyvendinimas – trys iš keturių proceso sričių (REQM, PP ir PMC) bendroje sumoje turi net 23 specifines praktikas, įvertintas aukščiausiu įgyvendinimo lygiu („F“), tai yra beveik 80 % visų šių proceso sričių specifinių praktikų. Deja, paskutinė projekto valdymo proceso sritis SAM išlieka neaprašyta, todėl jai priskirtos specifinės

praktikos tebėra neįgyvendintos net *Scrum* metodą praplėtus *Kanban* metodu. Diagramoje matomas ir stiprus palaikymo proceso sričių įgyvendinimas – MA proceso srities visos aštuonios specifinės praktikos yra visiškai įgyvendintos, o trys iš keturių PPQA specifinių praktikų yra daugiausia arba visiškai įgyvendintos. Situacija su paskutine palaikymo proceso sritimi lieka nepakitusi – visos septynios CM specifinės praktikos išlieka neįgyvendintos.

Specifinių praktikų įvertinimo statistika (4 lentelė) rodo, kiek iš viso kiekvienu

4 lentelė. Specifinių praktikų įvertinimo statistika

Įgyvendinimo lygis	Praktikų skaičius	Dalis nuo visų praktikų
F	33	61,11 %
L	2	3,70 %
P	3	5,56 %
N	16	29,63 %
Iš viso:	54	100,00 %

įgyvendinimo lygiu yra įvertinta praktiškų. 35 iš 54, arba 64,81 %, specifinių praktiškų yra įvertintos kaip visiškai arba daugiausia įgyvendintos *Scrum* metodą praplėtus *Kanban* metodu.

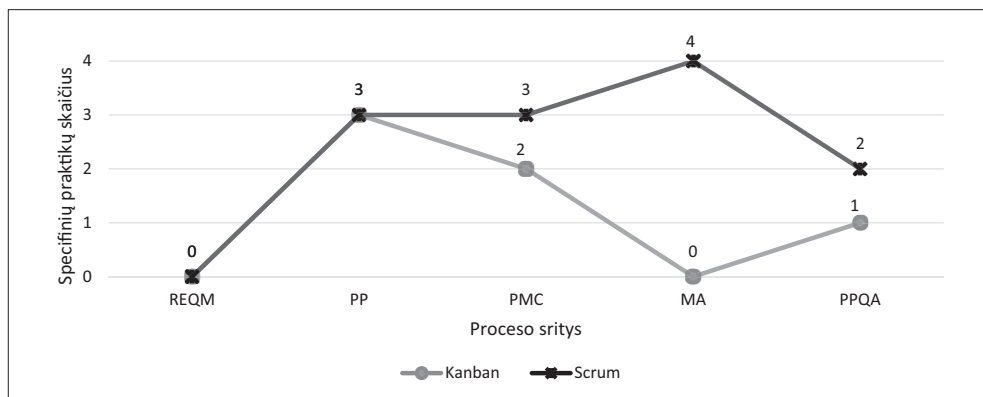
Procesų tobulinimas naudojant *Kanban*

Dabar bus atliktas kiekybinis abiejų įvertinimų palyginimas. Iš pradžių palyginami specifinių praktiškų įvertinimų kiekiai, vėliau aprašyta skaičiavimo sistema apskaičiuojami ir palyginami metodikų skaitiniai įvertinimai.

Suskaičiavus visas specifines praktikas, įvertintas kaip visiškai įgyvendinamas, buvo nubrėžta diagrama (3 pav.), vaizduojanti, kaip *Kanban* pagerina *CMMI-DEV* specifinių praktiškų įvertinimą. Diagramoje neįtrauktos viena projekto valdymo (SAM) ir viena palaikymo (CM) proceso sritys, nes jų specifinių praktiškų įgyvendinamumo neaprašo nei *Scrum*, nei *Kanban* metodai. Matoma, kad netenkinamą įvertinimą gavusių specifinių praktiškų skaičius sumažėja šešiais vienetais. Norint, kad diagramoje pavaizduotos penkios antro brandos lygio proceso sritys būtų laikomos įgyvendinamomis, organizacijai beliktų pagerinti tik šešių likusių specifinių praktiškų įgyvendinamumą (iš viso šioms proceso sritims priskirta 41 specifinė praktika).

kad palaikymo proceso srityse taikant tik *Scrum* metodą nebuvo nė vienos visiškai įgyvendinamos specifinės praktikos. Kita diagrama (4 pav.) vaizduoja, kaip sumažėja specifinių praktiškų, neleidžiančių patenkinti specifinius tikslus, skaičius. Tokios specifinės praktikos buvo įvertintos kaip iš dalies įgyvendinamos arba išvis neįgyvendinamos. Diagramoje neįtrauktos viena projekto valdymo (SAM) ir viena palaikymo (CM) proceso sritys, nes jų specifinių praktiškų įgyvendinamumo neaprašo nei *Scrum*, nei *Kanban* metodai. Matoma, kad netenkinamą įvertinimą gavusių specifinių praktiškų skaičius sumažėja šešiais vienetais. Norint, kad diagramoje pavaizduotos penkios antro brandos lygio proceso sritys būtų laikomos įgyvendinamomis, organizacijai beliktų pagerinti tik šešių likusių specifinių praktiškų įgyvendinamumą (iš viso šioms proceso sritims priskirta 41 specifinė praktika).

Siekiant pagerėjimą išreikšti skaitine reikšme naudojama balo skaičiavimo sistema. Įvertinimas balais skaičiuojamas taip: kiekviena visiškai įgyvendinama („F“) specifinė praktika verta 1 balo, kiekviena daugiausia įgyvendinama („L“) specifinė praktika verta 0,85 balo, kiekviena iš dalies įgyvendinama („P“) verta 0,50 balo



3 pav. Visiškai įgyvendinamų praktiškų palyginimas

ir kiekviena neįgyvendinama („N“) verta 0,15 balo. Balų reikšmės pasirinktos pagal ISO/IEC 15504 standartą, kuriame tokie balai atitinka procesų atributų įvertinimų viršutinius režius. Pagal šį standartą proceso atributas įvertinamas „F“, jei jo praktikų įverčių vidurkis yra nuo 86 % iki 100 %, „L“ – nuo 51 % iki 85 %, „P“ – nuo 16 % iki 50 % ir „N“ – nuo 0 % iki 15 %. Maksimalus įvertinimas tokioje sistemoje yra 54 balai, jei visų septynių antro CMMI-DEV brandos lygio proceso sričių specifinės praktikos yra visiškai įgyvendinamos, arba 41 balas, jeigu iš skaičiavimų pašalintume SAM ir CM proceso sritis, kurios nebuvo aprašytos nei *Scrum*, nei *Kanban*. Susumavus balus gautas toks įvertinimas (5 lentelė): *Scrum* metodas surinko 34,05 balo, o *Scrum* su *Kanban* – 38,60 balo. Įvertinimas padidėja daugiau nei 13 %. Jeigu įvertinimą lygintume su maksimaliu galimu įvertinimu (54), tai surinktų balų kiekis padidėja nuo 63,06 % iki 71,48 %, tačiau jeigu į skaičiavimus neįtraukiame SAM ir CM proceso sričių, tai surinktų balų kiekis padidėtų nuo 83,05 % iki 94,15 %. Vadinas, *Scrum* metodo, praplėsto *Kanban* metodu, įvertinimas viršija 86 %, o tai kituose vertinimo modeliuose, pavyzdžiui, „PKP Branda“, gali būti laikoma, kad šių metodų junginys visiškai įgyvendina penkias antro CMMI-DEV brandos lygio proceso sritis. Galutinis rezultatas yra aiškus: nors metodų derinys padės efektyviau valdyti organizacijos procesus, tačiau siekdama gauti antro CMMI-DEV brandos

lygio įvertinimą organizacija privalo turėti stiprius sutarties su tiekėju ir konfigūracijos valdymo įgūdžius.

Rezultatai ir išvados

- Nustatyta, kad, *Scrum* metodą praplėtus *Kanban* metodu, visos REQM ir MA proceso sričių specifinės praktikos bei 75 % PP ir PMC pasiekia visišką įgyvendinimą. Visiškai įgyvendinamų specifinių praktikų skaičius išaugo nuo 19 iki 33.
- Nustatytas pagerėjimas ir analizuojant prasčiausius įvertinimus gavusias praktikas: PMC ir PPQA proceso sritys sumažino neįgyvendinamų arba iš dalies įgyvendinamų praktikų kiekį vienu vienetu, o MA – net keturiais.
- Įvesta įvertinimo balais skaičiavimo sistema, kurioje nustatyta, kad *Scrum* metodą praplėtus *Kanban* metodu įvertinimas padidėja 13 %. Šioje skaičiavimo sistemoje vertinant išmetus SAM ir CM proceso sritis gauta, kad surenkamų balų skaičius pakilo nuo 83 % iki 94 % galimų balų.

Jei organizacijos, dirbančios pagal *Scrum* metodą, norėtų pagerinti savo procesus pernelyg jų nepakeisdamos, joms verta rinktis *Kanban* metodą *Scrum* metodui praplėsti. Deja, vien tik šių metodų nepakanka, kad būtų pasiektas antrasis CMMI-DEV brandos lygis. Verta paminėti, kad SAM proceso sritis gali būti įgyvendinama išorinėmis paslaugomis, todėl likusios reikiamos pastangos antram CMMI-DEV lygiui

5 lentelė. *Metodų įvertinimas balais*

Metodas	F (1)	L (0,85)	P (0,50)	N (0,15)	Balas	% nuo 54	% nuo 41
Scrum	19	10	8	17	34,05	63,06%	83,05%
Kanban	33	2	3	16	38,60	71,48%	94,15%

pasiękti santykinai nėra didelės: reikia įgyti tvirtus konfigūracijos valdymo įgūdžius ir pasiekti „F“ arba „L“ įgyvendinimo lygi likusioms šešioms specifinėms praktikoms.

Scrum metodą praplečiant kitu (ne *Kanban*) metodu būtų gaunami kitokie rezultatai, todėl ieškant geriausio metodų derinio turėtų būti atliekama daugiau teorinių bei praktinių tyrimų naudojant kitus metodus.

LITERATŪRA

ANDERSON, David J. ir kt. (2012). A Comparative Study of Scrum and Kanban Approaches on a Real Case Study Using Simulation. In *Agile Processes in Software Engineering and Extreme Programming*. Berlin: Springer, p. 123–137. ISBN 9783642206771.

ANDERSON, David J. (2010a). *Kanban. Successful Evolutionary Change for Your Technology Business*. Sequim: Blue Hole Press. ISBN 0984521402.

ANDERSON, David J. (2010b). *The Principles of the Kanban Method*. David J. Anderson & Associates. Prieiga per internetą: <<http://dja.com/principles-kanban-method-0>>.

CMMI Product Team (2010). CMMI® for Development, Version 1.3 (CMU/SEI-2010-TR-033). *Software Engineering Institute, Carnegie Mellon University*. Prieiga per internetą: <<http://www.sei.cmu.edu/library/abstracts/reports/10tr033.cfm>>.

DIAZ, Jessica; GARBAJOSA, Juan; CALVOMANZANO, Jose A. (2009). Mapping CMMI Level 2 to Scrum Practises: An Experience Report. In *Communications in Computer and Information Science*. Berlin: Springer, p. 93–104.

FOEGEN, Malte; CROOME, David (2011). How Scrum helps with CMMI. *wibas*. Prieiga per internetą: <https://www.wibas.com/media/filer_public/2013/10/01/wibas-scrum-for-cmmi_en.pdf>.

MARÇAL, Ana Sofia C., ir kt. (2007). Mapping CMMI Project Management Process Areas to SCRUM Practises. *Research Gate*. Prieiga per internetą: <https://www.researchgate.net/publication/4298820_Mapping_CMMI_project_management_process_areas_to_SCRUM_practises>.

OHNO, Taiichi (1988). *Toyota Production System*.

Portland: Productivity Press. ISBN 9780915299140.

POTTER, Neil; SAKRY, Mary (2009). Implementing Scrum (Agile) and CMMI Together. *The Process Group*, vol. 16, no. 2. Prieiga per internetą: <<http://www.processgroup.com/pgpostmar09.pdf>>.

SCAMPI Upgrade Team (2011). Standard CMMI Appraisal Method for Process Improvement (SCAMPI) A, Version 1.3: Method Definition Document (CMU/SEI-2011-HB-001). *Software Engineering Institute*. Prieiga per internetą: <<http://www.sei.cmu.edu/library/abstracts/reports/11hb001.cfm>>.

SJØBERG, Dag I. K.; JOHNSEN, Anders; SOLBERG, Jørgen (2012). Quantifying the Effect of Using Kanban versus Scrum: A Case Study. *IEEE Software*, vol. 29, no. 5, p. 47–53. ISSN 0740-7459.

SCHWABER, Ken; SUTHERLAND, Jeff (2013). *The Scrum Guide* [interaktyvus]. Prieiga per internetą: <<http://www.scrumguides.org/docs/scrumguide/v1/scrum-guide-us.pdf>>.

SUTHERLAND, Jeff (2004). Agile Development: Lessons Learned from the First Scrum. *Scrum Alliance*. Prieiga per internetą: <<https://www.scrumalliance.org/resources/35>>.

TAKEUCHI, Hirotaka; NONAKA, Ikujiro (1986). New New Product Development Game. *Harvard Business Review*, p. 137–146. Prieiga per internetą: <<https://hbr.org/1986/01/the-new-new-product-development-game>>.

VARTHANAN MOORTHY, Vishnu (2012). *CMMI with AGILE – Demystified* [interaktyvus]. Prieiga per internetą: <<https://www.slideshare.net/visvarthana/cmmi-with-agile-demystified-13792446>>.

SOFTWARE PROCESS ASSESSMENT IN SCRUM WITH A KANBAN-BASED ORGANIZATION

Vaidotas Pėkis, Stasys Peldzius

S u m m a r y

Seeking for a higher quality of developing products, organizations often look for ways to improve their software development process. Consequently, the probability of winning a contest increases, since projects can be finished faster together with sticking to budget and meeting deadlines. Various models of process assessment can be used to evaluate the development process of a company. Organizations tend to choose them using one or another software development methodology or framework in order to improve the assessment. But in the long-term period, the methodology chosen first often appears to be insufficient in achieving a higher maturity of

the process; therefore, this leads to the implementation of additional methodologies. It is critical for the organization to know how a chosen methodology will affect its maturity. This paper investigates how the CMMI-DEV assessment of the development process changes when the organization, using Scrum as the base of its software development process, extends it with Kanban. It was found that an organization combining Scrum together with Kanban would increase the number of fully implemented practices by 75%.

Keywords: software development methodology, Scrum, Kanban, CMMI-DEV, maturity level of organization.

2017 m. gegužės 31 d.