

## SCRUM METODO TAIKYMO GALIMYBĖS RENGiant UTENOS KOLEGIJOS INFORMATIKOS INŽINERIJOS STUDIJŲ KRYPTIES STUDENTŲ BAIGIAMŲJŲ DARBŲ PRAKTINĘ DALĮ

Jūratė Urbonienė, Donatas Bukelis, Arūnas Pranaitis

*Utenos kolegija, Verslo ir technologijų fakultetas*

*Maironio g. 18, Utena*

### Anotacija

*SCRUM* – tai lanksčiojo programavimo projektų valdymo metodas, kuris paremtas efektyviu projekte dalyvaujančios komandos bendradarbiavimu. Šis metodas plačiai taikomas IT įmonėse programinės įrangos kūrimo projektų metu. Metodo taikymas rengiant IT projektus studijų metu analizavo V. Steponavičienė ir J. Urnikienė [1], tuo tarpu šaltinių apie metodo taikymą rengiant informatikos inžinerijos krypties baigiamuosius darbus neaptikta. Dalį straipsnio autorių nuolat nagrinėja IT srities studentų baigiamųjų darbų kokybės gerinimo klausimus, yra atlikta analizė ir buvo taikytas *Lean* metodas rengiant baigiamojo darbo analitinę dalį [2], tačiau pastebėta, jog pagerėjus vienos analitinės dalies kokybei dar vis nemažai spragų matoma rengiant baigiamojo darbo praktinę dalį – neaktyvinant užsakovo ar būsimąjo vartotojo nuolatinio komunikavimo, lieka nerealizuotos svarbios ir reikalingos kuriamo produkto funkcijos. *Straipsnio tikslas* yra pristatyti *SCRUM* metodo taikymo galimybes rengiant Utenos kolegijos informatikos inžinerijos studijų krypties studentų baigiamųjų darbų praktinę dalį. Taigi, straipsnyje trumpai aptariamas *SCRUM* metodas ir pagrindiniai jo elementai, nagrinėjamos *SCRUM* metodo pritaikymo Utenos kolegijos IT studentų baigiamųjų darbų praktinės dalies rengimui galimybės bei pateikiamos praktinės išvalgos, kaip galėtų atrodyti *SCRUM* metodo elementai baigiamojo darbo rengimo procese, siekiant kuo geresnio galutinio rezultato – studentų baigiamuoju darbu kuriamo užsakomojo produkto (internetu svetainės, mobiliosios aplikacijos, sistemos konfigūravimo dokumento ar pan.), kuris atitiktų užsakovo reikalavimus ar vartotojų poreikius ir teiktų suderintą pageidaujamą funkcionalumą.

**Reikšminiai žodžiai:** *SCRUM* metodas, IT baigiamasis darbas

### 1. *SCRUM* lanksčiojo programavimo projektų valdymo metodas

*SCRUM* – tai darbo metodų struktūra (angl. *Framework*, toliau – *SCRUM* metodas), kuri paremta efektyviu komandos tarpusavio bendradarbiavimu [3]. Šis metodas leidžia kurti savaime veikiančias komandas, kurios užduoties įgyvendinimo metu skatinamos betarpiškai bendrauti. *SCRUM* yra darbo bei bendradarbiavimo metodų struktūra, tačiau *SCRUM* kūrėjai tvirtina, kad tai nėra metodologija [3]. *SCRUM* metodas naudojamas programinių produktų (toliau – produktų) kūrimui ir vystymui valdyti. Naudojant šį metodą visi veiklos procesai ar veikimo būdai yra taikomi, tam kad užtikrinti sklandų produkto kūrimo procesą. *SCRUM* nusako produktų kūrimo būdų ir jų valdymo santykinį efektyvumą [3]. Tai sudaro sąlygas šių procesų tobulinimui. *SCRUM* metodas yra lengvai suprantamas, tačiau sunkiai įsisavinamas bei valdomas praktiškai.

*SCRUM* metodo elementai yra *SCRUM* komandos ir jos nariams priskirtos rolės, įvykiai (ceremonijos), artefaktai ir taisyklės, kurios apibrėžia roles, įvykius, artefaktus ir jų sąveiką [3, 5]. *SCRUM* remiasi empirinės procesų kontrolės teorija (empirizmu) [6]. Empirizmas paremtas tuo, kad žinios įgyjamos iš patirties, o priimant sprendimus remiamasi tuo, kas jau žinoma.

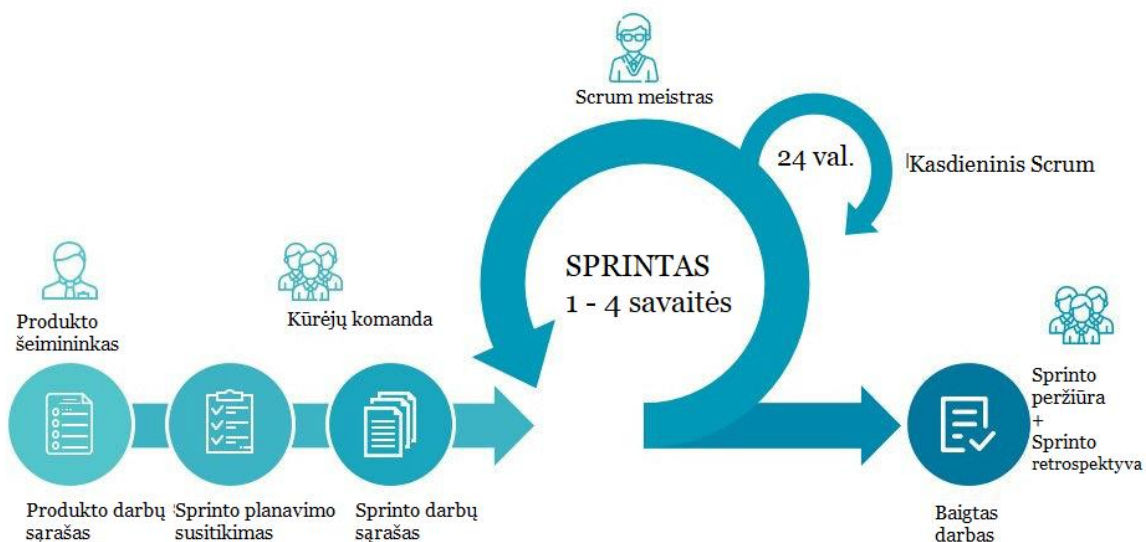
*SCRUM* komandos narių rolės gali būti **Produkto šeimininkas** (angl. *Product Owner*), **Kūrimo komanda** (angl. *Development Team*), **Scrum meistras** (angl. *Scrum Master*) [3, 4, 6]. *SCRUM* komandos narių atsakomybė ir funkcijos pateikiamos 1 lentelėje.

1 lentelė. SCRUM narių atsakomybių ir funkcijų aprašymas

SCRUM komandos rolė	Narių skaičius	Atsakomybė ir funkcijos
<i>Produkto šeimininkas</i>	Vienas asmuo	Atsakingas už maksimalią kūrimo komandos atliekamo darbo vertę, kurią užsakovui ar klientui sukuria ar sukurs pats produktas; Atsakingas už produkto kūrimo darbų sąrašo valdymą; Pagrindinės funkcijos: aiškus produkto darbų sąrašo apibūdinimas ir įforminimas; darbų sąrašo įrašų eiliškumo, prioritetų nustatymas; užtikrinimas, kad programuotojų komanda tinkamai supras darbų sąrašo įrašus.
<i>Kūrimo komanda</i>	Savarankiška grupė asmenų	Tai kūrėjai, kurie atlieka tam darbą, kuria ir vysto produktą pagal sudarytą darbų sąrašą; Savarankiškai pasirenka, koku būdu geriausiai atlikti paskirtus darbus; Komandos nariai turi turėti visas darbui atlikti reikiamas kompetencijas ir įgūdžius; Atsakomybė už galutinį rezultatą – sukurtą užsakovo reikalavimus ar vartotojo poreikius atitinkantį produktą – priklauso komandai kaip visumai.
<i>Scrum meistras</i>	Vienas asmuo	Tai komandos lyderis, kuris prižiūri kaip kūrimo komanda laikosi SCRUM metodo praktinių taikymų, taisyklių, taip pat užtikrina, kad darbo metu būtų naudojamas SCRUM metodas; Tarpininkauja sprendžiant klausimus su ne SCRUM komandai priklausančiais asmenimis; Yra atsakingas už SCRUM įvykių vyksmą, kliūčių, kylančių kūrimo komandai, šalinimą.

SCRUM įvykis (iteracija) – tai ribotos trukmės metodo dalis, kurios metu baigiama atlikti galutinio produkto kūrimo užduotis ir kurio tikslas užtikrinti visišką produkto kūrimo skaidrumą ir galutinio rezultato pritaikymą [3].

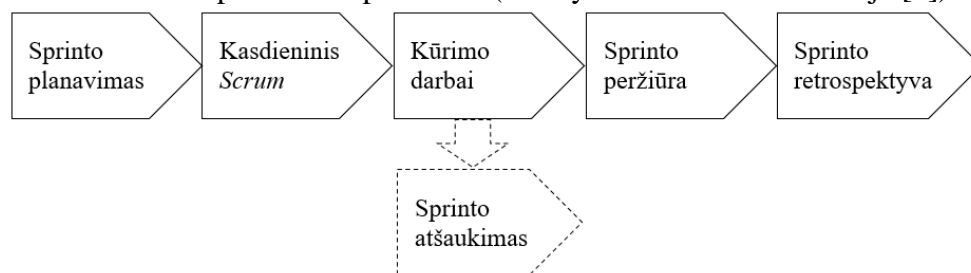
SCRUM metodo schematinis vaizdas ir elementų tarpusavio sąveika matoma 1 paveiksle [7].



1 pav. SCRUM metodo elementų tarpusavio sąveika

*Sprintas* (angl. *Sprint*) – tai pagrindinis SCRUM įvykis, kurio metu baigiamos atlikti kelios stambios produkto kūrimo užduotys, t. y. sukuriamas tam tikras galutinio produkto elementas. Sprintui darbai pasirenkami iš numatytų darbų sąrašo. Kiekvienas sprintas turi tikslą, apibrėžiantį, kas turi būti sukurta, planą, kuriuo remiantis bus kuriama, ir kuriamą galutinio produkto elementą.

Sprinto metu neleidžiami jokie pakeitimai, kurie galėtų įtakoti sprinto tikslą, nemažinami kokybės tikslai ir reikalavimai. Sprintui nustatomas tam tikras laikotarpis. Ilgiausias laikotarpis yra 30 dienų (4 savaitės), rekomenduojamas ir dažniausiai naudojamas – 2 savaitių. Šis laikotarpis skirtas sukurti užbaigtą, veikiantį ir užsakovo ar vartotojo lūkesčius atitinkantį kuriamo galutinio produkto prieaugį. Sprinto schematinis vaizdas pateiktas 2 paveiksle (sudaryta remiantis informacija [4]).



2 pav. Sprinto schematinis vaizdas

*Sprinto planavimas* – tai darbo ar darbų, kurie privalo būti baigti per sprintui numatytą laiką, iš darbų sąrašo parinkimas, t. y. darbų plano sudarymas. Tam organizuojamas sprinto planavimo susirinkimas. Sprinto darbų planą kuria visi *SCRUM* komandos nariai – *Produkto šeimininkas*, *Scrum meistras*, *Kūrimo komanda*. Sprinto planavimo susitikimo tikslas atsakyti į du klausimus: 1. Ką galima sukurti sprinto metu? Pasirenkami darbai iš darbų sąrašo, įvardijamas sprinto tikslas. 2. Kaip bus atlikti galutiniam veikiančiam produktui sukurti reikalingi darbai?

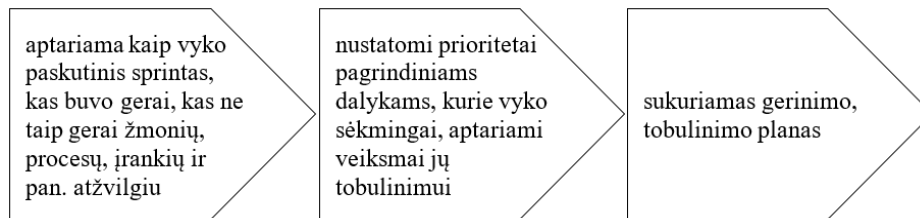
*Kasdieniniai Scrum* (angl. *Daily Scrum* arba *Daily Stand-up*) – tai daugiausiai 15 minučių trukmės susirinkimas, kurį rekomenduojama organizuoti ne darbo, o kurioje nors kitoje vietoje dažniausiai visai komandai stovint. Už *kasdieninių Scrumų* organizavimą atsakingas *Scrum meistras*, tačiau daugiausia kalba *Kūrimo komandos* nariai. Tokio susirinkimo tikslas – suderinti veiksmus ir sukurti planą artimiausioms 24 valandoms, t. y. kiekvienas *Kūrimo komandos* narys turi atsakyti į tokius klausimus: ką aš atlikau vakar; ką aš planuoju atlikti šiandien; ar yra kokių kliūčių, kurios trukdo man atlikti reikiamą darbą.

Sprinto pabaigoje atliekama *Sprinto peržiūra*, tam kad būtų patikrinta ar yra pasiektas numatytas tikslas ir sukurtas galutinio naudoti skirto produkto elementas, jei reikia pritaikomas ir darbų sąrašas. Susitikime dalyvauja *Kūrimo komanda* ir *Produkto šeimininkas*, pakviečiamos kitos suinteresuotos šalys, tokios kaip užsakovas, būsimi vartotojai. Susitikimo dalyviai aptaria, kas buvo atlikta sprinto metu. Sprinto peržiūros metu aptariamus dalykus galima matyti 2 lentelėje.

2 lentelė. Sprinto peržiūros metu aptariamų dalykų aprašas

SCRUM komandos narys (-iai)/ Suinteresuotas asmuo	Aptariamai dalykai
Produkto šeimininkas	paaikškina, kurie darbai iš darbų sąrašo buvo baigti, kurie ne
Kūrimo komanda	aptaria, kas sprinto metu vyko sėkmingai, kokios problemos ar kliūtys buvo iškilusios, kokie buvo jų sprendimo būdai
Kūrimo komanda	demonstruoja baigtus naudojimui skirtus produktus ar produktą, atsako į klausimus
Produkto šeimininkas	aptaria produkto darbų sąrašą, pateikia planuojamą darbų baigimo datą
Visa susitikimo grupė	aptaria, kas iš darbų sąrašo galėtų būti daroma toliau, tam, kad sprinto peržiūra pateiktų vertingą indėlį kitam sprinto planavimo susitikimui.

*Sprinto retrospektyva* – tai susirinkimas, kuris suteikia galimybę visai *SCRUM* komandai pagalvoti apie tai, ką galima pagerinti kitam sprintui. Susirinkimo eiga pateikta 3 paveiksle.



3 pav. Srinto retrospektyvos eiga

*Srinto atšaukimas* – sprintas gali būti atšauktas nepasibaigus numatytai srinto trukmei. Atšaukti srintą teisę turi tik *Produkto šeimininkas*, jeigu srinto tikslas tampa nereikalingu. Taip gali atsitikti pasikeitus užsakovo veiklos kryptčiai, rinkos sąlygoms ar technologijoms. Tačiau dėl nedidelės trukmės srinto atšaukimas retai kada turi prasmę [3, 4].

*SCRUM* artefaktai parodo reikiamą atlikti darbą arba sukurtą galutinio produkto vertę. Galimi *SCRUM* artefaktai – *Produkto darbų sąrašas* (angl. *Product Backlog*), *Srinto darbų sąrašas* (angl. *Sprint Backlog*) ir *Prieaugis* (angl. *Increment*) [4]. *SCRUM* artefaktai apibūdinimą galima matyti 3 lentelėje.

3 lentelė. *SCRUM* artefaktų apibūdinimas

<i>SCRUM</i> artefaktas	Apibūdinimas
<i>Produkto darbų sąrašas</i>	tai visų kuriamo (ų) galutinio produkto savybių, reikalingų funkcijų, reikalavimų, tobulinimų ir taisymų įgyvendinimo darbai. Šis darbų sąrašas niekada nebūna baigtinis, jis keičiasi kartu su kuriama kintančiu galutiniu naudoti skirtu produktu
<i>Srinto darbų sąrašas</i>	tai bet kuriam vienam srintui priskirti darbai iš <i>Produkto darbų sąrašo</i> , kuriuos atlikus per srintą bus pasiektas pageidaujamas rezultatas, sukuriama galutinis naudoti skirtas produkto elementas, kuris laikomas baigtu
<i>Prieaugis</i>	tai visų <i>Produkto darbų sąrašo</i> darbų, baigtų srinto metu, bei visų ankstesnių galutinio produkto elementų visuma. Srinto pabaigoje galutinis produkto elementas turi būti baigtas ir tinkamas naudoti bei atitikti <i>SCRUM</i> metodo užduoties baigtumo apibrėžtį

Galutinio produkto rezultatas yra atskirų produkto kūrimo darbų ir užduočių įvykdymas. *SCRUM* metodo atveju kiekvienas darbas, užduotis *Produkto darbų sąrašė* vadinamas *virtotojo pasakojimu* (angl. *User Story*). Kiekvienas užsakovo pageidavimas ar vartotojo poreikis galutiniam produktui, jo savybei, funkcijai ar pan. remiantis *SCRUM* metodu turi būti aprašytas tokiu formatu:

*Aš kaip <virtotojo tipas>, noriu <išreiškiamas tikslas>, jei tai bus įgyvendinta aš galėsiu <aprašoma priežastis ir teigiamos pasekmės>* (angl. *As a <type of user>, I want <some goal> so that <some reason>*) [8].

Visi vartotojo pasakojimai, kurie gali būti įtraukti į *Produkto darbų sąrašą* ar *Srinto darbų sąrašą*, turėtų būti racionalūs ir sukuriantys vertę galutiniam produktui bei jo vartotojams. Vartotojo pasakojimų kokybei ir tinkamumui įvertinti plačiausiai naudojamas yra *I.N.V.E.S.T* kriterijų rinkinys [9], kurio kriterijų apibūdinimas pateikiamas 4 lentelėje.

4 lentelė. *I.N.V.E.S.T* kriterijų aprašymas

Kriterijus	Angliškas pavadinimas	Apibūdinimas
I – Nepriklausomas	<i>Independent</i>	Vartotojo pasakojimas turi būti kiek įmanoma nepriklausomas nuo kitų vartotojo pasakojimų, kad būtų galima įgyvendinti ir pristatyti atskirai. Taip sukuriama atskiros funkcijos ir reikalavimai galutiniam produktui
N – Derybinis	<i>Negotiable</i>	Vartotojų pasakojimą turėtų būti galima aptarti ateityje, turėtų būti paliekama terpė deryboms. Vartotojo pasakojimas turėtų atspindėti reikalavimo esmę, poreikį, o ne būdus ir reikalavimus kaip jį įgyvendinti

Kriterijus	Angliškas pavadinimas	Apibūdinimas
V – Vertingas	<i>Valuable</i>	Vartotojo pasakojimas turi sukurti pridėtinę vertę ir padaryti galutinį produktą geresniu, labiau naudotinu ir tenkinti užsakovo reikalavimus, vartotojo pagrįstą poreikį
E – Įvertinamas	<i>Estimable</i>	Vartotojo pasakojimas turi būti pakankamai suprantamas, kad jį galima būtų suskirstyti į užduotis ir įvertinti. Dažniausiai vertinama per tai, kokią problemą užsakovui, vartotojui tai padės išspręsti prizmę
S – Smulkus	<i>Small</i>	Vartotojo pasakojimas turi būti ne per didelis, kurį būtų galima įgyvendinti per kelias dienas (per 40 darbo valandų)
T – Testuojamas	<i>Testable</i>	Vartotojo pasakojimas turi tinkamumo kriterijų, kad nuspręsti ar jis gali būti laikomas baigtu ir tenkinančiu užsakovo reikalavimus, vartotojo poreikius

Kiekvienas vartotojo pasakojimas turi turėti kriterijus, kuriais apibrėžiama, kad su tuo pasakojimu susijęs darbas yra baigtas. Tai yra vadinama **baigtumo apibrėžtimi** (angl. *Definition of Done* arba sutrumpintai *DoD*) [4, 10]. Baigtumo apibrėžtis turi būti aprašyta ir sutarta su užsakovu, vartotoju iš anksto. Turi būti numatyta, pagal kokius kriterijus galutinis produktas bus testuojamas, koks galutinis rezultatas užsakovą, vartotoją tenkintų. Rekomenduojama nepradėti jokių galutinio produkto kūrimo darbų, kol nėra sutarta dėl baigtumo apibrėžties konkrečiam vartotojo pasakojimui.

Užsakovas, vartotojai dažnai nurodo ilgą sąrašą savybių, funkcijų, kurios turi būti galutiniame produkte, tačiau didesnės apimties galutiniai produktai gali būti kuriami etapais ir visuma sukuriama per kelis sprintus. Todėl kuriant galutinį produktą būtina naudoti prioretizavimo metodus, padedančius nuspręsti, kurios funkcijos, savybės bus kuriamos konkrečiame galutinio produkto vystymo etape. Vienas iš populiarių **prioretizavimo metodų** yra **MoSCoW** metodas [11]. Pagal šį metodą užsakovo, vartotojo reikalavimai galutinio produkto savybėms, funkcijoms suskirstomi į:

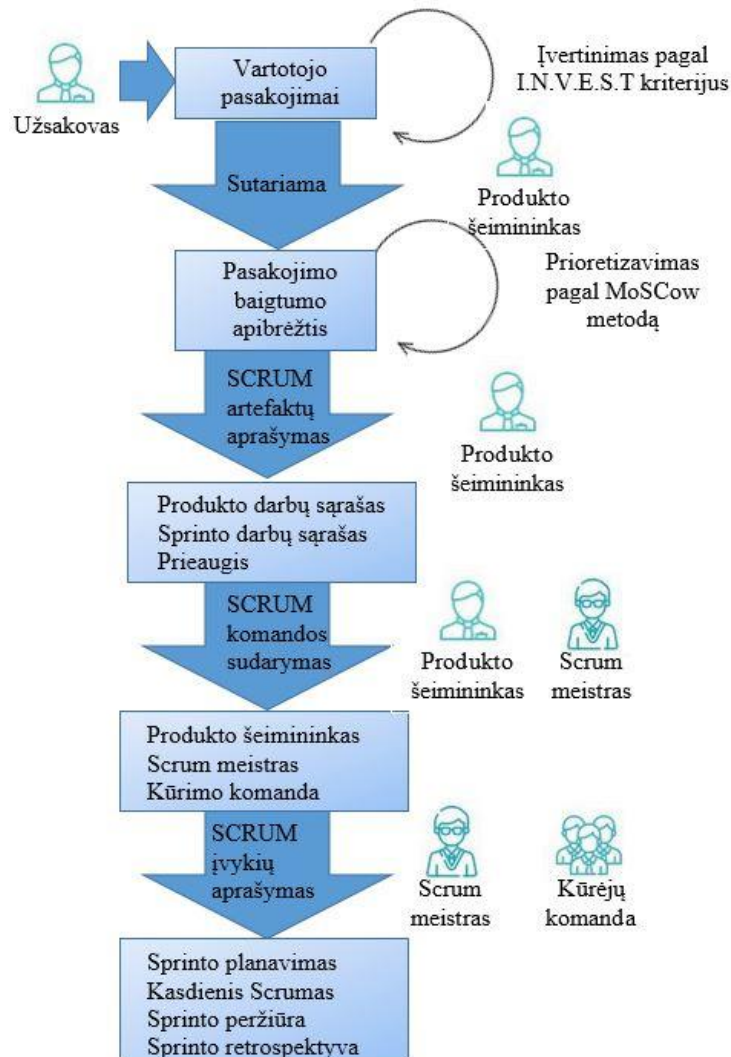
- **Mo** – Privalomi būti (angl. *Most Have*) – tai be jokių derybų privalomi reikalavimai kuriamam galutiniam produktui;
- **S** – Turėtų būti (angl. *Should Have*) – tai svarbūs reikalavimai, kurie nėra „gyvybiškai“ svarbūs, tačiau sukuria reikšmingą vertę galutiniam produktui;
- **Co** – Galėtų būti (angl. *Could Have*) – tai reikalavimai, kuriuos būtų gera turėti (angl. *Nice to have*), tačiau tai neturės didelės įtakos galutinio produkto patogumui naudoti, būtinam funkcionalumui;
- **W** – Nebus (angl. *Won't Have*) – tai reikalavimai, kurie tikrai nebus įgyvendinami konkrečiame etape, nes jie yra mažiausiai reikšmingi, svarbūs.

## 2. SCRUM metodo taikymas Utenos kolegijos Informatikos inžinerijos studijų krypties studentų baigiamųjų darbų praktinės dalies rengime

Susipažinus su **SCRUM** metodo komandos rolėmis, artefaktais, įvykiais, nutarta pabandyti tai panaudoti Utenos kolegijoje, siekiant patobulinti kolegijoje realizuojamos Informatikos inžinerijos studijų krypties Informacinių sistemų inžinerijos (ISI) studijų programos studentų baigiamųjų darbų praktinės dalies rengimo procesą ir išbandyti kaip šis metodas veikia praktikoje. Šio metodo taikymas įgalintų geriau susipažinti ir išsigilinti į užsakovo reikalavimus ar būsimų vartotojų poreikius kuriamam produktui bei įtakotų visų pageidaujamų funkcijų ir galimybių įdiegimą realizacijoje. Tuo pačiu tai skatintų studentų motyvaciją bei aktyvesnį įsitraukimą į produkto projektavimo ir realizacijos procesą bei inovatyvių problemų sprendimo priemonių paiešką. Šiuo tikslu pirmiausiai yra aprašomi **SCRUM** elementai, apibrėžiama kaip jie gali būti pritaikyti aptariamame procese – ISI studentų baigiamųjų darbų praktinės dalies rengime. **SCRUM** metodo taikymas vykdomas šiais etapais (schematinis vaizdas pateiktas 4 paveiksle):

1. Užsakovo reikalavimai, vartotojų poreikiai paverčiami **vartotojo pasakojimais**.
2. Vartotojų pasakojimai įvertinami pagal **I.N.V.E.S.T kriterijus**.
3. Kiekvienam vartotojo pasakojimui sutariama **baigtumo apibrėžtis**.

4. Produkto užsakovo reikalavimai, vartotojų poreikiai prioretizuojami pagal **MoSCoW metodą**.
5. Aprašomi **SCRUM** artefaktai: sudaromas **Produkto darbų sąrašas**.
6. Aprašomi **SCRUM** artefaktai: sudaromas **Sprinto darbų sąrašas**.
7. Aprašomi **SCRUM** artefaktai: nusprendžiama kas bus laikoma **Prieaugiu**.
8. Sudaroma **SCRUM komanda**: priskiriamos **Produkto šeimininko, Scrum meistro ir Kūrimo komandos** rolės.
9. Aprašomi **SCRUM įvykiai**, jų eiga, tikslas, laikas, dalyviai – **Sprinto planavimas, kasdienis Scrumas, Sprinto peržiūra, Sprinto retrospektyva**.
10. Po baigiamųjų darbų peržiūros katedroje, įvertinamas **SCRUM** metodo taikymo **efektyvumas**.



4 pav. **SCRUM** metodo taikymo rengiant ISI studentų baigiamųjų darbų praktinę dalį eiga

Nagrinėjamu atveju, Utenos kolegijos ISI studijų programos baigiamųjų darbų praktinės dalies rengimo **Produkto darbų sąrašas** – tai baigiamųjų darbų temų, užduotys, suskirstytos į kiek įmanoma mažesnes dalis, pagal reikalingas sukurti funkcijas, savybes, įvertinant visus užsakovo reikalavimus, vartotojų poreikius. Šis sąrašas bus parengtas tada, kai užsakovo reikalavimai bei vartotojo poreikiai bus tinkamai paversti vartotojo pasakojimais, įvertinti pagal *I.N.V.E.S.T* kriterijų rinkinį, turės baigtumo apibrėžtis bei bus įvertinti pritaikant *MoSCoW* prioretizavimo metodą.

Baigiamojo darbo užduotys – **vartotojo pasakojimai** turėtų būti surašytos pagal *Aš kaip <vartotojo tipas>, noriu <išreiškiamas tikslas>, jei tai bus įgyvendinta aš galėsiu <aprašoma priežastis ir teigiamos pasekmės>* formatą. Vartotojo pasakojimų pavyzdžiai ISI studijų programos



studentų baigiamojo darbo atveju galėtų būti: *Aš kaip <elektroninės parduotuvės vartotojas> noriu <turėti galimybę filtruoti prekes pagal pristatymo terminą> nes <tai aktualu, norint greitai pamatyti tas prekes, kurios gali būti pristatytos kuo greičiau>* arba *Aš kaip internetinės svetainės vartotojas, noriu, kad galėčiau prisiregistruoti prie sistemos saugiai, jei tai bus įgyvendinta, niekas kitas lengvai mano prisijungimo duomenimis nepasinaudos.*

Toliau užsakovo reikalavimai, vartotojų poreikiai studento kuriamam galutiniam produktui baigiamajame darbe – **vartotojo pasakojimai** – turi būti įvertinti **pagal I.N.V.E.S.T kriterijus**. Prireikus baigiamojo darbo vadovui (*Scrum meistriui*) būtina padėti studentui suderinti kriterijus su užsakovu. Pavyzdžiui *I.N.V.E.S.T* kriterijai smulkiems vartotojo pasakojimams – ISI studijų programos studentų baigiamojo darbo temos užduotims galėtų būti:

- **I** (Nepriklausomas) – *turi būti realizuota registracijos sistema, apmokėjimo sistema, užsienio kalbos pasirinkimo sistema ir pan.;*
- **N** (Derybinis) – *30 procentų užsakovo klientų yra iš Rusijos ar Latvijos, todėl puslapis su visa informacija ir rusų kalba yra būtinas, tačiau dėl 2 procentų Norvegijos klientų kurti puslapį su informacija norvegų kalba, neverta, nes jie gerai moka anglų kalbą;*
- **V** (Vertingas) – *apmokėjimo su MoQ galimybė, padarys svetainę patrauklią, tiems, kurie naudojami šiuo vis populiarėjančiu apmokėjimo būdu, suteiks pranašumą prieš konkurentus, kurie dar šios galimybės neįsidedė;*
- **E** (Įvertinamas) – *MoQ atsiskaitymo galimybė suteiks galimybę klientams greitai ir patogiai atsiskaityti mobiliuoju telefonu, nereikės prisijungti prie banko, pakaks tik nuskenuoti QR kodą;*
- **S** (Smulkus) – *registracijos svetainėje forma yra skirta tik registracijai, kontaktų-susisiekimų forma – tik bendravimui, todėl jos, kaip atskiros funkcijos, o ne viena, turi būti kuriamos ir testuojamos atskirai;*
- **T** (Testuojamas) – *registracijos forma testuojama generuojant laikinus besiregistruojančius vartotojus ir yra laikoma, kad ši funkcija yra baigta kurti, kai negalima prisiregistruoti naudojant slaptažodį, trumpesnę nei 8 simboliai.*

Kiekviena galutinio produkto kūrimo ar vystymo užduotis turi turėti su užsakovu ar galutiniu vartotoju sutartą **baigtumo apibrėžtį**, kuri turi būti aprašyta prie kiekvienos baigiamojo darbo užduoties. Nagrinėjamame ISI studijų programos baigiamųjų darbų praktinės dalies rengimo kontekste pavyzdžiu galėtų būti: *registracijos formoje prisijungimo, registracijos vardas turi būti sudarytas ne mažiau nei iš 10 simbolių, į elektroninį paštą turi būti siunčiama patvirtinimo nuoroda, tam kad užtikrinti, jog būtent tas asmuo registruojasi, o ne kitas naudojasi jo vardu ir duomenimis, slaptažodis turi būti sudarytas ne mažiau nei iš 8 simbolių, tarp kurių privalo būti bent viena didžioji raidė, skaičius ir specialus simbolis, telefono numeris turi būti įvestas su tarptautiniu kodu +370, o ne 8-etu, bei telefono numerį turi sudaryti 11 skaičių ir simbolis +.*

Sudarant **Produkto darbų sąrašą** ir įvertinant užsakovo išsakytus reikalavimus, vartotojo poreikius studento baigiamajame darbe kuriamam galutiniam produktui, pasinaudosime **MoSCoW** prioretizavimo metodu. Taip bus galima teisingai, pagal kiekvieno studento kompetenciją ir gebėjimus, įvertinti kas būtinai turi būti galutiniam produkte ir įtraukti į kurį nors sprintą, kas turi būti ir gali būti įtrauktas į sprinto darbų sąrašą kiekvienam studentui vėliau, kai bus įgyvendinti ir baigti pagal baigtumo apibrėžtį būtini užsakovo reikalavimai. Pavyzdžiui, baigiamuoju darbu kuriant internetinę svetainę privalomi įgyvendinti (**Mo, Must have**) darbai galėtų būti – *parengti informacijos pateikimo ir koregavimo galimybę, parengti internetinę svetainę, kad ji būtų vienodai atvaizduojama populiariausiomis interneto naršyklėmis, pritaikyta mobiliems įrenginiams; interneto svetainėje realizuoti vartotojų prisijungimo galimybę, turėtų būti (**S, Should have**) atliekami tokie darbai kaip, realizuoti vartotojų prisijungimo funkciją, įdiegti įvykių kalendorių, nuorodas į socialinius tinklus, galėtų būti atliekami (**C, Could Have**) tokie darbai kaip, realizuoti galimybę prenumeruoti naujienlaiškį, interneto svetainę pritaikyti naudoti žmonėms turintiems negalią, nereikšmingi darbai (**W, Won't Have**), kurie nebus atliekami yra sudaryti galimybę keisti interneto svetainės meniu juostos vietą. Studentų baigiamiesiems darbams rengti skirtam laikui einant į pabaigą, jei studentai yra įgyvendinę **Mo** ir **S** užsakovo reikalavimus, vartotojo poreikius jau įvykusių sprintų metų ir jie*

atitinka baigtumo apibrėžtį, galima likusiems sprintams skirti užduotis iš *Produkto darbų sąrašo*, kurių metu bus įgyvendinti užsakovo reikalavimai, priskirti prie *C* funkcijų. *W* reikalavimus, poreikius derėtų įtraukti į paskutinius sprintus, jei įvykusių sprintų peržiūros metu buvo nutarta, kad jie pateisino lūkesčius ir visa *SCRUM komanda* nutarė, kad laikas pereiti prie *W* reikalavimų įgyvendinimo. Nagrinėjamu ISI studentų baigiamųjų darbų rengimo atveju, pvz. kuriant IT inventorius apskaitos internetinę sistemą, *Produkto darbų sąrašas* galėtų atrodyti taip:

- sukurti vartotojo sąsajos dizainą ir elementų išdėstymo struktūrą;
- sukurti informacijos saugojimui duomenų bazę;
- realizuoti vartotojų prisijungimo funkciją;
- realizuoti duomenų įvedimo ir koregavimo funkciją;
- realizuoti galimybę nustatyti informacijos prieigos skirtingiems vartotojams teises;
- realizuoti užklausų vykdymo funkciją;
- realizuoti ataskaitų generavimo ir spausdinimo funkciją;
- parengti skolinamos įrangos registrą.

Sudarant *Sprinto darbų sąrašą*, kiekvienam sprintui per sprinto planavimo ceremoniją parenkamas galutinis studentų baigiamųjų darbų temų užduočių sąrašas – 2-3 užduotys kiekvienam 2 savaitių sprintui. Tai, ką studentai turi atlikti per kitas dvi savaites parenkama iš *Produkto darbų sąrašo*, kuris yra sudarytas iš smulkių užduočių – vartotojo pasakojimų, įvertintų pagal *I.N.V.E.S.T kriterijų* ir turinčių sutartą vienodai suprantamą baigtumo apibrėžtį. Atsižvelgiant į anksčiau minėtą *Produkto darbų sąrašą*, *Sprinto darbų sąrašas* galėtų būti: pagal užsakovo poreikius suprojektuoti *DB struktūrą*; parengti *IS loginę struktūrą*; sukurti reikiamas *DB lenteles*; sukurti vartotojų prisijungimo funkciją; parengti *DB elementų išdėstymą formose*; parengti užklausų vykdymo funkciją; parengti ataskaitų generavimo funkciją; parengti duomenų įvedimo ir redagavimo funkciją; parengti paieškos funkciją. *Sprintas* laikomas baigtu, jei studentai įgyvendino visas iškeltas užduotis, kurios leidžia užsakovui, vartotojui naudotis ta konkrečia galutinio produkto dalimi, funkcija, savybe. Pateiktu pavyzdžiu turėtų būti parengtas saugus vartotojo prisijungimas, sukurti vartotojo langai su valdymo elementais, vartotojas gali įvesti ir redaguoti jam leidžiamus duomenis, vykdomos reikiamos užklausos, generuojamos reikiamos ataskaitos.

Straipsnyje nagrinėjamu atveju *Prieaugis* – tai ISI studijų programos studentų baigiamojo darbo tikslą atitinkanti užduotis. Numatoma, ką studentai turi padaryti, tam kad įgyvendintų užsakovo reikalavimus ar vartotojo poreikius ir pasiektų išsikeltą baigiamojo darbo tikslą. Visų *Sprintų* pabaigoje galutinis produktas turi būti baigtas ir tinkamas naudoti. Be to, galutinis užsakovui ar vartotojui pristatomas produktas privalo atitikti sutartą produkto baigtumo apibrėžtį, kurią sudaro visų tarpinių užduočių baigtumo apibrėžtys, testavimo rezultatas. Atitikimą baigtumo apibrėžčiai turi patvirtinti tiek *SCRUM komanda*, tiek ir kitos suinteresuotos šalys – užsakovas arba vartotojas. Pavyzdžiui, Atlikus sukurtos *IS* testavimą matyti, kad visos pageidaujamos funkcijos ir galimybės veikia tinkamai.

Utenos kolegijos ISI studijų programos atveju *SCRUM komanda* galėtų atrodyti taip:

- *Produkto šeimininkas* galėtų būti Katedros vedėjas (-a) arba paskirtas už studijų programą atsakingas asmuo, kadangi galutinis produktas – tai visų konkrečios laidos studentų baigiamieji darbai;
- *Scrum meistrais* galėtų būti baigiamųjų darbų vadovai, kurie turės savo *Kūrimo komandą*, sudarytą iš jo temas pasirinkusių studentų. Praktikoje vienoje organizacijoje vienu metu vykdomi kelių produktų kūrimo darbai, tai kiekviena iniciatyva, užsakovas ar vartotojas gali turėti dedikuotus *Scrum meistrus*, kurie atsakingi už jiems paskirtų darbų įgyvendinimą;
- *Kūrimo komanda* – tai visi vieno vadovo studentai, kurie rengia baigiamąjį darbą.

Toliau remiantis projektų rengimo naudojantis *SCRUM* metodu praktika aprašomos *SCRUM* komandos funkcijos, kurias galima matyti 5 lentelėje.



**5 lentelė. SCRUM komandos narių funkcijų ir atsakomybių ISI baigiamųjų darbų rengimo kontekste aprašymas**

<b>SCRUM komandos rolė</b>	<b>Asmuo ar asmenų grupė</b>	<b>Funkcijos ir atsakomybės</b>
<i>Produkto šeimininkas</i>	Katedros vedėjas (-a) arba paskirtas už studijų programą atsakingas asmuo	valdo baigiamųjų darbų rengimo procesą; užtikrina, kad visi iš užsakovo surinkti reikalavimai ar vartotojų poreikiai būtų tinkamai aprašyti, kad juos vienodai suprastų tiek <i>Kūrimo komanda</i> (studentai), tiek ir <i>Scrum meistras</i> (baigiamojo darbo vadovas); patvirtina studentų darbų užsakovų ar būsimų vartotojų – <i>vartotojų pasakojimus</i> ; parengia <i>Produkto darbų sąrašą</i> ir pagal numatytus darbus, įvertina jų sukuriamą vertę; yra atsakingas už <i>Galutinių produktų sąrašo</i> sudarymą – visų baigiamųjų darbų temų užduočių patvirtinimą ir teisingą jų padalijimą į smulkesnes dalis – užduotis, jas įvertinant pagal <i>I.N.V.E.S.T</i> kriterijus; užtikrina, kad visos galutiniu produktu suinteresuotos šalys susitartų ir vienodai suprastų <i>baigtumo apibrėžtį</i>
<i>Scrum meistras</i>	Baigiamojo darbo vadovas	vadovauja <i>Kūrimo komandai</i> , prižiūri kaip jie laikosi metodo taikymo taisyklių ir praktikų; padeda studentams spręsti iškilusias užduočių įgyvendinimo problemas, konsultuoja studentus, organizuoja <i>kasdieninių Scrum</i> susirinkimus
<i>Kūrimo komanda</i>	Studentai	kuria ir vysto užsakovo reikalavimus ar vartotojo poreikius atitinkantį galutinį produktą; atlieka iš sudaryto <i>Produkto darbų sąrašo</i> konkrečiam <i>Sprinto darbų sąrašui</i> paskirtas užduotis; dalyvauja <i>kasdieninių Scrum</i> susirinkimuose

Planuojant ir organizuojant **SCRUM įvykius** rekomenduojama visą studentų baigiamajam darbui parengti skirtą laiką padalinti į sprintus, kurių trukmė dvi savaitės. Toliau turi būti suplanuojamas kiekvienas 2 savaitių trukmės sprintas. Pavyzdžiai galėtų būti, išanalizuoti *užduotį bei išsiaiškinti reikalavimus, išanalizuoti ir pasirinkti projekto realizavimo priemones, suprojektuoti ir sukurti duomenų bazę*.

Konkrečiau **Sprinto planą** kitoms dviem savaitėms pirmadieniais kuria visa SCRUM komanda, dalyvauja – Katedros vedėjas (-a) arba paskirtas už studijų programą atsakingas asmuo (kaip *Produkto šeimininkas*), baigiamojo darbo vadovas (kaip *Scrum meistras*) ir studentai (kaip *Kūrimo komanda*). Rekomenduojama organizuoti atskirus **Sprinto planavimo susirinkimus**, kuriuose dalyvautų tik *SCRUM meistrai* (baigiamojo darbo vadovai) bei jų vadovaujama *Kūrimo komanda* (studentai). Susirinkimo metu visi kartu turi nutarti, kurios užduotys ateinančias dvi savaites bus atliekamos. Komandų darbui, informacijos dalijimuisi bei užrašams gali būti naudojama komandinio darbo programinė įranga tokia kaip, *Stormboard*, *ScrumWorks*, virtualios mokymosi aplinkos (VMA) *Moodle* veikla *Vikis* ar pan. Vienu metu kiekvienai *Kūrimo komandai* (studentams) rekomenduojama priskirti 2-3 užduotis iš smulkaus jų temos užduočių (vartotojų pasakojimų) sąrašo, kad jie galėtų jas atlikti ir kad jos atitiktų baigtumo apibrėžtį. Susirinkimus galima organizuoti tiek kolegijos patalpose, tiek jungiantis nuotoliniu būdu naudojant vaizdo konferencijų programinę įrangą (VKPI) *Adobe Connect*, *Zoom*, *Skype*, VMA *Moodle* įrankį-veiklą *BigBlueButton* ar pan.

Taikant SCRUM metodą produkto kūrimo eigoje turi būti organizuojami **kasdieniai Scrumai**. Baigiamojo darbo vadovas (*Scrum meistras*) kiekvienos dienos pradžioje sutartu laiku, tarkim 9-tą valandą ryte, naudojantis suderinta VKPI *Adobe Connect*, *Zoom*, *Skype*, *BigBlueButton* ar pan. susirenka su studentais (*Kūrimo komanda*) aptarti praeitų 24 valandų rezultatus. Katedros vedėjas arba paskirtas už studijų programą atsakingas asmuo (*Produkto šeimininkas*) šiame susirinkime gali dalyvauti savo pasirinkimu, tačiau nors kartą per savaitę jo dalyvavimas pageidaujamas. Susirinkimo

metu kiekvienas studentas kas dieną papasakoja baigiamojo darbo vadovui: *ką atliko vakar; ką planuoja atlikti šiandien; ar yra kokių kliūčių, kurios trukdo atlikti sutartą darbą.*

Kiekvieno 2 savaitių sprinto pabaigoje penktadieniais kolegijos patalpose arba naudojant suderintą VKPI *Adobe Connect, Zoom, Skype, BigBlueButton* ar pan. visa *SCRUM* komanda – Katedros vedėjas arba paskirtas už studijų programą atsakingas asmuo, baigiamojo darbo vadovas, studentai – susirenka ***Sprinto peržiūrai***. Į šią peržiūrą taip pat galima pakviesti kitus *Scrum meistrus* (kitus baigiamųjų darbų vadovus), studentų kuriamų galutinių produkto užsakovus ar būsimus vartotojus. Rekomenduojama organizuoti atskirus susirinkimus su kiekvienu vadovu (*Scrum meistru*) ir jo studentais (*Kūrimo komanda*). Kiekvieno 2 savaitių sprinto peržiūros metu aptariama:

- Katedros vedėjas (-a) arba paskirtas už studijų programą atsakingas asmuo (kaip *Produkto šeimininkas*) paaiškina, kurie darbai iš *Produkto darbų sąrašo* buvo baigti, kurie dar ne;
- Studentai (kaip *Kūrimo komanda*) aptaria, kas sprinto metu vyko sėkmingai, kokios problemos ar kliūtys buvo iškilusios, kokie buvo jų sprendimo būdai. Baigiamųjų darbų vadovai papasakoja, ar studentai kreipėsi patarimo dėl iškilusių problemų sprendimo arba atsiradusių kliūčių pašalinimo, pasidalina patirtimi su kitais susirinkimo dalyviais, kaip būtų galima panašias problemas spręsti;
- Studentai (kaip *Kūrimo komanda*) demonstruoja baigtas realizuoti naudojimui skirto galutinio produkto funkcijas ir jų veikimą, atsako į susirinkimo dalyvių užduodamus klausimus. Vadovai vertina baigtas realizuoti užduotis ar jos atitinka sutartą baigtumo apibrėžtį;
- Katedros vedėjas (-a) arba paskirtas už studijų programą atsakingas asmuo (kaip *Produkto šeimininkas*) aptaria dar neatliktas *Produkto darbų sąrašo* užduotis (smulkias baigiamojo darbo užduotis), yra sutariama planuojamą darbų baigimo data;
- Visa susirinkime dalyvaujanti grupė aptaria, kas iš *Produkto darbų sąrašo* galėtų būti daroma toliau – kito artimiausio sprinto metu, tam, kad sprinto peržiūra pateiktų vertingą indėlį kito sprinto planavimo susitikimui. Studentai išreiškia pageidavimus kurias užduotis toliau norėtų atlikti.

Rekomenduojama rengti ir sprinto retrospektyvos susirinkimą, tam kad:

- aptarti kaip vyko paskutinis sprintas, kas buvo gerai, suprasti kokie sunkumai ar problemos buvo kilusios, kur daugiausiai strigo studentai, kaip jiems kuo greičiau galima padėti, kad jie kitame sprints galėtų lengviau atlikti užduotis;
- identifikuoti, kas konkrečiu metu yra ar buvo didžiausi trukdžiai;
- suderinti planą, kaip galima pašalinti trukdžius, jei reikia, galima įtraukti užsakovą ar būsimus vartotojus;
- įvardinti, kas labiausiai nesiseka studentams, ir paskirti mokymosi šaltinius tam, kad jie išmoktų atlikti tai, ko reikalaujama.

Utenos kolegijos ISI studijų programos baigiamojo darbo praktinės dalies rengimas trunka 2-2,5 mėnesio, todėl galutinio produkto kūrimo eigoje gali būti organizuojami 4-6 sprintai. Straipsnio rengimo metu *SCRUM* metodo taikymo etapai ir šiame straipsnyje aprašyti elementai buvo suplanuoti, tačiau realaus praktinio pritaikymo, t. y. galutinio produkto kūrimo, pabaiga numatyta vėliau, todėl išvadų apie metodo taikymo efektyvumą pateikti šiame straipsnyje galimybės nėra. Metodo taikymo efektyvumo rezultatus ISI studijų programos baigiamųjų darbų praktinės dalies rengimo procese planuojama aptarti kitame straipsnyje.

## Išvados

1. *SCRUM* metodas leidžia kurti ir vystyti galutinį programinės įrangos produktą efektyviai bendraujant visai komandai (*SCRUM* komandai – Katedros vedėjui (-ai) arba paskirtam už studijų programą atsakingam asmeniui, baigiamojo darbo vadovui, studentams, užsakovui ar būsimam vartotojui), kurią domina kuo geresnis galutinio produkto rezultatas. Kadangi Utenos kolegijos Informatikos inžinerijos studijų krypties studentų baigiamieji darbai yra užsakomieji arba pritaikomojo pobūdžio, *SCRUM* metodas leistų įtakoti galutinio produkto atitikimą užsakovo

- reikalavimams ar vartotojo poreikiams, tuo pasiekiant, kad įmonė ar organizacija gautų kokybišką ir jos poreikius atitinkantį ir praktiškai naudojamą produktą.
2. Produkto kūrimo darbų suskirstymas į smulkesnius etapus ir dažnas kuriamo produkto etapų aptarimas, sąvokų, užduočių, darbų atlikimo terminų vienareikšmiškumo suderinimas tarpusavyje bei peržiūrėjimas leidžia sukurti kokybišką bei užsakovo reikalavimus ar būsimų vartotojų poreikius atitinkantį galutinį produktą.
  3. *SCRUM* metodas nesunkiai gali būti taikomas Utenos kolegijos Informatikos inžinerijos studijų krypties studentų baigiamųjų darbų praktinės dalies rengime, tačiau šis metodas reikalauja didesnės visų, kuo geresniu galutinio produkto rezultatu suinteresuotų asmenų, motyvacijos nuosekliai vystyti galutinį produktą ir palaikyti pastovų tarpusavio ryšį nei įprastai, kai pradžioje yra suderinami reikalavimai ir tik po tam tikro 2-2,5 mėn. laikotarpio susitinkama aptarti galutinį produktą, kuriant užsakovo reikalavimus ar būsimų vartotojų poreikius atitinkantį programinės įrangos produktą. Taupant laiko sąnaudas realūs susitikimai gali vykti naudojantis suderinta VKPI *Adobe Connect*, *Zoom*, *Skype*, *BigBlueButton* ar pan., o produkto kūrimo komandų darbui, informacijos dalijimuisi bei užrašams gali būti naudojama komandinio darbo programinė įranga tokia kaip, *Stormboard*, *ScrumWorks*, virtualios mokymosi aplinkos (VMA) *Moodle* veikla *Vikis* ar pan.

### Naudotos literatūros ir šaltinių sąrašas

1. Vaida Steponavičienė, Jovita Urnikienė. AGILE projektų valdymo metodika IT projektų kūrimui aukštojoje mokykloje. *Aukštųjų mokyklų vaidmuo visuomenėje: iššūkiai, tendencijos ir perspektyvos*, Mokslo darbai, 2018, Nr. 1 (7), Alytus, ISSN 2029-9311
2. Jūratė Urbonienė, Donatas Bukelis. Lean metodologijos taikymas identifikuojant sprendimus Utenos kolegijos IT studentų baigiamųjų darbų analitinės dalies kokybei gerinti. Mokslo darbų žurnalas *Įžvalgos*, 2019 Nr. 2, p. 60-70, ISSN 2669-0330 (online)
3. Ken Schwaber, Jeff Sutherland. Galutinis Scrum gidas: žaidimo taisyklės, 2017. Prieiga internetu: <https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-Lithuanian.pdf>
4. Agile manifestas. Scrum gidas. Terminai. Vertimas iš: Ken Schwaber ir Jeff Sutherland Scrum gido (originalios versijos anglų kalba). Asociacija „Agile Lietuva“, 2014. Prieiga internetu: [http://www.agile.lt/uploads/editor/file/Scrum\\_gidas.pdf](http://www.agile.lt/uploads/editor/file/Scrum_gidas.pdf)
5. A Better Way Of Building Products. The Scrum Guide. Prieiga internetu: <https://www.scrum.org/resources/what-is-scrum>
6. Putting Agile and Scrum to work. Scrum Alliance 2018. Prieiga internetu: [http://info.scrumalliance.org/rs/510-STH-507/images/1250\\_WhitePaper\\_MediumRes.pdf](http://info.scrumalliance.org/rs/510-STH-507/images/1250_WhitePaper_MediumRes.pdf)
7. How to run Scrum efficiently in 2019? Quick guide for beginners. Prieiga internetu: <https://habr.com/en/company/hygger/blog/455022/>
8. User Stories. Prieiga internetu: <https://www.mountaingoatsoftware.com/agile/user-stories>
9. Bill Wake. INVEST in Good Stories, and SMART Tasks, 2003-2017. <https://xp123.com/articles/invest-in-good-stories-and-smart-tasks/>
10. Agile Practices. Definition of Done. Prieiga internetu: <https://www.leadingagile.com/2017/02/definition-of-done/>
11. Product Management MoSCoW Prioritization. Prieiga internetu: <https://www.productplan.com/glossary/moscow-prioritization/>

## **POSSIBILITIES OF APPLYING THE SCRUM METHOD IN PREPARING THE PRACTICAL PART OF THE FINAL THESES OF STUDENTS IN THE FIELD OF STUDIES OF INFORMATICS ENGINEERING OF UTENA UNIVERSITY OF APPLIED SCIENCES**

**Jūratė Urbonienė, Donatas Bukelis, Arūnas Pranaitis**

*Utena University of Applied Sciences, Faculty of Business and Technology*

*Maironio str. 18, Utena*

### **Summary**

SCRUM is a flexible programming project management method based on effective collaboration of the team involved in the project. This method is widely used in IT companies during software development projects. The article discusses the SCRUM method and its main elements, examines the possibilities of applying the SCRUM method to the preparation of the practical part of Utena University of Applied Sciences IT students' theses and provides practical insights to achieve the best possible end result - a student's thesis outsourced product (website, mobile application, system configuration document or etc.) implementation of harmonized functionality.

Keywords: SCRUM method, IT final thesis