

# Agilis módszertan kutatás-fejlesztés laborban

Ilyés Enikő

ilyese@inf.elte.hu  
ELTE IK

**Absztrakt:** Az agilis módszertanok iparban való elterjedtségét látva felmerülhet a kérdés: eredményesen alkalmazhatók ezek a módszerek az egyetemi kutató-fejlesztő munka keretében végzett szoftverfejlesztés esetében is? Mivel korábban sikeresen próbálkoztunk a Scrum használatával egy szoftverfejlesztést bemutató egyetemi kurzuson, 2018-ban megkíséreltük az agilis módszertanok alkalmazását a „txtUML” kutató-fejlesztő labor keretei között is. Ebben a cikkben e sajátos módszerünket mutatjuk be, rávilágítva azokra a kihívásokra, melyeket a végzett munka kutatás-fejlesztő jellege támasztott egy klasszikus szoftverfejlesztés gyakorlattal szemben.

**Kulcsszavak:** agilis, kutatás-fejlesztés labor, csapatmunka

## 1. Bevezetés

Az Eötvös Lóránd Tudományegyetem kutatóegyetem jellegének megfelelően az intézmény Informatikai Karán is több kutatás-fejlesztés csoport működik. Ezek keretén belül az egyetem munkatársai és hallgatói is rész vesznek kutatómunkában. A hallgatóknak van lehetőségük önkéntesen, ösztöndíj ellenében vagy tanulmányi kredit érték megszerzése céljából bekapcsolódni e laborok működésébe.

Egy-egy kutató-fejlesztő labor fókuszában általában egy nagyobb termék áll, melynek fejlesztése több kutatási kérdést vet fel. A kutató csoport tagjai ezt a terméket fejlesztik olyan módon, hogy egy-egy kutatási kérdést, -ágot egy-egy csoporttag/alcsoport jár jobban körbe, meglátásait megvitatja a többiekkel, majd eredményeit visszacsatolja a közös nagy termékbe.

A „Model Driven Development” kutatócsoport 2014 óta működik az Eötvös Loránd Tudományegyetemen. Egyik fő projektje a txtUML nevet viseli. A kivitelező csoport tagjainak rövid leírása a termékről: „A txtUML név egy mozaikszó, jelentése textual, executable, translatable – vagyis szövegalapú, végrehajtható és fordítható – UML. Az eszköz egy nyílt forráskódú szoftver, melynek célja a modell-vezérelt alkalmazásfejlesztés támogatása.

Szöveges (ld. forráskód) és grafikus (ld. UML diagramok) modellezési módszerek előnyeit egyesíti. Közvetlenül végrehajtható, diagramokon megjeleníthető és debuggolható, valamint más programozási nyelvekre fordítható UML modelleket lehet definiálni segítségével. A modellezés jól használható a szoftverfejlesztés korai, prototipizálási szakaszában. A támogatott nyelvek esetén az elkészült modellek akár eredeti formájukban is beágyazhatók meglévő rendszerekbe. Ilyen módon az alapul vett paradigmát akár a szoftverfejlesztés következő, általános célú programozási nyelvek feletti absztrakciós szintjeként is értelmezhetjük.” [1]

A txtUML csoport működésére az volt jellemző a megalakulást követő időszakban, hogy az ötletadó (Dévai Gergely) nagy lendülettel toborzott maga köré néhány hallgatót (4-5), akik rendszerint szakdolgozat, diplomamunka, TDK dolgozat megvalósítása keretében dolgoztak a termék egy-egy ágán. Ilyen módon a csoporttagok nagyon motiváltak voltak, az irányítás pedig erős, az ötletadónál centralizálódott. 2016-ban az ötletadó távozása és a más motivációval rendelkező csapattagok érkezése megváltoztatta a csoport működésének jellegét. A hallgatók nagy része „Szoftvertechnológia labor” tárgy teljesítése céljából kezdett el a csoportban dolgozni, ami azt eredményezte, hogy: fél évente sokkal több új tag csatlakozott a csoporthoz; akik projekt iránti elköteleződésére alapvetően fél év távlatában lehetett számítani (a tárgyfelvelet fél évre szól); és 4 kredit értéknek megfelelő munka volt tőlük elvárható (4 x 30 óra, azaz fél évente 120 óra). A korábban alkalmazott, erős moti-

vációra, hosszútávú elköteleződésre és erősen centralizált vezetésre alapuló módszer már nem bizonyult a munkaszervezés hatékony eszközének.

Mivel korábban sikeresen alkalmaztunk Scrum módszert klasszikus szoftverfejlesztés gyakorlati órán, az az ötlet született, hogy kezdjünk el agilis módszereket alkalmazni a txtUML kutatócsoport munkájának megszervezése céljából is. Az újszerű munkaszervezés megtervezése során azonban nyilvánvalóvá vált a kutatás-fejlesztés labor néhány olyan sajátossága, mely kétségessé tette, hogy a szoftverfejlesztés gyakorlat során már bevált módszerek módosítás mentes átörökítése lenne a leghatékonyabb.

### **1.1. Kutatás-fejlesztés labor sajátosságai a szoftverfejlesztés gyakorlattal szemben**

A kutatás-fejlesztés labor legkiemelkedőbb jellemzője a szoftverfejlesztés gyakorlattal szemben már a megnevezésben szembetűnik: maga a kutatás. Míg a fejlesztési feladatok térben és időben viszonylag jól körülhatárolhatók, becsülhetők, a kutatási feladatok sokkal nagyobb szabadságot kívánnak és több bizonytalanságot rejtnek. Ebből kifolyólag csak fölösleges nyomást okozhat az, ha szigorúan rögzített időkeretek közé próbáljuk szorítani teljesítésüket. Eszerint, míg a szoftverfejlesztés gyakorlat során alkalmazott Scrum és ezáltal a futamok rögzítettsége lendületet, ritmust és motivációt ad a csoport-tagok számára, a kutatás-fejlesztés labor esetében ez csak plusz terhet rak rájuk és korlátozza a kutatás esetében fontos egyéni szabadságot. Mivel korábban sikeresen alkalmaztunk Scrum módszert klasszikus szoftverfejlesztés gyakorlati órán, az az ötlet született, hogy kezdjünk el agilis módszereket alkalmazni a txtUML kutatócsoport munkájának megszervezése céljából is. Az újszerű munkaszervezés megtervezése során azonban nyilvánvalóvá vált a kutatás-fejlesztés labor néhány olyan sajátossága, mely kétségessé tette, hogy a szoftverfejlesztés gyakorlat során már bevált módszerek módosítás mentes átörökítése lenne a leghatékonyabb.

Egy másik sajátosság, hogy a szoftverfejlesztés gyakorlat hallgatói fél évet/egy évet vannak együtt, mindenki egyszerre csatlakozik a csoporthoz és egyszerre válik ki a csoportból a tárgy első, illetve utolsó óráján. A terméket általában az ötlettől kezdve ők építik fel, meglévő kódbázisra nem alapoznak. A kivitelezni kívánt szoftver teljességben megvalósítható a tárgy időkeretein belül. Ugyanez nem mondható el a kutatás-fejlesztés labor kapcsán. A hallgatók rendszerint egy már meglévő kutató-csoporthoz csatlakoznak és egy nagyobb kódbázissal való megismerkedéssel kezdik a munkát. Nem belátható, hogy mettől-meddig maradnak a kutató csoport tagjai (a minimum rendszerint fél év, maximum pedig a hátralévő tanulmányi éveik száma). Legtöbben nem látják a kutatás témáját egészében kibontakozni, csak a kutatás egy szakaszának lehetnek tanúi. Munkaszervezés szempontjából ez azért fontos, mert erőteljesen tekintettel kell legyünk arra, hogy a csapat egy-egy tagja saját folyamatait, szakaszait (pl. becsatlakozás a csapatba) kell összehangolja a teljes csapat folyamataival és ehhez kellő segítséget és motivációt szükséges biztosítanunk számára.

Látni kell azt is, hogy a szoftverfejlesztés gyakorlat tárgy hallgatói általában hasonló szintű előképzést követően kerülnek egy csoportba és a korábban elsajátított elméleti anyag gyakorlattá váltása a feladatuk. A kutatócsoportba elég különböző tudással rendelkező hallgatók kerülhetnek be és motivációjuk az elméleti és gyakorlati tudásban való fejlődés is lehet. Amikor tehát munkaszervezés módszert választunk a kutatás-fejlesztés labor számára, kellő lehetőséget kell biztosítanunk arra, hogy a tagok tanulhassanak egymástól, akár közösen gyarapodjanak elméleti és gyakorlati tudásban is.

Katrin Greßer és Renate Freisler „Agilis és sikeres vezetés” [1] című könyvéből olvashatjuk: „A vezetés nem más, mint csapatom számára keretet, s ezáltal biztonságot nyújtani, ugyanakkor a csapattagok kibontakozásához kellő szabadságot biztosítani.”; „A vezetők moderátorként funkcionálnak. A csapat kollektív intelligenciájára, szakértelmére és kompetenciáira támaszkodnak.” Tekintve a korábban bemutatott sajátosságokat, beláthatjuk: az agilis vezetés találó lehet a kutatás-fejlesztő

csoport jellegéhez, mindössze a megfelelő (szoftverfejlesztési gyakorlat esetében alkalmazottól némi-  
leg eltérő) munkaszervezési kereteket kell kialakítanunk.

## 2. Szakirodalmi kitekintés

Az agilis módszertanok egyetemi környezetben való alkalmazására részben klasszikus szoftverfejlesztési gyakorlatok módszertanának bemutatása kapcsán találunk példákat. Az Iowai Egyetem [2], Texasi Egyetem [3] oktatói Scrum módszer alkalmazásáról számolnak be. A Calgary Egyetemen [4] már Kanban és Lean módszerrel is bővül a paletta. A Marylandi Egyetemen [5] extrém programozással ötvözték a Scrumot, a Zürichi Egyetemen [6] hasonlóan. Saját Scrum-os tapasztalatainkat az „Esettanulmány: Agilis szoftverfejlesztés egyetemi kurzuson” című cikkünkben foglaltuk össze. [7]

A másik szakirodalmi csoport az agilis módszertanok kutatás-fejlesztés csoportban való alkalmazását taglalja. A „researchgate” honlapon [9] fórum keretében jelzik többen, hogy érdeklődnek Scrum módszer kutatócsoportban való használatára, vagy éppen már ki is próbálták. A vélemények megoszlanak a Scrum módszer és a kutatás-fejlesztés projekt jellegének összeegyeztethetőségét illetően. A Cseh Technikai Egyetem oktatója [10] a Scrum használatát az ipari partnerekkel való együttműködés szempontjából találta praktikusnak. A Minho Egyetem iFlow projektjében [11] ugyancsak ilyen céllal alkalmaztak Scrum-ot, viszont ők UML diagramok hangsúlyos használatával bővítették a módszertant, mert ezek értékes hozadékaról nem kívántak lemondani. A Lavras Egyetem [12] hét projekt kapcsán is hasznosnak nyilvánult a Scrum használatára, bár ennek kimutatására még nem álltak rendelkezésre megfelelő mérőszámok. A legnagyobb kihívás viszont már megfogalmazásra került: a csapattagokban való megfelelő hozzáállás (kommunikáció, együttműködés, közös felelősségvállalás) kialakítása.

## 3. Saját módszer

Ebben a fejezetben azt a sajátos agilis módszertant mutatjuk be, melyet a txtUML kutatás-fejlesztés laborban alkalmazunk 2018 tavaszi félévében. Ebben az időszakban összesen kilenc hallgató volt a csoport tagja, ezek közül három senior hallgató – vagyis olyan hallgató, aki már több mint egy éve dolgozik a kutató csoportban. Három hallgató teljesen új volt a csapatban, további három pedig már fél évvel korábban csatlakozott. A fejlesztő csapat munkáját egy projekt felelős (dr. Gregorics Tibor) és jómagam, mint módszertan felelős segítettük.

A továbbiakban szerepkörök, események és termékek mentén ismertetjük az alkalmazott módszert. Minden elem esetén tisztázzuk, hogy milyen célt szolgált.

### 3.1. Szerepkörök

Csapatunkban öt szerepkör érvényesül: projekt vezető, Scrum mester, technikai vezető, alcsoporthoz vezető, fejlesztő.

*Projekt vezető:* Képviselem a kutatócsoport céljait és létjogosultságát az egyetem vezetősége felé. Kapcsolatember a csoport eredményeinek széleskörű terjesztése érdekében (például: az egyetemi „Nyílt nap”, „Kutatók éjszakája” rendezvények esetén). Részt vesz a csoport találkozóin és ezen kívül is könnyen elérhető a csoport munkájával kapcsolatos döntések meghozása szempontjából. Szerepköre által biztos keretet és irányt ad a kutatócsoport működésének.

*Scrum mester:* A szerepkör megnevezése a Scrum módszertanból inspirálódott. A csoport működési értékeinek a védelmét hivatott biztosítani. Ezt különböző események kezdeményezésével, koordinálásával és példamutatással képes elérni. (például: kezdeményezi és koordinálja a napi Scrumot a hatékony megbeszélések érdekében; tréningeli a tagokat az elakadások hatékony jelentésére, stb.)

Részt vesz a csoport találkozóin és figyelemmel kíséri az egyes tagok csapatba való működésének folyamatát, valamint az egész csoport működésének dinamikáját.

*Technikai vezető:* Átfogó képe van a szoftverről, ugyanakkor sok részletben is jártas. Ő fogja össze a csapattagok munkáját technikai szempontból. Minden fontos technikai döntés vele is egyeztetendő. Mentoráló szerepe is van, vagyis arra hivatott, hogy átadja a többi fejlesztőnek a tudását, tapasztalatát, főleg az őket érintő feladatok kapcsán. Tetszés szerint fejlesztési-kutatási feladatokat is vállalhat. Elsődlegesen a csoport találkozóin elérhető, de gyakran ezen kívül, online kommunikáció által is válaszol a fejlesztők kérdéseire.

*Alcsoport vezető:* A txtUML csoport tevékenysége kapcsán négy különböző kutatás-fejlesztési feladatcsoport különíthető el. A következő megnevezésekkel hivatkozunk témájukra: nyelvi front-end; megjelenítés; C++ export; modelltesztelés. Azonos témakörű feladatokon dolgozó hallgatók alcsoportot alkotnak, melyet egy senior hallgató vezet, aki a témakörben leginkább otthon van. Ő jelöli ki azokat a kisebb kutatási-fejlesztési kérdéseket, melyen egy-egy csapattag dolgozhat és mentorálja a fejlesztőket ebben a folyamatban a csoport találkozó és online kommunikáció során. Rendszerint maga is fejleszt. A csoport találkozó alkalmával bemutatót és vitát szervezhez a saját témakörét illetően, hogy átadja tudását illetve alcsoportja eredményeit visszacsatolja a teljes kutatócsoporthoz.

*Fejlesztő:* A csapatba való csatlakozás első lépéseként minden új tag megismerkedik a txtUML-el felhasználói nézőpontból. Ezt követően kiválasztja azt a területet, mely kapcsán szeretne hozzájárulni a kutató-fejlesztő munkához és ezáltal egy alcsoport tagja lesz. Alcsoport vezetőjétől feladatokat kap, melyeken egyénileg (az alcsoport vezető mentorálásával) dolgozik. Részt vesz a csoport találkozóin, ahol figyelemmel kíséri a csoport haladását, tanul a többiektől és ő is átadja tapasztalatait. Kutató-fejlesztő munkával járul hozzá a szoftver termék bővítéséhez, finomításához.

## 3.2. Események

A txtUML kutatócsoport tagjaival hetente találkozunk egy két órát tartó ülés keretein belül. Ezek alkalmával valósulnak meg az agilis módszertanhoz tartozó események. A köztes időben mindenki egyénileg dolgozik a saját feladatán. Online konzultációra a slack (<https://slack.com/>) eszközt használjuk.

Az események kategóriába négy nagyobb egység tartozik: Előkészítés, Heti Rutin, Visszatekintés, Bemutató.

### 3.2.1. Előkészítés

A félév első időszakában előkészítés folyik. Ilyenkor a projekt vezető, technikai vezető és alcsoport vezető megbeszéli, hogy milyen fő irányok érvényesüljenek a félév során. Az új tagok ismerkednek a rendszerrel, a régebbi fejlesztők pedig jelzik, hogy milyen konkrét feladatokat vállalnának szívesen. A Scrum mester kiselőadás/tréning keretében ismerteti azokat az értékeket, melyek a csoport működésének hagyományához tartoznak és azokat a módszereket, melyeket a csoport a félév során használhat az értékek aládúcolására (Példa értékekre: hatékony csapatmegbeszélések, motiváció fenntartása, tudás átadása; példa ezeket alátámasztó módszerre: napi Scrum rituáléja.) Az előkészítés szakasz rendszerint az első 2-3 találkozó idejét veszi igénybe.

### 3.2.2. Heti rutin

Az előkészítés szakasz után egy sor olyan találkozás következik, melynek a felépítési váza a következő: hírek, napi Scrum, hét témája, vita közösen, vita kics csoportban.

A *hírek* részlegben elhangzik a csapattagok köszöntése; a Scrum mester jelzi a közösségnek, ha valaki hiányzást, késést jelentett; rövid reflektálás történhet a beküldött heti beszámolókkal kapcsolatban; a csoporttal kapcsolatos rövid hírek hangzanak el (Példa rövid hírre: “Felkértek bennünket, hogy vegyünk részt a kari Nyílt napon.” “Két csoporttársunk elnyerte a Kar Kiváló Hallgatója címet.

Gratulálunk nekik!”) Maximum 5 percet vesz igénybe ez a kis rituálé és segíti a hatékonyságot, fókuszálást indít el a tagokban.

A *napi Scrum* maximum 15 percet tartó rituálé. A technikai vezetők, alcsoport vezetők és a fejlesztők kötelező módon részt vesznek rajta. Minden tag három kérdésre válaszol röviden: „Mit valósítottam meg az elmúlt találkozó óta?”; „Mit tervezek megvalósítani a következő találkozóig?”; „Van-e valamilyen elakadásom?”. A kérdések és az időkeret szem előtt tartása abban segít, hogy lendületesen, hatékonyan haladjunk végig a csoporttagokat érintő kérdéseken. Ha ugyanis beleme gyünk olyan technikai részletekbe, melyek egy-egy tag számára fontosak, de a többi tag nem érintett bennük, akkor az utóbbiak könnyen nagyon unottá és demotiválttá válnak. Így tehát az ilyen jellegű tisztázni valóakra fény derül a napi Scrum folyamán, kulcs szavait a Scrum mester feljegyzi, de megbeszélésükre a találkozó második felében kerül sor, az érintettek, esetleg egyéb érdeklődők körében (*vita* közösen, *vita* kiscsoportban). A napi Scrum rövid tanulságok továbbadására viszont alkalmas terep, hiszen mindenki kiélezett figyelemmel vesz részt rajta. Másik hozadéka, hogy mivel a csapat-tagok hangosan fogalmaznak meg haladási ritmusukat illető gondolatokat, saját magukat és egymást is motiválják, lendületben tartják.

A *bét témája* rész a Scrum mesternek biztosít keretet arra, hogy a csoport értékeinek mélyítésén dolgozhasson a jelenlévőkkel. Általában 10-15 percet vesz igénybe és kiselőadás vagy tréning formájában valósul meg. A 2018-as első félév során olyan témák merültek fel, mint: egyéni motiváció; közös cél; közös felelősségvállalás; heti beszámoló hatékony megfogalmazása; munka becslése; bemutató tervezése; stb. Ezek átbeszélése hozzásegít ahhoz, hogy az új és régi tagok is elsajátítsanak olyan értékeket és módszereket melyek lehetővé tudják tenni, hogy egymás mellett dolgozó egyének helyett valódi csoportmunka alakuljon ki.

A *vita közösen* illetve *vita kiscsoportban* olyan kérdések körbejárására ad lehetőséget, melyek a napi Scrum során feljöttek, de a hatékony csoport-összshelyzet felmérés érdekében későbbre halasztottuk kibontásukat. Amennyiben mindenkit érintenek a csoportban (például egy komolyabb technikai döntés, mely a termék összképére van hatással) a leginkább bevonódott tagok felvezetik a megvitandó kérdést, majd mindenki együtt érvel annak váza mentén. Ilyen kérdésekből egy találkozó alkalmával több is feljöhethet, de akár egy sem. Ha csak egy pár személyt, esetleg csak egy fejlesztő-alcsoport vezető párost érint egy kérdés, akkor ők külön elvonulnak és meg tárgyalják azt. Példa erre, amikor a fejlesztő saját feladatát illetően elakad, vagy éppen befejezi azt és új feladatot kér. A *vita* rész kiváló lehetőséget nyújt a csapat tagoknak arra, hogy megosszák egymással tapasztalataikat, tudásukat egy adott téma kapcsán és együtt, egymás által fejlődjenek.

### 3.2.3. Visszatekintés

A visszatekintés során a csapattagoknak lehetőségük van reflektálni a közös munkára és együtt olyan irányokat, módszereket megfogalmazni, melyek bevetése vélhetően pozitív hatással lesz a következő munkaszakaszokra. A 2018.-as első félév során egy visszatekintő alkalom volt, a félév közepén, a hetedik találkozó alkalmával. A visszatekintést a Scrum mester koordinálta, aki két különböző színű, kis méretű papírlap csomagot osztott szét a csapattagok között. Az egyik szín a pozitív, a másik szín a negatív visszajelzéshez társult. Minden papírlapra szigorúan egy kulcsszót, gondolatot írhattak fel a csoporttagok, anonim módon. A visszagyűjtött papírlapokat „negatív/positív” illetve „technika/módszertan/csoport” tengelyek mentén rendszerezte a Scrum mester a táblán, egymás tetejére helyezve azokat a papírlapokat, melyek ugyanazt a gondolatot képviselték. Az így megjelenő kupacok arra utaltak, hogy egy-egy visszajelzést nagyobb súllyal érdemes kezelni, hiszen több csapattagot is hangsúlyosan érint. A visszajelzések áttekintését követően megbeszélés zajlott, mely során a csoport körbejárta a problémás területeket és ötleteket fogalmazott meg javításukat illetően. Egy konkrét példa: többen jelezték, hogy redundánsnak érzékelik az adminisztrációs feladatokat (heti jelentések írása, GitHub használata, napi Scrum). A megbeszélés során sikerült tisztázni, hogy mind-

egyiknek van külön hozadéka is, illetve a redundáns információkat hivatkozások használatával fel tudjuk oldani a továbbiakban.

### 3.2.4. Bemutató

A bemutató a félév záró alkalma, mely során a kutatás-fejlesztés labor minden tagja összegzi a félév során megvalósított munkáját és bemutatja eredményeit. Az alkalmon külső meghívottak is részt vehetnek. A menetrend logikusan felépített: a projekt vezető ismerteti a fő irányokat, a Scrum mester a munka menetének módszerét, a fejlesztők pedig -élen az alcsoport vezetőikkel- a megvalósított fejlesztéseket.

2018. július 2.-án került sor az idei bemutatóra, mely egyben a txtUML 0.7.0 release-ének bemutatója is volt. Az alkalmon régi csapattagok is részt vettek és nem maradt el a tortás-pezsngős ünnep sem, mely a közös sikert, mint motivációt hivatott erősíteni.

### 3.3. Termékek

A csoport munkájának elsődleges terméke a fejlesztett *kódbázis*, mely nyílt forráskód formájában elérhető a <https://github.com/ELTE-Soft/txtUML/> honlapon.

Ugyanezen a honlapon megtalálhatóak a szoftvertermékkel kapcsolatos feladatok, melyek *termékkívánságlistaként* is felfoghatók (issues fül alatt). Ezekhez adhatnak hozzá vagy ezekből válogathatnak a fejlesztők, alcsoport vezetők, technikai vezető – természetesen a projekt vezető által kijelölt irányoknak megfelelően.

Az alcsoportok mindegyikéhez tartozik egy-egy *tábla* (ugyancsak az említett honlapon, projects fül alatt), mely megjeleníti az alcsoportban éppen aktuális feladatokat. Olyan oszlopai vannak, mint: várakozó feladatok („to do”); fejlesztés alatt („in development”); tesztelés alatt („under testing”); felülvizsgálatra készen („pull request”); kész („done”). A tábla átfogó képet nyújt tehát az alcsoport munkájának jelenlegi állapotáról.

Ezeket kívül létezik egy belső honlap, ami tulajdonképpen egy űrlapot jelenít meg, ezen keresztül gyűjt egy háttér táblázatba információkat a tagok heti haladását illetően. A „*Heti beszámoló*” nevet viseli. Felületén egy elvégzett feladattal kapcsolatban a következő adatok adhatók meg: kivitelező neve; feladat típusa (pl. kutatás, fejlesztés, mentorálás, adminisztrálás, stb.); feladat rövid leírása; feladat kifejtett leírása (ha a rövid leírás nem elegendő); kapcsolódó GitHub feladat kódja (ha van ilyen); kiemelő tanulság (ha van ilyen). A heti beszámoló kitöltésére minden héten, a találkozót megelőző időpontban kell a fejlesztők időt szájanak és az előző találkozó óta végzett összes feladat kapcsán kitöltsék a megfelelő adatokat. Ennek kapcsán többször fogalmaztak meg a csapattagok ellenérzést, leginkább a túlzott adminisztráció vádja merült fel. Ennek ellenére megtartottuk a heti beszámolót a következő pozitív hatásaira tekintettel: mivel teljesítése megelőzi a napi Scrumot, elősegíti, hogy a csoporttagok összeszedetten, fókuszáltan számoljanak be a napi Scrum során; kutatáshoz és pályázat elszámoláshoz alkalmas adatsorokat állít elő; a később becsatlakozó csapattagok felzárkózásához nyújthat segítséget; önreflektáláshoz gyűjt adatokat – a személyes teljesítményt naplózza; olyan gyakorlat, mely fejleszti a csoportmunkához szükséges készségeket (együttműködés, szervezés, kommunikáció) és ráhangol a munkahelyi adminisztrációs követelményekre is.

## 4. Visszajelzések, eredmények

2018 tavaszi félévében 18 héten át alkalmaztuk a 3. fejezetben bemutatott módszertant. Megfigyeléseink a következők voltak: a heti találkozók hatékonyabbak, gördülékenyebbek és lendületesebbek lettek, mint korábban voltak; nőtt a kommunikációs interakciók száma a csapattagok között; az önálló munkavégzés mellett kezdett kialakulni a csoportos felelősség tudata; a rendszeres adminisztrálás kezdett elfogadottá, természetessé válni; tisztábbak lettek az egyéni és a saját célok.

A 7. heti visszatekintés során a csapattagok hasonló észrevételeket fogalmaztak meg. Pozitív jellemzőként emelték ki a csapatmunkát, mely ezúttal az al csoportok között is jobban érvényesült. Hárman is utaltak arra, hogy a célok összeszedése, világos megfogalmazása pozitívan hatott rájuk; a hatékonyabb kommunikációt is többen kiemelték. Rövidebb és jó hangulatú találkozókra számoltak be. Nagyon pozitívnak élték meg a csapattagokban lévő kompetenciát, mely a segítségnyújtással társulva sok tanulási lehetőséget nyújtott, főleg az új tagok számára. Ez utóbbi pozitív visszajelzés az alkalmazott módszertant annyiban illetheti, hogy az igyekezett teret biztosítani a tudás átadására a csoport struktúra (hierarchikus szerepkörök) és a heti rutin (főleg a napi Scrum és a közös/kiscsoportos viták) által.

Negatív visszajelzések a heti beszámoló kapcsán érkeztek elsősorban, melyet a csapattagok redundánsnak címkéztek. Panasz volt még a „folyamat túlformalizálása”. Mindkettőt némileg kezdeti kritikának éltük meg, vagyis az újdonságokkal szembeni alapvető emberi ellenállásnak. Reakciónk rá az volt, hogy adjunk még időt ezen elemek kapcsán, hogy pozitív hatásuk nyilvánvalóbbá válhasson hosszabb távon, addig pedig vita keretében mutassunk rá értékeikre. Ha későbbi visszatekintések is rávilágítanak még demotiváló hatásukra, akkor közös döntés nyomán természetesen elhagyhatók, módosíthatók.

A 17. héten a csapattagok kérdőíveket töltöttek ki a félévre vonatkozóan. Az ezek által begyűjtött információk is elemezhetők az alkalmazott módszertanra vonatkoztatva. A kérdőíveket 7 személy töltötte ki, vagyis a csapat 77 %-a. A számadatot bekérő kérdések esetén 1-től 5-ös skála állt a kitöltők rendelkezésére, ahol az „1” a legkevésbé, az „5” a leginkább szavaknak feleltethető meg. A módszertanra legdirektebb módon utaló kérdés volt: „Mit gondolsz, milyen mértékben függ össze a félév során megvalósított érték a projekt menedzselésével?”. A válaszok átlaga 3,71 volt. A „Mennyire jöttek át számodra az agilis módszertan értékei a közös munka során?” kérdésre ennél alacsonyabb átlagú válaszok érkeztek: 3,41. Ezeket mi nem tarjuk nagyon alacsony értékeknek, viszont ennél magasabbakat szeretnénk elérni. (Az agilis értékek átadására például ilyen céllal dolgoztunk ki egy külön tréninget, melyet a következő félévben már be is vetettünk.)

Voltak egyéb kérdések, amelyekre kapott átlagokat nagyon pozitívnak tekintünk és amelyek hátterében módszertani hatásokat is látni vélünk. Ezeket [kérdés-válaszok átlaga-magas átlag lehetséges módszertani alapja] hármaskban foglaljuk össze:

- „Mennyire tartottad (számodra) megfelelőnek a gyakorlat során általad megvalósított feladatot?” – 4,28 – A vita kiscsoportban és az al csoport vezetői mentoráló szerepe segítette, hogy a tagok személyre szabottan válasszanak feladatot és elégedettséget éljenek meg ennek kapcsán.
- „Mennyire volt megfelelő számodra a munkádra kapott visszajelzések száma, minősége (a félév folyamán)?” – 4,57 – A napi Scrum, vita közösen, vita kiscsoportban és az al csoport vezetői, technikai vezető mentoráló hozzáállása teret és keretet adott az egyénre vonatkozó visszajelzésekre.
- „Mennyire működött csapatként a csoport?” – 4,14 – A heti találkozók eleji közös hírfogadás (például közös gratuláció egy csapat tagnak), a napi Scrum mindenkit bevonó jellege, a közös értékeket tudatosító hét témája rész segítette, hogy kialakuljon a csapatszellem. A technikai vezető és az al csoportvezetők mentoráló hozzáállása is erősíthette az összetartozás érzetét.
- „Milyen volt a közérzet a csapatban a félév során?” – 4,71 – A heti rutin menete és a szerepkörök megfelelő kialakítása segítette, hogy mindenki elegendő figyelmet kapjon a csoportban és gördülékenyen tudjon a részeként működni.

Kiemelünk néhány szöveges formában megfogalmazott visszajelzést is. A legjobb tapasztalatok kapcsán írt kulcsszavak: „a csapatmunka, fejlődés, egymás ösztönzése”; „Közösen dolgoztunk egy érdekes feladaton.” Hisszük, hogy a módszertan hatékonysága is hozzájárult ahhoz, hogy a csapattagok legjobb tapasztalatként éljék meg a közös munkát. A „legnagyobb kihívás”-okat illetően felfi-

gyeltünk arra, hogy a fejlesztést és mentorálást együttesen végző csoporttagok nehézségekkel küzdenek az egyensúly megtartása kapcsán. Ehhez több segítséget szükséges nyújtanunk nekik a továbbiakban. A „min változtatnál” kategóriában a senior hallgatók eltávozásával (befejezik egyetemi tanulmányukat) kapcsolatos félelem is megjelent, mely utalás volt számunkra arra, hogy még több figyelmet kell szenteljünk a jövőben a projekt-tudás átadásának (seniortól fejlesztőnek irányba leginkább). Ugyancsak ebben a kategóriában fejeződött ki az igény a módszertan mélyebb megismerésére is mely további érv volt egy/több teljeskörű agilis módszertan tréning bevezetése mellett.

## 5. Összefoglalás

Az egyetemi környezetben működő kutatás-fejlesztés csoportok munkájának megszervezése érdekében inspirálódhatunk a szoftverfejlesztés iparban bevált agilis módszerekből. Több példát láthattunk arra, hogy ezek alkalmazása klasszikus szoftverfejlesztési gyakorlat órán sikereket hozott. Amikor viszont kutatás-fejlesztési laboron szeretnénk bevetni őket, szem előtt kell tartanunk néhány sajátos jellemzőt: a kutatás jellege nagyobb szabadságot kíván; kevésbé becsülhetővé teszi a feladatok méretét, a megvalósításukhoz szükséges időt; a kutatócsoportokba nagyobb rugalmassággal csatlakozhatnak be és válnak ki tagok; egy már meglévő kódbázissal és csoporttal szembesülnek az új tagok és egyéni ritmusukban kell tudjanak ehhez alkalmazkodni; nagyobb teret és konkrétabb lehetőségeket kell adni a tagoknak arra, hogy egymástól tanulhassanak, együtt fejlődhessenek a kutatás témáját illetően.

Ezeket a sajátosságokat szem előtt tartva dolgoztunk ki egy agilis módszertant a txtUML kutatócsoport számára. A projektvezető többnyire iránymutató szerepe kellő szabadságot és biztonságot ad a kutatáshoz, míg a Scrum mester a gördülékeny, ugyanakkor egyének számára rugalmasságot kínáló csapatmunkát segíti. A technikai vezető - alsó csoport vezető – fejlesztők hierarchikus lánc a tudás hatékony áramlását hivatott előidézni. Az előkészítés, hírek, hét témája, visszatekintés munkaszakaszok leginkább a csapat munka értékeinek tudatosítására biztosítanak terepet. A napi Scrum, a termék kívánságlista, a táblák és a heti beszámolók az átláthatóságot és ezáltal a hatékonyságot, motivációt kívánják erősíteni. A közös vita, vita kiscsoportban a tudás átadásának, az együttes fejlődésnek a bázisa.

A kipróbált módszertan kapcsán olyan visszajelzések érkeztek, miszerint hatékonyabb lett a kommunikáció a csoportban, tisztábbak a célok, kellemes a légkör, jobban érvényesül a csapatmunka, egyéni fejlődést élnek meg a tagok a tudás csapatban való áramlása által. A visszajelzések olyan továbbfejlesztési lehetőségekre is rámutattak, mint például az agilis értékek elmélyítését szolgáló tréning bevezetése. Összességében elégedettséget érzünk a módszertanunk kapcsán és továbbra is agilis vezetéssel kívánunk „hozzájárulni ahhoz, hogy emberek egy csoportja egyéni képességeit szennelvélyel és kreativitással értékek teremtésére használja.” [1]

## 6. Köszönetnyilvánítás

Köszönettel tartozunk a txtUML kutatócsoport minden tagjának, akik nyitottan és kísérletező kedvvel fogadták minden ötletünket saját módszerünk kialakítása érdekében. Készséges hozzáállásuk nélkül ez a tanulmány nem jöhetett volna létre.

A kutatási projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósult meg (EFOP-3.6.3-VEKOP-16-2017-00002).

## Irodalom

1. *Kutatócsoport: txtUML*  
<https://people.inf.elte.hu/ilyese/txtUML.html> (utoljára megtekintve: 2018.11.04.)
2. K. Greßer, R. Freisler.: *Agilis és sikeres vezetés*. Z-Press, Miskolc (2018)
3. J. G. Kuhl: *Incorporation of Agile Development Methodology into a Capstone Software Engineering Project Course*. In: College of Engineering at Iowa Research Online (ed.): 2014 ASEE North Midwest Section Conference, Iowa (2014)
4. T. Smith, K.M.L. Cooper, C.S. Longstreet: *Software Engineering Senior Design Course: Experiences with Agile Game Development in a Capstone Project*. In: ACM New York (ed.): ICSE 2011, Waikiki, Honolulu (2011)
5. C. Anslow, F. Maurer: *An Experience Report at Teaching a Group Based Agile Software Development Project Course*. In: ACM (ed.): SIGCSE'15, Kansas (2015)
6. D. F. Rico, H. H. Sayani: *Use of Agile Methods in Software Engineering Education*. In: IEEE Computer Society Washington (ed.): 2009 Agile Conference, Hannover (2009)
7. M. Kropp, A. Meier: *Teaching Agile Software Development at University Level*. In: IEEE (ed.): CSEE&T '13, San Francisco (2013)
8. E. Ilyés: *Esettanulmány: Agilis szoftverfejlesztés egyetemi kurzuson*. In: Webdidaktika alapítvány (ed.): INFODIDACT'2017, Zamárdi (2017)
9. *ResearchGate*  
[https://www.researchgate.net/post/Is anyone using Scrum in research projects at universities](https://www.researchgate.net/post/Is_anyone_using_Scrum_in_research_projects_at_universities) (utoljára megtekintve: 2018.10.04.)
10. M. Ota: *Scrum in Research*. In: Y. Luo (ed.): CDVE 2010, Calvia (2010)
11. N. Santos, J. M. Fernandes, M. S. Carvalho, P. V. Silva, F. A. Fernandes, M. P. Rebelo, D. Barbosa, P. Maia M. Couto, R. J. Machado: *Using Scrum Together with UML Models: A Collaborative University-Industry R&D Software Project*. In: O. Gervasi et al. (eds.): ICCSA 2016, Zhuzhou (2016)
12. I. R. Lima, T. de Castro Freire, H. A. X. Costa: *Adapting and Using Scrum in a Software Research and Development Laboratory* Revista de Sistemas de Informação da FSMA, 9. (2012) Trilha Principal (2012) 15-23