

КОМПАРАТИВНА АНАЛИЗА ОДАБРАНИХ КАРАКТЕРИСТИКА ТРАДИЦИОНАЛНЕ И АГИЛНЕ МЕТОДОЛОГИЈЕ РАЗВОЈА СОФТВЕРА

COMPARATIVE ANALYSIS ON SELECTED CHARACTERISTICS OF TRADITIONAL AND AGILE SOFTWARE DEVELOPMENT METHODOLOGY

Љубиша Мићин

Универзитет у Бањој Луци, Економски факултет
Република Српска, Босна и Херцеговина
Ljubisa.micic@ef.unibl.org

Абстракт: Модерни развој софтвера захтјева употребу једне или комбинацију више традиционалних или агилних методологија да би се постигла пуна ефективност и ефикасност. Међутим, разлика између ова два приступа није само у пројектном менаџменту, документацији и улогама. Ове разлике утичу на успјешност пројекта, задовољство клијената, мотивацију и интерес чланова развојног тима као и лакоћу схватања захтјева клијената. Рад анализира те разлике кроз неколико карактеристика развојних фаза пројекта и идентификује кључне изазове у планирању и имплементацији одговарајућих методологија развоја. Кроз рад је дат преглед литературе из ове области, експертског знања из традиционалних и агилних методологија као и компаративни преглед горе поменутих кључних разлика с циљем постизања што већег степена ефектности пројеката.

Кључне ријечи: Агилност, традиционално, waterfall, scrum, kanban

Abstract: Modern software development requires usage of just one or combination of traditional and agile software development methodologies in order to reach full efficiency and effectiveness. However, difference between those two is not just in project management, documentation and roles. Those differences effect project success, satisfaction of clients, motivation and interest of development team members as well as easiness of client project requirements. Paper analyses those differences in several characteristic project development phases and identifies key challenges in planning and implement appropriate development methodology. It is combination of literature review as well as expert knowledge in both, traditional and agile,

methodologies with outcome of review of above mention key differences in order to achieve as much as effective project outcome.

Key Words: agile, traditional, waterfall, scrum, kanban

УВОД

Савремени развој софтвера је у све већој мјери заснован примјеном неке од агилних методологија, међу којима се посебно истиче Scrum. Агилне методологије су вишеструко прихваћеније у ИТ компанијама и обично резултују квалитетнијом испоруком производа али и већом ефикасношћу и ефективношћу у самом развоју. Међутим, не постоје готова рјешења у одабиру методологија.

Нерјетко је дефинисање одговарајуће методологије и прилагођавање исте конкретном предузећу заиста озбиљан процес и захтјева анализу како самог предузећа тако и запослених, производа на којима се ради али и планова за будућност. То је основни мотив за овај рад – покушај дефинисања критеријума у одабиру одговарајуће методологије у развоју софтвера у ИТ компанијама као и фактора за процјену успјешности примјене исте. Међутим, само увођење неке од агилних методологија, у мањој или већој мјери, зависи од одређених предуслова који морају бити испуњени. Теорија познаје уобичајене предуслове али они могу одступати од регије до регије, предузећа до предузећа и наравно од врсте пројекта на којем се очекује имплементација одређене методологије. Такође, успјешност у примјени методологије нерјетко зависи од низа фактора, од организационих до физичких. То је такође дио проблематике коју третира овај рад. Надаље, успјешност примјене је увелико условљена профилима и особинама појединача који обављају кључне улоге у тиму. На примјер у Scrum-у, као најраспрострањенијој агилној методологији, те улоге су Scrum Мастер, Product Owner и чланови тима. Сваки члан тим би требало да посједује одређене особине да би примјена била успјешна. Идентификација тих особина, базирано на мишљењу стручњака са искуством у раду са Scrum-ом је један од циљева овог рада.

Уводни дио рада даје појашњења концепата агилног развоја софтвера, са нагласком на Scrum као најраспрострањенији методологију која је до сада показала најбоље резултате. Други дио рада даје приказ анкетног истраживања спроведеног на одабраном узорку стручњака са искуством у раду по различитим методологијама са простора Бање Луке и Босне и Херцеговине. Анкетирање је спроведено са циљем идентификације кључних карактеристика традиционалних и агилних методологија са аспекта стручњака који су примјењивали или примјењују исте.

Методологија која је кориштена у припреми рада обухвата:

- преглед литературе из научне области као и кориштење стручне литературе доступне на интернету,
- интервју са Scrum мастерима и члановима Scrum тимова,
- анкетно испитивање на одабраном узорку стручњака са искуством у примјени различитих методологија.

1. АГИЛНЕ МЕТОДОЛОГИЈЕ

Агилност се, бар судећи по Кембриџ и Оксфорд ријечницима, дефинише као способност брзог и лаганог кретања али и размишљања, дјеловања и рјешавања проблема. Из тих разлога није ни тешко закључити зашто се баш та ријеч употребила у називу специфичне групе методологоија карактеристичних за савремени развој софтвера. Агилне методологије обухватају читав сет методологоија и нерјетко се преплићу или користе заједно са оним традиционалним у ИТ предузећима фокусираним на ефикасност и ефективност у развоју софтвера. У агилним методологијама, али и традиционалним, као приоритетна и кључна проблематика се намеће прикупљање и дефинисање захтјева који су нерјетко немјерљиви, непрецизно дефинисани и без реалних ограничења у времену и простору. Примјена агилних методологоија у компанијама, како PriceWatersCoopers наводи (PriceWatersCoopers LL, 2014), у организацијама са малим или никаквим искуством у примјени агилних методологоија, је рад са високом ризиком за прекорачење рокова и одгањање лансирања производа.

Одабир одговарајућег оквира за развој софтвера зна бити од кључне важности у ефикасности али и ефективности пословања и постизању преузетих обавеза у задатом року. Одговарајући оквир и приступ може омогућити успјех кроз сарадњу и рад са стејхолдерим/ заинтересованим странама. Модерно глобално пословање, имајући у виду брзину којом се мјења, захтјева висок степен прилагођавања и прихватања модерних алата и модалитета пословања. Из тих разлога не чуди да су се баш агилне методологије истакле као најприхваћенији у развоју софтвера јер омогућавају баш то – висок степен прилагођавања промјенама уз уважавање и колаборацију са свим заинтересованим странама. То је од посебног значаја у ИТ компанијама фокусираним на развој софтвера јер се у њима, као рјетко гдје другдје, захтјева висок степен услаглашености модерних технологија и пословања.

Како PwC наводи у свом истраживању (PriceWatersCoopers LL, 2014), значај агилних методологоија можемо лако уочити кроз статистику и предвиђању Gartner-а и Forrester-а: 2000. године мање од 1% ИТ одјељења је чуло за агилне методологије док данас од 60 до 80% развојних тимова користи ове методологије као примарне у производњи софтвера. Такође, исти предвиђају да ће се тренд популаризације агилних методологоија наставити и да се очекује још већи проценат употребе ових методологоија у развоју софтвера. Међутим, иста студија наводи да и поред високог процента употребе, агилне методологије ипак не могу дати праве резултате без квалитетног општег менаџмента у планирању и комуникацији.

У агилним методологоијама почетни фокус је на оном шта треба да буде резултат тј. исход производње софтвера а мање енергије се усмјерава на како до тога доћи. Разлог томе јесте чињеница да се веома често, у методологоијама прије агилних, погрешно или недовољно прецизно дефинисао захтјев и атрибути

очекиваног производа што је на крају доводило да се вријеме узалудно трошило у раду на нечему што и није био захтјев клијента. Агилне методологије стављају фокус на прецизно дефинисање захтјева како би се вријеме и енергија тима уложила у креирање одговарајућег производа и/или сервиса који се очекује. Кроз итерације, тим усавршава и начин како да дође до жељеног производа тј. да повећа властити ефикасност кроз прецизно дефинисање активности, улога, рокова и слично. Али и овде је од велике важности како је производ који се очекује дефинисан и прецизност у томе свакако олакшава и рад на дефинисању методологије тј. конкретног начина развоја софтвера.

Такође, од великог је значаја и укљученост самог тима у процес јер кроз то они схватају о каквом производу се ради и какве карактеристике он треба посједовати. Основи циљ агилних методологија у процесу прикупљања и дефинисања захтјева али и испоруке кроз итерације јесте унапријеђење комуникације, усаглашености у раду али и укљученост клијента од почетка како би се постигли циљеви и потребе организације. Ола Сундин у својој брошури „Scrum у пет минута“, осим Scrum нао најпознатије агилне методологије наводи и Kanban, Extreme Programming (XP) и Lean Software Development уз напомену да се оне могу комбиновати и примјењивати заједно: Kanban за питања ефикасности, Lean за дефинисање принципа за развој организације, Scrum за организацију и планирање пројеката а XP као извршну методологију засновану на 12 принципа (Ола Сундин, 2015).

Упркос флексибилности одређене ствари требају бити што прецизније дефинисане као што су нпр.:

- Улоге унутар пројектног тима
- Временска ограничења (понекад ипак флексибилна али као оквир морају постојати)
- Непромјенљиве (нрп. да ли ће корисници производа бити само физичка лица)
- Поступан унос информација
- Правила понашања, рада и испоруке резултата
- Оквирни опис изгледа очекиваног производа (нпр. дизајн портала и сл.)
- Начин доношења одлука
- Алата који ће се у развоју користити

Од велике је важности и однос према евентуалним неуспјешним итерацијама. Битно је да се подстиче однос према лошијим извођењима и да се они не перципирају као неуспјех него да перцепција о њима буде као о „новој лекцији“ и још једном начину за унапријеђење и развој.

Неуспјешне итерације су у суштини шансе за унапријеђење процеса и поправку проблема у самом почетку како би се избјегле евентуалне дубље и комплексније грешке у наставку рада. PwC у својој анализи (PriceWatersCoopers LL, 2014) наводи да је од велике битности разумјевање самог процес у агилном развоју софтвера. Анализа даје и графички приказ процеса који се у зависности од

специфичности случаја може прилагодити и измјенити. Анализа даље наводи да је посебан значај у самом процесу планирања дефинисање функционалних захтјева односно шта прецизно систем треба да испоручује и од чега би се требао састојати. Препоручују да се до тих информација дође кроз разговор са корисницима кроз случајеве употребе. Са друге стране, ти иницијални разговори дају само оквир за рад а конкретни специфични захтјеви ће се дефинисати кроз сам развој. Те специфичности могу укључивати формирања, дизајн, попуњавање података, изглед и позиције на екрану, потврде уноса података и сл. Након самог дефинисања захтјева улази се у процес дефинисања историја за спринтове. У самом спринту од велике је важности стабилна и јасна комуникација са клијентом (индиректно преко Product Owner-а или директно са клијентом) са развојним тимом да би се, сада већ прецизније, дефинисало како систем треба да функционише.

То уједно даје клијентима и бољи увид у технички развој као и могућности одређених технологија које се користе у развоју софтвера. За разлику од традиционалних методологија за које се веже тзв. black box који карактерише одсуство клијента у самом процесу развоја, агилне методологије промовишу укљученост клијента у развој и његов допринос развоју. Тиме се добија, некад формално а већином суштинско, раније одобрење лансирања производа и повећава задовољство клијента.

Временско планирање и припрема документације, без обзира на флексибилност самог агилног развоја, ипак мора постојати у одређеној мјери. И поред чињенице да се сваки спринг посматра као засебна јединица посла, ипак је битно да постоји одређена документација и планирање, бар за конкретни дио посла, како би сваки спринт почео са првим даном када је планиран. Уобичајено је да се посљедња седмица сваког спринта посвети планирању наредног. Основни аспекти процеса прикупљања захтјева су јасна документација, садржај рада али и јасна контрола самог процеса. Поменута документација може обухватати листе, дијаграме, текстуална појашњења, сценарије кориштења и слично.

У овом дијелу процеса је од велике важности да сваки учесник има јасно разумјевање исхода који би требали да буду на крају. Главни учесници у овом дијелу процеса су менаџмент или други доносиоци одлука, девелопери и менаџери квалитета тзв. QA (Quality Assurance) који након дефинисања документације за сваки спринт морају имати слична ако не и иста поимања очекиваних исхода рада. Док документација треба да дефинише на чему се прецизно ради садржај треба да дефинише функционалности система, примјене и различите студије случаја. Такође, од велике је важности и сама контрола која заправо и дефинише агилност као такву. Тим се треба придржавати документације и правила рада како би се резултати на крају могли мјерити у складу са планираним а и како би се постигао жељени ниво ефикасности и ефикасности. Једном када се дефинише методологија сви чланови тима, не само

да је се морају придржавати, него би требало да имају и заједничко разумијевање исте.

Осим тога, битно је дефинисати датуме почетака, датуме завршетака али и датуме провјера и тестирања како би QA могао на вријеме да се припреми и квалитетно одради тестирања која су од велике важности и пружају самом процесу додатну дозу квалитета. Успјешно откривање грешака и дефеката система током самог процеса развоја је један од стубова самог агилног развоја. То и јесте вјероватно разлог зашто савремени агилни развој подразумева јаку улогу осигурања квалитета од планирања до испоруке производа. Такође, пријава грешака и неправилности се не очекује само од QA него и од сваког члана тима – то је не само препорука него и обавеза. Приликом детекције грешака се користе различите методологије: од откривања од стране QA преко стандардизованих процедура осигурања квалитета, до уочаваја грешака од стране развојног тима (кроз алате за интеракцију, кроз сам развој и сл.), путем print screen-ова и сл.

2. ТЕОРИЈСКА КОМПАРАЦИЈА АГИЛНИХ И ТРАДИЦИОНАЛНИХ МЕТОДОЛОГИЈА

Имајући у виду брзину којом се технологија мијења али и ефекат исте на пословање, посебно пословање ИТ компанија може се лако закључити да је потреба организационог прилагођавања и континуираног унапријеђење рада и више него очигледна. Модерно пословање захтјева унапријеђење повезаности и сарадње са клијентима/корисницима и то нерјетко баш уз употребу тих истих модерних технолошких алата. Употреба технологије постаје један од дијелова савремене тржишне конкурентности, нерјетко кључни дио пословних стратегија компанија. Континуирано унапријеђење је постало захтјев за компаније како би задржале своју тржишну позицију и евентуално унаприједиле свој положај.

Примјена агилних методологија ма колико била у тренду и потребна ипак зна бити и штетна за организацију, било да је прекомпликована или збуњујућа. То је донекле и оправдано јер је сам процес преласка са традиционалне методологије, која је сконцентрисана око документације и поприлично је унапријед дефинисана, на методологију која у фокус ставља захтјеве посла, тим и ефективност самог посла. Међутим, традиционалне методологије, као што је нпр. Waterfall карактерише „одвојеност“ фаза и недостатак комуникације из фазе у фазу што нерјетко доводи до развоја софтвера који уопште не одговара потребама клијента. Са друге стране, евентуални поновни почетак процеса може бити предуг и сувише компликован што може изазвати незадовољство клијената али и незадовољство унутар саме организације. У Waterfall-у у почетној фази имамо интезиван Waterfall и обично дуготрајан процес прикупљања захтјева међутим када се изађе из ове фазе комуникација са клијентом и бизнис аналитичаром скоро да престаје што онемогућава измјене у самим захтјевима које се нерјетко јављају у самом процесу развоја.

Обилна документација која настане као резултат фазе дефинисање захтјева се прослијеђује развојном тиму али се из фазе у фазу развоја прослијеђује остављајући могућност различитог дефинисања онога што су биле иницијалне потребе клијента. Унутар тима обично постоје различите дефиниције шта је заправо захтјев посла што оставља могућност различите интерпретације и неусаглашеног рада. То нас доводи до кључне мане Waterfall методологије али и других традиционалних методологија: недостатак комуникације и прецизности у раду што у крајњем исходу може довести до одгода лансирања производа али и вишеструко мање квалитетног производа који не одговара захтјевима клијента.

Укљученост тима у сам развој од почетне до крајње фазе унапријеђује њихово разумијевање очекиваног производа што утиче не само на ефикасност у раду него и повећава квалитет исхода рада. Са друге стране, циљ агилних методологија је испоруке производа вишег квалитета кроз промоцију колаборације и заједничког рада кроз инкрементални процес и максималну оптимизацију ресурса и буџета. Агилне методологије имају основни захтјев да унаприједе брзину реакције, повећају прецизност у дефинисању захтјева што све на крају треба резултовати испоруком производа који одговара захтјевима клијента у што већем обиму. Међутим, иако су агилне методологије иницијално настале из потребе веће флексибилности, нужност постојања структуре и пројектног планирања не губи на значају. Дапаче, пројектно планирање и дефинисање улога, обима посла и временских ограничења је чак и значајније у агилним методологија, додуше на другачији начин него у традиционалним методологијама.

Агилне методологије укупан обим посла који се очекује дијеле на мање дијелове ткз. Еpics (или популарно преузето и у српском језику: епици). Еpics дефинишу кључне сегменте који су потребни да се развију крзо пројекат. Еpics се даље дијеле на мање дијелове посла обично назване Stories (или опет преузето као оргинал у српски језик: сторие) унутар којих се дефинишу посебни задаци – Tasks, који су као и предходна два термина преузета у српски језик уз локално прилагођавања па се нерјетко умјесто задаци користи ријеч таскови. Оно што је од круцијалне важности у агилном пројектном планирању је дефинисање појединачних зависности и међусобних условљености што је свакако од велике важности у самој реализацији пројекта.

Такође, веома је значајно и правилно и прецизно одређивање приоритета у раду за шта су посебно заинтересовани сами чланови тима. Поменуто одређивање приоритета је увелико условљено захтјевима посла али и континуираној комуникацији са клијентима као једне од карактеристика агилних методологија. На крају имамо Sprints или српском језику прилагођене спринтове, који нису ништа друго него временски блокови кроз које реализујемо епике и сторие. Спринтови су оубичајени за Scrum агилну методологију иако се у другачијим називима појављују и у другим методологијама. Чланови пројектног тима у агилним методологијама морају имати заједничко и јасно шватање пројекта на

коме раде а то се дјелимично постиже и њиховима свакодневним укључивањем у само пројектно планирање, одређивање тежине задатака и слично. На самом почетку пројекта чланови пројектног тима дају заједничку процјену обима посла сваке од појединачних планираних активности. У овој фази пројекта тим настоји да дефинише колико сваки од захтјева посла заправо одузима времена за квалитетно извршење, који су модалитети контроле извршења, као и да успостави квалитетну и поуздану комуникацију са пројектним менаџментом. У оквиру ове фазе се прецизно дефинише основа за фазу развоја, процјене квалитета, тестирања и на крају испоруке клијентима. Прецизно дефинисање захтјева је од значајне помоћи самом тиму у процјени сложености посла и дефинисању његовом „score“-а тј. обима. Обим пројекта ће увелико дефинисати и сам Sprint Plan који је основни исход фазе планирања и почетка рада на пројекту. Оно што је повећава квалитет рада је свакако висок степен објективности тима у процјени сложености задатака али свакако и њихова свјесност да ће се захтјеви мјењати у самом развоју производа. За разлику од Waterfall-а код агилних методологија у почетној фази или више почетних фаза неће се доћи до прецизног и коначног обима захтјева али ће, уз квалитетан рад и напор самог тима, ипак на крају доћи до основног оквира који је, уколико је квалитетно припремљен, довољна основа развојном тиму за процјену очекиваних захтјева на крају развоја. У агилним методологијама почетна процјена је у суштини оквир за детаљно и прецизно дефинисање захтјева кроз саме спринтове. За тим је од изузетне важности да остане флексибилан у односу на промјене захтјева и додатне захтјеве како би се избјегле евентуалне фазе незадовољства и како би се што раније и што ефективније приступило наставку рада и на крају испоручио што квалитетнији производ. Код традиционалних метода је нерјетко то изазивало проблеме јер је тим сматрао да су сви захтјеви дефинисани кроз документацију и да је све додатно, а веома често се дешавало, заправо нови пројекат.

То не само да је утицало на квалитет испоручених производа него је и увелико утицало на само функционисање организације, задовољство тима и менаџмента али и на крају на укупно пословање и профитабилност оргнаизације. Такође, за разлику од агилних методологија гдје је процес прикупљања захтјева нерјетко трајао и по неколико мјесеци у агилним методологијама тај процес се сведе на неколико сати, посебно када је процес вођен од стране искусних пројект менаџера. Значај стручњака у области агилних методологија постаје водећа тема ИТ компанија широм свијета јер је од велике важности за успех предузећа. У опису њиховог посла је доношење одлука у кратким и ограниченим временским периодима, елеминисање губитака времена и енергије, ефективност и ефикасност пројектног тима али и успјешна и квалитетна комуникација са клијентима/корисницима производа. Обично је ријеч о професионалцима који не само да познају стручну област и технологије које се користе него су детаљно упознате са самом методологијом која се користи, квалитетима и манама чланова пројектног тима али и временским и сваким другим ресурсним ограничењима.

Табела 1 – Приказ основних разлика scrum и традиционалног пројект менаџмента (ScrumStudy, 2013) (превод: аутор)

	Scrum	Традиционални пројект менаџмент
Нагласак је на	Људима	Процесима
Документација	Минимална – само уколико је потребна	Обилмна
Стил процеса	Итеративан	Линеаран
Планирање унапријед	Скромно	Веома велико
Приоритетизација захтјева	Заснована на вриједносиза бизнис и редовно се ажурира	Фиксирано у пројектном плану
Обезбјеђење квалитета	Фокусирано на клијента	Фокусирано на процес
Организација	Самоорганизова-ност	Управљана организација
Менаџерски стил	Децентрализован	Централизован
Лидерство	Колаборативно, услужно лидерство	Наређивање и контрола
Мјерење перформанси	Вриједност за клијента	У складу са планом
ROI	На почетку или током пројектног живота	На крају пројектног живова
Укљученост клијента	Висока током цијелог пројекта	Варира, зависно од фазе животног циклуса

3. ИСТРАЖИВАЊЕ О СТАВОВИМА И ПЕРЦЕПЦИЈАМА СТРУЧЊАКА О СПЕЦИФИЧНОСТИМА РАЗЛИЧИТИХ МЕТОДОЛОГИЈА

За потребе овог рада, с циљем идентификације кључних разлика традиционалних и агилних методологија развоја софтвера, у периоду мај-јун 2017. године на простору БиХ, са фокусом на стручњаке из tech компанија, реализовано је анкетно испитивање.

Анкетиран је узорак од 42 ИТ стручњака који су упознати или су радили и са традиционалним и са агилним технологијама а који се налазе и/или су се налазили на позицијама пројектних менаџера, Product Owner-а, Scrum Master-а, ИТ предузетника/ власника компанија али и самих девелопера који су радили у окружењу које примјењује неки од горе поменутих приступа у развоју софтвера. Кључни резултати поменутог испитивања, посебно они повезани са идентификацијом разлика традиционалних и агилних методологија су приказани у наставку рада.

Илустрација 2 – Приказ одговора на питање колико искуства испитаници имају са различитим методологијама (ауторов допринос)



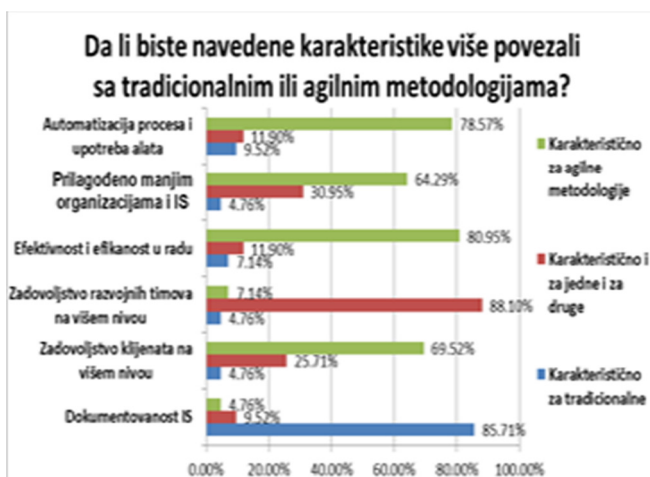
Око 80% испитаника је имало добро или солидно познавање традиционих и агилних методологија а ни један од испитаника није био без искуства у раду са једном од ове двије методологије. То је био резултат одабира циљаног узорка у локалним tech предузећима.

Илустрација 2 – Приказ одговора на питање о битности врсте методологије за успјешност пројекта (ауторов допринос)



Такође очекивано, већина испитаника и то преко 97% сматра да је одабир одговарајуће методологије битан за сам успјех пројекта а њих преко $\frac{3}{4}$ сматра да је сам одабир пресудно или веома битан за успјех пројекта. Имајући у виду да прва два графика потврђују да је одабрани узорак адекватан за дату тему у наставку анкетања је посвећена пажња идентификацији кључних разлика горе поменутих двију група методологија: традиционалних и агилних, од документованости ИС до задовољства клијената.

Илустрација 3 – Приказ одговора на питање о карактеристика специфичним за различите методологије (ауторов допринос)



Очекивано стручњаци су означили документованост као типичну карактеристику коју повезују са традиционалним методологијама а да карактеристике као што су задовољство клијената, ефикасности и ефикасност у раду и аутоматизација процеса ипак више везују за агилне методологије. Занимљиво је да су задовољство развојних тимова означили као карактеристику која не зависи од развојне методологије – скоро 90% стручњака сматра да ово задовољство не зависи од примјењен методологије.

Илустрација 3а – Приказ одговора на питање о повезаност карактеристика са одређеним врстама методологија (ауторов допринос)



Код питања за коју врсту методологије би повезали коју особину, испитаници су били сагласни да особине као што су снажна интеракција тима, прихватање и подстицање промјене, бољи испоручени производ као и континуирано

унапријеђење јесу карактеристике агилних методологија. Са друге стране, а што је уједно и кључна предност традиционалних методологија по одређеним стручњацима је дефинисаност финалног производа на самом почетку рада.

ЗАКЉУЧАК

Имајући у виду литературу која третира поменућу тематику као и кључне резултате анкетирања стручњака ван сумње се долази до неколико кључних закључака. Избор одговарајуће методологије јесте, нерјетко пресудно, битан за успјех самог пројекта. Приликом одабира методологије треба повести рачуна не само о врсти и величини информационог система него и о самом развојном тиму, карактеристикама клијента и његовом очекиваном задовољству, жељеној ефективности и ефиканости али и о пословној филозофији и начину рада саме организације (да ли је оријентисана на промјене и континуирано унапријеђење или преферира стабилност, сигурност и предвидивост и слично).

Стручњаци, што у већој или мањој мјери препознаје и сама литература, су идентификовали кључне карактеристике традиционалних методологија као што су документованост система, прецизно дефинисан жељени исход пројекта на самом почетку рада, фазност односно условљеност сваке фазе завршетком рада у предходној али и мане, као што су нерјетко недовољна ефективност и ефиканост у раду, минимална интеракција између развојног тима, нефлексибилност и немогућност или отежана могућност корекција у свакој наредној фази.

С друге стране, препознали су и кључне карактеристике агилних методологија као што су флексибилност и оријентисаност ка промјени, могућност прилагођавања клијенту током цијелог процеса развоја софтвера, већа укљученост и интеракција са клијентом у процесу развоја, снажна интеракција самог тима, подстицање на промјене како у самом развојном тиму тако и у организацији итд. што све у крајњем исходу води ка већем задовољству корисника што је вјероватно и један од разлога зашто се толики број организација одлучује за увођење агилних методологија.

Осим тога, стручњаци су указали на то да и поред теорисјких и практичних предности агилних методологија нерјетко се у пракси примјењују или хибридне методологије (комбинације агилних и традиционалних, у зависности о пројектима на којима се ради) или мјешавине различитих агилних методологија (нпр. Scrumban као мјешавина Scrum-а и KanBan-а).

Рад даје основу за даља истраживања с циљем идентификације конкретних изазова у раду са једном или другом врстом методологија али и за дефинисање кључних критеријума за одабир одређене врсте методологије у развоју информационог система.

ЛИТЕРАТУРА

- [1] Andrew Stellman, J. G., 2015. Learning agile. Sebastopol, CA: O'Reilly Media, Inc.
- [2] CMS - Center for a Medicare and Medicaid Services, 2008. <http://www.cms.gov>. <http://www.cms.gov/Research-Statistics-Data-and-Systems/CMS-Information-Technology/XLC/Downloads/SelectingDevelopmentApproach.pdf>
- [3] Jones, C., 2013. Evaluating Agile and Scrum with Other Software Methodologies, s.l.: InfoQ.
- [4] Krige, D., 2015. Selection criteria for a development methodology, s.l.: s.n.
- [5] N.N. Agile Manifesto, 2001. <http://www.agilemanifesto.org/iso/sr/>
- [6] PricewaterhouseCoopers LL, 2014 Adopting an Agile methodology
- [7] Sundin, O., 2015. Scrum Master Training. Banja Luka: Softhouse, Sveden.
- [8] VersionOne, 2013. 7th Annual State of Agile Development Survey, s.l.: VersionOne