

МЕТОДОЛОГИЈЕ УПРАВЉАЊА ПРОЈЕКТОМ СА АКЦЕНТОМ НА SCRUM PROJECT MANAGEMENT METHODOLOGIES WITH ACCENT ON SCRUM

Милица Говедарица, Факултет техничких наука, Нови Сад

Област – РАЧУНАРСТВО И АУТОМАТИКА

Кратак садржај - У раду су описане популарне методологије за управљање пројектом. Направљено је угрубо поређење двије главне групе методологија. Акцент рада је стављен на *scrum*, који је до детаља објашњен.

Кључне речи: *Методологије управљања пројектом, agile, scrum, sprint, backlog, product*

Abstract - *The paper gives an overview of the project management methodologies. The shallow comparison between the two main groups of methodologies is done. Accent of paper is on scrum, and scrum is explained to details.*

Keywords: *Project management methodologies, agile, scrum, sprint, backlog, product*

1. УВОД

Развијање модерних апликација ријетко је дјело појединца. Дизајнирање, кодирање, тестирање и паковање апликација захтјева *cross functional* тим (тим који се састоји од људи из различитих области) који чине дизајнери, програмери, тестери и графички дизајнери који заједно раде на задацима. Ово постаје још захтјевније узимајући у обзир чињеницу да су захтјеви чешће недефинисани и непотпуни него обрнуто. Комбиновање свега наведеног представља процес развијања апликације у циљу испуњавања очекивања корисника. Апликација треба да буде стабилна, да је достављена у временски остављеном року и унутар предвиђаног буџета.

Почетком 1970-тих креће развој разних методологија намјењених превазилажењу ових потешкоћа. На почетку најпопуларнија је била *waterfall* (водопад) методологија. Међутим, током 1980-тих, величина и сложеност апликација се повећава и расте потреба за флексибилношћу захтјева.

С једне стране ово значи да апликација неће у потпуности испуњавати захтјеве корисника, а са друге стране повећава се квалитет апликације и њена стабилност. Било како било, резултат ове методологије неријетко представљају апликације које не садрже карактеристике и функционалности неопходне да се постигне зацртани циљ, као и апликације које имају пуно мана због проблема с квалитетом.

НАПОМЕНА:

Овај рад проистекао је из мастер рада чији ментор је био др Александар Купусинац, ванр. проф.

2. ОСНОВНЕ МЕТОДОЛОГИЈЕ

Постоји много приступа управљању пројектом, али се посебно истичу два начина организације на које се касније ослањају остали. То су:

1. Традиционални приступ управљању пројектом и
2. Агилни

Традиционални приступ је линеаран, све фазе се изводе у секвенци. Пројекат је, унапријед, у потпуности испланиран без икаквог простора остављеног потенцијалним промјенама. Због ових разлога, традиционални приступ се сусреће са проблемима буџета и времена.

Агилни ставља акценат на тимски рад, сарадњу са клијентом и флексибилност. Представља итеративан процес, и труди се да у свакој итерацији спреми низ функционалности који може да буде показан клијенту.

Традиционални је најпогоднији за мале и једноставне пројекте, који прате линеарни приступ. Изненадне промјене или било какве компликације могу да блокирају читав процес и приморају тим да се врати уназад како би кренуо из почетка.

Агилни је најпогоднији за комплексне пројекте. Комплексни пројекти могу да имају више повезаних фаза и свака фаза може да буде зависна од много других за разлику од линеарног гдје једна фаза зависи тачно од једне друге фазе. Разлог због којег је агилни приступ бољи за комплексније пројекте јесте управо могућност адаптације.

На слици [1] је приказана табела поређења карактеристика традиционалног и агилног приступа.

Characteristics	Traditional Project Management	Agile Project Management
Organizational structure	Linear	Iterative
Project scale	Large-scale	Small and medium scale
Development model	Life cycle model	Evolutionary delivery model
User requirements	Clearly defined before coding or implementation	Interactive input
Client involvement	Low	High
Restart Cost	High	Low
Development process	Life cycle model	Evolutionary delivery model
Development model	Fixed	Easily changeable
Testing	Once coding is done	Every iteration
Architecture	Creates current and predictable requirements	Creates current requirements
Requirements	Standard and known in advance	Emergent with rapid changes

Слика 1. Традиционални наспрам агилног приступа

3. SCRUM

Scrum представља подкуп Агилне методологије, помоћу њега може да се имплементира агилно окружење. Пошто агилни процес нема кораке које треба испратити, **scrum** пружа начин да се унесе агилни процес у пројекат. Може да се користи као улазна тачка за друга агилна окружења. Није базиран само на софтверске пројекте, већ има доста ширу примјену.

Scrum је:

- Лаган
- Једноставан за разумјети
- Тежак за савладати

Scrum се користи при развоју софтвера, хардвера, уградбених система, интерактивних мрежа, аутономних возила, школа, влада, маретинга, управљања операцијама организација и скоро свему што користимо у свакодневном животу. Стручност **scrum**-а је у раду с комплексним проблемима. Посебно је ефективан при итеративном и инкременталном преносу знања. Суштину **scrum**-а чини мали тим људи.

Појединачан тим је веома флексибилан и лако се адаптира. Они настављају да раде сами, у склопу другог или других тимова који развијају, пуштају у промет, користе и одржавају производ који користи хиљаде људи. Они сарађују и заједно раде користећи софистициране развојне архитектуре и циљана окружења.

Оно што раздваја **scrum** од других агилних методологија јесу специфични концепти и праксе које су подјелени у три категорије:

- Улоге
- Артефакти
- Временске кутије

Scrum значајно повећава продуктивност и смањује вријеме развијања наспрам класичног **waterfall** процеса. Омогућава организацији да се прилагоди лако брзо мјењајућим захтјевима, и да произведе производ који испуњава еволуирајуће пословне захтјеве. Предности **scrum**-а:

- Повећање квалитета достављаних производа
- Боље подношење промјена(и очекивање промјена)
- Пружање бољих естимација, а при том коришћење мање времена да се постигну
- Боља контрола над производом као и његовим стањем

Три стуба којих свака имплементација емпиријске контроле процеса треба да се ослања су:

- Транспарентност
- Инспекција
- Адаптација

Scrum тим се састоји од:

- **Product owner**
- **Development team**
- **Scrum master**

Product owner је одговоран за повећање вриједности производа који је резултат рада **development** тима. Начин на који се ово ради варира од организације до организације, **scrum** тима до **scrum** тима и индивидуе до индивидуе.

Development тим се састоји од професионалаца који раде на достављању потенцијалних верзија инкремента **off-Done** које могу бити пуштене у коришћење на крају сваког **sprint**-а. Инкремент **off-Done** захтјева **sprint review**. Само чланови тима могу да креирају инкремент **off-Done**.

Scrum master је одговоран за промовисање и подстицање **scrum**-а као што је дефинисано у **Scrum Guide**. **Scrum master** ово постиже тако што помаже свима да разумију **scrum** теорију, правила и вриједности.

Срце **scrum**-а јесте **sprint**, спакован у временску кутију од мјесец или мање, током кога инкремент **off Done**, искористив и спреман. **Sprint** има константно трајање у току развојног труда. Нови **sprint** почиње одмах након закључка претходног **sprint**-а.

Циљ **sprint**-а представља оно што може да буде имплементирано из **product backlog**-а. Пружа смјернице тиму поводом тога зашто граде тај инкремент. Креиран је у току планирања **sprint**-а. Циљ **sprint**-а пружа мало флексибилности **development** тиму што се тиче функционалности које ће бити имплементирани. Селектоване ставке **product backlog**-а достављају једну кохерентну функционалност, која може да буде циљ **sprint**-а. Циљ може да буде било која кохерентност која захтјева да тим заједно ради на повезаним функционалностима наспрам оних засебних. Како тим ради, имају у уму циљ. Како би га постигли, имплементирају функционалности. Уколико се испостави да посао није оно што је очекивано, тим сарађује с **product owner**-ом о потенцијалним промјенама домена **product backlog**-а за текући **sprint**-а.

Scrum прописује четири формална догађаја за инспекцију и адаптацију:

- **Sprint Planning** – планирање спринта
- **Daily Scrum** – дневни састанци
- **Sprint Review** – ревизија спринта
- **Sprint Retrospective** – ретроспектива спринта

Планирање **sprint**-а даје одговоре на следећа питања:

- Шта ће бити одрађено у инкременту који представља резултат планираног **sprint**-а?
- Како да се постигне тај инкремент?

Daily Scrum је догађај ограничен на петнаест минута дневно у коме учествује **development** тим. Одвија се сваки дан у току **sprint**-а. На њему, тим планира посао који намјерава да одради у наредна двадесет-четири сата. Овако оптимизује сарадњу тима и перформансе прелазећи кроз посао урадјен у претходних двадесет-четири сата, тј. од последњег **daily scrum**-а, као и прогнозирање надлазећег посла. **Daily scrum** се одвија сваки дан на истом мјесту у исто вријеме, како би се смањила комплексност.

Структуру састанка поставља *development* тим, и може да буде извршена на различите начине уколико је фокус идаље на напретку ка циљу. Неки тимови ће да користе питања, други ће бити више окренути дискусији.

Примјери структуре:

- Шта сам јуче урадио како бих помогао тиму да постигне циљ?
- Шта ћу данас да урадим како бих помогао тиму да постигне циљ?
- Да ли видим неке препреке које спречавају мени или тим у достизању циља?

Чланови тима се често нађу након састанка како би детаљније продискутовали, или прилагодили, поново испланирали остатак посла.

Sprint review се одржава на крају сваког *sprint*-а како би се размотрио инкремент и адаптирао *product backlog* уколико је то потребно. У току овог догађаја, тим и дионичари разматрају оно што је постигнуто. Базирано на том и на свим измјенама, учесници разматрају следеће ставке које би могле да се одраде како би оптимизовале вриједност.

Ово је састанак информативног карактера, није састанак информисања о стању пројекта. Он такође служи за презентацију инкремента како би добили реакцију и потенцијалне измјене.

Ретроспектива представља могућност разматрања перформанси и прављење плана шта треба да се поправи за наредни *sprint*.

Product backlog представља сортирана листу која се састоји од свега што је потребно производу. Она је јединствени извор захтјева свих промјена које треба да се одраде на производу. *Product owner* је одговоран за *product backlog*, укључујући његов садржај, доступност и сортирање. Он није никад комплетан. Најранији развој креира преглед иницијалних и најбоље разјашњених захтјева. Напредује као што и производ и окружење у којем ће да се производ користи напредује. Динамички је, константно се мјења у циљу идентификације онога што је потребно производу како би био пристојан, компетитиван и користан. Уколико производ постоји, постоји и *product backlog*.

Sprint backlog је низ ставки из *backlog*-а селектоване за *sprint*, плус план како доставити инкремент производа и остварити циљ *sprint*-а. Представља прогнозу развојног тима о томе које ће функционалности да буду наредни инкремент и који посао је потребан да би се оне доставиле. Чини видљивим сав посао који развојни тим идентификује као неопходан за достизање *sprint* циља. Како би се осигурао константни напредак он укључује бар једну ставку унапређења високог приоритета, која је идентификована на претходној ретроспективи. Представља план са довољно детаља који се мјења са радом који је објашњен на дневном *scrum*-у. Развојни тим модификује спринт беклог у току спринта. Промјене се јављају како тим ради на плану и сазнаје више о послу који је потребан да се постигне циљ.

Инкремент представља збир свих ставки из *product backlog*-а које су завршене у току *sprint*-а и вриједност инкремента свих претходних *sprint*-ова. На крају *sprint*-а, нови инкремент мора бити *Done*, што значи да мора да буде у стању у ком може да се користи и да испуњава *scrum*-ову дефиницију *off Done*. Инкремент је тијело које се састоји из готовог посла који подржава емпиризам. Корак је ка циљу или визији. Он мора бити оспособљен за употребу без обзира да ли *product owner* одлучи да га достави.

Scrum се ослања на транспарентност. Одлуке да се оптимизује вриједност и контрола ризика су базиране на запаженом стању артефакта. Уколико је транспарентност комплетна, ове одлуке имају добру подлогу. У случају да није, онда ове одлуке могу да имају мане, вриједност може да опадне и ризици могу да се повећају.

Кад се ставка *product backlog*-а или инкремент прогласе за *Done*, сви треба да разумију шта то тачно значи. Дефиниција зависи од тима, али је кључно да сви разумију шта значи да је посао завршен, како би се осигурала транспарентност. Ово је дефиниција *off Done* за *scrum* тим, и користи се како би се процјенило који је посао готов.

Термини који се најчешће користе приликом рада унутар *scrum* окружења:

- **Daily Scrum** – кратки састанак на којем се описује шта се ради тај дан
- **Definition of Done** – кад је пројекат завршен и сусреће заједничка очекивања
- **Development Team** – одговорни за управљање послом који је повезан за сваки *sprint*
- **Forecast** – ствари коришћене за имплементацију *sprint*-а
- **Increment** – мали кораци који воде ка завршетку пројекта
- **Product Backlog** – посао који треба да се одради тачно задатим редом
- **Product Backlog Refinement** – када *product owner* и тим додају детаље *product backlog*-у
- **Product Owner** – менаџер који је одговоран за производ и тим
- **Ready** – обострано разумјевање *product backlog* када се планира *sprint*
- **Scrum** – окружење за сарадњу тима приликом рада на комплексним пројектима
- **Scrum Board** – визуелизација рада који је у току у виду табле
- **Scrum Master** – улога која је слична тренеру који помаже тиму коришћењем свог експертског знања
- **Scrum Team** – *product owner*, *development* тим и *scrum master*
- **Self-Organization** – аутономија тима унутар оквира пројектиних циљева
- **Sprint** – кратки задаци, секвенцијални, по завршетку једног одмах креће други
- **Sprint Backlog** – оно што тим треба да уради у *sprint*-у

- **Sprint Goal** – циљ *sprint*-а, сочно речено
- **Sprint Planning** – кратко планирање наредног *sprint*-а
- **Sprint Retrospective** – картка ретроспектива завршеног *sprint*-а
- **Sprint Review** – кратка оцјена *sprint*-а како би се унаприједио следећи
- **Stakeholder** – иницијатор пројекта који није члан тима
- **Velocity** – просјечна количина *product backlog* претворена у инкремент пројекта у току *sprint*-а

4. ЗАКЉУЧАК

Циљ овог раде јесте да помогне при одабиру методологије којом ће да се управља пројектом. На само питање како одабрати праву методологију има кратак одговор: зависи од пројекта. Углавном све се своди на повјерење, да ли клијенат вјерује агенцији да ће успјешно да достави пројекат, и да ли је спреман да плаћа неуспјех на путу до успјеха.

У крајности није битно за коју се методологију одлучите, докле год акценат остаје на оном што је заиста битно, а то је сам пројекат. Методологије служе само томе да олакшају процес достављања пројекта.

5. ЛИТЕРАТУРА

- [1] Jeff Sutherland and Ken Schwaber, Scrum Guide, 2010, <https://www.scrumguides.org/>
- [2] Jason Westland, Scrum Methodology, 2018, <https://www.projectmanager.com/blog/scrum-methodology>
- [3] Jeff Sutherland, Ken Schwaber and others, Agile manifesto, 2001, <http://agilemanifesto.org/>
- [4] Kira Carr, Agile Project Management Vs. Traditional Project Management, 2017, <https://www.knowledgehut.com/blog/agile/agile-project-management-vs-traditional-project-management>

Кратка биографија:

Милица Говедарица рођена је 1994. године у Требињу, Босна и Херцеговина. Завршила је гимназију „Перо Слијепчевић“ у Гацку 2013. године. Факултет техничких наука у Новом Саду уписала је 2013. године. Испунила је све обавезе и положила све испите предвиђене студијским програмом.