



ANALIZA PRIMENE AGILNOG PRISTUPA PROJEKTNOM MENADŽMENTU U SAVREMENOM POSLOVANJU

ANALYSIS OF AGILE APPROACH TO PROJECT MANAGEMENT IN MODERN BUSSINESS

Miljan Đumić, *Fakultet tehničkih nauka, Novi Sad*

Oblast – INŽENJERSKI MENADŽMENT

Kratak sadržaj – *Savremene poslovne strategije orijentisane su na stvaranje i zadržavanje vrednosti i prepoznaju važnost odnosa između upravljanja projektima i organizacije i upravljanja preduzećem. Kako bi sistem u celosti zadovoljio svrhu, neophodno je da se uskladi niz faktora koji obuhvataju radne procese, sisteme planiranja i izveštavanja i analize. U okviru rada biće analizirane prednosti ali i ograničenja savremenih oblika primene agilnog pristupa projektnom menadžmentu, a u cilju dobijanja što preciznijih informacija urađeno je i istraživanje.*

Ključne reči: *projektni menadžment, agilni pristup, savremeno poslovanje*

Abstract – *Modern business strategies are focused on creating and retaining value and recognize the importance of the relationship between project management and the organization and management of the company. In order for the system to fully meet its purpose, it is necessary to align a number of factors that include work processes, planning and reporting and analysis systems. The paper will analyze the advantages and limitations of modern forms of application of the agile approach to project management, and in order to obtain the most accurate information, research has been done.*

Keywords: *project management, agile approach, modern business*

1. REZIME

Agilni pristup je skup metoda razvoja softvera zasnovanih na iteraciji i postupni proces razvoja, gde zahtevi i razvoj evoluiraju kroz saradnju između samoorganizujućih, višefunkcionalnih timova koji omogućavaju brzu isporuku visokokvalitetnog softvera koji zadovoljava potrebe klijenta, a takođe se i prilagođava promenama u zahtevima.

Iako tradicionalne metodologije razvoja softvera, kao što su strukturirani i objektno orijentisani pristupi zasnovani na životnom ciklusu, i dalje dominiraju razvojem sistema nekoliko decenija i mnoga su istraživanja sprovedena u tradicionalnim metodologijama, Agilni razvoj softvera donosi svoj sopstveni skup novih izazova koji se bave

zadovoljenjem klijenta u smislu rane i kontinuirane isporuke softvera/proizvoda/usluge.

1.1. Cilj i predmet istraživanja

Cilj rada jeste da se istraži primena agilnog pristupa projektnom menadžmentu i različitih individualnih i organizacionih ishoda rada (opšte zadovoljstvo radom, percipirane prednosti rada, mogućnosti karijere i produktivnosti). U tom smislu će u radu biti identifikovani i opisani glavni faktori koji govore u prilog tome da agilni pristup razvoju poboljšava proces razvoja softvera kako bi se udovoljilo dinamičnim poslovnim okruženjima. Takođe će biti urađeno poređenje agilnih metodologija razvoja sa tradicionalnim metodologijama razvoja sistema i analiza trenutnog stanja usvajanja agilnih metodologija.

2. AGILNI PRISTUP RAZVOJU SOFTVERA U POREĐENJU SA TRADICIONALNIM

Agilni pristup razvoju softvera (engl. *Agile software development - ASD*) je glavna paradigma u polju softverskog inženjerstva koje je industrija široko prihvatila, a tokom protekle decenije sprovedena su mnoga istraživanja o agilnim metodologijama razvoja.

Tradicionalni pristup razvoju softverskih metodologija prati generičku inženjersku paradigmu zahteva, dizajna, gradnje i održavanja. Ove metodologije se nazivaju i vodopadima zasnovanim na klasičnoj paradigmi razvoja softvera. Takođe su poznata pod mnogim drugim imenima poput vođenih planom, vođenih dokumentacijom, teške metodologije i dizajn unapred [1].

| Ključna pitanja | Tradicionalni pristup | Agilni pristup |
|------------------------|---|---|
| Životni ciklus razvoja | Linearno; Model vodopada životnog ciklusa, spirala ili neka varijacija | Iterativni; Evolutivni model isporuke |
| Stil razvoja | Predvidljiv | Prilagodljiv |
| Zahtevi | Rano se mogu saznati, uglavnom stabilni; Jasno definisani i dokumentovani | Hitna promena, brza promena, nepoznata - Otkrivena tokom projekta |

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Bojan Lalić, red. prof.

| | | |
|--------------------------------|--|---|
| Arhitektura | Teška arhitektura za trenutne i buduće zahteve | YAGNI pravilo – “You aren’t going to need it” („Neće vam trebati“) |
| Menadžment | Procesno orijentisan; komanda i kontrola | Orijentisan na ljude; liderstvo i saradnja |
| Dokumentacija | Teška / detaljna eksplicitna znanja | Laka (zamenjena komunikacijom licem u lice); Prečutno znanje |
| Cilj | Predvidljivost i optimizacija | Istraživanje ili adaptacija |
| Članovi tima | Distribuirani timovi specijalista; Pristup eksternim znanjima i adekvatnim veštinama; planski orijentisani | Agilni, upućeni, zajednički i saradnički nastrojeni |
| Organizacija tima | Pre-strukturirani timovi | Timovi koji se samoorganizuju |
| Uključenost klijenta | Niska uključenost; pasivnost | Klijent na licu mesta i smatra se članom tima; Aktivno / proaktivno |
| Organizaciona kultura | Kultura komandovanja i upravljanja | Kultura liderstva i saradnje |
| Proces razvoja softvera | Univerzalni pristup i rešenje za obezbeđivanje predvidljivosti i visoke sigurnosti | Fleksibilan pristup prilagođen kolektivnom razumevanju kontekstualnih potreba kako bi se omogućio brži razvoj |
| Merenje uspešnosti | Usklađenost sa planom | Isporučena poslovna vrednost |

Tabela 2: Tradicionalne i agilne perspektive razvoja softvera [2]

Pojedini autori iz oblasti projektnog menadžmenta su došli do zaključka da su se tokom njihovog iskustva u razvoju projekata zahtevi često menjali za 25% ili više. Zbog stalnih promena u tehnologiji i poslovnim okruženju, izazov je za tradicionalni projektni menadžment da stvori kompletan set zahteva unapred [3]. Pojedini autori ističu i da je jedan od problema

tradicionalnog projektnog menadžmenta nesposobnost da se odgovori na promene koje često određuju uspeh ili neuspeh softverskog proizvoda.

3. INOVATIVNE AGILNE METODOLOGIJE

Postoji najmanje desetak inovativnih agilnih metodologija koje dele vrednosti i principe, ali se razlikuju u svojim naglascima. Stručnjaci često kombinuju različite pristupe, a u narednom delu teksta tabelarno su upoređeni oblici koji su najpopularniji i konteksti u kojima svaki najbolje funkcioniše.

| Okvir | Planirano olakšanje |
|----------------|---|
| Scrum | <ul style="list-style-type: none"> • Čitav obim posla podeljen je na razvoj kratkih ciklusa - Sprintova. • Trajanje Srinta je od dve do četiri nedelje. • Tim treba striktno da poštuje plan rada za svaki Sprint. • Ljudi uključeni u projekat imaju unapred definisane uloge. |
| Kanban | <ul style="list-style-type: none"> • Razvoj je zasnovan na vizuelizaciji radnog toka. • Trenutni rad (rad u toku) je prioritet. • Ne postoje vremenski ograničeni razvojni ciklusi. • Tim može promeniti plan rada u bilo kom trenutku. |
| Hybrid | <ul style="list-style-type: none"> • Agilni pristup i vodopad se dopunjuju. • Agilan razvoj softvera se održava pod uslovima Vodopada (fiksni rok, predviđeni budžet i temeljna procena rizika). |
| Bimodal | <ul style="list-style-type: none"> • Postoje dva odvojena načina rada - tradicionalni (režim 1) i agilni (režim 2). • Dva odvojena tima rade na projektima sa dva različita cilja. • Tim Mode 1 održava infrastrukturu IT sistema. • Tim Mode 2 isporučuje inovativne aplikacije. • Važna je saradnja među timovima. |
| Lean | <ul style="list-style-type: none"> • Okvir promoviše brzi razvoj softvera sa manje napora, vremena i troškova. • Razvojni ciklus je što kraći. • Rano isporučeni proizvod se kontinuirano poboljšava. • Tim je neovisan i ima širi spektar odgovornosti od onih u Scrum-u, Bimodalu i Hibrid-u. • Programeri takođe mogu da formulišu koncept proizvoda. |

| | |
|-----------|--|
| XP | <ul style="list-style-type: none"> • Fokus je na tehničkim aspektima razvoja softvera. • XP uvodi inženjerske prakse usmerene na pomoć programerima da napišu jasan kod. • Razvoj proizvoda uključuje dosledne faze: pisanje jezgra, testiranje, analiza, projektovanje i kontinuirana integracija koda. • Komunikacija licem u lice unutar tima i uključivanje klijenta u razvoj je presudno. |
|-----------|--|

Tabela 3: Ključne karakteristike agilnih metodologija [4]

Agilni razvoj softvera je krovni termin za širok spektar okvira i tehnika, koji dele iste principe i vrednosti. Svaka od njih ima svoja područja upotrebe i karakteristične osobine. Najpopularniji okviri su *Scrum*, *Kanban*, *Hibrid*, *Lean*, *Bimodal* i *XP*. U tabeli 3. prikazane su njihove ključne karakteristike.

3. SCRUM

Scrum je okvir koji pomaže ljudima, timovima i organizacijama da generišu vrednost kroz prilagodljiva rešenja za složene probleme.

Ukratko, Scrum zahteva Scrum Master-a da podstakne okruženje u kome:

1. Vlasnik proizvoda nalaže rad za složeni problem u zaostale proizvode.
2. Scrum tim pretvara izbor dela u Povećanje vrednosti tokom Sprintsa.
3. Scrum tim i njegove zainteresovane strane pregledaju rezultate i prilagođavaju se za sledeći sprint.
4. Ponavljanje [5].

Scrum je jednostavan i namerno nepotpun, definišući samo delove potrebne za primenu Scrum teorije; zasnovan je na kolektivnoj inteligenciji ljudi koji ga koriste. Umesto da ljudima daju detaljna uputstva, pravila Scrum-a vode njihove odnose i interakcije.

U okviru se mogu koristiti različiti procesi, tehnike i metode. Scrum se obavlja oko postojećih praksi ili ih čini nepotrebnim. Scrum čini vidljivom relativnu efikasnost trenutnog upravljanja, okruženja i radnih tehnika, tako da se mogu izvršiti poboljšanja.

Teorija skruma

Scrum je zasnovan na empirizmu i vitkom razmišljanju. Empirizam tvrdi da znanje potiče iz iskustva i donošenja odluka na osnovu onoga što se primećuje. *Lean* razmišljanje smanjuje otpad i fokusira se na najvažnije.

Scrum koristi iterativni, inkrementalni pristup za optimizaciju predvidljivosti i kontrolu rizika; angažuje grupe ljudi koji zajedno imaju sve veštine i stručnost da obave posao i dele ili steknu takve veštine po potrebi.

Scrum kombinuje četiri formalna događaja za inspekciju i adaptaciju u okviru događaja koji sadrži Sprint. Ovi događaji funkcionišu jer primenjuju empirijske Scrum-

ove stubove transparentnosti, inspekcije i prilagođavanja [5].

Transparentnost

Proces i rad koji nastaju moraju biti vidljivi onima koji izvode posao, kao i onima koji ga primaju. Sa Scrum-om, važne odluke se zasnivaju na percepciji stanja njegova tri formalna artefakta. Artefakti koji imaju malu transparentnost mogu dovesti do odluka koje umanjuju vrednost i povećavaju rizik. Transparentnost omogućava inspekciju. Inspekcija bez transparentnosti obmanjuje i rasipa [5].

Inspekcija

Scrum artefakti i napredak ka dogovorenim ciljevima moraju se često i marljivo pregledavati kako bi se otkrile potencijalno neželjene razlike ili problemi. Da bi pomogao u inspekciji, Scrum pruža ritam u vidu svojih pet događaja.

Inspekcija omogućava prilagođavanje. Inspekcija bez adaptacije smatra se besmislenom. Scrum događaji su dizajnirani da izazovu promene.

Adaptacija

Ako bilo koji aspekti procesa odstupaju od prihvatljivih granica ili ako je dobijeni proizvod neprihvatljiv, postupak koji se primenjuje ili materijali koji se proizvode moraju se prilagoditi. Prilagođavanje se mora izvršiti što je pre moguće kako bi se smanjilo dalje odstupanje.

Prilagođavanje postaje teže kada ljudi koji su uključeni nisu osnaženi ili samoupravni. Očekuje se da će Scrum tim prilagoditi trenutak kada sazna nešto novo kroz inspekciju.

Scrum vrednosti

Uspešna upotreba Scrum-a zavisi od toga da ljudi postanu veštiji u životu pet vrednosti:

posvećenost, fokus, otvorenost, poštovanje i hrabrost

Scrum tim se obavezuje da će postići svoje ciljeve i pružiti podršku jedni drugima. Njihov primarni fokus je rad na Sprintu kako bi postigli najbolji mogući napredak ka tim ciljevima. Scrum tim i njegove zainteresovane strane otvoreni su prema radu i izazovima. Članovi Scrum tima poštuju jedni druge kako bi bili sposobni, nezavisni ljudi i kao takvi ih poštuju ljudi sa kojima rade. Članovi Scrum tima imaju hrabrosti da urade pravu stvar i rade na teškim problemima.

3.1. Sprint

Sprintovi su otkucaji srca Scruma, gde se ideje pretvaraju u vrednost.

To su događaji fiksne dužine od jednog meseca ili manje radi postizanja doslednosti. Novi Sprint započinje odmah nakon zaključenja prethodnog Sprintsa.

Sav posao neophodan za postizanje cilja proizvoda, uključujući planiranje sprintsa, dnevne skrome, pregled sprintsa i retrospektivu sprintsa, odvija se u okviru Sprintsa.

Tokom Sprintsa:

- Ne prave se promene koje bi ugrozile cilj Sprintsa;

- Kvalitet se ne smanjuje;
- Zaostatak proizvoda se doručuje po potrebi; i,
- Obim se može razjasniti i ponovo pregovarati sa vlasnikom proizvoda [5].

Sprintovi omogućavaju predvidljivost osiguravajući inspekciju i prilagođavanje napretka ka cilju proizvoda najmanje svakog kalendarskog meseca. Kada je horizont Sprinta predugačak, cilj Sprinta može postati nevaljan, složenost može porasti i rizik. Mogu se koristiti kraći sprintovi da bi se stvorilo više ciklusa učenja i ograničio rizik troškova i napora na kraći vremenski okvir. Svaki Sprint može se smatrati kratkim projektom.

Postoje razne prakse za predviđanje napretka, poput *burn-downs*, *burn-ups*, ili kumulativnih tokova. Iako su se pokazale korisnim, one ne zamenjuju važnost empirizma. U složenim okruženjima nije poznato šta će se dogoditi. Samo ono što se već dogodilo može se koristiti za donošenje odluka koje gledaju u budućnost.

Sprint može biti otkazan ako Sprint cilj zastareva. Samo Vlasnik proizvoda ima ovlašćenje da otkáže Sprint [5].

Sprint cilj, proizvodi zaostalih proizvoda odabrani za Sprint, kao i plan za njihovo isporučivanje, zajedno se nazivaju *Sprint Backlog*. Planiranje sprinta je vremenski ograničeno na najviše osam sati za jednomesečni sprint. Za kraće Sprintove, događaj je obično kraći.

4. KANBAN

Još jedan uobičajeni okvir za upravljanje projektima je Kanban. Dosadašnja istraživanja iz oblasti agilnih metodologija govore u prilog tome da 43% kompanija koriste Kanban kao jedan od okvira za upravljanje projektima. Poreklo iz vizuelnog sistema karata koji se koristi u Toyotinoj proizvodnji kao metoda kontrole proizvodnje, Kanban je jednostavan, ali moćan pristup razvoju softverskih proizvoda. Prevedeno kao vizuelni signal sa japanskog, Kanban se fokusira na vizualizaciju toka posla i daje prioritet radu u toku (engl. *work in progress - WIP*), ograničavajući njegov obim kako bi ga efikasno prilagodio kapacitetima tima. Čim se zadatak završi, tim može uzeti sledeću stavku. Dakle, razvojni proces nudi veću fleksibilnost u planiranju, brži preokret, jasne ciljeve i transparentnost.

5. ZAKLJUČAK

U tradicionalnom upravljanju projektima, metrike se pretežno koriste da pokažu koliko pažljivo projekat prati troškove i raspored. Međutim, to su uglavnom procene odvojene od stvarnosti i ono što nedostaje je merenje efikasnosti. Zbog toga je u agilnom pristupu fokus na rezultatima, optimizaciji performansi i donošenju odluka na osnovu podataka. Na primer, kada se optimizuje radni proces i meri učinak tima, agilni pristup projektnom menadžmentu pruža metrike kao što su vreme izvođenja, vreme ciklusa, starenje u toku, protok itd.

Scrum je dominantna agilna metodologija koja diktira tok procesa. Koristi ga veliki broj kompanija, a mali procenat kompanija ga kombinuje sa drugim tehnikama. Scrum ima za cilj da održi snažnu saradnju između ljudi koji rade na složenim proizvodima, a detalji se menjaju ili dodaju. Zasnovan je na sistematskim interakcijama između tri glavne uloge sa definisanom odgovornošću: Scrum Master, vlasnik proizvoda i tim.

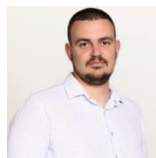
Pored Scrum metodologije koja je naprimerjivanija, primenom *Kanban* metodologije timovi mogu da izdaju mala izdanja i prilagođavaju se promenljivim prioritetima. Za razliku od Scruma, ne postoje Sprintovi sa njihovim unapred definisanim ciljevima. Kanban je usredsređen na obavljanje sitnih delova kako se pojave. Na primer, ako tester pronađe greške u proizvodu, programeri pokušavaju da ih odmah isprave.

Kanban, na primer, dobro funkcioniše nakon glavnog izdanja proizvoda i odgovara ažuriranju i održavanju. U praksi se neretko kombinuje Scrum sa Kanban tehnikama, koristeći takozvani Scrumban, umesto originalnih okvira.

6. LITERATURA

- [1] Boehm, B. (2002). Get ready for agile methods, withcare. *Computer*, 35(1), 64-69.
- [2] Moniruzzaman, A.B.M. & Hossain, S.A.(2013). Comparative Study on Agile Software Development Methodologies, *Global Journal of Computer Science and Technology Volume XIII Issue VII Version I*
- [3] Cho, J. (2008). Issues and Challenges of Agile Software Development with Scrum, *Issues in Information Systems*, 9(2), 188-195
- [4] <https://www.altexsoft.com/whitepapers/agile-project-management-best-practices-and-methodologies/>; pristupljeno: 22.07.2021.
- [5] Schwaber, K.&Sutherland, J.(2020). The Scrum Guide - The Definitive Guide to Scrum: The Rules of the Game

Kratka biografija:



Miljan Đumić rođen je u Novom Sadu 1996. god. Diplomski rad na Fakultetu tehničkih nauka iz oblasti Inženjerskog menadžmenta – Motivacija i demotivacija zaposlenih u zdravstvenj organizaciji odbranio je 2020.god.
kontakt: miljandjmic1996@gmail.com