

**IMPLEMENTACIJA AGILNIH METODOLOGIJA UPRAVLJANJA SOFTVERSKIM PROJEKTIMA - UPOREDNA ANALIZA PRIMENE U TEORIJI I PRAKSI****IMPLEMENTATION OF AGILE METHODOLOGIES IN SOFTWARE PROJECT MANAGEMENT: COMPARATIVE ANALYSIS OF THEORY AND PRACTICE**Dina Petrov, *Fakultet tehničkih nauka, Novi Sad***Oblast – INŽENJERSKI MENADŽMENT**

**Kratak sadržaj** – *Nastanak agilnih metodologija i radnih okvira poput Skrama korenito su promenili pristup upravljanju softverskim projektima. Mada oni predstavljaju moćne alate stvorene s namerom prevazilaženja izazova tradicionalnih pristupa, postavlja se pitanje o njihovoj realističnoj praktičnoj primeni u industriji, i da li su IT kompanije dosledne pravilima i principima ovih metodologija.*

**Ključne reči:** *agilne metodologije, Scrum, informacione tehnologije, upravljanje projektima, kompanije*

**Abstract** – *The emergence of agile methodologies and frameworks like Scrum has fundamentally changed the approach to managing software projects. While these methodologies represent powerful tools designed to overcome the challenges of traditional approaches, the question arises about their realistic practical application in the industry and whether IT companies consistently adhere to the rules and principles of these methodologies.*

**Keywords:** *agile methodologies, Scrum, project management, information technologies, companies*

**1. UVOD**

S naglim razvojem informacionih tehnologija, a samim tim i sve većom kompleksnošću softverskih projekata pojavila se izražena potreba za pouzdanijim, savremenijim i fleksibilnijim metodologijama upravljanja takvim projektima. Tradicionalni metod razvoja softvera, karakterističan za ovaj period i poznat kao metod vodopada (engl. Waterfall), sve manje je mogao da zadovolji potrebe eksponencijalno rastuće složenosti savremenih softverskih poduhvata.

Otuda na scenu stupaju agilne metodologije, s ciljem prevazilaženja ograničenja tradicionalnih metodologija i korenite promene pristupa razvoju i pogled na životni ciklus softverskih proizvoda. Agilne metodologije su bazirane na više različitih radnih okvira aktuelnih krajem 20. veka, među kojima je najrasprostranjeniji Skram (engl. *Scrum*) i dan-danas [1]. S razvojem agilnog pristupa, iz uloge projektnog menadžera razvijaju se potpuno nove, srodne uloge, kao npr. vlasnik proizvoda (engl. *Product Owner*), skram master (engl. *Scrum Master*), i sl.

**NAPOMENA:**

**Ovaj rad proistekao je iz master rada čiji je mentor bila doc. dr Danijela Ćirić – Lalić.**

Definišu se i potpuno nove "ceremonije" i alati upravljanja razvojem, kao i principi i vrednosti koje služe kao vodilje i motivatori pri upravljanju projektima. Međutim, današnje kompanije koje implementiraju agilni pristup neretko vrše određene modifikacije ovih "pravila", kako bi pronašle sopstveni, njima najpogodniji pristup. Ponekad su izmene samo u nazivima uloga i elemenata nekog konkretnog radnog okvira, a ponekad je originalni okvir gotovo neprepoznatljiv, ili je pak hibrid više različitih radnih okvira. Cilj rada jeste sagledati primenu agilnih metodologija, onako kako su zamišljene i kako su ih njihovi tvorcii definisali, u poređenju sa njihovom realističnom praktičnom primenom u industriji - u IT kompanijama Srbije i regiona. U okviru rada, pored teorijskog pregleda istorije agilnih metodologija i njihovih principa, izvršena je uporedna analiza agilnosti dve produktno orijentisane kompanije, a potom izvršeno anketno istraživanje o stavovima i realnoj implementaciji agilnog pristupa upravljanja softverskim projektima u IT kompanijama Srbije i regiona.

**2. AGILNE METODOLOGIJE**

Originalni, široko rasprostranjen metod upravljanja softverskim projektima bio je tradicionalni metod, takođe nazvan i metod Vodopada, koji je definisao Winston Rojs 1970. godine [2]. Rojsov metod vodopada koristi pristup raščlanjivanja kompleksnih softverskih poduhvata u faze koje se sekvencijalno izvršavaju jedna nakon druge. S usložnjavanjem softverskih poduhvata pokazalo se da metod vodopada nije optimalan – ispostavio se kao previše rigidan i nefleksibilan. Najveći problem koji se javlja prateći vodopad metod pri razvoju softvera jeste faza testiranja koja se izvršava na samom kraju projekta, odnosno nakon što je celokupno softversko rešenje isprogramirano. Dodatno, neretko dolazi do izmena u zahtevima u kasnijim fazama, razvoja, što metod vodopada nije u stanju da podrži.

Do početka dvadeset prvog veka, postojao je čitav set raznolikih, „agilnijih“ metodologija, među kojima su neke od najpoznatijih Skram, Kanban, Ekstremno programiranje, Lin (engl. *Lean*), Kristal i Razvoj vođen funkcionalnostima [4]. Elementi ovih metodologija služili su kao inspiracija za postavku jednog, zajedničkog pristupa i definisanje novog seta vrednosti i principa agilnih metodologija. Februara 2021. godine definisan je novi, agilni pristup, sa svojim setom vrednosti i principima, navedenim u dokumentu pod nazivom Agilni manifest

[5]. Postoje četiri ključne vrednosti koje naglašavaju važnost ljudi i interakcija ispred procesa i alata, fu funkcionalnog softvera ispred opširne dokumentacije, saradnje sa klijentom ispred pregovaranja o ugovorima i reagovanja na promene ispred praćenja unapred definisanog plana. Manifest takođe sadrži dvanaest principa koji promovišu fleksibilnost, kontinuiranu isporuku, saradnju, održivi razvoj, tehničku izvrsnost i jednostavnost, uz naglasak na često isporučivanje vrednosti i brzo prilagođavanje promenama.

### 3. SKRAM RADNI OKVIR

Skram (engl. *Scrum*) je radni okvir agilnog upravljanja softverskim projektima, nastao devedesetih godina prošlog veka. Tvorci Skrama, Džef Saterlend i Ken Šveber su kasnije u 2010. godini objavili zvanični Skram vodič (engl. *The Scrum Guide*) u kojem su predstavljene Skram uloge, ceremonije i artefakti, kao i uputstva za najbolje prakse primene radnog okvira. Skram je dizajniran da omogući timovima da rade efikasnije i prilagodljivije, koristeći iterativan pristup koji se fokusira na kontinuiranu isporuku vrednosti kroz kratke razvojne cikluse zvane sprintovi. Ovaj okvir naglašava timsku saradnju, fleksibilnost, i transparentnost, omogućavajući brzu adaptaciju na promene i kontinuirano poboljšanje proizvoda. Sprint u proseku traje nekoliko nedelja, i cilj sprinta je razviti novo poboljšanje funkcionalnog proizvoda – inkrement ili MVP (engl. *Minimum Viable Product*) do kraja iteracije, kada se vrši demonstracija inkrementa i dobija se povratna informacija od zainteresovanih strana/klijenata.

Dakle, poboljšanja proizvoda su redovna i česta, a prostor za povratne informacije od zainteresovanih strana i rano otkrivanje grešaka je maksimalizovan. Postepene isporuke gotovih MVP proizvoda osiguravaju da je uvek dostupna potencijalno upotrebljiva verzija radnog proizvoda krajnjim korisnicima. Skram tim je samoorganizovan i ukršteno-funkcionalan (engl. *cross-functional*) tim, što znači da su sposobni da sami procenjuju i određuju na koji način je najbolje pristupiti određenom zadatku ili poslu, umesto da njima rukovodi neko sa strane. Članovi tima poseduju sva potrebna znanja, umeća i kompetencije da nezavisno ostvaruju ciljeve posla pred sobom. Timski model u Skramu je dizajniran tako da optimizuje fleksibilnost, kreativnost i produktivnost njegovih članova. Skram radni okvir definiše nekoliko tipova uloga koje članovi Skram tima mogu nositi, među kojima su vlasnik proizvoda, razvojni tim i Skram master (engl. *Scrum Master*). Skram tim je samoorganizovan i ukršteno-funkcionalan (engl. *cross-functional*), što znači da su njegovi članovi sposobni da sami procenjuju i određuju na koji način je najbolje pristupiti određenom zadatku ili poslu, umesto da njima rukovodi neko sa strane.

Skram radni okvir definiše nekoliko tipova uloga koje članovi Skram tima mogu nositi, među kojima su vlasnik proizvoda, razvojni tim i Skram master (engl. *Scrum master*). Razvojni tim sam, ili sa vlasnikom proizvoda i Skram masterom, bira koje stavke iz bekloga preuzima za tekući sprint i poseduje sve potrebne veštine i znanja za njihov uspešan razvoj. Vlasnik proizvoda predstavlja sponu između tehničkog (razvojnog) tima i zainteresovanih strana, odnosno klijenata i njegov zadatak da maksima-

lizira vrednost proizvoda. Stvaranje i upravljanje beklogom proizvoda takođe spada u odgovornosti vlasnika proizvoda. To podrazumeva identifikaciju i definisanje stavki bekloga u obliku kartica, ali i njihovo sortiranje u skladu sa prioritetima, poslovnim ciljevima i potrebama zainteresovanih strana, ali i riziku, nužnosti i vrednosti koje donose. Skram master predstavlja „kičmu“ skrama i odgovoran je za promociju, podršku i sprovođenje svih procesa skram radnog okvira unutar kompanije/organizacije. Njegov zadatak je da pomaže članovima tima da nauče i razumeju teoriju, principe, pravila i vrednosti skrama, kao i da se oni pravilno primenjuju. Odgovoran je za produktivnost i postizanje ciljeva sastanaka, kao i da ne traju duže nego što je predviđeno (engl. *timeboxing*). Skram master dodatno pomaže razvojnom timu da bude produktivan i samostalan, jer ume da identifikuje i ukloni prepreke koje mogu ometati rad tima. To mogu biti tehnički problemi, nedostatak resursa ili čak spoljašnji pritisci klijenata.

Kada su u pitanju zadaci bekloga, u Skramu se definišu u obliku korisničkih priča (engl. *user story*) koje se pišu u sledećem formatu: Ja kao <ULOGA/ROLA>, želim da <AKTIVNOST>, kako bih dobio <BENEFIT>.

Korišćenje ovakvog formata za opisivanje funkcionalnosti koju softver treba da podrži ih čini jasnijim razvojnim timu jer stavlja fokus na krajnjeg korisnika i način na koji on očekuje da koristi posmatranu funkcionalnost. Svako korisničkoj priči, pored opisa u gorepomenutom formatu, takođe treba pridodati i tzv. poene priče (engl. *story points*) kao rezultat estimacije zahtevnosti priča. Svaka korisnička priča nosi određenu težinu implementacije koju razvojni tim zajedno procenjuje. Neophodno je da se estimacija radi timski, kako bi se poeni što realnije odredili i sprečile situacije u kojima jedna osoba ne uspe da predvidi sve okolnosti i problematične aspekte. Svi zadaci u toku razvoja aplikacija se organizuju u obliku kartica na Skram tabli, koja ima nekoliko različitih vrsta kolona: Beklog proizvoda, beklog sprinta, U toku, Revizija i QA, Sprint gotovo, Beklog gotovo.

Jedna kartica u toku izvršavanja zadatka prolazi kroz sve navedene kolone redom, a nekad se može i vratiti u prethodnu kolonu, kao npr. kada QA inženjeri otkriju nepravilnost u radu funkcionalnosti, kartica se vraća u “U toku” kolonu, kako bi se greška ispravila i ponovo proverila. Skram radni okvir definiše nekoliko različitih vrsta sastanaka (tzv. Skram ceremonije) karakterističnih za razvoj u sprintovima: planiranje sprinta, dnevni sastanak, revizija sprinta i retrospektiva sprinta. Sve ceremonije su striktno vremenski ograničene na predefinisano trajanje. Celokupan posao koji se planira u toku sprinta se određuje na samom početku sprinta, na događaju planiranja sprinta (engl. *sprint planning*), kada članovi razvojnog tima preuzimaju korisničke priče i na Skram tabli sebe dodaju na kartice, a zatim ih prebacuju u Sprint beklog kolonu. Dnevni sastanak (engl. *daily standup*), kao što ime kaže, je sastanak koji se održava svaki radni dan u toku sprinta. Takođe je vremenski ograničen, i to na maksimalno 15 minuta.

Na ovom sastanku, svaki član razvojnog tima ima priliku da kaže na čemu je radio prethodni dan, na čemu radi u tom momentu, i na čemu planira da radi nakon završenog trenutnog zadatka. Sastanak služi za praćenje napretka rada sprinta od strane celog razvojnog tima, kako bi svi

bili u toku sa radom jedni drugih i time stekli uvid u zajedničko funkcionisanje tima kao celine. Na reviziji sprinta (engl. *sprint review*), razvojni tim prezentuje urađeni posao demonstracijom inkrementa aktuelnog sprinta radi dobijanja povratnih informacija klijenata. Raspravlja se o tome šta je tokom sprinta dobro urađeno, koji su problemi nastajali i kako su ti problemi rešeni, kao i opciono šta je potrebno ispraviti u narednom sprintu

Dodatno se razmatra prioritet narednog sprinta kako bi na planiranju sprinta odabrala adekvatne korisničke priče. Konačno, sprint retrospektiva (engl. *sprint retrospective*) je prilika za Skram tim da izvrši sopstveni pregled ishoda sprinta (bez klijenata) i napravi plan za poboljšanja koja će biti primenjena u sledećem sprintu.

Održava se nakon revizije prethodnog sprinta, a pre planiranja narednog. Svrha ovog sastanka jeste oceniti kako je prethodni sprint prošao u odnosu na članove tima, odnose, procese i alate [6] [7].

#### **4. UPOREDNA ANALIZA IMPLEMENTACIJE AGILNIH METODOLOGIJA U IT KOMPANIJAMA**

Prva celina empirijskog dela rada obuhvata analizu implementacije agilnog pristupa upravljanju projektima u dve produktno orijentisane kompanije, i bazira se na ličnom iskustvu i opažanjima, kao i intervjuima sa zaposlenim licima tokom perioda obavljanja stručne prakse (u slučaju kompanije A) i zaposlenja autora rada (u slučaju kompanije B). Uočeno je da su kompanije A i B prošle kroz slične faze poslovanja od svog osnivanja pa sve do danas. Obe su inicijalno radile po modelu eksternog angažovanja, razvijajući proizvode za strane naručioca sve do prelaska na proizvodnu orijentisanost razvijanjem sopstvenog proizvoda. Nakon višegodišnjeg rada po "čistom" Skramu razvijajući tuđe proizvode, prelazak na razvijanje sopstvenih proizvoda ovim kompanijama je omogućio slobodu isprobavanja i modifikovanja različitih agilnih elemenata i drugačijih pristupa razvoju, pre konačnog konstruisanja najpogodnijeg pristupa za svoje potrebe.

Ono što se može uočiti kod kompanija produktne orijentisanosti jeste da je daleko lakše definisati sopstveni, hibridni agilni pristup razvoju kada razvijani proizvod pripada kompaniji i timu koji ga razvija. Mogućnost slobode donošenja odluka i uvažavanja mišljenja svih članova tima je mnogo veća kada ne postoji eksterni naručilac čiji se proizvod razvija i čija je "reč poslednja". I uloga vlasnika proizvoda ima potpuni smisao u slučaju proizvodno orijentisanih kompanija, jer su u takvom režimu rada vlasnici proizvoda slobodni da donose odluke o tome šta će se razvijati, u kojem redosledu i na koji način, bez potrebe da primarno ispunjavaju sve zahteve eksternog naručioca.

Nakon prelaska sa "čistom" Skramom na sopstveni pristup, smanjena je i strogost u definisanju i praćenju estimacija - u slučaju kompanije A u potpunosti, dok su u slučaju kompanije B estimacije i dalje prisutne pri razvoju epova, ali sa daleko manjim subjektivnim osećajem pritiska striktnog pridržavanja istim od strane zaposlenih.

Ni jedna ni druga kompanija više ne razvija u sprintovima već se zadaci obavljaju sekvencijalno kao u Kanbanu. Zadaci se prate vizuelno na tabli koja liči na hibrid Skram i Kanban table. U obe kompanije se neguje poverenje prema zaposlenima da odgovorno izvršavaju svoje

zadatke prema svojim mogućnostima i znanju, nasuprot rigidnom praćenju estimacija u svakom sprintu poenima priča i rokovima. U slučaju obe kompanije, nova verzija softvera (inkrement) dostupna je krajnjim korisnicima u proseku jednom nedeljno. Čestom dostavom novog inkrementa smanjuje se koštanje nastalih bagova, jer se oni brzo rešavaju i na ispravljenu verziju se ne čeka dugo nakon prijave problema. Iako obe kompanije negiraju korišćenje Skrama kao primarnog radnog okvira, mogu se primetiti primenjeni elementi Skrama i drugih metodologija s određenim modifikacijama.

Na primer, u kompaniji B uloga vlasnika proizvoda nosi naziv menadžer proizvoda, iako snosi gotovo identične odgovornosti kao vlasnik proizvoda u kompaniji A i u Skrama generalno. U slučaju obe kompanije, zadaci se pišu u formatu klasičnih korisničkih priča Skrama, mada se retko interno nazivaju korisničkim pričama. Ni jedna ni druga kompanija nisu imale potrebu za ulogom Skram Mastera, što je i očekivano uzimajući u obzir da su obe odbacile Skram kao striktan radni okvir.

Obe kompanije imaju ceremonije koje imaju ciljeve i zadatke kao Skram ceremonije, kao što su dnevni sastanci i retrospektive sprinta (mada ne postoji sprint već se gleda prethodni period određene dužine). Obe kompanije vremenom su došle do svoje lične metodologije razvoja softvera, koja se pokazala kao najpogodnija za njihove potrebe. Uočavaju se elementi Skrama i Kanbana, mada nijednu metodologiju ove kompanije ne navode kao svoj primarni pristup.

#### **5. ANKETNO ISTRAŽIVANJE PRIMENE AGILNIH METODOLOGIJA PRI UPRAVLJANJU RAZVOJEM SOFTVERA U IT KOMPANIJAMA SRBIJE I REGIONA**

Druga celina empirijskog dela rada obuhvata sprovedeno anketno istraživanje primene agilnih metodologija pri upravljanju razvojem softvera u IT kompanijama Srbije i regiona, sprovedeno leta 2024. godine putem interneta. Konačni uzorak ankete broji N = 80 ispitanika. U uzorku od 80 ispitanika, na osnovu rezultata pitanja o godini rođenja ispitanika, najveća većina pripada uzrastu od 25 do 34 godine — 56,3% ispitanika. Druga po redu je grupa u rasponu od 35 do 44 godine, koja čini 25% ispitanika. Na osnovu pitanja o polu ispitanika, čak 72,9% ispitanika je muškog pola. S obzirom na činjenicu da industrijom informacionih tehnologija i dalje dominiraju muškarci, ova informacija ne iznenađuje.

Kada je u pitanju priroda kompanije, gotovo je jednak broj ispitanika koji su naveli da rade u kompaniji eksternog angažovanja (47,5%) i onih koji rade u kompaniji proizvodne orijentisanosti (45%). Postoji i određeni procenat ispitanika čije kompanije rade u oba režima istovremeno (5%). Na pitanje "da li vaša kompanija koristi agilne metodologije", 94,9% ispitanika odgovorilo je potvrdno. Naredno pitanje odnosilo se na vrstu agilnih metodologija. Na prvom mestu se nalazi Skram, sa 43,8% ispitanika. Zanimljivo je da je Skram i na drugom mestu, sa 23,8%, ali "sa određenim modifikacijama". Dakle, Skram je prisutan kod 67,4% ispitanika, u nekom obliku. Na trećem mestu se nalazi Kanban sa 11% uzorka. Što se tiče tipova ceremonija karakterističnih za kompanije, uočeno je da čak i među onima koji su odabrali neku ceremoniju koja je

karakteristična za Skram radni okvir, nije prisutan isti procenat drugih Skram ceremonija. Npr, planiranje sprinta je prisutno kod 68.8% ispitanika, dok je revizija sprinta pristupa kod 46.3%. Ovo nam može govoriti da su u pitanju pomenute Skram modifikacije, ukoliko su timovi uvideli da ne dobijaju dovoljno dodatne vrednosti od svih Skram ceremonija. Kada je u pitanju uloga koja definiše zadatke bekloga, najveći broj ispitanika odabrao je vlasnika proizvoda/menadžera proizvoda (58,8%). Međutim, samo je nešto manji procenat odabranog odgovora Lead developer/Tech lead sa 57,5% i developeri sa 55%. Ova podela ne iznenađuje, jer čak i u Skramu je "dozvoljeno" da ostali članovi tima pomažu vlasniku proizvoda pri definisanju zadataka. Takođe, tehnička lica često jedina imaju uvid u nedostatke "ispod haube".

Kada je u pitanju uloga koja definiše prioritete stavki/zadatka u beklogu, ubedljivo najveći broj ispitanika odabrao je vlasnika proizvoda/menadžera proizvoda (61,3%). Na drugom mestu se nalazi *Lead developer/Tech lead* sa 46,3%. Na pitanje o vršenju procene težine i obima implementacije zadataka, 27,5% ispitanika odgovorilo je da se procena uopšte ne vrši. Procenu putem poena priča vrši 43,8% ispitanika. Slično, na pitanje o postojanju rokova za završetak implementacije zadataka, 40,7% ispitanika izjasnilo se za kraj iteracije. Na drugom mestu je odgovor da rokova nema, sa 25,9% ispitanika koji su odabrali ovu opciju. Kada je u pitanju prisutnost koncepta sprinta/iteracije, najveći broj ispitanika (74,5%) dalo je potvrđan odgovor, dok u 23,5% slučajeva sprintovi ne postoje.

Kada je u pitanju slaganje sa izjavom "Zadaci jednog sprinta se ne prenose u naredni sprint," najveći broj ispitanika (43,3%) dao je najnižu ocenu 1, na skali od 1 do 5. Ovo govori da neki teoretski principi i pravila u praksi nisu realni i izvodljivi, jer nije moguće predvideti sve moguće probleme i okolnosti. Slično, kada je u pitanju slaganje sa izjavom "U toku sprinta ne dolazi do promena u planiranom obimu posla za taj sprint," najveći broj ispitanika (36,1%) dao je ocenu 2.

Kada je u pitanju objavljivanje novog funkcionalnog proizvoda na kraju sprinta, najveći broj ispitanika (19%) dao je ocenu 3, dok je ocenu 4 dalo nešto manje - 27,9%. Vidimo da princip obavezne dostave funkcionalnog inkrementa nakon svakog sprinta nije uvek izvodljiv i moguć. Svoga 13,1% ispitanika dalo je najvišu ocenu slaganja sa ovim stavom.

Kada su u pitanju čini stavovi ispitanika o prednostima agilnosti, najčešći odgovori ticali su se fleksibilnosti, lakoci planiranja, transparentnosti i komunikaciji, brzog isporuci. Što se tiče negativnih stavova, na prvom mestu je preveliki fokus na različite sastanke i osećaj "gubljenja vremena" na iste, zatim manjak dugoročnog planiranja i ciljeva, i nedevojlno temeljan razvoj radi postizanja obima sprinta.

## 6. ZAKLJUČAK

Metodologije upravljanja softverskim projektima su prošle kroz značajne faze korenitih promena kroz istoriju, od Rojsovog modela Vodopada iz 1970. godine, preko razvoja "laganijih" radnih okvira devedesetih godina prošlog veka, do sastanka februara 2001. godine kada je nastao Agilni manifest.

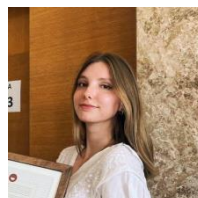
I dan-danas, Skram je jedan od najzastupljenijih agilnih radnih okvira, što se može videti i iz rezultata sprovedenog anketnog istraživanja. Rad je za cilj imao poređenje agilnih pristupa u teoriji, onako kako su ih njihovi tvorci zamislili, odnosno onako kako njihova "pravila" nalažu, i pri realnoj, praktičnoj primeni u industriji. Posmatranjem rezultata anketnog istraživanja, takođe možemo uvideti određene devijacije od "pravila" u kompanijama širom Srbije. Smatram da upravo u tome leži poenta agilnosti. Skram i slični srodni radni okviri i agilne metodologije nalažu čitav niz korisnih pravila, ceremonija i uloga, ali forsiranje timova da se striktno pridržavaju svakom "zakonu" dovodi do gubitka karakteristike prilagodljivosti i agilnosti koje se promovisu, a koje predstavljaju srž i temelj savremenih pristupa upravljanju razvojem softvera. Striktno pridržavanje pravilima i principima jednog savremenog agilnog radnog okvira može štetiti produktivnosti rada unutar timova ako ne postoji mogućnost primene određenih argumentovanih modifikacija metodologije.

Ako zagovornici agilnih metodologija posebno naglašavaju njihovu neograničenu prilagodljivost izvesnim promenama u zahtevima, smatram da je prihvatljivo, ili čak i poželjno, da se ta ista prilagodljivost primeni i na samu metodologiju. Agilne metodologije i radni okviri mogu biti izuzetno moćan alat upravljanja razvojem softverskih projekata, ali ne treba pratiti svako pravilo slepo, bez kritičke analize posmatranih pravila i principa u kontekstu svakog projekta ponaosob, jer se onda ponovo javljaju problemi strogosti tradicionalnih principa, koje bi trebalo da su agilne metodologije prevazišle.

## 7. LITERATURA

- [1] Digital.ai. (2023). *State of Agile Report*. Preuzeto sa Digital.ai
- [2] Royce, W. W. (1970). *Managing the development of large software systems: Concepts and techniques*. In *Proceedings of IEEE WESCON* (pp. 1-9). IEEE.
- [3] Larman, C., & Basili, V. R. (2003). *Iterative and Incremental Development: A Brief History* (pp 47-56). *IEEE Computer*, 36(6)
- [4] Cockburn, A. (2006). *Agile software development: The cooperative game* (2nd ed.) (pp 45-47). Addison-Wesley Professional.
- [5] Highsmith, J. (2001, February 11-13). *History: The Agile Manifesto*. *Agile Manifesto*. Preuzeto 2.9.2024. sa <https://agilemanifesto.org/history.html>
- [6] Schwaber, K., & Sutherland, J. (2020). *The Scrum guide*. Scrum.org. Preuzeto 02.09.2024. sa <https://scrumguides.org/scrum-guide.html>
- [7] Schwaber, K. (2004). *Agile project management with Scrum (Developer best practices)* (pp. 2-5, 53-60, 133-138). Microsoft Press

### Kratka biografija:



**Dina Petrov** rođena je u Zrenjani- nu 1999. godine. Master rad na Fakultetu tehničkih nauka iz oblasti Inženjerski menadžment – Pro- jektni menadžment odbranila je je 2024. godine.