

AGILNO PLANIRANJE I PROCJENJIVANJE

Milenko Savić

Goran Marjanović

Sažetak: U današnje vrijeme, kako velike tako i male, firme traže načine za kvalitetniji odgovor zahtjevima tržišta što ih tjera na uvođenje metodologija u vođenju projekata. Prilikom uvođenja metodologija, firme bi, u stvari, htjele biti konkurentnije na tržištu i profitabilnije, a to se popularno danas naziva agilne firme.

Ako želimo da projekat ima imalo šansi za uspjeh, jedan od najbitnijih činilaca za kojim moramo posegnuti je dobro razraditi metodologiju. Projektnih metodologija ima raznih, ugrubo ih možemo podijeliti na dvije grupe: tradicionalna metodologija i agilna metodologija.

Ključne riječi: preduzeće, metodologija, uspjeh, softver

AGILE BUSINESS AND ASSESSMENT

Abstract: Nowadays, as well as big ones, small enterprises are looking for ways to respond the requirements of the market in the best way. In order to achieve that, they are forced to introduce some new methodologies in project management. Enterprises do that with the aim to be more profitable and as concurrent at the market as possible. This kind of business is known as agile enterprise.

Supposing we want the project to have chance for success one of the most important elements we have to include is to develop methodology. There are numerous project methodologies. They can be divided into two groups: traditional methodology and agile methodology.

Key words: enterprise, methodology, success, software

1. UVOD

Tradicionalne metodologije se još nazivaju i vodopad ili slap (engl. *waterfall*), a naziv su dobile po strukturi svoje izvedbe jer se odvijaju po fazama u kojima jedna faza slijedi drugu fazu te u takvoj metodologiji nema iteracija.

Agilne metodologije (npr. *Scrum* i *Kanban*) su strukturisane na potpuno drugačiji način te govore o tome da su promjene tokom projekta čak poželjne i konstantne.

Radi lakšeg shvatanja, tradicionalne metodologije možemo opisati kao *strukturisane*, a agilne kao *adaptivne*.

Budući da se u posljednje vrijeme cijela softverska industrija oslanja i teži ka agilnom vođenju projekata, u ovom radu ćemo se osvrnuti upravo na takav pristup vođenju projekata te pokazati da i u takvom adaptivnom okruženju postoji cijeli niz prednosti, ali i prepreka radi kojih se neki projekti vraćaju tradicionalnim metodologijama.

2. DEFINICIJA PODRUČJA UPRAVLJANJA PROJEKTIMA

Budući da je proces standardizacije u raznim područjima drugačije definisan i u stalnom je razvoju, pojam upravljanja projektima u softverskoj industriji ćemo dati proizvoljan opis kako

bi se najbolje opisalo u duhu razumljivog jezika. PMI (*Project Management Institute*) koji definiše projekat kao „vremenski određeno nastojanje da se proizvede jedinstven proizvod, usluga ili rezultat“.¹³⁵

Iz ove definicije možemo izvući dva vrlo bitna zaključka kojim projekt razlikujemo od svakodnevnog operativnog posla.

Prvi je da je sam projekt privremena aktivnost koja ima svoj početak i kraj. Takođe unutar ovog osvrta zaključujemo da projektni tim koji je okupljen za sam razvoj projekta se nakon završetka projekta raspušta. Time možemo reći da se ljudski potencijali i resursi preraspoređuju na druge projekte i druge operativne poslove.

Drugi zaključak koji možemo izvesti je da je samim završetkom projekta stvoren jedinstveni proizvod.

Po drugim definicijama, projekat možemo definisati na druge načine.

Na primjer, Kerzner definiše projekt kao „bilo koji niz aktivnosti i zadaća koji imaju određeni cilj, koji treba ispuniti određene specifikacije, imaju određen početak i kraj, ograničena finansijska sredstva, imaju određene resurse na raspolaganju za trošenje (kako ljudske tako i tehničke) te su višefunkcionalne“.¹³⁶

Ako bismo uzimali u obzir strogo definiciju projekta po PMI standardu, upravljanje projektom bismo označili kao „primjenu znanja, vještina, alata i tehnika u projektnim aktivnostima kako bi se ispunili projektni zahtjevi“.¹³⁷

Upravljanje projektom uključuje:

- utvrđivanje zahtjeva,
- postavljanje jasnih i ostvarivih ciljeva,
- uspostavu ravnoteže između suprotstavljenih zahtjeva za kvalitetu, doseg, vrijeme i trošak,
- prilagođavanje specifikacija, planova i pristupa interesima i očekivanjima zainteresovanim stranama.

3. STATISTIKA I UZROCI (NE)USPJELIH PROJEKATA

U softverskoj industriji projekti značajno probijaju planirani budžet i obim planiranih resursa te time i vremenske okvire, a na kraju se ne isporuči ono što je planirano. Iz rezultata raznih istraživanja jasno možemo vidjeti da IT projekti imaju više šanse za neuspjeh nego za uspjeh.

Uzmimo za primjer istraživanje sprovedeno još davne 1995. godine koje je napravila firma *KPMG Consulting* u kojem je sastavljen izvještaj¹³⁸ o uzrocima i posljedicama neuspjeha projekta. Po izvještaju, glavni razlozi neuspješnih IT softverskih projekata su sljedeći:

- ciljevi projekta nisu potpuno definisani,
- loše planiranje i procjena,
- tehnologija je nova u strukturi organizacije,
- metodologija projekta upravljanja nije adekvatna ili uopšte ne postoji,
- projektni tim ne sadrži dovoljno iskusnih članova.

¹³⁵ PMI, *A Guide to the Project Management Body of Knowledge, Third Edition (PMBOOK Guide)*. Newton Square, PE: Project Management Institute, 2004.

¹³⁶ H. Kerzner, *Project Management, A Systems Approach to Planning, Scheduling and Controlling, Eight Edition*, Hoboken, NJ: John Wiley & Sons, Inc, 2003.

¹³⁷ K. Feralj, *Upravljanje informacijskim sistemima*. Zagreb: Fakultet elektrotehnike i računarstva, 2004.

¹³⁸ Cole A., *Runaway Projects – Causes and Effects*, *Software World, UK, Vol 26. No. 3, 1995.*

Kako smo već naveli, vodeći se statističkim podacima, šanse za neuspjeh softverskih projekata su veće od onih za sam uspjeh projekta. U prosjeku, 80% novih projekata je isporučeno s probijanjem roka, a tu se može uzeti u obzir i finansijsko ograničenje. Iz navedenih 80% gotovo pola, tj. 40%, se može označiti kao neuspjeli projekti, a od ostale polovine se u praksi pokazalo tek 10–20% kao projekt koji je implementirao i zadovoljio sve kriterijume uspješnosti (podaci iz OASIG iz 1996. godine).

Međutim, situacija se drastično statistički popravila ako posmatramo projekte unazad nekoliko godina. Samo statistici doprinosi i činjenica da se zadnjih godina i softverska industrija značajno razvila te da se „učilo na greškama“ i uvodile nove metodologije razvoja.

Po istraživanju američke firme *The Standish Group – The CHAOS Report*, koje je sprovedeno u desetogodišnjem periodu (1994–2004) navodi se da se u tom periodu situacija znatnije popravila, premda gledajući brojke ne tako značajno.

| Godina | Uspješnost projekta | Djelomična uspješnost projekta | Neuspješnost projekta |
|--------|---------------------|--------------------------------|-----------------------|
| 1994. | 16 % | 53 % | 31 % |
| 1996. | 27 % | 33 % | 40 % |
| 1998. | 26 % | 46 % | 28 % |
| 2000. | 28 % | 49 % | 23 % |
| 2002. | 34 % | 51 % | 15 % |
| 2004. | 29 % | 53 % | 18 % |

Tabela 1. - *The Standish Group - The CHAOS Report*

Po navedenom istraživanju, kao najbitniji uzroci neuspjeha softverskih projekata navedeni su sljedeći elementi:

- nejasni zahtjevi klijenata,
- neodgovarajuća kontrola promjena u projektu,
- nekontrolisana promjena vremenskog plana projekta,
- neodgovarajuće korišćenje tehnika i alata za upravljanje projektima,
- neodgovarajuće upravljanje troškovima projekta,
- neodgovarajuća komunikacija na projektu,
- neadekvatno planiranje,
- nerealna očekivanja,
- nekompletna specifikacija,
- neodgovarajuće testiranje,
- neodgovarajuće upravljanje rizicima,
- neodgovarajuće osiguravanje kvalitete proizvoda,
- drugi ljudski faktori...

Ono što jasno možemo izvući iz navedenog su sljedeće ključne riječi:

- nejasno,
- neodgovarajuće,
- nekompletno,
- nerealno.

Upravo zbog ovakvih ključnih faktora, cijela softverska industrija se okrenula korišćenju metodologija pri upravljanju projektima. Različiti projekti zahtijevaju i različite metodologije i različite pristupe, ali na kraju bitan je činilac da metodologija mora biti primijenjena.

4. PRIMJENA METODOLOGIJE

Metodologija je, prema PMI definiciji, sistem tehnika i praksi, procedura i pravila koje koristi onaj ko radi na području određene djelatnosti. Procedura je niz koraka koje je potrebno obaviti kako bi se ostvario nekakav cilj, te je korake potrebno obaviti po redosljedju.

Metodologija se pak može shvatiti kao niz ili skup smjernica i principa koji mogu biti primijenjeni u specifičnoj situaciji. U softverskoj industriji te smjernice možemo shvatiti kao popis zadataka (engl. *tasks*) koji moraju biti obavljani. Uz navedeno, metodologija može biti specifičan pristup problemu, razni modeli te razni obrasci korišćeni kroz životni ciklus projekta.

Metodologiju u softverskom svijetu nikako ne smijemo shvatiti kao jednoznačnu definiciju koja bi označila privremeno rješenje ili kuharicu (engl. *cookbook*) za uspješno izvođenje projekta.

Dobra metodologija mora sadržavati sve bitne procese upravljanja projektom. Sva projektna područja moraju biti uključena u procese i širiće se sa obimom projekta.

Po Kerzneru „karakteristike dobre metodologije moraju imati preporučeni stupanj detalja, uporabu predložaka, standardizirane tehnike planiranja, vremensko određivanje i kontrolu troškova, standardizirani oblik izvještavanja, fleksibilnost za brzi razvoj i primjenu na svim projektima, prihvaćenost i upotrebljivost u cijeloj organizaciji, prihvatljivost korisnicima te etiku dobro obavljenog posla“.¹³⁹

5. AGILNA METODOLOGIJA

U posljednjih nekoliko godina došlo je do procvata novog stila metodologije softvera koji je nazvan agilnom metodologijom. Agilna metodologija je prije svega kulturni manifest, sistem vrijednosti, principa i stilova ponašanja primjerenih brzom vremenu u kojem živimo i radimo te u kojem smo jednostavno pretrpani informacijama, mogućnostima i prilikama. Ovakav stil metodologije se oslanja na kompleksnije definisane uloge pojedinaca u projektu te na njihovu veću odgovornost, transparentniju komunikaciju i brze povratne informacije (engl. *feedback*).

Agilni manifest¹⁴⁰ propisuje „Proglas o metodi agilnog razvoja softvera“ te se definiše na sljedeći način.

„Tražimo bolje načine razvoja softvera razvijajući softver i pomažući drugima pri njegovom razvoju.

5.1. AGILNO PLANIRANJE

Za većinu softverskog razvoja se može reći da ima prisutnu haotičnu aktivnost u sebi, te se često parafrazira sa izrazom „*kodiraj i popravi*“ (engl. „*code and fix*“). Softver je najčešće pisan bez nekakvog dobro osmišljenog plana, a i sam arhitekturno-sistemska dizajn se svodi na odluke koje su donesene „*usput*“. To su najčešće male promjene te takođe i djeluje ako je osmišljeni sistem malen i nije kompleksan. Međutim, najčešće se događa da i mali sistemi s vremenom prerastaju u veće sisteme te postaje vrlo teško na jednostavan način dodavati nove funkcionalnosti. Takođe uz dodavanje novih funkcionalnosti neizbježno je uvođenje novih grešaka u sistem ako sistem rapidno raste.

Da bi bile izbjegnute ovakve promjene u sistemu, po svima neželjene, razvijena je i uvedena nova metodologija nazvana „*agilna metodologija planiranja*“.

¹³⁹H. Kerzner, *Project Management, A Systems Approach to Planning, Scheduling and Controlling, Eight Edition*, Hoboken, NJ: John Wiley & Sons, Inc, 2003.

¹⁴⁰ *Agilni manifest* – <http://agilemanifesto.org/iso/hr/manifesto.html>

Agilnim planiranjem, u stvari, želimo ubrzati procese i izbjeći, ili izostaviti, one dijelove koji nas usporavaju. Za mnoge je agilno planiranje ustvari izbjegavanje pretjerane birokracije, tj. pretjerano trošenje vremena na dokumentiranje sistema koje se tehnički nazivaju dokumentno-orijentisano planiranje (engl. *document-oriented planning*).

Na mnogo načina, agilno planiranje bi se nazvalo kodno orijentisano planiranje (engl. *code-oriented planning*) koje u suštini kazuje kako je izvorni kôd glavni izvor dokumentacije.

Međutim, mnogi imaju različite stavove o takvim pristupima, te nedostatak dokumentacije može dovesti do neželjenih posljedica.

Agilne metodologije moramo predstaviti na dva načina kako bismo ih razlikovali od inženjerskog pristupa metodologijama:

Agilne metodologije su prije adaptivne, tj. prilagodljive, nego predvidljive. Inženjerske metode imaju tendenciju planiranje velikih procesa u sitne detalje i za dugi period te takav pristup funkcioniše dobro dokle god se nešto ne promijeni. Međutim, inženjerske metode se odupiru promjenama. Nasuprot njima, agilne metode vole promjene i svaka promjena je dobrodošla.

Agilne metodologije su više ljudski nego procesno orijentisane. Inženjerske metode imaju za cilj definisati proces koji će raditi, bez obzira na to ko će ga koristiti i mijenjati. Nasuprot tome, agilne metode pokazuju da niti jedan definisani proces neće nadoknaditi vještinu razvojnog tima pa je uloga samog procesa da podrži ulogu razvojnog tima.

5.1.1. SCRUM

SCRUM je procesni framework koji se koristi za upravljanje kompleksnim okolinama. SCRUM je okvir unutar kojega se mogu koristiti razni procesi i tehnike pa time možemo reći da pojam SCRUM nije metodologija ili tehnika razvoja softvera. Moramo spomenuti da se SCRUM framework sastoji od SCRUM timova i njihovih pridruženih uloga.

SCRUM je definisan i uspostavljen na način da minimalizuje potrebe za sastancima te uspostavi pravilnosti u projektu. Koristi se vremenski definisani događaj koji ima svoje maksimalno vrijeme trajanja. Taj način i takav pristup osigurava da se dovoljno vremena koristi za planiranje bez uzaludnog trošenja vremena.

SCRUM isporučuje proizvode u iterativnim i inkrementalnim fazama, a ujedno i maksimalizuje povratne informacije krajnjeg klijenta jer se kroz iterativne faze češće dobije i povratna informacija.

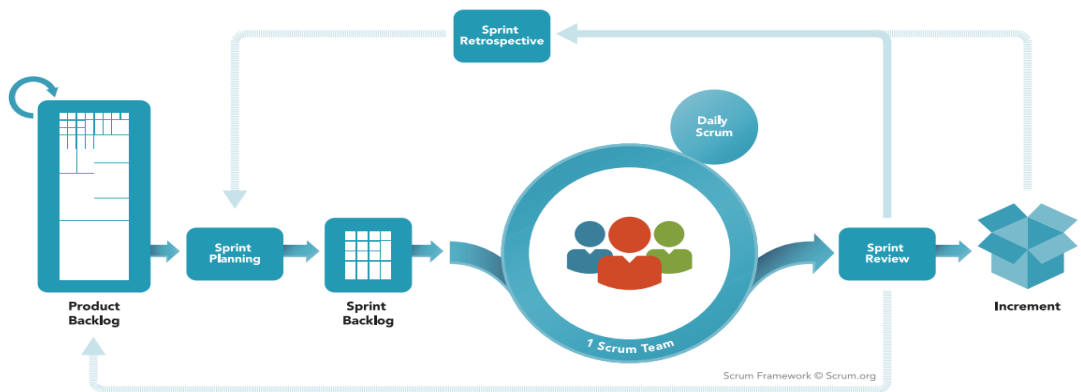
SCRUM se definiše na načina da je vlasnik proizvoda jedini odgovoran za upravljanje produktnim katalogom (engl. *Product Backlog*).

Pored vlasnika proizvoda, glavnu ulogu u SCRUM vode razvojni tim i Scrum Master. Razvojni tim se sastoji od profesionalaca koji rade konkretan posao i zadatke kako bi isporučili inkrement proizvoda na kraju svakog definisanog ciklusa sprinta.

Životni ciklus SCRUM frameworka¹⁴¹ je prikazan na sljedećoj ilustraciji.

¹⁴¹SCRUM Framework –

https://www.scrum.org/Portals/0/Documents/Collateral/ScrumFramework_17x11.pdf



Slika 1. SCRUM framework

Životni ciklus možemo opisati na sljedeći način:

- Vlasnik proizvoda unosi prijedloge i zahtjeve u produktni katalog,
- Sprint planiranje se odvija unutar cijelog tima te se zadaci (engl. *tasks*) raspoređuju po članovima tima te unose u sprint backlog,
- Svaki sprint je definisan istim periodom te se za taj određeni period i definiraju zadaci članovima tima. Period svakog sprintsa ne bi trebalo da bude duži od 2–3 nedjelje.
- Svaki sprint ima definiciju šta će se konkretno obaviti u navedenom periodu te na koji će se način odrediti i izraditi konačan proizvod.
- Dnevni scrum (engl. *Daily Scrum*) je sastanak u trajanju od 15 minuta koji služi razvojnom timu kao događaj za usklađivanje aktivnosti te je u planu da se na tom kratkom sastanku dogovore aktivnosti pojedinih članova razvojnog tima. Svaki član iznosi svoje stanovište šta je prethodno radio, te šta će raditi do idućeg dnevnog sastanka. Dnevni scrum se, po pravilu, uvijek održava na istom mjestu te u isto vrijeme da bi se smanjila kompleksnost. Scrum master je odgovoran da forsira tim za sudjelovanje u dnevnom sastanku te kratkom održavanju, ne duže od 15 minuta.
- Nakon svakog sprintsa (2–3 nedjelje) radi se Sprint Review te se na njemu odlučuje koliko je proizvod spreman za isporuku, tj. koje su funkcionalnosti spremne za krajnjeg korisnika. Moguće je da neke od funkcionalnosti nisu odrađene, te se na navedenom sastanku odlučuje što sa takvim neodrađenim funkcionalnostima. Najčešće se takve funkcionalnosti vraćaju u produktni katalog za neku drugu iteraciju ili se potpuno odbacuju s naznakom da se funkcionalnost neće implementirati.
- Nakon Sprint Reviewa radi se inkrement proizvoda i kratka retrospektiva sprintsa u kojem je poželjno da članovi tima iznesu poteškoće na koje su naišli tokom proteklog sprintsa kako se navedene poteškoće ne bi ponovile.
- Odmah po završetku prethodnog sprintsa započinje novi sprint sa sljedećom iteracijom i tako dokle god ima potrebe za održavanjem i razvojem proizvoda.

5.1.2. KANBAN

Kao odgovor na tradicionalnu metodologiju planiranja i procjenjivanja projektnih zadataka pojavila se i metodologija nazvana KANBAN. To je metoda za razvoj softverskih proizvoda i procesa s naglaskom na „u pravom trenutku“ (engl. *just-in-time*) isporuku. Pri navedenom se u ovom pristupu ne opterećuje razvojni tim. Naglasak je na tome da svaki pojedini član razvojnog tima sam uzima zadatke koji su popisani na ukupnoj listi zadataka.

S ovakvim pristupom proces do isporuke je korisniku jasno vidljiv napredak, a i svima koji sudjeluju u projektu.

Pokazalo se kao dobra praksa da se napredak na projektu na nekakav način vizualizuje i da bude dostupan svima na uvid.

Kanban metoda je poznata po tome da je projekat vizualiziran na način da su zadaci dostupni svima na zidu u obliku table na kojoj su nalijepljeni papirići sa zadacima. Ti papirići su takozvani *Kanban zadaci*, a tabla na kojoj se nalaze ti papirići se naziva *Kanban tabla*.

Kanban tabla se najčešće sastoji od 3 segmenta, a oni se najčešće dijele na:

- Dio u kojem se nalaze zadaci koje je potrebno napraviti u tekućem sprintu,
- Dio u kojem se nalaze odabrani zadaci od strane razvojnog tima,
- Dio u kojem se nalaze odrađeni zadaci.

Ovo je najjednostavniji način koji se pokazao da najbolje funkcioniše u svakoj vrsti projekta, bio on mali ili veliki, kratkoročni ili dugotrajni, jer su zadaci jasno vidljivi svima.

Na Kanban tabli su raspoređeni zadaci, a trenutno stanje zadataka je određeno natpisom u kojem su statusu. Na primjer „*treba uraditi*“ (engl. *To Do*), „*radi se*“ (engl. *Doing*) i „*urađeno*“ (engl. *Done*). Međutim, natpisi se mogu razlikovati u odnosu na različite timove, ovaj koji smo naveli je pogodan za razvojni tim, međutim ako se radi o timu koji se bavi testiranjem, Kanban tabla se može podijeliti u sljedeće segmente: „*testirano*“ (engl. *Tested*), „*potvrđeno*“ (engl. *Verified*), „*blokirano*“ (engl. *Blocked*).

Gotovo najbitnija karakteristika Kanban metode je posmatranje i poboljšanje procesa te je tokom životnog vijeka projekta potrebno obratiti pažnju na zadatke i njihovu cirkulaciju kroz sistem i kroz praćenje poboljšavati Kanban procese.

Najčešći oblik praćenja je način toka prenosa zadataka iz stupca u stubac. Trajanje pojedinog zadatka ne bi trebalo da bude duži od tri dana. Ako je potrebno više vremena, trebalo bi revidirati zadatak te ga podijeliti u više manjih zadataka i time ubrzati protok rješavanja dodijeljenog zadatka.

Distributivni timovi i virtuelna kanban ploča

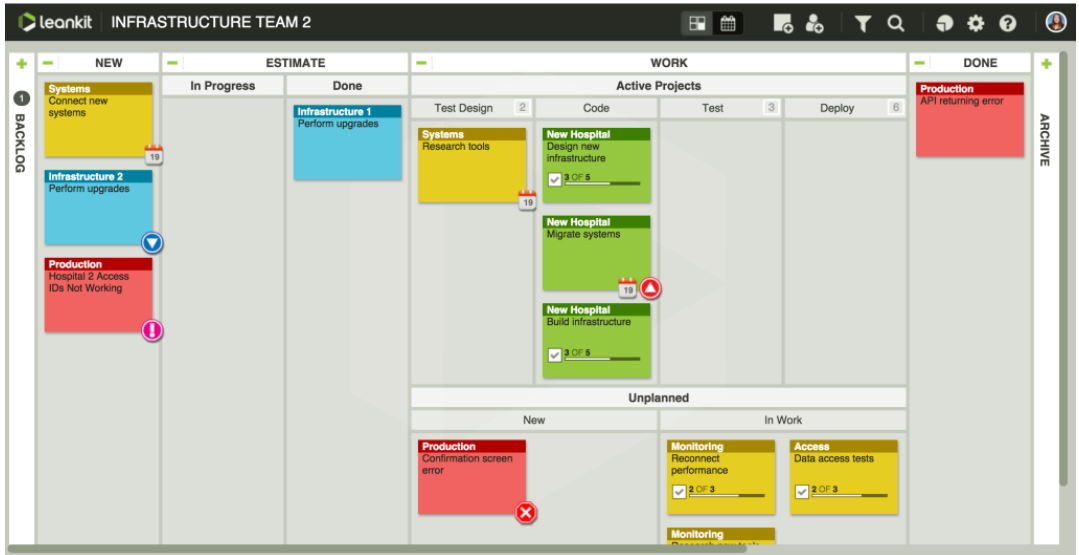
Poneki projekti zahtijevaju distributivne timove, što po prirodi posla, što po kompetenciji članova tima.

Distributivni timovi su timovi kod kojih se članovi tima ne nalaze na istoj lokaciji, čak možda ne samo u istoj kancelariji, nego možda čak u drugoj zgradi na drugoj lokaciji ili čak u drugom gradu pa čak i drugoj državi.

Uvijek se postavlja pitanje da li u ovom slučaju tim može raditi po Kanban metodologiji kada pojedini članovi tima nisu u mogućnosti fizički odabrati zadatak te ga prebaciti u drugi status.

Ovome je, naravno, doskočila softverska tehnologija te raznim alatima stvorila virtuelnu Kanban tablu.

Ovakvo stvorena tabla daje mogućnost timovima ili članovima tima koji se nalaze na udaljenoj lokaciji da budu informisani u svakom trenutku isto kao da se i nalaze u istoj prostoriji s drugim članovima tima na lokaciji.



Slika 2. Virtuelna Kanban tabla

Kako vidimo, prikazana virtuelna tabla¹⁴² gotovo nema razlike od stvarne fizičke table. Na virtuelnoj tabli su prikazani i virtuelni papirići sa zadacima koje treba obaviti te se na potpuno isti način odvija proces izvedbe i rješavanje zadataka.

Uvijek se postavlja dodatno pitanje: Kako će ovako udaljeni članovi tima znati da je neko uzeo novi zadatak? Da li je pravi zadatak odabran? Da li ima prioritelnijih zadataka?

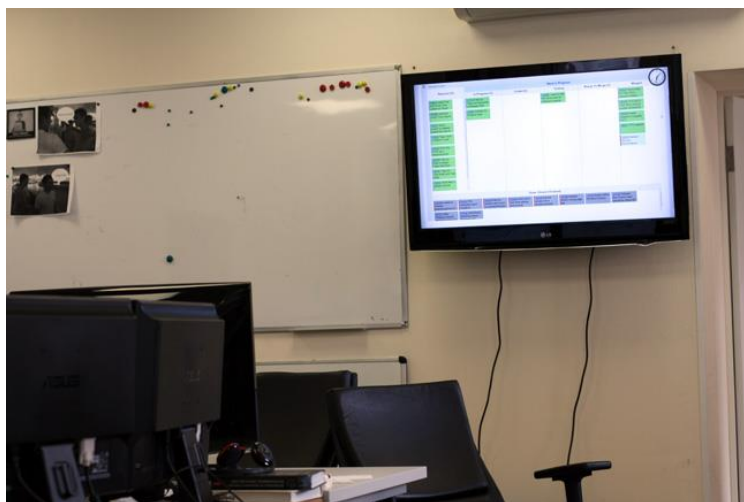
Odgovor na ovakvo pitanje smo dali u razmatranju SCRUM metodologije, a na potpuno isti način se primjenjuje i kod Kanban metodologije.

Odgovor je kratak i jasan: Raspored i izbor zadataka se odrađuje na 15-minutnom dnevnom sastanku.

Nema prepreka zašto se tim ne bi sastao svakodnevno i na taj način izabrao zadatke, a pri tome još i dodatno ukratko razjasnio moguće nedoumice. Nakon ovakvog odabira i raspodjele zadataka svaki član tima nastavlja raditi na svom zadatku.

U ovakvim distributivnim timovima takođe je praksa postavljanje virtuelnih tabli u vidu ekrana (monitor ili tv) s velikim dijagonalama na zid te time stvoriti fizičku Kanban tablu.

¹⁴² Kanban tabla – <https://leankit.com/learn/kanban/kanban-board>



Slika 3. Digitalna Kanban tabla na velikom ekranu

Kako vidimo na prikazanoj fotografiji,¹⁴³ virtualna Kanban tabla može takođe biti potpuno digitalna, a takođe vidljiva s radnog mjesta svim članovima tima. Prenos zadataka iz stupca u stubac se događa u realnom vremenu te je u svakom trenutku moguće vidjeti kako se projekt odvija te koliko zadataka još treba obaviti, koji su zadaci u toku izrade, te koji su zadaci već obavljani.

Ovakva virtualna Kanban tabla ima još i drugih benefita, a jedan od najznačajnijih, makar projekt menadžerima, je statistika.

Statistički podaci puno govore o samom projektu te se na vrijeme može uočiti i planirati projekat na drugi način ako se uvidi da projekat ide u pogrešnom smjeru, tj. da se zadaci ne obavljaju predviđenom brzinom, ili čak da se predviđeni zadaci obavljaju prebrzo pa je potrebno dodati još zadataka iz produktnog backloga u trenutni sprint.

6. ISTRAŽIVANJE I STATISTIČKI PODACI KORIŠĆENJA AGILNE METODOLOGIJE

Statistički podaci korišćeni u istraživanju su najviše od firme VerisonOne. Navedena firma svake godine sprovodi istraživanje u vezi sa upotrebom agilnih metoda.

U istraživanju se može učestvovati u bilo kojem trenutku ispunjavajući upitnik na stranicama <http://stateofagile.versionone.com>.

U ovom radu koristi se izvještaj o rezultatima iz 2015. godine (10th Annual State of Agile Development, 2015¹⁴⁴).

U navedenom izvještaju (upoređujući sa izvještajima prethodnih godina) vidimo da u 2015. godini 95% firmi koristi agilni razvoj softvera, što je mali pomak od 1% u odnosu na godinu prije, no veliki pomak u odnosu na period od dvije godine kada je na globalnom nivou taj podatak iznosio 88% (8th Annual State of Agile Development, 2013¹⁴⁵).

¹⁴³Kanban tabla na 42-inčnom TV – <https://www.targetprocess.com/blog/2009/08/kanban-board-on-42-inches-tv>

¹⁴⁴Statistički podaci, VerisonOne 10th Annual State of Agile Report 2015 – <https://versionone.com/pdf/VersionOne-10th-Annual-State-of-Agile-Report.pdf>

¹⁴⁵Statistički podaci, VerisonOne 8th Annual State of Agile Report 2013 – [https://www5.learningtree.com/certified-scrum-training/Files/2013-state-of-agile-survey LT.pdf](https://www5.learningtree.com/certified-scrum-training/Files/2013-state-of-agile-survey_LT.pdf)

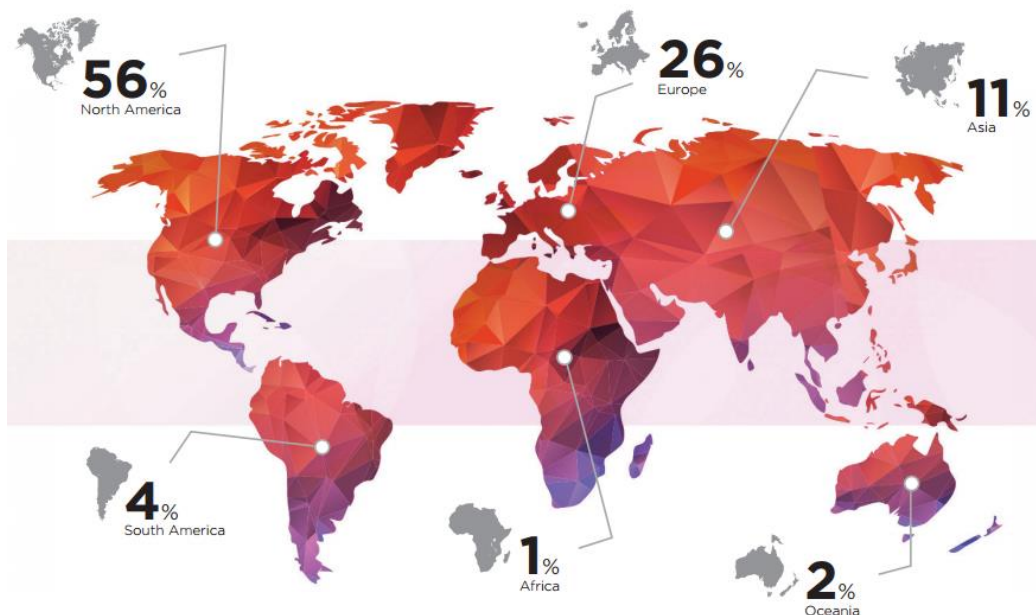
Company Experience



Slika 6. Iskustvo firmi u upotrebi agilnih metoda razvoja softvera

Istraživanje sprovedeno u ovom izvještaju firme VerisonOne je obuhvatao firme iz cijelog svijeta (56% Sjeverna Amerika, 26% Evropa, 11% Azija, 4% Južna Amerika, 1% Afrika i 2% Okeanija).

Takođe u navedenom izvještaju vidimo da su mnoge firme u porastu s korišćenjem agilnih metodologija u odnosu na prethodne godine.



Slika 7. Lokacije organizacija korišćene u istraživanju (2015)

Agilnim metodama dato je svojstvo iterativnosti i kontinuirane povratne informacije koje omogućuju uspješniju doradu softvera, ali i samu isporuku konačnog proizvoda. Takođe kontinuirano planiranje, testiranje, integracija i neki drugi oblici su najčešća prednost agilnih metoda.

Najveće prednosti i koristi koje firme dobiju od korišćenja agilnih metoda izražene su kroz 4 faktora:

- Povećanje vidljivosti projekta (engl. *Visibility*)
- Povećanje prilagodbe na projektu (engl. *Adaptability*)
- Povećanje poslovne vrijednosti projekta (engl. *Business Value*)

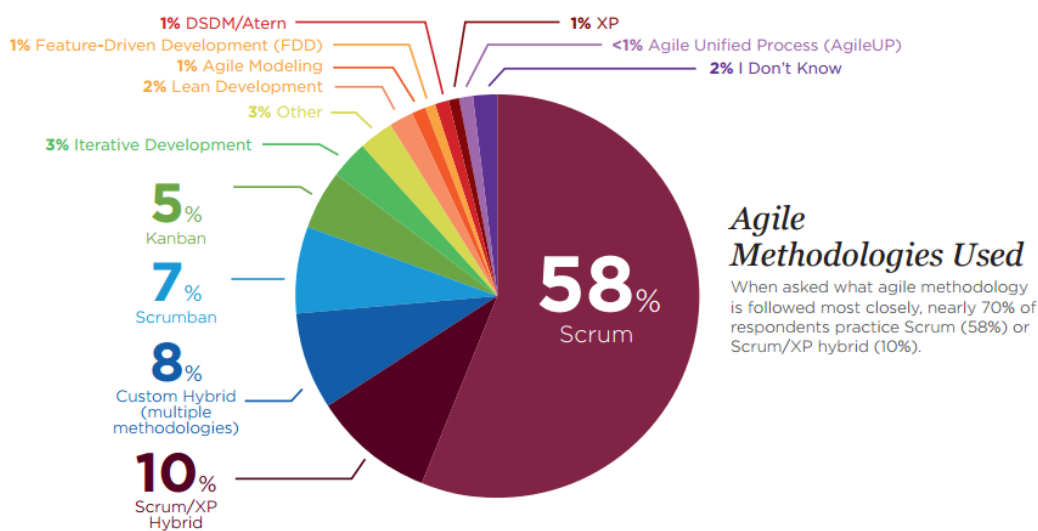
- Smanjenje rizika uspješnosti projekta (engl. *Risk*)

Po navedenom možemo zaključiti sljedeće ponašanje:

- Vidljivost projekta je gotovo kontinuirana za sve vrijeme trajanje projekta. U odnosu na tradicionalnu metodu je to znatno poboljšanje jer kod tradicionalnih metoda projekat ima najbolju vidljivost na početku i na kraju projekta, tokom cjelokupnog razvoja projekta vidljivost se gubi korišćenjem tradicionalnih metoda.
- Povećanje prilagodbe je takođe kontinuirani proces korišćenjem agilnih metoda, dok je vidljivo da se prilikom tradicionalnih metoda prilagodba drastično razlikuje i nije tako poželjna.
- Vrijednost projekta je vidljiva i najznačajnija već na samom početku projekta te zadržava i povećava svoju vrijednost tokom cijelog perioda projekta. Ovo ponašanje nam je direktno omogućeno stavkom vidljivosti projekta, te ima značajan uticaj na njegovu vrijednost. Tradicionalan način nam pokazuje kako je vrijednost projekta vidljiva tek na kraju, tj. pri isporuci proizvoda.
- Rizik je drastično smanjen, tj. prilikom promjena na projektu smanjujemo rizik jer se prilagođavamo promjenama. Tradicionalan način nam pokazuje da se rizik kontinuirano zadržava prilikom cijelog perioda projekta te naglo opada pri kraju projekta, tj. kod finalne isporuke.

7. NAJČEŠĆE KORIŠĆENE AGILNE METODE

Po istraživanju firme VerisonOne rezultati pokazuju sljedeću učestalost korišćenja pojedinih agilnih metoda.



Slika 6. Najčešće korišćene agilne metodologije

Zaključili smo da je Scrum najzastupljenija metodologija, a to se vidi i iz istraživanja jer istraživanje pokazuje brojku od 58% što je poprilično velika zastupljenost.

Međutim, ovoj brojci doprinosi i sama činjenica da se sama riječ „*agilno*“ ponekada poistovjećuje sa metodologijom „*Scrum*“ te time sama Scrum metodologija dobije na zvučnosti i važnosti u agilnim procesima.

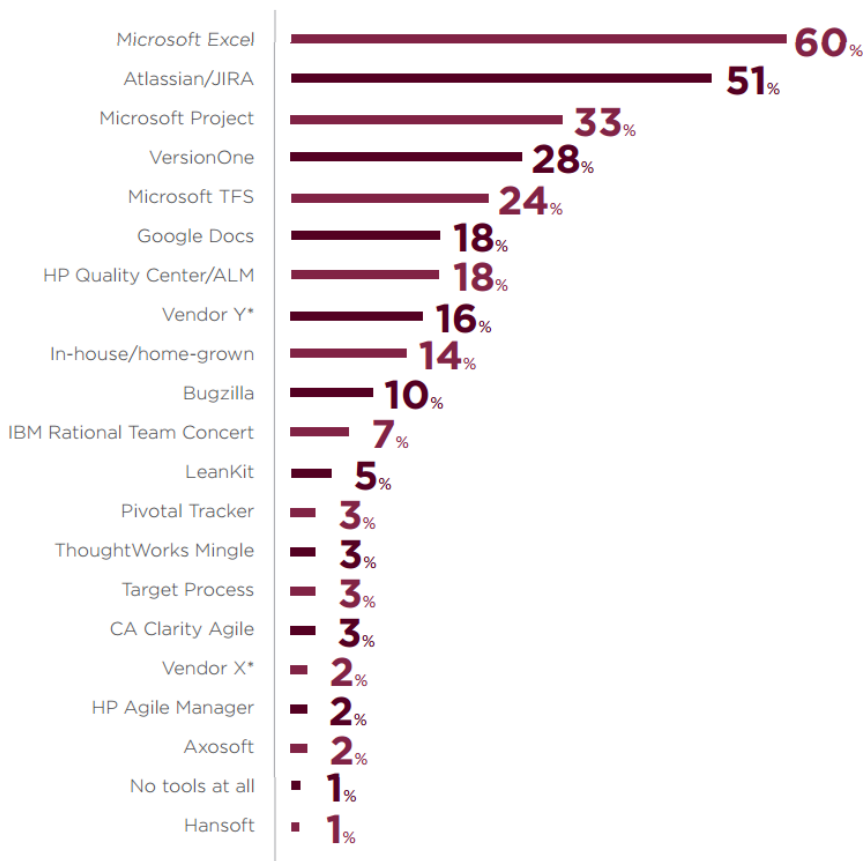
Pored vidljivo najzastupljenije metodologije, iz navedenog izvještaja istrage možemo vidjeti da se relativno često razvijaju i hibridne metodologije koje su nastale iz više agilnih metodologija. Iz toga vidimo kako su i metodologije prilagodljive te i sam nastanak novih imena metodologija nastaju iz imena postojećih metodologija.

Uzmimo za primjer ime metodologije „*Scrumban*“. Iz samog imena možemo zaključiti da je metodologija nastala iz dvije, a to su „*Scrum*“ i „*Kanban*“.

Ono što takođe možemo zaključiti iz novonastale metodologije, a i brojkama iz istraživanja je da je Scrumban metodologija manje zastupljena nego Scrum, ali više nego Kanban. Iz navedenog zaključujemo da je Kanban metodologija proširena sa Scrum metodologijom, a u dovoljnom omjeru da više nije čisti Kanban, a niti toliko širok kao Scrum.

8. ALATI ZA UPRAVLJANJEM PROJEKATA

Za uspješno upravljanje projektom moraju se koristiti i alati, te se i kod agilnih metodologija koriste alati koji su se pokazali efikasnim. Statistički podaci pokazuju sljedeće alate koji se koriste za upravljanje projektom.



Slika 7. Alati za upravljanje projektima (VerisonOne, 2015)

Kao što je vidljivo na prethodnoj ilustraciji, većina firmi (njih 60%) koriste Microsoft Excel kao alat za vođenje projekta, a tome se može pridodati kriterijum jednostavnosti upotrebe te

nastavak načina korišćenja po tradicionalnim metodologijama. Osnovni princip agilnih metoda je jednostavnost, tj. ne komplikovati oko dokumentacije. Po navedenom istraživanju vidimo da se i sama firma VerisonOne sa svojim alatima nalazi u vrhu na četvrtom mjestu.

ZAKLJUČAK

Ovaj rad je gotovo u potpunosti posvećen planiranju projekata s naznakom na agilne metodologije. Takođe prikazali smo razliku između tradicionalnih metodologija te njihovih prednosti kod manjih projekata.

Kod agilnih metodologija poput Scruma ili Kanbana situacija komunikacije je potpuno drugačija u odnosu na tradicionalne metodologije. Komunikacija prema klijentima je relativno otvorena i podložna promjenama. Predstavljene agilne metodologije prikazuju projektno izvođenje u iteracijama, tj. dostavljaju se određene funkcionalnosti kada su gotove, za razliku od tradicionalnog dostavljanja funkcionalnosti koje ovise o cjelokupnoj funkcionalnosti cjelokupnog proizvoda.

Kod agilnih metodologija uočava se kako su planiranja i procjene jednostavnije i tačnije jer se bazira na manjim grupama funkcionalnosti koje su rascjepkane u manje cjeline te ih je moguće tačnije procijeniti. Budući da su funkcionalnosti rascjepkane u manje cjeline, promjene ne utiču puno na cjelokupnu funkcionalnost proizvoda te je potrebno i čak poželjno da se uočene promjene na proizvodu događaju što češće jer se takvim pristupom, u stvari, pravi željena funkcionalnost proizvoda.

Kod tradicionalnih metoda ovakve promjene izazivaju neželjen efekat jer se čeka gotov proizvod, a tek onda i povratna informacija krajnjeg kupca. Tek nakon povratne informacije se može uvidjeti koje su potrebne promjene te se tek nakon tog vremena mogu raditi nove procjene i implementacije.

Kako vidimo, tradicionalne metodologije nisu poželjne kod velikih projekata jer se vrlo tromo dolazi do informacija za koje je potrebno raditi promjene.

Agilne metodologije su se pokazale vrlo adekvatnom primjenom u softverskoj industriji ne samo iz razloga brzine i efikasnosti, nego i činjenicom da su promjenjive i nisu vodene metodom da sve mora biti onako kako je definisano pravilnikom. Kod agilnih metodologija pravilnik je samo smjernica te se i sam pravilnik može mijenjati u odnosu na projektne potreba. Može se reći da se iz pravilnika uzme dio koji je poželjan, a isto tako i odbaci dio koji je nepoželjan, a da se pri tome ne naruši strategija agilnosti.

LITERATURA

1. PMI, *A Guide to the Project Management Body of Knowledge*, Third Edition (PMBOOK Guide). Newton Square, PE: Project Management Institute, 2004.
2. H. Kerzner, *Project Management, A Systems Approach to Planning, Scheduling and Controlling*, Eight Edition, Hoboken, NJ: John Wiley & Sons, Inc, 2003.
3. R.L. Kleim and I.S. Ludin, *Project Management Practitioner's Handbook*, AMACOM Books, 1998.
4. Cole A., *Runaway Projects – Causes and Effects*, Software World, UK, Vol 26. No. 3, 1995.

5. Nekorištena funkcionalnost, Mike Cohn, Founder of Mountain Goat Software – <https://www.mountaingoatsoftware.com/blog/are-64-of-features-really-rarely-or-never-used>
6. SCRUM Framework – https://www.scrum.org/Portals/0/Documents/Collateral/ScrumFramework_17x11.pdf
7. Kanban ploča – <https://leankit.com/learn/kanban/kanban-board>
8. Kanban ploča na 42 inčnom TV-u – <https://www.targetprocess.com/blog/2009/08/kanban-board-on-42-inches-tv>
9. Statistički podaci, VerisonOne 10th Annual State of Agile Report 2015 – <https://versionone.com/pdf/VersionOne-10th-Annual-State-of-Agile-Report.pdf>
10. Statistički podaci, VerisonOne 8th Annual State of Agile Report 2013 – https://www5.learningtree.com/certified-scrum-training/Files/2013-state-of-agile-survey_LT.pdf