

AGILNE METODE - OPIS I PRIMJER RAZVOJA INFORMACIJSKOG SUSTAVA

Čuturdić, Enna

Undergraduate thesis / Završni rad

2022

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Economics in Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Ekonomski fakultet u Osijeku**

Permanent link / Trajna poveznica: <https://urn.nsk.hr/urn:nbn:hr:145:557197>

Rights / Prava: [In copyright](#)/[Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2025-01-28**



Repository / Repozitorij:

[EFOS REPOSITORY - Repository of the Faculty of Economics in Osijek](#)



Sveučilište Josipa Jurja Strossmayera u Osijeku
Ekonomski fakultet u Osijeku
Preddiplomski sveučilišni studij Poslovna informatika

Enna Čuturdić

**AGILNE METODE – OPIS I PRIMJER RAZVOJA
INFORMACIJSKOG SUSTAVA**

Završni rad

Osijek, 2022.

Sveučilište Josipa Jurja Strossmayera u Osijeku
Ekonomski fakultet u Osijeku
Preddiplomski sveučilišni studij Poslovna informatika

Enna Čuturdić

**AGILNE METODE – OPIS I PRIMJER RAZVOJA
INFORMACIJSKOG SUSTAVA**

Završni rad

Kolegij: Oblikovanje i implementacija informacijskih sustava

JMBAG: 00102221393

E – mail: ecuturdic@efos.hr

Mentor: Prof.dr.sc. Josip Mesarić

Osijek, 2022.

Josip Juraj Strossmayer University of Osijek
Faculty of Economics in Osijek
Undergraduate Study Business Informatics

Enna Čuturdić


**AGILE METHODS - DESCRIPTION AND EXAMPLE OF INFORMATION SYSTEM
DESIGN**

Final paper

Osijek, 2022.

IZJAVA

O AKADEMSKOJ ČESTITOSTI, PRAVU PRIJENOSA INTELEKTUALNOG VLASNIŠTVA, SUGLASNOSTI ZA OBJAVU U INSTITUCIJSKIM REPOZITORIJIMA I ISTOVJETNOSTI DIGITALNE I TISKANE VERZIJE RADA

1. Kojom izjavljujem i svojim potpisom potvrđujem da je ZAVRŠNI
(navesti vrstu rada: završni / diplomski / specijalistički / doktorski) rad isključivo rezultat osobnoga rada koji se temelji na mojim istraživanjima i oslanja se na objavljenu literaturu. Potvrđujem poštivanje nepovredivosti autorstva te točno citiranje radova drugih autora i referiranje na njih.
2. Kojom izjavljujem da je Ekonomski fakultet u Osijeku, bez naknade u vremenski i teritorijalno neograničenom opsegu, nositelj svih prava intelektualnoga vlasništva u odnosu na navedeni rad pod licencom *Creative Commons Imenovanje – Nekomercijalno – Dijeli pod istim uvjetima 3.0 Hrvatska*. 
3. Kojom izjavljujem da sam suglasan/suglasna da se trajno pohrani i objavi moj rad u institucijskom digitalnom repozitoriju Ekonomskoga fakulteta u Osijeku, repozitoriju Sveučilišta Josipa Jurja Strossmayera u Osijeku te javno dostupnom repozitoriju Nacionalne i sveučilišne knjižnice u Zagrebu (u skladu s odredbama Zakona o znanstvenoj djelatnosti i visokom obrazovanju, NN br. 123/03, 198/03, 105/04, 174/04, 02/07, 46/07, 45/09, 63/11, 94/13, 139/13, 101/14, 60/15).
4. izjavljujem da sam autor/autorica predanog rada i da je sadržaj predane elektroničke datoteke u potpunosti istovjetan sa dovršenom tiskanom verzijom rada predanom u svrhu obrane istog.

Ime i prezime studenta/studentice: EMMA ČUTURDIĆ

JMBAG: 00102221393

OIB: 66344318261

e-mail za kontakt: emmaegg@gmail.com

Naziv studija: TRENDIPLOMSKI SVEUČILIŠNI STUDIJ POSLOVNA INFORMATIKA

Naslov rada: AGILNE METODE - OPIS I PRIMJER RAZVOJA INFORMACIJSKOG SUSTAVA

Mentor/mentorica diplomskog rada: Prof. dr. sc. JOSIP MESARIĆ

U Osijeku, 7.9. 2022. godine

Potpis Emma Č.

SAŽETAK

U radu su ukratko opisane tradicionalne i agilne metode upravljanja projektima te je potanko objašnjen Scrum, koji je najzastupljeniji među agilnim metodama. Objašnjen je Scrum tim i njegovi članovi. Scrum tim je mala grupa ljudi, a sastoji se od jednog Scrum Mastera, jednog Product Ownera i od Developera. Opisan je tijek Scrum-a. Sprint je srž Scruma, pri čemu dolazi do pretvorbe ideja u vrijednost. Sprintevi su događaji čija je duljina fiksna, traju mjesec dana ili manje, a njihova svrha jest postizanje dosljednosti. Tijekom Planiranja Sprinta priprema se posao kojeg je potrebno odraditi unutar Sprinta, a Dnevni Scrum provodi se u trajanju od 15 minuta s ciljem provjere napretka u ostvarenju Cilja Sprinta te eventualno prilagođavanja Sprint Backloga, u njemu sudjeluju Developeri te eventualno Product Owner ili Scrum master. Pregled Sprinta jest radni sastanak, cilj mu je provjera ishoda Sprinta i određivanje eventualnih prilagodbi u budućnosti, pri čemu tim predstavlja rezultate rada i vodi se rasprava o postignutom napretku u ostvarenju cilja. Retrospektiva se odvija radi planiranja načina povećanja kvalitete i učinkovitosti. Artefakti u Scrum-u su: Product Backlog (obveza: Cilj proizvoda), Sprint Backlog (obveza: Cilj Sprinta) te Inkrement (obveza: Definicija Gotovog). Ukratko je opisan informacijski sustav u bankarstvu, način upravljanja te njegove komponente. Pomoću alata Jira na pojednostavljenom primjeru prikazan je razvoj mobilne aplikacije u bankarstvu.

Ključne riječi: informacijski sustav, bankarstvo, agilne metode, Scrum.

ABSTRACT

This research paper briefly describes traditional and agile methods of project management, as well as Scrum, which is the most common Agile Method. Scrum team and its members are also explained. Scrum team is a small group of people that consists of one Scrum Master, one Product Owner and a Developer. Scrum flow is described as well. Sprint is a core of Scrum where ideas are transformed into values. Sprints are cases with fixed length that occur in duration of one month or less whose purpose is to achieve consistency. During Sprint Planning, work is being prepared for Sprint that needs to be done within it, Daily Scrum is carried out for 15 minutes with the goal of tracking progress achieved in the Sprint Goals and eventually adjusting the Sprint Backlog where Developers and also sometimes Product Owner or Scrum Master are participating. Sprint Review is a work meeting with a goal to review the outcome of the Sprint and to determine possible adjustments in the future, the team then presents results of the work which is followed by debate to discuss the achievements that are made. Retrospective takes place in order to plan ways for increasing quality and efficiency. The Scrum Artifacts are: Product Backlog (commitment: Product goal), Sprint Backlog (commitment: Sprint Goal) and Increment (commitment: Definition of Done). To summarize, banking information system, management methods and its components are all described. With the help of Jira Software, on a simple example, the development of a mobile application in banks is shown.

Key words: Information System, banking, Agile methods, Scrum.

SADRŽAJ

1. Uvod.....	1
1.1. Tradicionalne metode upravljanja projektima.....	1
1.2. Agilne metode upravljanja projektima.....	2
2. Cilj rada.....	5
3. SCRUM agilna metoda	5
3.1. Scrum tim	8
Developeri.....	9
Product Owner.....	9
Scrum Master	10
3.2. Tijek Scruma.....	12
Sprint.....	12
Planiranje Sprinta	13
Dnevni Scrum.....	14
Pregled Sprinta	14
Retrospektiva.....	15
3.3. Artefakti u Scrumu	16
3.4. Prednosti i nedostaci Scrum metodologije	17
4. Informacijski sustav u bankarstvu (IS).....	19
4.1. Upravljanje informacijskim sustavom banke.....	20
4.2. Komponente informacijskog sustava banke	21
5. Primjena SCRUM metode u bankarstvu	22
6. Rasprava	31
7. Zaključak.....	34
8. Literatura	35
Popis tablica	39
Popis slika	39

1. Uvod

Projektni menadžment suočava se s nizom izazova i nepoznanica. Pored toga, za uspješnost projekta, važan izbor koji se mora donijeti prije postavljanja samog projekta jest odabir metode vođenja projekta. Metode su mnogobrojne, a „najbolji savjet projektnim menadžerima jest da odaberu onu metodu koja najbolje odgovara njihovoj organizaciji i samom projektu“ (Thompson, 2019).

Upravljanje projektima je od velike važnosti za stvaranja novog, jedinstvenog proizvoda ili usluge. Kako bi upravljanje projektom bilo adekvatno, postoje različite metode za upravljanje projektima i njihov izbor ovisit će o značajkama željenog proizvoda/usluge te značajkama samog projekta. Dvije najveće skupine metoda upravljanja projektima su tradicionalne i agilne metode.

1.1. Tradicionalne metode upravljanja projektima

Tradicionalno upravljanje projektima definira se kao „primjena alata, vještina, tehnika i znanja u aktivnostima projekta s ciljem ispunjenja zahtjeva projekta. Tradicionalno upravljanje projektima sastoji se od 5 faza: pokretanje, planiranje, izvršavanje, nadgledanje i kontrola te zatvaranje. Svih pet faza izvršava se pod vodstvom i podrškom voditelja projekta kao i projektnog tima“ (Project Management Institute, 2013).

Tradicionalnim metodologijama smatraju se sve metodologije čiji je pristup slijedni (fazni) odnosno u kojim se nakon prelaska na sljedeći proces ne vraća na prethodni. Tradicionalne metodologije imaju uporište u vodopadnom modelu, struktura im je jasna, a prilagodljivost slaba. Opseg projekta je kod tih metoda jasno definiran, faze i procesi su strogo razloženi. Malena je podložnost promjenama. Kod tradicionalnih metodologija vremenski je okvir jasno definiran, rokovi su točno određeni, a rizici poznati i rijetko se događaju neočekivane situacije.

Zajedničke karakteristike tradicionalnih modela su (Awad, 2005):

- prediktivni pristup,
- sveobuhvatna dokumentacija,
- orijentiranost na proces,
- orijentiranost na alate.

Vodopadni model predstavlja je tradicionalne metodologije vođenja projekata, a njegovo osnovno obilježje jest nemogućnost prelaska na slijedeću fazu sve dok nije završena prethodna te je također onemogućen povratak na prethodnu fazu. Prvi korak u vodopadnom modelu jest analiza zahtjeva, a slijedi oblikovanje. Treći korak čini izgradnja, slijeda ga uvođenje te vodopadni model završava održavanjem i primjenom (Pavlič, 2011). Prednost vodopadnog modela jest kompletna definiranost zahtjeva projekta prije samog uvođenja, unaprijed definirano trajanje, cijena i opsežnost projekta, mogućnost prepoznavanja potencijalnih problema na samom početku projekta, opsežna i temeljita dokumentacija te jednostavnost testiranja sustava (scenariji unaprijed definirani u dokumentaciji) (Palmquist i sur., 2013). Kao najčešći nedostaci vodopadnog modela navodi se nemogućnost prelaska na slijedeću fazu sve dok nije završena prethodna, a u slučaju da se učini iznimka i vrati se na prethodnu fazu, uvelike se podiže cijena i produljuje trajanje projekta. Također, potencijalni je problem za klijente na samom početku projekta iznijeti sve tražene zahtjeve (Palmquist i sur., 2013).

Radi učestalih promjena i povećanja zahtjevnosti samih projekata, tradicionalne metode pokazale su se kao neučinkovito rješenje. Osnovni razlog napuštanja tradicionalnih metoda jest nedostatna uključenost klijenta u sve faze razvoja projekta, već spomenuta nemogućnost vraćanja na prethodnu fazu projekta te provođenje testiranja tek na njegovu završetku.

1.2. Agilne metode upravljanja projektima

Početakom 90-ih godina prošlog stoljeća IT sektor je započeo svoj brzi i široko obuhvatni rast, te su tzv. vodopadne (eng. *waterfall*) metode pokazale svoj nedostatak jer se *software* (engl. kompjuterski program) nije mogao pravovremeno isporučiti. Kao rezultat toga pojavio se tzv. *Agile* (engl. agilni, okretan) pokret čime je omogućen razvoj u skladu s klijentovim (naručiteljevim) željama te ispunjavanje zadanog vremenskog roka. Prema Thompson (2019), „agilne metode su se pojavile i dugo zadržale u IT sektoru, ali posljednjih godina prelaze u gotovo sve sfere i industrije u kojima je projektni menadžment od velikog značaja“.

Uporaba tradicionalnih metodologija dovela je do nezadovoljstva među voditeljima IT projekata, pa je 2001. godine održan sastanak sedamnaestero stručnjaka iz područja softverskog inženjerstva. Tada je nastao Agilni manifest koji se smatra prvotnim dokumentom i vodičem za agilno vođenje projekata, koji definira osnovne postavke budućih agilnih metoda. Četiri su osnovne poruke Agilnog manifesta (Beck i sur., 2001):

- Individualci i njihovi međusobni odnosi (interakcija) ispred procesa i alata
- Uporabljiv softver ispred opsežne dokumentacije
- Suradnja s klijentom (naručiteljem) ispred pregovaranja o ugovoru
- Reakcija na promjenu ispred ustrajanja na praćenju plana

Agilni manifest ima veliku važnosti jer je njime omogućeno postizanje veće uspješnosti i smatra se svojevrsnim vodičem za upravljanje projektima agilnim metodama. Agilnost je prema tome skup vrijednosti i principa kojima se potiče određeno ponašanje, a koje su primjerene za projekte čije je obilježje velika promjenjivost, primjerice IT sektor i telekomunikacije (Highsmith i Cockbur, 2001). Definirano je dvanaest načela agilnog projektnog menadžmenta prema kojima bi se projekti trebali voditi (Thompson, 2019):

1. Najveći prioritet je zadovoljstvo naručitelja koje se postiže brзом i kontinuiranom isporukom.
2. Prihvatljive su izmjene u svim fazama razvoja proizvoda.
3. Prihvatljiva je povećana učestalost isporuke proizvoda ili usluge, u razmacima od nekoliko tjedana ili mjeseci, s ciljem da razmaci budu što kraći.
4. Dionici (naručitelji) i razvojni tim usko surađuju prilikom razvijanja proizvoda ili usluge, tijekom cjelokupnog trajanja projekta.
5. Projekt se gradi oko grupe motiviranih pojedinaca.
6. Interakcija i komunikacija licem u lice je najučinkovitiji oblik komunikacije u razvojnom timu.
7. Funkcionalni proizvod jest primarno mjerilo uspjeha (svrha projekta nije samo proći kroz proces od početka do kraja).
8. Agilni procesi potiču i zagovaraju održivi razvoj.
9. Agilnost se poboljšava dobrim dizajnom i neprekidnim ustrajanjem na tehničkoj izvrsnosti.
10. Jednostavnost je element od najveće važnosti.
11. Tim redovno razmatra načine da postaje učinkovitiji te prema tome usklađuje i prilagođava svoje ponašanje.

Jedna od najvećih prednosti agilnog vođenja projekta svakako se očituje u odnosu s klijentom, točnije u njegovoj uključenosti. Isporuka gotovog proizvoda klijentu događa se

ranije i češće, što omogućuje klijentu da testira taj proizvod te da povratnu informaciju. Na takav način se unaprjeđuje i odnos unutar samog tima. Od prednosti agilnog vođenja projekata ističu se slijedeće (Sharma i sur., 2012):

- veća kvaliteta proizvoda,
- veće zadovoljstvo korisnika,
- povećan nadzor nad projektom,
- smanjenje rizika,
- porast zadovoljstva članova tima,
- veća predvidljivost projekta.

Mogući nedostaci agilnog vođenja projekata su slijedeći (Sharma, Sakar i Gupta, 2012):

- loš menadžment resursa,
- oskudna i neprecizna dokumentacija,
- izrada proizvoda u previše fragmenata,
- neprecizno definiranje zatvaranja projekta,
- zahtjevno mjerenje rezultata.

Najpoznatije agilne metode su:

- SCRUM
- Extreme Programming (XP)
- Crystal
- Dynamic Systems Development Method (DSDM)
- Lean Development (LD)
- Feature-Driven Development (FDD)
- Adaptive Software Development (Yuonan i Mamedov, 2020).

Tablica 1. Razlika između agilnih i tradicionalnih metoda (Awad, 2005):

	Agilne metode	Tradicionalne metode
Pristup	Adaptivni	Prediktivni
Mjerenje uspjeha	Poslovna vrijednost	Posvećenost planu
Veličina projekta	Mali	Veliki
Stil menadžmenta	Decentralizirani	Autokratski
Pogled na promjene	Prilagodljivost promjenama	Održivost promjena
Kultura	Vodstvo - suradnja	Naređivanje - kontrola
Dokumentacija	Oskudna	Opsežna
Naglasak	Orijentacija na ljude	Orijentacija na proces
Ciklusi	Brojni	Ograničeni
Domena	Nepredvidiva/istraživačka	Predvidiva
Planiranje unaprijed	Minimalno	Opsežno
Povratak na investitora	U ranoj fazi projekta	Na kraju projekta
Veličina tima	Mali/kreativni	Veliki

2. Cilj rada

Cilj ovog rada je dati prikaz karakteristika agilnih metoda te koristeći programski alat Jira prikazati kako se agilne metode primjenjuju na konkretnom primjeru projekta poboljšanja karakteristika IS-a banke.

3. SCRUM agilna metoda

Kada se govori o agilnim metodama, najzastupljeniji je Scrum. Scrum metodologija olakšava koordiniranu aktivnost programera koji „lome“ svoj rad na manje zadatke koji se mogu završiti unutar ciklusa ograničenog trajanja („sprinteva“), te je omogućeno praćenje napretka i ponovno planiranje tijekom redovnih sastanaka (Hidalgo, 2019). Termin „scrum“

definira se kao „holistički pristup fleksibilnom, autonomnom i dinamičkom timskom radu” (Nonaka i Takeuchi, 1995), te je izdvojeno šest osnovnih karakteristika:

1. Ugrađena nestabilnost – visoki menadžment nudi projektnom timu velik stupanj slobode te istovremeno postavlja izazovne ciljeve.
2. Samoorganizirajući projektni timovi – grupe preuzimaju inicijativu i razvijaju neovisan raspored njihovog rada.
3. Preklapajuće faze razvoja – umjesto pristupa po principu sekvenci gdje projekt prolazi kroz nekoliko faza korak po korak, preklapajući pristup povećava brzinu i fleksibilnost, zajedničku odgovornost i suradnju.
4. Višerazinsko učenje – članovi tima sudjeluju u kontinuiranom procesu pokušaja i pogrešaka, „učeci kroz rad“ kroz dvije dimenzije: putem višestrukih razina (individualna, grupna, institucionalna) te putem različitih funkcija.
5. Suptilna kontrola – iako su timovi velikom mjerom samostalni, menadžment prevenira nestabilnost, dvosmislenost i tenziju, dok se istovremeno vrši kontrola putem pritiska vršnjaka (engl. Peer pressure).
6. Organizacijski transfer učenja - sudionici prenose znanje ostalima izvan grupe, stvarajući uvjete za nove projekte i dajući zadatke pojedincima. Znanje se prenosi i organizacijski na način da se aktivnosti projekta pretvaraju u standardnu praksu (Hidalgo, 2019).

Glavni razlog zastupljenosti Scruma jest jednostavnost principa funkcioniranja, odnosno u mogućnosti njegove primjene na bilo kojem tipu projekta ili, ukoliko je riječ o cijelom poduzeću, politici poslovanja poduzeća i njegovom globalnom radu (Stellman i Greene, 2017).

Scrum nije proces, tehnika niti metoda, već se za njega može reći da je svojevrsan okvir za razvoj, isporuku i održavanje složenih proizvoda, a unutar tog okvira omogućeno je rješavanje kompleksnog problema, za što je potrebna prilagodljivost, učinkovitost i kreativnost. U okviru Scruma ne postoje metodološke upute za ostvarivanje ciljeva već je riječ o skupu vrijednosti, načela i aktivnosti koja se provode u poduzeću (Rubin, 2013). Vrijednosti Scruma su predanost, hrabrost, otvorenost, usredotočenost i poštovanje članova projektnog tima. Scrum je jednostavan, a njegov okvir je namjerno nepotpun te su njime definirani isključivo dijelovi potrebni za implementiranje teorije Scruma. Scrum je izgrađen na kolektivnoj inteligenciji osoba koje se njime koriste. Umjesto da osobama koje ga koriste

daju detaljne upute, pravila Scruma usmjeravaju njihove odnose i interakcije (Schwaber i Sutherland, 2020, str. 3).

Unutar okvira Scruma omogućena je upotreba raznih procesa, tehnika i metoda. Scrum se ili nadograđuje na postojeće prakse ili ih pak čini nepotrebnima. Scrum iznosi na vidjelo relativnu učinkovitost trenutnog načina upravljanja, okoline i tehnika rada, kako bi se oni mogli poboljšati.

Prema Schwaberu i Sutherlandu (2020, str. 3), „za Scrum je potreban Scrum Master koji potiče okruženje u kojem:

1. Product Owner uz pomoć Product Backloga definira slijed rada na određenom kompleksnom problemu.
2. Scrum tim tijekom Srinta odrađuje dio posla i time stvara Vrijednost (engl. Increment).
3. Scrum tim i dionici (engl. stakeholders) pregledavaju rezultate rada i prilagođavaju se za idući Sprint.“
4. Postupak se ponavlja.

Scrum prema Schwaberu i Sutherlandu (2020), čine artefakti koji se mogu se svrstati u tri skupine koje istovremeno predstavljaju glavne dijelove Scrum metode:

1. Product Backlog – predstavlja popis svih zadataka i svih predmeta koji se moraju izvršiti. Održavanje tog popisa je zadatak menadžera ili vlasnika proizvoda. Product Backlog sadrži potpune ili nepotpune želje vlasnika proizvoda (specifikacije) i korisnička iskustva koja pomažu razumijevanju načina izrade softverskog rješenja.
2. Sprint Backlog – osnovni popis svih korisničkih priča, zadataka i popravaka grešaka u sustavu, za koje je predviđeno da se naprave u određenom vremenskom periodu. Razvojni tim i vlasnici proizvoda održavanju sastanke čija je tema ono što se treba ili želi učiniti u sljedećem vremenskom razdoblju.
3. Inkrement – predstavlja iskoristiv proizvod ili značajku koja je spremna za izlazak nakon predodređenog vremenskog perioda.

Teorija Scruma počiva na empirizmu i *lean thinking* principu. Empirizam se bazira na tome da znanje proizlazi iz iskustva, a odluke se donose temeljem opažanja. *Lean thinking* smanjuje i/ili uklanja nepotrebne viškove i fokusira se na ono što je najvažnije.

Pristup u Scrumu je iterativan i inkrementalan (postupan), a „cilj je optimiziranje predvidljivosti i kontrole rizika, te angažiranje skupine individualaca koji zajedno imaju sve potrebne vještine i stručno znanje kako bi odradili posao, koji kasnije međusobno dijele ili stječu takve vještine, ovisno o potrebi“ (Schwaber i Sutherland, 2020, str. 3). Članovima Scrum tima omogućena je prilagodba klijentovim željama i potrebama, dok istovremeno vrše isporuku kvalitetnog proizvoda. Za Scrum se može reći da ima tri osnovne karakteristike: lagan je, jednostavno ga je razumjeti no teško ga je savladati (Schwaber i Sutherland, 2011).

3.1. Scrum tim

Za Scrum tim se može reći da je temeljna jedinica Scruma. „Scrum tim je mala grupa ljudi, a sastoji se od jednog Scrum Mastera, jednog Product Ownera i od Developera. Unutar samog Scrum tima nema pod-timova, ne postoji hijerarhija, već je to kohezivna grupa profesionalaca koja je u svakom trenutku usredotočena na jedan cilj, a to je Cilj proizvoda (engl. *Product Goal*)“ (Schwaber i Sutherland, 2011, str 5). Scrum timovi su međufunkcionalni (engl. *cross-functional*), dakle članovi tima posjeduju sve vještine koje su nužne za stvaranje proizvoda u svakom Sprintu. Scrum timovi su samoupravljački (engl. *self-managed*), dakle samostalno unutar tima donose odluku tko što radi, kada i kako radi.

Uobičajeni broj članova Scrum tima jest 10 ili manje, a osnovna ideja je da tim bude dovoljno malen da zadrži okretnost i fleksibilnost, a s druge strane da bude dovoljno velik kako bi se dovršio zadani zadatak unutar Sprints. Kroz iskustvo je uočeno da je unutar manjih timova komunikacija bolja, a produktivnost veća. U situaciji u kojoj Scrum tim postane prevelik, preporuča se reorganizacija tima u više kohezivnih Scrum timova, koji bi bili usredotočeni na isti proizvod, a svi timovi imali bi isti cilj proizvoda, Product Backlog i Product Ownera (Schwaber i Sutherland, 2020, str 5). Bilo bi idealno da 7 članova tima (službeni raspon 3 do 9) boravi u istoj prostoriji koja je zaštićena od vanjskih ometajućih podražaja (Saradha, 2016).

Sve aktivnosti u svezi rada na proizvodu odgovornost su Scrum tima: suradnja s dionicima, uporaba, provjera, održavanje, eksperimentiranje, istraživanje i razvoj itd. Organizacija strukturira Scrum tim i daje mu dovoljne ovlasti kako bi upravljao vlastitim radom. Rad održivim tempom u Sprintovima dovodi do povećane usredotočenosti i dosljednosti Scrum tima, a za stvaranje upotrebljivog, korisnog Inkrementa odgovoran je cijeli tim. Postoje tri specifične zone odgovornosti unutar Scrum tima: Developeri, Product Owner i Scrum Master. Svaki pojedini član tima ima vlastitu ulogu u upravljanju projektom

(Ingason i Jonasson, 2014), a svaka od tih uloga je neophodna za učinkovito odvijanje procesa (Haukur i Helgi 2014).

Developeri

Developeri (razvojni inženjeri) su usredotočeni na stvaranje upotrebljivog, korisnog Inkrementa u svim aspektima, u svakom Sprintu. Za developere je važno da posjeduju skup specifičnih vještina koje se često značajno razlikuju ovisno o domeni posla i vrlo često su širokog spektra.

Kada je riječ o Developerima, vrijede navedena pravila (Schwaber i Sutherland, 2011, str 5):

- Izrada plana Sprinta i Sprint Backloga je odgovornost Developera.
- Odgovornost Developera jest i osiguranje kvalitete - pridržavanje definicije učinjenog. (engl. *Definition of Done*).
- Jedna od odgovornosti Developera jest i svakodnevna prilagodba plana u skladu s Ciljem Sprinta (engl. *Sprint Goal*) odnosno procjena napretka prema cilju sprinta i planiranje budućih aktivnosti (Rubin, 2013).

Product Owner

Odgovornost Product Owner jest postizanje maksimalne vrijednosti proizvoda kojeg stvara Scrum tim, a način na koji to čini razlikuje se ovisno o pojedincima, timovima i organizacijama (Schwaber i Sutherland, 2011, str 6). Uloga Product Ownera je poznavati prioritete i želje svih dionika (Stakeholdera) kako bi mogao Developerima na odgovarajući način prenijeti njihove zadatke te kakav proizvod trebaju razviti (Rubin, 2013).

Odgovornost Product Ownera jest postizanje maksimalnog povrata od uloženog na način da identificira zahtjeve projekta nakon čega ih transformira u Product backlog. „ Jedan od najvažnijih zadataka Product Ownera jest razvijanje cilja proizvoda, informiranje o istome te upravljanje product backlogom. Upravljanje Product backlogom predstavlja jasno definiranje stavki na Product backlogu te upravljanje redoslijedom stavki, kako bi se omogućilo postizanje ciljeva na najučinkovitiji mogući način. Uzimajući u obzir Product backlog, Product Owner donosi odluku koji će zadatak imati prioritet u sljedećem sprintu, a revizija Product backloga provodi se tijekom cjelokupnog trajanja projekta. Product Owner je zadužen za optimiranje vrijednosti posla kojeg Development Tim odrađuje te je njegov

zadatak osigurati da je Product backlog dostupan, transparentan i jasan Developerima koji će prema njemu razvijati proizvod“ (Schwaber, Sutherland 2017).

Product Owner može samostalno obavljati navedene zadatke ili može na druge delegirati dio zaduženja. Neovisno o tome što odabere, on sam ostaje kao osoba koja je odgovorna za navedeno. Za uspjeh Product ownera neophodno je da cijela organizacija poštuje odluke koje oni donesu. Odluke Product ownera vidljive su kroz sadržaj i redoslijed stavki Product Backloga te kroz Inkrement koji je podložan pregledu tijekom Sprint Reviewa.

Product Owner nije odbor već jedan pojedinac koji može zastupati potrebe mnogih dionika putem Product Backloga. Svatko tko želi promijeniti Product Backlog, ima mogućnost to učiniti na način da pokuša u to uvjeriti Product Ownera (Schwaber i Sutherland, 2011, str 6).

Za ulogu Product Ownera može se reći da je jedna je od najvažnijih, ali i najtežih u Scrum procesu (Haukur i Helgi, 2014). Težina uloge Product Ownera leži u tome što je on odgovoran za dobit ili gubitak koji je ostvaren projektom, a u slučaju da proizvod koji Scrum tim razvija nema komercijalnu vrijednost, uloga Product Ownera je postizanje maksimalne moguće dodane vrijednost za projekt ili poduzeće (npr. poboljšanje poslovanje, zadovoljstvo korisnika i slično). Product Owner nema nikakvu ovlast nad timom, iako s njime blisko surađuje i komunicira, već pruža podršku timu i odgovoran je za donošenje važnih, velikih odluka kao i za uspjeh samog projekta (Haukur i Helgi, 2014). Jednako tako, Product Owner „mora biti u stanju postići ravnotežu između različitih zahtjeva proizvoda, realne mogućnosti ostvarivanja tih zahtjeva te postizanja vlastitog zadovoljstva tijekom procesa“ (Ingason i Jonasson, 2014).

Za Product Ownera bilo bi poželjno da posjeduje određene osobine i kvalitete. Product Owner trebao bi posjedovati stručnost i kompetencije te biti uvijek na raspolaganju timu. Vrlo važna osobina su i dobre komunikacijske i pregovaračke vještine te autoritativan stav. Prema Ingason i Jonasson (2014), „osnovni zadatak Product Ownera je upravljanje očekivanjima različitih dionika (stakeholdera)“.

Scrum Master

Odgovornost Scrum Mastera jest uspostavljanje Scruma na način da on svim ostalim sudionicima pomaže razumjeti teoriju i praksu, vrijednosti i pravila Scruma, kako unutar samog Scrum tima tako i u organizaciji. Za njega možemo reći da snosi odgovornost za

učinkovitost tima, kao i za promoviranje Scrum okvira i podršku (Schwaber i Sutherland, 2011, str 6). Uspjeh Scrum Mastera je od ključne važnosti za uspjeh Scruma, ali njegova se uloga najčešće pogrešno interpretira. Scrum Master nema leadersku niti tradicionalnu funkciju, već je njegova uloga da poboljšava uporabu Scruma kroz treninge, olakšavanje i brzo uklanjanje svih prepreka koje ometaju tim u ostvarenju rezultata (Keith, 2010).

Scrum Master doprinosi Scrum timu na različite načine:

- pruža pomoć članovima tima kako bi postali bolji u samostalnom upravljanju i međufunkcionalnosti,
- potiče tim da se fokusira na stvaranje Inkremenata velike vrijednosti, što je konzistentno s Definicijom Gotovog,
- sudjeluje u uklanjanju prepreka koje mogu ometati napredak tima,
- odgovoran je za to da se svi događaji u Scrumu održe, da se to dogodi unutar zadanog okvira vremena te da ishod tih događaja bude produktivan i koristan (Schwaber i Sutherland, 2011, str 6).

Scrum Master također:

- sudjeluje u pronalasku tehnika za efikasno definiranje Cilja proizvoda i pomaže Product Owneru u upravljanju Product Backlogom,
- pomaže Scrum timu u shvaćanju potrebe za jasnim i sažetim stavkama na Product Backlogu,
- pomaže u shvaćanju i prakticiranju agilnosti,
- potiče empirijski pristup u planiranju proizvoda u kompleksnom okruženju,
- olakšava suradnju dionika na njihov zahtjev ili po potrebi (Schwaber i Sutherland, 2011, str 7).

Scrum Master pruža usluge organizaciji na različite načine, primjerice:

- vodi organizaciju, te u ulozi coacha (trenera) pomaže organizaciji u prihvaćanju i implementiranju Scruma,
- planira i savjetuje ostale o načinima implementiranja Scruma unutar organizacije,
- uvodi promjene s cilje povećanja produktivnosti Scruma,
- surađuje s drugim Scrum Masterima kako bi se povećala učinkovitost primjene Scruma,

- uklanja moguće prepreke između dionika i Scrum timova (Schwaber i Sutherland, 2011, str 7).

Poželjne karakteristike koje bi trebao posjedovati Scrum Master su (Rubin, 2013.):

- posjedovanje kompetencija, odnosno odlično poznavanje Scruma,
- vještina adekvatnog postavljanja pitanja s ciljem poticanja ostalih sudionika na razmišljanje,
- strpljivost,
- sposobnost kvalitetne suradnje s mnogim sudionicima procesa,
- poticanje sinergijske suradnje u timu,
- zaštita tima od nepovoljnih utjecaja izvan tima,
- transparentnost u timu i izvan njega,
- odsustvo skrivenih namjera samog Scrum Mastera.

3.2. Tijek Scruma

Sprint

Za Sprint se može reći da je srž Scruma, pri čemu dolazi do pretvorbe ideja u vrijednost. Sprintevi su događaji čija je duljina fiksna, traju mjesec dana ili manje, a njihova svrha jest postizanje dosljednosti. S novim Sprintom se započinje neposredno nakon završetka prethodnog Srinta, a poželjno je da je tijekom razvoja proizvoda trajanje Sprinteva jednako. Unutar Srinta odvijaju se svi zadaci i događaji koji su nužni za ostvarenje Cilja proizvoda – Planiranje sprinta, Dnevni sprint, Pregled sprinta i Retrospektiva (Schwaber i Sutherland, 2011).

Za vrijeme trajanja Srinta vrijede sljedeća pravila:

- nema uvođenja promjena koje bi mogle dovesti do ugrožavanja Cilja Srinta,
- promjena ne smije dovesti do narušavanja kvalitete,
- ukoliko postoji potreba, razrađuje se Product Backlog,
- ulaskom u detalje i dobivanjem novih informacija, postoji mogućnost razjašnjavanja opsega posla i pregovaranja s Product Ownerom.

Sprint je moguće otkazati prije isteka vremena, no jedina osoba koja je za to ovlaštena je Product Owner, koji pak to možete učiniti ako na njega vrše pritisak drugi dionici procesa.

Potencijalni razlog za otkazivanje Sprinta jest prestanak aktualnosti (važnosti) cilja, do čega može doći uslijed promjene strateškog smjera poduzeća, promjene u tehnologijama i tržištu itd. S obzirom da su Sprintevi relativno kratki, njihovo otkazivanje nije čest slučaj i nepoželjno je jer dolazi do trošenja resursa.

Planiranje Sprinta

Tijekom Planiranja Sprinta priprema se posao kojeg je potrebno odraditi unutar Sprinta, a to je nužno jer Product Backlog sadrži više zahtjeva nego što tim može odraditi tijekom jednog sprinta. Rezultat planiranja jest zajednički plan odnosno odluka koje zahtjeve Scrum tim može odraditi u nadolazećem periodu. Planiranje sprinta je zajednički rezultat rada cijelog Scrum tima i ono ne traje dulje od osam sati za sprint jednomjesečnog trajanja, a za sprinteve kraćeg trajanja i njegovo trajanje je kraće. Uloga Product Ownera jest osiguravanje spremnosti sudionika na razgovor o najvažnijim stavkama Product Backloga (Schwaber i Sutherland, 2011, str 8).

Planiranje sprinta obuhvaća sljedeće teme:

1) Koja je vrijednost ovog Sprinta?

Product Owner daje prijedlog na koje sve načine proizvod može povećati svoju vrijednost i korisnost tijekom aktualnog Sprinta. Suradnjom cijelog tima dolazi do definiranja Cilja Sprinta koji objašnjava iz kojeg je razloga Sprint vrijedan dionicima. Prije kraja Planiranja sprinta, nužno je da je Cilj Sprinta definiran.

2) Što može biti učinjeno u ovom Sprintu?

Developeri u suradnji s Product Ownerom vrše izbor stavki s Product Backloga koje će biti uključene u trenutni Sprint. Navedene stavke tijekom Planiranja Sprinta tim mozet razraditi kako bi se poboljšalo razumijevanje i pouzdanost. Odluka o tome što sve može biti učinjeno tijekom sprinta može predstavljati izazov, a sposobnost tima da donese adekvatnu procjenu o opsegu posla koji će biti odrađen tijekom nadolazećeg sprinta temelji se na poznavanju Developera o njihovom dosadašnjem učinku i postignućima tima u prošlim sprintevima (Keith, 2010.). Jednako tako, što su Developeri više informirani o predstojećem vlastitom kapacitetu i Definiciji Gotovog, bit će sigurniji i točniji u prognoziranju Sprinta.

3) Na koji će se način obaviti izabrani posao?

Developeri planiraju postupke potrebne za stvaranje Inkrementa u skladu s Definicijom Gotovog za svaku pojedinačnu stavku Product Backloga. Često se navedene stavke razrađuju na manje, kraće stavke čije je trajanje jedan dan ili kraće, prema vlastitom nahođenju.

Prema Schwaber i Sutherland (2011, str 10). „za svaku odabranu stavku s Product Backloga Developeri planiraju rad potreban za stvaranje Inkrementa koji odgovara Definiciji Gotovog. To se često postiže razradom stavki s Product Backloga na manje radne stavke u trajanju od jednog dana ili kraće. Developeri to rade isključivo prema vlastitom nahođenju. Nitko im drugi ne govori kako pretvoriti stavke s Product Backloga u Inkremente vrijednosti.“ Cilj Sprinta zajedno sa odabranim stavkama Product Backloga i planom isporuke istih zajedno čine Sprint Backlog (Schwaber i Sutherland, 2011, str 10).

Dnevni Scrum

Dnevni Scrum provodi se s ciljem provjere napretka u ostvarenju Cilja Sprinta te u slučaju potrebe prilagođavanja Sprint Backloga. Riječ je o petnaestominutnom događaju u kojem sudjeluju Developeri te eventualno Product Owner ili Scrum master (u slučaju da se aktivno bave stavkama Sprint Backloga. Dnevni Scrum se odvija svakog radnog dana, u isto vrijeme na istom mjestu. Developerima je ostavljen izbor tehnike koju će koristiti za Dnevni Scrum, sve dok je on fokusiran na napredak u ostvarenju Cilja sprinta i dok je plan rada za slijedeći radni dan učinkovitost. Na taj način se poboljšava usredotočenost i samoupravljanje, postiže bolja komunikacija, olakšava otkrivanje prepreka, potiče brzo donošenje odluka i u konačnici smanjuje potreba za dugotrajnim sastancima (Schwaber i Sutherland, 2011, str. 9).

Pregled Sprinta

Pregled Sprinta jest radni sastanak, odvija se s ciljem provjere ishoda Sprinta i određivanja eventualnih prilagodbi u budućnosti, pri čemu tim predstavlja rezultate rada i vodi se rasprava o postignutom napretku u ostvarenju cilja. Drugim riječima, tijekom Pregleda Sprinta vrši se pregled postignutog i promjena, na temelju čega se suradnički odlučuje o tome što će se slijedeće učiniti i eventualno mijenjati Product Backlog. To je predzadnji događaj Sprinta nakon kojeg slijedi Retrospektiva, a za jednomjesečni sprint vremensko ograničenje Pregleda Sprinta iznosi maksimalno četiri sata (Schwaber i Sutherland, 2011, str 9).

Retrospektiva

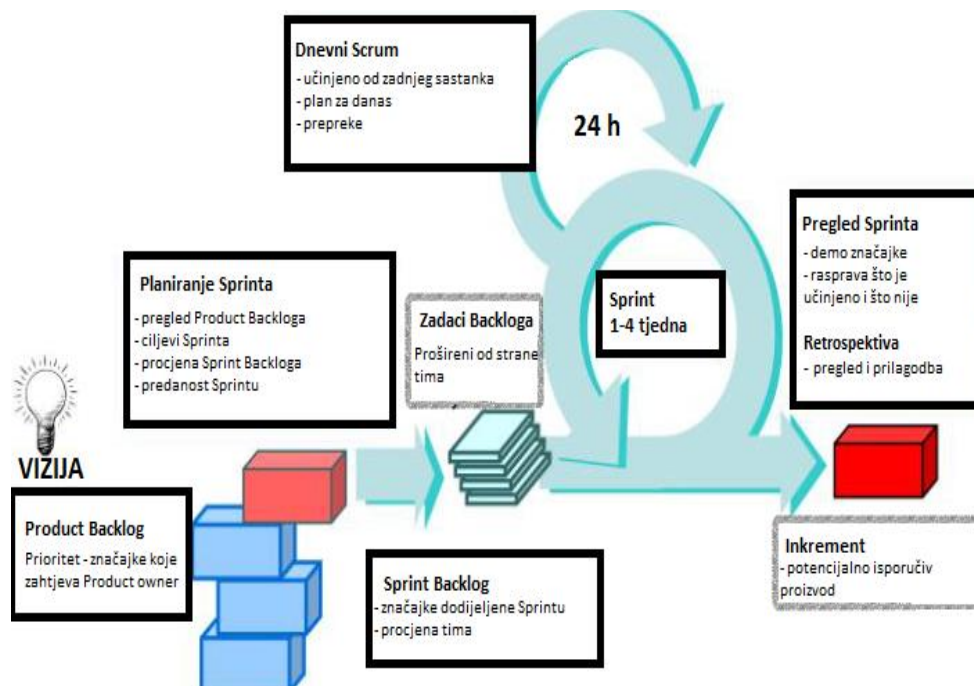
Retrospektiva se odvija s ciljem planiranja načina povećanja kvalitete i učinkovitosti. Ona je posljednji korak Srinta, te njeno vremensko ograničenje iznosi maksimalno tri sata ukoliko je riječ o jednomjesečnom Srintu.

Retrospektiva obuhvaća:

- pregled posljednjeg Srinta – pojedinci, međuodnosi, procesi, alati, Definicija Gotovog
- identifikacija pretpostavki zbog kojih je došlo do skretanja s planiranog puta te istraživanje njihovog podrijetla
- rasprava o tome što je dobro učinjeno tijekom Srinta
- rasprava o problemima uočenim tijekom Srinta i načina rješavanja, ili pak uzroka zbog kojih nisu riješeni
- prepoznavanje promjena koje su bile najkorisnije za povećanje učinkovitosti i prema potrebi dodavanje istih na Srint Backlog tijekom sljedećeg Srinta (Schwaber i Sutherland, 2011, str 9).

Za Retrospektivu Srinta može se reći da je to prilika za poboljšanje procesa, fokusirana na to da dobar Scrum tim postane izvrstan (Rubin, 2013.). Također, to je najvažnija, no najčešće zanemarena praksa koju poduzeća provode (Keith, 2010).

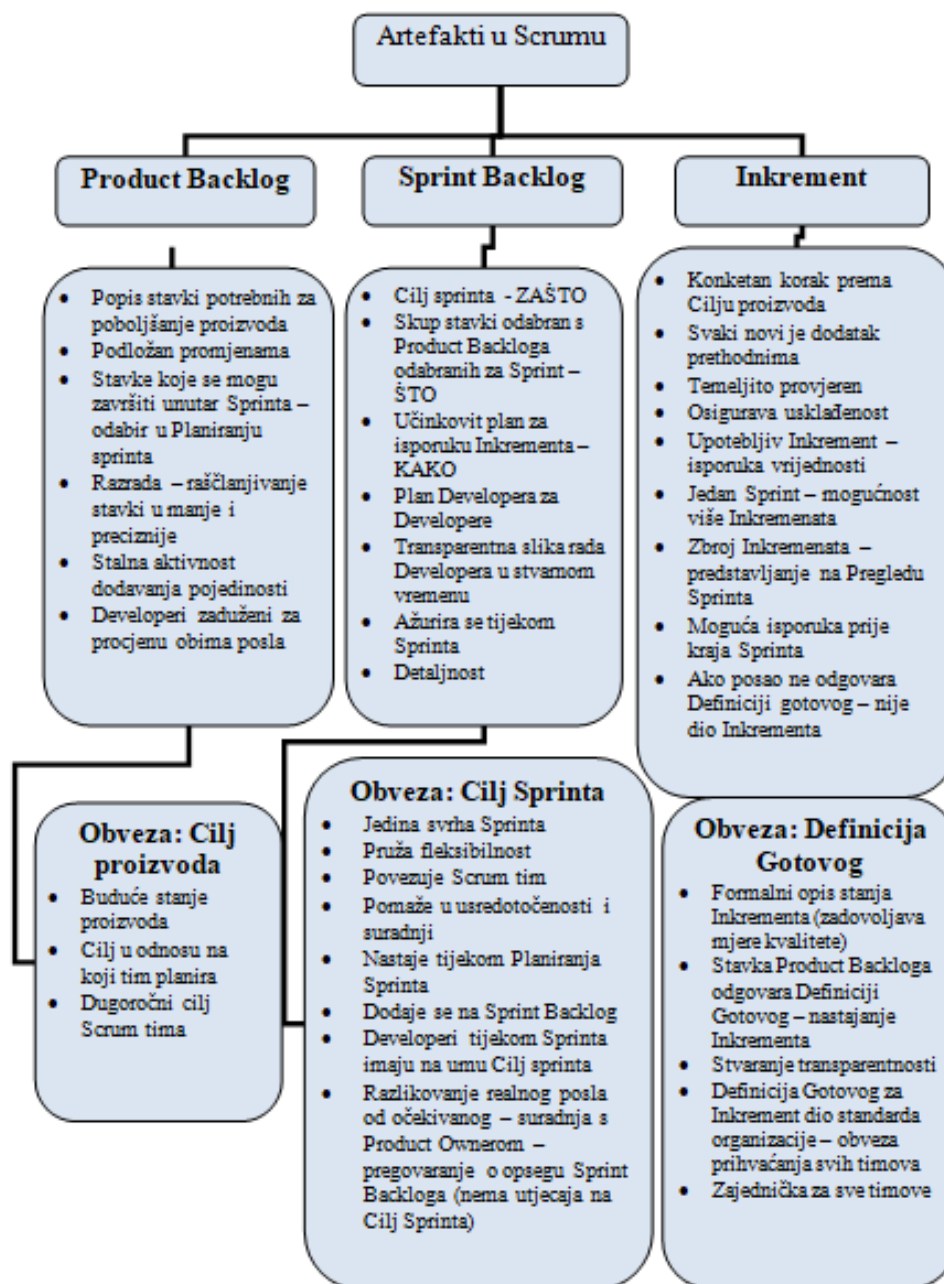
Tijek Scrum-a prikazan je na slici 1.



Slika 1. Tijek Scrum-a (Valpadasu i sur.,2020)

3.3. Artefakti u Scrumu

Artefakti u Scrumu odnose se na posao ili vrijednost, a osmišljeni su s ciljem maksimiziranja transparentnosti podataka. Uz svaki artefakt Scruma veže se jedna obveza Scrum tima, a cilj je osiguravanje pružanja informacija kojima se povećava transparentnost i usredotočenost i stvara mogućnost mjerenja napretka. Artefakti Scrum-a prikazani su na Slici 2.



Slika 2. Artefakti u Scrum-u (autorski rad, prema Schwaber i Sutherland, 2011).

3.4. Prednosti i nedostaci Scrum metodologije

S obzirom da je SCRUM metoda najrasprostranjenija agilna metoda, može se pretpostaviti da su njena obilježja uglavnom pozitivna. Kao najvažnije prednosti ističu se njena fleksibilnost i mogućnost prilagođavanja potrebama klijenta.

Prednosti SCRUM metodologije su slijedeće (Adell, 2013):

- kvalitetnije i brže napredovanje kao posljedica lakšeg uočavanja grešaka i njihovog otklanjanja,
- jasan i vidljiv razvoj projekta,
- redovni retrospektivni sastanci omogućuju lakšu kontrolu projekta,
- dobivanje povratne informacije naručitelja tijekom svakog novog sprinta poboljšava kvalitetu proizvoda i povećava zadovoljstvo dionika,
- olakšano uvođenje promjena uslijed kratkog trajanja sprinta i redovnog primanja informacija,
- svakodnevno održavanje sastanaka pridonosi transparentnosti među članovima tima, međusobnom pomaganju i izmjeni znanja te posljedično povećanju produktivnosti i zadovoljstva,
- kvalitetan proizvod lakše se isporučuje unutar zadanog roka,
- scrum funkcionira u bilo kojoj tehnologiji i industriji,
- minimaliziranje troškova u smislu procesa i upravljanja – rezultat brži i financijski isplativiji.

Iako su maksimalna uključenost naručitelja te njegova kontinuirana suradnja sa Scrum timom najčešće svrstane u pozitivne aspekte Scrum metode, ponekad mogu imati i negativan utjecaj. Učestale, konstantne promjene i često održavanje sastanaka ponekad produljuju razvojni proces (Highsmith i Cockburn, 2001). U situacijama kada naručitelj nije u stanju sam verbalizirati svoje zahtjeve, dolazi do povećanja ponavljanja procesa, smanjenja zadovoljstva i produljivanja roka isporuke. Važan čimbenik je veličina tima – provođenje Scrum metode u većim timovima nije nemoguće, ali je otežano.

Nedostaci Scrum metode su navedeni (Adell, 2013):

- nije definiran doseg projekta niti finalni datum isporuke proizvoda,
- u slučaju loše definiranosti zadatka, netočna je procjena troškova i potrebnog vremena i zadatak se može produljiti na nekoliko sprinteva,
- ako izostaje predanost članova tima, projekt neće biti dovršen ili će biti neuspješan,
- pogodan za manje, brže projekte – manji timovi,
- velika je važnost iskustva članova tima,

- Scrum Master mora imati povjerenje u tim – pretjerano stroga kontrola nad timom ima negativan učinak i dovodi do frustracije, pada morala i neuspjeha projekta,
- gubitak samo jednog člana tima uzrokuje veći zastoje u razvoju tima (mali broj članova tima).

4. Informacijski sustav u bankarstvu (IS)

Uporabom informacijske tehnologije omogućeno je korištenje i upravljanje informacijama te poticanje i unaprjeđenje poslovnih procesa s ciljem što djelotvornijeg ostvarivanja poslovnih ciljeva i postizanja konkurentske prednosti. Prema Hrvatskoj narodnoj banci (2013), informacijski sustav definira se kao sveobuhvatnost infrastrukture, organizacije, ljudi i postupaka kojima se vrši prikupljanje, obrada, generiranje i pohranjivanje informacija, kao i prijenos, prikaz, distribucija i raspolaganje njima.

U sektoru bankarstva, uslijed ubrzanog razvoja i sveprisutnosti informacijskih tehnologija unutar poslovnog okruženja, informacijski sustav ima poseban značaj (Hrvatska narodna banka, 2013):

- IS je jedan od presudnih faktora u poslovanju banke,
- IS povećava interakciju različitih sustava,
- IS definira temelj za razvijanje novih proizvoda i usluga te ima utjecaj na postizanje konkurentskih prednosti,
- IS pokreće reinženjering strateški važnih poslovnih procesa,
- IS dovodi do povećanja potrebe za novim investicijama u resurse informacijskog sustava,
- IS zahtijeva stjecanje i primjenu novih znanja iz područja struke (Hrvatska narodna banka, 2006).

Kako bi informacijski sustav bio kvalitetan, neophodno je da zadovolji sljedeća načela:

- IS jest model poslovne tehnologije nekog organizacijskog sustava,
- podaci su resurs poslovnog sustava,

- osnova razmatranja tijekom definiranja podsustava su poslovni procesi koji predstavljaju nepromjenjivi dio neke poslovne tehnologije,
- IS izgrađuje se pomoću integracije podsustava na osnovi zajedničkih podataka što se naziva modularnost,
- informacije vezane uz upravljanje i odlučivanje izvode se temeljem zbivanja na razini izvođenja (Klasić, Klarin, 2003).

Poslovni IS imaju direktan utjecaj na konkurentnost poslovanja, što se postiže na dva moguća načina:

- stvaranjem pozitivnog utjecaja na operativnu efikasnost poslovanja i/ili
- pokretanjem inovativnosti i promjenama u poslovanju u određenim okolnostima (Pejić Bach i sur., 2016).

4.1. Upravljanje informacijskim sustavom banke

Upravljanje IS-om banke podrazumijeva:

- Uspostavljanje primjerene organizacijske strukture (npr organizacijske jedinice za IT, za upravljanje rizikom itd)
- Uspostavljanje odgovarajućih funkcija i odbora (npr. voditelji)
- Uspostavljanje procesa upravljanja rizikom IS-a (procjena rizika) te uspostavljanje nadzora nad navedenim procesom. (Hrvatska narodna banka, 2006.)

Banka bi trebala definirati strategiju informacijskog sustava za koju je važno da bude usklađena s poslovnom strategijom banke, te da obuhvaća kratkoročne i dugoročne inicijative, pri čemu je važno uzeti u obzir slijedeće čimbenike:

- nove inicijative u poslovanju,
- promjene u organizaciji,
- razvoj tehnologije,
- regulatorne zahtjeve,
- potrebe za resursima i nadzorom,
- ograničenja (Hrvatska narodna banka, 2006).

4.2. Komponente informacijskog sustava banke

Arhitektura softverskog dijela IS-a e-bankarstva je troslojna, odnosno ima tri glavna dijela:

1. Front-end sustav (korisničko sučelje, eng. User interface)
2. Srednji dio sustava (eng. Middleware)
3. Back-end sustav – centralizirana obrada podataka u pozadini sustava (Al-Khatib, 2012):

Korisničko sučelje (eng. user interface, Front-end sustav) - predstavlja korisničko sučelje aplikacija, odnosno dio softverskog sustava IS banke kojeg korisnici vide i koriste tijekom autentikacije i uporabe financijskih usluga. Tijekom izrade korisničkog sustava najčešće se koriste programski jezici HTML, CSS, XHTML, JavaScript i PHP. U novije vrijeme je fokus na izradi sučelja gdje se posebna pažnja daje vizualnom izgledu sučelja, te na izradi sučelja za mobilne aplikacije (Zekić-Sušac, 2013).

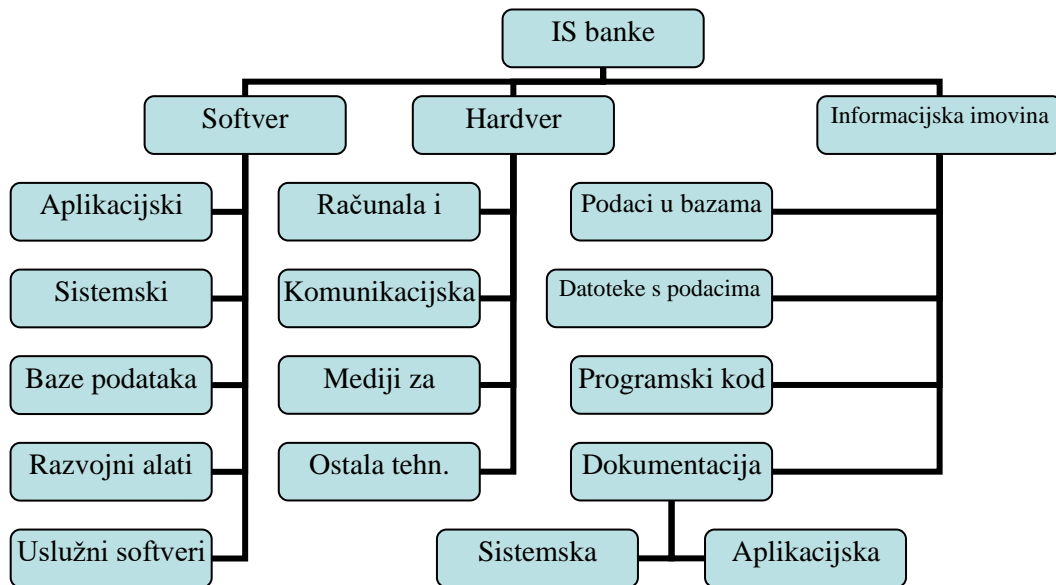
Smjernice za izradu front-end aplikacija usmjerene su na:

- Prilagodbu sučelja različitim dobnim skupinama, na način da bude atraktivno i jednostavno za korištenje i mladima koji se cijeli život koriste Internetom i mobilnim uređajima, kao i starijoj populaciji
- Pratiti preporuke dobrog dizajna i upotrebljivosti (eng. usability) aplikacije
- Koristiti sigurnosne protokole za zaštitu podataka i informirati korisnika o sigurnosti njegovih podataka (Zekić-Sušac, 2013).

Srednji dio sustava (eng. Middleware) - obuhvaća programe koji djeluju kao aplikacijski posrednik između više aplikacija, između aplikacije i baze podataka, ili između nekoliko baza podataka. Međusobna komunikacija aplikacija odvija se pomoću sustava za razmjenu poruka (eng. messaging – GET i PUT poruke, poruke koje služe za učitavanje podataka koje korisnik unese i za prijenos između aplikacija) (Zekić-Sušac, 2013).

Back-end dio sustava jest centralizirani računalni kapacitet banke, kojem se pristupa pomoću različitih kanala banke. Vršiti se prikupljanje transakcija u različitim oblicima, te aplikacije back – end sustava prevode zahjeve u oblik koji će stroj razumjeti i izvršiti. Za provođenje ovih funkcija potrebna je velika računalna snaga, a neophodne su dodatne sigurnosne mjere i tehnike programiranja na naprednoj razini. Baze podataka čine jezgru

back-end sustava (Zekić-Sušac, 2013). Na slici 3 prikazane su komponente informacijskog sustava banke.



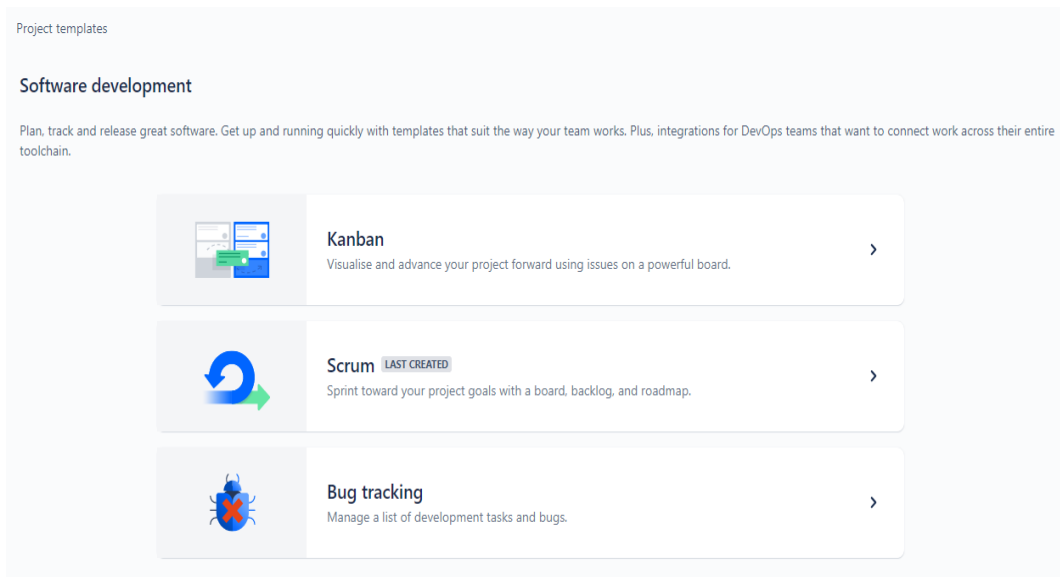
Slika 3. Komponente IS banke (autorski rad, prema Odluci o primjerenom upravljanju informacijskim sustavom, Hrvatska Narodna Banka, 2010).

5. Primjena SCRUM metode u bankarstvu

Glavni cilj ovog poglavlja je prikazati način na koji Scrum metoda funkcionira unutar razvojnog tima. Prikazat će se interakcija između Product Ownera, Scrum Mastera i razvojnog tima. Alat koji će biti prikazan je Jira, čije značajke omogućuju upravljanje projektima pa tako i projektima razvoja informacijskih sustava i drugih programskih rješenja na temelju agilnog (SCRUM) pristupa. Funkcionalnosti alata će se prikazati na pojednostavljenom primjeru razvoja mobilne aplikacije u bankarstvu. Za potrebe prikaza, projekt u alatu Jira biti će nazvan Banka 1.

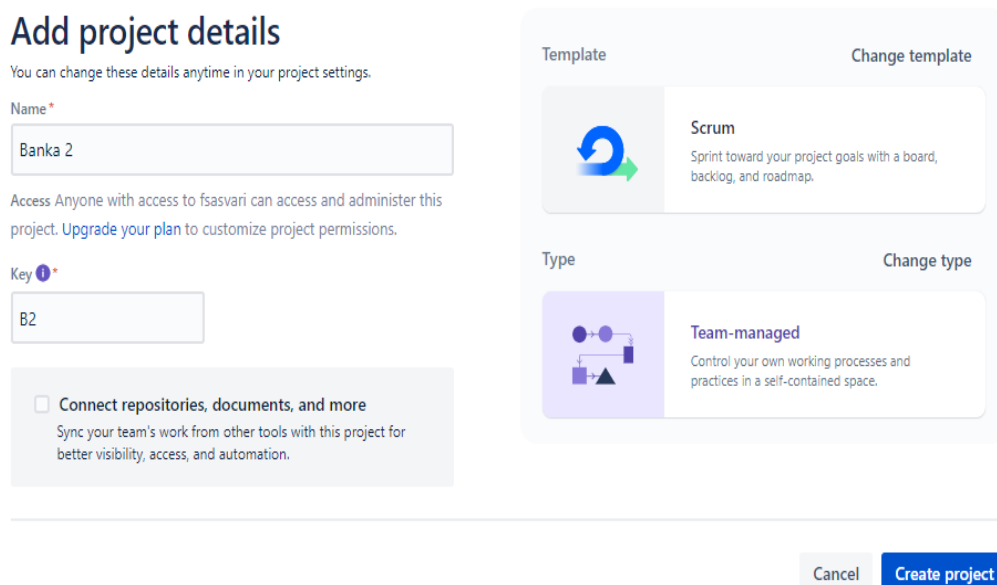
Zadatke definiraju Product Owner i Scrum Master zajedno s razvojnim timom tijekom planiranja sprinta. Nastoji se postići približno jednaka raspodjela developera po ticketima (story pointsima) kroz trenutni sprint.

Na slici 4 prikazan je izbor vrste prikaza boarda i backloga, u ovom slučaju odabire se Scrum.



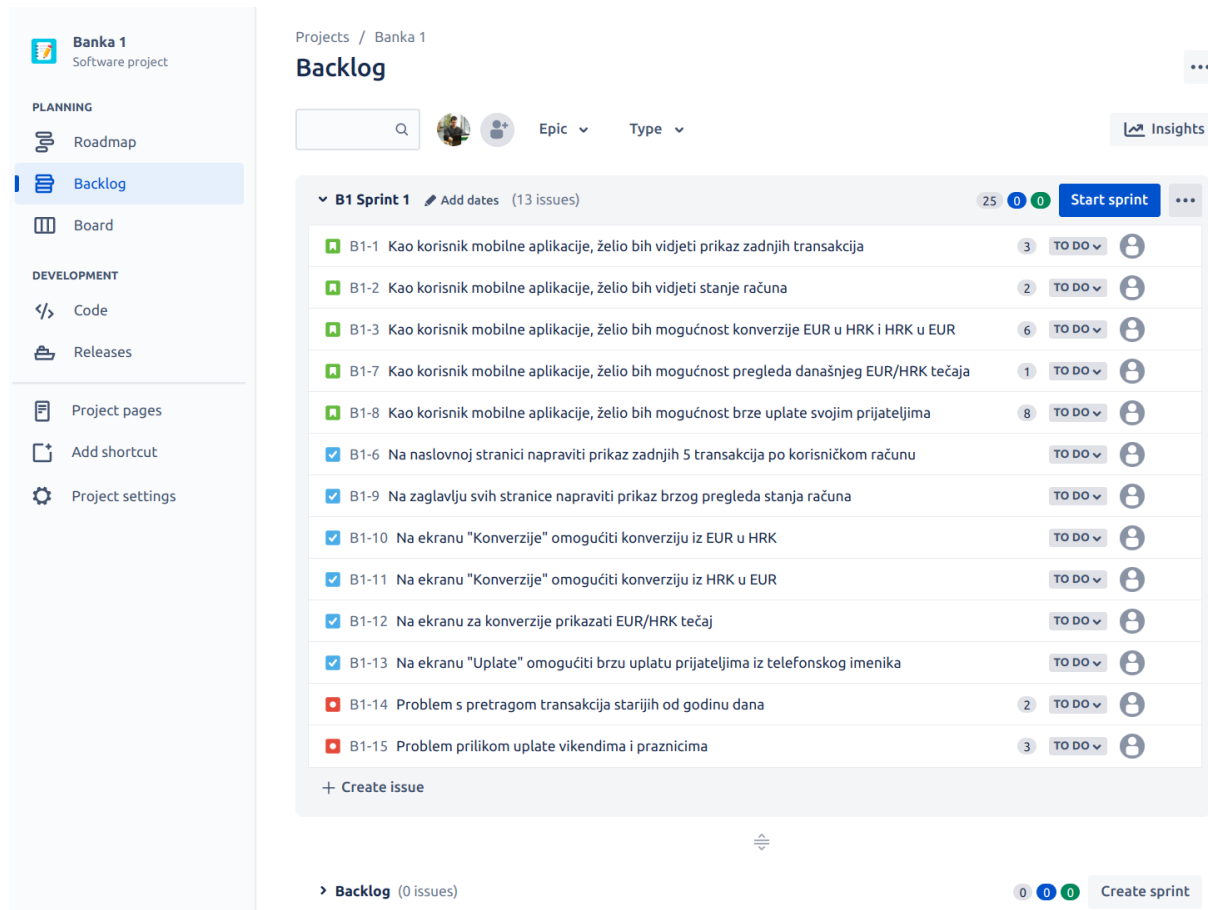
Slika 4. Kreiranje projekta – prvi korak

Zatim se unose naziv i ključ projekta („key“), oboje trebaju biti jedinstveni (slika 5).



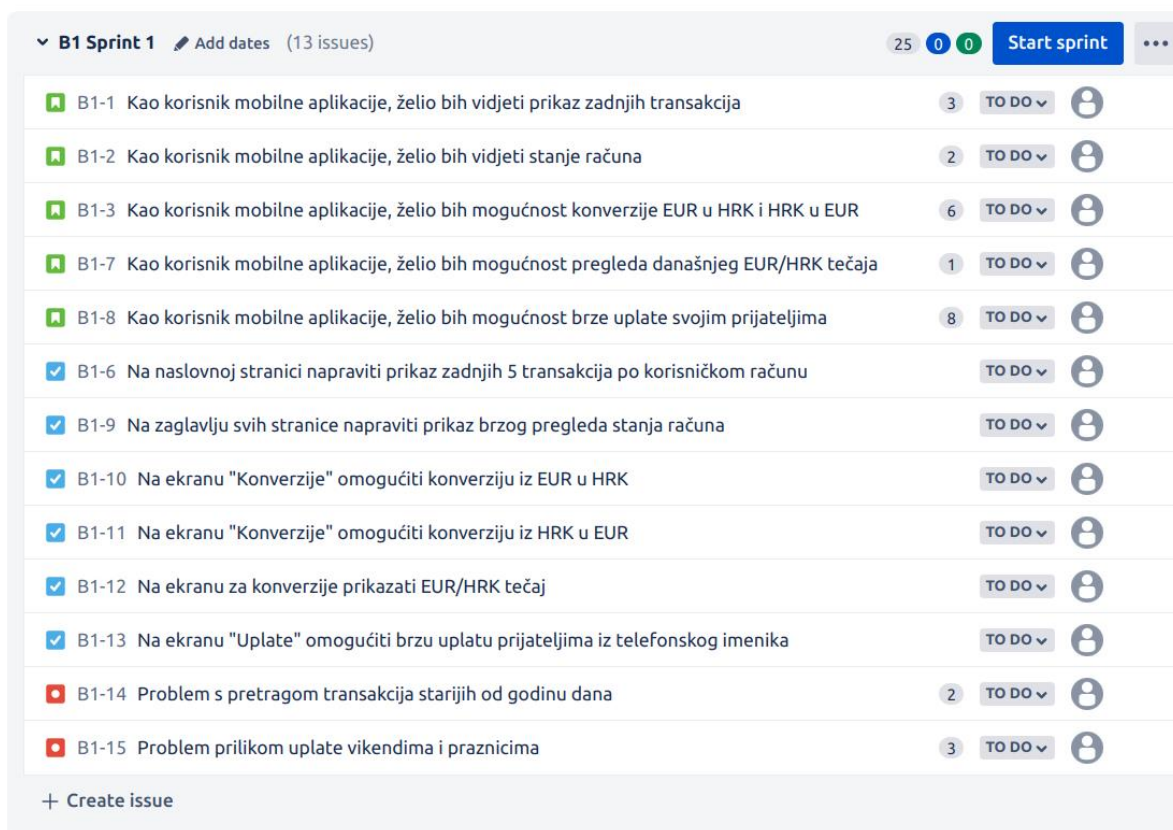
Slika 5. Kreiranje projekta – drugi korak

Na slici 6. prikazano je korisničko sučelje s nekoliko elemenata projekta, koji će biti posebno opisani, radi lakšeg razumijevanja.



Slika 6. Osnovno sučelje u alatu Jira (izvor: autor)

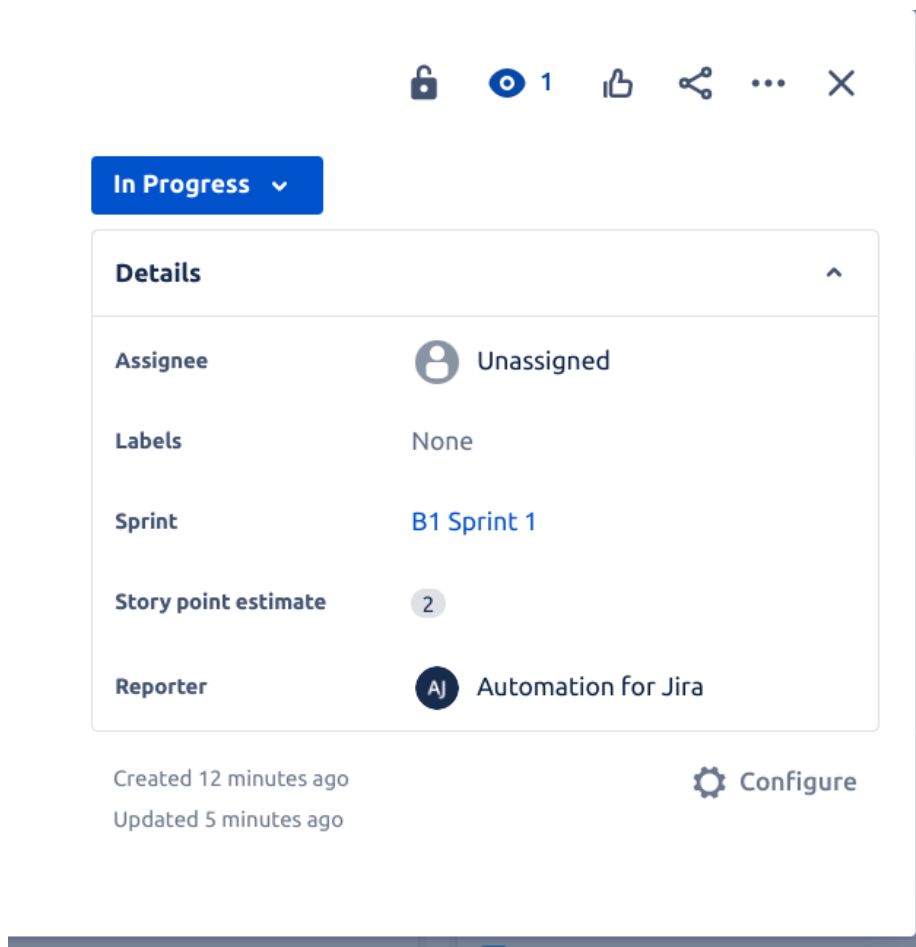
Kada Product Owner unese korisničke priče (User stories) u sučelje alata Jira, prikazuje se sučelje vidljivo na slici 7. Iz slike 5 vidljivo je nekoliko značajki. Svaka korisnička priča automatski je obilježena odnosno numerirana skraćenicom projekta i rednim brojem priče (sa svakom novom pričom, redni broj se povećava). Nije nužno da bude unešena kompletna korisnička priča već je dovoljno unijeti kratak i razumljiv naslov, a duljina opisa varira - važno je da je priča razumljiva i jasna razvojnom timu. Na slici 7 vidljivo je da uz svaku korisničku priču prikazana i faza u kojoj se trenutno nalazi, te osoba kojoj je dodijeljena.



Slika 7. Sučelje Backloga (izvor: autor)

Na slici 8 prikazano je nekoliko značajki sučelja korisničke priče:

1. Dodijeljena osoba (eng. Assignee) – član tima kojem je dodijeljena korisnička priča.
2. Oznake (eng. Labels) – omogućuju lakšu navigaciju po korisničkim pričama u sučelju te olakšavaju filtriranje korisničkih priča.
3. *Sprint* - korisnička priča se stavlja u *Sprint* kada to učini Product Owner
4. Procjena vremena u obliku bodova korisničkih priča (eng. Story point estimate) – može se i ne mora koristiti.
5. Izvjestitelj (eng. Reporter) – član tima koji je kreirao korisničku priču ili zadatak.
6. Automatizacija (eng. Automation) – automatsko dodijeljivanje korisničke priče drugom članu tima nakon što se pomakne u određeni stupanj azvoja.
7. Opis (eng. Description) – mogućnost unosa detaljnijeg objašnjenja korisničke priče kako bi bila jasnija i razumljivija svim članovima tima
8. Komentari (eng. Comments) - Primarna komunikacija svih dijelova *Scrum* metode. Product Owner ima mogućnost davanja mišljenja, prijedloga mogućih rješenja ili odgovaranja na pitanja ostalih članova razvojnog tima.



Slika 8. Prikaz sučelja korisničke priče (izvor: autor)

Elementi prikaza sučelja na slici 9 su sljedeći:

1. Naziv Sprinta (eng. Sprint name)
2. Trajanje Sprinta (eng. Duration)
3. Datum početka (eng. Start Date) i završetka Sprinta (eng. End Date)
4. Cilj Sprinta (eng. Sprint Goal).

Start Sprint

13 issues will be included in this sprint.

Sprint name *

Duration *

Start date *

End date *

Sprint goal

- završiti značajku konverzije EUR/HRK
- završiti prikaz zadnjih transakcija
- ispravak grešaka

Start **Cancel**

Slika 9. Prikaz sučelja postavki Sprinta (izvor: autor)

Na slici 10 prikazana su četiri stupca koja predstavljaju četiri faze

1. Za razvoj - početni stupac koji sadrži priče za koje je odlučeno da će ići u Sprint, odnosno stupac iz kojeg vuku sve korisničke priče u stupce s njegove desne strane.
2. Trenutno u razvoju – prikaz korisničkih priča koje su trenutno u razvoju
3. U testiranju – prikaz korisničkih priča koje su u procesu testiranja
5. Dovođeno - nakon što korisnička priča, to jest značajka dovršena („puštena van“), ona se premješta u ovaj stupac.

B1 Sprint 1

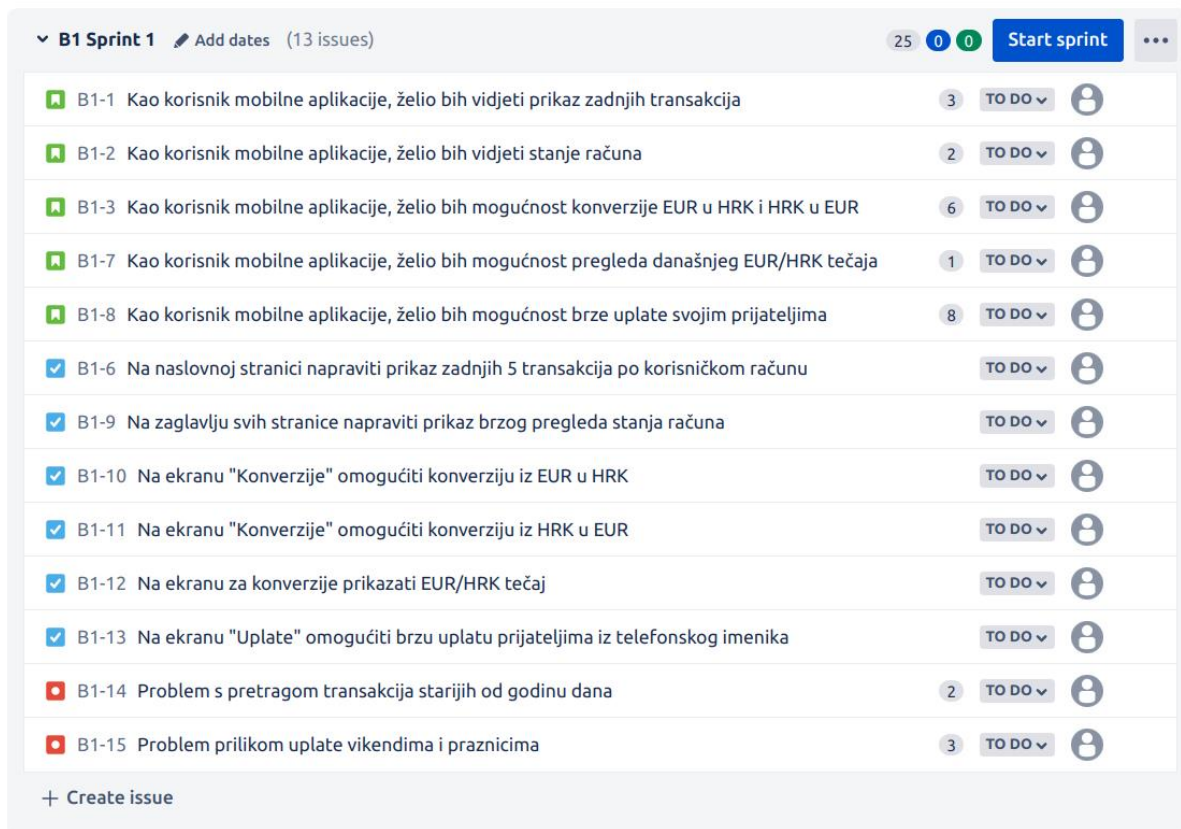
- završiti značajku konverzije EUR/HRK - završiti prikaz zadnjih transakcija - ispravak grešaka

The screenshot displays a Scrum board for 'B1 Sprint 1' with the following columns and tasks:

- ZA RAZVOJ 4 ISSUES**
 - Na ekranu za konverzije prikazati EUR/HRK tečaj (KONVERZIJA EUR/HRK) - B1-12 (checked)
 - Kao korisnik mobilne aplikacije, želio bih mogućnost pregleda današnjeg EUR/HRK tečaja (KONVERZIJA EUR/HRK) - B1-7 (1)
 - Kao korisnik mobilne aplikacije, želio bih mogućnost brze uplate svojim prijateljima (TRANSAKCIJE) - B1-8 (8)
 - Na ekranu "Uplate" omogućiti brzu uplatu prijateljima iz telefonskog imenika (TRANSAKCIJE) - B1-13 (checked)
- TRENUTNO U RAZVOJU 4 ISSUES**
 - Problem s pretragom transakcija starijih od godinu dana (B1-14) - 2
 - Kao korisnik mobilne aplikacije, želio bih mogućnost konverzije EUR u HRK i HRK u EUR (KONVERZIJA EUR/HRK) - B1-3 (6)
 - Na ekranu "Konverzije" omogućiti konverziju iz EUR u HRK (KONVERZIJA EUR/HRK) - B1-10 (checked)
 - Na ekranu "Konverzije" omogućiti konverziju iz HRK u EUR (KONVERZIJA EUR/HRK) - B1-11 (checked)
- U TESTIRANJU 3 ISSUES**
 - Problem prilikom uplate vikendima i praznicima (B1-15) - 3
 - Kao korisnik mobilne aplikacije, želio bih vidjeti stanje računa (TRANSAKCIJE) - B1-2 (2)
 - Na zaglavlju svih stranice napraviti prikaz brzog pregleda stanja računa (TRANSAKCIJE) - B1-9 (checked)
- DOVRŠENO 2 ISSUES**
 - Kao korisnik mobilne aplikacije, želio bih vidjeti prikaz zadnjih transakcija (KONVERZIJA EUR/HRK) - B1-1 (checked) - 3
 - Na naslovnoj stranici napraviti prikaz zadnjih 5 transakcija po korisničkom računu (TRANSAKCIJE) - B1-6 (checked)

Slika 10. Sučelje Scrum ploče

Product Owner koristi Backlog te vrši unos svih korisničkih priča vezanih za razvoj aplikacije u Backlog proizvoda. Priče su formirane od zahtjeva koje je Product Owner dobio od klijenta (uglavnom putem direktne komunikacije) koje je nakon toga pretvorio u korisničke priče koje bi trebale biti lako razumljive razvojnom timu. Primjeri su prikazani na slici 11.



Slika 11. Prikaz korisničkih priča (izvor: autor)

Ukoliko razvojni tim utvrdi da nema poteškoća s razumijevanjem korisničkih priča i ako se donese procijena da će zadaci biti izvršeni u zadanom roku, Product Owner zajedno sa Scrum Masterom dogovara datume početka i planiranog završetka Sprinta.

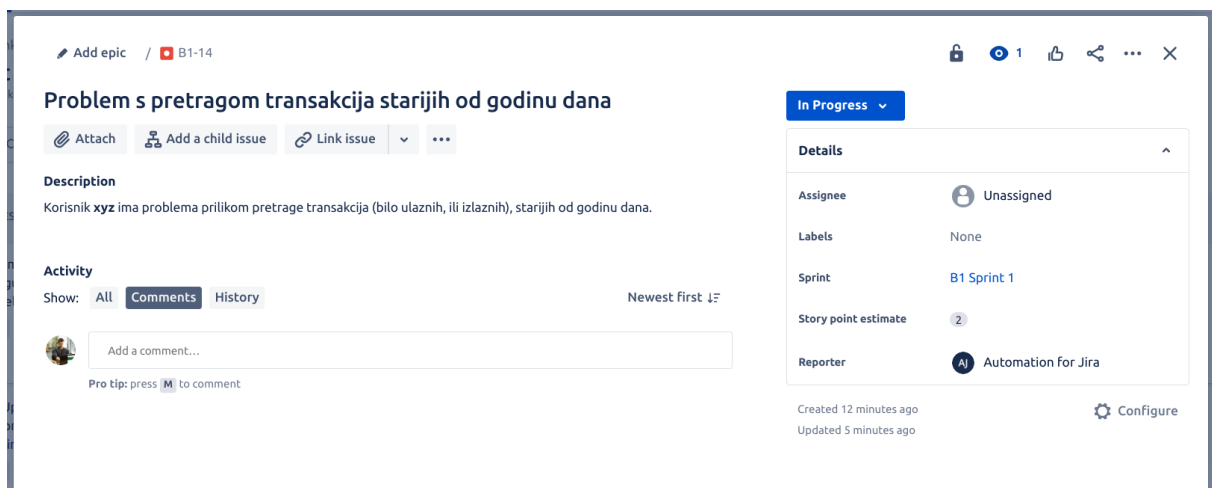
Korisničke priče moraju biti dodijeljene članovima razvojnog tima prije započinjanja Sprinta. Članovima tima je omogućeno da predlažu, odabiru i međusobno premještaju priču drugom članu tima ako vide potrebu za time. Scrum Master u dogovoru s Product Ownerom namješta Sprint na 2 tjedna i Sprint započinje.

Tijekom svakodnevnih sastanka, raspravlja se u trenutnom napretku tima u razvoju projekta. Komunikacija nije formalna, ne odgovara se na točno određena pitanja. Kada je Developeru dodijeljena određena korisnička priča, on ju smještaj u stupac „Trenutno u razvoju“. Kada je razvoj dovršen, korisnička priča dospjeva u stupac „U testiranju“. U slučaju da se tijekom testiranja utvrdi da korisnička priča ne odgovara implementaciji, moguće ju je vratiti u prethodni stupac dok se ne završe izmjene. Kada se za korisničku priču utvrdi da odgovara implementaciji, Product Owner donosi odluku o „puštanju van“ o čemu obavještava

ostale članove time. Kada su korisničke priče međusobno ovisne, neće izaći jedna prije druge – neće biti premještene u „Dovršeno“. Stupac „Dovršeno“ znači da su sve faze razvoja aplikacije završene i da je priča dostupna korisnicima, te kada su u njega smještene sve korisničke priče projekt se smatra dovršenim. Nakon toga slijede ispravci grešaka i dodavanja novih značajki pri čemu se proces ponavlja – dodaje se priča u Backlog i slijede svi navedeni koraci.

Oblik sučelja greške (slika 12) unaprijed je određen prema dogovoru razvojnog tima i Product Ownera. U ovom slučaju, postoji poteškoća s pretragom transakcija koje su starije od godine dana i sastoji se od slijedećih značajki:

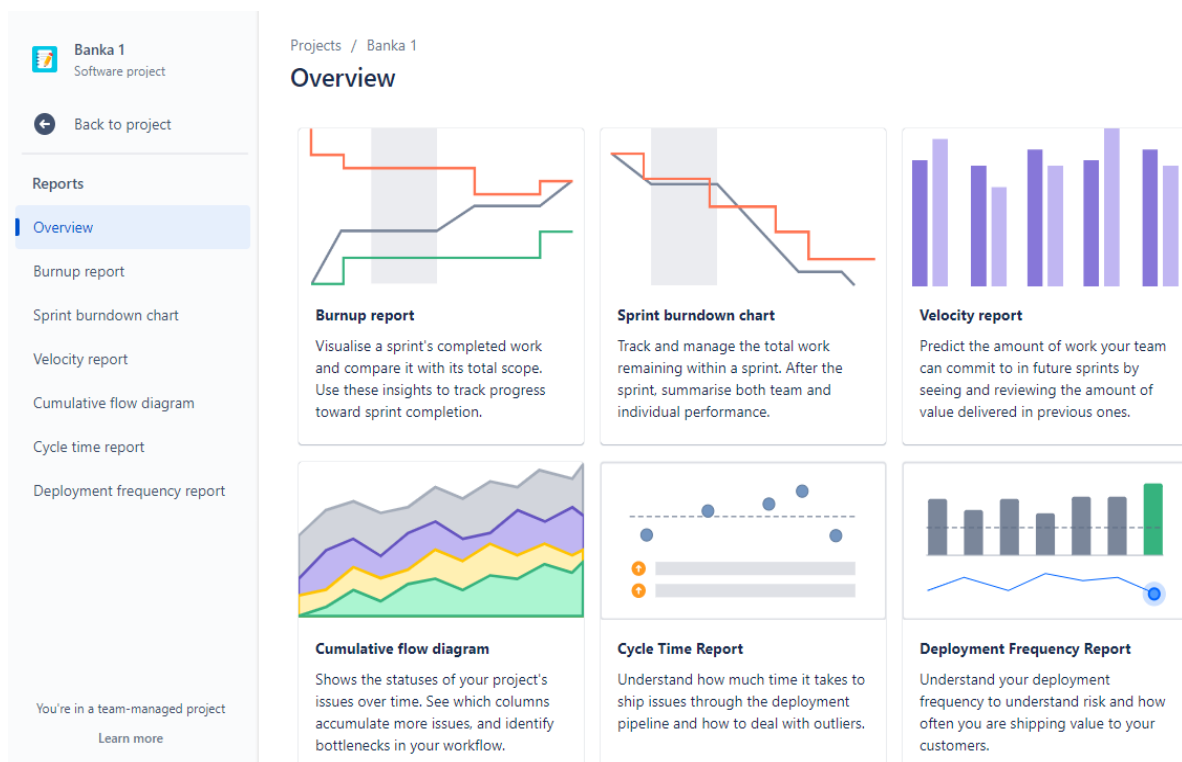
1. Naziv problema/greške
2. Mogućnosti dodavanja dokumenta (eng. Attach)
3. Mogućnosti dodavanja problema koji proizlazi iz trenutnog (eng. Add a child issue)
4. Mogućnosti povezivanja s drugim problemom (eng. Link issue)
5. Opis problema (eng. Description)
6. Aktivnosti (eng. Activity)
7. Prikaz sučelja korisničke priče sa svim pripadajućim značajkama



Slika 12. Prikaz sučelja greške (izvor: autor)

Na slici 13 prikazani su izvještaji (reports). Izvještaj „sagorijevanja“ (eng. Burnup report) prikazuje završeni posao u jednom sprintu, a grafikon (eng. Sprint burndown chart) omogućuje praćenje preostalog (nedovršenog) posla unutar pojedinog sprinta. Izvještaj o brzini (eng. Velocity report) omogućuje predviđanje brzine obavljenog posla u budućim sprintevima, a dijagram kumulativnog protoka (eng. cumulative flow diagram) prikazuje status poteškoća unutar projekta. Vremensko izvješće (eng. Cycle time report) pomaže razumijevanju vremena koje je potrebno za nadvladavanje poteškoća. Izvještaj učestalosti isporuke (eng. Deployment frequency report) prikazuje učestalost isporučivanja proizvoda naručiteljima.

Ukratko, pomoću izvještaja se može vidjeti napredak tima po danima u sprintu te koliko je pojedini sprint bio uspješan. Na kraju se može vidjeti i prilagoditi broj story pointsa i preveliko zauzeće tima te dodatno rasteretiti tim ili dodati veći broj ticketa i story pointsa u sami sprint.



Slika 13. Prikaz izvještaja

6. Rasprava

Pregledom istraživanja uočeno je da je većina dostupnih izvora bazirana na primjenu Scrum metodologije u IT sektoru, dok je manji broj istraživanja fokusiran na Scrum pristup u sektoru bankarstva. U istraživanju u kojem su uspoređivani agilni i tradicionalni pristup u vođenju IT projekata (Vresk, 2020) uočeno je da ispitanici u većem broju smatraju da su agilne metode u IT projektima bolji izbor od onih tradicionalnih u većini slučajeva. Kao najčešće razloge za to ispitanici su navodili činjenicu da svi zahtjevi odnosno značajke proizvoda najčešće nisu poznati na samome početku te se zahtjeva visoka razina fleksibilnosti metode upravljanja projektom kako bi se lako i brzo odgovorilo na navedene promjene. Također, proizvodi u IT sektorima često znaju biti zahtijevani od strane naručitelja s time da naručitelj nema u potpunosti viziju onoga što želi te. U slučaju da se takvi projekti provode tradicionalnim metodama, do prvog kontakta proizvoda i naručitelja došlo bi tek na kraju projekta, što bi vrlo vjerojatno dovelo do nezadovoljstva. Upravo zbog fleksibilnosti agilnih metoda i čestih promjena te konstante upućenosti naručitelja veće je njegovo zadovoljstvo i bolji je globalni uspjeh projekta. Kao prednost agilnih metoda ispitanici su navodili i veću uključenost tima i raznovrsnost mišljenja i gledišta pojedinog problema, a kao nedostatak učestale sastanke i manjak dokumentacije.

Rezultati istraživanja koje je proveo Awad (2005) ukazuju na to da ispitanici zaposleni u IT sektoru preferiraju agilne metode kada je riječ o projektima manje do srednje veličine, no kako su tradicionalne metode još uvijek bolji izbor za velike projekte.

Tijekom istraživanja provedenog u Filipinima (Ng, 2008), studenti su koristeći Scrum metodologiju radili na Capshaw projektu, te su iskazali vrlo pozitivne stavove prema upotrebi agilnih metoda, kao i klijenti – naručitelji. Tijekom Sprinteva, klijenti su bili u mogućnosti pratiti napredak stvaranja softvera, čime se povećao njihov interes i uključenost u projekt što je u konačnici utjecalo na pozitivnu evaluaciju od strane klijenta. Time je potvrđeno da se korištenjem agilnog pristupa povećala uključenost klijenta u projekt. Sa stajališta studenata, mnogi od njih su uočili učinkovitost planiranog načina razvoja softvera. Uporabom agilnih metoda u razvoju softvera, studenti su bili u mogućnosti pratiti i održavati napredak i razvoj. Bili su izloženi određenim aspektima razvojnih problema, od dinamike članova tima do kompleksnosti softvera. Kao negativnu stranu korištenja agilne metodologije, studenti su naveli veliku zahtjevnost u smislu vremena i resursa, te je Sprint produljen s 20 na 30 dana.

Dok je za neke to produljenje dalo dodatnu mogućnost odrađivanja zadatka, kod drugih ispitanika je došlo od opadanja truda koji su ulagali u projekt. Kao drugu negativnu stranu, studenti su navodili izostanak želje ili mogućnosti da iznesu tehničke ili timske probleme, iako su se sastanci redovno održavali, što je dovelo do povećanja stresa i burnouta.

Istraživanja o agilnim metodama vođenja projekta ukazala su na korist u smislu povećanja povjerenja i kohezije unutar tima (McHugh, Conboy i Lang, 2012). Empirijski dokazi upućuju na povezanost s različitim razinama dijeljenog vodstva, orijentacije tima, međufunkcionalnosti, internih procesa učenja i autonomije tima (Moe, Dingsøyr i Røyrvik, 2009; Stettina i Heijstek, 2011). Određena istraživanja dokazala su da se određene osobe ili tipovi osobnosti teško integriraju u agilne timove što dovodi do poteškoća (Whitworth i Biddle, 2007). Također, vremenski okvir potreban za razvijanje intelektualne aktivnosti, kao i motivacija za to mogu značajno varirati ovisno o vrsti projekta pa neki istraživači smatraju da bi trebala postojati ravnoteža između predvidljivih i adaptivnih formula za takav tip dinamičkog menadžmenta (Hidalgo, 2019).

Liang i Shekhar (2018) uočavaju da se u sektoru bankarstva manje od polovine projekata vodi agilnim metoda, te da postoji manjak znanja o istima. Iz tog razloga predlažu strategije pomoću kojih bi se moglo utjecati na probleme koji se javljaju prilikom korištenja agilnih metodologija u sektoru bankarstva: smanjenje veličine projekta, preoblikovanje kulture, adekvatan trening.

Izazov s kojim se suočava primjena agilne metodologije u financijskim institucijama je najčešće veći što je veća banka, gdje je potrebno koordinirati razvoj između odvojenih timova koji rade neovisno na pojedinoj komponenti softvera, primjerice dizajnu, analizi, razvoju ili testiranju. Timovi mogu biti razmješteni u više vremenskih zona ili čak kontinenata.

Drugi razlog nevoljkosti primjene agilne metodologije leži u činjenici da kada je riječ o aplikacijama za financijske usluge, najmanja pogreška može uzrokovati ogromne financijske gubitke te se transakcije moraju provesti s ekstremnom točnošću. Tehnologija ne smije zatajiti, i takav rizik je neprihvatljiv, kao i nemogućnost predvidljivosti ishoda. Upravo iz navedenih razloga mnoge institucije u financijskom sektoru ostaju vjerne modelu vodopada, gdje je dokumentacija opsežna (Saradha, 2016).

7. Zaključak

Prije započinjanja svakog projekta neophodan je odabir adekvatne metode kojom će se taj projekt voditi. Metoda se bira temeljem zahtjeva samog projekta, veličine tima i mogućnosti njegovih članova, te učestalosti promjena. Kada se govori o upravljanju projektima, razlikuju se tradicionalan i agilni pristup upravljanja projektima, te hibridni pristup upravljanja projektima (kombinacija metoda oba pristupa) koji je u zadnje vrijeme sve zastupljeniji. Među agilnim metodama izdvaja se Scrum.

Glavni razlog široke uporabe Scruma jest jednostavnost principa funkcioniranja, odnosno u njegovoj primjenjivosti na bilo kojem tipu projekta ili, ukoliko je riječ o cijelom poduzeću, politici poslovanja poduzeća i njegovom globalnom radu. Fleksibilnost Scrum metodologije i njena mogućnost prilagođavanja potrebama klijenta svrstavaju se u osnovne prednosti.

Najčešće poteškoće u primjeni Scrum metodologije vezane su uz nedefiniran doseg projekta i nedostatak finalnog datuma isporuke proizvoda, mogućnost neadekvatne procjene troškova i potrebnog vremena što može rezultirati produljenjem Srinta, primjerenost metode za manje timove i kraće projekte, ovisnost o predanosti i brojnosti članova tima.

Kako bi se Scrum metodologija provodila adekvatno i uspješno, neophodno je razumjeti da Scrum okvir propisuje smjernice, a kako bi se ostvario cilj projekta uloga poduzeća je da taj okvir prilagodi potrebama korisnika, mogućnostima tima i ostalim obilježjima svakog pojedinačnog projekta.

Literatura

Knjige

1. Keith., C. (2010). Agile Game Development with Scrum. Pearson India
2. Klasić, K., Klarin, K. (2009). Informacijski sustavi- načela i praksa. Zagreb. Intus informatika d.o.o.
4. Pavlić, M. (2011). Informacijski sustavi. Zagreb: Školska knjiga.
8. Pejić Bach, M. (2016). Informacijski sustavi, u knjizi: Varga, M, Strugar, I. (ur). Informacijski sustavi u poslovanju. Zagreb: Ekonomski fakultet.
9. Project Management Institute. (2017) Agile Practice Guide. Project Management Institute.
10. Rubin, K. (2013). Essential Scrum. New York: Addison-Wesley.
11. Spremić, M. (2004). Menadžment i elektroničko poslovanje. Zagreb: Narodne novine.
12. Stellman, A., Greene, J. (2017). Head first Agile. SAD: O'Reilly Media.
13. Thompson, E. (2019). Agile Project Management. Amazon Digital Services LLC
14. Yuonan, J., Mamedov, R. (2020). Agile Project Management in Banking. Stockholm: Södertörn University .

1. Awad, M. A. (2005). A Comparison between Agile and Traditional Software Development Methodologies. The University of Western Australia.
2. Al-Khatib, A.L. (2012). E-Banking: Survey. *International Journal of Advanced Research in Computer Science and Software Engineering*. 2 (10), pp. 12 – 19.
3. Haukur, I. J., Helgi T. I. (2014.) The role of the Product Owner in Scrum- comparison between theory and practise. *Procedia- Social and Behavioral Sciences*. 119(1), pp. 257-267.
4. Hema, V., Thota, S., Naresh Kumar, S., Padmaja, C., Rama Krishna, C. and Mahender, K. (2020). Scrum: An Effective Software Development Agile Tool. *IOP Conference Series: Materials Science and Engineering*. 981(2), pp. 1 – 10.
5. Hidalgo, E. (2019). Adapting the scrum framework for agile project management in science: case study of a distributed research initiative. *Heliyon*. 5(3), pp. 1 – 31.
6. Highsmith, J. and Cockburn, A. (2001). Agile software development: the business of innovation. *Computer*. 34(9), pp. 120-127.
7. McHugh, O., Conboy, K., Lang, M. (2012). Agile practices: the impact on trust in software project teams. *IEEE Softw*. 29 (3), pp. 71 - 76.
8. Moe, N.B., Dingsøy, T., Røyrvik, E.A. (2009). Putting agile teamwork to the test: a preliminary instrument for empirically assessing and improving agile software development. In: Abrahamsson, P., Marchesi, M., Maurer, F. (Eds.). *Agile Processes in Software Engineering and Extreme Programming XP 2009, Lecture Notes in Business Information Processing*, volume 31. Berlin: Springer, pp. 114 – 123.
9. Ng, G. C. (2018). A Study of an Agile methodology with scrum approach to the Filipino company-sponsored I.T. Capstone Program. *International Journal of Computing Sciences Research*. 2(2), pp. 68- 98.
10. Saradha, K. (2016). A Study on Adapting and using Scrum in Financial Sectors. *International Journal of Engineering Research & Technology (IJERT)*. 4 (05), pp. 10 – 5.

11. Sharma, S., Sarkar, D., Gupta, D. (2012). Agile Processes and Methodologies: A Conceptual Study. *International Journal on Computer Science and Engineering*. 4(5), pp. 892-898.
12. Stettina, C.J., Heijstek, W. (2011). Five agile factors: helping self-management to self-reflect. In: *European Conference on Software Process Improvement*. Berlin: Springer Heidelberg, pp. 84 - 96.
13. Sverrisdottir, H., Ingason, H. and Jonasson, H. (2014). The Role of the Product Owner in Scrum-comparison between Theory and Practices. *Procedia - Social and Behavioral Sciences*. 119, pp. 257-267.
14. Takeuchi, H., Nonaka, I. (1986). The New Product Development Game. *Harvard Business Review*. (January-February), pp. 137 – 146.
15. Whitworth, E., Biddle, R. (2007). The social nature of agile teams. In: *Agile Conference (AGILE)*. Washington: IEEE, pp. 26 -36.

Internetski izvori

1. Adell, L. (2013). Benefits & Pitfalls of using Scrum software development methodology. Preuzeto s <http://blog.belatrixsf.com/wp-content/uploads/kalins-pdf/singles/benefits-pitfalls-of-using-Scrum-software-development-methodology.pdf>. [pristupljeno 15.05.2022].
2. Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J., Thomas, D. (2001). Manifesto for Agile Software Development, <http://agilemanifesto.org/>. [pristupljeno 10.05.2022].
3. Hrvatska narodna banka. (2006). Smjernice za upravljanje informacijskim sustavom u cilju smanjenja rizika. Dostupno na: <https://www.hnb.hr/documents/20182/639854/h-smjernice-za-upravljanje-informacijskim-sustavom.pdf/e5579931-e846-47ab-af23-6809debef700>. [pristupljeno 17.05.2022].
4. Hrvatska narodna banka. (2010). Odluka o primjerenom upravljanju informacijskim sustavom. *Narodne novine*, br. 37/2010. Dostupno na: https://narodne-novine.nn.hr/clanci/sluzbeni/2010_03_37_958.html. [pristupljeno 17.05.2022].

5. Liang, J., Shekhar, S. (2018). Agile adoption in investment banks. Harrisburg University of Science and Technology. Dostupno na: http://digitalcommons.harrisburgu.edu/pmgt_dandt/34. [pristupljeno 17.05.2022].
6. Palmquist, S., Lapham, M., and Miller, S. (2013). Parallel Worlds: Agile and waterfall differences and similarities. Dostupno na: https://www.researchgate.net/publication/266054729_Parallel_Worlds_Agile_and_Waterfall_Differences_and_Similarities. [pristupljeno 17.05.2022].
7. Schwaber, K., Sutherland, J. (2017). Sveobuhvatni vodič kroz Scrum: Pravila igre [online]. Dostupno na <https://scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-Croatian.pdf>. [pristupljeno 15.05.2021].
8. Schwaber, K., Sutherland, J. (2020). Sveobuhvatni vodič kroz Scrum: Pravila igre [online]. Dostupno na <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Croatian.pdf>. [pristupljeno 15.05.2022].
9. Zekić-Sušac, M. (2013). ICT u bankarstvu – Informacijski sustav E – bankarstva. Dostupno na: http://www.efos.unios.hr/ict-u-bankarstvu/wp-content/uploads/sites/241/2013/04/pogl7_Info_sustav.pdf. [pristupljeno 17.05.2022].

Popis tablica

Tablica 1. Razlika između agilnih i tradicionalnih metoda.....	5
--	---

Popis slika

Slika 1. Tijek Scrum-a.....	16
Slika 2. Artefakti u Scrum-u	17
Slika 3. Komponente IS banke	22
Slika 4. Kreiranje projekta – prvi korak	23
Slika 5. Kreiranje projekta – drugi korak	23
Slika 6. Osnovno sučelje u alatu Jira.....	24
Slika 7. Sučelje Backloga.....	25
Slika 8. Prikaz sučelja korisničke priče.....	26
Slika 9. Prikaz sučelja postavki Srinta	27
Slika 10. Sučelje Scrum ploče.....	28
Slika 11. Prikaz korisničkih priča	29
Slika 12. Prikaz sučelja greške	30
Slika 13. Prikaz izvještaja	31