

# UPRAVLJANJE SOFTVERSKIM PROJEKTIMA

Murat Prašo, Emina Junuz, Indira Hamulić  
UNIVERZITET "DŽEMAL BIJEDIĆ", MOSTAR





Imena autora: Murat Prašo, Emina Junuz, Indira Hamulić  
Naslov djela: Upravljanje softverskim projektima  
Ime urednika: Emina Junuz  
Imena recenzenata: dr. sc. Senad Rahimić, dr. sc. Jasna Hamzabegović  
Ime ilustratora: Emina Junuz  
Ime lektora i korektora: Indira Hamulić  
Broj izdanja: prvo izdanje  
Naziv i sjedište izdavača: Univerzitet „Džemal Bijedić“ u Mostaru  
Godina izdavanja i godina štampanja: 2016  
Za elektronske publikacije potrebno je naznačiti URL – <http://up.fit.ba>

## Sadržaj

1. Osnovni koncepti o projektu .....	8
1.1. Upravljanje projektom .....	12
1.2. CASE alati za upravljanje projektom .....	14
1.3. Projektna terminologija [5].....	15
2. Životni ciklus razvoja softvera (SDLC – System Development Life Cycle).....	18
2.1. “Vodopadni pristup” (engl. “waterfall approach”) razvoja softvera.....	19
2.2. Paralelni pristup.....	20
2.3. Iterativni ili evolucijski model .....	20
2.4. Spiralni model .....	21
2.5. Iterativni razvoj i njegova usporedba sa sekvencijalnim razvojem.....	22
2.6. Promjene u strategiji IT industrije.....	23
3. Agilno upravljanje projektima .....	24
3.1. Agilne manifesto .....	24
3.2. Uloge u SCRUM projektu.....	29
3.2.1. Vlasnik proizvoda (engl. Product owner) .....	29
3.2.2. SCRUM Master.....	31
3.2.3. Razvojni Tim.....	31
3.3. Agilno planiranje i procjena vremena .....	32
3.4. Korisničke priče (engl. User Story).....	33
3.5. Product Backlog .....	33
3.6. Prioritizacija funkcionalnosti .....	34
3.7. Procjena veličine funkcionalnosti .....	35
3.8. Planiranje iteracije/sprint-a .....	35
3.9. Planiranje zadataka.....	36
3.10. Iteracije i praćenje progressa projekta .....	36
3.11. Dnevni SCRUM sastanak.....	37
3.12. Sprint review .....	38
3.13. Praćenje progressa i burndown grafikon.....	38
3.14. Retrospektiva iteracije/sprint-a .....	39
4. Microsoft Team Foundation Server (MS TFS) .....	41
4.1. Instalacija .....	41
4.2. Timski projekt (eng. Team project).....	42
4.3. Funkcionalnosti Team Foundation Server-a .....	42
4.3.1. Version Control .....	43
4.4. Povezivanje na TFS.....	44

5. Upravljanje projektnim ciklusom.....	48
5.1. Projektni ciklus.....	48
5.2. Upravljanje projektnim ciklusom.....	53
5.3 Planiranje upravljanja projektnim ciklusom i upravljački alati .....	56
6. Pristup pomoću logičkog okvira – alata za dizajn i analizu projekta.....	58
6.1. Uvod.....	58
6.2. Faza analize .....	59
6.2.1 Analiza problema .....	59
6.2.2 Analiza ciljeva.....	62
6.2.3 Analiza strategije (razmatranje alternativa) .....	63
6.3 Faza planiranja .....	63
6.3.1. Matrica logičkog okvira .....	64
6.3.2. Nivoi ciljeva .....	65
6.3.3. Pretpostavke .....	67
6.3.4. Faktori koji osiguravaju održivost.....	68
6.3.5. Objektivno provjerljivi indikatori (Objectively Verifiable Indicators,, OVI) .....	69
6.3.6. Izvori provjere podataka (Source of Verification, SOVs).....	70
6.3.7. Sredstva i troškovi.....	71
7. Korištenje logičkog okvira za formulisanje aktivnosti i terminiranje resursa.....	73
7.1. Raspored aktivnosti i resursa.....	73
7.1.1. Lista rasporeda aktivnosti.....	73
7.1.2. Prezentacija redoslijeda aktivnosti .....	76
7.2. Dodjela resursa aktivnostima. ....	76
7.2.1. Kontrolna lista specifikacije sredstava i redoslijeda .....	76
8. Korištenje logičkog okvira za ocjenu prijedloga projekata.....	79
8.1. Uvod.....	79
8.2. Vodič za ocjenu projektnog prijedloga .....	79
8.3. Alati za ocjenu kvaliteta.....	84
9. Monitoring i izvještavanje.....	88
9.1. Uvod.....	88
9.2. Oblikovanje sistema monitoringa.....	89
Analiza projektnih ciljeva .....	89
9.2.1. Pregled procedura implementacije .....	89
9.2.2. Pregled indikatora .....	91
9.2.3. Izvještavanje.....	91
10. Revizija i evaluacija projekta .....	95
10.1. Uvod.....	95

10.2. Kriterijii evaluacije.....	95
10.3. Veza s logičkim okvirom .....	96
10.3.1. Troškovi .....	96
10.3.2. Aktivnosti .....	96
10.3.3. Rezultati .....	97
10.3.4. Cilj projekta.....	97
10.3.5. Svrha projekta .....	98
10.4. Mogućnosti za evaluaciju.....	98
11. Literatura .....	100
Glosar .....	101





## 1. Osnovni koncepti o projektu

Svaki poduhvat sa određenim ciljem koji treba biti realiziran u određenom vremenu sa raspoloživim resursima i čiji proces realizacije i rezultati se mogu evaluirati po tačno utvrđenim kriterijima se smatra projektom.

Definicija projekta glasi: „**Projekat** je složen i neponovljiv poduhvat koji je usmjeren prema određenom cilju u budućnosti, a izvodi se sa ograničenim ljudskim, materijalnim i finansijskim resursima i u ograničenom vremenu [1].“

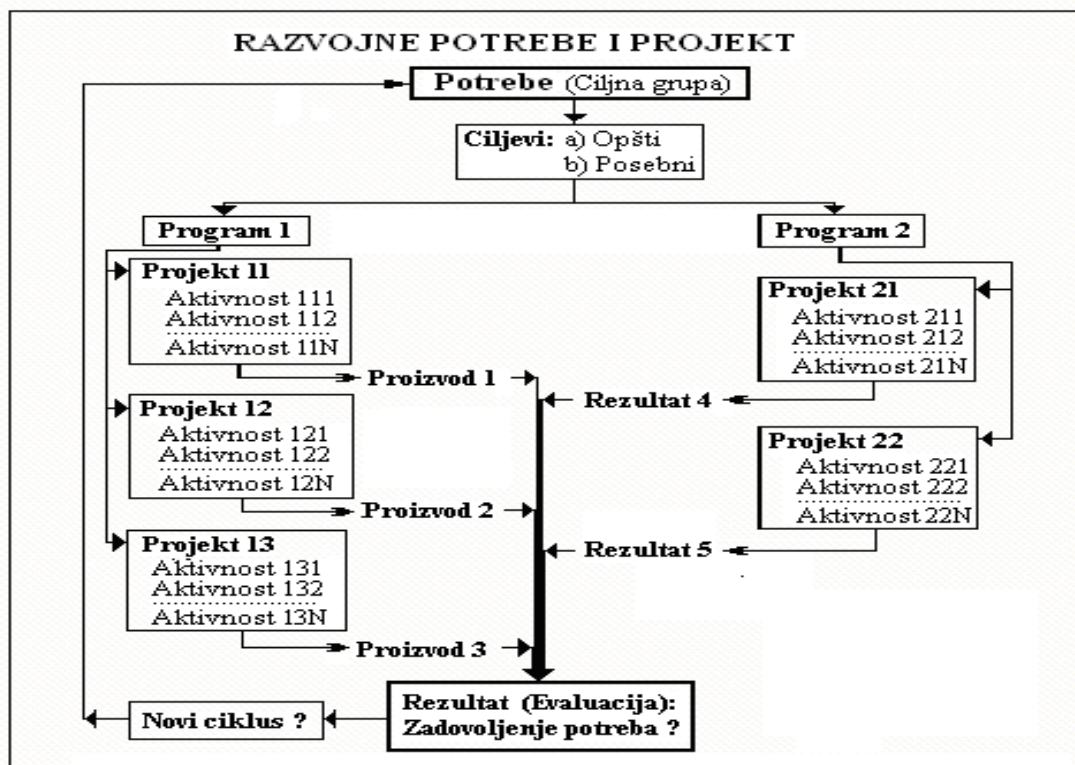
Svaki projekt ima jasno definiran cilj i rezultate. Projekt čini niz aktivnosti. Svaka aktivnost projekta također ima definiran cilj i rezultate. Cilj i rezultati svake aktivnosti su u saglasnosti sa osnovnim ciljem projekta.

Projekt je bilo koji posao koji se sastoji od više aktivnosti.

Projekti su obično dio programa. **Programom** se, obično, smatra skup raznorodnih aktivnosti koje se odvijaju kroz duži niz godina na datom prostoru, ili u različitim oblastima u datom vremenu, sa ciljem da se zadovolji više razvojnih potreba.

Projekt je podskup programa s kraćim trajanjem, manjim raspoloživim sredstvima za realizaciju, precizno definiranim rezultatom i potrebnim ulazima oblikovanim tako da zadovolje neku specifičnu razvojnu potrebu (slika 1). Projektima koji čine jedan program se nužno koordinirano upravljati kako bi se ostvario cilj ili ciljevi programa.

Svaki projekt sadrži rizik, jer je jedinstven i njegov se ishod ne može sa sigurnošću predvidjeti.



Slika 1: Organizaciona šema programa

Projekti se realiziraju u različitim područjima. Međunarodna asocijacija za upravljanje projektima (engl. International Project Management Association - IPMA) razlikuje 10 tipova (kategorija projekta):

- vojni/odbrambeni projekti,
- poslovni i projekti organizacionih promjena,
- projekti komunikacionih sistema,
- projekti specijalnih događaja,
- projekti industrijskih postrojenja,
- softverski projekti,
- internacionalni razvojni projekti,
- medijski projekti,
- razvoj proizvoda ili usluga i
- istraživačko-razvojni projekti.

U bilo kojoj oblasti potrebno je razlikovati jednostavne i složene projekte.

Jednostavne projekte nije potrebno prethodno detaljno planirati i dokumentirati svaki korak. Jednostavni projekti se realiziraju korak po korak i na kraju svakog koraka odlučiti da li i na koji način se prelazi na sljedeći korak.

Primjer jednostavnog kratkoročnog projekta bi mogao biti: „Odlazak kod obućara sa ciljem popravke cipela.“

1. Pitati kolegu gdje u gradu ima obućarska radnja.
2. Odnijeti cipele u obućarsku radnju.
3. Sačekati.
4. Cipele popravljene.

Ovakav tip projekata nije potrebno dokumentirati.

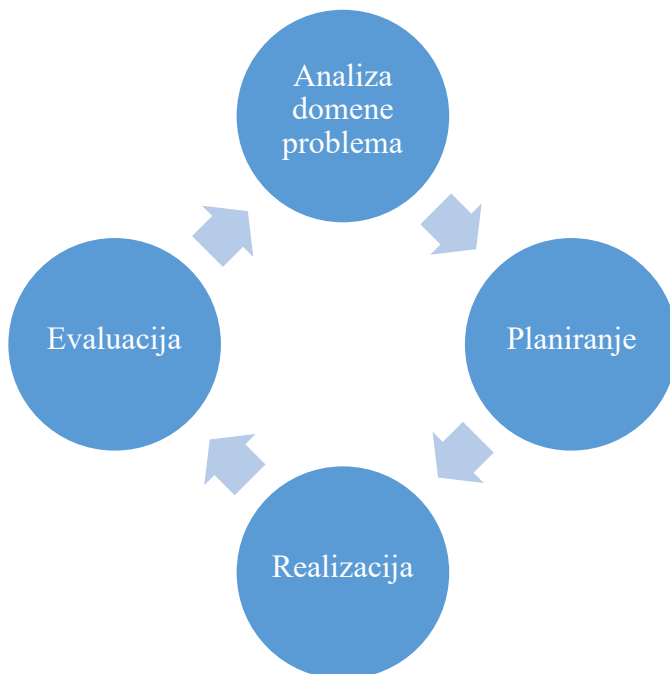
Složene ili kompleksne projekte je potrebno dobro planirati prije početka projekta i dokumentirati svaki korak projekta. Predmet ovoga kursa će upravo biti izučavanje procesa *kreiranja, realizacije i evaluacije* projekata.

Svaki projekt je plan rada u cilju realizacije jedne ideje. Kreirati projekt za realizaciju ideje znači odgovoriti na sljedeća pitanja:

- **Šta?** - Naziv *projekta* (*jednostavan*, jasan i iz njega se vidi čime se *projekt* bavi).
- **Uzrok?** – Napraviti analizu aktuelnog stanja i odrediti uzrok nastanka potrebe za projektom.
- **Zašto?** – Koji problemi će se riješiti realizacijom projekta.
- **Gdje?** – Geografska lokalizacija projekta (naselje, općina, država, ...)
- **Kako?** – Definiranje popisa aktivnosti projekta.
- **Ko?** – Ko realizira aktivnosti ili ko su odgovorne osobe za realizaciju aktivnosti?
- **Kada?** – Definirati datum početka projekta, datum završetka projekta, datume početka i završetka svake aktivnosti, datum srednjoročnoga izvještaja itd.
- **Koliko?** – Definirati troškove projekta (popis resursa potrebnih za realizaciju projekta i njihova cijena).

Osnovne faze svakog projekta su (slika 2):

1. Analiza domene problema (definiranje problema, traži se uzrok nastanka potrebe za projektom, te odgovori na pitanje zašto? i gdje?),
2. Planiranje (traži se odgovor na pitanja: kako?, ko?, kada? koliko?),
3. Realizacija (Izvedba prethodnih faza projekta će biti jednostavna ukoliko su prethodne faze ispravno urađene) i
4. Evaluacija (provjera da li su: postignuti zadani ciljevi, sve aktivnosti realizirane u zadanom roku sa predviđenim resursima. Ovo je i faza učenja iz iskustva u svrhu primjene stečenih znanja u budućim projektima).



Slika 2: Faze projekta

Ako projekt ne uspije također je koristan jer iz njega se može naučiti sve što je bilo dobro i ne ponoviti ono što je bilo loše.

Često je razlog neuspjeha projekta ne upravljanje projektom. Projekti ne završavaju u ugovorenim rokovima, ne isporučuju ugovorene funkcionalnosti i izlaze izvan ugovorenog budžeta zato jer se projektom ne upravlja ili se ne primjenjuju standardni procesi upravljanja.

## 1.1. Upravljanje projektom

**Upravljanje projektima ili projektni menadžment** je specijalizovana disciplina modernog menadžmenta koja se bavi upravljanjem različitim projektima radi poboljšanja efikasnosti njihove realizacije. To je jedan novi upravljački koncept kojim se, uz pomoć odgovarajućih metoda organizacije, planiranja i kontrole, usklađuju svi potrebni resursi i koordinira obavljanje potrebnih aktivnosti da bi se određeni projekat realizovao na najefikasniji način, odnosno u minimalnom (predviđenom) vremenu i sa minimalnim (predviđenim) troškovima [1].

**Projektni menadžer** je osoba zadužena da projekat dovede do završetka. On vodi projektni tim koji radi na upravljanju realizacijom projekta.

Projektni menadžer je osoba odgovorna za planiranje, koordinaciju i kontrolu projekta od početka do završetka, zadovoljavajući zahtjeve projekta i osiguravajući pravovremenu realizaciju unutar dozvoljenih troškova i u skladu sa standardima kvaliteta.

Koristi projektnog menadžmenta su:

- Unapređenje planiranja i omogućavanje efikasnog ostvarenja ciljeva,
- Postavljanje jasnih i mjerljivih ciljeva,
- Koordinisanje resursa,
- Utvrđivanje i upravljanje rizicima,
- Ušteda vremena potrebnog za realizaciju projekta ,
- Smanjenje troškova realizacije projekta,
- Ostvarivanje željenih rezultata.

Razvoj formalnoga upravljanja projektima započeo je 1950-ih kao potreba Ministarstva obrane Sjedinjenih Američkih Država za razvojem složenih vojnih sistema. Time se potvrđuje i činjenica da je područje upravljanja projektima nastalo iz tradicionalnih inženjerskih disciplina [2].

Područje upravljanja projektima, iako se razvilo iz tehničkih disciplina, s vremenom je pod utjecajem drugih područja sve više postalo multidisciplinarno. Tako za uspješan rad na cjelokupnom projektu treba uzeti u obzir, osim užeg područja upravljanja projektima, prije svega organizacijsku strukturu i okruženje projekta i znanje s područja primjene projekta,

standarde i pravni okvir, te općenito znanje iz poslovnog upravljanja i međuljudskih odnosa. Svaki od tih faktora može imati veliki uticaj na uspješnost projekta.

U početku je upravljanje projektima nametnuto iz potrebe za standardizacijom procesa i uključivalo je jasne ciljeve pa su timovi koji su dobili zadatak mogli pouzdano planirati. Najveći faktor daljnjega rasta područja bila je kompleksnost poslova unutar inženjerskih zanimanja [3]. U to je doba (1960-e) računarska industrija započela svoj jaki utjecaj na poslovna okruženja, te su se računari počeli sve više koristiti za poslovne potrebe. U tim ranim počecima za sve se projekte koristio isti pristup, naslijeđen od ostalih inženjerskih disciplina. Međutim, brzorastuće područje računarske industrije često je imalo za posljedicu neuspješne projekte, bilo zbog neispunjenja rokova bilo zbog očekivanja kupaca. Postajalo je očitiije da tradicionalni pristup nije prikladan za takve projekte. Paralelno s tradicionalnim pristupom počeli su se razvijati i novi pristupi, dok je ovaj tradicionalni i dalje ostao u primjeni. Nasuprot tradicionalnom inženjerskom pristupu, razvio se dinamični model, koji je morao odgovoriti na veće zahtjeve za kontrolom troškova nasuprot rezultatima i vrijednostima projekata, a i sve većim zahtjevima za bržim postignućem ciljeva projekta. Kao krajnji korak u razvoju, dovodeći sva ograničenja vezana uz povećanje koristi proizašle iz projekta, brzom razvoju i velikim promjenama unutar projektnog plana do velikoga izražaja, pojavio se ekstremni pristup upravljanja projektima [4].

Primjeri u softverskom inženjeringu:

- Tradicionalni pristup - model vodopada,
- Agilni pristup – inkrementalni model i
- Ekstremni pristup - ekstremno programiranje.

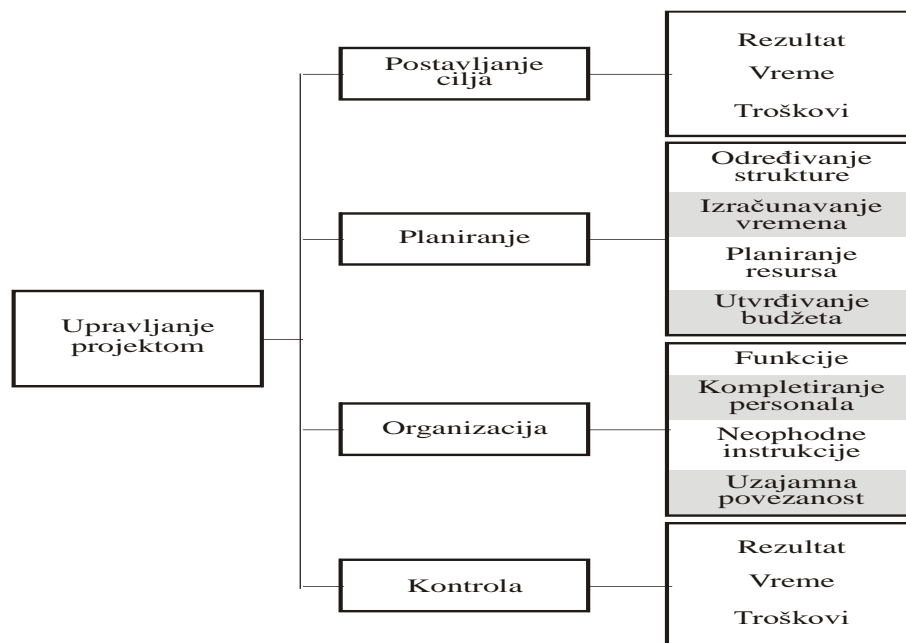
Upravljanje projektom predstavlja naučno zasnovan i u praksi potvrđen koncept kojim se uz pomoć odgovarajućih metoda organizacije, planiranja i kontrole vrši racionalno usklađivanje svih potrebnih resursa i koordinacija obavljanja potrebnih aktivnosti da bi se određeni projekat realizovao na najefikasniji način.

Upravljanje projektom je posao osiguranja krajnjih ciljeva projekta, uz suočavanje sa svim rizicima i problemima koji se javljaju u realizaciji. To je jedna specijalizovana oblast upravljanja koja je razvijena da bi se koordiniralo i upravljalo sa većim brojem kompleksnih aktivnosti u modernoj industriji.

Upravljanje projektom je umijeće kako izvoditi projekat saradnjom ljudi u dogovorenom vremenu, određenim sredstvima rada i troškovima sa željenim učinkom.

Koncept upravljanja projektom dakle, kao što se vidi iz prethodnih definicija, se zasniva na uspostavljanju i korišćenju takve organizacione forme koja omogućava najefikasniju realizaciju projekta, odnosno najefikasnije korišćenje raspoloživih metoda, resursa i ljudi za postizanje optimalnih rezultata u realizaciji projekta.

Na taj način se upravljanje projektom iskazuje kao kompletna koncepcija, koja obuhvata interdisciplinarnu primjenu više metoda i tehnika organizacije, planiranja i kontrole u cilju što efikasnije realizacije određenog projekta (slika 3).



Slika 3: Upravljanje projektom [1]

S obzirom da je ovo jedan opšti model upravljanja projektom, njegove osnovne funkcije (postavljanje cilja, planiranje, organizovanje i kontrola) su iste kao i kod drugih upravljačkih procesa, ali je njihova razrada usklađena sa potrebama upravljačkih procesa usmjerenih na realizaciju projekta.

## 1.2. CASE alati za upravljanje projektom

CASE (engl. Computer Aided Software Engineering) alati za upravljanje projektom koriste se zajedno sa metodologijom upravljanja projektom.

Popis najkorištenijih u 2014. godini i njihov kratak opis je dostupan na <http://www.capterra.com/project-management-software/>

U okviru ovog kursa studenti će koristiti Microsoft Project 2013 i Open project koji je free software i dostupan na <https://www.openproject.org>.

### 1.3. Projektna terminologija [5]

<b>Ciljna grupa projekta</b>	<b>Pojedinac ili grupa subjekata korisnika rezultata projekta.</b>
<b>Direktna opservacija</b>	<b>Opservacija projektnog tima na konkretnom terenu, u vezi s projektom.</b>
<b>Evaluacija</b>	<b>Ocjena stepena do kojeg su ostvareni zacrtani ciljevi projekta.</b>
<b>Fokus grupa</b>	<b>Mala radna grupa koja raspravlja o problemu u nastojanju da identifikuje osnovne među njima, bez vanjskih utjecaja.</b>
<b>Identifikacija prepreka</b>	<b>Proces otkrivanja potencijalnih smetnji realizaciji projekta.</b>
<b>Indikatori</b>	<b>Brojčani podaci o nekoj pojavi, zasnovani na odabranim kriterijima.</b>
<b>Intervju</b>	<b>Sistematski proces pribavljanja i bilježenja informacija kroz postavljanje pitanja.</b>
<b>Istraživanje dokumentacije</b>	<b>Sistematsko pretraživanje i analiza podataka i informacija koje su dobijene iz sekundarnih izvora.</b>
<b>Izlazi</b>	<b>Očekivani neposredni rezultati projekta.</b>
<b>Kontrola</b>	<b>Proces provjere saglasnosti predviđenih i ostvarenih nabavki, radova i troškova.</b>
<b>Kritičan put</b>	<b>Put u procesu pripreme i implementacije projekta duž kojeg svako zakašnjenje izaziva kašnjenje cijelog projekta.</b>
<b>Kvalitativno mjerenje</b>	<b>Nestandardizovana metoda ocjene efekata projekta</b>



<b>Kvantitativno mjerenje</b>	<b>Standardizovana metoda utvrđivanja mjerljivih dimenzija neke aktivnosti ili projekta.</b>
<b>Lista aktivnosti</b>	<b>Popis aktivnosti koje je nužno obaviti radi pripreme i realizacije projekta.</b>
<b>Monitoring</b>	<b>Proces tehničkog, finansijskog i drugih oblika svakodnevnog upravljanja u procesu formulacije i izvođenja projekta.</b>
<b>Nepredviđeni izdaci</b>	<b>Razuman iznos kojim se pokrivaju nepredviđene promjene uslova projekta radi pokrića rizika projekta.</b>
<b>Opšti cilj projekta</b>	<b>Glavni cilj za ciljnu grupu projekta.</b>
<b>Plan aktivnosti</b>	<b>Tekstualni i grafički prikaz rasporeda aktivnosti koji omogućava planiranje, izvršenje i nadzor projekta.</b>
<b>Predračun</b>	<b>Procjena troškova izvođenja projekta.</b>
<b>Pristup ključnom informatoru</b>	<b>Sistematsko intervjuisanje izabраниh osoba koje su u poziciji da imaju znanje iz prve ruke o korisnicima projekta.</b>
<b>Program</b>	<b>Skup projekata čijom sinergijom se ostvaruju optimalni efekti.</b>
<b>Projekt</b>	<b>Zaokružen podskup aktivnosti koje omogućavaju stvaranje izlaza.</b>
<b>Raščlanjivanje projekta</b>	<b>Sistematsko razlaganje projekta iz složenih u jednostavnije aktivnosti koje omogućavaju grafičku prezentaciju.</b>
<b>Rizici</b>	<b>Uticaji ili pojave koje mogu ugroziti formulisanje ili realizaciju projekta.</b>
<b>Tehnički aspekt projekta</b>	<b>Dio projekta u kom se razrađuju tehnički detalji, imajući na umu racionalnost projekta.</b>

<b>Učinak</b>	<b>Proizvod, usluga ili drugi rezultat neke od aktivnosti projekta ili projekta kao cjeline.</b>
<b>Ulazi</b>	<b>Serija aktivnosti i resursa bez kojih nije moguće postići transformaciju projekta u proizvod.</b>
<b>Upravljački ciklus</b>	<b>Logički slijed aktivnosti upravljanja koje se izvršavaju uglavnom prema predviđenom redu.</b>
<b>Upravljanje projektom</b>	<b>Niz vrijednosnih procjena i kontrolnih odluka koje imaju za cilj da zadovolje neku potrebu, korištenjem odgovarajućih resursa.</b>
<b>Vrijeme mirovanja</b>	<b>Period u pripremi ili realizaciji projekta u kom nije moguće razvijati neku aktivnost projekta a izvan je kontrole projektnog tima.</b>

## 2. Životni ciklus razvoja softvera (SDLC – System Development Life Cycle)

**Životni ciklus razvoja softvera (SDLC – System Development Life Cycle)** ili „razvojni proces“ opisuje „životni vijek“ od nastanka potrebe za softverom, preko implementacije, održavanja, pa sve dok se ne prestane koristiti.

U ranoj fazi razvoja informacijskih tehnologija razvoj softvera je bio fokusiran na tehnologiju, programiranje i tehničke vještine.

Stručnjaci koji su radili na razvoju su posjedovali zavidnu razinu tehničkih vještina, međutim sve se češće dešavalo da se kod velikih projekata zakaže u smislu pravovremene isporuke naručenog softvera.

Kako upotreba informacijskih tehnologija rasla sve je više rasla i svijest o potrebi za jedan općeprihvaćen standardiziran pristup razvoju softvera.

Ne postoji univerzalni proces za rješavanje problema koji može biti primjenjiv u svim situacijama u kojima se očekuje razvoj novog softvera.

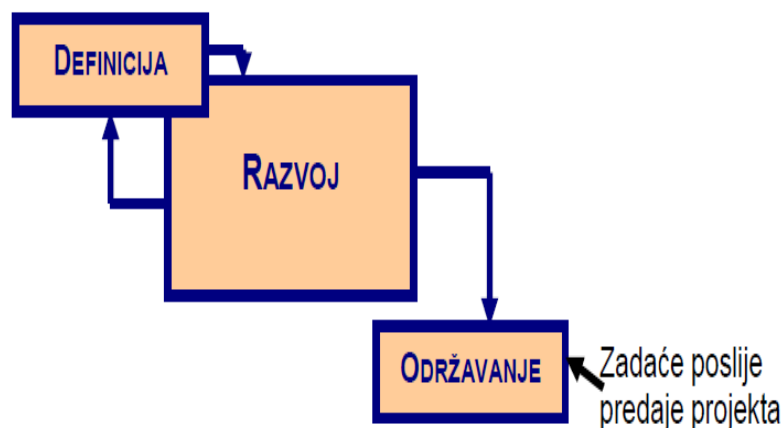
Pristupi razvoju softvera moraju pratiti potrebe konkretne situacije (okruženje, zakonski okvir, organizacijska struktura, strateški plan razvoja, operacioni planovi, postojeća infrastruktura itd.).

U nekim elementima procesi razvoja softvera mogu biti standardizirani do određenog stupnja. Životni ciklus razvoja sistema (eng. System development lifecycle-sdlc) je jedan od pokušaja da se postigne određeni stupanj standardizacije.

**Model procesa razvoja softvera** definira redoslijed razvojnih faza.

Na slici 4 je prikazana općenita struktura razvoja i održavanja softvera. Nakon okončanja razvojnog procesa prelazi se u fazu održavanja (60% posla). Potrebno je u fazi ugovaranja sa naručiocem proizvoda jasno odrediti šta podrazumjeva faza održavanja, da ne bi došlo do situacije u kojoj korisnik traži implementaciju nove verzije softvera ili potpuno novog softvera.

### Općenita struktura procesa razvoja IS:



Slika 4: Opća struktura razvoja i održavanja softvera

Koncept SDLC-a omogućuje:

- Sistematski i uređen pristup rješavanju problema koji se tiču obrade poslovnih informacija;
  - Sredstvo za upravljanje, usmjeravanje, nadzor i kontrolu procesa izgradnje IS-a
- To uključuje:

- Opis procesa ili koraka ili faza koje se trebaju slijediti,
- Isporuke-izvještaji, programi, dokumentacija i
- Bitne tačke razvoja - datumi završetka pojedinih faza razvoja.

Postoji više varijanti SDLC-a, a to su:

- Tradicionalni linearni (sekvencijalni) “vodopadni” (eng. “waterfall”),
- Paralelni model,
- Iterativni model ili evolucijski model,
- Spiralni model,
- Agilne metode itd.

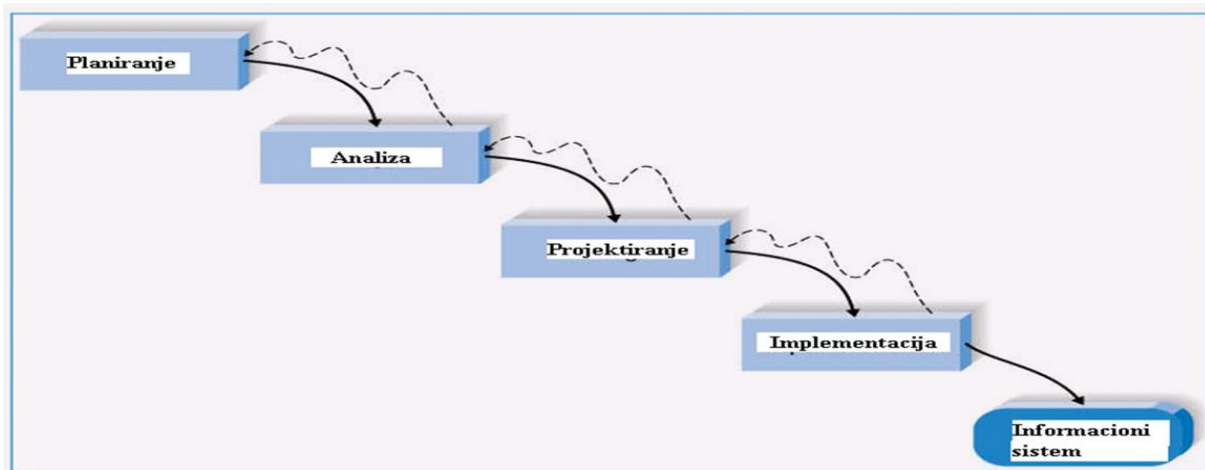
U nastavku će ukratko biti predstavljeni linearni, paralelni, iterativni i spiralni model, a posebno poglavlje će biti posvećeno agilnome i pristupu SCRUM.

## 2.1. “Vodopadni pristup” (engl. “waterfall approach”) razvoja softvera

Pri korištenju sekvencijalnih metodologija razvoja analitičar sistema i članovi razvojnog tima napreduju sekvencijalno od jedne do druge faze projekta. Svaka faza se u cjelosti završi i potom se prelazi na sljedeću fazu. Nije uobičajeno vraćati se na prethodnu fazu. Osnovne faze modela vodopada su prikazane na slici 5.

Ključne prednosti ovakve metodologije su:

- Softverski zahtjevi su identificirani mnogo prije nego što počne njegov razvoji
- Promjene u zahtjevima su sve manje kako projekat implementacije softvera napreduje.



Slika 5: Sekvencijalni pristup razvoju softvera

Osnovni nedostaci sekvencijalnog pristupa su:

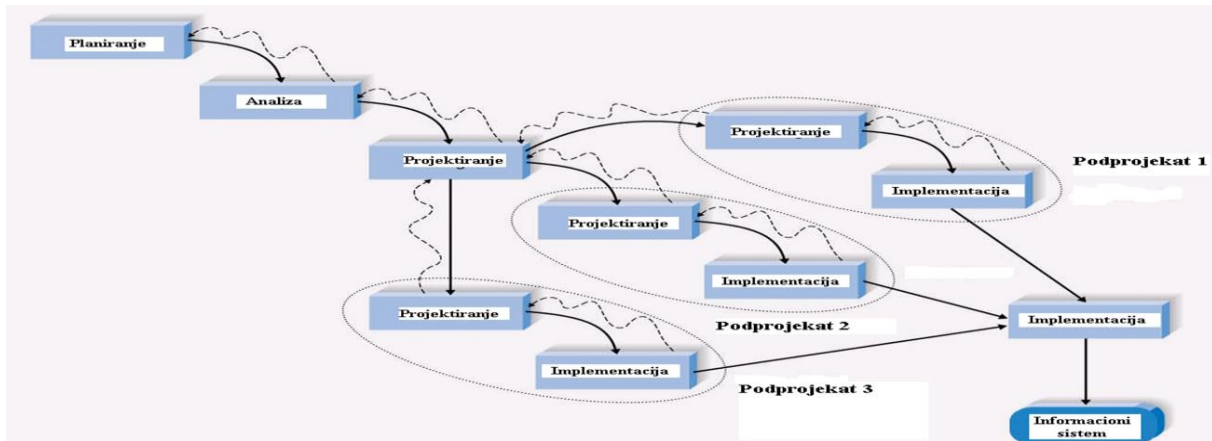
- Dizajn softvera mora biti u potpunosti završen prije početka implementacije,
- Između prijedloga u fazi analize i isporuke softvera protekne mnogo vremena što može dovesti i do paralize u analizi.

Paraliza u analizi zapravo se dešava zbog predetaljne analize. „Manje je više“, te je potrebno staviti fokus na ono što ne treba raditi i neće doći do paralize.

## 2.2. Paralelni pristup

Metodologija paralelnog razvoja nastoji smanjiti vrijeme između faze analize i isporuke softverskog proizvoda. Prvo se napravi dizajn cjelokupnog proizvoda a zatim se projekt dijeli na više podprojekata. Svi podprojekti, ukoliko ne postoji međusobna ovisnost među njima, se realiziraju paralelno. Za paralelnu realizaciju podprojekata potrebno je više resursa, npr. ljudskih resursa, ali je vrijeme realizacije projekta znatno skraćeno. Rezultati svakog od podprojekata potom se integriraju u jednu cjelinu, projekt.

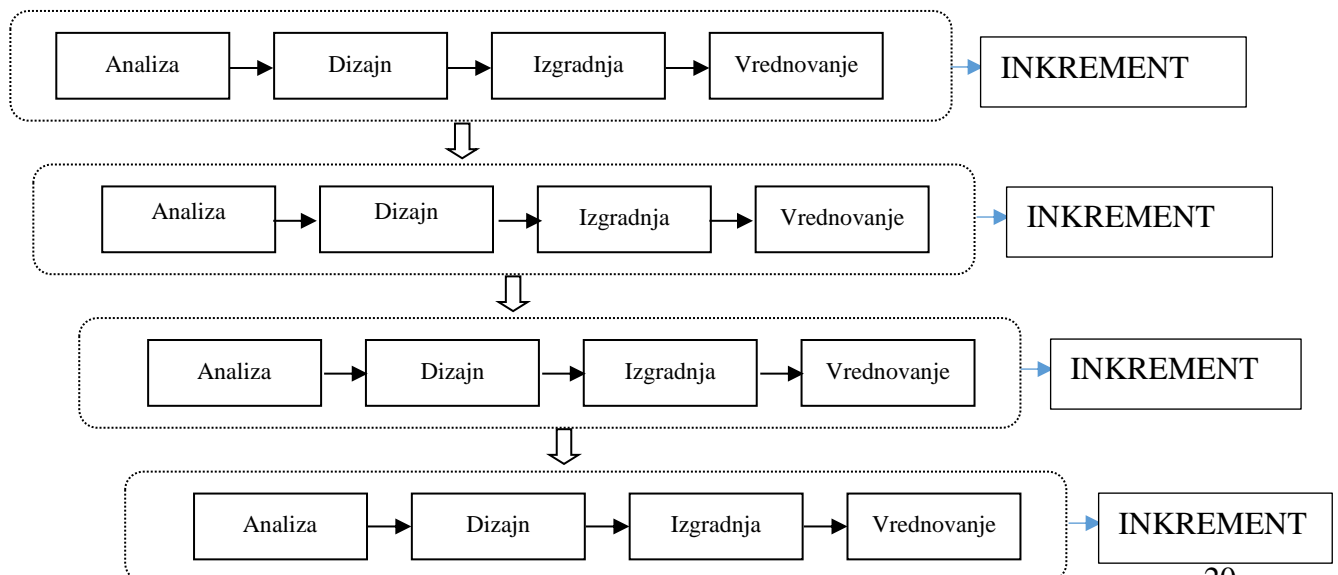
Faze paralelnog pristupa razvoju softvera su prikazane na slici 6.



Slika 6: Paralelni pristup razvoju softvera

## 2.3. Iterativni ili evolucijski model

Iterativni ili evolucijski model je model u kojem korištenje softverskog proizvoda mijenja perspektivu korisnika u odnosu na softverski proizvod. Svaka promjena perspektive može rezultirati generiranjem novih korisničkih zahtjeva. Na taj način softverski proizvod raste zajedno sa organizacijom u kojoj se koristi. Ovakav pristup razvoja podrazumjeva brzo prototipiranje, odnosno, gradnju dijelova softverskog proizvoda u inkrementima. Razvoj svakog inkrementa podrazumjeva realizaciju niza razvojnih aktivnosti. Svaki niz razvojnih aktivnosti omogućuje kreiranje funkcionalnog proizvoda koji se isporučuje i na temelju kojeg se generiraju budući zahtjevi. Na slici 7 je prikazan evolucijski pristup razvoju softvera.



Slika 7: Evolucijski pristup razvoju softvera

## 2.4. Spiralni model

Spiralni model nastoji kombinirati prednosti "top-down" i "bottom-up" pristupa.

Faze spiralnog modela su:

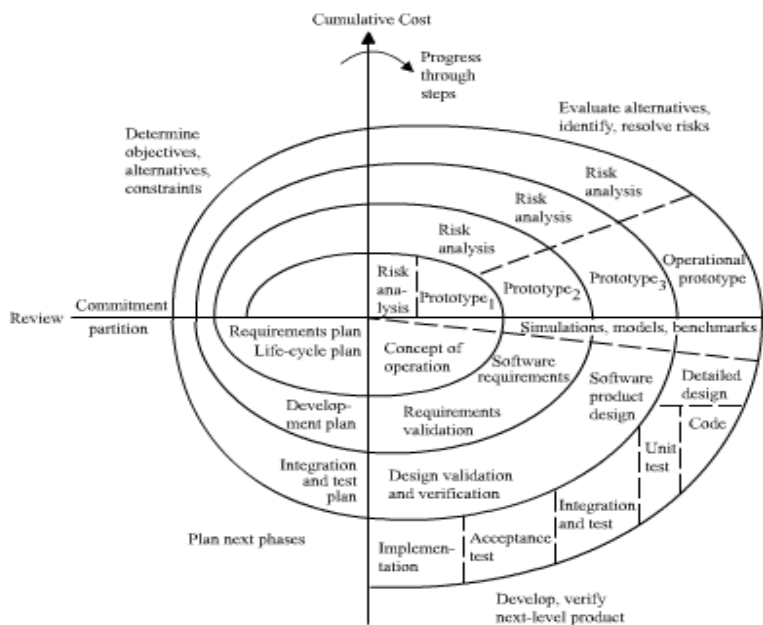
1. Analiza rizika i procjena alternativa;
2. Razvoj i verifikacija sljedećeg "proizvoda";
3. Planiranje sljedeće faze i
4. Pregled - Određivanje ciljeva, alternativa i ograničenja.

Na početku svake faze radi se analiza rizika i nastoje se utvrditi mogući rizici. Cilj analize rizika je eliminirati rizike ili barem ih svesti na najmanju moguću mjeru. Ako je rizik nastavka projekta prevelik, tada se projekat prekida.

Kao primjere rizika mogu se navesti:

- postojanje rizika da realizirani proizvod neće zadovoljavati postavljene zahtjeve;
- postojanje rizika da će cijena razvoja softverskog proizvoda premašiti korist koja se očekuje od njegove upotrebe.

Na slici 8 je dat grafički prikaz koji ilustrira ključne osobine spiralnog modela. Osa Y (ordinata) predstavlja kumulativni trošak. Svaka petlja spirale počevši od osi X prikazuje jednu fazu u razvoju softvera, bez obzira kako je ta faza realizirana (prototipski, evolucijski, ...). Pri svakom prolasku kroz osu X donosi se odluka da li će se nastaviti sa razvojem softvera.



In 1988, Barry Boehm

**The Spiral Model**

Slika 8: Spiralni model razvoja softvera

Na slici 9 je na drugačiji način prikazan spiralni model razvoja softvera.

Dio faze koji na slici 6 ide od tačke 1 do tačke 2 prikazuje analizu rizika, te procjenu alternativa.

Dio faze od tačke 2 do tačke 3 prikazuje razvoj i verifikaciju sljedećeg proizvoda.

Dio faze od tačke 3 do tačke 4 prikazuje planiranje sljedeće faze.

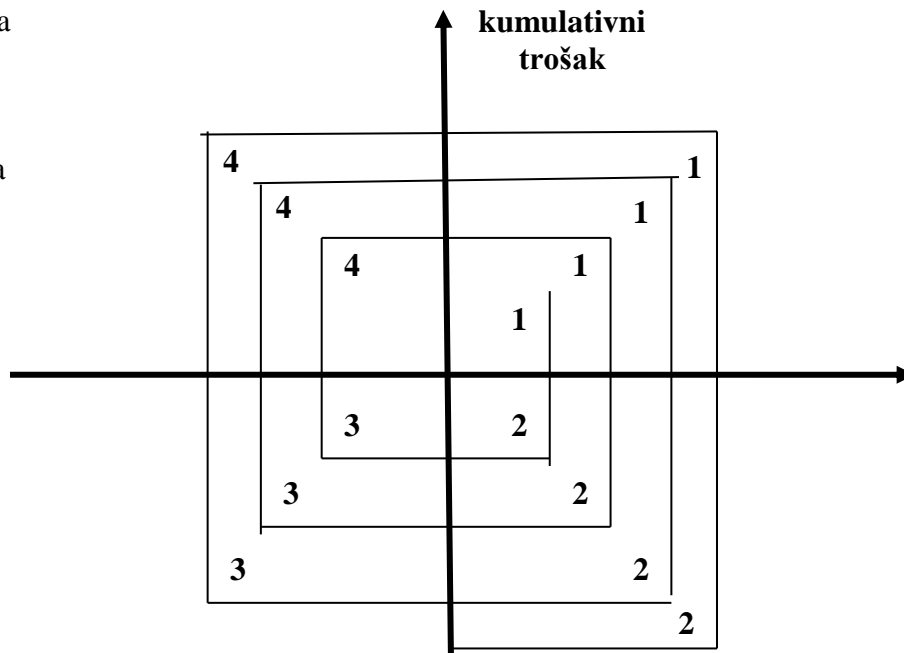
I na kraju, dio razvojne faze prikazan od tačke 4 do tačke 1 prikazuje pregled i određivanje ciljeva, mogućih alternativa i ograničenja.

integracija

gradnja

dizajn

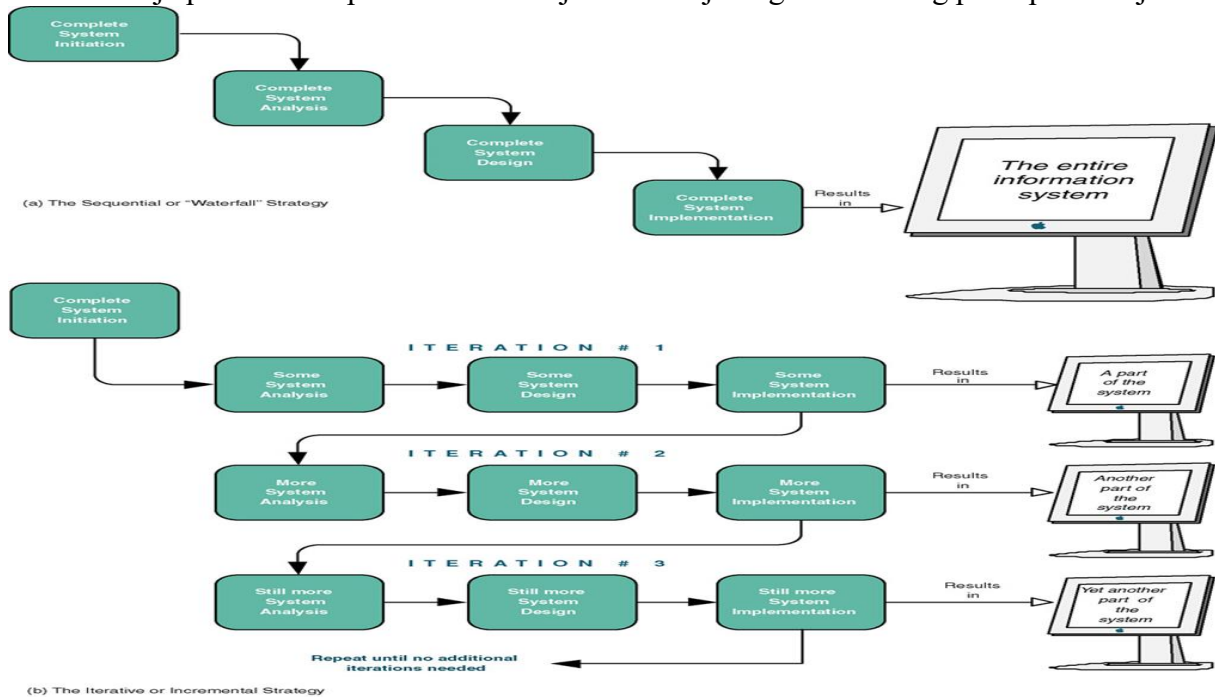
analiza



Slika 9: Spiralni model

## 2.5. Iterativni razvoj i njegova usporedba sa sekvencijalnim razvojem

Na slici 10 je prikazana usporedba korištenja sekvencijalnog i iterativnog pristupa razvoja.



Slika 10: Usporedba iterativnog i sekvencijalnog razvojnog procesa

## 2.6. Promjene u strategiji IT industrije

Transformacija poslovanja općenito dovela je do promjene u strategiji IT industrije. Zahtjevi se često mijenjaju te je potrebno često raditi nadogradnju softverskih rješenja, što ukazuje na sve veću potrebu korištenja agilnih i iterativnih procesa razvoja IS-a.

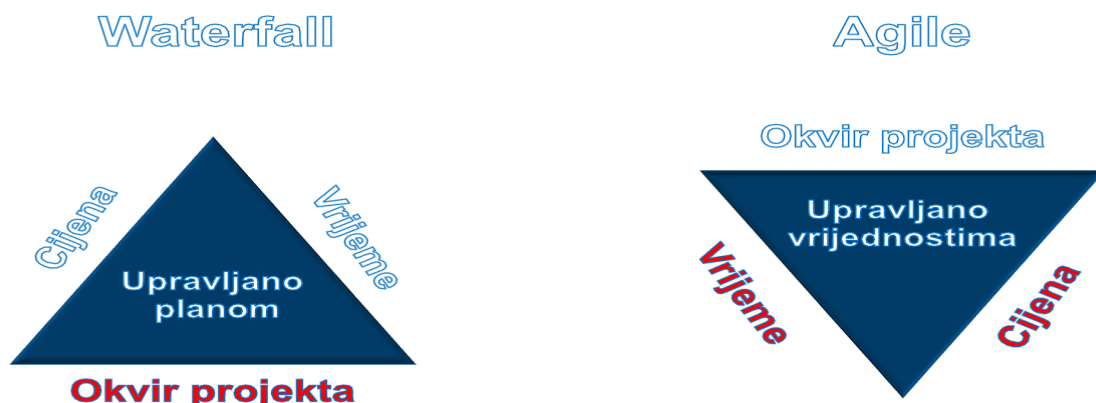
U tabeli jedan je prikazana usporedba sekvencijalnog modela vodopada i agilnih metoda razvoja (+ prednost, - nedostatak).

	<b>Agile</b>	<b>Waterfall</b>
Opseg projekta	-	+
Fleksibilan na promjene	+	-
Korisnički feedback	+	-
Strukturiranost	-	+
....		

Tabela 1: Agile vs. Waterfall model

Svaki projekt, pa i projekt razvoja softvera se može posmatrati kao trougao čije stranice su: cijena, vrijeme i okvir projekta (slika11). Obzirom da se u sekvencijalnom modelu, modelu vodopada, već u prvoj fazi definiraju korisnički zahtjevi i okvir projekta u sljedećim fazama je potrebno prilagođavati finansijske resurse i vrijeme.

Agilne metode dopuštaju kreiranje novih korisničkih zahtjeva i općenito mijenjati okvir projekta, vrijeme predaje projekta i finansijska sredstva su fiksna.



Slika 11: Mogućnosti prilagodbi modela vodopada i agilnih modela razvoja IS-a



### 3. Agilno upravljanje projektima

SCRUM je veoma jednostavan agilni pristup koji se koristi za upravljanje inovativnim projektima razvoja softvera. Glavni cilj SCRUM-a je rano, često i konzistentno dostavljanje inkremenata softverskog proizvoda. SCRUM omogućava timovima softverskih inženjera da dostignu visok nivo performansi i produktivnosti, fokusirajući se na ključne *poslovne vrijednosti* softvera koji se izgrađuje i eventualne *rizike*. Osnove SCRUM-a su male iteracije koje se nazivaju SPRINT (od 7 do 30 dana), agilno planiranje i procjena vremena, skladište dijelova proizvoda koji se moraju završiti, kratko opisani zadaci / korisničke priče i dnevni zadaci. SCRUM tim se sastoji od 5-10 članova od kojih je jedan vlasnik proizvoda, jedan je SCRUM master ili vođa tima i članovi tima koji se mogu specijalizirati za pojedine oblasti razvoja softvera.

SCRUM je jednostavan način upravljanja *inovativnim* projektima. Namijenjen je za realizaciju prije svega inovativnih proizvoda, projekata gdje su zahtjevi promjenjivi u toku razvoja projekta i ne pretjerano obimnih projekata. Za obimne projekte koristi se skalabilni agilni okvir (engl. Scaled Agile Framework - SAFe), dok za projekte koji imaju preciznu specifikaciju zahtjeva koja se neće mijenjati bolje je primijeniti sekvencijalne metodologije razvoja softvera i upravljanja projektom.

SCRUM može biti korišten za nove projekte ili za aktivnosti koje su već u toku. Originalno je dizajniran za korištenje u softverskom inženjeringu ali njegova primjena nije ograničena te može biti korišten za projekte razvoja informacionih sistema, kao i za upravljanje projektima iz bilo koje oblasti.

#### 3.1. Agilne manifesto

SCRUM je zasnovan na vrijednostima i principima Agilnog Manifesta<sup>1</sup>, koji predlaže drukčiji način upravljanja softverskim projektima. Agilni manifest je nastao u periodu od 11. do 13 februara 2001. godine sa ciljem pronalaska boljeg načina razvoja softvera i pomoću u razvoju softvera drugima. Potpisalo ga je 17 učesnika:

---

<sup>1</sup> <http://www.agilemanifesto.org/>

Kent Beck	James Grenning	Robert C. Martin
Mike Beedle	Jim Highsmith	Steve Mellor
Arie van Bennekum	Andrew Hunt	Ken Schwaber
Alistair Cockburn	Ron Jeffries	Jeff Sutherland
Ward Cunningham	Jon Kern	Dave Thomas
Martin Fowler	Brian Marick	

Osnovne 4 vrijednosti koje određuje Agilni maifest su prikazana u tabeli 1.

Tabela 2: Osnovne vrijednosti Agilnog manifesta

U fokusu	Pored
Pojedinac i odnosi među njima	Procesi, tehnologije i alati
Funkcionalan softver	Potpuna dokumentacija
Saradnja sa klijentom	Pregovaranje
Fleksibilnost na promjene	Praćenje plana

1. Agilni manifest se fokusira na pojedince i njihovu međusobnu interakciju u odnosu na procese, metodologije i alate. Važno je da su članovi tima visoko kvalificirani za zadatke koji su im povjereni i da su kreativni. Također je važno da članovi tima imaju osjećaj da rade ono što vole i sretni su kada projekt uspije. Svakako je potrebno imati definirane procese i alate, ali je puno važnije imati sposobne i kreativne članove tima.
2. Agilni manifest u prvi plan stavlja funkcionalan softver ispred potpune dokumentacije. Mnogo je važnije imati funkcionalan i koristan softver od potpune dokumentacije softvera sa greškama.
3. Agilni manifest se fokusira na saradnju sa klijentom i odnos prema klijentu je partnerski, a pregovori o ugovoru će se odvijati u partnerskoj atmosferi, a ne kao dvije suprotstavljene strane. Klijent je dio razvojnog tima kako bi se projekt mogao prilagoditi promjenama.
4. Jedan od osnovnih i važnijih principa Agilnog manifesta je fleksibilnost na promjene, te Agilni manifest fleksibilnost na promjene stavlja ispred praćenja plana projekta, što

je osnovna karakteristika PMBoK (engl. Project Management Book of Knowledge) projekata.

Agilni manifest je okvirno određen sa prethodno navedene 4 osnovne vrijednosti, a detaljnije određen sa sljedećih 12 principa:

1. Zadovoljstvo klijenta je najviši prioritet i postiže se ranom i kontinuiranom isporukom korisnog softvera.
2. Izmjene zahtjeva su prihvatljive, čak i u podmaklim fazama projekta. Agile koristi izmjene u svrhu povećanja konkurentske prednosti klijenta.
3. Ispravan softver isporučuje se u roku od nekoliko sedmica do nekoliko mjeseci dajući prednost kraćim intervalima.
4. Svakodnevna saradnja korisnika i softver inženjera je neophodna.
5. Projekte realiziraju timovi motiviranih pojedinaca. Potrebno je osigurati im adekvatno okruženje i podršku te povjeriti obavljanje posla.
6. Najefikasniji i najučinkovitiji način informiranja razvojnog tima i informiranja unutar razvojnog tima je razgovor licem u lice.
7. Ispravan softver je osnovna mjera napretka.
8. Agile zagovara održivi razvoj softvera. Investitori, softver inženjeri i korisnici bi trebali biti u stanju neograničeno dugo održavati stalni tempo rada.
9. Agile pospješuju kontinuirana briga o tehničkom kvalitetu softvera i dobar dizajn.
10. Od ključne važnosti je pojednostavljivanje, čime se povećava obim posla kojeg ne treba raditi.
11. Najbolja arhitektura, odredbe i dizajn proizilaze iz samoorganizirajućih timova.
12. Redovnom retrospektivom tim teži da postane učinkovitiji, te u skladu s time prilagođava i podešava svoje postupke.

Prvi princip Agilnog manifesta nastoji najvažnije funkcionalnosti za klijenta isporučiti što prije, kako bi klijent što ranije počeo eksploatirati dio vrijednosti softvera, te stekao osjećaj koliko cjelokupan proizvod može biti koristan. Dakle, cilj je da klijent vrlo rano osjeti zadovoljstvo korištenja softvera i da željno iščekuje svaki novi inkrement proizvoda. SCRUM prvu isporuku inkrementa proizvoda spremnog za produkciju radi najkasnije mjesec dana nakon početka realizacije projekta.

Drugi princip Agilnog manifesta naglašava fleksibilnost procesa upravljanja projektom i mogućnost izmjene zahtjeva u toku realizacije projekta, kako bi softverski proizvod u svakom momentu bio prilagođen zahtjevima i promjenama u okruženju poslovanja za koje se proizvodi nova vrijednost, novi softverski proizvod. Specifikacija zahtjeva opisuje osobine i kapacitet softverskog proizvoda ili proizvoda općenito, jer SCRUM se može primjeniti i na projekte iz oblasti koje su različite od softverskog inženjeringa i informacijsko-komunikacijskih tehnologija općenito.

Treći princip Agilnog manifesta naglašava važnost ispravnog softvera, odnosno verifikovanog (funkcionalni zahtjev implementiran), validiranog (funkcionalni zahtjev ispravno implementiran) i testiranog softvera, te ponovo insistira na kratkom roku isporuke sa ciljem da korisnik što prije počne koristiti inkrement proizvoda, koji predstavlja poslovnu vrijednost, i da što prije SCRUM tim dobije povratnu informaciju o inkrementu proizvoda, te na taj način umanju rizik od neuspjeha. Zahtjeve je moguće promijeniti, ukoliko nisu ispravno određeni, ali određeni datum predaje inkrementa proizvoda je nepromjenjiv.

Četvrti princip je stalna saradnja sa softver inženjera sa klijentom, kako bi se uvijek mogle pojasniti eventualne nedoumice o osobinama proizvoda. Iluzorno je očekivati da klijent svakodnevno radi sa SCRUM timom, jer je u najmanju ruku neobično da neko plaća izradu naručenog proizvoda i još da sudjeluje u njegovoj izgradnji. To bi se moglo poistovjetiti sa situacijom kada klijent odveze automobil na popravku u servis i serviser ga zamoli da mu pomogne u toku popravke, a uredno mu po završetku posla izda i naplati račun. U realnom svijetu softverske industrije u timu će najmanje jedan član igrati ulogu klijenta i redovno u skladu sa potrebama komunicirati sa klijentom. Ulogu klijenta u timu najčešće ima *product owner* ili sistem analitičar.

Peti princip definira potrebu da tim koji realizira projekt bude multidisciplinarni, te da članovi tima budu motivirani za posao koji obavljaju. Za motivaciju poslodavac im osigurava adekvatno okruženje i sve potrebne uvjete. Sve češće susrećemo softverske firme koje osim neophodnih resursa i uvjeta za rad uposlenicima osiguravaju i prostor za zabavu i organiziraju društvene aktivnosti, kako bi svaki pojedinac u što većoj mjeri podigao stepen motivacije i razvio timski duh. Peti princip Agilnog manifesta također naglašava važnost povjerenja u članove tima za posao koji im je povjeren. Svakako je preduvjet za visoki stepen povjerenja i osobine članova tima koje definira prva vrijednost Agilnog manifesta, kao što je visoka kvalificiranost članova tima za posao koji obavljaju. U timu nema šefova i uspjeh je zajednički.

Šesti princip definira kao najefikasniji oblik komunikacije, komunikaciju licem u lice (engl. face to face), te kako će u nastavku poglavlja biti navedeno, SCRUM koristi tri vrste sastanaka (dnevni, prije završetka svakog sprinta i retrospektive sprinta po završetku svakog sprinta). Sastanci takođe koriste za dobivanje povratne informacije (engl. feedback) o proizvodu i procesima koji se koriste.

Sedmi princip utvrđuje kako indikator napretka i uspjeha projekta ispravnost svakog inkrementa softvera. Insistira se na provjeri ispravnosti svakog inkrementa softverskog proizvoda. Funkcionalni zahtjevi su određeni karakteristikama i mogućnostima proizvoda, te prioritet se uspostavlja po poslovnim vrijednostima i rizicima. Dakle, funkcionalni zahtjevi koji predstavljaju veću *poslovnu vrijednost* i predstavljaju veći *rizik* će biti implementirane prije funkcionalnosti koje su ocijenjene da imaju manju poslovnu vrijednost.

Osmi princip Agilnog manifesta određuje da tim radi u ciklusima, te da u jednakim vremenskim intervalima (engl. time box) radi isporuku korisnih inkremenata proizvoda, što održava stalni tempo rada tima i smanjuje mogućnost članovima tima da budu neproduktivni.

Deveti princip Agilnog manifesta insistira na stalnoj brizi o tehničkom kvalitetu softvera i dobrog dizajna. Izlaz iz svake iteracije je dio softverskog proizvoda spremnog za produkciju, dakle upravo onakav proizvod kakav je klijent naručio i spreman za eksploataciju.

Deseti princip Agilnog manifesta teži optimizirati količinu posla pojednostavljivanjem procesa, karakteristika proizvoda, tehnika itd.

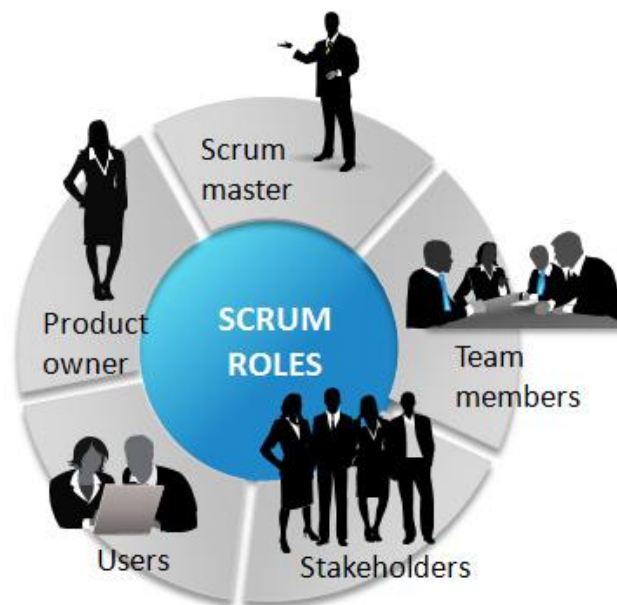
Jedanaesti princip Agilnog manifesta nastoji u cjelosti eliminirati hijerarhijsku strukturu tima koji realizira projekt. U SCRUMtimu postoje različite uloge, ali nema šefova. Svi članovi tima mogu donositi odluke koje dijele sa ostatkom tima. Tim se samoorganizira i svi članovi tima bez obzira na ulogu pomažu jedni drugima i na taj način se ostvaruje najbolja arhitektura, odredbe i dizajn.

Dvanaesti princip Agilnog manifesta definira potrebu retrospektive svakoga sprinta. Retrospektiva treba da sadrži detaljnu analizu korištenih procesa u toku sprinta i na osnovu dobivenih rezultata donosi odluke o redizajniranju korištenih procesa ili promjeni procesa u cjelosti sa ciljem optimizacije produktivnosti.

## 3.2. Uloge u SCRUM projektu

Jedan projekt može realizirati jedan ili više SCRUM timova. Koordinacija SCRUM timova koji rade na kompleksnim projektima je zahtjevan proces. U nastavku će biti predstavljene tri moguće uloge u jednom SCRUM timu.

SCRUM tim čine pojedinci koji mogu imati ulogu: vlasnik proizvoda (engl. product owner), SCRUM master, klijenti, budući korisnici, razvojni tim i sve zainteresirane strane za projekt općenito (slika 12).



Slika 12: SCRUM uloge

SCRUM tim je samoorganizirajući i često se kaže da članovi SCRUM tima nisu resursi projekta već „*decisions makers*“. Članovi tima znaju svoje zadatke i svako se organizira na način koji mu najviše odgovara. Ovo ne znači da uopšte ne postoje rukovodioci. Jedna firma ili organizacija ima menadžere i vođu projekta čija zaduženja su: angažiranje novih kadrova, ugovaranje, uspostavljanje, kontrola postizanja ciljeva, upravljanje materijalnim resursima, raspoređivanje resursa projektima itd.

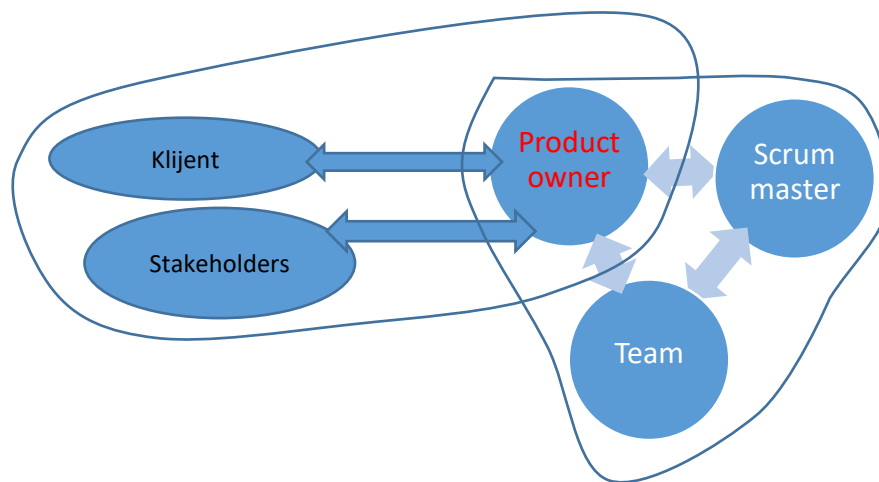
### 3.2.1. Vlasnik proizvoda (engl. Product owner)

Vlasnik proizvoda ili *product owner* je član tima sa najvećim i značajnim iskustvom i predstavlja glas klijenta u timu po zahtjevima četvrtog principa Agilnog manifesta. Ima viziju

proizvoda u cijelosti i razumije poslovnu vrijednost osobina proizvoda te može valorizirati osobine proizvoda sa ciljem određivanja prioriteta i redoslijeda njihove realizacije.

On određuje osobine i funkcionalnosti proizvoda, te ih po završetku implementacije također validira.

*Product owner* ne predstavlja nikakvog rukovodioca tima, već je član tima koji surađuje sa svim ostalim članovima tima. Jedna od njegovih važnih uloga je i uspostavljanje komunikacije (Slika 13) između sa jedne strane razvojnog tima, a sa druge strane klijenata, korisnika i svih zainteresiranih (stakeholders).



Slika 13: Centralna uloga vlasnika proizvoda između razvojnog tima i svih drugih zainteresiranih strana

Da bi pojedinac uspješno obavljao ulogu vlasnika proizvoda, potrebno je da ima dobre komunikacijske sposobnosti i sposobnost jasnog izražavanja zahtjeva. Ovu funkciju obavlja samo jedna osoba i sva ključna pitanja na jednom SCRUM projektu se postavljaju vlasniku proizvoda.

Uloge vlasnika proizvoda su:

- Određuje koje karakteristike i funkcionalnosti proizvoda će biti implementirane i kojim redoslijedom;
- Definiira željene rezultate projekta;
- Prioritizira funkcionalnosti u skladu sa tržišnom vrijednošću;
- Određuje datum puštanja proizvoda u produkciju;
- Prihvata ili odbija radne rezultate i
- Osigurava profitabilnost projekta (engl. Return of Investment - ROI).

### 3.2.2. SCRUM Master

SCRUM Master je vođa ili lider tima, ali ne rukovodilac, jer u SCRUM-u nema rukovodioca niti šefova. Odgovoran je za pravilno uvođenje i provođenje SCRUM-a, davanje podrške timu i vođenje tima kroz cjelokupni proces gradnje softvera, otklanjajući sve prepreke tima koje stoje na putu uspješnog realiziranja projekta.

Uloge SCRUM mastera su:

- Osigurava funkcionalnost i produktivnost tima;
- Omogućava tijesnu saradnju između svih uloga u timu;
- Otklanja prepreke;
- Štiti tim od vanjskih utjecaja;
- Osigurava poštivanje procesa, uključujući pozive na dnevne SCRUM sastanke, analizu iteracija (sprintova) i planiranje sprintova;
- Započinje dnevni SCRUM sastanak.

SCRUM master treba da ima između ostalog savjetodavnu ulogu i da je na raspolaganju svim članovima razvojnog tima za pomoć koju će on lično pružiti ili će iznaći neko drugo rješenje. On ne bi trebao nikada obraćati se timu sa uputama šta treba da rade, već je spreman za sva pitanja tima.

SCRUM master također je odgovora za pomoć u prilagodbi cjelokupne organizacije SCRUM procesima.

### 3.2.3. Razvojni Tim

Razvojni tim je samoorganizirajući tim. Odgovoran je za dostavljanje softverskih cjelina na kraju svake iteracije. Bitno je naglasiti da softver koji je rezultat svake iteracije se može pustiti u produkciju. Idealna brojnost razvojnog tima je  $7\pm 2$  člana. Svaki član tima ima različite osobine (znanja, kompetencije i vještine) što tim čini multidisciplinarnim. Znanje i sposobnosti članova tima bi se moglo predstaviti kao veliko slovo 'T'. Horizontalna linija na slovu 'T' predstavlja opće znanja člana tima o svim potrebnim tehnikama za izgradnju softverskog proizvoda, a vertikalna ekspertizu u određenoj oblasti, npr. ekspert za optimizaciju korisničkog



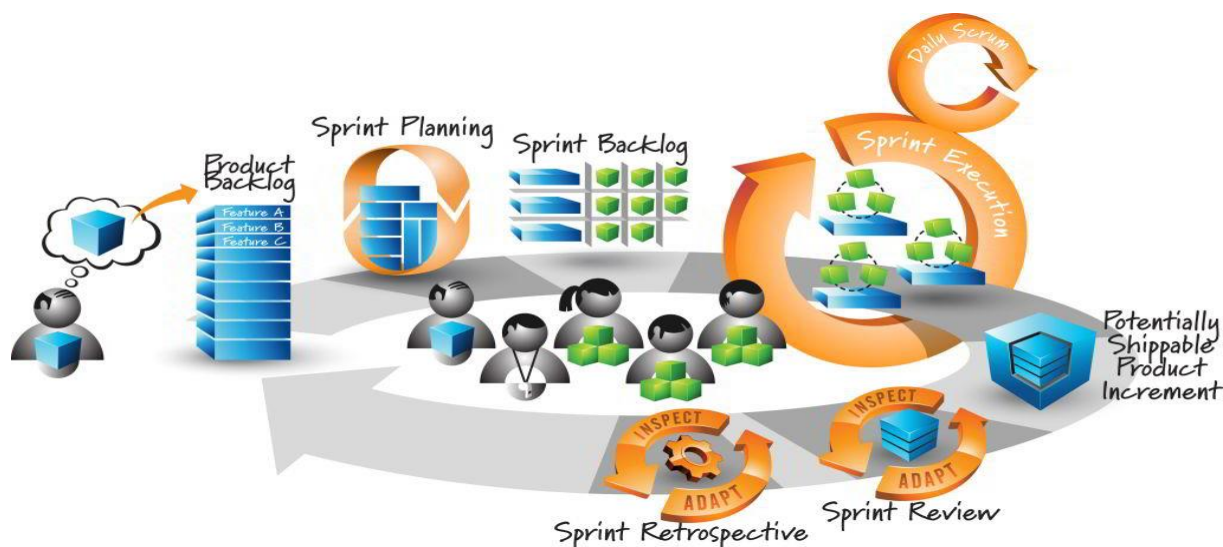
iskustava – UX. Kao što je već rečeno, razvojni tim na SCRUM projektu je sposoban samostalno organizirati se. Funkcije nisu striktno podijeljene, članovi tima mogu preuzeti različite funkcije, ovisno o njihovoj ekspertizi i sklonostima.

Tim bira ciljeve za sprint/iteraciju i određuje radne rezultate. Tim uživa apsolutno povjerenje te članovi tima imaju pravo da urade sve što je potrebno da bi se došlo do cilja iteracije. Kao što je već rečeno, samoorganiziraju svoj rad svjesni da postavljeni ciljevi iteracije moraju biti realizirani do dana koji je definiran za završetak iteracije. Razvojni tim je svjestan da bez obzira na neočekivane prepreke i probleme, definirani datum predaje inkrementa proizvoda je nepromjenljiv. Po završetku sprinta demonstriraju radne rezultate vlasniku proizvoda i ostalim zainteresiranim stranama.

### 3.3. Agilno planiranje i procjena vremena

Globalna pregled iterativnog ciklusa SCRUM projekta je predstavljen na slici 3.

Razvoj softverskog proizvoda počinje kada *product owner* ima jasnu viziju proizvoda u cijelosti. Plava kocka predstavlja finalni proizvod. *Product backlog* pohranjuje osobine finalnog proizvoda koje nisu fiksne, već promjenjive i u toku razvoja proizvoda mogu se dodavati nove ili brisati postojeće. Dakle, definicija karakteristika proizvoda ili funkcionalnosti je također cikličan proces. Ovisno o poslovnim vrijednostima i rizicima uspostavlja se redoslijed implementacije karakteristika proizvoda. Implementacija se radi u sprint-ovima. Tokom sprinta održavaju se dnevni sprint sastanci, a nakon sprinta „sprint review“ sastanak sa ciljem revidiranja i analize proizvoda sprinta, te prilagodbe sljedećih sprintova da bi konačni proizvod bio što kvalitetniji. Rezultat svakog sprinta je funkcionalan dio proizvoda (zelene kocke na slici 14).



Slika 14: Grafički pregled SCRUM-a [5]

Po završetku sprinta održava se također „*sprint retrospective*“ sastanak gdje se analiziraju procesi korišteni u toku sprinta, a ne više proizvod. Cilj je optimizirati procese i optimizirati rezultate projekta.

Kompletan proces je cikličan, što ilustrira siva kružna strelica na slici 3.

U nastavku će biti detaljno objašnjen svaki od artefakata SCRUM-a.

### 3.4. Korisničke priče (engl. User Story)

Korisničke priče ili *user stories* predstavljaju zapravo opis proizvoda iz perspektive budućih korisnika. SCRUM preporučuje definiciju korisničkih zahtjeva u terminima osobina proizvoda. Skup korisničkih priča predstavlja specifikaciju zahtjeva proizvoda, gdje svaka korisnička priča predstavlja jednu osobinu proizvoda. Zahtjevi za proizvod koji će riješiti poslovni problem se prikupljaju od svih zainteresiranih strana (službenika, menadžera, i ostalih članova ekipe).

Korisničke priče se u toku evolucije projekta mogu mijenjati, dodavati ili brisati.

### 3.5. Product Backlog

*Product backlog* čine sve korisničke priče sa utvrđenim prioritetima. Veći prioritet i prednost u implementaciji imaju važnije korisničke priče. Inicijalnu verziju Product Backlog-a kreira vlasnik proizvoda, ali je bitno naglasiti da svako može dodavati nove funkcionalnosti u

skladište. Prioritet funkcionalnosti određuje vlasnik proizvoda u skladu sa poslovnom i marketinškom vrijednošću rezultata implementacije korisničke priče.

Lista funkcionalnosti može sadržavati bilo šta, defekte, poboljšanja, projekte, probleme, rizike itd. Da bi se funkcionalni zahtjevi razlikovali od drugih elemenata u Product Backlogu, potrebno je da budu izraženi na način da izražavaju vrijednost za krajnjeg korisnika, npr.: „Kao korisnik, želim da klikom na dugme potvrdim unos novog računa“. Nefunkcionalni zahtjevi se također mogu nalaziti u Product Backlogu ali napisani na drukčiji način da bi se izrazila njihova tehnička ili druga priroda.

### 3.6. Prioritizacija funkcionalnosti

Nakon dodavanja funkcionalnosti u Product Backlog, potrebno je odrediti prioritete u skladu sa poslovnim vrijednostima i rizicima proizvoda. Vlasnik proizvoda je zadužen za određivanje prioriteta funkcionalnosti. Funkcionalnosti koje se nalaze na kraju liste prioriteta možda i neće nikada biti urađene i često su to stvari koje su nedovoljno ili pogrešno definirane. Sve ideje/funkcionalnosti koje dolaze od vlasnika proizvoda ili članova tima moraju biti poredane po prioritetu te tako jasno ukazivati na važnost pojedinih komponenti u odnosu na cijelu sliku proizvoda. Product backlog se može promatrati kao stog prioriteta (engl. stack) gdje se na vrhu nalaze funkcionalnosti sa većim prioritetom, a na dnu one sa najnižim prioritetom.

Jedna od velikih prednosti SCRUM-a u odnosu na druge metode je to što je prioritet zadataka uvijek jasno vidljiv i oslikava trenutne potrebe, tako da članovi tima nemaju nejasnoća u pogledu toga šta je potrebno raditi i kojim redoslijedom. Sve to omogućava veći fokus na trenutne zadatke svakog pojedinca i mogućnost promjene prioriteta tokom vremena.

Zadaci koji se nalaze na početku liste Product Backlog-a će definitivno biti urađene u bliskoj budućnosti pa je potrebno uložiti dodatni trud da se to što bolje razumije i objasni. Te funkcionalnosti moraju biti veoma dobro definirane, da bi se omogućilo timu da odmah krene sa radom. Vrlo je važno da tim zna kuda ide prije početka implemetacije, kako bi bio efikasan i ne bi lutao. Svaka funkcionalnost mora biti napisana na način da se može isporučiti krajnjem korisniku kao zasebna cjelina. SCRUM metoda angažira resurse onda kada je to potrebno, tako da ni pojedinačne funkcionalnosti nisu objašnjene detaljno sve do trenutka kada tim treba početi sa radom na tim cjelinama.

### 3.7. Procjena veličine funkcionalnosti

Funkcionalnosti iz *Product Backlog*-a se obično procjenjuju bodovima a ne mjernim vremenskim jedinicama. Procjena bodovima se odvija na način da se određenoj funkcionalnosti dodjeljuje više ili manje bodova u odnosu na količinu posla koju je potrebno uraditi. Ovakav način procjenjivanja je veoma neobičan ali se u praksi pokazalo da je veoma efikasan i fleksibilan.

Koristeći ovakav način procjene vremena, razvojnim timovima se više ne postavlja pitanje o tome koliko će dugo trajati razvoj na određenoj funkcionalnosti, već njena veličina.

SCRUMtimovi obično koriste neki brojčani sistem da bi označili veličinu funkcionalnosti, najčešće korišteni sistem je Fibonacci niz (1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987 ...)

Timovi obično iz ovog niza koriste brojeve od 1 do 21 da bi označili veličinu funkcionalnosti, i vremenom uče iz svojih grešaka, tako poboljšavajući vještine procjene veličine funkcionalnosti i prepoznavanja sličnih zadataka. S obzirom da se u Scrum-u koristi iterativni pristup, na početku svake iteracije/sprint-a, tim zajedno procjenjuje funkcionalnosti koje će biti urađene u bliskoj budućnosti, dodjelom ocjene veličine svim funkcionalnim cjelinama iz skladišta zadataka. Nakon što razvojni tim ponudi procjene veličine za funkcionalnosti iz trenutne iteracije, vlasnik proizvoda ima mogućnost uvida u date procjene da bi eventualno promijenio prioritet pojedinih funkcionalnosti. Ukoliko tim procjeni da realizacije jedne funkcionalnosti traje duže od jednog sprinta, funkcionalnost se dijeli na manje dijelove koji mogu biti realizirani u vremenu trajanja sprinta.

### 3.8. Planiranje iteracije/sprint-a

Prva odluka koja se mora donijeti prije planiranja prve iteracije/sprinta je njeno trajanje, u većini projekata je to period od jedne sedmice do mjesec dana (četiri sedmice). Kao zlatna sredina za brze i agilne timove se uzima period od dvije sedmice.

Nakon odluke vremenskog trajanja jedne iteracije/sprinta, biraju se funkcionalnosti koje se planiraju završiti u tom vremenskom periodu. Funkcionalnosti biraju direktno članovi razvojnog tima u skladu sa procjenama veličine i prioritetima. Nakon nekoliko iteracija, tim postepeno uči o količini posla koja se može realno uraditi za vrijeme jedne iteracije a *product owner* ima realan pokazatelj njihove brzine.

Brzina tima se jednostavno mjeri sabiranjem bodova za sve funkcionalnosti koje se završe u jednoj iteraciji. Ovaj pokazatelj obično je promjenjiv na početku svakog projekta, da bi postepeno postao stabilan i dao mogućnost vlasniku proizvoda da procjeni vrijeme potrebno da se cjelokupan proizvod pusti u produkciju.

### 3.9. Planiranje zadataka

Nakon što se funkcionalnosti visokog nivoa ocjene na način koji je objašnjen u prethodnom poglavlju, članovi tima dijele funkcionalnosti na manje zadatke sa tehničkim detaljima implementacije koji se ovaj put procjenjuju u satima. Razlika između ove metode u odnosu na druge je to da se zadaci procjenjuju onda kada je to potrebno, pred početak svake iteracije. Također, resursi za realizaciju zadatka ili aktivnosti se mobiliziraju na početku iteracije/sprinta. Svaki zadatak koji se dodaje u okviru određene funkcionalnosti mora sadržavati tehničke detalje implementacije i biti u skladu sa ostalim zadacima, tako kompletirajući jednu cjelinu kao završenu.

Procijenjeno vrijeme trajanja svakog pojedinačnog zadatka (npr. „kreirati bazu podataka“), ne bi trebalo biti duže od 24 sata, a preporučeno vrijeme je do 8 sati.

### 3.10. Iteracije i praćenje progressa projekta

Bitno je naglasiti da SCRUM propisuje agilni pristup izgradnji softvera, ali ne ulazi u konkretne inženjerske detalje, tako da se obično kombinira sa nekom inženjerskom disciplinom kao što je UP (Unified process), XP (Extreme programming), RUP (Rational unified process) i sl.

SCRUM ne objašnjava način na koji će softver biti izgrađen, već je njegova uloga da objasni agilnu praksu koja dovodi do poboljšanja procesa i uspješnog završetka projekta.

Konkretna izgradnja softvera u SCRUM projektu se odvija u iteracijama/sprint-ovima, a jedan od najvažnijih aspekata razvoja je kvalitetna saradnja između svih članova tima uključenih u projekat. Da bi SCRUM timovi radili na odgovarajući način potrebno je primjeniti sljedeće agilne principe:

- SCRUM timovima je potrebno omogućiti samostalno donošenje odluka (engl. decision makers), što doprinosi osjećaju odgovornosti.
- Trajanje iteracije je fiksno, funkcionalnosti dodijeljene jednoj iteraciji ili sprint-u se mogu dodavati i/ili brisati samo u zamjenu za druge funkcionalnosti.

- Funkcionalnost je završena samo ako je 100% kompletirana, veoma je bitno da se uzima samo po jedan zadatak i završi do kraja.
- Testiranje softvera je ugrađeno u cijeli životni vijek projekta, idealno koristeći TDD (Test Driven Development ).
- Nakon što tim usvoji plan za iteraciju, ne smije biti izmjena i distrakcija u toku rada, te plan mora biti realiziran u predviđenom vremenu za sprint. Ovo se najviše odnosi na izmjene koje inicira vlasnik proizvoda.

### 3.11. Dnevni SCRUM sastanak

Održavanje dnevnog SCRUM sastanka je veoma važna aktivnost u toku razvoja softvera. Sastanak se održava svaki dan na rutinski način u isto vrijeme i na istom mjestu. Cijeli tim prisustvuje sastanku, SCRUM Master započinje sastanak a svaki član tima pojedinačno mora odgovoriti na nekoliko pitanja:

1. Šta je urađeno od zadnjeg sastanka? (jučer)
2. Šta se planira uraditi do sljedećeg sastanka? (sutra)
3. Da li postoje prepreke u procesu progressa projekta? (prepreke)

Ako postoje problemi o kojima je potrebno detaljnije razgovarati, to se obično radi nakon SCRUM sastanka da bi se njegovo trajanje ograničilo na maksimalno 15 minuta. Osnovni cilj sastanka je preuzimanje odgovornosti svakog člana tima za sopstveni rad stvaranjem pritiska, jer je svakako neugodno izvijestiti tim da na primjer od posljednjeg sastanka niste ništa uradili. Cilj sastanka je također sinkronizacija posla i samog tima, kao i revizija i adaptacija posla.

Na početku svakog sprinta ovaj sastanak se koristi za planiranje sprinta. Planiranje sprinta se radi u sljedećim koracima:

1. Tim u dogovoru sa *vlasnikom proizvoda* dogovara koje su aktivnosti prioritetne za realizaciju.
2. Iz skupa dogovorenih prioritetnih aktivnosti tim uzima jednu aktivnost.
3. Tim dijeli aktivnost na manje zadatke.
4. Tim procjenjuje vrijeme potrebno za realizaciju aktivnosti.
5. Ukoliko se aktivnost može realizirati u kraćem vremenu od trajanja sprinta, tim uzima sljedeću po prioritetu aktivnost i ponovo izvršava korak 3 i sljedeće korake algoritma.

6. Ukoliko je tim planirao dovoljno posla za sprint, završava se planiranje sprinta.

### 3.12. Sprint review

Sprint review je sastanak koji se radi prije završetka sprinta. Cilj je analizirati šta je to bilo dobro, a šta loše u toku sprinta. Učesnici su svi članovi tima: stakeholders, vlasnik proizvoda, scrum master, svi članovi razvojnog tima, predstavnici klijenata i budućih korisnika, kao i druge zainteresirane strane ako postoje. Cilj je revidirati do sada urađeni dio proizvoda i na osnovu feedback-a prilagoditi sljedeće sprintove da bi konačni proizvod bio što kvalitetniji. Obično se na ovom sastanku prezentira urađeni dio proizvoda (demo).

Svrha Sprint Review sastanka je:

1. Članovi tima mogu pokazati šta su postigli i prezentirati njihov doprinos proizvodu.
2. Menadžmentu se na vrijeme prezentira šta je urađeno, što omogućava pravovremenu reakciju.
3. Pomaže timu da ostane fokusiran na ciljeve i rokove trenutne iteracije, jer demo veoma jasno pokazuje šta jeste ili nije urađeno.

### 3.13. Praćenje progresu i burndown grafikon

U tradicionalnim softverskim projektima testiranje se radi na samom kraju projekta, s toga je veoma teško odrediti kvalitet softvera prije samog kraja, kao ni procijeniti ukupni broj nedostataka koji će biti pronađeni.

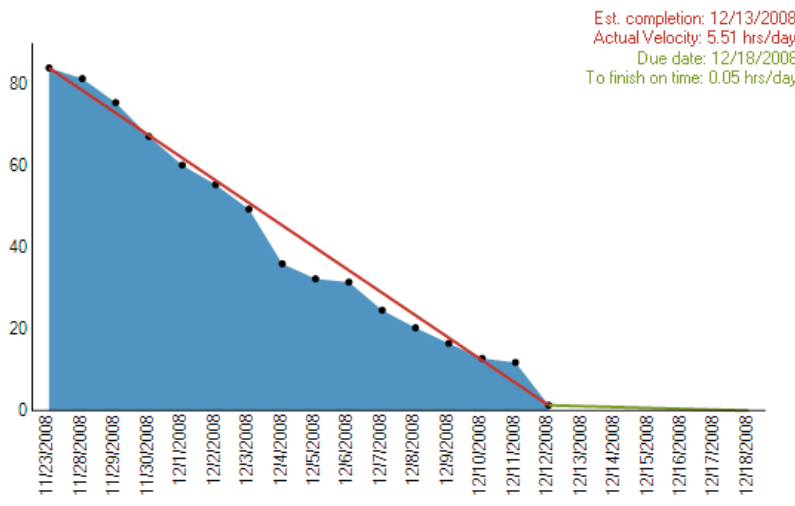
Agilni projekti integriraju testiranje u projekat od samog početka, tako da se navodno kompletirane cjeline smatraju završenim. Vlasnik proizvoda je uključen na dnevnoj osnovi u razvoj softvera i može jasno procijeniti da li treba napraviti promjene u toku procesa.

Svi agilni principi i dnevni SCRUM sastanci služe tome da se na jasan način može ocijeniti kvalitet i cjelovitost proizvoda.

Burndown grafikon je dodatni alat koji uveliko pomaže u procesu praćenja progresu na jednom SCRUM projektu. Omogućava praćenje dnevne produktivnosti. Nakon što su dnevni zadaci, koji se nalaze u sklopu neke funkcionalnosti, ocjenjeni sa brojem sati koji su potrebni da se to završi, sve to biva jasno prikazano na Burndown grafikonu. Na osi X prikazani su dani, a na osi Y ostatak posla u satima.

Članovi tima su dužni da svakodnevno ažuriraju broj sati koji su proveli na nekom zadatku i preostali broj sati za koji se procjenjuje da će funkcionalnost biti završena.

Navedeni grafikon prikazuje procijenjeno vrijeme u toku iteracije i realno vrijeme koje je potrebno da se završe sve odabrane funkcionalnosti sprint-a, razlika između ovih vremena jasno pokazuje da li tim kasni ili će uspjeti završiti sve aktivnosti na vrijeme. Rezultirajuća grafika se naziva burndown brzina (engl. burndown velocity) i teži 0 (slika 15).



Slika 15: Primjer burndown grafikona

### 3.14. Retrospektiva iteracije/sprint-a

Nakon Sprint Review sastanka se obično održava Sprint Retrospective sastanak. Sastanku prisustvuju vlasnik proizvoda, scrum master i razvojni tim.

Članovi tima na ovom sastanku prave retrospektivu urađenog na sljedeći način:

1. Osvrt na konačni burndown grafikon.
2. Analiza brzine tima.
3. Diskusija o pozitivnim dijelovima iteracije.
4. Diskusija o negativnim dijelovima iteracije.
5. Odrediti šta će tim uraditi drukčije u odnosu na prethodnu iteraciju.



Na ovom sastanku za razliku od SCRUM review sastanka se revidiraju korišteni SCRUM procesi, a ne proizvod. Cilj ovog sastanka je optimizirati korištene procese sa ciljem optimizacije planiranih rezultata.

Prednost SCRUM-a kao agilne metode u odnosu na druge metode upravljanja projektima se ogleda u fleksibilnosti na promjene i jednostavnosti korištenja. SCRUM je brza metoda koja se u potpunosti slaže sa trenutnim trendovima u svijetu informacijskih tehnologija i općenito poslovanja, gdje dolazi do veoma brzih promjena i konstantnih turbulencija. Sve više razvojnih timova u svijetu prihvata SCRUM i druge agilne metode. Timovi inženjera na SCRUM projektu obično bilježe rast produktivnosti, klijenti su zadovoljniji i konstantno su uključeni u projekat što doprinosi kvaliteti finalnog proizvoda.

## 4. Microsoft Team Foundation Server (MS TFS)

MS Team Foundation Server 2015 je namijenjen kao centar na kojem razvojni tim radi na softverskom projektu. Također ima alate koji potpomažu agilnoj metodi razvoja softvera. MS Team Foundation Server 2015 se instalira na jednom centralnom serveru i svi ostali članovi grupe se mogu na njega spojiti kako bi mogli koristiti njegove funkcionalnosti. MS Team Foundation Server 2015 je moguće direktno koristiti s Microsoft Visual Studio verzijama, ali je također napravljena tako da ga se može koristiti i preko drugih platformi.

Timovi koji se bave razvojem softvera nastoje poboljšati proces razvoja softvera, kvalitetu softvera, te komunikaciju i dijeljenje znanja u okviru tima sa ciljem optimizacije efikasnosti i troškova. U tu svrhu koriste različite proizvode, ovisno o karakteristikama projekta.

Team Foundation Server (TFS) je proizvod koji objedinjuje alate, procese i smjernice, što omogućava bolju komunikaciju između članova tima i transparentnost projekata, a na jednom mjestu pruža sve alate za projektovanje, razvoj i testiranje softverskog koda, koji uključuju alate za kontrolu nad resursima, praćenje radnih zadataka, upravljanje projektima, upravljanje programskim kodom i izvještavanje. Ovi alati su namijenjeni prvenstveno za velike grupe korisnika. MS TFS je kvalitetan alat koji je Microsoft koristio prilikom razvoja Windows Viste i novog Office-a. Najveću korist od ovog alata imat će korisnici koji će pratiti stanja projekata, definisati zadatke razvojnom timu te kontrolisati izvorni kod.

MS TFS je vrlo pogodan za SCRUM agilni pristup razvoja softvera.

U nastavku ovog poglavlja će biti predstavljene najvažnije karakteristike MS TFS-a i primjer njegove upotrebe.

### 4.1. Instalacija

MS Team Foundation Server 2015 se može instalirati na sljedeće načine, ovisno o potrebama i mogućnostima grupe:

- Single server – u kojem se čitav sistem (uključujući MS SQL server) instalira na jednom server računaru. Ovakav način instalacije je pogodan za manje grupe i za pregled funkcionalnosti MS Team Foundation Server 2015.
- Dual server – u kojem se MS SQL server instalira na jednom „data tier“ serveru, a TFS (Team Foundation Server) se instalira na „application tier“ serveru. Ovakav način instalacije je pogodan za grupe srednje veličine.

- Multiple server configuration – TFS podržava rad više hiljada korisnika i za tu svrhu TFS se može instalirati na veliki broj server računara. U tom slučaju ponovo je odvojen „data tier“ kojeg čini MS SQL server ili Always On Availability Group i više server računara na kojim su instalirani TFS.

#### 4.2. Timski projekt (eng. Team project)

Nakon instalacije TFS-a i uspostavljanja barem jedne kolekcije timskih projekata, mogu se izrađivati timski projekti. Timski projekti predstavljaju najnižu organizacijsku jedinicu koja se može napraviti. Projektu se dodjeljuju korisnici koji će ga realizirati. Projekt služi kao centralna tačka na kojoj će se upravljati izvornim code-om, tj. čitav repozitoriji (ili više njih) će biti pohranjen u okviru projekta i operacije kao „potvrđi“ (eng. „commit“) i „ažuriraj“ (eng. update) će se sve vršiti nad njim. MS TFS također omogućava saradnju između članova tima. To znači da pohranjuje backlog i sve informacije o saradnji koje su predviđene agilnom metodom. Backlog sadrži korisničke priče, te se svakoj korisničkoj priči dodjeljuje jedan ili više zadataka projekta (engl. task). Svaka zadatak može biti u jednom od sljedećih stanja: „new“, „active“ ili „closed“.

#### 4.3. Funkcionalnosti Team Foundation Server-a

TFS pruža veliki broj novih alata (neki su se do sada koristili samo interno u Microsoftu) da bi se poboljšala komunikacija, rad i produktivnost razvojnog tima.

TFS se sastoji od sljedećih komponenti:

- **Team Foundation Version Control (TFVC):** služi za praćenje izvornog koda, projektnih datoteka, dokumenata, izvještaja i predložaka koji se pohranjuju centralizirano na serveru. Članovi tima preuzimaju datoteke sa servera i rade lokalno promjene. Promjene se grupiraju u skupove promijena („set changes“).
- **GIT** je distribuiran sistem kontrole verzija u kojem svaki član grupe na svom računaru drži repozitorij čitavog izvornog code-a (s tim da se na TFS-u čuva jedan centralni repozitorij). GIT pruža skoro sve mogućnosti kao i TFVC s tim da postoje neke razlike. Kad član grupe napravi promjene na izvorni code-u, on zapravo pravi „potvrde“ (eng. „commit“) koje može proslijediti ostalim članovima team-a (s tim da mora prije toga preuzeti sa servera najnovije potvrde ostalih članova i tad može eventualno doći do konflikta koje mora riješiti). Na svim računarima članova se čuvaju informacije o svim

potvrđama i tako se može pregledati tko je što radio bez da je potrebno biti povezan na TFS. Grananje nije zasnovano na putanji i svaki član grupe može lokalno napraviti koliko želi grana bez da ih mora prenijeti na centralni repozitoriji na TFS-u. Može se slobodno prebaciti s jedne grane na drugu, ali prije toga mora potvrditi promjene s trenutne grane.

- **Work Item Tracking(WIT):** omogućava praćenje work item-a projekta u cijelosti.
- **Team Portal:** koristi se za komunikaciju u okviru tima i olakšava posao voditelju projekta. Omogućava generirati izvještaje, pregledavati work item-e i dokumente.
- **Team Foundation Build:** omogućava automatizirati „build“ projekta (kompajliranje svih datoteka u jedinstven proizvod koji se može pokrenuti).
- **Team Reporting:** omogućava pogled na podatke projekata u TFS-u, građen na SQL 2014 Reporting Services.
- **Project Managment:** integracija sa Microsoft Office Project 2013 koristeći Visual Studio Tools za Office (VSTO).
- **Integration Services:** web servisi koji dozvoljavaju vanjskim alatima da se integriraju u okruženje TFS-a.
- **Team Explorer:** „shell“ za Visual Studio koji podržava funkcionalnosti TFS-a.

#### 4.3.1. Version Control

Version Control ne služi samo za kontrolu izvornog koda, već je integriran sa TFS policama, e-mail obavijestima, automatskim „build-om“ i WIT-om[12]. U sebi sadrži alate kao što su branching, merging, shelving, atomic check-in itd.

##### 4.1.1.1. Integrirani Check-In

U fokusu „version control“ je changeset. TFS koristi changeset kao logički kontejner za pohranjivanje svih podataka o jednom check-in (izvorni kod, metadata itd.). Klikom na „pending changes“ i „check-in“ neke datoteke, Team Foundation stvara novi changeset u source repozitoriju i pridružuje mu broj koji je jedinstven na tom serveru koji se povećava za jedan kod slijedeće promjene. Na ovaj način je omogućeno praćenje verzija.

Važno je naglasiti da je „check-in“ u TFS-u nedjeljiv, tj. check-in će se obaviti u cijelosti ili nikako (slično kao transakcija u sistemu za upravljanje bazama podataka).

Osobina nedjeljivosti „check-in“-a je korisna na primjer u slučaju greške u komunikacijskom kanalu jer nije potrebno brinuti da li je samo dio koda prenesen na server.

Promijene se također mogu poništiti na nivou skupa promjena (changeset). Dakle, moguće je napraviti rollback skupa promjena.

Npr. Potrebno je napraviti „check-in“ 50 datoteka. U toku prenosa na server 50 datoteka dogodi se greška u komunikacijskom kanalu. Server će se nakon uočene greške vratiti u stanje u kojem se nalazio prije početka izvršenja prekinutog „check-in“-a zbog greške u komunikacijskom kanalu [13].

#### *3.1.1.2. Workspaces*

Workspace je klijentska strana kopije datoteka sa TFS version control-a. Sve promjene na datoteci se odražavaju na klijentskom workspace-u i stavljaju se pod „pending changes“.

Sve promjene koje klijent napravi će ostalim članovima tima biti vidljive tek kada napravi check-in [14].

#### *3.1.1.3. Branching and merging*

TFS version control omogućava timu da određeni dio koda podijeli na više dijelova („razgrana“) sa ciljem paralelnog rada. Kada nastali dio (grana) se izgradi do stabilnog stanja, onda se može integrirati nazad u originalni kod.

Opcija „merging“ se koristi za spajanje dijelova koda pojedinih grana. Ova funkcionalnost je namijenjena većim timovima [15].

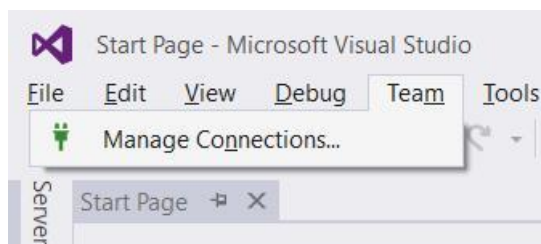
#### *3.1.1.4. Shelving*

Shelving se koristi da bi se nezavršeni dio posla pohranio na server. „Build“ aplikacije će biti neuspješan, ali promjene su pohranjene na server bez utjecaja na „build“. Pohranjene promjene na serveru su dostupne ostalim članovima tima [16].

### *4.4. Povezivanje na TFS*

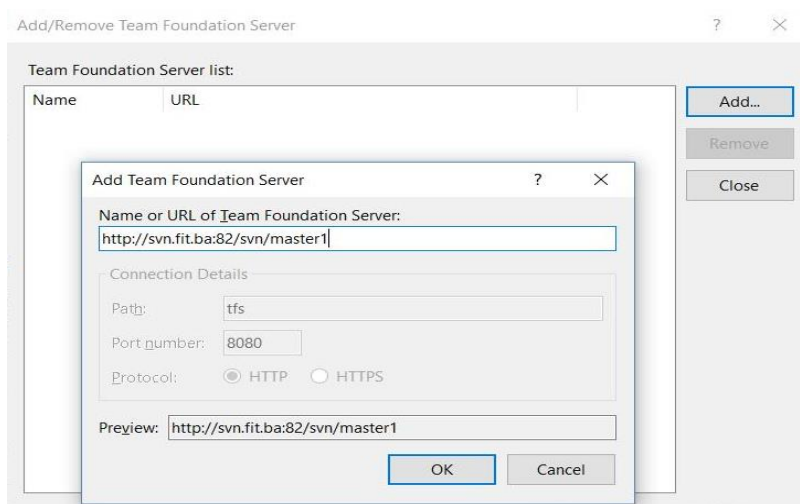
U nastavku će biti prikazan primjer povezivanja na TFS iz Visual studio-a, dodavanje novog projekta i pohranjivanje izmjena na projektu u TFS-u.

Za pokretanje TFS-a neophodno je prethodno pokrenuti **Visual Studio**. Nakon pokretanja Visual Studio-a potrebno je u glavnom meniju izabrati opciju **Team**, a zatim **Manage Connectios** (Slika 16).



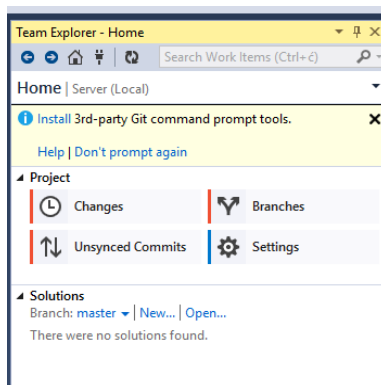
Slika 16: Pokretanje TFS-a

U prozoru koji se otvori (Slika 17) odabrati opciju **Servers**, a potom **Add** te upisati adresu servera na kojem će se nalaziti projekat koji će dijeliti članovi tima.



Slika 17: Upis adrese servera

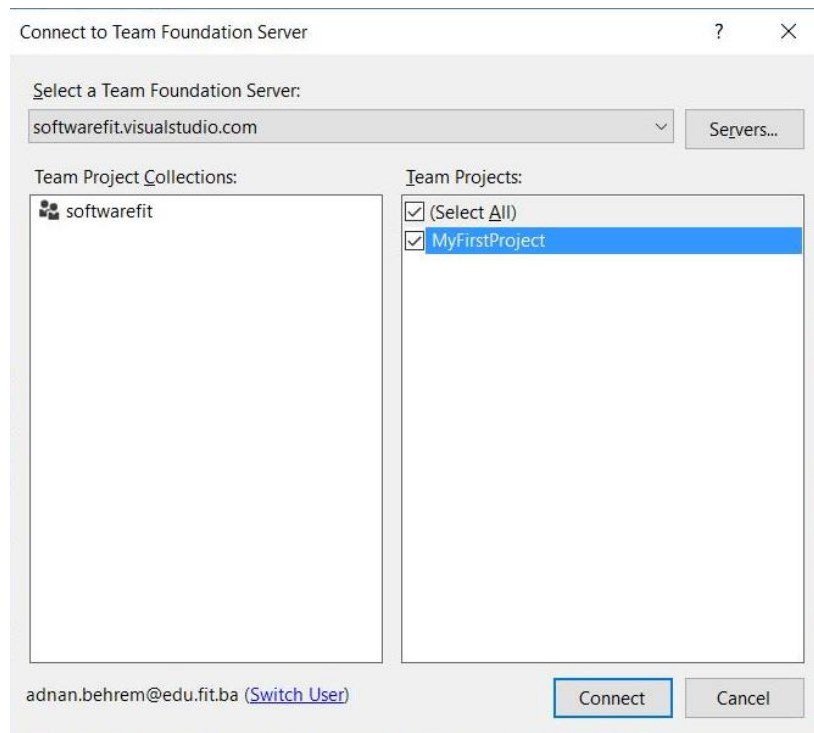
Ukoliko se kreira novi server, potrebno je odabrati opciju **New** i upisati naziv praznog direktorija koji će biti korišten kao server. Nakon toga otvara se Team explorer prikazan na slici 18.



Slika 18: Team explorer za novi prazan projekt

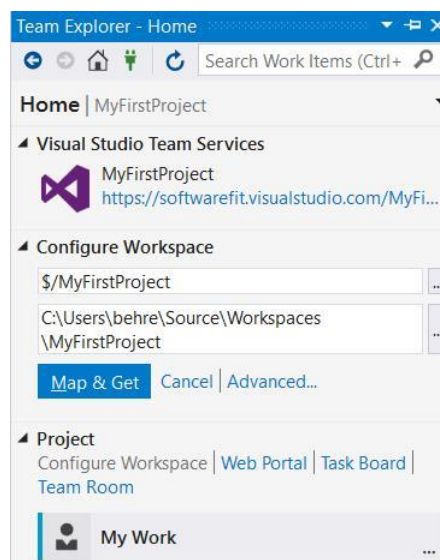
Ovisno o sigurnosnoj politici tima za pristup projektu potrebno je upisati **korisničko ime** i **lozinku**.

Nakon prijave na server odabrati projekt. Administrator koji je kreirao server dodjeljuje projekte korisnicima koji se prijave na server (Slika 19).



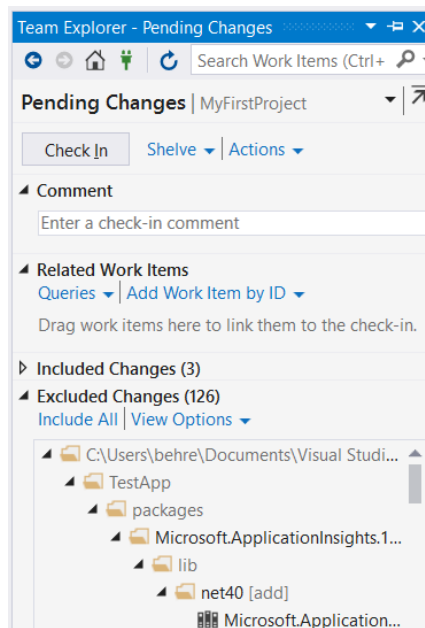
Slika 19: Odabir projekta

Nakon ovog koraka, sa desne strane, prikazat će se **Team Explorer** u kojem je potrebno izabrati opciju **Map & Get**, da bi se projekat učitao u TFS (Slika 12).



Slika 12: Učitavanje projekta u TFS

Poslije vremena potrebnog za učitavanja (ovisno u veličini datoteka projekta) sa desne strane, u **Solution Explorer-u**, mogu se vidjeti sve datoteke odabranog projekta. Nakon izmjene datoteke potrebno je desnim klikom odabrati opciju **Check-in** da bi TFS pohranio izmjene u repozitorij te da bi i drugi mogli pogledati izmjene koje su napravljene (Slika 20).



Slika 20: Spremanje izmjena

Može se zaključiti da je TFS, osim razvojnome timu vrlo koristan voditeljima projekta za praćenje napretka projekta. Ovo poglavlje predstavlja samo osnovu TFS-a i može pomoći budućim korisnicima pri odluci koji proizvod koristiti da bi osigurali konzistentan i kvalitetan razvojni proces koji vodi do kvalitetnog proizvoda.

Team Foundation Server 2015 je tek jedna komponenta alata Visual Studio 2015, koji nudi brojne mogućnosti za istraživanje i učenje novih komponenti.



## 5. Upravljanje projektnim ciklusom

Ovo poglavlje prikazuje projektni ciklus općenito, opisuje njegove faze i ima ulogu pomoćnog sredstva u upravljanju projektom. Generičke faze projekata općenito se mogu primijeniti i na softverske projekte iako je nastavku predstavljen generički ciklus upravljanja projektom. Predstavljen je opšti uvid u smisao i principe Upravljanja projektnim ciklusom i ukratko opisan način na koji projektni ciklus funkcionira.

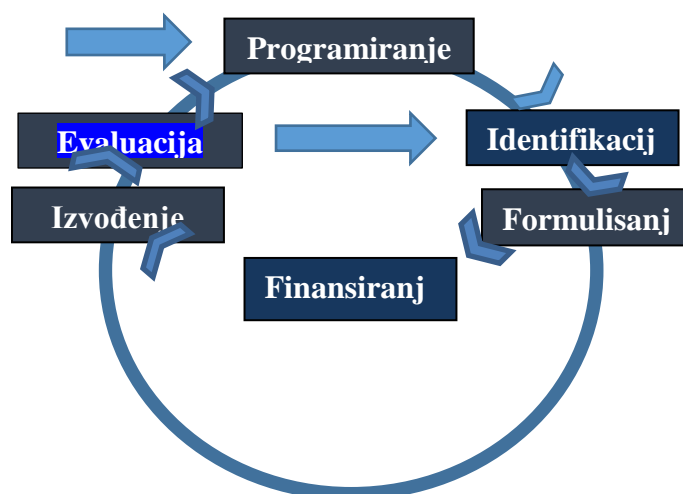
### 5.1. Projektni ciklus

Način na koji se projekti planiraju i izvode počiva na definiciji poznatoj kao projektni ciklus o kojem je bilo riječi u 1. poglavlju. Ciklus počinje identifikacijom ideje i nju razvija u plan rada (engl. work plan) koji je izvodljiv i može se evaluirati. Ideja projekta treba da bude usaglašena sa strategijom razvoja organizacije ili okružena kojem je namijenjena, te sve ključne odluke u u svim fazama životnog ciklusa projekta treba da budu u skladu sa strategijom razvoja okruženja kojem su namijenjeni ishodi projekta.

U softverskim projektima sama ideja treba da se razvije u specifikaciju korisničkih zahtjeva (engl. Software requirements specification - SRS) koji treba da sadrži detaljan plan, izgled i funkcionalnosti softverskog proizvoda. U okviru SCRUM-a to su „*user stories*“ i „*product backlog*“.

U softverskom projektu projektni partneri mogu biti timovi koji paralelno rade na razvoju istog softverskog proizvoda ili više verzija softverskog proizvoda, gdje je svaka verzija prilagođena jednom određenom tipu klijenta.

Generički gledano, projekt sadrži šest faza: osmišljavanje programa ili programiranje<sup>2</sup>, identifikacija, formulacija, finansiranje, izvođenje ili realizacija i evaluacija (Slika 21).



Slika 21: Projektni ciklus

Svaka od faza treba da bude prilagođena organizaciji na kojoj se realizira i za koju se realizira projekt.

Osnovne odrednice projektnog ciklusa su:

1. Ciklus definira ključne odluke, zahtjeve za informacijama i odgovornost u svakoj od faza.
2. Faze su progresivnog karaktera te svaka od njih mora biti u cjelini realizirana da bi se mogla uspješno realizirati naredna. Npr. u agilnom okviru razvoja softvera svaki sprint treba uspješno realizirati, te svaki sprint treba da rezultira funkcionalnim dijelom proizvoda da bi se krenulo sa realizacijom slijedećeg sprinta.
3. Ciklusi su osmišljeni tako da nakon evaluacije iskustava iz prethodnih projekata postanu ugrađiva pri osmišljavanju budućih programa i projekata. Npr. u agilnom okviru razvoja softvera se realiziraju sastanci nazvani retrospektiva iteracije.

Opis faza projektnog ciklusa je dat u nastavku.

Tokom faze **programiranja** analizira se stanje na tržištu ili na nivou okruženja gdje će se proizvod koristiti da bi se identificirali problemi, ograničenja, te mogućnosti za realizaciju konkretnog projekta. To podrazumijeva također analizu društveno-ekonomskih pokazatelja i prioriteta okruženja i finansijera ili donatora. Cilj je da se identificiraju i usaglase osnovni

<sup>2</sup> Programiranje u projektnoj terminologiji je različito od programiranja u oblasti računarskih nauka.

ciljevi i prioriteti tržišta za razvojnu saradnju i time stvori relevantan i izvodiv programski okvir u kojem se projekti mogu identificirati i pripremiti. U svakom od ovih prioriteta treba formulisati strategiju koja će uvažiti iskustva iz prethodnih projekata.

U fazi **identifikacije** treba prepoznati projektne ideje i druge razvojne akcije koje treba provesti da bi se moglo nastaviti istraživanje. To podrazumijeva konsultacije s potencijalnim korisnicima svake od akcija, analizu problema s kojima se susreću, i prepoznati opcije koje bi mogle pomoći projektu. Nakon toga je moguće donijeti odluku o relevantnosti tih projektних ideja (za korisnike projekta i programski okvir) i, također, koja od ideja treba u narednoj fazi (formulacija) biti detaljnije razrađena.

Tokom faze **formulacije** razrađuje se projektna ideja i pretvara u operativni plan projekta. U detaljnoj specifikaciji ideje treba da su uključeni korisnici, sve zainteresirane strane i drugi partneri u projektu, nakon čega treba procijeniti izvodivost (ima li šansi za uspjeh) i održivost (može li projekt dugoročno opstati i donijeti dugoročne prihode korisnicima projekta). Te procjene služe kao osnovica za donošenje odluke o nastavku rada na formuliranju finansijske konstrukcije projekta.

Tokom faze osmišljavanja **finansijske konstrukcije**, projektni prijedlog je predmet analize od strane finansijske agencije koja treba da donese odluku o tome da li će projekt biti finansiran ili ne. U tom postupku se finansijska agencija dogovara s partnerima o detaljima aranžmana i uslovima pod kojima će finansiranje biti izvedeno.

Finansijeri softverskih projekata obično su krajnji korisnici ili investitori koji će ostvarivati dobit na osnovu korištenja softverskog proizvoda od strane drugih korisnika. Klijenti će se odlučiti za ulaganje u softverski proizvod ukoliko će uvođenje softverskog proizvoda u poslovanje optimizirati njihovu dobit i učiniti ih konkurentnim u sektoru.

Sve češće softverski proizvodi definiraju plan isplativosti i monetizaciju na osnovu kvalitetnoga personaliziranog marketinškog plana.

Tokom faze **implementacije** dolazi do realizacije projekta. Ovdje se provodi postupak licitacije (tender) ili odabir najboljeg ponuđača i sklapaju ugovori o tehničkoj pomoći projektu, angažira se radna snaga, materijalni resursi i usluge. Tokom realizacije projekta potrebno je pratiti radove i provjeravati da li će projekt zaista postići zacrtane rokove i ciljeve. Proces praćenja i provjere radi voditelj projekta u saradnji sa korisnicima i svim zainteresiranim stranama, kao i

sa partnerima ukoliko ima više partnera uključeno u realizaciju projekta. U slučaju potrebe projekt se može adaptirati i modificirati uključujući i modifikaciju ciljeva postavljenih u fazi formulacije.

U praćenje softverskih projekata osim voditelja projekta obavezno su uključeni članovi tima zaduženi za testiranje inkremenata softverskog proizvoda i osiguranje kvaliteta. Njihovi izvještaji sadrže važne informacije za voditelja projekta u procesu informiranja viših instanci i u donošenju važnih odluka za budućnost projekta.

U fazi **evaluacije** finansijeri (klijenti, finansijska agencija ili partnerska zemlja) ocjenjuju projekt sa stanovišta izvršenja i postignutih rezultata i ishoda, te identificiraju stečena iskustva upotrebljiva u drugim projektima. Osim završne evaluacije, rade se i evaluacije u toku realizacije projekta sa ciljem da se uočena iskustva primijene u preostalom vijeku projekta.

U proces donošenja odluka često je korisno, a ponekad i neophodno uključiti neke ili sve zainteresirane strane kao i partnerske institucije u projektu. Iskustvo pokazuje da odluke u vezi projekta donesene s nedovoljno konsultacija s korisnicima i partnerima projekta i bez nužnih informacija nisu rezultirale dobrim ishodima. Cilj projektnog ciklusa je da se osigura da sve zainteresirane strane i partneri projekta učestvuju u donošenju odluka i da su odluke temeljene na dovoljnoj količini informacija i ispravnim informacijama.

Podjela projektnog ciklusa u šest faza pruža minimalnu osnovu za uspješnu pripremu prijedloga, izvođenje i evaluaciju projekta. Priprema projekta odvija se u društvenom i političkom kontekstu, u kojem su očekivanja povećana, zahtjevi su često konfliktni a aspiracije moraju biti usklađene. U fazi identifikacije projektna ideja može biti sistematički izgrađena znatno prije procesa pripreme. U fazi formulacije će projektna ideja biti detaljnije razvijena zbog čvrstog oslonca na potrebe korisnika i dovoljno je „vlasničkog“ karaktera da je glavni partneri projekta ne bi smatrali svojom.

U praksi, projektni ciklus može da se razlikuje zavisno od tipa programa na kojem se radi. Ipak je vrlo korisno prilagoditi tekuće korake u konkretnom projektu – koracima ovog projektnog ciklusa.

Popunite narednu tabelu da opišete projektni ciklus primijenjen na softverski projekt na kojim vi radite. Koliko se to razlikuje od gore opisanog generičkog ciklusa? Zašto postoje te razlike?

<b>Faze projektnog ciklusa</b> Npr. Specifikacija zahtjeva	<b>Glavne aktivnosti</b> Npr. programiranje	<b>Glavni izlazi projekta</b> Npr. mobilna aplikacija za prijavu poreza	<b>Učesnici</b> Npr. Ministarstvo finansija, IT i ekonomski fakulteti, IT firme, predstavnici poreznih obveznika

Tabela 3: Tabela projektog ciklusa softverskog proizvoda

## 5.2. Upravljanje projektnim ciklusom

Upravljanje projektnim ciklusom (engl. Project Cycle Management - PCM) etablirala je Evropska Komisija ranih 1990-tih godina radi usporedbe kvaliteta dizajna projekta i upravljanja sa ciljem da se unaprijedi efektivnost. Zaključeno je da je znatan broj projekata slabo proveden, a kao uzroci su navedeni slijedeći uzroci:

- Slabo planiranje i priprema projekata;
- Nerelevantnost projekata za njihove korisnike;
- Rizici nisu uzeti u obzir u dovoljnoj mjeri;
- Ignorirani su faktori koji utiču na dugoročnu održivost;
- Pouke iz ranijih iskustava su rijetko kada ugrađivane u praksu.

<u>Zašto PCM</u>	
<u>Iskustva:</u>	<u>PCM:</u>
• Nejasan strateški okvir	Sektorski pristup
• Ponudom vođeni projekti	Tražnjom vođena rješenja
• Nedovoljna analiza postojećeg stanja	Unaprijediti analizu
• Aktivnostima orijentirano planiranje	Cilju orijentirano planiranje
• Neprovjerljiv utjecaj projekta	Provjerljiv utjecaj
• Pritisak troškova	Naglasak na kvalitetu
• Kratkoročna vizija	Fokus na održivosti
• Neprecizni projektni dokumenti	Standardizirani formati

Slika 22: Smisao upravljanja projektnim ciklusom

**1. Upravljanje projektnim ciklusom integrira faze ciklusa projekta** pa su teme sistematski provjerene pristupom i metodologijom koji osiguravaju da ciljevi i problemi održivosti ostanu u fokusu.

„Upravljanje projektnim ciklusom obavezuje praktičare u dizajniranju projekta da se fokusiraju na realne potrebe korisnika zahtijevajući detaljnu ocjenu postojeće situacije i primjenjujući metodu logičkog okvira.... U oblikovanju projekta su od samog početka uključeni aspekti koji osiguravaju održivost. Jaka strana PCM-a je da su dokumenti projekta strukturirani saglasno standardiziranom formatu koji obuhvata sve relevantne teme, uključujući polaznu tezu na kojoj se projekt temelji. Te teme se u svakoj fazi projektnog ciklusa provjeravaju i po potrebi revidiraju i prenose u narednu fazu. Ovaj

sistem čini projektni koncept i kontekst u kojem se realizuje jasnim i vidljivim, te omogućava bolji monitoring i evaluaciju.“

#### **Principi upravljanja projektnim ciklusom**

- **Faze ciklusa projekta** - strukturirane i na informacijama zasnovano donošenje odluka
- **Klijentu orijentirano** – uključenost partnera projekta u donošenje odluka
- **Planiranje u logičkom okviru** – opsežna i konzistentna analiza
- **Održivost** – mehanizmi za neprekidan priliv koristi od projekta
- **Integralni pristup** – vertikalna integrisanost i standardizovana dokumentacija

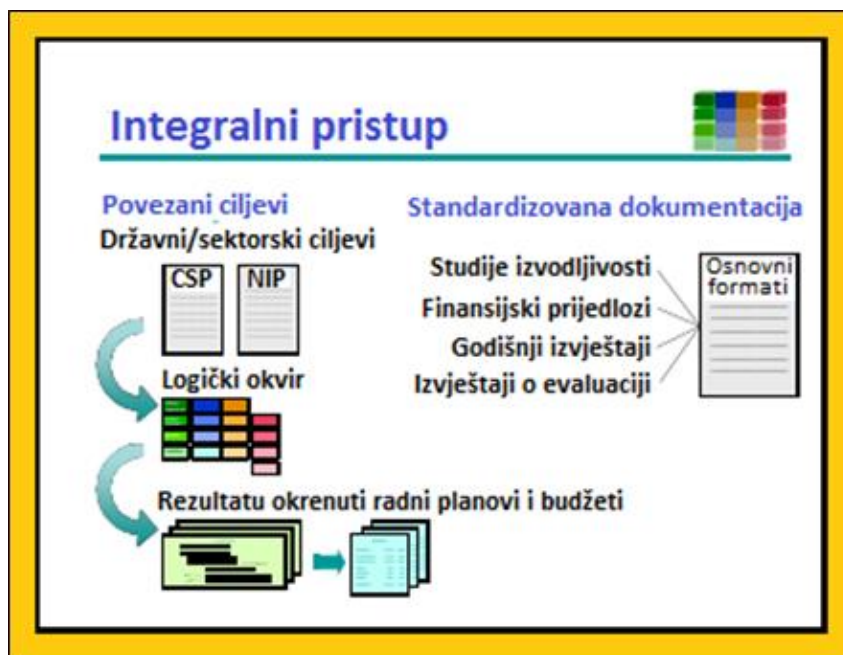
Slika 23: Principi upravljanja projektnim ciklusom

**2. Klijenti učestvuju zajedno sa budućim** korisnicima rezultata projekta u ključnim fazama ciklusa i u formulaciji ciljeva projekta na način da osiguraju održivu korist u toku upotrebnog vijeka projekta.

**3. Uključivanje aspekata održivosti u dizajn projekta** da bi se u konačnici postigla održiva korist iz projekta potrebno je i u fazi dizajna razmatrati aspekte održivosti projekta (engl. sustainability).

**4. Korištenje pristupa pomoću logičkog okvira** osigurava konzistentan analitički pristup oblikovanju projekta i upravljanju njime.

**5. Integralni pristup** povezuje ciljeve svakog projekta sa ciljevima strategije razvoja, nacionalnim, međunarodnim ili ciljeva sektora kojem je namijenjen proizvod. Integralni pristup također osigurava da radni planovi i budžeti budu pripremljeni na osnovu logičkog okvira projekta; a korištenjem osnovnog formata da se osigura konzistentan i obuhvatan tretman ključnih pitanja u životnom vijeku projekta (Slika 24).



Slika 24: Integralni pristup

**Osiguranje relevantnosti, izvodljivosti i održivosti projekta** osigurava se upravljanjem projektnim ciklusom koji spaja principe upravljanja, analitičke alate i tehnike i primjenjuje ih u strukturiranom procesu donošenja odluka kako bi bilo osigurano:

- Da su **projekti relevantni** za prihvaćenu strategiju i da odražavaju realne potrebe korisnika:
  - Projekti treba da su saglasni sa strategijom razvoja organizacije, sektora, nacionalnom strategijom te zajedničkim ciljevima partnera u međunarodnim projektima;
  - U proces planiranja, od početka, treba da su uključeni korisnici (naglašeno u SCRUM razvojnom okviru softvera);
  - Analiza problema mora biti temeljita, ali ne „uroniti“ u „paralizu u analizi“;
  - Ciljevi treba da su jasno definirani u pogledu koristi za ciljnu grupu.
- Da su **projekti izvodljivi** u smislu da ciljevi i rezultati mogu realno biti ostvareni u granicama konkretnog okruženja:
  - Ciljevi treba da su logični i mjerljivi;
  - Moraju biti uzeti u obzir rizici, pretpostavke i sposobnost izvođačkih agencija za realizaciju projekta;
  - Monitoring projekta mora biti koncentriran na relevantne ciljeve.



- **Da su projekti održivi**

- Kao dio oblikovanja projekta treba da su obuhvaćeni svi relevantni faktori koji utiču na održivost projekta;
- Rezultati evaluacije treba da se koriste za prenošenje iskustva na oblikovanje budućih projekata.

### 5.3 Planiranje upravljanja projektnim ciklusom i upravljački alati

Planiranje projekta i upravljački alati su praktični mehanizmi za postizanje relevantnosti, izvodljivosti i održivosti projekta. Glavni alat za upravljanje projektnim ciklusom je pristup pomoću logičkog okvira (engl. **Logical Framework Approach, LFA**).

Razmatranje u logičkom okviru predstavlja efikasnu tehniku koja omogućava svim zainteresiranim stranama projekta da identificiraju i analiziraju problem te odrede ciljeve i aktivnosti koje treba poduzeti za rješavanje problema. Koristeći strukturu logičkog okvira planeri provjeravaju koncept predloženoga projekta s aspekta relevantnosti, izvodljivosti i održivosti. Osim uloge koju ima u pripremama programa i projekata, on predstavlja i ključni alat u implementaciji i evaluaciji, je služi kao osnova za pripremu akcionih planova i osmišljavanje sistema monitoringa te kao okvir za evaluaciju.

Partneri projekta treba da su – što je više moguće – uključeni u timski rad jer, pored ostalog, snažno doprinose uspješnosti planiranja. Za uspješno korištenje logičkog okvira se, kao dodatna analitička sredstva i alati - koriste sredstva tehničke, ekonomske, društvene i okolinske analize. Alate koji koristi Evropska komisija su: Ocjena uticaja na okoliš (Environmental Impact Assessment), Analiza uticaja na polove (Gender Impact Analysis) i Finansijska i ekonomska analiza (Financial and Economic Analysis).

Sažetak ključnih pojmova poglavlja:

- ✓ Projekti se planiraju i izvode u sekvencama poznatim pod nazivom projektni ciklus. On predstavlja strukturu koja osigurava uključenost projektnih partnera, raspoloživost relevantnih informacija tako da se mogu donositi informatički temeljene odluke u ključnim fazama projekta.
- ✓ Šest faza projektnog ciklusa su progresivne. Svaka od njih vodi slijedećoj. U svakoj od njih potrebno je raspolagati informacijama koje omogućavaju korektno odlučivanje prije nego se pređe u narednu. Ciklus podrazumijeva da se iskustva iz završenih projekata mogu koristiti za poboljšanje budućih projekata.
- ✓ Razdvajanje faza identifikacije i formulacije projekta je posebno važno. U fazi identifikacije se analizira relevantnost projektne ideje i blagovremeno učvršćuje stav o njoj, dok u fazi formulacije projektna ideja može biti napuštena.
- ✓ Ciklus upravljanja projektom uvela je Evropska Komisija ranih 1990-tih godina radi unapređenja kvaliteta osmišljavanja projekata i upravljanja njima, te time poboljšala efektivnost projekata. PCM integrira faze ciklusa na način da se ključna pitanja sistematski provjeravaju.
- ✓ PCM povezuje principe upravljanja, analitičke alate i tehnike i ugrađuje ih u sistem odlučivanja kako bi bilo osigurano da projekt ostane relevantan za stvarne potrebe korisnika, da je izvodljiv i održiv.
- ✓ Glavni alat koji se koristi u oblikovanju i upravljanju projektnim ciklusom je logički okvir. Da bi bio uspješno korišten, logički okvir mora biti podržan drugim alatima tehničke, ekonomske, društvene i okolinske analize.

## 6. Pristup pomoću logičkog okvira – alata za dizajn i analizu projekta

U ovom poglavlju će se nešto detaljnije predstaviti pristup pomoću logičkog okvira i njegova uloga u oblikovanju projekta. To će biti učinjeno na jednom jednostavnom primjeru projekta.

### 6.1. Uvod

Pristup pomoću logičkog okvira je glavni alat koji se koristi za oblikovanje projekta u fazi njegove identifikacije i razrade. U fazi identifikacije on osigurava da projekta ideja postane relevantna, a u fazi razrade – da osigura izvodljivost i održivost. Pristup je podijeljen u dvije faze:

1. **Faza analize** – tokom koje se analizira postojeća situacija sa ciljem da se razvije vizija „**poželjne buduće situacije**“ i da se odaberu strategije koje će viziju učiniti ostvarljivom.
2. **Faza planiranja** – tokom koje se projektna ideja razrađuje do operativnih detalja.



Slika 25: Pristup pomoću logičkog okvira

Zbog značaja aktivnosti i dodjele resursa za integralni pristup, ovi alati su opisani u posebnoj poglavlju. Bez obzira na to oni predstavljaju integralni dio pristupa pomoću logičkog okvira.

## 6.2. Faza analize

Projekti se rade tako da adresiraju problem kojeg imaju korisnici. Korektno planirani projekti kojima su prepoznaju stvarne potrebe korisnika ne mogu se ostvariti ako nije provedena dobra analiza problema. Međutim, aktuelnu situaciju različito shvataju različite grupe partnera projekta. Zbog toga je – u fazi analize – veoma važno okupiti predstavnike svih zainteresiranih strana. Obično se to radi kroz radionice na kojima se problem i teme otvoreno diskutuju. Faza analize se izvodi kroz tri aktivnosti: analiza problema (1), analiza ciljeva (2) i analiza strategije (3).

### 6.2.1 Analiza problema

U fazi analize problema razmatraju se negativne strane postojeće situacije i uspostavlja uzročno-posljedična relacija („Cause and Effect“ relationship) između postojećih problema, tj.:

1. Identifikacija svih zainteresiranih strana i projektnih partnera na koje se odnosi predloženi projekt;
2. Identifikacija glavnih problema s kojima se susreću korisnici;
3. Razvoj stabla problema da se uspostavi uzročno-posljedični odnos.

**Analiza zainteresiranih strana i partnera projekta.** Ovom, polaznom, tačkom u analizi problema treba sve potencijalne neposredne i posredne korisnike projekta grupisati prema srodnosti interesa, bilo da se radi o pozitivnom ili negativnom uticaju projekta na njih. Koristeći se metodama posmatranja, proučavanja dokumentacije, ankete i intervjua i tehnikama diskusije omogućava se dokumentacija o radu partnera projekta i zaključcima. Ova dokumentacija omogućava projektnim planerima bolji uvid u pripremni proces, posebno u planiranju neophodnih istraživanja potrebnih prije organizacije namjenske radionice.<sup>3</sup>

**Spolne konsideracije ili uključivanje stvarnih partnera u projekt.** Veoma je važno da se, prije održavanja radionice, pribavi dovoljno informacija i izvrši njihova analiza. Te informacije dolaze iz različitih izvora kao što su intervjui, promatranje, izvještaji i

---

<sup>3</sup> Tehnika rada u radnim grupama formiranim prema zainteresiranosti učesnika u radionici, i rezime u plenarnoj sesiji pokazali su se kao efikasna metoda prikupljanja i dokumentiranja rezultata rada u radionici.

statistika. Očekivana relevantnost, izvodljivost i održivost neke intervencije postaju mnogo očitiji ako su svi učesnici u projektu blagovremeno uključeni u analizu situacije i pozvani da učestvuju u planiranoj radionici. Pri tome su neki projektni ciljevi neostvarivi npr., ako oba pola nisu konsultirana ili nisu diskutirane njihove respektivne uloge u projektnim aktivnostima.

U skoro svim društvima se muškarci i žene razlikuju u vezi s njihovim dnevnim zadacima, pristupu resursima i njihovoj kontroli i učešću u donošenju odluka. Ustvari, polna diskriminacija uvijek umanjuje efektivnost i uticaj projekta. Zbog toga je nužno da se u pripremi projekta, analizira mogući uticaj projekta na muškarce, žene i druge ciljane grupe (djecu, etničke manjine, socijalne grupe) **prije** donošenja važnih odluka o obliku intervencije, njenim ciljevima, strategijama i alokaciji resursa.

**Planiranje radionice.** Nakon što su relevantne informacije prikupljene i izvedena analiza, moguće je izvršiti pripreme za radionicu. Na osnovu raspoloživih informacija partneri projekta će identificirati „brain storming-om ključne probleme u datoj situaciji. Glavna tehnika koja se koristi u ovoj fazi je prikazivanje (crtanje) drveta problema. Taj problem se sređuje hijerarhijski. Svaki problem se najprije sažme a zatim utvrđuje njihova hijerarhija na slijedeći način:

- ✓ Ako problem predstavlja **uzrok** – spušta se na niži nivo
- ✓ Ako je neki problem **posljedica** drugog problema – pomjera se naviše
- ✓ Ako nije ni uzrok ni posljedica ostavlja se na nepromijenjenom nivou.

Naprimjer, ako je centralni problem „*neprepoznatljivost određene marke (engl. brand)*“ uzrok može biti u „*neprisutnosti na Internetu i nedostatak e-poslovanja*“ a posljedica može biti „*slaba prodaja i ograničeno plasiranje informacija o brandu*“.

U postupku formiranja stabla, na njega se kače preostali problemi dok se ne dobije kompletno stablo. Kad se to stablo dovrši definiran je i centralni problem. O tome da li je – tako identificirani– centralni problem zaista centralni treba postići saglasnost s interesnim grupama uključenim u analizu. Revizija analize problema može pokazati da je neki drugi problem važniji, ali to ne utiče na validnost analize. Nakon kompletiranja, stablo problema predstavlja obuhvatnu sliku **postojeće negativne situacije** (Slika 26).



Slika 26: Stablo ili piramida problema

Postoje dvije zajedničke teškoće – proistekle iz iskustva – tokom identifikacije i analize problema: neadekvatna specifikacija problema (1) i „odsustvo rješenja“ (2).

- Neadekvatna specifikacija problema nastaje kad problem nije predstavljen sa dovoljno detalja zbog kojeg nije identifikovana stvarna priroda problema. Tvrdnja kao što je „Slabo upravljanje“ mora biti izbačena jer treba preciznije otkriti njegov sadržaj, recimo „slaba finansijska kontrola“ ili „kašnjenje u isporuci ključnih usluga“, itd. Pri tome treba imati na umu da je postavljanje detalja stvar procjene od strane moderatora radionice i učesnika. To, dakle, zavisi od obuhvata i prirode projekta.
- Odsustvo rješenja je tvrdnja koja ne opisuje tekuće negativnosti stvarnog stanja nego opisuje odsustvo željene situacije. Naprimjer, „Odsustvo obučenog osoblja“ ne opisuje specifičan problem (osoblje ne raspolaže dovoljnim ili odgovarajućim znanjima) nego nosi rizik iskrivljavanja intervencije prema odsutnom rješenju (odsutno rješenje je „obuka“) i greške u odluci (mogućeg „angažovanja odgovarajućeg upravljačkog osoblja“). Zbog toga, uvijek treba biti pažljiv s tvrdnjom kao što je „odsustvo ...“.

## 6.2.2 Analiza ciljeva

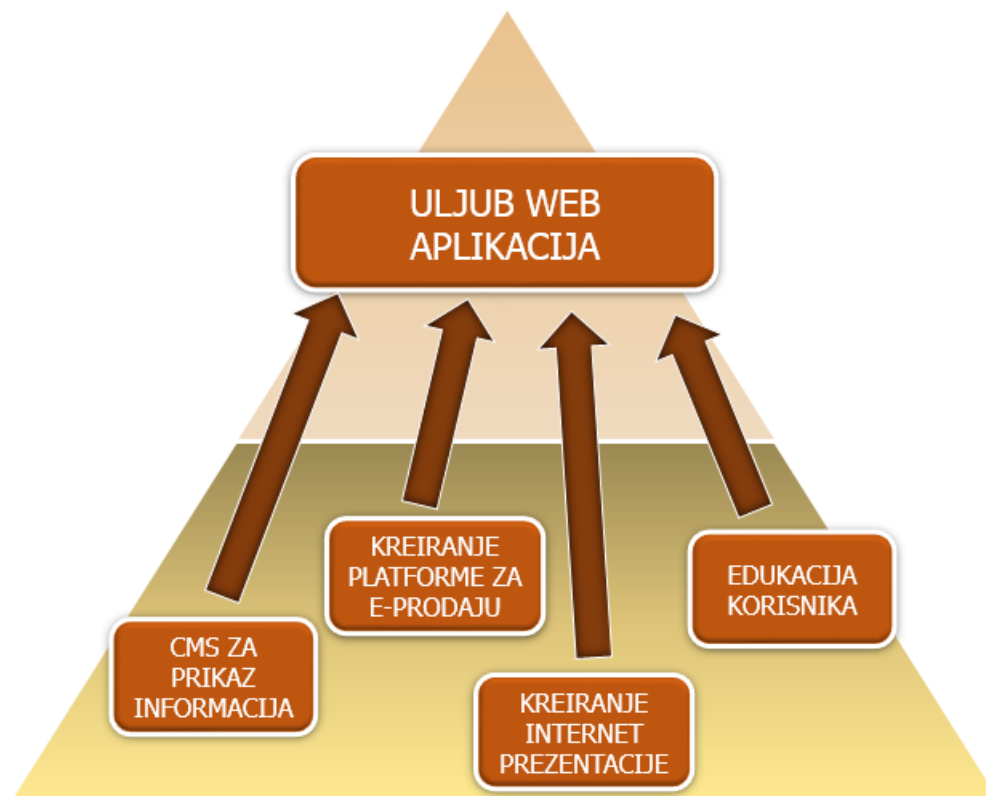
Dok analiza problema predstavlja negativne strane postojeće situacije, analiza ciljeva predstavlja pozitivnu stranu buduće željene situacije. To podrazumijeva prevođenje problema u ciljeve.

PROBLEM	CILJ
1. NEPREPOZNATLJIVOST ULJUB-A	1. UVOĐENJE ULJUB WEB APLIKACIJE
1.1 OGRANIČENO PLASIRANJE INFORMACIJA	1.1 UVOĐENJE CMS <sup>4</sup> ZA DIJELJENJE INFORMACIJA
1.2 NEZASTUPLJENOST NA INTERNETU	1.2 KREIRANJE INTERNET PREZENTACIJE
1.3 NEMOGUĆNOST PRODAJE RADOVA	1.3 KREIRANJE PLATFORME ZA E-PRODAJU
1.4 NEPOZNAVANJE TRENDOVA POSLOVANJA	1.4 EDUKACIJA KORISNIKA

Tabela 4: Prevođenje problema u ciljeve projekta

Stablo ili piramida ciljeva može se shvatiti kao slika stabla problema u ogledalu, a odnos „uzrok-posljedica“ pretvoriti u relaciju „sredstvo-učinak“. U razmatranju stabla ciljeva može se dogoditi da postoji logički jaz u početnom stablu ciljeva koji nije bio uočljiv na stablu problema, zbog toga je potrebno relacije „sredstvo-učinak“ preispitati i reorganizovati. Naime, ciljevi koji se odnose na sličnu oblast mogu se regrupirati u mrežu koja će stvoriti osnovu za analizu strategije. Kad se to presloži – stablo ciljeva treba da pruži sveobuhvatnu sliku **željene buduće situacije** (Slika 27).

<sup>4</sup> Eng. Content Management System – sistem za (dinamičko) upravljanje sadržajima; web aplikacija uz pomoć koje se može upravljati sadržajem bez poznavanja programiranja.



Slika 27: Stablo ili piramida ciljeva

### 6.2.3 Analiza strategije (razmatranje alternativa)

Posljednja aktivnost u fazi analize je izbor strategije koja će se koristiti za postizanje željenih ciljeva. Strateška analiza ima za cilj da se donese odluka o tome koji ciljevi (nižeg nivoa) će **biti uključeni u projekt** a koji će **ostati izvan njega**, te šta će se, nakon svih rektifikacija, definisati kao najviši cilj. Osim što se izvodi logičko razmatranje, strateška analiza treba da odgovori na pitanja izvodljivosti različitih (alternativnih) intervencija.

Zavisno od obuhvata projekta i količine rada koja to podrazumijeva, neki dijelovi strukture mogu imati sadržaj koji je moguće tretirati kao poseban projekt, ili sve skupa može predstavljati program kao skup projekata.

### 6.3 Faza planiranja

Glavni proizvod pristupa pomoću logičkog okvira je Matrica logičkog okvira. Logički okvir uspostavlja logiku intervencije projekta (aktivnosti koje se poduzimaju, rezultati koji iz njih proističu, projektni cilj i td.) i opisuje osnovne pretpostavke i rizike koji se nalaze u pozadini logike. Time se stvara osnova za provjeru izvodljivosti projekta. Za upravljanje i nadzor projekta, u logičkom okviru definišu se zadaci koje treba uraditi, resursi koje je potrebno



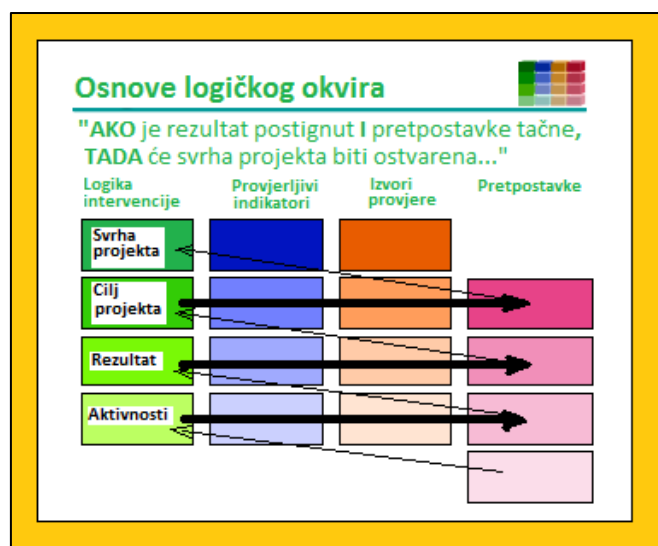
pribaviti i odgovornost upravljača projektom. U drugom i trećem stupcu (objektivno provjerljivi indikatori i izvori verifikacije) logički okvir daje okvir prema kojem će se moći provjeriti i evaluirati napredovanje projekta.

### 6.3.1. Matrica logičkog okvira

Prije detaljnog opisa logičkog okvira, treba imati na umu da logički okvir ne predstavlja čarobno rješenje za identifikaciju i razradu dobrih projekata. Izbacivanje ili unošenje aktivnosti u logički okvir („garbage in, garbage out“) ne smije biti rađeno mehanički. Kad se koristi ispravno – logički okvir veoma pomaže u uspostavljanju logičkih odnosa među aktivnostima, rezultatima, ciljevima i svrsi, u najmanju ruku za dobro obaviještene korisnike. Prema tome, logički okvir ne smije biti shvaćen kao običan skup mehaničkih procedura, nego kao pomoć u promišljanju. Treba ga, dakle, shvatiti kao dinamički alat koji je moguće dotjerivati tempom koji odgovara dinamici izvođenja projekta. On treba da se koristi radi vidljivosti projektne strukture i radi daljeg planiranja i upravljanja budžetom.

U fazi analize, projektni partneri mogu diskutovati probleme, ciljeve i strategije, a logički okvir pruža polazište za to. Jasnim definisanjem glavnog i parcijalnih ciljeva u hijerarhijskom poretku logički okvir postaje sredstvo provjere unutarnje logike projektnog plana, osiguravajući da aktivnosti vode ka rezultatima, ovi ka ciljevima itd. Planeri, u tome, moraju identifikovati kritične pretpostavke i rizike koji mogu ugroziti izvodljivost projekta, precizirati indikatore i izvore informacija koje treba koristiti u praćenju projekta. Sve te informacije nalaze se u jednom jedinom dokumentu kao sažetku projekta.

Mada koristan, što je dokazano u praksi, logički okvir ne predstavlja ni sveobuhvatan alat niti garantuje uspjeh projekta. Ovladavanje tim procesom zahtijeva vrijeme i obuku u vezi s konceptom i logikom ovog pristupa. Od planera se očekuje da – u vrlo sažetoj formi – sažmu složene ideje i relacije u jednostavne fraze koje mogu izgledati nejasne ili besmislene. To se prečesto karikira u „popunite obrasce“, ali takvim pristupom doći će se do nejasno definiranih ciljeva pa korisnici i partneri u projektu neće doživjeti projekt kao njihov projekt.



Slika 28: Matrica logičkog okvira

Logički okvir je – po formi – tabela ili matrica sa četiti stupca i (u najkraćoj formi) četiri reda. Njegova vertikalna logika otkriva šta projekt treba da napravi, razjašnjava uzročno-posljedične odnose i definiše pretpostavke i rizike koji se nalaze izvan kontrole projektnog tima koji ga realizuje. Horizontalna logika otkriva način na koji je moguće mjeriti učinke i resurse – specificira indikatore i izvore informacija u kojima je moguće izvršiti provjeru.

U tabeli 5 predstavljen je primjer logičkog okvira projekta.

<b>Svrha projekta:</b>	Unapređivanje organizacionih, funkcionalnih i finansijskih aspekata poslovanja ULJUB-a.
<b>Problem:</b>	Neprepoznatljivost ULJUB-a
<b>Cilj projekta:</b>	Implementirati aplikaciju za podršku rada ULJUB-a radi kreiranja web prepoznatljivosti, uvođenja elemenata e-poslovanja i pojačanja promocije Udruženja.
<b>Ulazi projekta:</b>	Samo novčana sredstva
<b>Izlazi projekta:</b>	Internet aplikacija sa elementima CMS-a i e-poslovanja
<b>Korisnici projekta:</b>	Svi korisnici ULJUB-a, posebno aktivni (prodavci)

Tabela 5: Primjer logičkog okvira projekta

### 6.3.2. Nivoi ciljeva

Ciljevi, odabrani za uključivanje u projekt, treba da se razlože prema logici intervencije u projektu – u prvom stupcu. Ovdje je bitno da su svi nivoi ciljeva korektno uneseni.

1. **Opšti cilj ili svrha** (engl. Overall Objectives, General Objectives) programa treba da objasni zašto je program bitan za društvo, **po dugoročnom učinku na korisnike i široj**

**koristi za druge grupe.** On treba da pokaže kako se program uklapa u strategiju razvoja, regionalne ili sektorske politike EU i vlada odnosno organizacija zemlje na koju se on odnosi. Opšti cilj neće biti postignut isključivo razrađenim programom nego na njega imaju utjecaja i drugi programi i projekti.

2. **Cilj projekta** (engl. Project Purpose) treba da prostekne iz centralnog problema u obliku **koristi koje će imati korisnici projekta ili ciljne grupe**, kao rezultat korištenja usluga iz projekta u periodu upotrebe.
3. **Rezultat** (engl. Result) opisuje **uslugu koja će biti pružena očekivanim korisnicima ili ciljnoj grupi** i on je neposredan proizvod projekta koji projektni tim treba da osigura po završetku istog. Rezultati treba da budu usmjereni prema glavnim uzrocima problema s kojim se susreće ciljna grupa. Da bi bila osigurana relevantnost rezultata za problem koji se tiče korisnika (ciljne grupe) treba da bude identifikovana tražnja korisnika za tom konkretnom vrstom usluge.
4. **Aktivnosti** (Activities) – treba unijeti dobra ili usluge koje će svaka od aktivnosti proizvesti, po njenom završetku.

Ključno za uspješno korištenje logičkog okvira je razumijevanje značenja definicija u njihovom izvedbenom obliku, posebno u shvatanju pojmova Rezultat i Projektni cilj.

Mada su menadžeri opsjednuti isporukom rezultata, oni ne mogu **kontrolirati** ponašanje ciljne grupe. Ostvarenje opšteg cilja projekta zahtijeva reakciju korisnika (engl. Beneficiary response) u smislu korištenja usluga projekta i iz toga izvlače koristi za sebe. Međutim, to ne znači da projektni tim ne snosi **odgovornost** za ostvarenje opšteg cilja ili svrhe projekta. U suštini, oni imaju jasnu odgovornost da zadovolje određenu potrebu i uvažavaju preferencije korisnika.



Slika 29: Odnos između rezultata i cilja projekta

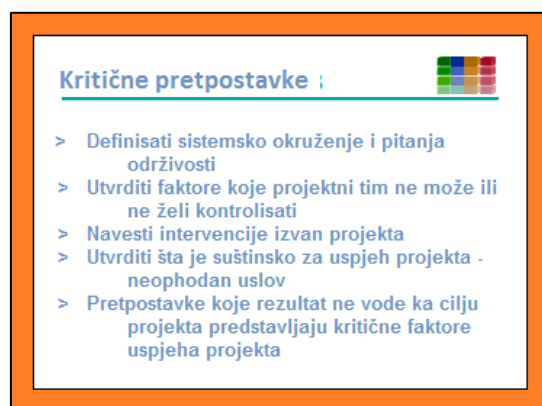
Jedna od konvencija u Upravljanju projektom ciklusom koja često uzrokuje nesporazum je da postoji samo jedan cilj projekta. Razlog za takvu definiciju je činjenica da ako se definiše više ciljeva to bi podrazumijevalo složeniji projekt i uzrokovalo nove probleme. Uz to, višestruki cilj bi mogao proizvesti međusobne konflikte ciljeva. Zbog toga je razjašnjenje pitanja šta projekt treba da proizvede ustvari kritičan korak u oblikovanju projekta.

Nakon što se partneri projekta usaglase šta treba da bude Cilj projekta, tada je moguće ciljeve nižeg ranga preslikati iz stabla ciljeva u matricu. Nakon toga je potrebno ponovo analizirati odnos između sredstava i učinaka da bi se – po potrebi – ugradile dodatne aktivnosti i rezultati.

### 6.3.3. Pretpostavke

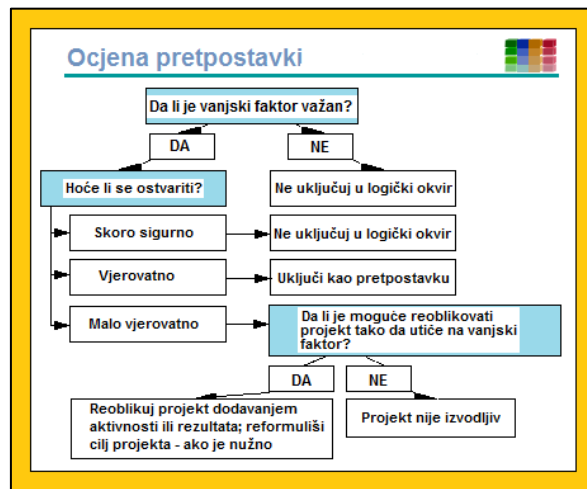
U fazi analize projekta može se zaključiti da projekt, u cjelini, neće ostvariti sve ciljeve prikazane u stablu ciljeva. Nakon što je strategija odabrana ciljevi koji nisu obuhvaćeni logikom intervencije, kao ni ostali vanjski faktori, ostaće izvan projekta. Oni će uticati na implementaciju projekta i njegovu dugoročnu održivost, ali će ostati izvan kontrole. Ti uslovi moraju biti ispunjeni ako se želi postići zacrtani opšti cilj, a treba ih uključiti kao pretpostavke u četvrtu kolonu logičkog okvira.

Preduslovi se razlikuju od pretpostavki utoliko što moraju biti ispunjeni prije početka projekta. Vjerovatnoća i značaj tih uslova koji treba da budu ispunjeni treba da se procjeni posebno, u ocjeni rizika projekta. Neki od njih mogu biti kritični, a drugi marginalnog karaktera.



Slika 30: Uloga pretpostavki

Koristan način ocjene značaja pretpostavki je pomoću algoritma datog u slici 31. Nakon što su one prepoznate treba ih iskazati u terminima željene situacije. Na taj način one mogu biti provjerene i ocijenjene.



Slika 31: Algoritam ocjene pretpostavki

#### 6.3.4. Faktori koji osiguravaju održivost

Za projekt se kaže da je održiv kad neprekidno – uz dužem periodu nakon što je prestala donatorska pomoć – donosi ciljnoj grupi koristi. U prošlosti je vrlo često utvrđeno da su projekti bili od slabe koristi ciljnoj grupi, jer nije uzet u obzir određeni broj faktora kritičnih za uspjeh. Iskustvo je pokazalo da dugoročna održivost projekata zavisi od sljedećih faktora:

- **Političke podrške** – obim do kojeg partnerska vlada pruža direktnu podršku nastavku projekta nakon perioda podrške donatora. Npr. u projekt implementacije univerzitetskog informacionog sistema potrebno je uključiti i nadležno ministarstvo obrazovanja.
- **Adekvatnosti tehnologije** – da li primijenjena tehnologija ima dugoročniji vijek upotrebe (mogućnost održavanja softverskog proizvoda, potpunost sigurnosnih propisa; osposobljenost lokalnog osoblja za korištenje i održavanje itd).
- **Institucionalnog i upravljačkog kapaciteta** – sklonosti i spremnosti korisničkih agencija da nastave pružati projektne usluge nakon isteka donatorske pomoći (ICT odjel će održavati proizvod i servisirati potrebe korisnika).
- **Ekonomskoj i finansijskoj vitalnosti** – da li periodične koristi projekta nadmašuju njegove troškove i da li predstavlja dugoročno održivu investiciju.

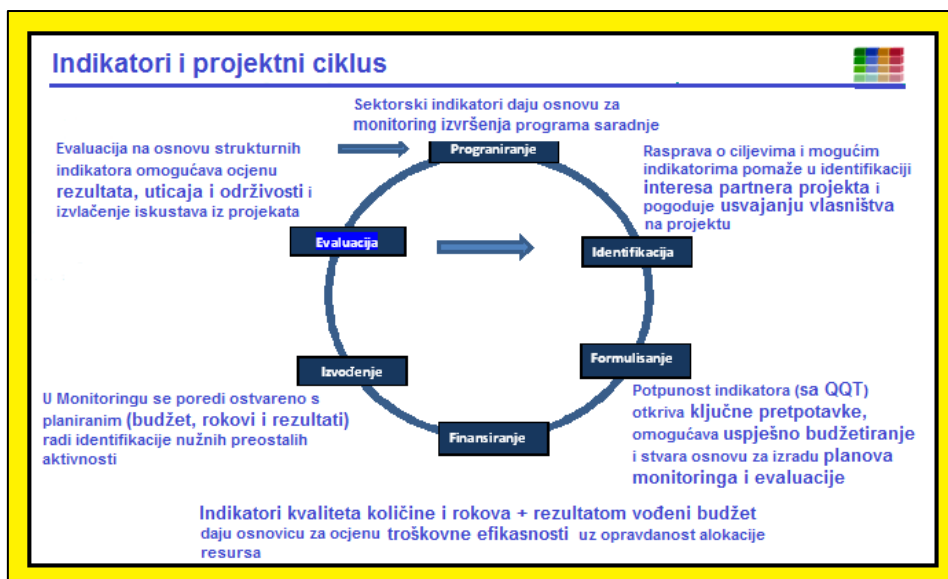
- **Socio-kulturna i spolna pitanja** koja utiču na motivaciju i učešće – obim do kojeg su identificirane potrebe svih korisničkih grupa naznačene projektom i utjecaj koji to ima na raspodjelu koristi u dužem roku.
- **Zaštite okoliša** – obima do kojeg projekt nastoji očuvati ili oštetiti ekološko okruženje i time podržati ili zanemariti ostvarenje dugoročnih koristi.

Navedeni faktori se ocjenjuju prema vjerovatnoći i značaju na isti način kao i vanjski faktori (korištenjem algoritma), bilo da su odbačeni kao nevažni, uključeni kao pretpostavke u logički okvir ili usmjerene na reoblikovanje projekta. Faktori su važni zbog toga što njihovo potezivanje može reducirati izvodljivost i održivost projekta.

#### 6.3.5. Objektivno provjerljivi indikatori (Objectively Verifiable Indicators,, OVIs)

Objektivno provjerljivi indikatori opisuju ciljeve projekta u mjerljivim pokazateljima i čine osnovicu za **mjerenje učinka (engl. performance measurement)**. Specifikacija ovih indikatora služi za provjeru održivosti ciljeva i čini osnovicu za sistem monitoringa projekta. Kad su identificirani, pokazatelji treba da budu osmišljeni tako da pokazuju količinu, kvalitet i vrijeme (Quantity, Quality, Time, QQT).

Cilj utvrđivanja sadržaja pokazatelja je da bilo koje lice – koje želi da utvrdi učinak projekta – dobije isti odgovor. To je mnogo lakše učiniti ako su pokazatelji zaista mjerljivi, teže je kad su u pitanju kvalitativne promjene. Tada je korisno odabrati više pokazatelja jer samo jedan rijetko kada može dati potpun odgovor. S druge strane, previše pokazatelja može izazvati zbrku i mnogo rada na njihovom pronalaženju. Pokazatelji su često potrebni tokom implementacije, ali kad to nije moguće treba računati s efektivnijim monitoringom. Uloga pokazatelja nije ograničena samo na monitoring i evaluaciju, kao što je vidljivo na slici 32. Oni igraju značajnu ulogu u svim fazama projektnog ciklusa.



Slika 32: Indikatori i projektni ciklus

### 6.3.6. Izvori provjere podataka (Source of Verification, SOVs)

Nakon definiranja indikatora treba utvrditi izvore informacija i način njihovog prikupljanja. To će pomoći da testiramo njihovu dostupnost i realističnost u pogledu utrošenog vremena, novca i ličnih napora. U provjeri je potrebno specificirati:

1. Format u kojem je informacija raspoloživa (npr., Izvještaj o napredovanju rada, računovodstvo projekta, snimci projekta, službena statistika itd.)
2. Ko treba da pribavi informacije
3. Kako često se informacije mogu dobiti (npr., mjesečno, kvartalno, godišnje itd.)

Izvore izvan projekta treba provjeriti sa stanovišta dostupnosti, raspoloživosti i relevantnosti. Također je potrebno predvidjeti rad i troškove prikupljanja tih informacija. Često postoji direktna veza između složenosti projekta i kompleksnosti prikupljanja podataka i to stvara troškove, kao što je pokazano slikom 33. Ako zahtijevani podaci nisu dostupni, biće potrebno potražiti nove izvore informacija. Isto vrijedi kad se radi o troškovima njihovog pribavljanja.



Slika 33: Međuzavisnost troškova i složenosti njihovog prikupljanja

### 6.3.7. Sredstva i troškovi

Sredstva su ljudski, materijalni i finansijski resursi potrebni za poduzimanje planiranih aktivnosti i upravljanje projektom. Sa ciljem da se pribave pouzdane procjene sredstava i njihovih troškova potrebno je da se planirane aktivnosti projekta – uključujući aktivnosti upravljanja – procijene sa što moguće više detalja. Posebnu pažnju treba posvetiti prikupljanju podataka o objektivno provjerljivim informacijama (OVIs).

Popuniti tabelu 6 za softverski projekt koji trenutno realizirate.

	Logika intervencije	Objektivno provjerljivi indikatori	Izvori verifikacije	Pretpostavke
<b>Opšti cilj</b>		•		
<b>Ciljevi projekta</b>		•		
<b>Rezultati</b>		•		
		•		
<b>Aktivnosti</b>				

**Preduslov:**

Tabela 6: Logički okvir projekta



Sažetak ključnih pojmova poglavlja:

- ✓ Za ispravan pristup stvarnim potrebama korisnika i uzimanje u obzir različitih pogleda raznih grupa partnera u projektu neophodno je da se u fazi analiziranja projekta okupe sve te grupe. Također je, za konzistenciju projekta, nužno uvažiti različitost uloga muškaraca i žena u projektu.
  
- ✓ Matrica logičkog okvira je glavni proizvod pristupa pomoću logičkog okvira. U njemu se definira logika intervencije projekta (ako se aktivnosti izvrše, rezultat će biti ostvaren, a za njim projektni cilj i td.) i opisuje važne pretpostavke i rizike na kojima počiva logika. Uz objektivno provjerljive indikatore i poznate izvore informacija logički okvir pruža i okvir u kojem je moguće nadzirati i evaluirati projekt.
  
- ✓ Pristup pomoću logičkog okvira nije sveobuhvatan alat i ne garantira projektu uspjeh po sebi. Prečesto primijenjeni pristup na principu „popuni kućice“ dovodi do slabo razrađenog projekta s nejasnim ciljevima i izostanak „vlasništva“ na projektu od strane partnera projekta.
  
- ✓ Posebno je važno osigurati da se svi nivoi ciljeva korektno navedeni:
  - **Opšti ciljevi ili svrha (engl. Overall Objectives, General Objectives)** – širi sektorski ili nacionalni programski ciljevi kojima projekt treba da doprinese
  - **Cilj projekta (engl. Project Purpose)** – održive koristi koje će projekt pružati korisnicima, institucijama ili sistemu.
  - **Rezultati (engl. Results)** – usluge koje projekt treba da pruža.
  - **Aktivnosti (engl. Activities)** – način na koji će dobra i usluge biti isporučeni.

## 7. Korištenje logičkog okvira za formulisanje aktivnosti i terminiranje resursa

U ovom poglavlju se objašnjava način na koji se koristi logički okvir za rezultatom vođene radne planove i budžete i predstavlja pristup korak po korak u osmišljavanju aktivnosti i terminiranju resursa.

### 7.1. Raspored aktivnosti i resursa

Nakon što je matrica logičkog okvira dovršena nastavlja se razrada operativnih detalja. Raspored aktivnosti je metoda prikazivanja aktivnosti projekta kojom se određuje njihov raspored i zavisnosti između njih te služi kao osnovica za utvrđivanje upravljačke odgovornosti za dovršenje svake od njih. Nakon što je raspored aktivnosti napravljen, moguće im je dodijeliti resurse i troškove. Budući da su ta oba alata izvedena iz aktivnosti u logičkom okviru, postoji i direktna logička veza između detaljne razrade plana i projektnih ciljeva.



Slika 34: Raspored aktivnosti i dodjela resursa

#### 7.1.1. Lista rasporeda aktivnosti

Kad je logički okvir dovršen moguće je ćeliju u kojoj je ispisana lista aktivnosti kopirati u obrazac za raspored aktivnosti. Taj obrazac treba da bude podešen na očekivano vrijeme realizacije projekta. Aktivnosti koje će se realizovati u prvoj godini mogu biti predstavljene s više detalja tako da pokazuju početak i kraj aktivnosti u konkretnoj sedmici, a u narednoj godini se kao vremenska jedinica mogu koristiti mjeseci. Ovo treba shvatiti kao prvu procjenu, koju

će kasnije mijenjati i podešavati naredni učesnici u projektu. Podaci iz prve procjene predstavljaju baznu liniju i pomoćno sredstvo za dodjelu resursa i troškova. Koraci u tome su:

**Korak 1 – Izrada liste glavnih aktivnosti.** Glavne aktivnosti – navedene u logičkom okviru – predstavljaju sažetke faza rada koje moraju biti izvršene da bi se ostvarili projektni ciljevi. Njih je moguće – sada – koristiti kao okvir unutar kojeg će trebati razraditi detalje te faze.

**Korak 2 – Razrada aktivnosti u upravljive zadatke.** Cilj razlaganja glavnih aktivnosti ili faza rada u detaljnije definisane pod-aktivnosti ili zadatke je da se dobiju dovoljno jednostavni zadaci koji se lako organizuju i izvršavaju. Svakom tako definisanom zadatku se može dodijeliti njegov kratkoročni cilj.

Glavna vještina ove razrade je u dobijanju tražene dubine razrade. Najčešća greška koja se pravi kod razrade projekata je u prevelikoj usitnjenosti zadataka. Ovo znači da razradu treba zaustaviti na onom nivou na kojem je moguće zadacima dodijeliti potrebne resurse i vrijeme trajanja te – generički definisano – lice koje će raspolagati dovoljnim informacijama da zna šta treba učiniti da bi obavilo posao.

**Korak 3 – Utvrđivanje redoslijeda i zavisnosti.** Nakon što su aktivnosti dovoljno detaljno razložene – treba ih povezati:

- **Prema redoslijedu (to sequence)** – definisati redoslijed kojim će svaka od njih biti izvođena
- **Prema zavisnosti** – odrediti koji zadatak mora biti završen da bi se mogao početi onaj koji se trenutno razmatra i zadaci koje dolaze kasnije a mogu se realizovati tek kad je konkretni završen.

To se može najlakše shvatiti na konkretnom primjeru. Izgradnja kuće sastoji se od brojnih – posebnih ali međusobno zavisnih – aktivnosti: iskop i izrada temelja, zidanje zidova, ugrađivanje vrata i prozora, malterisanje zidova, izrada krovne konstrukcije i pokrivanje, izrada vodovodnih i kanalizacionih instalacija itd. Prema redoslijedu, temelji se moraju iskopati i napraviti – da bi se zidovi mogli zidati; a zavisnosti pokazuju da ugradnju vrata i prozora nije moguće početi prije nego što je gradnja zidova dosegla određenu visinu; ili da malterisanje ne možete završiti prije postavljanja vodovodnih i kanalizacionih i drugih instalacija.

**Korak 4 – Procjena početka, trajanja i dovršetka aktivnosti.** Specifikacija vremena podrazumijeva realističnu procjenu trajanja svakog zadatka, unošenje te procjene u trajanje konkretnog zadatka, zatim određivanje dana početka i njegovog kraja. To vrijeme trajanja zadatka ponekad nije jednostavno odrediti. Ukoliko to ne možemo sami učiniti s dovoljnom pouzdanošću treba da konsultujemo tehnička lica koja bolje poznaju konkretnu materiju.

Nepouzdanost ovih procjena je česta greška koja dovodi do potcjenjivanja ili precjenjivanja potrebnog vremena za ostvarenje projekta, a uzrokovana je:

- Ispuštanjem ključnih aktivnosti ili zadataka
- Izostavljanjem neke linije zavisnosti iz sistema međuzavisnosti zadataka
- Dopuštanjem da zadaci konkurišu jedan drugom na isti resurs (rasporedom jednog lica na više zadataka, ili nekih elemenata opreme na dva ili više zadataka.
- Željom da se ostavi dobar utisak obećanjem brzog dovršetka.

**Korak 5 - Sažimanje poretka glavnih zadataka.** Nakon što su specificirana vremena pojedinačnih zadataka od kojih se sastoji neki glavni zadatak ili faza korisno je napraviti sažetak za tu konkretnu fazu, identifikacijom njenog početka, trajanja i završetka.

**Korak 6 - Utvrditi kontrolne tačke (Milestones, miljokazi)** Kontrolne tačke predstavljaju osnovicu za praćenje i upravljanje projektom. To su ključni događaji koji postaju mjerilo napredovanja projekta i istovremeno cilj za članove projektnog tima. Najjednostavniji način definisanja kontrolnih tačaka su datumi dovršetka neke od aktivnosti, npr., procjena potreba u obuci – dovršene do januara 20..godine.

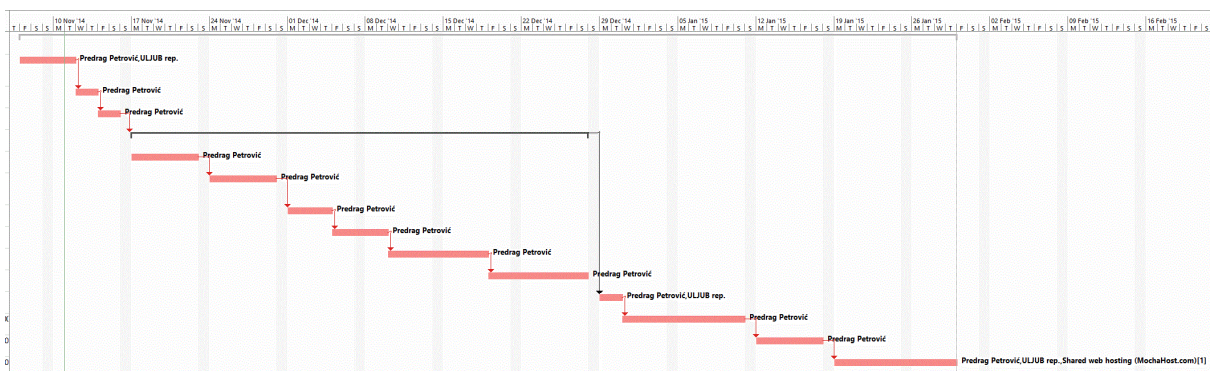
**Korak 7 – Identifikacija stručnjaka (expertise).** Kad su zadaci poznati moguće je identifikovati vrstu stručnjaka potrebnog za ostvarenje zadatka. Vrsta stručnjaka nam je često unaprijed poznata, ali je dobro iskoristiti ovu priliku za provjeru izvodljivosti akcionog plana u pogledu raspoloživosti ljudskih resursa.

**Korak 8 – Raspored zadataka u timu.** Ovdje nije u pitanju samo izjava o tome šta ko radi. S dodjelom zadataka dodjeljuje se i odgovornosti dostizanja kritičnih tačaka. Drugim riječima to je sredstvo kojim se definiše odgovornost svih članova tima uključujući njegovog vođu. Zbog toga se pri dodjeli zadataka mora uzeti u obzir sposobnost, znanja i iskustvo svakog člana tima. Pri dodjeljivanju zadataka članovima tima treba osigurati da svako od njih razumije šta se od

njega očekuje. Ukoliko to nije slučaj detalji koji opisuju konkretni zadatak moraju biti dublje razrađeni.

### 7.1.2. Prezentacija redoslijeda aktivnosti

Sve informacije o redoslijedu aktivnosti mogu biti prikazane grafički. Taj format se zove **Gantov dijagram (Gantt Chart)** Jedan njegov primjer dat je slikom 35. Format se može prilagođavati očekivanom trajanju projekta. Pregled opšteg karaktera može koristiti kvartalnu ili mjesečnu vremensku jedinicu a pojedinačni kvartalni radni plan – sedmicu kao vremensku jedinicu.



Slika 35: Primjer Gantvog dijagrama

## 7.2. Dodjela resursa aktivnostima.

Procjena troškova mora biti temeljena na pažljivom i sveobuhvatnom planiranju budžeta. To ima znatnog uticaja na odluku o investiranju u postupku ocjene projekta i za smireno izvođenje radova nakon što je dat znak za početak. Osim toga, lista aktivnosti treba biti kopirana u obrazac inputa i raspored aktivnosti. U tom slučaju svaka aktivnost može biti korištena za provjeru da li su sva nužna sredstva pribavljena.

### 7.2.1. Kontrolna lista specifikacije sredstava i redoslijeda

Nakon što su sve aktivnosti unesene u raspored potrebno je specificirati sredstva za njihovo izvođenje. Ta sredstva treba da su raspoređena u odgovarajuće kategorije troškova kako bi se omogućilo njihovo agregira-nje i rezimiranje informacija o troškovima. Npr., na slici 36 aktivnost osnivanja jedinice za planiranje podrazumijeva opremu, plate i naknade. Ovdje je potrebno definisati **Jedinicu mjere, Količinu za period, Troškove po jedinici i Troškove projekta.**

Korak 1: Kopiranje aktivnosti	Korak 2: Dodjela resursa	Korak 3: Unos kategorije troškova	Korak 4: Unos jed. mjere	Korak 5: Unos količina	Korak 6: Unos jed. troškova	Korak 7: Unos izvora finansir anja	Korak 8: Unos koda troškova	Korak 9: Raspored troškova	Korak 10: Sabiranje									
Aktivnosti/Resursi			Jed.mj ere	Količina po periodu				Troškovi po jed.	Izvor finans.	Kod troškova		Troškovi za period				Svega	Period.tr ošk./god.	
				Q1	Q2	Q3	Q4			EU	Vlada	Q1	Q2	Q3	Q4			
1.1 Dizajn i izvođenje programa obuke osoblja u brizi o pacijentima																		
Oprema																		
Kompjuter			kom	2				1000	EU	3.4	A/1.3	2000				2000		
Kopir aparat			kom	1				5000	EU	3.4	A/1.4	5000				5000		
Štampač			kom	2				500	EU	3.4	A/1.5	1000				1000		
Plate i naknade (lokalno osoblje)																		
Osoblje partnerske države			mjesec	6	6	6	6	1700	Vlada	5.2	B/2.1	10200	10200	10200	10200	40800		
Kancelarijsko osoblje			mjesec	3	3	3	3	900	Vlada	5.2	B/2.2	2700	2700	2700	2700	10800		
Itd.																		

Slika 36: Lista specifikacije resursa za aktivnosti

Troškovi projekta, također, treba da budu razgraničeni prema **Izvorima finansiranja** tako da je svim učesnicima u finansijskoj konstrukciji poznato šta i kada finansiraju. Budući da su **vrste troškova kodirane** moguće ih je dalje agregirati do kraja.

Sada je moguće **terminirati troškove** korištenjem jednostavnih formula kao što je množenje godišnje količine s jediničnim troškovima. Nakon što su izračunati **Ukupni troškovi** treba imati na umu da agencija za implementaciju treba pokriti i troškove održavanja po isteku vijeka projekta. **Ti – periodični – troškovi (Recurrent Costs)** mogu djelimično ili u cjelosti biti pokriveni iz povećanog prihoda generiranog iz projektnih aktivnosti. Bez obzira kako će se to dogoditi, ove troškove treba specificirati da bi se mogao kalkulisati njihov uticaj na budžet agencije za implementaciju.

Sažetak ključnih pojmova poglavlja:

- ✓ Raspored aktivnosti je metoda predstavljanja aktivnosti projekta, kojom se definiše redoslijed izvođenja i bilo kakve zavisnosti koje mogu postojati između njih. On služi i kao sredstvo identifikacije odgovornosti za izvođenje aktivnosti.
- ✓ Nakon što je dovršen logički okvir, postaje moguće kopiranje istih iz krajnjeg lijevog stupca u obrazac rasporeda aktivnosti. U tom obrascu se vrše preliminarne procjene koje će kasnije biti mijenjane od strane projektnog tima u skladu s dinamikom izvršenja.
- ✓ Po završetku rasporeda aktivnosti svakoj od njih treba dodijeliti inpute i definirati troškove. Troškovi moraju biti pažljivo procijenjeni i budžetirani. Oni će znatno uticati na donošenje odluke o finansiranju i služiti za nesmetano izvođenje projekta nakon što donesena odluka o njegovom početku.
- ✓ Lista aktivnosti mora biti kopirana u obrazac nabavki i rasporeda troškova. Svaka od tih aktivnosti treba da se koristi kao kontrolna lista za osiguranje potrebnih sredstava namijenjenih realizaciji aktivnosti.

## 8. Korištenje logičkog okvira za ocjenu prijedloga projekata

U ovom poglavlju se objašnjava način na koji se logički okvir koristi u za ocjenu projektnih prijedloga sa ciljem otkrivanja slabosti u prijedlogu projekta i za formulisanje pitanja za pripremne studije. Data su i uputstva ocjenu kvaliteta finansijskog prijedloga.

### 8.1. Uvod

U pripremnim fazama se pristup pomoću logičkog okvira koristi kao alat za planiranje. Međutim, on je i moćan alat i za naknadnu analizu projektnih prijedloga, jedina razlika je u tome što se umjesto izvora informacija o problemu – koristi prijedlog projekta.

Cilj primjene ovog pristupa na prijedlog projekta je da se otkriju **RELEVANTNOST, IZVODLJIVOST ILI ODRŽIVOST** projekta, na osnovu dodatne analize ili iz postojećih izvora. Treba uočiti da se ta tehnika primjenjuje u kancelarijskoj analizi postojećih prijedloga. i da ne postoji drugi način u participativnom planiranju kojim se može ući u srž osim pristupa pomoću logičkog okvira.

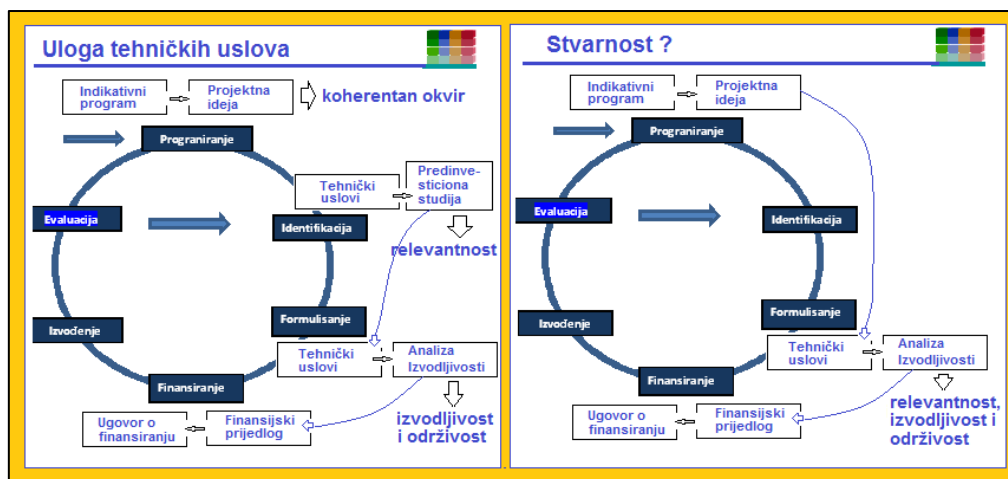
Programi obuke u upravljanju projektnim ciklusom nude dva alata za ocjenu projektnih prijedloga:

1. **Vodič za ocjenu projektnih prijedloga**, koji ima za cilj dublju analizu projektnog prijedloga prije faze formulacije. Njegov zadatak je da pomogne identifikaciji pitanja i tema koje treba da budu uključene u tehničke uslove (Terms of Reference) za izvođenje studije izvodljivosti.
2. **Alat za ocjenu kvaliteta**, kojim se provjerava kvalitet nacrtu prijedloga projekta prije njegovog podnošenja nadležnom finansijskom tijelu.

### 8.2. Vodič za ocjenu projektnog prijedloga

U idealnom slučaju, pojmovi relevantnosti, izvodljivosti i održivosti se koriste dvaput tokom pripreme projekta – jednom u Fazi identifikacije (kao dio preinvesticione studije – Pre-feasibility Study), a zatim detaljnije u Fazi formulacije (kao dio analize izvodljivosti). U praksi se često događa da se sve uzima za gotovo – ako je provedena samo jedna faza te analize, uz etiketu „korektno urađeno“ (ready-made). U odsustvu dvostruke provjere pripremljenosti prijedloga veoma je važno da procesni menadžeri osiguraju kvalitet tehničkih uslova u obliku paralelnog pregleda („one-shot“ exercise).





Slika 37: Uloga tehničkih uslova u pripremi projekta

Vodič za ocjenu sadrži **šest uputstava** kao okvir za analizu koherentnosti i potpunosti projektnog prijedloga. Taj pristup može odvesti do razgradnje i rekonstrukcije dizajna projekta sa ciljem da se otkriju praznine i nekonzistentnosti i, zatim, identifikuju pitanja koja treba uključiti u tehničke uslove za izradu analize izvodljivosti. Alat je podesan i za unošenje prijedloga u obrazac logičkog okvira ukoliko ga organizacija koja je podnijela prijedlog projekta nije ranije koristila.

### Uputstvo 1: Analiza ciljeva i problema

- ✓ Identifikovati i napraviti listu problema i ciljeva u dokumentu
- ✓ Izraditi piramidu problema
- ✓ Uporediti piramidu ciljeva s analizom problema
- ✓ Formulirati pitanja vezana za adekvatnost analize problema i za logičke praznine ili nekonzistentnosti u stablu ciljeva

Posebnu pažnju treba posvetiti pitanju koliko obuhvatno su obrađeni problemi i ograničenja ciljne grupe i drugih partnera u projektu. Razmjere u kojima analiza problema podržava ciljeve formulisane u prijedlogu će tada dati vidljivu osnovu za zaključak o tome da li su projektni ciljevi dobro podešeni identifikovanim potrebama. Ako se to u prijedlogu nije uspjelo – treba to pitanje uključiti u tehničke uslove studije izvodljivosti.

### Uputstvo 2: Identifikovati logiku intervencije i pretpostavke

- ✓ Identifikovati svrhu projekta iz piramide ciljeva
- ✓ Formulirati svrhu i ciljeve projekta, rezultate i aktivnosti

- ✓ Identifikovati pretpostavke iz projektnog prijedloga
- ✓ Procijeniti logičku koherentnost logike intervencije i pretpostavki
- ✓ Formulirati pitanja o logici i koherentnosti ciljeva

Prijedlog može ne biti pripremljen pomoću logičkog okvira ili da postavljeni ciljevi nisu saglasni s definicijama za Svrhu i Ciljeve projekta, Rezultate i Aktivnosti. U oba slučaja treba razjasniti logiku intervencije projekta uključujući rizike i pretpostavke s kojim će se susresti tokom realizacije.

Koristeći se piramidom ciljeva napravljenim prema Uputstvu br.1 kartice ciljeva treba smjestiti u matricu logičkog okvira tako da odgovaraju gore pomenutim definicijama Svrhe i Ciljeva, Rezultata i Aktivnosti. Mada neka od pitanja vezanih za logičku saglasnost mogu time biti identifikovana, primjenom logike intervencije i odgovarajućih pretpostavki moguće je preciznije provjeriti izvodljivost projekta. Na primjer:

- ✓ Svrha projekta objašnjava kako je projekt usklađen s državnim ili sektorskim ciljevima Komisije i partnerskih vlada ili institucija. Ako veza između projekta i ciljeva višeg reda nije vidljiva tada je potrebno pribaviti detaljnije informacije o relevantnosti projekta za sektorske ili programe države.
- ✓ Na bazi informacija iz projektnog prijedloga treba biti moguće da se formuliše Cilj projekta tako da opisuje koriti koje će imati ciljna grupa iz projektnog prijedloga (Rezultat). Ako to ni sada nije moguće, trebat će pribaviti nove informacije koje će dokazati relevantnost projekta za potrebe ciljne grupe.
- ✓ Ako cilj projekta može da se korektno formuliše, ali rezultat projekta izgleda nedovoljan da opravda očekivane koristi, treba pribaviti nove informacije o tome odakle mogu da se dobiju zahtijevane dodatne usluge (kao iz projekta).
- ✓ Akcije i aktivnosti projekta možda nisu logički dovoljne da se postigne željeni rezultat. Tada će trebati pribaviti dodatne informacije o tome kako je moguće osigurati očekivani Rezultat.
- ✓ Projekt može sadržavati opis vanjskih faktora koji se nalaze izvan projektnog okvira a ipak utiču na dostizanje projektnih ciljeva. Tada je moguće – na osnovu iskustava s takvim ili sličnim projektima identifikovati druge faktore. Svi oni treba da budu uključeni u logički okvir i prikazani u pozitivinom značenju – kao da su realizovani.

Nikad, dakle, ne oklijevajte tražiti prave informacije. Osim ocjene projektne logičke saglasnosti uvijek je potrebno osvrnuti se na ranije dokaze o uspjehu i promašajima i time koristiti lekcije iz prošlosti.

### Uputstvo 3: Ocjena pretpostavki

- ✓ Ocijeniti ulogu pretpostavki na osnovu algoritma pretpostavki
- ✓ Sugerisati modifikacije dizajna projekta
- ✓ Sugerisati pitanja o vanjskim faktorima

Cilj analize vanjskih faktora je da se utvrdi njihov uticaj na izvodljivost projekta, u pogledu stepena riskantnosti. Ovo zahtijeva razumijevanje relativnog značaja svakog od faktora za uspjeh projekta i vjerovatnoće da će se dogoditi. Ako postoji dovoljno podataka u prijedlogu o značaju i vjerojatnoći pretpostavki, tada se postavlja pitanje da li su ti faktori dovoljno istraženi tokom analize izvodljivosti. Možda u prijedlogu postoje informacije koje će voditi ka modifikaciji prijedloga. U tom slučaju će – u tehničkim uslovima analize izvodljivosti – morati biti uključeno i pitanje validnosti modifikacija.

### Uputstvo 4: Ocjena održivosti

- ✓ Identifikovati rezultate i aktivnosti koje će se nastaviti tokom upotrebnog vijeka projekta
- ✓ Ocjeniti izgled za održivost
- ✓ Formulirati pitanja o održivosti rezultata i aktivnosti

Ovdje se radi o dugoročnom aspektu pri oblikovanju projekta. Ako su koristi od projekta održive, tada bi trebalo računati s tim da će usluge iz njega biti pružane tokom upotrebnog vijeka. Cilj ocjene održivosti je da se identifikuju usluge (Rezultati) koje će se nastaviti i otkrije da li su ugrađeni u to dovoljni mehanizmi da se osigura njihova trajnost.

Ako se pokaže da analiza šest faktora nedovoljna za odgovor na pitanje održivosti, tada u studiji održivosti treba izvršiti dodatna istraživanja. Na primjer, izvedbena agencija treba da raspolaze dovoljnim znanjima i resursima ukoliko se ona stara za isporuku usluga. Ako nema dovoljno

informacija da se to dokaže, tada studijom izvodljivosti treba provjeriti resurse i sposobnost agencije i možda u oblikovanju projekta predvidjeti dodatne aktivnosti kojima će kapaciteti agencije biti unaprijeđeni.

### **Uputstvo 5: Utvrditi indikatore**

- ✓ Otkriti indikatore za Svrhu, Projektne ciljeve i Rezultate u prijedlogu projekta
- ✓ Ocijeniti indikatore sa stanovišta mjerljivosti količine, kvaliteta i vremena (QQT)
- ✓ Formulirati pitanja o mjerenju performansi

Indikatori stvaraju osnovicu za mjerenje učinka projekta i treba da budu mjerljivi uz prihvatljive troškove i u kapacitetu agencije koja projekt izvodi. Ako to nije moguće, treba postaviti nova pitanja u tehničkim uslovi-ma studije izvodljivosti.

### **Uputstvo 6: Priprema tehničkih uslova**

- ✓ Pregledati pitanja identifikovana tokom koraka 1-5
- ✓ Razmjestiti pitanja prema Relevantnosti, Izvodljivosti, Preduslova i Održivosti
- ✓ Ugraditi pitanja u **Odjeljak D: Teme koje treba proučiti** u Tehničkim uslovima za studiju izvodljivosti.

Nakon što je ocjena dovršena, identifikovana pitanja i teme treba složiti po prioritetima a zatim ugraditi u tehničke uslove studije izvodljivosti. Cilj studije izvodljivosti je da donosiocima odluke u vladama i Evropskoj Komisiji pruže dovoljno informacija za opravdanje prihvatanja, modifikacije ili odbacivanja projekta predložene-nog za finansiranje i izvođenje.

Ključni proizvod studije izvodljivosti je ocjena relevantnosti, izvodljivosti i održivosti predloženog projekta i detaljan plan rada zasnovan na strukturi logičkog okvira. Na slici 27 dat je okvirni sadržaj u zajedničkoj upotrebi u nekim od RELEX servisa.

## Indikativni sadržaj Tehničkih uslova za studiju izvodljivosti Usluge Ureda za pomoć i obuku u Upravljanju projektnim ciklusom

### Sadržaj

A. POZADINA STUDIJE .....	1
B. CILJEVI STUDIJE .....	1
C. REZULTATI STUDIJE .....	1
D. TEME KOJE TREBA PROUČITI .....	2
I) Relevantnost .....	2
II) Izvodljivost .....	2
III) Preduslovi .....	2
IV) Održivost .....	2
E. PLAN RADA .....	3
F. ZAHTIJEVANI NALAZI .....	4
G. IZVJEŠTAVANJE .....	4
H. VREMENSKI RASPORED .....	4
I. POMOĆ UGOVORNIH ORGANA KONSULTANTIMA .....	4
Dodataj 1. Okvрни sadržaj izvještaja o studiji izvodljivosti .....	5
1. Sažetak .....	5
2. Pozadina .....	5
3. Intervencije .....	6
4. Pretpostavke .....	6
5. Izvođenje .....	7
6. Faktori koji osiguravaju održivost .....	7
7. Monitoring i evaluacija .....	9
8. Zaključci i prijedlozi .....	9
Tehnički dodaci .....	9
Administrativni dodaci .....	9

### 8.3. Alati za ocjenu kvaliteta

Nakon analize izvodljivosti pravi se Izvještaj o studiji izvodljivosti i nacrt Finansijskog prijedloga. Za njihovu izradu također postoje alati, za osiguranje njihovog kvaliteta, posebno Finansijskog prijedloga koji treba da služi kao osnovica za finansijsku odluku.

Do ove faze oblikovanja projekta prijedlog je obrađivan u obrascu Logičkog okvira i taj proces ovdje nije nužno ponavljati. Umjesto njega će biti korišten alat Procjene kvaliteta i pristup pomoću kontrolne liste (engl. Checklist approach) kojim se razlažu ključne komponente relevantnosti, izvodljivosti i održivosti u obična pitanja i gradi okvir za brzu identifikaciju šupljina u informacijama iz Finansijskog prijedloga (Slika 38).

**Ocjena kvaliteta projektnog prijedloga  
„Akcija obuke u Upravljanju projektnim ciklusom  
Modul 3: PCM napredna obuka  
Dokument obuke**

Sadržaj

CILJ OCJENE KVALITETA.....	1
PARAMETRI OCJENE KVALITETA .....	1
1. Relevantnost.....	2
1.1 Da li su korisnici projekta jasno identifikovani? .....	2
1.2 Da li su problemi korisnici opisani dovoljno ? .....	2
1.3 Da li je analiza problema dovoljno obuhvatna ? .....	2
1.4 Da li Svrha projektna objašnjava zašto je projekt važan za društvo ? .....	3
1.5 Da li je projektni cilj definisan u terminima korisiti za ciljnu grupu ? .....	3
1.6 Postoji li potreba da rezultati bududemonstrirani ? .....	4
2. Izvodljivost .....	4
2.1 Da li prtojekt doprinosi svrsi (ako su pretpostavke tačne) ? .....	4
2.2 Jesu li rezultati opisani kao usluge koje će biti isporučene ciljnoj grupi ? .....	5
2.3 Da li će projektni cilj biti ostvaren <u>ako</u> se isporuči rezultat ? .....	5
2.4 Jesu li sredstva dovoljno opravdana pomoću kvantificiranih ciljeva ? .....	5
2.5 Da li su identifikovani važni vanjski uslovi projekta ? .....	6
2.6 Je li vjerovatnoća ostvarenja pretpostavki prihvatljiva ? .....	6
2.7 Da li je agencija za implementaciju projekta u stanju izvesti projekt ? .....	7
3. Održivost .....	7
3.1 Da li relevantne vlasti imaju politiku podrške projektu u njegovom upotrebnom vijeku ? .....	7
3.2 Da li je tehnologija prilagođena lokalnim uslovima ? .....	7
3.3 Da li će okoliš biti zaštićen u toku izvođenja i nakon puštanja u projekta u upotrebu ? .....	8
3.4 Postoji li sklonost projektnih korisnika da preuzmu projekt kao svojinu ? .....	8
3.5 Da li žene (i druge grupe ) iamju koristima i proizvodnim faktorima tokom i nakon projekta ? .....	9
3.6 Da li će agencija za implementaciju biti u stanju da osigura nastavak (follow-up) projekta ? .....	9
3.7 Da li finansijska i ekonomska analiza potvrđuju da je ovaj projekt efikasan, održiv i relevantan ? .....	10

Slika 38: Alat za ocjenu kvaliteta

ocjenu kvaliteta

## 1.1 Da li su korisnici jasno definisani

Jasan opis korisnika treba, u najmanju ruku, uljuči stav o njihovoj ekonomskoj i socijalnoj ulozi / položaju i geografskoj lokaciji. Zavisno od projekta i druge informacije mogu imati značaja kao što su nivo obrazovanja i osposobljenosti, vlasništvu i/ili pristupu resursima itd. Spolna slika ovih informacija je bitna radi osiguranja zadovoljenja potreba obaju polova. Starost, etnička pripadnost i druge socijalne osobine također mogu biti relevantne za sadržaj projekta.

**Indikator za rangiranje: Korisnici su jasno identifikovani ...**

	Kada ...
Potpuno	Korisnici su oopisani u detalje, uključujući socio-ekonomsku ulogu i položaj, geografsku lokaciju, polni sastav i navedeni drugi faktori
Prilično	Opis sadrži socio-ekonomske informacije, geografsku lokaciju i polni sastav, ali manjkaju detalji.
Slabo	Navedeni su samo neki elementi
Nikako	Nikakva specifična uloga niti lokacija nisu pomenuti.

Sažetak ključnih pojmova poglavlja:

- ✓ Logički okvir primjenjuje se u pripremi projekta. Ipak, to ostaje moćno sredstvo analize oblikovanja i razrade procesa u cjelini.
- ✓ Cilj primjene pristupa pomoću logičkog okvira je da se otkriju slabosti i praznine u dizajnu projekta. Te praznine odnose se na RELEVANTNOST, IZVODLJIVOST I ODRŽIVOST projekta.
- ✓ Upravljanje projektnim ciklusom nudi dva alata za ocjenu projektnih prijedloga: i) **Vodič za ocjenu** koji služi za dublju analizu prijedloga prije faze formulacije; i ii) **Alat za ocjenu kvaliteta**, kojim se ocjenjuje kvalitet nacрта finansijskog prijedloga prije njegovog podnošenja odgovarajućem finansijskom tijelu na odlučivanje.
- ✓ Vodič za ocjenu sadrži 6 uputstava koja daju okvir za analizu koherentnosti i obuhvatnosti projektnog prijedloga. Pristup je u razlaganju i rekonstrukciji projektnog dizajna sa ciljem da se otkriju praznine i nesaglasnosti i odatle identifikuju pitanja za uključivanje u tehničke uslove za studiju izvodljivosti.
- ✓ Alat za ocjenu kvaliteta nudi pristup pomoću kontrolne liste koja razlaže ključne komponente relevantnosti, izvodljivosti i održivosti u jednostavna pitanja i tako pruža okvir za brzu identifikaciju informacionih praznina u finansijskom prijedlogu.

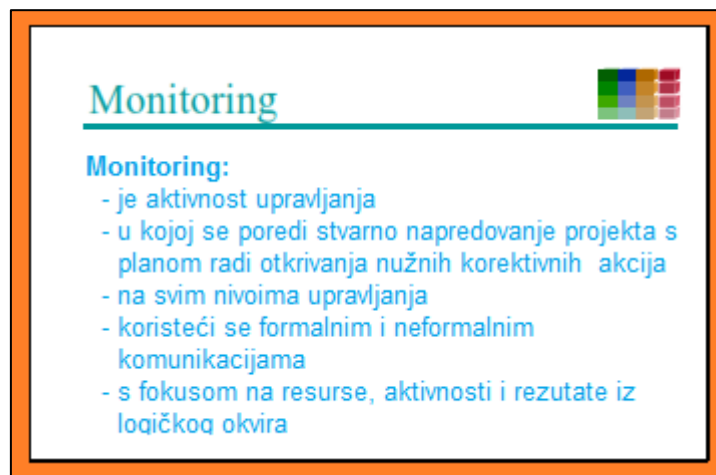


## 9. Monitoring i izvještavanje

Ovo poglavlje definiše monitoring i objašnjava njegovu ulogu u upravljanju projektom. U njemu se prikazuju osnovni koraci u oblikovanju sistema vođenja projekata i iznose glavne koristi efikasnog monitoringa kao i promašaji koje treba izbjeći.

### 9.1. Uvod

Nakon planiranja i osiguranja finansijske podrške projektu dolazi najvažniji dio projekta – izvođenje. Rijetko koji projekt se dovrši striktno onako kako je planiran. Često se, u praksi, događa da projekt dobije pravac i dinamiku koja nije očekivana tokom planiranja. Upravljački tim projekta ovdje ima važan i težak zadatak da održi dovoljnu kontrolu nad projektom kako bi bio zadržan na tragu ostvarenja predviđenih ciljeva. To se ostvaruje **monitoringom**, koji se može definisati kao sistematsko i neprekidno prikupljanje, analiza i korištenje informacija za kontrolu upravljanjem i donošenje odluka.



Slika 39: Monitoring

Monitoring projekta je integralni dio svakodnevnog upravljanja. Cilj mu je da osigura informacije pomoću kojih upravljački tim može identifikovati i riješiti izvođačke probleme i ocijeniti napredovanje u odnosu na ono što je originalno planirano.

Tok informacija između projekta i Komisije je predmet posebnog sistema koji funkcioniše na agregiranijem ili institucionalnom nivou. **Ovdje neće biti riječi o tom nivou monitoringa, jer je – ovdje – fokus na nivou projekta.** Ovaj sistem monitoringa je oblikovan radi prikupljanja sumarnih informacija o svim projektima kojima upravlja Komisija.

## 9.2. Oblikovanje sistema monitoringa

Držeći fokus na ciljevima projekta, sistem monitoringa na nivou projekta mora proći kroz pet koraka:

1. **Analiza ciljeva projekta** - treba da razjasni dizajn projekta. Dobar monitoring veoma je zavisan od jasnoće postavljenih ciljeva. Pristup pomoću logičkog okvira ovdje osigurava da su ciljevi korektno napisani, a da su aktivnosti definisane tako da vode ka predviđenim rezultatima i ciljevima. Logički redoslijed ovdje pojednostavljuje izbor indikatora praćenja.
2. **Pregled procedura izvođenja** - omogućava utvrđivanje potreba za informacijama na različitim nivoima upravljačke strukture. Dubina zahtijevanih informacija i njihova učestalost izvještavanja varira zavisno od nivoa upravljanja. Ustvari, ovim korakom se spajaju potrebe za informacijama sa ulogom u donošenju odluka.
3. **Provjera indikatora** - služi za mjerenje u približavanju ciljevima. U timu za izvođenje projekta primaran fokus je na fizičkom i finansijskom praćenju aktivnosti i rezultata. Alati za indikatore su snimci stvarnih učinaka radi poređenja troškova s predviđenim budžetom i fizičkog napredovanja s redoslijedom aktivnosti.
4. **Oblikovanje obrazaca za izvještavanje** - radi dobijanja relevantnih i blagovremenih informacija na različitim nivoima upravljanja i radi olakšavanja analize.
5. **Priprema plana implementacije sistema monitoringa** - kojim treba definirati osoblje, zahtijevana znanja i vještine i propisati sistem izvještavanja i odgovornosti u vezi s njim.

### Analiza projektnih ciljeva

Analiza ciljeva već je vršena u odjeljku 3.1. Pa ipak, ponekad protekne duži period od osmišljavanja do realizacije, nekad se promijene okolnosti ili partneri. Tada ima smisla implementaciju početi uvodnom radionicom na kojoj će partneri projekta pregledati projektne dokumente i ključne pretpostavke. Projektne ciljeve treba revidovati tako da se osigura njihova jasnoća i realističnost, specifičnost i mjerljivost. Kad se to učini biće stvorena osnovica za sistem monitoringa i evaluacije.

#### 9.2.1. Pregled procedura implementacije

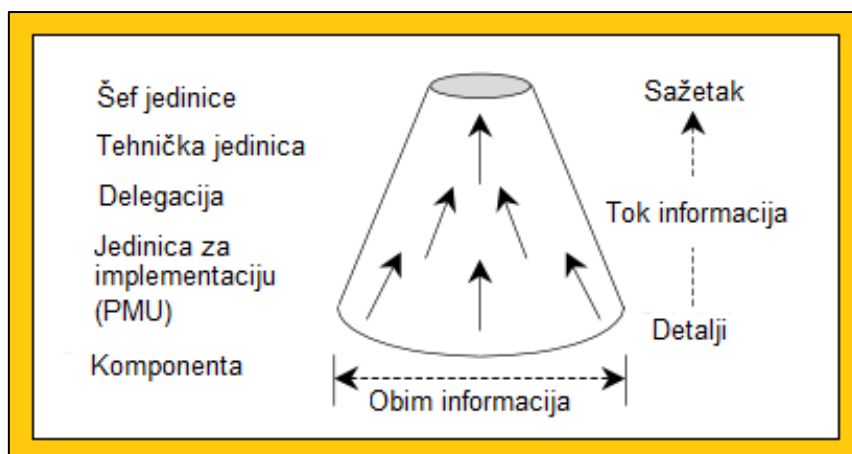
Logički okvir predstavlja okvir za identifikaciju potreba za informacijama u cjelini. Zbog toga je važno da informacione potrebe budu prilagođene različitim nivoima upravljačke strukture.

U stvarnosti dubina i učestalost informacionih detalja veoma zavisi od upravljačke strukture. Administrator projekta će zahtijevati dnevne informacije, a ugovarač njihove sažetke o postignućima rezultata ili devijacijama u odnosu na plan rada. Slika 40 ilustruje taj princip. Pregled procedura izvođenja radova ima za cilj da se definiše koja aktivnost treba biti izvedena i ko će to učiniti. To treba učiniti oslanjajući se na raspored aktivnosti.

Pregledom procedura izvođenja (ko šta radi) u saradnji s osobljem izvedbene agencije treba razjasniti razlute uloge, funkcije i odgovornosti kako bi se uspostavila veza između potreba za informacijama i nivoa upravljanja. Taj proces može biti izveden pomoću tabele u kojoj je naveden korisnik informacije, šta se zahtijeva, izvor informacija i ko je odgovoran podnijeti izvještaj.

Uspješno izvještavanje je zavisno od međusobnog razumijevanja korisnika i izvještača. Ali taj pristup je nedovoljan i ima slabosti: prva - jer se pretpostavlja da korisnik unaprijed zna šta mu treba, i druga – korisnik teži da dobije više informacija nego mu realno treba. Obje situacije se najčešće javljaju u ranim fazama realizacije projekta – kad uloge nisu dovoljno razjašnjene. Taj nesporazum se neće riješiti prije nego postane vidljiva treća slabost – da korisnik nema predstavu o tome koje mu informacije stoje na raspolaganju. Da bi se nesporazumi ove vrste prevazišli lica koja su zadužena za monitoring i evaluaciju (M&E) treba da vrše stalan pregled korisničkih zahtjeva kroz:

- ✓ Prisustvo sastancima za planiranje i dogovore da bi uočili šta od informacija nedostaje ili je suvišno za efikasno donošenje odluka
- ✓ Ohrabrenje dijaloga o sadržaju i formi izvještaja između korisnika i izvještača.



Slika 40: Potrebe za informacijama i nivoi upravljanja

### 9.2.2. Pregled indikatora

Izbor indikatora je već diskutovan u jednom od predhonih poglavlja. Međutim, često se u evaluacijama ističe da su indikatori slabo odabrani. Njihov zajednički problem je:

- **Odabir previše indikatora** – ljudi često imaju intenciju da precijene količinu informacija nužnih za donošenje odluka. Specifikacija potreba za informacijama podrazumijeva kompromis između količine informacija potrebnih za donošenje odluka i količine informacija koje donosilac odluke može praktično pregledati i analizirati. Također se prečesto događa da menadžer precijeni vlastite potrebe za informacijama pa na kraju zaključiti da je prosto nemoguće apsorbirati informacije sadržane u izvještajima.

Potrebe za informacijama podešavaju se nivou upravljanja i odabir indikatora to mora odražavati specifikacijom njihovog minimuma. Nešto veći broj indikatora potreban je pri dnevnom upravljanju, a agregirani pokazatelji pogoduju višim nivoima upravljanja.

- **Odabir kompleksnih pokazatelja** – čini najveći problem bilo zbog odsustva znanja ili odsustva potrebnih resursa za prikupljanje istih. Sredstvo za prevladavanje kompleksnim indikatora su kvalitativni (nestandardizovani) indikatori.
- **Indikator previše koncentrisani na napredak projekta** – ne daju dovoljno informacija o izvršenju projekta. Zajednički odgovor na ovu kritiku je da je nemoguće mjeriti uticaj tokom životnog vijeka projekta. Međutim, ako se koriste vodeći indikatori (engl. Leading Indicators<sup>5</sup>) - postaje moguće dobiti jasnu indikaciju o dostizanju ciljeva. Npr., ako su klijenti zadovoljni uslugama dobijenim projektom, tada je jasno da će oni nastaviti te usluge koristiti – a to predstavlja promjenu i njihovom ponašanju koja će se nastaviti u upotrebnom vijeku projekta. Izbor indikatora utjecaja je kritični dio oblikovanja projekta jer može izoštriti definiciju ciljeva i identifikaciju potencijalnih klijenata.

### 9.2.3. Izvještavanje

Menaadžeri projekta vrlo često, sedmično ili „preko noći“ žele utvrditi napredak projekta u pogledu raspoloživog budžeta ili planiranih aktivnosti. Mnogi od tih podataka su operativne naravi i služe za interne potrebe projektnog tima. Za izvještaje o napredovanju se kao ključni

---

<sup>5</sup> Vodeći indikator je približne (orijentacione) naravi i predstavlja zamjenu za indikatore uticaja.

indikatori mogu koristiti malo operativnih podataka i nešto od agregiranih podataka o opremi i materijalima.

Monitoring se ne može smatrati uspješnim samo prikupljanjem potrebnih informacija. One moraju biti i dostavljene u potrebnoj formi pravoj osobi i pravovremeno. Samo tada je, naime, moguće donositi blagovremene odluke i držati projekt pod kontrolom.

U sistemu izvještavanja moraju biti uspostavljeni mehanizmi komunikacije koji osiguravaju da potrebne informacije budu generirane i korištene na pravi način. U vezi s tim postoje dva tipa ovih mehanizama:

- **Izveštaji o napredovanju** – to su periodični sažeci (sedmični, mjesečni ili kvartalni) s fizičkim i finansijskim pokazateljima uključenim u logički okvir, redoslijed aktivnosti i raspored troškova. Nije dovoljna izjava „sve teče po planu“, moraju postojati i dokazi o tome.
- **Uvid u napredovanje** – poređen s planom, može biti pismeni ili oralni – oba otvaraju prostor za ocjenu postojećih rezultata i problema. Međutim uvidi mogu biti prečesti i drastični a tada se upada u iskušenje stalnog vraćanja planu i njegovom podešavanju u skladu s iskustvom. Ovo je prihvatljivo samo do određene tačke – u kojoj ne smije prevagnuti planiranje nad izvršenjem. Također se u periodima krize organizacije više fokusaju na dovršenje zadataka nego na proces kao cjelinu. Bolje je postizati rezultate nego stalno prepravljati plan.

### **Izveštaji o napretku**

Izveštaji o napretku obično su formalizovani i dopuštaju vremenska poređenja. Sadržaj izvještaja mora striktno odgovarati logičkom okviru i rezultatima – npr., redoslijedu aktivnosti, budžetima i redoslijedu troškova. U svakom od njih cilj je poznat: u logičkom okviru nalaze se indikatori napretka (definisani kvalitetom, količinom i vremenom, QQT), a u rasporedu aktivnosti – kontrolne tačke; u rasporedu troškova će troškovi biti procijenjeni i uneseni u kalendar.

Cilj izvještaja o napretku projekta je da se osiguraju ažurni podaci o postignućima – na osnovu pokazatelja i kontrolnih tačaka koristeći se slijedećim okvirom:

Podaci o **namjeranim postignućima** se porede sa

- Podacima o **ostvarenim rezultatima** radi utvrđivanja
  - značajnih **odstupanja od plana**, kao osnovice za
  - identifikaciju **problema i mogućnosti** sa ciljem da se identifikuju

- korektivne **akcije i alternative**.

U tom okviru, izvještajem treba obuhvatiti slijedeće oblasti:

- Sažetak aktuelnog stanja projekta prema indikatorima projektnog cilja i rezultata
- Glavne aktivnosti poduzete tokom izvještajnog perioda upoređene s planom aktivnosti
- Troškove tokom izvještajnog perioda i kumulativ s datumom izvještavanja, upoređen s budžetom i rasporedom troškova
- Procjenu broja opsluženih klijenata ili korisnika tokom perioda
- Tekuće i očekivane probleme uključujući preostale planirane aktivnosti
- Planirane glavne aktivnosti i rokove za naredni period.

Sažetak ključnih pojmova poglavlja:

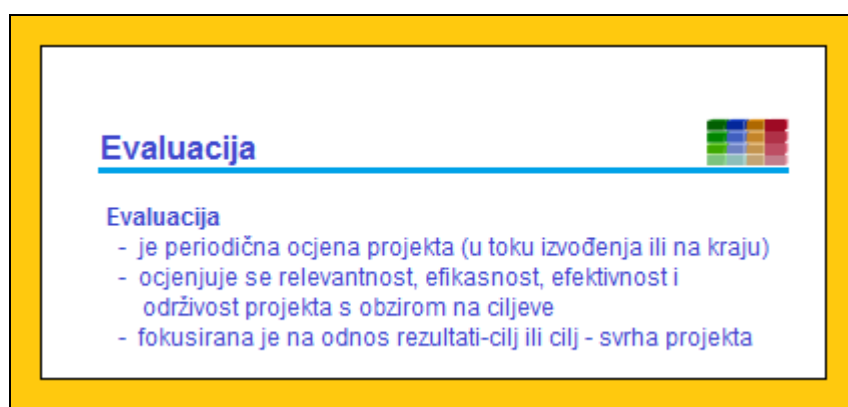
- ✓ Monitoring može biti definisan kao sistematsko i neprekidno prikupljanje, analiza i korištenje informacija sa ciljem kontrole upravljanja i donošenja odluka.
- ✓ Upravljanje projektom je integralni dio dnevnog upravljanja. Cilj mu je da osigura informacije pomoću kojih je moguće identifikovati i riješiti probleme primjene i ostvariti napredak u skladu s onim što je originalno planirano.
- ✓ U oblikovanju i specifikaciji sistema upravljanja postoji pet koraka:
  - Analiza projektnih ciljeva
  - Pregled procedura izvršenja zadataka
  - Provjera indikatora
  - Oblikovanje formi izvještaja
  - Izrada plana izvršenja sistema monitoringa.
- ✓ Potrebe za informacijama treba prilagoditi nivoima upravljačke strukture. U praksi, dubina zahtijevanih informacija i učestalost izvještavanja zavisi od nivoa upravljanja.
- ✓ Ponovnim pregledom procedura izvođenja projekta (šta ko radi) u saradnji s osobljem partnerske institucije razjašnjavaju se uloge, funkcije i odgovornost i stvara veza između potreba za informacijama i nivoa upravljanja.
- ✓ Zajednički problemu uočeni u izboru indikatora uključuju:
  - Izbor previše indikatora,
  - Izbor kompleksnih indikatora
  - Pretjerana koncentracija na pokazatelje progressa.

## 10. Revizija i evaluacija projekta

Ovim poglavljem se definiše evaluacija i glavni kriteriji za ocjenu projekata. Ono povezuje kriterije za evaluaciju s logičkim okvirom i definiše uobičajeni vremenski raspored evaluacija.

### 10.1. Uvod

Evaluacija se može definisati kao periodična ocjena relevantnosti, efikasnosti, efektivnosti, uticaja ekonomske i finansijske održivosti i održivosti projekta u kontekstu zacrtanih ciljeva. Cilj evaluacije je da se utvrde postignuća projekta u odnosu na planirana očekivanja te da se iskustva iz projekta iskoriste u poboljšanju budućih projekata i programa. Evaluacija počiva na rutinskim izvještajima podnesenim tokom implementacije a mogu sadržavati i dodatna istraživanja od strane vanjskih posmatrača ili posebno formiranih tijela.



Slika 41: Evaluacija

### 10.2. Kriterijii evaluacije

Glavna tema koja se tiče svih evaluacija je izbor kriterija. Komisija koristi slijedeće kriterije:

1. **Relevantnost** – prilagođenost ciljeva projekta problemu na koji se odnosi i fizičkom i političkom okruženju u kojem projekt egzistira
2. **Priprema i oblikovanje projekta** – Logika i potpunost planskog procesa projekta i interna logika i koherencija sadržaja projekta
3. **Efikasnost** – troškovi, brzina i upravljačka efikasnost u pretvaranju inputa i aktivnosti u ciljeve i kvalitet postignutog rezultata
4. **Efektivnost** – ocjena doprinosa rezultata ciljevima projekta i načina na koji su pretpostavke uticale na postignuća projekta



5. **Utjecaj** – utjecaj projekta na šire okruženje i doprinos širim sektorskim ciljevima sažetim u svrhu projekta
6. **Održivost** – vjerovatnost nastavka tokova koristi iz projekta, posebno nastavka projektnih aktivnosti i ostvarenju rezultata s posebnim svrhom na faktore političke podrške, ekonomskim i finansijskim faktorima, socio-kulturnim aspektima, spolnim, podešenosti tehnologije, ekološkim aspektima i iinstitucionalnom kapacitetu.

### 10.3. Veza s logičkim okvirom

Koraci u postupku evaluacije tijesno su povezani s hijerarhijskom strukturom cilja u oblikovanju projekta. Svi aspekti postignuća projekta se evaluiraju na osnovu ovog sistemskog pristupa.

#### 10.3.1. Troškovi


**Poređenje stvarnih s planiranim troškovima.** To je osnovica za analizu odstupanja (Variance). Da li je budžet revidovan? Kakva je bila priroda i blagovremenost nabavki u odnosu na plan? Da li su partnerske agencije i drugi donatori ispunili planirana obećanja? Ovi podaci se nalaze u izvještajima o napretku i ti izvještaji su ključni izvor podataka za evaluaciju.

#### 10.3.2. Aktivnosti

**Aktuelni rokovi i dovršetak porede se s planskim.** Je li bilo kašnjenja ili ušteda u vremenu? Koja organizacija je odgovorna za kašnjenje? Kakav je uticaj imala ta promjena na projekt? I ovi podaci nalaze se u izvještajima o napretku projekta.

## Indikatori i logički okvir



Nivo logičkog okvira	Tip informacija	Period prikupljanja	Koncept mjerenja
Svrha	Značajne razvojne promjene koje se tiču sektora Mjerenja zasnovana na trendovima, poželjno ključnih sektorskih pokazatelja N.B.: Projekt je samo jedan od mnogih utičućih faktora	Dovršetak projekta i kasnije	<b>ODRŽIVOST</b> "Nastavak tokova koristi" <b>UTICAJ</b> "Vršenje promjena u širem okruženju"
Cilj projekta	Realistične i održive promjene položaja ciljne grupe, specifične za projektnu intervenciju	Srednji rok i dovršetak projekta	<b>RELEVANTNOST</b> "Zadovoljenje potreba ciljne grupe"
 Reakcija korisnika	Rana upozorenja korisnika o vjerovatnoj uspješnosti projekta u vezi s <b>dostupnošću, korištenjem</b> i <b>zadovoljstvom</b> pruženim uslugama projekta	Kvartalno/godišnje [Prikupljeno uz monitoring rezultata.]	<b>VODEĆI INDIKATORI</b> Procjene relevantnih i održivih promjena
Rezultati	Kvantitativna i kvalitativna mjerila fizičkog napretka u u isporuci usluga Omjer troškova i input-output omjer izvršenja Poređenje aktuelnih postignuća s planiranim ciljevima	Kvartalno/Godišnje	<b>USPJEŠNOST</b> "Napredak prema ciljevima - radeći prave stvari" <b>EFIKASNOST</b> "Efikasna implementacija - radeći stvari ispravno"
Aktivnosti	Mjerenja temeljena na rasporedu aktivnosti Poređenja početka i završetka s planiranim datumima Odstupanja od planiranih termina Kontrolne tačke i događaji	Sedmično/Mjesečno	<b>EKONOMIJA</b> "Postizanje najboljih odnosa između troškova, kvaliteta i vremena"
Troškovi	Mjerenja zasnovana na budžetu Poređenja stvarnog s planiranim Analiza doprinosa vlade, donatora i korisnika Analiza varijanse troškova	Sedmično/Mjesečno	

Slika 42: Veza indikatora sa logičkim okvirom

### 10.3.3. Rezultati

*Pokazatelji načina na koji su aktivnosti transformirane u rezultat i uslugu.* Mnogi od ovih pokazatelja biće pokazatelji uspješnog dovršenja zadatka, npr., poslovnog centra podignutog i opremljenog. Drugi kvantificiraju ciljeve kao što je broj menadžera obučenih u tržišnoj analizi. Treći nivo se bavi efikasnošću projektnih aktivnosti.

Indikatori efikasnosti porede aktuelne ulaze kao omjer prema aktuelnim izlazima, npr., prosječni troškovi obuke po učesniku, procenat javnih službenika prekvalifikovanih i zaposlenih u privatnom sektoru. Većina takvih podataka može da se nađe u datotekama projekta ili prenesenih u izvještaj o napretku. Ti omjeri dopuštaju poređenja tokom vijeka projekta s nekim drugim projektima.

### 10.3.4. Cilj projekta

**Indikatori dostizanja održivih koristi za ciljnu grupu.** Ovi indikatori efektivnosti pokazuju da li je projekt postigao tražene ciljeve ili nije i da li su usluge iz projekta održive, tj. da li će se

one nastaviti pružati ciljnoj grupi po završetku vanjske pomoći. Ako stvarni učinak projekta ne odgovara planiranom, evaluator mora nastaviti istraživanje. Da li je učinak slab zbog nedovoljne početne analize problema, zbog nedostataka u njegovom dizajnu ili zbog neadekvatne implementacije? Ovdje od posebnog značaja tri faktora održivosti: 1) Da li je lokalni institucionalni i upravljački kapacitet projekta postignut? 2. Do koje mjere je razvijena neophodna politika podrške projektu? 3. Do koje mjere je postignuta finansijska održivost projekta?

Na kraju, evaluacijom treba ispitati primjenu standarda kvaliteta dobara odnosno usluga generiranih projektom postignuta prema viđenju korisnika, npr., Da li je osoblje koje je obučavano steklo potrebna znanja? Da li njihovi poslodavci smatraju ta znanja relevantnim i korisnim? Evaluacija efektivnosti i održivosti zahtijevaće od evaluatora da pribavi podatke izvan projektne organizacije, sastancima i posjetama korisnicima i drugim organizacijama.

#### 10.3.5. Svrha projekta

*Doprinos projekta širim sektorskim ciljevima.* Budući da je svaki pojedinačni projekt samo jedan elemenat programa aktivnosti, ocjena postignuća svrhe projekta može biti ostvarena kao dio tematske ili sektorske evaluacije, npr., evaluacije državnog programa“.

#### 10.4. Mogućnosti za evaluaciju

Pristup usvojen od mnogih agencija, uključujući Evropsku Komisiju, je da formalno evaluiraju izvještaje podnesene u određenim fazama projektnog ciklusa i dopune ih *ad-hoc* analizama. Tipični specifični izvještaji ove vrste su:

- **U toku izvođenja projekta (at Mid-Term)**, da se sagleda napredak i predlože promjene u toku preostalog vremena za realizaciju.
- **Po završetku projekta (at Project Completion)**, da se analiziraju utrošeni resursi, rezultati i napredak prema zacrtanim ciljevima. Cilj ove evaluacije je da se izvuku lekcije naučene tokom realizacije i upo-trijebe za unapređenje dizajna budućih projekata.

Osim navedenog, *ad-hoc* studije se koriste za istraživanje tema kao što su sektorski projekti konkretne zemlje; ili posebni tipovi intervencije u regiji kao projekt institucionalnog razvoja.

Prednost tematski studija je u tome što se jednom evaluacijom može obuhvatiti više projekata, a njeni rezultati se odnositi na šire političke ciljeve.

- Evaluacija može bi definirana kao periodična ocjena relevantnosti, efikasnosti, efektivnosti, ekonomskoj i finansijskoj vitalnosti i održivosti projekta u kontekstu zadanih ciljeva.
- Svrha evaluacije je da provjeri ostvarenja projekta u pogledu planiranih očekivanja i da se iskustva iz njega iskoriste za unapređenje budućih projekata i programa.
- U Eu se koriste slijedeći kriteriji: Relevantnost, Efikasnost, Efektivnost, Uticaj, Ekonomska i finansijska vitalnost i Održivost projekta.
- Pristup usvojen od strane mnogih agencija, uključujući Evropsku Komisiju, je programirati formalne evaluacione izvještaje u specifičnim fazama projektnog ciklusa i dopuniti ih ad-hoc analizama. Tipični specifični izvještaji ove vrste su:
  - **U toku izvođenja projekta** (at Mid-Term), da se sagleda napredak i predlože promjene u toku preostalog vremena za realizaciju.
  - **Po završetku projekta** (at Project Completion), da se analiziraju utrošeni resursi, rezultati i napredak prema zacrtanim ciljevima. Cilj ove evaluacije je da se izvuku lekcije naučene tokom realizacije i upotrijebe za unapređenje dizajna budućih projekata.

## 11. Literatura

1. <http://www.uup.ba/dokumenti.html> (17.2.2015.)
2. R.K. Wysocki and R. McGary, *Effective Project Management*, Third Edition. Indianapolis, IN: John Wiley & Sons, Inc, 2003.
3. H. Kerzner, *Project Management: A Systems Approach to Planning, Scheduling, and Controlling*, Eighth Edition. Hoboken, NJ: John Wiley & Sons, Inc, 2003.
4. R. Thomsett, "Extreme Project Management", Executive Report, Cutter Consortium, Vol.2, No. 2, 2001.
5. Murat Prašo, „Uvod u upravljanje projektom“, Univerziteteska knjiga Mostar, 2005.
6. <http://www.scrumalliance.org> (20.6.2016.)
7. Mike Cohn, *Agile Estimating and Planning*, First Printing 2005
8. [http://www.scrumalliance.org/pages/scrum\\_roles](http://www.scrumalliance.org/pages/scrum_roles) (20.6.2016.)
9. <http://www.mountangoatsoftware.com/topics/scrum> (20.6.2016.)
10. <https://heulwynroberts.wordpress.com/tag/scrum-master> (20.6.2016.)
11. (2009) Definicija i forma Team Foundation Server-a. [http://www.link-elearning.com/lekcija-Definicija-i-forma-Team-Foundation-Server-a\\_5225](http://www.link-elearning.com/lekcija-Definicija-i-forma-Team-Foundation-Server-a_5225) (29.6.2016.)
12. [https://msdn.microsoft.com/en-us/library/ms181280\(v=vs.90\).aspx](https://msdn.microsoft.com/en-us/library/ms181280(v=vs.90).aspx) (29.6.2016.)
13. [https://msdn.microsoft.com/en-us/library/ms181409\(v=vs.100\).aspx](https://msdn.microsoft.com/en-us/library/ms181409(v=vs.100).aspx) (29.6.2016.)
14. <https://www.visualstudio.com/en-us/docs/tfvc/create-work-workspaces> (29.6.2016.)
15. <https://www.visualstudio.com/en-us/docs/tfvc/merge-folders-files> (29.6.2016.)
16. <http://stackoverflow.com/questions/556981/what-is-shelving-in-tfs> (29.6.2016.)

## Glosar

Aktivnosti (Activities)	Specifični zadaci koje treba poduzeti tokom vijeka projekta sa ciljem da se ostvare rezultati.
Analiza ciljeva (Analysis of Objectives)	Identifikacija i provjera željenih budućih koristi kojima ciljna grupa daje prioritet. Proizvod analize ciljeva je drvo ciljeva.
Redoslijed aktivnosti (Activity schedule)	Gantogram, grafička prezentacija - slična štapićastom dijagramu - na kojoj se predstavljaju vrijeme, redoslijed i trajanje projektnih aktivnosti. Može se koristiti za utvrđivanje kontrolnih tačaka za praćenje napretka i dodjelu odgovornosti za dostizanje kritičnih tačaka u projektu.
Procjena (Appraisal)	Analiza prijedloga projekta da se prepoznaju njegove osobine i prihvatljivost u skladu sa definisanim kriterijima. To je posljednji korak prije donošenja odluke o finansiranju projekta. Procjenjuje se da li projekt prihvatljiv sa stanovišta njegove utemeljenosti, da li ciljevi odgovaraju problemu i da li su troškovi razumni.
Pretpostavke (Assumptions)	Vidjeti „Rizici i pretpostavke“
Komisija (Commission)	Komisija Evropske Unije
Odobrenje (Commitment)	Odobrenje je formalna odluka Komisije da odvoji određenu količinu novca za posebnu namjenu. Nikakvi troškovi ne mogu se priznati ako nisu odobreni od strane ovlaštenog organa.
Evaluacija (Evaluation)	Periodična ocjena efikasnosti, efektivnosti, uticaja, održivosti i relevantnosti projekta u odnosu na odabrane ciljeve.
Faza evaluacije (Evaluation Phase)	Šesta i konačna faza projektnog ciklusa u kojoj se projekt analizira u kontekstu ciljeva i iskustava koja treba ugraditi u buduće akcije.
Faktori održivosti (Factors Ensuring Sustainability)	Faktori poznati po znatnom uticaju na održivost koristi generiranih projektima u prošlosti a treba da budu uzeti u obzir pri oblikovanju budućih projekata.
Analiza izvodljivosti (Feasibility Study)	Analiza izvodljivosti tokom faze formulacije projekta provjerava da li je predloženi projekt dobro utemeljen i ima li izgleda da zadovolji potrebe budućih korisnika. Analizom treba oblikovati projekt do operativnih detalja uzimajući u obzir tehničke, ekonomske, finansijske, institucionalne, upravljačke, okolinske i socio-kulturne aspekte. Analiza treba da Evropskoj komisiji i partnerskoj vladi pruži dovoljno informacija za prihvatanje, modifikaciju ili odbacivanje projektnog prijedloga za finansiranje.
Finansijski sporazum (Financial Agreement)	Dokument potpisan od strane Evropske Komisije i partnerske zemlje odnosno slijednika o finansijskoj odluci. Obuhvata opis projekta ili programa za finansiranje. Predstavlja formalno odobrenje EU i partnerske zemlje da finansiraju opisane mjere.
Finansijski prijedlog (Financing Proposal)	Finansijski prijedlozi su nacrti dokumenata podnesenih od strane službi Komisije – Komitetu za finansiranje na mišljenje i Komisiji na odlučivanje. Opisuje opštu pozadinu, prirodu, obuhvat i ciljeve, te modalitete predloženih mjera i indicira predviđena ulaganja. Nakon prijema povoljnog mišljenja Finansijskog Komiteta, prijedlozi se

	razmatraju u Komisiji i donosi odluka o njima i sklapa te potpisuje finansijski dogovor s partnerskom zemljom.
Faza finansiranja (Financing Phase)	Četvrta faza projektnog ciklusa, tokom koje se projekti odobravaju za finansiranje i odabire ugovarač za implementaciju.
Faza formulacije (Formulation Phase)	Treća faza projektnog ciklusa tokom koje se razrađuju detalji na temelju analize izvodljivosti.
Gantogram (Gantt Chart)	Grafički metod prikazivanja informacija, često korišten radi terminiranja aktivnosti. Sličan štapićastom dijagramu.
Hijerarhija ciljeva (Hierarchy of Objectives)	Aktivnosti, rezultati, projektni cilj i svrha - onakvi kako su navedeni u logici intervencije.
Faza identifikacije (Identification Phase)	Druga faza projektnog ciklusa. Podrazumijeva elaboraciju projektne ideje sa stanovišta ciljeva, rezultata i aktivnosti – sa ciljem da se dođe do odgovora da li nastaviti rad - izradom analize izvodljivosti.
Faza formulacije (Formulation Phase)	Treća faza projektnog ciklusa. Podrazumijeva utvrđivanje detalja projekta na osnovu analize izvodljivosti, praćene provjerom od strane osoblja Evropske komisije da li osobine i konzistentnost projekta odgovaraju sektorskim politikama.
Faza izvođenja (Implementation Phase)	Peta faza projektnog ciklusa, tokom koje se projekt realizuje i nadzire u pogledu napretka prema zacrtanim ciljevima.
Indikativni programi (Indicative Programmes)	Njih priprema Evropska Komisija u saradnji s vladama partnerskih zemalja. Njima se daju opšta uputstva i principi saradnje s Evropskom Unijom. Specificiraju fokalne sektore i teme u zemlji ili regiji i mogu sadržavati brojne projektne ideje.
Integralni pristup (Integrative Approach)	Konzistentna provjera svih faza projektnog ciklusa sa ciljem u njegovom fokusu ostanu relevantnost, izvodljivost i održivost.
Logika intervencije (Intervention Logic)	Strategija sadržana u projektu. To je narativni opis projekta na svih četiri nivoa hijerarhije ciljeva korištenih u logičkom okviru.
Logički okvir (Logframe)	Matrica u kojoj su predstavljene logika intervencije, pretpostavke, objektivno provjerljivi pozatelji i izvori njihove provjere.
Pristup pomoću logičkog okvira (Logical Framework Approach, LFA)	Metodologija planiranja, upravljanja i evaluacije programa i projekata, uključujući analizu ciljeva i strategije, pripremu matrice i rasporeda aktivnosti i resursa.
Sredstva (Means)	Inputi zahtijevani za izvršenje rada (osoblje, oprema, materijali)
Kontrolne tačke (Milestones)	Vrsta objektivno provjerljivog indikatora kratkoročnih ciljeva (obično aktivnosti) koji olakšavaju mjerenje postignuća u toku projekta. Indiciraju i vrijeme kad se mogu donositi odluke.
Minotoring, nadzor ili praćenje izvršenja (Monitoring)	Sistematsko i neprekidno prikupljanje, analiza i upotreba informacija sa ciljem upravljačke kontrole i donošenja odluka.
Cilj (Objective)	Opis cilja projekta ili programa. U generičkom značenju to se odnosi na aktivnosti, rezultate, ciljeve i svrhu projekta.
Drvo ciljeva (Objective Tree)	Grafički prikaz projektnih intervencija, planiranih logički, polazeći od analize problema, predloženih sredstava, resursa i dovršetka.

Objektivno provjerljivi indikatori (Objectively Verified Indicators, OVI)	Mjerljivi indikatori koji pokazuju da li su ili ne dostignuti ciljevi na svakom nivou hijerarhije logičkog okvira. Čine osnovicu za oblikovanje odgovarajućeg sistema upravljanja projektom.
Svrha (Overall Objectives)	Ciljevi u širem sektorskom ili državnom programu kojem projekt treba da doprinese.
Preduslovi (Pre-conditions)	Preduslovi (ako postoje) vezani za odredbe o pomoći koje moraju biti ispunjene prije početka realizacije projekta.
Predinvesticijska studija (Pre-feasibility Study)	Prethodno istraživanje vođeno tokom faze identifikacije, kojim treba da se osigura da su svi problemi identifikovani, alternative prepoznate i najpovoljnija odabrana na temelju kriterija održivosti. Analiza treba da Evropskoj Komisiji i partnerskim vladama pruži informacije koje opravdavaju prihvatanje, modifikaciju ili odbacivanje predloženog projekta za fazu formulacije.
Analiza problema (Problem Analysis)	Strukturirano ispitivanje negativnih aspekata neke situacije sa ciljem uspostavljanja uzročno-posljedičnog odnosa.
Faza programiranja (Programming Phase)	Prva faza projektnog ciklusa tokom koje se priprema indikativni program.
Projektni ciklus (Project Cycle)	Projektni ciklus obuhvata život projekta počev od inicijalne ideje do njegovog dovršetka. Njime se stvara struktura koja osigurava su svi partneri projekta konsultovani, definišu ključne odluke, potrebne informacije i odgovornosti u svakoj fazi tako da se mogu donositi meritorne odluke u ključnim trenucima. Vodi ka evaluaciji u kojoj se praktična iskustva unose u buduće projekte.
Upravljanje projektnim ciklusom (Project Cycle Management)	Metodologija pripreme, implementacije i evaluacije projekata i programa temeljenih na integralnom pristupu i pristupu pomoću logičkog okvira
Cilj projekta (Project Purpose)	Glavni cilj projekta u terminima održive koristi za ciljnu grupu projekata. Ne odnosi se na usluge projekta (to su rezultati) niti na korištenje tih usluga, nego na koristi koje ciljna grupa izvlači iz rezultata korištenih usluga projekta.
Tekući troškovi, troškovi koji se ponavljaju (Recurrent Costs)	Troškovi koji nastaju radom i održavanjem nakon puštanja projekta u upotrebu.
Raspored resursa (Resource Schedule)	Budžet projekta
Rezultati (Results)	Izlaz stvoren poduzimanjem serije aktivnosti. Rezultati su ono što projekt treba da postigne s datumom dovršenja.
Rizici, ograničenja i pretpostavke (Risks, Constrains & Assumptions)	Vanjski faktori koji mogu uticati na napredak ili uspjeh projekta, na kojima upravljački tim nema direktnu kontrolu.



Izvori provjere (Sources of Verification)	Sredstva pomoću kojih indikatori ili kontrolne tačke mogu biti snimljeni i biti na raspolaganju projektom timu ili evaluatorima izvršenja projekta.
Partneri projekta (Stakeholders)	Pojedinci ili institucije s finansijskim ili intelektualnim interesom za rezultate projekta.
Analiza strategije (Strategy Analysis)	Kritička ocjena alternativnih puteva dostizanja ciljeva i izbor jednog ili više njih za uključivanje u projekt.
Održivost (Sustainability)	Ključni zahtjev uspješnosti projekta. To je sposobnost generiranja rezultata nakon što je prekinuta vanjska podrška. Mada je projekt vremenski ograničen, koristi iz njega treba da se nastave u eksploatacionom vijeku projekta bez potrebe za vanjskim inputima.
SWOT analysis	Analiza jakih ( <b>S</b> trengths) i slabih strana ( <b>W</b> eaknesses) te mogućnosti ( <b>O</b> pportunities) i prijetnji ( <b>T</b> hreats) s kojima se susreće organizacija. Alat koji se koristi pri procjeni projekta.
Tehnički uslovi (Terms of Reference)	Tehničkimuslovima definišu se zadaci koje treba da izvrši ugovarač, a opisuju pozadinu projekta i ciljeve, planirane aktivnosti, očekivane ulaze i izlaze, budžet, vremenski raspored i opis poslova.
Plan rada (Workplan)	Terminski plan koji prikazuje aktivnosti i resurse potrebne za dostizanje projektog rezultata i ciljeva.