



ASOCIJACIJA ZA
KVALITET I
STANDARDIZACIJU
SRBIJE

FESTIVAL KVALITETA 2005.
32. Nacionalna konferencija o kvalitetu
Kragujevac, 19. - 21. maj 2005.



COST BENEFIT ANALIZA I KVALITET SOFTVERA*

COST BENEFIT ANALYSIS AND SOFTWARE QUALITY*

mr Miladin Stefanović¹⁾, mr Slobodan Mitrović²⁾, mr Milan Erić³⁾

Rezime: Da bi se dobio kvalitetan softverski proizvod potrebno je na pravi način upravljati projekatima (re)inženjeringu softvera. Postoji veliki broj analiza koje je potrebno sprovesti da bi se uskandalili ekonomski okviri za zahtevanim nivoom kvaliteta softvera. Pri tome je potrebno definisati i kvantifikovati veliki broj parametara koji opisuju jedan softverski proizvod. U radu će biti prezentovana cost/benefit analiza sa procenom cene koštanja softvera, koja će biti implementirana na karakterističnim preduzećima uz donošenje generalnih zaključaka.

Ključne reči: kvalitet softvera, cena softvera, cost-benefit, COCOMO

Abstract: Management of software project has a great importance in process of software development. There large number of analysis that should be performed in order to have successful software (re)engineering project. One of the very important activities is compare and contrast economical and financial framework with software quality characteristics. During this process, a large number of parameters should be defined and measures. In this paper it will be presents const/benefit analysis, calculation of software costs, and these methods will be implemented on specific domestic companies. Some general conclusions will be made.

Key words: software quality, software costs, cost-benefit, COCOMO

1. UVOD

Glavno pitanje kvaliteta softvera predstavlja obezbeđivanje alata i tehnologija koji će omogućiti softverskoj industriji da razvija softverske prouizvode i službe koje su sigurne, pouzdane, a sa druge strane zadovoljavaju ekonomske i finansijske okvire koji su definisani u projektima razvoja ili reinženjeringu softvera [1]. Rešavalje ovih pitanja može u značajnoj meri poboljšati prilike u ICT industriji, kao i stanju IT sektora u

1) Mr Miladin Stefanović, CIM centar, Mašinski fakultet Kragujevac

2) Mr Slobodan Mitrović, Mašinski fakultet Kragujevac

3) Mr Milan Erić, Mašinski fakultet Kragujevac

* Rezultati deo projekta TR-6218A - "Razvoj softverskih rešenja u Internet-intranet okruženju za integrisani razvoj proizvoda i proces"

domaćoj privredi [2]. Kvalitet softvera je više dimenzionalni koncept koji se ne može jednostavno definisati [3]. Pri određivanju kvaliteta softvera potrebno je meriti više parametara. Ne postoji opšta saglasnost koje karakteristike softvera je potrebno meriti, niti kako ih kvantifikovati. Da bi se ostavrio potrebnii kvalitet softvera, potrebno je rešiti osnovnu grupu problema koja se javlja, i to: izbor potrebnih alata, tehnologija i cost-benefit analiza. Naročito je bitna su bitne analize koje prethode razvoju ili reinženjeringu softverskog proizvoda i to: analiza mogućih prilaza u impleemntaciji kvaliteta, analiza uvođenja promena, implementacija u koracima, odnos ostvarwnog kvaliteta i cene koštanja, analiza cene koštanja i ostzavrenih performansi i cost-benefit analiza. Jedan od izuzetno bitnih faktora u *Cost/Benefit* analizi je određivanje cene koštanja razvoja novog softvera ili reinženjeringu starog [4]. Sem toga, na troškovnoj strani moraju se uzeti u obzir troškovi nabavke novog softvera i eventualni troškovi koje bi nastali kao posledica procesa reinženjeringu informacionog sistema u preduzeću. Pri ovome je potrebno definisati veliki broj uticajnih parametara koji definišu složenost i osobine projekta softverskog (re)inženjeringu. U ovom radu će biti prikazana COCOMO metoda cost/benefit analize softvera. Metoda će biti implementirana na tri domaća preduzeća, prezentovaće se definisanje uticajnih faktora. Na osnovu rezultata analize biće izведен generalni zaključak o karakteristima procesa softverskog reinženjeringu u domaćim preduzećima, sa aspekta kvalitetnog projektovanja softvera.

2. COCOMO MODEL

COCOMO (*CO*nstructive *CO*st *M*odel) predstavlja model za procenu cene koštanja softvera koji je razvijen kao otvoreni model od strane Direktora Centra za Softverski inženjering na USC Dr Barry Boehm-a. Polazna osnova u izračunavanju u COCOMO modelu je korišćenje *Effort Equation* (jednačine ulaganja) koja ima za cilj procenjivanje veličine podatka osoba/mesec koja je neophodna za razvoje projekta. Svi ostali COCOMO rezultati, uključujući i procene sa zahtevima i održavanjem se izvode iz ove veličine. COCOMO proračuni se baziraju na procenjenoj veličini projekta koja se meri brojem linija izvornog koda *Source Lines of Code* (SLOC). SLOC se definiše kao: jedino linije koda koje se isporučuju kao deo softverskog proizvoda se broje - drajveri za testiranje i drugi softver za podršku se ne računaju; linije koda koji su generisali zaposleni programeri - kod kreiran od strane generatora koda se isključuje; jedan SLOC je jedna logička linija koda; deklaracije se računaju kao SLOC; komentari se ne računaju kao SLOC.

COCOMO II model sadrži i *Scale Drivers* - uticajne faktore koji najznačajnije utiču na vreme i trajanje projekta. Potrebno je proceniti pet uticajnih faktora koji su elementi jednačine ulaganja.

Uticajni faktori su: specifičnost, fleksibilnost razvoja, arhitektura / razrešavanje rizika, kohezija tima i zrelost procesa.

COCOMO II, takođe sadrži i 17 uticajnih faktora na cenu. Cenovni uticajni faktori određuju trud koji je neophodan za kompletiranje softverskog projekta. Na primer, ukoliko se razvija softver za kontrolu leta avio kompanije, uticajni faktor Zahtevana Pouzadnost Softvera - *Required Software Reliability* (RELY) i njegova cena biće postavljeni na vrlo visoku vrednost.

Jednačina ulaganja u COCOMO II modelu procenjuje potreban broj osoba/meseci *person/months* - PM na osnovu procene veličine projekta merenog u hiljadama SLOC ili KASLOC.

$$\text{Effort} = 2.94 * \text{EAF} * (\text{KASLOC})^E$$

Gde je:

- KASLOC - broj adaptiranih linija koda
- EAF - faktor podešavanja ulaganja (*Effort Adjustment Factor*) koji pripada grupi cenovnih faktora (*Cost Drivers*)
- E - eksponent dobijen iz pet veličinskih uticajnih faktora (*Scale Drivers*)

Faktori podešavanja ulaganja su proizvod svih uticajnih faktora.

COCOMO II jednačina plana predviđa broj meseci neophodnih za završetak softverskog projekta. Vreme trajanja projekta se bazira na *Effort* jednačini:

$$\text{Duration} = 3.67 * (\text{Effort})^{SE}$$

Gde je:

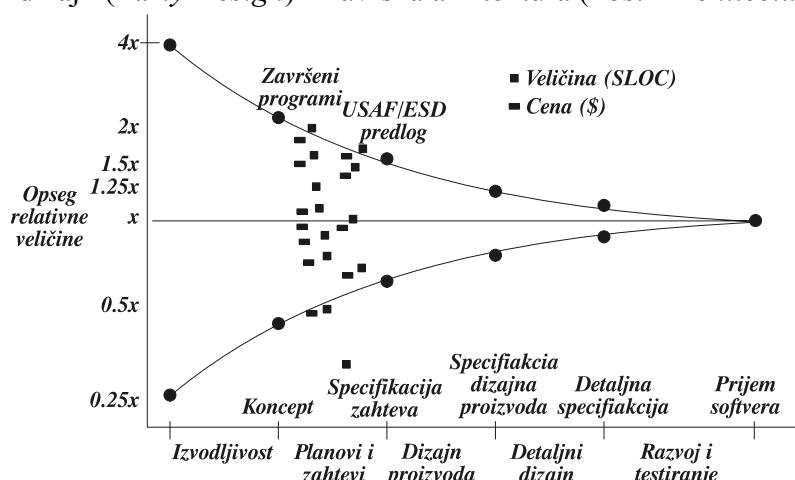
- Effort - vrednost dobijena iz *Effort* jednačine
- SE - eksponent za jednačinu plana dobijen iz pet veličinskih uticajnih faktora (*Scale Drivers*)

Jedan od najbitnijih faktora je SCED ili *Required Development Schedule* koji ukazuje na potencijalnu mogućnost da projekat zahteva više rada za razvoj nego što je to predviđeno kao optimum.

Pri korišćenje USC-COCOMO II softvera polazimo od nekih osnovnih prepostavki, a jedna od osnovnih je procenjivanje veličine projekta.

Sa slike 1 se vidi kolika može biti nepreciznost procenjivanja veličine i cene softvera u zavisnosti od faze projekta. Kako napreduje životni ciklus proizvoda, i kako se donose odluke bitne za razvoj proizvoda, priroda proizvoda i naravno veličina mogu bolje da se sagledaju. "Završeni programi" predstavljaju sedam stvarnih projekata razvijenih od strane tima koji je predvodio autor metode, dok "USAF/ESD predlog" se odnose na pet predloga *USA Air Force Electronic System Division* koji su dati u cilju ocenjivanja metode.

Pošto cena softvera može da se ocenjuje u različitim fazama razvoja projekta uvedeni su moduli Rani dizajn (*Early Design*) i Završna arhitektura (*Post-Architecture*)



Slika 1 - Procena veličine i cene softvera u zavisnosti od faze

U Fazi *Early Design* modela uključuju se pitanja korišćenja alternativnih softvera/arhitekture sistema i koncepta. U ovom stadijumu, obično, nema dovoljno poznatih informacija za fino kalibriranje modela. U ovoj fazi COCOMO II softvera

koriste se 7 cenovnih uticajnih faktora (i dva Sposobnost zaposlenih - *Personnel Capability* i Iskustvo zaposlenih - *Personnel Experience*, dok se u drugoj fazi *Post Architecture* modela koriste 6 faktora i različiti aspekti osposobljenosti, iskustva i kvaliteta zaposlenih.)

Post-Architecture model uključuje stvarni razvoj i održavanje softvera. Ova faza daje najbolje rezultate ukoliko razvija arhitekturu softvera, koja odgovara misiji sistema, konceptu funkcionisanja i ukoliko uspostavlja okvir za kompletan prozvod.

3. CASE STUDY – SOFTVERSKI REINŽENJERING U DOMAĆIM KOMPANIAJAM

Za sprovođenje *Cost/Benefit* analize korišćen je COCOMO II softverski paket i to u delu određivanja troškova razvoja, reinženjeringa softvera i definisanja potrebnih ljudskih i finansijskih resursa.

Za konkretni primer, reinženjering IS, na primeru preduzeća A, urađena je COCOMO II *Cost Benefit* Analiza i definisani parametri projekta,

Tabela 1. – Karakteristicki izabranih preduzeća

Organizacija A	Proizvodni program preduzeća je širokog spektra, sa tehnološkim usmerenjem na preradu metala rezanjem, prostornim i zapreminskim oblikovanjem i izradu alata za postupke kontrole. Program obuhvata pretežno specijalne proizvode i to uglavnom po porudžbini (za poznatog kupca). U preduzeću je zaposleno 113 radnika i može se svrstati u preduzeća srednje veličine. Postojeći aplikativni softver obuhvata sledeće module: marketing, kadrovska evidencija i obračun zarada, knjigovodstvo.
Organizacija B	Akcionarsko društvo za trgovinu na veliko i malo sa preko 90 prodajnih objekata. Preduzeće je srednje veličine i bavi se prodajom na veliko i malo. Postojeći aplikativni softver obuhvata: magacinsko poslovanje, nabavka i snabdevanje, prodaja.
Organizacija C	Program preduzeća je širokog spektra iz oblasti prerade metala. Program je orijentisan na male, i srednje serije metalnih proizvoda, koji se izrađuju i po narudžbini i za potrebe tržišta. U preduzeću je zaposleno 24 radnika i može se svrstati u mala preduzeća. Postojeći softver obuhvata: magacinsko poslovanje, nabavka i snabdevanje, prodaja.

Na početku su usvojene vrednosti za uticajne parametre. Potrebno je odrediti parametre kohezije i sposobnosti programerskog tima. Ovaj parametar se sastoji od tri pod parametara: sposobnosti analitičara - ACAP (*Analyst Capabilities*), sposobnosti programera - PCAP (*Programmer Capability*) i vremena koje je tim proveo na okupu PCON (*Personnel Continuity*). Svaki ovih faktora može biti ocenjen od 1 (vrlo nizak) do 5 (vrlo visok) tabela 1. Zatim je sva ova tri parametra sumiraju. Koristeći tabele koje sadrži programski paket biraju se vrednosti uticajnog parametra.

Tabela 1 - Parametri kohezije i sposobnosti programerskog tima

	Ekstremno nizak	Vrlo nizak	Nizak	Normalan	Visok	Vrlo visok	Ekstremno visok
Suma ACAP, PCAP, PCON	3, 4	5, 6	7, 8	9	10, 11	12, 13	14, 15

Za konkretni projekat dobijena je vrednost sume 7, što spada u kategoriju niskih vrednosti, prevashodno zato što je razvojni tim preduzeća bio kratko vremena na okupu, dok su im programeri bili sa malo iskustva.

Programersko iskustvo u radu na ciljnoj platformi ponderisano je na sledeći način koristeći pondere: iskustvo rada sa aplikacijama AEXP (*Application Experience*), iskustvo u radu sa platformom PEXP (*Platform Experience*) i iskustvo u radu sa jezicima i alatima LTEX (*Language and Tool Experience*) (tabela 2).

Tabela 2 - Programersko iskustvo u radu na ciljnoj platformi

	Ekstremno nizak	Vrlo nizak	Nizak	Normalan	Visok	Vrlo visok	Ekstremno visok
Suma AEXP, PEXP, LTEX	3, 4	5, 6	7, 8	9	10, 11	12, 13	14, 15
Programersko iskustvo	£ 3 meseca	5 meseci	9 meseci	1 godina	2 godine	4 godine	6 godine

Tabela 3 - Specifičnost i fleksibilnost zahteva

	Vrlo nizak	Nizak	Normalan	Visko	Vrlo visok	Ekstremno visok
PREC specifičnosti	Totalno specifičan	Uglavnom specifičan	Ponegde specifičan	Uglavnom poznat	U većem delu specifičan	Totalno poznat
FLEX Fleksibilnost zahteva	rigorozni	Ponegde fleksibilni	Delimično fleksibilni	Uglavnom usaglašeni	Ponegde usaglašeni	Generalni ciljevi
TEAM kohezija tima	Vrlo komplikovani odnosi	Ponegde komplikovani odnosi	U osnovi kooperativni odnosi	Većinom kooperativni odnosi	Visoko kooperativni odnosi	Kompletna interaktiv.

Zatamljena polja predstavljaju procenu faktora u slučaju analiziranog preduzeća. Postavljeni zahtevi su bili delimično fleksibilni, projekat je uglavnom bio poznat, a u timu su vladali uglavnom kooperativni odnosi (tabela 3). Preko definisanih opcija iz prethodnih tabela napravljena je selekcija uticajnih parametara iz tabela koje su date u okviru paketa COCOMO II.

Definisani su i parametri koji se odnose na zahtevanu pouzdanost softvera RELY (*Required Software Reliability*), veličinu baze podataka DATA (*Database Size*), kompleksnost proizvoda CPLX (*Product Complexity*), dokumentacije koja odgovara potrebama životnog ciklusa DOCU (*Documentation match to life-cycle needs*). RELY i DOCU imaju opseg od niskog do vrlo visokog, CPLX od vrlo niskog do ekstremno visokog. Ukupne sume ovih faktora čine faktor ukupne kompleksnosti i pouzdanosti RCPX koji može da se kreće od 5 do 21 (tabela 4)

Tabela 4 - Parametri RELY, DATA, CPLX, DOCU

	Ekstremno nizak	Vrlo nizak	Nizak	Normalan	Visko	Vrlo visok	Ekstremno visok
RELY, DATA, CPLX, DOCU	5, 6	7, 8	9 - 11	12	13 - 15	16 - 18	19 - 21
Naglasak na pouzdanosti i dokumentaciji	Vrlo mali	Mali	Delimič.	Osnovni	Jak	Vrlo jak	Ekstrem.
Kompleksnost projekta	Vrlo jednost.	Jednost.	Umereno jednost.	Osrednje komplek.	Kompl.	Vrlo kompl.	Ekstrem. komplek.
Veličina baze podataka	Mala	Mala	Mala	Srednja	Velika	Vrlo velika	Vrlo velika

Pošto je reč o preduzećima u metalopreradivačkoj industriji vrednost parametra RELY je umerena pošto eventualni otkaz softvera ne bi bio skopčan sa visokim finansijskim gubitcima (kao što bi to bila situacija da je reč o nekom bankarskom sistemu).

Pri čemu se RELY sračunava prema tabeli 5.

Parametar DATA se određuje prema forumuli 1, a rang se određuje preko tabele 6.

$$\frac{D}{P} = \frac{\text{Velicina baze podataka (Bytes)}}{\text{Velicina programa (SLOC)}} \quad (1)$$

Tabela 5 - Zahtevana pouzdanost softvera

	Vrlo nizak	Nizak	Normalan	Visok	Vrlo visok	Ekstremno visok
RELY	Beznačajni gubitci	Nizak, lako se pokrivaju gubitci	Umeren, lako se pokrivaju gubitci	Visoki finansijski gubitci	Rizik za ljudski život	

Tabela 6 - Parametar kompleksnosti DB

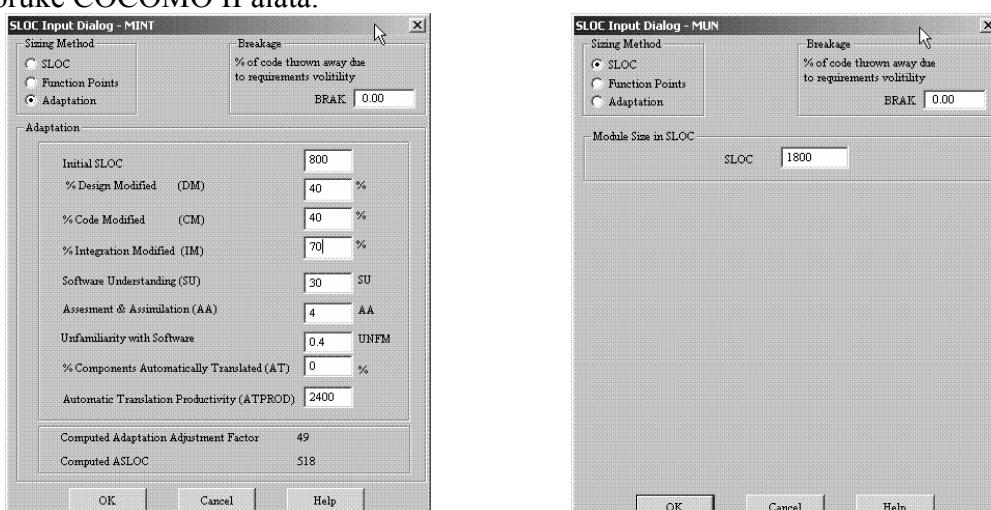
	Vrlo nizak	Nizak	Normalan	Visok	Vrlo visok	Ekstremno visok
DATA		DB bytes/ Pgm SLOC < 10	10 - D/P < 100	100 - D/P < 1000	D/P - 1000	

Parametar DOCU se određuje prema tabeli 7, a pošto u analiziranom preduzeću dokumentacije ima vrlo malo selektovana je najniža vrednost.

Tabela 7 - Vrednost parametra - dokumentovanost softvera

	Vrlo nizak	Nizak	Normalan	Visok	Vrlo visok	Ekstremno visok
DOCU	Mnoge potrebe životnog ciklusa nisu pokrivene	Neke potrebe životnog ciklusa nepokrivene	Glavne potrebe životnog ciklusa pokrivene	Pokrivene sve faze životnog ciklusa	Deteljno pokrivene faze	

Pored ovih postoje i drugi pomoći parametri koji su takođe definisani koristeći preporuke COCOMO II alata.



a) Izračunavanje SLOC u slučaju kada se vrši modifikacija nekog koda

b) Izračunavanje SLOC u slučaju da se vrši pisanje koda

Slika 2 - Izračunavanje SLOC

Veličina projekta definisana je preko broja linija koda, s tim što je izabrana opcija reinženjeringu postojećeg koda u obimu od 40% i podešeni parametri koji se tiču razumevanja i poznavanja softvera i razvojnog okruženja na osnovu podataka iz prethodnih tabela (slika 2).

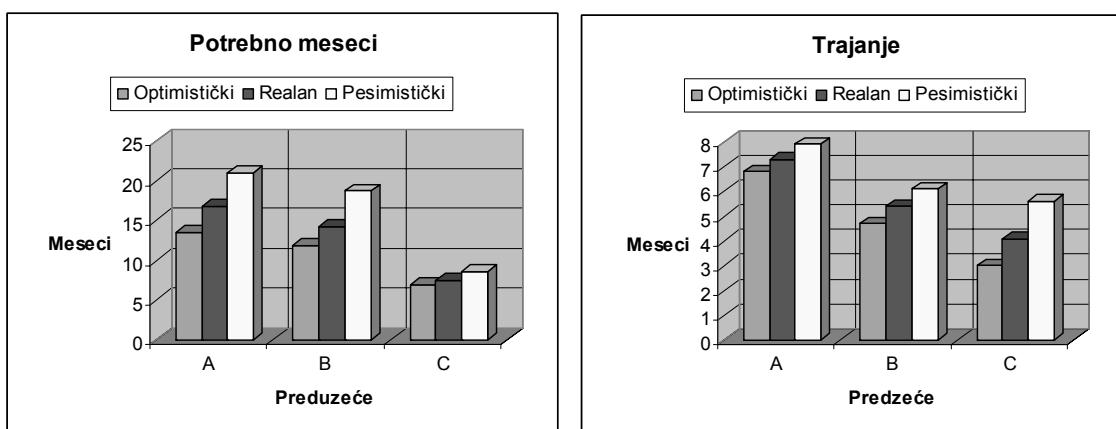
Definisani su osnovni parametri za svaki modul koji je potrebno kreirati ili redizajnirati. Pored naziva modula unosi se i cena radne snage, kao i parametri koji pokazuju obučenost radne snage, koherentnosti timova, slika 3. Za potrebe ove analize dati su parametri obučenosti i kohezije radnih timova ispod prosečnih vrednosti. Na osnovu ovih parametra izračunavaju se ukupni troškovi izrade, potrebna radna snaga i vreme za izradu, kao i nivo rizika.

Project Name: wSORIS		Scale Factor			Schedule					
		Development Model:			Post Architecture					
X	Module Name	Module Size	LABOR Rate (\$/month)	NOM Effort DEV	EST Effort DEV	PROD	COST	INST COST	Staff	RISK
MINT	A: 518	700.00	1.14	1.8	2.0	257.8	1406.75	2.7	0.3	0.0
MUN	S: 1800	700.00	1.14	6.1	7.0	257.8	4888.33	2.7	1.0	0.0
e-servis	S: 750	700.00	1.14	2.6	2.9	257.8	2036.80	2.7	0.4	0.0
MUF	A: 1270	700.00	1.14	4.3	4.9	257.8	3448.99	2.7	0.7	0.0

Total Lines of Code:	Estimated	Effort	Sched	PROD	COST	INST	Staff	RISK
4338	Optimistic	13.5	6.8	322.2	9424.70	2.2	2.0	
	Most Likely	16.8	7.3	257.8	11780.87	2.7	2.3	0.0
	Pessimistic	21.0	7.9	206.2	14726.09	3.4	2.7	

Slika 3 - Prikaz osnovnih parametara modula, kao i estimacija scenarija

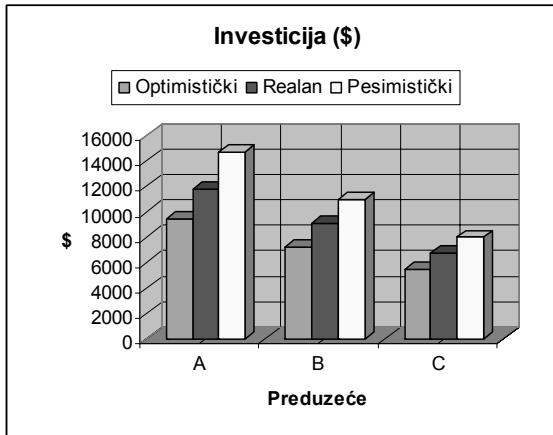
Zbog niske cene radne snage rizik je prikazan na niskom nivou, mada je realno da je taj rizik nešto veći.



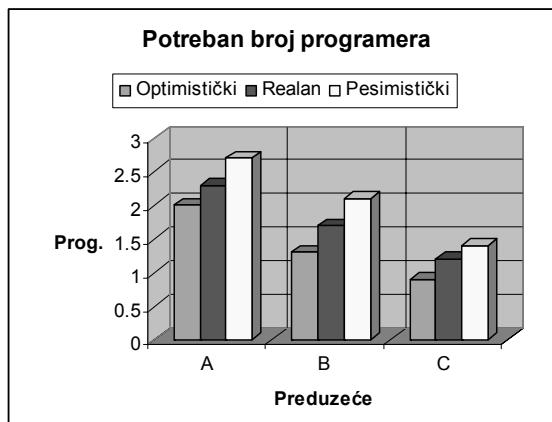
Slika 4 - Potrebno meseci u pesimističkom, realnom i optimističkom scenariju za preduzeća A, B, C

Slika 5 - Ukupno trajanje projekta u pesimističkom, realnom i optimističkom scenariju za preduzeća A, B, C

Na kraju su date procene za optimistički, realni i pesimistički scenario. Kao i ukupan broj čovek/meseca, broj zaposlenih, broj radnih meseci i ukupna cena koštanja projekata slika 3. Svi ovi podaci grafički su prezentovani na dijagramima 4 - 7. Za projekt u preduzeću A u realnom scenariju potrebno je 16.8 programer/meseci, projekat bi trajao 7.3 meseca, koštalo 11780\$ i zahtevao angažovanje 3 programera.



Slika 6 - Potrebne investicije u pesimističkom, realnom i optimističkom scenariju za preduzeća A, B, C

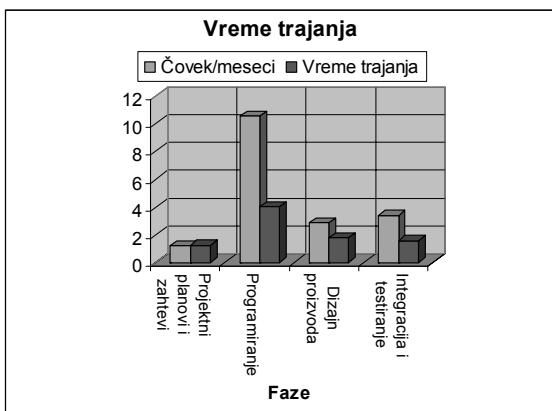


Slika 7- Potreban broj programera u pesimističkom, realnom i optimističkom scenariju za preduzeća A, B, C

Paket COCOMO II omogućava pregled parametara po fazama ukupnog projekta ili modula. Na slici su prikazani pomenuti parametri za karakteristične faze projekta. Grafička prezentacija podataka je data na slikama 8 i 9.



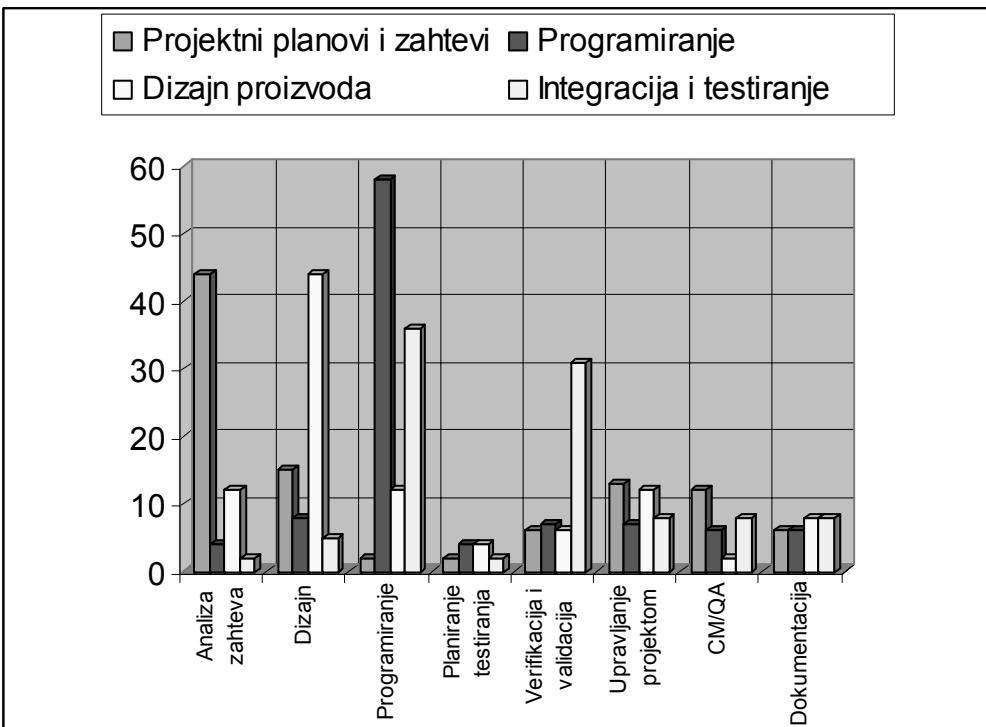
Slika 8 - Ukupna raspodela vremena po projektnim fazama



Slika 9 - Vreme trajanja pojedinih faza po mesecima

Na slike 9 se vidi da planiranje i definisanje zahteva za izradu projekta zahteva 7% vremena ukupnog trajanja projekta, dizajn projekta zauzima 15%, dok programiranje u širem smislu zahteva skoro 2/3 vremena ili 62%, i na kraju aktivnosti integracije i testiranja zahtevaju 16% raspoloživog vremena. Za samu aktivnost programiranja potrebno je 12 programera/meseci. Ova analiza je napravljena za podatke koji se odnose na preduzeće A, ali je raspodela vremena i troškova gotovo identična i za preduzeća B i C.

Glavne faze projekta su: pravljenje projektnih planova i zahteva, programiranje, dizajn proizvoda, analizirani su sa aspekta trajanja pojedinih aktivnosti. Analizirane aktivnosti su: analiza zahteva, dizajn, programiranje, planiranje testiranja, verifikacija i validacija, upravljanje projektom, obezbeđivanje kvaliteta, kao i potrebno vreme za kreiranje dokumentacije.



Slika 10 - Ukupna raspodela vremena pojedinih aktivnosti po fazama

Sva vremena su predstavljena u procentualnim iznosima i predstavljena na slici 10.

ZAKLJUČAK

COCOMO II softverski paket koji je korišćen za kalkulaciju daje i konačan stepen rizika realizacije projekta. Projekat je analiziran u opciji *Early Design* što za posledicu može da ima odstupanje dobijenih rezultata od stvarnih vrednosti. (videti sliku 1). Procenjeni rizik za ostvarivanje ovog projekta je nizak. Pošto su plate zaposlenih na razvoju i održavanju softvera u SCG niže nego prosečne plate u razvijenim zapadnim zemljama i USA (a pošto je softver kreiran u USA) izvršena je analiza u kojoj su postavljene prosečne plate radnika u USA i rezultat je takođe ukazivao na nizak rizik ostvarivanja projekta.

Takođe je izvršena analiza i za preduzeća B i C u modu *Early Design* i rezultati su, takođe, ukazivali na nizak rizik koji sa sobom nosi projekat softverskog reinženjeringu u domaćim kompanijama.

Osnovi razlozi koji čine ovaj i ovakve projekte reinženjeringu projektima niskog rizika su:

- nizak nivo integracije resursa i nizak nivo primene složenih IT koncepcata kod nasleđenih sistema tako da oni nisu kompleksni za razumevanje i reinženjering;
- izbor delimičnog reinženjeringu, pošto se vrši integracija nasleđenog sistema sa novim sistemom smanjeni su rizici;
- relativno niži nivo ciljnog sistema posle reinženjeringu, pošto preduzeća u metaloprerađivačkoj industriji obično nisu nosioci inovacionih promena u oblasti IT; to uslovjava da ciljni sistemi nisu preterano kompleksni, da se pred njih ne psotavljaju visoki zahtevi na polju pouzdanosti i sigurnosti a, baze podataka u ovim sistemima obično nisu velike;

- niska cena rada kvalifikovane radne snage zaposlene na razvoju i održavanju softvera.

Faktori koji negativno utiču i podižu rizik su:

- nešto niži stepen sposobljenost i kohezije radnih timova (posledica odlaska velikog broj stručnjaka) što ima kao rezultat niži nivo znanja o softverskim alatima i platformama i
- nepostojanje dokumentacije koja podržava životni ciklus softvera.

LITERATURA

- [1] Osterweil L., at all, "Strategic Direction in Software Quality", ACM Computer Surveys. Vol. 8, Dec 24. 1996.
- [2] Stefanović M, Arsovski S., " Uporedna analiza postojećeg stanja IS u domaćoj i inostranoj metaloprerađivačkoj industriji ", 30 Jupiter konferencija, Mašinski fakultet u Beogradu, Beograd April 2004 .
- [3] Somerville J., "Software Engineering", Addison-Wesley, Fifth-edition, 2001
- [4] Boehm B., "Software Engineering Economics", Prentice-Hall, 1981
- [5] Cappiello C., Francalanci C., Pernici B., Plebani P., Scannapieco M., "Data Quality Assurance in Cooperative Information Systems: a Multi-dimension Quality Certificate", IASI-CNR, Rome, Italy, 2002
- [6] Hu M., "Web Services Composition, Partition, and QoS in Distributed System Integration and Re-engineering", Police Information Technology Organisation, United Kingdom, 2003.
- [7] Unhelkar B., "Applying the UML to Enhance the Quality of Web Service", <http://www.methodscience.com>
- [8] Conti M., Kumar M., "Quality of Service in Web Service", IEEE Computer Magazine, 2001