



SE101 - RAZVOJ SOFTVERA I INŽENJERA SOFTVERA

Softverski projekti koji ne
uspevaju – razlozi i predlozi
rešenja

Lekcija 12

PRIRUČNIK ZA STUDENTE

SE101 - RAZVOJ SOFTVERA I INŽENJERA SOFTVERA

Lekcija 12

SOFTVERSKI PROJEKTI KOJI NE USPEVAJU – RAZLOZI I PREDLOZI REŠENJA

- ✓ Softverski projekti koji ne uspevaju – razlozi i predlozi rešenja
- ✓ Poglavlje 1: Socijalni i tehnički razlozi neuspeha projekata
- ✓ Poglavlje 2: Netačne procene i pogrešno planiranje
- ✓ Poglavlje 3: Netačni i previše optimistični izveštaji
- ✓ Poglavlje 4: Nerealni ciljevi
- ✓ Poglavlje 5: Neadekvatna kontrola kvaliteta
- ✓ Poglavlje 6: Znanje kao uzrok neuspeha projekata
- ✓ Poglavlje 7: Vežba - Pokazni primeri
- ✓ Poglavlje 8: Vežba - zadaci
- ✓ Zaključak

Copyright © 2017 – UNIVERZITET METROPOLITAN, Beograd. Sva prava zadržana. Bez prethodne pismene dozvole od strane Univerziteta METROPOLITAN zabranjena je reprodukcija, transfer, distribucija ili memorisanje nekog dela ili čitavih sadržaja ovog dokumenta., kopiranjem, snimanjem, elektronskim putem, skeniranjem ili na bilo koji drugi način.

Copyright © 2017 BELGRADE METROPOLITAN UNIVERSITY. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, without the prior written permission of Belgrade Metropolitan University.

▼ Uvod

UVOD

Cilj i sadržaj lekcije

Cilj lekcije predstavlja razumevanje razloga neuspeha softverskih projekata. U toku lekcije naučićete:

- šta lideri projekata i menadžeri kompanija vide kao glavne razloge neuspeha projekata
- koji su osnovni uzroci neuspeha
- razloge i tipove netačnih procena prilikom razvoja softvera
- šta sadrže mesečni izveštaji o statusu
- osnovne uzroke za nerealni pritisak na projekat
- kako neadekvatna kontrola kvaliteta može dovesti do neuspeha projekta
- koja je stopa defekata kod neuspešnih projekata
- kako neadekvatna kontrola kvaliteta može dovesti do neuspeha projekta
- uticaj znanja kao uzrok neuspeha projekata

▼ Poglavlje 1

Socijalni i tehnički razlozi neuspeha projekata

RAZLOZI NEUSPEHA PROJEKATA

Različite razloge za neuspeh softverskih projekata navode kako korporativni rukovodioci, tako i menadžeri samih softverskih projekata

Softver je važna, ali zabrinjavajuća tehnologija. Softverske aplikacije su pokretačka snaga savremenih poslovnih operacija, ali mnogi izvršni direktori vide softver kao jednu od glavnih problematičnih oblasti sa kojima se suočavaju velike korporacije. Niz pritužbi viših rukovodilaca protiv softverskih organizacija je dugačak, ali se može svesti na skup od tri veoma kritična problema koji se javljaju iznova i iznova u stotinama korporacija:

1. Procena - Softverski projekti se ne procenjuju ili ne planiraju sa prihvatljivom tačnošću.
2. Izveštaji - Izveštavanje o statusu softverskog projekta je često pogrešno i obmanjujuće.
3. Kvalitet i pouzdanost softvera su često neprihvatljivo loši.

Kada se intervjuišu i sami menadžeri softverskih projekata, oni se slažu da su tri glavne žalbe na softverske projekte stvarne i ozbiljne. Međutim, sa stanovišta softverskih menadžera, korporativni rukovodioci takođe doprinose softverskim problemima. U nastavku su tri osnovne primedbe upućene najvišem rukovodstvu:

1. Rukovodioci često odbacuju tačne i konzervativne procene.
2. Rukovodioci primenjuju štetan pritisak na plan projekta koji šteti kvalitetu.
3. Rukovodioci dodaju nove, velike zahteve u toku projekta razvoja softvera.

OSNOVNI UZROCI NEUSPEHA PROJEKATA

Postoji pet osnovnih faktora rizika za uspeh svakog softverskog projekta

Korporativni rukovodioci i menadžeri softverskih projekata imaju donekle različite stavove o tome zašto su problemi sa softverom tako rasprostranjeni. Međutim, u suštini i korporativni rukovodioci i menadžeri softvera vide iste probleme, ali ova pitanja izgledaju sasvim drugačije za svaku grupu.

U nastavku su navedeni osnovni uzroci pet faktora rizika za uspeh softvera:

1. Osnovni uzroci netačne **procene i planiranja rasporeda** .
2. Osnovni uzroci netačnog i optimističnog **izveštavanja** o statusu.
3. Osnovni uzroci nerealnih **pritisaka na raspored** .
4. Osnovni uzroci novih i promenljivih **zahteva** tokom razvoja.
5. Osnovni uzroci neadekvatne **kontrole kvaliteta**.

Ovih pet rizičnih oblasti su toliko kritične da se moraju kontrolisati ako je verovatno da će veliki projekti imati dobre šanse za uspešan ishod.

Svaki od navedenih faktora rizika detaljnije je opisan u objektima učenja koji slede.

▼ Poglavlje 2

Netačne procene i pogrešno planiranje

RAZLOZI I TIPOVI NETAČNIH PROCENA PRILIKOM RAZVOJA SOFTVERA

Problem zahtevanja preuranjene formalne procene pre potpunog definisanja zahteva postoji u softverskoj zajednici duže od 50 godina

Iz analize i diskusija o pitanjima procene sa nekoliko stotina menadžera i rukovodilaca u više od 75 kompanija između 1995. i 2006. utvrđeno je da su glavni uzroci problema sa procenom troškova sledeći:

1. Formalne procene se zahtevaju pre nego što se zahtevi u potpunosti definišu.
2. Istorijski podaci su retko dostupni za kalibraciju procena.
3. Dodati su novi zahtevi, ali prvobitna procena se ne može menjati.
4. Savremeni alati za procenu se ne koriste uvek na velikim softverskim projektima.
5. Konzervativne procene mogu biti poništene i zamenjene agresivnim procenama.

Prvo od navedenih pitanja procene – Formalne procene se zahtevaju pre nego što se zahtevi u potpunosti definišu – je endemski problem koji muči softversku zajednicu više od 50 godina. Problem rane procene nema savršeno rešenje od 2006. godine ali postoje neki pristupi koji mogu smanjiti rizike na prihvatljiv nivo.

Nekoliko alata za procenu troškova komercijalnog softvera ima režime rane procene koji mogu pomoći menadžerima u određivanju veličine projekta pre ispunjavanja potpunih zahteva, a zatim u proceni potreba za razvojnim osobljem, resursa, rasporeda, troškova, faktora rizika i kvaliteta. Za veoma rane procene, analiza rizika je ključni zadatak.

Ove rane procene imaju nivo poverenja koji u početku neće biti veoma visoki. Kako informacije postanu dostupne i zahtevi budu definisani, tačnost procene i nivo poverenja će se poboljšati. Međutim, procene troškova softvera izvršene pre potpunog razumevanja zahteva su suštinski teške. Zbog toga rane procene treba da uključuju nepredviđene uslove za promene zahteva i druge stavke troškova nizvodno.

ANALIZA UZROKA NETAČNE PROCENE

Stopa promene softverskih zahteva kreće se između 1 i 3 procenta po kalendarskom mesecu tokom faza projektovanja i kodiranja

Drugo pitanje procene – **Istorijski podaci su retko dostupni za kalibraciju procena** - usko je povezano sa prvim pitanjem. Kompanije kojima nedostaju istorijske informacije o osoblju, rasporedu, resursima, troškovima i nivoima kvaliteta iz sličnih projekata uvek su u opasnosti kada je u pitanju procena troškova softvera. Za one organizacije kojima nedostaju interni istorijski podaci, moguće je dobiti eksterne referentne informacije od brojnih konsultantskih organizacija. Međutim, obim eksternih referentnih podataka varira među industrijama, kao i izvori snabdevanja.

Treće pitanje procene – **Dodaju se novi zahtevi, ali se prvobitna procena ne može promeniti** - je pitanje novih i promenljivih zahteva bez mogućnosti promene prvobitne procene. Sada je poznato da se stopa promene softverskih zahteva kreće između 1 i 3 procenta po kalendarskom mesecu tokom faza projektovanja i kodiranja. Dakle, za projekat od 1.000 funkcionalnih poena i prosečnih 2 procenta mesečnih izmena tokom dizajna i kodiranja, nove funkcije koje se pojavljuju tokom dizajna i kodiranja će dodati oko 12 procenata konačnoj veličini aplikacije. Ova vrsta informacija može i treba da se koristi za preciziranje procena troškova softvera uključivanjem nepredviđenih troškova za očekivano povećanje zahteva. Kada se zahtevi promene, moguće je da neki projekti u nekim kompanijama revidiraju procenu kako bi odgovarali novom skupu zahteva. Međutim, mnogi projekti su primorani da pokušaju da ispune nove zahteve bez dodatnog vremena ili dodatnih sredstava. Brzina povećanja zahteva biće smanjena ako se koriste tehnologije kao što su dizajn zajedničkih aplikacija, izrada prototipa i inspekcije zahteva.

Četvrti problem procene – **Savremeni alati za procenu se ne koriste uvek na velikim softverskim projektima** - je neuspeh da se koriste najsavremenije metode za procenu troškova softvera. Neprikladno je koristiti gruba ručna pravila za važne projekte. Za velike softverske projekte koji imaju više od 1000 funkcionalnih tačaka, bilo koji od alata za procenu troškova komercijalnog softvera obično može da nadmaši ručne procene u smislu tačnosti, potpunosti i sposobnosti da se nosi sa nezgodnim situacijama kao što su povećanje broja zaposlenih i stopa rasta zahteva. Alati za procenu imaju još jednu veliku prednost: kada se dodaju nove funkcije ili promene zahtevi, ponovna procena obično traje samo nekoliko minuta. Pored toga, ovi alati će pratiti istoriju promena napravljenih tokom razvoja i, stoga, obezbediti koristan revizorski trag.

Peto i poslednje od glavnih pitanja – **Konzervativne procene mogu biti poništene i zamenjene agresivnim procenama** - je odbijanje konzervativnih ili tačnih procena troškova i planova razvoja od strane klijenata ili najviših rukovodilaca. Konzervativne procene su zamenjene agresivnijim koje su zasnovane na poslovnim potrebama, a ne na sposobnostima tima da isporuče. Za takve slučajeve najbolje rešenje je procena istorijskih podataka iz sličnih projekata.

▼ Poglavlje 3

Netačni i previše optimistični izveštaji

RAZLOZI NETAČNIH IZVEŠTAJA

Osnovni uzrok netačnog izveštavanja o statusu je taj što lideri projekata nisu obučeni za obavljanje ove važne aktivnosti

Jedan od najčešćih izvora trvenja između korporativnih rukovodilaca i softverskih menadžera je društveni problem da izveštaji o statusu softverskih projekata nisu tačni ili uverljivi. U slučaju za slučajem, mesečni izveštaji o statusu su optimistični da je sve po planu i pod kontrolom sve do malo pre planiranog starta kada se iznenada otkriva da nije sve bilo pod kontrolom i možda će biti potrebno još šest meseci.

Situacija dvosmislenog i neadekvatnog izveštavanja o statusu bila je uobičajena čak i vreme korišćenja modela vodopada za razvoj softvera. Netačno izveštavanje je još **češće u modernoj eri** gde spiralni model i druge alternative kao što su agilne metode i objektno orijentisana paradigma zamenjuju tradicionalne metode. Razlog je taj što ove nelinearne metode razvoja softvera nemaju istu preciznost za dovršavanje prekretnica kao što su to činile starije linearne softverske metodologije. Osnovni uzrok netačnog izveštavanja o statusu je taj što lideri projekata jednostavno nisu obučeni za obavljanje ove važne aktivnosti. Iznenadujuće je da se ni univerziteti ni mnogi interni programi obuke za menadžment ne bave izveštavanjem o statusu. Ako je projekat zaista pod kontrolom i po planu, onda izveštavanje o statusu neće oduzimati posebno vreme. Ali ako projekat izmiče kontroli, onda će izveštaji o statusu sadržati crvene zastavice ili odeljke upozorenja koji uključuju prirodu problema i plan da se projekat vrati pod kontrolu. Ovde će biti potrebno više vremena, ali ovo je vreme veoma dobro potrošeno.

Vreme i trud posvećeni pažljivom izveštavanju o statusu su jedno od najboljih ulaganja u softver koje kompanija može da napravi. Ovo ne bi trebalo da bude iznenađenje: izveštaji o stanju se dugo koriste za praćenje i kontrolu izgradnje drugih vrsta složenih inženjerskih projekata.

Nažalost, iz pregleda izveštaja o statusu brojnih projekata koji su završili na sudu zbog kršenja ugovora, netačno izveštavanje o statusu i dalje ostaje glavni faktor koji doprinosi prekoračenju troškova, prekoračenju rasporeda, kao i sudskim sporovima ako se projekat izvodi po ugovoru.

SADRŽAJ IZVEŠTAJA

Ključni element svakog mesečnog izveštaja je beleženje podataka o stvarnom broju grešaka u poređenju sa predviđenim brojem grešaka

U mesečnim izveštajima o statusu navodi se sedam opštih vrsta informacija:

1. Odstupanja troškova
2. Varijacije u rasporedu
3. Odstupanje od obima projekta
4. Varijanse otklanjanja defekta
5. Varijanse defekata
6. Završetak prekretnica
7. Problemi na koje se naišlo

Šest od ovih sedam elemenata izveštavanja su uglavnom kvantitativni, mada takođe mogu postojati objašnjenja zašto se varijacije javljaju, kao i njihov značaj. Ključni element svakog mesečnog izveštaja je **beleženje podataka o stvarnom broju grešaka u poređenju sa predviđenim brojem grešaka**. Ova aktivnost implicira postojanje alata i metoda za procenu kvaliteta.

Sledećim vrstama softvera su potrebni mesečni izveštaji o statusu:

- projekti čiji su ukupni troškovi razvoja značajni (>1.000.000\$),
- projekti čiji će ukupni rok razvoja biti duži od 12 kalendarskih meseci,
- projekti sa značajnom strateškom vrednošću za preduzeće,
- projekti kod kojih rizik od pomeranja može biti opasan (kao što su odbrambeni projekti),
- projekti sa značajnim interesovanjem za vrhunski korporativni menadžment,
- projekti kreirani po ugovoru sa kaznama za neizvršenje,
- projekti čiji je datum isporuke objavljen ili je važan za preduzeće.

▼ Poglavlje 4

Nerealni ciljevi

NEREALNI PRITISAK

Postoje četiri osnovna uzroka za nerealni pritisak na projekat

Nerealni pritisak na projekat od strane rukovodilaca ili clijenata je uobičajen faktor rizika za softver. Postoje četiri osnovna uzroka za nerealni pritisak na projekat:

1. veliki softverski projekti obično imaju duge rasporede od više od 36 meseci,
2. lideri projekata nisu u stanju da uspešno brane konzervativne procene,
3. istorijski podaci iz sličnih projekata nisu dostupni,
4. neka vrsta eksternog poslovnog roka utiče na raspored.

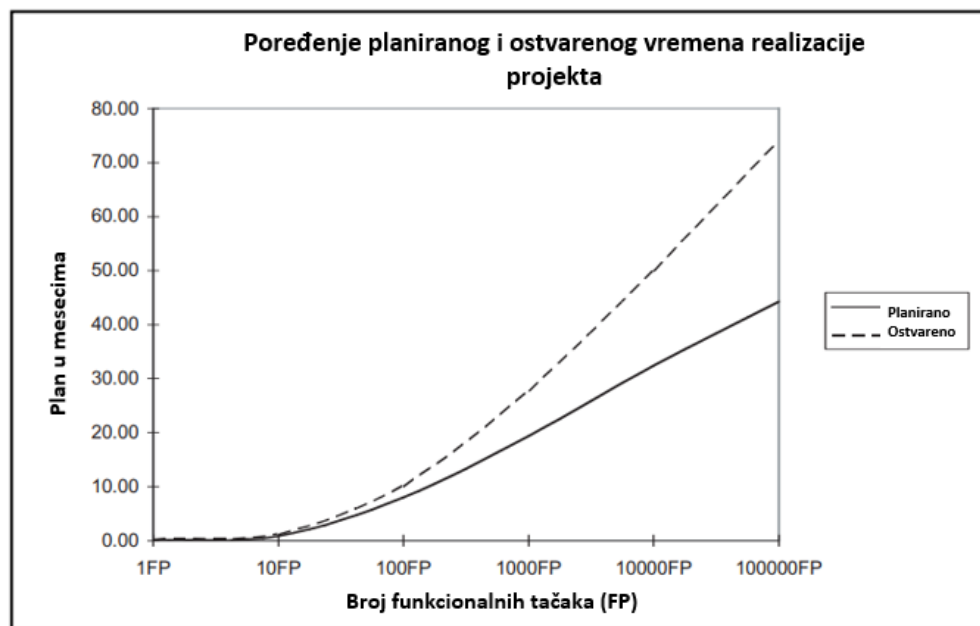
Slika 1 prikazuje iskustva američke industrije koja su proizašla iz nekoliko hiljada softverskih projekata. Gornja kriva prikazuje prosečno vreme isporuke u kalendarskim mesecima, dok donja kriva prikazuje planirano ili željeno vreme isporuke. Što je projekat veći, to je veći jaz između stvarnog i planiranog datuma isporuke aplikacije.

Dok korporativni rukovodioci možda žele da se veliki softverski projekat završi za 24 meseca, oni će gotovo sigurno prihvatiti 36-mesečni raspored kao činjenicu ukoliko znaju da nijedan od 50 sličnih projekata u njihovoj industriji nikada nije završen za manje od 36 meseci. Procene bi mogle biti poništene, ali tačni istorijski podaci će verovatno sprečiti da pritisak na planiranje projekta ne postane nerealan.

Najteže je rešiti problem kada neka vrsta spoljnog poslovnog roka utiče na dinamiku projekta. Nažalost, ovi poslovni rokovi su obično van kontrole kako lidera projekta, tako i tehničkog osoblja. Primeri spoljnih poslovnih rokova uključuju ugovorne obaveze, datume početka novih zakona koji zahtevaju softversku podršku ili neku vrstu tehničke situacije kao što su one povezane sa problemom. Takvi eksterni fiksni datumi ne mogu biti promenjeni ili bar ne promenjeni od strane osoblja projekta. Stoga kombinacija smanjenja funkcija, plus prekovremenog rada osoblja ostaje najčešći metod za rešavanje fiksnih i nepromenljivih datuma isporuke. Ako je propisani raspored prilično nemoguće postići, onda bi drastičnija opcija bila otkazivanje projekta.

POREĐENJE PLANIRANOG I OSTVARENOG VREMENA REALIZACIJE PROJEKTA

Slika 1 prikazuje grafik kreiran na osnovu iskustva iz američke industrije



Slika 4.1 Poređenje planiranog i ostvarenog vremena realizacije projekta [1,2]

NOVI I STALNO PROMENLJIVI ZAHTEVI

Prosečne stope promene od 2 procenta po kalendarskom mesecu ukazuju da su metode koje se trenutno koriste za prikupljanje i analizu početnih zahteva neadekvatne i da ih treba poboljšati

Osnovni uzroci promena zahteva su dinamična preduzeća. Stvarni zahtevi za softverom moraju se promeniti kao odgovor na nove poslovne potrebe. Međutim, prosečne stope promene od 2 procenta po kalendarskom mesecu ukazuju da su metode koje se koriste za prikupljanje i analizu početnih zahteva neadekvatne i da ih treba poboljšati. Ukupan akumulirani obim novih ili promenljivih zahteva može premašiti 50 procenata početnih zahteva kada se ukupni broj tačaka funkcije u fazi zahteva uporedi sa ukupnim vrednostima funkcionalnih tačaka pri primeni.

Najsavremenija kontrola promena zahteva uključuje sledeće:

- Zajednički odbor za kontrolu promena klijent/razvoj.
- Korišćenje dizajna zajedničkih aplikacija (DZA) za minimiziranje nizvodnih promena.
- Korišćenje formalnih prototipova da se minimiziraju nizvodne promene.
- Formalni pregled svih zahteva za izmene.
- Revidirane procene troškova i rasporeda za sve promene ispod 50 funkcionalnih tačaka.
- Određivanje prioriteta zahteva za promenom u smislu uticaja na poslovanje.
- Formalno dodeljivanje zahteva za izmene određenim izdanjima.
- Korišćenje automatizovanih alata za kontrolu promena sa mogućnostima unakrsnog referenciranja.

Jedna od uočenih posledica upotrebe formalnih DZA sesija je smanjenje broja zahteva za promenama na nižem nivou. Umesto da se neplanirani zahtevi pojavljuju po stopi od 1 do 3

procenta svakog meseca, studije o DZA od strane IBM-a i drugih kompanija su pokazale da neplanirane promene zahteva često padaju ispod 1 procenat mesečno zbog efikasnosti DZA tehnike. Prototipovi su takođe korisni u smanjenju stope promena zahteva. Obično su ključni ekrani, ulazi i izlazi napravljeni prototipom tako da korisnici imaju neko praktično iskustvo sa primerom završene aplikacije.

Međutim, promene će se uvek dešavati u velikim sistemima. Nije moguće zamrznuti zahteve bilo koje aplikacije iz stvarnog sveta. Stoga su vodeće kompanije spremne i sposobne da se nose sa promenama i ne dozvoljavaju da one postanu prepreka napretku.

▼ Poglavlje 5

Neadekvatna kontrola kvaliteta

KAKO NEADEKVATNA KONTROLA KVALITETA MOŽE DOVESTI DO NEUSPEHA PROJEKTA?

Osnovni uzrok loše kontrole kvaliteta je nedostatak čvrstih empirijskih podataka o isplativosti dobrog programa za kontrolu kvaliteta

Efikasna kontrola kvaliteta softvera je najvažniji pojedinačni faktor koji odvaja uspješne projekte od kašnjenja i katastrofa. Razlog za ovaj uspeh je taj što je pronalaženje i popravljavanje grešaka najskuplji element troškova za velike sisteme i zahteva više vremena od bilo koje druge aktivnosti. Osnovni uzrok loše kontrole kvaliteta je nedostatak čvrstih empirijskih podataka o isplativosti dobrog programa za kontrolu kvaliteta. Više od 50 godina empirijskih studija dokazalo je da projekti sa efikasnom kontrolom kvaliteta koštaju manje i imaju kraće vremenske rokove od sličnih projekata sa lošom kontrolom kvaliteta. Uspješna kontrola kvaliteta uključuje prevenciju kvarova, uklanjanje kvarova i aktivnosti merenja kvarova.

Fraza "*prevencija defekata*" uključuje sve aktivnosti koje minimiziraju verovatnoću stvaranja greške ili defekta na prvom mestu. Primeri aktivnosti za prevenciju defekata uključuju korišćenje Siks Sigma pristupa, korišćenje DZA za prikupljanje zahteva, korišćenje formalnih metoda projektovanja, korišćenje tehnika strukturisanog kodiranja i korišćenje biblioteka materijala za višekratnu upotrebu.

Fraza "*uklanjanje defekta*" podrazumeva sve aktivnosti koje mogu pronaći greške ili nedostatke u bilo kojoj vrsti rezultata. Primeri aktivnosti otklanjanja nedostataka uključuju inspekciju zahteva, inspekciju dizajna, inspekciju dokumenata, inspekciju koda i mnoge vrste testiranja.

Fraza "*merenje defekta*" uključuje mere grešaka pronađenih tokom razvoja, kao i defekte koje su korisnici prijavili nakon objavljivanja. Ove dve ključne mere omogućavaju vodećim kompanijama da izračunaju svoje stope efikasnosti otklanjanja kvarova ili procenite otkrivenih nedostataka pre objavljivanja softverskih aplikacija.

Neke aktivnosti istovremeno imaju koristi za prevenciju i uklanjanje kvara. Na primer, učešće u inspekciji dizajna i koda je veoma efikasno u smislu otklanjanja defekata, ali takođe koristi i prevenciji kvarova. Prevencija kvarova je potpomognuta jer učesnici inspekcije uče da izbegavaju vrste grešaka koje inspekcije otkrivaju.

Uspesne aktivnosti kontrole kvaliteta uključuju prevenciju kvarova, uklanjanje kvarova i merenja kvarova. Kombinacija aktivnosti prevencije i otklanjanja kvarova dovodi do nekih veoma značajnih razlika u ukupnim brojevima softverskih defekata između uspešnih i neuspešnih projekata.

STOPA DEFEKATA KOD USPEŠNIH I NEUSPEŠNIH PROJEKATA

Neuspešni projekti akumuliraju razvojne ukupne vrednosti od oko 7.0 defekata po funkcijskoj tački i uklanjaju samo oko 85 procenata njih pre isporuke

Uspešni projekti u rasponu od 10.000 funkcionalnih tačaka imaju broj isporučenih defekata oko 0.12 defekata po funkcijskoj tački ili 1200 ukupnih latentnih defekata. Od toga, oko 10 posto su prilično ozbiljni nedostaci, a ostali defekti su manji ili kozmetički nedostaci.

Nasuprot tome, neuspešni projekti akumuliraju razvojne ukupne vrednosti od oko 7.0 defekata po funkcijskoj tački i uklanjaju samo oko 85 procenata njih pre isporuke. Broj isporučenih defekata je oko 1.05 defekata po funkcijskoj tački ili 10.500 ukupno latentnih defekata. Od toga, oko 15 procenata su prilično ozbiljni nedostaci. Ovako veliki broj ozbiljnih defekata nakon isporuke veoma je zabrinjavajući za korisnike. Ako projekat ima više od oko 7.0 defekata po funkcijskoj tački i manje od 85 procenata efikasnosti uklanjanja, verovatno će biti otkazan jer nikada ne može uspešno izaći iz testiranja, a ciklus testiranja će se beznačajno odužiti.

Jedan od razloga zašto uspešni projekti imaju tako visoku efikasnost uklanjanja defekata u poređenju sa neuspešnim projektima je upotreba inspekcije dizajna i koda. Formalni dizajn i inspekcije koda u proseku imaju oko 65 procenata efikasnosti u pronalaženju nedostataka. Oni takođe poboljšavaju efikasnost testiranja obezbeđivanjem boljeg izvornog materijala za konstruisanje test slučajeva.

Neuspešni projekti obično izostavljaju inspekcije dizajna i koda i zavise isključivo od testiranja. Izostavljanje prethodne inspekcije uzrokuje tri ozbiljna problema:

1. Veliki broj defekata koji su i dalje prisutni kada testiranje počne usporava projekat do zastoja.
2. Stopa loših popravki za projekte bez inspekcija je alarmantno visoka.
3. Ukupna efikasnost uklanjanja defekta povezana samo sa testiranjem nije dovoljna da bi se postigle stope uklanjanja defekta veće od oko 80 procenata.

Srećom SEI, ISO standardi kvaliteta i pristup Siks Sigma doprineli su aktivnostima kontrole kvaliteta tokom proteklih 20 godina. Kao rezultat toga, sve veći broj velikih projekata je bio uspešan u poređenju sa sličnim projektima rađenim 1980-ih.

▼ Poglavlje 6

Znanje kao uzrok neuspeha projekata

NEDOSTATAK ZNANJA

Uzrok neuspeha projekta koji podrazumeva da menadžeri projekta nemaju dovoljno znanja je najlakše rešiti

Prvi od primarnih uzroka neuspeha projekta - *menadžeri projekta nemaju dovoljno znanja* - je najlakše ispraviti.

Jedno rešenje je da se obezbedi neophodna obuka. Džek Ferguson je tokom predavanja na radionicama Instituta za softversko inženjerstvo iznosio stav da se prečesto obuka o određenoj temi pruža prerano. Kada se obuka pruža prekasno, učenici su već shvatili potrebu i mogu lako da vide kako im principi koji se predaju mogu pomoći da budu uspešniji u ostvarivanju svojih projekata. To ne znači da je projekat u nevolji pre dobijanja obuke, ali se ta obuka pruža u odgovarajuće vreme u životnom ciklusu projekta. Na primer, pružanje detaljne obuke o zatvaranju projekta pre pokretanja projekta ima manju vrednost od pružanja takve obuke pošto projekat ulazi u fazu zatvaranja u životnom ciklusu, a članovi projekta sada shvataju važnost obuke. To, međutim, podrazumeva da projekat ima plan za obuku.

Nedostatak iskustva menadžera projekta se često smatra pitanjem šta je bilo prvo, kokoška ili jaje? Pojedincu je potrebno iskustvo da bi bio dobar menadžer, ali kako se može steći iskustvo ako im se ne pruži prilika da upravljaju? Odgovor je da se omogući menadžerima da steknu iskustvo tako što će ranije naučiti da upravljaju malim projektima čime stiču iskustvo i dobijaju poverenje za vođenje velikih projekata. Međutim, u praksi to nije uvek slučaj.

NEPRAVILNA PRIMENA ZNANJA

Postoji pet povezanih uzroka i posledica koji dovode do nepravilne primene znanja

Drugi osnovni uzrok neuspeha projekata - *menadžeri projekata ne primenjuju pravilno znanje koje imaju* - teže je otkloniti. Postoji pet povezanih uzroka i posledica:

- Ako je problem samo nedostatak pamćenja o onome što je trebalo da se uradi, onda blagi podsetnici od podređenih, kolega ili supervizora mogu biti katalizator za ispravljanje propusta. Kancelarije programa mogu da pruže mentorstvo projektnim menadžerima i pomognu u smislu pružanja obuke za osveženje znanja onima koji su ili zaboravili šta

treba da rade ili možda nisu upoznati sa aktuelnom politikom, direktivama, procedurama, itd.

- Priče obiluju projektima koji imaju zadatak da postignu nemoguće na osnovu nametnutih ograničenja. Na primer, menadžer projekta može da odredi da će projektu biti potrebno 36 meseci da se završi, ali je smer naniže da se obezbedi guma na rampi za 30 meseci. Jedini način da se to postigne je da se koriste prečice. Nažalost, korišćenje prečica obično rezultira produženjem projekta zbog prerade. Ono što bi u dobro planiranom projektu trajalo 36 meseci, sada traje 48 meseci. Jedini pravi lek za ovo pitanje je lični integritet: spremnost da se kaže višem rukovodstvu da je njihov zahtev nemoguće ostvariti, a zatim da im se pomogne da to shvate na osnovu trenutne tehnologije, politike, direktiva itd.
- Neki menadžeri projekata smatraju da su neke obavezne prakse upravljanja projektima od male vrednosti. Umesto da razgovaraju o svojim zabrinutostima sa višim menadžmentom, oni biraju da ignorišu princip sa kojim se ne slažu. Kada se uverenje da praksa ne dodaje vrednost spoji sa odlukom da se ne primeni, može doći do neuspeha projekta. Jedan od najboljih načina za rešavanje ovih problema je korišćenje istorijskih podataka iz drugih projekata da bi se potvrdila korist određenih praksi ili da bi se pokazale posledice njihovog nepoštovanja.
- Brojne su knjige koje podržavaju različite filozofije i metodologije upravljanja projektima. Neki menadžeri projekata neće verovati da su propisane filozofije i najbolje. Umesto da primenjuju metode koje smatraju pogrešnim, oni biraju da slede ono što smatraju ispravnim principima. Ova upotreba oskudnih resursa ima negativan uticaj na šanse za završetak projekta u okviru prvobitno dodeljenih troškova i rasporeda. Jedno rešenje za ovaj problem je postizanje dogovora između svih strana u vezi sa principima upravljanja projektom koji će se koristiti za ostatak projekta.
- Šta ukoliko primena čvrstih principa upravljanja projektom da bi se obezbedio uspeh projekta nije primarni cilj menadžera projekta? Iako se ovaj problem retko javlja, nekolicina menadžera projekata može videti svoju trenutnu poziciju kao odskočnu dasku za napredovanje na više pozicije. Njihov cilj može biti samo da pokažu dobre kratkoročne rezultate na račun ukupnog uspeha projekta.

▼ Poglavlje 7

Vežba - Pokazni primeri

UVOĐENJE SAP SISTEMA U LIDL

Prikaz prvog primera neuspelog projekta

U okviru dva časa vežbi studentima treba prikazati primere neuspeha softverskih projekata.

- **Uvođenje SAP sistema u Lidl supermarkete**

Lanac diskontnih supermarketa Lidl odustao je od SAP sistema samo tri godine nakon što je počeo rad sa SAP Retail-om. Kompanija je implementirala SAP sistem kako bi zamenila sopstveni sistem upravljanja robom koji je bio razvijen interno i koji je naišao na granice sopstvenog kapaciteta za inovacije i unapređenje. U to vreme, predstavnici Lidl-a izjavili su da je zastareli sistem ometan prekidima procesa, suvišnim skladištenjem matičnih podataka, prazninama u integraciji i funkcionalnim ograničenjima. Više interfejsa i decentralizovana struktura servera učinili su zadatak pokretanja i održavanja sistema sve složenijim.

Lidl je prvobitno počeo sa novim elektronskim sistemom upravljanja robom i informacionim sistemom u svojim austrijskim prodavnicama u maju 2015. godine, a planirao je da uvede sistem u 10.000 prodavnica i više od 140 logističkih centara. Međutim, ispostavilo se da je Lidl kasnije odlučio da odustane od projekta vrednog 500 miliona evra. U dopisu koji je šef Lidla poslao osoblju navodi se da se strateški ciljevi kako su prvobitno definisani projektom ne mogu postići bez da prodavac mora da potroši više nego što je želeo.

Nedavna studija je otkrila da je neuspeh SAP projekata da ispune očekivanja čest problem. Anketa od 113 pojedinaca u 105 kompanija otkrila je da samo 36% smatra da se njihov SAP projekat drži prvobitnog plana. Pored toga, nešto manje od polovine (48%) reklo je da njihov projekat nije postigao poslovne ciljeve. Izveštaj je takođe otkrio da je 52% anketiranih preduzeća reklo da je njihov projekat premašio budžet.

Teme za diskusiju:

1. Koji od 5 osnovnih uzroka neuspešnih projekata prepoznajete u ovom primeru i zašto? Netačne procene i pogrešno planiranje
2. Diskutujte o razlozima netačne procene prilikom planiranja navedenog projekta. (Objekat učenja 2)
3. Sa vaše tačke gledišta, koji su razlozi visokog procenta neuspeha SAP projekata?

PROJEKAT ZA AUTOMATIZACIJU PLATNOG SPISKA KANADSKJE VLADE - FENIX

Prikaz drugog primera neuspelog projekta

"Fenix" platni sistem je sistem za obradu platnog spiska za zaposlene u kanadskoj saveznoj vladi koji je kreirao IBM u junu 2011. koristeći softver PeopleSoft. Ovaj sistem prvi put je uveden 2009. godine kao deo Inicijative za transformaciju administracije plaćanja premijera Stivena Harpera, koja je nameravala da zameni kanadski 40 godina star sistem novim, štedljivim „komercijalnim gotovim proizvodom“. Do jula 2018. godine, Feniks je izazvao probleme sa platama blizu 80 procenata od 290.000 državnih službenika savezne vlade kroz nedovoljno, prekomerno i neplaćanje. Stalni komitet Senata za nacionalne finansije pokušao je da ispita uzroke neuspeha, održavši „osam sastanaka sa 28 svedoka, uključujući glavnog revizora Kanade, predstavnike sindikata, odeljenja i agencije, zvaničnike IBM-a, Ministra javnih službi i nabavki i službenika Tajnog saveta“ i posetivši lokaciju platnog sistema Miramiči tokom njihove istrage. U njihovom izveštaju „Problem sa plaćanjem u Feniksu: Rad na rešenju“ od 31. jula 2018. godine ovaj projekat je nazvan neuspehom i „međunarodnom sramotom“. Umesto uštede od 70 miliona dolara godišnje kako je planirano, u izveštaju se navodi da bi troškovi poreskih obveznika za rešavanje problema Feniksa mogli dostići ukupno 2,2 milijarde dolara do 2023. Kancelarija glavnog revizora Kanade je takođe izvršila nezavisnu reviziju i objavila izveštaj 2018. koji je zaključio da je projekat Fenix „bio neshvatljiv neuspeh u upravljanju projektom i nadzoru“.

Bilo je nekoliko uzroka za neuspeh projekta Fenix:

- Državni menadžeri su krivili nedostatak obuke za zaposlene, posebno one u novom platnom centru u Miramičiju.
- Savezni sindikati su krivili IBM, praveći poređenja sa sličnim projektom IBM-a koji je na kraju koštao 1,2 milijarde dolara.
- Bivša konzervativna vlada je okrivljena za prebrzo smanjenje zaposlenih i nedovoljno trošenje na obuku. Liberalna vlada je okrivljena za prebrzo uvođenje sistema i ignorisanje znakova upozorenja.

Teme za diskusiju:

- Koji od 5 osnovnih uzroka neuspešnih projekata prepoznajete u ovom primeru i zašto? Nerealan pritisak na projekat i nedostatak znanja
- Diskutujte o uzrocima nerealnog pritiska koji su doprineli neuspehu projekta. (Objekat učenja 4)
- Na koji način je nedostatak znanja uticao na neuspeh projekta i kako je mogao da bude izbegnut? (Objekat učenja 6)

MAC-OV OPERATIVNI SISTEM KOPLAND

Prikaz trećeg primera neuspelog projekta

Sve dok je Apple isporučivao Mekintoš računare, Mac operativni sistem (OS) je bio jasno merilo za kvalitetne računarske operativne sisteme. Međutim, do sredine 1990-tih, Mac-ov osnovni softver je počeo da izgleda po malo zastarelo. Sistem 7 je ostao superiorniji u odnosu na Vindovs 95, ali mnogim korisnicima je jaz izgledao manje očigledan nego ranije. Kopland, nazvan po američkom kompozitoru Aronu Koplandu, bio je projekat osmišljen da povрати konkurentsku prednost Apple-a. Najavljen u martu 1994. godine, OS je obećao mnoge funkcije koje danas izgledaju poznato. Stvari su izgledale na pravom putu neko vreme. U novembru 1995. Apple je objavio prvu Kopland beta verziju za odabranu grupu od oko 50 Mac programera. Međutim, stvari nikada nisu otišle dalje od toga. Uskoro je projekat manje ličio na novi operativni sistem, a više na ogromnu kolekciju novih tehnologija. Lista željenih karakteristika je rasla mnogo brže nego što su karakteristike mogle da se dopune. Kako je „paket“ rastao, testiranje je postajalo sve teže i inženjeri su još 1995. komentarisali da je Apple-ov najavljeni datum izlaska iz 1996. bio beznadežno optimističan.

Teme za diskusiju:

- Koji od 5 osnovnih uzroka neuspešnih projekata prepoznajete u ovom primeru i zašto? Nekontrolisano dodavanje novih zahteva
- Na koji način je dodavanje novih zahteva uticalo na neuspeh projekta? (Objekat učenja 2)
- Sa vaše tačke gledišta, na koji način je mogao biti sprečen neuspeh?

▼ Poglavlje 8

Vežba - zadaci

ZADACI ZA INDIVIDUALNI RAD STUDENATA

Zadaci za samostalno rešavanje na vežbi i kod kuće

Posle predavanja a pre časova vežbanja (održavaju se dva dana kasnije), poželjno je da pokušate da rešite neke od ovih zadataka radom kod kuće i da rezultate pošaljete mejlom, preko Zimbre, saradniku koji drži vežbe, najkasnije jedan sat pre održavanja vežbi. Ako neki zadatak nije urađen pre ili za vreme vežbi, preporučuje se studentu da ih uradi posle vežbi, kod kuće.

Obim zadatka može da zahteva najviše 30 do 40 minuta rada studenta. Svaki student mora da ima poseban zadatak, a na temu koja se ovde definiše. Studenti koji pošalju identična ili vrlo bliska rešenja, neće im ta rešenjabiti prihvaćena, tj. dobiće 0 poena.

Tekst domaćeg zadatka:

Odaberite sopstveni primer (različit za svakog studenta) neuspešnog softverskog projekta i objasnite osnovne razloge njegovog nespeha.

▼ Poglavlje 9

Zaključak

ZAKLJUČAK

Sumiranje stečenih znanja

- Postoji pet osnovnih faktora rizika za uspeh svakog softverskog projekta.
- Osnovni uzrok netačnog izveštavanja o statusu je taj što lideri projekata nisu obučeni za obavljanje ove važne aktivnosti.
- Ključni element svakog mesečnog izveštaja je beleženje podataka o stvarnom broju grešaka u poređenju sa predviđenim brojem grešaka.
- Postoje četiri osnovna uzroka za nerealni pritisak na projekat.
- Prosečne stope promene od 2 procenta po kalendarskom mesecu ukazuju da su metode koje se koriste za prikupljanje i analizu početnih zahteva neadekvatne i da ih treba poboljšati.
- Osnovni uzrok loše kontrole kvaliteta je nedostatak čvrstih empirijskih podataka o isplativosti dobrog programa za kontrolu kvaliteta.
- Uspešne aktivnosti kontrole kvaliteta uključuju prevenciju kvarova, uklanjanje kvarova i merenja kvarova.
- Uspešni projekti u rasponu od 10.000 funkcionalnih tačaka imaju broj isporučenih defekata oko 0.12 defekata po funkcijskoj tački ili 1200 ukupnih latentnih defekata, dok neuspešni projekti akumuliraju razvojne ukupne vrednosti od oko 7.0 defekata po funkcijskoj tački i uklanjaju samo oko 85 procenata njih pre isporuke.
- Uzrok neuspeha koji podrazumeva da menadžeri projekta nemaju dovoljno znanja je najlakše rešiti.
- Postoji pet povezanih uzroka i posledica koji dovode do nepravilne primene znanja.

LITERATURA

Pogodna literatura za učenje

Obavezna literatura:

1. Nastavni materijal za e-učenje na predmetu SE101 Razvoj softvera i inženjera softvera, Univeziitet Metropolitan, školska 2022/23. godina
Nastavni materijal je pripremljen korišćenjem reference 2.
2. Capers Jones, Social and Technical Reasons for Software Project Failures, CrossTalk, 2006
3. Timothy K. Perkins, Knowledge:The Core Problem of Project Failure, CrossTalk, 2006