

PLANIRANJE PROJEKTA

Poglavlje 23

Predmet: Razvoj softvera 2
Student: Uroš Jovanović 1047/2103
Profesor: Vladimir Filipović

Celine:

- Planiranje projekta
- Cena softvera
- Razvoj vođen planiranjem
- Raspored projekta
- Agilno planiranje
- Tehnike procene



Planiranje projekta

- Najvažniji posao menadžera projekta
- Plan projekta usmerava razvoj i pomaže u proceni napretka projekta
- Odvija se tokom 3 faze životnog ciklusa projekta:
 - U fazi ponude (spekulativno)
 - U inicijalnoj fazi razvoja (detaljnije)
 - Periodično tokom razvoja (reagovanje na promene, profinjenje plana)

Cena softvera

- **Troškovi** razvoja softvera se uglavnom sastoje iz 3 celine:
 - Radni trošak (plate za programere, inženjere, menadžere...)
 - Troškovi hardvera i softvera (uklj. održavanje)
 - Troškovi putovanja i obuke
- **Cena** razvijenog softvera = **troškovi** + **profit** (teorija)
 - U praksi ovo nije čest slučaj

Cena softvera

- U praksi prilikom formiranja cene treba uzeti više stvari u obzir
 - Organizacioni, ekonomski, politički, poslovni aspekti...
- Formiranje cene treba da bude grupna odluka
- Cenom do pobeđe (engl. *“Price to win”*)
 - Predlog cene na osnovu očekivanja klijenta
- U mnogim projektima cena je fiksiran faktor a zahtevi se menjaju shodno ceni

Cena softvera

- Neki od faktora koji utiču na formiranje cene:

Faktor	Opis
Probijanje na tržište	Prihvatanje niže cene zarad isticanja na tržištu
Nesigurna procena troškova	Povećanje cene usled nemogućnosti korektne procene troškova razvoja
Ugovorne obaveze	Korekcija cene na osnovu stavki ugovora (npr. isporuka izvornog koda uz softver)
Promenljivost zahteva	Niža početna cena koja se menja na više sa promenom zahteva klijenta
Finansijsko stanje	Nepovoljno finansijsko stanje razvijaoca obično rezultuje smanjenjem cena u cilju održavanja protoka novca

Razvoj vođen planiranjem

- Razvojni proces se **detaljno** planira
 - Raspodela posla, učesnici, rokovi, očekivani rezultati...
- Uglavnom se koristi za velike projekte, dok se za manje i srednje projekte više koristi agilno planiranje
- Prednosti:
 - Potencijalni problemi se otkrivaju pre samog početka razvoja
- Mane:
 - Pojedine odluke donešene za vreme ranog planiranja često moraju da se menjaju kako projekat odmiče

Razvoj vođen planiranjem

- Plan se obično sastoji iz sledećih delova :
 - Uvod (ciljevi, ograničenja)
 - Organizacija (učesnici, uloge)
 - Procena rizika (verovatnoća rizika, strategije)
 - Hardverski i softverski zahtevi (potrebni za razvoj)
 - Deljenje posla (deljenje na aktivnosti, ključne tačke)
 - Raspored projekta(aktivnosti, potrebno vreme, učesnici)
 - Mehanizmi za nadgledanje i izveštaj (koji mehanizmi, kakve izveštaje praviti i kada?)

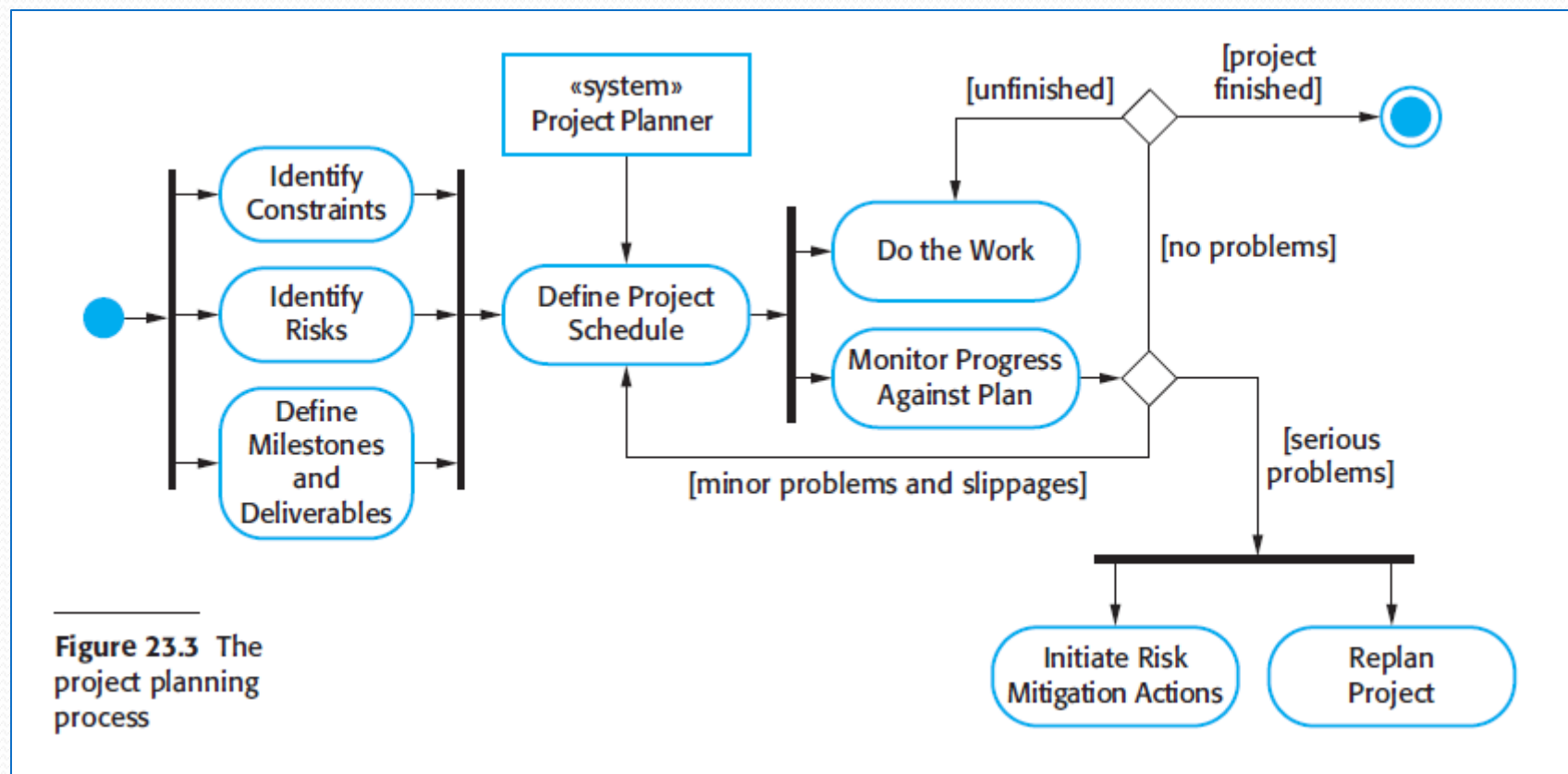
Razvoj vođen planiranjem

- Dodatni aspekti planiranja:

Plan	Opis
Plan kvaliteta	Opisuje kvalitet procedura i standarda u razvoju
Plan validacije	Opisuje način validacije sistema (korektnost, ispravnost...)
Plan menadžmenta konfiguracije	Opisuje strukture i procedure za menadžment konfiguracije
Plan održavanja	Predviđa zahteve i troškove održavanja
Plan razvoja osoblja	Opisuje kako će se veština i iskustvo zaposlenih razvijati

Razvoj vođen planiranjem

- Proces planiranja

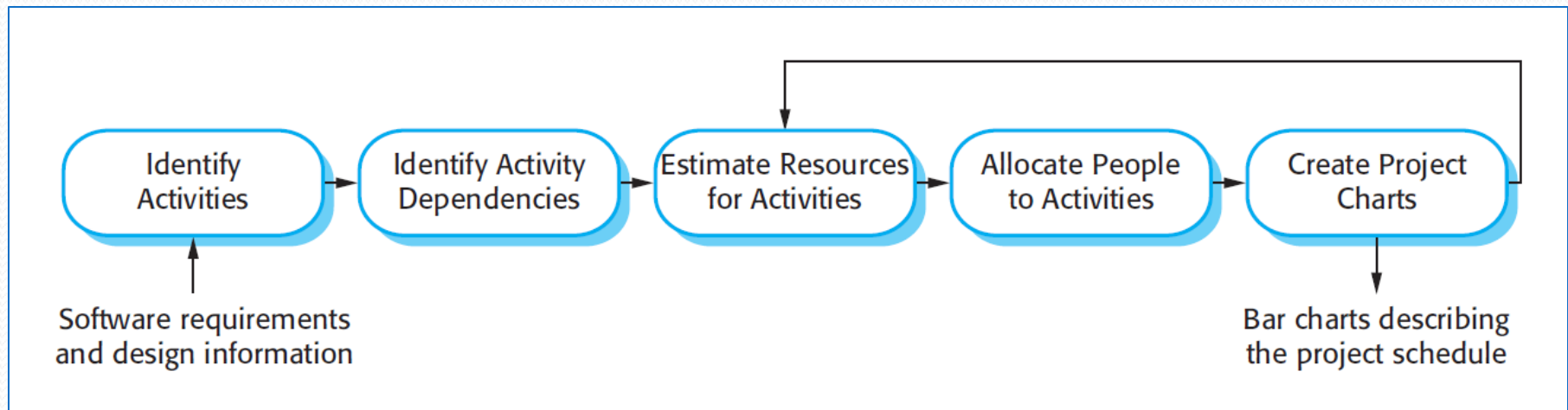


Raspored projekta

- Proces odlučivanja kako će projekat biti podeljen na manje zadatke i kada će se oni izvršavati
- Inicijalni raspored se obavlja na samom početku projekta
- Detaljnost rasporeda zavisi od načina planiranja
- Tehnički napredni projekti zahtevaju kontinualno osvežavanje rasporeda novim informacijama
- Dobar savet je napraviti optimističan raspored i na njega dodati vreme potrebno za saniranje eventualnih problema (dobro proceniti moguće probleme)

Raspored projekta

- Primer procesa raspoređivanja u razvoju vođenim planiranjem:



Raspored projekta

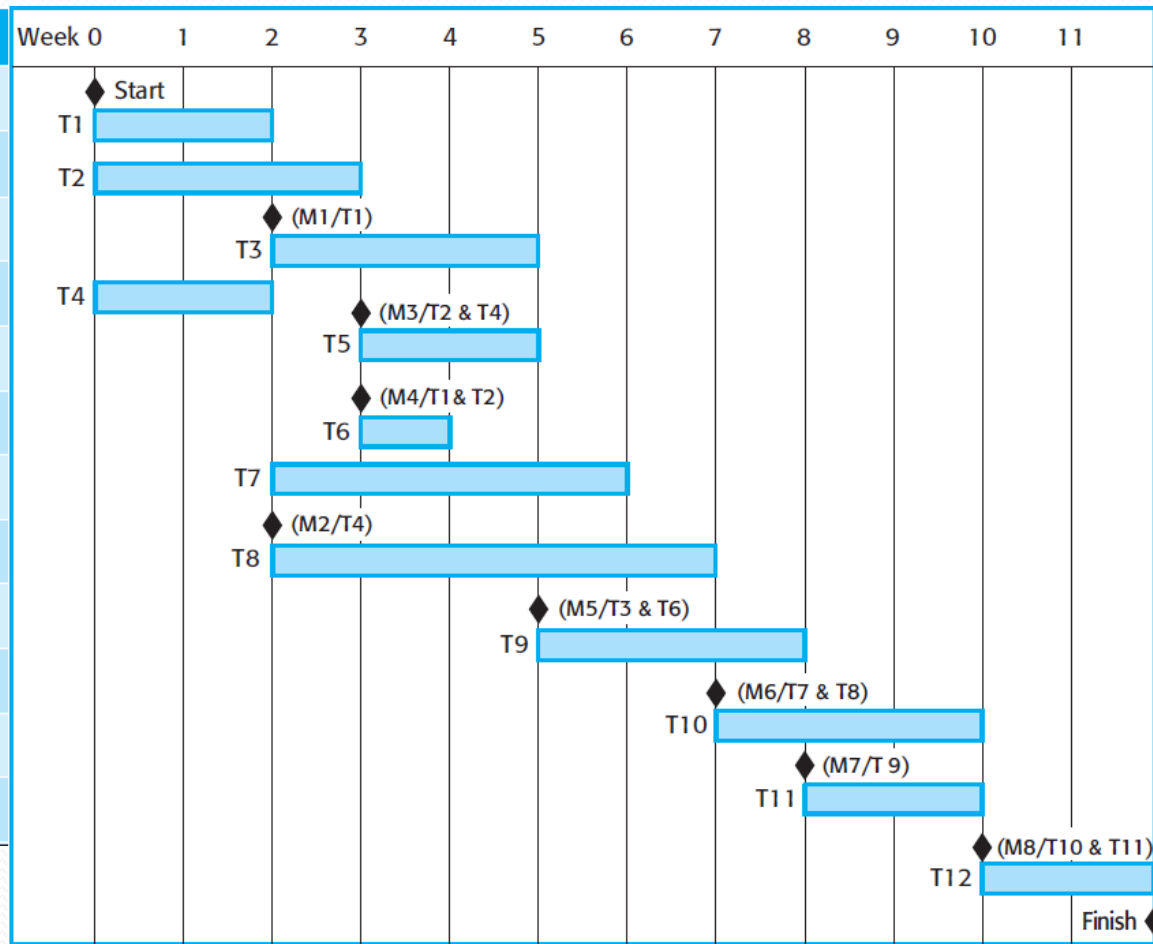
- Predstavljanje rasporeda:
 - **Grafikon** (vezan za kalendar, odgovornosti učesnika, proteklo vreme, početak i kraj aktivnosti, osmislio ga Henry Gantt – Gantt-ov dijagram)
 - **Mreža aktivnosti** (dijagram mreža, pokazuje zavisnosti između različitih aktivnosti)
- Raspored se obično generiše posebnim alatima
- Aktivnost je osnovni element planiranja, sadrži:
 - Vreme trajanja (u mesecima ili danima)
 - Procena rada (u jedinicama osoba-dan/osoba-mesec)
 - Krajnji rok
 - Definisana ciljna tačka (opipljiv rezultat)

Raspored projekta

- Tokom planiranja potrebno je definisati i ključne tačke (engl. “*Milestones*”)
- Ključne tačke predstavljaju pregled i procenu odrađenog posla, obično dokumentovano u nekom obliku
- Mogu biti vezane za jednu ili više (grupu) aktivnosti
- Specijalan tip ključne tačke je isporučiv materijal koji se šalje klijentu na procenu, obično kao rezultat neke veće celine ili bitne faze projekta

Raspored projekta

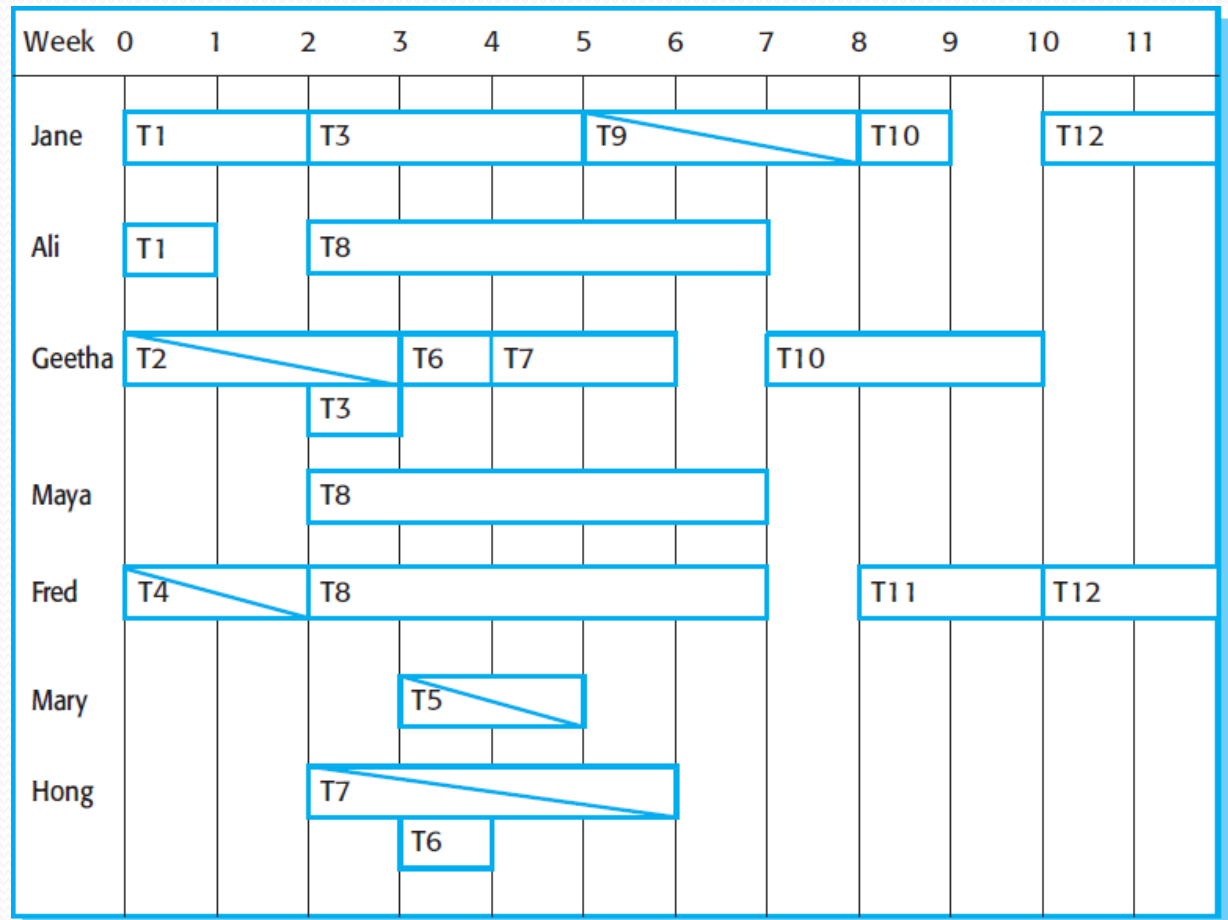
Task	Effort (person-days)	Duration (days)	Dependencies
T1	15	10	
T2	8	15	
T3	20	15	T1 (M1)
T4	5	10	
T5	5	10	T2, T4 (M3)
T6	10	5	T1, T2 (M4)
T7	25	20	T1 (M1)
T8	75	25	T4 (M2)
T9	10	15	T3, T6 (M5)
T10	20	15	T7, T8 (M6)
T11	10	10	T9 (M7)
T12	20	10	T10, T11 (M8)



Primer grafikona rasporeda

Raspored projekta

Raspored zaposlenih



Agilno planiranje

- Agilna metodologija predstavlja inkrementalni pristup razvoju softvera
- Softver se razvija i dostavlja klijentu u manjim celinama
- Funkcionalnost tih celina nije unapred isplanirana
- Najpoznatiji agilni pristupi su “Scrum” i ekstremno programiranje (“XP”) gde se planiranje vrši u 2 koraka:
 - Planiranje puštanja u rad (gleda unapred nekoliko meseci, određuje potrebne funkcionalnosti sistema pri puštanju u rad)
 - Planiranje iteracije (planira se naredna celina/inkrement koja obično traje 2-4 nedelje)

Agilno planiranje

- Proces planiranja u XP pristupu:



Agilno planiranje

- Prednosti:
 - Ceo tim ima pregled trenutnih zadataka u iteraciji i ko je zadužen za njih
 - Svaki član tima bira sebi zadatak – dodatna motivacija
- Mane:
 - Često aktivno učešće klijenta u razvoju nije moguće
 - Nije pogodno za velike timove gde se članovi često menjaju
- Agilno planiranje najbolje funkcioniše u malim, stabilnim timovima

Tehnike procene

- Često je teško napraviti procene u razvoju (troškovi, vreme, resursi...)
- U pojedinim slučajevima inicijalna procena X može biti promašena od $0.25X$ pa sve do $4X$
- Generalno, postoje 2 tehnike procena:
 - Procena bazirana na iskustvu
 - Oslanja se na prethodno iskustvo menadžera
 - Teško održiva usled brzog napretka tehnika razvoja softvera
 - Novi projekti često nisu slični prethodnim
 - **Algoritamsko modelovanje**

Tehnike procene

- **Algoritamsko modelovanje** koristi matematičke formule za procenu
- Formule zavise od veličine projekta, tipa softvera koji se razvija i ostalih timskih, procesnih i proizvodnih faktora
- Opšta formula ima oblik: $Trud = A * Veličina^B * M$
 - **A** – konstantan faktor, zavisi od tipa softvera koji se razvija i uobičajene lokalne organizacione prakse
 - **Veličina** – izražava veličinu softvera koji se razvija (broj linija koda ili neka druga mera veličine)
 - **B** – obično broj između 1 i 1.5
 - **M** – multiplikator koji predstavlja attribute procesa, proizvoda i razvoja poput iskustva razvojnog tima

Tehnike procene

- Nedostaci modela zasnovanih na algoritmima:
 - Teško je proceniti **Veličinu** u ranim fazama razvoja
 - Procena faktora **B** i **M** često je subjektivna
- Modelovanje algoritma je često kompleksno i teško za upotrebu, uzak opseg upotrebe
- Poželjno je razviti opseg procena (najgora, očekivana, najbolja) i primeniti formulu na svaku pojedinačno
- Poznavanje tipa softvera koji se razvija kao i predefinisani programski jezik i hardver povoljno utiču na procenu

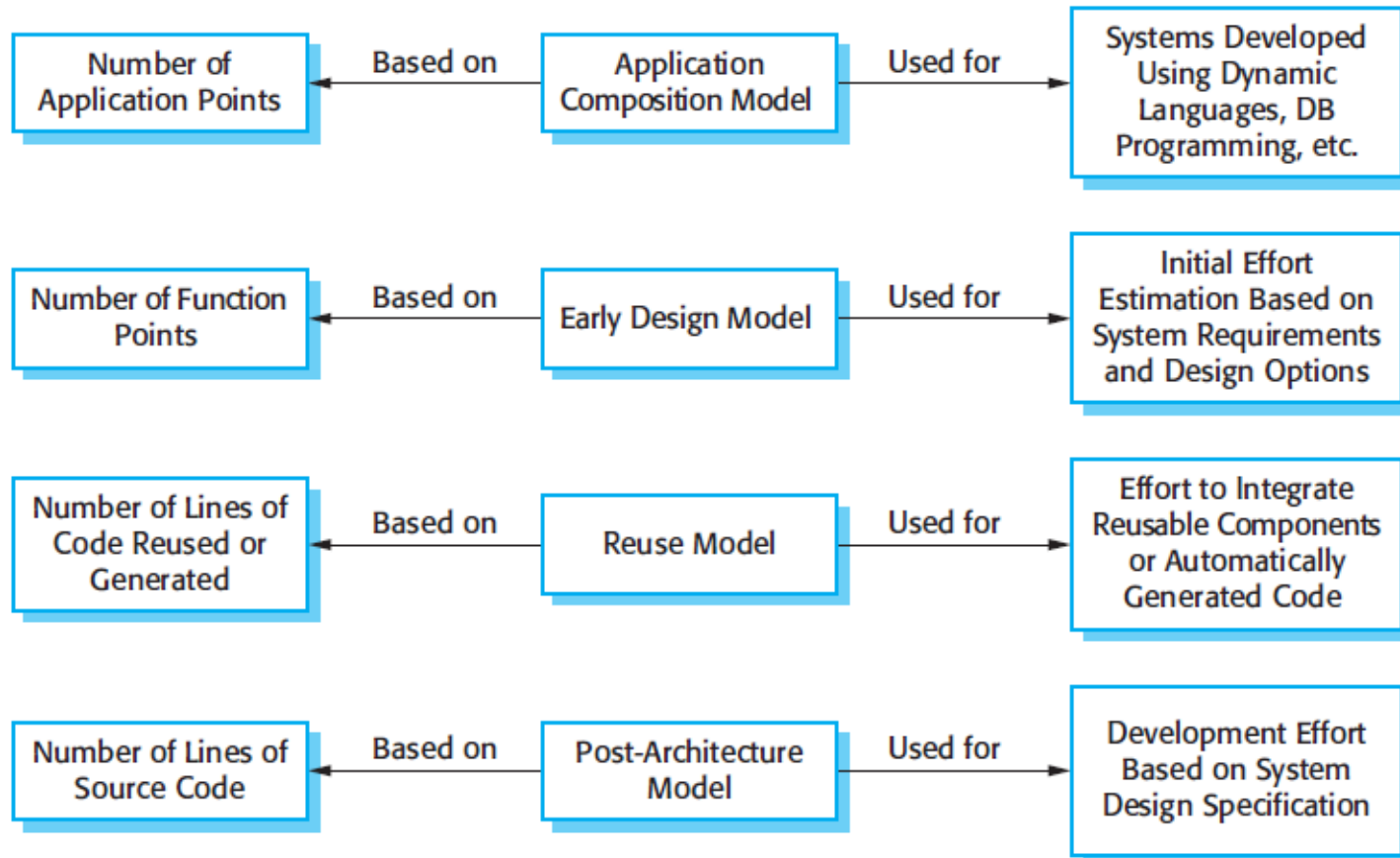
Tehnike procene

- **COCOMO II model**

- Spada u algoritamske modele
- Empirijski model
- Rezultat sakupljanja podataka velikog broja softverskih projekata
- Dobro dokumentovan i slobodan za javnu upotrebu
- Podržava moderne pristupe razvoja softvera
- Sastoji se iz 4 celine (podmodela)

Tehnike procene

- Delovi COCOMO II modela (podmodeli)



Tehnike procene

- Delovi COCOMO II modela (podmodeli):
 - **Kompozicija aplikacija**
 - Koristi se za nove projekte koji koriste postojeće komponente
 - Procena se bazira na tzv. “aplikacionim tačkama”
 - Formula: $PM = (NAP \times (1 - \%reuse/100))/PROD$
 - Ne uzima u obzir dodatni trud za ponovno iskorišćenje komponenata

Developer's experience and capability	Very low	Low	Nominal	High	Very high
ICASE maturity and capability	Very low	Low	Nominal	High	Very high
PROD (NAP/month)	4	7	13	25	50

Tehnike procene

- **Rani dizajn**

- Koristi se za brze inicijalne procene pre detaljnog projektnog plana (podrazumeva se da je dogovor sa klijentom napravljen)
- Korisno za pregled dostupnih opcija implementacije
- Formula: $\text{Effort} = A \times \text{Size}^B \times M$

Tehnike procene

- **Ponovna upotrebljivost**

- Dve vrste ponovno upotrebljivog koda
 - Black Box – upotrebljivost bez izmena i razumevanja koda
 - White Box – potrebne dodatne izmene i adaptacija
- Mnogi sistemi koriste automatski generisan kod (često iz modela predstavljenih UML-om)
- Formula: $PM_{Auto} = (ASLOC \times AT/100) / ATPROD$ // Estimate for generated code
- Procena potrebnog truda za implementaciju ponovno upotrebljenog koda se izražava preko broja linija ekvivalentnog novog koda
- Formula: $ESLOC = ASLOC \times (1 - AT/100) \times AAM$

Tehnike procene

- **Post-projektni model**

- Koristi se nakon što projektanti naprave strukturni dizajn
- Početna formula: $PM = A \times Size^B \times M$
- **Veličina** se obično izražava u linijama koda (KSLOC)
- U obzir veličine se uzima
 - Broj linija novog koda
 - Cena ponovnog upotrebljavanja koda
 - Broj linija koda čija je adaptacija potrebna za implementaciju
- Eksponent se računa na osnovu 5 faktora skaliranja
- $B = \text{suma faktora}/100 + 1.01$

Tehnike procene

- Faktori skaliranja

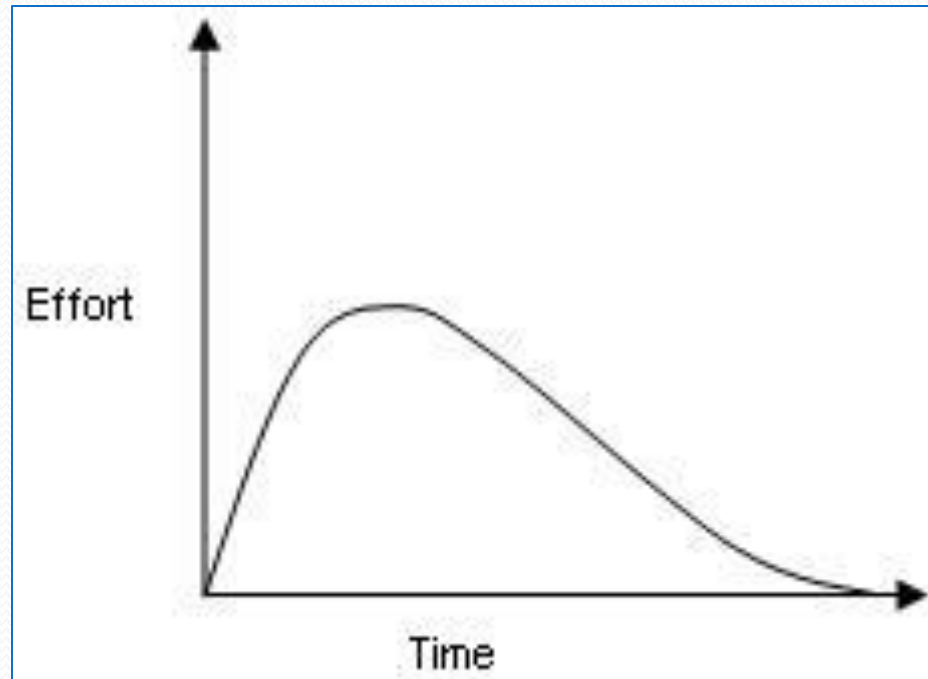
Faktor skaliranja	Objašnjenje
Prethodno iskustvo	Izražava upoznatost organizacije sa konkretnim tipom projekta
Fleksibilnost razvoja	Fleksibilnost razvojnog procesa. Uticaj klijenta na razvoj.
Procena rizika	Količina sprovedene analize rizika
Kohezija tima	Koliko dobro se članovi tima uklapaju u radu
Zrelost procesa	Odražava koliko je razvojni proces sazreo. Zreli procesi su korisni, automatizovani, pouzdani i konstantno se nadograđuju

Tehnike procene

- **Trajanje projekta i organizacija zaposlenih**
 - Pored procena troškova i truda, menadžeri moraju da procene i trajanje razvoja i br. potrebnih zaposlenih
 - COCOMO formula: $TDEV = 3 \times (PM)^{(0.33 + 0.2 \cdot (B - 1.01))}$
 - Često je procenjeno vreme veće od zahtevanog – dolazi do kompresije rasporeda
 - Kompresija rasporeda eksponencijalno utiče na rast potrebnog rada
 - Dodavanje zaposlenih radi ubrzanja razvoja uglavnom negativno utiče na produktivnost postojećeg tima

Tehnike procene

- Putnamov model procene





KRAJ