Procesne in projektne metrike

- Omogočajo inženirjem vpogled v potek projekta.
- Produktivnost/kvaliteta
- Zbrani podatki se analizirajo in primerjajo s podatki iz preteklih projektov
- Analiza (projektni vodje), zbiranje podatkov (inženirji)
- Brez meritev so ocene o poteku projekta subjektivne
- Z meritvami pa lahko ugotovimo različne trende (pozitivne, negativne, …) in ustrezno reagiramo

Postopek:

1) Definiramo množico projektnih metrik
2) Zberemo podatke, izračunamo metrike
3) Analiza in izpeljava zaključkov

Projektne metrike omogočajo:

- določiti status projekta
- zaslediti morebitna tveganja
- odkriti probleme preden ti postanejo kritični za cel projekt
- reorganizacijo
- ocenjevanje sposobnosti programerjev pri kontroli kvalitete produkta

Procesne metrike

Učinkovitost procesa merimo indirektno:

- Število odkritih napak pred dostavo PO uporabnikom
- Število odkritih napak s strani uporabnikov
- Število delovnih produktov
- Človeški napor, potreben za izdelavo (enota človek/mesec)
- Čas, potreben za izdelavo
- Časovno ujemanje procesa z načrtom

Projektne metrike

Projektne metrike iz prejšnjih projektov so osnova za oceno človeškega napora in časa, potrebnega za dokončanje trenutnega projekta.

Merimo:

- Število izdelanih modelov
- Čas, potreben za revizijo
- FP metrika
- Število vrstic kode
- Metrike programske opreme

Optimizacija časa, potrebnega za izdelavo

Ocena in izboljšava kvalitete PO

Metrike, normalizirane na velikost

Metrike, ki se normalizirajo glede na velikost projekta (LOC = število vrstic kode):

- Število napak na 1000 vrstic (KLOC)
- Cena na 1 KLOC
- Število strani dokumentacije na 1 KLOC
- Število napak na 1 človek/mesec
- Število KLOC na 1 človek/mesec

Pomanjkljivost: LOC je odvisna od programskega jezika!

Metrike, normalizirane na FP

Metrike, ki se normalizirajo glede na function point metriko

Lahko izračunamo podobne metrike kot pri metrikah, orientiranih na velikost (KLOC zamenjamo s FP metriko)

Metrika FP je neodvisna od programskega jezika!

Relacija med LOC in FP (v povprečju):

Programski jezik	LOC za 1 FP (v povprečju)
Assembler	337
C	162
C++	66
Java	63
JavaScript	58
Perl	60
Smalltalk	26
SQL	40
Visual basic	47

Objektno-orientirane metrike

- Število ključnih razredov (zelo neodvisne komponente, ki jih definiramo zgodaj v OO analizi)
- Število podpornih razredov (UI, dostop do baze, manipulacijski, računski, itd)
- Število podsistemov (skupek razredov, s katerimi realiziramo eno funkcionalnost, vidno uporabniku)

Metrike spletnih aplikacij

- Število statičnih spletnih strani
- Število dinamičnih spletnih strani
- Število internih povezav
- Število podatkovnih objektov (zapisi v podatkovnih bazah)
- Število povezav z zunanjimi sistemi
- Število statičnih vsebin (tekstovne, grafične, avdio, video informacije)
- Število dinamičnih vsebin
- Število izvršljivih funkcij (skriptov, appletov)

Metrike kvalitete

- Pravilnost (št napak na 1 KLOC), enostavnost vzdrževanja (MTTC – mean time to change), integriteta, uporabnost
- Učinkovitost pri odpravljanju napak: DRE = E / (D + E)
  - E – število napak, najdenih pred pred dostavo PO uporabnikom
  - D – število napak, ki jih najdejo uporabniki
- V splošnem: DRE = E_i / (E_i + E_{i+1})
  - E_i – število napak, najdenih v aktivnosti i

Ocenjevanje zahtevnosti projekta

- Osnovna akcija planiranja
- Po aktivnosti komunikacije je treba oceniti, koliko denarja, koliko človeškega napora, koliko ljudi in koliko časa bo potrebno za razvoj
- Če obstajajo metrike za pretekle projekte, je ocena lahko bolj natančna
- Načini ocenjevanja:
  - Pozno ocenjevanje
  - Ocenjevanje na osnovi podobnih projektov
  - Dekompozicija projekta in ocenjevanje na osnovi ocen posameznih delov
  - Uporaba enega izmed empiričnih modelov

Dekompozicija projekta

- Določanje obsega projekta (mehka logika, FP obseg, pristop s standardnimi komponentami, pristop sprememb)
- Določimo tri vrednosti: optimistično, pričakovano in pesimistično in izračunamo povprečje po formuli:

  $S = (S_{opt} + 4 \times S_p + S_{pes}) / 6$

- LOC in FP se uporabljata na dva načina:
  - kot osnovni spremenljivki za oceno “velikosti” vsakega elementa PO
  - kot osnovna metrika iz preteklih projektov za izračun obsega dela in stroškov
- Po oceni obsega projekta lahko projekt razdelimo na funkcije, razrede, poslovne procese, itd. in ocenimo obseg vsake take komponente posebej

Primer ocene na osnovi LOC metrike

Sistem za CAD aplikacijo za pomoč pri izdelavi mehaničnih komponent:

Funkcija	Ocena LOC
Uporabniški vmesnik	2300
2D geometrijska analiza	5300
3D geometrijska analiza	6800
Upravljanje s PB	3400
Pretvorba v grafične modele	5000
Upravljanje z zunanjimi napravami	2100
Moduli za prikaz na različnih napravah	8500
Skupaj:	33400

Če vemo, da je za podobne projekte povprečje 650 LOC / človek-mesec, lahko ocenimo trajanje projekta.
Če vemo, da za enega programerja podjetje porabi cca 3500 EUR/mesec, lahko ocenimo stroške za projekt.

Empirični modeli

- Ocenjujejo človeški napor kot funkcijo LOC ali FP metrike.
- Empirični podatki, iz katerih je izpeljana večina empiričnih modelov, so zbrani iz končnega števila projektov
- Zato noben model ni primeren za vse vrste PO in v vseh razvijalskih okoljih
- Večina jih ima obliko:

$$E = A + B \times (e_V)^C$$

E: človeški napor v človek-mesecih

A, B in C: empirično določene konstante

e_V: velikost projekta v LOC ali FP

Empirični modeli

- Walston-Felix model: $E = 5.2 \times KLOC^{0.91}$
- Bailey-Basili model: $E = 5.5 + 0.73 \times KLOC^{1.16}$
- Boehmov enostavni model: $E = 3.2 \times KLOC^{1.05}$
- Doty model za KLOC>9: $E = 5.288 \times KLOC^{1.047}$
- Albrecht-Gaffney model: $E = -91.4 + 0.355 \times FP$
- Kemerer model: $E = -37 + 0.96 \times FP$
- Regresijski model za male projekte: $E = -12.88 + 0.405 \times FP$

COCOMO II model

- Constructive Cost Model
- Uporablja objektne točke (NOP); indirektna mera, ki jo izračunamo iz števila zaslonov UI, poročil in ostalih komponent

Tip objekta	Enostaven	Srednji	Kompliciran
UI zaslon	1	2	3
Poročilo	2	5	8
3GL komponenta			10

- PROD je mera produktivnosti glede na kvaliteto programerjev

Kvaliteta programerjev	Zelo nizka	Nizka	Srednja	Visoka	Zelo visoka
PROD	4	7	13	25	50

- E = NOP / PROD

Hvala za pozornost! :)