

# Gestione – parte I

Rif. Ghezzi et al.  
Cap. 8 (8.1- 8.2)

# Management e suoi scopi

- *Management*, l'arte di far funzionare un sistema mediante persone

*La creazione e il mantenimento di un ambiente interno all'impresa dove gli individui possono operare in modo efficiente ed efficace per raggiungere gli obiettivi del gruppo (Koontz et al, 1980)*

- Creare un prodotto di alta qualità con risorse limitate
- Raggiungere obiettivi in parte contraddittori

# Scopi della gestione (management)

- Organizzare un'attività intellettuale, più problematica rispetto ad altre:
  - misura della produttività
  - pianificazione
  - stima di tempi e costi

# Attività di gestione

- Pianificazione
  - obiettivi
  - stima dei costi
  - rischi
- Acquisizione risorse
  - materiali
  - umane
- Controllo
  - direzione
  - monitoraggio
  - deviazioni dal piano

# Perché misurare?

- Rispondono all'esigenza del manager di un progetto software di disporre di una valutazione qualitativa e quantitativa del processo di sviluppo:

*You can't manage what you can't measure!*

# Supporto per:

- *stimare* la produttività e quindi fare confronti fra le diverse tecnologie e metodi di sviluppo per scegliere e migliorare
- *predire* la durata di un progetto e quindi il costo, utile sia per uso interno che per fare preventivi
- *pianificare* il lavoro di sviluppo in base alle risorse
- *controllare* che un progetto rispetti i tempi

# Supporto per:

*Stimare, predire, pianificare, controllare*

Da cui:

aumento della produttività

miglioramento della qualità

rispetto dei tempi

# Cosa misurare?

- In realtà non si vuole misurare solo il prodotto, ma l'intero processo di sviluppo. Ci sono quindi diversi tipi di grandezze di cui tenere conto.
- **Attributi del prodotto software:**
  - Dimensione
  - Complessità
  - Riusabilità
  - Altre

# Cosa misurare?

- **Attributi del processo**
  - Livello di sforzo (numero di ore/uomo o mesi/uomo)
  - Costi
  - Difetti
  - Durata

Queste grandezze servono a calcolare:

**Produttività=Dimensione/Sforzo**

**Qualità=Difetti/Dimensione**

# Quando misurare?

- La misurazione dovrebbe interessare tutte le fasi di sviluppo
  - Requisiti utente
  - Specifiche
  - Progetto
  - Scrittura del codice
  - Verifica
- La misurazione è tanto più utile quanto prima è effettuata.

# Metriche (*non esaustivo*)

- Teoria di Halstead
- Teoria di Mc Cabe
  
- Linee del codice (LOC, SLOC, etc)
- Function Point (FP), da specifiche espresse come diagrammi entity-relationship per la parte dati e specifiche in linguaggio naturale (uguali ai requisiti utente) o data flow diagrams per la parte processi

# Produttività

- Dimensione del software prodotto: come misurarla?
  - LOC (linee di codice, Lines of Code)
  - Punti funzione (Function Point, FP)
- Dipende dal tipo di applicazione e dagli strumenti (compreso il linguaggio) usati
- Produttività individuale e di squadra

# Costi

- Dipendenti dal numero dei progettisti necessari
- Stima iniziale
- Modelli
- Utilizzo:
  - valutazione della forza lavoro necessaria
  - confronto con i costi effettivi durante l'esecuzione del piano

# Rischi

- Eventi che possono portare a deviazioni dal piano
- Esempi:
  - difficoltà a trovare sviluppatori
  - uscita sul mercato di un software concorrente in anticipo
- Per ognuno:
  - valutazione della probabilità
  - analisi dell'impatto
  - contromisure

# Controllo di progetto

- Obiettivo: verificare che il progetto proceda secondo il piano
- Strumenti di controllo
  - Diagrammi di Gantt
  - Diagrammi PERT
- Gestione delle deviazioni dal piano
  - Piani realistici o prudenti per minimizzarle

# Miglioramento del processo

- I processi software sono complessi
- Influenzano
  - produttività
  - qualità
- Analisi e modellazione
- Modifica
- Modello di maturità

# Persone

- L'ingegneria del software è un'attività prevalentemente umana.
- La gestione delle persone è
  - decisiva
  - problematica
- Gestione di
  - individui
  - gruppi