

# Info 3 Teoria

Daniele Ravasio

## Indice

# 1 Metodologie Agili

## Variabili di controllo nello sviluppo di un processo software

1. **Tempo**, durata del progetto
2. **Qualità**, soddisfazione degli stakeholders
3. **Risorse**, personale, attrezzatura
4. **Scopo**, cosa c'è da fare, le feature da implementare

Queste variabili di controllo sono molto difficili da essere controllate tutte insieme, la via più semplice è quella di controllare lo **scopo**. La *teoria della modellazione del processo software* ha l'obiettivo di controllare le altre variabili.

## Modelli per il processo di sviluppo software

- Cascata (pianificato, lineare)
- Spirale (pianificato, iterativo)
- Agile (non pianificato, guidato dal test)
  - usato solitamente per processi "light"

### 1.1 Manifesto Agile

#### Metodologie Agili

C'è un approccio moderno alle pratiche di *sviluppo incrementale*, l'assunzione generale è "La comunicazione è necessariamente imperfetta". Inoltre l'aumento della documentazione non è necessariamente la risposta alle debolezze della pratica in evoluzione dello sviluppo.

Piuttosto ci sono certe attività complementari (**best practice** che possono aiutare ad aumentare la qualità o lo scopo delle interazioni come per esempio:

- pair programming
- timeboxing
- test-first development

### Manifesto Agile

Ci sono 4 valori dell'Agile e sono: Individui e interazioni, software lavorativo, collaborazione del cliente, rispondere al cambiamento, inoltre il Manifesto Agile segue 12 principi:

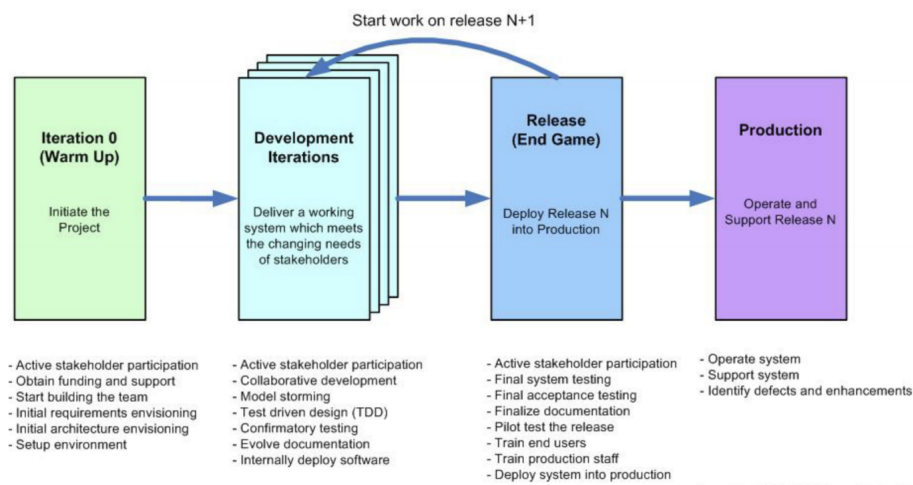
- 1 Soddisfazione del cliente
- 2 Accettare i cambi richiesti durante lo sviluppo del processo
- 3 Consegna frequente del software lavorativo
- 4 Collaborazione
- 5 Supporto, fiducia, motivazione persone coinvolte
- 6 Capacità di interazioni faccia-faccia
- 7 il Software lavorativo è la prima misura del progresso
- 8 Sviluppo sostenibile
- 9 Eccellenza tecnica e un buon design influenzano agilità

10 Semplicità

11 Team con auto-organizzazione interna

12 Auto-miglioramento

## 1.2 Processi Agili



Esiste anche l'approccio Extreme Programming (XP), segue uno specifico ciclo di vita che è: → **Planning** → **Design** → **Programmazione (refactoring)** → **Testing** → **Release**.

Dopo che è stata effettuata una release ed il progetto è andato in produzione si parte a lavorare sulla release N+1.

La consegna continua è data attraverso la pipeline dello sviluppo.

## 1.3 Esempio di *best practice* Agile

Ci sono varie best practice per i metodi agili:

•

ok

## 1.4 DevOps