

Orale APS

Giuseppe Facchi

27 giugno 2020

Indice

| | | |
|----------|---|----------|
| 1 | Processi per lo sviluppo software | 2 |
| 1.1 | Introduzione | 2 |
| 1.2 | Processo a cascata | 2 |
| 1.3 | Sviluppo iterativo ed evolutivo | 2 |
| 1.3.1 | Pianificazione iterativa, guidata dal rischio e dal cliente | 3 |
| 1.4 | UP | 3 |
| 1.4.1 | Le fasi di UP | 3 |
| 1.4.2 | Le discipline di UP | 3 |
| 1.5 | Agile | 4 |
| 1.6 | Fase di Ideazione | 4 |
| 2 | Requisiti evolutivi | 5 |
| 2.1 | Casi d'uso | 5 |
| 2.1.1 | Attori | 5 |
| 2.1.2 | Notazione: tre formati per i casi d'uso | 6 |
| 2.1.3 | Come trovare i casi d'uso | 6 |
| 2.1.4 | Verificare l'utilità dei casi d'uso | 6 |

1 Processi per lo sviluppo software

1.1 Introduzione

Un processo per lo sviluppo software definisce un approccio per la costruzione, il rilascio e la manutenzione del software. Esempi:

- **Processo a cascata**
- **UP**
- **Scrum**
- **Spirale**

1.2 Processo a cascata

Il processo software con ciclo di vita **a cascata** è basato su uno svolgimento **sequenziale** delle diverse attività.

Questo processo è **molto soggetto a fallimenti**, perché più si fa grande il software più è difficile implementare nuove features.

1.3 Sviluppo iterativo ed evolutivo

In questo approccio lo sviluppo è organizzato in una serie di mini-progetti brevi, di lunghezza fissa chiamati **iterazioni**. Il risultato di ciascuna iterazione è un **sistema eseguibile, testato e integrato**.

Può essere chiamato anche sviluppo iterativo e incrementale o sviluppo iterativo ed evolutivo.

Vantaggi

- **Minore probabilità di fallimento**
- Riduzione **precoce** dei rischi maggiori
- **Progresso visibile** fin dall'inizio
- **Feedback precoce**, con coinvolgimento dell'utente

Timeboxing: Le iterazioni hanno una **lunghezza fissata**

1.3.1 Pianificazione iterativa, guidata dal rischio e dal cliente

In ciascuna **iterazione** viene stabilito il piano di lavoro dettagliato per una sola iterazione. In *UP* viene effettuata alla **fine** di ciascuna iterazione, per decidere il piano dell'iterazione successiva.

1.4 UP

Unified Process è un processo **iterativo** per la costruzione di sistemi orientati agli oggetti. In particolare viene ampiamente adottato **RUP (Rational Unified Process)**, un suo raffinamento.

- Processo **iterativo**
- Le pratiche di UP forniscono un esempio di struttura rispetto a come **eseguire** e dunque come spiegare l'**OOA/D** (*Object-Oriented Analysis/Design*)
- UP è **flessibile** e può essere applicato usando un **approccio leggero e agile** come ad esempio **Scrum**

1.4.1 Le fasi di UP

- **Ideazione:** *NON è la previsione dei requisiti del modello a cascata, ma una fase di fattibilità*
- **Elaborazione**
- **Costruzione**
- **Transizione**

1.4.2 Le discipline di UP

Una **disciplina** è un insieme di attività e dei relativi elaborati di una determinata area. In UP un elaborato è un qualsiasi prodotto di lavoro. Ci sono diverse discipline in UP:

- **Modellazione del business:** Modello di dominio
- **Requisiti:** Modello dei Casi d'uso per definire requisiti funzionali e non funzionali

- **Progettazione:** Modello di Progetto

1.5 Agile

Non è possibile dare una definizione precisa di metodo agile perché le pratiche adottate variano notevolmente da metodo a metodo.

Una pratica di base è quella che prevede **iterazioni brevi**, con raffinamenti evolutivi dei piani, dei requisiti e del progetto.

Lo **scopo della modellazione e dei modelli** è di **agevolare la comprensione e la comunicazione**, **NON** di documentare.

1.6 Fase di Ideazione

Lo scopo della fase di ideazione è stabilire una visione iniziale comune per gli obiettivi del progetto.

- Non si definiscono tutti i requisiti nella fase di ideazione
- La maggior parte dell'analisi dei requisiti avviene durante la fase di elaborazione
- Modello dei Casi d'Uso

2 Requisiti evolutivi

Un sistema deve fornire un certo numero di funzionalità, relative alla gestione di alcune tipologie di informazione e al possesso di determinate qualità (sicurezza e prestazioni). Un requisito è una capacità o una condizione a cui il sistema deve essere conforme.

Requisiti funzionali Descrivono il comportamento del sistema in termini di funzionalità fornite ai suoi utenti. Possono essere espressi in forma di casi d'uso.

Requisiti non funzionali Non riguardano le specifiche funzioni del sistema, ma sono relativi a proprietà del sistema, ad esempio sicurezza, prestazioni, scalabilità, ecc.

2.1 Casi d'uso

In generale i casi d'uso sono storie scritte, testuali, di qualche attore che usa un sistema per raggiungere degli obiettivi. I casi d'uso non sono diagrammi, bensì testo.

Attore Qualcosa o qualcuno dotato di comportamento, come una persona o un'organizzazione o un sistema informatico.

Scenario (istanza di caso d'uso) E' una sequenza specifica di azioni e interazioni tra il sistema e alcuni attori.

Un caso d'uso è quindi una collezione di scenari correlati, sia di successo che di fallimento. I casi d'uso sono Requisiti Funzionali.

2.1.1 Attori

- **Attore Primario:** Utilizza direttamente i servizi del SuD (system under discussion) affinché vengano raggiunti gli obiettivi utente.
- **Attore Finale:** Vuole che il SuD sia utilizzato affinché vengano raggiunti dei suoi obiettivi. Spesso attore primario e finale coincidono (es. Cliente commercio elettronico).

- **Attore di Supporto:** Offre un servizio al SuD (es. Sistema di autorizzazione al pagamento).
- **Attore fuori scena:** Non è un attore primario, finale, di supporto (es. Governo).

2.1.2 Notazione: tre formati per i casi d'uso

- Formato breve
- Formato informale
- Formato dettagliato

2.1.3 Come trovare i casi d'uso

1. Scegliere i confini del sistema
2. Identificare gli attori primari
3. Identificare gli obiettivi di ciascun attore primario
4. Definire i casi d'uso che soddisfino gli obiettivi utente

2.1.4 Verificare l'utilità dei casi d'uso

Test del capo Il capo pone una domanda per cui ci sarà una risposta, se la risposta non soddisfa il capo il caso d'uso non è mirato a ottenere i risultati il cui valore sia misurabile. Non è però sempre vero (es. Autenticazione utente, concetto semplice ma difficilmente implementabile)

Test EBP (Elementary Business Process) Simile al test del capo. Un processo di business elementare è un'attività svolta da una persona in un determinato tempo e luogo, in risposta a un evento di business, che aggiunge valore e lascia i dati in uno stato consistente.

Test della dimensione Un caso d'uso deve essere costituito da più passi.

Esempi

- *Negoziare un contratto con un fornitore*: Troppo ampio per essere un EBP
- *Gestire una restituzione*: Passa il test del capo, è un EBP, le dimensioni vanno bene
- *Effettuare il login*: Non passa il test del capo
- *Spostare una pedina*: Non passa il test della dimensione