

Documentazione (10)

La documentazione deve essere una conseguenza dei processi organizzativi. Serve documentare nel modo meno intrusivo possibile. Fra le cose più importanti che fa la documentazione è fornire una misura sull'avanzamento del progetto (Piano di progetto), che fornisce dei **consuntivi** (stato fotografato) e **preventivi** (stime). I consuntivi sono finali se siamo alla fine del progetto, o parziali se siamo in corso. Documentare serve per dominare la complessità dei processi produttivi, attenuare la volatilità dei requisiti, facilitare il controllo di avanzamento.

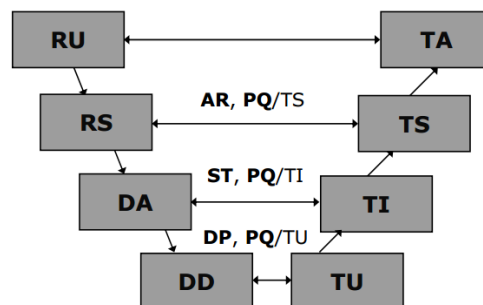
Misureremo le cose sulle quali possiamo e vogliamo fissare degli obiettivi di miglioramento. Misurazione per obiettivi, "ad hoc". Il responsabile di progetto deve avere un "cruscotto" con gli indicatori delle metriche che utilizziamo, e questi indicatori devono essere aggiornati. Vogliamo inoltre che il responsabile spenda il meno possibile per far questo devo documentare tutte le attività di pianificazione, gestione, sviluppo, verifica e validazione. Il piano ci dà degli obiettivi sui tempi e costi, le norme sono invece gli strumenti e le procedure che uso per rendere il piano fattibile. Il primo e più importante documento da realizzare (interno) sono le **norme di progetto**, che cresceranno nel tempo. Un altro documento importante è la specifica software, che è la descrizione ad alto livello del sistema.

Ogni architettura sw ha molte viste:

- **Modello statico:** Identifica le componenti principali e procede per decomposizione grafica;
- **Modello dinamico:** Illustra la struttura a processi del sistema;
- **Modello delle interfacce:** Definisce le interfacce del sistema;
- **Modello delle relazioni:** Identifica il flusso dei dati tra componenti distinti in relazione tra loro;
- **Modello di distribuzione:** Associazione tra nodi fisici e componenti logiche.

L'architettura va descritta in due documenti distinti:

- **Specifica tecnica** fissa linee e strategie di realizzazione, avvia la fase realizzativa ma non fissa i dettagli implementativi. Per ogni componente specifica la funzione svolta (con quali strutture dati, quali e quanti flussi di controllo) il tipo di dati in ingresso, in uscita, risorse logiche e fisiche necessarie per il suo funzionamento;
- **Definizione di prodotto:** consente di stimare costo e tempi di realizzazione, consente sviluppo parallelo dei componenti terminali. Decompone i componenti architetturali in moduli a grana fine finché ogni modulo ha dimensione, coesione, complessità e accoppiamento appropriati per la codifica in parallelo. In oltre questo documento deve fornire tutti i dettagli necessari alla codifica e alla verifica di ciascun modulo.



Per la documentazione si procede in modo **sequenziale**. In questo modo si suddividono i moduli grandi, in modo da agevolare i programmatori, i quali produrranno tanti moduli e li integreranno tra loro dopo la

validazione. Dentro il repository si devono caricare moduli già verificati in modo automatico e/o manuale (preferibilmente automatico perché la verifica manuale costa).

La progettazione di dettaglio fissa unità, fissa ciò che troveremo nel *ramo ascendente*. Quando ho unità buone le posso integrare (test di integrazione). Quando tutte le parti sono messe insieme allora posso fare un test di sistema. Nel percorso discendente il lavoro che riguarda la qualità deve includere specifiche progressive delle verifiche che farò per la produzione.

Stadi di preparazione dei documenti:

- Creazione
- Pulitura
- Rilascio

Le transizioni tra uno stato e l'altro hanno l'etichetta di "documento approvato".

NB *Tutto ciò che facciamo in discesa implica ciò che faremo in salita, e tutto ciò che facciamo in salita motiva quello che sta in discesa.*

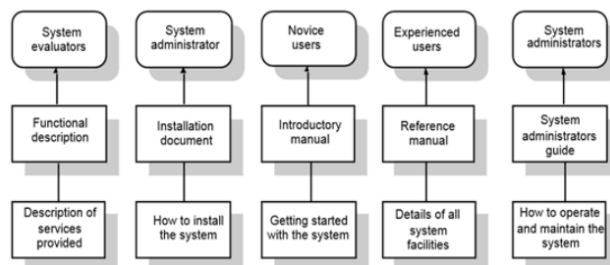
Il **tracciamento dei requisiti** fissa la relazione tra i prodotti del processo di sviluppo. Può essere fatto:

- **Forward:** in avanti (completezza), "non mi dimentico niente", ciascun ingresso a una fase deve essere messo in relazione con una specifica uscita di quella fase, fatto attraverso **matrici di tracciabilità**;
- **Backward:** all'indietro (necessità), ciascuna uscita di una fase deve essere messa in relazione con uno specifico ingresso a quella fase.

Il tracciamento deve essere assicurato ed automatizzato il più possibile. Dobbiamo avere **evidenza di copertura**. E' un lavoro potenzialmente oneroso e non può essere fatto a mano. I requisiti andranno identificati attraverso codici che li renderanno **univoci**. Un codice che definisca di cosa stiamo parlando (es. RU). Poi dovrò numerarli in modo gerarchico, in cui c'è uno spazio e un ambito. Non un numero in modo sequenziale ma raggruppato attorno a **categorie**. Non devo avere necessariamente solo numeri ma posso avere anche lettere.

Manuale utente: deve essere adatto alle caratteristiche dell'utente e alle caratteristiche dell'interfaccia utente. Un manuale serve come informazione supplementare. Si usa uno stile molto succinto e sintetico, non narrativo.

Tipi di manuali:



Stile di scrittura:

- Forma attiva invece che passiva;
- Correttezza grammaticale e tipografica;
- Frasi brevi e intorno a un solo fatto;

- Usare liste piuttosto che frasi;
- Paragrafi brevi, fatti di poche frasi;
- Stile non verboso;
- Terminologia precisa (glossario);
- Usare più punti di vista per descrizioni complesse;
- Usare sezioni e sottosezioni titolate.