

# Piano di qualifica

v0.5.0



<🔗> Farmacode

[farmacode.swe.unipd@gmail.com](mailto:farmacode.swe.unipd@gmail.com)

## Registro delle modifiche

Versione	Data	Scrittori	Revisori	Descrizione
0.5.0	2023-12-14	Pandolfo Mattia	Baggio Matteo	Prima stesura qualità di processo
0.4.0	2023-12-01	Carraro Alessandro	Rosson Lorenzo	Prima stesura qualità di prodotto
0.3.0	2023-11-27	Carraro Alessandro	Rosson Lorenzo	Stesura sezione qualità di architettura
0.2.0	2023-11-24	Carraro Alessandro	Rosson Lorenzo	Stesura sezione introduzione
0.1.0	2023-11-24	Carraro Alessandro	Rosson Lorenzo	Struttura iniziale del documento

# Indice

1) Introduzione .....	4
1.1) Scopo del documento .....	4
1.2) Glossario .....	4
1.3) Maturità e miglioramenti .....	4
1.4) Riferimenti .....	4
1.4.1) Riferimenti informativi .....	4
2) Piano di qualità .....	4
2.1) Qualità dell'architettura .....	4
2.1.1) Modularità .....	5
2.1.2) Disponibilità .....	5
2.1.3) Semplicità .....	5
2.1.4) Incapsulazione .....	5
2.1.5) Coesione .....	5
2.1.6) Accoppiamento .....	5
2.2) Qualità del software .....	6
2.3) Qualità di processo .....	7
2.3.1) Processi primari .....	7
2.3.2) Processi di supporto .....	7
2.3.3) Processi organizzativi .....	8
2.3.4) Metriche utilizzate .....	8
2.3.4.1) Processi primari .....	8
2.3.4.2) Processi di supporto .....	8
2.3.4.3) Processi organizzativi .....	9
3) Controllo di qualità .....	9
3.1) Test .....	9
3.2) Grafici .....	9
3.3) Tabelle .....	9
4) Miglioramenti .....	9

# 1) Introduzione

Il documento relativo al piano di qualità rappresenta un elemento di fondamentale importanza per i progetti di sviluppo software che mirano a conformarsi agli elevati standard di qualità definiti nei principi dell'ingegneria del software.

La qualità e la valutazione del prodotto sono due concetti indispensabili per fare confronti o determinare il grado di aderenza alle attese.

Il documento sarà utile a chi si occupa della creazione del prodotto, a chi userà il prodotto e a chi lo valuterà.

## 1.1) Scopo del documento

Il presente documento ha lo scopo di fornire una descrizione dettagliata e il più precisa possibile sulle metriche e sulle metodologie di controllo e misurazione della qualità nelle varie parti del prodotto software.

Verranno definiti gli obiettivi di qualità e i relativi processi e risorse necessarie per conseguirli, oltre alla definizione dei test con la loro documentazione: metodologie ed esito.

## 1.2) Glossario

Al fine di evitare eventuali equivoci o incomprensioni riguardo la terminologia utilizzata all'interno di questo documento, si è deciso di adottare un Glossario, con file apposito, in cui vengono riportate tutte le definizioni rigogliose delle parole ambigue utilizzate in ambito di questo progetto. Nel documento appena descritto verranno riportati tutti i termini definiti nel loro ambiente di utilizzo con annessa descrizione del loro significato.

La presenza di un termine all'interno del Glossario è evidenziata dal *colore blu*.

## 1.3) Maturità e miglioramenti

Questo documento è stato realizzato utilizzando un approccio incrementale, con lo scopo di semplificare i cambiamenti nel tempo in base alle reciproche esigenze decise da entrambi le parti, ovvero membri del gruppo di progetto e azienda proponente. Pertanto questo documento non può essere considerato esaustivo e completo.

## 1.4) Riferimenti

### 1.4.1) Riferimenti informativi

- T6 - Progettazione software (slide del corso di Ingegneria del Software);
- T7 - Qualità del software (slide del corso di Ingegneria del Software);
- T8 - Qualità di processo (slide del corso di Ingegneria del Software).

# 2) Piano di qualità

## 2.1) Qualità dell'architettura

La qualità di una buona architettura è definita da questi aspetti:

### Aspetti auto-esplicativi o non qualificabili

- Sufficienza;
- Robustezza;

### Aspetti considerati per qualificazione architettura

- Modularità;
- Disponibilità;

- Flessibilità;
- Riusabilità;
- Efficienza;
- Affidabilità;
- Sicurezza rispetto a malfunzionamenti;
- Sicurezza rispetto a intrusioni.
- Semplicità;
- Incapsulazione;
- Coesione;
- Basso accoppiamento;

### **2.1.1) Modularità**

Si intende la suddivisione dell'architettura in parti chiare e ben distinte.

I passaggi di qualità sono due:

- Determinare le parti che compongono l'interfaccia utente e quelle che compongono l'implementazione;
- Particolare attenzione va posta nell'evitare l'effetto domino, ovvero quando la modifica interna di una parte comporta modifiche anche al suo esterno.

Questi aspetti andranno verificati in fase di design e conseguentemente documentati. La strategia scelta per modularizzare è ricercando information hiding.

### **2.1.2) Disponibilità**

Anche detta availability, indica il grado di indisponibilità causata dalla manutenzione.

Questo aspetto andrà verificato in fase di design e conseguentemente documentato.

### **2.1.3) Semplicità**

Questo aspetto andrà verificato in fase di design e conseguentemente documentato.

In particolare andranno verificati:

- la presenza di elementi strettamente necessari;
- ogni elemento sarà il più semplice possibile, senza renderlo banale.

### **2.1.4) Incapsulazione**

Questo aspetto andrà verificato in fase di design e conseguentemente documentato.

In particolare andranno verificati:

- Rendere invisibile all'esterno le componenti architetture;
- Esporre solo l'interfaccia, nascondendo algoritmi e strutture dati usate per realizzarla;
- La manutenzione all'interfaccia non andrà a danneggiare l'implementazione.

### **2.1.5) Coesione**

Questo aspetto andrà verificato in fase di design e conseguentemente documentato.

In particolare andranno verificati:

- Funzionalità vicine stanno nella stessa componente;
- Minore interdipendenza fra componenti;
- Architettura più comprensibile;
- Giusto equilibrio fra modularità e coesione stessa.

La strategia scelta per modularizzare è ricercando information hiding.

### **2.1.6) Accoppiamento**

Questo aspetto andrà verificato in fase di design e conseguentemente documentato. In particolare andranno verificati:

- Dipendenze fra parti, esempio parti interne ed esterne;
- Accoppiamento sia minimizzato:
  - il grado U di utilizzo reciproco di M componenti.
- Metriche fan-in e fan-out:
  - SFIN, indice di utilità;
  - SFOUT, indice di dipendenza;
  - Una buona progettazione ha componenti con SFIN elevato.

## 2.2) Qualità del software

DA RISCRIVERE La Qualità è un aspetto fondamentale per la valutazione del prodotto, valutiamo per determinare il grado di conformità alle attese e ci si aspetta una valutazione da chi fa, da chi usa e da terze parti.

Chi sviluppa il prodotto avrà una visione intrinseca della qualità, ovvero conforme ai requisiti o idonea all'uso, chi usa il prodotto avrà una visione relativa della qualità, ovvero la soddisfazione del cliente ed infine terze parti che confrontano il prodotto avranno una visione quantitativa della qualità, ovvero una misurazione oggettiva o, appunto, per confronto.

Nel controllo di qualità bisogna assicurare conformità passo-passo invece che solo a fine corsa.

Obiettivo	Descrizione	Metriche
Funzionalità	Capacità del prodotto di offrire tutte le funzioni individuate nell'Analisi dei requisiti, soddisfacendo tutti i requisiti.	MPD3, MPD4 MPD5
Usabilità	Capacità di essere comprensibile e di facile utilizzo per l'utente, in modo da renderne piacevole l'esperienza.	MPD6, MPD7
Portabilità	Capacità di poter funzionare in diversi ambienti di esecuzione.	MPD8
Efficienza	Capacità di svolgere un compito nel minor tempo possibile e utilizzando la minor quantità possibile di risorse.	MPD9
Affidabilità	Capacità di svolgere i compiti anche in caso di problemi ed errori	MPD10
Copertura dei test	Capacità del prodotto di superare tutti i test a cui viene sottoposto, al fine di garantire una corretta implementazione dei requisiti individuati	MPD11, MPD12, MPD13, MPD14

===Metriche utilizzate

Codice	Nome metrica	Valore accettabile	Valore ottimale
MPD3	Copertura dei requisiti obbligatori	100%	100%

MPD4	Copertura dei requisiti desiderabili	$\geq 85\%$	100%
MPD5	Copertura dei requisiti opzionali	$\geq 70\%$	100%
MPD6	Facilità di utilizzo	$\leq 7$ click	$\leq 5$ click
MPD7	Tempo per l'apprendimento	$\leq 8$ minuti	$\leq 5$ minuti
MPD8	Versioni browser supportate	$\geq 80\%$	100%
MPD9	Tempo medio di risposta al comando di ricerca	5 secondi	3 secondi
MPD10	Gestione errori	$\geq 50\%$	100%
MPD11	Branch Coverage	$\geq 80\%$	100%
MPD12	Statement Coverage	$\geq 80\%$	100%
MPD13	Function Coverage	$\geq 80\%$	100%
MPD14	Line Coverage	$\geq 80\%$	100%

## 2.3) Qualità di processo

Per garantire la qualità dei processi, abbiamo utilizzato delle metriche che permettano di valutarli e accertarsi che questi raggiungano i corretti obiettivi di qualità previsti, per questo si è scelto di fare riferimento allo standard ISO/IE-C/IEEE 12207:1997, semplificando e adattando secondo le nostre necessità i processi in esso elencati.

La nomenclatura delle metriche utilizza l'acronimo MPC, ossia "Minimum Predictive Capability".

In questa sezione sono presentati i valori accettabili e i valori ottimali per i processi primari, di supporto e organizzativi.

### 2.3.1) Processi primari

Obiettivo	Descrizione	Metriche
Fornitura	Processo che consiste nell'individuare procedure e risorse adatte a soddisfare le necessità del cliente, secondo i termini e le condizioni stabiliti.	MPC1, MPC2, MPC3, MPC4, MPC5, MPC6, MPC7, MPC8
Sviluppo	Processo che ha lo scopo di determinare attività e compiti necessari alla realizzazione di un prodotto software che rispetti le esigenze del cliente.	MPC9, MPC10

### 2.3.2) Processi di supporto

Obiettivo	Descrizione	Metriche
Verifica	Processo che ha lo scopo di confermare che ciascun servizio realizzato soddisfi i requisiti specificati dal cliente.	MPC11, MPC12
Accertamento della qualità	Processo con lo scopo di assicurare che il prodotto e i servizi offerti siano conformi agli standard definiti, rispettando gli obiettivi di qualità.	MPC13

### 2.3.3) Processi organizzativi

Obiettivo	Descrizione	Metriche
Gestione organizzativa	Processo che si occupa di regolare le modalità di coordinamento del gruppo.	MPC14

### 2.3.4) Metriche utilizzate

#### 2.3.4.1) Processi primari

Codice	Nome metrica	Valore accettabile	Valore ottimale
MPC1	Estimated at Completion (EAC)	$\pm 5\%$ rispetto al preventivo	Uguale al preventivo
MPC2	Estimate to Complete (ETC)	$\geq 0$	$\leq$ EAC
MPC3	Earned Value (EV)	$\geq 0$	$\leq$ EAC
MPC4	Actual Cost (AC)	$\geq 0$	$\leq$ EAC
MPC5	Cost Variance (CV)	$\geq -10\%$	$\geq 0\%$
MPC6	Planned Value (PV)	$\geq 0$	$\leq$ Budget at Completion
MPC7	Schedule Variance (SV)	$\geq -10\%$	$\geq 0\%$
MPC8	Cost Performance Index (CPI)	$\pm 10\%$	0%
MPC9	Requirements Stability Index (RSI)	$\geq 80\%$	100%
MPC10	Satisfied Obligatory Requirements (SOR)	100%	100%

#### 2.3.4.2) Processi di supporto



Codice	Nome metrica	Valore accettabile	Valore ottimale
MPC11	Code Coverage (CC)	$\geq 80\%$	100%
MPC12	Passed Tests (PT)	$\geq 80\%$	100%
MPC13	Quality Metrics Satisfied (QMS)	$\geq 90\%$	100%

#### 2.3.4.3) Processi organizzativi

Codice	Nome metrica	Valore accettabile	Valore ottimale
MPC14	Non-calculated Risk (NCR)	$\leq 5$	0

### 3) Controllo di qualità

#### 3.1) Test

#### 3.2) Grafici

#### 3.3) Tabelle

### 4) Miglioramenti