

Piano di Qualifica

Contatti: swateng.team@gmail.com

Versione: 0.2



Registro delle Modifiche

| Versione | Data | Descrizione | Autore | Ruolo |
|----------|------------|--|--------------------|----------------|
| 0.2 | 28-11-2023 | Inizio stesura sezione qualità di Processo | Riccardo Costantin | Amministratore |
| 0.1 | 27-11-2023 | Stesura iniziale introduzione | Riccardo Costantin | Amministratore |

Indice

| | |
|--|---|
| 1 Introduzione | 4 |
| 1.1 Scopo | 4 |
| 2 Qualità di Processo | 5 |
| 2.1 Obiettivi | 5 |
| 2.2 Processi Primari | 5 |
| 2.2.1 Fornitura | 5 |
| 2.2.2 Sviluppo | 7 |
| 2.2.2.1 Progettazione architetture | 7 |
| 2.2.2.2 Codifica | 7 |
| 2.3 Processi di Supporto | 8 |
| 2.3.1 Documentazione | 8 |

1 Introduzione

Il *Piano di Qualifica* è un documento fondamentale che delinea le strategie e le attività mirate a garantire la qualità del prodotto. Adattandosi alle mutevoli esigenze e ai requisiti del progetto didattico, questo strumento si propone di guidare il team attraverso un percorso di miglioramento continuo, definendo metriche quantitative ed effettuando valutazioni qualitative che rispecchino l'efficacia ed efficienza del processo di sviluppo. Il piano, orientato all'incrementalità, mira a fornire una chiara roadmap per la valutazione continua, includendo la definizione dettagliata di test, criteri di accettazione e la documentazione accurata degli esiti ottenuti. Il documento costituisce un impegno concreto per assicurare la qualità in modo costante e definita nel corso dello sviluppo del progetto.

1.1 Scopo

Il presente documento ha i seguenti scopi:

- Definizione standard di qualità;
- Definizione metriche e criteri quantitativi per valutare l'efficacia del processo e del prodotto;
- Gestione dei test: definizione di test, la pianificazione delle attività di test, la valutazione dei risultati;
- Verifica e validazione del prodotto includendo test di unità, di integrazione, di sistema, di accettazione;
- Gestione dei Difetti e delle Anomalie riscontrate durante le attività di verifica;
- Miglioramento continuo attraverso l'adattamento del piano in base alle esperienze acquisite e alle modifiche nei requisiti del progetto.

2 Qualità di Processo

2.1 Obiettivi

Si riferisce all'efficacia ed efficienza dei processi e delle metodologie impiegate nello sviluppo, testing e mantenimento del software. Essa abbraccia l'insieme di attività, pratiche e metodi utilizzati lungo l'intero ciclo di vita dello sviluppo del software al fine di garantire che il prodotto finale soddisfi i requisiti specificati e venga consegnato puntualmente ed entro il budget previsto. In questo contesto, la qualità di processo è fondamentale per assicurare un'implementazione coerente, misurabile e ottimizzata delle fasi di sviluppo, contribuendo a conseguire gli obiettivi di qualità complessivi.

2.2 Processi Primari

2.2.1 Fornitura

Le metriche di progetto usate nella fornitura sono misure utilizzate per valutare diversi aspetti del processo di sviluppo e del prodotto risultante. Alcune delle metriche comuni utilizzate nella fornitura di progetti software includono:

- **TCP:** *Total Cost Project* indica la somma totale dei costi associati al progetto;
- **BAC:** *Budget At Completion* indica il budget totale pianificato per il completamento del progetto;
- **EV:** *Earned Value* rappresenta il valore prodotto dal progetto ossia il valore dei *deliverable_G* rilasciati fino al momento della misurazione in seguito alle attività svolte;
 - Formula: $EV = \text{Percentuale di completamento del lavoro} \cdot BAC$.
- **PV:** *Planned Value* il valore del lavoro pianificato fino a un dato momento;
 - Formula: $PV = \text{Percentuale di pianificazione del lavoro} \cdot BAC$.
- **AC:** *Actual Cost* il costo effettivo sostenuto fino a un dato momento;
- **CPI:** *Cost Performance Index* misura l'efficienza del costo del lavoro svolto fino a un dato momento;
 - Formula: $CPI = \frac{EV}{AC}$.
- **SPI:** *Schedule Performance Index* misura l'efficienza del tempo rispetto alla pianificazione del progetto. Fornisce un indicatore numerico che rappresenta il rapporto tra il lavoro effettivamente eseguito (o il valore guadagnato) e il lavoro pianificato fino a un determinato punto nel tempo. Aiuta a valutare quanto il progetto sta rispettando il programma pianificato;
 - Formula: $SPI = \frac{EV}{PV}$.

- **EAC:** *Estimated at Completion* revisione del valore stimato per la realizzazione del progetto, ossia il BAC rivisto allo stato corrente del progetto;
 - Formula: $EAC = \frac{BAC}{CPI}$.
- **ETC:** *Estimated To Completion* stima del costo aggiuntivo necessario per completare il progetto;
 - Formula: $ETC = EAC - AC$.
- **VAC:** *Variance at Completion* la differenza tra il budget previsto e il budget attuale alla fine del progetto;
 - Formula: $VAC = BAC - EAC$.
- **SV:** *Schedule Variance* indica se si è in linea, in anticipo o in ritardo rispetto alla schedulazione delle attività di progetto pianificate nella baseline;
 - Formula: $SV = EV - PV$.
- **CV:** *Cost Variance* la differenza tra il valore del lavoro effettivamente svolto e il costo effettivo del lavoro svolto fino a un dato momento;
 - Formula: $CV = EV - AC$.
- **BV:** *Budget Variance* indica se alla data corrente si è speso di più o di meno rispetto a quanto inizialmente previsto nel budget.
 - Formula: $BV = PV - AC$.

Le metriche della qualità di processo verranno identificate con:

MPC-[Nome abbreviato]

Legenda:

- **Nome abbreviato:** Nome della metrica specifica abbreviata.

| Metrica | Descrizione | Valore accettazione | Valore ideale |
|----------------|----------------------------|---------------------------------|----------------------|
| MPC-EV | Earned Value | ≥ 0 | $\leq EAC$ |
| MPC-PV | Planned Value | ≥ 0 | $\leq BAC$ |
| MPC-AC | Actual Cost | ≥ 0 | $\leq EAC$ |
| MPC-CPI | Cost Performance Index | tra 0.95 e 1.05 | > 1 |
| MPC-SPI | Schedule Performance Index | tra 0.95 e 1.05 | > 1 |
| MPC-EAC | Estimated at Completion | deviazione del $\pm 5\%$ | $= BAC$ |
| MPC-ETC | Estimated To Completion | ≥ 0 | $\leq EAC$ |
| MPC-VAC | Variance at Completion | deviazione del $\pm 5\%$ budget | ≥ 0 |

| | | | |
|--------|-------------------|---------------|-----|
| MPC-SV | Schedule Variance | tra -5% e +5% | >0 |
| MPC-CV | Cost Variance | tra -5% e +5% | >0 |
| MPC-BV | Budget Variance | $\pm 5\%$ | <0% |

2.2.2 Sviluppo

2.2.2.1 Progettazione architeturale

Sono state individuate due metriche principali:

1. **SFIN:** *Structure Fan In*;

- **Descrizione:** rappresenta il numero di moduli o componenti che chiamano un modulo o una funzione specifica;
- **Significato:** un fan-in elevato indica che molte parti del sistema dipendono da un particolare modulo. Questo può essere un segno positivo di riusabilità, in quanto il modulo è utilizzato in molte parti del sistema.

2. **SFOUT:** *Structure Fan Out*;

- **Descrizione:** rappresenta il numero di moduli o funzioni chiamati da un modulo o una funzione specifica;
- **Significato:** un fan-out elevato può indicare che un modulo ha molte dipendenze da altri moduli. Questo può portare a una maggiore complessità del sistema, poiché le modifiche in un modulo possono richiedere modifiche in molti altri moduli.

| Metrica | Descrizione | Valore accettazione | Valore ideale |
|----------|-------------------|---------------------|---------------|
| MPC-FIN | Structure Fan In | ? | ? |
| MPC-FOUT | Structure Fan Out | ? | ? |

2.2.2.2 Codifica

- **Complessità Ciclomatica:** quantifica la complessità del codice misurando il numero di percorsi linearmente indipendenti attraverso il grafo di controllo di flusso del programma. Più è alta la complessità ciclomatica, maggiore è la complessità del codice;
- **Numero di violazioni delle regole di codifica:** conta il numero di violazioni delle regole di codifica stabilite per il progetto;
- **Numero di bug per linea di codice:** misura la densità di difetti nel codice, espressa come il numero di *bug_G* riscontrati per ogni 1000 linee di codice (KLOC);
- **Cicli di complessità essenziale:** misura la complessità intrinseca del software, escludendo la complessità dovuta alle scelte di implementazione;

- **Rapporto di churn del codice:** indica la frequenza con cui le linee di codice sono modificate nel tempo, fornendo informazioni sulla stabilità del codice e sulla necessità di frequenti modifiche.

2.3 Processi di Supporto

2.3.1 Documentazione

- **IG:** *Indice Gulpease* metrica utilizzata per valutare la leggibilità di un testo in lingua italiana. L'Indice Gulpease tiene conto di due variabili linguistiche: la lunghezza delle parole e la lunghezza delle frasi. La formula per calcolare l'indice è la seguente:

$$IG = 89 + \frac{300 \cdot \frac{Nf - Nl}{Np}}{Np}$$

Dove:

- **Nf** → indica il numero delle frasi;
- **Nl** → indica il numero delle lettere;
- **Np** → indica il numero delle parole.

L'indice fornisce un punteggio che varia da 0 a 100. Di seguito le possibili interpretazioni:

- 0-29: Testo difficile da leggere;
 - 30-49: Testo leggibile con sforzo;
 - 50-59: Testo abbastanza leggibile;
 - 60-69: Testo leggibile;
 - 70-79: Testo facile da leggere;
 - 80-89: Testo molto facile da leggere;
 - 90-100: Testo estremamente facile da leggere.
- **Correttezza ortografica:** Numero errori grammaticali o ortografici per documento.