

# SWEG

## Software Engineering Group

*swegsoftware@gmail.com*

### Piano di Qualifica

#### Informazioni sul documento

Redattori:	Gabriel R.	Marco B.	Andrea M.	Davide S.
Verificatori:	Andrea M.	Mircea P.	Davide S.	
Approvazione:	Davide M.			
Destinatari:	T. Vardanega	R. Cardin	Zero12	
Uso:	Esterno			
Versione:	1.0.0			

## Registro dei Cambiamenti - Changelog

<i>Versione</i>	<i>Data</i>	<i>Autore</i>	<i>Verificatore</i>	<i>Dettaglio</i>
1.0.0	2023-02-05	Davide Milan	Gabriel Rovesti	Approvazione
0.0.9	2023-01-29	Marco Bernardi	Mircea Plamadeala	Inserimento Grafici
0.0.8	2023-01-15	Andrea Meneghello	Mircea Plamadeala	Correzione sezioni e aggiunta termini glossario
0.0.7	2023-01-10	Gabriel Rovesti	Marco Bernardi	Correzione sezioni e ampliamento sezione 5 e sottosezioni
0.0.6	2022-12-19	Andrea Meneghello	Mircea Plamadeala	Correzione e ampliamento parti sezione 3
0.0.5	2022-12-18	Marco Bernardi	Davide Sgrazzutti	Conclusione sezione 2
0.0.4	2022-12-17	Marco Bernardi	Andrea Meneghello	Scrittura sezione 4
0.0.3	2022-12-16	Davide Sgrazzutti	Andrea Meneghello	Scrittura sezione 3
0.0.2	2022-12-10	Gabriel Rovesti	Milo Spadotto	Scrittura sezioni 1 e 2 e prime bozze sottosezioni
0.0.1	2022-12-07	Gabriel Rovesti	Milo Spadotto	Definizione struttura documento e scheletro sezioni

# Sommario

<b>Sommario</b>	<b>2</b>
Elenco delle immagini	3
Elenco delle tabelle	3
Elenco dei grafici	4
<b>1 Introduzione e scopo</b>	<b>5</b>
1.1 Glossario	5
1.2 Maturità e miglioramenti	5
1.3 Riferimenti	6
1.3.1 Riferimenti normativi	6
1.3.2 Riferimenti informativi	6
<b>2 Qualità di processo</b>	<b>7</b>
2.1 Scopo ed obiettivi	7
2.2 Processi primari	7
2.2.1 Fornitura	7
2.2.2 Sviluppo	8
2.2.2.1 Progettazione architettuale	8
2.2.2.2 Progettazione di dettaglio	9
2.2.2.3 Codifica	9
2.3 Processi di supporto	9
2.3.1 Documentazione	9
2.3.2 Gestione della qualità	10
2.3.3 Verifica	10
<b>3 Qualità di prodotto</b>	<b>11</b>
3.1 Obiettivi	11
3.1.1 Metriche	12
<b>4 Test e specifiche</b>	<b>15</b>
4.1 Tipologie di test	16
4.1.1 Test di Unità	16
4.1.2 Test di Integrazione	16
4.1.3 Test di Sistema	16

4.1.4 Test di Accettazione	16
4.1.5 Test di Regressione	16
<b>A Resoconto delle attività di verifica</b>	<b>18</b>
A.1 Fornitura	18
A.1.1 MPC-AC e MPC-ETC: Actual Cost e Estimated to Completion	18
A.1.2 MPC-EV e MPC-PV: Earned Value e Planned Value	19
A.1.3 MPC-SV: Schedule Variance	19
A.1.4 MPC-CV: Cost Variance	20
A.1.5 MPC-EAC: Estimated at Completion	20
A.2 Documentazione	21
A.2.1 MPC-IG: Indice Gulpease	21
A.2.2 MPC-CO: Correttezza Ortografica	22
<b>B Valutazioni per il miglioramento</b>	<b>22</b>
B.1 Valutazione sull'organizzazione	22
B.2 Valutazione sugli strumenti utilizzati	23
B.3 Valutazione sui ruoli	23

## Elenco delle immagini

Immagine 1: Modello a V

## Elenco delle tabelle

Tabella 1: Metriche di fornitura

Tabella 2: Metriche di sviluppo

Tabella 3: Metriche di progettazione di dettaglio

Tabella 4: Metriche di codifica

Tabella 5: Metriche Gulpease

Tabella 6: Metriche di gestione della qualità

Tabella 7: Metriche di coverage

Tabella 8: Obiettivi di qualità

Tabella 9: Metriche e riferimenti

Tabella 10: Copertura dei requisiti

Tabella 11: Tempo di risposta medio

Tabella 12: Metriche di usabilità

Tabella 13: Metriche di affidabilità

Tabella 14: Metriche di complessità ciclomatica

Tabella 15: Metriche di supporto dei browser

Tabella 16: Valutazione sull'organizzazione

Tabella 17: Valutazione sugli strumenti utilizzati

Tabella 18: Valutazione sui ruoli

## **Elenco dei grafici**

Grafico 1: Metriche MPC-AC e MPC-ETC

Grafico 2: Metriche MPC-EV e MPC-PV

Grafico 3: Metrica MPC-SV

Grafico 4: Metrica MPC-CV

Grafico 5: Metrica MPC-EAC

Grafico 6: Metrica MPC-IG

Grafico 7: Metrica MPC-CO

# 1 Introduzione e scopo

Il Piano di Qualifica è un documento che si valuta di modificare incrementalmente, in particolare per definire le metriche di valutazione del prodotto, che saranno definite conformemente ai requisiti e alle aspettative del proponente, al fine di poter correttamente definire la qualità del prodotto, attraverso un processo di miglioramento continuo e che, per sua natura, tende a diventare incrementale nel corso del tempo e quando viene definita baseline. Per tutti questi motivi, la qualità viene definita da un insieme di processi che cerchino di definire metriche di misurazione di efficacia ed efficienza (misure quantitative che serviranno da valutazione nel corso di realizzazione del progetto didattico).

A livello pratico, il presente documento ha lo scopo di:

- Definire le metriche e le opportune metodologie di controllo e misurazione;
- Definire la quantità e la qualità di test e relative metriche;
- Definire l'applicazione dei test e documentare l'esito ottenuto, definendo se può conformemente alle attese sulla base delle metriche definite.

## 1.1 Glossario

Al fine di evitare incomprensioni relative alla terminologia usata all'interno del documento, viene fornito un Glossario nel file apposito, tale da non avere terminologie ambigue nell'attività progettuale individuata e dandone una definizione precisa. Ogni termine avrà nel documento una lettera G come apice, per meglio evidenziare la loro appartenenza al documento indicato.

## 1.2 Maturità e miglioramenti

Il presente documento è redatto con un approccio incrementale, al fine di poter implementare facilmente cambiamenti nel corso del tempo a seconda di esigenze concordate bilateralmente tra membri del gruppo e proponente. Pertanto, non può essere considerato definitivo e completo in questa versione.

## 1.3 Riferimenti

### 1.3.1 Riferimenti normativi

- Capitolato C4-Piattaforma di localizzazione testi:  
<https://www.math.unipd.it/~tullio/IS-1/2022/Progetto/C4.pdf>
- Norme di Progetto.

### 1.3.2 Riferimenti informativi

- Slide T08 del corso di Ingegneria del Software<sup>G</sup> - Qualità del software (o di prodotto):  
<https://www.math.unipd.it/~tullio/IS-1/2022/Dispense/T08.pdf>
- Slide T09 del corso di Ingegneria del Software - Qualità di processo:  
<https://www.math.unipd.it/~tullio/IS-1/2022/Dispense/T09.pdf>
- Verifica e validazione:  
<https://www.math.unipd.it/~tullio/IS-1/2022/Dispense/T10.pdf>
- ISO<sup>G</sup>/IEC<sup>G</sup> 9126:2001 SWE Product Quality;
- ISO/IEC 14598:1999 SW Product Evaluation;
- ISO/IEC 25000:2005 SQuaRE: Systems and software Quality Requirements and Evaluation:
  - 25010:2011 Quality model;
  - 25020:2019 Quality measurement framework;
  - 25030:2007 Quality requirements;
  - 25040:2011 Quality evaluation.
- ISO 9000:2015;
- ISO 9004:2018;
- ISO/IEC 33020:2019.

## 2 Qualità di processo

### 2.1 Scopo ed obiettivi

La qualità è determinata univocamente dai processi che lo compongono, misurata mettendo in atto delle metriche che permettano di valutare tali processi e accertarsi che raggiungano i corretti obiettivi di qualità previsti. In particolare, si fa riferimento al cosiddetto *Ciclo PDCA (Plan - Do - Check - Act)*, atto a garantire un miglioramento continuo nell'utilizzo dei processi e delle risorse tramite pianificazione, successiva verifica con le metriche previste ed integrazione in fase di produzione. Di seguito, i processi individuati e i livelli di qualità previsti per ciascuno.

In particolare, per ciascuna metrica si opera una breve descrizione, dando un'idea comprensiva dell'attuazione e dei valori considerati accettabili in fase di controllo (check) qualità.

### 2.2 Processi primari

#### 2.2.1 Fornitura

Per questa fase di processo, si individuano tutte le scelte operate in fase di sviluppo per le singole risorse, secondo l'impegno assunto e scandito nelle singole fasi di progetto. In particolare, si individuano le misure da prevedere e attuare per le procedure ripetute nella fase di realizzazione del progetto didattico secondo i termini e condizioni stabiliti.

L'acronimo principale di riferimento è MPC, cioè Minimum Predictive Capability (MPC), metrica utilizzata per valutare la capacità di un modello di previsione o di apprendimento automatico di generare previsioni precise. In altre parole, l'MPC è il valore minimo della precisione delle previsioni che il modello deve raggiungere per essere considerato accettabile.

Di queste, individuiamo ogni termine utile:

- *BAC (Budget At Completion)* - Costo totale di progetto preventivato al completamento;
- *EAC (Estimated At Completion)* - Valore stimato per i compiti attualmente da realizzare (attuale)
  - Il calcolo viene dato dalla divisione di BAC per CPI (Cost Performance Index).
- *ETC (Estimated To Completion)* - Stima del costo finale alla data della misurazione;
- *EV (Earned Value)* - Valore ottenuto fino al momento calcolato (attuale)
  - Il calcolo viene dato dal lavoro svolto in percentuale moltiplicato per EAC.
- *PV (Earned Value)* - Attività lavorativa fino al momento calcolato (attuale)
  - Il calcolo viene dato dal lavoro pianificato in percentuale moltiplicato per BAC.
- *AC (Actual Cost)*: Budget speso fino al momento calcolato;
- *CV (Cost Variance)*: Differenza tra budget utilizzabile e quello usato effettivamente
  - Il calcolo viene dato da  $EV - AC$ .
- *SV (Schedule Variance)* - Varianza (a livello di anticipo/ritardo) rispetto a quanto previsto/schedulato
  - Il calcolo viene dato da  $EV - PV$ ;
  - Se ha valore negativo, si è in ritardo rispetto alle previsioni.
- *BV (Budget Variance)* - Varianza (a livello di bilancio) rispetto al budget preventivato in fase di completamento progettuale
  - Il calcolo viene dato da  $AC - CV$ .



<i><b>Metrica</b></i>	<i><b>Descrizione</b></i>	<i><b>Valore accettabile</b></i>	<i><b>Valore Ottimale</b></i>
MPC-ETC	Estimated to Completion	$\geq 0\%$	$\leq \text{EAC}$
MPC-EAC	Estimated at Completion	Errore del $\pm 5\%$ rispetto al preventivo	Corrispondente al preventivo
MPC-EV	Earned Value	$\geq 0$	$\leq \text{EAC}$
MPC-PV	Planned Value	$\geq 0$	$\leq \text{BAC}$
MPC-AC	Actual Cost	$\geq 0$	$\leq \text{EAC}$
MPC-CV	Cost Variance	$\geq -5\%$	$\geq 0\%$
MPC-SV	Schedule Variance	$\geq -10\%$	$\geq 0\%$
MPC-BV	Budget Variance	$\pm 10\%$	$\leq 0\%$

Tabella 1: Metriche di fornitura

## 2.2.2 Sviluppo

### 2.2.2.1 Progettazione architeturale

#### **SFIN - Structural Fan-In**

Indice di utilità, indica quante componenti utilizzano un modulo specifico.

Un **SFIN** alto indica che il componente viene utilizzato molto.

#### **SFOUT - Structural Fan-Out**

Indice di dipendenza, indica il numero di componenti utilizzate dalla componente in esame.

Un **SFOUT** indica un alto accoppiamento della componente.

<i><b>Metrica</b></i>	<i><b>Descrizione</b></i>	<i><b>Valore accettabile</b></i>	<i><b>Valore Ottimale</b></i>
MPC-SFIN	Structural Fan-in	/	/
MPC-SFOUT	Structural Fan-out	/	/

Tabella 2: Metriche di sviluppo

### 2.2.2.2 Progettazione di dettaglio

#### **NM - Number of Methods**

Indica il numero medio di metodi per package. Un numero eccessivo potrebbe indicare la necessità di refactoring.

<b>Metrica</b>	<b>Descrizione</b>	<b>Valore accettabile</b>	<b>Valore Ottimale</b>
MPC-NM	Number of Methods	3-11	3-8

Tabella 3: Metriche di progettazione di dettaglio

### 2.2.2.3 Codifica

#### **BLC - Bugs for Line of Code**

Indice del numero di righe di codice contenenti *bug* ed errori al proprio interno.

#### **VNUD - Variabili Non Utilizzate e Non Definite**

Indice del numero di variabili non utilizzate e non definite. Esse sono considerate errori di programmazione e possono causare *bug* nel software: le prime allocano memoria inutilmente e rendono il codice meno pulito, le seconde sono dichiarate ma non inizializzate ad un valore noto definito prima di essere utilizzate.

<b>Metrica</b>	<b>Descrizione</b>	<b>Valore accettabile</b>	<b>Valore Ottimale</b>
MPC-BLC	Bugs for Line of Code	0-70	0-25
MPC-VND	Variabili non utilizzate e non definite	0	0

Tabella 4: Metriche di codifica

## 2.3 Processi di supporto

Per questa fase di processo, si individuano tutti i compiti ed attività attuabili per realizzare quanto richiesto a livello di prodotto software. In particolare, si intende tutto questo come processo ampio, atto a raccogliere tutte le attività di *manutenzione*, strutturazione e pianificazione del prodotto, guidando la realizzazione in modo incrementale, atto alle modifiche continuative.

### 2.3.1 Documentazione

#### **IG - Indice Gulpease**

Indice di leggibilità del testo tarato sulla lingua italiana. Considera due variabili linguistiche: la lunghezza della parola e la lunghezza della frase rispetto al numero delle lettere.

$$IG = 89 + \frac{300 * Nf - 10 * Nl}{Np}$$

Dove:

- $Nf$ : numero di frasi;
- $Nl$ : numero di lettere;
- $Np$ : numero di parole.

In generale, **IG**:

- < 80 : difficili da leggere per chi ha la licenza elementare;
- < 60 : difficili da leggere per chi ha la licenza media;
- < 40 : difficili da leggere per chi ha un diploma superiore.

### **CO - Correttezza Ortografica**

Numero di errori grammaticali o ortografici per documento.

<b>Metrica</b>	<b>Descrizione</b>	<b>Valore accettabile</b>	<b>Valore Ottimale</b>
MPC-IG	Indice Gulpease	40-100	60-100
MPC-CO	Correttezza Ortografica	0	0

Tabella 5: Metriche Gulpease

## **2.3.2 Gestione della qualità**

### **QMS - Quality Metrics Satisfied**

Percentuale di metriche di qualità soddisfatte.

$$QMS = \frac{NQMS}{TQM} \cdot 100$$

Dove:

- NQMS (Number of Quality Metrics Satisfied): numero di metriche di qualità soddisfatte;
- TMQ (Total number of Quality Metrics): numero di metriche di qualità totali.

<b>Metrica</b>	<b>Descrizione</b>	<b>Valore accettabile</b>	<b>Valore Ottimale</b>
MPC-QMS	Quality Metrics Satisfied	≥90%	100%

Tabella 6: Metriche di gestione della qualità

## **2.3.3 Verifica**

### **CC - Code Coverage**

Misura della quantità di codice di un programma che viene eseguita durante un test.

Il code coverage viene utilizzato per valutare la qualità dei *test* e per verificare che il codice sia stato adeguatamente testato. Un alto livello di code coverage è solitamente considerato un segno di un buon set di test, poiché indica che il codice è stato eseguito in molti modi diversi e che le diverse parti del programma sono state adeguatamente testate.

### SC - Statement Coverage

Tecnica di test di tipo *white box*, prevede l'esecuzione di tutte le istruzioni presenti nel codice sorgente almeno una volta.

Questa *metrica* viene utilizzata per calcolare il numero di istruzioni eseguite almeno una volta.

### BC - Branch Coverage

Indice di quante diramazioni del codice vengono eseguite dai test. Un "ramo" è uno dei possibili percorsi di esecuzione che il codice può seguire dopo che un'istruzione decisionale (es. *if*) viene valutata.

### DCC - Decision/Condition Coverage

Il Decision/Condition Coverage è un criterio di copertura del codice utilizzato nei test del software.

<i>Metrica</i>	<i>Descrizione</i>	<i>Valore accettabile</i>	<i>Valore Ottimale</i>
MPC-CC	Code Coverage	>70%	>90%
MPC-SC	Statement Coverage	>70%	>85%
MPC-BC	Branch Coverage	>50%	>75%
MPC-DCC	Decision/Condition Coverage	-	-

Tabella 7: Metriche di coverage

## 3 Qualità di prodotto

Dopo un'attenta analisi per individuare le proprietà utili per la gestione del ciclo di vita del software si è cercato di trovare quali caratteristiche siano necessarie per la realizzazione di un prodotto di qualità.

L'acronimo principale di riferimento è MPD, cioè Mean Percentage Difference (MPD), metrica utilizzata per valutare la differenza percentuale media tra due valori. Ad esempio, può essere utilizzata per calcolare la differenza percentuale media tra i valori di previsione e i valori effettivi di una variabile in un modello di previsione o di apprendimento automatico.

In altre parole, l'MPD indica la percentuale media di errore tra le previsioni del modello e i valori effettivi della variabile in questione. Un valore di MPD inferiore indica che le previsioni del modello sono più precise, mentre un valore più elevato indica una maggiore imprecisione.

### 3.1 Obiettivi

<i>Obiettivo</i>	<i>Descrizione</i>	<i>Metrica</i>
Funzionalità	la capacità di un prodotto software di fornire funzioni che soddisfino esigenze stabilite nei requisiti descritti all'interno dell'Analisi dei Requisiti V.1.0.0.	MPD-CR

Efficienza	la capacità di fornire appropriate prestazioni relativamente alla quantità di risorse usate.	MPD-TM
Usabilità	Creazione di un prodotto che abbia un apprendimento, una comprensione e un utilizzo che sia semplice ed intuitivo, alla portata di ogni utente.	MPD-TA MPD-RO MPD-EU
Affidabilità	la capacità del prodotto software di essere sempre disponibile e tollerante agli errori quando usato in date condizioni per un dato periodo.	MPD-FD
Manutenibilità	la capacità del software di essere modificato, includendo correzioni, miglioramenti o adattamenti.	MPD-CC
Portabilità	la capacità del software di essere trasportato da un ambiente di sviluppo ad un altro.	MPD-VS

Tabella 8: Obiettivi di qualità

### 3.1.1 Metriche

<i><b>Metrica</b></i>	<i><b>Nome</b></i>	<i><b>Valore Accettabile</b></i>	<i><b>Valore Ottimale</b></i>
MPD-CR	Copertura dei requisiti	100%	100%
MPD-TM	Tempo di risposta medio	3 secondi	2 secondi
MPD-TA	Tempo apprendimento	10 minuti	5 minuti
MPD-RO	Raggiunta dell'obiettivo	7 click	5 click
MPD-EU	Errori dell'utente	2	0
MPD-FD	Failure Density	100%	100%
MPD-CC	Complessità Ciclomantica	25-35%	30-45%
MPD-VS	Versioni Supportate	100%	100%

Tabella 9: Metriche e riferimenti

## 3.2 Funzionalità

### 3.2.1 Metriche

#### CR - Copertura dei requisiti

Viene calcolata la percentuale dei requisiti soddisfatti:

$$RC_{obb} = \frac{Nos}{Not} \cdot 100$$

$N_{os}$  : numero requisiti obbligatori soddisfatti;

$N$  : numero requisiti obbligatori totali.

<i><b>Metrica</b></i>	<i><b>Nome</b></i>	<i><b>Valore Accettabile</b></i>	<i><b>Valore Ottimale</b></i>
MPD-CR	Copertura dei requisiti	100%	100%

Tabella 10: Copertura dei requisiti

### 3.3 Efficienza

#### 3.3.1 Metriche

##### TM - Tempo di risposta medio

Il tempo impiegato dal software dalla gestione ed elaborazione di una richiesta fino al risultato finale fornito.

<i><b>Metrica</b></i>	<i><b>Nome</b></i>	<i><b>Valore Accettabile</b></i>	<i><b>Valore Ottimale</b></i>
MPD-TM	Tempo di risposta medio	3 secondi	2 secondi

Tabella 11: Tempo di risposta medio

### 3.4 Usabilità

#### 3.4.1 Metriche

##### TA - Tempo Apprendimento

Tempo necessario all'utente per imparare ad utilizzare tutte le funzionalità del software.

##### RO - Raggiunta dell'obiettivo

Interazioni necessarie all'utente per raggiungere il risultato voluto

## EU - Errori dell'utente

Interazioni errate che l'utente compie prima di raggiungere il risultato voluto

<i><b>Metrica</b></i>	<i><b>Nome</b></i>	<i><b>Valore Accettabile</b></i>	<i><b>Valore Ottimale</b></i>
MPD-TA	Tempo apprendimento	10 minuti	5 minuti
MPD-RO	Raggiunta dell'obiettivo	7 click	5 click
MPD-EU	Errori dell'utente	2	0

Tabella 12: Metriche di usabilità

## 3.5 Affidabilità

### 3.5.1 Metriche

#### FD - Failure Density

Viene calcolata in percentuale l'affidabilità di un prodotto software:

$$FD = \frac{T_f}{T_e} \cdot 100$$

$T_f$  : numero di test falliti;

$T_e$  : numero di test eseguiti.

<i><b>Metrica</b></i>	<i><b>Nome</b></i>	<i><b>Valore Accettabile</b></i>	<i><b>Valore Ottimale</b></i>
MPD-FD	Failure Density	100%	100%

Tabella 13: Metriche di affidabilità

## 3.6 Manutenibilità

### 3.6.1 Metriche

#### CC - Complessità Ciclomantica

La complessità ciclomantica viene calcolata utilizzando il grafo di controllo di flusso tramite la formula

$$v(G) = e - n + p$$

e: numero degli archi in G;

n: numero dei nodi in G;

p: numero delle componenti connesse da ogni arco.

<i>Metrica</i>	<i>Nome</i>	<i>Valore Accettabile</i>	<i>Valore Ottimale</i>
MPD-CC	Complessità Ciclomatica	0-30	0-10

Tabella 14: Metriche di complessità ciclomatica

### 3.7 Portabilità

#### 3.7.1 Metriche

##### BS - Browser supportati

Viene calcolata la percentuale delle versioni del browser supportate tramite la formula:

$$VS = \frac{Bs}{Bp} \cdot 100$$

$B_s$  : versioni di browser supportate;

$B_p$  : versioni di browser stabilite da supportare.

<i>Metrica</i>	<i>Nome</i>	<i>Valore Accettabile</i>	<i>Valore Ottimale</i>
MPD-BS	Browser supportati	100%	100%

Tabella 15: Metriche di supporto dei browser

## 4 Test e specifiche

Nella seguente sezione verranno espresse in maniera dettagliata le varie metodologie di test, gli obiettivi del testing e i criteri di successo utilizzati durante lo sviluppo del *prodotto*.

Il gruppo SWAG per perseguire la correttezza del *prodotto* e facilitare la *fase di validazione*, svolgerà la verifica in parallelo allo sviluppo (*Modello a V*<sup>G</sup>).

I test dovranno essere resi il più automatici possibile, per evitare che la fase di testing rallenti la produzione.

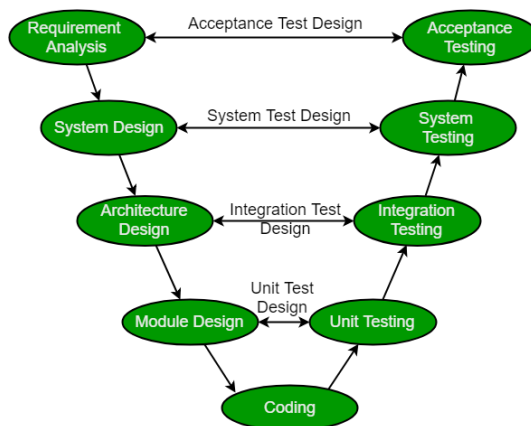


Immagine 1: Modello a V



## 4.1 Tipologie di test

### 4.1.1 Test di Unità

I test di *unità* sono un tipo di test che viene utilizzato per verificare il funzionamento di una singola unità di codice all'interno di un software.

Una unità di codice può essere una funzione, una classe o qualsiasi altra porzione di codice che svolge una specifica attività all'interno del software.

In questa prima versione del *Piano di Qualifica* non è ancora definita l'unità, e di conseguenza anche i test.

L'unità verrà definita con l'inizio del processo di progettazione e sviluppo software.

### 4.1.2 Test di Integrazione

I test di integrazione sono un tipo di test che viene utilizzato per verificare il funzionamento delle diverse componenti di un software quando vengono integrate tra loro e sono particolarmente utili per identificare e risolvere eventuali problemi di integrazione.

Inoltre, i test di integrazione possono essere utilizzati per verificare che il software soddisfi i requisiti prestabiliti in modo completo e che sia pronto per essere messo in produzione.

In questa prima versione del *Piano di Qualifica* non sono ancora definiti i test di integrazione poiché non sono ancora state individuate le varie componenti del *prodotto*.

### 4.1.3 Test di Sistema

I test di sistema vengono utilizzati per verificare il funzionamento del software come sistema completo, inclusi tutti i componenti e le interfacce con gli altri sistemi. I test di sistema hanno lo scopo di verificare che il software soddisfi i requisiti prestabiliti e che sia pronto per essere messo in produzione.

### 4.1.4 Test di Accettazione

I test di accettazione sono un tipo di test che viene utilizzato per verificare che il software soddisfi i requisiti prestabiliti dal *capitolato* e che sia pronto per essere consegnato al *committente* o messo in produzione.

Vengono svolti alla presenza del *committente*.

### 4.1.5 Test di Regressione

I test di regressione sono un tipo di test che viene utilizzato per verificare che le modifiche apportate ad un software non influiscano negativamente sulle sue funzionalità esistenti, sono particolarmente utili per garantire che il software continui a funzionare correttamente anche dopo aver apportato modifiche o aggiornamenti.

Consistono nella ripetizione selettiva di test di unità, integrazione e sistema.

## 5 Resoconto test

# A Resoconto delle attività di verifica

## A.1 Fornitura

### A.1.1 MPC-AC e MPC-ETC: Actual Cost e Estimated to Completion

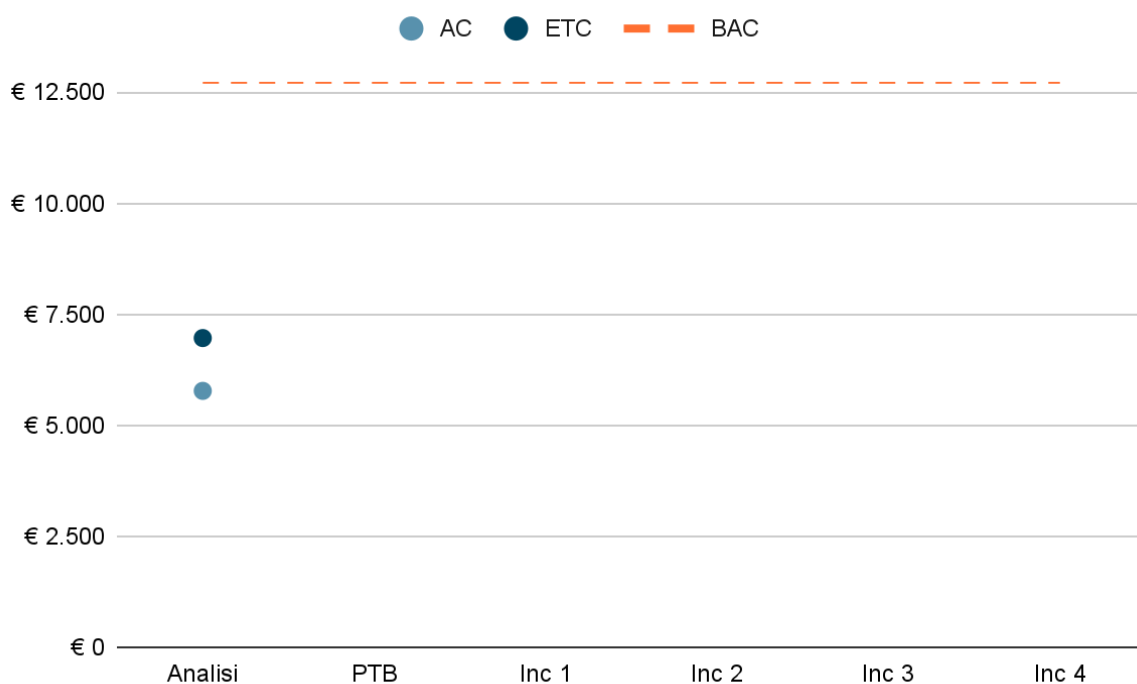


Grafico 1: Metriche MPC-AC e MPC-ETC

### A.1.2 MPC-EV e MPC-PV: Earned Value e Planned Value

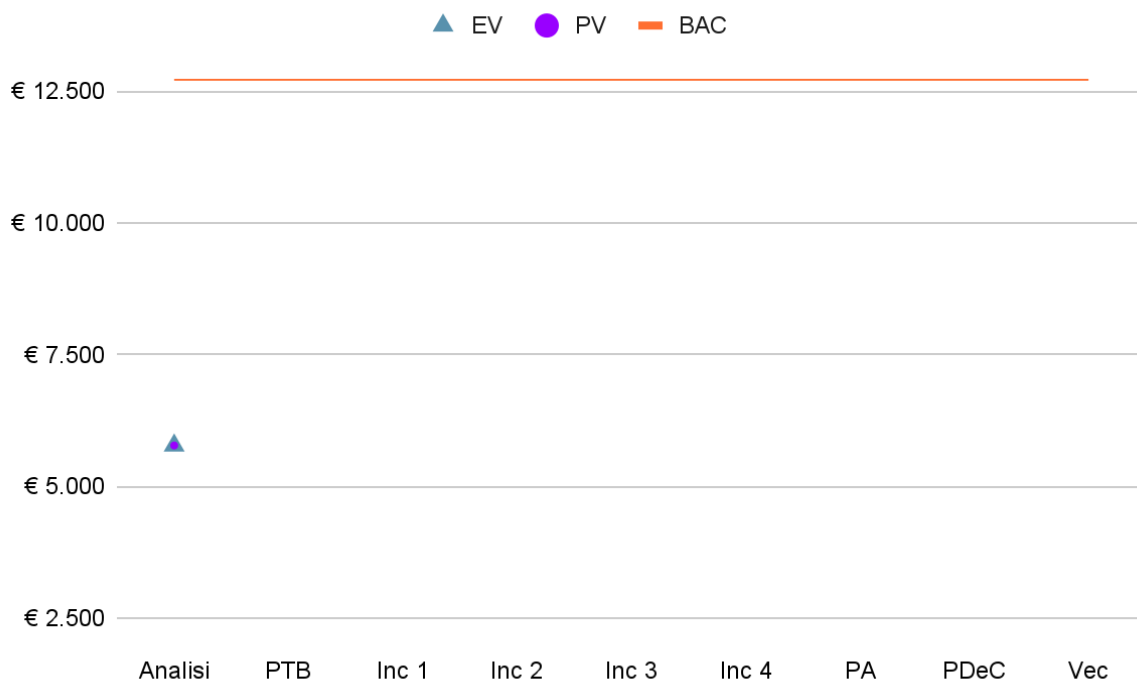


Grafico 2: Metriche MPC-EV e MPC-PV

### A.1.3 MPC-SV: Schedule Variance

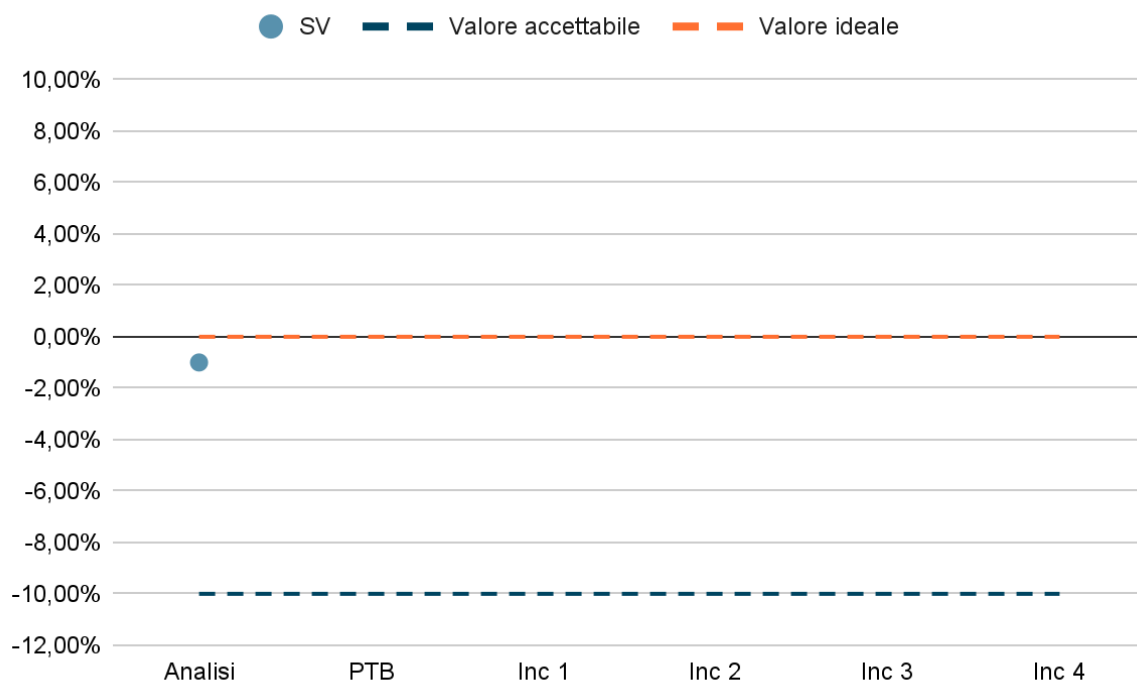


Grafico 3: Metrica MPC-SV

### A.1.4 MPC-CV: Cost Variance

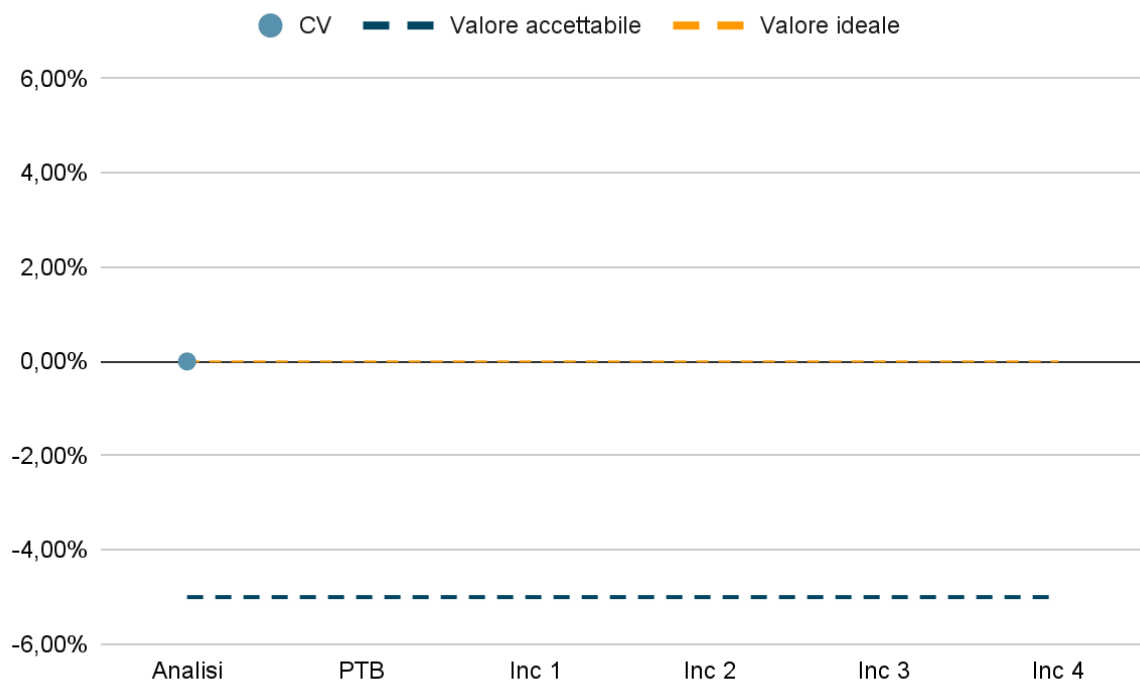


Grafico 4: Metrica MPC-CV

### A.1.5 MPC-EAC: Estimated at Completion

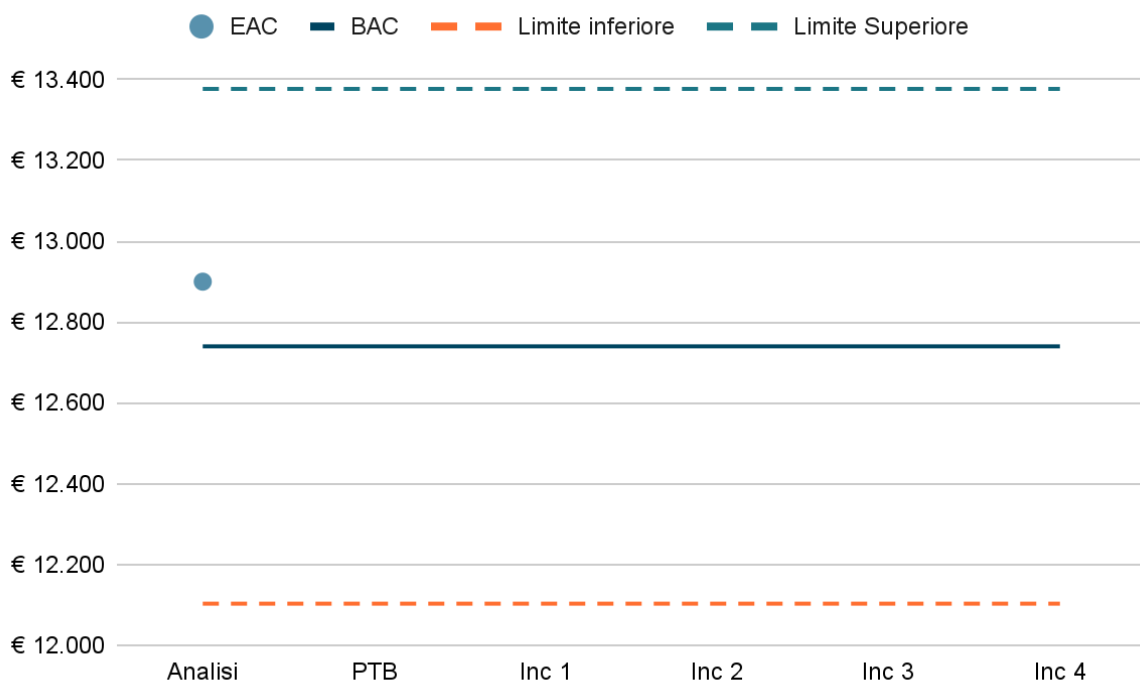


Grafico 5: Metrica MPC-EAC

## A.2 Documentazione

### A.2.1 MPC-IG: Indice Gulpease

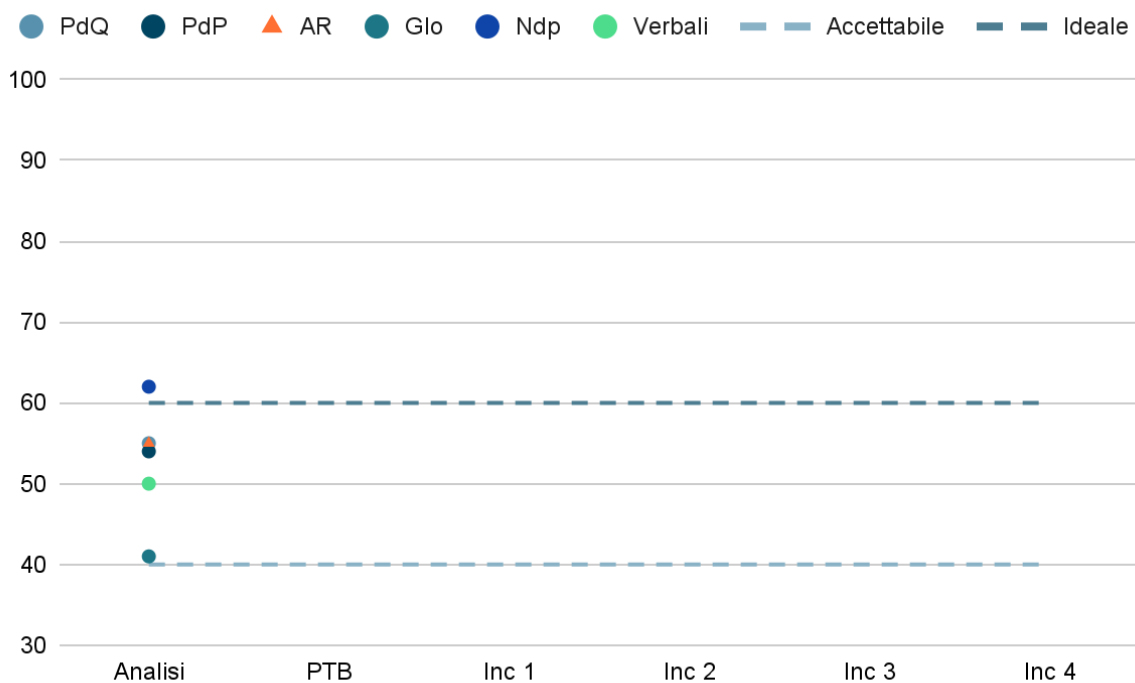


Grafico 6: Metrica MPC-IG

## A.2.2 MPC-CO: Correttezza Ortografica

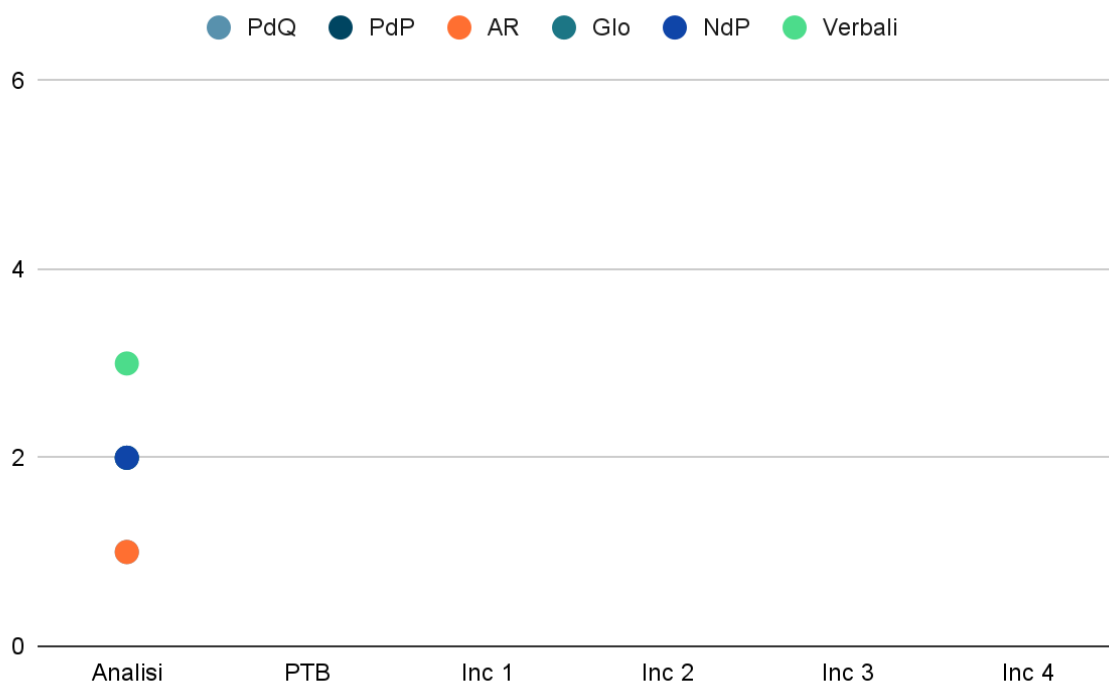


Grafico 7: Metrica MPC-CO

## B Valutazioni per il miglioramento

Nella seguente sezione viene riportata la valutazione sul lavoro fino alla data di modifica del documento. Questa ha lo scopo di inserire osservazioni sulle criticità presenti e sulle possibili correzioni da adottare come miglioramenti.

### B.1 Valutazione sull'organizzazione

<i>Criticità</i>	<i>Descrizione</i>	<i>Gravità</i>	<i>Soluzione</i>
Suddivisione dei compiti	Il gruppo ha incontrato svariate difficoltà nell'organizzazione dei ruoli e nel garantire il parallelismo di tutte le operazioni, evitando membri inoccupati e formazione di sottogruppi	Media	Il gruppo ha deciso di convergere le proprie forze per fare in modo di realizzare le attività con maggiore coesione con gli altri membri del gruppo
Verifica	Nelle fasi del progetto e nella successiva scrittura dei documenti, non è stata posta abbastanza attenzione su questo punto, richiedendo	Media	Il gruppo ha deciso di svolgere i dovuti controlli ai documenti ad ogni termine dei periodi di sprint per avere una visione più chiara sullo sviluppo

	reiterazioni e maggiore controllo		
Rotazione dei ruoli	Il gruppo ha incontrato svariate difficoltà nell'organizzazione dei ruoli e nel garantire la corretta riorganizzazione delle singole mansioni	Media	Il Responsabile ha posto un controllo più granulare sulle attività di gruppo, al fine di massimizzare quanto prodotto ed evitare dispersioni di sorta

Tabella 16: Valutazione sull'organizzazione

## B.2 Valutazione sugli strumenti utilizzati

<i>Criticità</i>	<i>Descrizione</i>	<i>Gravità</i>	<i>Soluzione</i>
Scarsa conoscenza delle tecnologie	Il gruppo ha incontrato svariate difficoltà nell'organizzazione dei ruoli e nella composizione asincrona dei compiti	Media	Il gruppo ha deciso, in accordo col Responsabile, di garantire un maggiore controllo sul breve periodo
Verificatore	Nelle fasi del progetto, non è stato svolto con abbastanza serietà tale ruolo, risultando in reiterazioni frequenti	Bassa	Si è deciso di porre maggiore attenzione, incrementando il numero e basando la verifica su attività precise e determinate

Tabella 17: Valutazione sugli strumenti utilizzati

## B.3 Valutazione sui ruoli

<i>Criticità</i>	<i>Descrizione</i>	<i>Gravità</i>	<i>Soluzione</i>
Responsabile	Il gruppo ha incontrato svariate difficoltà nell'organizzazione dei ruoli e nella composizione asincrona dei compiti	Media	Il gruppo ha deciso, in accordo col Responsabile, di garantire un maggiore controllo sul breve periodo
Verificatore	Nelle fasi del progetto, non è stato svolto con abbastanza serietà tale ruolo, risultando in reiterazioni frequenti	Bassa	Si è deciso di porre maggiore attenzione, incrementando il numero e basando la verifica su attività precise e determinate

Tabella 18: Valutazione sui ruoli