



Università degli Studi di Padova

Laurea in Informatica

Corso di Ingegneria del Software

Anno Accademico 2023/2024



Gruppo: SWEet16

Email: sweet16.unipd@gmail.com

Piano di Qualifica

Redattori:	Alberto C., Alberto M., Alex S., Bilal El M., Giovanni Z.	
Verificatori:	Alberto M., Alex S., Bilal El M.	
Amministratore:	Alberto M.	
Destinatari:	T. Vardanega	R. Cardin
Versione:	3.0.0	

Registro delle modifiche

Versione	Data	Autore	Verificatore	Descrizione
3.0.0	2024/06/21	Alex S.		Approvazione per il rilascio
2.1.0	2024/06/23	Alex S.	Bilal El M.	Stesura test di unità
2.0.0	2024/06/21	Alex S.		Approvazione per il rilascio
1.2.1	2024/04/10	Alberto M.	Alex S.	Stesura dei test
1.2.0	2024/06/21	Alberto C.	Alex S.	Correzione grafici
1.1.0	2024/06/21	Alex S.	Bilal El M.	Aggiornamento resoconto
1.0.0	2024/04/15	Alex S.		Approvazione per il rilascio
0.6.1	2024/04/10	Alberto C.	Alex S.	Modifica grafici resoconto attività di verifica
0.6.0	2024/04/08	Giovanni Z.	Alberto M.	Stesura qualità di processo
0.5.0	2024/04/02	Alberto C.	Alex S.	Stesura resoconto delle attività di verifica
0.4.0	2024/03/25	Giovanni Z.	Alex S.	Stesura qualità di prodotto
0.3.0	2024/03/22	Alberto C.	Alberto M.	Stesura test
0.2.0	2024/03/19	Bilal El M.	Alberto M.	Stesura introduzione
0.1.0	2024/02/16	Alex S.	Alberto M.	Stesura scheletro

Indice

1	Introduzione	5
1.1	Scopo del documento	5
1.2	Glossario	5
1.3	Riferimenti	5
1.3.1	Riferimenti normativi	5
1.3.2	Riferimenti informativi	5
2	Qualità di processo	6
2.1	Scopo	6
2.2	Processi primari	6
2.2.1	Fornitura	6
2.2.2	Sviluppo	7
2.2.2.1	Progettazione architettuale	7
2.2.2.2	Progettazione di dettaglio	8
2.2.2.3	Codifica	8
2.3	Processi di supporto	8
2.3.1	Documentazione	8
2.3.2	Gestione delle qualità	9
2.3.2.1	Metriche	9
2.3.3	Verifica	9
3	Qualità di prodotto	10
3.1	Scopo	10
3.2	Usabilità	10
3.2.1	Obiettivi	10
3.3	Manutenibilità	10
3.4	Affidabilità	11
3.5	Efficienza	12
3.6	Funzionalità	12

3.7	Compatibilità	12
4	Test	13
4.1	Tipologie di test	13
4.1.1	Test di unità	13
4.1.2	Test di integrazione	14
4.1.3	Test di sistema	14
4.1.4	Test di accettazione	14
4.1.5	Test di regressione	14
4.2	Specifica dei test	14
4.2.1	Test di unità	15
5	Resoconto delle attività di verifica	17
5.1	Fornitura	17
5.1.1	MPC-AC e MPC-ETC: Actual Cost e Estimated to Completion	17
5.1.2	MPC-EV e MPC-PV: Earned Value e Planned Value	18
5.1.3	MPC-SV: Schedule Variance	18
5.1.4	MPC-CV: Cost Variance	19
5.1.5	MPC-EAC: Estimated at Completion	19
5.2	Documentazione	20
5.2.1	MPC-IG: Indice Gulpease	20
5.2.2	MPC-CO: Correttezza Ortografica	20

1 Introduzione

1.1 Scopo del documento

Lo scopo di questo documento è stabilire standard e obiettivi per valutare la qualità dei processi e dei prodotti lungo l'intero ciclo di progetto.

L'obiettivo è stabilire la qualità del prodotto in modo accurato, attraverso un processo di miglioramento continuo che evolve nel tempo, soprattutto quando viene stabilita una linea di base. La qualità è definita da una serie di processi che mirano a definire metriche per valutare l'efficacia ed efficienza.

Queste misure quantitative saranno utilizzate per valutare il progresso del progetto.

In pratica questo documento si propone di:

- Stabilire metriche e procedure di controllo e misurazione appropriate;
- Definire la quantità e la qualità dei test e le relative metriche;
- Applicare i test e documentare i risultati ottenuti, verificando se corrispondono alle aspettative basate sulle metriche definite.

1.2 Glossario

Al fine di evitare possibili ambiguità o incomprensioni riguardanti la terminologia usata nel documento, è stato deciso di adottare un glossario in cui vengono riportate le varie definizioni. In questa maniera in esso verranno posti tutti i termini specifici del dominio d'uso con relativi significati.

La presenza di un termine all'interno del glossario viene indicata applicando una " ^G " ad apice della parola.

1.3 Riferimenti

1.3.1 Riferimenti normativi

- *Capitolato d'appalto*^G C3 - Easy Meal:
<https://www.math.unipd.it/~tullio/IS-1/2023/Progetto/C3.pdf>.

1.3.2 Riferimenti informativi

- Qualità del software:
<https://www.math.unipd.it/~tullio/IS-1/2023/Dispense/T7.pdf>;
- Qualità di processo:
<https://www.math.unipd.it/~tullio/IS-1/2023/Dispense/T8.pdf>;
- Verifica e validazione:
<https://www.math.unipd.it/~tullio/IS-1/2023/Dispense/T9.pdf>;
- Verifica e validazione, analisi statica:
<https://www.math.unipd.it/~tullio/IS-1/2023/Dispense/T10.pdf>;
- Verifica e validazione, analisi dinamica (aka testing):
<https://www.math.unipd.it/~tullio/IS-1/2023/Dispense/T11.pdf>;

- ISO/IEC 9126:2001:
<https://www.iso.org/standard/35733.html>;
- Indice di Gulpease:
https://it.wikipedia.org/wiki/Indice_Gulpease;
- Glossario:
<https://github.com/SWEet16-SWE-Group/docs/blob/main/RTB/Documentazione%20Esterna/Glossario.pdf>.

Tutti i riferimenti (normativi e informativi) a risorse web soggette a variazione sono stati consultati il 2024/04/03.

2 Qualità di processo

2.1 Scopo

La qualità di un prodotto è influenzata dalla qualità dei processi che lo compongono.

È quindi necessario dotarsi di metriche che permettano di valutare tali processi e garantire che essi raggiungano gli obiettivi di qualità fissati.

Per garantire una corretta implementazione ed un mantenimento costante, si seguirà il *ciclo di Deming*, meglio conosciuto come PDCA, che prevede un approccio iterativo funzionale all'attuazione di un miglioramento continuo.

In questa sezione si espongono le metriche scelte ed i livelli di qualità accettabili e ottimali per ciascuna di esse.

Le metriche hanno un identificativo avente come prefisso l'acronimo QPC (Qualità Processo) seguito dal codice della singola metrica.

2.2 Processi primari

2.2.1 Fornitura

- **Budget at Completion (QPC-BAC):** Totale preventivato del progetto.
- **QPC-AC - Actual Cost:** Costo sostenuto per il progetto al momento del calcolo;
- **QPC-ETC - Estimated to Completion:** Stima del valore per la realizzazione delle rimanenti attività;
- **QPC-EAC - Estimated at Completion:** Costo finale stimato alla data della misurazione, revisione del **QPC-BAC**;

$$\text{Formula} : \text{QPC-AC} + \text{QPC-ETC}$$

- **QPC-EV - Earned Value:** Importo guadagnato per il lavoro svolto al momento del calcolo;

$$\text{Formula} : (\% \text{Lavoro svolto}) \cdot \text{QPC} - \text{EAC}$$

- **QPC-PV - Planned Value:** Importo pianificato in base al lavoro svolto, al momento del calcolo;

$$\text{Formula} : (\% \text{Lavoro pianificato}) \cdot \text{QPC-BAC}$$

- **QPC-SV - Schedule Variance:** Stato (anticipo/ritardo) della pianificazione. Un valore negativo indica che si è in ritardo rispetto alla pianificazione;

$$\text{Formula base : } \text{QPC-EV} - \text{QPC-PV}$$

$$\text{Formula in \% : } \text{QPC-EV} / \text{QPC-PV} \cdot 100$$

- **QPC-CV - Cost Variance:** Differenza tra il budget a disposizione e quello effettivamente utilizzato. Un valore negativo indica che si sta lavorando in perdita.

$$\text{Formula : } \text{QPC-EV} - \text{QPC-AC}$$

$$\text{Formula in \% : } \text{QPC-EV} / \text{QPC-AC} \cdot 100$$

Metrica	Descrizione	Valore accettabile	Valore ideale
QPC-AC	Actual Cost		
QPC-ETC	Estimated to Completion	$\geq 0\%$	$\leq \text{QPC-EAC}$
QPC-EAC	Estimated at Completion	Errore del $\pm 3\%$ rispetto a QPC-BAC	$= \text{QPC-BAC}$
QPC-EV	Earned Value	≥ 0	$\leq \text{QPC-EAC}$
QPC-PV	Planned Value	≥ 0	$\leq \text{QPC-BAC}$
QPC-SV	Schedule Variance	$\geq -10\%$	≥ 0
QPC-CV	Cost Variance	$\geq -5\%$	≥ 0

Tabella 1: Metriche e obiettivi fornitura

2.2.2 Sviluppo

2.2.2.1 Progettazione architettuale

- **QPC-SFIN - Structural Fan-in:** Indice di utilità, indica quante componenti utilizzano un determinato modulo. Un valore alto indica che il componente è molto usato;
- **QPC-SFOUT - Structural Fan-out:** Indice di dipendenza, indica quante componenti sono utilizzate dalla componente in esame. Un valore elevato indica che quest'ultima utilizza molte componenti esterne ad essa.

Metrica	Descrizione	Valore accettabile	Valore ideale
QPC-SFIN	Structural Fan-in	-	-
QPC-SFOUT	Structural Fan-out	-	-

Tabella 2: Metriche e obiettivi progettazione architettuale

2.2.2.2 Progettazione di dettaglio

- **QPC-NM - Number of Methods:** Indica il numero medio di metodi per package. Un numero eccessivo potrebbe indicare la necessità di refactoring.

Metrica	Descrizione	Valore accettabile	Valore ideale
QPC-NM	Number of Methods	3-11	3-8

Tabella 3: Metriche e obiettivi progettazione di dettaglio

2.2.2.3 Codifica

- **QPC-BLC - Bugs for Line of Code:** Indice del numero di righe di codice contenenti bug ed errori al proprio interno;
- **QPC-VNU - Variabili Non Utilizzate:** Indice di un errore di programmazione, le variabili non utilizzate sporcano il codice e fanno allocare memoria inutilmente;
- **QPC-VND - Variabili Non Definite:** Fonte comune di bug nel software, sono variabili dichiarate ma non inizializzate ad un valore noto definito prima di essere utilizzate.

Metrica	Descrizione	Valore accettabile	Valore ideale
QPC-BLC	Bugs for Line of Code	0-70	0-25
QPC-VNU	Variabili Non Utilizzate	0	0
QPC-VND	Variabili Non Definite	0	0

Tabella 4: Metriche e obiettivi codifica

2.3 Processi di supporto

2.3.1 Documentazione

QPR-DOC: Indice di Gulpease

L'Indice di Gulpease è un indice di leggibilità di un testo tarato sulla lingua italiana. Si basa sulla seguente formula:

$$GULPEASE = 89 + \frac{(NF \cdot 300) - (10 \cdot NL)}{NP}$$

- **NF:** Numero frasi;
- **NL:** Numero lettere;
- **NP:** Numero parole.

In generale risulta la seguente suddivisione:

- $GULPEASE \leq 80$: Difficile da leggere per chi ha un licenza elementare;

- $GULPEASE \leq 60$: Difficile da leggere per chi ha un licenza media;
- $GULPEASE \leq 40$: Difficile da leggere per chi ha un diploma superiore.

Metrica	Descrizione	Valore accettabile	Valore ideale
QPR-DOC	Indice di Gulpease	$GULPEASE \geq 40$	$GULPEASE \geq 60$

Tabella 5: Metriche Documentazione

2.3.2 Gestione delle qualità

2.3.2.1 Metriche

- **QPC-QMS: Quality Metrics Satisfied:** Percentuale di metriche di qualità soddisfatte.

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

Metrica	Descrizione	Valore accettabile	Valore ideale
QPC-QMS	Quality Metrics Satisfied	$\geq 90\%$	100%

Tabella 6: Metriche e obiettivi gestione della qualità

2.3.3 Verifica

- **QPC-CC - Code Coverage:** Definisce la misura della quantità di codice di un programma che viene eseguita durante uno specifico test.
Una percentuale alta indica che il codice è stato testato in modo approfondito nelle sue diverse parti e quindi vi è una minore probabilità che ci siano bug;
- **QPC-SC - Statement Coverage:** Metrica utilizzata per calcolare il numero di istruzioni eseguite almeno una volta.
Utilizza una tecnica di test chiamata white box che prevede l'esecuzione di tutte le istruzioni presenti nel codice sorgente almeno una volta;
- **QPC-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.

Metrica	Descrizione	Valore accettabile	Valore ideale
QPC-CC	Code Coverage	$\geq 80\%$	90%
QPC-SC	Statement Coverage	$\geq 70\%$	85%
QPC-BC	Branch Coverage	$\geq 50\%$	75%

Tabella 7: Metriche e obiettivi verifica

3 Qualità di prodotto

3.1 Scopo

Facendo riferimento allo standard ISO/IEC 9126:2001, vengono di seguito riportate le caratteristiche che il prodotto deve avere per essere considerato di qualità.

Vengono inoltre riportate le relative metriche atte a definire un metodo di valutazione del prodotto finale.

Le metriche hanno un identificativo avente come prefisso l'acronimo QPR (Qualità Prodotto) seguito dal codice della singola metrica.

3.2 Usabilità

- **QPR-TA - Tempo Apprendimento:** Tempo necessario all'utente per apprendere l'utilizzo del prodotto;
- **QPR-NP - Numero Passi:** Numero di passi necessari per raggiungere lo scopo voluto;
- **QPR-TE - Tempo Esplorazione:** Tempo speso nell'esplorazione del prodotto;
- **QPR-NE - Numero Errori:** Numero di errori commessi dall'utente prima di raggiungere lo scopo voluto.

3.2.1 Obiettivi

Metrica	Descrizione	Valore accettabile	Valore ideale
QPR-TA	Tempo Apprendimento	10 minuti	5 minuti
QPR-NP	Numero Passi	20 click	10 click
QPR-TE	Tempo Esplorazione	5 minuti	3 minuti
QPR-NE	Numero Errori	5	0

Tabella 8: Metriche usabilità

3.3 Manutenibilità

- **QPR-CC - Complessità Ciclomatica:** Valutazione della complessità di un algoritmo calcolata utilizzando il grafo di controllo del flusso tramite la formula

$$v(G) = L - N + P$$

- **V(G):** Numero ciclomatico relativo al grafo G;
 - **L:** Numero di archi nel grafo;
 - **N:** Numero di nodi del grafo;
 - **P:** Numero dei componenti del grafo disconnessi.
- **QPR-NPM - Numero parametri per metodo:** Numero di parametri passati ai metodi. Un valore troppo grande può indicare un metodo troppo complesso;

- **QPR-FCC - Facilità di comprensione del codice:** Un codice comprensibile permette una manutenibilità e gestione migliore. Viene misurata con la seguente formula

$$R = \frac{NR_{com}}{NR_{tot}}$$

che indica il rapporto tra le righe di commenti (NR_{com}) e le righe di codice (NR_{tot}).

Metrica	Descrizione	Valore accettabile	Valore ideale
QPR-CC	Complessità Ciclomatica	≤ 25	≤ 10
QPR-NPM	Numero parametri per metodo	≤ 8	≤ 4
QPR-FCC	Facilità di comprensione del codice	≥ 0.10	≥ 0.20

Tabella 9: Metriche Manutenibilità

3.4 Affidabilità

- **QPR-AFD - Failure Density:** Indica l'affidabilità del software. Si ricava dal rapporto tra i test falliti e quelli eseguiti.

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

- T_f : Test falliti;
- T_t : Test totali.

- **QPR-ACC - Code Coverage:** Indica la percentuale di codice eseguito durante i test.

Metrica	Descrizione	Valore accettabile	Valore ideale
QPR-AFD	Failure Density	90%	100%
QPR-ACC	Code Coverage	80%	100%

Tabella 10: Metriche Affidabilità

3.5 Efficienza

- **QPR-TMR - Tempo di Risposta Medio:** Il tempo impiegato dal software dalla gestione ed elaborazione di una richiesta fino al risultato finale fornito.

Metrica	Descrizione	Valore accettabile	Valore ideale
QPR-TMR	Tempo di Risposta Medio	3 secondi	2 secondi

Tabella 11: Metriche efficienza

3.6 Funzionalità

QPR-RC: Requirements Coverage, indica la percentuale dei requisiti soddisfatti. Per il calcolo del valore accettato si considerano solo i requisiti obbligatori.

Formula valore accettato:

$$RC_{obb} = \frac{NR_{os}}{NR_{ot}} \cdot 100$$

- RC_{obb} : Requirements Coverage obbligatori;
- NR_{os} : Numero di requisiti obbligatori soddisfatti;
- NR_{ot} : Numero di requisiti obbligatori totali.

Formula valore ideale:

$$RC_i = \frac{NR_s}{NR_t} \cdot 100$$

- RC_i : Requirements Coverage ideali;
- NR_s : Numero di requisiti ideali soddisfatti;
- NR_t : Numero di requisiti ideali totali.

Metrica	Descrizione	Valore accettabile	Valore ideale
QPR-RC	Requirements Coverage	100% RC_{obb}	100% RC_i

Tabella 12: Metriche Funzionalità

3.7 Compatibilità

QPR-CB: Compatibilità Browser, indica la percentuale di browser supportati in relazione a quelli previsti.

Formula valore accettato:

$$CB = \frac{BW_s}{BW_p} \cdot 100$$

- CB : Compatibilità browser;

- BW_s : Browser supportati;
- BW_p : Browser previsti.

Metrica	Descrizione	Valore accettabile	Valore ideale
QPR-CB	Compatibilità Browser	100%	100%

Tabella 13: Metriche Compatibilità

4 Test

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 SWEet16, durante lo sviluppo dell' RTB, ha eseguito esclusivamente un'unica tipologia di test sui vari componenti utilizzati nella programmazione del PoC. Per perseguire la correttezza del prodotto e facilitare la fase di validazione, la verifica è stata svolta in parallelo allo sviluppo (*Modello a V^G*). I test dovranno essere resi il più automatici possibile, per evitare che la fase di testing rallenti la produzione.

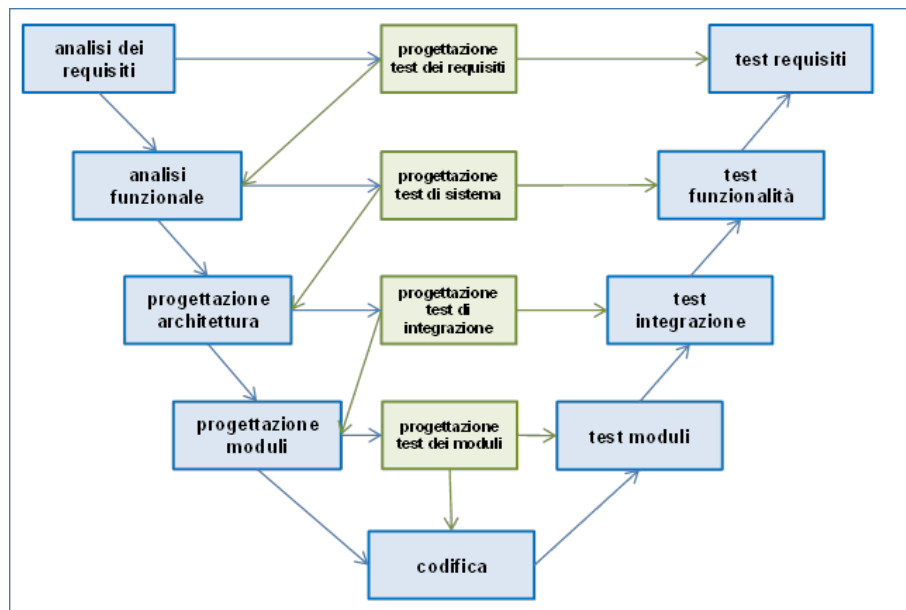


Immagine 1: Modello a V

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. Questa viene 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.

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. In particolare, questa tipologia di test mira a soddisfare tutti i requisiti funzionali e la maggior parte di quelli non funzionali, compresi aspetti relativi all'usabilità, sicurezza, performance e vulnerabilità.

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 e mirano a soddisfare pienamente i requisiti, accertandosi di avere un prodotto funzionante e soddisfacente le aspettative progettuali iniziali.

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*, verificando quindi di non perdere funzionalità nella progressiva realizzazione del prodotto software.

4.2 Specifica dei test

Al fine di garantire di creare una denominazione uniforme e per facilitare la comprensione, i test sono identificati da un codice come segue:

T[Tipologia][Identificativo]

Dove:

- **Tipologia** indica il tipo del test eseguito:
 - U: Per Test di Unità;
 - I: Per Test di Integrazione;

- S: Per Test di Sistema;
- A: Per Test di Accettazione;
- R: Per Test di Regressione.

- **Identificativo** del test in oggetto.

Ciascuna tipologia di test sarà rappresentata da apposite tabelle, comprensive di identificativo, descrizione e stato. Come riportato precedentemente, al momento sono stati effettuati esclusivamente i *test di unità*.

4.2.1 Test di unità

Tabella 14:

Identificativo	Descrizione	Stato
TU1	Si verifica il corretto funzionamento di axios-client	Superato
TU2	Si verifica il corretto funzionamento di ClientService	Superato
TU3	Si verifica il corretto funzionamento di ContextProvider	Superato
TU4	Si verifica il corretto funzionamento di CreazioneProfiloCliente	Superato
TU5	Si verifica il corretto funzionamento di CreazioneProfiloRistoratore	Superato
TU6	Si verifica il corretto funzionamento di DashBoardCliente	Superato
TU7	Si verifica il corretto funzionamento di DashboardRistoratore	Superato
TU8	Si verifica il corretto funzionamento di DivisioneContoPagamento	Superato
TU9	Si verifica il corretto funzionamento di FormIngredienti	Superato
TU10	Si verifica il corretto funzionamento di FormOrdinazione	Superato
TU11	Si verifica il corretto funzionamento di FormPietanze	Superato
TU12	Si verifica il corretto funzionamento di FormPrenotazione	Superato
TU13	Si verifica il corretto funzionamento di GestioneIngredienti	Superato
TU14	Si verifica il corretto funzionamento di GestioneMenu	Superato
TU15	Si verifica il corretto funzionamento di IntolleranzeService	Superato
TU16	Si verifica il corretto funzionamento di Layout	Superato
TU17	Si verifica il corretto funzionamento di LinkInvito	Superato
TU18	Si verifica il corretto funzionamento di Login	Superato

Continued on next page

Tabella 14: (Continued)

TU19	Si verifica il corretto funzionamento di Menu	Superato
TU20	Si verifica il corretto funzionamento di ModificaInfoAccount	Superato
TU21	Si verifica il corretto funzionamento di ModificaProfiloCliente	Superato
TU22	Si verifica il corretto funzionamento di ModificaProfiloRistoratore	Superato
TU23	Si verifica il corretto funzionamento di NotFound	Superato
TU24	Si verifica il corretto funzionamento di Notifiche	Superato
TU25	Si verifica il corretto funzionamento di ResturantCard	Superato
TU26	Si verifica il corretto funzionamento di Ristorante	Superato
TU27	Si verifica il corretto funzionamento di Ristoranti	Superato
TU28	Si verifica il corretto funzionamento di router	Superato
TU29	Si verifica il corretto funzionamento di SearchBar	Superato
TU30	Si verifica il corretto funzionamento di SelezioneProfilo	Superato
TU31	Si verifica il corretto funzionamento di SignUp	Superato
TU32	Si verifica il corretto funzionamento di SingolaPrenotazioneCliente	Superato
TU33	Si verifica il corretto funzionamento di SingolaPrenotazioneRistoratore	Superato
TU34	Si verifica il corretto funzionamento di AllergeniControllerTest	Superato
TU35	Si verifica il corretto funzionamento di AuthControllerTest	Superato
TU36	Si verifica il corretto funzionamento di ClientControllerTest	Superato
TU37	Si verifica il corretto funzionamento di ExampleTest	Superato
TU38	Si verifica il corretto funzionamento di IngredienteControllerTest	Superato
TU39	Si verifica il corretto funzionamento di InvitiTest	Superato
TU40	Si verifica il corretto funzionamento di NotificheControllerTest	Superato
TU41	Si verifica il corretto funzionamento di OrdinazioniTest	Superato
TU42	Si verifica il corretto funzionamento di PietanzaControllerTest	Superato
TU43	Si verifica il corretto funzionamento di PrenotazioneControllerTest	Superato
TU44	Si verifica il corretto funzionamento di ProfileControllerTest	Superato

Continued on next page

Tabella 14: (Continued)

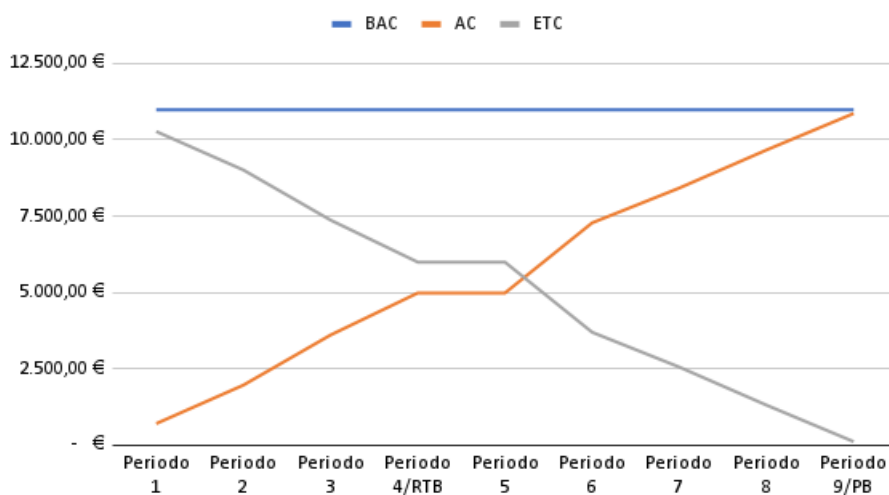
TU45	Si verifica il corretto funzionamento di RistoratoreControllerTest	Superato
TU46	Si verifica il corretto funzionamento di RistoratoreTest	Superato
TU47	Si verifica il corretto funzionamento di UserControllerTest	Superato

5 Resoconto delle attività di verifica

5.1 Fornitura

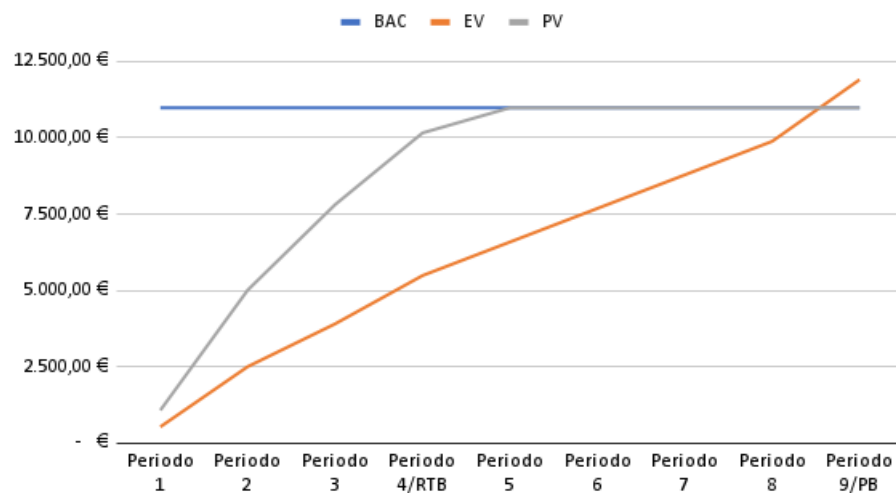
5.1.1 MPC-AC e MPC-ETC: Actual Cost e Estimated to Completion

Grafico 1: Metriche MPC-AC e MPC-ETC



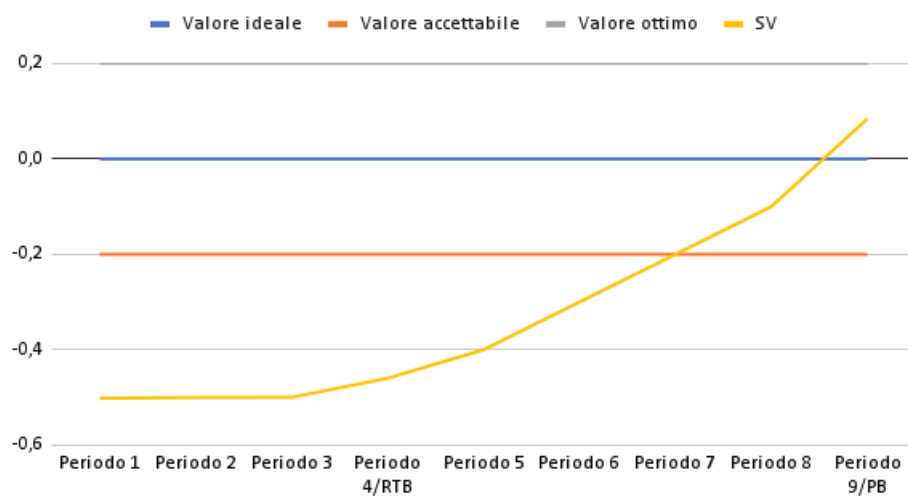
5.1.2 MPC-EV e MPC-PV: Earned Value e Planned Value

Grafico 2: Metriche MPC-EV e MPC-PV



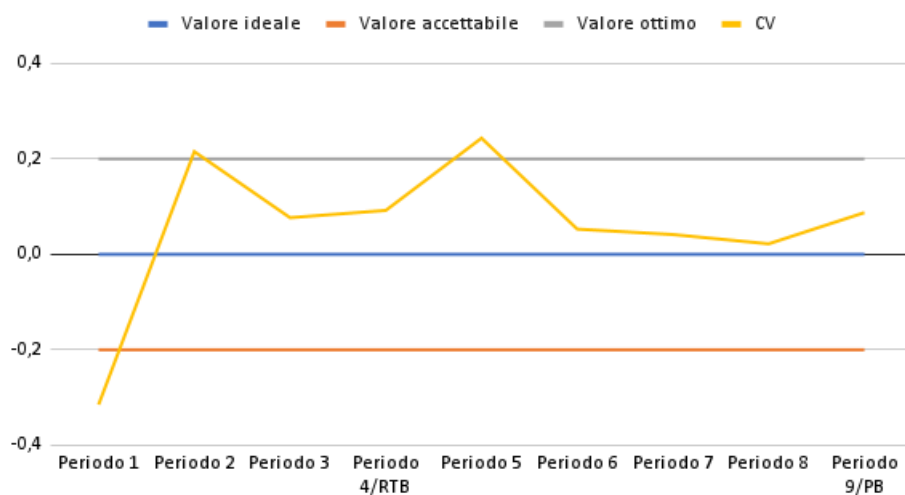
5.1.3 MPC-SV: Schedule Variance

Grafico 3: Metrica MPC-SV



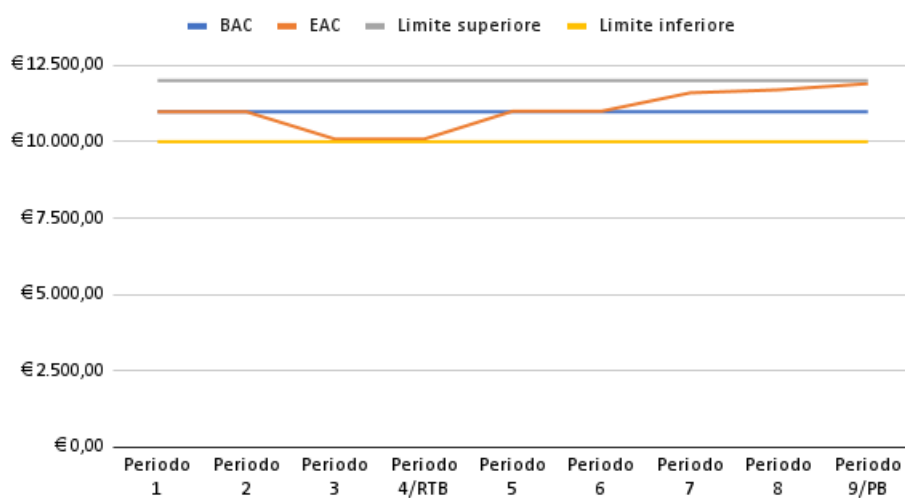
5.1.4 MPC-CV: Cost Variance

Grafico 4: Metrica MPC-CV



5.1.5 MPC-EAC: Estimated at Completion

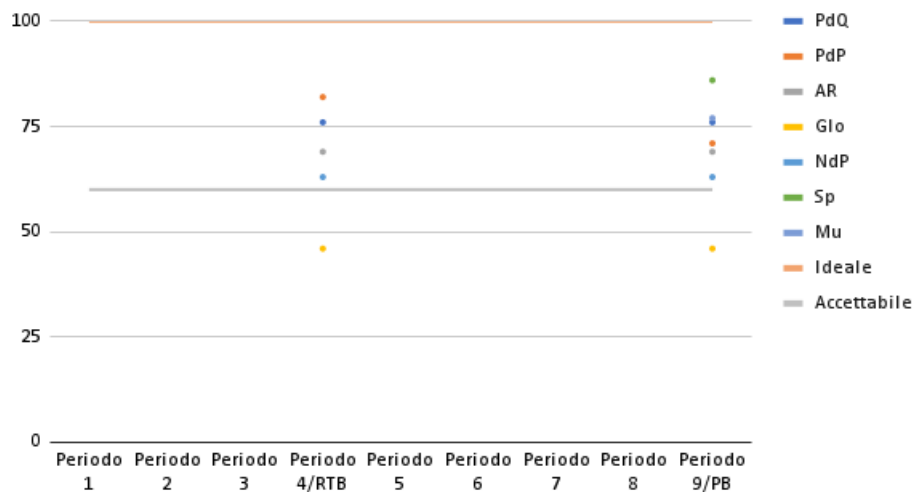
Grafico 5: Metrica MPC-EAC



5.2 Documentazione

5.2.1 MPC-IG: Indice Gulpease

Grafico 6: Metrica Gulpease



5.2.2 MPC-CO: Correttezza Ortografica

Grafico 7: Metrica MPC-CO

