



PEBKAC

Gruppo: 11

Email: pebkacswe@gmail.com

Docs: <https://pebkac-swe-group-11.github.io>

GitHub: <https://github.com/PEBKAC-SWE-Group-11>



Università degli Studi di Padova

Corso di Laurea: Informatica

Corso: Ingegneria del Software

Anno Accademico: 2024/2025

Piano di qualifica

Informazioni sul documento:

Verificatore	Derek Gusatto Alessandro Benin
Redattore	Matteo Piron Matteo Gerardin Derek Gusatto
Uso	Esterno
Destinatari	Tullio Vardanega Riccardo Cardin Vimar S.p.A.

Registro delle versioni

Versione	Data	Autore	Ruolo	Descrizione
1.0.0	2025-01-25	Derek Gusatto	Responsabile	Approvazione e rilascio
0.2.0	2025-01-24	Alessandro Benin	Verificatore	Verifica completa
0.1.6	2025-01-24	Derek Gusatto	Responsabile	§1.4 Riferimenti §5.5 Considerazioni finali in vista della revisione RTB
0.1.5	2025-01-24	Derek Gusatto	Responsabile	
0.1.4	2025-01-23	Derek Gusatto	Responsabile	Correzioni tabelle metriche, aggiunta grafici, correzione minori varie
0.1.3	2025-01-22	Derek Gusatto	Responsabile	Correzioni minori
0.1.2	2025-01-19	Matteo Gerardin	Amministratore	Test da TS-21 a TS-32 e da TA-21 a TA-25, eliminazione test end-to-end
0.1.1	2025-01-18	Derek Gusatto	Progettista	Test da TS-1 a TS-20 e da TA-1 a TA-20
0.1.0	2025-01-15	Derek Gusatto	Verificatore	Verifica §1, §2, §3, §4
0.0.3	2024-12-23	Matteo Gerardin	Amministratore	§4 Strategie di testing (Struttura e definizioni)
0.0.2	2024-12-06	Matteo Piron	Amministratore	§2 Qualità di Processo, §3 Qualità di Prodotto
0.0.1	2024-12-04	Matteo Piron	Amministratore	§1 Introduzione

Indice

1	Introduzione	6
1.1	Scopo del documento	6
1.2	Scopo del prodotto	6
1.3	Glossario	6
1.4	Riferimenti	6
1.4.1	Riferimenti normativi	6
1.4.2	Riferimenti informativi	6
2	Qualità di processo	8
2.1	Processi primari	8
2.1.1	Fornitura	8
2.1.2	Sviluppo	8
2.1.2.1	Codifica	8
2.2	Processi di supporto	8
2.2.1	Documentazione	8
2.2.2	Gestione della qualità	9
3	Qualità del prodotto	10
3.1	Funzionalità	10
3.2	Manutenibilità	10
3.3	Affidabilità	10
3.4	Efficienza	10
4	Strategie di testing	11
4.1	Notazione dei test	11
4.2	$Test_G$ di unità	12
4.3	$Test_G$ di integrazione	12
4.4	$Test_G$ di sistema	12
4.5	$Test_G$ di accettazione	17
5	Cruscotto delle metriche	21
5.1	Qualità di processo - Fornitura	21
5.1.1	Earned Value, Actual Cost, Planned Value	21
5.1.2	Estimated To Completion, Estimated At Completion	22
5.1.3	Budget Variance	22
5.1.4	Cost Variance, Schedule Variance	23
5.1.5	Cost Performance Index	24
5.2	Qualità di processo - Documentazione	24
5.2.1	Indice di Gulpease	24
5.3	Qualità di processo - Gestione della qualità	25
5.3.1	Metriche non soddisfatte	25
5.4	Qualità di prodotto - Funzionalità	25
5.4.1	Requisiti soddisfatti	25
5.5	Considerazioni finali in vista della revisione RTB	26

Elenco delle figure

1	Earned Value, Actual Cost, Planned Value	21
2	Estimated To Completion, Estimated At Completion	22
3	Estimated To Completion, Estimated At Completion	22
4	Cost Variance, Schedule Variance	23
5	Cost Performance Index	24
6	Indice di Gulpease	24
7	Metriche non soddisfatte	25
8	Requisiti soddisfatti	25

Elenco delle tabelle

1	Metriche per il processo di fornitura	8
2	Metriche per il processo di codifica	8
3	Metriche per il processo di documentazione	8
4	Metriche per il processo di gestione delle qualità	9
5	Metriche _G per la funzionalità _G del prodotto	10
6	Metriche _G per la manutenibilità _G del prodotto	10
7	Metriche _G per l'affidabilità _G del prodotto	10
8	Metriche _G per l'efficienza del prodotto	10
9	Stato dei <i>test_G</i> di sistema	16
10	Stato dei <i>test_G</i> di accettazione	20

1 Introduzione

1.1 Scopo del documento

Il presente documento ha l'obiettivo di definire le strategie di verifica e validazione messe in atto per garantire la qualità del prodotto e dei processi associati al progetto. La creazione del piano di qualifica è progressiva e incrementale nel tempo per consentire al team di riportare gli esiti delle verifiche effettuate nel tempo, con l'intento di mantenere ed eventualmente incrementare la qualità dell'intero progetto.

1.2 Scopo del prodotto

Il progetto "Vimar GENIALE" mira a sviluppare un'applicazione intelligente che supporti installatori elettrici nell'uso di dispositivi Vimar_G, facilitando l'accesso alle informazioni tecniche sui prodotti, rispondendo a domande poste in linguaggio naturale. La tecnologia alla base prevede l'uso di modelli di LLM_G e di tecniche RAG_G , con una struttura di gestione basata su $container_G$ e integrata in un ambiente $cloud_G$. Il sistema include tre componenti principali: una *applicativo web responsive_G*, un *applicativo server_G* e un'*infrastruttura cloud-ready_G*.

1.3 Glossario

Per evitare ambiguità relative al linguaggio utilizzato nei documenti, viene fornito il Glossario V1.0.0, nel quale si possono trovare tutte le definizioni di termini che hanno un significato specifico che vuole essere disambiguato. Tali termini sono marcati con una G a pedice.

1.4 Riferimenti

1.4.1 Riferimenti normativi

- Norme di Progetto v1.0.0
- PD1 - Regolamento del progetto didattico
<https://www.math.unipd.it/~tullio/IS-1/2024/Dispense/PD1.pdf>
- Capitolato d'Appalto C2: Vimar GENIALE
<https://www.math.unipd.it/~tullio/IS-1/2024/Progetto/C2.pdf>

1.4.2 Riferimenti informativi

- T7 - Qualità del Software
<https://www.math.unipd.it/~tullio/IS-1/2024/Dispense/T07.pdf>
- T8 - Qualità del processo
<https://www.math.unipd.it/~tullio/IS-1/2024/Dispense/T08.pdf>
- T9 - Verifica e Validazione: Introduzione
<https://www.math.unipd.it/~tullio/IS-1/2024/Dispense/T09.pdf>

- **T10 - Verifica e Validazione: Analisi statica**
<https://www.math.unipd.it/~tullio/IS-1/2024/Dispense/T10.pdf>
- **T11 - Verifica e Validazione: Analisi dinamica (Testing)**
<https://www.math.unipd.it/~tullio/IS-1/2024/Dispense/T11.pdf>
- **Glossario v1.0.0**

2 Qualità di processo

La qualità di processo si basa sull'idea che, per realizzare un prodotto conforme a specifici standard qualitativi, è essenziale monitorare e migliorare regolarmente i processi che lo generano. Questo principio si applica all'intera gamma di attività, pratiche e metodologie impiegate durante il ciclo di vita del *software*_G. In altre parole, la qualità dei processi ha l'obiettivo di andare a conformare la qualità del prodotto in modo tale da garantire sempre che gli standard definiti nel documento *Norme di Progetto* vengano rispettati ed eventualmente migliorati. Di seguito sono elencate le metriche che il team si impegna a rispettare per garantire l'eccellenza nei processi.

2.1 Processi primari

2.1.1 Fornitura

Metrica	Descrizione	Valore Accettabile	Valore Ideale
CV	Cost variance	± 150	0
PV	Planned Value	≥ 0	$\leq BAC$
EV	Earned Value	≥ 0	$\leq EAC$
AC	Actual Cost	≥ 0	$\leq EAC$
CPI	Cost Performance Index	tra 0.95 e 1.05	1
EAC	Estimated At Completion	$\pm 5\%$ del budget preventivo	budget preventivo
ETC	Estimated To Completion	≥ 0	$\leq EAC$
SV	Schedule Variance	± 150	0%
BV	Budget Variance	$\geq 10\%$	0%
BAC	Budget At Completion	-	-

Tabella 1: Metriche per il processo di fornitura

2.1.2 Sviluppo

2.1.2.1 Codifica

Metrica	Descrizione	Valore accettabile	Valore ideale
SC	Statement Coverage	0	$\geq 100\%$

Tabella 2: Metriche per il processo di codifica

2.2 Processi di supporto

2.2.1 Documentazione

Metrica	Descrizione	Valore accettabile	Valore ideale
IG	Indice Gulpease	$\geq 65\%$	100

Tabella 3: Metriche per il processo di documentazione

2.2.2 Gestione della qualità

Metrica	Descrizione	Valore accettabile	Valore ideale
MNS	Metriche Non Soddisfatte	≤ 3	0

Tabella 4: Metriche per il processo di gestione delle qualità

3 Qualità del prodotto

La qualità del prodotto si concentra sulla valutazione del $software_G$ sviluppato, ponendo l'accento su caratteristiche come facilità d'uso, funzionalità_G, affidabilità_G, capacità di manutenzione_G e, in senso più ampio, sulle prestazioni complessive del prodotto. L'obiettivo principale del $team_G$ è anche quello non solo di soddisfare le attese del cliente fornendo un prodotto $software_G$ che implementi le funzionalità_G volute, ma che lo faccia seguendo precisi $standard_G$ di qualità. Vengono quindi fornite di seguito le metriche_G che il $team_G$ si impegna a soddisfare nel contesto della qualità del prodotto.

3.1 Funzionalità

Metrica	Descrizione	Valore accettabile	Valore ideale
ROS _G	Requisiti Obbligatori Soddisfatti	100%	0
RDS _G	Requisiti Desiderabili Soddisfatti	$\geq 0\%$	$\geq 75\%$
RPS _G	Requisiti Opzionali Soddisfatti	$\geq 0\%$	$\geq 50\%$

Tabella 5: Metriche_G per la funzionalità_G del prodotto

3.2 Manutenibilità

Metrica	Descrizione	Valore accettabile	Valore ideale
SFIN _G	<i>Structure Fan In_G</i>	da determinare	da determinare
SFOUT _G	<i>Structure Fan Out_G</i>	da determinare	da determinare

Tabella 6: Metriche_G per la manutenibilità_G del prodotto

3.3 Affidabilità

Metrica	Descrizione	Valore accettabile	Valore ideale
PTCP _G	<i>Passed Test Cases Percentage_G</i>	$\geq 80\%$	100%
CC _G	<i>Code Coverage_G</i>	$\geq 80\%$	100%

Tabella 7: Metriche_G per l'affidabilità_G del prodotto

3.4 Efficienza

Metrica	Descrizione	Valore accettabile	Valore ideale
TDE _G	Tempo Di Elaborazione _G	da determinare	da determinare

Tabella 8: Metriche_G per l'efficienza del prodotto

4 Strategie di testing

Per dimostrare che i *requisiti_G* individuati dagli analisti ed elencati nella sezione omonima dell'Analisi dei Requisiti siano soddisfatti, è necessario che vengano realizzati dei *test_G* appositi che verranno eseguiti sul prodotto sia in fase di codifica che in fase di *verifica_G* e *revisione_G*.

I *test_G* realizzabili possono essere suddivisi in quattro categorie principali:

- **Test_G di unità:** verificano il corretto funzionamento di una singola unità di codice indipendente (ad esempio una funzione), assicurandosi che produca i risultati attesi al variare dei possibili input_G, e vengono generalmente automatizzati per facilitare l'individuazione degli errori durante la fase di sviluppo;
- **Test_G di integrazione:** verificano il corretto funzionamento delle interazioni tra diverse unità di codice o componenti di un sistema, assicurandosi che, una volta integrati, i vari moduli lavorino insieme senza problemi, rilevando eventuali errori nelle interfacce e nei flussi di dati tra di essi;
- **Test_G di sistema:** verificano il comportamento complessivo di un'intera applicazione o sistema, testando tutte le sue componenti integrate per assicurarsi che soddisfatti i *requisiti_G* funzionali e non funzionali, assicurandosi di valutare il sistema nel suo insieme simulando l'uso reale per identificare eventuali problemi di performance, sicurezza o compatibilità;
- **Test_G di accettazione:** verificano se un sistema o una parte di esso soddisfa i *requisiti_G* e le aspettative degli utenti o del cliente, venendo eseguiti prima del *rilascio_G* del *software_G* per confermare che il prodotto finale sia pronto per l'uso e conforme alle specifiche concordate.

4.1 Notazione dei test

È stato deciso come notazione per identificare univocamente i *test_G* la seguente:

T[Tipologia][Numero]

Tipologia indica la tipologia del *test_G*:

- **U:** di unità;
- **I:** di integrazione;
- **S:** di sistema;
- **A:** di accettazione.

Ogni *test_G* si trova in uno **Stato**, che può essere:

- **V:** verificato. Questo stato indica che il *test_G* ha fornito un esito positivo;
- **NV:** non verificato. Questo stato indica che il *test_G* ha fornito un esito negativo;
- **NI:** non implementato. Questo stato indica che il *test_G* non è ancora stato implementato, e quindi non fornisce nessun esito.

4.2 *Test_G* di unità

I *test_G* di unità sono una tipologia di *test_G* utilizzata per verificare singoli componenti o unità di codice in isolamento, al fine di garantire che funzionino correttamente. Un'unità di codice può essere una funzione, un metodo, una classe o un modulo, a seconda del livello di granularità scelto. I *test_G* di unità vengono solitamente scritti dagli sviluppatori durante o immediatamente dopo la scrittura del codice e vengono utilizzati per:

- Validare il comportamento del codice, assicurandosi che ogni unità fornisca risultati corretti per un determinato insieme di input;
- Facilitare la manutenzione del *software_G*, individuando rapidamente errori introdotti da modifiche;
- Promuovere la modularità, progettando concettualmente componenti indipendenti e riutilizzabili.

Per la realizzazione di questa categoria di *test_G* per questo progetto saranno utilizzati i *framework_G* *Pytest_G* e *unittest_G* per Python, dato che quest'ultimo è il linguaggio scelto per la realizzazione del *backend_G*.

I *test_G* di unità, insieme ai *test_G* di integrazione, come richiesto nel capitolato, devono avere un *coverage_G* minimo pari al 75% (opzionalmente un *coverage_G* minimo pari al 90%).

4.3 *Test_G* di integrazione

I *test_G* di integrazione sono una tipologia di *test_G* progettata per verificare la capacità di diversi componenti o moduli di un sistema di funzionare insieme. I *test_G* di integrazione non mirano a testare singoli moduli in modo indipendente, come fa il *test_G* di unità, che si concentra su unità di codice isolate. Le caratteristiche principali dei *test_G* di integrazione sono:

- Monitorare i problemi di comunicazione tra moduli;
- Garantire la corretta configurazione e gestione delle dipendenze tra moduli; Testare il sistema in condizioni più vicine a quelle reali rispetto a quanto avviene con i *test_G* di unità.

I *test_G* di integrazione, insieme ai *test_G* di unità, come richiesto nel capitolato, devono avere un *coverage_G* minimo pari al 75% (opzionalmente un *coverage_G* minimo pari al 90%).

4.4 *Test_G* di sistema

I *test_G* di sistema sono una tipologia di *test_G* attraverso la quale vengono testati il comportamento e la funzionalità di un sistema completo nel suo insieme. Viene eseguito dopo che tutti i componenti o moduli sono stati integrati e serve a garantire che il sistema soddisfi i *requisiti_G* funzionali e non funzionali specificati. I *test_G* di sistema valutano il *software_G* in un ambiente il più possibile vicino a quello reale, simulando gli utenti di tale *software_G*. Le caratteristiche chiave dei *test_G* di sistema sono:

- Verifica dei *requisiti_G* funzionali: assicurarsi che il sistema fornisca le funzionalità previste;
- Verifica dei *requisiti_G* non funzionali: verifica di prestazioni, sicurezza, usabilità, scalabilità...
- *Test_G* End-to-End: valutazione di flussi di lavoro completi, inclusa l'interazione con altri sistemi o applicazioni esterne;
- Valutazione della conformità: garantire che il sistema aderisca a specifici standard o regolamenti.

Codice	Descrizione	<i>Requisito_G</i>	Stato
TS-1	Il sistema deve permettere al modulo <i>AI_G</i> di interrogare il <i>database_G</i> in modo efficace per recuperare informazioni sui prodotti. Le informazioni restituite devono essere corrette, coerenti e aggiornate, garantendo che gli utenti ottengano risposte utili per le loro domande.	RF.O.026, RF.O.027	NI
TS-2	Deve essere garantita l'integrazione fluida tra il <i>frontend_G</i> e il <i>backend_G</i> , con l'elaborazione delle richieste degli utenti tramite <i>GUI_G</i> e la restituzione di risposte tempestive e accurate nella stessa interfaccia.	UC1, UC6	NI
TS-3	Quando l'utente inserisce una domanda che supera il limite di caratteri predefinito, il sistema deve gestire questa situazione restituendo un messaggio di errore chiaro e comprensibile.	RF.O.009, UC7	NI
TS-4	La <i>pipeline_G</i> di estrazione automatizzata dei dati dal sito Vimar deve raccogliere e indicizzare in modo efficiente le informazioni, rendendole disponibili per le interrogazioni degli utenti.	RF.O.020, RF.O.025	NI
TS-5	Il sistema deve essere avviabile tramite <i>Docker Compose_G</i> e garantire che tutti i componenti dell'infrastruttura containerizzata funzionino correttamente in un ambiente scalabile e portabile.	RV.O.002, RV.O.007	NI
TS-6	La <i>dashboard_G</i> amministrativa deve fornire agli amministratori una panoramica chiara e aggiornata sull'utilizzo del sistema, incluse statistiche dettagliate sulle richieste effettuate dagli utenti.	RF.O.003, UC14	NI
TS-7	La <i>pipeline_G</i> di indicizzazione automatizzata deve operare senza interventi manuali, consentendo l'aggiornamento continuo delle informazioni nel <i>database_G</i> in base ai nuovi dati raccolti.	RF.O.025, RF.O.026	NI
TS-8	Deve essere possibile estrarre correttamente informazioni utili dai file <i>PDF_G</i> , inclusi schemi elettrici e manuali tecnici, rendendoli disponibili per la consultazione e il download.	RF.O.023, RF.D.022	NI
TS-9	Il sistema deve identificare e gestire richieste relative a argomenti proibiti, restituendo un messaggio appropriato che informa l'utente della restrizione.	RF.O.005, UC15	NI

Codice	Descrizione	<i>Requisito_G</i>	Stato
TS-10	L'infrastruttura del sistema deve supportare la scalabilità, consentendo l'esecuzione simultanea su più nodi senza compromettere le prestazioni o l'affidabilità.	RQ.O.002, RV.O.007	NI
TS-11	Il sistema deve funzionare correttamente su tutti i browser compatibili specificati, garantendo un'esperienza utente coerente e priva di errori.	RV.O.012 - RV.O.016	NI
TS-12	Le <i>API_G</i> devono essere protette con meccanismi di autenticazione adeguati, come le <i>API-Key_G</i> , per garantire accessi sicuri e controllati ai servizi del sistema.	RV.O.006, RF.O.027	NI
TS-13	Il <i>database_G</i> deve essere aggiornabile con nuove informazioni sui prodotti in modo che il sistema possa fornire risposte aggiornate e precise agli utenti.	RF.O.021, RF.O.024	NI
TS-14	La <i>dashboard_G</i> deve mostrare statistiche aggiornate in tempo reale, consentendo agli amministratori di monitorare l'utilizzo del sistema con una visione dinamica e dettagliata.	RF.P.016, UC14.3	NI
TS-15	Il sistema deve supportare sessioni multiple di utenti simultanei, garantendo l'isolamento delle sessioni e l'integrità dei dati per ciascun utente.	RQ.D.001, UC1	NI
TS-16	Il sistema deve essere in grado di rispondere accuratamente a domande poste in lingua italiana, utilizzando il modello <i>AI_G</i> per generare risposte appropriate.	RF.O.001, UC16	NI
TS-17	Le <i>sessioni_G</i> di conversazione devono poter essere salvate e recuperate senza perdita di dati, consentendo agli utenti di riprendere le interazioni da dove erano state interrotte.	RF.O.012, UC3	NI
TS-18	Le risposte del sistema devono includere immagini o schemi tecnici, quando richiesto, garantendo la loro corretta visualizzazione nell'interfaccia utente.	RF.O.027, UC17	NI
TS-19	La <i>dashboard_G</i> amministrativa deve fornire statistiche dettagliate sui <i>feedback_G</i> ricevuti dagli utenti, inclusi conteggi di risposte positive e negative.	RF.D.019, UC14.5	NI
TS-20	Gli utenti devono poter ricercare e recuperare conversazioni salvate in precedenza, garantendo l'accesso a tutte le informazioni contenute nelle conversazioni archiviate.	RF.P.040, UC9	NI

Codice	Descrizione	<i>Requisito_G</i>	Stato
TS-21	Il sistema deve gestire la situazione in maniera corretta e mostrare un messaggio di errore chiaro e dettagliato nel caso in cui un utente tenti di salvare una conversazione senza che ne sia stata creata alcuna.	UC4, RF.O.034	NI
TS-22	Il sistema deve consentire all'installatore di visualizzare la data e l'ora di invio di ogni messaggio nella cronologia delle conversazioni.	RF.P.041, RF.P.042	NI
TS-23	Il sistema deve consentire agli amministratori di azzerare il conteggio dei <i>feedback_G</i> positivi e negativi ricevuti, mostrando un messaggio di conferma.	RF.P.044, RF.P.045	NI
TS-24	Il sistema supporta correttamente la creazione di nuove <i>sessioni_G</i> di conversazione, garantendo che siano archiviate e accessibili successivamente.	RF.O.035, UC1	NI
TS-25	Il sistema consente agli utenti di accedere al contenuto dei documenti di interesse direttamente dalla risposta fornita.	RF.P.043, UC17	NI
TS-26	L'interfaccia utente del sistema deve essere completamente <i>responsive_G</i> , adattandosi a diversi dispositivi senza errori di layout.	RQ.D.001	NI
TS-27	Il sistema deve inviare un avviso in caso di errore durante l'indicizzazione automatica dei dati provenienti dal sito Vimar.	RF.O.025, RF.O.026	NI
TS-28	Il sistema deve includere una funzione di "ripristino" per le <i>sessioni_G</i> interrotte a causa di errori tecnici, garantendo che nessun dato venga perso.	UC3, RF.O.012	NI
TS-29	Il sistema deve mostrare un messaggio chiaro quando l'utente cerca di accedere alla <i>dashboard_G</i> con credenziali errate.	UC13	NI
TS-30	Il sistema deve restituire un messaggio specifico quando non esistono messaggi pregressi nella cronologia della conversazione selezionata.	UC10	NI
TS-31	Il sistema deve restituire un messaggio chiaro e dettagliato quando si tenta di accedere a una funzionalità riservata agli amministratori senza aver effettuato l'accesso.	UC12	NI
TS-32	Il sistema deve garantire che, durante la ricerca di informazioni, vengano restituiti solo prodotti pertinenti alla categoria selezionata dall'installatore.	UC6	NI

Tabella 9: Stato dei *test_G* di sistema
16 di 26

4.5 *Test_G* di accettazione

I *test_G* di accettazione sono una tipologia di *test_G* che verifica che un sistema o un'applicazione soddisfi *requisiti_G* ed aspettative concordate con il cliente o contro l'utente finale. Normalmente, questi *test_G* sono condotti sul ciclo finale del processo di sviluppo, prima della pubblicazione o della consegna del prodotto. Questi *test_G* hanno l'obiettivo di:

- Confermare la conformità ai *requisiti_G* funzionali: verificare che il sistema realizzi le funzionalità richieste;
- Verificare che sia appropriato per l'utilizzo nel mondo reale: assicurarsi che il sistema sia pronto per un ambiente di produzione.
- Dare all'ente proprietario la capacità di approvare o rifiutare il sistema: un *test_G* di accettazione di successo è proprio l'ultimo passo di approvazione per il *rilascio_G*.

Codice	Descrizione	Casi d'uso	Stato
TA-1	Il sistema deve consentire a un installatore di inserire una richiesta tramite conversazione libera e ricevere informazioni dettagliate, incluse descrizioni dei prodotti, schemi elettrici e manuali tecnici.	UC6	NI
TA-2	Un amministratore deve poter accedere alla <i>dashboard_G</i> inserendo credenziali corrette, garantendo l'accesso solo a utenti autorizzati.	UC12	NI
TA-3	Gli utenti devono poter fornire <i>feedback_G</i> sull'accuratezza delle risposte ricevute, e il sistema deve registrare correttamente tali <i>feedback_G</i> .	UC11	NI
TA-4	Il sistema deve consentire agli utenti di salvare le conversazioni e confermare che il salvataggio sia avvenuto con successo.	UC3	NI
TA-5	Le richieste relative a argomenti non pertinenti devono essere bloccate, e il sistema deve restituire un messaggio di cortesia senza fornire ulteriori risposte.	UC15	NI
TA-6	Gli utenti devono poter visualizzare lo storico delle conversazioni in corso, con tutte le risposte mostrate in ordine cronologico.	UC9	NI
TA-7	Deve essere possibile eliminare una conversazione salvata, con conferma visibile dell'avvenuta cancellazione.	UC5	NI
TA-8	Nel caso in cui venga superato il limite massimo di conversazioni consentite, il sistema deve notificare l'utente con un messaggio chiaro.	UC2	NI

Codice	Descrizione	Casi d'uso	Stato
TA-9	Gli amministratori devono poter accedere a statistiche di utilizzo tramite la <i>dashboard_G</i> , con informazioni dettagliate su tutte le richieste effettuate.	UC14	NI
TA-10	Gli amministratori devono poter visualizzare il numero di richieste effettuate tramite conversazione libera, direttamente dalla <i>dashboard_G</i> .	UC14.1	NI
TA-11	Gli utenti devono confermare l'eliminazione delle conversazioni salvate prima che queste vengano definitivamente rimosse dal sistema.	UC5	NI
TA-12	Il sistema deve fornire risposte dettagliate e accurate relative a prodotti appartenenti agli impianti Smart, su richiesta dell'utente.	UC6	NI
TA-13	Gli utenti devono ricevere risposte chiare e precise per prodotti relativi agli impianti Domotici, con tutte le informazioni pertinenti.	UC6	NI
TA-14	Il sistema deve estrarre automaticamente informazioni dal sito di Vimar e indicizzarle correttamente per consentire interrogazioni rapide.	UC6	NI
TA-15	Deve essere garantita la gestione di richieste per prodotti non presenti, con un messaggio di cortesia che informa l'utente dell'assenza di dati.	UC8	NI
TA-16	Gli utenti devono poter scaricare documenti <i>PDF_G</i> , come manuali tecnici, direttamente dal sistema.	UC17	NI
TA-17	Nel caso di superamento del limite massimo di caratteri consentiti in una domanda, il sistema deve notificare l'errore in modo chiaro.	UC7	NI

Codice	Descrizione	Casi d'uso	Stato
TA-18	Utilizzando un menù guidato, gli utenti devono poter accedere facilmente alle informazioni sui prodotti di loro interesse.	UC18	NI
TA-19	Il sistema deve restituire risposte che includano immagini, come schemi elettrici, garantendo la loro corretta visualizzazione nell'interfaccia utente.	UC17	NI
TA-20	Gli amministratori devono avere accesso a statistiche relative ai $feedback_G$ positivi ricevuti, visualizzandole nella $dashboard_G$ in modo dettagliato.	UC14.4	NI
TA-21	Gli installatori devono ricevere un messaggio di conferma visibile quando forniscono $feedback_G$ sul sistema.	UC11	NI
TA-22	Gli installatori devono poter recuperare una lista delle loro conversazioni salvate, con possibilità di visualizzare il contenuto completo di ciascuna.	UC9	NI
TA-23	Gli amministratori devono poter visualizzare il numero di richieste effettuate tramite conversazione libera, direttamente dalla $dashboard_G$.	UC14.2	NI
TA-24	Gli utenti devono poter interrompere una conversazione guidata e tornare al menu principale senza perdere lo storico delle interazioni effettuate fino a quel momento.	UC5	NI
TA-25	Gli amministratori devono poter visualizzare le statistiche aggiornate sul numero di parole utilizzate nelle richieste, direttamente dalla $dashboard_G$.	UC14.3	NI

Tabella 10: Stato dei $test_G$ di accettazione

5 Cruscotto delle metriche

5.1 Qualità di processo - Fornitura

5.1.1 Earned Value, Actual Cost, Planned Value

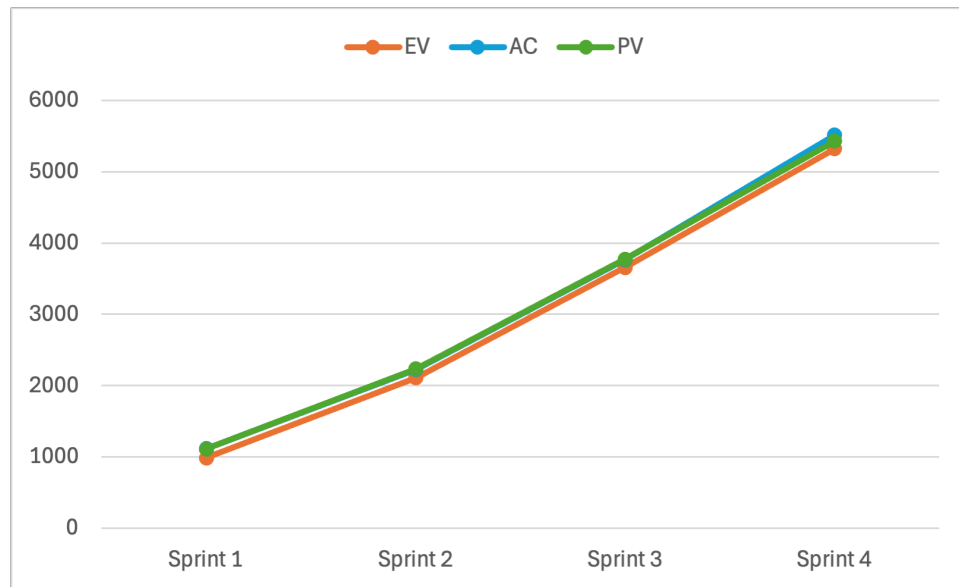


Figura 1: Earned Value, Actual Cost, Planned Value

Analisi

Queste metriche indicano un buon andamento essendo sempre abbastanza sovrapposte su tutti gli *sprint_G*, anche se si nota un graduale distaccamento verso il quarto *sprint_G* a causa dell'aumento dei costi per il raggiungimento degli obiettivi.

5.1.2 Estimated To Completion, Estimated At Completion

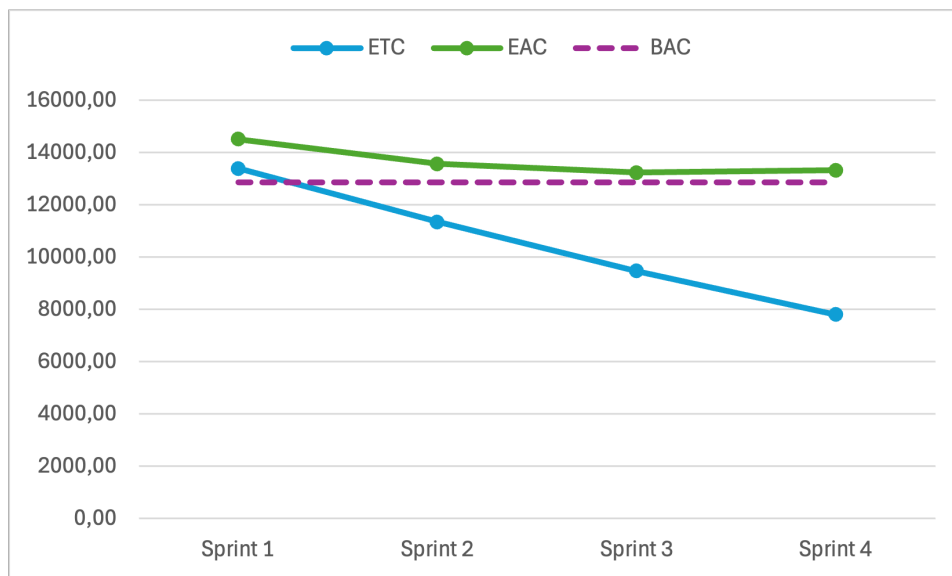


Figura 2: Estimated To Completion, Estimated At Completion

Analisi

Dopo un inizio non ottimale è possibile notare un riallineamento. L'Estimation At Completion si è riallineato al Budget At Completion, ma è sempre rimasto superiore, anche se di poco, a causa del costo maggiore per il raggiungimento di tutti gli obiettivi degli *sprint_G*. L'Estimated To Completion è sempre stato gradualmente discendente.

5.1.3 Budget Variance

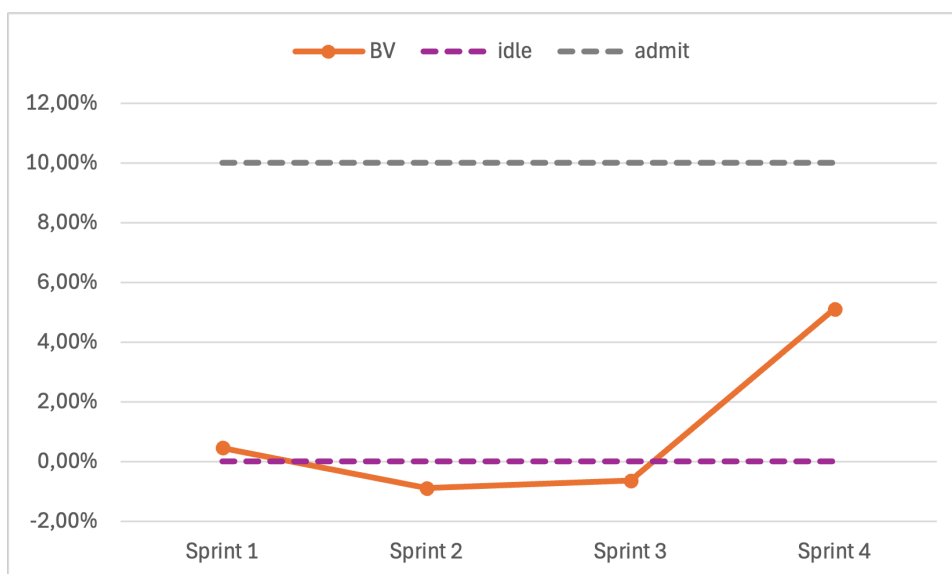


Figura 3: Estimated To Completion, Estimated At Completion

Analisi

Anche se la Budget Variance è sempre rimasta nei limiti ammissibili, è evidente come nel quarto *sprint_G*, dove la variazione è notevole, le ore preventivate fossero insufficienti e questo deve spingere a una più corretta pianificazione basata sulle retrospettive precedenti.

5.1.4 Cost Variance, Schedule Variance

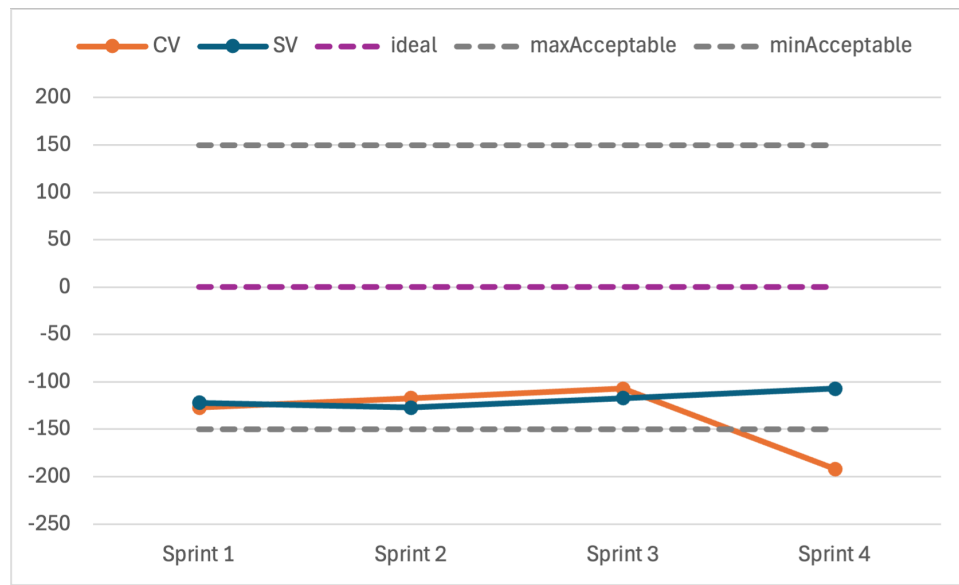


Figura 4: Cost Variance, Schedule Variance

Analisi

Per quanto i valori, soprattutto della Schedule Variance, siano praticamente sempre rimasti accettabili, non si notano dei netti miglioramenti, ma anzi un peggioramento, come in altre metriche, nel corso del quarto *sprint_G*. Questo deve spingere a una migliore gestione delle *risorse_G* per il raggiungimento degli obiettivi entro i costi preventivati.

5.1.5 Cost Performance Index

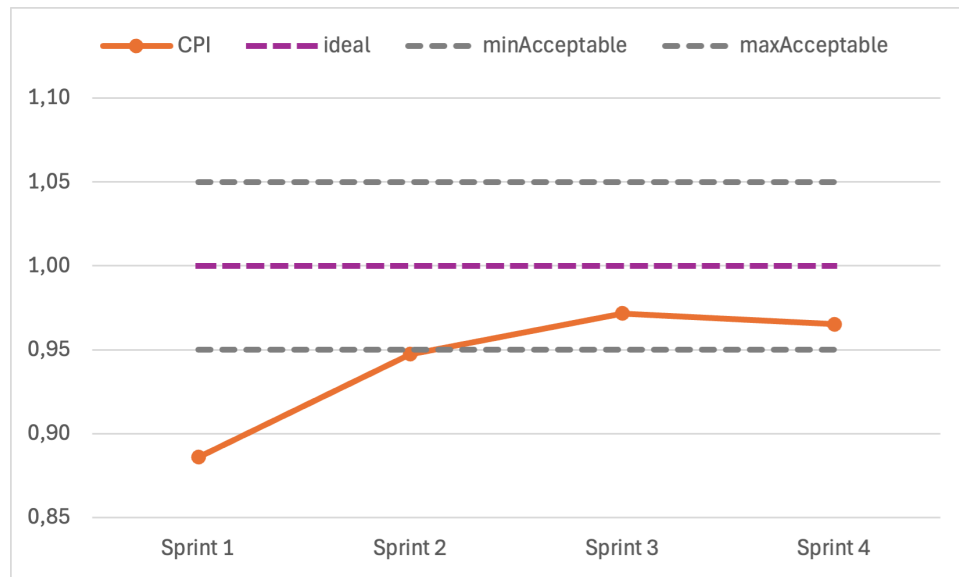


Figura 5: Cost Performance Index

Analisi

È evidente come il Cost Performance Index sia migliorato nel corso degli $sprint_G$, fino a rientrare nei limiti ammissibili, l'obiettivo deve essere quello di seguire il trend attuale per puntare al valore ideale.

5.2 Qualità di processo - Documentazione

5.2.1 Indice di Gulpease

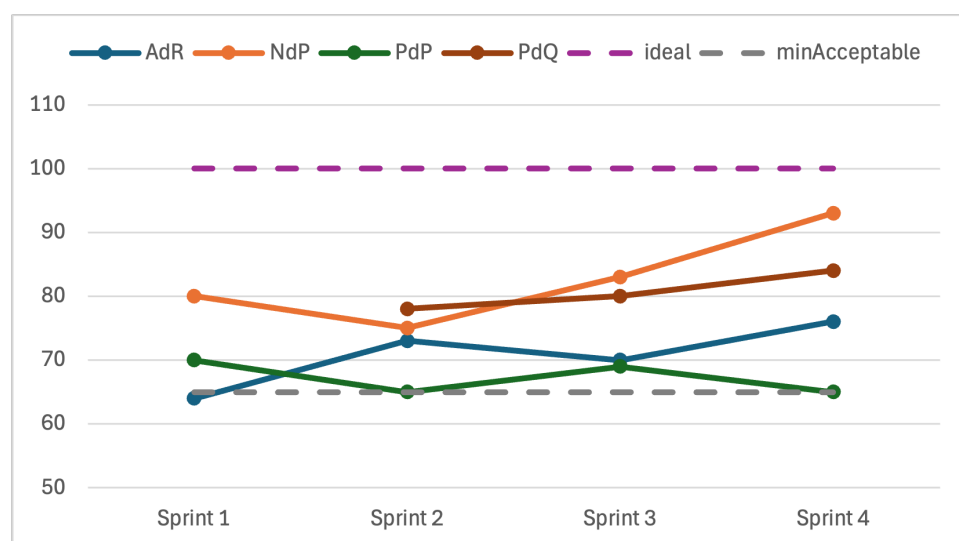


Figura 6: Indice di Gulpease

Analisi

Al termine del secondo *sprint_G* tutti i documenti erano in lavorazione ed entro i limiti minimi di leggibilità, si nota peraltro un miglioramento nella leggibilità di tutti i documenti ad esclusione del Piano di Progetto, sul quale il gruppo dovrà concentrarsi di più per garantire una scrittura comprensibile.

5.3 Qualità di processo - Gestione della qualità

5.3.1 Metriche non soddisfatte

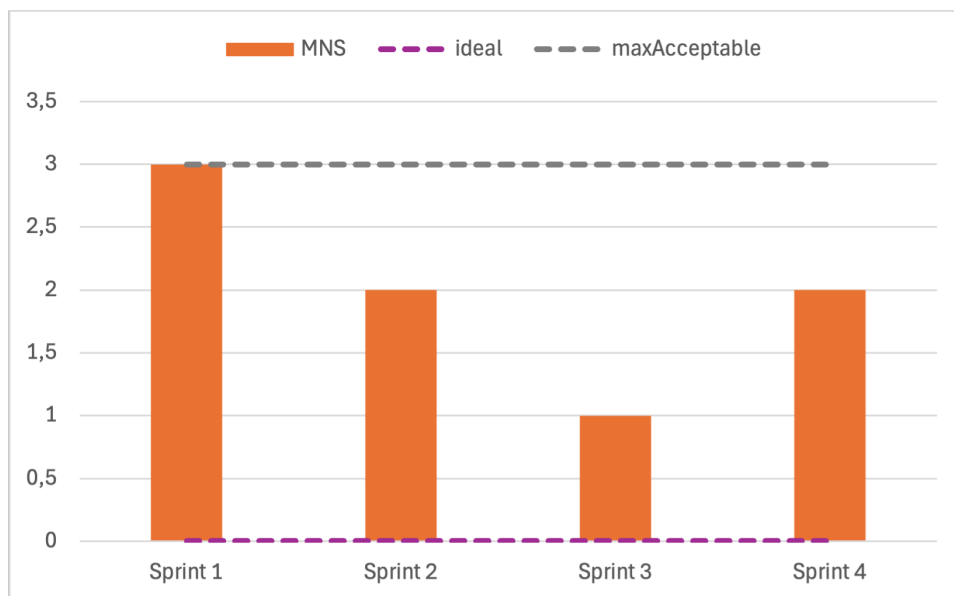


Figura 7: Metriche non soddisfatte

Analisi

La metrica che non risulta mai soddisfatta è EAC per la quale si rimanda all'analisi specifica. Altre metriche che risultano non soddisfatte sono ETC (primi due *sprint_G*), CPI (primo *sprint_G*) e CV (quarto *sprint_G* per il quale si rimanda all'analisi specifica).

5.4 Qualità di prodotto - Funzionalità

5.4.1 Requisiti soddisfatti

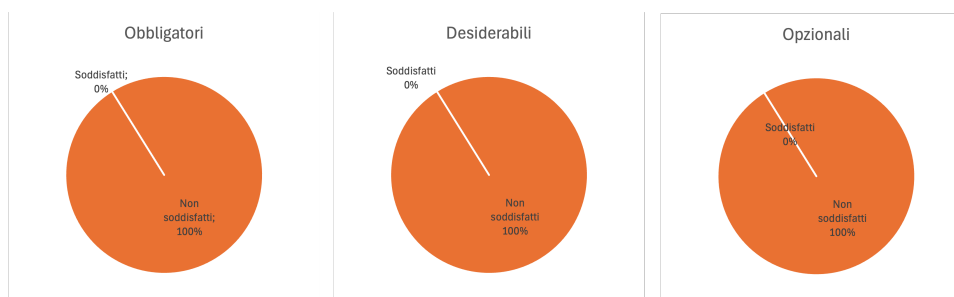


Figura 8: Requisiti soddisfatti

Analisi

Il progetto fino ad ora si è concentrato sulla creazione di un *Proof of Concept_G*, e non su prodotto finale con il quale soddisfare i *requisiti_G*, per cui è normale che nel cruscotto non risulti soddisfatto alcun *requisito_G*.

5.5 Considerazioni finali in vista della revisione RTB

Sin dalle prime fasi del progetto il gruppo si è posto l'obiettivo di avere un *Way of Working_G* chiaro e puntuale, anche se di natura incrementale, in quanto questo deve essere gradualmente ampliato e raffinato con il passare del tempo. Nonostante il *Way of Working_G* sia progressivamente migliorato con l'avanzare del progetto, alcune aree necessitano di essere ancora migliorate per far sì che la qualità di processo si rifletta positivamente sulla qualità di prodotto.

La comunicazione, all'interno del team e con il Proponente_G è stata fin da subito stabile e costruttiva, sia in modalità sincrona, permettendo riunioni efficaci di durata non eccessiva, che asincrona, permettendo una buona organizzazione del lavoro.

Il processo di automiglioramento che il gruppo si è impegnato ad assumere si è concentrato, e continuerà a concentrarsi, anche sulle modalità di gestione delle difficoltà e degli imprevisti, per escogitare contromisure efficaci che permettano di non sprecare *risorse_G*. Un obiettivo di vitale importanza per il gruppo nel corso dei prossimi *sprint_G* è l'aggiornamento rigoroso e puntuale del Piano di Progetto e del Piano di Qualifica con i dati rilevanti ed aggiornati, al fine da poter essere sfruttati al meglio.

x