



Code7Crusaders

Software Development Team

Piano di Qualifica^G

Membri del Team:

Enrico Cotti Cottini, Gabriele Di Pietro, Tommaso Diviesti
Francesco Lapenna, Matthew Pan, Eddy Pinarello, Filippo Rizzolo

Versioni

Ver.	Data	Autore	Verificatore	Descrizione
2.0	14/04/2025	Eddy Pinarello	Gabriele Di Pietro	Modifiche e approvazione documento
1.1	10/03/2025	Francesco Lapenna	Gabriele Di Pietro	Correzione errori RTB, riscritta sezione 6
1.0	11/02/2025	Gabriele Di Pietro	Filippo Rizzolo	Approvazione documento
0.5	06/02/2025	Gabriele Di Pietro	Matthew Pan	Stesura sezione 3.2
0.4	20/01/2025	Matthew Pan	Filippo Rizzolo	Stesura sezione 3.1 - Test Sistema
0.3	16/12/2024	Gabriele Di Pietro	Matthew Pan	Stesura sezione 5
0.2	10/12/2024	Gabriele Di Pietro	Francesco Lapenna	Aggiunte tabelle
0.1	05/12/2024	Gabriele Di Pietro	Enrico Cotti Cottini	Prima stesura del documento

Indice

1	Introduzione	4
1.1	Obiettivo del Documento	4
1.2	Glossario	4
1.3	Riferimenti	4
1.3.1	Normativi	4
1.3.2	Informativi	4
2	Qualità di processo	6
2.1	Processi di base e/o primari	6
2.1.1	Fornitura	6
2.1.2	Sviluppo	6
2.2	Processi di Supporto	7
2.2.1	Documentazione	7
2.2.2	Gestione della qualità	8
2.2.3	Risoluzione dei Problemi	8
2.3	Processi organizzativi	8
2.3.1	Pianificazione	8
3	Qualità di prodotto	9
3.1	Funzionalità	9
3.2	Affidabilità	9
3.3	Usabilità	9
3.4	Efficienza	9
3.5	Manutenibilità	10
4	Metodologie e Testing	11
4.1	Test di Unità	11
4.2	Test di Integrazione	12
4.3	Test di Sistema	12
4.4	Test di Accettazione	16
5	Cruscotto valutazione della qualità	17
5.1	Qualità processo di Fornitura	17
5.1.1	1PBM-PV - Planned Value e 4PBM-EV - Earned Value	17
5.1.2	5PBM-AC - Actual Cost e 2PBM-ETC - Estimate to Complete	18
5.1.3	6PBM-SV - Schedule Variance e 7PBM-CV - Cost Variance	19
5.1.4	3PBM-EAC - Estimated at Completion	19
5.2	Qualità processo di Documentazione	20
5.2.1	1PSM-IG - Indice Gulpease	20
5.3	Qualità del processo di gestione della qualità	21
5.3.1	7PSM-QMS - Metriche di Qualità Soddisfatte	21
5.4	Qualità del processo di pianificazione	22
5.4.1	1POM-RSI - Requirements Stability Index	22
6	Iniziative di automiglioramento per la qualità	23
6.1	Introduzione	23
6.2	Valutazione sull'organizzazione	23
6.3	Valutazione sui Ruoli	24

6.4	Valutazione sugli Strumenti	25
6.5	Considerazioni Finali	25

Elenco delle tabelle

1	Metriche di qualità per il processo di Fornitura	6
2	Metriche di qualità per il processo di Progettazione	6
3	Metriche di qualità per il processo di Codifica	7
4	Metriche di qualità per il processo di Verifica	7
5	Metriche di qualità per il processo di Documentazione	7
6	Metriche di qualità per il processo di Gestione della Qualità	8
7	Metriche di qualità per il processo di Risoluzione dei Problemi	8
8	Metriche di qualità per il processo di Pianificazione	8
9	Metriche di qualità di funzionalità del prodotto	9
10	Metriche di qualità di affidabilità del prodotto	9
11	Metriche di qualità di usabilità del prodotto	9
12	Metriche di qualità di efficienza del prodotto	9
13	Metriche di qualità di manutenibilità del prodotto	10
14	Legenda per il Test	11
17	Test di Sistema	15
18	Test di Accettazione	16
19	Contromisure adottate per migliorare l'organizzazione	23
20	Contromisure adottate per migliorare la gestione dei ruoli	24
21	Contromisure adottate per migliorare il rapporto con gli strumenti	25

Elenco delle figure

1	Proiezione di PV ed EV	17
2	Proiezione di AC e ETC	18
3	Proiezione di SV e CV	19
4	Proiezione di EAC	19
5	Indice di Gulpease per documento	20
6	Metriche di qualità soddisfatte	21
7	Metriche di qualità soddisfatte	22

1 Introduzione

1.1 Obiettivo del Documento

Il documento ha lo scopo di definire le strategie di verifica e validazione per assicurare il corretto funzionamento e uno standard di qualità dello strumento sviluppato e delle attività che lo accompagnano. Sarà sottoposto a revisioni continue, così da poter seguire l'evoluzione del progetto.

1.2 Glossario

Il Glossario^G è uno strumento utilizzato per risolvere eventuali dubbi su termini specifici utilizzati nella redazione del documento. Esso conterrà la definizione dei termini evidenziati e sarà consultabile al seguente link. I termini presenti in tale documento saranno evidenziati da una 'G' al pedice.

1.3 Riferimenti

1.3.1 Normativi

- **Regolamento del progetto**
<https://www.math.unipd.it/~tullio/IS-1/2024/Dispense/PD1.pdf>
- **Norme del Progetto**
https://code7crusaders.github.io/docs/RTB/documentazione_interna/norme_di_progetto.html

1.3.2 Informativi

- **Standard ISO/IEC 25010**
<https://iso25000.com/index.php/en/iso-25000-standards/iso-25010>
- **Standard ISO/IEC 12207:1995**
https://www.math.unipd.it/~tullio/IS-1/2009/Approfondimenti/ISO_12207-1995.pdf
- **Qualità di prodotto**
<https://www.math.unipd.it/~tullio/IS-1/2024/Dispense/T07.pdf>
- **Qualità di processo**
<https://www.math.unipd.it/~tullio/IS-1/2024/Dispense/T08.pdf>
- **Verifica e validazione**
 - Introduzione
<https://www.math.unipd.it/~tullio/IS-1/2024/Dispense/T09.pdf>
 - Analisi Statica
<https://www.math.unipd.it/~tullio/IS-1/2024/Dispense/T10.pdf>
 - Analisi Dinamica
<https://www.math.unipd.it/~tullio/IS-1/2024/Dispense/T11.pdf>
- **Capitolato d'appalto C7**
<https://www.math.unipd.it/~tullio/IS-1/2024/Progetto/C7.pdf>
- **Verbali esterni ed interni**
<https://code7crusaders.github.io/docs/RTB/index.html>
<https://code7crusaders.github.io/docs/PB/index.html>

- **Analisi dei requisiti V2.0**

https://code7crusaders.github.io/docs/PB/documentazione_esterna/analisi_dei_requisiti.html

- **Glossario V2.0^G**

https://code7crusaders.github.io/docs/RTB/documentazione_interna/glossario.html

2 Qualità di processo

La qualità di processo è un criterio fondamentale ed è alla base di ogni prodotto che rispecchi lo stato dell'arte. Per raggiungere tale obiettivo è necessario sfruttare delle pratiche rigorose che consentano lo svolgimento di ogni attività in maniera ottimale. Al fine di valutare nel miglior modo possibile la qualità del prodotto e l'efficacia dei processi, sono state definite delle metriche, meglio specificate nel documento Norme di ProgettoG e qui di seguito riepilogate. Esse sono state suddivise utilizzando lo **Standard** ISO/IEC12207:1995, il quale separa i processi di ciclo di vita del software in processi di base e/o primari, processi di supporto e processi organizzativi.

2.1 Processi di base e/o primari

2.1.1 Fornitura

Codice	Nome	Ammissibile	Ottimale
1PBM-PV	Planned Value ^G	$PV \geq 0$	$PV \leq BAC$
2PBM-ETC	Estimated to Complete	$ETC \geq 0$	$ETC \leq EAC$
3PBM-EAC	Estimated at Completion	$EAC \leq BAC + 10\%$	$EAC \leq BAC$
4PBM-EV	Earned Value ^G	$EV \geq 0$	$EV \leq EAC$
5PBM-AC	Actual Cost ^G	$AC \geq 0$	$AC \leq EAC$
6PBM-SV	Scheduled Variance ^G	$SV \geq -10\%$	$SV \geq 0\%$
7PBM-CV	Cost Variance ^G	$CV \geq -10\%$	$CV \geq 0\%$
8PBM-CPI	Cost Performance Index	$CPI \geq 0.8$	$CPI \geq 1$
9PBM-SPI	Scheduled Performance Index	$SPI \geq 0.8$	$SPI \geq 1$
10PBM-OTDR	On-Time Delivery Rate	$OTDR \geq 90\%$	$OTDR \geq 95\%$

Tabella 1: Metriche di qualità per il processo di Fornitura

2.1.2 Sviluppo

2.1.2.1 Progettazione (14PBM)

Codice	Nome	Ammissibile	Ottimale
14PBM-PG	Profondità delle Gerarchie	$PG \leq 7$	$PG \leq 5$

Tabella 2: Metriche di qualità per il processo di Progettazione

2.1.2.2 Implementazione (15PBM - 18PBM)

Codice	Nome	Ammissibile	Ottimale
15PBM-PPM	Parametri per Metodo	$PPM \leq 7$	$PPM \leq 5$
16PBM-CPC	Campi per Classe	$CPC \leq 8$	$CPC \leq 5$
17PBM-LCPM	Linee di Commento per Metodo	$LCPM \geq 50$	$LCPM \geq 20$

Tabella 3: Metriche di qualità per il processo di Codifica

2.1.2.3 Verifica e Validazione (8PSM-CC - 12PSM-PTCP)

Codice	Nome	Ammissibile	Ottimale
8PSM-CC	Code Coverage	$CC \geq 80\%$	$CC = 100\%$
12PSM-PTCP	Passed Test Case Percentage	$PTCP \geq 90\%$	$PTCP \geq 100\%$

Tabella 4: Metriche di qualità per il processo di Verifica

2.2 Processi di Supporto

2.2.1 Documentazione

(1PSM-IG - 2PSM-CO)

Codice	Nome	Ammissibile	Ottimale
1PSM-IG	Indice di Gulpease	$IG \geq 50$	$IG \geq 75$
2PSM-CO	Correttezza Ortografica	$CO = 0$ errori	$CO = 0$ errori

Tabella 5: Metriche di qualità per il processo di Documentazione

2.2.2 Gestione della qualità

(3PSM-FU - 7PSM-QMS)

Codice	Nome	Ammissibile	Ottimale
7PSM-QMS ^G	Metriche di Qualità Soddisfatte	$QMS^G \geq 90\%$	$QMS^G \geq 90\%$

Tabella 6: Metriche di qualità per il processo di Gestione della Qualità

2.2.3 Risoluzione dei Problemi

(13PSM-RMR - 14PSM-NCR)

Codice	Nome	Ammissibile	Ottimale
13PSM-RMR	Risk Mitigation Rate	$RMR \geq 80\%$	$RMR = 100\%$
14PSM-NCR	Richi non Calcolati	$NCR \leq 3$	$NCR = 0$

Tabella 7: Metriche di qualità per il processo di Risoluzione dei Problemi

2.3 Processi organizzativi

2.3.1 Pianificazione

(1POM-RSI)

Codice	Nome	Ammissibile	Ottimale
1POM-RSI	Requirements Stability Index ^G	$RSI \geq 75\%$	$RSI = 100\%$

Tabella 8: Metriche di qualità per il processo di Pianificazione

3 Qualità di prodotto

La sezione Qualità di Prodotto del Piano di Qualifica definisce i criteri e le metriche adottate per garantire che il software sviluppato soddisfi i requisiti di qualità previsti. Questa sezione descrive gli attributi fondamentali del prodotto, come affidabilità, manutenibilità, usabilità e prestazioni, e le strategie adottate per monitorarne e migliorarne la qualità durante il ciclo di sviluppo. L'obiettivo è assicurare che il software sia conforme agli standard richiesti e risponda efficacemente alle esigenze degli utenti finali.

3.1 Funzionalità

Codice	Nome	Ammissibile	Ottimale
11PBM-PRO	Percentuale Requisiti Obbligatori	$PRO = 100\%$	$PRO = 100\%$
12PBM-PRD	Percentuale Requisiti Desiderabili	$PRD \geq 30\%$	$PRD = 100\%$
13PBM-PRF	Percentuale Requisiti Facoltativi	$PRF \geq 0\%$	$PRF = 100\%$

Tabella 9: Metriche di qualità di funzionalità del prodotto

3.2 Affidabilità

Codice	Nome	Ammissibile	Ottimale
9PSM-BC	Branch Coverage	$BC \geq 80\%$	$BC = 100\%$
10PSM-SC	Statement Coverage	$SC \geq 80\%$	$SC = 100\%$
11PSM-FD	Failure Density	$FD \leq 15\%$	$FD = 0\%$

Tabella 10: Metriche di qualità di affidabilità del prodotto

3.3 Usabilità

Codice	Nome	Ammissibile	Ottimale
3PSM-FU	Facilità di Utilizzo	$FU \geq 3$ errori	$FU \geq 0$ errori
4PSM-TA	Tempo di Apprendimento	$TA \leq 12$ minuti	$TA \leq 8$ minuti
6PSM-TE	Tempo di Elaborazione	$TE \leq 10$ secondi	$TE \leq 5$ secondi

Tabella 11: Metriche di qualità di usabilità del prodotto

3.4 Efficienza

Codice	Nome	Ammissibile	Ottimale
5PSM-TR	Tempo di Risposta	$TR \leq 8$ secondi	$TR \leq 4$ secondi

Tabella 12: Metriche di qualità di efficienza del prodotto

3.5 Manutenibilità

Codice	Nome	Ammissibile	Ottimale
18PBM-CCM	Complessità Ciclomatica Metrica	$CCM \leq 6$	$CCM \leq 3$

Tabella 13: Metriche di qualità di manutenibilità del prodotto

4 Metodologie e Testing

In questa sezione si illustrano le metodologie di *Testing* adottate per garantire il rispetto dei vincoli individuati nella sezione *Requisiti* del documento Analisi dei Requisiti. I test sono suddivisi in cinque categorie:

1. Test di unità
2. Test di integrazione
3. Test di Sistema
4. Test di Accettazione

Verranno elencate le varie tipologie di test eseguite, indicando il codice del test, una breve descrizione di ciò che viene verificato e lo stato di avanzamento del test, espresso come segue.

S	Test Superato
NS	Test NON Superato
NI	Test NON Implementato

Tabella 14: Legenda per il Test

4.1 Test di Unità

I test di unità hanno l'obiettivo di verificare il corretto funzionamento delle singole unità software, ovvero delle funzioni, classi o metodi che compongono il sistema. Vengono eseguiti in modo isolato, senza dipendenze esterne, per assicurare che ogni componente si comporti come previsto in ogni possibile condizione di input. Questi test sono fondamentali per individuare errori in fase precoce dello sviluppo e garantire la qualità del codice. I risultati ottenuti permettono inoltre di semplificare la fase di integrazione e facilitare la manutenzione futura del software.

Codice	Descrizione	Stato
1T-U	Verifica della copertura per tutti gli adapters , tra cui 'ConversationPostgresAdapter', 'FaissAdapter', 'LangChainAdapter', 'MessagePostgresAdapter', 'SupportMessagePostgresAdapter', 'TemplatePostgresAdapter', 'UserPostgresAdapter'.	S
2T-U	Verifica della copertura per tutti i controllers , come 'AddFileController', 'AuthenticationController', 'ChatController', 'DeleteConversationController', 'GetMessageController', 'GetSupportMessageController', ecc.	S
3T-U	Verifica della copertura per i DTOs , inclusi 'AnswerDTO', 'ConversationDTO', 'MessageDTO', 'TemplateDTO', 'UserDTO', 'SupportMessageDTO', ecc.	S
4T-U	Verifica della copertura per le entities , come 'MessageEntity', 'SupportMessageEntity', 'UserEntity', 'ConversationEntity', 'FileEntity', ecc.	S

Codice	Descrizione	Stato
5T-U	Verifica della copertura per tutti i repositories , tra cui 'ConversationPostgresRepository', 'FaissRepository', 'LangChainRepository', 'MessagePostgresRepository', 'SupportMessagePostgresRepository', 'UserPostgresRepository', ecc.	S
6T-U	Verifica della copertura per i services , inclusi 'AddFileService', 'AuthenticationService', 'ChatService', 'GenerateAnswerService', 'DeleteConversationService', ecc.	S
7T-U	Verifica della copertura per tutti i models , tra cui 'UserModel', 'MessageModel', 'FileModel', 'TemplateModel', 'SupportMessageModel', 'QuestionModel', ecc.	S
8T-U	Verifica della copertura per le classi nei ports , come 'AuthenticationPort', 'DeleteConversationPort', 'GenerateAnswerPort', 'SaveMessagePort', 'ValidationPort', ecc.	S
9T-U	Verifica della copertura per gli usecases , intesi come componenti applicativi che coordinano il flusso tra controller, servizi e repository per casi d'uso specifici.	S

4.2 Test di Integrazione

I test di integrazione si concentrano sulla verifica del corretto funzionamento tra i vari moduli del sistema, in particolare nel contesto dell'accesso ai dati tramite PostgreSQL. Questi test mirano a garantire che le interazioni tra repository e database siano affidabili e coerenti con le aspettative.

Codice	Descrizione	Stato
1T-I	Verifica dell'integrazione tra 'ConversationPostgresRepository' e il database, assicurandosi che salvataggio, recupero e aggiornamento delle conversazioni funzionino correttamente.	S
2T-I	Verifica dell'integrazione tra 'MessagePostgresRepository' e il database, con test sui metodi di creazione, lettura e rimozione dei messaggi.	S
3T-I	Verifica dell'integrazione tra 'SupportMessagePostgresRepository' e il database, testando la gestione dei messaggi di supporto.	S
4T-I	Verifica dell'integrazione tra 'TemplatePostgresRepository' e il database, assicurando che le operazioni su template siano eseguite correttamente.	S

4.3 Test di Sistema

I test di sistema sono finalizzati alla verifica del soddisfacimento dei requisiti richiesti ed evidenziati nel documento Analisi dei Requisiti^G. Questi test vengono effettuati sul sistema nel suo complesso, per verificare che il software funzioni correttamente e che sia in grado di eseguire le operazioni richieste.

Codice	Descrizione	Stato
1T-S	Verificare che il caricamento dei dati semantici aziendali avvenga correttamente nei formati accettati.	S
2T-S	Verificare che il sistema gestisca correttamente documenti in formati non compatibili.	S
3T-S	Verificare che i testi vengano suddivisi correttamente in blocchi.	S
4T-S	Verificare che i blocchi di testo vengano trasformati in vettori tramite l'Embedding ^G Model.	S
5T-S	Verificare che i vettori siano memorizzati e indicizzati correttamente nel database vettoriale.	S
6T-S	Verificare che l'utente possa inviare una domanda attraverso l'interfaccia utente.	S
7T-S	Verificare che la query venga gestita correttamente tramite API REST ^G e inoltrata al sistema.	S
8T-S	Verificare che l'Embedding ^G Model trasformi la domanda in una rappresentazione vettoriale.	S
9T-S	Verificare che la ricerca nel database vettoriale restituisca i vettori più simili.	S
10T-S	Verificare che il sistema LLM ^G costruisca la risposta utilizzando il contesto fornito.	S
11T-S	Verificare che la risposta venga inviata correttamente al dispositivo dell'utente tramite API REST ^G .	S
12T-S	Verificare che l'utente registrato possa avviare e gestire una conversazione con il bot.	S
13T-S	Verificare che l'utente possa richiedere e ricevere informazioni sui prodotti durante una conversazione.	S
14T-S	Verificare che l'utente possa salvare una conversazione avviata.	S
15T-S	Verificare che l'utente possa visualizzare le conversazioni precedentemente salvate.	S
16T-S	Verificare che l'utente possa recuperare e riprendere una conversazione salvata.	S
17T-S	Verificare che l'utente possa eliminare una conversazione salvata.	S
18T-S	Verificare che l'accesso al sistema sia consentito solo con credenziali valide.	S
19T-S	Verificare che il sistema blocchi gli utenti non registrati.	S
20T-S	Verificare che il sistema prevenga attacchi come SQL Injection.	S

Codice	Descrizione	Stato
21T-S	Verificare che l'utente possa inviare feedback positivo o negativo sulla qualità della conversazione.	S
22T-S	Verificare che l'amministratore possa creare template di domande e risposte.	S
23T-S	Verificare che l'amministratore possa modificare template di domande e risposte.	NI
24T-S	Verificare che l'amministratore possa eliminare un template esistente.	S
25T-S	Verificare che il sistema blocchi la creazione di template in formato non valido.	S
26T-S	Verificare che l'amministratore possa monitorare le prestazioni del sistema dalla dashboard.	NI
27T-S	Verificare che l'amministratore possa visualizzare i feedback forniti dagli utenti.	S
28T-S	Verificare che l'amministratore possa importare dati da documenti esterni.	S
29T-S	Verificare che il sistema blocchi l'importazione di file non compatibili.	S
30T-S	Verificare che l'amministratore possa visualizzare le richieste di assistenza degli utenti.	S
31T-S	Verificare che l'amministratore possa segnalare una richiesta di assistenza presa in carico.	S
32T-S	Verificare che l'amministratore possa rispondere agli utenti via e-mail.	S
33T-S	Verificare che l'amministratore possa visualizzare l'utilizzo generale del servizio.	NI
34T-S	Verificare che l'amministratore possa visualizzare i costi del sistema.	NI
35T-S	Verificare che lo schema di progettazione della base di dati sia conforme ai requisiti.	S
36T-S	Verificare che il codice prodotto sia disponibile in formato sorgente tramite repository pubblici.	S
37T-S	Verificare che la documentazione descrittiva del sistema di raccomandazione sia completa e accessibile.	S
38T-S	Verificare che la documentazione riassuntiva delle metriche e dei risultati sia conforme ai requisiti.	S
39T-S	Verificare che l'LLM ^G sia integrato correttamente tramite API.	S
40T-S	Verificare che sia stato implementato almeno un database relazionale e che funzioni correttamente.	S

Codice	Descrizione	Stato
41T-S	Verificare che sia stato implementato almeno un database vettoriale e che funzioni correttamente.	S
42T-S	Verificare che sia stato implementato un embedding model, locale o tramite API.	S
43T-S	Verificare che la WebApp consenta di comunicare correttamente con il chatbot.	S

Tabella 17: Test di Sistema

4.4 Test di Accettazione

I test di Accettazione vengono effettuati per verificare che il Software soddisfi i requisiti richiesti e consentono di ultimare il processo di validazione finale.

Codice	Descrizione	Stato
TA01	Verificare che il sistema accetti documenti nei formati <code>.pdf</code> e <code>.txt</code> in input	S
TA02	Verificare che i documenti vengano suddivisi in blocchi di testo.	S
TA03	Verificare che il modello di embedding generi rappresentazioni vettoriali dei blocchi di testo.	S
TA04	Verificare che i vettori generati siano memorizzati nel database vettoriale.	S
TA05	Verificare che l'utente possa inviare domande tramite l'interfaccia della web app.	S
TA06	Verificare che la domanda venga inoltrata al sistema tramite API REST ^G .	S
TA07	Verificare che la domanda venga trasformata in una rappresentazione vettoriale.	S
TA08	Verificare che il sistema recuperi i vettori più simili dal database vettoriale.	S
TA09	Verificare che il sistema LLM ^G costruisca una risposta basata sulla domanda e sul contesto.	S
TA10	Verificare che la risposta venga inviata all'utente tramite API REST ^G .	S
TA11	Verificare che l'utente registrato possa avviare una conversazione con il bot.	S
TA12	Verificare che l'utente possa salvare una conversazione.	S
TA13	Verificare che il login con username e password funzioni.	S
TA14	Verificare la protezione contro SQL Injection e altri attacchi.	S
TA15	Verificare che l'utente possa fornire un feedback sulla conversazione.	S
TA16	Verificare che l'amministratore possa monitorare le prestazioni del sistema tramite dashboard.	S
TA17	Verificare che l'utente possa eliminare una conversazione salvata.	S
TA18	Verificare che l'utente possa inviare richieste di assistenza per contattare un operatore umano.	S

Tabella 18: Test di Accettazione

5 Cruscotto valutazione della qualità

5.1 Qualità processo di Fornitura

5.1.1 1PBM-PV - Planned Value e 4PBM-EV - Earned Value

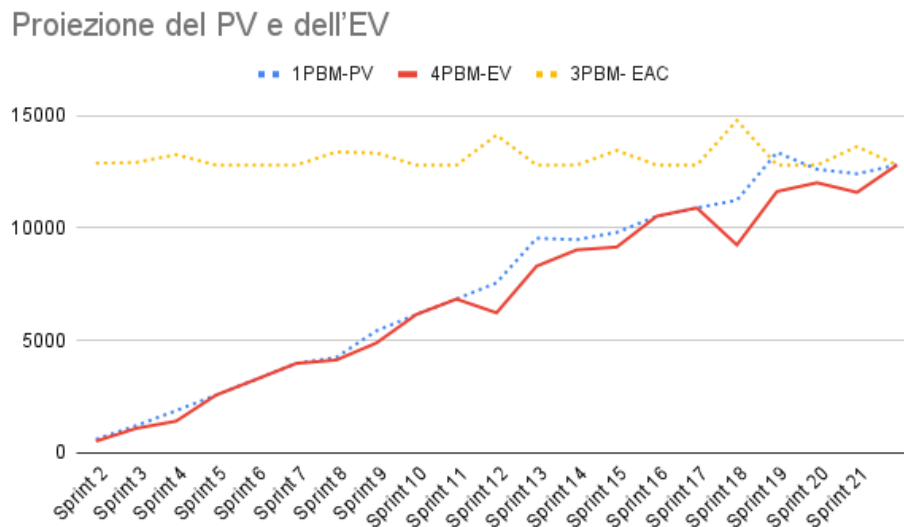


Figura 1: Proiezione di PV ed EV

Il grafico mostra l'andamento di Planned Value (PV) ed Earned Value (EV) nel tempo, evidenziando una forte sovrapposizione tra le due curve. Questo implica che il team di progetto ha seguito fedelmente il piano iniziale, eseguendo le attività previste nei tempi stabiliti. La convergenza quasi lineare di PV ed EV verso il Estimate at Completion (EAC) indica una distribuzione omogenea del lavoro lungo tutto l'orizzonte temporale del progetto. Ciò suggerisce che il progetto non ha subito ritardi significativi né ha registrato accelerazioni improvvise, ma ha mantenuto un ritmo costante di avanzamento. L'aderenza tra PV ed EV è un segnale positivo in termini di gestione del progetto, poiché significa che le attività sono state completate secondo le stime iniziali, senza deviazioni rilevanti. Questo può derivare da una pianificazione accurata, un'efficace allocazione delle risorse e una buona esecuzione da parte del team.

5.1.2 5PBM-AC - Actual Cost e 2PBM-ETC - Estimate to Complete

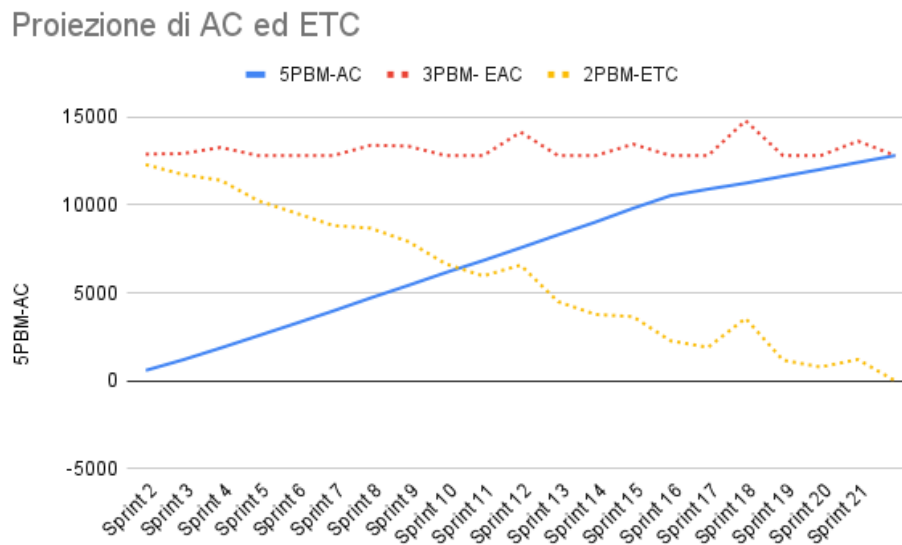


Figura 2: Proiezione di AC e ETC

Il grafico mostra l'andamento di tre metriche fondamentali nella gestione dei costi di progetto:

- Actual Cost (AC): il costo effettivamente sostenuto fino a un determinato momento. Questo valore cresce progressivamente nel tempo, indicando il consumo di risorse economiche man mano che il progetto avanza.
- Estimate to Complete (ETC): la stima dei costi necessari per completare il progetto. Si osserva un andamento decrescente, segno che, con l'avanzare delle attività, il budget residuo necessario si riduce.
- Estimate at Completion (EAC): la stima del costo totale previsto al completamento del progetto. La stabilità di questa metrica nel tempo suggerisce che non si stanno verificando scostamenti significativi rispetto al budget iniziale.

La stabilità dell'Estimate at Completion è un segnale positivo, poiché indica che le previsioni di spesa fatte in fase di pianificazione si stanno rivelando accurate e che il progetto non sta subendo variazioni di costo significative. In altre parole, i costi effettivi e quelli stimati rimangono allineati, suggerendo una gestione finanziaria efficace e senza imprevisti di rilievo.

5.1.3 6PBM-SV - Schedule Variance e 7PBM-CV - Cost Variance

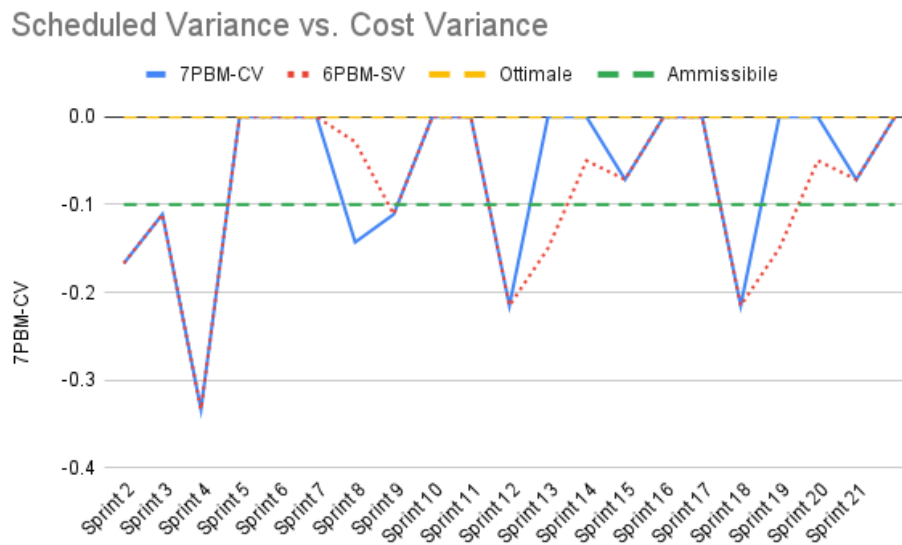


Figura 3: Proiezione di SV e CV

Il grafico mette in evidenza l'andamento di Scheduled Variance (SV) e Cost Variance (CV) nel tempo, mostrando come questi due indicatori siano quasi sempre sovrapposti. Questo significa che il progetto è gestito con un buon livello di efficienza, sia in termini di pianificazione che di controllo dei costi. L'andamento simile di SV e CV indica che le attività vengono completate nei tempi previsti senza significative deviazioni dal budget. Questo riflette un bilanciamento efficace tra l'esecuzione dei lavori e il controllo delle spese. Nel complesso, il grafico evidenzia un buon equilibrio tra tempistiche e risorse, dimostrando un efficace processo di gestione del progetto.

5.1.4 3PBM-EAC - Estimated at Completion

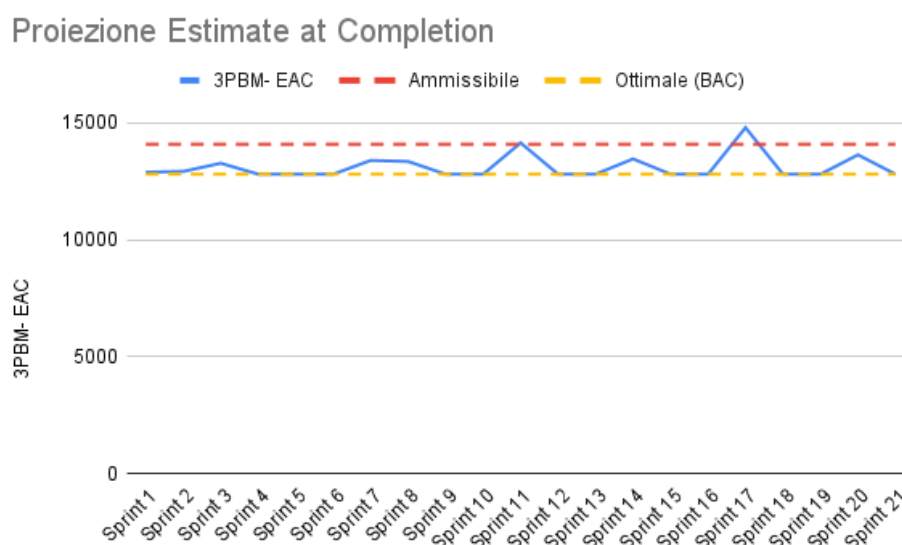


Figura 4: Proiezione di EAC

Il grafico rappresenta l'andamento dell'Estimate at Completion (EAC) rispetto al costo preventivato durante il corso del progetto. Si osserva che, nella maggior parte del tempo, l'EAC rimane allineato con il budget iniziale, indicando un controllo efficace dei costi e una gestione finanziaria coerente con le previsioni.

Tuttavia, emergono alcuni momenti critici in cui l'EAC si avvicina alla soglia massima ammissibile, segnalando possibili situazioni di rischio, come ritardi, variazioni nei costi delle risorse o inefficienze operative. Nonostante queste fluttuazioni, il valore dell'EAC rientra rapidamente nei parametri ottimali, suggerendo l'adozione di misure correttive tempestive che hanno permesso di riportare il progetto in linea con il budget.

Questo comportamento evidenzia una gestione attenta della valutazione dei costi a finire, con interventi mirati a contenere gli scostamenti e garantire che il progetto si mantenga all'interno delle soglie di sostenibilità economica.

5.2 Qualità processo di Documentazione

5.2.1 1PSM-IG - Indice Gulpease

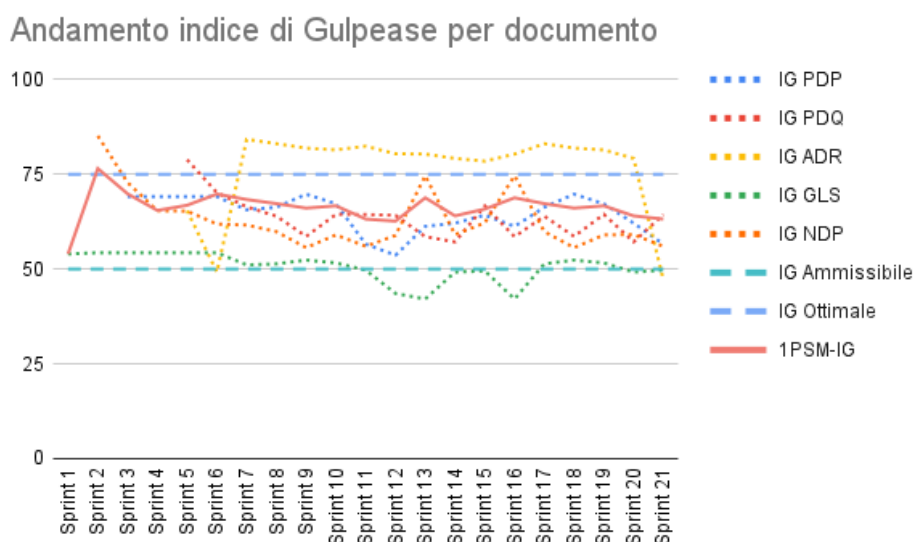


Figura 5: Indice di Gulpease per documento

L'indice di Gulpease è una metrica utilizzata per valutare la leggibilità di un testo in lingua italiana. L'indice varia tra 0 e 100, dove valori più alti indicano una maggiore leggibilità. Dal grafico emerge che l'indice di Gulpease della documentazione analizzata si mantiene costantemente in un intervallo compreso tra il valore ammissibile e quello ottimale. Ciò implica che la documentazione è chiara e ben strutturata e permette una fruizione veloce e immediata, riducendo il tempo di comprensione del contenuto.

5.3 Qualità del processo di gestione della qualità

5.3.1 7PSM-QMS - Metriche di Qualità Soddisfatte

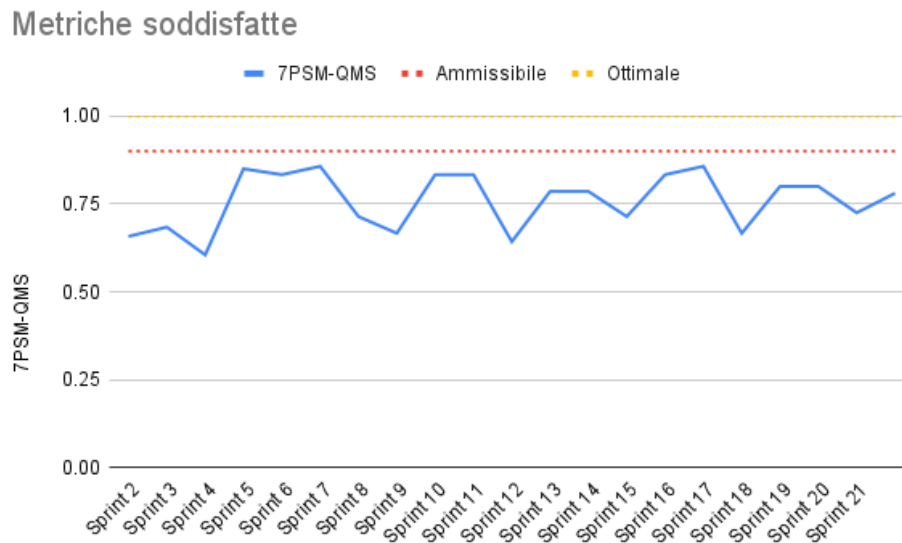


Figura 6: Metriche di qualità soddisfatte

Come evidenziato dal grafico, la quantità di metriche soddisfatte si colloca appena al di sotto del range ammissibile. Questo indica che, sebbene il progetto sia vicino a raggiungere gli standard richiesti, non ha ancora soddisfatto pienamente i criteri stabiliti. Tuttavia, è importante considerare che il calcolo delle Metriche di Qualità Soddisfatte include anche metriche relative alla qualità del codice e ai test effettuati su di esso. Poiché il codice non è ancora stato sviluppato in questa fase del progetto, è naturale che il valore complessivo del QMS risulti inferiore. Di conseguenza, il risultato attuale non rappresenta necessariamente una criticità, ma piuttosto una condizione prevista, destinata a migliorare man mano che lo sviluppo del codice procede e vengono introdotti i relativi test.

5.4 Qualità del processo di pianificazione

5.4.1 1POM-RSI - Requirements Stability Index

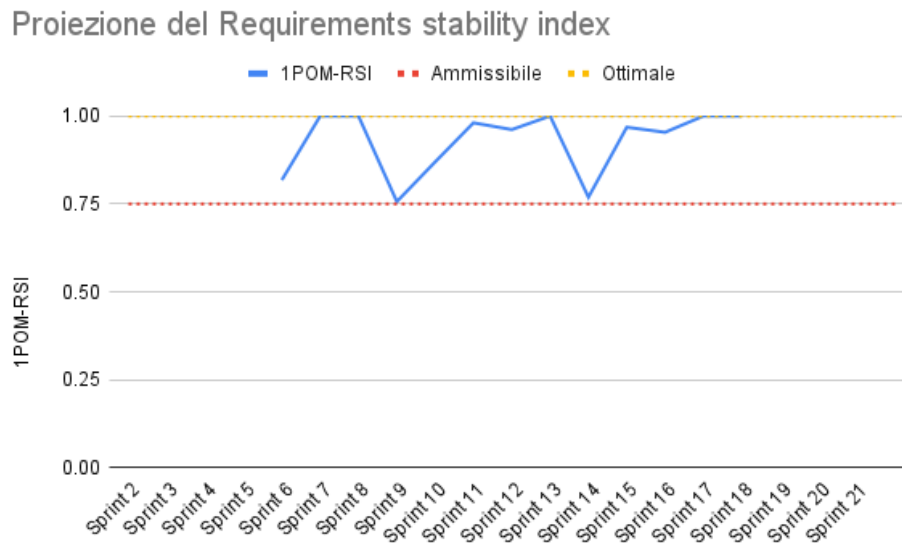


Figura 7: Metriche di qualità soddisfatte

Il Requirements Stability Index (RSI) è un indicatore che misura la stabilità dei requisiti di un progetto nel tempo. Un RSI elevato indica che i requisiti sono rimasti sostanzialmente invariati, mentre un valore basso suggerisce modifiche frequenti e potenzialmente destabilizzanti per lo sviluppo. L'andamento del grafico mostra che, a partire dalla fase di analisi, i requisiti hanno subito variazioni limitate e coerenti con le previsioni iniziali. Questo suggerisce un processo di gestione dei requisiti ben controllato, con revisioni e aggiornamenti minimi che non hanno compromesso la stabilità complessiva del progetto. L'oscillazione contenuta dell'indice evidenzia un buon livello di maturità nella definizione dei requisiti, riducendo il rischio di impatti negativi su tempi, costi e qualità del prodotto finale.

6 Iniziative di automiglioramento per la qualità

6.1 Introduzione

In questa sezione vengono descritte le azioni intraprese per migliorare la qualità del progetto *Software*. Ogni iniziativa è stata identificata attraverso l'esperienza accumulata durante lo sviluppo del progetto, mano a mano che emergono problematiche specifiche. Essendo questa la nostra prima esperienza con un progetto di tale complessità, è stato necessario affrontare numerosi tentativi per capire come organizzarci e gestire le diverse attività in modo efficace. Durante il percorso, siamo riusciti a riconoscere i punti di forza e le aree di miglioramento nel nostro lavoro, individuando così gli aspetti su cui focalizzarci per ottimizzare il processo.

6.2 Valutazione sull'organizzazione

Descrizione problema	Contromisura
Mancanza di tracciabilità delle attività rende difficile l'avanzamento produttivo e la pianificazione del lavoro	Implementazione di una Project Board su GitHub per migliorare la gestione e il monitoraggio delle attività
Attività che non venivano concluse per il tempo previsto rallentando il lavoro	Introduzione di una data di scadenza per ogni attività e creazione di tag specifici per categorizzare i problemi
Difficoltà nel comunicare la rendicontazione delle ore durante ogni sprint	Utilizzata la roadmap di <i>GitHub</i> per la costruzione dei diagrammi di Gantt, in modo tale da iniziare a prevedere il tempo per ogni singola attività e avere uno schema di riferimento
Difficoltà nel comunicare e organizzare le riunioni o eventi	Utilizzo di un calendario condiviso tramite l'uso di <i>Google Calendar</i> per pianificare le riunioni e le attività da svolgere
Diversi software di scrittura per stendere i documenti	Standardizzazione dell'uso di \LaTeX per la stesura dei documenti, e creazione di template per specifici documenti come i Verbali. Tutto al fine di migliorare la coerenza, la manutenibilità e la leggibilità della documentazione all'interno del progetto.

Tabella 19: Contromisure adottate per migliorare l'organizzazione

6.3 Valutazione sui Ruoli

Ruolo	Descrizione problema	Contromisura
Verificatore	Modificare o aggiornare documenti senza prima averli sottoposti a verifica non è conforme ai buoni standard progettuali	Il verificatore è tenuto a compilare ad ogni modifica la tabella relativa per ciascun documento

Tabella 20: Contromisure adottate per migliorare la gestione dei ruoli

6.4 Valutazione sugli Strumenti

Strumento	Descrizione problema	Contromisura
Glossario	Difficoltà nel riconoscere se una parola in un documento è stata definita nel glossario	Creazione di uno script Python che verifica la presenza di ogni parola definita nel glossario nei documenti.
Sito Web	Visitare il repository per leggere la documentazione risulta scomodo e macchinoso	Creazione di un sito web di facile accesso dove è possibile scaricare i pdf per la documentazione, fornendo la possibilità di visualizzare i documenti più piccoli direttamente sulle pagine.
LaTeX	Difficoltà nell'uso di questo linguaggio, soprattutto per i membri che non lo conoscevano	Ogni membro ha intrapreso un percorso di apprendimento per colmare le proprie lacune

Tabella 21: Contromisure adottate per migliorare il rapporto con gli strumenti

6.5 Considerazioni Finali

Il processo di automiglioramento è un'attività continua e fondamentale per garantire la qualità del progetto. Le contromisure adottate hanno permesso di risolvere o ridurre i problemi individuati, migliorando l'efficienza e l'efficacia del lavoro svolto. Il gruppo si impegna a mantenere un approccio proattivo e collaborativo per individuare e risolvere tempestivamente eventuali criticità. Continueremo a monitorare e valutare le nostre pratiche, implementando ulteriori miglioramenti ove necessario, per assicurare il successo del progetto e la soddisfazione degli stakeholder.