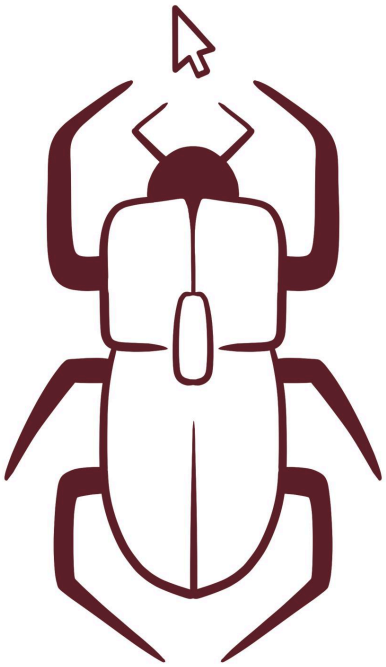


Piano di Qualifica

v0.8.1

Registro delle Modifiche

Data	Versione	Descrizione	Redattore	Verificatore
2026/01/27	0.8.1	Aggiunta grafico BV-SV, revisione Metodi di Testing e aggiunta test, modifiche minori al documento	Zago Alice	
2026/01/22	0.7.0	Grafici CPI-SPI, EAC, RSI e SGA	Zago Alice	
2026/01/21	0.6.0	Cruscotto di valutazione, grafico PV-AC-EV	Zago Alice	
2026/01/17	0.5.2	Revisione Automiglioramento	Suar Alberto	Zago Alice
2026/01/13	0.5.1	Rielaborazione introduzione documento e qualità di processo	Suar Alberto	Zago Alice
2025/12/02	0.5.0	Modifica tabelle qualità di processo, inserimento tabelle qualità di prodotto	Zago Alice	Suar Alberto
2025/12/30	0.4.0	Iniziati metodi di testing, inserimento tabelle test	Berengan Riccardo	Suar Alberto
2025/12/28	0.3.0	Processi secondari e processi organizzativi con tabelle soglie metriche, iniziata sezione automiglioramento e qualità di prodotto	Zago Alice	Suar Alberto
2025/12/27	0.2.0	Qualità di processo, processi primari	Zago Alice	Suar Alberto
2025/12/26	0.1.0	Inizio stesura documento, introduzione, scopo e riferimenti	Zago Alice	Suar Alberto
2025/12/23	0.0.0	Creazione documento	Zago Alice	Suar Alberto

Indice

Indice tabelle

1 Introduzione

1.1 Contesto del Progetto

Il presente documento descrive il Piano di Qualifica relativo al progetto Code Guardian, commissionato dall'azienda Var Group e realizzato dal gruppo di studenti Skarab Group nell'ambito del corso di Ingegneria del Software presso l'Università degli Studi di Padova.

L'obiettivo del progetto è lo sviluppo di una piattaforma ad Agenti per l'audit e la remediation automatizzata delle vulnerabilità nei repository GitHub, in conformità con quanto definito dal capitolato **C2**.

1.2 Finalità del Documento

Il Piano di Qualifica definisce l'impostazione metodologica per la gestione della qualità, specificando come il gruppo intenda prevenire, rilevare e correggere i difetti.

Il documento costituisce il riferimento primario per il Responsabile e per i Verificatori, strutturando gli obiettivi nelle seguenti macro-aree:

- **Piano della Qualità (Quality Assurance)**: definizione della strategia di gestione della qualità, identificando gli standard di riferimento (in particolare ISO/IEC 25010), le metriche di misurazione e le relative soglie di accettazione/ottimalità.
- **Controllo di Qualità (Quality Control)**: pianificazione operativa delle attività di Verifica (analisi statica, test dinamici) per garantire la correttezza tecnica degli artefatti prodotti.
- **Validazione di Prodotto**: definizione delle procedure necessarie per accertare la conformità del sistema rispetto alle aspettative degli Stakeholder e ai requisiti del capitolato.

TODO: PDSA, non PDCA

- **Miglioramento Continuo**: applicazione di meccanismi retroattivi (basati sul ciclo Plan-Do-Check-Act) che utilizzano i risultati delle misurazioni per ottimizzare i processi e il **Way of Working** in corso d'opera.

1.3 Traguardi Qualitativi

L'assicurazione della qualità segue l'approccio incrementale del progetto, fissando obiettivi specifici per le due principali milestones:

1.3.1 Revisione dei Requisiti e della Tecnologia (RTB)

Per la milestone RTB (**06/02/2026**), le attività di qualità si concentrano sulla correttezza formale e sulla fattibilità tecnica:

- **Qualità dei Documenti**: Verifica approfondita della documentazione (Analisi dei Requisiti, PdP, NdP) tramite analisi statica e walkthrough, per garantire assenza di ambiguità e coerenza interna (Indice di Gulpease).
- **Qualità del Prototipo (PoC)**: L'attività di verifica è focalizzata esclusivamente sulla **dimostrazione della fattibilità tecnica** (Technology Baseline), con particolare attenzione all'interazione Agenti-LLM. Il testing in questa fase ha valore *sperimentale e propedeutico*: esso funge da caso di studio per calibrare le metriche e validare le strategie di verifica che saranno poi applicate in modo sistematico ed estensivo sul MVP.

1.3.2 Revisione di Accettazione (Product Baseline – PB)

Per il rilascio finale (**21/03/2026**), il focus si sposta sulla robustezza, sulla copertura e sulla soddisfazione dei requisiti:

- **Qualità del Prodotto (MVP)**: Esecuzione completa dei test di unità, integrazione e sistema. Validazione finale rispetto ai requisiti funzionali e prestazionali del capitolato.

- **Qualità del Codice**: Rispetto dei vincoli di stile, assenza di **code smells** e raggiungimento delle soglie di copertura del codice (Code Coverage) definite nel presente piano.
- **Validazione Utente**: Verifica dell'usabilità tramite test di accettazione (UAT) basati sui casi d'uso principali.

1.4 Glossario

Al fine di prevenire ambiguità interpretative, è stato redatto un Glossario che definisce in modo univoco la terminologia tecnica, gli acronimi e i concetti di dominio utilizzati all'interno della documentazione. Le occorrenze dei termini presenti nel Glossario sono evidenziate nel testo mediante apposita formattazione.

Versione aggiornata del Glossario: Link al Glossario

1.5 Riferimenti

1.5.1 Riferimenti Normativi

I seguenti documenti hanno valore vincolante per la definizione delle strategie di qualità e per le attività di verifica:

- **Capitolato C2**: Piattaforma ad agenti per l'audit e la remediation dei repository software.
  https://www.math.unipd.it/~tullio/IS-1/2025/Progetto/C2.pdf
- **Norme di Progetto**: Il documento definisce il "Way of Working", stabilendo gli strumenti e le procedure che questo Piano si occupa di misurare.
  Documento interno

1.5.2 Riferimenti Informativi

- **ISO/IEC 25010:2011**: Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE).
  https://iso25000.com/index.php/en/iso-25000-standards/iso-25010
- **ISO/IEC 12207:2008**: Systems and software engineering — Software life cycle processes.
  https://ieeexplore.ieee.org/document/4475826
- **Dispense del corso di Ingegneria del Software – Qualità del software**
  https://www.math.unipd.it/~tullio/IS-1/2025/Dispense/T06.pdf

1.6 Qualità di Processo

La garanzia della qualità del prodotto finale è intrinsecamente legata alla qualità dei processi produttivi che lo generano. Per il progetto Code Guardian, la gestione dei processi mira a rendere il Way of Working sostenibile, tracciabile e soggetto a miglioramento continuo attraverso l'applicazione del ciclo PDCA.

1.6.1 Centralizzazione delle Metriche e Obiettivi

Il presente documento costituisce il riferimento unico, autoritativo e analitico per la gestione della qualità del progetto Code Guardian. Mentre le Norme di Progetto definiscono le procedure operative, gli strumenti e le responsabilità per l'estrazione dei dati, il Piano di Qualifica ha il compito di centralizzare la "scienza della misurazione" del gruppo.

In particolare, ogni metrica qui esposta è corredata da:

- **Identificativo univoco:** (MPC per il processo, MPD per il prodotto);
- **Formulazione matematica:** Per garantire l'oggettività del calcolo;
- **Soglie di Valutazione:** Distinte in "Accettabilità" (requisito minimo per la validazione) e "Ottimalità" (target di eccellenza desiderato).

Ogni scostamento rilevato tra i valori misurati e le soglie qui definite viene analizzato durante le retrospettive di fine Sprint. Tali evidenze costituiscono la base oggettiva per l'attivazione di azioni correttive o per la ricalibrazione delle soglie stesse, garantendo che il processo di qualità evolva insieme alla maturità del team.

1.6.2 Processi Primari: Fornitura e Sviluppo

Questi processi definiscono le attività core per la realizzazione del software. Il monitoraggio si focalizza sul rispetto dei vincoli di tempo e budget (tramite la metodologia EVM) e sulla gestione rigorosa dell'ambito di progetto.

ID	Nome	Formula	V. Accettabile	V. Ottimo
MPC01	Budget at Completion (BAC)	Preventivo	Preventivo	Preventivo
MPC02	Planned Value (PV)	PV	≥ 0	Da Piano
MPC03	Actual Cost (AC)	AC	$\leq EAC$	$\leq EV$
MPC04	Earned Value (EV)	EV	$\geq 90\% PV$	$\geq PV$
MPC05	Budget Variance (BV)	$BV = BAC - EAC$	≥ 0	> 0
MPC06	Schedule Variance (SV)	$SV = EV - PV$	$> -10\%$ BAC	≥ 0
MPC07	Cost Performance Index (CPI)	$CPI = \frac{EV}{AC}$	$0.90 \leq v \leq 1.10$	1.00
MPC08	Schedule Performance Index (SPI)	$SPI = \frac{EV}{PV}$	$0.90 \leq v \leq 1.10$	1.00
MPC09	Estimate at Completion (EAC)	$EAC = \frac{BAC}{CPI}$	$\leq BAC + 5\%$	$\leq BAC$

Table 1: Soglie metriche per il processo di Fornitura (EVM)

Riferimento: Norme di Progetto, Sezione [Stabilità dei Requisiti] Il monitoraggio della stabilità dei requisiti è cruciale per prevenire lo **scope creep**, specialmente a seguito delle revisioni correttive post-S2.

ID	Nome	Formula	V. Accettabile	V. Ottimo
MPC10	Requirements Stability Index	$RSI = \frac{R_{\text{tot}} - \Delta R}{R_{\text{tot}}} \times 100$	$\geq 75\%$	100%

Table 2: Soglie metriche per il processo di Sviluppo

1.6.3 Processi di Supporto

I processi di supporto garantiscono l'integrità e la verificabilità degli artefatti. La leggibilità della documentazione (Indice di Gulpease) e la copertura dei test sono i parametri cardine per assicurare la manutenibilità futura.

ID	Nome	Formula	V. Accettabile	V. Ottimo
MPC11	Gulpease Index	$89 + \frac{300(L_f) - 10(L_p)}{F_p}$	≥ 40	≥ 60
MPC12	Correttezza Ortografica	Errori segnalati	0	0
MPC13	Code Coverage	$\frac{\text{Linee coperte}}{\text{Linee totali}} \times 100$	$\geq 70\%$	$\geq 80\%$
MPC14	Test Success Rate	$\frac{\text{Passati}}{\text{Eseguiti}} \times 100$	100%	100%

Table 3: Soglie metriche Documentazione e Verifica

1.6.4 Processi Organizzativi

Misurano l'efficienza interna del team Skarab Group nell'auto-organizzarsi e nel rispettare gli impegni presi durante gli Sprint.

ID	Nome	Formula	V. Accettabile	V. Ottimo
MPC15	Metrics Satisfaction	$\frac{\text{Metriche OK}}{\text{Metriche Tot}} \times 100$	$\geq 90\%$	100%
MPC16	Sprint Goal Achievement	$\frac{\text{Completati}}{\text{Pianificati}} \times 100$	$\geq 80\%$	100%

Table 4: Soglie metriche Organizzative

1.7 Qualità di Prodotto

La qualità di prodotto valuta il software consegnato rispetto ai requisiti e alle caratteristiche intrinseche definite dallo standard ISO/IEC 25010.

1.7.1 Adeguatezza Funzionale e Affidabilità

Si misura la capacità del sistema di svolgere i compiti richiesti e di rimanere operativo senza guasti critici, parametro fondamentale per un tool di audit.

ID	Nome	Formula	V. Accettabile	V. Ottimo
MPD01	Copertura Req. Obbligatori	$\frac{\text{Soddisfatti}}{\text{Totale Obbl.}} \times 100$	100%	100%
MPD04	Failure Density	$\frac{\text{N. guasti}}{\text{KLOC}}$	≤ 0.5	0
MPD05	Availability	$\frac{\text{Tempo Up}}{\text{Tempo Tot}} \times 100$	$\geq 98\%$	$\geq 99.9\%$

Table 5: Metriche Adeguatezza e Affidabilità

1.7.2 Manutenibilità e Sicurezza

Data la natura del progetto Code Guardian, queste metriche rappresentano il valore distintivo del prodotto. Un codice manutenibile e privo di vulnerabilità è condizione necessaria per l'accettazione.

ID	Nome	Formula	V. Accettabile	V. Ottimo
MPD08	Comment Density	$\frac{\text{Linee commento}}{\text{Linee codice}} \times 100$	$\geq 15\%$	$20\% - 25\%$
MPD09	Cyclomatic Complexity	$V(G)$	≤ 15	≤ 10
MPD10	Coupling (Fan-out)	Dipendenze esterne	≤ 6	≤ 3
MPD11	Vulnerability Detection	N. vulnerabilità critiche	0	0

Table 6: Metriche Manutenibilità e Sicurezza

TODO: Svolgere una revisione approfondita delle metodologie di testing, facendo ovviamente riferimento alla sezione di descrizione delle NdP

2 Metodi di Testing

Questa sezione definisce la strategia di testing per il progetto *CodeGuardian*. Skarab Group ha adottato un approccio di testing multilivello che copre:

- **Test di Sistema (TS)**.
- **Test di Unità (TU)**.
- **Test di Accettazione (TA)**.

2.1 Convenzioni di Nomenclatura

2.2 Test di Sistema

Di seguito, viene riportata la tabella che definisce i Test di Sistema (TS) necessari per validare il comportamento descritto nei Casi d'Uso.

ID Test	Descrizione	Tipo	UC Riferimento
TS-1	Verifica procedura di registrazione completa con dati validi.	Funzionale	UC1
TS-1.1	Verifica validazione formato Username: lunghezza e caratteri ammessi.	Validazione	UC1.1
TS-1.2	Verifica unicità Username: tentativo di registrazione con username già esistente.	Sicurezza	UC1.1
TS-1.3	Verifica validazione formato Email: sintassi e dominio valido.	Validazione	UC1.2
TS-1.4	Verifica unicità Email: tentativo di registrazione con email già presente.	Sicurezza	UC1.2
TS-1.5	Verifica complessità Password: rispetto dei criteri di sicurezza.	Sicurezza	UC1.3
TS-2	Verifica login con credenziali corrette.	Funzionale	UC2
TS-2.1	Verifica gestione errore per Username non esistente o errato.	Sicurezza	UC2.1
TS-2.2	Verifica gestione errore per Password errata.	Sicurezza	UC2.2

ID Test	Descrizione	Tipo	UC Riferimento
TS-2.3	Verifica validazione formato input in fase di login.	Validazione	UC2.1, UC2.2
TS-3	Verifica flusso completo OAuth con GitHub.	Integrazione	UC3
TS-3.1	Verifica rifiuto collegamento da parte dell'utente.	Funzionale	UC3.1
TS-3.2	Verifica gestione errore codice GitHub mancante o non valido.	Integrazione	UC3.2
TS-3.3	Verifica gestione errore codice GitHub già associato ad altro utente.	Sicurezza	UC3.2
TS-4	Verifica invio richiesta analisi con URL valido e opzioni selezionate.	Funzionale	UC4
TS-4.1	Verifica validazione URL Repository: formato e dominio GitHub.	Validazione	UC4.1
TS-4.2	Verifica accessibilità Repository: URL privato o inesistente.	Integrazione	UC4.1
TS-4.3	Verifica obbligatorietà selezione aree di interesse.	Validazione	UC4.2
TS-4.4	Verifica blocco analisi per report già up-to-date.	Ottimizzazione	UC4.3
TS-4.5	Verifica blocco analisi concorrente già in corso.	Stato	UC4.3
TS-5	Verifica visualizzazione lista report e apertura dashboard di dettaglio.	Funzionale	UC5
TS-5.1	Verifica gestione caso "Nessun report disponibile".	Funzionale	UC5.2
TS-5.2	Verifica selezione dati specifici da visualizzare.	Funzionale	UC5.3
TS-5.3	Verifica rendering completo report con tutti i dati selezionati.	UI/UX	UC5.4

ID Test	Descrizione	Tipo	UC Riferimento
TS-6	Verifica filtro temporale sui report passati con intervallo valido.	Funzionale	UC6
TS-6.1	Verifica errore per mancata selezione intervallo temporale.	Validazione	UC6.1
TS-6.2	Verifica errore intervallo temporale incoerente.	Validazione	UC6.2
TS-6.3	Verifica errore intervallo temporale troppo ampio.	Validazione	UC6.2
TS-6.4	Verifica messaggio “Nessun report nel periodo selezionato”.	Funzionale	UC6.2
TS-7	Verifica rendering grafico comparativo tra report.	UI/UX	UC7
TS-7.1	Verifica interazione con grafico per dettagli specifici.	UI/UX	UC7
TS-8	Verifica rendering tabella comparativa con indicatori di variazione.	UI/UX	UC8
TS-9	Verifica presenza sezioni Analisi Codice (Statica, Dipendenze, OWASP).	Contenuto	UC9
TS-9.1	Verifica correttezza conteggio vulnerabilità totali.	Logica	UC9.4
TS-9.2	Verifica visualizzazione analisi statica del codice.	Contenuto	UC9.1
TS-9.3	Verifica visualizzazione analisi librerie e dipendenze.	Contenuto	UC9.2
TS-9.4	Verifica visualizzazione report sicurezza OWASP.	Contenuto	UC9.3
TS-10	Verifica presenza sezioni Analisi Documentazione.	Contenuto	UC10
TS-10.1	Verifica rilevamento errori di spelling.	Contenuto	UC10.1

ID Test	Descrizione	Tipo	UC Riferimento
TS-10.2	Verifica calcolo completezza documentazione.	Logica	UC10.2
TS-11	Verifica visualizzazione numero totale vulnerabilità repository.	Contenuto	UC11
TS-12	Verifica coerenza Metadati Report (data, commit, richiedente).	Integrazione	UC12
TS-12.1	Verifica visualizzazione data report.	Contenuto	UC12.1
TS-12.2	Verifica visualizzazione commit analizzati.	Contenuto	UC12.2
TS-12.3	Verifica visualizzazione richiedente.	Contenuto	UC12.3
TS-13	Verifica disconnessione account GitHub con conferma.	Funzionale	UC13
TS-13.1	Verifica richiesta conferma prima della disconnessione.	Sicurezza	UC13.1
TS-14	Verifica esportazione report in formati supportati.	Funzionale	UC14
TS-14.1	Verifica errore per mancata selezione formato.	Validazione	UC14.1
TS-14.2	Verifica generazione file dopo conferma.	Funzionale	UC14.2
TS-15	Verifica modifica password con validazione corretta.	Funzionale	UC15
TS-15.1	Verifica errore per password corrente mancante o errata.	Sicurezza	UC15.1
TS-15.2	Verifica errore per nuova password non conforme.	Validazione	UC15.2
TS-15.3	Verifica errore per nuova password uguale alla precedente.	Validazione	UC15.2
TS-15.4	Verifica ricezione conferma modifica avvenuta.	Funzionale	UC15.4

ID Test	Descrizione	Tipo	UC Riferimento
TS-16	Verifica visualizzazione suggerimenti di remediation.	Funzionale	UC16
TS-16.1	Verifica gestione caso “Nessuna issue identificata”.	Funzionale	UC16.1
TS-16.2	Verifica visualizzazione dettaglio singola issue.	Contenuto	UC16.2
TS-17	Verifica creazione ambiente sandbox per analisi.	Integrazione	UC17
TS-17.1	Verifica gestione errore durante creazione sandbox.	Integrazione	UC17.1
TS-18	Verifica lettura richieste utente da parte orchestratore.	Integrazione	UC18
TS-18.1	Verifica esecuzione analisi completa quando richiesta.	Funzionale	UC18.1
TS-18.2	Verifica processamento richieste specifiche su aree.	Funzionale	UC18.2
TS-19	Verifica analisi vulnerabilità dipendenze con remediation.	Sicurezza	UC19
TS-19.1	Verifica accettazione remediation proposte.	Funzionale	UC19.1
TS-19.2	Verifica rifiuto remediation proposte.	Funzionale	UC19.2
TS-20	Verifica rilevamento segreti e token esposti.	Sicurezza	UC20
TS-20.1	Verifica rifiuto remediation segreti.	Funzionale	UC20.1
TS-20.2	Verifica revoca automatica se integrata con provider.	Integrazione	UC20.2
TS-20.3	Verifica visualizzazione risultati rilevamento.	Contenuto	UC20.3
TS-21	Verifica conformità licenze dipendenze.	Compliance	UC21

ID Test	Descrizione	Tipo	UC Riferimento
TS-21.1	Verifica integrazione con processo approvazione legale.	Integrazione	UC21.1
TS-22	Verifica revisione PR automatizzata.	Integrazione	UC22
TS-22.1	Verifica esecuzione test automatici.	Funzionale	UC22.1
TS-22.2	Verifica suggerimenti modifica automatici (codemods).	Funzionale	UC22.2
TS-23	Verifica monitor qualità codice con metriche.	Qualità	UC23
TS-23.1	Verifica integrazione con tool metriche esterni.	Integrazione	UC23.1
TS-23.2	Verifica suggerimenti KPI e obiettivi qualità.	Qualità	UC23.2
TS-24	Verifica suggerimenti refactoring codice.	Qualità	UC24
TS-24.1	Verifica impatto tramite test automatizzati.	Qualità	UC24.1
TS-24.2	Verifica applicazione automatica sotto supervisione.	Funzionale	UC24.2
TS-24.3	Verifica visualizzazione suggerimenti.	Contenuto	UC24.3
TS-25	Verifica generazione changelog e release notes.	Funzionale	UC25
TS-25.1	Verifica rilevamento breaking changes.	Funzionale	UC25.1
TS-25.2	Verifica pubblicazione automatica su GitHub Release.	Integrazione	UC25.2
TS-25.3	Verifica visualizzazione e approvazione changelog.	Funzionale	UC25.3
TS-26	Verifica analisi test e coverage.	Qualità	UC26
TS-26.1	Verifica replay test intermittenti.	Qualità	UC26.1

ID Test	Descrizione	Tipo	UC Riferimento
TS-26.2	Verifica suggerimenti per test addizionali.	Qualità	UC26.2
TS-26.3	Verifica visualizzazione report test/ coverage.	Contenuto	UC26.3
TS-27	Verifica applicazione policy CI/CD pre-merge.	Compliance	UC27
TS-27.1	Verifica gestione eccezioni approvate manualmente.	Compliance	UC27.1
TS-27.2	Verifica policy dinamiche per branch differenti.	Compliance	UC27.2
TS-27.3	Verifica visualizzazione risultati policy.	Contenuto	UC27.3
TS-28	Verifica generazione report programmabili e alert.	Funzionale	UC28
TS-28.1	Verifica filtri e template report.	Funzionale	UC28.1
TS-28.2	Verifica azioni automatiche su alert critici.	Funzionale	UC28.2
TS-28.3	Verifica visualizzazione report programmati.	Contenuto	UC28.3
TS-29	Verifica recupero e avvio tool esterni di analisi.	Integrazione	UC29
TS-29.1	Verifica gestione tool esterno non disponibile.	Integrazione	UC29.1
TS-29.2	Verifica richiesta analisi del codice a tool esterno.	Integrazione	UC29.2
TS-29.3	Verifica richiesta analisi documentazione a tool esterno.	Integrazione	UC29.3
TS-29.4	Verifica richiesta analisi OWASP a tool esterno.	Integrazione	UC29.4
TS-30	Verifica generazione report finale completo.	Integrazione	UC30

ID Test	Descrizione	Tipo	UC Riferimento
TS-30.1	Verifica integrazione analisi singole nel report.	Integrazione	UC30.1
TS-31	Verifica trasferimento report a sistema frontend.	Integrazione	UC31
TS-32	Verifica notifica utente disponibilità nuovo report.	Funzionale	UC32
TS-34	Verifica notifica completamento analisi al frontend.	Integrazione	UC34
TS-34.1	Verifica retry invio messaggio completamento.	Integrazione	UC34.1
TS-35	Verifica gestione errore critico durante analisi.	Integrazione	UC35
TS-35.1	Verifica retry invio messaggio fallimento.	Integrazione	UC35.1
TS-36	Verifica salvataggio metadati repository.	Persistenza	UC36
TS-37	Verifica esistenza repository analizzata.	Funzionale	UC37
TS-37.1	Verifica gestione caso nessuna repository analizzata.	Funzionale	UC37.1
TS-38	Verifica salvataggio report analisi nel database.	Persistenza	UC38
TS-38.1	Verifica gestione errore durante salvataggio.	Persistenza	UC38.1
TS-39	Verifica salvataggio metriche aggregate.	Persistenza	UC39
TS-39.1	Verifica gestione errore durante salvataggio metriche.	Persistenza	UC39.1
TS-40	Verifica invio credenziali al sistema backend.	Integrazione	UC40
TS-40.1	Verifica gestione errore durante trasferimento.	Integrazione	UC40.1

ID Test	Descrizione	Tipo	UC Riferimento
TS-41	Verifica gestione codice OAuth GitHub.	Integrazione	UC41
TS-41.1	Verifica gestione errore durante scambio codice.	Integrazione	UC41.1

Table 7: Tabella dei Test di Sistema

2.3 Test di Unità

In questa sezione vengono definiti i Test di Unità volti a verificare il corretto funzionamento delle singole componenti software, basati sui requisiti funzionali identificati.

ID Test	Requisito Riferimento	Descrizione	Risultato Atteso
TU-1.1	FR1	Verifica rendering pagina di registrazione.	La pagina viene visualizzata correttamente con tutti i campi.
TU-1.2	FR3	Verifica inserimento username nel campo dedicato.	Il valore inserito viene accettato e memorizzato.
TU-1.3	FR6, FR7	Verifica validatore Username: lunghezza minima 4 e massima 20 caratteri.	La funzione restituisce `false` per valori fuori range.
TU-1.4	FR8	Verifica validatore Username: solo caratteri alfanumerici.	La funzione restituisce `false` per caratteri speciali.
TU-1.5	FR9	Verifica messaggio errore per username non valido.	Il messaggio di errore viene visualizzato correttamente.
TU-1.6	FR10	Verifica controllo unicità username nel database.	La funzione restituisce `false` se username già esistente.
TU-1.7	FR4	Verifica inserimento email nel campo dedicato.	Il valore inserito viene accettato e memorizzato.
TU-1.8	FR11, FR12	Verifica validatore Email: presenza '@' e dominio valido.	La funzione restituisce `false` per email malformate.
TU-1.9	FR13	Verifica messaggio errore per email non valida.	Il messaggio di errore viene visualizzato correttamente.
TU-1.10	FR14	Verifica controllo unicità email nel database.	La funzione restituisce `false` se email già esistente.
TU-1.11	FR5	Verifica inserimento password nel campo dedicato.	Il valore inserito viene accettato e memorizzato (mascherato).
TU-1.12	FR15	Verifica validatore Password: lunghezza minima 8 caratteri.	La funzione restituisce `false` per password troppo corte.

ID Test	Requisito Riferimento	Descrizione	Risultato Atteso
TU-1.13	FR16, FR17	Verifica validatore Password: presenza maiuscola e minuscola.	La funzione restituisce `false` se mancano lettere maiuscole o minuscole.
TU-1.14	FR18, FR19	Verifica validatore Password: presenza numero e carattere speciale.	La funzione restituisce `false` se mancano numeri o caratteri speciali.
TU-1.15	FR20	Verifica messaggio errore per password non conforme.	Il messaggio indica specificamente i criteri mancanti.
TU-1.16	FR2	Verifica invio richiesta registrazione tramite pulsante.	La richiesta viene inviata al backend con i dati corretti.
TU-2.1	FR21	Verifica rendering pagina di autenticazione.	La pagina viene visualizzata con campi username e password.
TU-2.2	FR22	Verifica inserimento username per autenticazione.	Il valore inserito viene accettato.
TU-2.3	FR23	Verifica inserimento password per autenticazione.	Il valore inserito viene accettato (mascherato).
TU-2.4	FR24	Verifica validazione lunghezza username (4-20 caratteri).	La funzione restituisce `false` per valori fuori range.
TU-2.5	FR25	Verifica messaggio errore username non conforme.	Il messaggio di errore viene visualizzato.
TU-2.6	FR26	Verifica controllo esistenza username nel database.	La funzione restituisce `false` se username non esiste.
TU-2.7	FR27	Verifica messaggio errore password non conforme.	Il messaggio di errore viene visualizzato.
TU-2.8	FR28	Verifica controllo correttezza password.	La funzione verifica l'hash e restituisce il risultato corretto.
TU-3.1	FR29	Verifica accesso sezione collegamento GitHub.	La sezione viene visualizzata correttamente.

ID Test	Requisito Riferimento	Descrizione	Risultato Atteso
TU-3.2	FR30	Verifica visualizzazione avviso redirect a GitHub.	L'avviso viene mostrato prima del redirect.
TU-3.3	FR31, FR32	Verifica gestione conferma/ rifiuto collegamento.	Il sistema procede o annulla in base alla scelta.
TU-3.4	FR33	Verifica processo OAuth completo con GitHub.	Il codice viene ricevuto e processato correttamente.
TU-3.5	FR34	Verifica gestione errore codice GitHub non ricevuto.	Il messaggio di errore viene visualizzato.
TU-3.6	FR35	Verifica controllo codice già associato ad altro utente.	La funzione restituisce `false` se codice duplicato.
TU-3.7	FR36	Verifica validazione formato codice GitHub.	La funzione restituisce `false` per formato non valido.
TU-4.1	FR37	Verifica accesso sezione richiesta analisi.	La sezione viene visualizzata correttamente.
TU-4.2	FR38	Verifica inserimento URL repository nel campo.	Il valore inserito viene accettato.
TU-4.3	FR39, FR40	Verifica validatore URL: protocollo https e dominio github.com.	La funzione restituisce `false` per URL non validi.
TU-4.4	FR41	Verifica messaggio errore URL non conforme.	Il messaggio di errore viene visualizzato.
TU-4.5	FR42	Verifica controllo accessibilità repository.	La funzione verifica l'esistenza e l'accesso al repository.
TU-4.6	FR43	Verifica messaggio errore URL mancante.	Il messaggio viene mostrato se campo vuoto.
TU-4.7	FR44	Verifica selezione aree di interesse.	Le aree selezionate vengono memorizzate.
TU-4.8	FR45	Verifica messaggio errore nessuna area selezionata.	Il messaggio viene visualizzato.
TU-4.9	FR46	Verifica invio richiesta analisi tramite pulsante.	La richiesta viene inviata al backend.

ID Test	Requisito Riferimento	Descrizione	Risultato Atteso
TU-4.10	FR47	Verifica controllo report up-to-date.	La funzione confronta hash commit e restituisce il risultato.
TU-4.11	FR48	Verifica controllo analisi in corso.	La funzione verifica lo stato e restituisce il risultato.
TU-5.1	FR49	Verifica accesso sezione visualizzazione report.	La sezione viene visualizzata correttamente.
TU-5.2	FR50	Verifica rendering elenco report disponibili.	L'elenco viene popolato con i report esistenti.
TU-5.3	FR51	Verifica selezione report dall'elenco.	Il report selezionato viene caricato.
TU-5.4	FR52	Verifica messaggio nessun report disponibile.	Il messaggio viene visualizzato se lista vuota.
TU-5.5	FR53	Verifica messaggio errore nessun report selezionato.	Il messaggio viene mostrato.
TU-5.6	FR54	Verifica selezione dati specifici da visualizzare.	Le opzioni selezionate vengono memorizzate.
TU-5.7	FR55	Verifica messaggio errore nessun dato selezionato.	Il messaggio viene visualizzato.
TU-5.8	FR56	Verifica rendering completo dettagli analisi.	Tutti i dati selezionati vengono visualizzati correttamente.
TU-6.1	FR57	Verifica selezione intervallo temporale.	L'intervallo viene memorizzato correttamente.
TU-6.2	FR58	Verifica conferma selezione intervallo.	L'azione di conferma viene registrata.
TU-6.3	FR59	Verifica modifica intervallo dopo selezione.	Il nuovo intervallo sovrascrive il precedente.
TU-6.4	FR60	Verifica messaggio errore intervallo non selezionato.	Il messaggio viene visualizzato.
TU-6.5	FR61	Verifica validazione intervallo temporale.	La funzione verifica coerenza e ampiezza.

ID Test	Requisito Riferimento	Descrizione	Risultato Atteso
TU-6.6	FR62	Verifica messaggio nessun report nel periodo.	Il messaggio viene visualizzato se nessun risultato.
TU-6.7	FR63	Verifica controllo incoerenza intervallo (Start > End).	La funzione restituisce `false` per intervalli incoerenti.
TU-6.8	FR64	Verifica controllo ampiezza massima intervallo.	La funzione restituisce `false` se intervallo troppo ampio.
TU-7.1	FR65	Verifica rendering grafico comparativo.	Il grafico viene visualizzato con i dati corretti.
TU-7.2	FR66	Verifica interazione con punti dati grafico.	I dettagli vengono mostrati al click/hover.
TU-7.3	FR67	Verifica rendering tabella comparativa.	La tabella viene popolata correttamente.
TU-7.4	FR68	Verifica calcolo indicatori variazione.	Gli indicatori mostrano diff corretto rispetto report precedente.
TU-9.1	FR69	Verifica visualizzazione sezione analisi codice.	La sezione viene renderizzata correttamente.
TU-9.2	FR70	Verifica rendering report analisi statica.	I dati dell'analisi statica vengono mostrati.
TU-9.3	FR71	Verifica rendering analisi librerie/dipendenze.	L'elenco dipendenze viene visualizzato.
TU-9.4	FR72	Verifica rendering report sicurezza OWASP.	I risultati OWASP vengono mostrati.
TU-9.5	FR73	Verifica calcolo totale vulnerabilità codice.	Il conteggio corrisponde alla somma delle vulnerabilità.
TU-10.1	FR74	Verifica visualizzazione sezione analisi documentazione.	La sezione viene renderizzata correttamente.
TU-10.2	FR75	Verifica visualizzazione errori spelling.	Gli errori rilevati vengono elencati.

ID Test	Requisito Riferimento	Descrizione	Risultato Atteso
TU-10.3	FR76	Verifica calcolo completezza documentazione.	La percentuale viene calcolata correttamente.
TU-11.1	FR77	Verifica visualizzazione vulnerabilità totali repository.	Il numero totale viene calcolato e mostrato.
TU-12.1	FR78	Verifica visualizzazione area metadati.	La sezione metadati viene renderizzata.
TU-12.2	FR79	Verifica formattazione data report.	La data viene visualizzata nel formato corretto.
TU-12.3	FR80	Verifica visualizzazione commit analizzati.	L'elenco commit viene mostrato.
TU-12.4	FR81	Verifica visualizzazione richiedente.	Il nome utente richiedente viene mostrato.
TU-13.1	FR82	Verifica processo disconnessione account GitHub.	Il collegamento viene rimosso correttamente.
TU-13.2	FR83	Verifica azione pulsante Disconnetti.	Il pulsante avvia il processo di disconnessione.
TU-13.3	FR84	Verifica conferma disconnessione.	La conferma viene richiesta e processata.
TU-14.1	FR85	Verifica processo esportazione report.	Il report viene esportato correttamente.
TU-14.2	FR86	Verifica selezione formato esportazione.	Il formato selezionato viene applicato.
TU-14.3	FR87	Verifica messaggio errore formato non selezionato.	Il messaggio viene visualizzato.
TU-14.4	FR88	Verifica conferma esportazione.	Il file viene generato dopo conferma.
TU-15.1	FR89	Verifica accesso sezione modifica password.	La sezione viene visualizzata.
TU-15.2	FR90	Verifica inserimento password corrente.	Il valore inserito viene accettato (mascherato).

ID Test	Requisito Riferimento	Descrizione	Risultato Atteso
TU-15.3	FR91	Verifica messaggio errore password corrente mancante.	Il messaggio viene visualizzato.
TU-15.4	FR92	Verifica controllo correttezza password corrente.	La funzione verifica l'hash correttamente.
TU-15.5	FR93	Verifica inserimento nuova password.	Il valore inserito viene accettato (mascherato).
TU-15.6	FR94	Verifica messaggio errore nuova password mancante.	Il messaggio viene visualizzato.
TU-15.7	FR95	Verifica validazione conformità nuova password.	La funzione applica tutti i criteri di sicurezza.
TU-15.8	FR96	Verifica controllo uguaglianza password nuova/precedente.	La funzione restituisce `false` se identiche.
TU-15.9	FR97	Verifica conferma modifica password.	La password viene aggiornata nel database.
TU-15.10	FR98	Verifica messaggio conferma modifica avvenuta.	Il messaggio di successo viene visualizzato.
TU-16.1	FR99	Verifica visualizzazione suggerimenti remediation.	La sezione viene renderizzata correttamente.
TU-16.2	FR100	Verifica rendering lista issue identificate.	L'elenco issue viene popolato.
TU-16.3	FR101	Verifica messaggio nessuna issue identificata.	Il messaggio viene visualizzato se lista vuota.
TU-16.4	FR102	Verifica visualizzazione dettaglio remediation.	I dettagli vengono mostrati correttamente.
TU-17.1	FR103	Verifica creazione ambiente sandbox.	L'ambiente viene creato correttamente via Docker.
TU-17.2	FR104	Verifica intercettazione errori creazione sandbox.	Gli errori vengono catturati e gestiti.
TU-17.3	FR105	Verifica comunicazione errori al frontend.	Il messaggio di errore viene trasmesso.

ID Test	Requisito Riferimento	Descrizione	Risultato Atteso
TU-18.1	FR106	Verifica lettura richieste utente.	Le richieste vengono parsate correttamente.
TU-18.2	FR107	Verifica esecuzione analisi completa.	Tutti i moduli di analisi vengono attivati.
TU-18.3	FR108	Verifica processamento richieste specifiche.	Solo i moduli richiesti vengono attivati.
TU-18.4	FR109	Verifica analisi completa per repository nuova.	L'analisi completa viene forzata.
TU-19.1	FR110	Verifica analisi vulnerabilità dipendenze.	Le vulnerabilità vengono rilevate correttamente.
TU-19.2	FR111	Verifica accettazione remediation vulnerabilità.	Le remediation vengono applicate.
TU-19.3	FR112	Verifica rifiuto remediation.	Le remediation vengono scartate.
TU-20.1	FR113	Verifica rilevamento segreti e token.	I segreti esposti vengono identificati.
TU-20.2	FR114	Verifica rifiuto remediation segreti.	Le remediation vengono scartate.
TU-20.3	FR115	Verifica revoca automatica segreti.	La revoca viene eseguita se integrazione attiva.
TU-20.4	FR116	Verifica visualizzazione risultati rilevamento.	I risultati vengono mostrati correttamente.
TU-24.1	FR117	Verifica suggerimenti refactoring.	I suggerimenti vengono generati correttamente.
TU-24.2	FR118	Verifica applicazione automatica refactor.	Il refactor viene applicato sotto supervisione.
TU-24.3	FR119	Verifica visualizzazione suggerimenti refactoring.	I suggerimenti vengono mostrati.
TU-25.1	FR120	Verifica generazione changelog.	Il changelog viene creato correttamente.

ID Test	Requisito Riferimento	Descrizione	Risultato Atteso
TU-25.2	FR121	Verifica rilevamento breaking changes.	Le breaking changes vengono segnalate.
TU-25.3	FR122	Verifica pubblicazione automatica GitHub Release.	La release viene pubblicata.
TU-25.4	FR123	Verifica visualizzazione e approvazione changelog.	Il changelog può essere revisionato e approvato.
TU-26.1	FR124	Verifica analisi test e coverage.	I dati di coverage vengono calcolati.
TU-26.2	FR125	Verifica riesecuzione test intermittenti.	I test flaky vengono rieseguiti.
TU-26.3	FR126	Verifica suggerimenti test addizionali.	I gap di coverage vengono identificati.
TU-26.4	FR127	Verifica visualizzazione report test/coverage.	Il report viene mostrato correttamente.
TU-27.1	FR128	Verifica applicazione policy CI/CD.	Le policy vengono verificate.
TU-27.2	FR129	Verifica gestione eccezioni manuali.	Le eccezioni vengono processate.
TU-27.3	FR130	Verifica policy dinamiche per branch.	Policy diverse vengono applicate per branch.
TU-27.4	FR131	Verifica visualizzazione risultati policy.	I risultati vengono mostrati.
TU-28.1	FR132	Verifica generazione report programmabili.	I report vengono generati automaticamente.
TU-28.2	FR133	Verifica configurazione filtri e template.	Filtri e template vengono applicati.
TU-28.3	FR134	Verifica azioni automatiche su alert.	Le azioni vengono eseguite.
TU-28.4	FR135	Verifica visualizzazione report programmati.	I report vengono mostrati.

Table 8: Tabella dei Test di Unità

2.4 Test di Accettazione

In questa sezione vengono definiti i Test di Accettazione, volti a validare il sistema rispetto ai requisiti utente e agli scenari d'uso reali.

ID Test	UC Riferimento	Descrizione	Risultato Atteso

Table 9: Tabella dei Test di Accettazione

3 Cruscotto di Valutazione

Il presente cruscotto costituisce il sistema di monitoraggio attraverso il quale Skarab Group valuta oggettivamente l'andamento del progetto. Le metriche qui raccolte rappresentano l'evidenza empirica necessaria per attivare il ciclo *PDCA* (Plan-Do-Check-Act), trasformando i dati grezzi in informazioni per il miglioramento continuo.

3.1 Finalità del Cruscotto

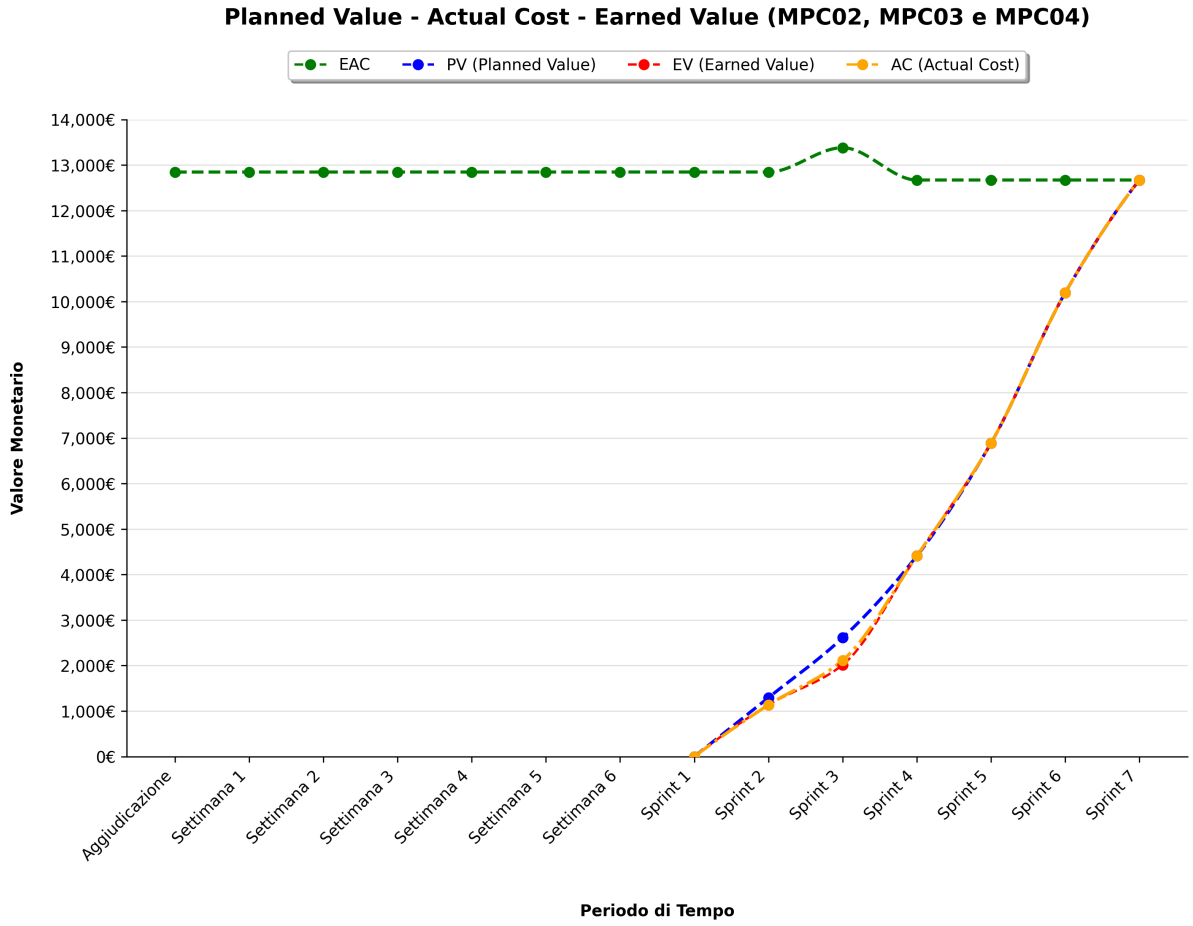
Il cruscotto di valutazione assolve a due funzioni fondamentali:

- **Monitoraggio Proattivo:** Consente di rilevare scostamenti rispetto alle soglie definite all'interno del *Piano di Qualifica* stesso, evitando che criticità latenti si trasformino in blocchi operativi.
- **Tracciabilità Storica:** Documenta l'evoluzione delle prestazioni del gruppo nel tempo, permettendo di identificare pattern ricorrenti e validare l'efficacia delle azioni correttive implementate.

È importante evidenziare che il periodo iniziale, dall'aggiudicazione fino all'avvio formale delle attività di progetto (*Sprint 1*), ha rappresentato una fase di "palestra" durante la quale il gruppo si è dedicato allo studio approfondito delle tecnologie necessarie, partecipando anche a sessioni di formazione organizzate dall'azienda proponente Var Group.

3.2 Processi Primari: Fornitura e Sviluppo

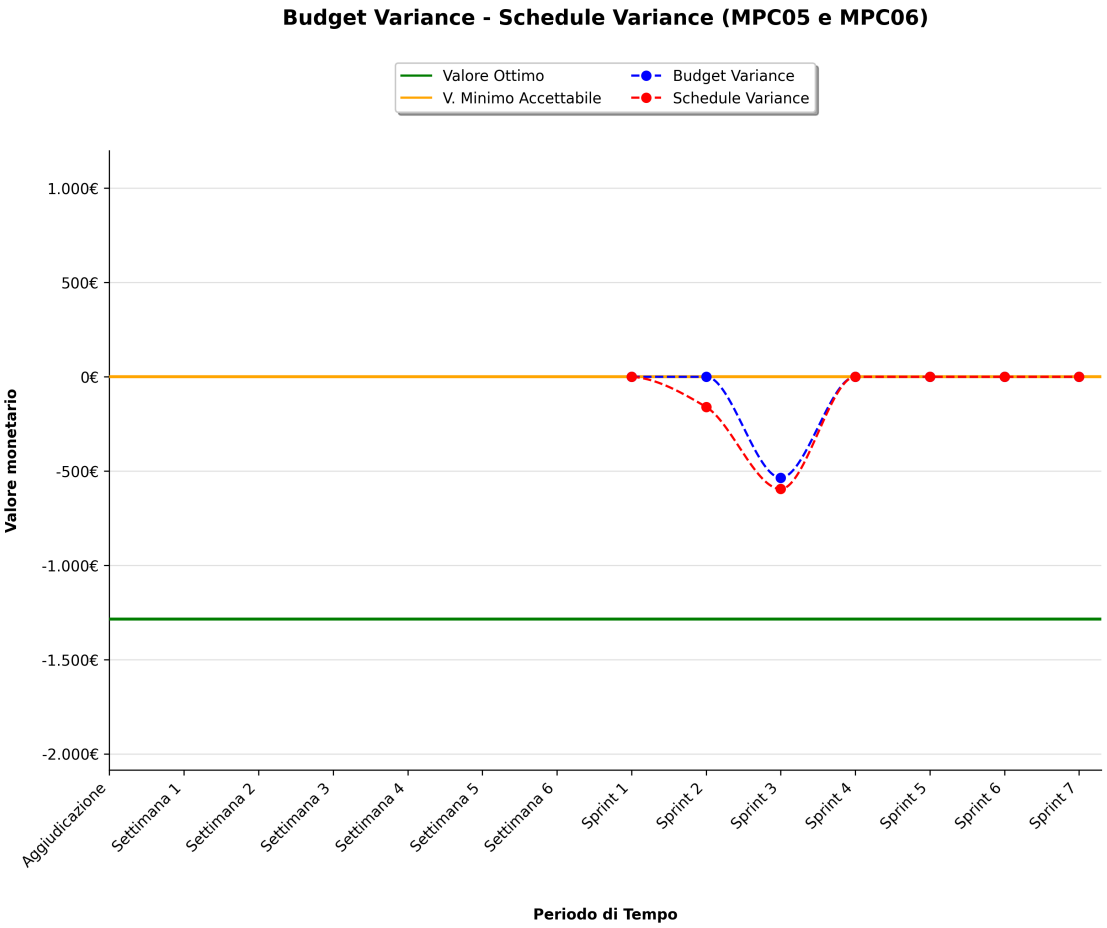
3.2.1 Planned Value - Actual Cost - Earned Value (MPC02, MPC03 e MPC04)



Dopo la fase iniziale, in cui le attività di formazione e setup sono state gestite come investimento interno senza gravare sul budget, il progetto è entrato nella fase operativa con l'avvio dello *Sprint 1*. In questa prima iterazione Skarab Group ha mostrato un buon equilibrio economico, completando il lavoro con un dispendio di risorse coerente con il valore prodotto, pur registrando un lieve ritardo rispetto alla pianificazione ideale.

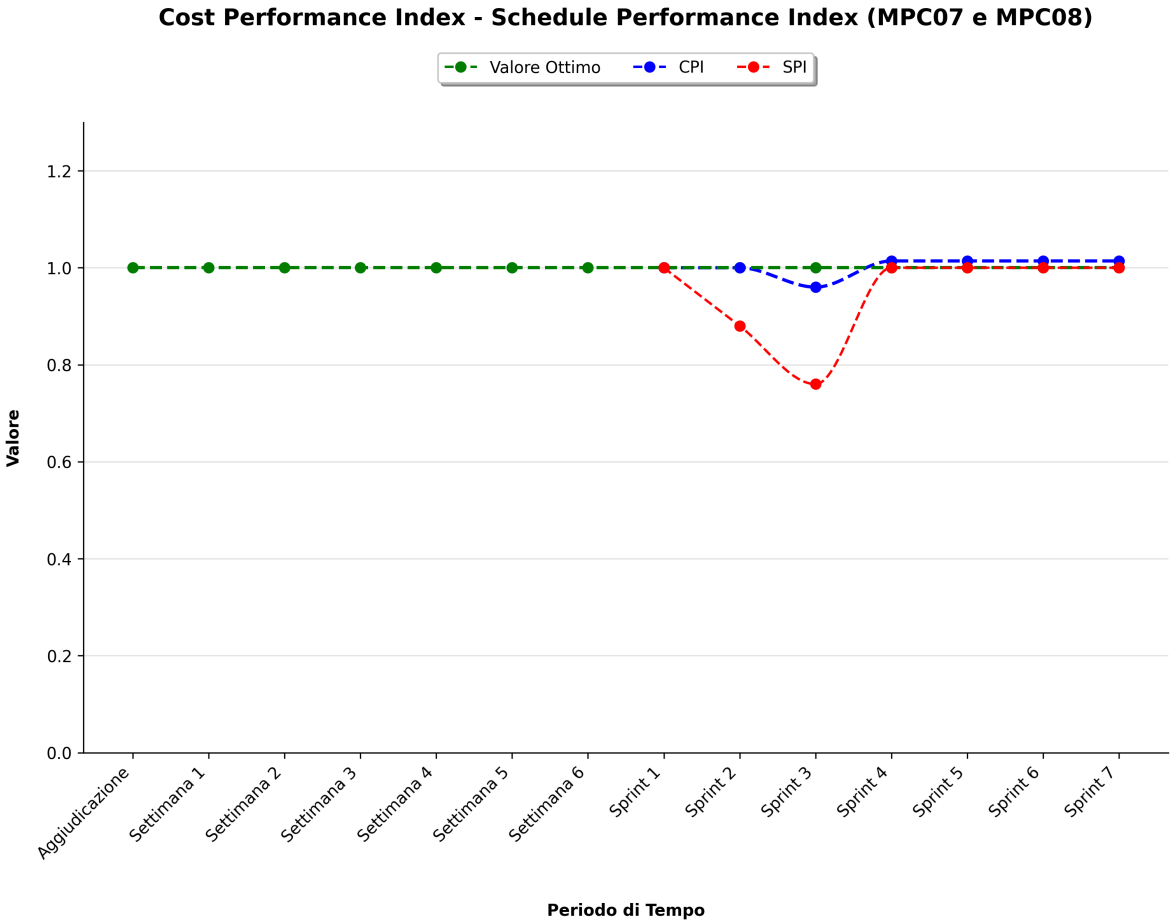
Tuttavia, la situazione ha subito una variazione significativa durante lo *Sprint 2*: a fronte di un incremento del *Planned Value* (PV) e dell'*Actual Cost* (AC), l'*Earned Value* (EV) ha subito una flessione. Questo testimonia l'insorgere di inefficienze produttive e debito tecnico, legati alla necessità di ricalibrare task qualitativamente insufficienti che hanno rallentato la produzione.

3.2.2 Budget Variance - Schedule Variance (MPC05 e MPC06)



Il grafico monitora la salute economica e temporale del progetto a partire dallo *Sprint 1* durante il quale la *Schedule Variance* (SV) mostra una leggera flessione. Quest'ultima si è accentuata nello *Sprint 2*, riflettendosi anche sulla *Budget Variance* (BV).

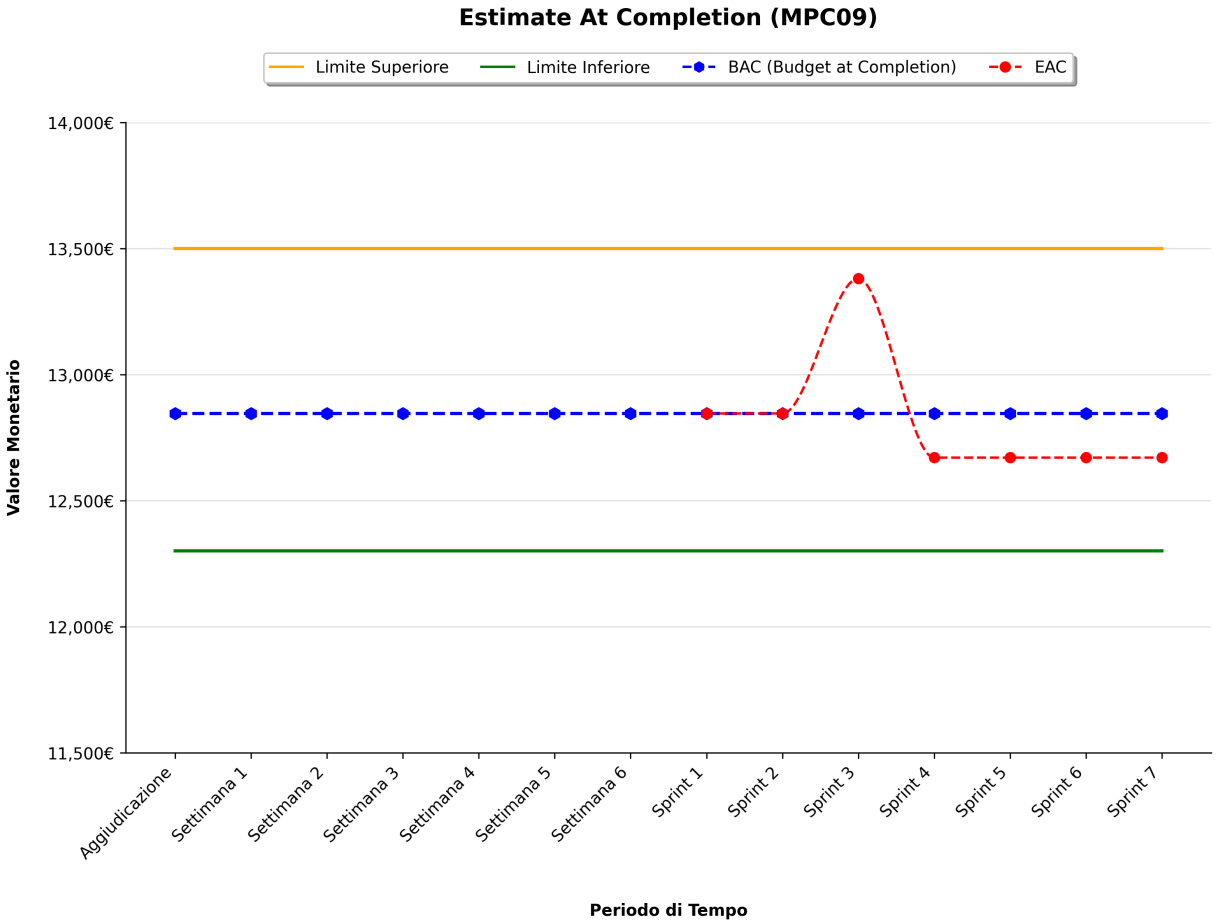
3.2.3 Cost Performance Index - Schedule Performance Index (MPC07 e MPC08)



Dal grafico è possibile notare come, inizialmente, lo *Schedule Performance Index* (SPI) sia inferiore a 1, indicando un leggero ritardo fisiologico. La buona gestione dei costi è invece documentata dal *Cost Performance Index* (CPI) che, essendo pari a 1, indica un ottimo utilizzo del budget.

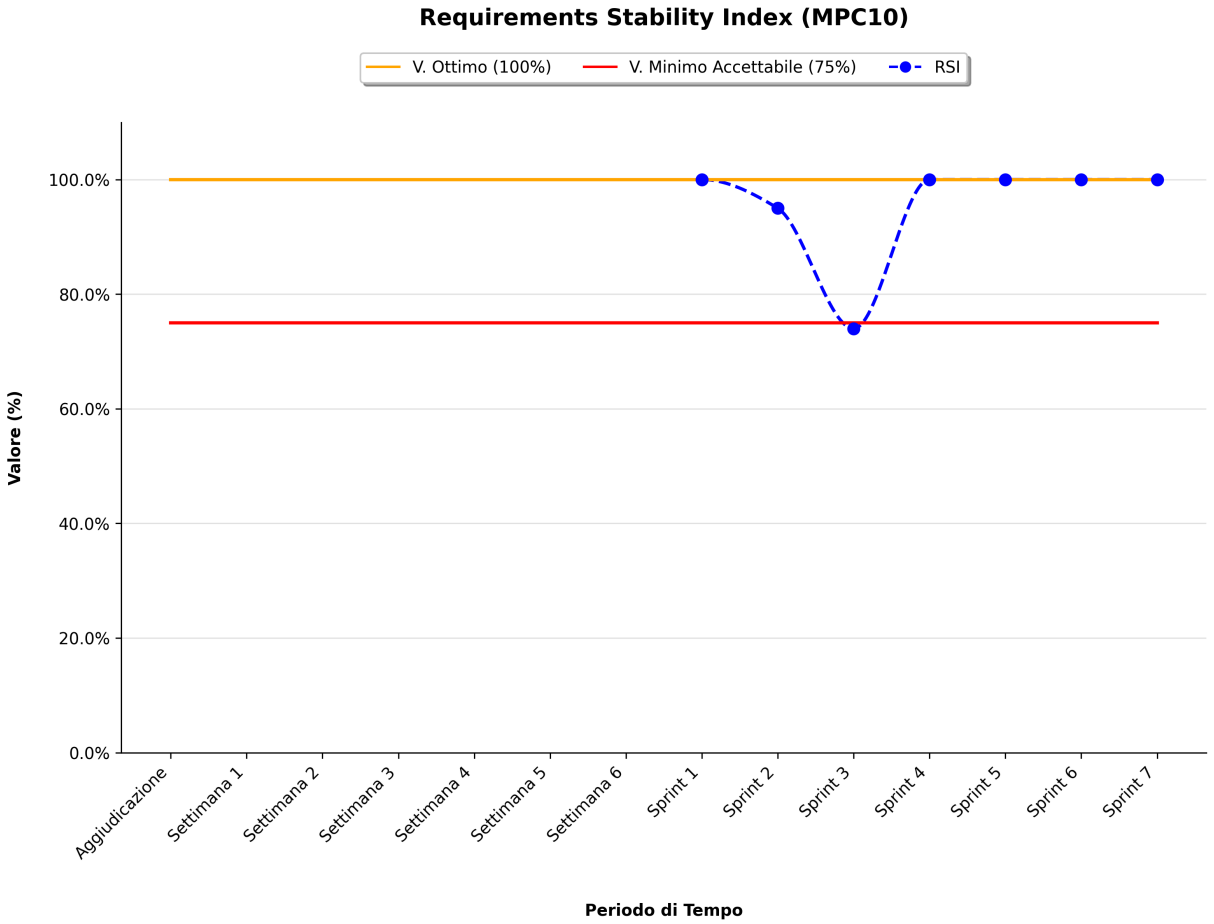
La situazione è peggiorata nel corso dello *Sprint 2*, durante il quale si è verificato un crollo dello *Schedule Performance Index* (SPI) che segnala un ritardo critico rispetto alla pianificazione. La cattiva gestione temporale è in contrasto con il *Cost Performance Index* (CPI) che si mantiene abbastanza stabile, confermando che il problema non è di natura economica ma organizzativa.

3.2.4 Estimate at Completion (MPC09)



Dopo una fase iniziale di stabilità coincidente con il budget originale, è possibile notare come una gestione inefficiente delle risorse abbia spinto la previsione di spesa verso il limite massimo.

3.2.5 Requirements Stability Index (MPC10)



Il *Requirements Stability Index* (RSI) registra un peggioramento nel corso dello *Sprint 2*. Tale flessione è riconducibile a una sottostima iniziale dei requisiti impliciti e all'emersione di ulteriori requisiti in seguito al colloquio con il Prof. Cardin: il team ha dovuto apportare modifiche significative per aggiungere i requisiti non tracciati in precedenza dagli Analisti.

3.3 Processi di Supporto

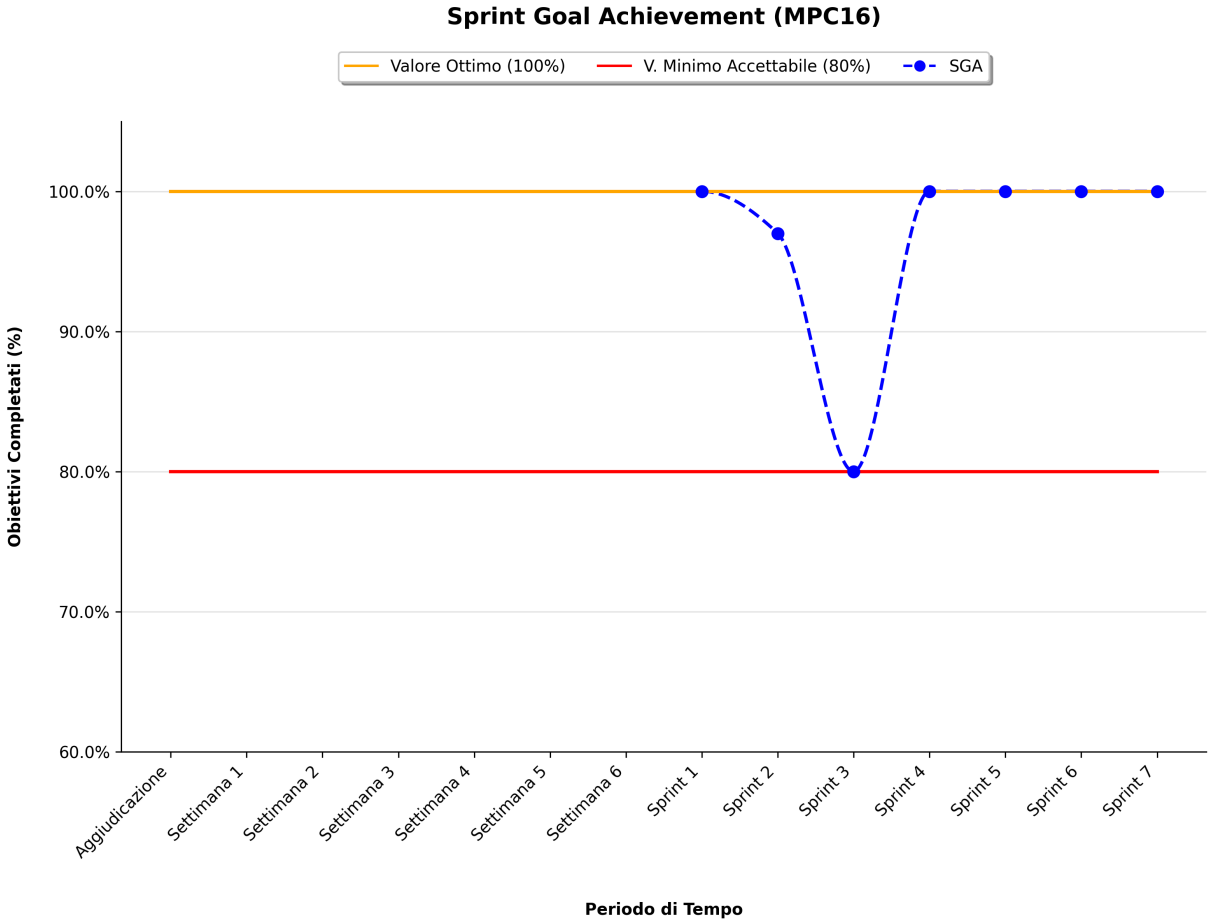
3.3.1 Gulpease Index (MPC11)

3.3.2 Correttezza Ortografica (MPC12)

3.4 Processi Organizzativi

3.4.1 Metrics Satisfaction (MPC15)

3.4.2 Sprint Goal Achievement (MPC16)



Dal grafico è possibile osservare la buona efficacia operativa dimostrata durante lo *Sprint 1* dal team, che è riuscito a completare gli obiettivi prefissati quasi nella loro interezza. Durante lo *Sprint 2*, invece, la metrica ha subito una flessione poiché gli obiettivi prefissati non sono stati pienamente raggiunti.

3.5 Automiglioramento

TODO: Definire ulteriori problemi emersi durante lo sviluppo del progetto

3.5.1 Introduzione

Il miglioramento continuo risulta fondamentale per garantire la qualità del progetto Code Guardian. Seguendo il Way of Working definito nelle **Norme di Progetto**, il team effettua retrospettive periodiche per identificare i colli di bottiglia operativi e implementare soluzioni correttive secondo il ciclo PDCA. Le valutazioni sono state suddivise in tre ambiti critici identificati durante lo sviluppo iniziale.

3.5.1.1 Valutazione Tecnologica

L'adozione di nuovi strumenti ha richiesto una fase di adattamento per garantire che l'infrastruttura tecnologica supportasse, e non ostacolasse, la produttività.

Strumento	Problema	Decisione presa
Typst	Curva di apprendimento ripida e rischio di disomogeneità stilistica nei documenti.	Studio autonomo obbligatorio e creazione di template condivisi per centralizzare la logica di formattazione.
Issue Tracking System	Frammentazione delle informazioni tecniche e incomprensioni sui requisiti.	Centralizzazione della comunicazione asincrona su Jira, con obbligo di risoluzione dei dubbi tramite commenti tracciabili sulle singole task.

Table 10: Ottimizzazione tecnologica

3.5.1.2 Valutazione Organizzativa

Il coordinamento di un gruppo numeroso ha richiesto un passaggio da una comunicazione informale a una struttura più gerarchica e definita.

Criticità	Soluzione Organizzativa
Difficoltà di allineamento immediato su decisioni logistiche e urgenze.	Definizione di canali gerarchici: Telegram per le urgenze, Discord per il lavoro sincrono e i meeting di allineamento.
Sovrapposizione di sforzi o "buchi" operativi dovuti alla dimensione del gruppo (7 persone).	Suddivisione in sotto-gruppi di lavoro tematici per ridurre il rumore comunicativo e aumentare la focalizzazione.

Table 11: Miglioramento dell'efficienza organizzativa

3.5.1.2.1 Valutazione delle Responsabilità

Per evitare lo stallo decisionale e risolvere ambiguità metodologiche, è stato necessario definire chiaramente i confini d'azione dei ruoli e attivare canali di supporto esterni.

Problema di Ruolo / Criticità	Azioni di Risposta
Mancanza di una visione d'insieme su documenti complessi come l'Analisi dei Requisiti.	Nomina di un referente responsabile per ogni macro-documento, incaricato di supervisionare la coerenza finale e il rispetto delle scadenze.
Forte difficoltà degli Analisti nel tracciare correttamente attori e sistemi in conformità con gli standard richiesti.	Attivazione di una strategia di chiarimento a più livelli: • Consultazione diretta con il **Prof. Cardin** per risolvere dubbi metodologici; • Confronto costruttivo con altri gruppi di progetto per allineamento sugli standard; • Richiesta di intervento dell'azienda proponente (Var Group) per chiarire il perimetro operativo del sistema.
Incertezza sulla validazione degli incrementi prodotti e rischio di errori latenti.	Rafforzamento del ruolo dei Verificatori, con l'introduzione di una revisione obbligatoria "a quattro occhi" prima di ogni merge sul repository principale.

Table 12: Definizione e gestione delle responsabilità e risoluzione blocchi metodologici

3.5.1.3 Conclusioni

Il processo di automiglioramento ha permesso di trasformare le criticità iniziali — tipiche di un gruppo numeroso che opera su tecnologie nuove — in punti di forza procedurali. L'integrazione tra responsabilità chiare, strumenti di tracking e canali di comunicazione dedicati garantisce la sostenibilità del progetto verso la milestone PB.