



Piano di Qualifica

Informazioni sul documento

Versione	3.0.0
Data di Creazione	2016-12-04
Data ultima modifica	2017-04-09
Stato	Approvato
Redazione	Jordan Gottardo Marco Pasqualini
Verifica	Giulia Petenazzi
Approvazione	Daniel De Gaspari
Uso	Esterno
Lista di distribuzione	Professor Tullio Vardanega Professor Riccardo Cardin Zephyrus RiskApp
Email di riferimento	zephyrus.swe@gmail.com

Registro delle modifiche

Versione	Data	Autore	Ruolo	Descrizione
3.0.0	2017-04-09	Daniel De Gaspari	<i>Responsabile</i>	Approvazione documento
2.1.0	2017-04-06	Giulia Petenazzi	<i>Verificatore</i>	Verifica documento
2.0.3	2017-04-05	Marco Pasqualini	<i>Amministratore</i>	Aggiunta sezione 3.4 "Test di unità"
2.0.2	2017-04-05	Marco Pasqualini	<i>Amministratore</i>	Aggiunta sezione D.2.2 "Verifica del software"
2.0.2	2017-03-22	Marco Pasqualini	<i>Amministratore</i>	Correzione post RP: eliminata sezione E; riorganizzata sezione D per obiettivo invece che per periodo
2.0.1	2017-03-22	Jordan Gottardo	<i>Amministratore</i>	Correzione post RP: Spostata sezione Test da appendice a corpo del documento (sezione 3)
2.0.1	2017-03-22	Jordan Gottardo	<i>Amministratore</i>	Correzione post RP: Eliminata sezione 3
2.0.0	2017-02-30	Leonardo Brutesco	<i>Responsabile</i>	Approvazione documento
1.2.0	2017-02-26	Jordan Gottardo	<i>Verificatore</i>	Verifica documento
1.1.1	2017-02-27	Giulia Petenazzi	<i>Analista</i>	Corretta descrizione "Superamento dei test pianificati". Aggiunti test di validazione, sistema, integrazione
1.1.0	2017-02-26	Jordan Gottardo	<i>Verificatore</i>	Verifica documento
1.0.8	2017-02-22	Giulia Petenazzi	<i>Analista</i>	Eliminazione obiettivo "Validazione Web"; cambiati valore accettabile da 100% a 80% per obiettivo "Implementazione funzionalità desiderabili" in §2.2.2.2; Rinominato "Superamento dei test richiesti" in "Superamento dei test pianificati" in §2.2.2.9;
1.0.7	2017-02-22	Marco Pasqualini	<i>Analista</i>	Stesura § 2.2.1
1.0.6	2017-02-06	Marco Pasqualini	<i>Analista</i>	Stesura § 3.3

Versione	Data	Autore	Ruolo	Descrizione
1.0.5	2017-02-05	Giulia Petenazzi	<i>Analista</i>	Sostituita nome section "Copertura dei test" con "Superamento dei test"
1.0.4	2017-02-04	Giulia Petenazzi	<i>Analista</i>	Section documentazione del codice: sostituito "Avere" con "Scrivere", "anche essere soggetti esterni" con "essere soggetti esterni"
1.0.3	2017-02-04	Giulia Petenazzi	<i>Analista</i>	Estesa descrizione delle section "Rispetto del budget", "Leggibilità e comprensibilità", "Adesione alle norme interne", "Correttezza ortografica", "Correttezza concettuale", "Implementazione delle funzionalità obbligatorie", "Implementazione delle funzionalità desiderabili", "Basso numero di statement per metodo", "Basso numero di parametri per metodo", "Basso numero di campi dati per classe", "Basso grado di accoppiamento", "Bassa complessità ciclomatica", "Basso numero di variabili dichiarate e non utilizzate", "Correzione delle situazioni di fallimento"
1.0.2	2017-02-03	Giulia Petenazzi	<i>Analista</i>	Cambiato nomi delle section "Visione generale della strategia di gestione della qualità" in "Obiettivi di qualità" e di "La strategia di gestione della qualità nel dettaglio" in "Organizzazione della gestione della qualità". Ristrutturato il contenuto di tali section

Versione	Data	Autore	Ruolo	Descrizione
1.0.1	2017-01-29	Giulia Petenazzi	<i>Analista</i>	Messi a glossario i termini solo la prima volta che compaiono in ogni section del documento
1.0.0	2017-01-07	Giulia Petenazzi	<i>Responsabile</i>	Approvazione
0.2.0	2017-01-06	Leonardo Brutesco	<i>Verificatore</i>	Verifica documento
0.1.1	2017-01-06	Jordan Gottardo	<i>Analista</i>	Correzione errori
0.1.0	2017-01-06	Leonardo Brutesco	<i>Verificatore</i>	Verifica documento
0.0.7	2017-01-04	Marco Pasqualini	<i>Verificatore</i>	Aggiunta sezione resoconto delle attività di verifica - fase analisi
0.0.6	2016-12-13	Marco Pasqualini	<i>Analista</i>	Aggiunte sezioni CMM, ISO/IEC-9126, Test, PDCA
0.0.5	2016-12-12	Jordan Gottardo	<i>Analista</i>	Aggiunta sezione la strategia di gestione della qualità nel dettaglio
0.0.3	2016-12-05	Jordan Gottardo	<i>Analista</i>	Aggiunta sezione visione generale della strategia di gestione
0.0.2	2016-12-04	Jordan Gottardo	<i>Analista</i>	Aggiunta introduzione
0.0.1	2016-12-04	Jordan Gottardo	<i>Analista</i>	Creazione template e indice

Indice

1	Introduzione	1
1.1	Scopo del documento	1
1.2	Scopo del prodotto	1
1.3	Glossario	1
1.4	Riferimenti	1
1.4.1	Riferimenti normativi	1
1.4.2	Riferimenti informativi	2
2	Obiettivi di qualità	3
2.1	Qualità di processo	3
2.1.1	Tutti i processi	3
2.1.1.1	Miglioramento costante	4
2.1.1.2	Rispetto della pianificazione	4
2.1.1.3	Rispetto del budget	4
2.1.1.4	Completezza dell'analisi dei rischi	4
2.1.2	Processo di documentazione	5
2.1.2.1	Impegno nella documentazione	5
2.1.2.2	Qualità del template	5
2.1.2.3	Qualità delle immagini	5
2.1.2.4	Tracciamento delle modifiche	6
2.1.3	Processo di sviluppo	6
2.1.3.1	Impegno nella codifica	6
2.1.3.2	Assegnazione scenari principali	6
2.1.3.3	Copertura requisiti obbligatori	7
2.1.3.4	Basso grado di accoppiamento	7
2.1.3.5	Alto grado di utilità	7
2.2	Qualità di prodotto	7
2.2.1	Qualità dei documenti	8
2.2.1.1	Leggibilità e comprensibilità	8
2.2.1.2	Adesione alle norme interne	8
2.2.1.3	Correttezza ortografica	8
2.2.1.4	Correttezza concettuale	9
2.2.1.5	Basso livello di annidamento dell'indice	9
2.2.2	Qualità del software	9
2.2.2.1	Implementazione delle funzionalità obbligatorie	9
2.2.2.2	Implementazione delle funzionalità desiderabili	10
2.2.2.3	Basso numero di statement per metodo	10
2.2.2.4	Basso numero di parametri per metodo	10
2.2.2.5	Basso numero di campi dati per classe	10
2.2.2.6	Bassa complessità ciclomatica	11
2.2.2.7	Assenza di variabili dichiarate e non utilizzate	11
2.2.2.8	Documentazione del codice	11
2.2.2.9	Superamento dei test pianificati	11
2.2.2.10	Robustezza	12
2.2.2.11	Correzione delle situazioni di fallimento	12
2.2.2.12	Copertura degli statement	12
2.2.2.13	Copertura dei branch	12
2.2.3	Corrispondenza obiettivo - caratteristica di qualità	13

3	Test	14
3.1	Test di validazione	14
3.2	Test di sistema	25
3.3	Test di integrazione	29
3.4	Test di unità	30
A	CMM	35
A.1	Struttura	35
A.2	Livelli	35
B	ISO/IEC 9126	37
B.1	Modello della qualità del software	37
B.1.1	Modello della qualità esterna e interna	37
B.1.2	Modello della qualità in uso	38
B.2	Qualità esterna e relative metriche	38
B.3	Qualità interna e relative metriche	38
B.4	Qualità in uso e relative metriche	38
C	PDCA	39
C.1	Fasi	39
D	Resoconto delle attività di verifica	40
D.1	Verifica dei processi	40
D.1.1	Tutti i processi	40
D.1.1.1	Miglioramento costante	40
D.1.1.1.1	Periodo An	40
D.1.1.1.2	Periodo PI	41
D.1.1.1.3	Periodo PCV	41
D.1.1.2	Rispetto della pianificazione	43
D.1.1.2.1	Periodo An	43
D.1.1.2.2	Periodo PI	43
D.1.1.2.3	Periodo PCV	43
D.1.1.3	Rispetto del budget	44
D.1.1.3.1	Periodo An	44
D.1.1.3.2	Periodo PI	44
D.1.1.3.3	Periodo PCV	44
D.1.1.4	Completezza dell'analisi dei rischi	45
D.1.1.4.1	Periodo An	45
D.1.1.4.2	Periodo PI	45
D.1.1.4.3	Periodo PCV	45
D.1.2	Processo di documentazione	46
D.1.2.1	Impegno nella documentazione	46
D.1.2.1.1	Periodo PI	46
D.1.2.1.2	Periodo PCV	46
D.1.2.2	Qualità del template	47
D.1.2.2.1	Periodo PI	47
D.1.2.2.2	Periodo PCV	47
D.1.2.3	Qualità delle immagini	48
D.1.2.3.1	Periodo PI	48
D.1.2.3.2	Periodo PCV	48

D.1.2.4	Tracciamento delle modifiche	49
D.1.2.4.1	Periodo PI	49
D.1.2.4.2	Periodo PCV	49
D.1.3	Processo di sviluppo	50
D.1.3.1	Impegno nella codifica	50
D.1.3.1.1	Periodo PCV	50
D.1.3.2	Assegnazione scenari principali	51
D.1.3.2.1	Periodo PI	51
D.1.3.2.2	Periodo PCV	51
D.1.3.3	Copertura requisiti obbligatori	52
D.1.3.3.1	Periodo PI	52
D.1.3.3.2	Periodo PCV	52
D.1.3.4	Basso grado di accoppiamento	53
D.1.3.4.1	Periodo PI	53
D.1.3.4.2	Periodo PCV	53
D.1.3.5	Alto grado di utilità	54
D.1.3.5.1	Periodo PI	54
D.1.3.5.2	Periodo PCV	54
D.2	Verifica dei prodotti	55
D.2.1	Verifica dei documenti	55
D.2.1.1	Leggibilità e comprensibilità	55
D.2.1.1.1	Periodo An	55
D.2.1.1.2	Periodo PI	56
D.2.1.1.3	Periodo PCV	56
D.2.1.2	Adesione alle norme interne	57
D.2.1.2.1	Periodo An	57
D.2.1.2.2	Periodo PI	57
D.2.1.2.3	Periodo PCV	58
D.2.1.3	Correttezza ortografica	59
D.2.1.3.1	Periodo An	59
D.2.1.3.2	Periodo PI	59
D.2.1.3.3	Periodo PCV	60
D.2.1.4	Correttezza concettuale	61
D.2.1.4.1	Periodo An	61
D.2.1.4.2	Periodo PI	61
D.2.1.4.3	Periodo PCV	62
D.2.1.5	Basso livello di annidamento dell'indice	63
D.2.1.5.1	Periodo PI	63
D.2.1.5.2	Periodo PCV	63
D.2.2	Verifica del software	64
D.2.2.1	Implementazione delle funzionalità obbligatorie	64
D.2.2.1.1	Periodo PCV	64
D.2.2.2	Implementazione delle funzionalità desiderabili	65
D.2.2.2.1	Periodo PCV	65
D.2.2.3	Basso numero di statement per metodo	66
D.2.2.3.1	Periodo PCV	66
D.2.2.4	Basso numero di parametri per metodo	67
D.2.2.4.1	Periodo PCV	67
D.2.2.5	Basso numero di campi dati per classe	68

D.2.2.5.1	Periodo PCV	68
D.2.2.6	Bassa complessità ciclomatica	69
D.2.2.6.1	Periodo PCV	69
D.2.2.7	Assenza di variabili dichiarate e non utilizzate	70
D.2.2.7.1	Periodo PCV	70
D.2.2.8	Documentazione del codice	71
D.2.2.8.1	Periodo PCV	71
D.2.2.9	Superamento dei test pianificati	72
D.2.2.9.1	Periodo PCV	72
D.2.2.10	Robustezza	73
D.2.2.10.1	Periodo PCV	73
D.2.2.11	Correzione delle situazioni di fallimento	74
D.2.2.11.1	Periodo PCV	74
D.2.2.12	Copertura degli statement	75
D.2.2.12.1	Periodo PCV	75
D.2.2.13	Copertura dei branch	76
D.2.2.13.1	Periodo PCV	76

1 Introduzione

1.1 Scopo del documento

Il presente documento ha lo scopo di fissare in modo quantitativo gli obiettivi di qualità di processo e di prodotto che il *team_G* si è prefissato. Inoltre verranno illustrate le strategie di verifica e validazione utilizzate per raggiungere tali obiettivi e la loro effettiva applicazione nei vari periodi di sviluppo.

1.2 Scopo del prodotto

Lo scopo del prodotto consiste nella creazione di un'interfaccia web contenente una mappa geografica su cui potranno essere rappresentati:

- il processo produttivo aziendale;
- gli scenari di danno;
- i risultati dell'analisi dei rischi.

Il prodotto verrà utilizzato da agenti assicuratori per l'inserimento delle informazioni utili allo svolgimento dell'analisi dei rischi dell'assicurando.

L'interfaccia dovrà essere in grado di connettersi ai sistemi preesistenti di *RiskApp* per la memorizzazione e gestione dei dati inseriti. A causa del requisito di integrabilità, che è stato deciso di soddisfare, l'applicazione da sviluppare sarà parte integrante dell'attuale applicazione del proponente.

1.3 Glossario

Allo scopo di rendere più semplice e chiara la comprensione dei documenti viene allegato il *Glossario v2.0.0*, nel quale verranno raccolte le spiegazioni di terminologia tecnica o ambigua, abbreviazioni ed acronimi. Per evidenziare un termine presente in tale documento, esso verrà marcato con il pedice _G. Solo la prima occorrenza del termine in ogni sezione sarà marcata per non appesantire la lettura del documento.

Tutti i termini del glossario evidenziati sono link ipertestuali al glossario stesso; affinché funzionino correttamente è necessario che la posizione delle directory e dei file forniti non venga alterata.

1.4 Riferimenti

1.4.1 Riferimenti normativi

- *Norme di progetto v3.0.0*;
- *capitolato_G d'appalto C3*: <http://www.math.unipd.it/tullio/IS-1/2016/Progetto/C3.pdf>;
- *standard ISO_G/IEC_G 9126:2001*: https://en.wikipedia.org/wiki/ISO/IEC_9126;
- *Capability Maturity Model(CMM_G)*: https://en.wikipedia.org/wiki/Capability_Maturity_Model.

1.4.2 Riferimenti informativi

- *Piano di progetto v3.0.0*;
- *Plan-Do-Check-Act_G*: <https://en.wikipedia.org/wiki/PDCA>;
- *indice di Gulpease_G*: http://it.wikipedia.org/wiki/Indice_Gulpease;
- slide del corso di Ingegneria del Software - Qualità del software:
<http://www.math.unipd.it/tullio/IS-1/2016/Dispense/L10.pdf>;
- slide del corso di Ingegneria del Software - Qualità di processo:
<http://www.math.unipd.it/tullio/IS-1/2016/Dispense/L11.pdf>;
- metriche di progetto: https://it.wikipedia.org/wiki/Metriche_di_progetto.

2 Obiettivi di qualità

In questa sezione vengono espressi gli obiettivi di qualità che il *team_G* si è prefissato. Data la difficoltà (e in alcuni casi l'impossibilità) nel misurare direttamente la qualità, sono stati scelti standard, modelli e metriche. Ognuno di questi fa uso di scale differenti e fissate a priori. Per ogni criterio, il team ha fissato dei range di valori accettabili e ottimi. A prescindere dal livello raggiunto in ogni misurazione, l'obiettivo da perseguire è il miglioramento continuo della qualità, realizzata attraverso il ciclo *PDCA_G*. Nel caso in cui non si raggiungesse l'obiettivo minimo, dovranno essere attuate misure correttive come previsto dalle *Norme di progetto v3.0.0*. Siccome la qualità non è una proprietà intrinseca dei processi, è fondamentale dotarsi di buoni strumenti per effettuare le misurazioni.

Ogni volta che viene effettuata una misurazione si ottiene un valore, ovvero una misura. Per poter comparare due valori è necessario rapportarli su una scala: facendo ciò si crea una metrica. È possibile trovare una descrizione degli strumenti e delle metriche nelle *Norme di progetto v3.0.0*.

Tenendo conto degli obiettivi che verranno stabiliti, il valore può avere giudizio:

- **negativo:** obiettivo non raggiunto. È necessario fare ulteriori verifiche o correzioni. Per alcune metriche, le azioni da intraprendere in caso di valori negativi sono descritte nelle *Norme di progetto v3.0.0*;
- **accettabile:** obiettivo raggiunto, soglia di accettabilità superata. Se il criterio a cui il valore si riferisce è un obiettivo importante in ottica PDCA, è necessario attuare azioni per migliorare la qualità;
- **ottimale:** obiettivo raggiunto, soglia di ottimalità superata. Anche in questo caso vale quanto detto nel punto precedente riguardo il miglioramento continuo.

2.1 Qualità di processo

Garantire la qualità dei processi è fondamentale se si vogliono ottenere prodotti di qualità. L'unico modo di garantire *quality assurance_G* è far sì che i processi siano normati e misurati. Inoltre, è possibile anche ottenere maggiore efficienza, efficacia e ripetibilità dei risultati. Gli obiettivi relativi a questi ambiti sono illustrati nel *Piano di progetto v3.0.0*. Le caratteristiche che i processi dovrebbero avere sono le seguenti:

- un processo dovrebbe essere in grado di migliorare continuamente le proprie performance:
 - le performance di un processo dovrebbero essere misurabili;
 - un processo dovrebbe perseguire obiettivi quantitativi di miglioramento.
- i processi e le loro attività dovrebbero rispettare i tempi e i costi stabiliti dal *Piano di progetto v3.0.0*.

Seguono gli obiettivi e le relative metriche riguardanti la qualità di processo che il team ha stabilito.

2.1.1 Tutti i processi

In questa sezione sono definiti gli obiettivi generici per tutti i processi.

2.1.1.1 Miglioramento costante

Per quantificare il livello di performance raggiunto dai processi, si è deciso di adottare il modello *CMM_G*. L'obiettivo è migliorare costantemente tale livello, secondo quanto definito dal ciclo PDCA.

- **metrica utilizzata:** Livello CMM (*LCMM*);
- **valore negativo:** 1;
- **valore accettabile:** 2 e 3;
- **valore ottimale:** 4 e 5.

Per una descrizione più dettagliata del modello CMM, si faccia riferimento all'[appendice A](#).

2.1.1.2 Rispetto della pianificazione

Rispettare la pianificazione del lavoro stabilita nel *Piano di progetto v3.0.0* è fondamentale per evitare ritardi e garantire la qualità del processo. Qualora non la si rispettasse, è molto probabile che il processo non abbia le caratteristiche di qualità desiderate.

- **metrica utilizzata:** Schedule Variance (*SV*);
- **valore negativo:** $SV \geq 5$ giorni;
- **valore accettabile:** $0 < SV \leq 4$ giorni;
- **valore ottimale:** $SV \leq 0$ giorni.

2.1.1.3 Rispetto del budget

Rispettare il budget stabilito nel *Piano di progetto v3.0.0* è un obiettivo importante per evitare inefficienze nell'utilizzo delle risorse. Il team desidera che il costo effettivo non si discosti eccessivamente da quanto pianificato.

- **metrica utilizzata:** Cost Variance (*CV*);
- **valore negativo:** $CV > 10\%$;
- **valore accettabile:** $0\% < CV \leq 10\%$;
- **valore ottimale:** $CV \leq 0\%$.

2.1.1.4 Completezza dell'analisi dei rischi

Il team desidera che l'analisi dei rischi sia il più completo possibile, così da ridurre la probabilità di subire danni da rischi non preventivati.

- **metrica utilizzata:** *RNP*;
- **valore negativo:** $RNP > 2$;
- **valore accettabile:** $1 \leq RNP \leq 2$;
- **valore ottimale:** $RNP = 0$.

2.1.2 Processo di documentazione

Oltre agli obiettivi precedentemente enunciati, relativi a tutti i processi, il team ha identificato degli obiettivi particolari riguardanti il processo di documentazione.

2.1.2.1 Impegno nella documentazione

Il team desidera che il tempo impiegato nella redazione dei documenti non venga sprecato. Inoltre, una scrittura del documento troppo veloce potrebbe significare scarsa attenzione e quindi generare troppi errori. È quindi necessario fissare un giusto grado di produttività.

- **metrica utilizzata:** Righe Documento Per Ora (*RDPO*);
- **valore negativo:** $RDPO > 30, RDPO < 5$;
- **valore accettabile:** $5 \leq RDPO < 15$;
- **valore ottimale:** $15 \leq RDPO \leq 30$.

2.1.2.2 Qualità del template

L'obiettivo è realizzare un template di qualità per garantire che gli *Analisti* che redigono i documenti non abbiano necessità di decidere la struttura del documento. Inoltre, il template fornirà tutti i comandi necessari per la stesura del documento. Il team desidera minimizzare il numero di comandi aggiuntivi richiesti dagli *Analisti* al *Responsabile* in quanto tutte le necessità in questo ambito dovrebbero già essere soddisfatte dal template.

- **metrica utilizzata:** Numero Comandi Richiesti (*NCR*);
- **valore negativo:** $NCR > 3$;
- **valore accettabile:** $0 < NCR \leq 3$;
- **valore ottimale:** $NCR = 0$.

2.1.2.3 Qualità delle immagini

Il team desidera che le immagini incluse nei documenti siano di qualità. È stata decisa una risoluzione verticale minima per tutte le immagini che compaiono nei documenti, sia esterni che interni. È preferibile avere a disposizione un'immagine ad alta risoluzione e ridurne la grandezza utilizzando gli appositi comandi \LaTeX invece di avere un'immagine di scarsa qualità ma di giuste dimensioni dall'inizio. Tuttavia, è necessario non eccedere con la risoluzione per non appesantire troppo il documento e i tempi di compilazione.

- **metrica utilizzata:** Risoluzione Verticale (*RV*);
- **valore negativo:** $RV < 720, RV > 400$;
- **valore accettabile:** $2160 \leq RV \leq 4000$;
- **valore ottimale:** $720 \leq RV < 2160$.

2.1.2.4 Tracciamento delle modifiche

L'obiettivo è rendere le modifiche ai documenti tracciabili. Un'alta tracciabilità semplifica di molto l'attività dei *Verificatori* ed evita che essi debbano chiedere supporto ai redattori del documento riguardo le sezioni modificate. Ogni task completato relativo ad un documento deve produrre un avanzamento di versione con un relativo inserimento nel registro delle modifiche, come definito dalle *Norme di progetto v3.0.0*.

- **metrica utilizzata:** Percentuale Tracciamento Modifiche (*PTM*);
- **valore negativo:** $PTM < 100\%$;
- **valore accettabile:** $PTM = 100\%$;
- **valore ottimale:** $PTM = 100\%$.

2.1.3 Processo di sviluppo

Oltre al processo di documentazione, anche il processo di sviluppo ha degli obiettivi specifici. Esso infatti è particolarmente vasto e copre molte attività, come l'analisi dei requisiti, la progettazione e la codifica.

2.1.3.1 Impegno nella codifica

Il team desidera che il tempo impiegato nella produzione di codice non sia sprecato. Tuttavia, una scrittura di un elevato numero di *statement_G* in poco tempo provoca spesso errori, codice confusionario e istruzioni inutili. È necessario quindi porsi un obiettivo che non sia ai due estremi.

- **metrica utilizzata:** Righe Codice Per Ora (*RCPO*);
- **valore negativo:** $RCPO > 20$, $RCPO < 3$;
- **valore accettabile:** $3 \leq RCPO \leq 10$;
- **valore ottimale:** $10 < RCPO \leq 20$.

2.1.3.2 Assegnazione scenari principali

Ad ogni use case dev'essere assegnato uno scenario principale, dato che aiuta la comprensione del flusso principale degli eventi. Il team desidera che nessuno use case sia privo di scenario principale.

- **metrica utilizzata:** Use Case senza Scenario Principale (*UCSP*);
- **valore negativo:** $UCSP > 0$;
- **valore accettabile:** $UCSP = 0$;
- **valore ottimale:** $UCSP = 0$.

2.1.3.3 Copertura requisiti obbligatori

Il team desidera che l'attività di progettazione produca un'architettura di qualità. Infatti, l'obiettivo è progettare componenti che siano in grado come minimo di soddisfare tutti i requisiti obbligatori.

- **metrica utilizzata:** Percentuale di Requisiti Obbligatori Coperti ($PROC$);
- **valore negativo:** $PROC < 100\%$;
- **valore accettabile:** $PROC = 100\%$;
- **valore ottimale:** $PROC = 100\%$.

2.1.3.4 Basso grado di accoppiamento

Il grado di accoppiamento indica le dipendenze uscenti da una *componente_G* del sistema verso le altre. Un numero di dipendenze troppo elevato è sintomo di scarsa coesione e quindi di cattiva progettazione. Eliminare completamente l'accoppiamento tuttavia è impossibile; uno degli obiettivi della progettazione è quello di mantenere basso tale livello.

- **metrica utilizzata:** Grado di Accoppiamento (GA);
- **valore negativo:** $GA > 10$;
- **valore accettabile:** $3 < GA \leq 10$;
- **valore ottimale:** $GA \leq 3$.

2.1.3.5 Alto grado di utilità

Il grado di utilità indica le dipendenze entranti in una componente del sistema. Un alto grado di utilità per le componenti è sintomo di buona progettazione, in quanto viene favorito il riuso.

- **metrica utilizzata:** Grado di Utilità (GU);
- **valore negativo:** $GU = 0$;
- **valore accettabile:** $1 < GU \leq 5$;
- **valore ottimale:** $GU > 5$.

2.2 Qualità di prodotto

Oltre alla qualità dei processi, il team desidera anche garantire determinate caratteristiche di qualità dei prodotti. Per raggiungere questo obiettivo, è necessario che il processo con cui tale prodotto viene realizzato sia controllato e vincolato. A tal fine, è stato scelto di seguire lo standard *ISO_G/IEC_G 9126:2001*.

Le tipologie di prodotti che verranno realizzati sono due:

- documenti;
- software.

2.2.1 Qualità dei documenti

Il team si pone come obiettivo la produzione di documenti di qualità. Essi sono infatti fondamentali per la comprensione del prodotto software fin dal concepimento, sia da parte di soggetti interni che esterni. Le metriche che il team ha scelto per i documenti sono il più oggettive possibili. Da sole non garantiscono la qualità generale del documento, quindi è necessario un'ulteriore e accurata verifica, soprattutto per evitare gli errori concettuali e di forma. Seguono gli obiettivi e le metriche riguardanti la qualità dei documenti che il team si è prefissato.

2.2.1.1 Leggibilità e comprensibilità

La leggibilità e comprensibilità dei documenti sono caratteristiche fondamentali affinché essi siano utili a coloro che li leggono. Dovrà essere posta particolare attenzione alla lunghezza delle frasi e alla complessità delle parole utilizzate. Il team desidera che i testi siano comprensibili da persone con almeno un diploma superiore.

- **metrica utilizzata:** *Indice Gulepease_G* (*IG*);
- **valore negativo:** $IG < 40$;
- **valore accettabile:** $40 \leq IG < 60$;
- **valore ottimale:** $IG \geq 60$.

2.2.1.2 Adesione alle norme interne

Aderire alle regole di stesura dei documenti definite nelle *Norme di progetto v3.0.0* è fondamentale per assicurare l'omogeneità del testo e della terminologia. Alcuni esempi di norme riguardanti i documenti sono quelle relative ai loro nomi (e relativa versione), agli elenchi puntati, ai ruoli dei membri, ecc. L'obiettivo è aderire in modo completo alle norme interne, eliminando tutti gli errori che le violino.

- **metrica utilizzata:** Errori riguardanti le Norme interne e Non Corretti (*ENNC*);
- **valore negativo:** $ENNC > 0$;
- **valore accettabile:** $ENNC = 0$;
- **valore ottimale:** $ENNC = 0$.

2.2.1.3 Correttezza ortografica

Il team desidera che i documenti prodotti siano completamente esenti da errori ortografici rilevati e non corretti. I *Verificatori*, oltre ad utilizzare gli strumenti automatici definiti nelle *Norme di progetto v3.0.0*, dovranno prestare particolare attenzione durante la lettura del documento per scoprire errori non rilevati.

- **metrica utilizzata:** Errori Ortografici Non Corretti (*EONC*);
- **valore negativo:** $EONC > 0$;
- **valore accettabile:** $EONC = 0$;
- **valore ottimale:** $EONC = 0$.

2.2.1.4 Correttezza concettuale

L'obiettivo è ridurre il più possibile il numero di errori concettuali rinvenuti e non corretti. Gli errori concettuali sono più gravi di quelli ortografici, in quanto veicolano un messaggio sbagliato al lettore. I *Verificatori*, in caso di dubbio sulla correzione di errori concettuali, dovranno utilizzare la procedura di gestione delle anomalie definita nelle *Norme di progetto v3.0.0*.

- **metrica utilizzata:** Errori Concettuali Non Corretti (*ECNC*)
- **valore negativo:** $ECNC > 5\%$;
- **valore accettabile:** $ECNC \leq 5\%$;
- **valore ottimale:** $ECNC = 0\%$.

2.2.1.5 Basso livello di annidamento dell'indice

L'obiettivo è contenere il livello di annidamento dei paragrafi del documento. Un livello di annidamento troppo elevato appesantisce la leggibilità del documento. Una struttura tabellare è un'alternativa preferibile all'eccessivo annidamento dei paragrafi, qualora i contenuti ne beneficiassero.

- **metrica utilizzata:** Livello Annidamento Indice (*LAI*);
- **valore negativo:** $LAI > 5$;
- **valore accettabile:** $3 < LAI \leq 5$;
- **valore ottimale:** $LAI \leq 3$.

2.2.2 Qualità del software

Il team desidera che il software prodotto sia di qualità.

Seguono gli obiettivi e le metriche riguardanti la qualità del software che il team si è prefissato.

2.2.2.1 Implementazione delle funzionalità obbligatorie

Il software deve implementare completamente le funzionalità descritte nei requisiti obbligatori. È fondamentale che i requisiti obbligatori siano soddisfatti, in quanto senza di essi non si avrebbe un prodotto accettabile.

- **metrica utilizzata:** Implementazione delle Funzionalità Obbligatorie (*IFO*)
- **valore negativo:** $IFO < 100\%$;
- **valore accettabile:** $IFO = 100\%$;
- **valore ottimale:** $IFO = 100\%$.

2.2.2.2 Implementazione delle funzionalità desiderabili

Il software deve implementare il maggior numero possibile delle funzionalità descritte nei requisiti desiderabili. Pur non essendo obbligatoriamente richieste dal proponente, il team reputa che esse siano di particolare importanza per la realizzazione di un software di qualità.

- **metrica utilizzata:** Implementazione delle Funzionalità Desiderabili (*IFD*);
- **valore negativo:** $IFD < 80\%$;
- **valore accettabile:** $80\% \leq IFD < 100\%$;
- **valore ottimale:** $IFD = 100\%$.

2.2.2.3 Basso numero di statement per metodo

Al fine di fornire metodi facilmente comprensibili, il team desidera contenere il numero di statement di ognuno di essi.

- **metrica utilizzata:** Numero di *Statement_G* per Metodo (*NSM*);
- **valore negativo:** $NSM > 60$;
- **valore accettabile:** $30 < NSM \leq 60$;
- **valore ottimale:** $NSM \leq 30$.

2.2.2.4 Basso numero di parametri per metodo

Un numero troppo elevato di parametri per metodo influenza negativamente la comprensibilità dello stesso e l'attività di codifica.

- **metrica utilizzata:** Numero di Parametri per Metodo (*NPM*);
- **valore negativo:** $NPM > 12$;
- **valore accettabile:** $5 < NPM \leq 12$;
- **valore ottimale:** $NPM \leq 5$.

2.2.2.5 Basso numero di campi dati per classe

Una classe con un numero troppo elevato di campi dati indica che essa non è abbastanza specializzata. Quasi sicuramente è possibile spezzarla in due o più classi con un minore numero di campi dati.

- **metrica utilizzata:** Numero Campi Dati Per Classe (*NCDPC*);
- **valore negativo:** $NCDPC > 15$;
- **valore accettabile:** $10 < NCDPC \leq 15$;
- **valore ottimale:** $NCDPC \leq 10$.

2.2.2.6 Bassa complessità ciclomatica

La complessità ciclomatica indica il numero di cammini linearmente indipendenti presenti all'interno del codice. Un valore particolarmente elevato implica una grande difficoltà durante l'esecuzione di testing, in quanto alcuni rami del grafo di controllo del programma potrebbero essere difficilmente raggiungibili. Una bassa complessità ciclomatica aiuta a raggiungere la copertura del 100% del codice durante la creazione ed esecuzione dei test.

- **metrica utilizzata:** Numero Ciclomático (NC);
- **valore negativo:** $NC > 20$;
- **valore accettabile:** $10 < NC \leq 20$;
- **valore ottimale:** $NC \leq 10$.

2.2.2.7 Assenza di variabili dichiarate e non utilizzate

Il team desidera che non siano presenti variabili dichiarate e non utilizzate all'interno del codice. La presenza di una di esse complicherebbe la leggibilità del codice e indicherebbe uno statement completamente inutile.

- **metrica utilizzata:** Numero di Variabili dichiarate e Non Utilizzate ($NVNU$);
- **valore negativo:** $NVNU > 0$;
- **valore accettabile:** $NVNU = 0$;
- **valore ottimale:** $NVNU = 0$.

2.2.2.8 Documentazione del codice

Scrivere codice documentato è importante per garantire manutenibilità e comprensibilità dello stesso. Il mezzo con cui si intende raggiungere tale obiettivo è commentare il codice. Verrà posta particolare attenzione nello scrivere commenti comprensibili anche a eventuali manutentori, che potranno essere soggetti esterni.

- **metrica utilizzata:** Rapporto tra le linee di Commento e le linee di Codice (RCC);
- **valore negativo:** $RCC < 10\%$, $300\% > RCC$;
- **valore accettabile:** $10\% \leq RCC$;
- **valore ottimale:** $30\% \leq RCC \leq 300\%$.

2.2.2.9 Superamento dei test pianificati

Assicurare il superamento dei test è fondamentale per poter verificare la corretta implementazione delle funzionalità previste dai requisiti.

- **metrica utilizzata:** Superamento dei Test Pianificati (STP);
- **valore negativo:** $STP < 80\%$;
- **valore accettabile:** $80 \leq STP < 90\%$;
- **valore ottimale:** $90\% \leq STP \leq 100\%$.

2.2.2.10 Robustezza

Il prodotto non deve interrompere il suo funzionamento al verificarsi di situazioni anomale e di errore. È preferibile la segnalazione dell'errore all'arresto improvviso.

- **metrica utilizzata:** Breakdown Avoidance (BA);
- **valore negativo:** $BA < 80\%$;
- **valore accettabile:** $80 \leq BA < 95\%$;
- **valore ottimale:** $BA \geq 95\%$.

2.2.2.11 Correzione delle situazioni di fallimento

Il prodotto deve superare la maggior parte dei test che provino a compromettere la sua stabilità. Una situazione di fallimento scoperta e non corretta indica che il risultato di un test non è stato sfruttato al massimo. L'obiettivo è correggere tutte le situazioni di fallimento scoperte durante i test.

- **metrica utilizzata:** Failure Avoidance (FA);
- **valore negativo:** $FA < 80\%$;
- **valore accettabile:** $80 \leq FA < 95\%$;
- **valore ottimale:** $FA \geq 95\%$.

2.2.2.12 Copertura degli statement

Ottenere una copertura degli statement elevata in fase di testing indica che durante un test vengono eseguite molte linee di codice di un metodo. Più linee di codice sono testate, più facile è scoprire gli errori.

- **metrica utilizzata:** Statement Coverage (SC);
- **valore negativo:** $SC < 70\%$;
- **valore accettabile:** $70 \leq SC < 90\%$;
- **valore ottimale:** $SC \geq 90\%$.

2.2.2.13 Copertura dei branch

La copertura dei branch indica quanti flussi logici di un metodo vengono testati da un test. È un tipo di copertura più potente rispetto a quella degli statement in quanto è in grado di scoprire più errori.

- **metrica utilizzata:** Branch Coverage (BC);
- **valore negativo:** $BC < 70\%$;
- **valore accettabile:** $70 \leq BC < 90\%$;
- **valore ottimale:** $BC \geq 90\%$.

2.2.3 Corrispondenza obiettivo - caratteristica di qualità

Gli obiettivi (e quindi anche le metriche) che il team ha scelto per misurare la qualità del software si possono riferire a:

- qualità interna;
- qualità esterna;
- qualità in uso.

Come descritto nello standard ISO/IEC 9126:2001, ogni metrica corrisponde a determinate caratteristiche di qualità. Segue una tabella che associa ogni metrica scelta alla relativa caratteristica. Inoltre è specificato a quale tipologia di qualità essa fa riferimento.

Obiettivo	Metrica	Tipo	Caratteristica
Implementazione funzionalità obbligatorie	IFO	Esterna	Funzionalità
Implementazione funzionalità desiderabili	IFD	Esterna	Funzionalità
Basso numero di statement per metodo	NSM	Interna	Manutenibilità
Basso numero di parametri per metodo	NPM	Interna	Manutenibilità
Basso numero di campi dati per classe	NCD	Interna	Manutenibilità
Bassa complessità ciclomatica	NC	Interna	Manutenibilità
Assenza di variabili dichiarate e non utilizzate	NVNU	Interna	Manutenibilità
Documentazione del codice	RCC	Interna	Manutenibilità
Superamento dei test pianificati	STP	Esterna	Affidabilità
Robustezza	BA	Esterna	Affidabilità
Correzione delle situazioni di fallimento	FA	Esterna	Affidabilità
Copertura degli statement	SC	Esterna	Affidabilità
Copertura dei branch	BC	Esterna	Affidabilità

Tabella 1: Mappa Metriche-Caratteristiche

3 Test

I test, eseguiti tramite analisi dinamica, sono attività che servono a verificare che il software prodotto implementi le funzionalità richieste. Una caratteristica fondamentale dei test è la ripetibilità: i risultati che essi forniscono devono essere deterministici, in modo da eseguire azioni correttive in caso gli esiti non siano quelli attesi. Per tracciare i test eseguiti e i risultati ottenuti sarà necessario produrre dei log di facile consultazione.

Le tabelle che descrivono i test utilizzano le seguenti abbreviazioni:

- **NI:** non implementato;
- **I:** implementato;
- **NS:** non soddisfatto;
- **S:** soddisfatto.

3.1 Test di validazione

I test di validazione, eseguiti durante il collaudo finale, servono a verificare che il software soddisfi le richieste del proponente.

Test	Descrizione	Operazioni	Stato	Esito
TVDF24.4	L'utente intende cambiare la modalità di visualizzazione della mappa	Viene richiesto di: <ol style="list-style-type: none"> 1. aprire l'applicazione; 2. cliccare sul pulsante per cambiare la visualizzazione della mappa. 	N.I	N.S
TVDF25	L'utente intende avviare il tutorial	Viene richiesto di: <ol style="list-style-type: none"> 1. aprire l'applicazione; 2. premere sul pulsante relativo al tutorial. 	N.I	N.S
TVFF11.3	L'utente intende compilare i dati dell'arco di tipo Trasporto.	Viene richiesto di: <ol style="list-style-type: none"> 1. avviare la procedura di aggiunta arco; 2. selezionare che l'arco è di tipo Trasporto; 3. compilare i dati dell'arco nell'area informativa. 	N.I	N.S

Test	Descrizione	Operazioni	Stato	Esito
TVFF14.5	L'utente intende modificare i dati dell'arco di tipo Trasporto.	Viene richiesto di: <ol style="list-style-type: none"> avviare la procedura di modifica arco; modificare i dati dell'arco nell'area informativa. 	N.I	N.S
TVFF26	L'utente intende avviare l'assistente vocale.	Viene richiesto di: <ol style="list-style-type: none"> aprire l'applicazione; premere sul pulsante relativo all'assistente vocale. 	N.I	N.S
TVOF1.1	L'utente intende disegnare il perimetro dell'asset sulla mappa.	Viene richiesto di: <ol style="list-style-type: none"> avviare la procedura di aggiunta asset; tracciare il perimetro dell'asset sulla mappa utilizzando gli strumenti a disposizione. 	N.I	N.S
TVOF1.2	L'utente intende compilare i dati dell'asset.	Viene richiesto di: <ol style="list-style-type: none"> avviare la procedura di aggiunta asset; compilare i campi richiesti nell'area informativa. 	N.I	N.S
TVOF10	L'utente intende eliminare un nodo.	Viene richiesto di: <ol style="list-style-type: none"> aprire l'applicazione; selezionare un nodo per visualizzarne le informazioni; premere sul pulsante "Elimina" nell'area informativa; confermare l'eliminazione del nodo. 	N.I	N.S

Test	Descrizione	Operazioni	Stato	Esito
TVOF11	L'utente intende aggiungere un nuovo arco.	Viene richiesto di: <ol style="list-style-type: none"> 1. avviare l'applicazione;; 2. inserire un asset se non ne sono presenti; 3. inserire almeno due nodi; 4. cliccare sul pulsante "+"; 5. selezionare "Aggiungi arco"; 6. selezionare il nodo di origine dell'arco; 7. selezionare il nodo di destinazione dell'arco; 8. compilare i dati dell'arco; 9. confermare l'aggiunta dell'arco. 	N.I	N.S
TVOF11.1	L'utente intende disegnare un arco.	Viene richiesto di: <ol style="list-style-type: none"> 1. avviare la procedura di aggiunta arco; 2. selezionare nodo di origine dell'arco; 3. selezionare nodo di destinazione dell'arco. 	N.I	N.S
TVOF12	L'utente intende visualizzare le informazioni di un arco.	Viene richiesto di: <ol style="list-style-type: none"> 1. aprire l'applicazione; 2. inserire un arco se non ne sono presenti; 3. selezionare l'arco dalla mappa per visualizzarne le informazioni nell'area informativa. 	N.I	N.S

Test	Descrizione	Operazioni	Stato	Esito
TVOF14	L'utente intende modificare un arco.	Viene richiesto di: <ol style="list-style-type: none"> 1. aprire l'applicazione; 2. selezionare un arco dalla mappa per visualizzarne le informazioni; 3. cliccare il pulsante "Modifica" nell'area informativa; 4. modificare il nodo di destinazione o di origine o i dati dell'arco; 5. confermare la modifica. 	N.I	N.S
TVOF14.1	L'utente intende modificare il nodo di origine di un arco.	Viene richiesto di: <ol style="list-style-type: none"> 1. avviare la procedura di modifica arco; 2. selezionare un arco dalla mappa per visualizzarne le informazioni; 3. cliccare sul pulsante ; 4. selezionare un nuovo nodo di origine dalla mappa. 	N.I	N.S
TVOF14.2	L'utente intende modificare il nodo di destinazione di un arco.	Viene richiesto di: <ol style="list-style-type: none"> 1. avviare la procedura di modifica arco; 2. selezionare un arco dalla mappa per visualizzarne le informazioni; 3. cliccare sul pulsante "Modifica destinazione" nell'area informativa; 4. selezionare un nuovo nodo di destinazione dalla mappa. 	N.I	N.S

Test	Descrizione	Operazioni	Stato	Esito
TVOF15	L'utente intende eliminare un arco.	Viene richiesto di: <ol style="list-style-type: none"> 1. aprire l'applicazione; 2. selezionare un arco per visualizzarne le informazioni; 3. premere sul pulsante "Elimina" nell'area informativa; 4. confermare l'eliminazione dell'arco. 	N.I	N.S
TVOF16	L'utente intende aggiungere un nuovo scenario di danno.	Viene richiesto di: <ol style="list-style-type: none"> 1. aprire l'applicazione; 2. premere sul pulsante "+"; 3. selezionare "Aggiungi scenario"; 4. disegnare lo scenario di danno sulla mappa; 5. compilare i dati dello scenario nell'area informativa; 6. confermare l'inserimento. 	N.I	N.S
TVOF16.1	L'utente intende compilare le informazioni dello scenario di danno.	Viene richiesto di: <ol style="list-style-type: none"> 1. avviare la procedura di inserimento scenario; 2. compilare le informazioni nell'area informativa. 	N.I	N.S
TVOF16.1.7	L'utente intende disegnare lo scenario di danno su mappa.	Viene richiesto di: <ol style="list-style-type: none"> 1. avviare la procedura di inserimento scenario; 2. disegnare lo scenario di danno su mappa con gli strumenti a disposizione. 	N.I	N.S

Test	Descrizione	Operazioni	Stato	Esito
TVOF17	L'utente intende visualizzare le informazioni di uno scenario di danno.	Viene richiesto di: <ol style="list-style-type: none"> 1. aprire l'applicazione; 2. selezionare la tab "Scenari" nell'area informativa; 3. selezionare lo scenario di danno da visualizzare dall'area informativa. 	N.I	N.S
TVOF19	L'utente intende modificare uno scenario.	Viene richiesto di: <ol style="list-style-type: none"> 1. aprire l'applicazione; 2. selezionare uno scenario di danno per visualizzarne le informazioni; 3. cliccare sul pulsante "Modifica" nell'area informativa; 4. modificare il disegno su mappa o i dati nell'area informativa; 5. confermare la modifica dello scenario. 	N.I	N.S
TVOF19.1	L'utente intende modificare le informazioni di uno scenario di danno.	Viene richiesto di: <ol style="list-style-type: none"> 1. avviare la procedura di modifica scenario di danno; 2. modificare le informazioni dello scenario nell'area informativa. 	N.I	N.S
TVOF19.1.7	L'utente intende modificare il disegno di uno scenario di danno.	Viene richiesto di: <ol style="list-style-type: none"> 1. avviare la procedura di modifica scenario di danno; 2. modificare il disegno dello scenario su mappa con gli strumenti a disposizione. 	N.I	N.S

Test	Descrizione	Operazioni	Stato	Esito
TVOF2	L'utente intende visualizzare le informazioni di un asset.	Viene richiesto di: <ol style="list-style-type: none"> 1. aprire l'applicazione; 2. cliccare sul perimetro di un asset. 	N.I	N.S
TVOF20	L'utente intende eliminare uno scenario di danno.	Viene richiesto di: <ol style="list-style-type: none"> 1. aprire l'applicazione; 2. selezionare la tab "Scenari" nell'area informativa; 3. selezionare uno scenario nell'area informativa per visualizzarne le informazioni; 4. cliccare sul pulsante "Elimina" nell'area informativa; 5. confermare l'eliminazione dello scenario. 	N.I	N.S
TVOF21	L'utente intende avviare l'analisi di danno.	Viene richiesto di: <ol style="list-style-type: none"> 1. aprire l'applicazione; 2. disegnare il processo produttivo sulla mappa, inserendo almeno un asset; 3. cliccare sul tab "Analisi" nell'area informativa; 4. selezionare su quali scenari di danno vuole calcolare l'analisi; 5. premere il pulsante "Avvia analisi". 	N.I	N.S

Test	Descrizione	Operazioni	Stato	Esito
TVOF22	L'utente intende visualizzare il risultato di un'analisi di danno precedentemente eseguita.	Viene richiesto di: <ol style="list-style-type: none"> 1. aprire l'applicazione; 2. cliccare sul tab "Analisi" nell'area informativa; 3. selezionare l'analisi di danno di cui vuole visualizzare i risultati. 	N.I	N.S
TVOF24.1	L'utente intende aumentare il livello di ingrandimento della mappa.	Viene richiesto di: <ol style="list-style-type: none"> 1. aprire l'applicazione; 2. cliccare sul pulsante che aumenta l'ingrandimento della mappa oppure scrollare verso l'alto con la rotellina del mouse oppure effettuare la gesture "pinch in" su tablet. 	N.I	N.S
TVOF24.2	L'utente intende diminuire il livello di ingrandimento della mappa.	Viene richiesto di: <ol style="list-style-type: none"> 1. aprire l'applicazione; 2. cliccare sul pulsante che diminuisce l'ingrandimento della mappa oppure scrollare verso il basso con la rotellina del mouse oppure effettuare la gesture "pinch out" su tablet. 	N.I	N.S
TVOF24.3	L'utente intende spostarsi sulla mappa.	Viene richiesto di: <ol style="list-style-type: none"> 1. aprire l'applicazione; 2. cliccare e trascinare su un punto della mappa oppure effettuare la gesture "pan" su tablet. 	N.I	N.S

Test	Descrizione	Operazioni	Stato	Esito
TVOF4	L'utente intende modificare un asset.	Viene richiesto di: <ol style="list-style-type: none"> 1. aprire l'applicazione; 2. selezionare il perimetro di un asset dalla mappa per visualizzarne le informazioni; 3. cliccare sul pulsante "Modifica" nell'area informativa; 4. modificare il disegno o i dati dell'asset; 5. confermare la modifica. 	N.I	N.S
TVOF4.1	L'utente intende modificare il disegno di un asset.	Viene richiesto di: <ol style="list-style-type: none"> 1. avviare la procedura di modifica asset; 2. modificare il perimetro dell'asset con gli strumenti a disposizione. 	N.I	N.S
TVOF4.2	L'utente intende modificare i dati dell'asset.	Viene richiesto di: <ol style="list-style-type: none"> 1. avviare la procedura di modifica asset; 2. modificare i dati dell'asset nell'area informativa. 	N.I	N.S
TVOF47	L'utente intende eliminare i risultati di un'analisi di danno precedentemente calcolata.	Viene richiesto di: <ol style="list-style-type: none"> 1. aprire l'applicazione; 2. cliccare sul tab "Analisi" nell'area informativa; 3. selezionare un'analisi di danno per visualizzarne i risultati; 4. cliccare sul pulsante "Elimina" nell'area informativa; 5. confermare l'eliminazione dell'analisi. 	N.I	N.S

Test	Descrizione	Operazioni	Stato	Esito
TVOF5	L'utente intende eliminare un asset.	Viene richiesto di: <ol style="list-style-type: none"> 1. aprire l'applicazione; 2. selezionare il perimetro di un asset per modificarne le informazioni; 3. cliccare sul pulsante "Elimina" nell'area informativa; 4. confermare di voler eliminare un asset. 	N.I	N.S
TVOF6	L'utente intende inserire un nuovo nodo.	Viene richiesto di: <ol style="list-style-type: none"> 1. aprire l'applicazione; 2. inserire almeno un asset, se non ne sono presenti; 3. cliccare sul pulsante "+"; 4. selezionare "Aggiungi nodo"; 5. selezionare l'asset di appartenenza del nodo sulla mappa; 6. posizionare il nodo all'interno dell'asset; 7. compilare i dati del nodo; 8. confermare l'aggiunta del nodo. 	N.I	N.S
TVOF6.1	L'utente intende selezionare l'asset di appartenenza del nodo.	Viene richiesto di: <ol style="list-style-type: none"> 1. avviare la procedura di aggiunta nodo; 2. cliccare sul perimetro di un asset sulla mappa. 	N.I	N.S

Test	Descrizione	Operazioni	Stato	Esito
TVOF6.12	L'utente intende compilare i dati del nodo.	Viene richiesto di: <ol style="list-style-type: none"> 1. avviare la procedura di aggiunta nodo; 2. selezionare l'asset di appartenenza del nodo e posizionarlo all'interno del perimetro; 3. compilare i dati del nodo nell'area informativa. 	N.I	N.S
TVOF6.2	L'utente intende posizionare un nodo all'interno di un asset.	Viene richiesto di: <ol style="list-style-type: none"> 1. avviare la procedura di aggiunta nodo; 2. selezionare l'asset di appartenenza del nodo; 3. posizionare il nodo all'interno dell'asset. 	N.I	N.S
TVOF7	L'utente intende visualizzare le informazioni di un nodo.	Viene richiesto di: <ol style="list-style-type: none"> 1. avviare l'applicazione; 2. selezionare un nodo dalla mappa. 	N.I	N.S
TVOF9	L'utente intende modificare un nodo.	Viene richiesto di: <ol style="list-style-type: none"> 1. aprire l'applicazione; 2. selezionare un nodo dalla mappa per visualizzarne le informazioni; 3. modificare l'asset di appartenenza o i dati del nodo; 4. confermare le modifiche. 	N.I	N.S

Test	Descrizione	Operazioni	Stato	Esito
TVOF9.1	L'utente intende modificare l'asset di appartenenza del nodo.	Viene richiesto di: <ol style="list-style-type: none"> 1. avviare la procedura di modifica nodo; 2. cliccare sul pulsante "Modifica asset di appartenenza"; 3. selezionare il perimetro di un nuovo asset dalla mappa; 4. posizionare il nodo all'interno dell'asset. 	N.I	N.S
TVOF9.12	L'utente intende modificare i dati del nodo.	Viene richiesto di: <ol style="list-style-type: none"> 1. avviare la procedura di modifica nodo; 2. modificare i dati nell'area informativa. 	N.I	N.S

Tabella 2: Riepilogo test di validazione

3.2 Test di sistema

I test di sistema servono a verificare il corretto funzionamento delle componenti dell'intero sistema.

Test	Descrizione	Stato	Esito
TSFF11.3	Viene verificato che il sistema permetta di compilare i dati dell'arco di tipo Trasporto.	N.I	N.S
TSFF14.5	Viene verificato che il sistema permetta la modifica dei dati dell'arco di tipo Trasporto.	N.I	N.S
TSFF16	Viene verificato che il sistema permetta l'aggiunta di un nuovo scenario di danno.	N.I	N.S
TSFF16.1	Viene verificato che il sistema permetta la compilazione delle informazioni di uno scenario di danno durante l'inserimento di uno scenario.	N.I	N.S

Test	Descrizione	Stato	Esito
TSFF16.1.7	Viene verificato che il sistema permetta il disegno di uno scenario di danno su mappa.	N.I	N.S
TSFF17	Viene verificato che il sistema permetta la visualizzazione di uno scenario di danno.	N.I	N.S
TSFF19	Viene verificato che il sistema permetta la modifica di uno scenario di danno precedentemente inserito.	N.I	N.S
TSFF19.1	Viene verificato che il sistema permetta la modifica delle informazioni dello scenario di danno.	N.I	N.S
TSFF19.1.7	Viene verificato che il sistema permetta la modifica del disegno dello scenario di danno.	N.I	N.S
TSFF20	Viene verificato che il sistema permetta l'eliminazione di uno scenario di danno precedentemente inserito.	N.I	N.S
TSFF21	Viene verificato che il sistema permetta l'avvio di un'analisi di danno.	N.I	N.S
TSFF22	Viene verificato che il sistema permetta la visualizzazione del risultato di un'analisi di danno precedentemente eseguita su mappa.	N.I	N.S
TSFF24.4	Viene verificato che il sistema permetta di cambiare la modalità di visualizzazione della mappa.	N.I	N.S
TSFF25	Viene verificato che il sistema permetta l'avvio e la fruizione del tutorial.	N.I	N.S
TSFF26	Viene verificato che il sistema permetta l'avvio e la fruizione dell'assistente vocale.	N.I	N.S

Test	Descrizione	Stato	Esito
TSFF47	Viene verificato che il sistema permetta l'eliminazione dei risultati di un'analisi di danno precedentemente calcolata.	N.I	N.S
TSOF1	Viene verificato che il sistema permetta l'aggiunta di un nuovo asset.	N.I	N.S
TSOF1.1	Viene verificato che il sistema permetta di disegnare il perimetro dell'asset su mappa durante l'aggiunta di un nuovo asset.	N.I	N.S
TSOF1.2	Viene verificato che il sistema permetta di compilare i dati dell'asset durante l'aggiunta dell'asset.	N.I	N.S
TSOF10	Viene verificato che il sistema permetta l'eliminazione di un nodo precedentemente inserito.	N.I	N.S
TSOF11	Viene verificato che il sistema permetta l'aggiunta di un nuovo arco.	N.I	N.S
TSOF11.1	Viene verificato che il sistema permetta di disegnare un arco, scegliendo nodo di origine e di destinazione.	N.I	N.S
TSOF12	Viene verificato che il sistema permetta la visualizzazione delle informazioni di un arco precedentemente inserito.	N.I	N.S
TSOF14	Viene verificato che il sistema permetta la modifica di un arco precedentemente inserito.	N.I	N.S
TSOF14.1	Viene verificato che il sistema permetta la modifica del nodo di origine dell'arco.	N.I	N.S
TSOF14.2	Viene verificato che il sistema permetta la modifica del nodo di destinazione dell'arco.	N.I	N.S
TSOF15	Viene verificato che il sistema permetta l'eliminazione di un arco precedentemente inserito.	N.I	N.S

Test	Descrizione	Stato	Esito
TSOF2	Viene verificato che il sistema permetta la visualizzazione di un asset precedentemente inserito.	N.I	N.S
TSOF24.1	Viene verificato che il sistema permetta l'aumento del livello di ingrandimento della mappa.	N.I	N.S
TSOF24.2	Viene verificato che il sistema permetta la diminuzione del livello di ingrandimento della mappa.	N.I	N.S
TSOF24.3	Viene verificato che il sistema permetta lo spostamento sulla mappa.	N.I	N.S
TSOF4	Viene verificato che il sistema permetta la modifica di un asset precedentemente inserito.	N.I	N.S
TSOF4.1	Viene verificato che il sistema permetta la modifica del perimetro di un asset precedentemente inserito.	N.I	N.S
TSOF4.2	Viene verificato che il sistema permetta la modifica dei dati di un asset precedentemente inserito.	N.I	N.S
TSOF5	Viene verificato che il sistema permetta l'eliminazione di un asset precedentemente inserito.	N.I	N.S
TSOF6	Viene verificato che il sistema permetta l'aggiunta di un nuovo nodo.	N.I	N.S
TSOF6.1	Viene verificato che il sistema permetta la selezione di un asset di appartenenza durante l'aggiunta di un nuovo nodo.	N.I	N.S
TSOF6.12	Viene verificato che il sistema permetta la compilazione dei dati del nodo durante l'aggiunta di un nuovo nodo.	N.I	N.S

Test	Descrizione	Stato	Esito
TSOF6.2	Viene verificato che il sistema permetta il posizionamento di un nodo all'interno del perimetro di un asset durante l'aggiunta di un nuovo nodo.	N.I	N.S
TSOF7	Viene verificato che il sistema permetta la visualizzazione delle informazioni di un nodo precedentemente inserito.	N.I	N.S
TSOF9	Viene verificato che il sistema permetta la modifica di un nodo precedentemente inserito.	N.I	N.S
TSOF9.1	Viene verificato che il sistema permetta di modificare l'asset di appartenenza di un nodo.	N.I	N.S
TSOF9.12	Viene verificato che il sistema permetta la modifica dei dati del nodo.	N.I	N.S

Tabella 3: Riepilogo test di sistema

3.3 Test di integrazione

I test di integrazione servono a verificare il corretto funzionamento di più unità. Più precisamente, l'obiettivo è quello di testare i vari *package*, sia singolarmente che nel loro insieme.

Test	Package	Descrizione	Stato	Esito
TI1	DeGeOP	Viene verificato che l'applicazione Web carichi correttamente le librerie JavaScript utilizzate.	N.I	N.S
TI2	CallManagerPkg	Viene verificato che sia funzionante il collegamento del gestore delle chiamate con il server RiskApp	N.I	N.S
TI3	CallManagerPkg	Viene verificato che i dati inviati dal gestore delle chiamate siano salvati correttamente sul server RiskApp	N.I	N.S
TI4	ActionCreatorsPkg	Viene verificato che l'Action Creators riceva correttamente gli input dalla View e crei le action ad essi associate.	N.I	N.S

T15	ActionCreatorsPkg	Viene verificato che il gestore delle chiamate invii una richiesta all'Action Creators per una creazione di un'azione.	N.I	N.S
T16	ActionCreatorsPkg	Viene verificato che l'Action Creators carichi correttamente lo store	N.I	N.S
T17	StorePkg	Viene verificato che lo Store comunichi con il Reducer	N.I	N.S
T18	ReducerPkg	Viene verificato che il Reducer gestisca correttamente tutte le azioni inviate allo store.	N.I	N.S
T19	ViewPkg	Viene verificato che la View funzioni correttamente permettendo il caricamento e la visualizzazione della pagina	N.I	N.S

Tabella 4: Riepilogo test di unità

3.4 Test di unità

I test di unità servono a verificare il corretto funzionamento della singola unità, ovvero della più piccola parte di lavoro realizzabile dal singolo programmatore.

Test	Descrizione e componenti	Stato	Esito
TU1	Viene verificato che il concretePolygon sia creato correttamente. Metodi testati: • PolygonPkg::ConcretePolygonFactory.createPolygon().	I	S
TU2	Viene verificato che i campi di un oggetto Asset vengano validati o meno. Metodi testati: • ProcessPkg::Asset.assetIsValid(); • ProcessPkg::Asset.testDescriptionValidation(); • ProcessPkg::Asset.testNameValidation(); • ProcessPkg::Asset.testSurfaceValidation(); • ProcessPkg::Asset.testUnitValueValidation().	I	S

TU3	Viene verificato che i campi di un oggetto Node vengano validati o meno. Metodi testati: <ul style="list-style-type: none"> • <code>ProcessPkg::Node.nodesValid()</code>; • <code>ProcessPkg::Node.testNameValidation()</code>. 	I	S
TU4	Viene verificato che i campi di un oggetto ResourceNode vengano validati o meno. Metodi testati: <ul style="list-style-type: none"> • <code>ProcessPkg::ResourceNode.nodesValid()</code>. 	I	S
TU5	Viene verificato che i campi di un oggetto ExitNode vengano validati o meno. Metodi testati: <ul style="list-style-type: none"> • <code>ProcessPkg::ExitNode.nodesValid()</code>. 	I	S
TU6	Viene verificato che i campi di un oggetto MachineNode vengano validati o meno. Metodi testati: <ul style="list-style-type: none"> • <code>ProcessPkg::MachineNode.nodesValid()</code>; • <code>ProcessPkg::MachineNode.testCapacityValidation()</code>; • <code>ProcessPkg::MachineNode.testProcessingTimeValidation()</code>; • <code>ProcessPkg::MachineNode.testValueValidation()</code>. 	I	S
TU7	Viene verificato che i campi di un oggetto QueueNode vengano validati o meno. Metodi testati: <ul style="list-style-type: none"> • <code>ProcessPkg::QueueNode.nodesValid()</code>; • <code>ProcessPkg::QueueNode.testCapacityValidation()</code>. 	I	S
TU8	Viene verificato che i campi di un oggetto SourceNode vengano validati o meno. Metodi testati: <ul style="list-style-type: none"> • <code>ProcessPkg::SourceNode.nodesValid()</code>; • <code>ProcessPkg::SourceNode.testLeadTimeValidation()</code>. 	I	S
TU9	Viene verificato che l'AssetReducer esegua correttamente l'azione ricevuta sulla lista di Asset. Metodi testati: <ul style="list-style-type: none"> • <code>ReducerPkg::AssetReducer.assetReducer()</code>. 	I	S

TU10	Viene verificato che l'NodeReducer correttamente l'azione ricevuta sulla lista di Node. Metodi testati: • ReducerPkg::NodeReducer.nodeReducer().	I	S
TU11	Viene verificato che l'EdgeReducer esegua correttamente l'azione ricevuta sulla lista di Edge. Metodi testati: • ReducerPkg::EdgeReducer.edgeReducer().	I	S
TU12	Viene verificato che l'OptionReducer esegua correttamente l'azione ricevuta sull'oggetto Options. Metodi testati: • ReducerPkg::OptionReducer.optionReducer().	I	S
TU13	Viene verificato che il Reducer invochi il reducer corretto rispetto all'azione ricevuta. Metodi testati: • ReducerPkg::Reducer.generalReducer().	I	S
TU14	Viene verificato che l>ActionCreator crei correttamente le azioni relative ad un oggetto Asset. Metodi testati: • ActionCreatorsPkg::AssetActionCreator.deleteAsset(); • ActionCreatorsPkg::AssetActionCreator.editAsset(); • ActionCreatorsPkg::AssetActionCreator.insertAsset().	I	S
TU15	Viene verificato che l'NodeCreator crei correttamente le azioni relative ad un oggetto Node. Metodi testati: • ActionCreatorsPkg::NodeActionCreator.deleteNode(); • ActionCreatorsPkg::NodeActionCreator.editNode(); • ActionCreatorsPkg::NodeActionCreator.insertNode().	I	S
TU16	Viene verificato che l'EdgeCreator crei correttamente le azioni relative ad un oggetto Edge. Metodi testati: • ActionCreatorsPkg::EdgeActionCreator.deleteEdge(); • ActionCreatorsPkg::EdgeActionCreator.editEdge(); • ActionCreatorsPkg::EdgeActionCreator.insertEdge().	I	S

TU17	Viene verificato che l'OptionCreator crei correttamente le azioni relative all'oggetto Options. Metodi testati: • ActionCreatorsPkg::OptionActionCreator.insertOptions().	I	S
TU19	Viene verificato che DeGeOPView modifichi correttamente il suo stato tramite il metodo handleChange. Metodi testati: • DeGeOPViewPkg::DeGeOPView.handleChange().	N.I	N.S
TU20	Viene verificato che DeGeOPView esegua le chiamate allo store con gli oggetti corretti. Metodi testati: • DeGeOPViewPkg::DeGeOPView.assetSaveEditAction(); • DeGeOPViewPkg::DeGeOPView.assetSaveInsertAction(); • DeGeOPViewPkg::DeGeOPView.doDeleteAsset(); • DeGeOPViewPkg::DeGeOPView.doDeleteEdge(); • DeGeOPViewPkg::DeGeOPView.doDeleteNode(); • DeGeOPViewPkg::DeGeOPView.nodeSaveEditAction(); • DeGeOPViewPkg::DeGeOPView.nodeSaveInsertAction().	N.I	N.S
TU21	Viene verificato che DeGeOPView crei la sidebar corretta rispetto al suo stato. Metodi testati: • DeGeOPViewPkg::DeGeOPView.sidebarFactory().	N.I	N.S
TU22	Viene verificato che InsertAssetSidebar venga renderizzata con le proprietà corrette. Metodi testati: • SidebarPkg::InsertAssetSidebar.render().	N.I	N.S
TU23	Viene verificato che InsertNodeSidebar sidebar venga renderizzata con le proprietà corrette. Metodi testati: • SidebarPkg::InsertNodeSidebar.render().	N.I	N.S
TU24	Viene verificato che InsertEdgeSidebar sidebar venga renderizzata con le proprietà corrette. Metodi testati: • SidebarPkg::InsertEdgeSidebar.render().	N.I	N.S

TU25	Viene verificato che ViewAssetSidebar sidebar venga renderizzata con le proprietà corrette. Metodi testati: • SidebarPkg::ViewAssetSidebar.render().	N.I	N.S
TU26	Viene verificato che ViewNodeSidebar sidebar venga renderizzata con le proprietà corrette. Metodi testati: • SidebarPkg::ViewNodeSidebar.render().	N.I	N.S
TU27	Viene verificato che ViewEdgeSidebar sidebar venga renderizzata con le proprietà corrette. Metodi testati: • SidebarPkg::ViewEdgeSidebar.render().	N.I	N.S
TU28	Viene verificato che TwoButtons venga renderizzata con le proprietà corrette. Metodi testati: • ButtonsPkg::TwoButtons.render().	N.I	N.S
TU29	Viene verificato che ThreeButtons venga renderizzata con le proprietà corrette. Metodi testati: • ButtonsPkg::ThreeButtons.render().	N.I	N.S

Tabella 5: Riepilogo test di unità

A CMM

Il **CMM_G** (Capability Maturity Model) è un modello che mira a migliorare e rifinire i processi software di un'organizzazione. Il modello descrive un percorso evolutivo a cinque livelli riguardante processi sempre più maturi e organizzati.

Il CMM è stato sviluppato dal **SEI_G** e promosso e finanziato dal Dipartimento della Difesa statunitense per valutare la qualità dei processi software delle organizzazioni che collaboravano con esso.

Il modello fornisce:

- una base concettuale a cui appoggiarsi per valutare il livello dei processi;
- un insieme di best practices consolidate negli anni da esperti e utilizzatori;
- un linguaggio comune e una visione condivisa;
- un metodo per definire un miglioramento in ambito organizzativo.

A.1 Struttura

Il modello è costituito da cinque aspetti:

- **livelli di maturità:** il CMM identifica un processo continuo di maturazione a cinque livelli (in cui il maggiore è uno stato ideale dove i processi sono sistematicamente gestiti da una combinazione di ottimizzazione e miglioramento del processo);
- **area chiave di processo:** identifica un insieme di attività correlate che, quando eseguite assieme, raggiungono un insieme di obiettivi considerati importanti;
- **obiettivi:** gli obiettivi di un'area chiave di processo riassumono gli stati che devono sussistere affinché tale area sia implementata in modo efficace e duraturo. La quantità di obiettivi soddisfatti indica il livello di capability raggiunto dall'organizzazione in un dato livello di maturità. Gli obiettivi denotano l'ambito, i limiti e lo scopo di ogni area chiave di processo;
- **caratteristiche comuni:** includono le pratiche che implementano e regolamentano le aree chiave di processo. Esistono cinque tipi di caratteristiche comuni:
 - impegno nell'esecuzione;
 - abilità nell'esecuzione;
 - attività eseguite;
 - misurazioni e analisi;
 - verifica e implementazione.
- **pratiche chiave:** descrivono gli elementi di infrastruttura e prassi che contribuiscono all'implementazione e regolamentazione dell'area.

A.2 Livelli

Sono presenti cinque livelli:

- **livello 1 - Iniziale:** i processi in questo livello hanno la tendenza ad essere non documentati e in uno stato di continuo cambiamento. Date queste caratteristiche, l'esito molto spesso dipende dallo sforzo dei singoli e non si considera essere ripetibile. I processi vengono riadattati di volta in volta, risultando caotici e scarsamente controllabili;
- **livello 2 - Ripetibile:** I processi di questo livello sono generalmente ripetibili, eventualmente con buoni risultati. La disciplina, se presente, pur non essendo rigorosa, aiuta a sostenere i processi durante i periodi di elevato carico di lavoro;
- **livello 3 - Definito:** i processi cominciano ad essere standardizzati, in quanto la disciplina è più rigorosa e la documentazione più completa. Inoltre sono soggetti ad un certo livello di miglioramento nel lungo periodo;
- **livello 4 - Gestito:** i processi sono controllati quantitativamente in accordo alle metriche di processo prestabilite. L'amministrazione aziendale può adeguare e adattare i processi a particolari progetti senza perdite sostanziali di qualità o deviazioni dalle specifiche;
- **livello 5 - Ottimizzato:** i processi in questo livello hanno come obiettivo il miglioramento continuo delle loro performance attraverso miglioramenti tecnologici sia incrementali che innovativi.

B ISO/IEC 9126

Lo standard *ISO_G/IEC_G 9126:2001* prevede una serie di normative e linee guida nate dalla collaborazione tra ISO e IEC per descrivere un modello della qualità del software.

Esso si suddivide in quattro parti:

- modello della qualità del software (9126-1);
- metriche per la qualità esterna (9126-2);
- metriche per la qualità interna (9126-3);
- metriche per la qualità in uso (9126-4).

B.1 Modello della qualità del software

Il modello della qualità del software viene definito nella prima parte dello standard e viene suddiviso in:

- modello della qualità esterna e interna;
- modello della qualità in uso.

Le caratteristiche contenute in tali modelli sono misurabili attraverso l'utilizzo di metriche.

B.1.1 Modello della qualità esterna e interna

Il modello della qualità esterna e interna classifica la qualità del software con sei caratteristiche generali:

- **funzionalità:** rappresenta la capacità del prodotto software di fornire le funzioni necessarie per operare in determinate condizioni, cioè in un determinato contesto;
- **affidabilità:** rappresenta la capacità del prodotto software di mantenere un certo livello di prestazioni quando viene usato in condizioni specifiche e per un intervallo di tempo fissato;
- **usabilità:** rappresenta la capacità del prodotto software di essere comprensibile. Un software è considerato usabile in proporzione alla facilità con cui gli utenti operano per sfruttare a pieno le funzionalità che il software realizza;
- **efficienza:** rappresenta la capacità del prodotto software di realizzare le funzioni richieste nel minor tempo possibile, utilizzando le risorse a disposizione nel miglior modo possibile;
- **manutenibilità:** rappresenta la capacità del prodotto software di essere modificato (a costi accessibili e in tempi rapidi). Le modifiche possono includere correzioni, adattamenti o migliorie del software; Le ultime possono essere richieste in seguito a cambiamenti nell'ambiente, nei requisiti o nelle specifiche funzionali;
- **portabilità:** rappresenta la capacità del prodotto software di poter essere trasportato da un ambiente all'altro (in modo sufficientemente veloce). L'ambiente include aspetti hardware e software.

B.1.2 Modello della qualità in uso

Gli attributi del modello della qualità in uso vengono suddivisi nelle seguenti quattro categorie:

- **efficacia:** rappresenta la capacità del prodotto software di permettere all'utente di raggiungere obiettivi specifici con accuratezza e completezza in uno specifico contesto d'utilizzo;
- **produttività:** rappresenta la capacità del prodotto software di permettere all'utente di impiegare un numero definito di risorse, in relazione all'efficienza raggiunta in uno specifico contesto di utilizzo;
- **sicurezza fisica:** rappresenta la capacità del prodotto software di raggiungere un livello accettabile di rischio per i dati, le persone, il business, la proprietà o gli ambienti in uno specifico contesto di utilizzo;
- **soddisfazione:** rappresenta la capacità del prodotto software di soddisfare gli utenti in uno specifico contesto di utilizzo.

B.2 Qualità esterna e relative metriche

È la qualità del prodotto software vista dall'esterno nel momento in cui esso viene eseguito e testato in un ambiente di prova. Le metriche associate ne misurano i comportamenti rilevabili:

- dai test;
- dall'operabilità;
- dall'osservazione in un contesto specifico.

Tali metriche vengono selezionate sulla base delle caratteristiche che il prodotto finale dovrà dimostrare durante la sua esecuzione.

B.3 Qualità interna e relative metriche

È la qualità del prodotto software vista dall'interno e fa riferimento alle caratteristiche implementative quali la sua architettura e il codice che ne deriva.

Le metriche associate si applicano al software non eseguibile (es: il codice sorgente) e alla documentazione. Le misure effettuate permettono di prevedere il livello di qualità esterna ed in uso del prodotto finale poiché gli attributi interni influenzano le caratteristiche esterne e quelle in uso.

B.4 Qualità in uso e relative metriche

È la qualità del prodotto software dal punto di vista dell'utilizzatore che ne fa uso all'interno di uno specifico sistema e contesto. Le metriche associate misurano il grado con cui il prodotto software permette agli utenti di svolgere, in un contesto operativo specifico, le proprie attività in modo:

- efficace;
- produttivo;
- sicuro;
- soddisfacente.

C PDCA

Il **PDCA_G** (conosciuto anche come "Ciclo di Deming" o "Ciclo di Shewhart") è un metodo per la gestione delle attività di processo ripetibili e misurabili e per la manutenibilità dei processi stessi. È un metodo iterativo suddiviso in quattro *fasi_G* (Plan-Do-Check-Act, da cui l'acronimo) e assicura un non decremento della qualità ad ogni ciclo. Fissati degli obiettivi di miglioramento desiderati si iterano le attività previste dal PDCA fino al raggiungimento degli stessi. I miglioramenti ai quali si fa riferimento sono legati all'efficienza e all'efficacia. Migliorare l'efficienza significa usare meno risorse per fare lo stesso lavoro. Migliorare l'efficacia significa divenire più conformi alle aspettative.

C.1 Fasi

Sono presenti quattro fasi:

- **Plan:** vengono definiti gli obiettivi di miglioramento, le strategie da utilizzare per perseguire tali obiettivi e il modo in cui queste verranno utilizzate.
Per far ciò si svolgono i seguenti passi:
 1. si svolge una prima fase di identificazione del problema (ad esempio un processo da migliorare) nella quale saranno raccolti dei dati in seguito a delle misurazioni;
 2. viene analizzato il problema e vengono individuati gli aspetti negativi, decidendone la loro importanza e le priorità di intervento;
 3. vengono definiti gli obiettivi di massima in modo chiaro e quantitativo, indicando i benefici ottenibili con il loro raggiungimento. Vengono inoltre specificati i tempi necessari per la loro attuazione, gli indicatori e gli strumenti di controllo necessari.
- **Do:** viene attuato ciò che è stato pianificato per risolvere il problema. Nello stesso tempo si devono anche raccogliere i dati necessari all'analisi che verrà svolta in seguito;
- **Check:** consiste nel verificare i risultati ottenuti (per efficienza ed efficacia) in seguito all'attuazione delle strategie di miglioramento. Essi saranno analizzati e studiati (anche attraverso grafici e tabelle riassuntive) in modo tale da avere una visione chiara di quanto rilevato. Se gli obiettivi sono stati raggiunti, ovvero se è avvenuto un miglioramento, si può passare alla fase successiva; in caso contrario è necessario ripetere il ciclo PDCA sullo stesso problema, analizzando gli stadi del ciclo precedente e individuando le cause del mancato raggiungimento degli obiettivi stabiliti.

L'esito del processo può essere di tre tipi:

- miglioramento secondo le aspettative;
 - miglioramento superiore alle aspettative;
 - miglioramento inferiore alle aspettative.
- **Act:** i miglioramenti individuati vengono regolamentati e integrati nello standard dell'organizzazione e tutti i membri del **gruppo_G** vengono informati e conseguentemente formati. Verrà quindi eseguita una nuova iterazione dell'intero ciclo.

D Resoconto delle attività di verifica

Il resoconto delle attività di verifica è stato suddiviso per obiettivo. All'interno di ogni obiettivo è presente un resoconto per ogni periodo. Per avere una lista dei periodi, consultare il *Piano di progetto v3.0.0*.

D.1 Verifica dei processi

Seguono gli esiti delle verifiche riguardanti i processi. Per avere informazioni dettagliate sugli scopi dei processi e sulle attività che li compongono, consultare le *Norme di progetto v3.0.0*.

D.1.1 Tutti i processi

D.1.1.1 Miglioramento costante

D.1.1.1.1 Periodo An

Processo	Metrica	Valore	Giudizio
Fornitura	LCMM	2	Accettabile
Sviluppo	LCMM	2	Accettabile
Documentazione	LCMM	2	Accettabile
Verifica	LCMM	2	Accettabile
Gestione delle infrastrutture	LCMM	2	Accettabile
Gestione dei processi	LCMM	2	Accettabile
Apprendimento	LCMM	2	Accettabile

Tabella 6: Resoconto miglioramento costante - periodo An

Il livello CMM_G dei processi di Fornitura, Sviluppo, Documentazione, Verifica, Gestione delle infrastrutture, Gestione di Processo e Apprendimento in questo periodo è pari a 2. Dopo lo stato iniziale, durato quasi fino a metà periodo, in cui i processi si trovavano in uno stato caotico, il rispetto delle *Norme di progetto v1.0.0* e l'adozione di strumenti automatici ha portato ad un guadagno di ripetibilità. Alcuni esempi di tali strumenti sono i correttori ortografici e lo script per il calcolo dell'indice di leggibilità per quanto riguarda i processi di Documentazione e Verifica e l'utilizzo di $Trender_G$ per quanto riguarda l'attività di analisi dei requisiti del processo di Sviluppo.

Tutti i processi non sono standardizzati ad un livello tale da raggiungere il livello 3 della scala. Inoltre, la disciplina non è ancora molto rigorosa. L'obiettivo per i prossimi periodi è migliorare tale livello.

D.1.1.1.2 Periodo PI

Processo	Metrica	Valore	Giudizio
Sviluppo	LCMM	2	Accettabile
Documentazione	LCMM	3	Accettabile
Verifica	LCMM	3	Accettabile
Gestione dei processi	LCMM	3	Accettabile
Gestione delle infrastrutture	LCMM	2	Accettabile
Gestione della configurazione	LCMM	3	Accettabile
Apprendimento	LCMM	2	Accettabile

Tabella 7: Resoconto miglioramento costante - periodo PI

Il livello CMM_G dei processi di Sviluppo, Gestione delle infrastrutture e Apprendimento in questo periodo è pari a 2. Per quanto riguarda il processo di Sviluppo, il livello CMM è rimasto invariato rispetto al periodo precedente in quanto è stata introdotta l'attività di progettazione ad alto livello. Essendo un'attività mai svolta prima dai membri del $team_G$, la disciplina pur essendo standardizzata non è ancora rigorosa. I processi di Gestione delle infrastrutture e di Apprendimento, non risultano ancora completamente ripetibili.

Il livello CMM dei processi di Documentazione, Verifica, Gestione dei processi e Gestione della configurazione in questo periodo è pari a 3. La standardizzazione dei processi sopracitati risulta maggiore a quella rilevata nel periodo precedente, con una disciplina rigorosa. L'obiettivo per i prossimi periodi è migliorare il livello CMM.

D.1.1.1.3 Periodo PCV

Processo	Metrica	Valore	Giudizio
Sviluppo	LCMM	1	Non accettabile
Documentazione	LCMM	3	Accettabile
Verifica	LCMM	3	Accettabile
Validazione	LCMM	3	Accettabile
Gestione dei processi	LCMM	3	Accettabile
Gestione delle infrastrutture	LCMM	2	Accettabile
Gestione della configurazione	LCMM	3	Accettabile
Apprendimento	LCMM	2	Accettabile

Tabella 8: Resoconto miglioramento costante - periodo PCV

Il livello CMM_G dei processi di Gestione delle infrastrutture e Apprendimento in questo periodo è pari a 2. Per quanto riguarda il processo di Sviluppo, il livello CMM è rimasto invariato rispetto

al periodo precedente in quanto è stata introdotta l'attività di codifica. Essendo un'attività mai svolta prima dai membri del *team_G*, la disciplina pur essendo standardizzata non è ancora rigorosa. I processi di Gestione delle infrastrutture e di Apprendimento, non risultano ancora completamente ripetibili.

Il livello CMM dei processi di Documentazione, Verifica, Gestione dei processi e Gestione della configurazione in questo periodo è pari a 3.

Il livello CMM del processo di Sviluppo è 1. Le norme di codifica sono estremamente superficiali e la loro applicazione è quasi nulla. I verificatori ritengono che il codice prodotto durante questo periodo sia di qualità inaccettabile. Sono già state prese decisioni per cercare di risolvere la situazione.

Tutti i processi non sono ancora ad un livello di ripetibilità tale da poter essere riadattati senza perdite sostanziali di qualità, pertanto nessuno raggiunge il livello 4.

D.1.1.2 Rispetto della pianificazione

Per una descrizione dell'obiettivo, consultare [Rispetto della pianificazione](#).

D.1.1.2.1 Periodo An

Metrica	Valore	Giudizio
SV	3 giorni	Accettabile

Il ritardo riscontrato nel periodo di Analisi è pari a 3 giorni. Dato che il ritardo è all'interno della soglia di accettabilità, il *teamG* è ancora in grado di rispettare la scadenza. L'obiettivo è cercare di evitare ritardi nei periodi successivi.

D.1.1.2.2 Periodo PI

Metrica	Valore	Giudizio
SV	4 giorni	Accettabile

Il ritardo riscontrato nel periodo di Progettazione Logica è pari a 4 giorni. Il ritardo rilevato si pone ai limiti della soglia di accettabilità. Il team è ancora in grado di rispettare la scadenza ma l'obiettivo è cercare di evitare ritardi aggiuntivi nei periodi successivi.

D.1.1.2.3 Periodo PCV

Metrica	Valore	Giudizio
SV	10 giorni	Non accettabile

Il ritardo riscontrato nel periodo di Progettazione di dettaglio, Codifica e Validazione è pari a 10 giorni. La metrica assume un valore non accettabile. I motivi del ritardo sono dovuti principalmente ai problemi con le tecnologie riscontrati in questo periodo e alla scarsa applicazione delle norme di codifica.

L'obiettivo per il prossimo periodo è cercare di recuperare il tempo perso, concentrandosi maggiormente sulle funzionalità obbligatorie e lasciando in secondo piano quelle opzionali.

D.1.1.3 Rispetto del budget

Per una descrizione dell'obiettivo, consultare [Rispetto del budget](#).

D.1.1.3.1 Periodo An

Metrica	Valore	Giudizio
CV	0%	Ottimale

Non sono state riscontrate spese aggiuntive. La metrica assume quindi un valore ottimale.

D.1.1.3.2 Periodo PI

Metrica	Valore	Giudizio
CV	0%	Ottimale

Non sono state riscontrate spese aggiuntive. La metrica assume quindi un valore ottimale.

D.1.1.3.3 Periodo PCV

Metrica	Valore	Giudizio
CV	-1.34%	Ottimale

In questo periodo è stato speso meno di quanto preventivato, quindi la metrica assume un valore ottimale.

D.1.1.4 Completezza dell'analisi dei rischi

Per una descrizione dell'obiettivo, consultare [Completezza dell'analisi dei rischi](#).

D.1.1.4.1 Periodo An

Metrica	Valore	Giudizio
RNP	0 rischi	Ottimale

Dall'inizio del progetto non sono sorti rischi non preventivati, pertanto la metrica assume un valore ottimale.

D.1.1.4.2 Periodo PI

Metrica	Valore	Giudizio
RNP	0 rischi	Ottimale

Nel periodo di Progettazione Logica non sono sorti rischi non preventivati, pertanto la metrica assume un valore ottimale.

D.1.1.4.3 Periodo PCV

Metrica	Valore	Giudizio
RNP	0 rischi	Ottimale

Nel periodo preso in esame non sono sorti rischi non preventivati, pertanto la metrica assume un valore ottimale.

D.1.2 Processo di documentazione

Alcuni obiettivi relativi al processo di documentazione non hanno un resoconto per il periodo di Analisi in quanto sono stati stabiliti alla fine di esso.

D.1.2.1 Impegno nella documentazione

Per una descrizione dell'obiettivo, consultare [Impegno nella documentazione](#).

D.1.2.1.1 Periodo PI

Metrica	Valore	Giudizio
RDPO	14	Accettabile

La metrica assume un valore accettabile; nonostante ciò si può ancora migliorare la produttività.

D.1.2.1.2 Periodo PCV

Metrica	Valore	Giudizio
RDPO	18	Accettabile

La metrica assume un valore ottimale; la produttività dei documenti è aumentata rispetto al periodo sufficiente grazie alla maggiore esperienza dei membri del team.

D.1.2.2 Qualità del template

Per una descrizione dell'obiettivo, consultare [Qualità del template](#).

D.1.2.2.1 Periodo PI

Metrica	Valore	Giudizio
NCR	0	Ottimale

La metrica assume un valore ottimale. Non sono stati richiesti comandi aggiuntivi nel periodo in esame, il che denota la presenza di un template in grado di soddisfare le necessità.

D.1.2.2.2 Periodo PCV

Metrica	Valore	Giudizio
NCR	0	Ottimale

La metrica assume un valore ottimale. Non sono stati richiesti comandi aggiuntivi nel periodo in esame, il che denota la presenza di un template in grado di soddisfare le necessità.

D.1.2.3 Qualità delle immagini

Per una descrizione dell'obiettivo, consultare [Qualità delle immagini](#).

D.1.2.3.1 Periodo PI

Metrica	Valore	Giudizio
RV	722	Accettabile

La metrica assume un valore accettabile. La qualità dell'immagine risulta essere sufficiente per una chiara visualizzazione. Il team si impegna comunque a migliorare ulteriormente la qualità delle immagini.

D.1.2.3.2 Periodo PCV

Metrica	Valore	Giudizio
RV	722	Accettabile

La metrica assume un valore accettabile. La qualità dell'immagine risulta essere sufficiente per una chiara visualizzazione. Il team si impegna comunque a migliorare ulteriormente la qualità delle immagini.

D.1.2.4 Tracciamento delle modifiche

Per una descrizione dell'obiettivo, consultare [Tracciamento delle modifiche](#).

D.1.2.4.1 Periodo PI

Metrica	Valore	Giudizio
PTM	100%	Ottimale

La metrica assume un valore ottimale. Tutte le modifiche effettuate ai documenti sono state tracciate nell'apposito registro.

D.1.2.4.2 Periodo PCV

Metrica	Valore	Giudizio
PTM	100%	Ottimale

La metrica assume un valore ottimale. Tutte le modifiche effettuate ai documenti sono state tracciate nell'apposito registro.

D.1.3 Processo di sviluppo

Alcuni obiettivi relativi al processo di documentazione non hanno un resoconto per il periodo di Analisi in quanto sono stati stabiliti alla fine di esso.

D.1.3.1 Impegno nella codifica

Per una descrizione dell'obiettivo, consultare [Impegno nella codifica](#).

D.1.3.1.1 Periodo PCV

Metrica	Valore	Giudizio
RCPO	4	Accettabile

La metrica assume un valore accettabile. L'impegno nella codifica è relativamente basso in quanto tutti i programmatori sono inesperti. L'obiettivo per il prossimo periodo è di aumentare il livello di produttività nella codifica.

D.1.3.2 Assegnazione scenari principali

Per una descrizione dell'obiettivo, consultare [Assegnazione scenari principali](#).

D.1.3.2.1 Periodo PI

Metrica	Valore	Giudizio
UCSP	0	Ottimale

La metrica assume un valore ottimale. Tutti gli use case hanno uno scenario principale assegnato.

D.1.3.2.2 Periodo PCV

Metrica	Valore	Giudizio
UCSP	0	Ottimale

La metrica continua ad assumere un valore ottimale. Tutti gli use case hanno uno scenario principale assegnato.

D.1.3.3 Copertura requisiti obbligatori

Per una descrizione dell'obiettivo, consultare [Copertura requisiti obbligatori](#).

D.1.3.3.1 Periodo PI

Metrica	Valore	Giudizio
PROC	100%	Ottimale

La metrica assume un valore ottimale. Tutti i requisiti obbligatori sono stati assegnati alle componenti progettate.

D.1.3.3.2 Periodo PCV

Metrica	Valore	Giudizio
PROC	100%	Ottimale

La metrica continua ad assumere un valore ottimale. Tutti i requisiti obbligatori sono stati assegnati alle componenti progettate.

D.1.3.4 Basso grado di accoppiamento

Per una descrizione dell'obiettivo, consultare [Basso grado di accoppiamento](#).

D.1.3.4.1 Periodo PI

Metrica	Valore	Giudizio
GA	26	Non accettabile

La metrica assume un valore non accettabile. La componente FactorySidebarPkg risulta avere un grado di accoppiamento pari a 26. Il team ha tuttavia ritenuto necessario mantenere tale componente. Il grado di accoppiamento così elevato è dato dal fatto che la famiglia di componenti che la factory deve gestire è numerosa.

D.1.3.4.2 Periodo PCV

Metrica	Valore	Giudizio
GA	26	Non accettabile

La metrica assume un valore non accettabile. Le motivazioni rimangono le stesse del periodo precedente.

D.1.3.5 Alto grado di utilità

Per una descrizione dell'obiettivo, consultare [Alto grado di utilità](#).

D.1.3.5.1 Periodo PI

Metrica	Valore	Giudizio
GU	0	Non accettabile

La metrica assume un valore non accettabile. La componente CallManagerPkg risulta avere un grado di utilità pari a 0. Il team ha tuttavia ritenuto necessario mantenere tale componente. Tale grado di utilità è dovuto al fatto che CallManagerPkg si sottoscrive allo store per mantenere aggiornato il server con i dati contenuti nel primo. Questo tipo di interazione non genera una dipendenza entrante in CallManagerPkg.

D.1.3.5.2 Periodo PCV

Metrica	Valore	Giudizio
GU	0	Non accettabile

La metrica assume un valore non accettabile. Le motivazioni rimangono le stesse del periodo precedente.

D.2 Verifica dei prodotti

D.2.1 Verifica dei documenti

Nel periodo di Analisi, i documenti sono stati analizzati principalmente tramite *walkthrough_G* data la scarsa esperienza dei verificatori. Gli errori più ricorrenti sono stati annotati e serviranno a creare una lista per la successiva attività di verifica, da effettuare utilizzando *inspection_G*.

D.2.1.1 Leggibilità e comprensibilità

Per una descrizione dell'obiettivo, consultare [Leggibilità e comprensibilità](#).

D.2.1.1.1 Periodo An

Documento	Metrica	Valore	Giudizio
<i>Piano di progetto v1.0.0</i>	IG	58	Accettabile
<i>Piano di qualifica v1.0.0</i>	IG	58	Accettabile
<i>Norme di progetto v1.0.0</i>	IG	61	Ottimale
<i>Studio di fattibilità v1.0.0</i>	IG	52	Accettabile
<i>Analisi dei requisiti v1.0.0</i>	IG	45	Accettabile
<i>Glossario v1.0.0</i>	IG	56	Accettabile
<i>VerbaleInterno_1_20161202</i>	IG	79	Ottimale
<i>VerbaleInterno_2_20161220</i>	IG	79	Ottimale
<i>VerbaleInterno_3_20161227</i>	IG	79	Ottimale
<i>VerbaleInterno_4_20170103</i>	IG	79	Ottimale
<i>VerbaleEsterno_1_20161203</i>	IG	73	Ottimale
<i>VerbaleEsterno_2_20161227</i>	IG	68	Ottimale

Tabella 9: Resoconto leggibilità e comprensibilità - periodo An

Tutti i documenti presentano un *indice Gulpese_G* ad un livello almeno accettabile; ciò dovrebbe garantire una lettura non particolarmente difficoltosa da parte di soggetti con almeno licenza superiore. Il documento che assume il valore più basso è l'*Analisi dei requisiti v1.0.0*. Questo è dovuto al fatto che esso è un documento particolarmente tecnico e i contenuti sono esposti sotto forma di tabelle.

D.2.1.1.2 Periodo PI

Documento	Metrica	Valore	Giudizio
<i>Piano di progetto v2.0.0</i>	IG	61	Ottimale
<i>Piano di qualifica v2.0.0</i>	IG	57	Accettabile
<i>Norme di progetto v2.0.0</i>	IG	63	Ottimale
<i>Analisi dei requisiti v2.0.0</i>	IG	48	Accettabile
<i>Specifica tecnica v1.0.0</i>	IG	51	Accettabile
<i>Glossario v2.0.0</i>	IG	56	Accettabile
<i>VerbaleInterno_5_201700206</i>	IG	62	Ottimale
<i>VerbaleInterno_6_20170221</i>	IG	66	Ottimale
<i>VerbaleEsterno_3_20170209</i>	IG	66	Ottimale

Tabella 10: Resoconto leggibilità e comprensibilità - periodo PI

Tutti i documenti presentano un *indice Gulpease_G* ad un livello almeno accettabile; ciò dovrebbe garantire una lettura non particolarmente difficoltosa da parte di soggetti con almeno licenza superiore. Il documento che assume il valore più basso è l'*Analisi dei requisiti v2.0.0*. Questo è dovuto al fatto che esso è un documento particolarmente tecnico e i contenuti sono esposti sotto forma di tabelle.

D.2.1.1.3 Periodo PCV

Documento	Metrica	Valore	Giudizio
<i>Piano di progetto v3.0.0</i>	IG	57	Ottimale
<i>Piano di qualifica v3.0.0</i>	IG	65	Accettabile
<i>Norme di progetto v3.0.0</i>	IG	64	Ottimale
<i>Analisi dei requisiti v3.0.0</i>	IG	47	Accettabile
<i>Specifica tecnica v2.0.0</i>	IG	51	Accettabile
<i>Glossario v2.0.0</i>	IG	56	Accettabile
<i>Definizione di prodotto v1.0.0</i>	IG	57	Accettabile
<i>Manuale utente v1.0.0</i>	IG	58	Accettabile
<i>Manuale manutentore v1.0.0</i>	IG	55	Accettabile
<i>VerbaleInterno_7_20170315</i>	IG	66	Ottimale
<i>VerbaleInterno_8_20170406</i>	IG	62	Ottimale
<i>VerbaleEsterno_4_20170319</i>	IG	61	Ottimale

Tabella 11: Resoconto leggibilità e comprensibilità - periodo PCV

D.2.1.2 Adesione alle norme interne

Per una descrizione dell'obiettivo, consultare [Adesione alle norme interne](#).

D.2.1.2.1 Periodo An

Documento	Metrica	Valore	Giudizio
<i>Piano di progetto v1.0.0</i>	ENNC	0	Ottimale
<i>Piano di qualifica v1.0.0</i>	ENNC	0	Ottimale
<i>Norme di progetto v1.0.0</i>	ENNC	0	Ottimale
<i>Studio di fattibilità v1.0.0</i>	ENNC	0	Ottimale
<i>Analisi dei requisiti v1.0.0</i>	ENNC	0	Ottimale
<i>Glossario v1.0.0</i>	ENNC	0	Ottimale
<i>VerbaleInterno_1_20161202</i>	ENNC	0	Ottimale
<i>VerbaleInterno_2_20161220</i>	ENNC	0	Ottimale
<i>VerbaleInterno_3_20161227</i>	ENNC	0	Ottimale
<i>VerbaleInterno_4_20170103</i>	ENNC	0	Ottimale
<i>VerbaleEsterno_1_20161203</i>	ENNC	0	Ottimale
<i>VerbaleEsterno_2_20161227</i>	ENNC	0	Ottimale

Tabella 12: Resoconto adesione alle norme interne - periodo An

Per tutti i documenti non risultano errori residui che violino le norme interne, pertanto le metriche hanno un valore ottimale.

D.2.1.2.2 Periodo PI

Documento	Metrica	Valore	Giudizio
<i>Piano di progetto v2.0.0</i>	ENNC	0	Ottimale
<i>Piano di qualifica v2.0.0</i>	ENNC	0	Ottimale
<i>Norme di progetto v2.0.0</i>	ENNC	0	Ottimale
<i>Analisi dei requisiti v2.0.0</i>	ENNC	0	Ottimale
<i>Specifica tecnica v1.0.0</i>	ENNC	0	Ottimale
<i>Glossario v2.0.0</i>	ENNC	0	Ottimale
<i>VerbaleInterno_5_201700206</i>	ENNC	0	Ottimale
<i>VerbaleInterno_6_20170221</i>	ENNC	0	Ottimale
<i>VerbaleEsterno_3_20170209</i>	ENNC	0	Ottimale

Tabella 13: Resoconto adesione alle norme interne - periodo PI

Per tutti i documenti non risultano errori residui che violino le norme interne, pertanto le metriche hanno un valore ottimale.

D.2.1.2.3 Periodo PCV

Documento	Metrica	Valore	Giudizio
<i>Piano di progetto v3.0.0</i>	ENNC	0	Ottimale
<i>Piano di qualifica v3.0.0</i>	ENNC	0	Ottimale
<i>Norme di progetto v3.0.0</i>	ENNC	0	Ottimale
<i>Analisi dei requisiti v3.0.0</i>	ENNC	0	Ottimale
<i>Specifica tecnica v2.0.0</i>	ENNC	0	Ottimale
<i>Glossario v2.0.0</i>	ENNC	0	Ottimale
<i>Definizione di prodotto v1.0.0</i>	ENNC	0	Ottimale
<i>Manuale utente v1.0.0</i>	ENNC	0	Ottimale
<i>Manuale manutentore v1.0.0</i>	ENNC	0	Ottimale
<i>VerbaleInterno_7_20170315</i>	ENNC	0	Ottimale
<i>VerbaleInterno_8_20170406</i>	ENNC	0	Ottimale
<i>VerbaleEsterno_4_20170319</i>	ENNC	0	Ottimale

Tabella 14: Resoconto adesione alle norme interne - periodo PCV

Per tutti i documenti non risultano errori residui che violino le norme interne, pertanto le metriche hanno un valore ottimale.

D.2.1.3 Correttezza ortografica

Per una descrizione dell'obiettivo, consultare [Correttezza ortografica](#).

D.2.1.3.1 Periodo An

Documento	Metrica	Valore	Giudizio
<i>Piano di progetto v1.0.0</i>	EONC	0	Ottimale
<i>Piano di qualifica v1.0.0</i>	EONC	0	Ottimale
<i>Norme di progetto v1.0.0</i>	EONC	0	Ottimale
<i>Studio di fattibilità v1.0.0</i>	EONC	0	Ottimale
<i>Analisi dei requisiti v1.0.0</i>	EONC	0	Ottimale
<i>Glossario v1.0.0</i>	EONC	0	Ottimale
<i>VerbaleInterno_1_20161202</i>	EONC	0	Ottimale
<i>VerbaleInterno_2_20161220</i>	EONC	0	Ottimale
<i>VerbaleInterno_3_20161227</i>	EONC	0	Ottimale
<i>VerbaleInterno_4_20170103</i>	EONC	0	Ottimale
<i>VerbaleEsterno_1_20161203</i>	EONC	0	Ottimale
<i>VerbaleEsterno_2_20161227</i>	EONC	0	Ottimale

Tabella 15: Resoconto correttezza ortografica - periodo An

Dopo l'analisi automatica dei correttori ortografici e quella mediante walkthrough da parte dei *Verificatori* non sono stati rilevati ulteriori errori che violano le norme interne, pertanto le metriche assumono un valore ottimale.

D.2.1.3.2 Periodo PI

Documento	Metrica	Valore	Giudizio
<i>Piano di progetto v2.0.0</i>	EONC	0	Ottimale
<i>Piano di qualifica v2.0.0</i>	EONC	0	Ottimale
<i>Norme di progetto v2.0.0</i>	EONC	0	Ottimale
<i>Analisi dei requisiti v2.0.0</i>	EONC	0	Ottimale
<i>Specifica tecnica v1.0.0</i>	EONC	0	Ottimale
<i>Glossario v2.0.0</i>	EONC	0	Ottimale
<i>VerbaleInterno_5_201700206</i>	EONC	0	Ottimale
<i>VerbaleInterno_6_20170221</i>	EONC	0	Ottimale
<i>VerbaleEsterno_3_20170209</i>	EONC	0	Ottimale

Tabella 16: Resoconto correttezza ortografica - periodo PI

Dopo l'analisi automatica dei correttori ortografici e quella mediante walkthrough da parte dei *Verificatori* non sono stati rilevati ulteriori errori che violano le norme interne, pertanto le metriche assumono un valore ottimale.

D.2.1.3.3 Periodo PCV

Documento	Metrica	Valore	Giudizio
<i>Piano di progetto v3.0.0</i>	EONC	0	Ottimale
<i>Piano di qualifica v3.0.0</i>	EONC	0	Ottimale
<i>Norme di progetto v3.0.0</i>	EONC	0	Ottimale
<i>Analisi dei requisiti v3.0.0</i>	EONC	0	Ottimale
<i>Specifica tecnica v2.0.0</i>	EONC	0	Ottimale
<i>Glossario v2.0.0</i>	EONC	0	Ottimale
<i>Definizione di prodotto v1.0.0</i>	EONC	0	Ottimale
<i>Manuale utente v1.0.0</i>	EONC	0	Ottimale
<i>Manuale manutentore v1.0.0</i>	EONC	0	Ottimale
<i>VerbaleInterno_7_20170315</i>	EONC	0	Ottimale
<i>VerbaleInterno_8_20170406</i>	EONC	0	Ottimale
<i>VerbaleEsterno_4_20170319</i>	EONC	0	Ottimale

Tabella 17: Resoconto correttezza ortografica - periodo PCV

Dopo l'analisi automatica dei correttori ortografici e quella mediante walkthrough da parte dei *Verificatori* non sono stati rilevati ulteriori errori che violano le norme interne, pertanto le metriche assumono un valore ottimale.

D.2.1.4 Correttezza concettuale

Per una descrizione dell'obiettivo, consultare [Correttezza concettuale](#).

D.2.1.4.1 Periodo An

Documento	Metrica	Valore	Giudizio
<i>Piano di progetto v1.0.0</i>	ECNC	0	Ottimale
<i>Piano di qualifica v1.0.0</i>	ECNC	0	Ottimale
<i>Norme di progetto v1.0.0</i>	ECNC	0	Ottimale
<i>Studio di fattibilità v1.0.0</i>	ECNC	0	Ottimale
<i>Analisi dei requisiti v1.0.0</i>	ECNC	0	Ottimale
<i>Glossario v1.0.0</i>	ECNC	0	Ottimale
<i>VerbaleInterno_1_20161202</i>	ECNC	0	Ottimale
<i>VerbaleInterno_2_20161220</i>	ECNC	0	Ottimale
<i>VerbaleInterno_3_20161227</i>	ECNC	0	Ottimale
<i>VerbaleInterno_4_20170103</i>	ECNC	0	Ottimale
<i>VerbaleEsterno_1_20161203</i>	ECNC	0	Ottimale
<i>VerbaleEsterno_2_20161227</i>	ECNC	0	Ottimale

Tabella 18: Resoconto correttezza concettuale - periodo An

Per tutti i documenti non sono stati rilevati errori concettuali non corretti, pertanto le metriche assumono un valore ottimale.

D.2.1.4.2 Periodo PI

Documento	Metrica	Valore	Giudizio
<i>Piano di progetto v2.0.0</i>	ECNC	0	Ottimale
<i>Piano di qualifica v2.0.0</i>	ECNC	0	Ottimale
<i>Norme di progetto v2.0.0</i>	ECNC	0	Ottimale
<i>Analisi dei requisiti v2.0.0</i>	ECNC	0	Ottimale
<i>Specifica tecnica v1.0.0</i>	ECNC	0	Ottimale
<i>Glossario v2.0.0</i>	ECNC	0	Ottimale
<i>VerbaleInterno_5_201700206</i>	ECNC	0	Ottimale
<i>VerbaleInterno_6_20170221</i>	ECNC	0	Ottimale
<i>VerbaleEsterno_3_20170209</i>	ECNC	0	Ottimale

Tabella 19: Resoconto correttezza concettuale - periodo PI

D.2.1.4.3 Periodo PCV

Per tutti i documenti non sono stati rilevati errori concettuali non corretti, pertanto le metriche assumono un valore ottimale.

Documento	Metrica	Valore	Giudizio
<i>Piano di progetto v3.0.0</i>	ECNC	0	Ottimale
<i>Piano di qualifica v3.0.0</i>	ECNC	0	Ottimale
<i>Norme di progetto v3.0.0</i>	ECNC	0	Ottimale
<i>Analisi dei requisiti v3.0.0</i>	ECNC	0	Ottimale
<i>Specifica tecnica v2.0.0</i>	ECNC	0	Ottimale
<i>Glossario v2.0.0</i>	ECNC	0	Ottimale
<i>Definizione di prodotto v1.0.0</i>	ECNC	0	Ottimale
<i>Manuale utente v1.0.0</i>	ECNC	0	Ottimale
<i>Manuale manutentore v1.0.0</i>	ECNC	0	Ottimale
<i>VerbaleInterno_7_20170315</i>	ECNC	0	Ottimale
<i>VerbaleInterno_8_20170406</i>	ECNC	0	Ottimale
<i>VerbaleEsterno_4_20170319</i>	ECNC	0	Ottimale

Tabella 20: Resoconto correttezza concettuale - periodo PCV

Per tutti i documenti non sono stati rilevati errori concettuali non corretti, pertanto le metriche assumono un valore ottimale.

D.2.1.5 Basso livello di annidamento dell'indice

Per una descrizione dell'obiettivo, consultare [Basso livello di annidamento dell'indice](#).

D.2.1.5.1 Periodo PI

Documento	Metrica	Valore	Giudizio
<i>Piano di progetto v2.0.0</i>	LA	4	Accettabile
<i>Piano di qualifica v2.0.0</i>	LA	4	Accettabile
<i>Norme di progetto v2.0.0</i>	LA	5	Accettabile
<i>Analisi dei requisiti v2.0.0</i>	LA	2	Ottimale
<i>Specifica tecnica v1.0.0</i>	LA	5	Accettabile
<i>Glossario v2.0.0</i>	LA	1	Ottimale
<i>VerbaleInterno_5_201700206</i>	LA	1	Ottimale
<i>VerbaleInterno_6_20170221</i>	LA	1	Ottimale
<i>VerbaleEsterno_3_20170209</i>	LA	1	Ottimale

Tabella 21: Resoconto basso livello di annidamento dell'indice - periodo PI

Per tutti i documenti il livello di annidamento dell'indice risulta accettabile. In molti casi è stata preferita una struttura tabellare come alternativa all'eccessivo annidamento.

D.2.1.5.2 Periodo PCV

Documento	Metrica	Valore	Giudizio
<i>Piano di progetto v3.0.0</i>	LA	5	Accettabile
<i>Piano di qualifica v3.0.0</i>	LA	4	Accettabile
<i>Norme di progetto v3.0.0</i>	LA	5	Accettabile
<i>Analisi dei requisiti v3.0.0</i>	LA	2	Ottimale
<i>Specifica tecnica v2.0.0</i>	LA	5	Accettabile
<i>Glossario v2.0.0</i>	LA	1	Ottimale
<i>Definizione di prodotto v1.0.0</i>	LA	5	Ottimale
<i>Manuale utente v1.0.0</i>	LA	2	Ottimale
<i>Manuale manutentore v1.0.0</i>	LA	4	Accettabile
<i>VerbaleInterno_7_20170315</i>	LA	1	Ottimale
<i>VerbaleInterno_8_20170406</i>	LA	1	Ottimale
<i>VerbaleEsterno_4_20170319</i>	LA	1	Ottimale

Tabella 22: Resoconto basso livello di annidamento dell'indice - periodo PCV

Per tutti i documenti il livello di annidamento dell'indice risulta accettabile. In molti casi è stata preferita una struttura tabellare come alternativa all'eccessivo annidamento.

D.2.2 Verifica del software

Gli obiettivi riguardanti il software sono stati verificati solamente dal periodo Progettazione di dettaglio, Codifica e Validazione in poi. Nei periodi precedenti non esisteva ancora codice, quindi la verifica di tali obiettivi non è stata possibile.

D.2.2.1 Implementazione delle funzionalità obbligatorie

Per una descrizione dell'obiettivo, consultare [Implementazione delle funzionalità obbligatorie](#).

D.2.2.1.1 Periodo PCV

Metrica	Valore	Giudizio
IFO	0	Non accettabile

Non sono state ancora implementate funzionalità obbligatorie: dato che i test di sistema non sono stati ancora implementati e superati, non è possibile confermare l'effettiva implementazione delle funzionalità. Questo rivela un grave ritardo nei lavori. Tuttavia, i test di unità sono già stati pianificati e superati parzialmente.

L'obiettivo per il prossimo periodo è implementare completamente le funzionalità obbligatorie.

D.2.2.2 Implementazione delle funzionalità desiderabili

Per una descrizione dell'obiettivo, consultare [Implementazione delle funzionalità desiderabili](#).

D.2.2.2.1 Periodo PCV

Metrica	Valore	Giudizio
IFD	0	Non accettabile

Non sono state ancora implementate funzionalità desiderabili: dato che i test di sistema non sono stati ancora implementati e superati, non è possibile confermare l'effettiva implementazione delle funzionalità. Questo rivela un grave ritardo nei lavori. Tuttavia, i test di unità sono già stati pianificati e superati parzialmente.

L'obiettivo per il prossimo periodo è implementare le funzionalità desiderabili per raggiungere l'obiettivo associato.

D.2.2.3 Basso numero di statement per metodo

Per una descrizione dell'obiettivo, consultare [Basso numero di statement per metodo](#).

D.2.2.3.1 Periodo PCV

Metrica	Valore	Giudizio
NSM	16	Ottimale

Il metodo con massimo numero di statement è il constructor della classe Asset. Tale valore è all'interno della soglia di ottimalità.

L'obiettivo per il prossimo periodo è cercare di non codificare metodi con numero di statement maggiore.

D.2.2.4 Basso numero di parametri per metodo

Per una descrizione dell'obiettivo, consultare [Basso numero di parametri per metodo](#).

D.2.2.4.1 Periodo PCV

Metrica	Valore	Giudizio
NPM	3	Ottimale

Il metodo con più alto numero di parametri è handleChange della classe DeGeOPView. Il valore è ottimale

L'obiettivo per il prossimo periodo è cercare di non codificare metodi con un numero di parametri maggiore.

D.2.2.5 Basso numero di campi dati per classe

Per una descrizione dell'obiettivo, consultare [Basso numero di campi dati per classe](#).

D.2.2.5.1 Periodo PCV

Metrica	Valore	Giudizio
NCDPC	19	Non accettabile

La classe con il massimo numero di campi dati è mapWrapper. Non è stato ritenuto sensato dividere le funzionalità offerte da questa classe in quanto esse sono fortemente coese. È stato preferito mantenere basso il numero di dipendenze per mantenere un basso livello di accoppiamento.

D.2.2.6 Bassa complessità ciclomatica

Per una descrizione dell'obiettivo, consultare [Bassa complessità ciclomatica](#).

D.2.2.6.1 Periodo PCV

Metrica	Valore	Giudizio
NC	5	Ottimale

Il metodo con il massimo numero di complessità ciclomatica è `componentDidUpdate` della classe `DeGeOPView`. Il valore è ottimale.

L'obiettivo per il prossimo periodo è cercare di non codificare metodi con complessità ciclomatica maggiore in modo da poter ottenere un branch coverage elevato più facilmente durante l'implementazione dei test.

D.2.2.7 Assenza di variabili dichiarate e non utilizzate

Per una descrizione dell'obiettivo, consultare [Assenza di variabili dichiarate e non utilizzate](#).

D.2.2.7.1 Periodo PCV

Metrica	Valore	Giudizio
NVNU	0	Ottimale

Non risultano esserci variabili dichiarate e non utilizzate. La metrica assume quindi un valore ottimale.

D.2.2.8 Documentazione del codice

Per una descrizione dell'obiettivo, consultare [Documentazione del codice](#).

D.2.2.8.1 Periodo PCV

Metrica	Valore	Giudizio
RCC	76%	Ottimale

La metrica assume un valore ottimale. Il codice risulta essere documentato in maniera ottimale.

D.2.2.9 Superamento dei test pianificati

Per una descrizione dell'obiettivo, consultare [Superamento dei test pianificati](#).

D.2.2.9.1 Periodo PCV

Metrica	Valore	Giudizio
STP	13.93%	Non accettabile

La metrica assume un valore fortemente negativo. Ciò denota un grave ritardo nei lavori. L'obiettivo per il prossimo periodo è implementare e superare i test mancanti.

D.2.2.10 Robustezza

Per una descrizione dell'obiettivo, consultare [Robustezza](#).

D.2.2.10.1 Periodo PCV

Metrica	Valore	Giudizio
BA	100%	Ottimale

I test effettuati non hanno rilevato situazioni anomale che comportino l'interruzione del funzionamento del prodotto. La metrica assume quindi un valore ottimale

D.2.2.11 Correzione delle situazioni di fallimento

Per una descrizione dell'obiettivo, consultare [Correzione delle situazioni di fallimento](#).

D.2.2.11.1 Periodo PCV

Metrica	Valore	Giudizio
FA	100%	Ottimale

Tutte le situazioni di fallimento rilevate dai test sono state corrette, pertanto la metrica assume un valore ottimale.

L'obiettivo è continuare a risolvere tutte le situazioni di fallimento rilevate durante l'esecuzione dei test.

D.2.2.12 Copertura degli statement

Per una descrizione dell'obiettivo, consultare [Copertura degli statement](#).

D.2.2.12.1 Periodo PCV

Metrica	Valore	Giudizio
SC	100%	Ottimale

I test effettuati eseguono almeno una volta ogni statement del codice che testano.
La metrica assume quindi un valore ottimale.

D.2.2.13 Copertura dei branch

Per una descrizione dell'obiettivo, consultare [Copertura dei branch](#).

D.2.2.13.1 Periodo PCV

Metrica	Valore	Giudizio
BC	100%	Ottimale

I test effettuati eseguono almeno una volta ogni branch del codice che testano.
La metrica assume quindi un valore ottimale.