



Piano di Qualifica

Gruppo MINT – Progetto MaaS

Informazioni sul documento

Versione	4.0.0
Redazione	Fabiano Tavallini, Michael Ogbuachi, Navid Taha
Verifica	Tommaso Zagni, Thomas Fuser, Michael Ogbuachi
Approvazione	Enrico Canova
Uso	Esterno
Distribuzione	Prof. Tullio Vardanega Prof. Riccardo Cardin Gruppo MINT

Descrizione

Questo documento descrive le operazioni di verifica e validazione seguite dal gruppo MINT durante la realizzazione del progetto MaaS.



Registro delle modifiche

Versione	Data	Collaboratori	Descrizione
4.0.0	28-07-2016	Enrico Canova (Responsabile)	Approvazione del documento.
3.1.0	28-07-2016	Michael Ogbuachi (Verificatore)	Verifica del documento.
3.0.6	27-07-2016	Fabiano Tavallini, Tommaso Zagni (Progettisti)	Appendice C: incrementato il resoconto delle attività di verifica.
3.0.5	27-08-2016	Fabiano Tavallini, Tommaso Zagni (Progettisti)	Appendice B: specificato in modo migliore lo stato di avanzamento dei test.
3.0.4	26-08-2016	Fabiano Tavallini (Progettista)	Sezione 2: definiti quantitativamente gli obiettivi di qualità, fine ristrutturazione documento. Fine rivisitazione test di Unità
3.0.3	26-08-2016	Fabiano Tavallini (Progettista)	Sezione 3: Aggiunte le metriche delle funzionalità obbligatorie e desiderabili.
3.0.2	25-08-2016	Fabiano Tavallini (Progettista)	Aggiunta sezione 3 con la strategia di qualifica in dettaglio e relative metriche. Inizio rivisitazione test di Unità.
3.0.1	25-08-2016	Fabiano Tavallini (Progettista)	Inizio ristrutturazione documento in base alle segnalazioni in sede di Revisione di Progettazione.
3.0.0	18-05-2016	Enrico Canova (Responsabile)	Approvazione del documento.
2.2.0	18-05-2016	Tommaso Zagni, Michael Ogbuachi (Verificatori)	Verifica del documento.
2.1.0	17-05-2016	Tommaso Zagni, Thomas Fuser (Verificatori)	Verifica del documento.
2.0.5	16-05-2016	Navid Taha, Michael Ogbuachi (Progettisti)	Sezione A5: stesura Test di Unità.
2.0.4	15-05-2016	Navid Taha, Fabiano Tavallini (Progettisti)	Sezione A4: stesura Test di Integrazione.
2.0.3	14-05-2016	Fabiano Tavallini (Progettista)	Sezione A3: stesura Test di Sistema.
2.0.2	13-05-2016	Michael Ogbuachi, Navid Taha (Progettisti)	Sezione A2: stesura Test di Validazione.
2.0.1	13-05-2016	Fabiano Tavallini (Progettista)	Sezione A1: stesura Livelli di Testing.



2.0.0	22-04-2016	Thomas Fuser, Fabiano Tavallini (Responsabili)	Approvazione del documento.
1.2.0	22-04-2016	Enrico Canova, Fabiano Tavallini, Michael Ogbuachi (Verificatori)	Verifica del documento.
1.1.0	21-04-2016	Enrico Canova, Navid Taha (Verificatori)	Verifica del documento.
1.0.3	20-04-2016	Thomas Fuser (Amministratore)	Appendice D: stesura avanzamento PDCA.
1.0.2	19-04-2016	Tommaso Zagni, Navid Taha (Amministratori)	Sezione 2 spostata in Appendice C: Qualità.
1.0.1	18-04-2016	Tommaso Zagni (Amministratore)	Ridefinizione scheletro del documento.
1.0.0	27-03-2016	Michael Ogbuachi (Responsabile)	Approvazione del documento.
0.2.0	27-03-2016	Tommaso Zagni, Fabiano Tavallini (Verificatori)	Verifica finale del documento.
0.1.0	26-03-2016	Tommaso Zagni (Verificatore)	Verifica del documento fino a 3.7.
0.0.7	25-03-2016	Michael Ogbuachi, Fabiano Tavallini (Amministratori)	Appendice A: Stesura Resoconto delle attività di verifica.
0.0.6	24-03-2016	Fabiano Tavallini, Thomas Fuser (Amministratori)	Sezione 4: Stesura Gestione Amministrativa della Revisione.
0.0.5	22-03-2016	Michael Ogbuachi, Navid Taha (Amministratori)	Sezione 3.7.3: fine stesura. Completamento stesura Misure e Metriche.
0.0.4	21-03-2016	Michael Ogbuachi (Amministratore)	Inizio stesura Misure e Metriche. Completate sezioni 3.7.1 e 3.7.2.
0.0.3	20-03-2016	Enrico Canova, Fabiano Tavallini (Amministratori)	Inizio stesura sezione 3. Completate le sezioni da 3.1 a 3.6.
0.0.2	19-03-2016	Fabiano Tavallini, Navid Taha (Amministratori)	Sezione 2: Stesura definizione obiettivi di qualità.
0.0.1	18-03-2016	Fabiano Tavallini (Amministratore)	Impostazione scheletro del documento. Sezione 1: Stesura Introduzione.



Indice

1	Introduzione	6
1.1	Scopo del documento	6
1.2	Scopo del prodotto	6
1.3	Glossario	6
1.4	Riferimenti	6
1.4.1	Normativi	6
1.4.2	Informativi	7
2	Visione generale della strategia di gestione della qualità	8
2.1	Obiettivi di qualità	8
2.1.1	Qualità di processo	8
2.1.1.1	Miglioramento costante	9
2.1.1.2	Rispetto dei tempi	9
2.1.1.3	Rispetto del budget	9
2.1.1.4	Produttività	10
2.1.1.5	Impegno	10
2.1.2	Qualità di prodotto	10
2.1.2.1	Qualità dei documenti	11
2.1.2.1.1	Leggibilità e comprensibilità	11
2.1.2.1.2	Correttezza ortografica	11
2.1.2.1.3	Correttezza concettuale	12
2.1.2.2	Qualità del software	12
2.1.2.2.4	Funzionalità obbligatorie	12
2.1.2.2.5	Funzionalità desiderabili	13
2.1.2.2.6	Manutenibilità e comprensibilità del codice	13
2.1.2.2.7	Affidabilità del software	14
2.1.2.2.8	Copertura dei test	15
2.2	Organizzazione temporale	15
3	Strategia di gestione della qualità nel dettaglio	16
3.1	Responsabilità	16
3.2	Risorse	16
3.2.1	Umane	16
3.2.2	Tecnologiche	16
3.3	Misure e metriche	17
3.3.1	Metriche per i processi	17
3.3.1.1	Modello SPICE	17
3.3.1.2	Schedule Variance	17
3.3.1.3	Budget Variance	18
3.3.1.4	Produttività	18
3.3.1.5	Impegno	19
3.3.2	Metriche per il prodotto	20
3.3.2.1	Metriche per i documenti	20
3.3.2.1.1	Gulpease	20
3.3.2.1.2	Errori ortografici rinvenuti e non corretti	21
3.3.2.1.3	Errori concettuali rinvenuti e non corretti	21



3.3.2.2	Metriche per il software	21
3.3.2.2.4	Copertura requisiti obbligatori	22
3.3.2.2.5	Copertura requisiti desiderabili	22
3.3.2.2.6	Numero di metodi - NOM	23
3.3.2.2.7	Numero di parametri per metodo	23
3.3.2.2.8	Variabili non utilizzate e non definite	23
3.3.2.2.9	Complessità ciclomatica	24
3.3.2.2.10	Metriche di Halstead	24
3.3.2.2.11	Maintainability index	26
3.3.2.2.12	Statement Coverage	26
3.3.2.2.13	Branch Coverage	26
3.3.2.2.14	Copertura dei test	26
4	Gestione amministrativa della revisione	28
4.1	Comunicazione delle anomalie	28
4.2	Procedure di controllo per la qualità di processo	28
	Appendici	29
A	Qualità	30
A.1	Qualità di processo	30
A.1.1	Standard ISO/IEC 15504	30
A.1.2	Ciclo di Deming	31
A.2	Qualità di prodotto	32
A.2.1	Standard ISO/IEC 9126	32
B	Pianificazione dei test	34
B.1	Livelli di testing	34
B.2	Test di validazione	35
B.3	Test di sistema	42
B.4	Test di integrazione	56
B.5	Test di unità	59
C	Resoconto delle attività di verifica	77
C.1	Riassunto delle attività di verifica	77
C.1.1	Revisione dei Requisiti	77
C.1.2	Revisione di Progettazione	77
C.1.3	Revisione di Qualifica	77
C.2	Dettaglio delle verifiche tramite analisi	77
C.2.1	Analisi Preliminare	77
C.2.1.1	Documenti	77
C.2.2	Progettazione Architettuale e di Dettaglio	78
C.2.2.1	Documenti	78
C.2.3	Codifica e Test	78
C.2.3.1	Documenti	78
C.2.3.2	Software	79
C.3	Dettaglio dell'esito delle revisioni	79
C.3.1	Revisione dei Requisiti	80
C.3.2	Revisione di Progettazione	80



Elenco delle tabelle

2	Tracciamento Metriche software - Caratteristiche di qualità	22
A.1	Test di Validazione	41
A.2	Test di Sistema	55
A.3	Descrizione test d'Integrazione	58
A.4	Test di Unità	76
A.5	Esiti verifica documenti in fase di Analisi	78
A.6	Esiti verifica documenti, Progettazione Architetturale	78
A.7	Esiti verifica documenti, Progettazione di Dettaglio e Codifica	78
A.8	Esiti metriche software, Codifica e Test	79
A.9	Esiti metriche software, Codifica e Test	79

Elenco delle figure

1	Il ciclo di miglioramento dei processi	29
2	Continuous quality improvement with PDCA	31
3	V-Model per il testing software	35
4	Sequenza d'integrazione delle componenti	56



1 Introduzione

L'obiettivo primario è la *qualità_G* del prodotto e dei suoi processi, ottenibile mediante una serie di controlli stabiliti inizialmente. L'assenza di queste verifiche, combinata ad un *team di sviluppo_G* con più componenti senza particolari accortezze e competenze, porta al progressivo deterioramento del materiale prodotto, sia esso codice sorgente o documentazione.

Bisogna pertanto prevenire l'inserimento, all'interno del *repository_G*, di materiale non congruo alle *Norme di Progetto v4.0.0* poiché si avvierebbe un graduale degrado della sua qualità.

1.1 Scopo del documento

Il Piano di Qualifica illustra la strategia di *verifica_G* e *validazione_G* che il gruppo MINT ha deciso di adottare per lo svolgimento del progetto. È necessario dimensionare la qualità dei prodotti e dei processi, operazione che non rientra nei normali ruoli di progettazione, bensì rappresenta una *funzione aziendale_G*. Secondo le strategie riportate in questo documento il *Committente_G* sarà in grado di valutare oggettivamente quanto è stato prodotto e disporrà di una solida base di verifica.

1.2 Scopo del prodotto

L'obiettivo preposto dal Committente per il progetto *MaaS_G* (*MongoDB_G* as an admin Service), è quello di costruire un servizio web che incorpora *MaaP_G* (MongoDB as an admin Platform) e lo rende direttamente disponibile, attraverso il Web, per molteplici società. L'architettura di MaaS servirà gruppi di utenti rendendogli disponibile una condivisione dedicata di una sua istanza, includendo tutti i dati e le funzionalità in loro possesso.

1.3 Glossario

Ogni occorrenza di termini tecnici, di dominio e gli acronimi sono evidenziati in corsivo e marcati con la lettera G in pedice. I relativi significati sono riportati nel documento *Glossario v4.0.0*.

1.4 Riferimenti

Vengono elencati qui di seguito i riferimenti sui quali si basano il presente documento stesso e l'organizzazione delle attività di verifica e validazione.

1.4.1 Normativi

- **Norme di progetto:** *Norme di Progetto v4.0.0*;
- **Capitolato d'appalto C4:** RedBabel, MaaS <http://www.math.unipd.it/~tullio/IS-1/2015/Progetto/C4.pdf>;
- **Standard ISO/IEC 15504:** http://en.wikipedia.org/wiki/ISO/IEC_15504;



- **Standard ISO/IEC 9126:** http://en.wikipedia.org/wiki/ISO/IEC_9126;
- **Standard IEEE 610.12-90:** https://cow.ceng.metu.edu.tr/Courses/download_courseFile.php?id=2677.

1.4.2 Informativi

- **Piano di Progetto:** *Piano di Progetto v4.0.0*;
- **SWEBOK v3:** capitolo 10;
- **Slides del corso di Ingegneria del Software mod. A: Qualità del software** <http://www.math.unipd.it/~tullio/IS-1/2015/Dispense/L08.pdf>;
- **Slides del corso di Ingegneria del Software mod. A: Qualità del processo_G** <http://www.math.unipd.it/~tullio/IS-1/2015/Dispense/L09.pdf>;
- **Software Engineering 9th - I. Sommerville (Pearson, 2011):** capitoli 24 e 26;
- **Metriche del software G - Ercole F. Colonese:** http://www.colonese.it/00-Manuali_Pubblicatii/08-Metriche%20del%20software_v1.0.pdf;
- **Ciclo di Deming:** https://it.wikipedia.org/wiki/Ciclo_di_Deming;
- **Indice Gulpease:** https://it.wikipedia.org/wiki/Indice_Gulpease.



2 Visione generale della strategia di gestione della qualità

La strategia adottata ha lo scopo di cercare efficienza nel perseguire determinati obiettivi di qualità per ottenere un riscontro affidabile e numericamente trattabile che consenta di assicurare il grado di qualità predeterminato. Con tale strategia si cerca inoltre di automatizzare parte delle attività di verifica; tale scelta necessita l'uso di tool adeguatamente configurati. L'aspettativa è la riduzione del lavoro manuale permettendo così una validazione semplificata.

2.1 Obiettivi di qualità

In questa sezione vengono definiti quantitativamente gli obiettivi di qualità che il gruppo *MINT* si impegna a perseguire durante lo svolgimento dell'intero progetto. Per determinare il raggiungimento degli obiettivi si fa uso di standard, modelli e metriche, per ognuno dei quali fissiamo dei valori per identificare, nelle diverse scale di misurazione, le soglie di accettazione ed ottimalità. Si fa notare che le metriche citate di seguito sono trattate in maggiore dettaglio nella sezione 3.3.

2.1.1 Qualità di processo

Assicurare la qualità dei processi è un fattore determinante per garantire la qualità del prodotto. In questo modo viene ottimizzato l'uso delle risorse, i costi vengono contenuti e la stima dei rischi risulta più semplice.

Il gruppo desidera che i processi seguano le seguenti caratteristiche:

- Performance continuamente migliorabili;
 - performance costantemente misurabili;
 - persecuzione continua di obiettivi quantitativi di miglioramento;
- Rispetto dei tempi indicati nel *Piano di Progetto v4.0.0*;
- Costi effettivi in linea con il *Piano di Progetto v4.0.0*.

Le linee guida per il miglioramento continuo della qualità dei processi, seguono il modello *PDCA*_G¹. In particolare esso permette un controllo delle attività di processo ripetibili e misurabili garantendo la manutenzione dei processi stessi incrementandone la qualità. Vengono così descritte come devono essere attuate le procedure di controllo:

- Pianificazione dettagliata;
- Monitoraggio delle attività pianificate;
- Definizione delle risorse necessarie al conseguimento degli obiettivi;
- Utilizzo di metriche per verificare il miglioramento della qualità dei processi.

All'interno del *Piano di Progetto v4.0.0* è descritta in dettaglio la pianificazione di queste attività.

Per quantificare la qualità dei processi, si adottano le metriche indicate in seguito.

¹Si veda appendice A.1.2 per approfondimenti



2.1.1.1 Miglioramento costante

Per quantificare la capacità dei processi di misurare le proprie performance e di porsi obiettivi quantitativi di miglioramento, si è deciso di adottare il modello $SPICE_g$ ², definito nello standard ISO/IEC 15504, ai fini di una valutazione oggettiva dei processi. In particolare si vuole raggiungere almeno il livello 2 previsto in tale modello, mentre il livello ottimale che speriamo di raggiungere è il 4.

Riassumendo:

- **Modello utilizzato:** SPICE;
- **Soglia di accettabilità:** livello 2;
- **Soglia di ottimalità:** livello 4.

Per approfondire la scelta delle soglie di accettazione e ottimalità si consulti la metrica alla sezione 3.3.1.1.

2.1.1.2 Rispetto dei tempi

Per capire se le attività di un processo sono in ritardo rispetto a quanto pianificato all'interno del *Piano di Progetto v4.0.0* viene utilizzata la Schedule Variance. Si desidera che il ritardo accumulato sia minore del 10% rispetto al totale pianificato, sarebbe invece ottimale essere in linea, o addirittura in anticipo, con quanto pianificato.

Riassumendo:

- **Metrica utilizzata:** Schedule Variance;
- **Soglia di accettabilità:** ritardo massimo del 10% rispetto a quanto pianificato;
- **Soglia di ottimalità:** ritardo minore o uguale a 0% rispetto a quanto pianificato.

Per una descrizione dettagliata della metrica utilizzata si faccia riferimento alla sezione 3.3.1.2.

2.1.1.3 Rispetto del budget

Per capire se i costi di un processo rientrano nel budget previsto dal *Piano di Progetto v4.0.0* viene utilizzata la Budget Variance. Si desidera avere costi non superiori al budget a disposizione per più del 10%, sarebbe invece ottimale essere in linea con il preventivo o che si spendesse di meno.

Riassumendo:

- **Metrica utilizzata:** Budget Variance;
- **Soglia di accettabilità:** costi maggiori al massimo del 10% rispetto al preventivo;
- **Soglia di ottimalità:** valore minore o uguale a 0%.

Per una descrizione dettagliata della metrica utilizzata si faccia riferimento alla sezione 3.3.1.3.

²Si veda appendice A.1 per approfondimenti



2.1.1.4 Produttività

Per conoscere le performance della produttività dei diversi componenti del gruppo si misurano la produttività di documentazione, di codifica e di test.

Riassumiamo le diverse metriche ed i rispettivi range desiderati:

- **Metrica utilizzata:** Produttività di documentazione;
- **Range di accettabilità:** [70 - 100];
- **Soglia di ottimalità:** valore maggiore o uguale a 100.

- **Metrica utilizzata:** Produttività di codifica;
- **Range di accettabilità:** [5 - 20];
- **Range di ottimalità:** [2 - 10].

- **Metrica utilizzata:** Produttività di test;
- **Range di accettabilità:** [5 - 10];
- **Soglia di ottimalità:** valore maggiore o uguale a 10.

Per una descrizione dettagliata di tali metriche si faccia riferimento alla sezione 3.3.1.4.

2.1.1.5 Impegno

Per valutare oggettivamente l'impegno dei diversi componenti del gruppo si mette in rapporto il tempo produttivo impiegato e la produttività vista in precedenza. Si desidera che l'impegno del gruppo rientri in un valore descritto dalla metrica tra 0,5 e 0,6, sarebbe invece ottimale avere un impegno superiore a 0,6.

Riassumendo:

- **Metrica utilizzata:** Impegno;
- **Range di accettabilità:** [0,5 - 0,6];
- **Soglia di ottimalità:** valore maggiore di 0,6.

Per una descrizione dettagliata della metrica utilizzata si faccia riferimento alla sezione 3.3.1.5.

2.1.2 Qualità di prodotto

Oltre alla qualità di processo, sono necessari degli obiettivi rivolti direttamente alla qualità del prodotto per massimizzare l'efficacia. Il progetto prevede la produzione di due tipologie di prodotto: i documenti e il software. Grazie ai seguenti processi verrà garantito il controllo di qualità del prodotto:

- **Software Quality Assurance (SQA_e):** l'insieme delle attività realizzate al fine di garantire il raggiungimento degli obiettivi di qualità; è importante che tale processo sia preventivo e non correttivo;



- **Verifica:** assicura che l'esecuzione delle attività dei processi svolti non introduca errori nel prodotto. Durante l'intera durata del progetto verranno svolte attività di verifica sugli output dei processi, accertando che esso sia corretto, completo e rispetti regole, convenzioni e procedure;
- **Validazione:** la conferma oggettiva che assicura che i prodotti finali soddisfino i requisiti e le aspettative attese.

Nelle prossime sezioni vengono definiti gli obiettivi che il gruppo *MINT* intende raggiungere, suddivisi per tipologia di prodotto. Per ogni obiettivo vengono specificati i criteri con i quali si effettuano le misurazioni sulla qualità e per ogni metrica scelta vengono inoltre dichiarate le soglie di accettazione e di ottimalità.

2.1.2.1 Qualità dei documenti

Il gruppo ha selezionato delle metriche per garantire la qualità dei documenti prodotti, in particolare essi devono seguire le seguenti caratteristiche:

- devono essere comprensibili alla lettura da parte di individui dotati di una licenza superiore;
- devono essere corretti a livello ortografico;
- devono essere corretti dal punto di vista concettuale.

Vengono definiti ora in modo quantitativo gli obiettivi di qualità per la documentazione e le relative metriche per perseguirli.

2.1.2.1.1 Leggibilità e comprensibilità

Per valutare la leggibilità dei documenti prodotti dal gruppo e il loro livello di comprensione da parte di persone con una licenza superiore, viene adottato l'indice Gulpease. Si desidera che la documentazione del progetto abbia un indice di leggibilità compreso almeno tra 40 e 60, viene considerato ottimale un indice superiore a 60.

Riassumendo:

- **Metrica utilizzata:** Gulpease;
- **Range di accettabilità:** [40 - 60];
- **Range di ottimalità:** [60 - 100].

Per una descrizione dettagliata della metrica utilizzata si faccia riferimento alla sezione 3.3.2.1.1.

2.1.2.1.2 Correttezza ortografica

Per sapere quanto i documenti siano effettivamente corretti a livello ortografico si adotta la percentuale di errori ortografici rinvenuti e non corretti. Il gruppo desidera che tutti gli errori trovati vengano corretti, dunque l'obiettivo minimo è anche quello ottimale.

Riassumendo:

- **Metrica utilizzata:** Errori ortografici rinvenuti e non corretti;
- **Range di accettabilità:** correzione di tutti gli errori trovati;
- **Range di ottimalità:** correzione di tutti gli errori trovati.



Per una descrizione dettagliata della metrica utilizzata si faccia riferimento alla sezione 3.3.2.1.2.

2.1.2.1.3 Correttezza concettuale

Per sapere quanto i documenti siano effettivamente corretti dal punto di vista concettuale si adotta la percentuale di errori concettuali rinvenuti e non corretti. Il gruppo desidera che gli errori trovati e non corretti non superino almeno il 5%, l'obiettivo ottimale è che tutti gli errori vengano corretti.

Riassumendo:

- **Metrica utilizzata:** Errori concettuali rinvenuti e non corretti;
- **Range di accettabilità:** percentuale di errori non corretti minore del 5%;
- **Range di ottimalità:** percentuale di errori non corretti pari a 0%.

Per una descrizione dettagliata della metrica utilizzata si faccia riferimento alla sezione 3.3.2.1.3.

2.1.2.2 Qualità del software

Al fine di garantire la qualità del prodotto software, il gruppo ha deciso di adottare lo standard ISO/IEC 9126³, esso classifica la qualità del software e definisce delle metriche utili per la sua misurazione. In particolare il gruppo si prefigge di garantire le seguenti qualità per il prodotto software:

- deve possedere le funzionalità descritte dai requisiti obbligatori;
- deve possedere le funzionalità descritte dai requisiti desiderabili;
- il codice deve risultare manutenibile e facilmente comprensibile;
- deve risultare affidabile e robusto;
- deve essere testato in ogni sua parte per garantirne il funzionamento.

2.1.2.2.4 Funzionalità obbligatorie

Il prodotto deve implementare tutte le funzionalità descritte dai requisiti obbligatori. Per monitorare lo stato di implementazione di tali funzionalità si rapportano i requisiti obbligatori completati con quelli ancora da completare.

- **Metrica utilizzata:** Copertura requisiti obbligatori;
- **Soglia di accettabilità:** 100% dei requisiti obbligatori soddisfatti;
- **Soglia di ottimalità:** 100% dei requisiti obbligatori soddisfatti.

Per una descrizione dettagliata della metrica utilizzata si faccia riferimento alla sezione 3.3.2.2.4.

³Si veda appendice A.2 per approfondimenti



2.1.2.2.5 Funzionalità desiderabili

Il prodotto deve implementare tutte le funzionalità descritte dai requisiti desiderabili. Per monitorare lo stato di implementazione di tali funzionalità si rapportano i requisiti desiderabili completati con quelli ancora da completare.

- **Metrica utilizzata:** Copertura requisiti desiderabili;
- **Soglia di accettabilità:** 100% dei requisiti desiderabili soddisfatti;
- **Soglia di ottimalità:** 100% dei requisiti desiderabili soddisfatti.

Per una descrizione dettagliata della metrica utilizzata si faccia riferimento alla sezione 3.3.2.2.5.

2.1.2.2.6 Manutenibilità e comprensibilità del codice

Il gruppo ha stabilito l'adozione di diverse metriche per capire quanto il codice sviluppato sia effettivamente manutenibile e comprensibile. In particolare vengono analizzati diversi aspetti per garantire tale obiettivo di qualità, essi si basano sul fatto che una funzione o modulo complesso è quasi sicuramente anche inefficiente. Vediamo in dettaglio quantitativo le metriche prese in considerazione ed i relativi valori che il gruppo desidera raggiungere.

- **Metrica utilizzata:** Numero di metodi (NOM);
- **Range di accettabilità:** [5 - 10];
- **Range di ottimalità:** [1 - 5].

Per una descrizione dettagliata di questa metrica si faccia riferimento alla sezione 3.3.2.2.6.

- **Metrica utilizzata:** Numero di parametri per metodo;
- **Range di accettabilità:** [3 - 5];
- **Range di ottimalità:** [0 - 3].

Per una descrizione dettagliata di questa metrica si faccia riferimento alla sezione 3.3.2.2.7.

- **Metrica utilizzata:** Variabili non utilizzate e non definite;
- **Soglia di accettabilità:** 0;
- **Soglia di ottimalità:** 0.

Per una descrizione dettagliata di questa metrica si faccia riferimento alla sezione 3.3.2.2.8.

- **Metrica utilizzata:** Complessità ciclomatica;
- **Range di accettabilità:** [10 - 25];
- **Range di ottimalità:** [0 - 10].

Per una descrizione dettagliata di questa metrica si faccia riferimento alla sezione 3.3.2.2.9.

- **Metrica utilizzata:** Halstead difficulty per function;
- **Range di accettabilità:** [15 - 30];
- **Range di ottimalità:** [0 - 15].

Per una descrizione dettagliata di questa metrica si faccia riferimento alla sezione 3.3.2.2.10.

- **Metrica utilizzata:** Halstead volume per function;
- **Range di accettabilità:** [1000 - 1500];
- **Range di ottimalità:** [20 - 1000].

Per una descrizione dettagliata di questa metrica si faccia riferimento alla sezione 3.3.2.2.10.

- **Metrica utilizzata:** Halstead effort per function;
- **Range di accettabilità:** [300 - 400];
- **Range di ottimalità:** [0 - 300].

Per una descrizione dettagliata di questa metrica si faccia riferimento alla sezione 3.3.2.2.10.

- **Metrica utilizzata:** Maintainability index;
- **Range di accettabilità:** [70 - 90];
- **Range di ottimalità:** [>90].

Per una descrizione dettagliata di questa metrica si faccia riferimento alla sezione 3.3.2.2.11.

2.1.2.2.7 Affidabilità del software

Il software prodotto deve essere affidabile e robusto, senza interruzioni o anomalie. Per perseguire tale obiettivo di qualità si adottano due metriche utili a monitorare l'efficacia dei test pianificati sul codice: la Statement Coverage e la Branch Coverage. In esse si analizza rispettivamente il numero di linee di codice e il numero di flussi logici effettivamente eseguiti nel processo di testing, per facilità di lettura questi indici sono espressi in percentuale sui rispettivi totali.

- **Metrica utilizzata:** Statement Coverage;
- **Range di accettabilità:** [70 - 85];
- **Range di ottimalità:** [85 - 100].

Per una descrizione dettagliata di questa metrica si faccia riferimento alla sezione 3.3.2.2.12.

- **Metrica utilizzata:** Branch Coverage;
- **Range di accettabilità:** [70 - 85];
- **Range di ottimalità:** [85 - 100].

Per una descrizione dettagliata di questa metrica si faccia riferimento alla sezione 3.3.2.2.13.



2.1.2.2.8 Copertura dei test

Per garantire il funzionamento del prodotto, il software deve essere testato in ogni sua parte. Nei test si controllano le funzioni previste dai requisiti. Si desidera eseguire con successo almeno l'80% dei test, sarebbe invece ottimale avere una copertura almeno del 90%.

Riassumendo:

- **Metrica utilizzata:** Copertura dei test;
- **Range di accettabilità:** [80 - 90];
- **Range di ottimalità:** [90 - 100].

Per una descrizione dettagliata della metrica utilizzata si faccia riferimento alla sezione 3.3.2.2.14.

2.2 Organizzazione temporale

Viene verificata la qualità dei singoli processi e dei loro output come risultato dei periodi descritti nel *Piano di Progetto v4.0.0*. Riportiamo qui le fasi e le relative attività di verifica:

- **Analisi Preliminare:** Periodo che controlla il rispetto dei processi e della documentazione prodotta rispetto le *Norme di Progetto v4.0.0*. Sarà inoltre verificata la corrispondenza tra requisiti e casi d'uso.
- **Analisi di Dettaglio:** Periodo di verifica dei processi incrementali relativi all'analisi e ai nuovi documenti di progettazione.
- **Progettazione Architettuale e di Dettaglio:** Verifica che i test siano adeguatamente pianificati come descritto nel *Piano di Progetto v4.0.0* ed eseguiti secondo quanto descritto nelle *Norme di Progetto v4.0.0*.
- **Codifica e Test:** Periodo di verifica dei processi incrementali relativi alla progettazione assieme alla verifica delle attività di codifica tramite tecniche di analisi statica e dinamica.
- **Validazione e Collaudo:** Periodo di attuazione dei test pianificati e verifica dei processi incrementali relativi alla codifica.



3 Strategia di gestione della qualità nel dettaglio

Il *Piano di Progetto v4.0.0* fissa una serie di scadenze improrogabili, risulta necessario definire con chiarezza una strategia di qualifica efficace. Gli incrementi sulla documentazione o sul codice possono essere di natura programmata, quindi prefissati nel calendario, oppure possono insorgere come inaspettati. In questo caso sarà necessario programmare le dovute modifiche; è questo il caso di *bug*_G o errori (vedi paragrafo 4.1). La qualità di ogni incremento è basata sul fatto che la struttura di qualifica garantisce il rispetto delle *Norme di Progetto v4.0.0*. Tale lavoro verrà svolto con l'aiuto di automatismi che segnaleranno le problematiche rilevate in modo da permettere una rapida correzione. L'utilizzo di software apposito permette di eseguire controlli mirati senza consumare risorse umane. L'implementazione di tali controlli viene descritta nelle *Norme di Progetto v4.0.0*.

3.1 Responsabilità

La responsabilità delle verifiche è attribuita al Responsabile di progetto e ai Verificatori. All'interno del *Piano di Progetto v4.0.0* sono definiti i compiti e le modalità di attuazione.

3.2 Risorse

Il funzionamento del processo di verifica è garantito grazie al consumo di risorse, distinguibili nelle categorie a seguire.

3.2.1 Umane

Le figure coinvolte sono il Responsabile di progetto e il Verificatore. I processi da loro effettuati consumano ore di produttività contabilizzate e schedate secondo il *Piano di Progetto v4.0.0*. Le ore di produttività sono fissate dalle regole di progetto (<http://www.math.unipd.it/~tullio/IS-1/2015/Progetto/PD01b.html>) in un minimo di 85 e un massimo di 105 ore individuali. Il *Piano di Progetto v4.0.0* determina la distribuzione di tali quote orarie con la relativa retribuzione. Ai fini della qualifica si potrà parlare di ore di produttività, tralasciandone l'aspetto economico, in quanto non rientra nel dominio del documento succitato;

3.2.2 Tecnologiche

Riguardano i mezzi utilizzati per gli automatismi, la qualità e la loro gestione. Trattandosi esclusivamente di mezzi informatici, vengono consumate unità di calcolo considerate a costo nullo. Tale considerazione si basa sul fatto che tutti i tipi di elaborazioni informatiche sono svolte su mezzi per i quali non è richiesto né un contributo economico, né un quantitativo temporale abbastanza consistente da poter essere considerato degno di nota. Le modalità del loro impiego sono descritte all'interno del documento *Norme di Progetto v4.0.0*.



3.3 Misure e metriche

Per monitorare l'andamento dei processi e la qualità del prodotto vengono adottate delle metriche che rendono misurabili e valutabili i processi, i documenti ed il software prodotto. Vengono qui riportate nel dettaglio le metriche citate nel capitolo precedente, suddividendole per la loro natura.

3.3.1 Metriche per i processi

Tali metriche monitorano e prevedono l'andamento delle principali variabili critiche del progetto: tempi e costi. Sono utilizzate metriche di tipo consultivo le quali consentono un riscontro immediato sullo stato attuale del progetto; ad ogni incremento verranno valutati tali indici e se necessario verranno stabiliti opportuni provvedimenti da parte del Responsabile di progetto.

3.3.1.1 Modello SPICE

Per controllare e verificare la qualità dei processi, il gruppo adotterà le metriche fornite dal modello $SPICE_G$ dove per ogni fase di lavoro si andrà a fornire un indice che descriverà la qualità della fase presa in esame. L'indice sarà relativo ad una scala già definita da SPICE. Effettuando questo tipo di verifiche il team avrà subito un riscontro della qualità del processo. SPICE ci consente di individuare la maturità di un processo, essa può assumere un valore che va da 1 (il peggiore) a 5 (il migliore).

Mettendo ora in relazione i risultati di tale modello con le soglie da noi stabilite otteniamo quanto segue:

- il valore 1 è considerato negativo;
- i valori 2 e 3 sono considerati accettabili;
- i valori 4 e 5 sono considerati ottimali.

3.3.1.2 Schedule Variance

Permette di calcolare le tempistiche rispetto la schedulazione delle attività pianificate alla data corrente. È un indicatore di efficacia soprattutto nei confronti del Cliente.

$$SV = BCWP - BCWS$$

Dove:

- **BCWP**: indica il valore delle attività realizzate alla data corrente;
- **BCWS**: indica il costo pianificato per realizzare le attività di progetto alla data corrente.



Quindi con:

- **SV>0**: il lavoro prodotto è in anticipo rispetto quanto pianificato;
- **SV<0**: il lavoro è in ritardo;
- **SV=0**: il lavoro è in linea con quanto stabilito.

3.3.1.3 Budget Variance

Permette di calcolare i costi rispetto alla data corrente. È un indicatore che ha un valore unicamente contabile e finanziario.

$$BV = BCWS - ACWP$$

Dove:

- **BCWS**: indica il costo pianificato per realizzare le attività di progetto alla data corrente;
- **ACWP**: indica il costo effettivamente sostenuto per realizzare le attività di progetto alla data corrente.

Quindi:

- **BV>0**: il budget speso è minore rispetto quanto pianificato;
- **BV<0**: il budget è maggiore di quanto pianificato in ritardo;
- **BV=0**: il budget speso è in linea con quanto stabilito.

3.3.1.4 Produttività

Produttività di documentazione

Indica la produttività media di documentazione delle risorse impiegate, valutando quindi le persone coinvolte durante i diversi stadi del progetto.

$$\text{Produttività di documentazione} = \text{Parole} / \text{Ore persona}$$

Dove:

- **Parole**: indica il numero di parole presente nei documenti;
- **Ore persona**: indica il numero di ore produttive dei componenti del gruppo.

Parametri utilizzati

- **Range negativo**: [<70];
- **Range accettazione**: [$70 - 100$];
- **Range ottimale**: [≥ 100].



Produttività di test

Indica la produttività media dei test realizzati.

$$\text{Produttività di test} = \text{Numero di test} / \text{Ore persona}$$

Dove:

- **Numero di test:** indica il numero di test eseguiti;
- **Ore persona:** indica il numero di ore produttive dei componenti del gruppo.

Parametri utilizzati

- **Range negativo:** [<5];
- **Range accettazione:** [5 - 10];
- **Range ottimale:** [>10].

Produttività di codifica

Indica la produttività media delle attività di codifica. Una bassa produzione di linee di codice non denota uno scarso impegno, bensì sottolinea l'efficienza di produttività impiegata.

$$\text{Produttività di codifica} = \text{LOCs} / \text{Ore persona}$$

Dove:

- **LOCs:** indica il numero di linee di codice prodotte;
- **Ore persona:** indica il numero di ore produttive dei componenti del gruppo.

Parametri utilizzati

- **Range negativo:** [>20];
- **Range accettazione:** [5 - 20];
- **Range ottimale:** [2 - 10].

3.3.1.5 Impegno

Indica l'impegno richiesto dal gruppo per la realizzazione del progetto.

$$\text{Impegno} = \text{Dimensione} / \text{Produttività}$$

Dove:

- **Dimensione:** indica il tempo produttivo impiegato;
- **Produttività:** indica la media delle produttività totali (di documentazione, di test, di codifica).



Parametri utilizzati

- **Range negativo:** $[-0,4]$;
- **Range accettazione:** $[0,5 - 0,6]$;
- **Range ottimale:** $[>0,6]$.

3.3.2 Metriche per il prodotto

Vengono qui riportate nel dettaglio le metriche per i prodotti del progetto. Tali metriche monitorano e prevedono le performance di sviluppo del progetto, ad ogni incremento verranno valutati tali indici e se necessario verranno stabiliti opportuni provvedimenti da parte del Responsabile di progetto.

3.3.2.1 Metriche per i documenti

La qualità dei documenti è garantita dalla loro leggibilità, dalla loro correttezza ortografica e concettuale. Si utilizzano quindi delle metriche utili a monitorare tali caratteristiche attese, descritte in dettaglio di seguito.

Il Diario delle modifiche viene incluso in ogni documento mostrando così lo storico del documento.

3.3.2.1.1 Gulpease

L'indice Gulpease è un indice per la leggibilità di un testo tarato sulla lingua italiana. Questo indice ha il vantaggio di utilizzare la lunghezza delle parole in lettere anziché in sillabe, semplificandone il calcolo automatico. Permette di misurare la complessità dello stile di un documento. In questo calcolo sono considerate due variabili linguistiche: la lunghezza della parola e la lunghezza della frase rispetto al numero delle lettere.

$$\text{Indice Gulpease} = 89 + \frac{300 * (\text{numero delle frasi}) - 10 * (\text{numero delle lettere})}{\text{numero delle parole}}$$

Il range dei risultati varia tra 0 e 100, dove 100 indica la leggibilità massima e 0 quella minima. In generale i documenti con:

- un indice inferiore a 80 sono difficili da leggere per chi ha la sola licenza elementare;
- un indice inferiore a 60 sono difficili da leggere per chi ha la sola licenza media;
- un indice inferiore a 40 sono difficili da leggere per chi ha un diploma superiore.

Parametri utilizzati

- **Range negativo:** $[0 - 40]$;
- **Range accettazione:** $[40 - 100]$;
- **Range ottimale:** $[60 - 100]$.



3.3.2.1.2 Errori ortografici rinvenuti e non corretti

Questa metrica risulta utile per determinare quanto un documento sia corretto dal punto di vista ortografico. Supponendo infatti che gli strumenti automatici descritti nelle *Norme di Progetto v4.0.0* siano in grado di trovare tutti (o almeno la maggioranza) degli errori ortografici contenuti nei documenti, la correttezza ortografica non può che basarsi sul numero di errori rinvenuti ma successivamente non corretti. Notare che per errori corretti si intende una scorrettezza ortografica revisionata manualmente da parte di un Verificatore. Le correzioni automatiche infatti, non essendo attendibili, non sono prese in considerazione.

Parametri utilizzati

- **Range negativo:** percentuale di errori non corretti maggiore allo 0%;
- **Range accettazione:** percentuale di errori non corretti pari a 0%;
- **Range ottimale:** percentuale di errori non corretti pari a 0%.

3.3.2.1.3 Errori concettuali rinvenuti e non corretti

Questa metrica è necessaria per determinare quanto un documento sia corretto dal punto di vista concettuale. Supponendo infatti che inseguito alle revisioni di progetto siano stati trovati tutti (o almeno la maggioranza) degli errori di questo tipo contenuti nei documenti, la correttezza concettuale non può che basarsi sul numero di errori rinvenuti ma successivamente non corretti. Notare che per errori corretti si intende un errore fatto notare dal committente in sede di revisione o dai Verificatori e successivamente corretto.

Parametri utilizzati

- **Range negativo:** percentuale di errori non corretti maggiore al 5%;
- **Range accettazione:** percentuale di errori non corretti minore del 5%;
- **Range ottimale:** percentuale di errori non corretti pari a 0%.

3.3.2.2 Metriche per il software

Per garantire la qualità del software, il gruppo *MINT*, ha deciso di adottare delle metriche con il compito di monitorare la qualità interna, esterna e quella in uso. In base alle risorse a disposizione e agli obiettivi di qualità preposti per il software, si fa quindi riferimento al modello e alle relative metriche citati nel capitolo precedente: lo standard ISO/IEC 9126⁴.

Di seguito vengono elencate le metriche per il software prodotto e le relative caratteristiche di qualità che intendono valutare.

⁴Si veda appendice A.2 per approfondimenti



Metriche scelte	Caratteristiche di qualità
Copertura requisiti obbligatori	Funzionalità
Copertura requisiti desiderabili	Funzionalità
Numero di metodi per package	Manutenibilità
Numero di parametri per metodo	Manutenibilità
Variabili non utilizzate e non definite	Manutenibilità
Complessità ciclomatica	Manutenibilità
Metriche di Halstead	Manutenibilità
Maintainability index	Manutenibilità
Statement coverage	Affidabilità
Branch coverage	Affidabilità
Copertura dei test	Affidabilità

Tabella 2: Tracciamento Metriche software - Caratteristiche di qualità

Vengono ora trattate nel dettaglio le metriche adottate.

3.3.2.2.4 Copertura requisiti obbligatori

Questa metrica ci permette di verificare in ogni momento lo stato dell'implementazione dei requisiti obbligatori. Essa controlla infatti il rapporto percentuale tra i requisiti obbligatori soddisfatti e il numero totale dei requisiti obbligatori richiesti.

$$\text{Copertura requisiti obbligatori} = \frac{\text{Numero di requisiti obbligatori soddisfatti} \cdot 100}{\text{Numero totale di requisiti obbligatori}}$$

Parametri utilizzati

- **Range negativo:** [$<100\%$];
- **Soglia accettazione:** 100%;
- **Soglia ottimale:** 100%.

3.3.2.2.5 Copertura requisiti desiderabili

Questa metrica ci permette di verificare in ogni momento lo stato dell'implementazione dei requisiti desiderabili. Essa controlla infatti il rapporto percentuale tra i requisiti desiderabili soddisfatti e il numero totale dei requisiti desiderabili richiesti.

$$\text{Copertura requisiti desiderabili} = \frac{\text{Numero di requisiti desiderabili soddisfatti} \cdot 100}{\text{Numero totale di requisiti desiderabili}}$$

Parametri utilizzati

- **Range negativo:** [$<100\%$];
- **Soglia accettazione:** 100%;
- **Soglia ottimale:** 100%.



3.3.2.2.6 Numero di metodi - NOM

Il Number of Methods è una metrica usata per calcolare la media delle occorrenze dei metodi per package. Un package non dovrebbe contenere un numero eccessivo di metodi. Valori superiori al range ottimale massimo potrebbero indicare una necessità di maggiore decomposizione del package.

Parametri utilizzati

- **Range negativo:** [>10];
- **Range accettazione:** [$5 - 10$];
- **Range ottimale:** [$1 - 5$].

3.3.2.2.7 Numero di parametri per metodo

Un numero elevato di parametri per un metodo potrebbe evidenziare un metodo troppo complesso.

Non c'è una regola forte per il numero di parametri possibili in un metodo o costruttore, citando Robert Martin, in Clean Code⁵ :

“The ideal number of arguments for a function is zero (niladic). Next comes one (monadic), followed closely by two (dyadic). Three arguments (triadic) should be avoided where possible. More than three (polyadic) requires very special justification – and then shouldn't be used anyway.”

e Steve McConnell, in Code Complete⁶:

“limit the number of a routine's parameters to about seven, seven is a magic number for people's comprehension”

Vengono quindi seguite le linee guida dei seguenti parametri.

Parametri utilizzati

- **Range negativo:** [>5];
- **Range accettazione:** [$3 - 5$];
- **Range ottimale:** [$0 - 3$].

3.3.2.2.8 Variabili non utilizzate e non definite

La presenza di variabili non utilizzate viene considerata *pollution_G*, pertanto non viene tollerata. Tali occorrenze vengono rilevate analizzando l'*Abstract syntax tree_G* (AST) eseguendo una cross-reference tra le variabili dichiarate e quelle inizializzate. Per sua natura, *Javascript_G* non blocca l'insorgenza di tali occorrenze, pertanto si rischia di dichiarare una variabile e poi utilizzarne una con nome leggermente diverso, oppure semplicemente dichiarare una variabile che in seguito non verrà mai utilizzata.

⁵Robert Martin, Clean Code: A Handbook of Agile Software Craftsmanship. Prentice Hall (2008)

⁶Steve McConnell, Code Complete: A Practical Handbook of Software Construction. Microsoft Press (2004)



Parametri utilizzati

- **Range negativo:** $[>0]$;
- **Soglia accettazione:** 0;
- **Soglia ottimale:** 0.

3.3.2.2.9 Complessità ciclomatica

La complessità ciclomatica è una metrica software che indica la complessità di un programma misurando il numero di cammini linearmente indipendenti attraverso il grafo di controllo di flusso. Nel grafo sopracitato i nodi corrispondono a gruppi indivisibili di istruzioni, mentre gli archi connettono due nodi se il secondo gruppo di istruzioni può essere eseguito immediatamente dopo il primo gruppo. Tale indice può essere applicato indistintamente a singole funzioni, *moduli*, metodi o package di un programma. Si vuole utilizzare tale metrica per limitare la complessità durante le attività di sviluppo del prodotto software. Può rivelarsi utile durante il testing per determinare il numero di casi di test necessari, infatti l'indice di complessità è un limite superiore al numero di test necessari per raggiungere il coverage completo del modulo testato. Inoltre, uno studio ha mostrato forti corrispondenze tra le metriche di complessità e il livello di coesione nei package presi in esame⁷.

Parametri utilizzati

- **Range negativo:** $[>25]$;
- **Range accettazione:** $[10 - 25]$;
- **Range ottimale:** $[0 - 10]$.

3.3.2.2.10 Metriche di Halstead

Le metriche di *Halstead_c* oltre a rappresentare indici di complessità, permettono di identificare le proprietà misurabili del software e le relative relazioni. Si basano sull'osservazione che una metrica dovrebbe valutare l'implementazione di un algoritmo in linguaggi differenti ed essere indipendente dall'esecuzione su una specifica piattaforma.

Sono identificati i seguenti dati all'interno di un problema:

- **n_1** : indica il numero di operatori distinti;
- **n_2** : indica il numero di operandi distinti;
- **N_1** : indica il numero totale di operatori;
- **N_2** : indica il numero totale di operandi.

Da cui si ottiene:

- **$n = n_1 + n_2$** : vocabolario della funzione;
- **$N = N_1 + N_2$** : lunghezza della funzione.

⁷Stein, C., G. Cox and L. Etzkorn, 2005. Exploring the Relationship between Cohesion and Complexity. J. Comput. Sci., 1: 137-144.



Data la scarsa disponibilità nella rete di valori di riferimento, i range specificati sono frutto di un confronto tra il $report_G$ sulla complessità di una libreria *open source*_G presa come esempio (<https://github.com/philbooth/complexity-report/blob/master/EXAMPLE.md>) e i valori dichiarati in <http://www.mccabe.com/pdf/McCabeIQMetrics.pdf>. Questi valori vengono dichiarati momentanei (RR) e saranno da rivalutare sia considerando altre fonti, sia considerando i valori rilevati in parti del codice che il gruppo considera come riferimento.

Halstead difficulty per function

Il livello di difficoltà di una funzione misura la propensione all'errore ed è proporzionale al numero di operatori presenti.

$$D = \left(\frac{n1}{2}\right) * \left(\frac{N2}{n2}\right)$$

Parametri utilizzati

- **Range negativo:** [>30];
- **Range accettazione:** [15 - 30];
- **Range ottimale:** [0 - 15].

Halstead volume per function

Il volume descrive la dimensione dell'implementazione di un algoritmo e si basa sul numero di operazioni eseguite e sugli operandi di una funzione. Il volume di una function senza parametri composta da una sola linea è 20, mentre un indice superiore a 1000 indica che probabilmente la funzione esegue troppe operazioni.

$$V = N * \log_2 n$$

Parametri utilizzati

- **Range negativo:** [>1500];
- **Range accettazione:** [1000 - 1500];
- **Range ottimale:** [20 - 1000].

Halstead effort per function

Lo sforzo per implementare o comprendere il significato di una funzione è proporzionale al volume e al suo livello di difficoltà.

$$E = V * D$$

Parametri utilizzati

- **Range negativo:** [>400];
- **Range accettazione:** [300 - 400];
- **Range ottimale:** [0 - 300].



3.3.2.2.11 Maintainability index

Questa metrica⁸ è una scala logaritmica da $-\infty$ a 171, calcolata sulla base delle linee di codice logiche, della complessità ciclomatica e dall'indice Halstead effort. Un valore alto indica una maggiore manutenibilità.

Parametri utilizzati

- **Range negativo:** [<70];
- **Range accettazione:** [70 - 90];
- **Range ottimale:** [>90].

3.3.2.2.12 Statement Coverage

Permette di calcolare quante linee di codice di ciascuna unità sono eseguite almeno una volta nell'esecuzione dei test. Tale metrica è espressa in percentuale.

Parametri utilizzati

- **Range negativo:** [<70];
- **Range accettazione:** [70 - 85];
- **Range ottimale:** [85 - 100].

3.3.2.2.13 Branch Coverage

Permette di calcolare quanti rami della logica di flusso sono attraversati almeno una volta durante l'esecuzione dei test. Tale metrica è espressa in percentuale.

Parametri utilizzati

- **Range negativo:** [<70];
- **Range accettazione:** [70 - 85];
- **Range ottimale:** [85 - 100].

3.3.2.2.14 Copertura dei test

Questa metrica esamina la percentuale di successo dei test ricavati dai requisiti e dalle relative funzionalità che il software dovrà ottenere. Indica infatti la percentuale dei test eseguiti con successo.

$$\text{Copertura dei test} = \frac{\text{Numero di test superati} \cdot 100}{\text{Numero totale di test pianificati}}$$

⁸Definita nel 1991 da Paul Oman e Jack Hagemester alla University of Idaho.



Parametri utilizzati

- **Range negativo:** [<80];
- **Range accettazione:** [80 - 90];
- **Range ottimale:** [90 - 100].



4 Gestione amministrativa della revisione

4.1 Comunicazione delle anomalie

Identificare le anomalie permette la correzione dei difetti ricercati dal processo di *Software Quality Management_G* e informa il Responsabile di progetto sullo stato del prodotto. Analizzare e catalogare le anomalie è utile per discutere, durante revisioni e riunioni, su quali modifiche e correzioni applicare e con quale priorità. Di seguito è presente la lista delle definizioni di anomalie (IEEE 610.12-90) adottate dal gruppo:

- **Error:** differenza riscontrata tra il risultato di una computazione e il valore teorico atteso (e.g. uscita dal range di accettazione degli indici di misurazione);
- **Fault:** un passo, un processo o un dato definito in modo erraneo (e.g. violazioni di norme tipografiche da parte di un documento). Corrisponde a quanto viene definito come bug;
- **Failure:** il risultato di un fault (e.g. incongruenza del prodotto con funzionalità indicate nell'analisi dei requisiti, incongruenza del codice con il design del prodotto);
- **Mistake:** azione umana che produce un risultato errato (e.g. anomalie nel repository).

La distinzione delle anomalie consente di impostare le metriche per valutarne l'andamento e in alcuni casi predirlo, in particolare è stata scelta la metrica che conta il numero di bug per lines of code. Il gruppo utilizzerà un *SCR_G* (Software Change Request) individuato nelle *Norme di Progetto v4.0.0*.

4.2 Procedure di controllo per la qualità di processo

Le procedure di controllo per la qualità di processo hanno il fine di migliorare la qualità del prodotto e diminuire i costi e tempi di sviluppo. Esistono due approcci principali:

- **A maturità di processo:** riflette le buone pratiche di management e tecniche di sviluppo. L'obiettivo primario è la qualità del prodotto e la prevedibilità dei processi;
- **Agile:** sviluppo iterativo senza l'overhead della documentazione e di tutti gli aspetti pre-determinabili. Ha come caratteristica la responsività ai cambiamenti dei requisiti cliente e uno sviluppo rapido.

Il team adotterà il primo approccio, essendo più adatto ad un gruppo inesperto. Con una visione proattiva si cerca di avere maggior controllo e previsione sulle attività da svolgere. Questa viene anche indicata come *best practice_G* per gruppi poco esperti.

Il processo con maggiore influenza sulla qualità del sistema non è quello di sviluppo ma quello di progettazione. È qui che le capacità e le esperienze dei singoli danno un contributo decisivo. Il miglioramento dei processi è un processo ciclico composto da tre sotto-processi:

- **Misurazione del processo:** misura gli attributi del progetto, punta ad allineare gli obiettivi con le misurazioni effettuate. Questo forma una *baseline_G* che aiuta a capire se i miglioramenti hanno avuto effetto;
- **Analisi del processo:** vengono identificate le problematiche ed i colli di bottiglia dei processi;

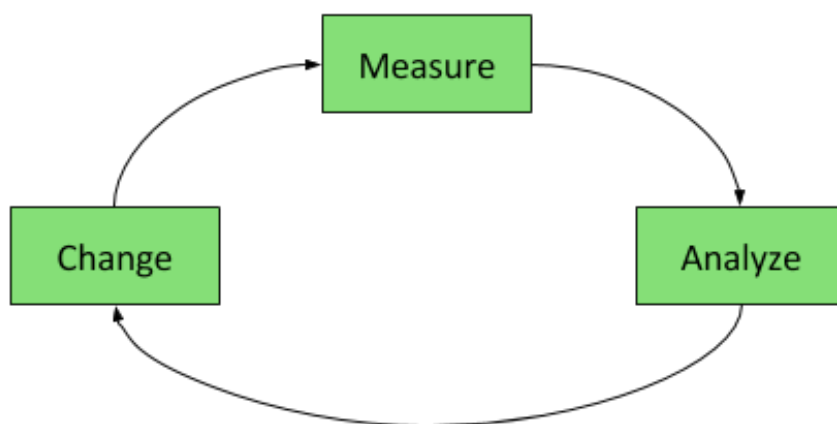


Figura 1: Il ciclo di miglioramento dei processi

- **Modifiche del processo:** i cambiamenti vengono proposti in risposta alle problematiche riscontrate.

Il team procederà nel seguente modo:

- Nella sezione Dettaglio delle verifiche tramite analisi (C.2) verranno inserite le misurazioni rilevate sulle le metriche descritte in Misure e Metriche (3.3);
- L'analisi viene effettuata i giorni precedenti alle consegne previste dal committente. Il Resoconto delle attività di verifica (C) contiene l'analisi del processo e le relative considerazioni comprendenti le problematiche riscontrate;
- Le modifiche al processo vengono attuate all'inizio del processo incrementale successivo. Queste attività sono programmate nel *Piano di Progetto v4.0.0*.

A Qualità

La qualità perseguita nel presente documento si basa sugli standard ISO/IEC 15504 e ISO/IEC 9126 con l'obiettivo di approfondirne incrementalmente la copertura.

A.1 Qualità di processo

La qualità del processo è un fattore determinante per la qualità del prodotto. Si è deciso di perseguirla servendosi dei modelli $SPICE_G$ e $PDCA_G$.

A.1.1 Standard ISO/IEC 15504

La qualità di processo definita in questo standard come SPICE (Software Process Improvement and Capability Determination), specifica come la qualità è collegata alla maturazione dei processi. Il $team_G$ ha scelto questo standard ai fini di una valutazione oggettiva dei processi, per darne un giudizio di maturità e per individuare azioni migliorative. Vengono individuati dei livelli di maturità al quale il fornitore può fare riferimento per determinare le proprie capacità organizzative. Vengono definiti:

- **Modelli di riferimento:**

- Dimensione del processo;
- Livelli di maturità dei processi:
 - * **5:** Ottimizzato
 - * **4:** Predicibile
 - * **3:** Stabilito
 - * **2:** Gestito
 - * **1:** Eseguito
 - * **0:** Incompleto

La capacità di un processo viene misurata tramite degli attributi che sono assimilabili alle metriche dei processi individuate in 3.3.1, in particolare la $Schedule Variance_G$ permette di capire se un processo è incompleto o gestito; il gruppo raggiungerà uno stato accettabile quando i processi diventeranno predicibili ossia quando la $Schedule Variance$ subirà al più lievi oscillazioni;

- **Stime:** si concretizzano in una struttura per la misurazione composta da:
 - I processi di misurazione, indicati nel *Piano di Progetto v4.0.0*;
 - Un modello per la misurazione identificabile in questo documento;
 - Gli strumenti utilizzati, specificati nelle *Norme di Progetto v4.0.0*.
- **Competenze e Qualifiche di chi controlla:** lo standard redige in modo rigoroso una serie di attività volte a formare chi opera l'attività di verifica e di stesura del Piano di

Qualifica. Tali competenze sono assenti all'interno del gruppo e, considerato che effettuare una formazione in linea con quanto specificato dallo standard sarebbe impossibile, tutti i membri si impegnano a studiare ed applicare al meglio quanto descritto in questo documento.

A.1.2 Ciclo di Deming

La qualità va ricercata non sul prodotto bensì sui processi alla base del prodotto, per questo il team ha scelto il metodo PDCA per il controllo delle attività di processo ripetibili e misurabili e per la manutenibilità dei processi stessi. Esso prevede l'iterazione ripetuta tra i quattro stadi definiti di seguito, assicurando un incremento della qualità ad ogni ciclo.

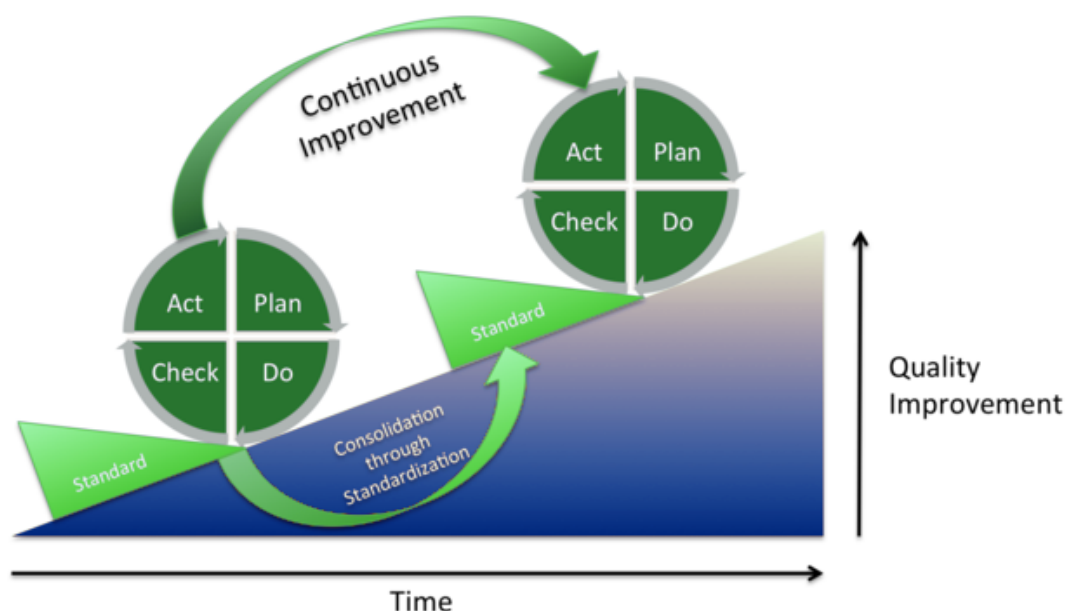


Figura 2: Continuous quality improvement with PDCA

1. **PLAN:** vengono stabiliti gli obiettivi e i processi di miglioramento necessari per raggiungere la qualità attesa, nel dettaglio:
 - Identificare il problema, o i processi da migliorare; per descrivere il problema è necessario raccogliere i dati tramite misurazioni;
 - Analizzare il problema e individuare gli effetti negativi, definendo la loro importanza e le priorità di intervento;
 - Definire gli obiettivi di massima in modo chiaro e quantitativo, indicando i benefici ottenibili con il suo raggiungimento. Devono essere definiti anche i tempi, gli indicatori e gli strumenti di controllo.



2. **DO**: viene implementato il punto precedente, applicando le soluzioni individuate al problema;
3. **CHECK**: verificare i risultati delle azioni intraprese; un confronto con i risultati attesi sarà il riscontro se quanto operato va nella direzione giusta. Vanno considerate metriche come la Schedule Variance (vedi 3.3.1.2) e la completezza dei risultati attesi soddisfatti, vanno elaborati grafici e tabelle per avere una visione chiara di quanto rilevato. Se si è raggiunto l'obiettivo definito nello stadio di Plan, si può passare allo stadio di Act, altrimenti è necessario ripetere un nuovo ciclo PDCA sullo stesso problema, analizzando i vari stadi del ciclo precedente individuandone le cause del non raggiungimento dell'obiettivo stabilito;
4. **ACT**: La soluzione individuata viene standardizzata e tutti i membri del gruppo vengono informati e formati. Si potrà eseguire tramite riunioni o strumenti di messaggistica interna al gruppo. Terminato questo stadio si procederà con una nuova iterazione a partire dal punto 1.

A.2 Qualità di prodotto

Sono necessari degli obiettivi rivolti direttamente alla qualità del prodotto per massimizzare l'efficacia. Lo standard ISO/IEC 9126 classifica la qualità del software e definisce delle metriche per la sua misurazione.

A.2.1 Standard ISO/IEC 9126

Prendendo come riferimento questo standard il team MINT si impegna a garantire nel prodotto MaaS le qualità da esso definite.

Esso si suddivide in:

- **Quality model**: classifica la qualità del software in un set di caratteristiche che verranno approfondite nel corso del progetto:
 - **Functionality**: viene controllata grazie al tracciamento dei requisiti individuati ed analizzati e i componenti;
 - **Reliability**: viene dimostrata combinando i test;
 - **Usability**: viene controllata con i test di validazione, inoltre la stesura del manuale d'uso aiuterà a verificarne l'usabilità e ad intervenire laddove necessario;
 - **Efficiency**: combinando analisi statica e dinamica controlliamo che il prodotto sia efficiente;
 - **Maintainability**: viene realizzata con l'utilizzo di design pattern e la stesura di documentazione dettagliata;
 - **Portability**: essendo $MaaS_G$ un applicazione Web non ci sono particolari problemi di portabilità per gli utenti.
- **External metrics**: sono le metriche rilevate tramite analisi dinamica specificate in 3.3;
- **Internal metrics**: sono le metriche rilevate in analisi statica specificate in 3.3;



- **Quality in use metrics:** si tratta di metriche rilevabili allo stato di prodotto *usabile* in condizioni reali, si rimanda la definizione di tale aspetto a quando verranno trattate le considerazioni sull'usabilità del prodotto in uno scenario di utilizzo reale, questo deve avvenire non oltre la *Progettazione di Dettaglio e Codifica*.



B Pianificazione dei test

Si vuole adottare una strategia di verifica del software tramite test opportunamente predeterminati, garantendo almeno un test per ogni requisito. Essi sono l'applicazione delle tecniche di verifica dinamica introdotte nelle *Norme di Progetto v4.0.0*. Tali attività, oltre a richiedere l'esecuzione del programma, devono poter essere ripetibili, ossia tramite delle specifiche su come riprodurre i test vogliamo che il loro output sia deterministico. È importante che i test di unità vengano svolti in parallelo, dando precedenza alle unità che producono risultati utili alla comprensione del loro funzionamento integrato. L'ambiente di testing deve soddisfare tale obiettivo. L'attività di test deve produrre un \log_G che specifica quando e chi ha eseguito il test e con quali input; l'insorgenza di $failure_G$ deve essere tracciata e catalogata.

B.1 Livelli di testing

Il testing del software viene suddiviso in livelli differenti e si concretizzano in un'esecuzione bottom-up che avanza sequenzialmente alle attività di codifica e di validazione. I test che si andranno ad applicare sono di cinque tipi:

- **Test di Validazione (TV):** viene verificato che il prodotto soddisfi quanto richiesto dal proponente individuando delle macro azioni da eseguire sul sistema che un normale utente svolge comunemente;
- **Test di Sistema (TS):** sono test relativi al comportamento dell'intero sistema ossia viene verificato che la sua architettura generale funzioni complessivamente bene;
- **Test di Integrazione (TI):** vengono verificate le componenti del sistema contenute nella *Specifica Tecnica v0.0.0*, ossia viene verificato che i $package_G$ siano funzionanti e in grado di funzionare nel loro insieme;
- **Test di Unità (TU):** viene testata ogni unità, ossia la più piccola parte di lavoro assegnabile ad un programmatore. In questo progetto una unità corrisponde ad una `function` o a un `method`;
- **Test di Regressione (TR):** possono essere test di tutte le tipologie succitate che devono mostrare il funzionamento del prodotto a seguito di una modifica.

La specifica delle ultime due tipologie viene riservata alla prossima revisione.

La figura 3 illustra come i test elencati vengono distribuiti durante il ciclo di sviluppo del prodotto.

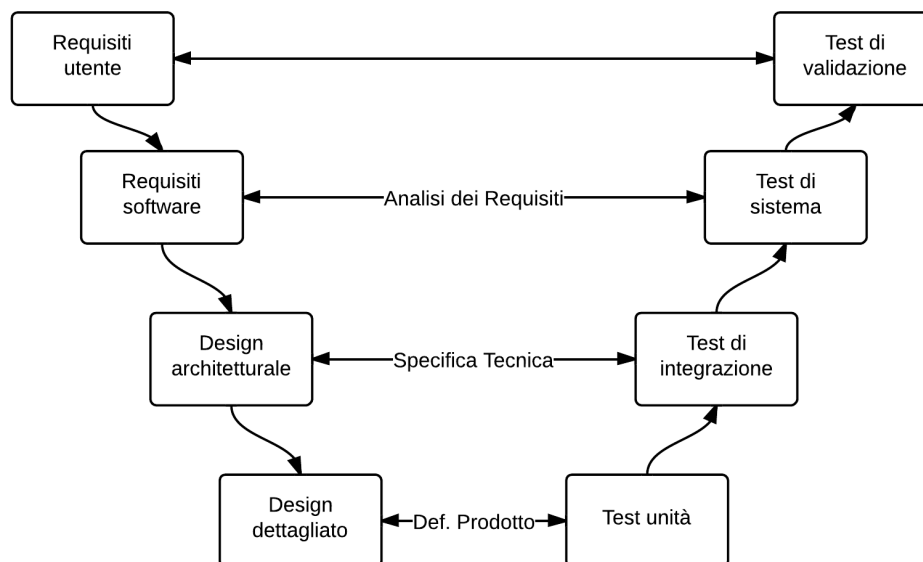


Figura 3: V-Model per il testing software

B.2 Test di validazione

In questa sezione vengono elencati i test di validazione per verificare che il prodotto sia conforme alle attese. I test si svolgono seguendo e verificando tutti passi di cui si compongono. I test eseguiti sono indicati con una *E* (Eseguito), mentre i test non eseguiti sono indicati con *N.E* (Non Eseguito). I test implementati sono indicati con una *I* (Implementato), mentre i test non implementati sono indicati con *N.I* (Non Implementato).



Test di Validazione	Descrizione	Stato
TV-RS1O 1	L'utente non autenticato intende autenticarsi inserendo i dati necessari in un'apposita form nella pagina di autenticazione. All'utente è richiesto di: <ul style="list-style-type: none">• Raggiungere la pagina di autenticazione;• Inserire l'email e la password;• Procedere con l'autenticazione.	N.E - N.I
TV-RS1O 2.1	L'utente non autenticato intende registrarsi compilando i campi della form nella pagina di registrazione. All'utente è richiesto di: <ul style="list-style-type: none">• Raggiungere la pagina di registrazione;• Inserire l'email nel campo apposito;• Inserire la password conforme ai vincoli richiesti;• Inserire la conferma della password;• Inserire il nome dell'azienda che desidera creare;• Procedere con la registrazione.	N.E - N.I
TV-RS1O 2.2	L'utente non autenticato intende unirsi ad un'azienda già esistente, che l'ha invitato tramite email. All'utente è richiesto di: <ul style="list-style-type: none">• Raggiungere la pagina di registrazione grazie al link ricevuto per email;• Inserire la password conforme ai vincoli richiesti;• Inserire la conferma della password;• Procedere con la registrazione.	N.E - N.I



TV-RS1D 3	<p>L'utente autenticato è intenzionato a modificare i dati del proprio account mediante una pagina web dedicata. All'utente è richiesto di:</p> <ul style="list-style-type: none">• Essere autenticato;• Aprire la pagina relativa alle modifiche dell'account;• Decidere se modificare il proprio avatar;• Decidere se modificare la propria password;• Decidere se modificare i propri dati anagrafici quali nome, cognome e data di nascita;• Decidere se modificare il proprio sesso;• Rendere effettive le modifiche.	N.E - N.I
TV-RS10 4	<p>Il Proprietario o l'Amministratore è intenzionato a connettere o disconnettere un database oppure modificarne i permessi mediante una pagina web dedicata. É richiesto di:</p> <ul style="list-style-type: none">• Essere autenticato;• Aprire la pagina relativa alla gestione database;• Decidere se connettere un database inserendo il nome del nuovo database;• Decidere se disconnettere un database;• Decidere se modificare i permessi di un database ai membri dell'azienda;• Decidere se cercare un database appartenente all'azienda;• Rendere effettive le modifiche.	N.E - N.I



TV-RS10 5	<p>L'utente autenticato è intenzionato ad eseguire operazioni di lettura, scrittura e modifica dei permessi in ambito dei DSLIS, mediate l'utilizzo di un editor testuale e un'interfaccia che visualizza la lista delle definizioni DSL. All'utente è richiesto di:</p> <ul style="list-style-type: none">• Essere autenticato;• Aprire l'editor di testo;• Decidere se creare un nuovo DSLIS o modificare quelli già esistenti. É possibile definire una Dashboard, una Collection, un Document e una Cell;• Decidere se eseguire una ricerca dei DSLIS sfoltendo i risultati della lista;• Decidere se eseguire un DSLIS;• Decidere se eseguire una Action sui dati visualizzati da un DSLIS;• Decidere se esportare un DSLIS;• Decidere se importare un DSLIS;• Decidere se eliminare i DSLIS già esistenti;• Decidere se visualizzare il codice di un DSLIS;• Decidere se modificare i permessi di lettura, scrittura o esecuzione associati ad un DSLIS.	N.E - N.I
TV-RS10 6	<p>Il Proprietario o l'Amministratore è intenzionato a manipolare gli account degli altri utenti dell'azienda. É richiesto di:</p> <ul style="list-style-type: none">• Essere autenticato come Proprietario o Amministratore;• Aprire la pagina relativa alla gestione account dei membri dell'azienda;• Decidere se rimuovere un utente del tipo Amministratore, Membro o Ospite;• Aggiungere un nuovo membro all'azienda, invitandolo tramite email e affidandogli un determinato ruolo;• Decidere di modificare il ruolo di un utente promuovendolo o facendolo retrocedere di ruolo;• Rendere effettive le modifiche.	N.E - N.I



TV-RS10 7	L'utente autenticato è intenzionato a ad eseguire il logout dal sistema. È richiesto di: <ul style="list-style-type: none">• Essere autenticato;• Eseguire il logout tramite l'apposito tasto presente in ogni pagina di MaaS.	N.E - N.I
TV-RS10 8	L'utente non autenticato è intenzionato a recuperare la propria password, ripristinandola. È richiesto di: <ul style="list-style-type: none">• Recarsi alla pagina di recupero password;• Inserire la propria email e richiedere il ripristino;• Seguire il link ricevuto nell'email inviata dal sistema;• Inserire la nuova password nella pagina raggiunta tramite il link. Essa dovrà rispettare i requisiti ed essere inserita una seconda volta per conferma;• Confermare i dati inseriti.	N.E - N.I
TV-RS10 9	L'utente autenticato è intenzionato a modificare le proprie preferenze. È richiesto di: <ul style="list-style-type: none">• Essere autenticato;• Decidere se modificare la Dashboard attiva tra quelle disponibili;• Decidere se effettuare una ricerca tra Dashboard disponibili;• Rendere effettive le modifiche.	N.E - N.I
TV-RS10 10	L'utente autenticato è intenzionato a visualizzare la pagina relativa alla propria Dashboard. È richiesto di: <ul style="list-style-type: none">• Essere autenticato;• Recarsi alla pagina di visualizzazione della Dashboard attiva.	N.E - N.I
TV-RS10 12	L'utente autenticato, è intenzionato a cercare un altro utente iscritto al sistema. È richiesto di: <ul style="list-style-type: none">• Essere autenticato;• Effettuare una ricerca inserendo il nome o la mail dell'utente desiderato.	N.E - N.I



TV-RS10 13.1	<p>Il Super-Amministratore è intenzionato a gestire le aziende contenute nel database interno del sistema. È richiesto di:</p> <ul style="list-style-type: none">• Essere autenticato come Super-Amministratore;• Recarsi alla pagina di gestione delle aziende;• Decidere se ricercare un'azienda specifica;• Decidere se modificare il nome di un'azienda;• Decidere se eliminare un'azienda;	N.E - N.I
TV-RS10 13.2	<p>Il Super-Amministratore è intenzionato a gestire gli utenti di un'azienda contenuti nel database interno del sistema. È richiesto di:</p> <ul style="list-style-type: none">• Essere autenticato come Super-Amministratore;• Recarsi alla pagina di gestione degli utenti di un'azienda;• Decidere se modificare il nome di un utente;• Decidere se modificare la password di un utente;• Decidere se eliminare un utente;	N.E - N.I
TV-RS10 14	<p>Il Super-Amministratore è intenzionato a impersonificare un utente. È richiesto di:</p> <ul style="list-style-type: none">• Essere autenticato come Super-Amministratore;• Recarsi alla pagina di visualizzazione degli utenti di un'azienda;• Effettuare una ricerca sulla lista selezionando l'utente da impersonificare.	N.E - N.I
TV-RS10 15	<p>Il Super-Amministratore è intenzionato ad autenticarsi mediante un'apposita pagina web. È richiesto di:</p> <ul style="list-style-type: none">• Non essere autenticato;• Recarsi alla pagina autenticazione del Super Amministratore;• Inserire l'email e la password;• Procedere con l'autenticazione.	N.E - N.I



TV-RS10 16	<p>Il Super-Amministratore è intenzionato ad eseguire il logout. È richiesto di:</p> <ul style="list-style-type: none">• Essere autenticato come Super-Amministratore;• Selezionare l'apposito pulsante di logout presente in tutte le pagine.	N.E - N.I
TV-RS10 18	<p>Il Super-Amministratore è intenzionato a cercare un utente. È richiesto di:</p> <ul style="list-style-type: none">• Essere autenticato come Super-Amministratore;• Effettuare una ricerca inserendo il nome o l'email dell'utente desiderato.	N.E - N.I

Tabella A.1: Test di Validazione



B.3 Test di sistema

Vengono qui descritti i test di sistema che andranno a verificare il funzionamento complessivo delle componenti. I test eseguiti sono indicati con una *E* (Eseguito), mentre i test non eseguiti sono indicati con *N.E* (Non Eseguito). I test implementati sono indicati con una *I* (Implementato), mentre i test non implementati sono indicati con *N.I* (Non Implementato).



Test di Sistema	Descrizione	Stato
TS-RS1O 1.1	Verificare che durante l'autenticazione di un utente l'indirizzo email venga immesso tramite un campo di testo apposito.	N.E - I
TS-RS1O 1.2	Verificare che durante l'autenticazione di un utente, la password venga immessa tramite un capo di testo apposito.	N.E - I
TS-RS1O 1.3	Verificare che, in caso di fallimento dell'autenticazione di un utente, il sistema visualizzi una pagina di errore.	N.E - I
TS-RS1O 1.4	Verificare che, in caso di successo dell'autenticazione di un utente, il sistema lo reindirizzi ad una pagina contenente la dashboard attiva.	N.E - I
TS-RS1O 2.1.1	Verificare che durante la registrazione di un nuovo Proprietario l'indirizzo email venga immesso tramite un campo di testo apposito.	N.E - I
TS-RS1O 2.1.2	Verificare che durante la registrazione di un nuovo Proprietario la password e la sua conferma vengano immessi tramite dei campi di testo appositi.	N.E - I
TS-RS1O 2.1.2.1	Verificare che durante la registrazione di un nuovo Proprietario venga richiesto che la password sia composta di un minimo di 6 caratteri.	N.E - I
TS-RS1O 2.1.2.2	Verificare che durante la registrazione di un nuovo Proprietario venga richiesto che la password sia composta di un massimo di 12 caratteri.	N.E - I
TS-RS1O 2.1.2.3	Verificare che durante la registrazione di un nuovo Proprietario venga richiesto che la password sia composta da caratteri alfanumerici con possibili caratteri di punteggiatura.	N.E - I
TS-RS1O 2.1.3	Verificare che durante la registrazione di un nuovo Proprietario il nome dell'azienda da creare venga immesso tramite un campo di testo apposito.	N.E - I
TS-RS1O 2.2.1	Verificare che durante la registrazione di un nuovo utente invitato la password e la sua conferma vengano immessi tramite dei campi di testo appositi.	N.E - I



TS-RS1O 2.2.1.1	Verificare che durante la registrazione di un nuovo utente invitato venga richiesto che la password sia composta di un minimo di 6 caratteri.	N.E - I
TS-RS1O 2.2.1.2	Verificare che durante la registrazione di un nuovo utente invitato venga richiesto che la password sia composta di un massimo di 12 caratteri.	N.E - I
TS-RS1O 2.2.1.3	Verificare che durante la registrazione di un nuovo utente invitato venga richiesto che la password sia composta da caratteri alfanumerici con possibili caratteri di punteggiatura.	N.E - I
TS-RS1O 2.3	Verificare che, in caso di fallimento della registrazione di un utente per email non valida, il sistema visualizzi una pagina di errore specificando tale causa.	N.E - I
TS-RS1O 2.4	Verificare che, in caso di fallimento della registrazione di un utente per password non valida, il sistema visualizzi una pagina di errore specificando se le due password inserite non corrispondono o se la password non soddisfa la struttura richiesta.	N.E - I
TS-RS1O 2.5	Verificare che, in caso di fallimento della registrazione di un Proprietario per nome azienda già presente, il sistema visualizzi una pagina di errore specificando tale causa.	N.E - I
TS-RS1O 2.6	Verificare che, in caso di successo della registrazione di un utente, il sistema visualizzi una pagina di avvenuta registrazione.	N.E - I
TS-RS1D 3.1	Verificare che il sistema permetta di modificare il proprio avatar mediante un apposito bottone che consente di fare l'upload di una nuova immagine.	N.E - I
TS-RS1O 3.2	Verificare che il sistema permetta di modificare la propria password mediante un apposito campo di testo.	N.E - I
TS-RS1O 3.3	Verificare che, in caso di fallimento della modifica della password, il sistema visualizzi un messaggio di errore.	N.E - I
TS-RS1D 3.4	Verificare che il sistema metta a disposizione una pagina per la modifica dei dati anagrafici dell'utente autenticato.	N.E - I



TS-RS1D 3.4.1	Verificare che, durante la modifica dei dati anagrafici, il sistema metta a disposizione campo di testo per la modifica del nome dell'utente.	N.E - I
TS-RS1D 3.4.2	Verificare che, durante la modifica dei dati anagrafici, il sistema metta a disposizione campo di testo per la modifica del cognome dell'utente.	N.E - I
TS-RS1D 3.4.3	Verificare che, durante la modifica dei dati anagrafici, il sistema metta a disposizione campo di testo per la modifica della data di nascita dell'utente.	N.E - I
TS-RS1D 3.4.4	Verificare che, durante la modifica dei dati anagrafici, il sistema metta a disposizione campo di testo per la modifica del sesso dell'utente.	N.E - I
TS-RS1O 4.1	Verificare che il sistema metta a disposizione del Proprietario e degli Amministratori la possibilità di connettere un nuovo database e fornisca un campo di testo per l'inserimento del nome del database.	N.E - I
TS-RS1O 4.2	Verificare che il sistema metta a disposizione del Proprietario e degli Amministratori la possibilità di disconnettere un database e fornisca un bottone apposito.	N.E - I
TS-RS1O 4.3	Verificare che il sistema metta a disposizione del Proprietario e degli Amministratori la possibilità di modificare i permessi di accesso ad un database e fornisca un'interfaccia apposita, contenente la lista dei membri dell'azienda ed un'apposita checkbox per ognuno di essi.	N.E - I
TS-RS1O 4.4	Verificare che il sistema visualizzi un messaggio di errore al Proprietario o agli Amministratori che cerchino un database non presente nel sistema.	N.E - I
TS-RS1O 4.5	Verificare che il sistema permetta al Proprietario o agli Amministratori di cercare un database dell'azienda presente nel sistema.	N.E - I
TS-RS1F 5.1	Verificare che il sistema fornisca un editor di testo con funzionalità avanzate, come l'evidenziazione sintattica, l'autocompletamento, l'indentazione ed i suggerimenti.	N.E - I



TS-RS1O 5.2	Verificare che il sistema permetta di creare nuove DSLIS mediante un'interfaccia apposita contenente l'editor testuale e degli appositi bottoni per compilare e salvare il codice.	N.E - I
TS-RS1O 5.3	Verificare che il sistema permetta di modificare le DSLIS mediante un'interfaccia apposita contenente l'editor testuale e degli appositi bottoni per compilare e salvare il codice.	N.E - I
TS-RS1O 5.4	Verificare che il sistema permetta di eliminare una DSLIS mediante un apposito bottone presente in ogni elemento della lista delle istruzioni.	N.E - I
TS-RS1O 5.5	Verificare che il sistema permetta di visualizzare il codice delle DSLIS mediante un'interfaccia apposita contenente un editor testuale.	N.E - I
TS-RS1O 5.6	Verificare che il sistema permetta di modificare i permessi di esecuzione, scrittura e lettura dei DSLIS mediante un'apposita interfaccia contenente la lista degli utenti dell'azienda iscritta a MaaS.	N.E - I
TS-RS1O 5.6.1	Verificare che il sistema permetta di modificare i permessi di esecuzione di un DSLIS mediante un apposito checkbox.	N.E - I
TS-RS1O 5.6.2	Verificare che il sistema permetta di modificare i permessi di scrittura di un DSLIS mediante un apposito checkbox.	N.E - I
TS-RS1O 5.6.3	Verificare che il sistema permetta di modificare i permessi di lettura di un DSLIS mediante un apposito checkbox.	N.E - I
TS-RS1O 5.6.4	Verificare che il sistema fornisca un'interfaccia contenente la lista dei DSLIS a cui l'utente ha accesso.	N.E - I
TS-RS1O 5.6.5	Verificare che il sistema fornisca una funzionalità di ricerca per sfoltire i dati visualizzati nella lista dei DSLIS.	N.E - I
TS-RS1O 5.6.6	Verificare che il sistema fornisca un'interfaccia contenente la lista degli utenti dell'azienda iscritta a MaaS.	N.E - I
TS-RS1O 5.6.7	Verificare che il sistema fornisca una funzionalità di ricerca per sfoltire i dati visualizzati nella lista degli utenti.	N.E - I



TS-RS1O 5.7	Verificare che il sistema permetta di definire una Dashboard mediante la parola chiave dashboard .	N.E - I
TS-RS1O 5.7.1	Verificare che il sistema permetta di definire una Row di una Dashboard mediante la parola chiave row .	N.E - I
TS-RS1O 5.7.1.1	Verificare che il sistema permetta di definire una Collection all'interno di una Row di una Dashboard mediante la parola chiave collection .	N.E - I
TS-RS1O 5.7.1.2	Verificare che il sistema permetta di definire un Document all'interno di una Row di una Dashboard mediante la parola chiave document .	N.E - I
TS-RS1O 5.7.1.3	Verificare che il sistema permetta di definire una Cell all'interno di una Row di una Dashboard mediante la parola chiave cell .	N.E - I
TS-RS1O 5.7.2	Verificare che il sistema permetta di definire il nome di una Dashboard mediante la parola chiave name .	N.E - I
TS-RS1O 5.8	Verificare che il sistema permetta di definire una Collection mediante la parola chiave collection .	N.E - I
TS-RS1O 5.8.1	Verificare che il sistema permetta di definire l'identità di una Collection mediante le parole chiave name label , id e weight .	N.E - I
TS-RS1O 5.8.1.1	Verificare che il sistema permetta di definire il nome di una collection MongoDB, a cui la Collection si deve referenziare, mediante la parola chiave name .	N.E - I
TS-RS1O 5.8.1.2	Verificare che il sistema permetta di definire il nome da visualizzare di una Collection, mediante la parola chiave label .	N.E - I
TS-RS1O 5.8.1.3	Verificare che il sistema permetta di definire l'id di una Collection mediante la parola chiave id e che di default esso assuma il valore del campo name .	N.E - I
TS-RS1O 5.8.1.4	Verificare che il sistema permetta di definire l'ordine di visualizzazione di una Collection all'interno della lista dei DSLIS mediante la parola chiave weight e che di default esso assuma il valore 0.	N.E - I



TS-RS1O 5.8.2	Verificare che il sistema permetta di definire l'index di una Collection mediante la parola chiave index .	N.E - I
TS-RS1O 5.8.2.1	Verificare che il sistema permetta di definire gli attributi da visualizzare di una Collection mediante la parola chiave column .	N.E - I
TS-RS1O 5.8.2.2	Verificare che il sistema permetta di definire l'ordinamento degli attributi di una Collection mediante le parole chiave sortBy e order .	N.E - I
TS-RS1O 5.8.2.3	Verificare che il sistema permetta di definire il limite di elementi da visualizzare di una Collection mediante la parola chiave perPage .	N.E - I
TS-RS1O 5.8.2.4	Verificare che il sistema permetta di definire la query di una Collection mediante la parola chiave query .	N.E - I
TS-RS1O 5.8.2.5	Verificare che il sistema permetta di definire gli attributi selezionabili di una Collection mediante la parola chiave selectable .	N.E - I
TS-RS1O 5.8.2.6	Verificare che il sistema permetta di definire gli attributi ordinabili di una Collection mediante la parola chiave sortable .	N.E - I
TS-RS1O 5.8.3	Verificare che il sistema permetta di definire le Action di una Collection mediante la parola chiave action . Verificare inoltre che esso fornisca le parole chiave export e sendEmail per definire le rispettive azioni.	N.E - I
TS-RS1O 5.9	Verificare che il sistema permetta di definire un Document mediante la parola chiave document .	N.E - I
TS-RS1O 5.9.1	Verificare che il sistema permetta di definire gli attributi da visualizzare di un Document mediante la parola chiave row .	N.E - I
TS-RS1O 5.9.2	Verificare che il sistema permetta di definire il tipo degli attributi di un Document mediante la parola chiave type e fornisca inoltre le parole chiave array , object , link e image per definire i vari tipi.	N.E - I
TS-RS1O 5.9.3	Verificare che il sistema permetta di definire le Action di un Document mediante la parola chiave action . Verificare inoltre che esso fornisca le parole chiave export e sendEmail per definire le rispettive azioni.	N.E - I



TS-RS1O 5.10	Verificare che il sistema permetta di definire una Cell mediante la parola chiave cell .	N.E - I
TS-RS1O 5.10.1	Verificare che il sistema permetta di definire la query di una Cell mediante la parola chiave query .	N.E - I
TS-RS1O 5.10.2	Verificare che il sistema permetta di definire l'ordinamento dei risultati della query di una Cell mediante le parole chiave sortBy e order .	N.E - I
TS-RS1O 5.10.3	Verificare che il sistema permetta di definire il tipo di una Cell mediante la parola chiave type e fornisca inoltre le parole chiave string , number , link , image e date per definire i vari tipi.	N.E - I
TS-RS1O 5.10.4	Verificare che il sistema permetta di definire il valore arbitrario di una Cell mediante la parola chiave value .	N.E - I
TS-RS1O 5.11	Verificare che il sistema permetta di definire una Action mediante la parola chiave action .	N.E - I
TS-RS1O 5.11.1	Verificare che il sistema permetta di definire l'Export di una Action mediante la parola chiave export .	N.E - I
TS-RS1O 5.11.2	Verificare che il sistema permetta di definire la Send Email di una Action mediante la parola chiave sendEmail .	N.E - I
TS-RS1O 5.12	Verificare che il sistema visualizzi un messaggio di errore nel caso in cui l'utente cerchi un DSLIS non memorizzato al suo interno.	N.E - I
TS-RS1O 5.13	Verificare che il sistema visualizzi un messaggio di errore nel caso in cui l'utente compili un DSLIS in cui sono presenti errori di sintassi.	N.E - I
TS-RS1O 5.14	Verificare che il sistema permetta di eseguire un DSLIS, mettendo a disposizione dell'utente un'interfaccia apposita contenente la lista dei DSLIS a cui ha accesso.	N.E - I
TS-RS1O 5.14.1	Verificare che il sistema consenta all'utente di eseguire un'Action sui dati visualizzati da un DSLIS, fornendo degli appositi bottoni.	N.E - I
TS-RS1O 5.14.1.1	Verificare che il sistema consenta all'utente di eseguire un'Export sui dati visualizzati da un DSLIS, fornendo degli appositi bottoni.	N.E - I



TS-RS1O 5.14.1.2	Verificare che il sistema consenta all'utente di eseguire una Send Email sui dati visualizzati da un DSLIS, fornendo degli appositi bottoni.	N.E - I
TS-RS1O 5.15	Verificare che il sistema permetta di importare un DSLIS, fornendo un bottone apposito.	N.E - I
TS-RS1O 5.16	Verificare che il sistema permetta di esportare un DSLIS, fornendo un bottone apposito.	N.E - I
TS-RS1O 5.17	Verificare che il sistema permetta di ricercare un DSLIS, tra quelli a cui l'utente ha accesso.	N.E - I
TS-RS1O 5.18	Verificare che il sistema permetta di visualizzare la lista completa dei DSLIS a cui l'utente ha accesso.	N.E - I
TS-RS1O 6.1	Verificare che il sistema permetta al Proprietario e agli Amministratori di rimuovere un utente dell'azienda, mediante un'apposita interfaccia grafica.	N.E - I
TS-RS1O 6.1.1	Verificare che il sistema permetta al Proprietario e agli Amministratori di rimuovere un Amministratore dell'azienda.	N.E - I
TS-RS1O 6.1.2	Verificare che il sistema permetta al Proprietario e agli Amministratori di rimuovere un Membro dell'azienda.	N.E - I
TS-RS1O 6.1.3	Verificare che il sistema permetta al Proprietario e agli Amministratori di rimuovere un Ospite dell'azienda.	N.E - I
TS-RS1O 6.2	Verificare che il sistema permetta al Proprietario e agli Amministratori di aggiungere un utente all'azienda, mediante un'apposita interfaccia grafica.	N.E - I
TS-RS1O 6.2.1	Verificare che il sistema permetta al Proprietario e agli Amministratori di invitare un nuovo Amministratore all'azienda, mediante un campo di testo per l'inserimento dell'email.	N.E - I
TS-RS1O 6.2.2	Verificare che il sistema permetta al Proprietario e agli Amministratori di invitare un nuovo Membro all'azienda, mediante un campo di testo per l'inserimento dell'email.	N.E - I
TS-RS1O 6.2.3	Verificare che il sistema permetta al Proprietario e agli Amministratori di invitare un nuovo Ospite all'azienda, mediante un campo di testo per l'inserimento dell'email.	N.E - I



TS-RS1O 6.3	Verificare che il sistema permetta al Proprietario e agli Amministratori di modificare il ruolo di un utente dell'azienda.	N.E - I
TS-RS1O 6.3.1	Verificare che il sistema permetta al Proprietario e agli Amministratori di promuovere un utente dell'azienda ad un ruolo superiore.	N.E - I
TS-RS1O 6.3.1.1	Verificare che il sistema permetta al Proprietario e agli Amministratori di promuovere un Ospite o un Membro dell'azienda al ruolo di Amministratore.	N.E - I
TS-RS1O 6.3.1.2	Verificare che il sistema permetta al Proprietario e agli Amministratori di promuovere un Ospite dell'azienda al ruolo di Membro.	N.E - I
TS-RS1O 6.3.2	Verificare che il sistema permetta al Proprietario e agli Amministratori di retrocedere un utente dell'azienda ad un ruolo inferiore.	N.E - I
TS-RS1O 6.3.2.1	Verificare che il sistema permetta al Proprietario e agli Amministratori di retrocedere un qualsiasi utente dell'azienda (escluso il Proprietario) al ruolo di Ospite.	N.E - I
TS-RS1O 6.3.2.2	Verificare che il sistema permetta al Proprietario e agli Amministratori di retrocedere un qualsiasi utente dell'azienda (escluso il Proprietario) al ruolo di Membro.	N.E - I
TS-RS1O 7	Verificare che il sistema permetta all'utente autenticato di effettuare il logout mediante un apposito bottone presente in ogni interfaccia di MaaS.	N.E - I
TS-RS1O 8.1	Verificare che il sistema permetta all'utente non autenticato di recuperare la propria password, mediante un'apposita interfaccia contenente un campo di testo per l'inserimento dell'email.	N.E - I
TS-RS1O 8.2	Verificare che il sistema, durante il recupero della password, invii un'email contenente un link attraverso il quale l'utente non autenticato possa effettuare il reset della propria password.	N.E - I
TS-RS1O 8.3	Verificare che il sistema permetta ad un utente non autenticato che abbia richiesto il recupero della password, di accedere ad un'apposita interfaccia che consenta il reset della propria password, tramite il link ricevuto nell'email di recupero.	N.E - I



TS-RS1O 8.3.1	Verificare che il sistema permetta ad un utente non autenticato che abbia richiesto il recupero della password, di scegliere una nuova password che sostituirà la precedente e di confermarla, mediante dei campi di testo appositi.	N.E - I
TS-RS1O 8.3.1.1	Verificare che durante il recupero password di un utente venga richiesto che la password sia composta di un minimo di 6 caratteri.	N.E - I
TS-RS1O 8.3.1.2	Verificare che durante il recupero password di un utente venga richiesto che la password sia composta di un massimo di 12 caratteri.	N.E - I
TS-RS1O 8.3.1.3	Verificare che durante il recupero password di un utente venga richiesto che la password sia composta da caratteri alfanumerici con possibili caratteri di punteggiatura.	N.E - I
TS-RS1O 8.4	Verificare che durante il recupero password di un utente il sistema visualizzi un messaggio di errore se l'email inserita per il recupero non rientra tra quelle degli iscritti a MaaS.	N.E - I
TS-RS1O 8.5	Verificare che durante il recupero password di un utente il sistema visualizzi un messaggio di errore se la password inserita per il recupero non rispetta la struttura richiesta oppure se non corrisponde a quella inserita per la conferma.	N.E - I
TS-RS1O 8.6	Verificare che il sistema visualizzi un messaggio di avvenuto reset della password nel caso in cui il suo recupero vada a buon fine.	N.E - I
TS-RS1O 9.1	Verificare che il sistema permetta all'utente di modificare la Dashboard attiva mediante un'apposita interfaccia contenente la lista di quelle a cui ha accesso.	N.E - I
TS-RS1O 9.2	Verificare che durante la modifica della Dashboard attiva il sistema permetta all'utente di cercare le Dashboard tra quelle a cui ha accesso.	N.E - I
TS-RS1O 9.3	Verificare che durante la modifica della Dashboard attiva di un utente il sistema visualizzi un messaggio di errore se la Dashboard cercata non rientra tra quelle a cui ha accesso.	N.E - I
TS-RS1O 10	Verificare che il sistema permetta all'utente autenticato di visualizzare la propria Dashboard attiva mediante un'apposita interfaccia.	N.E - I



TS-RS1O 11	Verificare che il sistema visualizzi all'utente un messaggio di errore per aver cercato un utente non presente nel sistema MaaS.	N.E - I
TS-RS1O 12	Verificare che il sistema permetta di cercare un utente attraverso un'apposita interfaccia, contenente un campo di testo in cui inserire il nome o l'email dell'utente da cercare.	N.E - I
TS-RS1O 13.1	Verificare che il sistema permetta al Super-Amministratore di gestire le aziende registrate, mediante un'apposita interfaccia grafica.	N.E - I
TS-RS1O 13.1.1	Verificare che il sistema consenta al Super-Amministratore di visualizzare la lista delle aziende registrate.	N.E - I
TS-RS1O 13.1.2	Verificare che il sistema consenta al Super-Amministratore di ricercare un'azienda tra quelle registrate.	N.E - I
TS-RS1O 13.1.3	Verificare che il sistema visualizzi al Super-Amministratore un messaggio di errore per aver cercato un'azienda non presente tra quelle registrate.	N.E - I
TS-RS1O 13.1.4	Verificare che il sistema consenta al Super-Amministratore di modificare il nome di un'azienda mediante un apposito campo di testo.	N.E - I
S-RS1O 13.1.5	Verificare che il sistema consenta al Super-Amministratore di eliminare un'azienda mediante un apposito bottone.	N.E - I
TS-RS1O 13.2	Verificare che il sistema permetta al Super-Amministratore di gestire gli utenti registrati, mediante un'apposita interfaccia grafica.	N.E - I
TS-RS1O 13.2.1	Verificare che il sistema consenta al Super-Amministratore di visualizzare la lista degli utenti registrati.	N.E - I
TS-RS1O 13.2.2	Verificare che il sistema visualizzi al Super-Amministratore un messaggio di errore per aver cercato un utente non presente tra quelli registrati.	N.E - I
TS-RS1O 13.2.3	Verificare che il sistema consenta al Super-Amministratore di modificare l'email di un utente mediante un apposito campo di testo.	N.E - I



TS-RS1O 13.2.4	Verificare che il sistema consenta al Super-Amministratore di modificare la password di un utente mediante un apposito campo di testo.	N.E - I
TS-RS1O 13.2.5	Verificare che il sistema consenta al Super-Amministratore di eliminare un utente mediante un apposito bottone.	N.E - I
TS-RS1O 14.1	Verificare che il sistema permetta al Super-Amministratore di impersonificare un utente registrato, mediante un'apposita interfaccia grafica che permetta di selezionarlo.	N.E - I
TS-RS1O 14.1.1	Verificare che il sistema permetta al Super-Amministratore di impersonificare un Proprietario.	N.E - I
TS-RS1O 14.1.2	Verificare che il sistema permetta al Super-Amministratore di impersonificare un Amministratore.	N.E - I
TS-RS1O 14.1.3	Verificare che il sistema permetta al Super-Amministratore di impersonificare un Membro.	N.E - I
TS-RS1O 14.1.4	Verificare che il sistema permetta al Super-Amministratore di impersonificare un Ospite.	N.E - I
TS-RS1O 14.2	Verificare che il sistema permetta al Super-Amministratore di visualizzare la lista degli utenti registrati.	N.E - I
TS-RS1O 14.3	Verificare che il sistema permetta al Super-Amministratore di ricercare un utente tra quelli registrati, mediante un'apposita barra di ricerca.	N.E - I
TS-RS1O 15.1	Verificare che durante l'autenticazione di un Super-Amministratore l'indirizzo email venga immesso tramite un campo di testo apposito.	N.E - I
TS-RS1O 15.2	Verificare che durante l'autenticazione di Super-Amministratore la password venga immessa tramite un capo di testo apposito.	N.E - I
TS-RS1O 15.3	Verificare che, in caso di fallimento dell'autenticazione di un Super-Amministratore, il sistema visualizzi una pagina di errore.	N.E - I
TS-RS1O 16	Verificare che il sistema permetta al Super-Amministratore di eseguire il logout mediante un apposito bottone presente in tutte le interfacce.	N.E - I



TS-RS1O 17	Verificare che il sistema visualizzi al Super-Amministratore un messaggio di errore nel caso in cui abbia ricercato un utente non presente nel sistema.	N.E - I
TS-RS1O 18	Verificare che il sistema permetta al Super-Amministratore di ricercare un utente mediante l'inserimento del nome o dell'email in una barra di ricerca apposita.	N.E - I
TS-RS2O 1	Verificare che il sistema suddivida i risultati derivanti da grandi quantità di dati elaborati in diverse pagine, per garantire la sua stabilità.	N.E - I
TS-RS4O 1	Verificare che il sistema venga implementato nella piattaforma Heroku.	N.E - I
TS-RS4O 2	Verificare che il sistema venga sviluppato su un framework Node.js di alto livello, ad esempio LoopBack.	N.E - I
TS-RS4O 3	Verificare che il sistema venga pubblicato e versionato in una repository di GitHub o BitBucket.	N.E - I
TS-RS4O 4	Verificare che il sistema si basi su un database sviluppato in MongoDB di versione 3.x o superiore.	N.E - I
TS-RS4O 5	Verificare che il sistema utilizzi Node.js come back-end in versione LTS (Long Term Support) Argon (v4.2.0).	N.E - I
TS-RS4O 6	Verificare tutte le password del sistema MaaS siano persistenti e criptate in qualsiasi momento, usando la libreria scrypt.	N.E - I
TS-RS4O 7	Verificare che il sistema si avvalga dei JWT (JSON Web Tokens) per i link del ripristino della password e per l'invito di nuovi utenti.	N.E - I
TS-RS4O 8	Verificare che il sistema verifichi l'autenticità di un utente tramite il controllo di email e password confrontando i dati con quelli presenti nel database interno di MaaS.	N.E - I
TS-RS4O 9	Verificare che il sistema verifichi la correttezza de dati inseriti dall'utente in fase di registrazione.	N.E - I

Tabella A.2: Test di Sistema

B.4 Test di integrazione

I test di integrazione servono per verificare che tutti i diversi componenti del sistema comunichino correttamente tra di loro, e che vi sia all'interno del software il flusso di dati atteso. Verrà utilizzata una strategia di integrazione incrementale per poter sviluppare e verificare più componenti in parallelo. Questo metodo permette di dare priorità ai test relativi alle componenti che vengono ritenute più importanti. In questo modo sarà possibile partire dalle componenti che soddisfano i requisiti obbligatori fino ad integrarli con le componenti che soddisfano i requisiti opzionali. Permette anche di restringere la ricerca dell'errore in caso di test fallito, in quanto sarà molto probabile che l'errore risulti dal nuovo componente o dalle sue interazioni con il sistema corrente. Non sarà da escludere il caso in cui il test fallisca per il fatto che la nuova istanza di test adoperi un campione di input non precedentemente trattato, portando così il sistema a generare un errore.

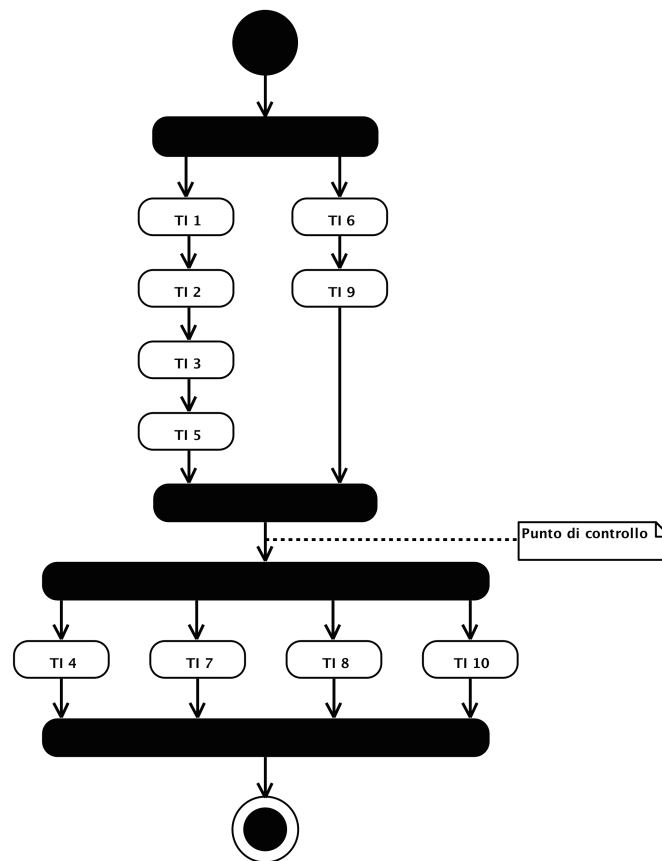


Figura 4: Sequenza d'integrazione delle componenti

Si è scelto di utilizzare un approccio top-down il quale garantisce che le componenti di più alto livello siano testate non appena sono implementate. Man mano che si procede con la codifica delle componenti di più basso livello, queste vengono integrate e viene eseguito il relativo test. Grazie all'integrazione incrementale delle componenti del sistema, è più semplice



determinare quale componente crea problemi e le funzioni di più alto livello sono testate prima.

I test eseguiti sono indicati con una *E* (Eseguito), mentre i test non eseguiti sono indicati con *N.E* (Non Eseguito). I test implementati sono indicati con una *I* (Implementato), mentre i test non implementati sono indicati con *N.I* (Non Implementato).



Test	Descrizione	Componenti aggiunte	Stato
TI 1	Si verifica che l'applicazione Web carichi correttamente le librerie JavaScript utilizzate.	Front-end	N.E - I.
TI 2	Si verifica che sia funzionante il collegamento con il server e con l'ActionCreators.	Front-end::WebAPIUtils	N.E - I.
TI 3	Si verifica che l'ActionCreators riceva correttamente gli input dalla View e crei le action ad essi associate.	Front-end::ActionCreators	N.E - I.
TI 4	Si verifica che gli Stores sia caricati correttamente e comunichino i cambiamenti alla View.	Front-end::Stores	N.E - I.
TI 5	Si verifica che la View funzioni correttamente permettendo il caricamento e la visualizzazione di tutte le pagine.	Front-End::View	N.E - I.
TI 6	Si verifica che il Middleware si integri correttamente nella gestione delle richieste che arrivano al server.	Back-end::Middlewares	N.E - I.
TI 7	Si verifica che il Model si integri correttamente della gestione dell'inserimento, della modifica, della creazione e dell'eliminazione consistente dei dati.	Back-end::Model	N.E - I.
TI 8	Si verifica che il Datasource si integri correttamente nella gestione dei Database esterni.	Back-end::Datasource	N.E - I.
TI 9	Si verifica che le RESTAPIs mappino correttamente le richieste HTTP da parte del client.	Back-end::RESTAPIs	N.E - I.
TI 10	Si verifica che il Connectors si connetta correttamente ai database esterni e al servizio di posta elettronica.	Back-end::Connectors	N.E - I.

Tabella A.3: Descrizione test d'Integrazione



B.5 Test di unità

Di seguito vengono elencati i test di unità pianificati. Si ritiene sia strettamente necessario, per comprendere interamente il contenuto della seguente tabella, leggere queste informazioni avendo a portata di mano il *tracciamento metodi-test*, il quale può essere trovato all'interno del documento *Definizione di Prodotto v2.0.0*.

Di seguito vengono elencati i test di unità pianificati. I test eseguiti sono indicati con una *E* (Eseguito), mentre i test non eseguiti sono indicati con *N.E* (Non Eseguito). I test implementati sono indicati con una *I* (Implementato), mentre i test non implementati sono indicati con *N.I* (Non Implementato).

Nome	Descrizione	Stato
TU - 1	Verifica che un utente sia stato inserito correttamente nel database di MaaS e che non ci siano errore.	Eseguito.
TU - 2	Verifica che un utente si a riuscito ad autenticarsi al sistema MaaS, senza che siano avvenuti errori che non dipendono da lui.	Eseguito.
TU - 3	Verifica della corretta esecuzione di una modifica su un utente.	Eseguito.
TU - 4	Verifica che un utente riesca a disconnettersi senza che avvengano errori.	Eseguito.
TU - 5	Verifica che si riescano ad ottenere i dati degli utenti di un'azienda.	Eseguito.
TU - 6	Verifica che si riescano ad ottenere i dati relativi ad un utente di un'azienda.	Eseguito.
TU - 7	Controllo dell'effettiva eliminazione di un utente dal database di MaaS.	Eseguito.
TU - 8	Verifica che si riesca ad ottenere una lista di utenti aventi un determinato valore tra i propri dati.	Eseguito.
TU - 9	Controllo che un utente che ha dimenticato la propria password riesca ad accedere al sistema di recupero password.	Eseguito.
TU - 10	Verifica che si riesca a cambiare il ruolo di un utente.	Eseguito.
TU - 11	Verifica che si riesca a generare (tramite l'Action sendEmail) una mail di invito per un nuovo utente e ad inviarla.	Eseguito.
TU - 12	Controllo della possibilità di effettuare il login da parte del Super Amministratore.	Eseguito.
TU - 13	Controllo che il Super Amministratore riesca a disconnettersi dal sistema, senza che avvengano errori.	Eseguito.



TU - 14	Si verifica che il Super Amministratore possa impersonare un utente iscritto al sistema, assumendone quindi identità, ruolo e privilegi.	Eseguito.
TU - 15	Si controlla che un Super Amministratore possa, tramite questo metodo, visualizzare una lista contenente tutte le aziende iscritte al sistema.	Eseguito.
TU - 16	Si controlla che un possibile Proprietario possa creare un'azienda ed inserirla nel database di MaaS.	Eseguito.
TU - 17	Viene verificato il funzionamento del metodo che restituisce al Super Amministratore le informazioni relative ad una specifica azienda.	Eseguito.
TU - 18	Viene verificato il funzionamento di un metodo che permette al Super Amministratore di modificare i dati di un'azienda, a scopo di supporto, qualora un Proprietario non riesca a farlo.	Eseguito.
TU - 19	Si controlla che un Super Amministratore riesca ad eliminare un'azienda dal database del sistema MaaS.	Eseguito.
TU - 20	Verifica della possibilità, da parte di un Super Amministratore, di ottenere le informazioni relative ad un'azienda.	Eseguito.
TU - 21	Si verifica che un Amministratore o Proprietario riesca a collegare un database esterno alla propria azienda, senza che si verifichino errori durante il processo.	Eseguito.
TU - 22	Si controlla che un Amministratore o Proprietario possa eliminare l'origine dei dati (database esterno) di un'azienda, senza che si verifichino errori durante il processo.	Eseguito.
TU - 23	Verifica che gli Amministratori di un'azienda iscritta al sistema MaaS siano in grado ottenere specifici dati, appartenenti alla propria azienda, conservati in un database esterno collegato.	Eseguito.
TU - 24	Si controlla che gli Amministratori di un'azienda possano cercare un database esterno collegato.	Eseguito.
TU - 25	Si controlla che un Amministratore di un'azienda possa concedere i permessi di accesso al database esterno ad un utente iscritto alla stessa azienda.	Eseguito.
TU - 26	Si controlla che gli Amministratori di un'azienda possa negare i permessi di accesso al database esterno ad un utente iscritto alla stessa azienda.	Eseguito.



TU - 27	Si verifica che un utente possa visualizzare una lista contenente tutte le Collection a cui ha accesso.	Eseguito.
TU - 28	Si verifica che un utente possa creare una nuova Collection, senza che si presentino errori.	Eseguito.
TU - 29	Si controlla che un utente possa modificare una Collection a cui ha accesso, qualora avesse permessi sufficienti.	Eseguito.
TU - 30	Si controlla che un utente possa eliminare una Collection a cui ha accesso, qualora avesse permessi sufficienti.	Eseguito.
TU - 31	Si controlla che un qualsivoglia utente possa ottenere il codice di una Collection, ossia il DSLIS da cui essa ha avuto origine.	Eseguito.
TU - 32	Si verifica la possibilità da parte di un utente di ricercare un insieme di Collection, filtrando i risultati in base ad un valore.	Eseguito.
TU - 33	Si controlla che un utente possa eseguire una Collection (ovvero il suo DSLIS) a cui ha accesso, qualora avesse permessi sufficienti, senza che avvengano errori.	Eseguito.
TU - 34	Verifica che il metodo abbia inviato via mail, tramite una Action <code>sendEmail</code> , le informazioni renderizzate da una Collection.	Eseguito.
TU - 35	Si verifica che un utente possa esportare i dati visualizzati in una Collection, senza che si verifichino errori.	Eseguito.
TU - 36	Si controlla che un utente possa esportare il codice DSLIS generatore di una Collection, senza che si verifichino errori.	Eseguito.
TU - 37	Si controlla che un utente possa importare il codice DSLIS generatore di una Collection.	Eseguito.
TU - 38	Si verifica che un utente possa visualizzare una lista contenente tutti i Document a cui ha accesso.	Eseguito.
TU - 39	Si verifica che un utente possa creare un nuovo Document, senza che si presentino errori.	Eseguito.
TU - 40	Si controlla che un utente possa modificare un Document a cui ha accesso, qualora avesse permessi sufficienti.	Eseguito.
TU - 41	Si controlla che un utente possa eliminare un Document a cui ha accesso, qualora avesse permessi sufficienti.	Eseguito.
TU - 42	Si verifica la possibilità da parte di un utente di ricercare un insieme di Document, filtrando i risultati in base ad un valore.	Eseguito.



TU - 43	Si controlla che un utente possa eseguire un Document (ovvero il suo DSLIS) a cui ha accesso, qualora avesse permessi sufficienti, senza che avvengano errori.	Eseguito.
TU - 44	Si controlla che un utente possa eseguire un Document appartenente ad una Collection (ovvero il suo DSLIS) a cui ha accesso, qualora avesse permessi sufficienti, senza che avvengano errori.	Eseguito.
TU - 45	Verifica che il metodo abbia inviato via mail, tramite una Action <code>sendEmail</code> , le informazioni renderizzate da un Document.	Eseguito.
TU - 46	Si verifica che un utente possa esportare i dati visualizzati in un Document, senza che si verifichino errori.	Eseguito.
TU - 47	Si controlla che un qualsivoglia utente possa ottenere il codice di un Document, ossia il DSLIS da cui esso ha avuto origine.	Eseguito.
TU - 48	Si controlla che un utente possa esportare il codice DSLIS generatore di un Document, senza che si verifichino errori.	Eseguito.
TU - 49	Si controlla che un utente possa importare il codice DSLIS generatore di un Document.	Eseguito.
TU - 50	Si verifica che un utente possa visualizzare una lista contenente tutte le Dashboard a cui ha accesso.	Eseguito.
TU - 51	Si verifica che un utente possa creare una nuova Dashboard, senza che si presentino errori.	Eseguito.
TU - 52	Si controlla che un utente possa modificare una Dashboard a cui ha accesso, qualora avesse permessi sufficienti.	Eseguito.
TU - 53	Si controlla che un utente possa eliminare una Dashboard a cui ha accesso, qualora avesse permessi sufficienti.	Eseguito.
TU - 54	Si controlla che un qualsivoglia utente possa ottenere il codice di una Dashboard, ossia il DSLIS da cui essa ha avuto origine.	Eseguito.
TU - 55	Si verifica la possibilità da parte di un utente di ricercare un insieme di Dashboard, filtrando i risultati in base ad un valore.	Eseguito.
TU - 56	Si controlla che un utente possa eseguire una Dashboard (ovvero il suo DSLIS) a cui ha accesso, qualora avesse permessi sufficienti, senza che avvengano errori.	Eseguito.



TU - 57	Si controlla che un utente possa esportare il codice DSLIS generatore di una Dashboard, senza che si verifichino errori.	Eseguito.
TU - 58	Si controlla che un utente possa importare il codice DSLIS generatore di una Dashboard.	Eseguito.
TU - 59	Si verifica che un utente possa visualizzare una lista contenente tutte le Cell a cui ha accesso.	Eseguito.
TU - 60	Si verifica che un utente possa creare una nuova Cell, senza che si presentino errori.	Eseguito.
TU - 61	Si controlla che un utente possa modificare una Cell a cui ha accesso, qualora avesse permessi sufficienti.	Eseguito.
TU - 62	Si controlla che un utente possa eliminare una Cell a cui ha accesso, qualora avesse permessi sufficienti.	Eseguito.
TU - 63	Si verifica la possibilità da parte di un utente di ricercare un insieme di Cell, filtrando i risultati in base ad un valore.	Eseguito.
TU - 64	Si controlla che un utente possa eseguire una Cell (ovvero il suo DSLIS) a cui ha accesso, qualora avesse permessi sufficienti, senza che avvengano errori.	Eseguito.
TU - 65	Si controlla che un qualsivoglia utente possa ottenere il codice di una Cell, ossia il DSLIS da cui essa ha avuto origine.	Eseguito.
TU - 66	Si controlla che un utente possa esportare il codice DSLIS generatore di una Cell, senza che si verifichino errori.	Eseguito.
TU - 67	Si controlla che un utente possa importare il codice DSLIS generatore di una Cell.	Eseguito.
TU - 68	Verifica che, all'atto dell'esportazione del codice di un DSLIS corretto, venga effettuata una conversione ugualmente corretta di esso nel formato JSON.	Eseguito.
TU - 69	Verifica che, all'atto dell'esportazione del codice di un DSLIS corretto, venga effettuata una conversione ugualmente corretta di esso nel formato CSV.	Eseguito.
TU - 70	Si verifica che funzioni correttamente la comunicazione tra front-end e back-end tramite i middleware.	Eseguito.
TU - 71	Si verifica che funzioni correttamente il controllo delle richieste fatte dal client al server, contenenti url errati.	Eseguito.



TU - 72	Si verifica che funzioni correttamente il controllo delle richieste fatte dal client al server, nel momento in cui quest'ultimo non sia in grado di gestirle.	Eseguito.
TU - 73	Si controlla che il server venga inizializzato correttamente.	Eseguito.
TU - 74	Verifica che sia possibile inviare tramite HTTP una richiesta per la sostituzione dell'avatar di un utente, e che sia possibile ricevere dal server un messaggio di buona riuscita o d'errore.	Eseguito.
TU - 75	Verifica che sia possibile inviare tramite HTTP una richiesta per la modifica dei dati personali di un utente, e che sia possibile ricevere dal server un messaggio di buona riuscita o d'errore.	Eseguito.
TU - 76	Verifica che sia possibile inviare tramite HTTP una richiesta per la sostituzione della password di un utente, e che sia possibile ricevere dal server un messaggio di buona riuscita o d'errore.	Eseguito.
TU - 77	Verifica che sia possibile inviare tramite HTTP una richiesta per la visualizzazione degli utenti di un'azienda, oltre che dei loro dati.	Eseguito.
TU - 78	Verifica che sia possibile inviare tramite HTTP una richiesta per la visualizzazione dei dati di un utente.	Eseguito.
TU - 79	Verifica che sia possibile inviare tramite HTTP una richiesta per l'eliminazione di un utente, e che sia possibile ricevere dal server un messaggio di buona riuscita o d'errore.	Eseguito.
TU - 80	Verifica che sia possibile inviare tramite HTTP una richiesta per la ricerca di un insieme di utenti, aventi una determinata proprietà, e che sia possibile ricevere dal server una lista o un messaggio d'errore.	Eseguito.
TU - 81	Verifica che sia possibile inviare tramite HTTP una richiesta per per il sistema di recupero password, e che sia possibile ricevere dal server un messaggio di buona riuscita o d'errore.	Eseguito.
TU - 82	Verifica che sia possibile inviare tramite HTTP una richiesta per la modifica del ruolo di un utente, e che sia possibile ricevere dal server un messaggio di buona riuscita dell'operazione.	Eseguito.
TU - 83	Verifica che sia possibile inviare tramite HTTP una richiesta per l'invito di un potenziale utente del sistema MaaS (con relativo ruolo), e che sia possibile ricevere dal server un messaggio di buona riuscita o d'errore.	Eseguito.



TU - 84	Si verifica che, attraverso una richiesta HTTP, si possa visualizzare una lista contenente tutte le Collection a cui si ha accesso, e che sia possibile ricevere dal server un messaggio d'errore.	Eseguito.
TU - 85	Si verifica che, attraverso una richiesta HTTP, si possa creare una nuova Collection, e che sia possibile ricevere dal server un messaggio di buona riuscita o d'errore.	Eseguito.
TU - 86	Si controlla che, attraverso una richiesta HTTP, si possa modificare una Collection a cui si ha accesso, qualora si avessero permessi sufficienti, e che sia possibile ricevere dal server un messaggio di buona riuscita o d'errore.	Eseguito.
TU - 87	Si controlla che, attraverso una richiesta HTTP, si possa eliminare una Collection a cui si ha accesso, qualora si avessero permessi sufficienti, e che sia possibile ricevere dal server un messaggio di buona riuscita dell'operazione.	Eseguito.
TU - 88	Si controlla che, attraverso una richiesta HTTP, si possa ottenere il codice di una Collection, ossia il DSLIS da cui essa ha avuto origine, e che sia possibile ricevere dal server un messaggio di buona riuscita dell'operazione.	Eseguito.
TU - 89	Si verifica la possibilità, attraverso una richiesta HTTP, di ricercare un insieme di Collection, filtrando i risultati in base ad un valore, e che sia possibile ricevere dal server una lista o un messaggio d'errore.	Eseguito.
TU - 90	Si controlla che, attraverso una richiesta HTTP, si possa eseguire una Collection (ovvero il suo DSLIS) a cui ha accesso, qualora avesse permessi sufficienti, senza che avvengano errori, e che sia possibile ricevere dal server il risultato dell'operazione in formato JSON.	Eseguito.
TU - 91	Verifica che, attraverso una richiesta HTTP, il metodo abbia inviato via mail, tramite una Action <code>sendEmail</code> , le informazioni renderizzate da una Collection, e che sia possibile ricevere dal server un messaggio di buona riuscita dell'operazione.	Eseguito.
TU - 92	Si verifica che, attraverso una richiesta HTTP, si possano esportare i dati visualizzati in una Collection, e che sia possibile ricevere dal server un messaggio di buona riuscita dell'operazione.	Eseguito.
TU - 93	Si controlla che, attraverso una richiesta HTTP, si possa esportare il codice DSLIS generatore di una Collection, e che sia possibile ricevere dal server un messaggio di buona riuscita.	Eseguito.



TU - 94	Si controlla che, attraverso una richiesta HTTP, si possa importare il codice DSLIS generatore di una Collection, e che sia possibile ricevere dal server un messaggio di buona riuscita o d'errore.	Eseguito.
TU - 95	Si verifica che, attraverso una richiesta HTTP, si possa visualizzare una lista contenente tutti i Document a cui si ha accesso, e che sia possibile ricevere dal server un messaggio d'errore.	Eseguito.
TU - 96	Si verifica che, attraverso una richiesta HTTP, si possa creare un nuovo Document, e che sia possibile ricevere dal server un messaggio di buona riuscita o d'errore.	Eseguito.
TU - 97	Si controlla che, attraverso una richiesta HTTP, si possa modificare un Document a cui si ha accesso, qualora si avessero permessi sufficienti, e che sia possibile ricevere dal server un messaggio di buona riuscita o d'errore.	Eseguito.
TU - 98	Si controlla che, attraverso una richiesta HTTP, si possa eliminare un Document a cui si ha accesso, qualora si avessero permessi sufficienti, e che sia possibile ricevere dal server un messaggio di buona riuscita dell'operazione.	Eseguito.
TU - 99	Si controlla che, attraverso una richiesta HTTP, si possa ottenere il codice di un Document, ossia il DSLIS da cui essa ha avuto origine, e che sia possibile ricevere dal server un messaggio di buona riuscita dell'operazione.	Eseguito.
TU - 100	Si verifica la possibilità, attraverso una richiesta HTTP, di ricercare un insieme di Document, filtrando i risultati in base ad un valore, e che sia possibile ricevere dal server una lista o un messaggio d'errore.	Eseguito.
TU - 101	Si controlla che, attraverso una richiesta HTTP, si possa eseguire un Document (ovvero il suo DSLIS) a cui ha accesso, qualora avesse permessi sufficienti, senza che avvengano errori, e che sia possibile ricevere dal server il risultato dell'operazione in formato JSON.	Eseguito.
TU - 102	Verifica che, attraverso una richiesta HTTP, il metodo abbia inviato via mail, tramite una Action <code>sendEmail</code> , le informazioni renderizzate da una Document, e che sia possibile ricevere dal server un messaggio di buona riuscita dell'operazione.	Eseguito.
TU - 103	Si verifica che, attraverso una richiesta HTTP, si possano esportare i dati visualizzati in un Document, e che sia possibile ricevere dal server un messaggio di buona riuscita dell'operazione.	Eseguito.



TU - 104	Si controlla che, attraverso una richiesta HTTP, si possa esportare il codice DSLIS generatore di un Document, e che sia possibile ricevere dal server un messaggio di buona riuscita.	Eseguito.
TU - 105	Si controlla che, attraverso una richiesta HTTP, si possa importare il codice DSLIS generatore di un Document, e che sia possibile ricevere dal server un messaggio di buona riuscita o d'errore.	Eseguito.
TU - 106	Si verifica che, attraverso una richiesta HTTP, si possa visualizzare una lista contenente tutte le Dashboard a cui si ha accesso, e che sia possibile ricevere dal server un messaggio d'errore.	Eseguito.
TU - 107	Si verifica che, attraverso una richiesta HTTP, si possa creare una nuova Dashboard, e che sia possibile ricevere dal server un messaggio di buona riuscita o d'errore.	Eseguito.
TU - 108	Si controlla che, attraverso una richiesta HTTP, si possa modificare una Dashboard a cui si ha accesso, qualora si avessero permessi sufficienti, e che sia possibile ricevere dal server un messaggio di buona riuscita o d'errore.	Eseguito.
TU - 109	Si controlla che, attraverso una richiesta HTTP, si possa eliminare una Dashboard a cui si ha accesso, qualora si avessero permessi sufficienti, e che sia possibile ricevere dal server un messaggio di buona riuscita dell'operazione.	Eseguito.
TU - 110	Si verifica la possibilità, attraverso una richiesta HTTP, di ricercare un insieme di Dashboard, filtrando i risultati in base ad un valore, e che sia possibile ricevere dal server una lista o un messaggio d'errore.	Eseguito.
TU - 111	Si controlla che, attraverso una richiesta HTTP, si possa eseguire una Dashboard (ovvero il suo DSLIS) a cui ha accesso, qualora avesse permessi sufficienti, senza che avvengano errori, e che sia possibile ricevere dal server il risultato dell'operazione in formato JSON.	Eseguito.
TU - 112	Si controlla che, attraverso una richiesta HTTP, si possa esportare il codice DSLIS generatore di una Dashboard, e che sia possibile ricevere dal server un messaggio di buona riuscita.	Eseguito.
TU - 113	Si controlla che, attraverso una richiesta HTTP, si possa ottenere il codice di una Dashboard, ossia il DSLIS da cui essa ha avuto origine, e che sia possibile ricevere dal server un messaggio di buona riuscita dell'operazione.	Eseguito.



TU - 114	Si controlla che, attraverso una richiesta HTTP, si possa importare il codice DSLIS generatore di una Dashboard, e che sia possibile ricevere dal server un messaggio di buona riuscita o d'errore.	Eseguito.
TU - 115	Si verifica che, attraverso una richiesta HTTP, si possa visualizzare una lista contenente tutte le Cell a cui si ha accesso, e che sia possibile ricevere dal server un messaggio d'errore.	Eseguito.
TU - 116	Si verifica che, attraverso una richiesta HTTP, si possa creare una nuova Cell, e che sia possibile ricevere dal server un messaggio di buona riuscita o d'errore.	Eseguito.
TU - 117	Si controlla che, attraverso una richiesta HTTP, si possa modificare una Cell a cui si ha accesso, qualora si avessero permessi sufficienti, e che sia possibile ricevere dal server un messaggio di buona riuscita o d'errore.	Eseguito.
TU - 118	Si controlla che, attraverso una richiesta HTTP, si possa eliminare una Cell a cui si ha accesso, qualora si avessero permessi sufficienti, e che sia possibile ricevere dal server un messaggio di buona riuscita dell'operazione.	Eseguito.
TU - 119	Si verifica la possibilità, attraverso una richiesta HTTP, di ricercare un insieme di Cell, filtrando i risultati in base ad un valore, e che sia possibile ricevere dal server una lista o un messaggio d'errore.	Eseguito.
TU - 120	Si controlla che, attraverso una richiesta HTTP, si possa eseguire una Cell (ovvero il suo DSLIS) a cui ha accesso, qualora avesse permessi sufficienti, senza che avvengano errori, e che sia possibile ricevere dal server il risultato dell'operazione in formato JSON.	Eseguito.
TU - 121	Si controlla che, attraverso una richiesta HTTP, si possa esportare il codice DSLIS generatore di una Cell, e che sia possibile ricevere dal server un messaggio di buona riuscita.	Eseguito.
TU - 122	Si controlla che, attraverso una richiesta HTTP, si possa ottenere il codice di una Cell, ossia il DSLIS da cui essa ha avuto origine, e che sia possibile ricevere dal server un messaggio di buona riuscita dell'operazione.	Eseguito.
TU - 123	Si controlla che, attraverso una richiesta HTTP, si possa importare il codice DSLIS generatore di una Cell, e che sia possibile ricevere dal server un messaggio di buona riuscita o d'errore.	Eseguito.



TU - 124	Si verifica che, attraverso una richiesta HTTP, si possa creare un nuovo database esterno, e che sia possibile ricevere dal server un messaggio di buona riuscita o d'errore.	Eseguito.
TU - 125	Si controlla che, attraverso una richiesta HTTP, si possa eliminare un database esterno a cui si ha accesso, qualora si avessero permessi sufficienti, e che sia possibile ricevere dal server un messaggio di buona riuscita dell'operazione.	Eseguito.
TU - 126	Si verifica che, attraverso una richiesta HTTP, si possa visualizzare una lista contenente tutti i database esterni a cui si ha accesso, e che sia possibile ricevere dal server un messaggio d'errore.	Eseguito.
TU - 127	Si verifica la possibilità, attraverso una richiesta HTTP, di ricercare un insieme di database esterni, filtrando i risultati in base ad un valore, e che sia possibile ricevere dal server una lista o un messaggio d'errore.	Eseguito.
TU - 128	Si verifica che, attraverso una richiesta HTTP, si possano attribuire ad un utente permessi di accesso ad un database esterno, e che sia possibile ricevere dal server un messaggio di buona riuscita dell'operazione.	Eseguito.
TU - 129	Si verifica che, attraverso una richiesta HTTP, si possano negare ad un utente i permessi di accesso ad un database esterno, e che sia possibile ricevere dal server un messaggio di buona riuscita dell'operazione.	Eseguito.
TU - 130	Si verifica che, attraverso una richiesta HTTP, si possa visualizzare una lista contenente tutte le Company a cui si ha accesso, e che sia possibile ricevere dal server un messaggio d'errore.	Eseguito.
TU - 131	Si verifica che, attraverso una richiesta HTTP, si possa creare una nuova Company, e che sia possibile ricevere dal server un messaggio di buona riuscita o d'errore.	Eseguito.
TU - 132	Si verifica che, attraverso una richiesta HTTP, si possano visualizzare i dati di una determinata Company a cui si ha accesso, e che sia possibile ricevere dal server il risultato dell'operazione.	Eseguito.
TU - 133	Si controlla che, attraverso una richiesta HTTP, si possa modificare una Company a cui si ha accesso, qualora si avessero permessi sufficienti, e che sia possibile ricevere dal server un messaggio di buona riuscita o d'errore.	Eseguito.



TU - 134	Si controlla che, attraverso una richiesta HTTP, si possa eliminare una Company a cui si ha accesso, qualora si avessero permessi sufficienti, e che sia possibile ricevere dal server un messaggio di buona riuscita dell'operazione.	Eseguito.
TU - 135	Si verifica la possibilità, attraverso una richiesta HTTP, di ricercare un insieme di Company, filtrando i risultati in base ad un valore, e che sia possibile ricevere dal server una lista o un messaggio d'errore.	Eseguito.
TU - 136	Si controlla che, attraverso una richiesta HTTP, il Super Amministratore riesca ad impersonare un qualsiasi utente del sistema MaaS, e che sia possibile ricevere dal server un messaggio di buona riuscita dell'operazione.	Eseguito.
TU - 137	Si controlla che, attraverso una richiesta HTTP, un utente possa essere registrato al sistema MaaS, e che sia possibile ricevere dal server un messaggio di buona riuscita o d'errore.	Eseguito.
TU - 138	Si controlla che, attraverso una richiesta HTTP, un utente possa autenticarsi al sistema MaaS, e che sia possibile ricevere dal server un messaggio di buona riuscita o d'errore.	Eseguito.
TU - 139	Si controlla che, attraverso una richiesta HTTP, un Super Amministratore possa autenticarsi al sistema MaaS, e che sia possibile ricevere dal server un messaggio di buona riuscita o d'errore.	Eseguito.
TU - 140	Si controlla che, attraverso una richiesta HTTP, un utente possa disconnettersi dal sistema MaaS, e che sia possibile ricevere dal server un messaggio di buona riuscita dell'operazione.	Eseguito.
TU - 141	Si controlla che, attraverso una richiesta HTTP, un Super Amministratore possa disconnettersi dal sistema MaaS, e che sia possibile ricevere dal server un messaggio di buona riuscita dell'operazione.	Eseguito.
TU - 142	Si verifica che venga creata una Action di tipo EDIT_USER_AVATAR_RESPONSE e che venga inoltrata alla Store UserStore .	Eseguito.
TU - 143	Si verifica che venga creata una Action di tipo EDIT_USER_PERSONAL_DATA_RESPONSE e che venga inoltrata alla Store UserStore .	Eseguito.
TU - 144	Si verifica che venga creata una Action di tipo EDIT_USER_PASSWORD_RESPONSE e che venga inoltrata alla Store UserStore .	Eseguito.



TU - 145	Si verifica che venga creata una Action di tipo GET_USERS_RESPONSE e che venga inoltrata alla Store UserStore.	Eseguito.
TU - 146	Si verifica che venga creata una Action di tipo GET_USER_RESPONSE e che venga inoltrata alla Store UserStore.	Eseguito.
TU - 147	Si verifica che venga creata una Action di tipo DELETE_USER_RESPONSE e che venga inoltrata alla Store UserStore.	Eseguito.
TU - 148	Si verifica che venga creata una Action di tipo SEARCH_USER_RESPONSE e che venga inoltrata alla Store UserStore.	Eseguito.
TU - 149	Si verifica che venga creata una Action di tipo FORGOT_PASSWORD_RESPONSE e che venga inoltrata alla Store UserStore.	Eseguito.
TU - 150	Si verifica che venga creata una Action di tipo CHANGE_ROLE_RESPONSE e che venga inoltrata alla Store UserStore.	Eseguito.
TU - 151	Si verifica che venga creata una Action di tipo SEND_INVITE_RESPONSE e che venga inoltrata alla Store UserStore.	Eseguito.
TU - 152	Si verifica che venga creata una Action di tipo GET_COLLECTIONS_RESPONSE e che venga inoltrata alla Store CollectionStore.	Eseguito.
TU - 153	Si verifica che venga creata una Action di tipo CREATE_COLLECTION_RESPONSE e che venga inoltrata alla Store CollectionStore.	Eseguito.
TU - 154	Si verifica che venga creata una Action di tipo EDIT_COLLECTION_RESPONSE e che venga inoltrata alla Store CollectionStore.	Eseguito.
TU - 155	Si verifica che venga creata una Action di tipo DELETE_COLLECTION_RESPONSE e che venga inoltrata alla Store CollectionStore.	Eseguito.
TU - 156	Si verifica che venga creata una Action di tipo RETRIEVE_COLLECTION_DSLIS_RESPONSE e che venga inoltrata alla Store CollectionStore.	Eseguito.
TU - 157	Si verifica che venga creata una Action di tipo SEARCH_COLLECTION_RESPONSE e che venga inoltrata alla Store CollectionStore.	Eseguito.



TU - 158	Si verifica che venga creata una Action di tipo EXECUTE_COLLECTION_RESPONSE e che venga inoltrata alla Store <code>CollectionStore</code> .	Eseguito.
TU - 159	Si verifica che venga creata una Action di tipo SEND_EMAIL_RESPONSE e che venga inoltrata alla Store <code>CollectionStore</code> .	Eseguito.
TU - 160	Si verifica che venga creata una Action di tipo EXPORT_RESPONSE e che venga inoltrata alla Store <code>CollectionStore</code> .	Eseguito.
TU - 161	Si verifica che venga creata una Action di tipo EXPORT_DSLIS_RESPONSE e che venga inoltrata alla Store <code>CollectionStore</code> .	Eseguito.
TU - 162	Si verifica che venga creata una Action di tipo IMPORT_DSLIS_RESPONSE e che venga inoltrata alla Store <code>CollectionStore</code> .	Eseguito.
TU - 163	Si verifica che venga creata una Action di tipo GET_DOCUMENT_RESPONSE e che venga inoltrata alla Store <code>DocumentStore</code> .	Eseguito.
TU - 164	Si verifica che venga creata una Action di tipo CREATE_DOCUMENT_RESPONSE e che venga inoltrata alla Store <code>DocumentStore</code> .	Eseguito.
TU - 165	Si verifica che venga creata una Action di tipo EDIT_DOCUMENT_RESPONSE e che venga inoltrata alla Store <code>DocumentStore</code> .	Eseguito.
TU - 166	Si verifica che venga creata una Action di tipo DELETE_DOCUMENT_RESPONSE e che venga inoltrata alla Store <code>DocumentStore</code> .	Eseguito.
TU - 167	Si verifica che venga creata una Action di tipo SEARCH_DOCUMENT_RESPONSE e che venga inoltrata alla Store <code>DocumentStore</code> .	Eseguito.
TU - 168	Si verifica che venga creata una Action di tipo EXECUTE_DOCUMENT_RESPONSE e che venga inoltrata alla Store <code>DocumentStore</code> .	Eseguito.
TU - 169	Si verifica che venga creata una Action di tipo SEND_EMAIL_RESPONSE e che venga inoltrata alla Store <code>DocumentStore</code> .	Eseguito.
TU - 170	Si verifica che venga creata una Action di tipo EXPORT_DSLIS_RESPONSE e che venga inoltrata alla Store <code>DocumentStore</code> .	Eseguito.



TU - 171	Si verifica che venga creata una Action di tipo RETRIEVE_DOCUMENT_DSLIS_RESPONSE e che venga inoltrata alla Store DocumentStore.	Eseguito.
TU - 172	Si verifica che venga creata una Action di tipo EXPORT_RESPONSE e che venga inoltrata alla Store DocumentStore.	Eseguito.
TU - 173	Si verifica che venga creata una Action di tipo IMPORT_DSLIS_RESPONSE e che venga inoltrata alla Store DocumentStore.	Eseguito.
TU - 174	Si verifica che venga creata una Action di tipo GET_DASHBOARDS_RESPONSE e che venga inoltrata alla Store DashboardStore.	Eseguito.
TU - 175	Si verifica che venga creata una Action di tipo CREATE_DASHBOARD_RESPONSE e che venga inoltrata alla Store DashboardStore.	Eseguito.
TU - 176	Si verifica che venga creata una Action di tipo EDIT_DASHBOARD_RESPONSE e che venga inoltrata alla Store DashboardStore.	Eseguito.
TU - 177	Si verifica che venga creata una Action di tipo DELETE_DASHBOARD_RESPONSE e che venga inoltrata alla Store DashboardStore.	Eseguito.
TU - 178	Si verifica che venga creata una Action di tipo SEARCH_DASHBOARD_RESPONSE e che venga inoltrata alla Store DashboardStore.	Eseguito.
TU - 179	Si verifica che venga creata una Action di tipo EXECUTE_DASHBOARD_RESPONSE e che venga inoltrata alla Store DashboardStore.	Eseguito.
TU - 180	Si verifica che venga creata una Action di tipo EXPORT_DSLIS_RESPONSE e che venga inoltrata alla Store DashboardStore.	Eseguito.
TU - 181	Si verifica che venga creata una Action di tipo RETRIEVE_DASHBOARD_DSLIS_RESPONSE e che venga inoltrata alla Store DashboardStore.	Eseguito.
TU - 182	Si verifica che venga creata una Action di tipo IMPORT_DSLIS_RESPONSE e che venga inoltrata alla Store DashboardStore.	Eseguito.
TU - 183	Si verifica che venga creata una Action di tipo GET_CELL_RESPONSE e che venga inoltrata alla Store CellStore.	Eseguito.



TU - 184	Si verifica che venga creata una Action di tipo <code>CREATE_CELL_RESPONSE</code> e che venga inoltrata alla Store <code>CellStore</code> .	Eseguito.
TU - 185	Si verifica che venga creata una Action di tipo <code>EDIT_CELL_RESPONSE</code> e che venga inoltrata alla Store <code>CellStore</code> .	Eseguito.
TU - 186	Si verifica che venga creata una Action di tipo <code>DELETE_CELL_RESPONSE</code> e che venga inoltrata alla Store <code>CellStore</code> .	Eseguito.
TU - 187	Si verifica che venga creata una Action di tipo <code>SEARCH_CELL_RESPONSE</code> e che venga inoltrata alla Store <code>CellStore</code> .	Eseguito.
TU - 188	Si verifica che venga creata una Action di tipo <code>EXECUTE_CELL_RESPONSE</code> e che venga inoltrata alla Store <code>CellStore</code> .	Eseguito.
TU - 189	Si verifica che venga creata una Action di tipo <code>EXPORT_DSLIS_RESPONSE</code> e che venga inoltrata alla Store <code>CellStore</code> .	Eseguito.
TU - 190	Si verifica che venga creata una Action di tipo <code>RETRIEVE_CELL_DSLIS_RESPONSE</code> e che venga inoltrata alla Store <code>CellStore</code> .	Eseguito.
TU - 191	Si verifica che venga creata una Action di tipo <code>IMPORT_DSLIS_RESPONSE</code> e che venga inoltrata alla Store <code>CellStore</code> .	Eseguito.
TU - 192	Si verifica che venga creata una Action di tipo <code>CREATE_EXTERNAL_DATABASE_RESPONSE</code> e che venga inoltrata alla Store <code>ExternalDatabaseStore</code> .	Eseguito.
TU - 192	Si verifica che venga creata una Action di tipo <code>DELETE_EXTERNAL_DATABASE_RESPONSE</code> e che venga inoltrata alla Store <code>ExternalDatabaseStore</code> .	Eseguito.
TU - 194	Si verifica che venga creata una Action di tipo <code>GET_EXTERNAL_DATABASES_RESPONSE</code> e che venga inoltrata alla Store <code>ExternalDatabaseStore</code> .	Eseguito.
TU - 195	Si verifica che venga creata una Action di tipo <code>SEARCH_EXTERNAL_DATABASE_RESPONSE</code> e che venga inoltrata alla Store <code>ExternalDatabaseStore</code> .	Eseguito.
TU - 196	Si verifica che venga creata una Action di tipo <code>ALLOW_EXTERNAL_DATABASE_ACCESS_RESPONSE</code> e che venga inoltrata alla Store <code>ExternalDatabaseStore</code> .	Eseguito.



TU - 197	Si verifica che venga creata una Action di tipo DE-NY_EXTERNAL_DATABASE_ACCESS_RESPONSE e che venga inoltrata alla Store ExternalDatabaseStore.	Eseguito.
TU - 198	Si verifica che venga creata una Action di tipo GET_COMPANIES_RESPONSE e che venga inoltrata alla Store CompanyStore.	Eseguito.
TU - 199	Si verifica che venga creata una Action di tipo CREA-TE_COMPANY_RESPONSE e che venga inoltrata alla Store CompanyStore.	Eseguito.
TU - 200	Si verifica che venga creata una Action di tipo GET_COMPANY_RESPONSE e che venga inoltrata alla Store CompanyStore.	Eseguito.
TU - 201	Si verifica che venga creata una Action di tipo EDIT_COMPANY_RESPONSE e che venga inoltrata alla Store CompanyStore.	Eseguito.
TU - 202	Si verifica che venga creata una Action di tipo DELE-TE_COMPANY_RESPONSE e che venga inoltrata alla Store CompanyStore.	Eseguito.
TU - 203	Si verifica che venga creata una Action di tipo SEARCH_COMPANY_RESPONSE e che venga inoltrata alla Store CompanyStore.	Eseguito.
TU - 204	Si verifica che venga creata una Action di tipo CREATE_USER_RESPONSE e che venga inoltrata alla Store SessionStore.	Eseguito.
TU - 205	Si verifica che venga creata una Action di tipo LOGIN_USER_RESPONSE e che venga inoltrata alla Store SessionStore.	Eseguito.
TU - 206	Si verifica che venga creata una Action di tipo LOGIN_SUPER_AMMINISTRATORE_RESPONSE e che venga inoltrata alla Store SessionStore.	Eseguito.
TU - 207	Si verifica che venga creata una Action di tipo LOGOUT_USER_RESPONSE e che venga inoltrata alla Store SessionStore.	Eseguito.
TU - 208	Si verifica che venga creata una Action di tipo LO-GOUT_SUPER_AMMINISTRATORE_RESPONSE e che venga inoltrata alla Store SessionStore.	Eseguito.
TU - 209	Si verifica che il metodo faccia da interfaccia a tutti i metodi omonimi delle classi della view.	Eseguito.



TU - 210	Si verifica che il metodo restituisca un oggetto JavaScript rappresentante lo stato della corrispondente Store, senza che avvengano errori.	Eseguito.
TU - 211	Si verifica che si possa emettere un segnale indicante un cambiamento nella corrispondente Store, senza che avvengano errori.	Eseguito.
TU - 212	Si verifica che si possa aggiungere alla Store una callback in ascolto di un segnale di cambiamento specifico, senza che avvengano errori.	Eseguito.
TU - 213	Si verifica che si possa togliere dalla Store una callback in ascolto di un segnale di cambiamento specifico, senza che avvengano errori.	Eseguito.
TU - 214	Si verifica che il metodo restituisca un oggetto JavaScript rappresentante lo stato della Store per la memorizzazione degli osservatori (le view), senza che avvengano errori.	Eseguito.

Tabella A.4: Test di Unità

C Resoconto delle attività di verifica

C.1 Riassunto delle attività di verifica

C.1.1 Revisione dei Requisiti

L'attività di verifica svolta dai Verificatori è avvenuta come determinato dal *Piano di Progetto v4.0.0* al termine della stesura di ogni documento previsto. La verifica svolta sui documenti e sui processi è avvenuta seguendo le indicazioni delle *Norme di Progetto v4.0.0* e misurando le metriche indicate in 3.3.2.1.

C.1.2 Revisione di Progettazione

L'attività di verifica svolta dai *Verificatori* è avvenuta come determinato dal *Piano di Progetto v4.0.0* al termine della stesura di ogni documento previsto. La verifica svolta sui documenti è avvenuta seguendo le indicazioni delle *Norme di Progetto v4.0.0* e misurando le metriche indicate in 3.3.2.1. Successivamente si è proceduto con le misurazioni delle metriche relative ai documenti. Sono quindi state effettuate le misurazioni mediante le metriche sui processi per valutarne la bontà e fornire una base per la pianificazione dei cicli *PDCA_G*.

C.1.3 Revisione di Qualifica

L'attività di verifica svolta dai *Verificatori* è avvenuta come determinato dal *Piano di Progetto v4.0.0* al termine della stesura di ogni documento previsto. La verifica svolta sui documenti è avvenuta seguendo le indicazioni delle *Norme di Progetto v4.0.0* e misurando le metriche indicate in 3.3.2.1. Le anomalie evidenziate non incidono in modo determinante sulla consistenza del prodotto del processo di documentazione. Le metriche hanno contribuito al controllo sui processi permettendo di monitorare e misurare il loro andamento.

L'attività di verifica svolta dai *Verificatori* è avvenuta come determinato dal *Piano di Progetto v4.0.0* al termine degli incrementi di ogni documento. La verifica svolta sui documenti è avvenuta seguendo le indicazioni delle *Norme di Progetto v4.0.0* e misurando le metriche indicate in 3.3.2.1. La verifica ha coinvolto principalmente le correzioni segnalate dal proponente a seguito della revisione di qualifica.

C.2 Dettaglio delle verifiche tramite analisi

C.2.1 Analisi Preliminare

C.2.1.1 Documenti

Vengono qui riportati i valori dell'indice Gulpease per ogni documento durante l'analisi e relativo esito basato sui range stabiliti in 3.3.2.1.



Documento	Valore indice	Esito
<i>Analisi dei Requisiti v1.0.0</i>	54	superato
<i>Glossario v1.0.0</i>	51	superato
<i>Norme di Progetto v1.0.0</i>	55	superato
<i>Piano di Progetto v1.0.0</i>	60	superato
<i>Piano di Qualifica v1.0.0</i>	57	superato
<i>Studio di Fattibilità v1.0.0</i>	50	superato

Tabella A.5: Esiti verifica documenti in fase di Analisi

C.2.2 Progettazione Architeturale e di Dettaglio

C.2.2.1 Documenti

Vengono qui riportati i valori dell'indice Gulpease per ogni documento durante la progettazione architeturale e relativo esito basato sui range stabiliti in 3.3.2.1.1.

Documento	Valore indice	Esito
<i>Analisi dei Requisiti v3.0.0</i>	51	superato
<i>Glossario v3.0.0</i>	49	superato
<i>Norme di Progetto v3.0.0</i>	47	superato
<i>Piano di Progetto v3.0.0</i>	73	superato
<i>Piano di Qualifica v3.0.0</i>	50	superato
<i>Definizione di Prodotto v3.0.0</i>	58	superato

Tabella A.6: Esiti verifica documenti, Progettazione Architeturale

C.2.3 Codifica e Test

C.2.3.1 Documenti

Vengono qui riportati i valori dell'indice Gulpease per ogni documento durante la progettazione di dettaglio e codifica, e relativo esito basato sui range stabiliti in 3.3.2.1.1.

Documento	Valore indice	Esito
<i>Analisi dei Requisiti v4.0.0</i>	54	superato
<i>Norme di Progetto v4.0.0</i>	53	superato
<i>Piano di Progetto v4.0.0</i>	52	superato
<i>Piano di Qualifica v4.0.0</i>	51	superato
<i>Definizione di Prodotto v4.0.0</i>	42	superato
<i>Glossario v4.0.0</i>	49	superato

Tabella A.7: Esiti verifica documenti, Progettazione di Dettaglio e Codifica



C.2.3.2 Software

Vengono qui riportate le misure rilevate con le metriche sull'analisi statica e dinamica del codice. Per ogni metrica si riportano i valore calcolati mantenendo una separazione tra backend e frontend. Per una descrizione delle metriche si rimanda alla sezione 3.3.2.2.

Backend

Metrica	Valore medio	Valore massimo	Esito
Complessità ciclomatica	1.38	6	superato
Numero di metodi (NOM)	/	155	superato
Numero di parametri per metodo	1.68	5	superato
Halstead difficulty per function	3.42	16.90	superato
Halstead volume per function	100.76	1303.56	superato
Halstead effort per function	653.64	9778.01	superato
Maintainability index	77.05	56.57	superato
Statement Coverage	/	63.45%	non superato
Branch Coverage	/	52.8%	non superato

Tabella A.8: Esiti metriche software, Codifica e Test

Frontend

Metrica	Valore medio	Valore massimo	superato
Complessità ciclomatica	1.34	5	superato
Numero di metodi (NOM)	/	62	superato
Numero di parametri per metodo	1.39	2	superato
Halstead difficulty per function	3.46	21.36	superato
Halstead volume per function	102.23	671.55	superato
Halstead effort per function	572.79	14342.28	superato
Maintainability index	71.98	50.2	superato
Statement Coverage	/	66.2%	non superato
Statement Coverage	/	48.69%	non superato

Tabella A.9: Esiti metriche software, Codifica e Test

I valori che non raggiungono i parametri di accettazione rappresentano una mancanza che verrà colmata nel prossimo periodo di sviluppo.

C.3 Dettaglio dell'esito delle revisioni

Lo sviluppo di questo progetto didattico si basa sull'attraversamento di quattro revisioni presiedute dal committente. Tre delle quattro revisioni produrranno delle segnalazioni degli errori riscontrati da parte del committente, deve seguire un report di come sono state risolte in ogni documento.



C.3.1 Revisione dei Requisiti

Per la Revisione dei Requisiti le segnalazioni da parte del committente sono state corrette:

- *Norme di Progetto*: il documento non ha subito modifiche in quanto considerato "eccellente";
- *Analisi dei Requisiti*: le *Norme di Progetto v4.0.0* descrivono la modalità di consegna che è stata ben definita includendo la generazione dei nomi dei documenti con la relativa versione. Inoltre sono stati rivisti e ampliati tutti i requisiti e casi d'uso segnalati dal committente;
- *Piano di Progetto*: sono stati approfonditi i rischi evidenziando i riscontri. Inoltre sono state rinominate le "fasi" raggiungendo una denominazione più significativa sugli stati di avanzamento.
- *Piano di Qualifica*: sono stati spostati i contenuti discorsivi, relativi alla qualità, in appendice. Sono stati definiti i test di validazione, sistema e integrazione ottenendo così una migliore correlazione con quanto specificato nelle Norme di Progetto.
- *Glossario*: il documento non ha subito modifiche in quanto considerato positivo.

C.3.2 Revisione di Progettazione

Per la Revisione di Progettazione le segnalazioni da parte del committente sono state corrette:

- *Norme di Progetto*: il documento non ha subito modifiche in quanto considerato "buono";
- *Analisi dei Requisiti*: il documento non ha subito modifiche in quanto le correzioni per la Revisione di Progettazione sono state giudicate buone.
- *Piano di Progetto*: sono state aggiornate le sezioni "Analisi dei Rischi" e "Consuntivo". Nella prima sezione sono stati modificate le voci "riscontro" valutando l'efficacia di attuazione delle strategie di mitigazione e sono stati rivisitati alcuni rischi sulla base di quanto riscontrato nella fasi affrontate. Nella seconda sezione è stata posta maggiore attenzione all'emersione di miglioramenti dell'efficacia e dell'efficienza sulla base dei riscontri economici.
- *Piano di Qualifica*: sono stati definiti quantitativamente gli obiettivi di qualità mettendoli in relazione alle metriche scelte. Quest'ultime invece sono state spostate in una sezione di approfondimento della strategia. Sono stati definiti in modo migliore gli stati di avanzamento dei test ottenendo così una migliore correlazione con il modello V.
- *Glossario*: il documento non ha subito modifiche in quanto considerato positivo.
- *Definizione di Prodotto*: sono stati corretti i dettagli progettuali segnalati ed esplicitati i formati JSON utilizzati;
- *Piano di Progetto*: rivista la struttura in modo da presentare in modo più leggibile la parte di preventivo/consuntivo.