



## Piano di Qualifica

### Informazioni sul documento

<b>Versione</b>	4.0.0
<b>Redazione</b>	Alberto De Agostini Andrea Giacomo Baldan Francesco Agostini
<b>Verifica</b>	Michael Munaro
<b>Approvazione</b>	Davide Trevisan
<b>Uso</b>	Esterno
<b>Lista di Distribuzione</b>	ScalateKids Prof. Tullio Vardanega Prof. Riccardo Cardin



## Diario delle modifiche

Versione	Autore	Ruolo	Data	Descrizione
4.0.0	Davide Trevisan	Responsabile	2016-07-05	Approvazione Documento
3.3.0	Michael Munaro	Verificatore	2016-07-04	Verifica sezione B.4
3.2.1	Francesco Agostini	Analista	2016-07-03	Stesura metriche sezione B.4
3.2.0	Michael Munaro	Verificatore	2016-06-10	Verifica sezione test
3.1.2	Francesco Agostini	Analista	2016-06-9	Modifica sezione test
3.1.1	Alberto De Agostini	Analista	2016-06-9	Modifica sezione test
3.1.0	Michael Munaro	Verificatore	2016-06-05	Verifica sezione B4
3.0.1	Giacomo Vanin	Analista	2016-06-05	Stesura budget e schedule variances di mezzo periodo (sezione B4)
3.0.0	Francesco Agostini	Responsabile	2016-05-16	Approvazione documento
2.3.0	Michael Munaro	Verificatore	2016-05-15	Verifica incrementi
2.2.1	Andrea Giacomo Baldan	Analista	2016-05-14	Aggiunta tabella dei valori delle metriche
2.2.0	Michael Munaro	Verificatore	2016-05-03	Verifica sezioni B.3.1 e C3
2.1.2	Alberto De Agostini	Analista	2016-05-02	Aggiunta sezione test di unità (C3)
2.1.1	Alberto De Agostini	Analista	2016-05-01	Aggiunta sezione B.3.1 (BV e SV primo maggio)
2.1.0	Michael Munaro	Verificatore	2016-04-29	Verifica modifiche per correzioni RP suggerite
2.0.2	Francesco Agostini	Analista	2016-04-28	Rimozione parti sezione due per spostamento in Norme di Progetto
2.0.1	Davide Trevisan	Analista	2016-04-26	Incremento sezione metriche per i processi
2.0.0	Marco Boseggia	Responsabile	2016-04-10	Approvazione documento
1.3.0	Michael Munaro	Verificatore	2016-04-05	Verifica incremento consuntivo
1.2.1	Andrea Giacomo Baldan	Analista	2016-04-05	Incremento sezione consuntivo per le attività di progettazione
1.2.0	Davide Trevisan	Verificatore	2016-04-02	Verifica sezioni test di validazione e di integrazione
1.1.1	Alberto De Agostini	Progettista	2016-04-01	Stesura sezioni test di validazione e di integrazione
1.1.0	Marco Boseggia	Verificatore	2016-02-26	Verifica sezioni 2.2, 2.8.1 e sottosezioni
1.0.2	Francesco Agostini	Analista	2016-02-24	Stesura sezione 2.8.1 e sottosezioni
1.0.1	Francesco Agostini	Analista	2016-02-19	Ristesura sezione 2.2 in seguito alle correzioni del docente
1.0.0	Alberto De Agostini	Responsabile	2016-01-21	Approvazione documento



0.6.0	Andrea Giacomo Baldan	Verificatore	2016-01-21	Verifica sezione Resoconto attività di Verifica
0.5.2	Francesco Agostini	Analista	2016-01-21	Stesura sezione Resoconto attività di Verifica
0.5.1	Francesco Agostini	Analista	2016-01-20	Creazione scheletro sezione Resoconto attività di Verifica
0.5.0	Michael Munaro	Verificatore	2016-01-20	Verifica Pianificazione dei test
0.4.2	Marco Boseggia	Analista	2016-01-18	Integrazione sezione Pianificazione dei test
0.4.1	Giacomo Vanin	Analista	2016-01-17	Stesura sezione Pianificazione dei test
0.4.0	Michael Munaro	Verificatore	2016-01-16	Verifica Correzioni sezione Standard di Qualità
0.3.1	Giacomo Vanin	Analista	2016-01-16	Correzioni sezione Standard di Qualità
0.3.0	Andrea Giacomo Baldan	Verificatore	2016-01-12	Verifica sezione Standard di Qualità
0.2.1	Marco Boseggia	Analista	2016-01-09	Stesura sezione Standard di Qualità
0.2.0	Michael Munaro	Verificatore	2016-01-07	Verifica Correzione effettuata
0.1.1	Giacomo Vanin	Analista	2016-01-08	Correzione sezione Gestione amministrativa della revisione
0.1.0	Andrea Giacomo Baldan	Verificatore	2016-01-05	Verifica capitoli stesi in precedenza
0.0.3	Giacomo Vanin	Analista	2016-01-04	Stesura sezione Gestione amministrativa della revisione
0.0.2	Francesco Agostini	Analista	2016-01-02	Stesura sezione Visione generale della strategia di verifica
0.0.1	Andrea Giacomo Baldan	Amministratore	2015-12-16	Creazione scheletro del documento



## Indice

<b>1</b>	<b>Sommario</b>	<b>1</b>
1.1	Scopo del documento	1
1.2	Scopo del Prodotto	1
1.3	Glossario	1
1.4	Riferimenti	1
1.4.1	Normativi	1
1.4.2	Informativi	1
<b>2</b>	<b>Visione generale della strategia di verifica</b>	<b>3</b>
2.1	Definizione obiettivi	3
2.1.1	Qualità di processo	3
2.1.2	Qualità di prodotto	3
2.2	Procedure di controllo di qualità di processo	3
2.3	Procedure di controllo di qualità di prodotto	3
2.4	Organizzazione	4
2.5	Pianificazione strategica e temporale	4
2.6	Responsabilità	5
2.7	Tecniche di analisi	5
2.7.1	Analisi statica	5
2.7.1.1	Walkthrough	5
2.7.1.2	Inspection	5
2.7.2	Analisi dinamica	5
2.7.2.1	Test di unità	5
2.7.2.2	Test di integrazione	5
2.7.2.3	Test di sistema	6
2.7.2.4	Test di regressione	6
2.7.2.5	Test di accettazione	6
2.8	Metriche di valutazione	6
2.8.1	Metriche per i processi	6
2.8.1.1	Budget Variance (BV)	6
2.8.1.2	Schedule Variance (SV)	7
2.8.1.3	Soddisfacimento requisiti obbligatori	7
2.8.1.4	Richieste al secondo	7
2.8.2	Metriche per i documenti	7
2.8.2.1	Indice Gulpease	7
2.8.3	Metriche per il software	8
2.8.3.1	Linee di codice per linee di commento	8
2.8.3.2	Copertura del codice	8
2.8.3.3	Numero di attributi per classe	9
2.8.3.4	Numero di parametri per metodo	9
2.8.3.5	Complessità ciclomatica	9
2.8.3.6	Livelli di annidamento	9



<b>A Standard di qualità</b>	<b>11</b>
A.1 Standard ISO/IEC 15504 . . . . .	11
A.2 Ciclo di Deming . . . . .	12
A.3 Standard ISO/IEC 9126 . . . . .	13
<b>B Resoconto attività di verifica</b>	<b>16</b>
B.1 Attività di Analisi . . . . .	16
B.1.1 Documenti . . . . .	16
B.2 Attività di Progettazione . . . . .	16
B.2.1 Documenti . . . . .	16
B.3 Attività di Codifica . . . . .	16
B.3.1 Budget Variance e Schedule Variance . . . . .	16
B.3.2 Soddisfamento requisiti obbligatori . . . . .	17
B.3.3 Richieste al secondo . . . . .	17
B.3.4 Documenti . . . . .	17
B.3.5 Codice . . . . .	17
B.3.5.1 Linee di codice per linee di commento . . . . .	18
B.3.5.2 Copertura del codice . . . . .	18
B.3.5.3 Numero di attributi per classe . . . . .	18
B.3.5.4 Numero di parametri per metodo . . . . .	18
B.3.5.5 Complessità ciclomatica . . . . .	19
B.3.5.6 Livelli di annidamento . . . . .	19
B.4 Attività di Validazione . . . . .	19
B.4.1 Budget Variance e Schedule Variance . . . . .	19
B.4.2 Soddisfamento requisiti obbligatori . . . . .	19
B.4.3 Documenti . . . . .	20
B.4.4 Codice . . . . .	20
B.4.4.1 Linee di codice per linee di commento . . . . .	20
B.4.4.2 Copertura del codice . . . . .	21
B.4.4.3 Numero di attributi per classe . . . . .	21
B.4.4.4 Numero di parametri per metodo . . . . .	21
B.4.4.5 Complessità ciclomatica . . . . .	21
B.4.4.6 Livelli di annidamento . . . . .	22
<b>C Pianificazione dei test</b>	<b>23</b>
C.1 Test di sistema . . . . .	23
C.2 Test di integrazione . . . . .	27
C.2.1 Tracciamento componenti-test di integrazione . . . . .	29
C.3 test di unità . . . . .	30
C.3.1 descrizione . . . . .	30
C.4 Test di validazione . . . . .	37
C.4.1 Test TV1 . . . . .	38
C.4.2 Test TV2 . . . . .	38
C.4.3 Test TV3 . . . . .	38
C.4.4 Test TV4 . . . . .	38



C.4.5 Test TV5 . . . . .	39
C.4.6 Test TV6 . . . . .	39
C.4.7 Test TV7 . . . . .	39
C.4.8 Test TV8 . . . . .	39
C.4.9 Test TV9 . . . . .	40
C.4.10Test TV10 . . . . .	40
C.4.11Test TV11 . . . . .	40
C.4.12Test TV12 . . . . .	40
C.4.13Test TV13 . . . . .	41
C.4.14Test TV14 . . . . .	41
C.4.15Test TV15 . . . . .	41
C.4.16Test TV16 . . . . .	41
C.4.17Test TV17 . . . . .	42
C.4.18Test TV18 . . . . .	42
C.4.19Test TV19 . . . . .	42
C.4.20Test TV20 . . . . .	42
C.4.21Test TV21 . . . . .	43
C.4.22Test TV22 . . . . .	43



# 1 Sommario

## 1.1 Scopo del documento

Il seguente documento ha lo scopo di delineare le strategie che il gruppo *ScalateKids* ha deciso di adottare per garantire degli obiettivi qualitativi da applicare ai processi sfruttati per lo sviluppo del progetto **ActorBase**. Per raggiungere tali obiettivi è necessario svolgere un'attività<sub>G</sub> di verifica continua in modo da rilevare e correggere errori o malfunzionamenti minimizzando lo spreco di risorse.

## 1.2 Scopo del Prodotto

Implementazione di un database NoSQL<sub>G</sub> di tipo key-value<sub>G</sub> orientato alla gestione di grandi moli di dati utilizzando il modello ad attori<sub>G</sub> su JVM<sub>G</sub>, comprensivo di un *Domain Specific Language* (DSL<sub>G</sub>) da utilizzare da riga di comando per poter interagire con il database.

Il progetto dovrà essere pubblicato su *GitHub* sotto licenza *MIT*.

## 1.3 Glossario

Tutti i termini di carattere tecnico o fraintendibile e gli acronimi sono raccolti nel file [Glossario v3.0.0](#); ogni occorrenza di parole nel *Glossario* è indicata da una "G" in pedice.

## 1.4 Riferimenti

### 1.4.1 Normativi

- **Capitolato d'appalto C1:** *Actorbase: a NoSQL DB based on the Actor model*;  
<http://www.math.unipd.it/~tullio/IS-1/2015/Progetto/C1.pdf>
- **Norme di Progetto:** [Norme di Progetto v4.0.0](#).

### 1.4.2 Informativi

- **Piano di Progetto:** [Piano di Progetto v4.0.0](#)
- **Dispense fornite dall'insegnamento Ingegneria del Software mod. A:**  
<http://www.math.unipd.it/~tullio/IS-1/2015/>
- **Standard ISO<sub>G</sub>/IEC<sub>G</sub> 15504:**  
[https://en.wikipedia.org/wiki/ISO/IEC\\_15504](https://en.wikipedia.org/wiki/ISO/IEC_15504)
- **Standard ISO<sub>G</sub>/IEC<sub>G</sub> 9126:**  
[https://it.wikipedia.org/wiki/ISO/IEC\\_9126](https://it.wikipedia.org/wiki/ISO/IEC_9126)



- **Ciclo di Deming<sub>c</sub>:**  
<https://en.wikipedia.org/wiki/PDCA>





## 2 Visione generale della strategia di verifica

### 2.1 Definizione obiettivi

#### 2.1.1 Qualità di processo

Per perseguire la qualità del prodotto è necessario garantire la qualità dei processi attuati per la sua creazione. Per raggiungere questo scopo i *Verificatori* con approvazione del *Responsabile* hanno trovato utile adottare lo standard ISO<sub>G</sub>/IEC<sub>G</sub> 15504<sup>1</sup> denominato SPICE (Software Process Improvement and Capability dEtermination). Il ciclo di Deming<sub>G</sub><sup>2</sup> (noto anche come ciclo PDCA<sub>G</sub>) definisce un metodo di controllo costante sui processi, consentendone un miglioramento continuo. Questo miglioramento consente ai processi di aumentare il livello di maturità nella scala data dallo standard ISO<sub>G</sub>/IEC<sub>G</sub> 15504.

#### 2.1.2 Qualità di prodotto

I *Verificatori* con approvazione del *Responsabile* hanno deciso di seguire lo standard ISO<sub>G</sub>/IEC<sub>G</sub> 9126<sup>3</sup> che ha lo scopo di delineare delle metriche per la misurazione della qualità di prodotto. Questa è fondamentale per garantire che il prodotto software<sub>G</sub> finale funzioni correttamente e rispetti degli obiettivi di qualità prefissati.

### 2.2 Procedure di controllo di qualità di processo

Per applicare correttamente il PDCA<sub>G</sub> è necessario che le seguenti condizioni siano soddisfatte:

- Ci sia una pianificazione accurata dei processi;
- Le risorse siano divise in maniera chiara nella pianificazione;
- I processi siano controllati.

La pianificazione descritta nel [Piano di Progetto v4.0.0](#) è stata fatta cercando di soddisfare queste condizioni. Per avere una misurazione oggettiva e quantitativa della qualità su ogni processo sono state stabilite delle metriche di valutazione descritte nella sezione [2.8.1](#). Seguire le regole descritte nelle [Norme di Progetto v4.0.0](#) per l'attuazione di tali verifiche e misurazioni.

L'insieme di queste condizioni e metriche consente l'utilizzo del PDCA<sub>G</sub> garantendo un miglioramento continuo della qualità di ogni processo e, di conseguenza, del prodotto finale.

### 2.3 Procedure di controllo di qualità di prodotto

Il controllo di qualità del prodotto si suddividerà in tre parti:

- Attuazione di tecniche di analisi statica e dinamica descritte nella sezione [2.7](#);

---

<sup>1</sup>Per approfondimenti consultare la sezione [A.1](#).

<sup>2</sup>Per approfondimenti consultare la sezione [A.2](#).

<sup>3</sup>Per approfondimenti consultare la sezione [A.3](#).



- Verifica costante dell'output di ogni attività<sub>G</sub>, i cui risultati verranno riportati nell'appendice B;
- Validazione, ovvero la conferma che il prodotto risponda correttamente ai requisiti attesi.

Per avere una misurazione oggettiva e quantitativa della qualità su ogni prodotto sono state stabilite delle metriche di valutazione per documenti e software descritte nelle sezioni, rispettivamente, 2.8.2 e 2.8.3. Seguire le regole descritte nelle [Norme di Progetto v4.0.0](#) per l'attuazione di tali verifiche e misurazioni.

## 2.4 Organizzazione

Per garantire la qualità del prodotto finale verranno effettuate attività<sub>G</sub> di verifica su ogni processo attuato. I controlli su ogni processo avranno come obiettivo garantire la qualità sia del processo stesso sia dell'eventuale prodotto da esso ricavato.

Ogni fase descritta in [Piano di Progetto v4.0.0](#) necessita di attività<sub>G</sub> di verifica diverse a causa della differenza natura del loro output. Di seguito sono elencate le strategie di verifica usate per ciascuna fase:

- **Analisi:** durante questa fase verrà verificato che ogni documento creato sia conforme a quanto descritto nel documento [Norme di Progetto v4.0.0](#), inoltre sarà necessario controllare che ogni requisito sia associato a un caso d'uso e viceversa;
- **Progettazione:** durante questa fase l'attività<sub>G</sub> di verifica si concentrerà sul processo incrementale dei documenti prodotti in fase di Analisi e sui processi attuati per l'attività<sub>G</sub> di progettazione assicurandosi che ogni documento e ogni processo sia conforme a quanto descritto nelle [Norme di Progetto v4.0.0](#);
- **Codifica:** durante questa fase l'attività<sub>G</sub> di verifica si concentrerà sul processo incrementale dei documenti prodotti in fase di Analisi e sui processi attuati per l'attività<sub>G</sub> di codifica assicurandosi che ogni documento e ogni processo sia conforme a quanto descritto nelle [Norme di Progetto v4.0.0](#).
- **Verifica e Validazione:** durante questa fase l'attività<sub>G</sub> di verifica continuerà ad essere sul processo incrementale dei documenti prodotti nelle precedenti fasi. Inoltre sarà effettuata una corposa attività<sub>G</sub> di verifica per il collaudo del prodotto.

In ogni documento viene inoltre tenuto un diario delle modifiche per permettere di mantenere uno storico delle attività<sub>G</sub> svolte su di esso e delle responsabilità per ogni attività<sub>G</sub>.

## 2.5 Pianificazione strategica e temporale

L'attività<sub>G</sub> di redazione di ogni documento sarà sempre anticipata da uno studio sulla struttura e sui contenuti del documento interessato. In questo modo si riduce la possibilità di errori concettuali e il tempo effettivo di scrittura del documento viene ridotto.

Il processo di verifica deve essere ben organizzato e pianificato per rispettare le scadenze fissate nel [Piano di Progetto v4.0.0](#).

L'attività di verifica segue le metodologie descritte nelle [Norme di Progetto v4.0.0](#). Esse sono state pensate in maniera tale da consentire l'individuazione e la correzione degli errori il più presto possibile evitando che si diffondano.



## 2.6 Responsabilità

Per organizzare al meglio il lavoro e in particolare le attività<sub>c</sub> di verifica e validazione vengono assegnati questi compiti ai soli *Verificatori* e al *Responsabile*. Inoltre le attività<sub>c</sub> di verifica non potranno essere svolte da un membro che ha partecipato allo sviluppo di tale sezione per garantire assenza di conflitto di interessi.

## 2.7 Tecniche di analisi

### 2.7.1 Analisi statica

L'analisi statica è una tecnica atta a trovare errori e anomalie. Questa tecnica è applicabile sia alla documentazione sia al codice<sub>c</sub>.

#### 2.7.1.1 Walkthrough

Come descritto nelle [Norme di Progetto v4.0.0](#) verrà effettuata la tecnica walkthrough nelle fasi iniziali di ogni attività<sub>c</sub> per passare il prima possibile ad attività<sub>c</sub> di *Inspection* (paragrafo seguente).

#### 2.7.1.2 Inspection

Successivamente all'attività<sub>c</sub> di 2.7.1.1 Walkthrough verrà fatta attività<sub>c</sub> di inspection (seguendo *Norme di Progetto* (paragrafo precedente)) rispetto alla lista di controllo ottenuta in precedenza.

### 2.7.2 Analisi dinamica

L'analisi dinamica si applica solo al prodotto software e serve a trovare errori o difetti nell'implementazione. I test effettuati devono essere *ripetibili*; questa caratteristica è fondamentale poiché un test deve produrre sempre lo stesso risultato dato un input in un ambiente per trovare errori.

#### 2.7.2.1 Test di unità

Test per verificare ogni unità del software prodotto. Per unità si intende la più piccola quantità di software che è utile verificare singolarmente.

L'obiettivo di questi test è trovare ed eliminare possibili errori creati dai *programmatore*.

#### 2.7.2.2 Test di integrazione

Test per verificare che l'integrazione tra due o più unità software funzionino correttamente.

In mancanza di unità questi test possono essere svolti tramite la creazione di una o più componenti fittizie che simulano il comportamento delle unità con cui l'unità creata deve interfacciarsi.



### 2.7.2.3 Test di sistema

Test per verificare la totale copertura dei requisiti software emersi in fase di *Analisi dei Requisiti*. Questi test verificano la totalità del prodotto software al suo raggiungimento di una versione definitiva.

### 2.7.2.4 Test di regressione

Test da effettuare per verificare che una modifica del software non pregiudichi il funzionamento del sistema o di altre parti del prodotto prima correttamente funzionanti.

Questa attività<sub>G</sub> dev'essere svolta tenendo conto del tracciamento dei requisiti per individuare i giusti test di unità, di integrazione e di sistema che possono essere influenzati dalle modifiche effettuate.

### 2.7.2.5 Test di accettazione

Questo corrisponde al collaudo vero e proprio del software da svolgersi col *Proponente*.

## 2.8 Metriche di valutazione

Per rendere quantificabile il processo di verifica verranno usate delle metriche elencate sotto. Soltanto due tipi di range<sub>G</sub> saranno accettati:

- **Ottimale:** contiene i valori che si deve cercare di ottenere, valori al di fuori di questo range<sub>G</sub> dovranno essere verificati e, se necessario, discussi dal gruppo;
- **Accettazione:** valori per avere un'accettazione del prodotto.

### 2.8.1 Metriche per i processi

Per garantire un monitoraggio e un controllo dei processi è stato scelto di utilizzare un indice sui costi e un indice temporale.

#### 2.8.1.1 Budget Variance (BV)

Questo indice indica se alla data corrente si è in linea con i costi preventivati.

Per calcolare questo indice basta moltiplicare le ore di lavoro pianificate per i rispettivi costi e sottrarle alla moltiplicazione tra le ore preventivate per i rispettivi costi. Se  $BV_G \geq 0$  significa che le spese sostenute sono minori rispetto ai costi preventivati.

##### Range accettabili:

- Range<sub>G</sub> ottimale:  $\geq 0$ ;
- Range<sub>G</sub> di accettazione:  $\geq -600$ .



### 2.8.1.2 Schedule Variance (SV)

Questo indice indica se alla data corrente si è in anticipo o in ritardo rispetto alla pianificazione temporale. Per calcolare questo indice basta sottrarre le ore effettive alle ore pianificate. Il risultato indica una quantità di ore lavoro.

Se  $SV_G \geq 0$  significa che si è in anticipo rispetto a quanto pianificato.

**Range accettabili:**

- $Range_G$  ottimale:  $\geq 0$ ;
- $Range_G$  di accettazione:  $\geq -40$ .

### 2.8.1.3 Soddisfacimento requisiti obbligatori

Questo indice serve per valutare la percentuale di completezza di requisiti obbligatori soddisfatti. Questo indice si calcola dividendo i requisiti soddisfatti con quelli totali moltiplicando per cento.

**Range accettabili:**

- $range_G$  ottimale:  $= 100$ ;
- $range_G$  di accettazione:  $= 100$ .

### 2.8.1.4 Richieste al secondo

Questo indice serve per valutare il numero di richieste che il sistema riesce a soddisfare in un secondo.

**Range accettabili:**

- $range_G$  ottimale:  $> 30$ ;
- $range_G$  di accettazione:  $15-30$ .

## 2.8.2 Metriche per i documenti

Per la stesura dei documenti è stato scelto di utilizzare un *indice di leggibilità* per la lingua italiana.

### 2.8.2.1 Indice Gulpease

Questo indice indica il grado di leggibilità di un documento.

L'indice di Gulpease prende in considerazione principalmente tre variabili:

- Numero delle frasi;
- Numero delle lettere;
- Numero delle parole.

La formula applicata per calcolare il suddetto indice è:



$$89 + \frac{300 * (\text{numerodellefrasi}) - 10 * (\text{numerodellelettere})}{\text{numerodelleparole}} \quad (1)$$

Lo script applicato su un documento produrrà in output un valore intero compreso tra zero e cento, in particolare i significati sono i seguenti:

- Indice inferiore a 80 rappresenta un documento difficile da leggere per persone con una licenza di scuola elementare;
- Indice inferiore a 60 rappresenta un documento difficile da leggere per persone con una licenza di scuola media inferiore;
- Indice inferiore a 40 rappresenta un documento difficile da leggere per persone con una licenza di diploma superiore;

**Range accettabili:**

- Range<sub>G</sub> ottimale: 60 - 100;
- Range<sub>G</sub> di accettazione: 45 - 60.

### 2.8.3 Metriche per il software

Per perseguire una buona qualità software sono state pensate diverse metriche di valutazione descritte nei prossimi paragrafi. Tuttavia questa sezione è una sezione che potrà subire cambiamenti durante le prossime revisioni.

#### 2.8.3.1 Linee di codice per linee di commento

Applicato su un file produrrà un valore in output che rappresenta la percentuale di righe di commento rispetto alle linee di codice<sub>G</sub>.

**Range accettabili:**

- Range<sub>G</sub> ottimale: 30 - 100%;
- Range<sub>G</sub> di accettazione: 20 - 30%.

#### 2.8.3.2 Copertura del codice

La copertura del codice<sub>G</sub> indica il numero di istruzioni che vengono eseguite durante i test rispetto alla totalità delle istruzioni. Per avere una minor possibilità di presenza di errori nel codice<sub>G</sub> si dovrà avere una percentuale il più alta possibile di istruzioni testate.

Tuttavia metodi molto semplici che non necessitano testing andranno a influire negativamente in questo ambito.

**Range accettabili:**

- Range<sub>G</sub> ottimale: 60 - 100%;



- Range<sub>G</sub> di accettazione: 40 - 60%.

### 2.8.3.3 Numero di attributi per classe

Un numero troppo elevato di attributi in una classe potrebbe indicare il non soddisfacimento del principio di *single responsibility*<sub>G</sub>. Questo potrebbe significare la necessità di una divisione in più classi relazionate tra loro per dividere le diverse funzionalità o un possibile errore di progettazione.

**Range accettabili:**

- Range<sub>G</sub> ottimale: 3 - 6;
- Range<sub>G</sub> di accettazione: 0 - 12.

### 2.8.3.4 Numero di parametri per metodo

Un numero troppo elevato di parametri in un metodo potrebbe indicare il non soddisfacimento del principio di *single responsibility*<sub>G</sub>. Questo potrebbe significare la necessità di una divisione in sotto metodi per dividere le diverse funzionalità o un possibile errore di progettazione.

**Range accettabili:**

- Range ottimale: 0 - 4;
- Range di accettazione: 4 - 8.

### 2.8.3.5 Complessità ciclomatica

La complessità ciclomatica è usata per misurare la complessità di parti di codice<sub>G</sub> come funzioni, metodi, classi o moduli.

Questa è calcolata attraverso un grafo di controllo del flusso del programma.

La complessità ciclomatica di una sezione di codice<sub>G</sub> è il numero di cammini linearmente indipendenti attraverso il codice<sub>G</sub> sorgente. Per esempio, se il codice<sub>G</sub> sorgente non contiene punti decisionali come IF o cicli FOR, allora la complessità sarà 1, poiché esiste un solo cammino nel sorgente (e quindi nel grafo). Se il codice<sub>G</sub> ha un singolo IF contenente una singola condizione, allora ci saranno due cammini possibili: il primo se l'IF viene valutato a TRUE e un secondo se l'IF viene valutato a FALSE.

**Range accettabili:**

- Range<sub>G</sub> ottimale: 0 - 10;
- Range<sub>G</sub> di accettazione: 10 - 15.

### 2.8.3.6 Livelli di annidamento

Indica il numero di livelli di annidamento di funzioni e metodi. Un valore alto di questo indice implica un livello di complessità molto elevato, aumentando la probabilità di commettere errori.

**Range accettabili:**

- Range<sub>G</sub> ottimale: 1 - 4;



- Range<sub>G</sub> di accettazione 5 - 8.





## A Standard di qualità

### A.1 Standard ISO/IEC 15504

Lo standard ISO<sub>c</sub>/IEC<sub>c</sub> 15504 definisce un metodo per controllare continuamente la qualità di ogni processo. Questo metodo sfrutta un modello denominato SPY (Software Process Assessment and Improvement) per la valutazione di processo.

Questo standard fornisce i seguenti nove attributi di qualità di processo:

- **Process performance:** il processo riesce a trasformare input identificabili in output identificabili.;
- **Performance management:** Il processo viene attuato e controllato seguendo una pianificazione che garantisca il raggiungimento degli obiettivi;
- **Work product management:** Il processo viene attuato e controllato seguendo una pianificazione che garantisca risultati documentati e verificati;
- **Process definition:** Il processo viene attuato seguendo standard di processo per garantire i risultati;
- **Process resource:** Il processo ha risorse adeguate per la sua attuazione;
- **Process measurement:** I risultati raggiunti e le misure rilevate durante l'attuazione di un processo sono stati usati per assicurarsi che tale processo supporti il raggiungimento di specifici obiettivi;
- **Process control:** Ogni processo è controllato attraverso la raccolta, analisi ed utilizzo delle misure di prodotto e di processo rilevate, per migliorare, eventualmente, le sue modalità di attuazione;
- **Process change:** Le modifiche alla definizione, gestione e attuazione di ogni processo sono controllate;
- **Continuous improvement:** Le modifiche ad ogni processo sono identificate ed implementate per assicurare il continuo miglioramento nel raggiungimento degli obiettivi rilevanti per l'organizzazione.

Questi attributi vengono utilizzati per suddividere la maturità di un processo nei seguenti sei livelli:

- **Livello 0 - Incompleto:** Il processo non raggiunge i suoi obiettivi;
- **Livello 1 - Attuato:** Il processo raggiunge i suoi obiettivi ma non è controllato. Il raggiungimento di questo livello è dimostrato dal possesso dell'attributo di *Process Performance*;
- **Livello 2 - Gestito:** Il processo è attuato, pianificato, tracciato, verificato e migliorato sulla base di obiettivi ben definiti. Il raggiungimento di questo livello è dimostrato dal possesso dell'attributo di *Performance management* e *Work product management*;
- **Livello 3 - Definito:** Il processo è attuato, pianificato e controllato su procedure basate sui principi del software engineering. Il raggiungimento di questo livello è dimostrato dal possesso dell'attributo di *Process definition* e *Process resource*;
- **Livello 4 - Predicibile:** Il processo è stabilizzato ed è attuato all'interno di definiti limiti riguardo i risultati attesi, le performance e le risorse impiegate. Il raggiungimento di questo livello è dimostrato dal possesso dell'attributo di *Performance measurement* e *Process control*;

- **Livello 5 - Ottimizzante:** Il processo è predicibile ed in grado di adattarsi per raggiungere obiettivi specifici e rilevanti per la organizzazione. Il raggiungimento di questo livello è dimostrato dal possesso dell'attributo di *Process change* e *Continuous improvement*.

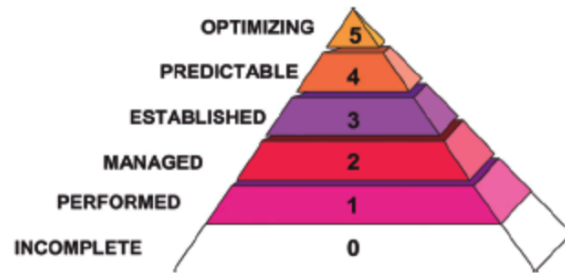


Figura 1: Visione piramidale dei sei livelli di maturità di processo.

Infine per stabilire in che livello si trovi un processo ci sono quattro livelli di possesso per ogni attributo:

- **N:** Non posseduto;
- **P:** Parzialmente posseduto;
- **L:** Largamente posseduto;
- **F:** Pienamente (Fully) posseduto.

## A.2 Ciclo di Deming

Il ciclo di Deming<sub>G</sub>, anche chiamato PDCA<sub>G</sub>, è un modello che consente il miglioramento continuo dei processi. Si suddivide nelle seguenti quattro attività<sub>G</sub>:

- **Plan:** attività<sub>G</sub> di pianificazione di attività<sub>G</sub>, risorse, scadenze e responsabilità;
- **Do:** attività<sub>G</sub> di esecuzione delle attività<sub>G</sub> pianificate;
- **Check:** attività<sub>G</sub> di verifica dei risultati dell'attività<sub>G</sub> di esecuzione, questi risultati verranno poi confrontati con i risultati pianificati nella fase di pianificazione;
- **Act:** attività<sub>G</sub> di messa in pratica del miglioramento dei processi sfruttando i risultati dell'attività<sub>G</sub> di verifica per trovare e modificare gli aspetti critici.

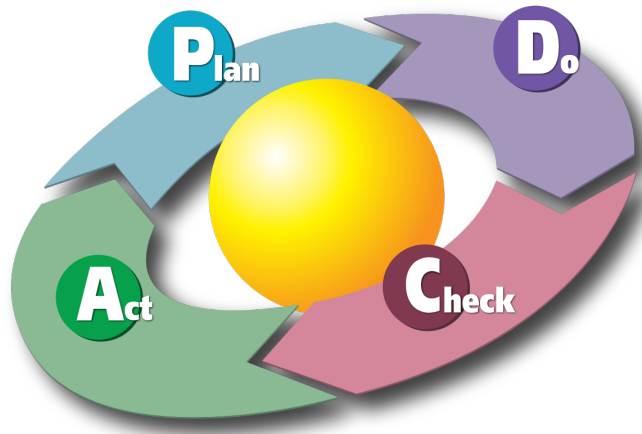


Figura 2: Ciclo di Deming

### A.3 Standard ISO/IEC 9126

Lo standard ISO<sub>c</sub>/IEC<sub>c</sub> 9126 definisce un modello per migliorare l'organizzazione e i processi, avendo come conseguenza il miglioramento della qualità del prodotto finale. I criteri qualitativi sono divisi in tre macro<sub>c</sub> aree:

- **Qualità interna:** è la qualità del prodotto software vista dall'interno. Si riferisce alla valutazione del codice<sub>c</sub> sorgente e dell'architettura prima che il prodotto sia ultimato;
- **Qualità esterna:** è la qualità del prodotto software vista dall'esterno. Si riferisce al risultato dei test condotti sul prodotto ultimato in un ambiente di prova;
- **Qualità in uso:** è la qualità del prodotto vista dall'utilizzatore finale del prodotto. Si riferisce alla qualità dopo il rilascio dell'applicazione.

Non potendo testare la qualità in uso nell'ambito di questo progetto si è scelto di concentrarsi sulla qualità interna ed esterna. Queste qualità nello standard ISO<sub>c</sub>/IEC<sub>c</sub> 9126 sono ulteriormente suddivise in sei categorie:

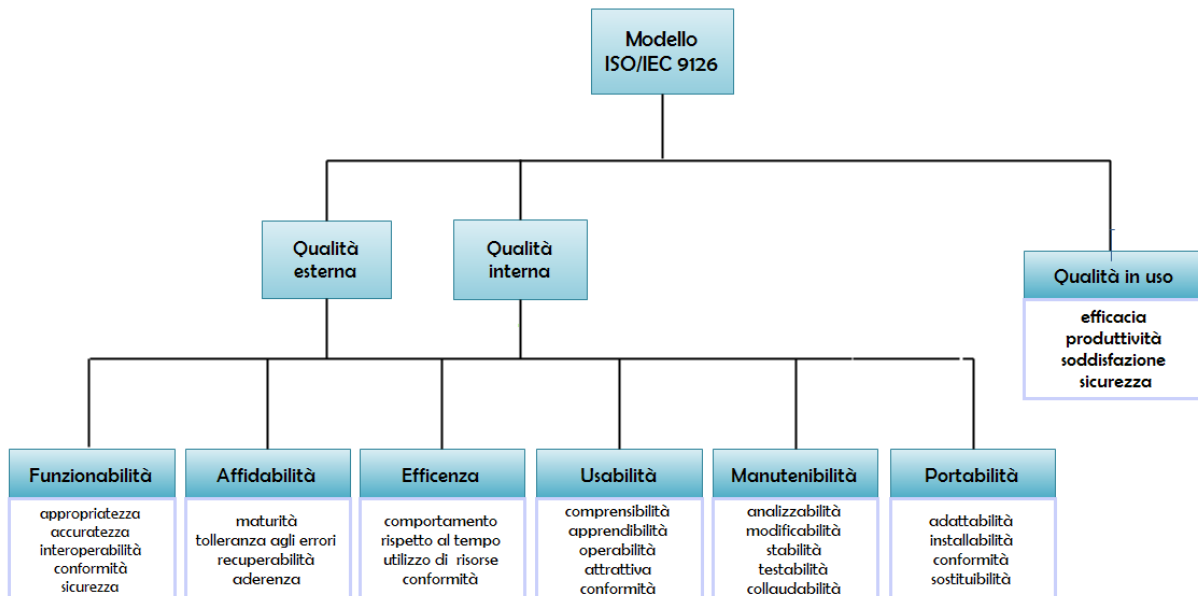


Figura 3: Modello dello standard ISO/IEC 9126

- **Funzionalità:** capacità del prodotto di fornire funzioni che soddisfino esigenze stabilite. Questa categoria si suddivide in:
  - **Appropriatezza:** capacità del prodotto di fornire un appropriato insieme di funzioni che rispondano a determinati requisiti;
  - **Accuratezza:** capacità del prodotto di fornire dei risultati corretti o con precisione concordata;
  - **Interoperabilità:** capacità del prodotto di interagire con uno o più sistemi esterni;
  - **Conformità:** capacità del prodotto di aderire a standard del settore di applicazione;
  - **Sicurezza:** capacità del prodotto di proteggere i dati, ad esempio negando l'accesso a utenti non autorizzati.
- **Affidabilità:** capacità del prodotto di mantenere uno specificato livello di prestazioni durante un determinato periodo. Questa categoria si suddivide in:
  - **Maturità:** capacità del prodotto di evitare che si verifichino risultati non corretti a causa di errori nel software;
  - **Tolleranza agli errori:** capacità del prodotto di mantenere uno specificato livello di prestazioni anche in presenza di errori nel software o di uso scorretto del prodotto;



- **Recuperabilità:** capacità del prodotto di ristabilire un livello adeguato di prestazioni in seguito a un malfunzionamento;
- **Aderenza:** capacità del prodotto di a standard riguardanti l'affidabilità.
- **Efficienza:** capacità del prodotto di fornire prestazioni appropriate in relazione alle risorse usate. Questa categoria si suddivide in:
  - **Comportamento rispetto al tempo:** capacità del prodotto di fornire i risultati in tempi adeguati;
  - **Utilizzo delle risorse:** capacità del prodotto di utilizzare le risorse in modo adeguato;
  - **Conformità:** capacità del prodotto di aderire a standard sull'efficienza.
- **Usabilità:** capacità del prodotto di essere capito e usato dall'utente. Questa categoria si suddivide in:
  - **Comprensibilità:** facilità di comprensione del prodotto;
  - **Apprendibilità:** capacità di ridurre l'impegno richiesto all'utente per imparare come usare correttamente il prodotto finale;
  - **Operabilità:** capacità del prodotto di fare in modo che gli utenti riescano a farne uso per i propri scopi;
  - **Attrattiva:** capacità del prodotto di essere piacevole per l'utente;
  - **Conformità:** capacità del prodotto di aderire a standard sull'usabilità.
- **Manutenibilità:** capacità del prodotto di essere modificato e migliorato. Questa categoria si suddivide in:
  - **Analizzabilità:** facilità di analisi del codice, per la localizzazione di errori;
  - **Modificabilità:** capacità del prodotto di permettere la sostituzione di componenti per implementare una determinata modifica;
  - **Stabilità:** capacità del prodotto di evitare errori derivanti da modifiche errate;
  - **Testabilità:** capacità del prodotto di essere facilmente testato rispetto alle modifiche apportate.
- **Portabilità:** capacità del prodotto di essere trasportato in un altro ambiente di lavoro. Questa categoria si suddivide in:
  - **Adattabilità:** capacità del prodotto di essere adattato per ambienti operativi diversi senza modifiche diverse da quelle fornite;
  - **Installabilità:** capacità del prodotto di essere installato in specifici ambienti;
  - **Conformità:** capacità del prodotto di aderire a standard sulla portabilità;
  - **Sostituibilità:** capacità del prodotto di essere utilizzato al posto di un altro software per svolgere gli stessi compiti.



## B Resoconto attività di verifica

### B.1 Attività di Analisi

#### B.1.1 Documenti

Nella tabella di seguito sono riportati i valori degli indici di Gulpease per ogni documento prodotto durante l'attività<sub>c</sub> di Analisi. L'esito della verifica dipenderà dal valore di questo indice come descritto nella sezione [2.8.2.1](#).

Documento	Valore indice	Esito
<i>Studio di Fattibilità v1.0.0</i>	54.18	Superato
<i>Norme di Progetto v1.0.0</i>	52.04	Superato
<i>Piano di Progetto v1.0.0</i>	54.87	Superato
<i>Piano di Qualifica v1.0.0</i>	52.82	Superato
<i>Analisi dei Requisiti v1.0.0</i>	68.06	Superato
<i>Glossario v1.0.0</i>	47.42	Superato

### B.2 Attività di Progettazione

#### B.2.1 Documenti

Nella tabella di seguito sono riportati i valori degli indici di Gulpease per ogni documento prodotto durante l'attività<sub>c</sub> di Progettazione. L'esito della verifica dipenderà dal valore di questo indice come descritto nella sezione [2.8.2.1](#).

Documento	Valore indice	Esito
<i>Specifica Tecnica v1.0.0</i>	68	Superato
<i>Norme di Progetto v2.0.0</i>	61	Superato
<i>Piano di Progetto v2.0.0</i>	64	Superato
<i>Piano di Qualifica v2.0.0</i>	64	Superato
<i>Analisi dei Requisiti v2.0.0</i>	88	Superato
<i>Glossario v2.0.0</i>	52	Superato

### B.3 Attività di Codifica

#### B.3.1 Budget Variance e Schedule Variance

Viene di seguito riportata una tabella contenente i valori per le metriche descritte in sezione [2.8.1](#). Questa tabella rappresenta un consuntivo calcolato in data primo maggio. Il consuntivo di periodo sarà disponibile nel documento [Piano di Progetto v4.0.0](#).



Ruolo	Ore	Costo
Responsabile	+3	+90
Analista	0	0
Amministratore	+3	+60
Progettista	-2	-44
Programmatore	-1	-15
Verificatore	+2	+30
<b>Totale</b>	<b>+5</b>	<b>+121</b>

### B.3.2 Soddisfacimento requisiti obbligatori

Attualmente la percentuale di requisiti obbligatori soddisfatti è di: **80%** Il valore della metrica è al di sotto del range di accettazione, questo è dovuto al fatto che il codice non è ancora ultimato. Il gruppo prevede che il range ottimale di 100% al termine del progetto.

### B.3.3 Richieste al secondo

Abbiamo effettuato diversi test per misurare questa metrica. Riportiamo una media dei valori ottenuti usando lo strumento GNU parallel, come descritto in [Norme di Progetto v4.0.0](#).

Il test prevede l'invio di duemila richieste da cinque processi paralleli per un totale di diecimila richieste.

La media ottenuta è di: **38**.

Questo valore rientra nel range ottimale.

### B.3.4 Documenti

Nella tabella di seguito sono riportati i valori degli indici di Gulpease per ogni documento prodotto durante l'attività<sub>c</sub> di codifica. L'esito della verifica dipenderà dal valore di questo indice come descritto nella sezione [2.8.2.1.t](#)

Documento	Valore indice	Esito
<i>Definizione di Prodotto v1.0.0</i>	75	Superato
<i>Specifica Tecnica v2.0.0</i>	74	Superato
<i>Norme di Progetto v3.0.0</i>	61	Superato
<i>Piano di Progetto v3.0.0</i>	64	Superato
<i>Piano di Qualifica v3.0.0</i>	64	Superato
<i>Analisi dei Requisiti v3.0.0</i>	89	Superato
<i>Glossario v3.0.0</i>	52	Superato
<i>Developer Manual v1.0.0</i>	66	Superato
<i>User Manual v1.0.0</i>	68	Superato



### B.3.5 Codice

Di seguito vengono riportati i risultati dei test di analisi statica e dinamica effettuati sul codice. Per ragioni di spazio e leggibilità si omettono le tabelle complete dei test. Per ogni test si riportano almeno il valore medio riscontrato ed eventuali risultati oltre il range di accettazione definiti nella sezione 2.8.3, oltre alla giustificazione della loro presenza.

#### B.3.5.1 Linee di codice per linee di commento

In questo paragrafo è possibile vedere il rapporto tra linee di codice e linee di commento, suddivisi per modulo:

- **CLI:**

Files	righe vuote	righe di commento	righe di codice	% commento su codice
28	133	834	419	66%

- **Driver:**

Files	righe vuote	righe di commento	righe di codice	% commento su codice
11	134	887	300	75%

- **ActorSystem:**

Files	righe vuote	righe di commento	righe di codice	% commento su codice
31	437	1817	1087	63%

Come visibile dalle tabelle tutti i valori soddisfano il range ottimale.

#### B.3.5.2 Copertura del codice

La percentuale di codice coperto da test attualmente è del: **72%**.  
Questo valore rientra nel range ottimale.

#### B.3.5.3 Numero di attributi per classe

Di seguito è riportato il numero medio di attributi per classe del codice di actorbase divisi per modulo:

- **CLI:** 1,04;
- **Driver:** 2,69;
- **Actorsystem:** 2,39.

Come si può notare tutti i valori rientrano nel range di accettazione.





#### B.3.5.4 Numero di parametri per metodo

Di seguito è riportato il numero medio di parametri per metodo del codice di actorbase suddiviso per moduli:

- **CLI**:0,1;
- **Driver**:0,63;
- **ActorSystem**:0,59.

Come si può notare tutti i valori rientrano nel range ottimale.

#### B.3.5.5 Complessità ciclomatica

Di seguito è riportata la media della complessità ciclomatica dei vari moduli:

- **CLI**:2,05;
- **Driver**:1,31;
- **Actorsystem**:2,93.

Come si può notare tutti i valori rientrano nel range ottimale.

#### B.3.5.6 Livelli di annidamento

In seguito è riportata la media dei livelli di annidamento dei vari moduli:

- **Cli**:1,78;
- **Driver**:1,56;
- **ActorSystem**:1,83.

Come si può notare tutti i valori rientrano nel range ottimale.

### B.4 Attività di Validazione

#### B.4.1 Budget Variance e Schedule Variance

Viene di seguito riportata una tabella contenente i valori per le metriche descritte in sezione [2.8.1](#).

Questa tabella rappresenta un consuntivo calcolato in data cinque giugno. Il consuntivo di periodo sarà disponibile nel documento [Piano di Progetto v4.0.0](#).



Ruolo	Ore	Costo
Responsabile	+2	+60
Analista	+2	+50
Amministratore	+2	+40
Progettista	+4	+88
Programmatore	-2	-30
Verificatore	+1	+15
<b>Totale</b>	<b>+9</b>	<b>+223</b>

#### B.4.2 Soddisfacimento requisiti obbligatori

Attualmente la percentuale di requisiti obbligatori soddisfatti è di: **100%**.

#### B.4.3 Documenti

Nella tabella di seguito sono riportati i valori degli indici di Gulpease per ogni documento prodotto durante l'attività<sub>c</sub> di validazione. L'esito della verifica dipenderà dal valore di questo indice come descritto nella sezione 2.8.2.1.

Documento	Valore indice	Esito
<i>Definizione di Prodotto v2.0.0</i>	72	Superato
<i>Specifica Tecnica v3.0.0</i>	74	Superato
<i>Norme di Progetto v4.0.0</i>	60	Superato
<i>Piano di Progetto v4.0.0</i>	65	Superato
<i>Piano di Qualifica v4.0.0</i>	63	Superato
<i>Analisi dei Requisiti v4.0.0</i>	88	Superato
<i>Glossario v4.0.0</i>	58	Superato
<i>Developer Manual v2.0.0</i>	74	Superato
<i>User Manual v2.0.0</i>	69	Superato

#### B.4.4 Codice

Di seguito vengono riportati i risultati dei test di analisi statica e dinamica effettuati sul codice. Per ragioni di spazio e leggibilità si omettono le tabelle complete dei test. Per ogni test si riportano almeno il valore medio riscontrato, e dove possibile dalle tecnologie utilizzate anche il massimo valore riscontrato ed eventuali risultati oltre il range di accettazione definiti nella sezione 2.8.3, oltre alla giustificazione della loro presenza. Si è cercato di essere il più precisi possibile.

##### B.4.4.1 Linee di codice per linee di commento

In questo paragrafo è possibile vedere il rapporto tra linee di codice e linee di commento, suddivisi per modulo:

- **CLI:**



Files	righe vuote	righe di commento	righe di codice	% commento su codice
29	124	1311	735	64%

- **Driver:**

Files	righe vuote	righe di commento	righe di codice	% commento su codice
13	200	1270	1198	51%

- **ActorSystem:**

Files	righe vuote	righe di commento	righe di codice	% commento su codice
29	465	1861	1715	52%

Inoltre tutti i rapporti commento su codice dei singoli file soddisfano il range ottimale.

#### B.4.4.2 Copertura del codice

Di seguito viene riportata la copertura dei test espressa in percentuale, affiancata dal rapporto coperte/totali. Si fa presente che la maggior parte del codice non coperto da test è composto da metodi banali (si pensi ai getter o setter). Nel caso del modulo driver è per questo motivo non si è tenuto conto nel totale riportato dei package data e exception, in quanto falsavano pesantemente la metrica per i motivi sopra citati.

Modulo	% delle classi	% dei metodi	% delle linee
<b>Cli</b>	68%(86/127)	56%(131/235)	51%(245/480)
<b>Driver</b>	62%(48/77)	51%(117/228)	59%(313/533)
<b>Actorsystem</b>	52%(105/202)	43%(220/510)	53%(498/945)

Il gruppo stima che il 10-15% del codice non banale non sia coperto da test. Tutti i valori rientrano comunque nel range di accettazione.

#### B.4.4.3 Numero di attributi per classe

Di seguito è riportato il numero medio di attributi per classe del codice di actorbase divisi per modulo. Da notare che nella media non sono state considerate eventuali classi prive di attributi.

- **CLI:** 3,67 massimo:10 (in CommandLoop); da notare che il dato è stato calcolando non considerando i vari comandi, che per ovvi motivi falsavano al ribasso il dato; rientrano comunque tutti nel range ottimale;
- **Driver:** 2,76 massimo:6 (nell' object Status). Non si è tenuto conto del package exceptions in quanto falsava al ribasso la media; rientrano comunque tutti nel range ottimale;
- **Actorsystem:** per ottenere risultati più significativi si dividono gli attori dai messaggi:
  - Attori: 2,93; massimo: 9 (la classe storekeeper);
  - Messaggi:1,79; massimo: 5 (il messaggio insertTo).

Come si può notare tutti i valori rientrano nel range di accettazione.



#### B.4.4.4 Numero di parametri per metodo

Di seguito è riportato il numero medio di parametri per metodo del codice di actorbase suddiviso per moduli:

- **CLI**:0,26; massimo:4 (nel metodo getDriver di DriverConnection in Connection.scala);
- **Driver**:1,18; massimo:5 (nello specifico il metodo apply in ActorbaseDriver.scala);
- **ActorSystem**:0,80; massimo:4 (il metodo encrypt in CryptoUtils.scala).

Come si può notare tutti i valori rientrano nel range ottimale, a parte un'unica eccezione.

#### B.4.4.5 Complessità ciclomatica

Di seguito è riportata la media della complessità ciclomatica dei vari moduli:

- **CLI**:3,30;
- **Driver**:3,63;
- **Actorsystem**:3,26.

#### B.4.4.6 Livelli di annidamento

In seguito è riportata la media dei livelli di annidamento dei vari moduli:

- **Cli**:1,75; massimo: 5;
- **Driver**:1,58; massimo:4;
- **ActorSystem**:1,86; massimo:7;

Come si può notare tutti i valori rientrano nel range di accettazione.



## C Pianificazione dei test

Di seguito sono descritti i test di sistema. In futuro questa sezione verrà aggiornata con i test di integrazione e di unità.

### C.1 Test di sistema

I test di sistema permettono di verificare il comportamento del sistema rispetto ai requisiti descritti nell'[Analisi dei requisiti v4.0.0](#).

Vengono presentate due tabelle, la prima contenente una sintesi dello stato dei test di sistema, la seconda descrive tutti i test dei requisiti ritenuti meritevoli di verifica.

Test totali di sistema	Test di sistema in stato OK	Percentuale di test di sistema superati
71	69	96%

Test	Descrizione	Stato	Requisito
TS.OBF1	Viene verificato che il sistema dovrà fornire la possibilità di distribuire il carico all'interno di un cluster scalabile	OK	OBF1
TS.OBF1.1	Viene verificato che il cluster <sub>G</sub> dovrà fornire la possibilità di connettersi ai singoli nodi <sub>G</sub> del cluster <sub>G</sub>	OK	OBF1.1
TS.OBF1.1.1	Viene verificato che il singolo nodo <sub>G</sub> dovrà fornire un server per ricevere comandi dall'esterno e inviare risposte	OK	OBF1.1.1
TS.OBF1.1.10.1	Viene verificato che il sistema ad attori dovrà permettere di autenticarsi dall'esterno mediante interazioni tra Clientactor <sub>G</sub> , Mainactor <sub>G</sub> , Storefinder <sub>G</sub> e AuthActor <sub>G</sub>	OK	OBF1.1.10.1
TS.OBF1.1.10.2.1	Viene verificato che il sistema ad attori dovrà permettere di creare collezioni <sub>G</sub> mediante l'interazione tra Clientactor <sub>G</sub> , Mainactor <sub>G</sub> , Storefinder <sub>G</sub> , Storekeeper <sub>G</sub> e AuthActor <sub>G</sub>	OK	OBF1.1.10.2.1
TS.OBF1.1.10.2.2	Viene verificato che il sistema ad attori dovrà permettere di visualizzare una lista delle collezioni <sub>G</sub> mediante l'interazione con Clientactor <sub>G</sub>	OK	OBF1.1.10.2.2
TS.OPF1.1.10.2.3	Viene verificato che il sistema ad attori dovrà permettere di modificare il nome delle collezioni <sub>G</sub> mediante l'interazione tra Clientactor <sub>G</sub> , Mainactor <sub>G</sub> , Storefinder <sub>G</sub> e AuthActor <sub>G</sub>	OK	OPF1.1.10.2.3
TS.OBF1.1.10.2.4	Viene verificato che il sistema ad attori dovrà permettere di cancellare collezioni <sub>G</sub> mediante l'interazione tra Clientactor <sub>G</sub> , Mainactor <sub>G</sub> , Storefinder <sub>G</sub> , Storekeeper <sub>G</sub> , AuthActor <sub>G</sub> e Ninja <sub>G</sub>	OK	OBF1.1.10.2.4
TS.OBF1.1.10.2.5	Viene verificato che il sistema ad attori dovrà permettere di aggiungere collaboratori a collezioni <sub>G</sub>	OK	OBF1.1.10.2.5



TS.OBF1.1.10.2.6	Viene verificato che il sistema ad attori dovrà permettere di rimuovere collaboratori da collezioni <sub>G</sub> mediante l'interazione tra Clientactor <sub>G</sub> , Mainactor <sub>G</sub> , Storefinder <sub>G</sub> , Storekeeper <sub>G</sub> , AuthActor <sub>G</sub> e Ninja <sub>G</sub>	OK	OBF1.1.10.2.6
TS.OBF1.1.10.2.7	Viene verificato che il sistema ad attori dovrà permettere di esportare collezioni <sub>G</sub> mediante l'interazione tra Clientactor <sub>G</sub> , Mainactor <sub>G</sub> , Storefinder <sub>G</sub> , Storekeeper <sub>G</sub> , AuthActor <sub>G</sub> e Ninja <sub>G</sub>	OK	OBF1.1.10.2.7
TS.OBF1.1.10.3.1	Viene verificato che il sistema ad attori dovrà permettere di inserire item <sub>G</sub> mediante l'interazione tra Clientactor <sub>G</sub> , Mainactor <sub>G</sub> , Storefinder <sub>G</sub> , Storekeeper <sub>G</sub> , Ninja <sub>G</sub> e Manager <sub>G</sub>	OK	OBF1.1.10.3.1
TS.OBF1.1.10.3.2	Viene verificato che il sistema ad attori dovrà permettere di creare una collezione <sub>G</sub> nel caso di inserimento item <sub>G</sub> a collezione <sub>G</sub> inesistente mediante interazioni tra Clientactor <sub>G</sub> , Mainactor <sub>G</sub> e AuthActor <sub>G</sub>	OK	OBF1.1.10.3.2
TS.OBF1.1.10.3.3	Viene verificato che il sistema ad attori dovrà inviare un messaggio d'errore verso l'esterno nel caso in cui la chiave dell'item <sub>G</sub> da inserire senza sovrascrittura sia già presente, mediante interazioni tra Clientactor <sub>G</sub> , Mainactor <sub>G</sub> , Storefinder <sub>G</sub> e Storekeeper <sub>G</sub>	OK	OBF1.1.10.3.3
TS.OBF1.1.10.3.4	Viene verificato che il sistema ad attori dovrà permettere di cancellare item <sub>G</sub> da collezioni <sub>G</sub> mediante interazioni tra Clientactor <sub>G</sub> , Main <sub>G</sub> , Storefinder <sub>G</sub> , Storekeeper <sub>G</sub> e Ninja <sub>G</sub>	OK	OBF1.1.10.3.4
TS.OBF1.1.10.3.5	Viene verificato che il sistema ad attori dovrà inviare un messaggio d'errore verso l'esterno nel caso in cui il nome della collezione <sub>G</sub> a cui inserire nuovi item <sub>G</sub> sia inesistente, mediante interazioni tra Clientactor <sub>G</sub> ed esterno	OK	OBF1.1.10.3.5
TS.OBF1.1.10.4	Viene verificato che il sistema ad attori dovrà permettere di ricercare item <sub>G</sub> tra collezioni <sub>G</sub> mediante interazioni tra Clientactor <sub>G</sub> , Main <sub>G</sub> , Storefinder <sub>G</sub> e Storekeeper <sub>G</sub>	OK	OBF1.1.10.4
TS.OBF1.1.10.5	Viene verificato che il sistema ad attori dovrà permettere di modificare la password utente di uno username <sub>G</sub> mediante interazioni tra Clientactor <sub>G</sub> , Main <sub>G</sub> , Storefinder <sub>G</sub> , AuthActor <sub>G</sub>	OK	OBF1.1.10.5
TS.OBF1.1.10.6	Viene verificato che il sistema ad attori dovrà permettere la disconnessione di client esterni mediante interazioni con Clientactor <sub>G</sub>	OK	OBF1.1.10.6
TS.OBF1.1.10.7.1	Viene verificato che il sistema ad attori dovrà permettere di aggiungere nuovi utenti all'interno del database <sub>G</sub> mediante interazioni tra Clientactor <sub>G</sub> , Main <sub>G</sub> , Storefinder <sub>G</sub> e AuthActor <sub>G</sub>	OK	OBF1.1.10.7.1
TS.OBF1.1.10.7.2	Viene verificato che il sistema ad attori dovrà permettere la rimozione di un utente dal database <sub>G</sub> mediante interazioni tra Clientactor <sub>G</sub> , Main <sub>G</sub> , Storefinder <sub>G</sub>	OK	OBF1.1.10.7.2



TS.OBF1.1.10.7.3	Viene verificato che il sistema ad attori dovrà permettere di effettuare un reset della password di un utente mediante interazioni tra Clientactor <sub>G</sub> , Main <sub>G</sub> , Storefinder <sub>G</sub> , AuthActor <sub>G</sub>	OK	OBF1.1.10.7.3
TS.OBF2	Viene verificato che il sistema dovrà fornire una console <sub>G</sub> per inviare comandi al server (CLI)	OK	OBF2
TS.OBF2.1	Viene verificato che il sistema dovrà fornire un Domain Specific Language per comunicare con il database <sub>G</sub>	OK	OBF2.1
TS.OBF2.1.1	Viene verificato che il DSL <sub>G</sub> dovrà fornire un comando per effettuare l'autenticazione utilizzabile mediante CLI	OK	OBF2.1.1
TS.OBF2.1.2.1	Viene verificato che il DSL <sub>G</sub> dovrà fornire un comando per la creazione di una nuova collezione <sub>G</sub> utilizzabile mediante CLI	OK	OBF2.1.2.1
TS.OBF2.1.2.2	Viene verificato che il DSL <sub>G</sub> dovrà fornire un comando per elencare i nomi delle collezioni <sub>G</sub> presenti all'interno del database <sub>G</sub> utilizzabile mediante CLI	OK	OBF2.1.2.2
TS.OBF2.1.2.3	Viene verificato che il DSL <sub>G</sub> dovrà fornire un comando per cancellare una o più collezioni <sub>G</sub> utilizzabile mediante CLI	OK	OBF2.1.2.3
TS.OPF2.1.2.4	Viene verificato che il DSL <sub>G</sub> dovrà fornire un comando per modificare il nome delle collezioni utilizzabile mediante CLI	OK	OPF2.1.2.4
TS.OBF2.1.2.5	Viene verificato che il DSL <sub>G</sub> dovrà fornire un comando per aggiungere collaboratori <sub>G</sub> ad una collezione <sub>G</sub> del sistema utilizzabile mediante CLI	Failing	OBF2.1.2.5
TS.OBF2.1.2.6	Viene verificato che il DSL <sub>G</sub> dovrà fornire un comando per rimuovere un collaboratore <sub>G</sub> da una collezione <sub>G</sub> del sistema utilizzabile mediante CLI	Failing	OBF2.1.2.6
TS.OBF2.1.2.7	Viene verificato che il DSL <sub>G</sub> dovrà fornire un comando per l'esportazione di collezioni <sub>G</sub> su file JSON <sub>G</sub> utilizzabile mediante CLI	OK	OBF2.1.2.7
TS.OBF2.1.3.1	Viene verificato che il DSL <sub>G</sub> dovrà fornire un comando per inserire un nuovo item <sub>G</sub> utilizzabile mediante CLI	OK	OBF2.1.3.1
TS.OBF2.1.3.1.1	Viene verificato che il DSL <sub>G</sub> dovrà fornire un comando per inserire un item <sub>G</sub> specificandone gli attributi da CLI	OK	OBF2.1.3.1.1
TS.DEF2.1.3.1.2	Viene verificato che il DSL <sub>G</sub> dovrà fornire un comando per inserire nuovi item <sub>G</sub> da file JSON <sub>G</sub> utilizzabile mediante CLI	OK	DEF2.1.3.1.2
TS.OBF2.1.3.6	Viene verificato che il DSL <sub>G</sub> dovrà fornire un comando per cancellare un item <sub>G</sub> specificandone gli attributi da CLI	OK	OBF2.1.3.6
TS.DEF2.1.4	Viene verificato che il DSL <sub>G</sub> dovrà fornire un comando per richiedere aiuto generale sul sistema utilizzabile mediante CLI	OK	DEF2.1.4
TS.DEF2.1.5	Viene verificato che il DSL <sub>G</sub> dovrà fornire un comando per richiedere aiuto sull'utilizzo di uno specifico comando del sistema utilizzabile mediante CLI	OK	DEF2.1.5
TS.OBF2.1.6	Viene verificato che il DSL <sub>G</sub> dovrà fornire un comando per permettere di effettuare una ricerca su una o più collezioni <sub>G</sub> all'interno del sistema; utilizzabile mediante CLI	OK	OBF2.1.6



TS.OBF2.1.7	Viene verificato che il DSL <sub>G</sub> dovrà fornire un comando per la modifica della propria password utilizzabile mediante CLI	OK	OBF2.1.7
TS.OBF2.1.8	Viene verificato che il DSL <sub>G</sub> dovrà fornire un comando per effettuare il logout dal sistema utilizzabile mediante CLI	OK	OBF2.1.8
TS.OBF2.1.9.1	Viene verificato che il DSL <sub>G</sub> dovrà fornire un comando per l'aggiunta di un nuovo utente all'interno del sistema utilizzabile mediante CLI	OK	OBF2.1.9.1
TS.OBF2.1.9.2	Viene verificato che il DSL <sub>G</sub> dovrà fornire un comando per la rimozione di un utente dal sistema	OK	OBF2.1.9.2
TS.OBF2.1.9.3	Viene verificato che il DSL <sub>G</sub> dovrà fornire un comando per effettuare il reset della password di un utente alla password standard	OK	OBF2.1.9.3
TS.DEF3	Viene verificato che dovrà essere fornito un driver <sub>G</sub> Scala <sub>G</sub> per interfacciarsi con il database <sub>G</sub>	OK	DEF3
TS.DEF3.1	Viene verificato che il driver <sub>G</sub> dovrà permettere di effettuare l'autenticazione all'interno del sistema.	OK	DEF3.1
TS.DEF3.2.1	Viene verificato che il driver <sub>G</sub> dovrà permettere un la creazione di una nuova collezione <sub>G</sub>	OK	DEF3.2.1
TS.DEF3.2.2	Viene verificato che il driver <sub>G</sub> dovrà permettere di elencare i nomi delle collezioni <sub>G</sub> presenti all'interno del database <sub>G</sub>	OK	DEF3.2.2
TS.DEF3.2.3	Viene verificato che il driver <sub>G</sub> dovrà permettere di cancellare una o più collezioni <sub>G</sub>	OK	DEF3.2.3
TS.OPF3.2.4	Viene verificato che il driver <sub>G</sub> dovrà permettere di modificare il nome delle collezioni	OK	OPF3.2.4
TS.DEF3.2.5	Viene verificato che il driver <sub>G</sub> dovrà permettere di aggiungere collaboratori <sub>G</sub> ad una collezione <sub>G</sub> del sistema	OK	DEF3.2.5
TS.DEF3.2.6	Viene verificato che il driver <sub>G</sub> dovrà permettere di rimuovere un collaboratore <sub>G</sub> da una collezione <sub>G</sub> del sistema	OK	DEF3.2.6
TS.DEF3.2.7	Viene verificato che il driver <sub>G</sub> dovrà permettere di esportare collezioni <sub>G</sub> su file JSON <sub>G</sub>	OK	DEF3.2.7
TS.DEF3.3.1	Viene verificato che il driver <sub>G</sub> dovrà permettere di inserire un nuovo item <sub>G</sub>	OK	DEF3.3.1
TS.DEF3.3.1.1	Viene verificato che il driver <sub>G</sub> dovrà permettere di inserire un nuovo item <sub>G</sub> specificandone gli attributi	OK	DEF3.3.1.1
TS.DEF3.3.1.2	Viene verificato che il driver <sub>G</sub> dovrà permettere di inserire nuovi item <sub>G</sub> da file JSON <sub>G</sub>	OK	DEF3.3.1.2
TS.DEF3.3.2	Viene verificato che il driver <sub>G</sub> dovrà permettere di cancellare uno o più item <sub>G</sub> dal sistema	OK	DEF3.3.2
TS.DEF3.4	Viene verificato che il driver <sub>G</sub> dovrà permettere di effettuare ricerche su una o più collezioni <sub>G</sub> all'interno del sistema	OK	DEF3.4
TS.DEF3.6.1	Viene verificato che il driver <sub>G</sub> dovrà permettere a utenti amministratori di aggiungere un nuovo utente al sistema	OK	DEF3.6.1





TS.DEF3.6.2	Viene verificato che il driver <sub>G</sub> dovrà permettere a utenti amministratori di rimuovere un utente dal sistema	OK	DEF3.6.2
TS.DEF3.6.3	Viene verificato che il driver <sub>G</sub> dovrà permettere a utenti amministratori di effettuare il reset della password ad un utente all'interno del sistema	OK	DEF3.6.3
TS.DEF3.7	Viene verificato che il driver <sub>G</sub> dovrà permettere di modificare la propria password	OK	DEF3.7
TS.DEF3.8	Viene verificato che il Driver <sub>G</sub> dovrà strutturare i dati in output in maniera navigabile	OK	DEF3.8
TS.OBV4	Viene verificato che il sistema dovrà funzionare in ambienti Ubuntu 14.04 o superiore	OK	OBV4
TS.OBV5	Viene verificato che il sistema dovrà funzionare su JVM <sub>G</sub> versione 8	OK	OBV5
TS.OBV6	Viene verificato che il sistema dovrà utilizzare la libreria Akka versione 2.4.2	OK	OBV6
TS.OBV7	Viene verificato che il sistema dovrà essere implementato utilizzando il linguaggio Scala versione 2.11	OK	OBV7
TS.DEV8	Viene verificato che il sistema dovrà funzionare in ambienti Windows 10 o superiori	OK	DEV8
TS.DEV9	Viene verificato che il sistema dovrà funzionare su sistemi OSX El Capitan o superiori	OK	DEV9
TS.OBQ10	Viene verificato che dovrà essere fornito un manuale utente redatto in lingua inglese	OK	OBQ10
TS.OBQ11	Viene verificato che dovranno essere rispettate tutte le norme e le metriche sulla stesura del codice come riportato nelle <i>Norme di Progetto</i>	OK	OBQ11
TS.OBQ12	Viene verificato che dovrà essere prodotta documentazione in Scaladoc del prodotto	OK	OBQ12

## C.2 Test di integrazione

I test di integrazione permettono di verificare la corretta integrazione ed il corretto flusso dei dati all'interno del sistema, sia relazioni tra package<sub>G</sub> sia il funzionamento di ogni singolo package<sub>G</sub> di ogni componente.

L'approccio utilizzato è di tipo bottom-up<sub>G</sub>, in questo modo è possibile integrare prima le parti con minore dipendenza funzionale e, nel caso di inserimento di parti difettose, è possibile retrocedere ad uno stato testato e sicuro in maniera semplice. Il diagramma seguente non rispetta il formalismo UML<sub>G</sub> 2.0 ed è utilizzato per semplificare l'illustrazione della strategia di integrazione:

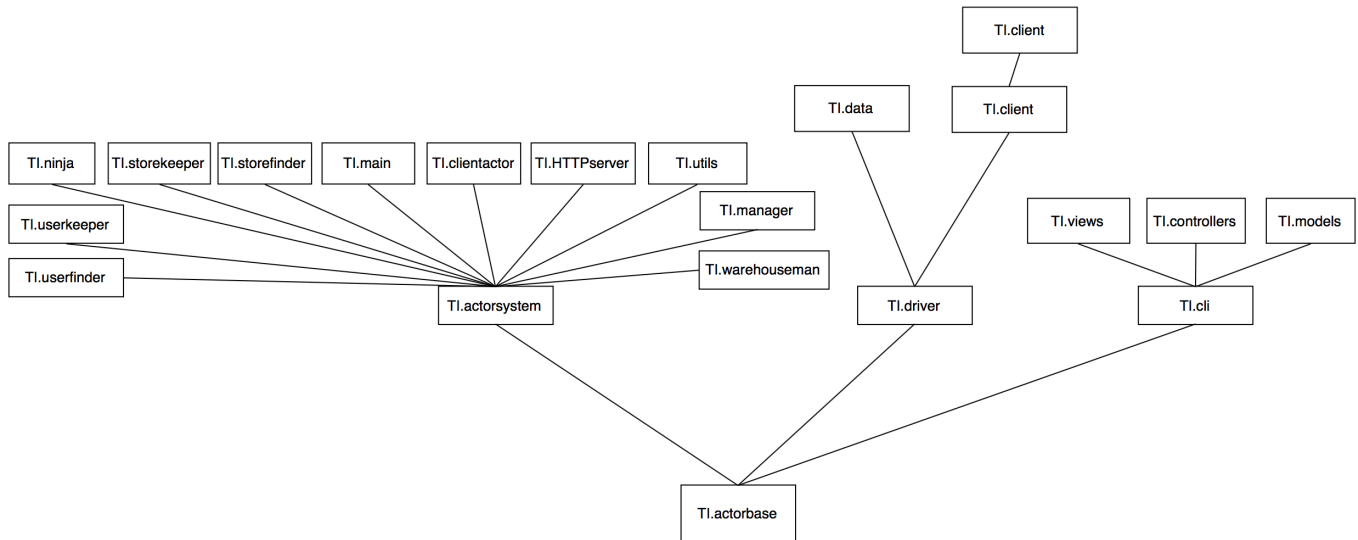


Figura 4: Diagramma informale della strategia di integrazione

Vengono ora presentate due tabelle. Nella prima c'è una sintesi dello stato dei test di integrazione. Se lo stato viene lasciato vuoto significa che la funzionalità non è stata implementata. Nella seconda vengono elencati e descritti tutti i test.

Test totali di integrazione	Test di integrazione in stato OK	Percentuale di test di integrazione superati
21	21	95%

Test	Descrizione	Componente	Stato
TI.actorbase	Test di integrazione finale per le componenti cli, driver e actorsystem	actorbase	OK
TI.cli	Test di integrazione finale per models, views e controllers	cli	OK
TI.models	Verifica della corretta esecuzione dei comandi mediante integrazione con componente driver	models	OK
TI.controllers	Verifica del parsing dei comandi e della corretta integrazione con models e views	controllers	OK
TI.views	Verifica del corretto funzionamento delle procedure di input e output	views	OK
TI.driver	Test di integrazione finale per client, data e api	driver	OK
TI.client	Verifica della comunicazione tra client e server e corretto funzionamento dei comandi	client	OK
TI.data	Test di controllo sulla corretta organizzazione dell'output in strutture navigabili e modificabili	data	OK



TI.api	Verifica della corretta integrazione con le componenti client e le strutture di creazione richieste HTTP <sub>G</sub>	api	OK
TI.actorsystem	Test di integrazione finale per HTTPserver, storefinder, storekeeper, ninja, manager, AuthActor, main, clientactor, warehouseman	actorsystem	OK
TI.utils	Verifica delle corrette procedure di organizzazione in strutture navigabili, crittazione e decrittazione dei dati di persistenza su disco	utils	OK
TI.HTTPServer	Test di verifica del corretto funzionamento del server HTTP	HTTPserver	OK
TI.clientactor	Test di verifica del parsing dei comandi ricevuti dall'esterno e della corretta esecuzione dei messaggi	clientactor	OK
TI.main	Verifica della corretta esecuzioni dei messaggi dedicati	main	OK
TI.storefinder	Verifica della corretta esecuzioni dei messaggi dedicati	storefinder	OK
TI.storekeeper	Verifica della corretta esecuzioni dei messaggi dedicati	storekeeper	OK
TI.AuthActor	Verifica della corretta esecuzioni dei messaggi dedicati	AuthActor	OK
TI.manager	Verifica della corretta esecuzioni dei messaggi dedicati	manager	OK
TI.ninja	Verifica della corretta esecuzioni dei messaggi dedicati	ninja	
TI.warehouseman	Verifica della corretta esecuzioni dei messaggi dedicati	warehouseman	OK

### C.2.1 Tracciamento componenti-test di integrazione

Componente	Test
actorbase	TI.actorbase
actorbase::cli	TI.cli
actorbase::cli::models	TI.models
actorbase::cli::controllers	TI.controllers
actorbase::cli::views	TI.views
actorbase::driver	TI.driver
actorbase::driver::client	TI.client
actorbase::driver::client::api	TI.api
actorbase::driver::data	TI.data
actorbase::actorsystem	TI.actorsystem
actorbase::actorsystem::utils	TI.utils
actorbase::actorsystem::HTTPServer	TI.HTTPServer
actorbase::actorsystem::clientactor	TI.clientactor
actorbase::actorsystem::main	TI.main
actorbase::actorsystem::storefinder	TI.storefinder
actorbase::actorsystem::storekeeper	TI.storekeeper
actorbase::actorsystem::AuthActor	TI.AuthActor
actorbase::actorsystem::manager	TI.manager
actorbase::actorsystem::ninja	TI.ninja
actorbase::actorsystem::warehouseman	TI.warehouseman



## C.3 test di unità

### C.3.1 descrizione

I test di unità servono al test di piccole unità del software associate ad un requisito.

Vengono ora presentate due tabelle, nella prima si può trovare una sintesi dello stato dei test, nella seconda vengono elencati e descritti tutti i test di unità previsti.

Test totali di unità	Test di unità in stato OK	Percentuale di test di unità superati
111	107	96%

Test	Descrizione	Stato	Requisito
TU.OBF1.1.1.1	Si verifica All'interno del singolo nodo <sub>G</sub> dovrà essere implementato l'attore HTTPserver <sub>G</sub> per gestire le connessioni provenienti dall'esterno	OK	OBF1.1.1.1
TU.OBF1.1.1.1.1	Si verifica che L'attore di tipo HTTPServer <sub>G</sub> dovrà gestire la ricezione del messaggio httprequest	OK	OBF1.1.1.1.1
TU.OBF1.1.2	Si verifica che All'interno del singolo nodo <sub>G</sub> dovrà essere implementato l'attore Clientactor <sub>G</sub> per gestire il ciclo di vita di una connessione dall'esterno	OK	OBF1.1.2
TU.OBF1.1.2.1	Si verifica che L'attore di tipo Clientactor <sub>G</sub> dovrà gestire la ricezione del messaggio Response	OK	OBF1.1.2.1
TU.OBF1.1.2.2	Si verifica che L'attore di tipo Clientactor <sub>G</sub> dovrà gestire la ricezione del messaggio MapResponse	OK	OBF1.1.2.2
TU.OBF1.1.3	Si verifica che L'attore di tipo Clientactor <sub>G</sub> dovrà gestire la ricezione del messaggio ListResponse	OK	OBF1.1.3
TU.OBF1.1.3.1	Si verifica che L'attore di tipo Main <sub>G</sub> dovrà gestire la ricezione del messaggio CreateCollection	OK	OBF1.1.3.1
TU.OBF1.1.3.4	Si verifica che L'attore di tipo Main <sub>G</sub> dovrà gestire la ricezione del messaggio GetFrom	OK	OBF1.1.3.4
TU.OBF1.1.3.5	Si verifica che L'attore di tipo Main <sub>G</sub> dovrà gestire la ricezione del messaggio AddContributor	OK	OBF1.1.3.5
TU.OBF1.1.3.7	Si verifica che L'attore di tipo Main <sub>G</sub> dovrà gestire la ricezione del messaggio RemoveContributor	OK	OBF1.1.3.7
TU.OBF1.1.3.8	Si verifica che L'attore di tipo Main <sub>G</sub> dovrà gestire la ricezione del messaggio CompleteTransaction	OK	OBF1.1.3.8
TU.OBF1.1.3.9	Si verifica che L'attore di tipo Main <sub>G</sub> dovrà gestire la ricezione del messaggio RemoveFrom	OK	OBF1.1.3.9
TU.OBF1.1.4	Si verifica che All'interno del singolo nodo dovrà essere implementato l'attore Storefinder <sub>G</sub> per la gestione delle collezioni <sub>G</sub> del database <sub>G</sub>	OK	OBF1.1.4



TU.OBF1.1.4.1	Si verifica che L'attore di tipo StoreFinder <sub>G</sub> dovrà gestire la ricezione del messaggio UpdateCollectionSize	OK	OBF1.1.4.1
TU.OBF1.1.4.3	Si verifica che L'attore di tipo StoreFinder <sub>G</sub> dovrà gestire la ricezione del messaggio Get	OK	OBF1.1.4.3
TU.OBF1.1.4.4	Si verifica che L'attore di tipo StoreFinder <sub>G</sub> dovrà gestire la ricezione del messaggio Remove	OK	OBF1.1.4.4
TU.OBF1.1.4.5	Si verifica che L'attore di tipo StoreFinder <sub>G</sub> dovrà gestire la ricezione del messaggio Insert	OK	OBF1.1.4.5
TU.OBF1.1.4.6	Si verifica che L'attore di tipo StoreFinder <sub>G</sub> dovrà gestire la ricezione del messaggio GetAllItems	OK	OBF1.1.4.6
TU.OBF1.1.4.7	Si verifica che L'attore di tipo StoreFinder <sub>G</sub> dovrà gestire la ricezione del messaggio PartialMapTransaction	OK	OBF1.1.4.7
TU.OBF1.1.5	Si verifica che All'interno del singolo nodo <sub>G</sub> dovrà essere implementato l'attore Storekeeper <sub>G</sub> per gestire il contenuto delle collezioni <sub>G</sub>	OK	OBF1.1.5
TU.OBF1.1.5.1	Si verifica che L'attore di tipo StoreKeeper <sub>G</sub> dovrà gestire la ricezione del messaggio Persist	OK	OBF1.1.5.1
TU.OBF1.1.5.2	Si verifica che L'attore di tipo StoreKeeper <sub>G</sub> dovrà gestire la ricezione del messaggio Get	OK	OBF1.1.5.2
TU.OBF1.1.5.3	Si verifica che L'attore di tipo StoreKeeper <sub>G</sub> dovrà gestire la ricezione del messaggio GetAll	OK	OBF1.1.5.3
TU.OBF1.1.5.4	Si verifica che L'attore di tipo StoreKeeper <sub>G</sub> dovrà gestire la ricezione del messaggio InsertItem	OK	OBF1.1.5.4
TU.OBF1.1.5.5	Si verifica che L'attore di tipo StoreKeeper <sub>G</sub> dovrà gestire la ricezione del messaggio RemoveItem	OK	OBF1.1.5.5
TU.OBF1.1.5.6	Si verifica che L'attore di tipo StoreKeeper <sub>G</sub> dovrà gestire la ricezione del messaggio InitMn	OK	OBF1.1.5.6
TU.OBF1.1.5.7	Si verifica che L'attore di tipo StoreKeeper <sub>G</sub> dovrà gestire la ricezione del messaggio BecomeNinja	not implemented	OBF1.1.5.7
TU.OBF1.1.6	Si verifica che All'interno del singolo nodo <sub>G</sub> dovrà essere implementato l'attore Warehouseman <sub>G</sub> per gestire la persistenza dei dati su disco	OK	OBF1.1.6
TU.OBF1.1.6.1	Si verifica che L'attore di tipo Warehouseman <sub>G</sub> dovrà gestire la ricezione del messaggio Read	OK	OBF1.1.6.1
TU.OBF1.1.6.2	Si verifica che L'attore di tipo Warehouseman <sub>G</sub> dovrà gestire la ricezione del messaggio Save	OK	OBF1.1.6.2
TU.OBF1.1.6.3	Si verifica che L'attore di tipo Warehouseman <sub>G</sub> dovrà gestire la ricezione del messaggio Clean	OK	OBF1.1.6.3
TU.OBF1.1.7	Si verifica che All'interno del singolo nodo <sub>G</sub> dovrà essere implementato l'attore Ninja <sub>G</sub> per gestire la ridondanza dei dati contenuti negli Storekeeper <sub>G</sub>	Not implemented	OBF1.1.7
TU.OBF1.1.7.1	Si verifica che L'attore di tipo Ninja <sub>G</sub> dovrà gestire la ricezione del messaggio Update	Not implemented	OBF1.1.7.1



TU.OBF1.1.7.2	Si verifica che L'attore di tipo Ninja <sub>G</sub> dovrà poter cambiare il proprio contesto ricevendo il messaggio BecomeSK e importare tutti i messaggi dello StoreKeeper	not implemented	OBF1.1.7.2
TU.OBF1.1.8	Si verifica che All'interno del singolo nodo <sub>G</sub> dovrà essere implementato l'attore Manager <sub>G</sub> per la gestione del carico degli attori di tipo Storekeeper <sub>G</sub> e Storefinder <sub>G</sub>	OK	OBF1.1.8
TU.OBF1.1.8.1	Si verifica che L'attore di tipo Manager <sub>G</sub> dovrà gestire la ricezione del messaggio OneMore	OK	OBF1.1.8.1
TU.OBF1.1.9	Si verifica che All'interno del singolo nodo dovrà essere implementato l'attore Authactor <sub>G</sub> per la gestione degli utenti	OK	OBF1.1.9
TU.OBF1.1.9.1	Si verifica che L'attore di tipo AuthActor <sub>G</sub> dovrà gestire la ricezione del messaggio AddCredential	OK	OBF1.1.9.1
TU.OBF1.1.9.2	Si verifica che L'attore di tipo AuthActor <sub>G</sub> dovrà gestire la ricezione del messaggio UpdateCredential	OK	OBF1.1.9.2
TU.OBF1.1.9.3	Si verifica che L'attore di tipo AuthActor <sub>G</sub> dovrà gestire la ricezione del messaggio RemoveCredential	OK	OBF1.1.9.3
TU.OBF1.1.9.4	Si verifica che L'attore di tipo AuthActor <sub>G</sub> dovrà gestire la ricezione del messaggio Authenticate	OK	OBF1.1.9.4
TU.OBF1.1.9.5	Si verifica che L'attore di tipo AuthActor <sub>G</sub> dovrà gestire la ricezione del messaggio AddCollectionTo	OK	OBF1.1.9.5
TU.OBF1.1.9.6	Si verifica che L'attore di tipo AuthActor <sub>G</sub> dovrà gestire la ricezione del messaggio RemoveCollectionFrom	OK	OBF1.1.9.6
TU.OBF1.1.9.7	Si verifica che L'attore di tipo AuthActor <sub>G</sub> dovrà gestire la ricezione del messaggio ListCollectionOf	OK	OBF1.1.9.7
TU.OBF1.1.10	Si verifica che Il sistema dovrà permettere di interagire con il sistema ad attori all'interno di ogni singolo nodo <sub>G</sub>	OK	OBF1.1.10
TU.OBF1.1.10.1.1	Si verifica che Il sistema ad attori dovrà inviare un messaggio di errore verso l'esterno in caso di credenziali errate, mediante interazioni tra AuthActor <sub>G</sub> e Clientactor <sub>G</sub>	OK	OBF1.1.10.1.1
TU.OBF1.1.10.2	Si verifica che Il sistema dovrà permettere di effettuare operazioni sulle collezioni <sub>G</sub>	OK	OBF1.1.10.2
TU.OBF1.1.10.2.1.1	Si verifica che Il sistema ad attori dovrà inviare un messaggio d'errore verso l'esterno in caso di nome collezione <sub>G</sub> già presente all'interno del sistema durante la creazione di nuova collezione <sub>G</sub> mediante l'interazione tra Clientactor <sub>G</sub> e Mainactor <sub>G</sub>	OK	OBF1.1.10.2.1.1
TU.OBF1.1.10.2.4.1	Si verifica che Il sistema ad attori dovrà inviare un messaggio d'errore verso l'esterno nel caso in cui il nome della collezione <sub>G</sub> da cancellare sia inesistente, mediante interazioni tra Clientactor <sub>G</sub> ed esterno	OK	OBF1.1.10.2.4.1



TU.OBF1.1.10.2.5.1	Si verifica che Il sistema ad attori dovrà permettere di aggiungere collaboratori a collezioni <sub>G</sub> con permessi lettura/scrittura mediante l'interazione tra Clientactor <sub>G</sub> , Mainactor <sub>G</sub> , Storefinder <sub>G</sub> , Storekeeper <sub>G</sub> , AuthActor <sub>G</sub> e Ninja <sub>G</sub>	OK	OBF1.1.10.2.5.1
TU.OBF1.1.10.2.5.2	Si verifica che Il sistema ad attori dovrà permettere di aggiungere collaboratori a collezioni <sub>G</sub> con permessi in sola lettura mediante l'interazione tra Clientactor <sub>G</sub> , Mainactor <sub>G</sub> , Storefinder <sub>G</sub> , Storekeeper <sub>G</sub> , AuthActor <sub>G</sub> e Ninja <sub>G</sub>	OK	OBF1.1.10.2.5.2
TU.OBF1.1.10.2.5.3	Si verifica che Il sistema ad attori dovrà inviare un messaggio d'errore verso l'esterno nel caso in cui il nome della collezione <sub>G</sub> a cui aggiungere collaboratori sia inesistente, mediante interazioni tra Clientactor <sub>G</sub> ed esterno	OK	OBF1.1.10.2.5.3
TU.OBF1.1.10.2.5.4	Si verifica che Il sistema ad attori dovrà inviare un messaggio d'errore verso l'esterno nel caso in cui lo username <sub>G</sub> da aggiungere tra i collaboratori di una collezione <sub>G</sub> sia inesistente, mediante interazioni tra Clientactor <sub>G</sub> , Main <sub>G</sub> e Storefinder <sub>G</sub>	OK	OBF1.1.10.2.5.4
TU.OBF1.1.10.2.6.1	Si verifica che Il sistema ad attori dovrà inviare un messaggio d'errore verso l'esterno nel caso in cui il nome della collezione <sub>G</sub> a cui rimuovere collaboratori sia inesistente, mediante interazioni tra Clientactor <sub>G</sub> ed esterno	OK	OBF1.1.10.2.6.1
TU.OBF1.1.10.3	Si verifica che Il sistema ad attori dovrà permettere di effettuare operazioni sugli item <sub>G</sub>	OK	OBF1.1.10.3
TU.OBF1.1.10.5.1	Si verifica che Il sistema ad attori dovrà inviare un messaggio d'errore verso l'esterno nel caso in cui durante la procedura di modifica della password le credenziali siano errate, mediante interazioni tra Clientactor <sub>G</sub> e AuthActor <sub>G</sub>	OK	OBF1.1.10.5.1
TU.OBF1.1.10.7	Si verifica che Il sistema ad attori dovrà permettere di effettuare operazioni di gestione utenti mediante interazioni tra Clientactor <sub>G</sub> , Main <sub>G</sub> , Storefinder <sub>G</sub> , AuthActor <sub>G</sub>	OK	OBF1.1.10.7
TU.OBF1.1.10.7.1.1	Si verifica che Il sistema ad attori dovrà inviare un messaggio d'errore verso l'esterno nel caso in cui il username <sub>G</sub> dell'utente da inserire all'interno del database <sub>G</sub> sia già esistente.	OK	OBF1.1.10.7.1.1



TU.OBF1.1.10.7.2.1	Si verifica che Il sistema ad attori dovrà inviare un messaggio d'errore verso l'esterno nel caso in cui lo username <sub>G</sub> dell'utente da rimuovere sia inesistente, mediante interazioni tra Clientactor <sub>G</sub> , Main <sub>G</sub> , Storefinder <sub>G</sub>	OK	OBF1.1.10.7.2.1
TU.OBF1.1.10.7.3.1	Si verifica che Il sistema ad attori dovrà inviare un messaggio d'errore verso l'esterno nel caso in cui lo username <sub>G</sub> dell'utente a cui resettare la password sia inesistente, mediante interazioni tra Clientactor <sub>G</sub> , Main <sub>G</sub> , Storefinder <sub>G</sub>	OK	OBF1.1.10.7.3.1
TU.OBF2.1.1.3	Si verifica che La CLI dovrà visualizzare un messaggio di errore esplicativo nel caso di credenziali di accesso non riconosciute	OK	OBF2.1.1.3
TU.OBF2.1.2	Si verifica che Il DSL dovrà fornire una serie di comandi per poter eseguire operazioni sulle collezioni <sub>G</sub> utilizzabile mediante CLI	OK	OBF2.1.2
TU.OBF2.1.2.7.2	Si verifica che Il comando per esportare collezioni <sub>G</sub> su file dovrà permettere di inserire il path del file su cui esportare le collezioni <sub>G</sub>	OK	OBF2.1.2.7.2
TU.OBF2.1.2.8	Si verifica che La CLI dovrà visualizzare un messaggio esplicativo in caso di inserimento di un nome collezione <sub>G</sub> già esistente all'interno del sistema durante la procedura di creazione nuova collezione <sub>G</sub>	OK	OBF2.1.2.8
TU.OBF2.1.2.11	Si verifica che La CLI dovrà visualizzare un messaggio di errore esplicativo in caso di inserimento nome collezione <sub>G</sub> da cancellare non esistente all'interno del sistema durante la procedura di cancellazione collezione <sub>G</sub>	OK	OBF2.1.2.11
TU.OBF2.1.2.12	Si verifica che La CLI dovrà visualizzare un messaggio di errore esplicativo in caso di inserimento nome collezione <sub>G</sub> non esistente all'interno del sistema durante la procedura di aggiunta collaboratore <sub>G</sub> a collezione <sub>G</sub>	OK	OBF2.1.2.12
TU.OBF2.1.2.13	Si verifica che La CLI dovrà visualizzare un messaggio di errore esplicativo in caso di inserimento nome collezione <sub>G</sub> non esistente all'interno del sistema durante la procedura di rimozione collaboratore <sub>G</sub> da collezione <sub>G</sub>	OK	OBF2.1.2.13
TU.OBF2.1.2.14	Si verifica che La CLI dovrà visualizzare un messaggio di errore esplicativo in caso di inserimento di uno username non esistente all'interno del sistema durante la procedura di aggiunta collaboratore <sub>G</sub> a collezione <sub>G</sub>	OK	OBF2.1.2.14
TU.DEF2.1.3	Si verifica che Il DSL dovrà fornire una serie di comandi per poter eseguire operazioni sugli item <sub>G</sub> utilizzabile mediante CLI	OK	DEF2.1.3





TU.DEF2.1.3.2	Si verifica che La CLI dovrà visualizzare un messaggio di errore esplicativo in caso di inserimento di un path <sub>G</sub> che punta a un file inesistente o non di tipo JSON <sub>G</sub> all'interno del filesystem durante la procedura di inserimento item da file	OK	DEF2.1.3.2
TU.DEF2.1.3.3	Si verifica che La CLI dovrà visualizzare un messaggio di errore esplicativo in caso di inserimento di un path <sub>G</sub> che punta ad un file JSON <sub>G</sub> non correttamente formato all'interno del filesystem	OK	DEF2.1.3.3
TU.OBF2.1.3.4	Si verifica che La CLI dovrà visualizzare un messaggio di errore esplicativo in caso di inserimento di una chiave già esistente durante la procedura di inserimento nuovo item <sub>G</sub> con flag <sub>G</sub> di sovrascrittura attivo	OK	OBF2.1.3.4
TU.OBF2.1.3.5	Si verifica che La CLI dovrà visualizzare un messaggio di errore esplicativo in caso di inserimento nome collezione <sub>G</sub> non esistente all'interno del sistema durante la procedura di cancellazione item <sub>G</sub>	OK	OBF2.1.3.5
TU.OBF2.1.7.4	Si verifica che La CLI dovrà visualizzare un messaggio di errore esplicativo in caso di password attuale non riconosciuta durante la procedura di modifica password	OK	OBF2.1.7.4
TU.OBF2.1.7.5	Si verifica che La CLI dovrà visualizzare un messaggio di errore esplicativo in caso di inserimento nuova password che non rispetta i vincoli imposti dal sistema durante la procedura di modifica della password	OK	OBF2.1.7.5
TU.OBF2.1.7.6	Si verifica che La CLI dovrà visualizzare un messaggio di errore esplicativo in caso di inserimento password di conferma non riconosciuta dal sistema durante la procedura di modifica della password	OK	OBF2.1.7.6
TU.OBF2.1.9	Si verifica che Il DSL dovrà fornire una serie di comandi per effettuare operazioni di gestione degli utente all'interno del sistema da parte di un utente amministratore utilizzabili mediante CLI	OK	OBF2.1.9
TU.OBF2.1.9.4	Si verifica che La CLI dovrà visualizzare un messaggio di errore esplicativo in caso di inserimento di uno username già censito a sistema durante la procedura di aggiunta nuovo utente	OK	OBF2.1.9.4
TU.OBF2.1.9.5	Si verifica che La CLI dovrà visualizzare un messaggio di errore esplicativo in caso di inserimento di uno username non esistente all'interno del sistema durante la procedura di rimozione utente	OK	OBF2.1.9.5



TU.OBF2.1.9.6	Si verifica che La CLI dovrà visualizzare un messaggio di errore esplicativo in caso di inserimento di uno username non esistente all'interno del sistema durante la procedura di reset della password di un utente	OK	OBF2.1.9.6
TU.DEF3.1.3	Si verifica che Il driver <sub>G</sub> dovrà lanciare un'eccezione nel caso di credenziali non riconosciute	OK	DEF3.1.3
TU.DEF3.2	Si verifica che Il driver <sub>G</sub> dovrà permettere l'esecuzione di comandi per poter eseguire operazioni sulle collezioni <sub>G</sub>	OK	DEF3.2
TU.DEF3.2.1	Si verifica che Il driver <sub>G</sub> dovrà permettere un la creazione di una nuova collezione <sub>G</sub>	OK	DEF3.2.1.2
TU.DEF3.2.1.2	Si verifica che Il driver <sub>G</sub> dovrà lanciare un'eccezione in caso di nome collezione <sub>G</sub> già censito durante la procedura di creazione	OK	DEF3.2.1.2
TU.DEF3.2.3.2	Si verifica che Il driver <sub>G</sub> dovrà lanciare un'eccezione in caso di inserimento nome collezione <sub>G</sub> da cancellare non esistente all'interno del sistema	OK	DEF3.2.3.2
TU.DEF3.2.4.3	Si verifica che Il driver <sub>G</sub> dovrà lanciare un'eccezione in caso di inserimento di un nome collezione <sub>G</sub> da modificare non esistente all'interno del sistema durante la procedura di modifica nome collezione <sub>G</sub>	OK	DEF3.2.4.3
TU.DEF3.2.4.4	Si verifica che Il driver <sub>G</sub> dovrà lanciare un'eccezione in caso di inserimento di un nuovo nome collezione <sub>G</sub> già censito a sistema durante la procedura di modifica nome collezione <sub>G</sub>	OK	DEF3.2.4.4
TU.DEF3.2.5.3	Si verifica che Il driver <sub>G</sub> dovrà lanciare un'eccezione in caso di inserimento nome collezione <sub>G</sub> a cui aggiungere un collaboratore <sub>G</sub> non esistente all'interno del sistema	OK	DEF3.2.5.3
TU.DEF3.2.5.4	Si verifica che Il driver <sub>G</sub> dovrà lanciare un'eccezione in caso di inserimento username non presente all'interno del sistema durante la procedura di aggiunta collaboratore <sub>G</sub> a collezione <sub>G</sub>	OK	DEF3.2.5.4
TU.DEF3.2.6.3	Si verifica che Il driver <sub>G</sub> dovrà lanciare un'eccezione in caso di inserimento nome collezione <sub>G</sub> a cui rimuovere un collaboratore <sub>G</sub> non esistente all'interno del sistema	OK	DEF3.2.6.3
TU.DEF3.2.7.3	Si verifica che Il driver <sub>G</sub> dovrà esportare tutto il contenuto del sistema in caso di inserimento di una lista nomi collezioni <sub>G</sub> vuota	OK	DEF3.2.7.3
TU.DEF3.3	Si verifica che Il driver <sub>G</sub> dovrà permettere di eseguire comandi per poter eseguire operazioni sugli item <sub>G</sub>	OK	DEF3.3
TU.DEF3.3.1.2.2	Si verifica che Il driver <sub>G</sub> dovrà lanciare un'eccezione in caso il file JSON <sub>G</sub> non sia presente nel filesystem secondo path <sub>G</sub> specificato	OK	DEF3.3.1.2.2



TU.DEF3.3.1.2.3	Si verifica che Il driver <sub>G</sub> dovrà lanciare un'eccezione in caso di inserimento di un path <sub>G</sub> che punta ad un file JSON <sub>G</sub> non correttamente formato	OK	DEF3.3.1.2.3
TU.DEF3.3.1.4	Si verifica che Il driver <sub>G</sub> dovrà lanciare un'eccezione in caso di inserimento di un item <sub>G</sub> con flag <sub>G</sub> di sovrascrittura non attivo e una chiave già esistente all'interno della collezione <sub>G</sub>	OK	DEF3.3.1.4
TU.DEF3.3.2.3	Si verifica che Il driver <sub>G</sub> dovrà lanciare un'eccezione nel caso di inserimento di un nome collezione <sub>G</sub> non esistente all'interno del sistema durante la procedura di cancellazione item <sub>G</sub>	OK	DEF3.3.2.3
TU.OBF3.6	Si verifica che Il driver <sub>G</sub> dovrà permettere di effettuare operazioni di gestione degli utente all'interno del sistema da parte di un utente amministratore	OK	OBF3.6
TU.OBF3.6.4	Si verifica che Il driver <sub>G</sub> dovrà lanciare un'eccezione nel caso in cui l'utente amministratore inserisca uno username già censito all'interno del sistema durante la procedura di aggiunta nuovo utente	OK	OBF3.6.4
TU.OBF3.6.5	Si verifica che Il driver <sub>G</sub> dovrà lanciare un'eccezione nel caso di inserimento di uno username non presente all'interno del sistema durante la procedura di rimozione utente da parte di utente amministratore	OK	OBF3.6.5
TU.OBF3.6.6	Si verifica che Il driver <sub>G</sub> dovrà lanciare un'eccezione nel caso di inserimento di uno username non presente all'interno del sistema durante la procedura di reset della password di un utente da parte di utente amministratore	OK	OBF3.6.6
TU.OBF3.7.3	Si verifica che Il driver <sub>G</sub> dovrà lanciare un'eccezione nel caso di inserimento della password attuale errata durante la procedura di modifica password	OK	OBF3.7.3
TU.OBF3.7.4	Si verifica che Il driver <sub>G</sub> dovrà lanciare un'eccezione nel caso di inserimento di una nuova password che non rispetta i vincoli del sistema durante l'operazione di modifica password	OK	OBF3.7.4
TU.OBF3.8.1	Si verifica che Il Driver <sub>G</sub> dovrà poter restituire sequenze di collezioni <sub>G</sub> navigabili	OK	OBF3.8.1
TU.OBF3.8.2	Si verifica che Il Driver <sub>G</sub> dovrà poter restituire collezioni <sub>G</sub> navigabili	OK	OBF3.8.2
TU.OBF3.8.3	Si verifica che Il Driver <sub>G</sub> dovrà poter restituire item <sub>G</sub>	OK	OBF3.8.3

## C.4 Test di validazione

I test di validazione servono all'accertamento che il prodotto realizzato sia conforme alle attese.  
Per ogni test vengono descritti i passi che un utente deve eseguire per testare i requisiti ad esso associati.



Si noti che i seguenti test eccetto **TV1**, **TV3** e **TV4** rappresentano le operazioni possibili sul database, per questo motivo tutti questi test saranno effettuabili sia tramite l'utilizzo della CLI<sub>G</sub> sia tramite l'utilizzo del Driver<sub>G</sub>. Nei test di validazione in cui è previsto l'inserimento di parametri per l'esecuzione di una operazione sul database si è scelto di non descrivere dettagliatamente tutti i parametri da inserire poiché questa descrizione è già stata redatta nell'[Analisi dei Requisiti 4.0.0](#).

#### C.4.1 Test TV1

L'utente intende aggiungere un nodo al cluster<sub>G</sub>.

L'utente deve:

- Avviare il server actorbase (TV1.1);
- Adattare il file di configurazione per potersi connettere al cluster<sub>G</sub> (TV1.2);
- Avviare actorbase sul nuovo nodo (TV1.3);
- Verificare tramite console l'avvenuta connessione al cluster<sub>G</sub> (TV1.4).

#### C.4.2 Test TV2

L'utente intende autenticarsi presso il sistema.

L'utente deve:

- Inserire il comando per effettuare l'autenticazione (TV2.1);
- Inserire i parametri per effettuare l'autenticazione (TV2.2);
- Verificare che l'autenticazione ha avuto successo controllando la risposta ricevuta dal server o eseguendo un comando disponibile solo ad utenti autenticati (TV2.3).

#### C.4.3 Test TV3

L'utente intende richiedere aiuto generale.

L'utente deve:

- Inserire il comando di aiuto generale attraverso l'uso della CLI<sub>G</sub> (TV3.1);
- Verificare che l'output della richiesta di aiuto è stato stampato in output sulla CLI<sub>G</sub> (TV3.2).

#### C.4.4 Test TV4

L'utente intende richiedere aiuto specifico di un comando.

L'utente deve:

- Inserire il comando di aiuto attraverso l'uso della CLI<sub>G</sub> (TV4.1);



- Inserire il nome del comando di cui vuole ricevere informazioni (TV4.2);
- Verificare che l'output della richiesta di aiuto è stato stampato in output sulla CLI<sub>G</sub> (TV4.3).

#### C.4.5 Test TV5

L'utente intende creare una collezione<sub>G</sub>.

L'utente deve:

- Inserire il comando per la creazione collezione<sub>G</sub> (TV5.1);
- Inserire il nome della collezione<sub>G</sub> da creare (TV5.2);
- Verificare l'avvenuta creazione della collezione<sub>G</sub> controllando la risposta ricevuta dal server o usando il comando di visualizzazione lista delle collezioni (TV5.3).

#### C.4.6 Test TV6

L'utente intende visualizzare la lista delle collezioni<sub>G</sub>.

L'utente deve:

- Inserire il comando per la visualizzazione lista delle collezioni<sub>G</sub> (TV6.1);
- Verificare l'avvenuta esecuzione del comando controllando la risposta ricevuta dal server (TV6.2).

#### C.4.7 Test TV7

L'utente intende cancellare una collezione<sub>G</sub>.

L'utente deve:

- Inserire il comando per la cancellazione collezione<sub>G</sub> (TV7.1);
- Inserire i parametri per cancellare la collezione<sub>G</sub> (TV7.2);
- Verificare l'avvenuta esecuzione del comando controllando la risposta ricevuta dal server o usando il comando di visualizzazione lista delle collezioni (TV7.3).

#### C.4.8 Test TV8

L'utente intende aggiungere un collaboratore ad una collezione<sub>G</sub>.

L'utente deve:

- Inserire il comando per l'aggiunta collaboratore a collezione<sub>G</sub> (TV8.1);
- Inserire i parametri per aggiungere il collaboratore alla collezione<sub>G</sub> (TV8.2);
- Verificare l'avvenuta esecuzione del comando controllando la risposta ricevuta dal server (TV8.3).



#### C.4.9 Test TV9

L'utente intende rimuovere un collaboratore da una collezione<sub>G</sub>.

L'utente deve:

- Inserire il comando per la rimozione di un collaboratore da un collezione<sub>G</sub> (TV9.1);
- Inserire i parametri per rimuovere un collaboratore da una collezione<sub>G</sub> (TV9.2);
- Verificare l'avvenuta esecuzione del comando controllando la risposta ricevuta dal server (TV9.3).

#### C.4.10 Test TV10

L'utente intende esportare una o più collezioni<sub>G</sub> su file.

L'utente deve:

- Inserire il comando per l'esportazione di collezioni<sub>G</sub> su file (TV10.1);
- Inserire i parametri per esportare collezioni<sub>G</sub> su file (TV10.2);
- Verificare l'avvenuta esecuzione del comando controllando la risposta ricevuta dal server o controllando il file salvato su disco (TV10.3).

#### C.4.11 Test TV11

L'utente intende inserire un item<sub>G</sub>. Se la collezione<sub>G</sub> su cui si vuole inserire l'item<sub>G</sub> non è presente nel sistema essa verrà creata.

L'utente deve:

- Inserire il comando per l'inserimento di item<sub>G</sub> (TV11.1);
- Inserire i parametri per inserire l'item<sub>G</sub> (TV11.2);
- Verificare l'avvenuta esecuzione del comando controllando la risposta ricevuta dal server o usando il comando di ricerca sulla collezione<sub>G</sub> giusta con la chiave appena usata (TV11.3).

#### C.4.12 Test TV12

L'utente intende inserire un item<sub>G</sub> importando da file.

L'utente deve:

- Inserire il comando per l'importazione di item<sub>G</sub> da file (TV12.1);
- Inserire i parametri per inserire gli item<sub>G</sub> tramite importazione (TV12.2);
- Verificare l'avvenuta esecuzione del comando controllando la risposta ricevuta dal server o usando il comando di ricerca sugli item<sub>G</sub> appena inseriti (TV12.3).



#### C.4.13 Test TV13

L'utente intende cancellare un item<sub>G</sub>.

L'utente deve:

- Inserire il comando per la cancellazione di item<sub>G</sub> (TV13.1);
- Inserire i parametri per cancellare l'item<sub>G</sub> (TV13.2);
- Verificare l'avvenuta esecuzione del comando controllando la risposta ricevuta dal server o usando il comando di ricerca sulla collezione<sub>G</sub> giusta con la chiave appena usata per effettuare la cancellazione (TV13.3).

#### C.4.14 Test TV14

L'utente intende effettuare una ricerca di item<sub>G</sub> su una o più collezioni<sub>G</sub>.

L'utente deve:

- Inserire il comando per la ricerca di item<sub>G</sub> (TV14.1);
- Inserire i parametri per la ricerca di un item<sub>G</sub> su una o più collezioni<sub>G</sub> (TV14.2);
- Verificare l'avvenuta esecuzione del comando controllando la risposta ricevuta dal server (TV14.3).

#### C.4.15 Test TV15

L'utente intende visualizzare tutti gli item<sub>G</sub> contenuti in una collezione<sub>G</sub>.

L'utente deve:

- Inserire il comando per la ricerca di item<sub>G</sub> (TV15.1);
- Inserire i parametri per la ricerca di tutti gli item<sub>G</sub> contenuti in una collezione<sub>G</sub> (TV15.2);
- Verificare l'avvenuta esecuzione del comando controllando la risposta ricevuta dal server (TV15.3).

#### C.4.16 Test TV16

L'utente intende effettuare una ricerca di un item<sub>G</sub> sull'intero database (secondo i suoi permessi come specificato nel documento [Analisi dei Requisiti 4.0.0](#)).

L'utente deve:

- Inserire il comando per la ricerca di item<sub>G</sub> (TV16.1);
- Inserire i parametri per la ricerca di un item<sub>G</sub> sull'intero database (TV16.2);
- Verificare l'avvenuta esecuzione del comando controllando la risposta ricevuta dal server (TV16.3).

**C.4.17 Test TV17**

L'utente intende visualizzare l'intero database (secondo i suoi permessi come specificato nel documento [Analisi dei Requisiti 4.0.0](#)).

L'utente deve:

- Inserire il comando per la ricerca di item<sub>c</sub> (TV17.1);
- Inserire i parametri per la ricerca per l'intero database (TV17.2);
- Verificare l'avvenuta esecuzione del comando controllando la risposta ricevuta dal server (TV17.3).

**C.4.18 Test TV18**

L'utente intende effettuare modificare la propria password.

L'utente deve:

- Inserire il comando per la modifica password (TV18.1);
- Inserire i parametri per la modifica password (TV18.2);
- Verificare l'avvenuta esecuzione del comando controllando la risposta ricevuta dal server o effettuando una deautenticazione seguita da una autenticazione con la nuova password (TV18.3).

**C.4.19 Test TV19**

L'utente intende effettuare deautenticazione dal server.

L'utente deve:

- Inserire il comando per effettuare il logout (TV19.1);
- Verificare l'avvenuta esecuzione del comando controllando la risposta ricevuta dal server o eseguendo un comando disponibile solo ad utenti autenticati (che dovrà non essere disponibile) (TV19.2).

**C.4.20 Test TV20**

L'utente amministratore intende aggiungere un utente al sistema.

L'utente deve:

- Inserire il comando per aggiungere un utente al sistema (TV20.1);
- Inserire i parametri per l'aggiunta utente (TV20.2);
- Verificare l'avvenuta esecuzione del comando controllando la risposta ricevuta dal server (TV20.3).





#### **C.4.21 Test TV21**

L'utente amministratore intende rimuovere un utente dal sistema.

L'utente deve:

- Inserire il comando per la rimozione utente (TV21.1);
- Inserire i parametri per la rimozione utente (TV21.2);
- Verificare l'avvenuta esecuzione del comando controllando la risposta ricevuta dal server (TV21.3).

#### **C.4.22 Test TV22**

L'utente amministratore intende effettuare un reset della password di un utente.

L'utente deve:

- Inserire il comando per effettuare il reset password (TV22.1);
- Inserire i parametri per il reset password (TV22.2);
- Verificare l'avvenuta esecuzione del comando controllando la risposta ricevuta dal server (TV22.3).



## Elenco delle figure

1	Visione piramidale dei sei livelli di maturità di processo. . . . .	12
2	Ciclo di Deming . . . . .	13
3	Modello dello standard ISO/IEC 9126 . . . . .	14
4	Diagramma informale della strategia di integrazione . . . . .	28