



Test Incident Report

Tech4All

Riferimento	C06_TIR_2.0
Versione	2.0
Data	15/01/2025
Destinatario	Top Management
Presentato da	Marco Capuano, Giovanni Cerchia, Arcangelo Ciaramella, Silvana De Martino, Giovanni Esposito, Luigi Nasta, Giovanni Salsano, Giuseppe Staiano
Approvato da	



Revision History

Data	Versione	Descrizione	Autori
05/01/2025	1.0	Creazione Test Incident Report.	Tutti.
07/01/2025	1.1	Stesura Test Registrazione.	Giuseppe Staiano, Giovanni Salsano.
07/01/2025	1.2	Stesura Test Login.	Giovanni Esposito.
07/01/2025	1.3	Stesura Test Visualizzazione Account.	Silvana De Martino.
07/01/2025	1.4	Stesura Test Creazione Tutorial e Test Visualizza lista Tutorial.	Marco Capuano, Giovanni Cerchia.
07/01/2025	1.5	Stesura Test Creazione Feedback e Visualizza Feedback.	Giovanni Salsano, Giuseppe Staiano.
07/01/2025	1.6	Stesura Test Creazione Quiz e Svolgimento Quiz.	Giovanni Esposito, Silvana De Martino, Luigi Nasta, Arcangelo Ciaramella.
08/01/2025	1.7	Revisione e aggiunta Test Svolgimento Quiz.	Silvana De Martino, Giovanni Esposito.



08/01/2025	1.8	Prima Revisione.	Marco Capuano, Arcangelo Ciaramella.
15/01/2025	2	Revisione Test Incident Report.	Giuseppe Staiano, Giovanni Salsano.



Team members

Nome	Ruolo nel progetto	Acronimo	Informazioni di contatto
Ferdinando Boccia	Project Manager	FB	f.boccia28@studenti.unisa.it
Domenico D'Antuono	Project Manager	DD	d.dantuono7@studenti.unisa.it
Silvana De Martino	Team Member	SDM	s.demartino30@studenti.unisa.it
Luigi Nasta	Team Member	LN	l.nasta4@studenti.unisa.it
Giovanni Salsano	Team Member	GSA	g.salsano14@studenti.unisa.it
Arcangelo Ciaramella	Team Member	AC	a.ciaramella7@studenti.unisa.it
Giovanni Esposito	Team Member	GE	g.esposito282@studenti.unisa.it
Giovanni Cerchia	Team Member	GC	g.cerchia6@studenti.unisa.it
Marco Capuano	Team Member	MC	m.capuano37@studenti.unisa.it
Giuseppe Staiano	Team Member	GS	g.staiano11@studenti.unisa.it



Sommario

1. Introduzione	6
2. Elenco Risultati Casi di Test	6
2.1 Test Registrazione (tester: Giuseppe Staiano, Giovanni Salsano)	6
2.1.1 Risultati	6
2.1.2 Osservazioni	6
2.2 Test Login (tester: Giovanni Esposito)	8
2.2.1 Risultati	8
2.2.2 Osservazioni	8
2.3.1 Risultati	9
2.3.2 Osservazioni	9
2.3.3 Conclusioni	9
2.4.1 Risultati	10
2.4.2 Osservazioni	10
2.4.3 Conclusioni	10
2.5.1 Risultati	11
2.5.2 Osservazioni	11
2.5.3 Conclusioni	11
2.6.1 Risultati	12
2.6.2 Osservazioni	12
2.6.3 Conclusioni	12
2.7.1 Risultati	14
2.7.2 Osservazioni	14
2.8.1 Risultati	15

1. Introduzione

In questo documento verranno visualizzati i risultati dell'esecuzione dei casi di test, che verranno messe poi sotto indagine nel TSR, nel caso in cui vengano scovati failure.

2. Elenco Risultati Casi di Test

Tabella per identificare i casi di test passati oppure no e poi specificare ogni caso di test non passato.

2.1 Test Registrazione (tester: Giuseppe Staiano, Giovanni Salsano)

```
PASS src/test/registration.test.ts
AutenticazioneService - Test registrazione
  ✓ TC Email già registrata (5 ms)
  ✓ TC Formato password non valido (1 ms)
  ✓ TC Nome non valido
  ✓ TC Cognome non valido (1 ms)
  ✓ TC Formato email non valido (1 ms)
  ✓ TC Registrazione completata (5 ms)

Test Suites: 1 passed, 1 total
Tests:       6 passed, 6 total
Snapshots:   0 total
Time:        1.66 s, estimated 2 s
```

2.1.1 Risultati

- Totale Test Eseguiti: 6
- Test Superati: 6
- Test Falliti: 0
- Stato Generale: Tutti i test sono stati superati con successo.

2.1.2 Osservazioni

L'analisi dei test evidenzia che la classe AutenticazioneService.ts, situata nel pacchetto src/app/services, ha superato tutti i controlli strumentali effettuati tramite Jest. I test sono stati completati con tempi variabili e non si riscontrano anomalie o errori nei comportamenti testati.

2.1.3 Conclusioni

I risultati positivi indicano che:

- I comportamenti previsti sono stati implementati correttamente;



- L'applicazione risponde in modo conforme alle specifiche per tutti gli scenari testati.

Questi esiti confermano l'affidabilità della classe `AutenticazioneService.ts` per quanto riguarda le funzionalità testate.

2.2 Test Login (tester: Giovanni Esposito)

```
PASS src/test/login.test.ts
AutenticazioneService - Test login
  ✓ TC Email o password mancanti (3 ms)
  ✓ TC Formato email non valido (1 ms)
  ✓ TC Lunghezza email non valida
  ✓ TC Formato password non valido (1 ms)
  ✓ TC Password troppo lunga (1 ms)
  ✓ TC Utente non trovato (1 ms)
  ✓ TC Errore interno del server (42 ms)
  ✓ TC Login completato con successo (1 ms)
  ✓ TC Email o password mancanti (1 ms)
  ✓ TC Errore interno del server (6 ms)

Test Suites: 1 passed, 1 total
Tests:       10 passed, 10 total
Snapshots:   0 total
Time:        2.258 s, estimated 4 s
```

2.2.1 Risultati

- Totale Test: 10
- Test Passati: 10
- Test Falliti: 0
- Stato dei Test: Tutti i test sono superati con successo.

2.2.2 Osservazioni

L'analisi dei test evidenzia che la classe `AutenticazioneService.ts`, situata nel pacchetto `src/app/services`, ha superato tutti i controlli strumentali effettuati tramite Jest. I test sono stati completati con tempi variabili e non si riscontrano anomalie o errori nei comportamenti testati.

2.2.3 Conclusioni

I risultati positivi indicano che:

- I comportamenti previsti sono stati implementati correttamente;
- L'applicazione risponde in modo conforme alle specifiche per tutti gli scenari testati.

Questi esiti confermano l'affidabilità della classe `AutenticazioneService.ts` per quanto riguarda le funzionalità testate.

2.3 Test Visualizzazione Account (tester: Silvana De Martino)

```
PASS src/test/account.test.ts
AccountService - visualizzaDati
  ✓ Dovrebbe mostrare i dati dell'utente se l'utente esiste (5 ms)
  ✓ Dovrebbe restituire un messaggio d'errore se l'utente non esiste (1 ms)
  ✓ Dovrebbe restituire un messaggio d'errore se non viene dato nessun id utente (1 ms)
AccountService - visualizzaUtenti
  ✓ Restituisce tutti gli utenti se gli utenti esistono (1 ms)
  ✓ dovrebbe restituire un messaggio quando viene scatenata un'eccezione (31 ms)
  ✓ dovrebbe restituire un messaggio d'errore se il metodo DAO restituisce null (1 ms)
  ✓ dovrebbe restituire un messaggio d'errore quando viene scatenata un'eccezione generica (8 ms)

Test Suites: 1 passed, 1 total
Tests:       7 passed, 7 total
Snapshots:   0 total
Time:        2.168 s, estimated 12 s
```

2.3.1 Risultati

- Totale Test: 7
- Test Passati: 7
- Test Falliti: 0
- Stato dei Test: Tutti i test sono superati con successo.

2.3.2 Osservazioni

L'analisi dei test evidenzia che la classe AccountService.ts, situata nel pacchetto src/app/services, ha superato tutti i controlli strumentali effettuati tramite Jest. I test sono stati completati con tempi variabili e non si riscontrano anomalie o errori nei comportamenti testati.

2.3.3 Conclusioni

I risultati positivi indicano che:

- I comportamenti previsti sono stati implementati correttamente;
- L'applicazione risponde in modo conforme alle specifiche per tutti gli scenari testati.

Questi esiti confermano l'affidabilità della classe AccountService.ts per quanto riguarda le funzionalità testate.

2.4 Test Creazione Tutorial e Visualizzazione lista Tutorial (tester: Marco Capuano, Giovanni Cerchia)

```
PASS src/test/tutorial.test.ts
TutorialService - Test creazioneTutorial
  ✓ TC Lunghezza titolo invalida (2 ms)
  ✓ TC Formato del titolo non corretto (1 ms)
  ✓ TC Formato della grafica non corretto (1 ms)
  ✓ TC Lunghezza testo invalida (1 ms)
  ✓ TC Categoria non trovata (6 ms)
  ✓ TC Creazione tutorial completata (1 ms)
TutorialService - Test visualizzazioneListaTutorial
  ✓ TC Lista tutorial vuota (1 ms)
  ✓ TC Errore durante la restituzione della lista di tutorial (27 ms)
  ✓ TC Lista tutorial restituita correttamente (1 ms)

Test Suites: 1 passed, 1 total
Tests:       9 passed, 9 total
Snapshots:   0 total
Time:        1.339 s, estimated 4 s
```

2.4.1 Risultati

- Totale Test: 9
- Test Passati: 9
- Test Falliti: 0
- Stato dei Test: Tutti i test sono superati con successo.

2.4.2 Osservazioni

L'analisi dei test evidenzia che la classe TutorialService.ts, situata nel pacchetto src/app/services, ha superato tutti i controlli strumentali effettuati tramite Jest.

I test coprono i due use case specificati nel R.A.D. nella sezione “Gestione Tutorial”, con tempi di esecuzione variabili, e non si riscontrano anomalie o errori nei comportamenti testati.

2.4.3 Conclusioni

I risultati positivi indicano che:

- I comportamenti previsti per la creazione e la visualizzazione dei tutorial sono stati implementati correttamente;
- L'applicazione risponde in modo conforme a quanto specificato nei due use case testati.

Questi esiti confermano l'affidabilità della classe TutorialService.ts per quanto riguarda le funzionalità testate.

2.5 Test Creazione Feedback (tester: Giovanni Salsano, Giuseppe Staiano)

```
PASS src/test/creafeedback.test.ts
FeedbackService - Creazione Feedback
  ✓ TC Recensione vuota (3 ms)
  ✓ TC Voto assente (1 ms)
  ✓ TC Lunghezza recensione troppo corta (1 ms)
  ✓ TC Lunghezza recensione troppo lunga
  ✓ TC Feedback esistente (1 ms)
  ✓ TC Feedback completato (2 ms)

Test Suites: 1 passed, 1 total
Tests:       6 passed, 6 total
Snapshots:  0 total
Time:        2.423 s
```

2.5.1 Risultati

- Totale Test: 6
- Test Passati: 6
- Test Falliti: 0
- Stato dei Test: Tutti i test sono superati con successo.

2.5.2 Osservazioni

L'analisi dei test evidenzia che la classe FeedbackService.ts, situata nel pacchetto src/app/services, ha superato tutti i controlli strumentali effettuati tramite Jest. I test sono stati completati con tempi variabili e non si riscontrano anomalie o errori nei comportamenti testati.

2.5.3 Conclusioni

I risultati positivi indicano che:

- I comportamenti previsti sono stati implementati correttamente;
- L'applicazione risponde in modo conforme alle specifiche per tutti gli scenari testati.

Questi esiti confermano l'affidabilità della classe FeedbackService.ts per quanto riguarda le funzionalità testate.

2.6 Test Visualizzazione Feedback (tester: Giovanni Salsano, Giuseppe Staiano)

```
PASS src/test/visualizzafeedback.test.ts
FeedbackService - VisualizzaFeedbackUtenteTutorial
  ✓ TC UserID mancante (5 ms)
  ✓ TC TutorialID mancante (1 ms)
  ✓ TC Feedback non trovato (1 ms)
  ✓ TC Feedback trovato (1 ms)
  ✓ dovrebbe gestire gli errori in modo appropriato (58 ms)
FeedbackService - VisualizzaFeedbackTutorial
  ✓ dovrebbe restituire un errore se tutorialId è mancante
  ✓ dovrebbe restituire un errore se il feedback non viene trovato (1 ms)
  ✓ dovrebbe restituire il feedback se trovato
  ✓ dovrebbe gestire gli errori in modo appropriato (7 ms)
FeedbackService - VisualizzaFeedbackUtente
  ✓ dovrebbe restituire un errore se userId è mancante (1 ms)
  ✓ dovrebbe restituire un errore se il feedback non viene trovato (1 ms)
  ✓ dovrebbe restituire il feedback se trovato (1 ms)
  ✓ dovrebbe gestire gli errori in modo appropriato (6 ms)

Test Suites: 1 passed, 1 total
Tests:      13 passed, 13 total
Snapshots:  0 total
Time:       3.147 s
```

2.6.1 Risultati

- Totale Test: 13
- Test Passati: 13
- Test Falliti: 0
- Stato dei Test: Tutti i test sono superati con successo.

2.6.2 Osservazioni

L'analisi dei test evidenzia che la classe FeedbackService.ts, situata nel pacchetto src/app/services, ha superato tutti i controlli strumentali effettuati tramite Jest. I test sono stati completati con tempi variabili e non si riscontrano anomalie o errori nei comportamenti testati.

Si specifica che sono stati sviluppati solo i test case relativi a "VisualizzaFeedbackUtenteTutorial" in quanto i test case per "VisualizzaFeedbackUtente" e "VisualizzaFeedbackTutorial" presentano logiche identiche. Per questo motivo, si è ritenuto sufficiente testare uno solo dei tre casi per verificare il corretto funzionamento della classe FeedbackService.ts.

2.6.3 Conclusioni

I risultati positivi indicano che:



- I comportamenti previsti sono stati implementati correttamente;
- L'applicazione risponde in modo conforme alle specifiche per tutti gli scenari testati.

Questi esiti confermano l'affidabilità della classe `FeedbackService.ts` per quanto riguarda le funzionalità testate.

2.7 Test Creazione Quiz (tester: Arcangelo Ciaramella)

```
PASS src/test/creazionequiz.test.ts
QuizService
  ✓ TC Lunghezza domanda non valida (corta) (7 ms)
  ✓ TC Lunghezza domanda non valida (troppo lunga) (2 ms)
  ✓ TC Lunghezza risposta non valida (troppo corta) (2 ms)
  ✓ TC Lunghezza risposta non valida (troppo lunga) (2 ms)
  ✓ TC Creazione Quiz Corretta (2 ms)

Test Suites: 1 passed, 1 total
Tests:       5 passed, 5 total
Snapshots:   0 total
Time:        2.738 s, estimated 5 s
```

2.7.1 Risultati

- Totale Test: 5
- Test Passati: 5
- Test Falliti: 0
- Stato dei Test: Tutti i test sono superati con successo.

2.7.2 Osservazioni

Dall'analisi dei test emerge che la classe QuizService.ts, situata nel pacchetto src/app/services, ha superato con successo tutti i controlli effettuati utilizzando Jest. I test sono stati completati in tempi variabili e non sono state rilevate anomalie o malfunzionamenti nei comportamenti analizzati

2.7.3 Conclusioni

I risultati ottenuti evidenziano che:

- I comportamenti previsti sono stati implementati correttamente;
- L'applicazione si comporta in modo conforme alle specifiche per tutti gli scenari testati.

Questi risultati confermano l'affidabilità della classe QuizService.ts relativamente alle funzionalità sottoposte a verifica.

2.8 Test Svolgimento Quiz (tester: Giovanni Esposito, Silvana De Martino, Luigi Nasta)

```
PASS src/test/svolgimento.test.ts
QuizService - getQuizByTutorialId
  ✓ dovrebbe restituire il quiz e le sue domande e risposte (5 ms)
  ✓ Dovrebbe lanciare un errore se tutorial non esiste (34 ms)
  ✓ Dovrebbe lanciare un errore se TutorialId è null (19 ms)
  ✓ Dovrebbe lanciare un errore se TutorialId è undefined (8 ms)
  ✓ Dovrebbe lanciare un errore se TutorialId è zero (7 ms)
  ✓ Dovrebbe dare errore se non viene trovato nessun quiz per il tutorial (6 ms)
QuizService - eseguiQuiz
  ✓ Dovrebbe aggiornare il numero di quiz superati in utente se il quiz viene passato (1 ms)
  ✓ Esecuzione corretta di un quiz (1 ms)
  ✓ Dovrebbe generare errore se l'id utente ha valore minore o uguale a 0 (7 ms)
  ✓ Dovrebbe generare errore se l'id quiz ha valore minore o uguale a 0 (7 ms)
  ✓ Dovrebbe restituire un messaggio di errore se il quiz non viene trovato (1 ms)
  ✓ Dovrebbe gestire un errore generico durante l'esecuzione del quiz (9 ms)
QuizService - eliminaQuiz
  ✓ Dovrebbe eliminare le risposte relative alle domande di un quiz che si sta eliminando (1 ms)
  ✓ Dovrebbe gestire gli errori generici (6 ms)
  ✓ Dovrebbe restituire errore se il quiz non è stato trovato (1 ms)

Test Suites: 1 passed, 1 total
Tests:       15 passed, 15 total
Snapshots:   0 total
Time:        2.073 s, estimated 4 s
```

2.8.1 Risultati

- Totale Test: 15
- Test Passati: 15
- Test Falliti: 0
- Stato dei Test: Tutti i test sono superati con successo.

2.8.2 Osservazioni

L'analisi dei test evidenzia che la classe QuizService.ts, situata nel pacchetto src/app/services, ha superato tutti i controlli strumentali effettuati tramite Jest. I test sono stati completati con tempi variabili e non si riscontrano anomalie o errori nei comportamenti testati.

Nello specifico sono stati sviluppati solo i test case relativi a "getQuizByTutorialId", "eseguiQuiz" e "eliminaQuiz".

2.8.3 Conclusioni

I risultati positivi indicano che:

- I comportamenti previsti sono stati implementati correttamente;
- L'applicazione risponde in modo conforme alle specifiche per tutti gli scenari testati.



Questi esiti confermano l'affidabilità della classe QuizService.ts per quanto riguarda le funzionalità testate.