



UNIVERSITÀ DEGLI STUDI  
DI SALERNO

---

## MuSe Remix Plugin

---

# Test Execution Report

### Docente / Tutor

Andrea De Lucia

Gerardo Iuliano

### Studenti

Carmine Calabrese (CC)  
0522501853

Daniele Carangelo (DC)  
0522501696

Data	Versione	Autori
11/09/2025	1.0	DC, CC

# Table of Contents

<b>1</b>	<b>Introduzione</b>	<b>2</b>
<b>2</b>	<b>Risultati Ottenuti</b>	<b>2</b>
2.1	Unit Testing . . . . .	2
2.2	Integration Testing - API . . . . .	3
2.3	System Testing . . . . .	3
<b>3</b>	<b>Incidenti Ricontrati</b>	<b>3</b>
<b>4</b>	<b>Execution tool report</b>	<b>4</b>
4.1	Jest testing . . . . .	4
4.2	Playwright testing . . . . .	5

# 1 Introduzione

Il presente documento riassume i risultati ottenuti durante l'attività di testing svolta nel corso della fase di manutenzione. L'obiettivo è fornire una panoramica complessiva sull'efficacia dei test eseguiti, valutando la copertura, gli esiti e l'eventuale presenza di anomalie. I dati presentati includono l'esecuzione dei test unitari e di integrazione, in riferimento alle modifiche introdotte con le `change request` CR\_01 e CR\_02.

## 2 Risultati Ottenuti

In questa sezione vengono riportati i risultati ottenuti durante la fase di testing, che hanno permesso di verificare la correttezza delle funzionalità implementate.

### 2.1 Unit Testing

In questa sottosezione vengono riportati i risultati dei test di unità eseguiti con il framework *Jest*. I test sono stati progettati per verificare il corretto funzionamento delle singole funzioni e dei moduli in isolamento.

I test di unità sono stati eseguiti sui due moduli principali riportati nella seguente tabella:

- **app.js:** Implementa la logica di comunicazione verso il tool MuSe tramite una API REST.
- **useRemixClient.js:** Client React che gestisce il collegamento con l'IDE Remix

Table 1: Riepilogo dei risultati dei test unitari

Metrica	Valore
Numero totale di test eseguiti	<b>35<sup>1</sup></b>
Test superati	<b>35</b>
Test falliti	<b>0</b>

Table 2: Code Coverage – Riepilogo per file

File	% Statements	% Branches	% Functions	% Lines
app.js	91.766	63.63	100	94.00
useRemixClient.js	90.09	92.85	83.33	92.30
<b>Totale</b>	<b>90.928</b>	<b>78.33</b>	<b>91.22</b>	<b>93.15</b>

<sup>1</sup>In questa sezione è presente anche il test case di regressione eseguito

## 2.2 Integration Testing - API

Questa sottosezione descrive i risultati dei test di integrazione delle API eseguiti utilizzando *Supertest*. L'obiettivo è verificare la corretta comunicazione tra client e server, la corretta esecuzione degli endpoint e la gestione degli errori.

Table 3: Riepilogo dei test di integrazione sulle API

Metrica	Valore
Numero totale di endpoint testati	<b>10</b>
Test superati	<b>10</b>
Test falliti	<b>0</b>

## 2.3 System Testing

Questa sottosezione descrive i risultati ottenuti durante il System Testing. Tramite l'utilizzo di framework come *Playwright*, è stato possibile automatizzare scenari realistici di utilizzo del tool, simulando l'interazione di un utente e testando tutte le funzionalità principali.

Table 4: Riepilogo dei risultati del System Testing

Metrica	Valore
Numero totale di scenari testati	<b>7</b>
Scenari superati	<b>6</b>
Scenari falliti	<b>1</b>

## 3 Incidenti Riscontrati

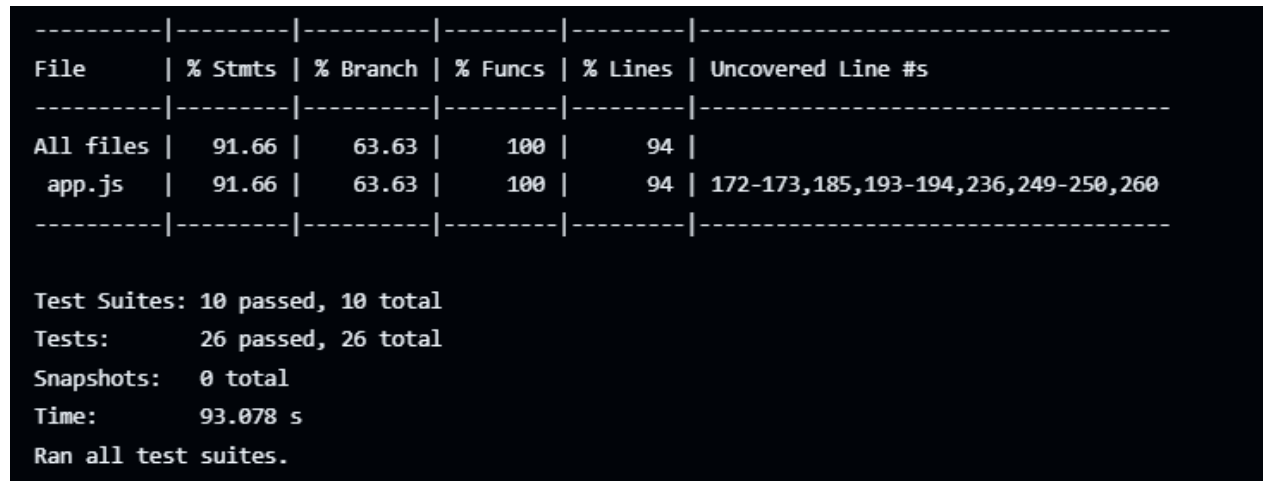
Nel corso delle attività di testing sono stati rilevati alcuni incidenti, di diversa gravità, documentati di seguito. La maggior parte è stata risolta, altri sono stati classificati come non bloccanti o ancora in fase di analisi.

Table 5: Elenco degli incidenti riscontrati durante il testing

ID	Categoria	Descrizione	Stato
TIR_01	Integration	Conflitti su file condivisi causati dalla parallelizzazione.	Risolto
TIR_02	System	Scenario "Test" flaky: esiti incoerenti tra esecuzioni.	Ignorato
TIR_03	Unit	Status restituiti in maniera errata.	Risolto
TIR_04	Unit	File temporanei non rimossi tra i test.	Risolto
TIR_05	Integration	Timeout troppo bassi nei test su API Test (Il tool MuSe impiega molto tempo per la fase di testing sui contratti mutati)	Risolto
TIR_06	Unit	Uso non mockato di fs causa overflow di memoria.	Risolto

## 4 Execution tool report

### 4.1 Jest testing



```

-----|-----|-----|-----|-----|-----
File      | % Stmts | % Branch | % Funcs | % Lines | Uncovered Line #s
-----|-----|-----|-----|-----|-----
All files | 91.66   | 63.63    | 100     | 94      |
app.js    | 91.66   | 63.63    | 100     | 94      | 172-173,185,193-194,236,249-250,260
-----|-----|-----|-----|-----|-----

Test Suites: 10 passed, 10 total
Tests:       26 passed, 26 total
Snapshots:   0 total
Time:        93.078 s
Ran all test suites.

```

Figure 1: Test esecuzione Back-end

File	% Stmts	% Branch	% Funcs	% Lines	Uncovered Line #s
All files	90.9	92.85	83.33	92.3	
useRemixClient.js	90.9	92.85	83.33	92.3	54,83-84,223-224,228-229,233-234
Test Suites: 1 passed, 1 total					
Tests: 19 passed, 19 total					
Snapshots: 0 total					
Time: 1.327 s					
Ran all test suites.					

Figure 2: Test execution Front-end

## 4.2 Playwright testing

Q

All6

✓ Passed6

Failed0

Flaky0

🔄 Skipped1

11/09/2025, 14:34:41 Total time: 1.4m

▼ e2e-plugin-test.spec.ts

✓ No mutant generated

e2e-plugin-test.spec.ts:68

27.0s

✓ Load plugin in Remix

e2e-plugin-test.spec.ts:140

7.0s

✓ Execute mutation

e2e-plugin-test.spec.ts:150

23.8s

✓ No mutant selected

e2e-plugin-test.spec.ts:178

7.4s

✓ No contract selected

e2e-plugin-test.spec.ts:201

9.7s

✓ No test file found

e2e-plugin-test.spec.ts:221

8.7s

Figure 3: Test execution System