

# TIR

## Test Incident Report



### MUSEUM MANAGEMENT **SYSTEM**

Riferimento	NC15_TIR_v1
Versione	1.0
Data	03/02/2024
Destinatario	Prof. Carmine Gravino
Presentato da	Team NC15
Approvato da	--



## Sommario

---

1.	Introduzione .....	3
1.1.	Scopo del documento .....	3
1.2.	Riferimenti .....	3
2.	Ambiente di Esecuzione .....	4
2.1.	Testing di Unità.....	4
3.	Test Incident Report .....	5

## Revision History

---

Data	Versione	Descrizione	Autori
03/02/2025	0.1	Prima stesura	AB
04/02/2025	0.5	Report test case VIS	AB, SN
05/02/2025	0.7	Report test case US e PER	VB
05/02/2025	1.0	Report test case AD	SN

## Team Members

---

Nome	Cognome	Acronimo	Contatto
Salvatore	Nocera	SN	s.nocera7@studenti.unisa.it
Angelo	Barone	AB	a.barone69@studenti.unisa.it
Vincenzo	Bucciero	VB	v.bucciero2@studenti.unisa.it

# 1.Introduzione

---

## 1.1. Scopo del documento

Il documento corrente riporta e descrive le attività di Testing effettuate per garantire il corretto funzionamento della Piattaforma web Museum Management System.

All'interno del documento verranno riportate le funzionalità testate, con i relativi strumenti utilizzati. Le funzionalità testate saranno:

- Registrazione alla piattaforma - Visitatore.
- Acquisto di un biglietto - Utente.
- Aggiunta di una nuova Opera - Personale.
- Aggiunta di un nuovo Avviso - Admin.

## 1.2. Riferimenti

Test Plan relativo al progetto Museum Management System: **TP**

Test Case Specification relativo al Progetto Museum Management System: **TCS**

Test Summary Report relativo al Progetto Museum Management System: **TSR**

## 2. Ambiente di Esecuzione

---

### 2.1. Testing di Unità

Per l'esecuzione dei test è stato sfruttato il framework di testing JUnit, in collaborazione con l'ambiente di sviluppo IntelliJ. JUnit offre la possibilità di testare le singole unità (unit test), e considerando che i test unitari potrebbero dipendere da classi che recuperano dati da servizi web, database in tempo reale o altre classi che potrebbero influenzare il risultato del test, è stato necessario ricorrere all'uso di mock che hanno consentito di emulare un servizio web o un database in tempo reale, fornendo risultati specifici in base alla situazione. Per simulare tali dipendenze, è stato impiegato il pacchetto Mockito.

### 3. Test Incident Report

Test Case ID	Test Incident ID	Tester	Data	Output
TC_VIS_1	TC_VIS_1	BARONE	4/02/2025	Passed
TC_VIS_2	TC_VIS_2	BARONE	4/02/2025	Passed
TC_VIS_3	TC_VIS_3	BARONE	4/02/2025	Passed
TC_VIS_4	TC_VIS_4	BARONE	4/02/2025	Passed
TC_VIS_5	TC_VIS_5	BARONE	4/02/2025	Passed
TC_VIS_6	TC_VIS_6	BARONE	4/02/2025	Passed
TC_VIS_7	TC_VIS_7	BARONE	4/02/2025	Passed
TC_VIS_8	TC_VIS_8	BARONE	4/02/2025	Passed
TC_VIS_9	TC_VIS_9	NOCERA	4/02/2025	Passed
TC_VIS_10	TC_VIS_10	NOCERA	4/02/2025	Passed
TC_VIS_11	TC_VIS_11	NOCERA	4/02/2025	Passed
TC_US_1	TC_US_1	BUCCIERO	5/02/2025	Passed
TC_US_2	TC_US_2	BUCCIERO	5/02/2025	Passed
TC_PER_1	TC_PER_1	BUCCIERO	5/02/2025	Passed
TC_PER_2	TC_PER_2	BUCCIERO	5/02/2025	Passed
TC_PER_3	TC_PER_3	BUCCIERO	5/02/2025	Passed
TC_PER_4	TC_PER_4	BUCCIERO	5/02/2025	Passed
TC_PER_5	TC_PER_5	BUCCIERO	5/02/2025	Passed
TC_PER_6	TC_PER_6	BUCCIERO	5/02/2025	Passed
TC_PER_7	TC_PER_7	BUCCIERO	5/02/2025	Passed
TC_AD_1	TC_AD_1	NOCERA	5/02/2025	Passed
TC_AD_2	TC_AD_2	NOCERA	5/02/2025	Passed



TC_AD_3	TC_AD_3	NOCERA	5/02/2025	Passed
TC_AD_4	TC_AD_4	NOCERA	5/02/2025	Passed

Esecuzione	Successi	Fallimenti
Testing di Unità	24	0