

# Test Execution Report for Mark

a.prisco50@studenti.unisa.it 0522501976

c.ranieri7@studenti.unisa.it 0522501977

June 30, 2025

## 1 Version History

Versione	Data	Descrizione
0.1	29/06/2025	Stesura della struttura del documento.
1.0	29/06/2025	Prima stesura del documento.

## 2 Riferimenti

All'interno del documento vengono citate le seguenti informazioni.

Riferimento	Descrizione
IGES_Mark_TPD	Documento relativo alla pianificazione della fase di testing.
IGES_Mark_TCS	Documento relativo alla specifica dei casi di test.
IGES_Mark_CCR	Documento relativo alla comprensione del codice in cui sono stati identificati una serie di problemi relativi ai dettagli tecnici del sistema.

## 3 Introduzione

Di seguito sono riportate tutte le informazioni relative all'esecuzione del piano di testing dopo l'intervento di additive reengineering.

**Obiettivo del Documento** Questo documento ha l'obiettivo di tenere traccia di tutte le informazioni relative ai risultati della fase di testing, le quali sono state raccolte in tabelle dettagliate e presentate nelle sezioni seguenti.

**Contesto del Progetto** Mark è un tool scritto in Python che dato un progetto, tramite l'ausilio di una Knowledge Base e delle detection rules, può classificare progetti ML in tre diverse categorie: ML-Consumer, ML-Producer e ML-Producer e Consumer.

## 4 Risultati della Fase di Testing Funzionale

Di seguito sono riportati i risultati dei test descritti nei documenti **IGES\_Mark\_TPD** e **IGES\_Mark\_TCS**.

**Nota.** Di seguito sono riportati i risultati per la fase di testing eseguita dopo l'aggiunta della GUI.

ID	Comportamento Rilevato	Esito Test
EA_0	Il sistema non va in crash a causa di path errato.	PASSED
EA_1	Nessuna classificazione effettuata, creati solo <i>results_first_step.csv</i> e <i>results_consumer.csv</i> .	PASSED
EA_2	Corretta classificazione del progetto come producer.	PASSED
EA_3	Corretta classificazione del progetto come consumer.	PASSED
EA_4	Corretta classificazione del progetto come producer e consumer.	PASSED
EA_5	Corretta classificazione del progetto come no-producer-consumer.	PASSED
EA_6	Corretta classificazione dei progetti come producer e consumer.	PASSED
EA_7	Corretta classificazione dei progetti come consumer e no-producer-consumer.	PASSED
EA_8	Corretta classificazione dei progetti come producer-consumer e producer.	PASSED
EA_9	Corretta classificazione dei progetti come no-producer-consumer, producer, consumer e producer-consumer.	PASSED

ID	Comportamento Rilevato	Esito Test
CL_0	Il sistema non va in crash a causa dell'assenza del file indicato come input.	PASSED
CL_1	Nessuna clonazione effettuata, come atteso in caso di file vuoto.	PASSED
CL_2	Nessuna clonazione effettuata per l'unica repository elencata, come previsto.	PASSED
CL_3	Nessuna clonazione effettuata per le repository elencate, come previsto.	PASSED

## 5 Risultati della Fase di Testing Strutturale

Di seguito sono riportati i risultati dei test descritti nei documenti **IGES\_Mark\_TPD** e **IGES\_Mark\_TCS**.

**Nota.** I test sono stati eseguiti una sola volta perché per le funzioni implementate si è raggiunta immediatamente una coverage media dell'80%.

**Nota.** I test sono stati effettuati sia con *no\_repos2* impostato a *True* sia a *False*.

**Nota.** Di seguito si riporta il significato delle colonne:

- **Tot** – Numero totale di righe del modulo.
- **Non** – Righe non coperte da alcun test.
- **Parz** – Righe parzialmente coperte (ad esempio, condizioni booleane valutate solo in parte).
- **Salt** – Righe saltate esplicitamente durante l'esecuzione.
- **Cov.** – Percentuale di copertura totale raggiunta sul modulo.
- **Righe non coperte / Note** – Dettagli sulle righe non eseguite e note di copertura generate dallo strumento di analisi (es. salti condizionali o blocchi non attraversati).

Modulo	Tot	Non	Parz	Salt	Cov.	Righe non coperte / Note
<i>cloner.py</i>	86	7	14	3	90%	44, 112→exit, 123–125, 128–130

**Nota.** Copertura aggiornata dopo l'aggiunta della GUI.