



2021/22

# TSR

## Test Summary Report

TEST SUMMARY REPORT

GERARDO LEONE, ANTONIO SANTOSUOSSO, MARIO LEZZI

Sommario

**Revision History** ..... 2

**Team Members** ..... 2

1. Introduzione..... 3

2. Relazione con altri documenti..... 3

3. Testing Unitario e di Integrazione..... 4

## Revision History

Data	Versione	Descrizione	Autori
25/01/2022	0.1	Prima Stesura	SA, ML
12/02/2022	0.1	Revisione	ML

## Team Members

Nome	Ruolo nel progetto	Acronimo	Informazione di contatto
Lezzi Mario	Team Member	LM	m.lezzi@studenti.unisa.it
Leone Gerardo	Team Member	LG	g.leone35@studenti.unisa.it
Santosuosso Antonio	Team Member	SA	a.santosuosso3@studenti.unisa.it

## 1. Introduzione

UniCinema si propone di migliorare le interazioni tra clienti e botteghino, cercando di limitare i rischi dell'emergenza sanitaria e proponendo un sistema in grado di fornire al cliente un modo veloce e semplice per acquistare biglietti online.

Il documento di Test Plan ha l'obiettivo di descrivere ed analizzare le attività di Testing per la piattaforma UniCinema. Il fine è quello di garantire che ogni aspetto funzioni in modo corretto.

All'interno del documento sono riportate le strategie di testing adottate, quali funzionalità saranno testate e gli strumenti scelti per la rilevazione degli errori, con lo scopo di presentare al cliente finale una piattaforma priva di malfunzionamenti.

Sono state pianificate attività di testing per le seguenti gestioni:

- Gestione Account
- Gestione Film

## 2. Relazione con altri documenti

Di seguito vengono elencate le relazioni tra il presente documento e gli altri documenti di testing.

### **Test Plan (TP)**

Il TSR fa riferimento alle attività di testing specificate nel TP.

### **Test Case Specification (TCS)**

Il TSR contiene il sunto dell'esecuzione dei test di sistema specificati nel TCS.

### **Test Incident Report (TIR)**

Il TSR contiene il sunto dei risultati sull'esecuzione specificati nel TIR.

### 3. Testing Unitario e di Integrazione

Nel corso del progetto, i test unitari e di integrazioni sono stati scritti e divisi in due distinte classi di test. In caso di problemi, era lo stesso sviluppatori a dover risolvere.

Per motivi di budget limitato, abbiamo preferito non effettuare il testing di integrazione su tutte le funzionalità del sito, ma soltanto su alcune che abbiamo ritenuto maggiormente prioritarie rispetto ad altre.

Il tool utilizzato per effettuare il testing unitario è jaCoCo, utilizzato tramite il tool di building Maven dell'IDE di IntelliJIDEA.

La tabella riporta la coverage raggiunta:

Branch Coverage	Instructions Coverage
77%	89%

Per un report completo si lascia il link al report di Code Coverage:

<https://bloodmask.github.io/UniCinema.github.io/site/jacoco/index.html>

## 4. Testing di sistema

Per il test di sistema si sono definite varie test suites tramite il tool Selenium IDE, per Chrome. Nello specifico, si è generata una Test Suite per Gestione (insieme di RF) e ogni esecuzione di una test suite ha richiesto il riavvio del sistema, per motivi legati al refresh dello storage. Di seguito vengono riportati i risultati delle esecuzioni dei tests.

Data Esecuzione	#Fallimenti	#Successi
07/02/2022	4	94
07/02/2022	0	98

**Nota:** il TC\_6.12 è stato testato manualmente per motivi legati al funzionamento del tool Selenium IDE.