



TSR

Test Summary Report

Stellar Library

Riferimento	NC10_TSR_ver.1.0
Versione	1.0
Data	09/01/2026
Destinatario	Prof. Carmine Gravino
Presentato da	NC10 Team



Revision History

Data	Versione	Descrizione	Autori
25/12/2025	0.1	Prima stesura	CG
09/01/2026	1.0	Revisione Finale	CG, BR, AP



Team members

Nome	Ruolo nel progetto	Acronimo	Informazioni di contatto
Antonio Plumitallo	Team Member	AP	a.plumitallo@studenti.unisa.it
Riccardo Odierna	Team Member	RO	r.odierna1@studenti.unisa.it
Biagio Radino	Team Member	BR	b.radino@studenti.unisa.it
Christian Antonio Genovese	Team Member	CG	c.genovese18@studenti.unisa.it



Sommario

Revision History	2
Team members	3
1 Introduzione.....	5
2 Relazione con altri documenti	5
3 Testing Unitario e di Integrazione.....	5
4 Testing di sistema	6



1 Introduzione

Il sistema StellarLibrary è una piattaforma e-commerce web-based progettata per supportare l'acquisto, la gestione e la scoperta di libri online. Il sistema permette agli utenti di consultare un catalogo organizzato, effettuare ricerche con filtri avanzati (titolo, autore, genere, prezzo, disponibilità), visualizzare schede prodotto complete di recensioni, gestire un carrello persistente e completare in sicurezza la procedura di checkout.

All'interno del documento sono riportate gli esiti dei test progettati nelle fasi precedenti 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 Autenticazione
- Gestione Checkout e Carrello
- Gestione Recensioni
- Gestioni Catalogo e Ricerca

2 Relazione con altri documenti

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

Relazione con il Test Plan

Il Test Summary Report fa riferimento alle attività di testing specificate nel Test Plan.

Relazione con il Test Case Specification

Il Test Summary Report contiene il sunto dell'esecuzione dei test di sistema specificati nel Test Case Specification.

Relazione con il Test Incident Report

Il Test Summary Report contiene il sunto dei risultati sull'esecuzione specificati nel Test Incident Report.



3 Testing Unitario

Nel corso del progetto, i test di unità sono stati sviluppati e organizzati in apposite classi di test, ciascuna dedicata alla verifica delle funzionalità delle classi implementate. Prima di proporre una modifica, ogni membro del team verificava in ambiente locale il corretto superamento dei test relativi alle classi modificate. Le nuove classi di test e le relative modifiche venivano quindi sottomesse tramite pull request sulla repository GitHub e sottoposte a revisione da parte di un altro membro del gruppo. Il merge delle modifiche veniva effettuato solo dopo la verifica del corretto funzionamento dei test; in caso di esito negativo, lo sviluppatore responsabile provvedeva all'individuazione e alla risoluzione del problema prima di riproporre le modifiche.



Laurea Triennale in informatica-Università di
Salerno Corso di *Ingegneria del Software* -
Prof.ssa C. Gravino

Nel corso e a fine progetto è stato usato il tool JaCoCo per la raccolta di metriche sulla coverage del test. Di seguito vengono riportati i risultati delle classi che implementano tali use case:

UC - GCC7 - Confermare un ordine

Branch Coverage	Line Coverage
44%	48%

UC - GA - Registrazione

Branch Coverage	Line Coverage
70%	65%

UC -GCR - Filtraggio del catalogo libri

Branch Coverage	Line Coverage
26%	26%

UC -GR - Inserimento di una recensione

Branch Coverage	Line Coverage
62%	72%

Le metriche di coverage riportate si riferiscono alle classi di service che implementano la logica dello use case testato viste nella loro interezza.

Ad esempio la coverage della classe *GestioneRecensioniService* include anche metodi non pertinenti allo use case “Inserimento di una recensione”, i quali non sono stati sottoposti a test di unità, al contrario di quelli inerenti agli use case specificati.

Per un report completo si lascia il link al report di JaCoCo:

[Report-jacoco](#)