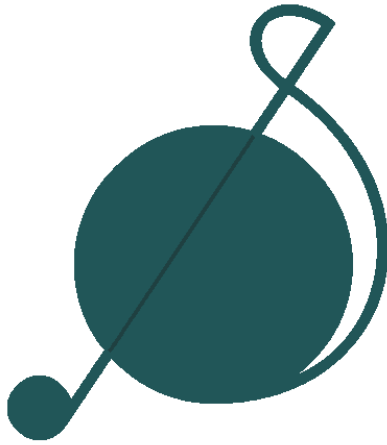




Laurea Triennale in informatica-Università di Salerno
Corso di
Ingegneria del software- Prof. Carmine Gravino



Spotibase

Test Summary Report

Spotibase

Docente

Gravino Carmine

Studenti

Silvio Venturino	0512108837
Umberto Della Monica	0512107742
Gagliarde Nicolapio	0512106980
Alessandro Aquino	0512106527

Revision History

Data	Versione	Descrizione	Autori
10/01/21	0.1	Creazione documento	SV
13/01/21	0.2	Modifica testing di sistema	SV
14/01/21	1.0	Modifica testing unitario, aggiunta link	Tutto il team

Team Members

Nome	Ruolo nel progetto	Email	Acronimo
Alessandro Aquino	Team member	a.aquino33@studenti.unisa.it	AA
Nicolapio Gagliarde	Team member	n.gagliarde@studenti.unisa.it	NG
Umberto Della Monica	Team member	u.dellamonica2@studenti.unisa.it	UM
Silvio Venturino	Team member	s.venturino@studenti.unisa.it	SV

Sommario

1. [Introduzione](#)
2. [Relazioni con altri documenti](#)
3. [Testing unitario e di integrazione](#)
4. [Testing di sistema](#)

1.Introduzione

Spotibase intende fornire agli utenti una piattaforma musicale che offre lo streaming on demand di canzoni, integrate con le funzionalità di un e-commerce.

Il documento di Test Plan ha l'obiettivo di descrivere ed analizzare le attività di Testing per il sistema Spotibase. Lo scopo è quello di garantire che ogni aspetto funzioni in modo corretto, prima del rilascio del sistema.

Di seguito si elencano le strategie adottate per il testing, le funzionalità da testare e le tecnologie utilizzate per rilevare gli errori. Il fine è quello di presentare al cliente una piattaforma priva di malfunzionamenti.

Sono state pianificate attività di testing per le seguenti gestioni:

- Gestione Canzone
- Gestione Carrello
- Gestione Playlist
- Gestione di Autenticazione
- Gestione Preferenza

2.Relazioni 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 e di integrazione

Il testing unitario verrà effettuato sulla parte di sistema relativa agli use case definiti nel RAD. In particolare verranno testate le Gestioni sopra citate.

Il testing unitario è stato suddiviso per tier. In particolare sono stati testati il data tier contenente i DAO e il logic tier contenente le servlet.

Per ogni classe DAO e servlet è stata creata una classe di test. Una volta testata la classe di test, si lancia l'intera suite di test.

Non si riportano i risultati del testing di integrazione, poichè il sistema risultava già implementato e quindi non è stato effettuato. Per completezza si riporta nel codice un esempio di testing di integrazione.

Il tool utilizzato per effettuare testing di unità e di integrazione è JaCoCo, utilizzato tramite il tool di building Maven dell' ide IntelliJ.

La coverage raggiunta viene riporta di seguito:

Branch Coverage	Instructions Coverage
78%	80%

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

<https://gagliardenicolapio.github.io/spotibase/jacoco/>

4. Testing di sistema

Il testing di sistema verrà effettuato sulla base dei test case definiti nel TPD.

In particolare si testeranno solo le funzionalità che permettono all'utente di inserire dati e di interagire con l'interfaccia del sistema.

Per il testing di sistema si definiscono varie test suites, tramite il tool Selenium IDE di Chrome.

I risultati delle esecuzioni dei tests sono riportati nella cartella "System Testing Selenium"

all'interno della cartella "Testing".