­

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Test Summary Report  Progetto  SalernArte   |  |  | | --- | --- | | Versione | 0.1 | | Data | 17/02/2023 | | Destinatario | Studenti di Ingegneria del Software 2021/22 | | Presentato da | Martino Lucia [0512105234], Longo Marco [0512105945], Della Pepa Alessia [0512105720] | | Approvato da |  | |

Membri del Team

|  |  |  |  |
| --- | --- | --- | --- |
| **Nome** | **Matricola** | **Acronimo** | **Informazioni di contatto** |
| Della Pepa Alessia | 0512105720 | DPA | a.dellapepa5@studenti.unisa.it |
| Longo Marco | 0512105945 | LM | m.longo36@studenti.unisa.it |
| Martino Lucia | 0512105234 | ML | l.martino11@studenti.unisa.it |

RevisionHistory

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versione** | **Descrizione** | **Autori** |
| 17/02/2023 | 0.1 | Prima stesura: struttura del documento | LM |

Sommario

[Membri del Team 2](#_Toc127546408)

[RevisionHistory 2](#_Toc127546409)

[1. Introduzione 2](#_Toc127546410)

[2. Relazione con gli altri documenti 3](#_Toc127546411)

[3. Testing Unitario e di Integrazione 3](#_Toc127546412)

[4. Testing di sistema 4](#_Toc127546413)

1. Introduzione

SalernArte si propone come obiettivo principale di realizzare un’agenzia online specializzata nella vendita di biglietti riguardanti mostre d’arte ed eventi teatrali e culturali nel salernitano. L’obiettivo è quello di facilitare a tutti i cittadini la ricerca di iniziative culturali salernitane, raggruppando queste ultime in un unico ambiente semplice ed intuitivo, e di ottimizzare il lavoro di organizzatori di eventi che si interfacciano ai cittadini.

Nel documento di Test Plan abbiamo descritto le modalità di Testing che sono state scelte e che sono state utilizzate per la piattaforma SalernArte.

Dal momento che il sistema è stato diviso in tre gestioni, per facilitare la divisione degli incarichi e la leggibilità tra i membri del team, abbiamo deciso di mantenere la suddivisione anche nelle attività di testing che verranno divise nelle seguenti gestioni:

* Gestione Utente
* Gestione Eventi
* Gestione Carrello

1. Relazione con gli altri documenti

In questa sezione si descrivono le relazioni tra il Test Summary Report e i documenti utilizzati per la descrizione degli eventi di Testing.

**Relazione con il Test Plan**

Nel Test Plan abbiamo descritto le attività di testing a cui il Test Summary Report si riferisce.

**Relazione con il Test Case Specification**

Si descrive nel seguente documento il risultato dell’esecuzione del testing di sistema effettuato nel Test Case Specification.

**Relazione con il Test Incident Report**

Il Test Incident Report effettua una panoramica riassuntiva del testing di sistema effettuato e che viene ripreso dal Test Summary Report.

1. Testing Unitario e di Integrazione

Durante la stesura delle classi di test abbiamo proceduto a suddividere il lavoro da svolgere nelle due tipologie di classi di test. Ogni membro del team aveva il compito di lavorare nella Gestione assegnata e implementare le classi di testing che gli appartenevano; si procedeva con lo sviluppo delle classi di test d’unità e infine con quelle d’integrazione.

Si procedeva poi alla verifica del testing delle classi sviluppate: nel caso in cui le classi di test implementate contenessero degli errori, il team member proprietario della classe procedeva alla correzione degli errori e infine effettuava il push sulla repository GitHub.  
Quando si effettuava la pull request il sistema eseguiva i test che erano stati caricati per verificarne la correttezza.   
Per dimostrare la raggiunta della Branche Coverage abbiamo utilizzato il tool JaCoCo che ha restituito i seguenti risultati:

Immagine che contiene testo, screenshot, computer, monitor

Descrizione generata automaticamente

Nel percorso “DOCUMENTAZIONE\Test d’unità e integrazione” sono reperibili tutti i report dell’esecuzione dei test; il report della coverage presentata è reperibile all’interno del file [index.html](Test%20d'unità%20e%20integrazione/index.html).

1. Testing di sistema

Terminata la fase di testing d’unità e di integrazione abbiamo effettuato un test completo del sistema delle funzionalità testate utilizzando il tool Selenium IDE, per Chrome. Per ogni Gestione individuata e assegnata in precedenza ad ogni team member abbiamo generato una Test Suite dove all’avvio del sistema si settava l’ambiente in modo che la pre-condizione fosse verificata e al termine dell’esecuzione, una volta controllato l’Assert assegnato, è stato resettato il sistema alla formazione originale per non intaccare i successivi test case.

Ogni volta che si verificava un errore del test case si procedeva ad individuare l’esecuzione errata e, una volta modificata, si procedeva con l’esecuzione della Test Suite corrispondente.

Di seguito, si presentano i risultati ottenuti dall’esecuzione dei tests: