­

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Test Plan  Progetto  SalernArte   |  |  | | --- | --- | | Versione | 0.1 | | Data | 28/11/2022 | | 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** |
| 28/11/2022 | 0.1 | Prima stesura: struttura del documento | DPA, ML, LM |
| 30/11/2022 | 0.2 | Funzionalità da testare e non: gestione utente | ML |
| 1/12/2022 | 0.3 | Gestione Eventi: Funzionalità da testare e non, Test cases: richiesta inserimento evento, richiesta modifica evento e ricerca evento | DPA |
| 02/12/2022 | 0.4 | Gestione Carrello: funzionalità da testare e non  Gestione Utente: funzionalità da testare e non, Test Cases: registrazione organizzatore ed utente, login, modifica profilo organizzatore | LM |
| 03/12/2022 | 0.5 | Gestione Eventi: Terminati tutti i test cases | DPA |
| 05/12/2022 | 0.6 | Modifica profilo amministratore: funzionalità da testare e non  Registrazione scolaresca: funzionalità da testare e non | ML |
| 08/01/2023 | 0.7 | Modificata tabella modifica profilo amministratore, creazione tabella modifica profilo utente | ML |
| 30/01/2023 | 0.8 | Revisione del documento ed aggiunta delle sezioni mancanti e di autori all’interno delle tabelle | DPA |
| 06/02/2023 | 0.9 | Revisione del documento ed aggiunta delle sezioni mancanti e di autori all’interno delle tabelle | LM |

Sommario

[Membri del Team 2](#_Toc126090955)

[RevisionHistory 2](#_Toc126090956)

[1. Introduzione 4](#_Toc126090957)

[2. Relazione con gli altri documenti 4](#_Toc126090958)

[3. Panoramica del sistema 4](#_Toc126090959)

[4. Funzionalità da testare e non testare 5](#_Toc126090960)

[4.1 Gestione Utente 5](#_Toc126090961)

[4.2 Gestione Eventi 5](#_Toc126090962)

[4.3 Gestione Carrello 5](#_Toc126090963)

[5. Pass/Fail criteria 6](#_Toc126090964)

[6. Approccio 6](#_Toc126090965)

[6.1 Testing di unità 6](#_Toc126090966)

[6.2 Testing funzionale 6](#_Toc126090967)

[6.3 Testing di Sistema 6](#_Toc126090968)

[6.4 Testing di integrazione 6](#_Toc126090969)

[7. Sospensione e ripristino 6](#_Toc126090970)

[8. Strumenti per il testing (hardware/software) 7](#_Toc126090971)

[9. Test cases 7](#_Toc126090972)

[9.1 Gestione Utente 7](#_Toc126090973)

[9.1.1 Registrazione Utente 7](#_Toc126090974)

[9.1.2 Modifica profilo utente 10](#_Toc126090975)

[9.1.3 Login 13](#_Toc126090976)

[9.1.4 Modifica Profilo Organizzatore 13](#_Toc126090977)

[9.1.5 Registrazione Organizzatore 17](#_Toc126090978)

[9.1.6 Modifica Profilo Amministratore 19](#_Toc126090979)

[9.1.7 Registrazione Scolaresca 22](#_Toc126090980)

[9.1.8 Modifica profilo scolaresca 23](#_Toc126090981)

[9.2 Gestione Eventi 25](#_Toc126090982)

[9.2.1 Richiesta inserimento Evento 25](#_Toc126090983)

[9.2.2 Richiesta modifica Evento 29](#_Toc126090984)

[9.2.3 Ricerca Evento 33](#_Toc126090985)

[9.3 Gestione Carrello 33](#_Toc126090986)

[9.3.1 Aggiungi al carrello 33](#_Toc126090987)

[9.3.2 Modifica Carrello 34](#_Toc126090988)

[10. Testing schedule 34](#_Toc126090989)

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.

Questo documento ha come obiettivo quello di descrivere le modalità di Testing che sono state scelte e che verranno utilizzate per la piattaforma SalernArte. Nello specifico andremo quindi a descrivere i criteri che sono stati utilizzati per scegliere le funzionalità da testare, in che modo queste attività sono state divise tra i componenti del gruppo, descriveremo l’approccio scelto per il Testing della piattaforma per poi entrare nel dettaglio dei TestCase individuati e di vari criteri che andranno seguiti.

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

Per la stesura e l’individuazione dei test case si fa riferimento agli altri documenti che sono stati prodotti:

* [Requirements Analysis Document](2-Requirements%20Analysis%20Document.docx) (RAD), da cui abbiamo ricavato tramite l’analisi dei requisiti funzionali e non funzionali i test case che verranno presentati nel seguente documento
* [System Design Document](3-System%20Design%20Document%20.docx) (SDD), da cui verrà presa la suddivisione in sottosistemi che dovrà essere rispettata all’interno dei test case.
* [Object Design Document](4-Object%20Design%20Document.docx) (ODD), da rivedere.

1. Panoramica del sistema

Il Sistema proposto è basato sullo stile architetturale Three Tier. Questo tipo di architettura è stata scelta poiché si sposa perfettamente per lo sviluppo di Web Application, come previsto dal nostro progetto.

Per la parte del Front-end saranno utilizzati HTML5 e CSS3.Per quanto riguarda il Back-end verranno utilizzati Java e Javascript. Per la gestione del database saranno usati MySQL e JDBC.

1. Funzionalità da testare e non testare

Le funzionalità che non verranno testate sono quelle che non prevedono l’inserimento di input da parte dell’utente - ad esempio attività riguardanti esclusivamente visualizzazioni di dati oppure l’utilizzo di un bottone. I requisiti funzionali di priorità alta che non vengono testati in questa fase per i motivi sopracitati, verranno testati successivamente con diverse tecniche e saranno quindi trattati in una diversa documentazione.

Di seguito la lista delle funzionalità di cui si effettuerà il testing per le varie gestioni:

## 4.1 Gestione Utente

Le funzionalità che saranno testate relative alla Gestione Utente sono le seguenti:

* Registrazione utente
* Registrazione scolaresca
* Registrazione organizzatore
* Modifica profilo utente
* Modifica profilo organizzatore
* Modifica profilo scolaresca
* Modifica profilo amministratore
* Login

## 4.2 Gestione Eventi

Le funzionalità che saranno testate relative alla Gestione Eventi sono le seguenti:

* Richiesta inserimento evento
* Richiesta modifica evento
* Ricerca evento

## 4.3 Gestione Carrello

Le funzionalità che saranno testate relative alla Gestione Carrello saranno:

* Aggiungi al carrello
* Modifica carrello

1. Pass/Fail criteria

La fase di testing ha come scopo quello di provare il sistema al fine di controllare che i requisiti stabiliti con il cliente siano stati rispettati e di individuare quanti più errori possibili che saranno quindi successivamente corretti. Ricordiamo che è possibile provare la presenza di errori in un sistema, ma non la loro assenza.

Sulla base dei requisiti individuati nella fase di analisi verranno scritti i casi di test (Test cases), che saranno valutati utilizzando un oracolo che rappresenta il comportamento atteso che dovrà essere confrontato con il risultato effettivo dell’esecuzione.

Un test è considerato un successo (pass) se, per un determinato input, il risultato atteso è diverso da quello osservato ed è stato quindi rilevato un malfunzionamento.

Un test verrà considerato un fallimento (fail) se, per un determinato input, il risultato atteso e quello osservato coincidono e non è stato rilevato un malfunzionamento.

Il testing sarà considerato valido nel momento in cui i vincoli elencati di seguito saranno stati tutti rispettati:

* Testare tutti i requisiti funzionali di alta e media priorità
* Raggiungere una branch coverage non inferiore al 75%

1. Approccio

Possiamo dividere le fasi di Test del sistema rappresentato dalla piattaforma SalernArte in 3 fasi: Test di Sistema, Testi di Integrazione e Test di Unità.

La progettazione delle attività di test avverrà seguendo l’ordine in cui sono state presentate le fasi di test, ma i test verranno successivamente eseguiti seguendo l’ordine inverso: ossia effettuando prima Testing di Unita, poi di Integrazione ed infine Test di Sistema.

Successivamente alla stesura del Requirements Analysis Document verrà progettato il Test di Sistema tramite l’analisi dei Requisiti individuati.

Successivamente alla stesura del System Design Document ed alla conseguente divisione dell’architettura del sistema in sottosistemi e layer, verranno prese decisioni relative alla progettazione dei Test di Unità e di Integrazione.

Durante la fase di implementazione verranno eseguite periodiche attività di revisione del codice eseguite dall’intero team tramite meet settimanali organizzati sulla piattaforma Teams. Per ogni sottosistema implementato verranno prodotte prima le classi che effettueranno il Test di Unità e successivamente quelle relative al Test di Integrazione.

Dopo la fase di implementazione e del Testing di Unita e di Integrazione, verrà eseguito il Test di Sistema .

## 6.1 Testing di Sistema

Il Test di Sistema sarà eseguito con il supporto del tool Selenium IDE, che permette di automatizzare l’esecuzione del test di sistema registrando le azioni che un utente può eseguire all’interno del browser e permettendo di definire un oracolo per ogni esecuzione. Verrà fatto il deploy del sistema in localhost per permettere di eseguire i test di sistema.

A causa del basso budget a disposizione, non si assicura l’esecuzione delle seguenti categorie di Testing di Sistema:

* Performance Testing
* Pilot Testing
* Acceptance Testing
* Installation Testing

### 6.1 Testing funzionale

L’obiettivo di del test Funzionale è la validazione dei requisiti funzionali individuati in fase di Analisi Dei Requisiti. Tramite l’analisi dei requisiti funzionali vengono individuati i Test Case che andremo a definire in questo documento, che saranno scelti tra le funzionalità chiave del sistema e tra quelle che hanno più probabilità di generare fault, ossia quelle che prevedono l’inserimento di input da parte dell’utente.

## 6.2 Testing di Integrazione

Per l’esecuzione del Test di Integrazione verrà utilizzato un approccio di tipo Bottom Up, scelto perché è la strategia che meglio si adatta a sistemi di tipo Object-oriented come il nostro. La scrittura delle classi di test avverrà tramite l’utilizzo del framework *JUnit,* mentre per effettuare lo Stub delle componenti si utilizzerà la funzionalità di Mock fornita dalla libreria *Mockito* . Verrà utilizzato *Maven* per l’esecuzione automatizzata delle classi di test ed infine per il calcolo della coverage del sistema verrà utilizzato *JaCoCo*.

Facendo riferimento al Diagramma Architetturale definito all’interno del System Design Document e che verrà riportato di seguito:



Il Test di Integrazione verrà effettuato a partire dal layer Storage dove verrà effettuata l’integrazione della classe service con la rispettiva componente DAO da cui dipende, dopodichè si salirà fino al layer Application dove verrà testata la classe controller integrando la componente service da cui dipende. La chiamata alla classe controller (e quindi la generazione della request che chiama la classe controller) verrà simulata utilizzando Mockito.

Il nome delle classi di test seguiranno la seguente regola: Presa una classe di nome *NomeClasse* , la rispettiva classe che effettuerà il test di integrazione sarà nominata *NomeClasseTestIntegrazione*.

## 6.3 Testing di Unità

Per l’esecuzione del Test di Unità si è deciso di utilizzare un approccio di tipo BlackBox, eseguendo quindi il testing delle operazioni che implementano le funzionalità individuate durante l’analisi dei requisiti e nello specifico dando priorità alle funzionalità individuate precedentemente tramite i Test Case che ,come abbiamo detto, sono state scelte tra quelle funzionalità con il più alto rischio di generare fault a causa della necessità dell’inserimento di input da parte dell’utente. Successivamente si proseguirà con il test delle restanti funzionalità di priorità alta e media.

Nello specifico i metodi di test che faranno riferimento ai Test Case individuati, avranno come nome del metodo di test il TestCaseID corrispondente, sostituendo al carattere “.” ,non utilizzabile all’interno del nome di un metodo, il carattere “p” .

Di seguito un esempio:

* TestCaseID: TC\_1.1\_1
* firma metodo: void TC\_1p1\_1();

Il nome delle classi di test seguiranno la seguente regola: Presa una classe di nome *NomeClasse* , la rispettiva classe che effettuerà il test di Unità sarà nominata *NomeClasseTest*.

Seguendo sempre la logica del Testing di tipo Bottom Up, verranno testate per unità le classi all’interno di ogni layer, prima di essere integrate. Verrà escluso il testing delle interfacce e delle classi di tipo Entity, queste ultime poiché fornite solo di metodi come getters, setters e toString.

Anche in questa fase come nella precedente è previsto:

* l’utilizzo del framework *JUnit;*
* l’utilizzo di *Mockito* per isolare la componente testata generando opportuni stub e driver;
* L’utilizzo di *Maven* e *JaCoCo* per l’esecuzione automatizzata dei test e la generazione del file di report e della branch coverage del sistema

1. Sospensione e ripristino

**Criteri di sospensione**

Il testing verrà eseguito fino alla sua completa terminazione anche nel caso in cui venissero generati dei failure. Solo nel caso in cui un errore nella definizione della classe di test dovesse generare un interruzione del test al momento dell’esecuzione, allora il test potrà essere momentaneamente sospeso per procedere alla correzione della classe di test.

**Criteri di ripristino**

Nel momento in cui gli errori individuati nelle classi di test sono stati risolti, il test dovrà riprendere la sua esecuzione.

1. Strumenti per il testing (hardware/software)

L’hardware utilizzato per il testing è un computer, in quanto il sistema non è stato ancora rilasciato.

1. Test cases

Per la definizione dei Test Frame sarà utilizzato l’approccio di Category Partition studiato. Verranno partizionati gli input in classi di equivalenza che permetteranno di ridurre il numero di test case trattati. L’output atteso sarà definito da un oracolo umano, derivato dall’analisi delle specifiche dei requisiti e dai casi d’uso.

## 9.1 Gestione Utente

### 9.1.1 Registrazione Utente

|  |  |  |
| --- | --- | --- |
| **Autore** | | **LM** |
| **Parametro: Nome** | | |
| **FORMATO:**  **^[ a-zA-Z\u00C0-\u00ff]{1,50}$** | | |
| **Nome Categoria** | **Scelte per categorie** | |
| Formato [FN] | 1. Rispetta il formato = false [error]  2. Rispetta il formato = true [PROPERTY FN\_OK] | |
| **Parametro: Cognome** | | |
| **FORMATO:**  **^[ a-zA-Z\u00C0-\u00ff]{1,50}$** | | |
| **Nome Categoria** | **Scelte per categorie** | | |
| Formato [FC] | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FC\_OK] | |
| **Parametro: Sesso** | | |
| **FORMATO:**  **“Uomo” “Donna” “Altro”** | | |
| **Nome Categoria** | **Scelte per categorie** | | |
| Formato [FS] | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FS\_OK] | |
| **Parametro: Data di nascita** | | |
| **FORMATO:**  **[GG/MM/AAAA]** | | |
| **Nome Categoria** | **Scelte per categorie** | | |
| Formato [FD] | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FD\_OK] | |
| Inserimento [DS] | 1. Precedente alla data odierna [if **formatoFDok** && dataDiNascita < dataOdierna] [property **inserimentoDSok**] 2. Uguale alla data odierna [if **formatoFDok** && dataDiNascita == dataOdierna] [error] 3. Successiva alla data odierna [if **formatoFDok** && dataDiNascita > dataOdierna] [error] | |
| **Parametro: E-mail** | | |
| **FORMATO:**  **^\\w+([\\.-]?\\w+)\*@\\w+([\\.-]?\\w+)\*(\\.\\w+){1,100}$** | | |
| **Nome Categoria** | **Scelte per categorie** | | |
| Formato [FE] | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FE\_OK] | |
| **Parametro: Password** | | |
| **FORMATO:**  **^(?=.\*[A-Z])(?=.\*\d)(?=.\*\W).{6,30}$** | | |
| **Nome Categoria** | **Scelte per categorie** | | |
| Formato [FP] | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FP\_OK] | |
| **Parametro: ConfermaPassword** | | |
| **FORMATO:**  **^(?=.\*[A-Z])(?=.\*\d)(?=.\*\W).{6,30}$** | | |
| **Nome Categoria** | **Scelte per categorie** | | |
| Match [MCP] | 1. Match con password= false[error] 2. Match con password= true [PROPERTY MP OK] | |
| Formato [FCP] | 1. Rispetta il formato = false [error]  2. Rispetta il formato = true [PROPERTY FPC OK] | |

|  |  |  |
| --- | --- | --- |
| **Test case ID** | **Test Frame** | **Esito** |
| TC\_1.1\_1 | FN1 | Errore: Nome non valido |
| TC\_1.1\_2 | FN2 , FC1 | Errore: Cognome non valido |
| TC\_1.1\_3 | FN2, FC2, FS1 | Errore: Sesso non valido |
| TC\_1.1\_4 | FN2, FC2, FS2, FD1 | Errore: Data di nascita non valida |
| TC\_1.1\_5 | FN2, FC2, FS2, FD2, DS2 | Errore: Data di nascita uguale alla data odierna |
| TC\_1.1\_6 | FN2, FC2, FS2, FD2, DS3 | Errore: Data di nascita successiva alla data odierna |
| TC\_1.1\_7 | FN2, FC2, FS2, FD2, DS1, FE1 | Errore: E-mail non valida |
| TC\_1.1\_8 | FN2, FC2, FS2, FD2, DS1, FE2, FP1 | Errore: Password non valida |
| TC\_1.1\_9 | FN2, FC2, FS2, FD2, DS1, FE2, FP2, MCP1 | Errore: Le password non coincidono |
| TC\_1.1\_10 | FN2, FC2, FS2, FD2, DS1, FE2, FP2, MCP2, FCP1 | Errore: conferma password non valido |
| TC\_1.1\_11 | FN2 FC2 FS2 FD2 DS1 FE2 FP2 MCP2 FCP2 | Corretto |

### 9.1.2 Modifica profilo utente

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Autore** | | **ML** |  | |
| **Parametro: Nome** | | |
| **FORMATO:**  **^[ a-zA-Z\u00C0-\u00ff]{1,50}$** | | |
| **Nome Categoria** | **Scelte per categorie** | |
| Formato [FN] | 1. Rispetta il formato = false [error]  2. Rispetta il formato = true [PROPERTY FN\_OK] | |
| **Parametro: Cognome** | | |
| **FORMATO:**  **^[ a-zA-Z\u00C0-\u00ff]{1,50}$** | | |
| **Nome Categoria** | **Scelte per categorie** | | |
| Formato [FC] | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FC\_OK] | |
| **Parametro: Sesso** | | |
| **FORMATO:**  **“Uomo” “Donna” “Altro”** | | |
| **Nome Categoria** | **Scelte per categorie** | | |
| Formato [FS] | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FS\_OK] | |
| **Parametro: Data di nascita** | | |
| **FORMATO:**  **[GG/MM/AAAA]** | | |
| **Nome Categoria** | **Scelte per categorie** | | |
| Formato [FD] | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FD\_OK] | |
| Inserimento [DS] | 1. Precedente alla data odierna [if **formatoFDok** && dataDiNascita < dataOdierna] [property **inserimentoDSok**] 2. Uguale alla data odierna [if **formatoFDok** && dataDiNascita == dataOdierna] [error] 3. Successiva alla data odierna [if **formatoFDok** && dataDiNascita > dataOdierna] [error] | |
| **Parametro: E-mail** | | |
| **FORMATO:**  **^\\w+([\\.-]?\\w+)\*@\\w+([\\.-]?\\w+)\*(\\.\\w+){1,100}$** | | |
| **Nome Categoria** | **Scelte per categorie** | | |
| Formato [FE] | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FE\_OK] | |
| **Parametro: VecchiaPassword** | | |
| **FORMATO:**  **^(?=.\*[A-Z])(?=.\*\d)(?=.\*\W).{6,30}$** | | |
| **Nome Categoria** | **Scelte per categorie** | | |
| Formato [FP] | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FP\_OK] | |
| **Paramentro: NuovaPassword** | | |
| **FORMATO:**  **^(?=.\*[A-Z])(?=.\*\d)(?=.\*\W).{6,30}$** | | |
| **Nome categoria** | | **Scelte per categorie** |
| Formato [FG] | | 1. Rispetta il formato = false [error]   Rispetta il formato = true [PROPERTY FP\_OK] |
| **Parametro: ConfermaNuovaPassword** | | |
| **FORMATO:**  **^(?=.\*[A-Z])(?=.\*\d)(?=.\*\W).{6,30}$** | | |
| **Nome Categoria** | **Scelte per categorie** | | |
| Match [MCP] | 1. Match con password= false[error] 2. Match con password= true [PROPERTY MP OK] | |
| Formato [FCP] | 1. Rispetta il formato = false [error]  2. Rispetta il formato = true [PROPERTY FPC OK] | |

|  |  |  |
| --- | --- | --- |
| **Test case ID** | **Test Frame** | **Esito** |
| TC\_1.2\_1 | FN1 | Errore: Nome non valido |
| TC\_1.2\_2 | FN2, FC1 | Errore: Cognome non valido |
| TC\_1.2\_3 | FN2, FC2, FS1 | Errore: Sesso non valido |
| TC\_1.2\_4 | FN2, FC2, FS2, FD1 | Errore: Data di nascita non valida |
| TC\_1.2\_5 | FN2, FC2, FS2, FD2, DS2 | Errore: Data di nascita uguale alla data odierna |
| TC\_1.2\_6 | FN2, FC2, FS2, FD2, DS3 | Errore: Data di nascita successiva alla data odierna |
| TC\_1.2\_7 | FN2, FC2, FS2, FD2, DS1, FE1 | Errore: E-mail non valida |
| TC\_1.2\_8 | FN2, FC2, FS2, FD2, DS1, FE2, FP1 | Errore: VecchiaPassword non valida |
| TC\_1.2\_9 | FN2, FC2, FS2, FD2, DS1, FE2, FP2, FG7 | Errore: NuovaPassword non valida |
| TC\_1.2\_10 | FN2, FC2, FS2, FD2, DS1, FE2, FP2, FG8, MCP1 | Errore: Le password non coincidono |
| TC\_1.2\_11 | FN2, FC2, FS2, FD2, DS1, FE2, FP2, FG8, MCP2, FCP1 | Errore: conferma password non valido |
| TC\_1.2\_12 | FN2, FC2, FS2, FD2, DS1, FE2, FP2, FG8, MCP2, FCP2 | Corretto |

### 9.1.3 Login

|  |  |  |  |
| --- | --- | --- | --- |
| |  |  | | --- | --- | | **Autore** | **LM** | | |
| **Parametro: E-mail** | |
| **FORMATO:**  **^\\w+([\\.-]?\\w+)\*@\\w+([\\.-]?\\w+)\*(\\.\\w+){1,100}$** | |
| **Nome Categoria** | **Scelte per categorie** |
| Match [ME] | 1. Match con email = false[error] 2. Match con email = true [PROPERTY ME OK] |
| **Parametro: Password** | |
| **FORMATO:**  **^(?=.\*[A-Z])(?=.\*\d)(?=.\*\W).{6,30}$** | |
| **Nome Categoria** | **Scelte per categorie** |
| Match [MP] | 1. Match con password= false[error] 2. Match con password= true [PROPERTY MP OK] |

|  |  |  |
| --- | --- | --- |
| **Test case ID** | **Test Frame** | **Esito** |
| TC\_1.3\_1 | ME1 | Errore: Email non corretta o non valida |
| TC\_1.3\_2 | ME2, MP1 | Errore: La password non è corretta o non valida |
| TC\_1.3\_3 | ME2, MP2 | Corretto |

### 9.1.4 Modifica Profilo Organizzatore

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Autore** | **LM** | | | |
| **Parametro: Nome** | | |
| **FORMATO:**  **^[ a-zA-Z\u00C0-\u00ff]{1,50}$** | | |
| **Nome Categoria** | **Scelte per categorie** | |
| Formato [FN] | 1. Rispetta il formato = false [error]  2. Rispetta il formato = true [PROPERTY FN\_OK] | |
| **Parametro: Cognome** | | |
| **FORMATO:**  **^[ a-zA-Z\u00C0-\u00ff]{1,50}$** | | |
| **Nome Categoria** | | **Scelte per categorie** |
| Formato [FC] | | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FC\_OK] |
| **Parametro: Sesso** | | |
| **FORMATO:**  **“Uomo” “Donna” “Altro”** | | |
| **Nome Categoria** | | **Scelte per categorie** |
| Formato [FS] | | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FS\_OK] |
| **Parametro: Data di nascita** | | |
| **FORMATO:**  **[GG/MM/AAAA]** | | |
| Formato [FD] | | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FD\_OK] |
| Inserimento [DS] | | 1. Precedente alla data odierna [if **formatoFDok** && dataDiNascita < dataOdierna] [property **inserimentoDSok**] 2. Uguale alla data odierna [if **formatoFDok** && dataDiNascita == dataOdierna] [errore] 3. Successiva alla data odierna [if **formatoFDok** && dataDiNascita > dataOdierna] [errore] |
| **Parametro: E-mail** | | |
| **FORMATO:**  **^\\w+([\\.-]?\\w+)\*@\\w+([\\.-]?\\w+)\*(\\.\\w+){1,100}$** | | |
| **Nome Categoria** | | **Scelte per categorie** |
| Formato [FE] | | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FE\_OK] |
| **Parametro: Password** | | |
| **FORMATO:**  **^(?=.\*[A-Z])(?=.\*\d)(?=.\*\W).{6,30}$** | | |
| **Nome Categoria** | | **Scelte per categorie** |
| Formato [FP] | | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FP\_OK] |
| **Parametro: Biografia** | | |
| **FORMATO:** | | |
| **Nome Categoria** | | **Scelte per categorie** |
| Lunghezza [LB] | | 1. Lunghezza<1 = false[error] 2. Lunghezza>=1 = true [PROPERTY LE OK] |
| **Parametro: Conferma Password** | | |
| **FORMATO:**  **^(?=.\*[A-Z])(?=.\*\d)(?=.\*\W).{6,30}$** | | |
| **Nome Categoria** | | **Scelte per categorie** |
| Match [MCP] | | 1. Match con password= false[error] 2. Match con password= true [PROPERTY LCP OK] |
| Formato [FCP] | | 1. Rispetta il formato = false [error]  2. Rispetta il formato = true [PROPERTY FCP\_OK] |
| **Parametro: IBAN** | | |
| **FORMATO:**  **^(it|IT)[0-9]{2}[A-Za-z][0-9]{10}[0-9A-Za-z]{12}$** | | |
| **Nome Categoria** | | **Scelte per categorie** |
| Formato [FI] | | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FE\_OK] |

|  |  |  |
| --- | --- | --- |
| **Test case ID** | **Test Frame** | **Esito** |
| TC\_1.4\_1 | FN1 | Errore: Nome utente non valido |
| TC\_1.4\_2 | FN2, FC1 | Errore: Cognome non valido |
| TC\_1.4\_3 | FN2, FC2, FS1 | Errore: Sesso non valido |
| TC\_1.4\_4 | FN2, FC2, FS2, FD1 | Errore: Data di nascita non valida |
| TC\_1.4\_5 | FN2, FC2, FS2, FD2, DS2 | Errore: Data di nascita uguale alla data odierna |
| TC\_1.4\_6 | FN2, FC2, FS2, FD2, DS3 | Errore: Data di nascita successiva alla data odierna |
| TC\_1.4\_7 | FN2, FC2, FS2, FD2, DS1, FE1 | Errore: Email non valida |
| TC\_1.4\_8 | FN2, FC2, FS2, FD2, DS1, FE2, FP1 | Errore: Password non valida |
| TC\_1.4\_9 | FN2, FC2, FS2, FD2, DS1, FE2, FP2, LB1 | Errore: La biografia non è valida |
| TC\_1.4\_10 | FN2, FC2, FS2, FD2, DS1, FE2, FP2, LB2, MCP1 | Errore: Le password non coincidono |
| TC\_1.4\_11 | FN2, FC2, FS2, FD2, DS1, FE2, FP2, LB2, MCP2, FCP1 | Errore: Il conferma password non è valido |
| TC\_1.4\_12 | FN2, FC2, FS2, FD2, DS1, FE2, FP2, LB2, MCP2, FCP2, FI1 | Errore: IBAN non valido |
| TC\_1.4\_13 | FN2 FC2 FS2 FD2 DS1 FE2 FP2 LB2 MCP2 FCP2 FI2 | Corretto |

### 9.1.5 Registrazione Organizzatore

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Autore** | **LM** | | | |
| **Parametro: Nome** | | |
| **FORMATO:**  **^[ a-zA-Z\u00C0-\u00ff]{1,50}$** | | |
| **Nome Categoria** | **Scelte per categorie** | |
| Formato [FN] | 1. Rispetta il formato = false [error]  2. Rispetta il formato = true [PROPERTY FN\_OK] | |
| **Parametro: Cognome** | | |
| **FORMATO:**  **^[ a-zA-Z\u00C0-\u00ff]{1,50}$** | | |
| **Nome Categoria** | **Scelte per categorie** | |
| Formato [FC] | | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FC\_OK] |
| **Parametro: Sesso** | | |
| **FORMATO:**  **“Uomo” “Donna” “Altro”** | | |
| Formato [FS] | | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FS\_OK] |
| **Parametro: Data di nascita** | | |
| **FORMATO:**  **[GG/MM/AAAA]** | | |
| Formato [FD] | | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FD\_OK] |
| Inserimento [DS] | | 1. Precedente alla data odierna [if **formatoFDok** && dataDiNascita < dataOdierna] [property **inserimentoDSok**] 2. Uguale alla data odierna [if **formatoFDok** && dataDiNascita == dataOdierna] [errore] 3. Successiva alla data odierna [if **formatoFDok** && dataDiNascita > dataOdierna] [errore] |
| **Parametro: E-mail** | | |
| **FORMATO:**  **^\\w+([\\.-]?\\w+)\*@\\w+([\\.-]?\\w+)\*(\\.\\w+){1,100}$** | | |
| Formato [FE] | | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FE\_OK] |
| **Parametro: Password** | | |
| **FORMATO:**  **^(?=.\*[A-Z])(?=.\*\d)(?=.\*\W).{6,30}$** | | |
| Formato [FP] | | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FP\_OK] |
| **Parametro: Biografia** | | |
| **FORMATO:** | | |
| Lunghezza [LB] | | 1. Lunghezza<1 = false[error] 2. Lunghezza>=1 = true [PROPERTY LE OK] |
| **Parametro: Conferma Password** | | |
| **FORMATO:**  **^(?=.\*[A-Z])(?=.\*\d)(?=.\*\W).{6,30}$** | | |
| Match [MCP] | | 1. Match con password= false[error] 2. Match con password= true [PROPERTY LCP OK] |
| Formato [FCP] | | 1. Rispetta il formato = false [error]  2. Rispetta il formato = true [PROPERTY FCP\_OK] |
| **Parametro: IBAN** | | |
| **FORMATO:**  **^(it|IT)[0-9]{2}[A-Za-z][0-9]{10}[0-9A-Za-z]{12}$** | | |
| Formato [FI] | | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FE\_OK] |

|  |  |  |
| --- | --- | --- |
| **Test case ID** | **Test Frame** | **Esito** |
| TC\_1.5\_1 | FN1 | Errore: Nome utente non valido |
| TC\_1.5\_2 | FN2, FC1 | Errore: Cognome non valido |
| TC\_1.5\_3 | FN2, FC2, FS1 | Errore: Sesso non valido |
| TC\_1.5\_4 | FN2, FC2, FS2, FD1 | Errore: Data di nascita non valida |
| TC\_1.5\_5 | FN2, FC2, FS2, FD2, DS2 | Errore: Data di nascita uguale alla data odierna |
| TC\_1.5\_6 | FN2, FC2, FS2, FD2, DS3 | Errore: Data di nascita successiva alla data odierna |
| TC\_1.5\_7 | FN2, FC2, FS2, FD2 , DS1, FE1 | Errore: Email non valida |
| TC\_1.5\_8 | FN2, FC2, FS2, FD2 , DS1, FP1 | Errore: Password non valida |
| TC\_1.5\_9 | FN2, FC2, FS2, FD2 , DS1, FP2, LB1 | Errore: La biografia non è valida |
| TC\_1.5\_10 | FN2, FC2, FS2, FD2 , DS1, FP2, LB2, MCP1 | Errore: Le password non coincidono |
| TC\_1.5\_11 | FN2, FC2, FS2, FD2 , DS1, FP2, LB2, MCP2, FCP1 | Errore: Il conferma password non è valido |
| TC\_1.5\_12 | FN2, FC2, FS2, FD2 , DS1, FP2, LB2, MCP2, FCP2, FI1 | Errore: IBAN non valido |
| TC\_1.5\_13 | FN2 FC2 FS2 FD2 DS1 FE2 FP2 LB2 MCP2 FCP2 FI2 | Corretto |

### 9.1.6 Modifica Profilo Amministratore

|  |  |  |  |
| --- | --- | --- | --- |
| |  |  | | --- | --- | | **Autore** | **ML** | | |
| **Parametro: Nome** | |
| **FORMATO:**  **^[ a-zA-Z\u00C0-\u00ff]{1,50}$** | |
| **Nome Categoria** | **Scelte per categorie** |
| Formato [FN] | 1. Rispetta il formato = false [error]  2. Rispetta il formato = true [PROPERTY FN\_OK] |
| **Parametro: Cognome** | |
| **FORMATO:**  **^[ a-zA-Z\u00C0-\u00ff]{1,50}$** | |
| **Nome Categoria** | **Scelte per categorie** |
| Formato [FC] | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FC\_OK] |
| **Parametro: E-mail** | |
| **FORMATO:**  **^\\w+([\\.-]?\\w+)\*@\\w+([\\.-]?\\w+)\*(\\.\\w+){1,100}$** | |
| **Nome Categoria** | **Scelte per categorie** |
| Formato [FE] | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FE\_OK] |
| **Parametro: Password** | |
| **FORMATO:**  **^(?=.\*[A-Z])(?=.\*\d)(?=.\*\W).{6,30}$** | |
| **Nome Categoria** | **Scelte per categorie** |
| Formato [FP] | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FP\_OK] |
| **Parametro: ConfermaPassword** | |
| **FORMATO:**  **^(?=.\*[A-Z])(?=.\*\d)(?=.\*\W).{6,30}$** | |
| **Nome Categoria** | **Scelte per categorie** |
| Match [MCP] | 1. Match con password= false[error] 2. Match con password= true [PROPERTY MP OK] |
| Formato [FCP] | 1. Rispetta il formato = false [error]  2. Rispetta il formato = true [PROPERTY FPC OK] |

|  |  |  |
| --- | --- | --- |
| **Test case ID** | **Test Frame** | **Esito** |
| TC\_1.6\_1 | FN1 | Errore: Nome non valido |
| TC\_1.6\_2 | FN2, FC3 | Errore: Cognome non valido |
| TC\_1.6\_3 | FN2, FC4, FS4, FD4, DS4, FE3 | Errore: E-mail non valida |
| TC\_1.6\_4 | FN2, FC4, FS4, FD4, DS4, FE4, FP3 | Errore: Password non valida |
| TC\_1.6\_5 | FN2, FC4, FS4, FD4, DS4, FE4, FP4, MCP3 | Errore: Le password non coincidono |
| TC\_1.6\_6 | FN2, FC4, FS4, FD4, DS4, FE4, FP4, MCP4, FCP1 | Errore: Il conferma password non è valido |
| TC\_1.6\_7 | FN2, FC4, FS4, FD4, DS4, FE4, FP4, MCP4, FCP2 | Corretto |

### 9.1.7 Registrazione Scolaresca

|  |  |  |  |
| --- | --- | --- | --- |
| |  |  | | --- | --- | | **Autore** | **ML** | | |
| **Parametro: Istituto** | |
| **FORMATO:**  **^[ a-zA-Z\u00C0-\u00ff]{1,100}$** | |
| **Nome Categoria** | **Scelte per categorie** |
| Formato [FI] | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FS\_OK] |
| **Parametro: E-mail** | |
| **FORMATO:**  **^\\w+([\\.-]?\\w+)\*@\\w+([\\.-]?\\w+)\*(\\.\\w+){1,100}$** | |
| **Nome Categoria** | **Scelte per categorie** |
| Formato [FE] | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FE\_OK] |
| **Parametro: Password** | |
| **FORMATO:**  **^(?=.\*[A-Z])(?=.\*\d)(?=.\*\W).{6,30}$** | |
| **Nome Categoria** | **Scelte per categorie** |
| Formato [FP] | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FP\_OK] |
| **Parametro: ConfermaPassword** | |
| **FORMATO:**  **^(?=.\*[A-Z])(?=.\*\d)(?=.\*\W).{6,30}$** | |
| **Nome Categoria** | **Scelte per categorie** |
| Match [MCP] | 1. Match con password= false[error] 2. Match con password= true [PROPERTY MP OK] |
| Formato [FCP] | 1. Rispetta il formato = false [error]  2. Rispetta il formato = true [PROPERTY FPC OK] |

|  |  |  |
| --- | --- | --- |
| **Test case ID** | **Test Frame** | **Esito** |
| TC\_1.7\_1 | FI1 | Errore: Istituto non valido |
| TC\_1.7\_2 | FI2, FE1 | Errore: Email non valida |
| TC\_1.7\_3 | FI2, FE2, FP1 | Errore: Password non valida |
| TC\_1.7\_4 | FI2, FE2, FP2, MCP1 | Errore: Le password non coincidono |
| TC\_1.7\_5 | FI2, FE2, FP2, MCP2, FCP1 | Errore: Conferma password non valida |
| TC\_1.7\_6 | FI2, FE2, FP2, MCP2, FCP2 | Corretto |

### 9.1.8 Modifica profilo scolaresca

|  |  |  |  |
| --- | --- | --- | --- |
| |  |  | | --- | --- | | **Autore** | **ML** | | |
| **Parametro: Istituto** | |
| **FORMATO:**  **^[ a-zA-Z\u00C0-\u00ff]{1,100}$** | |
| **Nome Categoria** | **Scelte per categorie** |
| Formato [FI] | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FS\_OK] |
| **Parametro: E-mail** | |
| **FORMATO:**  **^\\w+([\\.-]?\\w+)\*@\\w+([\\.-]?\\w+)\*(\\.\\w+){1,100}$** | |
| **Nome Categoria** | **Scelte per categorie** |
| Formato [FE] | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FE\_OK] |
| **Parametro: Password** | |
| **FORMATO:**  **^(?=.\*[A-Z])(?=.\*\d)(?=.\*\W).{6,30}$** | |
| **Nome Categoria** | **Scelte per categorie** |
| Formato [FP] | 1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FP\_OK] |
| **Parametro: ConfermaPassword** | |
| **FORMATO:**  **^(?=.\*[A-Z])(?=.\*\d)(?=.\*\W).{6,30}$** | |
| **Nome Categoria** | **Scelte per categorie** |
| Match [MCP] | 1. Match con password= false[error] 2. Match con password= true [PROPERTY MP OK] |
| Formato [FCP] | 1. Rispetta il formato = false [error]  2. Rispetta il formato = true [PROPERTY FPC OK] |

|  |  |  |
| --- | --- | --- |
| **Test case ID** | **Test Frame** | **Esito** |
| TC\_1.8\_1 | FI1 | Errore: Istituto non valido |
| TC\_1.8\_2 | FI2, FE1 | Errore: Email non valida |
| TC\_1.8\_3 | FI2, FE2, FP1 | Errore: Password non valida |
| TC\_1.8\_4 | FI2, FE2, FP2, MCP1 | Errore: Le password non coincidono |
| TC\_1.8\_5 | FI2, FE2, FP2, MCP2, FCP1 | Errore: Conferma password non valida |
| TC\_1.8\_6 | FI2, FE2, FP2, MCP2, FCP2 | Corretto |

## 9.2 Gestione Eventi

### 9.2.1 Richiesta inserimento Evento

|  |  |  |  |
| --- | --- | --- | --- |
| |  |  | | --- | --- | | **Autore** | **DPA** | | |
| **Parametro: Tipo Evento** | |
| **FORMATO: è una di queste stringhe: “mostra”, “teatro”** | |
| **Nome Categoria** | **Scelte per categorie** |
| FORMATO [FTE] | 1. Rispetta il formato = False [error] 2. Rispetta il formato= True [PROPERTY FTE\_OK] |
| **Parametro: Nome Evento** | |
| **FORMATO: ^[0-9°A-zÀ-ù ‘-]{1,50}$** | |
| **Nome Categoria** | **Scelte per categorie** |
| FORMATO [FNE] | 1. Rispetta il formato = False [error] 2. Rispetta il formato= True [PROPERTY FNE\_OK] |
| **Parametro: Descrizione Evento** | |
| **FORMATO: ^[0-9°A-zÀ-ù ‘-]{1,320}$** | |
| |  |  | | --- | --- | | **Nome Categoria** | **Scelte per categorie** | | |
| FORMATO[FDE] | 1. Rispetta il formato = False [error] 2. Rispetta il formato= True [PROPERTY FDE\_OK] |
| **Parametro: Foto Evento** | |
| **Nome Categoria** | **Scelte per categorie** |
| Grandezza File [GFFE] | 1. Grandezza > 15MB = False [error] 2. Grandezza <= 15MB = True [PROPERTY GFFE\_OK] |
| Dimensione [DFE] | 1. Dimensione > 1024x768 pixel = False [error] 2. Dimensione <= 1024x768 pixel= True [PROPERTY DFE\_OK] |
| **Parametro: Numero Biglietti** | |
| **FORMATO: ^([1-9]+[.,][0-9]+)$** | |
| **Nome Categoria** | **Scelte per categorie** |
| FORMATO [FNB] | * 1. Rispetta il formato= False [error]   2. Rispetta il formato= True [PROPERTY FNB\_OK] |
| **Parametro: Prezzo Biglietti** | |
| **FORMATO: ^([0-9]+[.,]\*)+** | |
| **Nome Categoria** | **Scelte per categorie** |
| FORMATO [FPB] | 1. Rispetta il formato= False [error] 2. Rispetta il formato= True [PROPERTY FPB\_OK] |
| **Parametro: Data inizio** | |
| **FORMATO:**  **gg/mm/aaaa** | |
| **Nome Categoria** | **Scelte per categorie** |
| FORMATO [FDI] | 1. Rispetta il formato= False [error] 2. Rispetta il formato= True [PROPERTY FDI\_OK] |
| VINCOLO [VDI] | 1. DataInizio<=DataAttuale =False [error] 2. DataInizio >DataAttuale = True [PROPERTY VDI\_OK] |
| **Parametro: Data fine** | |
| **FORMATO: gg/mm/aaaa** | |
| **Nome Categoria** | **Scelte per categorie** |
| FORMATO [FDF] | 1. Rispetta il formato= False [error] 2. Rispetta il formato= True [PROPERTY FDF\_OK] |
| VINCOLO [VDF] | 1. DataFine <= DataInizio =False [error] 2. DataFine >DataInizio = True [PROPERTY VDF\_OK] |
| **Parametro: Sede** | |
| **FORMATO: ^[0-9°A-zÀ-ù ‘-]{1,100}$** | |
| **Nome Categoria** | **Scelte per categorie** |
| FORMATO [FSE] | 1. Rispetta il formato= False [error] 2. Rispetta il formato= True [PROPERTY FSE\_OK] |
| **Parametro: Indirizzo** | |
| **FORMATO: ^[0-9°A-zÀ-ù ‘-]{2,30}$** | |
| **Nome Categoria** | **Scelte per categorie** |
| FORMATO [FIND] | 1. Rispetta il formato= False [error] 2. Rispetta il formato=True [PROPERTY FIND\_OK] |

|  |  |  |
| --- | --- | --- |
| **Test case ID** | **Test Frame** | **Esito** |
| TC\_2.1\_1 | FTE1 | Errore: il Tipo Evento non rispetta il formato |
| TC\_2.1\_2 | FTE2, FNE1 | Errore: il Nome Evento non rispetta il formato |
| TC\_2.1\_3 | FTE2, FNE2, FDE1 | Errore: la Descrizione Evento non rispetta il formato |
| TC\_2.1\_4 | FTE2, FNE2, FDE2, GFFE1 | Errore: il file Foto Evento ha una grandezza superiore ai 15MB |
| TC\_2.1\_5 | FTE2, FNE2, FDE2, GFFE2, DFE1 | Errore: il file Foto Evento ha una dimensione maggiore di 1024x768 pixel |
| TC\_2.1\_6 | FTE2, FNE2, FDE2, GFFE2, DFE2, FNB1 | Errore: il Numero Biglietti non rispetta il formato |
| TC\_2.1\_7 | FTE2, FNE2, FDE2, GFFE2, DFE2, FNB2, FPB1 | Errore: il Prezzo Biglietti non rispetta il formato |
| TC\_2.1\_8 | FTE2, FNE2, FDE2, GFFE2, DFE2, FNB2, FPB2, FDI1 | Errore, la Data Inizio non rispetta il formato |
| TC\_2.1\_9 | FTE2, FNE2, FDE2, GFFE2, DFE2, FNB2, FPB2, FDI2, VDI1 | Errore, la Data Inizio inserita è precedente o contemporanea alla Data Attuale |
| TC\_2.1\_10 | FTE2, FNE2, FDE2, GFFE2, DFE2, FNB2, FPB2, FDI2, VDI2, FDF1 | Errore, la Data Fine non rispetta il formato |
| TC\_2.1\_11 | FTE2, FNE2, FDE2, GFFE2, DFE2, FNB2, FPB2, FDI2, VDI2, FDF2, VDF1 | Errore, la Data Fine inserita è precedente o contemporanea alla Data Inizio |
| TC\_2.1\_12 | FTE2, FNE2, FDE2, GFFE2, DFE2, FNB2, FPB2, FDI2, VDI2, FDF2, VDF2, FSE1 | Errore, la Sede non rispetta il formato |
| TC\_2.1\_13 | FTE2, FNE2, FDE2, GFFE2, DFE2, FNB2, FPB2, FDI2, VDI2, FDF2, VDF2, FSE2, FIND1 | Errore, l’indirizzo non rispetta il formato |
| TC\_2.1\_14 | FTE2, FNE2, FDE2, GFFE2, DFE2, FNB2, FPB2, FDI2, VDI2, FDF2, VDF2, FSE2, FIND2 | Corretto |

### 9.2.2 Richiesta modifica Evento

|  |  |  |  |
| --- | --- | --- | --- |
| |  |  | | --- | --- | | **Autore** | **DPA** | | |
| **Parametro: Tipo Evento** | |
| **FORMATO: è una di queste stringhe: “mostra”, “teatro”** | |
| **Nome Categoria** | **Scelte per categorie** |
| FORMATO [FTE] | 1. Rispetta il formato = False [error] 2. Rispetta il formato= True [PROPERTY FTE\_OK] |
| **Parametro: Nome Evento** | |
| **FORMATO: ^[0-9°A-zÀ-ù ‘-]{1,50}$** | |
| **Nome Categoria** | **Scelte per categorie** |
| FORMATO [FNE] | 1. Rispetta il formato = False [error] 2. Rispetta il formato= True [PROPERTY FNE\_OK] |
| **Parametro: Descrizione Evento** | |
| **FORMATO: ^[0-9°A-zÀ-ù ‘-]{1,320}$** | |
| |  |  | | --- | --- | | **Nome Categoria** | **Scelte per categorie** | | |
| FORMATO[FDE] | 1. Rispetta il formato = False [error] 2. Rispetta il formato= True [PROPERTY FDE\_OK] |
| **Parametro: Foto Evento** | |
| **Nome Categoria** | **Scelte per categorie** |
| Grandezza File [GFFE] | 1. Grandezza > 15MB = False [error] 2. Grandezza <= 15MB = True [PROPERTY GFFE\_OK] |
| Dimensione [DFE] | 1. Dimensione > 1024x768 pixel = False [error] 2. Dimensione <= 1024x768 pixel= True [PROPERTY DFE\_OK] |
| **Parametro: Numero Biglietti** | |
| **FORMATO: ^[0-9]+$** | |
| **Nome Categoria** | **Scelte per categorie** |
| FORMATO [FNB] | 1. Rispetta il formato= False [error] 2. Rispetta il formato= True [PROPERTY FNB\_OK] |
| **Parametro: Prezzo Biglietti** | |
| **FORMATO: ^([0-9]+[.,]\*)+** | |
| **Nome Categoria** | **Scelte per categorie** |
| FORMATO [FPB] | 1. Rispetta il formato= False [error] 2. Rispetta il formato= True [PROPERTY FPB\_OK] |
| **Parametro: Data inizio** | |
| **FORMATO:**  **gg/mm/aaaa** | |
| **Nome Categoria** | **Scelte per categorie** |
| FORMATO [FDI] | 1. Rispetta il formato= False [error] 2. Rispetta il formato= True [PROPERTY FDI\_OK] |
| **Parametro: Data fine** | |
| **FORMATO: gg/mm/aaaa** | |
| **Nome Categoria** | **Scelte per categorie** |
| FORMATO [FDF] | 1. Rispetta il formato= False [error] 2. Rispetta il formato= True [PROPERTY FDF\_OK] |
| VINCOLO [VDF] | 1. DataFine <= DataInizio =False [error] 2. DataFine >DataInizio = True [PROPERTY VDF\_OK] |
| **Parametro: Sede** | |
| **^[0-9°A-zÀ-ù ‘-]{1,100}$** | |
| **Nome Categoria** | **Scelte per categorie** |
| FORMATO [FSE] | 1. Rispetta il formato= False [error] 2. Rispetta il formato= True [PROPERTY FSE\_OK] |
| **Parametro: Indirizzo** | |
| **FORMATO: ^[0-9°A-zÀ-ù ‘-]2,30}$** | |
| **Nome Categoria** | **Scelte per categorie** |
| FORMATO [FIND] | 1. Rispetta il formato= False [error] 2. Rispetta il formato=True [PROPERTY FIND\_OK] |

|  |  |  |
| --- | --- | --- |
| **Test case ID** | **Test Frame** | **Esito** |
| TC\_2.2\_1 | FTE1 | Errore: il Tipo Evento non rispetta il formato |
| TC\_2.2\_2 | FTE2, FNE1 | Errore: il Nome Evento non rispetta il formato |
| TC\_2.2\_3 | FTE2, FNE2, FDE1 | Errore: la Descrizione Evento non rispetta il formato |
| TC\_2.2\_4 | FTE2, FNE2, FDE2, GFFE1 | Errore: il file Foto Evento ha una grandezza superiore ai 15MB |
| TC\_2.2\_5 | FTE2, FNE2, FDE2, GFFE2, DFE1 | Errore: il file Foto Evento ha una dimensione maggiore di 1024x768 pixel |
| TC\_2.2\_6 | FTE2, FNE2, FDE2, GFFE2, DFE2, FNB1 | Errore: il Numero Biglietti non rispetta il formato |
| TC\_2.2\_7 | FTE2, FNE2, FDE2, GFFE2, DFE2, FNB2, FPB1 | Errore: il Prezzo Biglietti non rispetta il formato |
| TC\_2.2\_8 | FTE2, FNE2, FDE2, GFFE2, DFE2, FNB2, FPB2, FDI1 | Errore, la Data Inizio non rispetta il formato |
| TC\_2.2\_9 | FTE2, FNE2, FDE2, GFFE2, DFE2, FNB2, FPB2, FDI2, FDF1 | Errore, la Data Fine non rispetta il formato |
| TC\_2.2\_10 | FTE2, FNE2, FDE2, GFFE2, DFE2, FNB2, FPB2, FDI2, FDF2, VDF1 | Errore, la Data Fine inserita è precedente o contemporanea alla Data Inizio |
| TC\_2.2\_11 | FTE2, FNE2, FDE2, GFFE2, DFE2, FNB2, FPB2, FDI2, FDF2, VDF2, FSE1 | Errore, la Sede non rispetta il formato |
| TC\_2.2\_12 | FTE2, FNE2, FDE2, GFFE2, DFE2, FNB2, FPB2, FDI2, FDF2, VDF2, FSE2, FIND1 | Errore, l’indirizzo non rispetta il formato |
| TC\_2.2\_13 | FTE2, FNE2, FDE2, GFFE2, DFE2, FNB2, FPB2, FDI2, FDF2, VDF2, FSE2, FIND2 | Corretto |

### 9.2.3 Ricerca Evento

|  |  |  |  |
| --- | --- | --- | --- |
| |  |  | | --- | --- | | **Autore** | **DPA** | | |
| **Parametro: Testo Ricerca** | |
| **FORMATO: ^[0-9°A-zÀ-ù ‘-]\*\\*$** | |
| **Nome Categoria** | **Scelte per categorie** |
| FORMATO [FTR] | 1. Rispetta il formato= False [error] 2. Rispetta il formato=True [PROPERTY FTR\_OK] |

|  |  |  |
| --- | --- | --- |
| **Test case ID** | **Test Frame** | **Esito** |
| TC\_2.3\_1 | FTR1 | Errore: il Testo Ricerca non rispetta il formato |
| TC\_2.3\_2 | FTR2 | Corretto |

## 9.3 Gestione Carrello

### 9.3.1 Aggiungi al carrello

|  |  |  |  |
| --- | --- | --- | --- |
| |  |  | | --- | --- | | **Autore** | **LM** | | |
| **Parametro: Quantità** | |
| **FORMATO** | |
| **Nome Categoria** | **Scelte per categorie** |
| Quantità [QC] | 1. Quantita < 1 = false [error]  2. 1<Quantità<numBiglietti = true [PROPERTY QA\_OK] |

|  |  |  |
| --- | --- | --- |
| **Test case ID** | **Test Frame** | **Esito** |
| TC\_3.1\_1 | QC1 | Errore: Quantità non valida |
| TC\_3.1\_2 | QC2 | Corretto |

### 9.3.2 Modifica Carrello

|  |  |  |  |
| --- | --- | --- | --- |
| |  |  | | --- | --- | | **Autore** | **LM** | | |
| **Parametro: Quantità** | |
| **FORMATO:** | |
| **Nome Categoria** | **Scelte per categorie** |
| Quantità [QC] | 1. Quantita < 1 = false [error]  2. 1<Quantità<numBiglietti = true [PROPERTY QC\_OK] |

|  |  |  |
| --- | --- | --- |
| **Test case ID** | **Test Frame** | **Esito** |
| TC\_3.2\_1 | QC1 | Errore: Quantità non valida |
| TC\_3.2\_2 | QC2 | Corretto |

1. Testing schedule

Le attività di pianificazione del testing avverranno successivamente alla stesura dei documenti di System Design e Object Design.

La scrittura dei casi di test avverrà in contemporanea con lo sviluppo del codice.

L’esecuzione dei test verrà svolta durante l’implementazione del sistema e successivamente verranno rieseguiti tutti i casi di test quando l’implementazione del sistema sarà terminata per garantirne il corretto funzionamento.