



# Test Plan

## C03 – Lumen

Riferimento	2025_C03_TP
Versione	1.0
Data	4/12/2025
Destinatario	Prof.ssa Filomena Ferrucci, Prof. Fabio Palomba
Presentato da	<p>C03 Team Lumen:</p> <ul style="list-style-type: none"><li>• Luca Afeltra (LA)</li><li>• Michele Chierchia (MC)</li><li>• Alessandro Cigliano (AC)</li><li>• Matteo De Stasio (MDS)</li><li>• Vittorio Denysenko (VD)</li><li>• Giovanni Di Rosa (GR)</li><li>• Marianna Diograzia (MD)</li><li>• Felicia Riccio (FR)</li><li>• Maurizio Santangelo (MS)</li><li>• Gloria Scarallo (GS)</li><li>• Giovanni Pio Scardone (GPS)</li></ul>
Approvato da	



## Revision History

Data	Versione	Descrizione	Autori
04/12/2025	0.1	Aggiunta sospensione e ripristino	FR
05/12/2025	0.2	Aggiunta Panoramica del sistema	MD, MS
06/12/2025	0.3	Aggiunta Feature da testare/da non testare	MDS
06/12/2025	0.4	Aggiunta Pass/Fail criteria	VD
06/12/2025	0.5	Aggiunta del Testing Schedule	MD
06/12/2025	0.6	Aggiunta delle Responsabilità	GR
06/12/2025	0.7	Aggiunta delle Relazioni con altri documenti	GS
06/12/2025	0.8	Aggiunta dell'Approccio (Test di unità)	LA
06/12/2025	0.9	Aggiunta del Materiale di Testing	LA
06/12/2025	0.10	Aggiunta dell'introduzione	GPS
07/12/2025	0.11	Aggiunta dei Test Cases	Tutti
15/12/2025	1.0	Approvazione del documento	Tutti



## Sommario

---

<b>REVISION HISTORY</b>	<b>2</b>
<b>SOMMARIO</b>	<b>3</b>
<b>TEAM MEMBERS</b>	<b>5</b>
<b>1. INTRODUZIONE</b>	<b>6</b>
<b>2. RELAZIONE CON ALTRI DOCUMENTI</b>	<b>6</b>
<b>3. PANORAMICA DEL SISTEMA</b>	<b>7</b>
<b>4. FEATURES DA TESTARE/DA NON TESTARE</b>	<b>8</b>
<b>5. PASS/FAIL CRITERIA</b>	<b>9</b>
<b>6. APPROCCIO (TEST DI UNITÀ)</b>	<b>9</b>
<b>7. SOSPENSIONE E RIPRISTINO</b>	<b>10</b>
CRITERI DI SOSPENSIONE	10
CRITERI DI RIPRISTINO	10
<b>8. RESPONSABILITÀ</b>	<b>10</b>
<b>9. MATERIALI DI TESTING</b>	<b>10</b>
<b>10. TEST CASES</b>	<b>11</b>
10.1 GESTIONE ACCOUNT	11
10.1.1 Registrazione	11
10.1.2 Visualizzazione delle informazioni del profilo	14
10.1.3 Modifica delle informazioni del profilo	14
10.2 GESTIONE CONTENUTI	17
10.2.1 Aggiunta di un racconto	17
10.3 GESTIONE VOLONTARIATO	18
10.3.1 Partecipazione ad un evento	18
10.3.2 Creazione di un evento	18
10.3.3 Modifica di un evento	20
10.3.4 Avviare una raccolta fondi	22
10.4 GESTIONE RICHIESTE	24
10.4.1 Richiedere affiliazione	24
10.4.2 Accettare richiesta di affiliazione	25



**Lumen**

*Insieme per un futuro luminoso.*

10.5 GESTIONE RICERCA	26
10.5.1 Eseguire la ricerca di un utente per nome	26
<b>11 TESTING SCHEDULE</b>	<b>26</b>



## Team Members

Nome	Acronimo	Ruolo nel Progetto	Informazioni di contatto
<b>Rosa Carotenuto</b>	RC	Project Manager	r.carotenuto16@studenti.unisa.it
<b>Luigi Guida</b>	LG	Project Manager	l.guida15@studenti.unisa.it
<b>Luca Afeltra</b>	LA	Team Member	l.afeltra4@studenti.unisa.it
<b>Michele Chierchia</b>	MC	Team Member	m.chierchia13@studenti.unisa.it
<b>Alessandro Cigliano</b>	AC	Team Member	a.cigliano1@studenti.unisa.it
<b>Matteo De Stasio</b>	MDS	Team Member	m.destasio3@studenti.unisa.it
<b>Vittorio Denysenko</b>	VD	Team Member	v.denysenko@studenti.unisa.it
<b>Giovanni Di Rosa</b>	GR	Team Member	g.dirosa2@studenti.unisa.it
<b>Marianna Diograzia</b>	MD	Team Member	m.diograzia@studenti.unisa.it
<b>Felicia Riccio</b>	FR	Team Member	f.riccia14@studenti.unisa.it
<b>Maurizio Santangelo</b>	MS	Team Member	m.santangelo11@studenti.unisa.it
<b>Gloria Scarallo</b>	GS	Team Member	g.scarallo1@studenti.unisa.it
<b>Giovanni Pio Scardone</b>	GPS	Team Member	g.scardone@studenti.unisa.it



## 1. Introduzione

---

Lumen si propone di creare una piattaforma digitale semplice ed efficiente dedicata al mondo del volontariato, con l'obiettivo di ottimizzare e semplificare l'incontro tra i bisognosi e coloro che offrono aiuto. La piattaforma è progettata per gestire in modo fluido l'interazione tra tre attori fondamentali: gli enti benefici, che organizzano eventi e offrono supporto; i volontari, risorsa essenziale per le attività di volontariato; e i beneficiari, ovvero coloro che effettuano richieste di aiuto.

Il documento di Test Plan ha l'obiettivo di descrivere ed analizzare le attività di testing pianificate per la piattaforma Lumen, al fine di assicurarsi che ogni funzionalità e aspetto del sistema funzionino correttamente.

All'interno del documento sono riportate le strategie di testing adottate, le specifiche funzionalità che saranno sottoposte a verifica e gli strumenti scelti per la rilevazione e correzione degli errori, con lo scopo di rilasciare una piattaforma priva di malfunzionamenti e capace di rispondere efficacemente alle necessità degli utenti.

## 2. Relazione con altri documenti

---

I casi di test sono stati definiti sulla base delle funzionalità del sistema e fanno riferimento ai documenti prodotti nelle fasi precedenti del progetto.

### **Relazione con il Requirements Analysis Document (RAD)**

I test case riportati nel Test Plan derivano dai requisiti funzionali e non funzionali descritti nel RAD.

### **Relazione con il System Design Document (SDD)**

I test case sono coerenti con la suddivisione in sottosistemi e componenti definita nell'SDD.

Le verifiche previste rispettano quindi l'architettura logica individuata nella fase di design e garantiscono che le interazioni tra le parti del sistema si comportino come progettato.

### **Relazione con il Design Pattern Document (DPD)**

I casi di test sono inoltre allineati con le scelte progettuali descritte nel Design Pattern Document. In particolare, i test tengono conto dei pattern architettonici e strutturali adottati



## 3. Panoramica del sistema

---

Il sistema proposto si basa su un'architettura web di tipo three-tier, che garantisce una chiara separazione tra livello di presentazione, logica applicativa e livello di persistenza dei dati.

Questa organizzazione consente di ottenere un sistema maggiormente manutenibile, testabile e scalabile nel tempo.

Per la parte di front-end e la generazione delle view verranno utilizzati:

- React (con Vite) per la realizzazione di un'interfaccia utente reattiva, che facilita il riuso dei componenti e la gestione dello stato;
- HTML5 per la struttura semantica delle pagine;
- CSS3 per la definizione dello stile, in coerenza con le linee guida grafiche;
- JavaScript per la gestione della logica lato client e delle interazioni dinamiche con l'utente.

Per quanto riguarda la logica applicativa di back-end verranno utilizzati:

- Spring Boot
- Maven

Per la gestione della persistenza dei dati verrà utilizzato il DBMS PostgreSQL, soluzione robusta e scalabile che consente una memorizzazione affidabile delle informazioni del sistema.



## 4. Features da testare/da non testare

---

### Gestione Account

- Registrazione account
- Modifica informazioni del profilo
- Visualizzazione delle informazioni del profilo

### Gestione Contenuti

- Aggiunta di un racconto nella bacheca delle storie

### Gestione Volontariato

- Partecipazione ad un evento
- Creazione di un evento
- Modifica di un evento
- Avviare una raccolta fondi

### Gestione Richieste

- Richiedere affiliazione
- Accettare richiesta di affiliazione

### Gestione Ricerca

- Eseguire la ricerca di un Utente per nome

Le attività di testing non coinvolgeranno le funzionalità correlate a requisiti funzionali di bassa o media priorità.

Il requisito funzionale “Ricerca Geografica” (che rappresenta il modulo di intelligenza artificiale), che permette all’utente di ricercare il volontario e/o l’ente in base alla propria posizione geografica, facente parte della Gestione ricerca, verrà testato in futuro con differenti tecniche e sarà quindi trattato in una diversa documentazione.



## 5. Pass/Fail criteria

---

L'obiettivo del testing è individuare i difetti (faults) presenti nel sistema per poterli correggere. Ogni test viene valutato confrontando l'output effettivo del sistema con un risultato previsto (l'oracolo), basato sui requisiti funzionali.

Un test ha successo (pass) se, dato un input al sistema, l'output ottenuto è diverso dall'output atteso dall'oracolo.

Un test fallisce (fail) se, dato un input al sistema, l'output ottenuto è uguale all'output atteso dall'oracolo.

Tutto il testing sarà considerato valido se tutti i seguenti vincoli saranno rispettati:

- Tutti i requisiti funzionali ad alta priorità devono essere testati.

## 6. Approccio (Test di unità)

---

Nel paragrafo dedicato ai Test di Unità del Test Plan, l'attenzione è rivolta alla verifica delle interazioni tra i componenti service e DAO (Data Access Object), elementi fondamentali per la gestione e la persistenza dei dati all'interno dell'architettura software.

I test di unità hanno l'obiettivo di isolare e validare le singole unità di codice, concentrandosi nello specifico sul corretto comportamento dei service nei confronti dei DAO. In particolare, tali test mirano a garantire che i service invochino i metodi dei DAO in maniera corretta, coerente e conforme alle specifiche, assicurando l'accuratezza delle operazioni di accesso e manipolazione dei dati.

Per raggiungere questo obiettivo, viene adottata la tecnica del **mocking** dei DAO. Il mocking consente di simulare il comportamento dei componenti di accesso ai dati mediante versioni controllabili e semplificate, permettendo di testare i service in completo isolamento dagli strati inferiori dell'architettura, quali database o servizi esterni. Questa strategia riduce la complessità dei test e rende più affidabile l'individuazione di eventuali anomalie nel comportamento dei service.

L'implementazione dei test avviene tramite il framework di testing **JUnit**. L'esecuzione automatica dei test e il processo di build sono gestiti attraverso strumenti di automazione come **Maven**, garantendo ripetibilità e integrazione continua del processo di testing.

Al fine di aumentare ulteriormente la qualità del codice, vengono adottati strumenti di analisi statica e formattazione. In particolare, per il backend si fa affidamento sui controlli forniti da **Checkstyle**, mentre per il frontend vengono utilizzati i tool **ESLint** e **Prettier**, assicurando il rispetto degli standard di codifica, la leggibilità del codice e una maggiore manutenibilità complessiva del software.



## 7. Sospensione e ripristino

---

In questa sezione verranno definiti i criteri di sospensione del test e le attività di test che dovranno essere ripetute quando si riprende il test.

### Criteri di sospensione

Il testing non verrà sospeso fino alla sua terminazione, anche in caso di rilevazione di una failure. Il testing potrà essere temporaneamente sospeso solo nel caso in cui, durante l'esecuzione, si verifichi un errore legato alla definizione di uno dei test case stessi.

### Criteri di ripristino

Il testing riprenderà solo dopo l'individuazione e correzione dei fault.

## 8. Responsabilità

---

I team members si divideranno in due sottogruppi che si occuperanno rispettivamente dello sviluppo del lato backend e del lato frontend.

**Degli undici team members che compongono il team, verranno assegnati cinque team members al lato backend e sei team members al frontend.**

Ogni team member conoscerà tutto il lato di testing del sistema realizzato.

## 9. Materiali di testing

---

Le risorse che verranno utilizzate per l'esecuzione del testing si compongono di materiale redatto in precedenza, risorse hardware e risorse software.

Per quanto concerne il materiale redatto, le risorse utilizzate saranno il Requirements Analysis Document, il System Design Document e Design Pattern Document che identificheranno le componenti da testare.

Relativamente alla componente hardware, è necessario il semplice utilizzo di un computer.

Infine, altre risorse che verranno adoperate, saranno le **Github Actions**, fornite da Github, per l'integrazione continua (CI/CD).



## 10. Test cases

### 10.1 Gestione Account

#### 10.1.1 Registrazione

Parametro: Nome	
Nome Categoria	Scelte per la categoria
Lunghezza [LN]	Lunghezza >100 = false[error] Lunghezza <=100 = true [PROPERTY LN_OK]
Parametro: Cognome	
Nome Categoria	Scelte per la categoria
Lunghezza [LC]	Lunghezza >100 = false[error] Lunghezza <=100 = true [PROPERTY LC_OK]
Parametro: Email	
Formato	
$^{\text{[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.\text{[a-zA-Z]}\{2,\}}$}$	
Nome Categoria	Scelte per la categoria
Formato [FE]	Rispetta il formato = false [error] Rispetta il formato = true [PROPERTY FE_OK]
Parametro: Password	
Nome Categoria	Scelte per la categoria
Lunghezza [LP]	Lunghezza <= 0 OR Lunghezza >255 = false [error] . 0 < Lunghezza >= 255 = true [PROPERTY LP_OK]
Parametro: Numero Telefonico	
Formato	
$^{\text{d}\{10\}}$$	
Nome Categoria	Scelte per la categoria
Formato [FNT]	Rispetta il formato = false [error]



# Lumen

Insieme per un futuro luminoso.

	Rispetta il formato = true [PROPERTY FNT_OK]
<b>Parametro: Ruolo</b>	
<b>Formato</b> ^(Volontario   Beneficiario   Ente)\$	
<b>Nome Categoria</b>	<b>Scelte per la categoria</b>
Formato [FR]	Rispetta il formato = false [error] Rispetta il formato = true [PROPERTY FR_OK]
<b>Parametro: Ambito</b>	
<b>Nome Categoria</b>	<b>Scelta Categoria</b>
Lunghezza [LA]	Lunghezza >100 = false[error] Lunghezza <=100 = true [PROPERTY LN_OK]
Lunghezza [LA]	Lunghezza >100 = false[error] Lunghezza <=100 = true [PROPERTY LP_OK]
<b>Parametro: Immagine</b>	
<b>Formato</b> ^data:image/[a-zA-Z0-9+-]+;base64,([a-zA-Z0-9+/]{4})*([a-zA-Z0-9+/]{3}= [a-zA-Z0-9+/]{2}==)?\$	
Formato [FI]	Rispetta il formato = false [error] Rispetta il formato = true [PROPERTY FI_OK]
<b>Parametro: CAP</b>	
<b>Formato</b> ^\d{5}\$	
<b>Nome Categoria</b>	<b>Scelte per la categoria</b>
Formato [FC]	Rispetta il formato = false [error] Rispetta il formato = true [PROPERTY FC_OK]
<b>Parametro: Provincia</b>	
<b>Formato</b> ^[A-Z]{2}\$	



Formato [FP]	Rispetta il formato = false [error] Rispetta il formato = true [PROPERTY FP_OK]
<b>Parametro: Città</b>	
Nome Categoria	Scelte per la categoria
<b>Parametro: Strada</b>	
Nome Categoria	Scelte per la categoria

Test Case ID	Test frame	Esito
TC_1.1.1_1	LN1	Errato: Nome troppo lungo
TC_1.1.1_2	LN2, LC1	Errato: Cognome troppo lungo
TC_1.1.1_3	LN2, LC2, FE1	Errato: Email in formato errato
TC_1.1.1_4	LN2, LC2, FE2, LP1	Errato: Lunghezza errata della password
TC_1.1.1_5	LN2, LC2, FE2, LP2, FNT1	Errato: numero di telefono in formato errato
TC_1.1.1_6	LN2, LC2, FE2, LP2, FNT2, FR1	Errato: Ruolo in formato errato
TC_1.1.1_7	LN2, LC2, FE2, LP2, FNT2, FR2, LA1	Errato: Lunghezza errata dell'ambito
TC_1.1.1_8	LN2, LC2, FE2, LP2, FNT2, FR2, LA2, FI1	Errato: Immagine in formato errato
TC_1.1.1_9	LN2, LC2, FE2, LP2, FNT2, FR2, LA2, FI2, FC1	Errato: CAP in formato non valido



TC_1.1.1_10	LN2, LC2, FE2, LP2, FNT2, FR2, LA2, FI2, FC2, FP1	Errato: Provincia in formato non valido
TC_1.1.1_11	LN2, LC2, FE2, LP2, FNT2, FR2, LA2, FI2, FC2, FP2, LCI1	Errato: Lunghezza errata della città
TC_1.1.1_12	LN2, LC2, FE2, LP2, FNT2, FR2, LA2, FI2, FC2, FP2, LCI2, LS1	Errato: Lunghezza errata della strada
TC_1.1.1_13	LN2, LC2, FE2, LP2, FNT2, FR2, LA2, FI2, FC2, FP2, LCI2, LS2	Corretto

## 10.1.2 Visualizzazione delle informazioni del profilo

Parametro: Email	
Formato ^ [a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}\\$	
Nome Categoria	Scelte per la categoria
Formato[FE]	<ol style="list-style-type: none"> <li>Rispetta il formato = false [error]</li> <li>Rispetta il formato = true [PROPERTY FIS_OK]</li> </ol>

Test Case ID	Test frame	Esito
TC_1.1.2_1	FE	Errato: Email in un formato non valido

## 10.1.3 Modifica delle informazioni del profilo

Parametro: Nome	
Nome Categoria	Scelte per la categoria
Lunghezza [LN]	<ol style="list-style-type: none"> <li>Lunghezza &gt;100 = false[error]</li> <li>Lunghezza &lt;=100 = true [PROPERTY LN_OK]</li> </ol>
Parametro: Cognome	
Nome Categoria	Scelte per la categoria
Lunghezza [LC]	<ol style="list-style-type: none"> <li>Lunghezza &gt;100 = false[error]</li> <li>Lunghezza &lt;=100 = true [PROPERTY LC_OK]</li> </ol>



Parametro: Numero Telefonico	
	<b>Formato</b> <code>^\d{10}\$</code>
<b>Nome Categoria</b>	<b>Scelte per la categoria</b>
Formato [FNT]	1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FNT_OK]
Parametro: Ambito	
<b>Nome Categoria</b>	<b>Scelte per la categoria</b>
Lunghezza[LA]	1 Lunghezza >100 = false[error] 2 Lunghezza <=100 = true [PROPERTY LP_OK]
Parametro: Immagine	
	<b>Formato</b> <code>^data:image/[a-zA-Z0-9+-.]++;base64,([a-zA-Z0-9+/]{4})*([a-zA-Z0-9+/]{3}= [a-zA-Z0-9+/]{2}==)?\$</code>
Formato[FI]	1 Rispetta il formato = false [error] 2 Rispetta il formato = true [PROPERTY FI_OK]
Parametro: CAP	
	<b>Formato</b> <code>^\d{5}\$</code>
<b>Nome Categoria</b>	<b>Scelte per la categoria</b>
Formato[FC]	1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FC_OK]
Parametro: Provincia	
	<b>Formato</b> <code>^[A-Z]{2}\$</code>
Formato[FP]	1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FP_OK]
Parametro: Città	
<b>Nome Categoria</b>	<b>Scelte per la categoria</b>
Lunghezza [LCI]	1 Lunghezza >100 = false[error] 2 Lunghezza <=100 = true [PROPERTY LCI_OK]
Parametro: Strada	
<b>Nome Categoria</b>	<b>Scelte per la categoria</b>
Lunghezza [LS]	1 Lunghezza >100 = false[error] 2 Lunghezza <=100 = true [PROPERTY LS _OK]



Test Case ID	Test frame	Esito
TC_1.1.3_1	LN1	Errato: Nome troppo lungo
TC_1.1.3_2	LN2, LC1	Errato: Cognome troppo lungo
TC_1.1.3_3	LN2, LC2, FNT1	Errato: Numero di telefono in formato errato
TC_1.1.3_4	LN2, LC2, FNT2, LA1	Errato: Ambito troppo lungo
TC_1.1.3_5	LN2, LC2, FNT2, LA2, FI1	Errato: Immagine in formato errato
TC_1.1.3_6	LN2, LC2, FNT2, LA2, FI2, FC1	Errato: CAP in formato non valido
TC_1.1.3_7	LN2, LC2, FNT2, LA2, FI2, FC2, FP1	Errato: Provincia in formato non valido
TC_1.1.3_8	LN2, LC2, FNT2, LA2, FI2, FC2, FP2, LCI1	Errato: Lunghezza errata della città
TC_1.1.3_9	LN2, LC2, FNT2, LA2, FI2, FC2, FP2, LCI2, LS1	Errato: Lunghezza errata della strada
TC_1.1.3_10	LN2, LC2, FNT2, LA2, FI2, FC2, FP2, LCI2, LS2	Corretto



## 10.2 Gestione Contenuti

### 10.2.1 Aggiunta di un racconto

Parametro: Titolo	
Nome Categoria	Scelte per la categoria
Lunghezza [LT]	<ol style="list-style-type: none"><li>1. Lunghezza <math>\leq 0</math> OR Lunghezza <math>&gt; 255</math> = false error</li><li>2. <math>0 &lt; \text{Lunghezza} \leq 255</math> = true [PROPERTY LT_OK]</li></ol>
Parametro: Descrizione Racconto	
Nome Categoria	Scelte per la categoria
Lunghezza [LDR]	<ol style="list-style-type: none"><li>1. Lunghezza <math>\leq 0</math> = false [error]</li><li>2. Lunghezza <math>&gt; 0</math> = true [PROPERTY LDR_OK]</li></ol>
Parametro: Immagine	
Formato: ^data:image/[a-zA-Z0-9+-.]++;base64,([a-zA-Z0-9+/]{4})*([a-zA-Z0-9+/]{3}= [a-zA-Z0-9+/]{2}==)?\$	
Nome Categoria	Scelte per la categoria
Formato [FI]	<ol style="list-style-type: none"><li>1. Immagine presente AND formato non valido= false [error]</li><li>2. Immagine assente OR immagine presente AND formato valido [PROPERTY FI_OK]</li></ol>

Test Case ID	Test frame	Esito
TC_1.2.1_1	LT1	Errato: Titolo non valido (troppo lungo)
TC_1.2.1_2	LT1	Errato: Titolo non valido (vuoto)
TC_1.2.1_3	LT2, LDR1	Errato: Descrizione racconto non valida (vuota)
TC_1.2.1_4	LT2, LDR2, FI1	Errato: Formato dell'immagine non valido
TC_1.2.1_5	LT2, LDR2, FI2	Corretto



## 10.3 Gestione Volontariato

### 10.3.1 Partecipazione ad un evento

Parametro: Email	
Formato ^ [a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}\$	
Nome Categoria	Scelte per la categoria
Formato[FE]	<ol style="list-style-type: none"><li>3. Rispetta il formato = false [error]</li><li>4. Rispetta il formato = true [PROPERTY FIS_OK]</li></ol>
Parametro: Ruolo	
Formato ^(Volontario Beneficiario Ente)\$	
Nome Categoria	Scelte per la categoria
Formato[FR]	<ol style="list-style-type: none"><li>1. Rispetta il formato = false [error]</li><li>2. Rispetta il formato = true [PROPERTY FNTS_OK]</li></ol>

Test Case ID	Test frame	Esito
TC_1.3.1_1	FE	Errato: Email in un formato non valido
TC_1.3.1_2	FR	Errato: Ruolo non corrispondente

### 10.3.2 Creazione di un evento

Parametro: Titolo	
Nome Categoria	Scelte per la categoria
Lunghezza [LT]	<ol style="list-style-type: none"><li>1. Lunghezza <math>\leq 0</math> OR Lunghezza <math>&gt; 50</math> = false error</li><li>2. Lunghezza <math>\leq 50</math> = true [PROPERTY LT_OK]</li></ol>
Parametro: Descrizione Evento	
Nome Categoria	Scelte per la categoria



Lunghezza [LDE]	<ol style="list-style-type: none"> <li>1. Lunghezza <math>\leq 0</math> OR Lunghezza <math>&gt; 500 = \text{false}</math> [error]</li> <li>2. <math>&lt;</math> Lunghezza <math>\leq 500 = \text{true}</math> [PROPERTY LDE_OK]</li> </ol>
<b>Parametro: Numero Massimo Partecipanti</b>	
Nome Categoria	<b>Scelte per la categoria</b>
Valore [VNMP]	<ol style="list-style-type: none"> <li>3. <math>N &lt; 1 = \text{false}</math> [error]</li> <li>4. <math>N \geq 1 = \text{true}</math> [PROPERTY VNMP_OK]</li> </ol>
<b>Parametro: Data Evento</b>	
Nome Categoria	<b>Scelte per la categoria</b>
Validità [VDT]	<ol style="list-style-type: none"> <li>1. Data in formato non valido OR Data <math>&lt;</math> data odierna = false [error]</li> <li>2. Data in formato valido AND Data <math>\geq</math> data odierna = true [PROPERTY VDT_OK]</li> </ol>
<b>Parametro: Luogo Evento</b>	
Nome Categoria	<b>Scelte per la categoria</b>
Lunghezza [LLE]	<ol style="list-style-type: none"> <li>1. Lunghezza <math>\leq 0</math> OR Lunghezza <math>&gt; 100 = \text{false}</math> [error]</li> <li>2. <math>0 &lt; \text{Lunghezza} \leq 100 = \text{true}</math> [PROPERTY LLE_OK]</li> </ol>
<b>Parametro: Immagine</b>	
<b>Formato:</b> ^data:image/[a-zA-Z0-9+-.]++;base64,([a-zA-Z0-9+/]{4})*([a-zA-Z0-9+/]{3}= [a-zA-Z0-9+/]{2}==)?\$	
Nome Categoria	<b>Scelte per la categoria</b>
Formato (FI)	<ol style="list-style-type: none"> <li>1. Immagine presente AND formato non valido = false [error]</li> <li>2. Immagine assente OR Immagine presente AND formato valido = true [PROPERTY FI_OK]</li> </ol>

Test Case ID	Test frame	Esito
TC_1.3.2_1	LT1	Errato: Titolo non valido (vuoto o troppo lungo)
TC_1.3.2_2	LT2, LDE1	Errato: Descrizione evento non valida (vuota o troppo lunga)



TC_1.3.2_3	LT2, LDE2, VDT1	Errato: Data evento non valida (formato errato o nel passato)
TC_1.3.2_4	LT2, LDE2, VDT2, LLE1	Errato: Luogo evento non valido (vuoto o troppo lungo)
TC_1.3.2_5	LT2, LDE2, VDT2, LLE2, VNMP1	Errato: Numero massimo partecipanti non valido ( $N < 1$ )
TC_1.3.2_6	LT2, LDE2, VDT2, LLE2, VNMP2, FI1	Errato: Immagine in formato non valido
TC_1.3.2_7	LT2, LDE2, VDT2, LLE2, VNMP2, FI2	Corretto

### 10.3.3 Modifica di un evento

Parametro: Titolo	
Nome Categoria	Scelte per la categoria
Lunghezza [LT]	Lunghezza $\leq 0$ OR Lunghezza $> 50$ = false error Lunghezza $\leq 50$ = true [PROPERTY LT_OK]
Parametro: Descrizione Evento	
Nome Categoria	Scelte per la categoria
Lunghezza [LDE]	Lunghezza $\leq 0$ OR Lunghezza $> 500$ = false [error] $<$ Lunghezza $\leq 500$ = true [PROPERTY LDE_OK]
Parametro: Numero Massimo Partecipanti	
Nome Categoria	Scelte per la categoria
Valore [VNMP]	1. $N < 1$ = false [error] 2. $N \geq 1$ = true [PROPERTY VNMP_OK]
Parametro: Data Evento	
Nome Categoria	Scelte per la categoria



Validità [VDT]	Data in formato non valido OR Data < data odierna = false [error] Data in formato valido AND Data ≥ data odierna = true [PROPERTY VDT_OK]
<b>Parametro: Luogo Evento</b>	
Nome Categoria	Scelte per la categoria
Lunghezza [LLE]	Lunghezza ≤ 0 OR Lunghezza > 100 = false [error] 0 < Lunghezza ≤ 100 = true [PROPERTY LLE_OK]
<b>Parametro: Immagine</b>	
Formato: ^data:image/[a-zA-Z0-9+-]+;base64,([a-zA-Z0-9+/]{4})*([a-zA-Z0-9+/]{3}= [a-zA-Z0-9+/]{2}==)?\$	
Nome Categoria	Scelte per la categoria
Formato (FI)	Immagine presente AND formato non valido= false [error] Immagine assente OR Immagine presente AND formato valido = true [PROPERTY FI_OK] la formulaa

Test Case ID	Test frame	Esito
TC_1.3.3_1	LT1	Errato: Titolo non valido (vuoto o troppo lungo)
TC_1.3.3_2	LT2, LDE1	Errato: Descrizione evento non valida (vuota o troppo lunga)
TC_1.3.3_3	LT2, LDE2, VDT1	Errato: Data evento non valida (formato errato o nel passato)
TC_1.3.3_4	LT2, LDE2, VDT2, LLE1	Errato: Luogo evento non valido (vuoto o troppo lungo)
TC_1.3.3_5	LT2, LDE2, VDT2, LLE2, VNMP1	Errato: Numero massimo partecipanti non valido (N < 1)



TC_1.3.3_6	LT2, LDE2, VDT2, LLE2, VNMP2, FI1	Errato: Immagine in formato non valido
TC_1.3.3_7	LT2, LDE2, VDT2, LLE2, VNMP2, FI2	Corretto

### 10.3.4 Avviare una raccolta fondi

Parametro: Titolo				
Nome Categoria	Scelte per la categoria			
Lunghezza [LT]	1. Lunghezza > 255 = false[error] 2. Lunghezza <= 255 = true [PROPERTY LN_OK]			
Esistenza [ET]	1. Titolo non inserito = false[error] 2. Titolo inserito = true [PROPERTY ET_OK]			
Parametro: Obiettivo				
<b>Formato:</b> $^{\backslash d\{1,8\}\backslash .\backslash d\{2\}}$$				
Nome Categoria	Scelte per la categoria			
Formato [FO]	1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FO_OK]			
Esistenza [EO]	1. Obiettivo non inserito = false [error] 2. Obiettivo inserito = true [PROPERTY EO_OK]			
Parametro: DataChiusura				
<b>Formato:</b> $^{\backslash d\{4\}\backslash /(0[1-9]\mid 1[0-2])\backslash /(0[1-9]\mid [12]\backslash d\mid 3[01])$$				
Nome Categoria	Scelte per la categoria			
Formato [FDC]	1. Rispetta il formato OR DataChiusura < DataApertura = false [error] 2. Rispetta il formato AND Datachiusura >= DataApertura = true [PROPERTY FDC_OK]			



Esistenza [EDC]	1. DataChiusura non inserita = false[error] 2. DataChiusura inserita = true[PROPERTY EDC_OK]
-----------------	---

Test Case ID	Test frame	Esito
TC_1.3.4_1	ET1	Errato: Titolo non inserito
TC_1.3.4_2	ET2, LT1	Errato: Lunghezza del titolo invalida
TC_1.3.4_3	ET2, LT2, EO1	Errato: Obiettivo non inserito
TC_1.3.4_4	ET2, LT2, EO2, FO1	Errato: Formato dell'obiettivo errato
TC_1.3.4_5	ET2, LT2, EO2, FO2, EDC1	Errato: DataChiusura non inserita
TC_1.3.4_6	ET2, LT2, EO2, FO2, EDC2, FDC1	Errato: DataChiusura non valida (formato errato/ è minore rispetto alla data di inizio)
TC_1.3.4_7	ET2, LT2, EO2, FO2, EDC2, FDC2	Corretto



## 10.4 Gestione Richieste

### 10.4.1 Richiedere affiliazione

Parametro: Email Volontario	
<b>Formato</b> ^ [a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}\$	
Nome Categoria	Scelte per la categoria
Formato[FEV]	<ol style="list-style-type: none"><li>Rispetta il formato = false [error]</li><li>Rispetta il formato = true [PROPERTY FEV_OK]</li></ol>
Parametro: Ruolo Volontario	
<b>Formato</b> ^(Volontario Beneficiario Ente)\$	
Nome Categoria	Scelte per la categoria
Formato[FRV]	<ol style="list-style-type: none"><li>Rispetta il formato = false [error]</li><li>Rispetta il formato = true [PROPERTY FRV_OK]</li></ol>
Parametro: Email Ente	
<b>Formato</b> ^ [a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}\$	
Nome Categoria	Scelte per la categoria
Formato[FEE]	<ol style="list-style-type: none"><li>Rispetta il formato = false [error]</li><li>Rispetta il formato = true [PROPERTY FEE_OK]</li></ol>
Parametro: Ruolo Ente	
<b>Formato</b> ^(Volontario Beneficiario Ente)\$	
Nome Categoria	Scelte per la categoria
Formato[FRE]	<ol style="list-style-type: none"><li>Rispetta il formato = false [error]</li><li>Rispetta il formato = true [PROPERTY FRE_OK]</li></ol>

Test Case ID	Test frame	Esito
TC_1.4.1_1	FEV1	Errato: Utente non trovato.



TC_1.4.1_2	FEV2, FRV1	Errato: Solo i volontari possono richiedere l'affiliazione.
TC_1.4.1_3	FEV2, FRV2, FEE1	Errato: Ente destinatario mancante.
TC_1.4.1_4	FEV2, FRV2, FEE2, FRE1	Errato: Puoi fare la richiesta solo ad un ente.
TC_1.4.1_5	FEV2, FRV2, FEE2, FRE2	Corretto

## 10.4.2 Accettare richiesta di affiliazione

Parametro: Email Ente	
Formato	
$^{\text{[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}}\$}$	
Nome Categoria	Scelte per la categoria
Lunghezza [FEE]	1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FEE_OK]
Parametro: Ruolo Ente	
Formato	
$^{\text{(Volontario Beneficiario Ente)}}\$$	
Nome categoria	Scelte per la categoria
Formato[FRE]	1. Rispetta il formato = false [error] 2. Rispetta il formato = true [PROPERTY FRE_OK]

Test Case ID	Test frame	Esito
TC_1.4.2_1	FEE	Errato: Email in formato errato
TC_1.4.2_2	FRE	Errato: Ruolo in formato errato



## 10.5 Gestione Ricerca

### 10.5.1 Eseguire la ricerca di un utente per nome

Parametro: Nome	
Nome Categoria	Scelte per la categoria
Lunghezza [LN]	1. Lunghezza > 100 = false error 2. Lunghezza $\leq$ 100 = true [PROPERTY LN_OK]

Test Case ID	Test frame	Esito
TC_1.5.1_1	LN1	Errato: Nome non inserito
TC_1.5.1_2	LN2	Corretto

## 11 Testing Schedule

Le attività di pianificazione del testing seguiranno quanto definito nei capitoli precedenti e avranno inizio immediatamente dopo la fase di design, che fornisce le informazioni necessarie per organizzare e pianificare le verifiche sul sistema.

La definizione e la scrittura dei casi di test verranno svolte parallelamente allo sviluppo del codice, così da mantenere un collegamento diretto tra quanto progettato e quanto realizzato, e permettere di verificare fin da subito il corretto comportamento delle varie funzionalità.

L'esecuzione dei test avverrà sia durante che al termine dell'implementazione. Durante lo sviluppo saranno effettuati i test di unità e i test di integrazione, mentre una volta conclusa l'implementazione verranno eseguiti tutti i test di sistema. Al termine, i test principali saranno ripetuti per verificare che non si siano introdotti malfunzionamenti e per produrre i report finali.

Per una descrizione più dettagliata delle tempistiche generali si rimanda ai documenti di management dedicati alla pianificazione del progetto.