|  |
| --- |
| **Università degli Studi di Salerno Corso di Ingegneria del Software** |

**LetsMeet  
RAD- Requirement Analysis Documentation  
Versione 2.1**

Data: 13/12/2018

|  |  |
| --- | --- |
| Progetto: LetsMeet | Versione: 1.0 |
| Documento: RAD | Data: 13/12/2018 |

**Coordinatore del progetto:**

|  |  |
| --- | --- |
| Nome | Matricola |
|  |  |
| Gerardo Benevento | 0512104584 |

**Partecipanti:**

|  |  |
| --- | --- |
| Nome | Matricola |
|  |  |
| Vittorio Aiello | 0512104524 |
| Gerardo Benevento | 0512104584 |
| Raffaele Sansone | 0512104974 |

|  |  |
| --- | --- |
| **Scritto da:** | Gerardo Benevento, Raffaele Sansone, Vittorio Aiello |

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Versione | Descrizione | Autore |
| 13/12/2018 | 1.0 | Prima stesura del RAD | GB; VA; RS |
| 15/12/2018 | 1.1 | Rettifica del Class Diagram | GB; VA |
| 19/12/2018 | 2.0 | Rettifica del Class Diagram, Sequence e State | GB; VA; RS |
| 15/01/2019 | 2.1 | Aggiunta caso d’uso: CancellazioneEvento | GB;VA;RS |

Sommario

1. [1. Introduzione 3](#_Toc433975253)

[1.1 Scopo del sistema](#_Toc433975254)

[1.2 Ambito del sistema](#_Toc433975255)

[1.3 Obiettivi e criteri di successo del sistema](#_Toc433975256)

[1.4 Definizioni, acronimi e abbreviazioni](#_Toc433975257)

[1.5 Riferimenti](#_Toc433975258)

[1.6 Panoramica](#_Toc433975259)

1. [2. Sistema corrente 3](#_Toc433975260)
2. [3. Sistema proposto 3](#_Toc433975261)

[3.1 Panoramica](#_Toc433975262)

[3.2 Requisiti funzionali](#_Toc433975263)

[3.3 Requisiti no funzionali](#_Toc433975264)

[3.4 Modello di sistema](#_Toc433975265)

1. [4. Glossario 3](#_Toc433975266)

# 1. Introduzione

## 1.1 Scopo del sistema

Il sistema che si vuole sviluppare ha come scopo la creazione di eventi in tempo reale da parte degli utenti e partecipazione a quest’ultimi, garantendo un continuo monitoraggio per poter comprendere l’indice di gradimento dell’evento, quanti utenti vorranno parteciparvi e l’effettivo grado di affluenza, riprodotto in tempo reale, degli utenti all’evento.

Il sistema estende questa capacità nella creazione degli eventi a scopi anche commerciali, con aziende o imprese che decidono di affiliarsi mettendo a disposizione dei punti acquisibili unicamente con la verifica della partecipazione da parte degli utenti agli eventi creati dagli enti ed eventuale spesa di questi punti su un portale esterno.

## 1.2 Ambito del sistema

Il sistema troverà uso in un qualsiasi contesto in cui ci sia la necessità di organizzare e/o monitorare eventi; sia in chiave personale che professionale. Gli esempi di contesto possono variare dall’organizzazione di una serata in discoteca da parte di un gruppo di frequentatori fino alla pubblicizzazione dello stesso evento da parte del locale

In generale l’ambito del sistema lo si ritrova in ogni contesto sociale in cui c’è necessità di trovare o pubblicizzare eventi di tutti i tipi

## 1.3 Obiettivi e criteri di successo del sistema

LetsMeet si propone di offrire un network real-time in cui gli utenti possono creare e condividere eventi sul territorio tramite geolocalizzazione, dando la possibilità a tutti gli utilizzatori della piattaforma, di poter venire a conoscenza di eventi culturali e sportivi, feste, ed ogni categoria di avvenimento che si svolge sul territorio utilizzando dei criteri di ricerca come la vicinanza.

La geolocalizzazione ed altri metodi di verifica permetteranno agli utenti di essere riconosciuti come partecipanti agli eventi cui si recano e conseguentemente esprimere un giudizio sullo stesso, il quale influenzerà il feedback dell’evento e del suo creatore.

LetsMeet metterà a disposizione degli utenti metodi per poter comunicare tra di loro e con gli organizzatori degli eventi, attraverso delle sezioni dedicate per ogni evento, in maniera pubblica.

La piattaforma inoltre offre la possibilità di guadagnare punti, partecipando a eventi particolari, che potranno essere utilizzati per beneficiare di sconti presso aziende convenzionate e potrà dare l’occasione di creare eventi con partnership esterne dando dei benefit agli stessi partecipanti, dettati dalle partnership, gestendo un sistema di controllo della partecipazione.

## 1.3.1 Obiettivo 2

La piattaforma LetsMeet si propone come obiettivo quello di essere una piattaforma utilizzata per scoprire eventi di ogni tipo nelle proprie vicinanze favorendo lo sviluppo di relazioni iterpersonali. Ulteriormente quello di essere anche una piattaforma di pubblicizzazione per eventi sociali di varia natura e offrire al creatore dell’evento metriche adatte a comprendere l’indice di gradimento e statistiche real-time sull’evento

## 1.4 Definizioni, acronimi e abbreviazioni

**LM:** LetsMeet La piattaforma descritta in questo documento anche detta sistema

**MP**: “Meet” points i punti utilizzabili dagli utenti per ottenere diversi item da terze parti

**DASHBOARD**: Piattaforma informatica interattiva, che consente l’uso di strumenti facilmente accessibili.

**MAPPA:** Mappa interattiva con l’utente.

## 1.5 Riferimenti

## Bernd Bruegge & Allen H. Dutoit, *Object-Oriented Software Engineering: Using UML, Patterns and Java*, (2nd edition),Prentice-Hall, 2003.

## 1.6 Panoramica

Al secondo punto del documento verrà presentato il sistema corrente.

Al terzo punto verrà presentato il sistema proposto con i relativi requisiti funzionali e non funzionali. Attraverso scenari e use case verrà stabilito chi sono gli attori del sistema e come gli attori individuati interagiscono con il sistema.   
Verrà presentata la struttura del sistema tramite modello a oggetti e sequence diagram.

Verranno realizzati dei mock-up per rappresentare l’interfaccia grafica.

# 2. Sistema corrente

È evidente la mancanza di strumenti adatti alla creazione, monitoraggio e visualizzazione di eventi sul territorio. Considerando i social attualmente più utilizzati come Facebook, Twitter e Instagram sembra mancare una componente che si rifletta sul mondo reale, nonostante Facebook includa una sorta di gestione degli eventi quest’ultima non è adatta ad essere utilizzata in real-time. Nasce allora l’esigenza di spostare il focus da un mondo completamente virtuale, come quello dei social, ad uno più ricco di interazioni sociali basati su incontri frontali.

La promozione di eventi diventa difficile per i piccoli organizzatori non riuscendo ad incrementare il numero di partecipanti ad eventi, vista la mancanza di visibilità ed interattività sulle moderne piattaforme. Tal volta gli stessi utenti delle piattaforme vengono abbandonati su di esse, senza nessun punto di riferimento ed informativo.

Attualmente i social moderni permettono la creazione di un evento e la sua pubblicazione, senza applicare una gestione dell’evento stesso. Questo porta all’idea di un completo abbandono dell’inserzione e l’impressione di una community completamente assente.

La maggior parte delle piattaforme che si propongono di fornire un servizio sulla gestione di eventi, non offrono nessuna prestazione real-time alla community che ne fa parte; allontanando potenziali inscritti che non trovano nessun tipo di gratificazione e interattività nell’utilizzo del mondo che offrono le stesse piattaforme.

Le maggiori piattaforme per la creazione di eventi forniscono una descrizione limitata e facoltativa degli incontri e il luogo dove essi avverranno senza un aggiornamento constante dell’affollamento dovuto all’evento creato, né una mappa interattiva per la scoperta di nuovi avvenimenti, né un sistema di votazione offerto ai partecipanti dell’evento per giudicare quest’ultimo dopo il suo svolgimento.

Il sistema proposto vuole superare le barriere imposte dai vecchi sistemi dei social network, in quanto si munisce di sistemi per poter creare eventi in tempo reale da parte degli utenti, garantendo un continuo monitoraggio di quest’ultimi per poter comprendere le esigenze e i feedback da parte degli utenti che vi partecipano.

Gli utenti hanno bene mondo di poter partecipare agli eventi, visualizzando gli ultimi creati nelle loro vicinanze con le relative informazioni fra qui anche i partecipanti effettivi durante l’evento in continuo aggiornamento.

# 3. Sistema proposto

## 3.1 Panoramica

Il sistema da noi proposto è un’applicazione web, l’obiettivo e quello di poter dare la possibilità agli utenti di poter creare attraverso una mappa interattiva eventi in tempo reale, i quali hanno una locazione ben precisa dettata dalla stessa interazione con la mappa oppure attraverso un sistema di ricerca della località.

Gli eventi potranno essere riconosciuti attraverso tipologia per poter dare la possibilità agli utenti di poter interagire unicamente con quelli che più gli interessano.

Ogni evento avrà una scheda di informazione dettata dal creatore dell’evento, compresa una sezione di messaggistica dove gli eventuali partecipanti dell’evento potranno comunicare, per poter essere aggiornati sull’evento stesso.

I partecipanti degli eventi, confermata la loro presenza agli eventi ogni qualvolta si recano nel luogo dell’evento, potranno votare attraverso una sistema di rating molto minimale se hanno gradito o meno l’evento, inficiando sul feedback sia dell’evento che dell’utente creatore dell’evento stesso, il quale a sua volta avrà una sua pagina personale, visitabile, dove gli utenti possono leggere gli eventi creati e partecipati da lui.

La piattaforma si mette a disposizione anche alle aziende ed enti esterni che vogliono promuovere le loro iniziative mettendo a disposizione dei punti speciali chiamati punti “Meet” agli utenti che vi partecipano.

Tali punti potranno essere spesi sia per un eventuale promozione dell’azienda o ente esterno oppure su di una piattaforma legata al sistema LetsMeet.

## 3.2 Requisiti funzionali

**Gestione Account:**

RF1: Il sistema darà l’opportunità all’utente di autenticarsi all’interno della piattaforma.

RF2: Il sistema darà l’opportunità di disconnettersi dalla piattaforma.

RF3: Il sistema darà la possibilità di registrare un nuovo account.

RF4: Il sistema darà la possibilità ad utenti speciali chiamati “Super-Admin” di registrare i Moderatori.

RF5: Il sistema darà la possibilità di poter cercare gli utenti che hanno un account.

RF6: Il sistema darà a disposizione all’utente la sua pagina di profilo con eventuali eventi, feedback globale.

**Gestione Eventi:**

RF7: Il sistema darà all’utente la possibilità di visionare gli eventi limitrofi alla sua posizione attuale.

RF8: Il sistema darà la possibilità di creare un evento specificando tipologia, descrizione e la posizione, attraverso la mappa.

RF9: Il sistema darà all’utente la possibilità di votare un evento a cui ha partecipato.

RF10: Il sistema darà all’utente la possibilità di esplicitare la sua partecipazione ad un evento.

RF11: Il sistema darà all’utente la possibilità di verificare la sua partecipazione ad un evento.

RF12: Il sistema darà la possibilità all’utente di cercare eventi.

RF13: Il sistema darà la possibilità di visionare le informazioni di un evento.

RF14: Il sistema darà all’utente la possibilità di poter scrivere commenti nella sezione dedicata degli eventi.

RF15: Il sistema darà all’utente la possibilità di poter visionare i commenti scritti ad un evento.

RF16: Il sistema darà all’utente la possibilità di poter cancellare un evento.

**Gestione Segnalazioni:**

RF17: Il sistema darà la possibilità all’utente di poter segnale ai moderatori eventi.

RF18: Il sistema darà la possibilità all’utente di segnalare ai moderatori commenti inviati.

RF19: Il sistema darà la possibilità ai moderatori di poter notificare le segnalazioni agli utenti accettando le segnalazioni, sospendendo un utente.

RF20: Il sistema darà la possibilità ai moderatori di rifiutare l’eventuale segnalazione fatta da un utente.

**3.3 Requisiti non funzionali**

**3.3.1 Usabilità**

Si pensa che la piattaforma sarà utilizzata per la maggiore da utenti già familiari ai più comuni social network per questo si utilizzerà uno stile simile a queste ultime per facilitare l’integrazione degli utenti sulla piattaforma e massimizzare l’intuitività delle funzionalità offerte. Per ulteriormente facilitare l’utente nell’utilizzo della piattaforma verrà creata una pagina FAQ che coprirà le funzionalità di base dell’applicazione.  
Per le azienda partner verrà invece fornita una documentazione estensiva mirata ad illustrare le API che permetteranno il corretto utilizzo delle funzionalità automatiche della piattaforma e l’integrazione con software terzo

**3.3.2 Affidabilità**

Il sistema non ha un forte bisogno di affidabilità oltre i seguenti accorgimenti:

* bisogna prevedere un meccanismo per la sospensione del servizio per manutenzione
* i dati persistenti (quali quantità di MP posseduti da un utente) non dovranno essere influenzati da un possibile situazione d’errore e mantenere la coerenza e consistenza
* autenticazione da parte delle aziende terze per l’utilizzo delle API
* metodologie che impediscano l’utilizzo indebito dei MP

**3.3.3 Prestazioni**

Il prodotto software deve fornire una piacevole navigazione all’utente, quindi avere dei tempi di risposta inferiore ai 700ms sotto traffico moderato quando l’utente dispone di una connessione che non sia limitante. Requisiti stringenti sono richiesti dalla natura real-time della monitorizzazione degli eventi e delle sezioni pubbliche di discussione sugli eventi. I particolari requisiti richiesti per ogni tipologia di operazione saranno indicati nei rispettivi casi d’uso

**3.3.4 Supportabilità**

Il software che implementerà l’applicazione server dovrà essere deployabile in formato war tramite Tomcat e quindi di conseguenza sarà supportata su qualsiasi sistema in cui è presente l’ambiente java e Tomcat, con l’aggiunta di eventuali librerie necessarie

I client supportati saranno i browser Chrome e Firefox in quanto tra i più utilizzati

**3.3.5 Implementazione**

Poiché il è richiesto che il SW sia fornito in formato war sarà implementato in JAVA, specificatamente JAVA SE 1.8

**3.3.6 Interfaccia**

Non ci sono particolari requisiti di interfacciamento con sistemi legacy

**3.3.7 Packaging**

Il SF sarà disponibile per il cliente in formato war scaricabile dal sito aziendale congiuntamente alle eventuali librerie necessarie sotto forma di archivio. L’installazione consisterà esclusivamente della copia del file e delle librerie all’interno delle cartelle indicate nel file README all’interno dell’archivio

**3.3.8 Legali**

N/G

## 3.4 Modello di Sistema

**3.4.1 Scenari**

**3.1 Partecipazione evento sportivo “Play-off Seria A 2”**

Francesco vuole partecipare ad un evento sportivo per trascorrere una domenica pomeriggio all’insegna dello sport quindi decide di effettuare l’accesso sulla piattaforma “LetsMeet”, visiona sulla sua bacheca, sottoforma di tabella, gli eventi consigliatigli e che sono nei pressi della sua locazione attuale, ma non trova ciò che cerca.

Francesco allora decide di utilizzare la mappa interattiva offerta dall’applicazione per trovare e identificare gli eventi sul territorio e volendo partecipare ad un evento sportivo, filtra attraverso un menu a finestra tutti gli eventi che non riguardano lo sport nel loro tag descrittivo. Francesco trova l’evento che cercava:” Play-off Seria A 2”, clicca sull’icona dell’evento situato sulla mappa e visualizza la pagina di info dell’evento stesso. Al suo interno trova: il luogo dell’evento, la data, l’ora, il numero di utenti che vogliono partecipare all’evento: “Play-off Seria A 2” e il suo creatore, l’utente Raffaele.

Francesco decide di partecipare all’evento comunicando la sua adesione attraverso un pulsate apposito della piattaforma ed a Raffaele attraverso una notifica, viene comunicata la partecipazione all’evento sportivo “Play-off Seria A 2” da parte di Francesco.

L’utente Francesco accede alla bacheca messaggi dell’evento per verificare se qualche altro utente abbia aggiunto dettagli all’evento stesso, mediante messaggio, ma essendo casualmente la bacheca vuota, Francesco abbandona la sezione dedicata all’evento e la piattaforma e all’orario prestabilito dettato dalle informazioni date all’evento, si reca al luogo specificato per “Play-Off Seria A 2”.

Arrivato nel luogo prestabilito, Francesco accede alla piattaforma “LetsMeet” con il proprio account e condivide la sua partecipazione mediante il pulsante apposito, facendo in modo che a Raffaele venga inviata una notifica che lo informa di tale cosa ed incrementando il numero dei partecipanti all’evento, costantemente aggiornato sulla piattaforma.

Francesco e Raffaele, attraverso la bacheca comune dell’evento, specificano di incontrarsi vicino agli spalti nord dello stadio, attuale luogo d’incontro dell’evento, per vedere la partita assieme.

**3.2 Rating di un evento svolto**.

Carletto, lo scorso sabato sera, ha utilizzato la piattaforma “LetsMeet” venendo a conoscenza di una festa in un parco nei pressi di casa sua, decise di parteciparvi condividendo la sua scelta con tutti gli utenti dell’evento attraverso l’utilizzo della piattaforma.

Il giorno seguente, Carletto decide di dare una votazione all’evento a cui ha partecipato il giorno precedente quindi accede alla piattaforma “LetsMeet” con le proprie credenziali.

Carletto dalla pagina Home della piattaforma, si sposta alla pagina dedicata al suo profilo dove gli vengono mostrate le sue generalità, richieste durante alla registrazione sulla piattaforma, insieme agli eventi da lui creati e partecipati divisi in modo evidente dalla piattaforma.

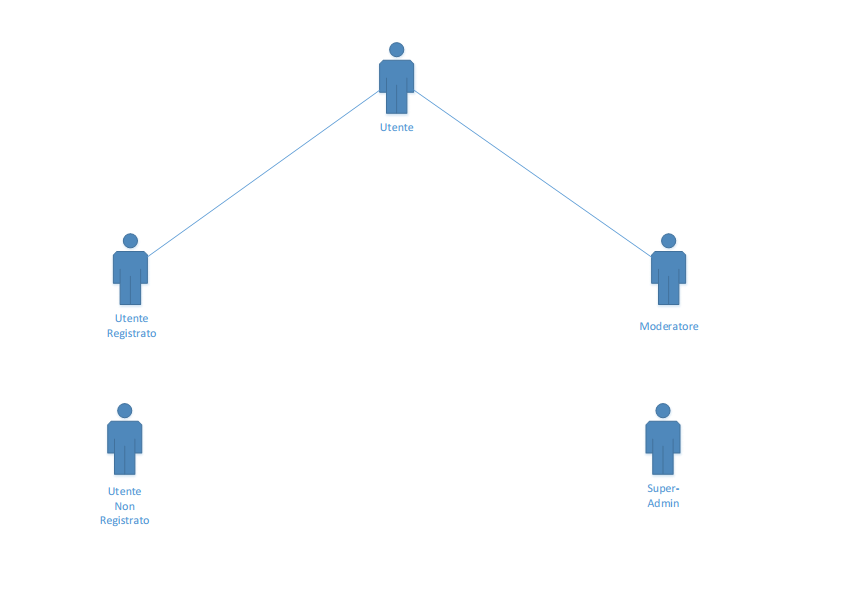
Viene informato, attraverso un segnale visivo, che può votare l’evento della festa nel parco a cui ha partecipato il giorno prima, usufruendo del sistema di rating offerto dalla piattaforma stessa.

Carletto, essendo particolarmente soddisfatto dell’organizzazione dell’evento del giorno prima, decide di votare in modo positivo l’evento cliccando il pulsante “pollice all’insù” anziché “pollice all’ingiù” e condivide la sua scelta attraverso il pulsante apposito ritornando sulla pagina profilo subito dopo.

A Pasquale, creatore dell’evento della festa nel parco vicino casa di Carletto, dopo le ore prestabilite per il voto, gli viene comunicato il giudizio finale che gli utenti hanno avuto riguardo alla festa e il suo feedback personale cambia in base a queste votazioni.

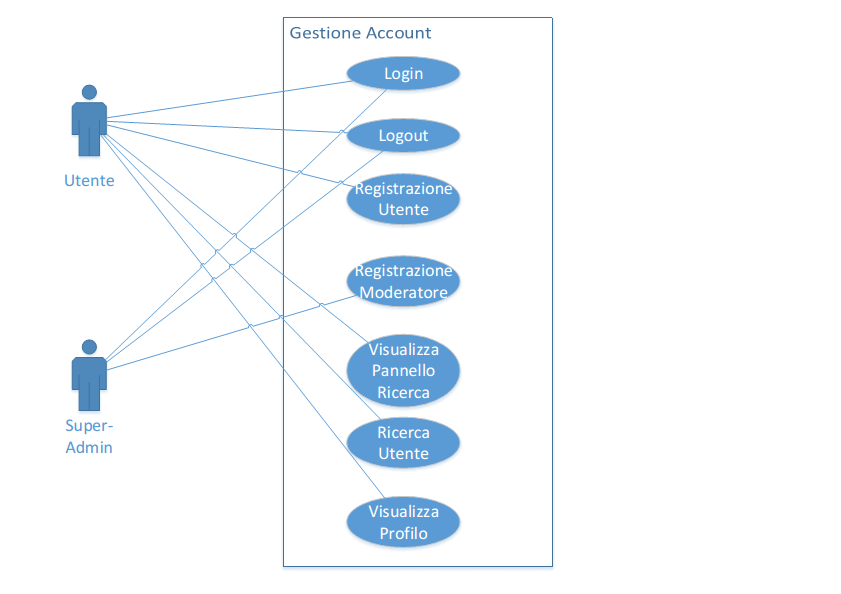
**3.3** **Creazione dell’evento culturale “Formazione gruppo di studio universitario”**

Mirco, studente di informatica all’università di Fisciano, ha da poco iniziato il suo primo anno accademico e non conosce ancora nessuno. Mirco decide di utilizzare la piattaforma LetsMeet per creare un evento pubblico con la finalità di conoscere persone che seguono il suo stesso corso di studi. Apre la piattaforma e seleziona sulla mappa fornita la zona dove risiede la biblioteca universitaria per selezionare quel luogo come punto d’incontro del suo evento. A questo punto pensa di voler creare un evento basato sulla formazione di un gruppo di studio con la finalità di assimilare più nozioni possibili nella materia “Architettura degli Elaboratori”. A questo punto Mirco decide il titolo del suo evento chiamandolo “gruppo di studio universitario” e lo inserisce nelle informazioni riguardanti l’evento da creare. Completa le informazioni inserendo la data, il luogo, l’orario e una piccola descrizione per specificare il motivo per il quale sta creando un evento. Mirco seleziona la categoria che ritiene più appropriata al suo evento. Tra le categorie disponibili Mirco seleziona “evento culturale” e dopo aver controllato che tutte le informazioni inserite sono corrette, posta l’evento sulla piattaforma in attesa di adesioni.

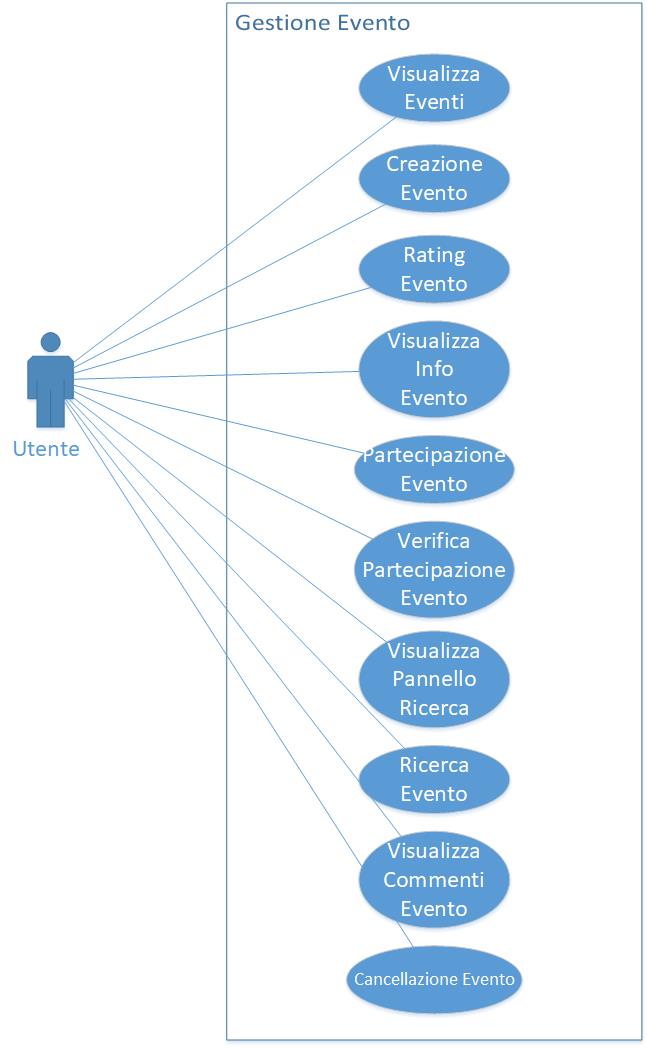
****

**Leggenda: (Sugli utenti)**

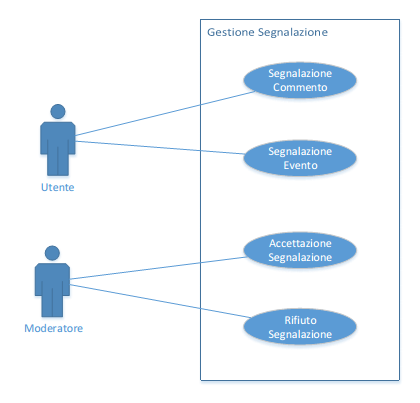
**Gestione Account**

****

**Gestione Evento:**

****

**Gestione Segnalazione:**

****

### 3.4.2 Use Case

|  |  |  |
| --- | --- | --- |
| Nome caso d’uso | LoginUtenteRegistrato | |
| Partecipanti | UtenteRegistrato. | |
| ID | UCLOGIN | |
| Flusso di eventi: | UTENTE  1.UtenteRegistrato attiva la funzione di login tramite bottone di Login  3. UtenteRegistrato riempie i form e sottomette le informazioni al sistema. | SISTEMA  2. Il sistema presenta un form di Login ad UtenteRegistrato per l’inserimento di username e password.  4. Il sistema riconosce UtenteRegistrato attraverso le credenziali.  5. UtenteRegistrato viene reindirizzato alla Home Page. |
| Condizioni di entrata | UtenteRegistrato deve aver svolto il caso d’uso: UCREGISTRAZIONEUT | |
| Eccezioni | Nel punto 4. Il sistema non riconosce UtenteRegistrato, gli notifica l’errore e attende un nuovo inserimento. | |
| Condizioni di uscita | Il sistema riconosce UtenteRegistrato e lo reindirizza alla pagina home | |
| Requisiti di funzionalità | Il sistema riconosce UtenteRegistrato entro 20 secondi | |

|  |  |  |
| --- | --- | --- |
| Nome caso d’uso | Logout UtenteRegistrato | |
| ID | UCLOGOUT | |
| Partecipanti | UtenteRegistrato | |
| Flusso di eventi: | UTENTE  1. UtenteRegistrato attiva la funzione Logout dal proprio profilo | SISTEMA  2. Il sistema disconnette UtenteRegistrato |
| Condizioni di entrata | UtenteRegistrato deve aver svolto il caso d’uso: UCLOGIN | |
| Eccezioni |  | |
| Condizioni di uscita | UtenteRegistarto non è più loggato | |
| Requisiti di funzionalità | Il sistema effettua il logoutl’UtenteRegistrato entro 10 secondi | |

|  |  |  |
| --- | --- | --- |
| Nome caso d’uso | Creare evento | |
| ID | UCCEVENTO | |
| Partecipanti | UtenteRegistrato. | |
| Flusso di eventi: | UTENTE  1. UtenteRegistrato clicca in un punto voluto sulla mappa interattiva  3. UtenteRegistrato compila i form proposti | SISTEMA  2. Il sistema risponde presentando ad UtenteRegistrato un form per le informazioni dell’evento che richiede la tipologia dell’evento, nome dell’evento, descrizione dell’evento, data e orario di inizio e di fine e località dell’evento, già inserito attraverso l’interazione al punto 1 con la mappa  4. Il Sistema controlla le informazioni sottomesse e crea un evento e invia una notifica di avvenuta transizione ad UtenteRegistrato |
| Condizioni di entrata | UtenteRegistrato deve aver svolto il caso d’uso: UCLOGIN &  UtenteRegistrato deve aver svolto il caso d’uso: UCVISUALIZZAZIONEEV | |
| Eccezioni |  | |
| Condizioni di uscita | L’evento è stato creato | |
| Requisiti di funzionalità | Il sistema rende disponibile l’evento agli altri utenti entro 3 minuti | |

|  |  |  |
| --- | --- | --- |
| Nome caso d’uso | CancellazioneEvento | |
| ID | UCCANCEV | |
| Partecipanti | UtenteRegistrato | |
| Flusso di eventi: | UTENTE  1. UtenteRegistrato attiva la funzione “Cancella” da sopra al profilo d’informazione dell’evento scelto | SISTEMA  2. Il sistema elimina l’evento selezionato e notifica il successo dell’operazione all’utente. |
| Condizioni di entrata | UtenteRegistrato deve aver svolto il caso d’uso: UCLOGIN  UtenteRegistrato deve aver svolto il caso d’uso:UCVISUALIZZAZIONEEV | |
| Eccezioni |  | |
| Condizioni di uscita | UtenteRegistarto ha cancellato l’evento | |
| Requisiti di funzionalità | Il sistema effettua la cancellazione dell’evento entro 10 secondi | |

|  |  |  |
| --- | --- | --- |
| Nome caso d’uso | Invio Commento | |
| ID | UCCOMMENTO | |
| Partecipanti | UtenteRegistrato | |
| Flusso di eventi: | UTENTE  1. UtenteRegistrato compila un form per il commento con il testo voluto e lo invia | SISTEMA  2. Il sistema mostra nel CommentBoundary, dell’evento scelto precedentemente durante il caso d’uso: UCVISCOMMENTI, il nuovo commento |
| Condizioni di entrata | UtenteRegistrato deve aver completato il caso d’uso: UCVISCOMMENTI | |
| Eccezioni |  | |
| Condizioni di uscita | Il commento inserito da UtenteRegistrato in CommentForm viene visualizzato in CommentBoundary e sarà disponibile per la visualizzazione agli altri utenti | |
| Requisiti di funzionalità | Il sistema invia e salva il messaggio entro 1 minuto | |

|  |  |  |
| --- | --- | --- |
| Nome caso d’uso | Visualizza commenti evento | |
| ID | UCVISCOMMENTI | |
| Partecipanti | UtenteRegistrato. | |
| Flusso di eventi: | UTENTE  1. UtenteRegistrato attiva la visualizzazione dei commenti | SISTEMA  2. Il sistema fornisce ad UtenteRegistrato un form per l’inserimento del proprio commento CommentFrom e permette la visualizzazione di quelli già esistenti dell’evento precedentemente scelto nel caso d’uso: UCVINFOEV tramite CommentBoundary |
| Condizioni di entrata | UtenteRegistrato deve aver svolto il caso d’uso: UCLOGIN  & UtenteRegistrato deve aver terminato il caso d’uso: UCVINFOEV | |
| Eccezioni |  | |
| Condizioni di uscita | UtenteRegistrato visualizza i messaggi dell’evento selezionato. | |
| Requisiti di funzionalità |  | |

|  |  |  |
| --- | --- | --- |
| Nome caso d’uso | Partecipazione evento | |
| ID | UCPEVENTO | |
| Partecipanti | UtenteRegistrato. | |
| Flusso di eventi: | UTENTE  1. UtenteRegistrato attiva la funzione partecipa | SISTEMA  2. Il sistema aggiorna le statistiche dell’evento precedentemente scelto nel caso d’uso: UCVINFOEV e notifica l’utente della riuscita dell’operazione |
| Condizioni di entrata | L’UtenteRegistrato deve aver svolto il caso d’uso: UCVISEVENTO | |
| Eccezioni |  | |
| Condizioni di uscita | L’UtenteRegistrato informa il sistema della sua eventuale partecipazione. | |
| Requisiti di funzionalità | Il sistema aggiorna il numero dei possibili partecipanti entro 10 minuto. | |

|  |  |  |
| --- | --- | --- |
| Nome caso d’uso | Convalida partecipazione evento standard | |
| ID | UCCONVPART | |
| Partecipanti | UtenteRegistrato | |
| Flusso di eventi: | UTENTE  1.UtenteRegistrato clicca sul bottone di validazione per attivare la funzione di validazione | SISTEMA  2.Il sistema prende atto della partecipazione e aggiorna le statistiche relative all’evento selezionato |
| Condizioni di entrata | UtenteRegistrato deve aver svolto il caso d’uso: UCLOGIN  & UtenteRegistrato deve aver completato il caso d’uso: UCVISINFOEV  & UtenteRegistrato deve situarsi nei pressi del luogo specificato dell’evento richiesto | |
| Eccezioni |  | |
| Condizioni di uscita | Le statistiche dell’evento selezionato vengono aggiornate tenendo conto della nuova partecipazione convalidata | |
| Requisiti di funzionalità | Il sistema aggiorna il numero dei partecipanti entro 1 minuto | |

|  |  |  |
| --- | --- | --- |
| Nome caso d’uso | Visualizza info evento. | |
| ID | UCVINFOEV | |
| Partecipanti | UtenteRegistrato | |
| Flusso di eventi: | UTENTE  1. UtenteRegistrato clicca sul bottone di info associato ad un evento | SISTEMA  2.Il sistema restituisce le informazioni dell’evento a cui il bottone info era associato: tipologia dell’evento, nome dell’evento, la sua descrizione, la locazione e la fascia oraria in cui si svolge, info sul feedback |
| Condizioni di entrata | UtenteRegistrato deve aver svolto il caso d’uso: UCLOGIN. | |
| Eccezioni |  | |
| Condizioni di uscita | Le info dell’evento sono visibili ad UtenteRegistrato | |
| Requisiti di funzionalità | Il sistema fornisce le informazioni entro 10 secondi | |

|  |  |  |
| --- | --- | --- |
| Nome caso d’uso | Visualizza profilo personale | |
| ID | UCVPROFILOUT | |
| Partecipanti | UtenteRegistrato | |
| Flusso di eventi: | UTENTE  1.UtenteRegistrato attiva la funzione “Profilo” cliccando sul bottone di visualizzazione profilo | SISTEMA  2.Il sistema restituisce ad UtenteRegistrato le informazioni che ha emesso durante la registrazione più gli eventi creati da lui creati ed il suo rating |
| Condizioni di entrata | UtenteRegistrato deve aver svolto il caso d’uso: UCLOGIN | |
| Eccezioni |  | |
| Condizioni di uscita | UtenteRegistrato ha a disposizione i dati fornitigli dal sistema | |
| Requisiti di funzionalità | Il sistema fornisce all’UtenteRegistrato la pagina profilo entro 10 secondi | |

|  |  |  |
| --- | --- | --- |
| Nome caso d’uso | Segnalazione commento | |
| ID | UCSEGNALAZIONEMESS | |
| Partecipanti | UtenteRegistrato | |
| Flusso di eventi: | UTENTE  1.L’UtenteRegistrato valuta un commento non adatto e attiva la funzione “Segnala Commento”. | SISTEMA  2.Il sistema accoglie la richiesta e invia la segnalazione al Moderatore  3.Il sistema notifica all’utente l’invio della segnalazione |
| Condizioni di entrata | UtenteRegistrato deve aver svolto il caso d’uso: UCVISCOMMENTI | |
| Eccezioni |  | |
| Condizioni di uscita | Il sistema prende in considerazione la richiesta | |
| Requisiti di funzionalità |  | |

|  |  |  |
| --- | --- | --- |
| Nome caso d’uso | Accettazione segnalazione evento | |
| ID | UCACCSEGNALAZIONEEV | |
| Partecipanti | Moderatore | |
| Flusso di eventi: | UTENTE  1. Moderatore accoglie la richiesta di un utente segnalatore mediante segnalazione del sistema  2. Moderatore giudica l’item segnalato e segnala l’utente inviante del messaggio incriminato attivando la funzione “Segnala UtenteRegistrato”. | SISTEMA  3.Il sistema accoglie la richiesta inviando una notifica di avviso all’utente incriminato. |
| Condizioni di entrata | Deve essere stata fatta una segnalazione di un messaggio da parte di un utente | |
| Eccezioni |  | |
| Condizioni di uscita | L’utente inviante del messaggio incriminato riceve la notifica e il suo account viene sospeso e viene cancellato l’evento. | |
| Requisiti di funzionalità | Il sistema fornisce la segnalazione 20 minuti. | |

|  |  |  |
| --- | --- | --- |
| Nome caso d’uso | Accettazione segnalazione commento | |
| ID | UCACCSEGNALAZIONECM | |
| Partecipanti | Moderatore | |
| Flusso di eventi: | UTENTE  1. Moderatore accoglie la richiesta di un utente segnalatore mediante segnalazione del sistema  2. Moderatore giudica l’item segnalato e segnala l’utente inviante del messaggio incriminato attivando la funzione “Segnala UtenteRegistrato”. | SISTEMA  3.Il sistema accoglie la richiesta inviando una notifica di avviso all’utente incriminato. |
| Condizioni di entrata | Deve essere stata fatta una segnalazione di un messaggio da parte di un utente | |
| Eccezioni |  | |
| Condizioni di uscita | L’utente inviante del messaggio incriminato riceve la notifica e il suo account viene sospeso e viene cancellato il commento. | |
| Requisiti di funzionalità | Il sistema fornisce la segnalazione 20 minuti. | |

|  |  |  |
| --- | --- | --- |
| Nome caso d’uso | Segnalazione Evento | |
| ID | UCSEGNALAZIONEV | |
| Partecipanti | UtenteRegistrato | |
| Flusso di eventi: | UTENTE  1. UtenteRegistrato ritiene l’evento selezionato inappropriato e decide di attiva la funzione di segnalazione dell’evento | SISTEMA  2.Il sistema accoglie la richiesta e invia la segnalazione al Moderatore |
| Condizioni di entrata | L’UtenteRegistrato deve aver svolto il caso d’uso: UCINFOEV | |
| Eccezioni |  | |
| Condizioni di uscita | Il sistema prende in considerazione la richiesta | |
| Requisiti di funzionalità |  | |

|  |  |  |
| --- | --- | --- |
| Nome caso d’uso | Rifiuta Segnalazione | |
| ID | UCRIFSEGNALAZIONE | |
| Partecipanti | Moderatore | |
| Flusso di eventi: | UTENTE  1. Moderatore accoglie la richiesta di un utente segnalatore mediante segnalazione del sistema ed entra nella sezione incriminata  2. Moderatore giudica l’evento secondo i criteri standard e non attiva la funzione “Segnala UtenteRegistrato”. | SISTEMA  3. Il sistema elimina la segnalazione |
| Condizioni di entrata | UtenteRegistrato deve aver svolto il caso d’uso: UCSEGNALAZIONEEV | |
| Eccezioni |  | |
| Condizioni di uscita | La segnalazione viene eliminata | |
| Requisiti di funzionalità | Il sistema fornisce la segnalazione entro 20 minuti. | |

|  |  |  |
| --- | --- | --- |
| Nome caso d’uso | Registrazione Utente | |
| ID | UCREGISTRAZIONEUT | |
| Partecipanti | UtenteNonRegistrato | |
| Flusso di eventi | UTENTE  1.UtenteNonRegistrato attiva la funzione di registrazione alla piattaforma cliccando sul pulsante di SignUp  3. UtenteNonRegistrato inserisce le proprie informazioni anagrafiche, la mail che vuole utilizzare e la password da utilizzare per accedere al sistema all’interno del form proposto dal sistema  4. Utente non registrato conferma i dati inseriti  8.Utente recupera il codice e lo inserisce nel form | SISTEMA  2. Viene presentata la pagina di registrazione all’interno del browser web di UtenteNonRegistrato contenete il form di registrazione  5. Il sistema verifica la formattazione dei dati immessi.  6. Il sistema invia una e-mail all’indirizzo da confermare contente un codice di verifica  7.mostra un form in cui è possibile inserire tale codice  9. Il sistema conferma l’indirizzo e-mail e invia UtenteNonRegistrato sulla Homepage |
| Condizione di entrata | Utente non registrato acceda alla piattaforma | |
| Eccezioni | Al 6. UtenteNonRegistrato non completa il caso d’uso UCVERMAIL ed il sistema non abilita un nuovo account per UtenteNonRegistrato | |
| Condizioni di uscita | Utente ha un profilo personale sulla piattaforma | |
| Requisiti di funzionalità | Utente potrà accedere al suo nuovo profilo personale entro 5 minuti | |

|  |  |  |
| --- | --- | --- |
| Nome caso d’uso | Visualizzazione Eventi | |
| ID | UCVISUALIZZAZIONEEV | |
| Partecipanti | UtenteRegistrato | |
| Flusso di eventi | UTENTE  1. UtenteRegistrato attiva la funzione “Mappa” cliccando sul bottone di visualizzazione mappa | SISTEMA  2.Il sistema restituisce la mappa del territorio, visualizzando gli eventi creati precedentemente. |
| Condizione di entrata | UtenteRegistrato deve aver completato il caso d’uso UCLOGIN | |
| Eccezioni |  | |
| Condizioni di uscita | L’UtenteRegistrato ha la panoramica del territorio con i suoi eventi. | |
| Requisiti di funzionalità | La mappa sarà resa possibile entro 2 minuti. | |

|  |  |  |
| --- | --- | --- |
| Nome caso d’uso | Rating Evento | |
| ID | UCRATINGEV | |
| Partecipanti | UtenteRegistrato | |
| Flusso di eventi | UTENTE  1. UtenteRegistrato esprime il proprio giudizio tramite il form di rating | SISTEMA  2.Il sistema elabora il voto e aggiorna il  feedback dell’evento e dell’utente a termine votazione |
| Condizione di entrata | UtenteRegistrato deve aver svolto il caso d’ uso UCLOGIN  & UtenteRegistrato completa il caso d’ uso UCINFOEV | |
| Eccezioni |  | |
| Condizioni di uscita | Il feedback dell’evento visibile dagli utenti della piattaforma è cambiato. | |
| Requisiti di funzionalità | La modifica del feedback è assicurata entro 2 minuti dalla sottomissione del voto. | |

|  |  |  |
| --- | --- | --- |
| Nome caso d’uso | Registrazione Moderatore | |
| ID | UCREGISTRAZIONEMOD | |
| Partecipanti | Super-admin | |
| Flusso di eventi | UTENTE  1. Super-admin inserisce all’interno del form l’username dell’account che si vuole far diventare Moderatore e si inviano le informazioni al sistema | SISTEMA  2. Il sistema elabora le informazioni e rende UtenteRegistrato indicato nel form Moderatore, notifica l’avvenuta operazione a Super-admin |
| Condizione di entrata | Super-Admin deve aver effettuato il caso d’uso: UCLOGIN | |
| Eccezioni | L’username di UtenteRegistrato non è valido e l’invio dei dati è impossibilitato | |
| Condizioni di uscita | UtenteRegistrato indicato è diventato Moderatore | |
| Requisiti di funzionalità | Il sistema trasforma UtenteRegistrato in Moderatore entro 1 minuti | |

|  |  |  |
| --- | --- | --- |
| Nome caso d’uso | Rimuovi Moderatore | |
| ID | UCDELMOD | |
| Partecipanti | Super-admin | |
| Flusso di eventi | UTENTE  1. Super-admin inserisce all’interno del form l’username dell’account che si vuole privare dei privilegi da moderatore e si inviano le informazioni al sistema | SISTEMA  2. Il sistema elabora le informazioni e rende Moderatore indicato nel form UtenteRegistrato, notifica l’avvenuta operazione a Super-admin |
| Condizione di entrata | Super-Admin deve aver effettuato il caso d’uso: UCLOGIN | |
| Eccezioni | L’username di Moderatore non è valido e l’invio dei dati è impossibilitato | |
| Condizioni di uscita | Moderatore indicato è diventato UtenteRegistrato | |
| Requisiti di funzionalità | Il sistema trasforma Moderatore in UtenteRegistrato entro 1 minuti | |

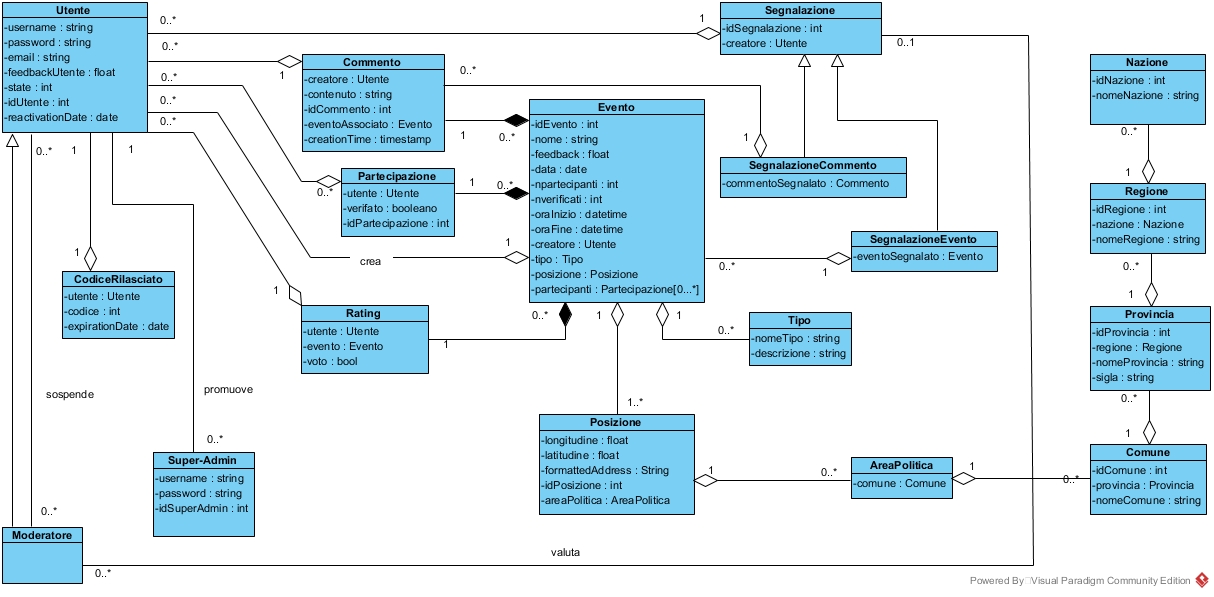
|  |  |  |
| --- | --- | --- |
| Nome caso d’uso | Visualizza pannello ricerca | |
| ID | UCVISRIC | |
| Partecipanti | Utente | |
| Flusso di eventi | UTENTE  1.Utente attiva la funzione “Ricerca” dalla pagina principale | SISTEMA  2.Il sistema restituisce ad Utente una pagina in cui sono presenti un selettore per scegliere il tipo di item da ricercare e varie proprietà per cui ricercare come: tipo di item da ricercare, keyword, fasce orarie, limiti di zona |
| Condizione di entrata | Utente deve aver svolto il caso d’ uso UCLOGIN | |
| Eccezioni |  | |
| Condizioni di uscita | Utente ha la possibiltà di utilizzare il pannello della ricerca | |
| Requisiti di funzionalità |  | |

|  |  |  |
| --- | --- | --- |
| Nome caso d’uso | Ricerca Evento | |
| ID | UCRICEEV | |
| Partecipanti | Utente | |
| Flusso di eventi: | UTENTE  1. Utente esprime la sua volontà di ricercare eventi attraverso un selettore  3. Utente compila il form con le informazioni che ritiene accurate per cercare l’evento desiderato e sottomette il form | SISTEMA  2.Il sistema presenta all’utente il form per l’inserimento di luogo, range chilometrico in cui ricercare, data e orario di inizio e data e orario di fine  4. Il sistema valuta la richiesta e mostra una pagina contente gli item che soddisfano i requisiti di ricerca ad Utente |
| Condizioni di entrata | Utente deve aver completato il caso d’uso UCVISRIC | |
| Eccezioni | Al 2. Il sistema potrebbe non trovare corrispondenze ai dati immessi e il sistema avvisa l’Utente con un avviso di mancata corrispondenza. | |
| Condizioni di uscita | L’Utente visualizza una lista di eventi corrispondenti ai dati di ricerca immessi al passo 3. | |
| Requisiti di funzionalità | Il sistema fornisce ad Utente la lista delle corrispondenze entro 40 secondi | |

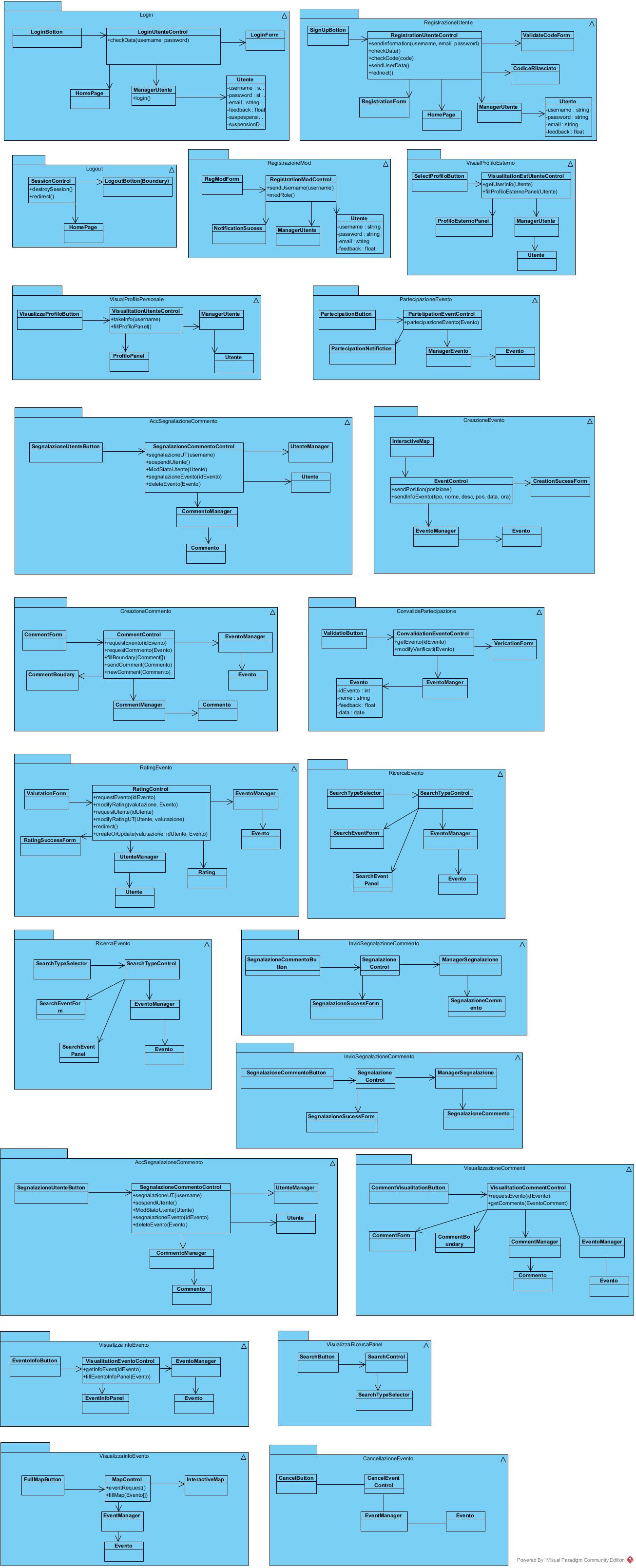
|  |  |  |
| --- | --- | --- |
| Nome caso d’uso | Ricerca Utente | |
| ID | UCRICEUT | |
| Partecipanti | Utente | |
| Flusso di eventi: | UTENTE  1. Utente esprime la sua volontà di ricercare utenti attraverso un selettore  3. Utente compila il form con le keyword che ritiene accurate per cercare l’utente desiderato e sottomette il form | SISTEMA  2.Il sistema presenta all’utente il form per l’inserimento delle keyword secondo cui effettuare la ricerca  4. Il sistema valuta la richiesta e mostra una pagina contente gli utenti che soddisfano i requisiti di ricerca ad Utente |
| Condizioni di entrata | Utente deve aver completato il caso d’uso UCVISRIC | |
| Eccezioni | Al 2. Il sistema potrebbe non trovare corrispondenze ai dati immessi e il sistema avvisa l’Utente con un avviso di mancata corrispondenza. | |
| Condizioni di uscita | L’Utente visualizza una lista di utenti corrispondenti ai dati di ricerca immessi al passo 3. | |
| Requisiti di funzionalità | Il sistema fornisce ad Utente la lista delle corrispondenze entro 40 secondi | |

### 3.4.3 Modello ad oggetti

**Class Diagram – Classi**



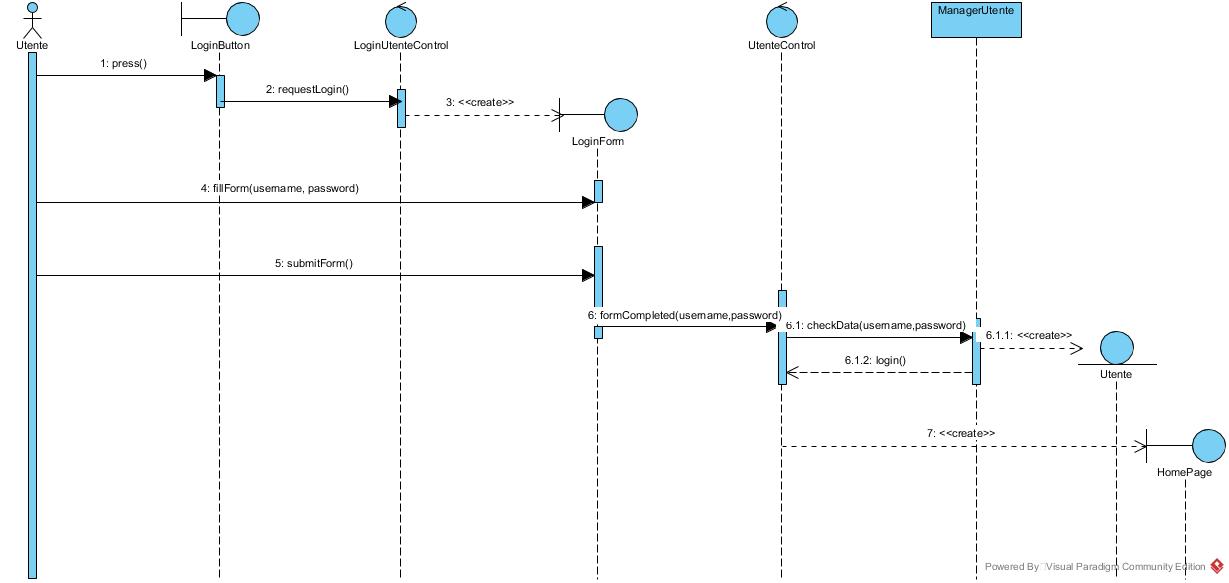
**Class Diagram– Casi d’uso**



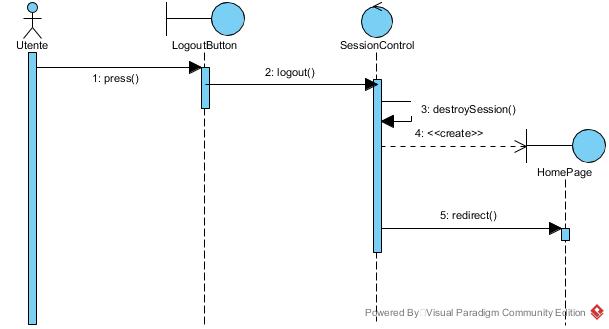
### 3.4.4 Modelli dinamici

**Sequence Diagram**

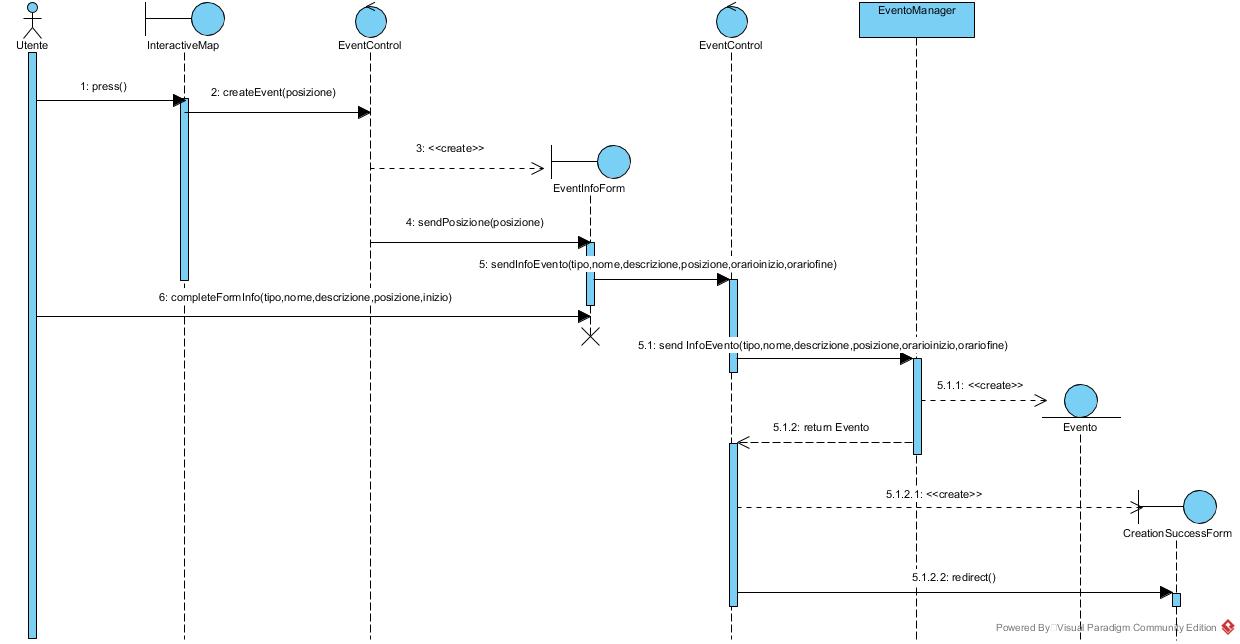
UCLOGIN



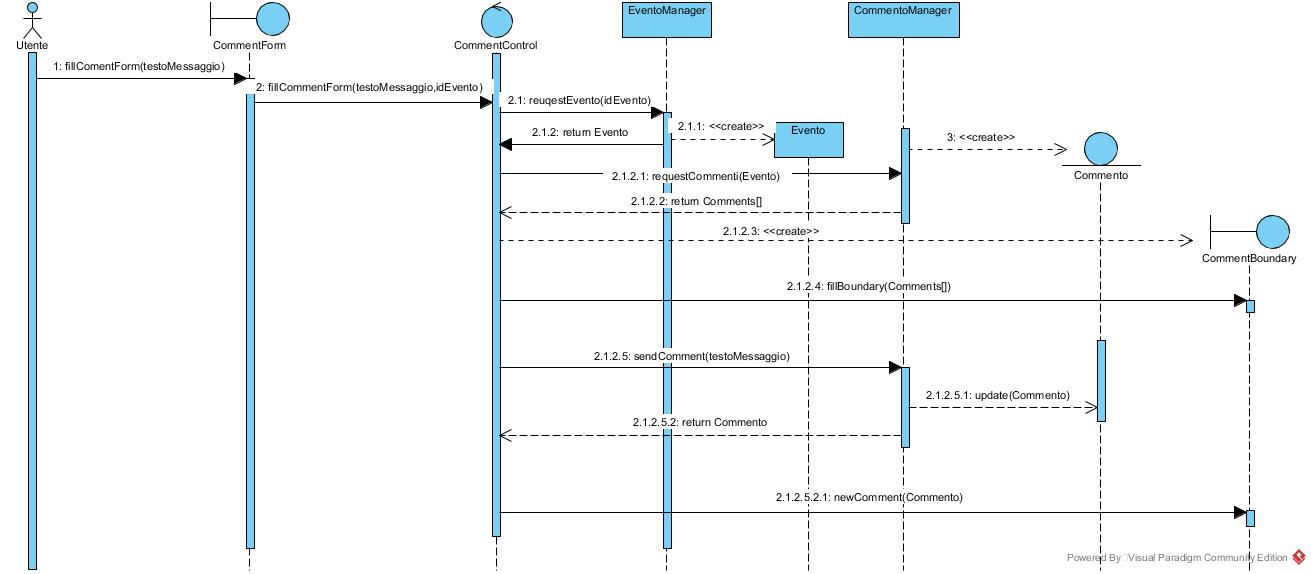
UCLOGOUT



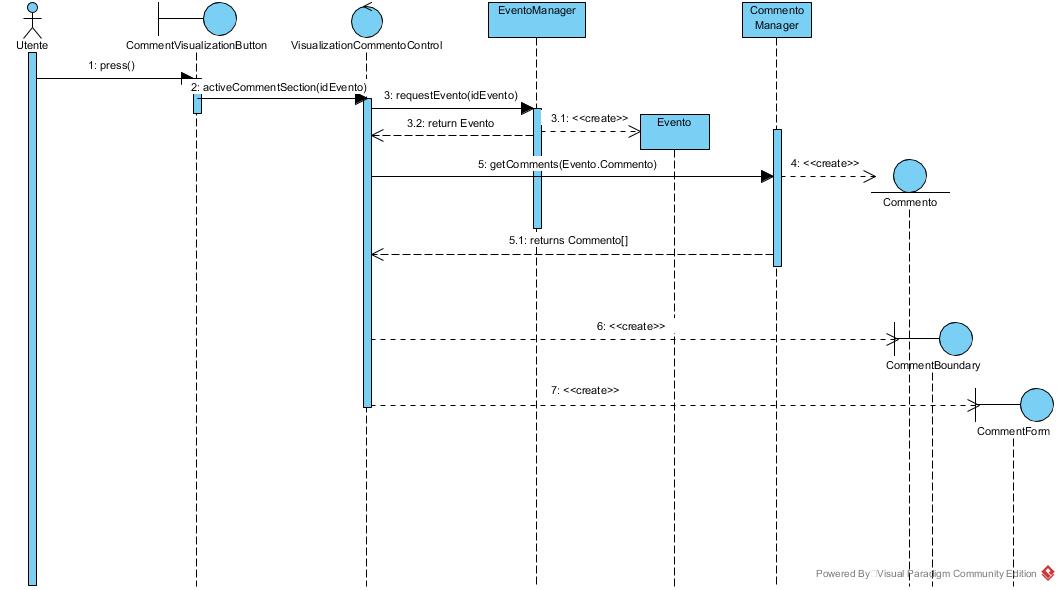
UCCEVENTO



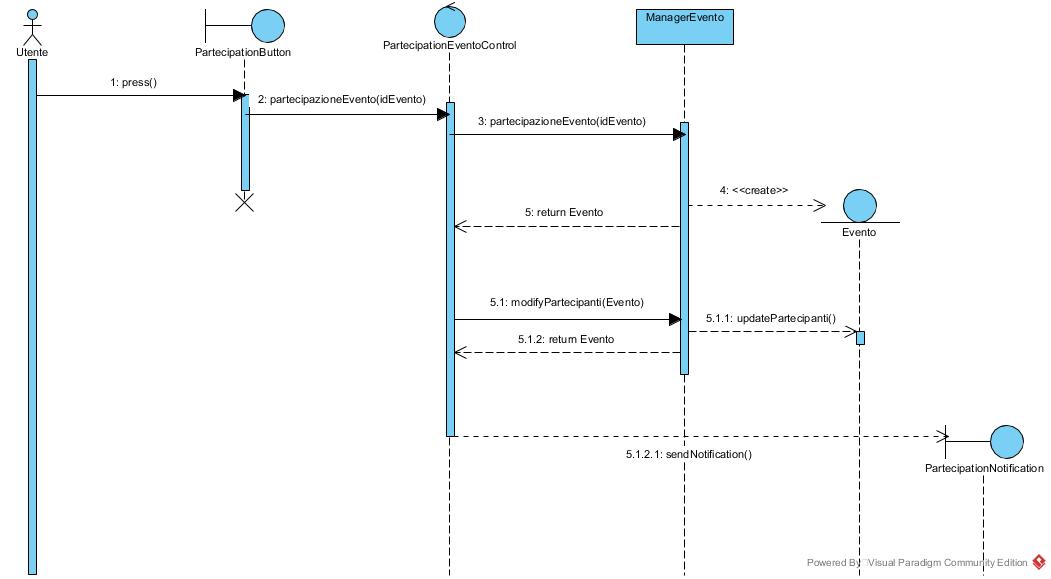
UCCOMMENTO



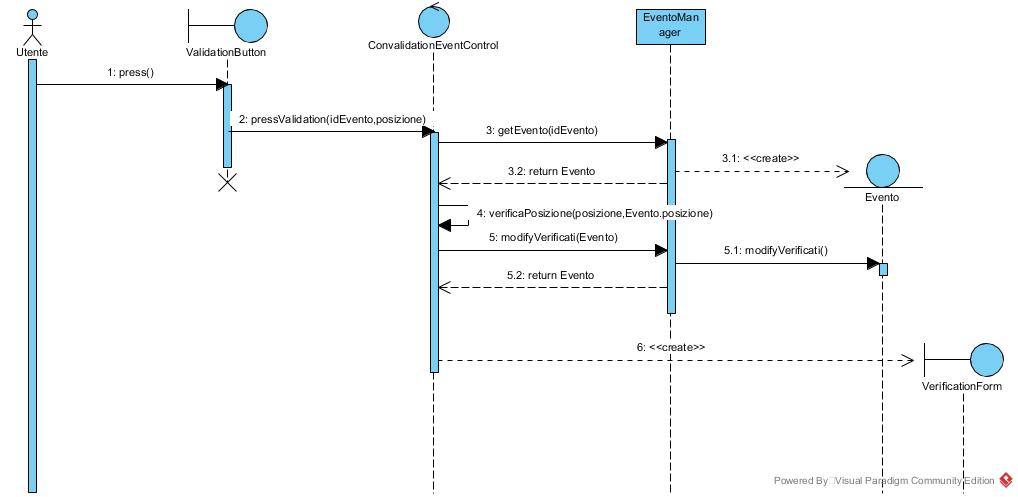
UCVISCOMMENTI



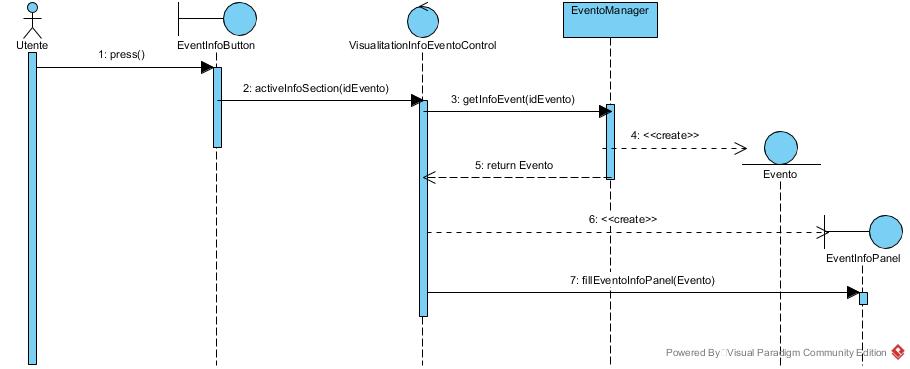
UCPEVENTO



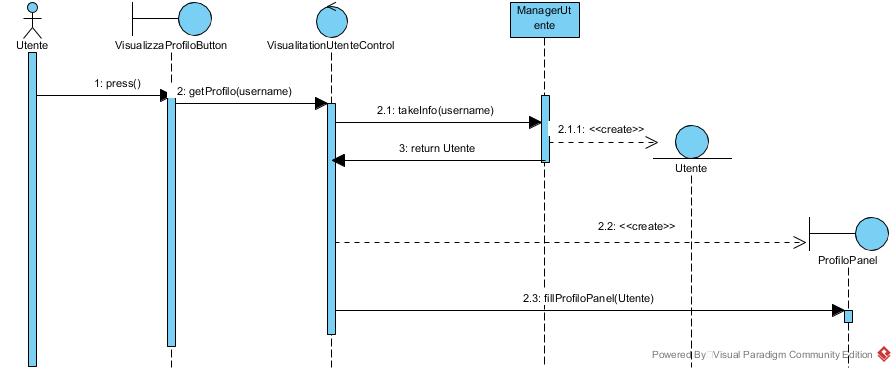
UCCONVPART



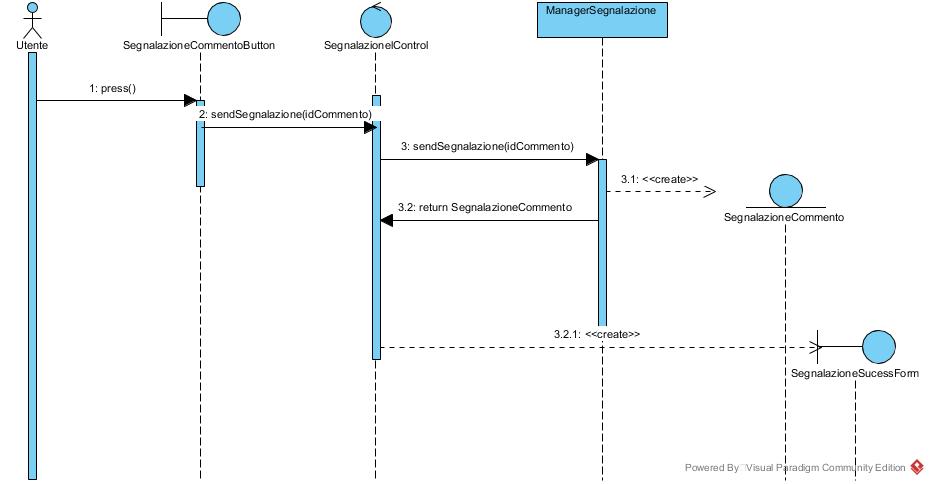
UCVINFOEV



UCVPROFILOUT



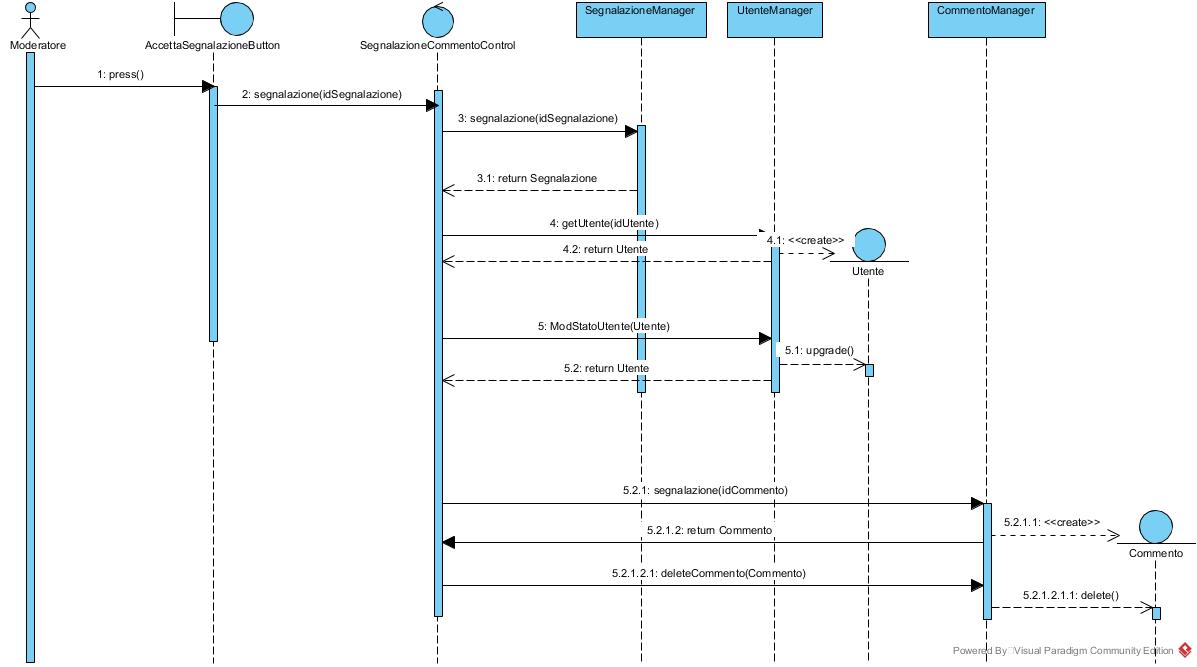
UCSEGNALAZIONEMESS



UCSEGNALAZIONEEV



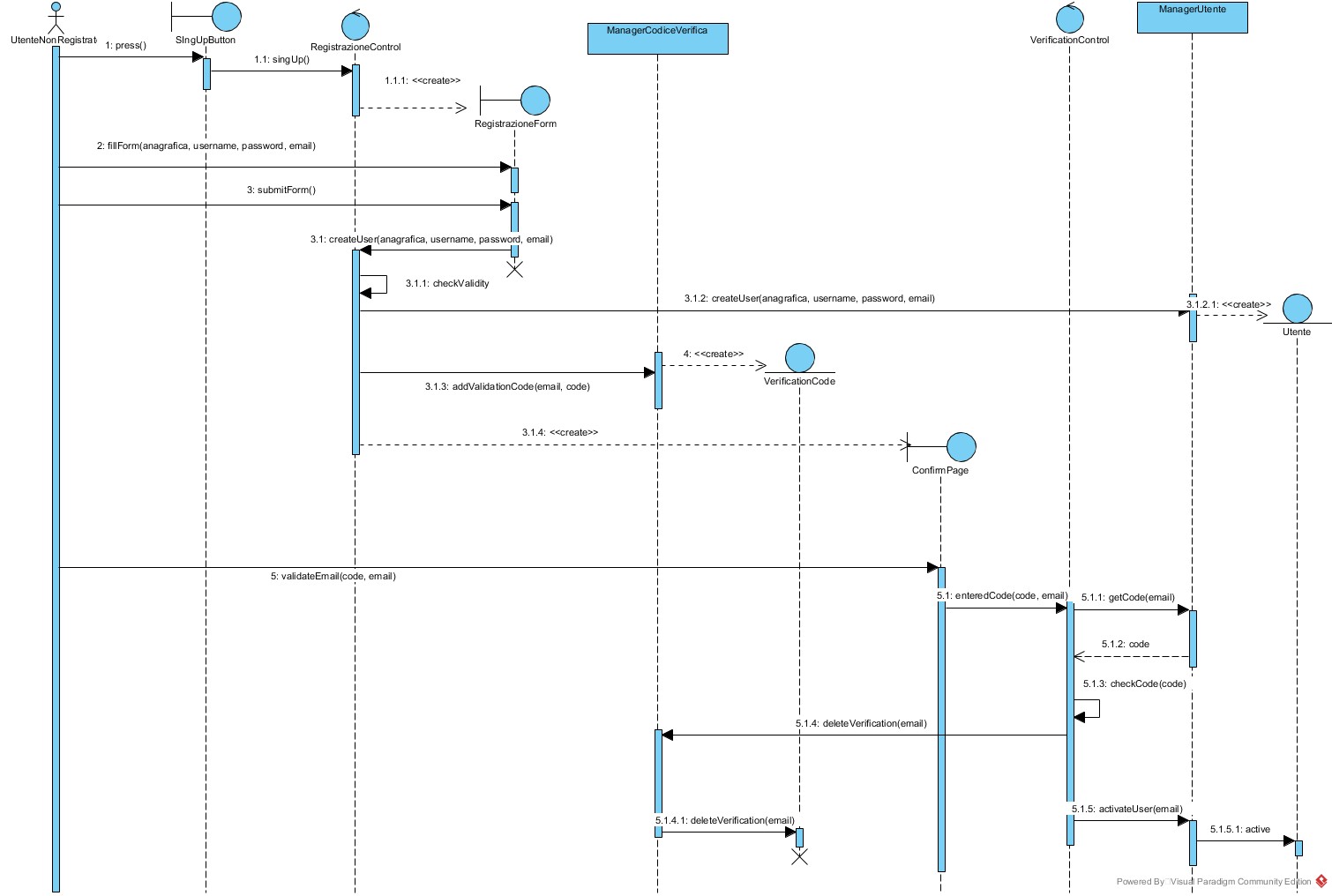
UCACCSEGNALAZIONECM



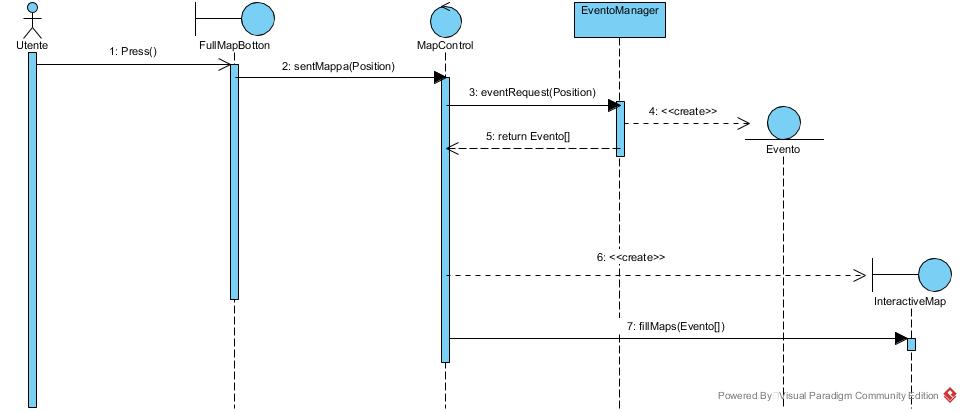
UCSEGNALAZIONEEV



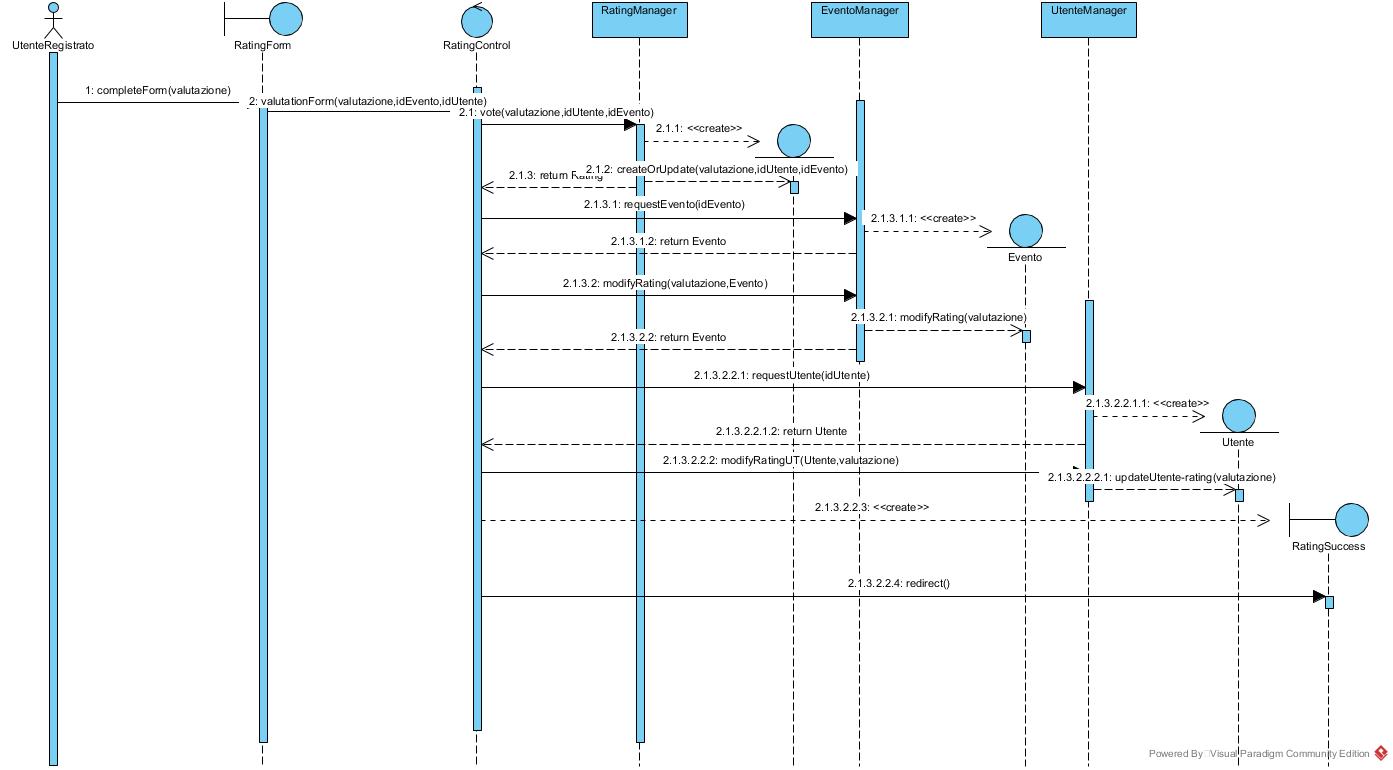
UCREGISTRAZIONEUT



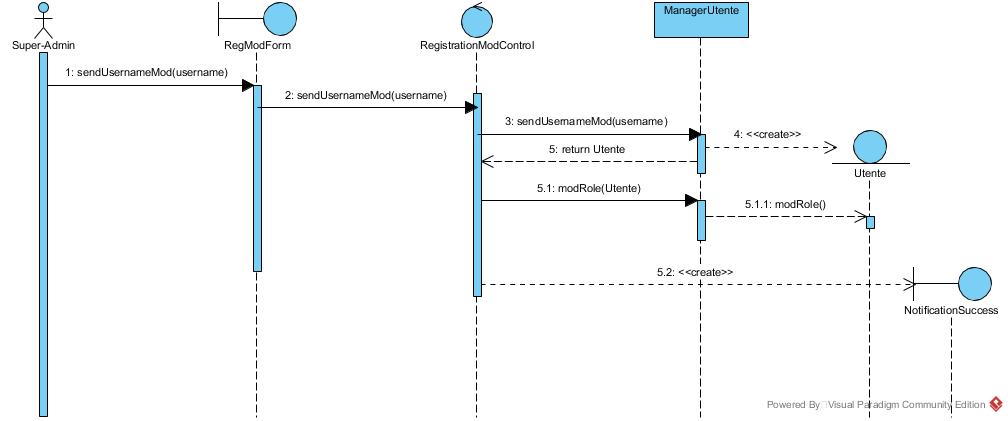
UCVISUALIZZAZIONEEV

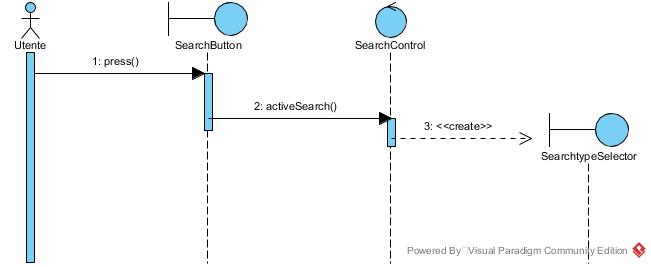


UCRATINGEV

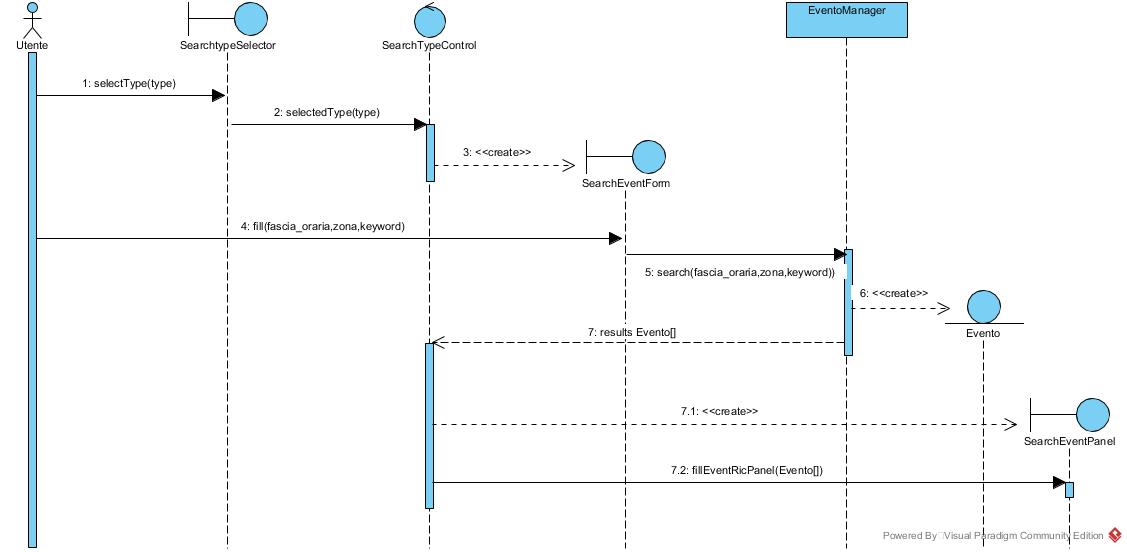


UCREGITSRAZIONEMOD

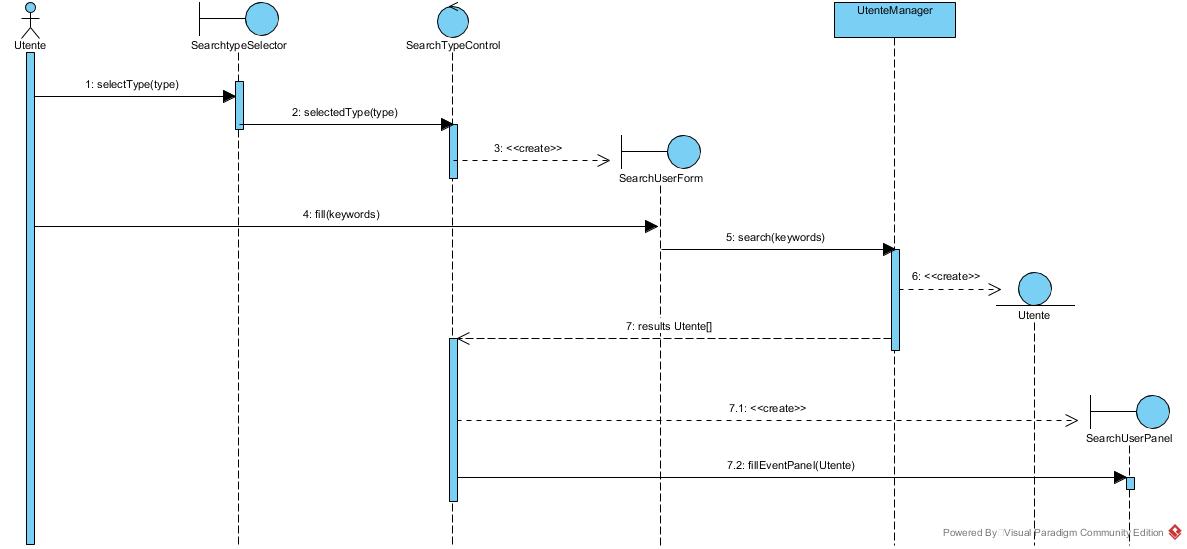


UCVISRIC

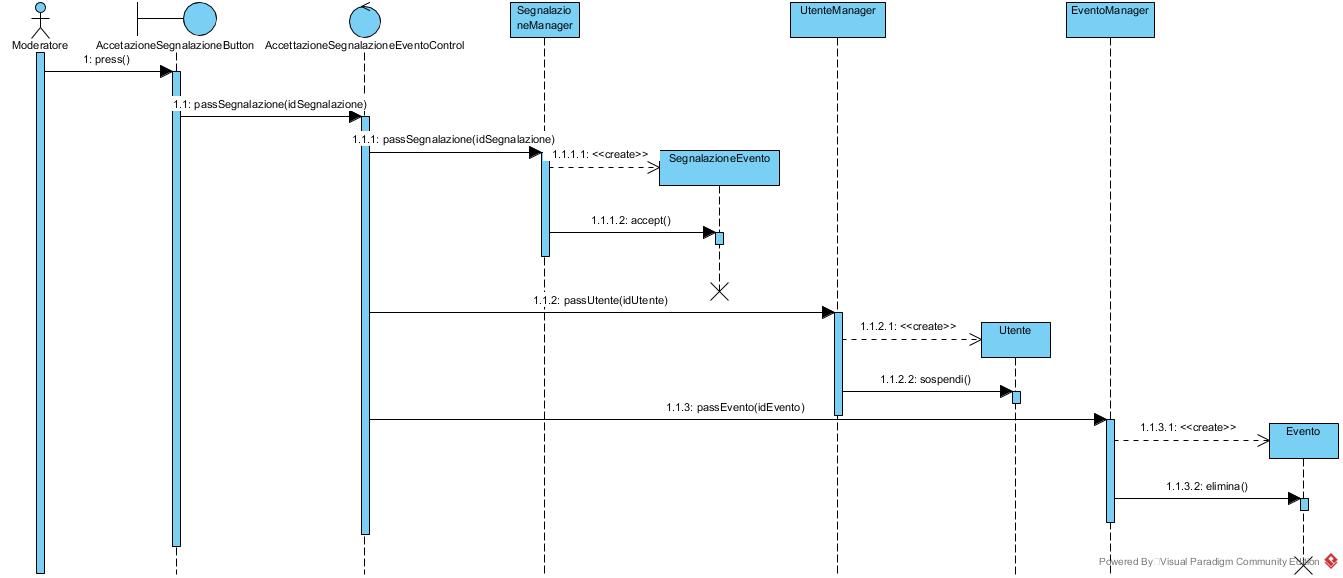
UCRICEEV



UCRICEUT



UCACCETTAZIONEEV

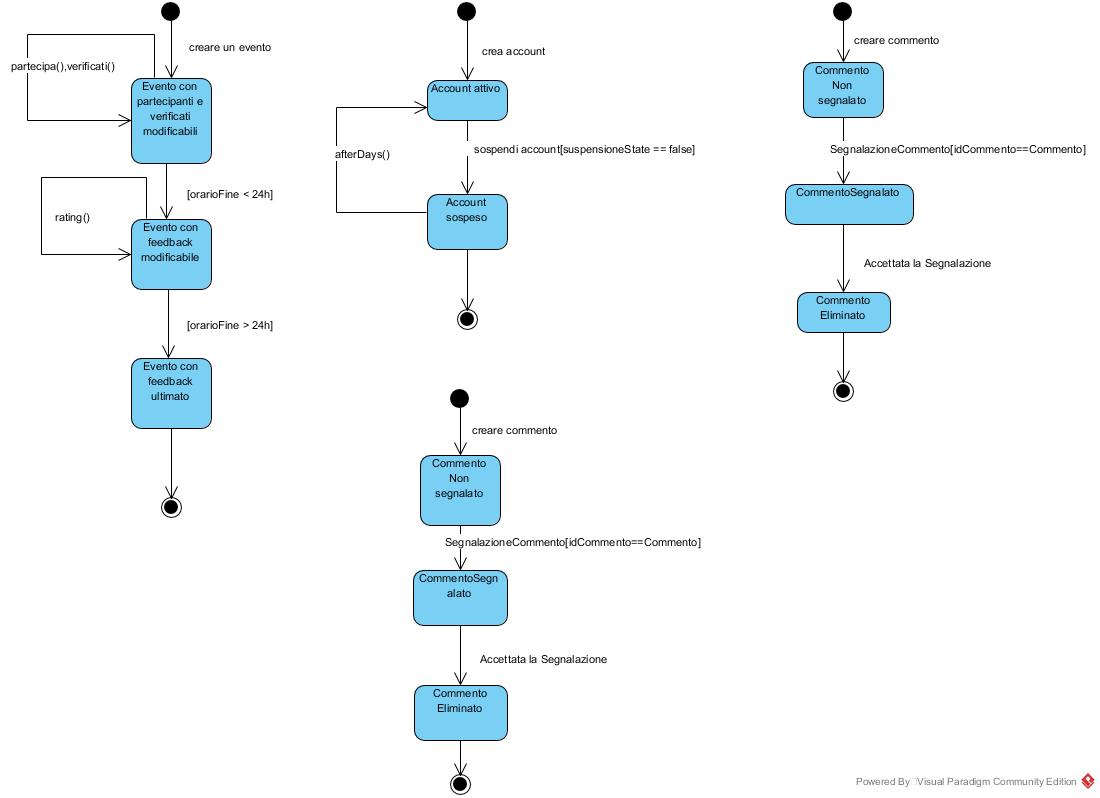


**STATECHARTDIAGRAM**

RatingEvento

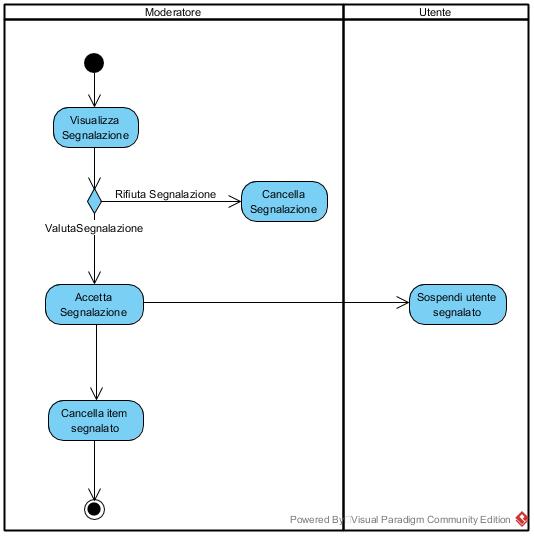
SospensioneAccount

Cancellazione Item(Evento/Commento)

****

**Activity Diagram**

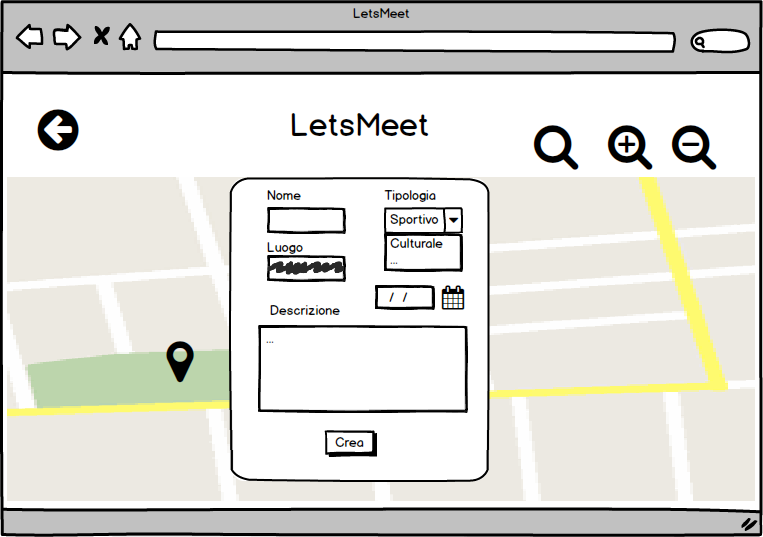
Activity Diagram – Segnalazione Item

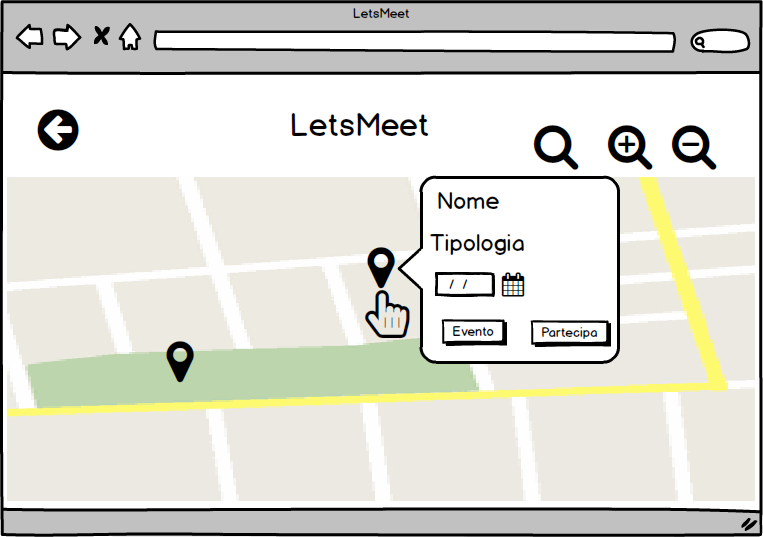


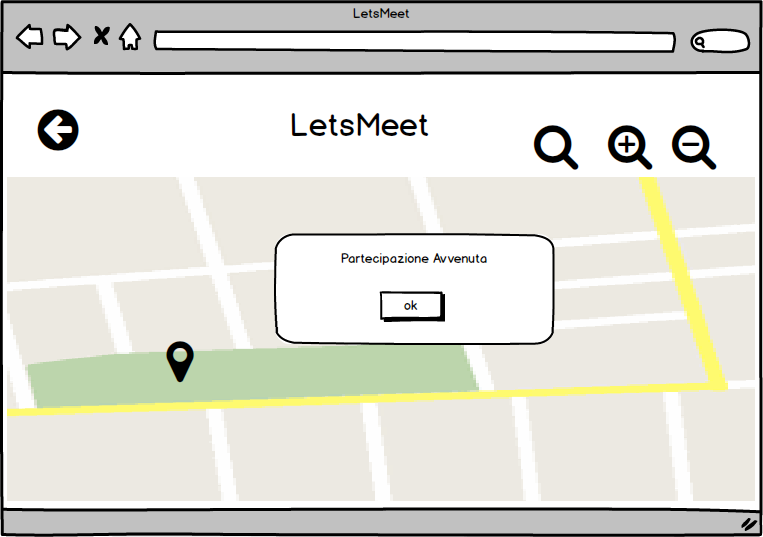
### 3.4.5 Interfaccia utente e mock-ups

### 

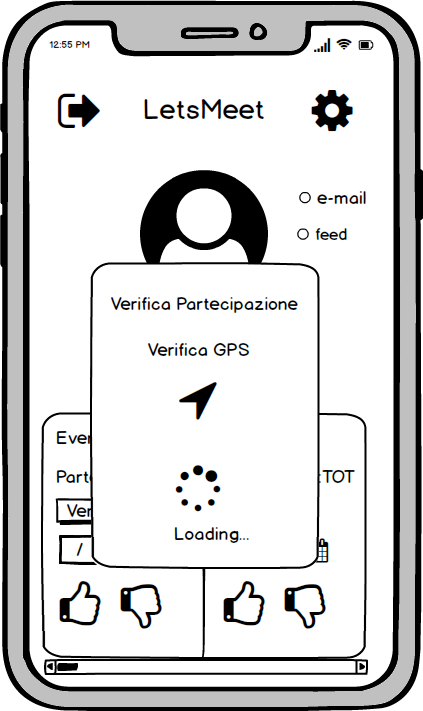
****

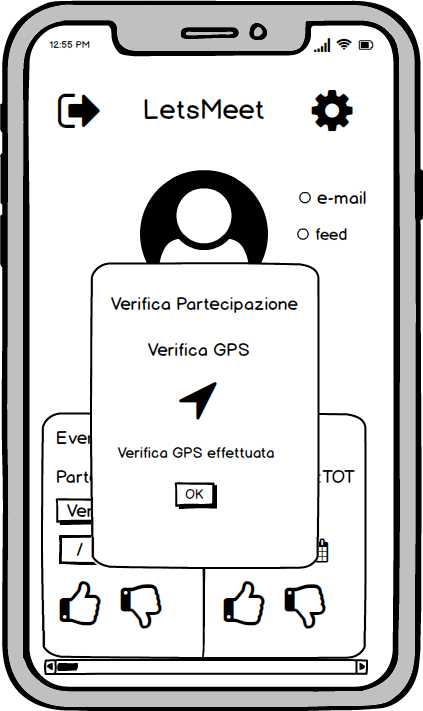
****

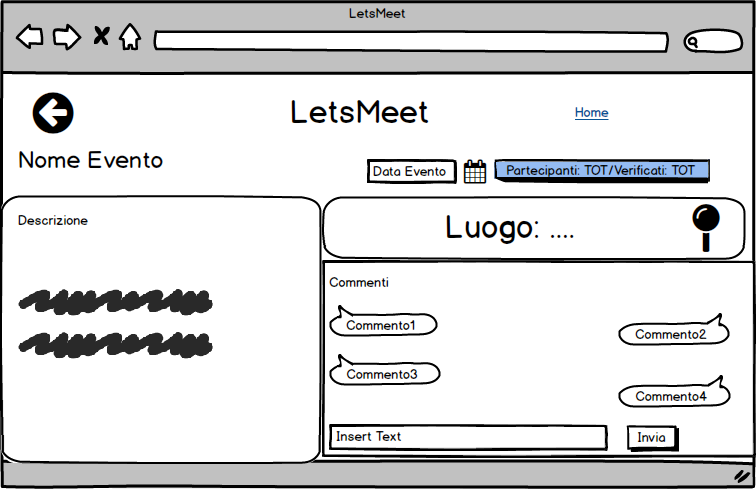
****

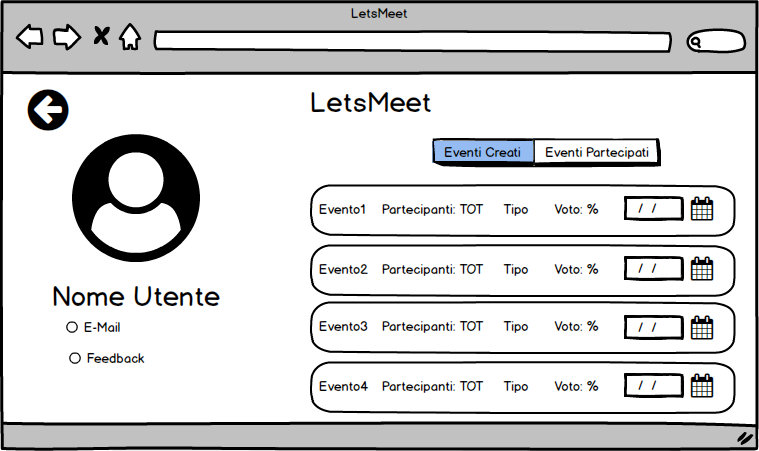
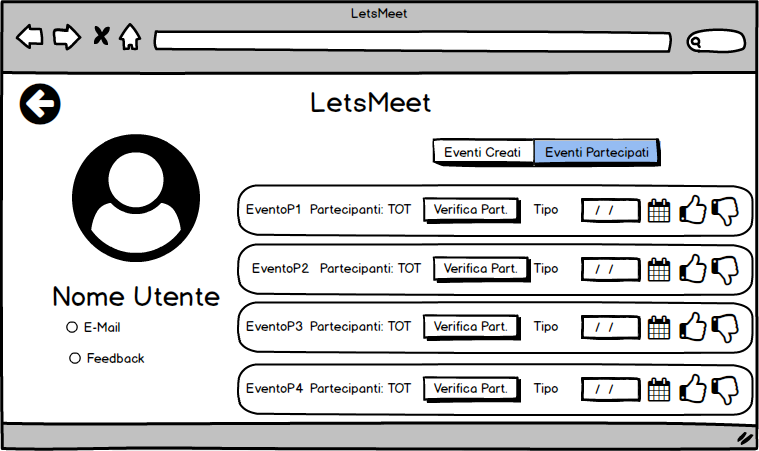
****

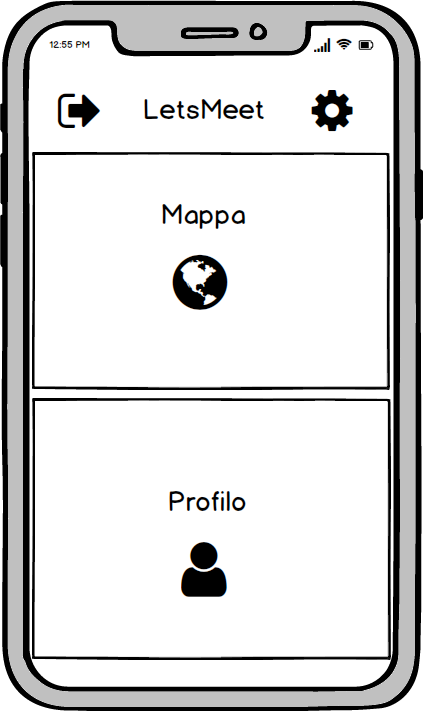
****

****

****

****

****

****

# 4. Glossario