

**Università degli Studi di Salerno**

**Corso di Ingegneria del Software**

**BeVoyager**

**RAD**

**Versione 3.5**

*BeVoyager*

**Data: 03/01/2017**

**Partecipanti:**

Nome	Matricola
Donato Tiano	0512102916
Alessandro Longobardi	0512102910
Paolo Zirpoli	0512102862
Salvatore Ruggiero	0512103002

Scritto da:	
-------------	--

## Revision History

Data	Versione	Descrizione	Autore
13/10/2016	1.0	Presentazione degli scenari, descrizione dei requisiti funzionali e non funzionali e stesura del Problem Statement.	
21/10/2016	2.0	Descrizione e diagrammi di casi d'uso e requisiti speciali.	
04/11/2016	3.0	Aggiunta di sequence diagrams, class diagrams e data dictionary.	
01/01/2017	3.1	Revisione completa del Capitolo 1 e dei relativi sottoparagrafi.	
02/01/2017	3.2	Revisione completa del Capitolo 2 e dei relativi sottoparagrafi.	
02/01/2017	3.3	Revisione completa degli scenari.	
03/01/2017	3.4	Revisione completa dei casi d'uso.	
03/01/2017	3.5	Revisione completa di Object Model e Sequence Diagrams, Statechart Diagrams.	

Sommario	
1. Introduzione.....	7
2. Proposed System .....	7
2.1. Problem Statement .....	7
2.2 Obiettivi primari .....	8
2.3 Requisiti Funzionali .....	8
2.3 Requisiti Non Funzionali.....	9
2.4 Target Environment.....	11
2.5 System Models .....	12
2.5.1 Scenari .....	12
2.5.1.1 Accesso all'itinerario modificabile .....	12
2.5.1.1.1 Viaggio con sconosciuti a Londra.....	12
2.5.1.1.2 Ricerca di un albergo.....	12
2.5.1.2 Accesso all'itinerario non modificabile.....	13
2.5.1.2.1 Partecipazione a viaggio per Roma .....	13
2.5.1.3 Ricerca di itinerari in base ai filtri.....	13
2.5.1.3.1 Ricerca di itinerari per Stoccolma .....	13
2.5.1.4 Tour Operator .....	13
2.5.1.4.1 Diventare un Tour Operator .....	13
2.5.1.4.2 Rinnovare pacchetto Tour Operator .....	13
2.5.1.4.3 Ritiro guadagni.....	14
2.5.1.5 Feedback.....	14
2.5.1.5.1 Dare feedback ad un viaggio creato da un Tour Operator.....	14
2.5.1.5.2 Dare feedback ad un utente .....	14
2.5.1.5.2 Dare feedback ad un itinerario .....	14
2.5.1.6 Viaggio privato.....	15
2.5.1.6.1 Creazione di un viaggio privato .....	15
2.5.1.6.2 Adesione ad un viaggio privato.....	15
2.5.1.7 Profilo utente .....	15
2.5.1.7.1 Identificazione partecipanti .....	15
2.5.1.8 Info itinerario.....	16
2.5.1.8.1 Visualizzare informazioni su un itinerario .....	16
2.5.2 Use Case Model .....	17

2.5.2.1 Accesso alla pagina di un luogo .....	17
2.5.2.2 Accesso al profilo dell'utente .....	17
2.5.2.3 Accesso ad un itinerario .....	18
2.5.2.4 Acquisto Pacchetto Tour Operator .....	19
2.5.2.5 Acquisto viaggio creato ad Tour Operator .....	20
2.5.2.6 Segnalazione.....	21
2.5.2.7 Ricerca luogo.....	21
2.5.2.8 Ricerca itinerario per luogo .....	22
2.5.2.9 Eliminazione account .....	23
2.5.2.10 Ricerca utente .....	23
2.5.2.11 Eliminazione itinerario .....	24
2.5.2.12 Eliminazione luogo .....	24
2.5.2.13 Eliminazione feedback itinerario.....	25
2.5.2.14 Eliminazione feedback luogo .....	26
2.5.2.15 Eliminazione feedback utente .....	26
2.5.2.16 Creazione viaggio - Utente.....	27
2.5.2.17 Cancellazione viaggio .....	28
2.5.2.18 Cancellazione creazione viaggio .....	29
2.5.2.19 Annullamento Cancellazione creazione viaggio .....	29
2.5.2.20 Conferma viaggio .....	29
2.5.2.21 Creazione itinerario .....	30
2.5.2.22 Aggiunta luogo ad un itinerario già creato .....	30
2.5.2.23 Rimuovi luogo da un itinerario .....	31
2.5.2.24 Creazione sondaggio per aggiungere un luogo ad un viaggio.....	31
2.5.2.25 Votazione sondaggio .....	32
2.5.2.26 Dare feedback ad un luogo .....	32
2.5.2.27 Dare feedback ad un utente .....	33
2.5.2.28 Dare feedback ad un itinerario .....	34
2.5.2.29 Filtra viaggi per data.....	35
2.5.2.30 Filtra viaggi per luogo .....	35
2.5.2.31 Gestione notifiche.....	35
2.5.2.32 Imposta notifica come già letta .....	36
2.5.2.33 Login .....	36

2.5.2.34 Logout .....	37
2.5.2.35 Partecipazione ad un viaggio.....	37
2.5.2.36 Password dimenticata .....	38
2.5.2.37 Registrazione .....	39
2.5.2.38 Rinnovo Pacchetto Tour Operator.....	40
2.5.2.39 Aggiungere un luogo al database .....	41
2.5.3 Object Model .....	42
2.5.3.1 Data dictionary .....	42
2.5.3.2 Class Diagram .....	44
2.5.4 Dynamic Models .....	45
2.5.4.1 Sequence diagrams .....	45
2.5.4.1.1 Accesso al profilo utente .....	45
2.5.4.1.2 Accesso alla pagina di un itinerario .....	46
2.5.4.1.3 Accesso pagina di un luogo.....	47
2.5.4.1.4 Acquisto viaggio da TourOperator.....	48
2.5.4.1.5 Aggiunta Luogo In Un Itinerario Già Creato .....	49
2.5.4.1.6 Cancellazione Viaggio .....	50
2.5.4.1.7 Conferma Viaggio .....	51
2.5.4.1.8 Creazione sondaggio per aggiungere un luogo ad un viaggio.....	52
2.5.4.1.9 Creazione Itinerario .....	53
2.5.4.1.10 Creazione Viaggio.....	54
2.5.4.1.11 Dare Feedback Ad Un Itinerario .....	55
2.5.4.1.12 Dare Feedback Ad Un Luogo.....	56
2.5.4.1.13 Dare Feedback Ad Un Utente .....	57
2.5.4.1.14 Eliminazione feedback itinerario .....	58
2.5.4.1.15 Eliminazione feedback luogo .....	59
2.5.4.1.16 Eliminazione feedback utente .....	60
2.5.4.1.17 Eliminazione itinerario.....	61
2.5.4.1.18 Eliminazione luogo .....	62
2.5.4.1.19 Eliminazione Account .....	63
2.5.4.1.20 Filtra viaggi per data .....	63
2.5.4.1.21 Gestione notifiche .....	64
2.5.4.1.22 Imposta notifiche come già lette .....	65

2.5.4.1.23 Login .....	66
2.5.4.1.24 Logout .....	67
2.5.4.1.25 Partecipazione ad un viaggio.....	67
2.5.4.1.26 Password dimenticata - Recupero Password .....	68
2.5.4.1.27 Registrazione.....	69
2.5.4.1.28 Ricerca Itinerario In Base Al Luogo .....	70
2.5.4.1.29 Ricerca Luogo .....	71
2.5.4.1.30 Ricerca Utente.....	72
2.5.4.1.31 Rimuovi Luogo Da Itinerario.....	72
2.5.4.1.32 Rinnovo pacchetto Tour Operator.....	73
2.5.4.1.33 Segnalazione .....	74
2.5.4.1.34 Votazione Sondaggio .....	75
2.5.4.2 Statechart Diagrams .....	76
2.5.4.2.1 Utente .....	76
2.5.4.2.2 Route .....	76
2.5.4.2.3 Pacchetto Tour Operator .....	77
2.5.4.2.4 Feedback Luoghi .....	77

# 1. Introduzione

In questo documento verranno descritti tutti i requisiti che il sistema dovrà rispettare per fornire la giusta esperienza agli utenti. Verranno elencati in dettaglio i requisiti funzionali e non funzionali e, tramite la stesura di scenario e casi d'uso, tutte le possibili azioni che chi utilizza la piattaforma può fare, partendo dalla registrazione fino alle singole operazioni che possono essere eseguite.

## 2. Proposed System

### 2.1. Problem Statement

Un viaggiatore va incontro a molte difficoltà durante l'organizzazione del proprio viaggio.

La scelta della meta è solo una piccola parte di un'operazione molto più complessa.

Nella formazione di un gruppo per un viaggio, risulta necessario coordinare diverse persone con desideri differenti che potrebbero non essere pienamente d'accordo su tutte le scelte. C'è chi vuole visitare un luogo particolare, o partecipare ad un evento, e ,spesso, trovare un punto d'incontro è difficile.

Differentemente, una persona potrebbe non avere voglia d'impegnarsi nella creazione di un itinerario da seguire, e preferirebbe averne uno già pronto.

Oppure una persona con la voglia di viaggiare potrebbe non avere amici disposti a partire ma vorrebbe comunque condividere con qualcuno la sua avventura; ma incontrare le persone giuste proprio al momento in cui si vuole organizzare un viaggio è molto difficile.

Nell'organizzazione di un viaggio, bisogna inoltre decidere i vari luoghi da visitare, quali eventi seguire, come fare per non spendere troppo.

Molto spesso può succedere di non sapere quali eventi poter trovare in un determinato luogo, o un posto di cui non si conosceva l'esistenza e si finisce con il perdere un'opportunità.

L'organizzazione di un viaggio è un'operazione complicata, ancor di più se si guarda al fatto che tutte le fasi dell'organizzazione si svolgono in maniera dispersiva: si utilizza un sito per acquistare i biglietti aerei, un altro per cercare informazioni sui luoghi, ancora un altro per trovare eventi a cui si vuole partecipare.

Una persona non troppo organizzata finirà con il ritrovarsi sommerso da link salvati nei preferiti, pagine stampate dei più svariati itinerari e tanto stress.

Perché non avere una piattaforma dove poter trovare tutto questo? E' proprio questo l'obiettivo del nostro progetto.

## 2.2 Obiettivi primari

Gli obiettivi del progetto BeVoyager sono:

- 1) Fornire una piattaforma unica dove gli utenti siano in grado di gestire tutte le varie parti riguardanti l'organizzazione di un viaggio, dalla scelta dell'itinerario alla prenotazione dei biglietti, fino al controllo degli orari con comodità;
- 2) Semplificare la composizione dell'itinerario, mettendo in evidenza luoghi ed eventi interessanti che si possono seguire e permettendo di salvarli in un unico punto di facile consultazione;
- 3) Fornire una piattaforma condivisa, dove gruppi di utenti possano tutti collaborare senza difficoltà alle decisioni riguardanti il viaggio, mettendo a confronto varie possibilità in maniera semplice e concisa;

Favorire la scoperta e le nuove conoscenze, mettendo in comunicazione persone a distanza che condividono il desiderio di esplorare una meta comune.

## 2.3 Requisiti Funzionali

BeVoyager mette a disposizione i seguenti servizi:

### - Creazione di un viaggio

L'utente sarà in grado di creare un itinerario, comprensivo di luoghi da visitare e attività da seguire. L'itinerario può essere pubblico o privato, modificabile o meno.

### - Partecipare ad un viaggio

L'utente può decidere di partecipare ad un viaggio creato da altri utenti.

### - Votare per prendere decisioni relative al viaggio

Gli utenti che decidono di partecipare ad un viaggio "modificabile", possono tenere votazioni per decidere quali attività svolgere e quali luoghi visitare, in modo tale da offrire a tutti i partecipanti la possibilità di far valere le proprie idee.

### -Gestione utente



L'utente potrà gestire il proprio profilo, caricando foto, aggiungendo descrizioni riguardanti sé stesso e i propri interessi, etc...

**- Evidenziare offerte e sconti**

Il sistema permette la ricerca e la messa in evidenza di offerte relative all'itinerario definito, per esempio verranno evidenziate le presenze di coupon per sconti o particolari eventi che si tengono nei luoghi che si intende visitare

**- Possibilità di diventare tour operator**

Un utente può decidere di diventare "Tour Operator". Tale azione comporta la creazione di itinerari di viaggio predefiniti disponibili all'acquisto da parte di altri utenti che non desiderano creare un viaggio da zero. Un tour operator riceve un compenso ad ogni acquisto di un proprio itinerario.

**- Gestione dei feedback utenti**

Si possono lasciare feedback su altri utenti con cui si è viaggiato, in modo da dare la possibilità a coloro che utilizzano la piattaforma di rendersi conto se una persona è compatibile come compagno di viaggio o meno.

**- Gestione feedback itinerario**

Si possono lasciare feedback sugli itinerari seguiti, sui luoghi ed eventi, per dare la possibilità ad altri utenti di capire se un itinerario fa al caso proprio oppure no, o se un itinerario è stressante da seguire o meno

## **2.3 Requisiti Non Funzionali**

**- Performance**

Il sistema deve essere in grado di offrire un'esperienza d'uso fluida agli utenti con tempi di risposta minimi. Essendo una piattaforma web, visualizzabile da browser, l'interfaccia deve risultare leggera e con requisiti tecnici non elevati. Il sistema deve essere in grado di mostrare agli utenti le offerte migliori per voli e eventi.

**- Interface**

I risultati delle ricerche vengono mostrati in una lista all'interno della pagina web. Gli utenti possono selezionare con un click il risultato preferito. Le pagine relative ai viaggi devono visualizzare immagini e descrizioni testuali dei luoghi inclusi nell'itinerario.

**- Supportability**

- Il sistema deve essere espandibile, dando la possibilità di supportare nuove funzionalità, come nuovi motori di ricerca per luoghi ed eventi. Da ciò deriva un sistema modulare, capace di facilitare possibili aggiornamenti.

**- Implementation**

Il back-end di BeVoyager deve essere realizzato utilizzando la tecnologia Java Servlet come base. Il front-end utilizza HTML, CSS e Javascript per offrire una compatibilità attraverso tutti i browser disponibili. Il sistema deve rispettare il pattern MVC. Il sito utilizzerà componenti off-the-shelf.

**- Trasparency**

Le operazioni di input/output a sistemi esterni non devono essere trasparenti agli utenti della piattaforma. Ciò significa che non saranno presenti schermate di redirect verso pagine esterne, ma i risultati verranno visualizzati nella pagina web della piattaforma stessa.

**- Scalability**

La piattaforma deve essere in grado di poter accogliere una grande quantità di utenti che accedono ed operano contemporaneamente.

**- Availability**

Durante gli aggiornamenti della piattaforma il sistema deve comunque gestire le richieste e le risposte di un utente iscritto. Ciò significa minimizzare i tempi di downtime. Inoltre il sistema deve garantire all'utente la disponibilità di un supporto tecnico in caso di eventuali difficoltà o problemi strutturali.

**- Security**

Ogni utente/tour operator dovrà avere delle credenziali di accesso trattate in maniera sicura dal nostro sistema.

## 2.4 Target Environment

La piattaforma è di tipo web, in quanto tale viene acceduta tramite browser. Tutti gli utenti che utilizzano un browser aggiornato, con supporto a cookie e Javascript attivo devono essere in grado di poter visualizzare ed usufruire di tutte le funzionalità offerte dal sito senza problemi.

Il database della piattaforma viene realizzato in MySQL, con la possibilità di essere ospitato sia su macchine UNIX che Windows. L'applicazione Web verrà sviluppata tramite tecnologie Java. Inoltre il deployment del server di riferimento per il supporto del database e dell'applicazione Web sarà fornito da Amazon Web Server (AWS).

Il dominio della piattaforma sarà gestito da *Register.it*, un servizio di hosting che si occupa di questo.

## 2.5 System Models

### 2.5.1 Scenari

Gli scenari rappresentano degli utilizzi concreti fatti da utenti registrati che sfruttano le funzionalità offerte dalla piattaforma.

#### 2.5.1.1 Accesso all'itinerario modificabile

##### 2.5.1.1.1 Viaggio con sconosciuti a Londra

*Flusso di eventi:*

- 1) Salvatore ha voglia di viaggiare verso Londra, ma non trova amici che con lui vogliano condividere tale esperienza, e soprattutto non vogliono perdere tempo ad organizzare il tutto
- 2) Salvatore cerca "Londra" su BeVoyager, inserendo le sue preferenze, come il budget a disposizione.
- 3) Tra i risultati trova un itinerario pubblico e modificabile, creato da persone che non conosce.
- 4) Salvatore controlla l'itinerario proposto, lo trova interessante e di suo gradimento.
- 5) Salvatore controlla i feedback delle persone che stanno partecipando al viaggio al momento. Vede che hanno tutti buoni feedback.
- 6) Salvatore decide dunque di partecipare al viaggio, e si aggiunge ai partecipanti.

##### 2.5.1.1.2 Ricerca di un albergo

*Flusso di eventi:*

- 1) Salvatore è un utente di BeVoyager che ha aderito alla partecipazione di un viaggio verso la Papuasias. Dato che lui e gli altri partecipanti hanno finito di organizzarsi sull'itinerario, vogliono scegliere un albergo conveniente dove pernottare. Salvatore quindi accede alla piattaforma BeVoyager.
- 2) Salvatore effettua il login e gli appare la homepage, dalla quale può spostarsi nella lista degli itinerari stabiliti. Tra questi c'è "Viaggio in Papuasias" che era proprio quello che Salvatore cercava. Quindi accede al link relativo alla pagina.
- 3) Salvatore a questo punto clicca sulla sezione "Scegli albergo" che lo porta immediatamente ad una form di ricerca. Salvatore quindi inserisce la destinazione in cui si troverà l'albergo in cui pernosterà.
- 4) Quello che Salvatore ha inserito viene passato alla piattaforma Trivago, che tramite il proprio motore di ricerca genera una lista di alberghi economici dove potrà pernottare.
- 5) Infine a Salvatore appare la lista di alberghi, la scorre, e seleziona quello che gli piace.

## **2.5.1.2 Accesso all'itinerario non modificabile**

### **2.5.1.2.1 Partecipazione a viaggio per Roma**

*Flusso di eventi:*

- 1) Antonio è una persona che ama viaggiare, ma non ama perdersi in chiacchiere nell'organizzazione del viaggio così cerca su internet degli itinerari da intraprendere a Roma e accede alla piattaforma BeVoyager.
- 2) Dà un sguardo ai molti itinerari proposti nella capitale e inizia a cercare fra quelli proposti da altri utenti della piattaforma.
- 3) Una volta visti tutti, sceglie di registrarsi su BeVoyager e di accedere ad un itinerario proposto da un utente della piattaforma; infine sceglie di partecipare a tale itinerario.

## **2.5.1.3 Ricerca di itinerari in base ai filtri**

### **2.5.1.3.1 Ricerca di itinerari per Stoccolma**

*Flusso di eventi:*

- 1) Luca vuole fare un viaggio a Stoccolma, ma è totalmente ignaro dei luoghi che si possono visitare, quindi cerca su BeVoyager degli itinerari per Stoccolma e accede al sito.
- 2) Luca, dopo aver visto vari itinerari per la sua destinazione, ne trova due che gli piacciono e aggiunge entrambi alla sua pagina dei preferiti.

## **2.5.1.4 Tour Operator**

### **2.5.1.4.1 Diventare un Tour Operator**

*Flusso di eventi:*

- 1) Giorgio decide di voler diventare un Tour Operator e vuole farlo online. Il suo obiettivo è ricavare una percentuale dall'organizzazione di itinerari.
- 2) Giorgio si registra a BeVoyager e va alla pagina "Pacchetti Tour Operator", dove c'è una vasta scelta di pacchetti per la visibilità dell'itinerario che vorrà proporre.
- 3) Giorgio sceglie e clicca sul pacchetto che più fa comodo a lui, inserisce i suoi dati, e lo acquista per una certa somma di denaro.

### **2.5.1.4.2 Rinnovare pacchetto Tour Operator**

*Flusso di eventi:*

- 1) Giorgio è un tour operator di BeVoyager che ha acquistato un pacchetto "tour operator".
- 2) Trascorso il periodo di durata del pacchetto, esso scade.
- 3) A Giorgio viene data la possibilità di rinnovare il suo pacchetto.

4) Giorgio decide quindi di rinnovare il pacchetto acquistato in precedenza o sceglierne uno diverso.

#### **2.5.1.4.3 Ritiro guadagni**

*Flusso di eventi:*

- 1) Fabio è un tour operator di BeVoyager e vorrebbe ritirare i guadagni ottenuti dalle proposte di viaggi. Apre il browser e accede alla piattaforma BeVoyager.
- 2) Successivamente Fabio accede alla piattaforma, va nella sua pagina personale e clicca su ritira guadagni, dove gli viene mostrata la lista di itinerari pubblicati, il ricavo per ognuno di questi, e infine, il ricavo totale.
- 3) Alla fine Fabio clicca sull'inserzione per la politica di gestione del ricavo per i tour operator e su "Accetta".
- 4) Il ricavo gli viene trasferito sul codice IBAN inserito al momento della sua iscrizione.

#### **2.5.1.5 Feedback**

##### **2.5.1.5.1 Dare feedback ad un viaggio creato da un Tour Operator**

*Flusso di eventi:*

- 1) Aldo è appena tornato da un viaggio fatto con BeVoyager, lui ha scelto di fare un viaggio creato da un tour operator e vuole esprimere un parere su di esso.
- 2) Aldo allora si collega alla piattaforma, e nel suo profilo utente ritrova il viaggio che aveva scelto.
- 3) Utilizza la funzione di rilascio feedback sul viaggio, e dà il suo parere. Il feedback sul viaggio si aggiungerà a quelli visibili sulla pagina del tour operator che lo ha creato.

##### **2.5.1.5.2 Dare feedback ad un utente**

*Flusso di Eventi:*

- 1) Andrea è appena tornato da un viaggio in Nuova Zelanda, organizzato in maniera pubblica e modificabile con BeVoyager, e si sente di esprimere un parere sulle persone con cui ha viaggiato, e quindi accede al sito BeVoyager.
- 2) Andrea esegue l'accesso, e va nella pagina Viaggi, presente sulla sua pagina personale, e sceglie quello appena concluso.
- 3) Andrea ha viaggiato con due sconosciuti. Due, con uno si è trovato bene, mentre con l'altro non è andato d'accordo.
- 4) Andrea accede sulla pagina di ognuno di loro e lascia un parere anonimo, attraverso un commento, ed esprime il perchè si è trovato male o bene con tali persone, e soprattutto se sono affidabili o meno.

##### **2.5.1.5.2 Dare feedback ad un itinerario**

*Flusso di eventi:*

- 1) Andrea ha partecipato ad un viaggio a Miami, organizzato da lui ed dei suoi amici tramite

BeVoyager. Vuole quindi dare una sua opinione all'itinerario a cui ha preso parte in modo che altre persone interessate possano rendersi conto della qualità dell'itinerario.

2) Andrea, dopo aver effettuato il login, accede alla lista dei suoi itinerari. Da qui, gli compare una pagina relativa agli itinerari a cui ha preso parte e si sono conclusi.

3) Andrea accede alla pagina dell'itinerario scelto e va nella sezione relativa ai feedback, dove vengono visualizzati tutti i commenti lasciati dagli utenti.

4) Andrea scrive ed invia il suo feedback, che verrà visualizzato insieme agli altri.

## **2.5.1.6 Viaggio privato**

### **2.5.1.6.1 Creazione di un viaggio privato**

*Flusso di eventi:*

1) Antonio vuole viaggiare con degli amici e organizzarsi per andare a Venezia, ma non vuole che persone a lui sconosciute entrino a far parte di questa esperienza. Quindi accede a BeVoyager.

2) Antonio va sulla sezione "Crea Viaggio" e gli appare una pagina di condizioni, dove tra queste c'è la possibilità di dichiarare il viaggio privato o pubblico.

3) Antonio sceglie la condizione "Viaggio privato" e avanza alla creazione del viaggio, dove gli verrà chiesto quale amico, presente sulla piattaforma, vuole far partecipare al suo itinerario.

4) Quindi inserisce un gruppo di amici che vuole far partecipare e termina la creazione del viaggio.

### **2.5.1.6.2 Adesione ad un viaggio privato**

1) Giuseppe è un utente di BeVoyager e vi accede frequentemente. Antonio è un utente di BeVoyager che si fida di Giuseppe come compagno di viaggio e lo inserisce nel gruppo di partecipanti di un suo itinerario privato.

2) A Giuseppe arriva una notifica di Antonio riguardante la partecipazione al viaggio privato. Giuseppe quindi accede alla sezione "Notifiche" e trova la richiesta di Antonio.

3) Giuseppe allora aderisce alla sua partecipazione al viaggio cliccando su "Partecipa" e, da questo momento in poi, può visualizzare l'itinerario e organizzarsi con gli altri partecipanti.

## **2.5.1.7 Profilo utente**

### **2.5.1.7.1 Identificazione partecipanti**

*Flusso di eventi:*

1) Donato è un utente di BeVoyager che ha aderito ad una proposta di itinerario per Milano. Ora però Donato vuole sapere con chi ha a che fare in questo viaggio verso Milano, quindi accede alla piattaforma BeVoyager.

2) Dopo aver effettuato il login, accede a "I miei itinerari" e sulla pagina apparirà una lista di itinerari a cui ha aderito alla partecipazione. A questo punto Donato trova l'itinerario che stava cercando.

- 3) Successivamente Donato accede all'itinerario e gli appare la pagina relativa. Quindi Donato potrà conoscere la lista degli utenti che hanno aderito a tale itinerario.
- 4) Donato sceglie uno degli utenti, quindi accede alla sua pagina di profilo.
- 5) Donato quindi, una volta apparso il profilo, accede alla sezione "Feedback", ovvero quella relativa ai commenti e alle valutazioni delle persone che già hanno viaggiato con tale utente.
- 6) Infine Donato legge i feedback sull'utente in modo tale da sapere di più sulla persona con cui si potrà trovare a viaggiare.

## **2.5.1.8 Info itinerario**

### **2.5.1.8.1 Visualizzare informazioni su un itinerario**

*Flusso di eventi:*

- 1) Rosaria sta cercando un itinerario per la gita a Milano che dovrà fare con i genitori, e vuole avere più informazioni sui luoghi che potrebbe visitare. Cerca quindi su internet informazioni e possibili luoghi da visitare e va sul sito BeVoyager
- 2) Rosaria nota che ci sono molte mete interessanti da visitare a Milano. Una volta visualizzati, ne sceglie uno in particolare e decide di visualizzarne le informazioni per saperne di più.



## 2.5.2 Use Case Model

### 2.5.2.1 Accesso alla pagina di un luogo

**Attori partecipanti:** UtenteRegistrato

**Condizione d'entrata:**

- L'Utente Registrato ricerca un luogo da visitare

**Flusso di eventi:**

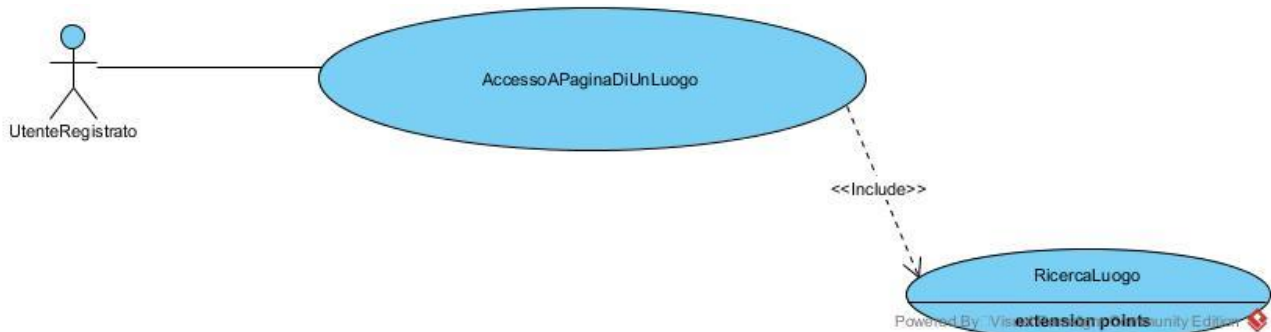
- Viene visualizzata una pagina che lista i risultati della ricerca effettuata dall'utente
- L'utente seleziona il risultato che desidera
- Vengono mostrati i contenuti relativi al risultato selezionato.

**Condizione di uscita:**

- L'utente esce dalla pagina del risultato

**Casi eccezionali:**

- L'Utente Registrato effettua una ricerca che non produce risultati.



### 2.5.2.2 Accesso al profilo dell'utente

**Attori partecipanti:** UtenteRegistrato

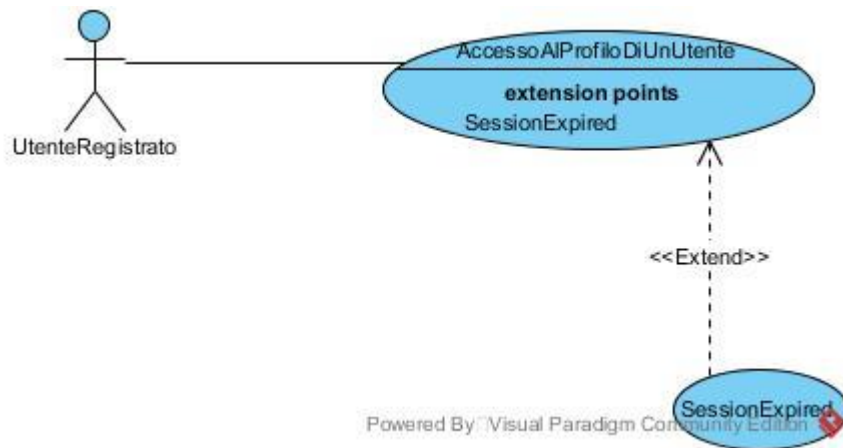
**Condizione d'entrata:** L'Utente Registrato clicca sul link che riporta al suo profilo

**Flusso di eventi:**

- Il sistema ricerca informazioni relative all'utente.
- Il sistema reindirizza l'utente sulla pagina del profilo relativa all'utente ricercato.

**Condizione di uscita:** L'utente visualizza la pagina.

**Casi eccezionali:** Dopo un certo periodo di tempo passato in maniera inattiva sul profilo, avverrà il logout dell'utente. non esiste. Utente non esistente.



### 2.5.2.3 Accesso ad un itinerario

**Attori partecipanti:** Utente

**Condizione d'entrata:** L'Utente ricerca un itinerario.

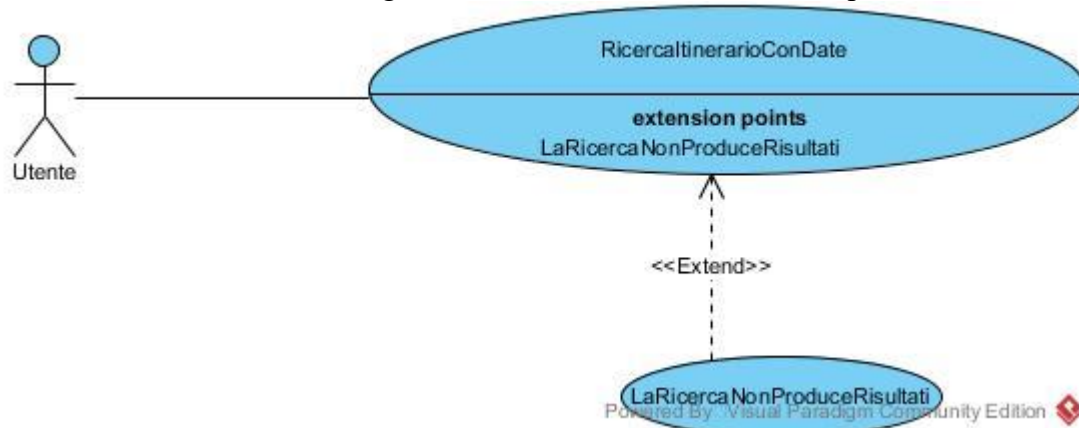
**Flusso di eventi:**

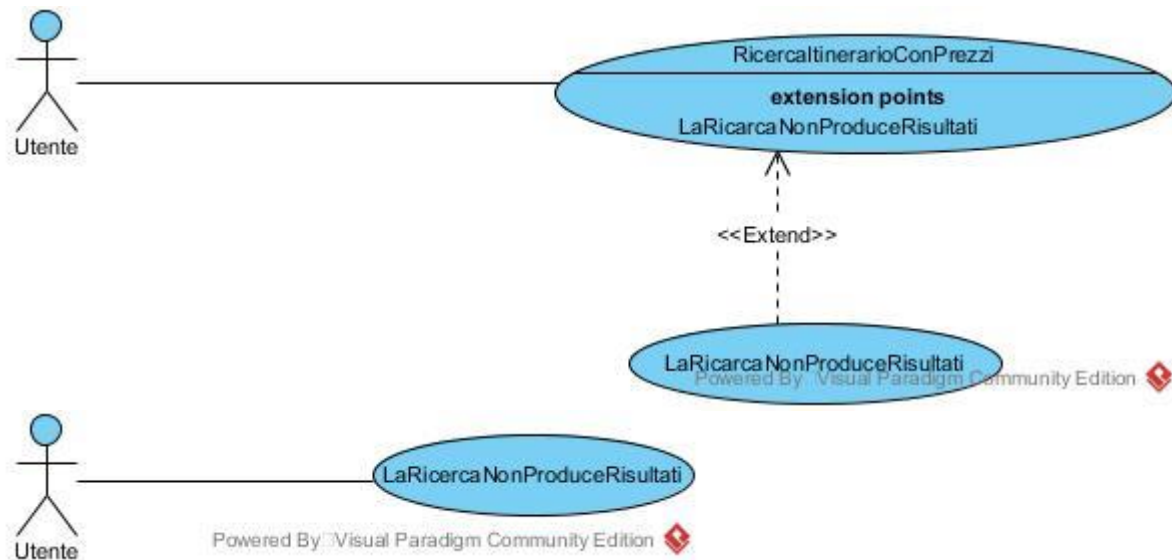
- Il sistema visualizza una pagina in cui vengono elencati i risultati della ricerca effettuata dall'utente
- L'utente seleziona il risultato che desidera
- Il sistema mostra la pagina contenente i contenuti relativi al risultato selezionato.

**Condizione di uscita:** L'utente visualizza la pagina prodotta.

**Requisiti speciali:** Nessuno

**Casi eccezionali:** L'Utente Registrato effettua una ricerca che non produce risultati.





### 2.5.2.4 Acquisto Pacchetto Tour Operator

**Attori:** UtenteRegistrato, SistemaPagamento

**Entry Condition:** L'Utente accede alla pagina relativa all'acquisto dei pacchetti per Tour Operator

**Flusso di eventi:**

-Seleziona il tipo di pacchetto che desidera

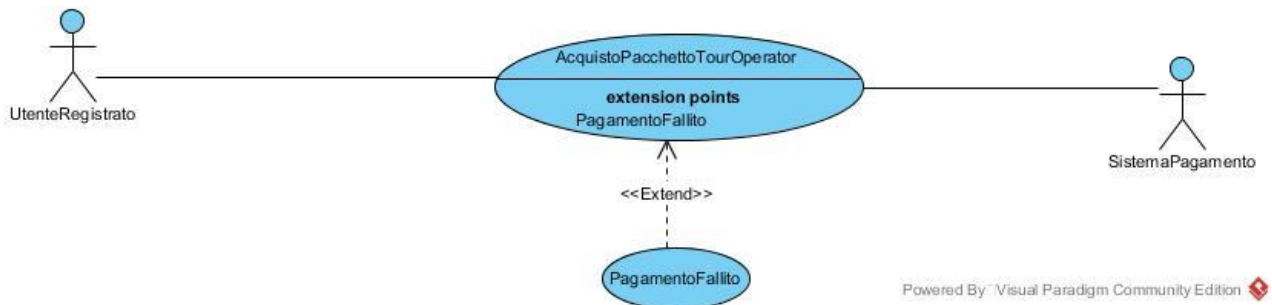
- Il sistema fornisce le informazioni relative al pacchetto selezionato

-L'utente inserisce i dati per il pagamento e da la conferma

- Il sistema effettua le operazioni relative al pagamento tramite il SistemaPagamento
- Il sistema riceve conferma da SistemaPagamento. A pagamento effettuato, il sistema aggiorna i dati dell'utente, fornendo i privilegi da Tour Operator

**Exit Condition:** L'utente diventa Tour Operator

**Caso Eccezionale:** Il pagamento non viene concluso.



### 2.5.2.5 Acquisto viaggio creato ad Tour Operator

**Attori:** UtenteRegistrato, SistemaPagamento

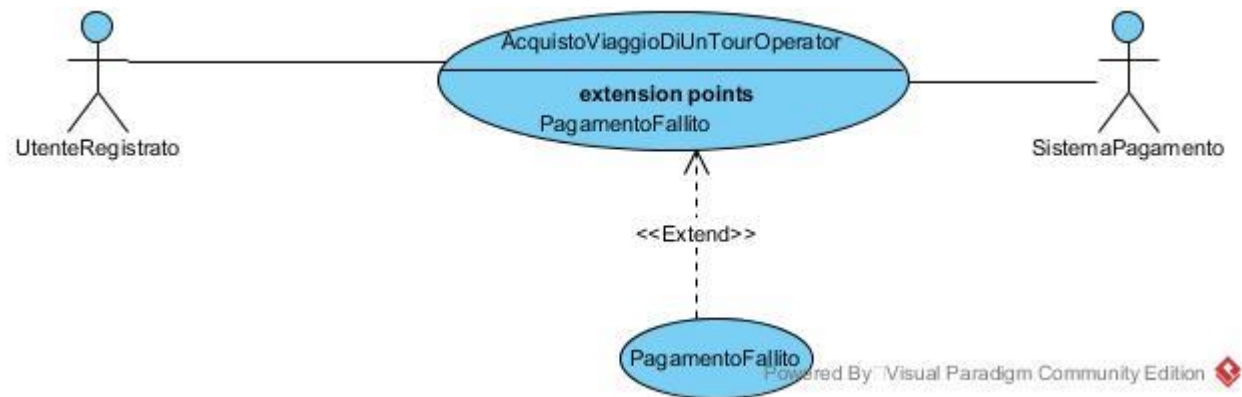
**Entry condition:** L'utente visualizza un viaggio organizzato da un Tour Operator

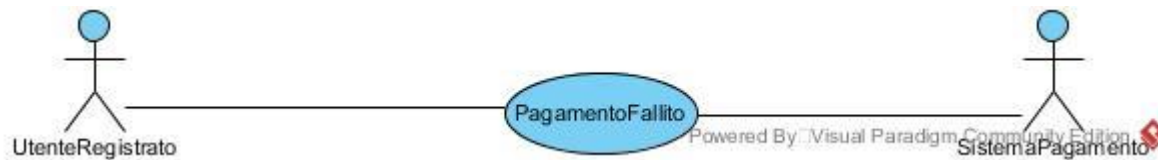
**Flusso di eventi:**

- L'utente clicca sull'opzione di acquisto
  - Il sistema mostra una schermata in cui l'utente può immettere i dati di pagamento
- L'utente inserisce i suoi dati ed accetta il pagamento
  - Il sistema riceve conferma da SistemaPagamento. A pagamento effettuato, il sistema aggiunge l'utente al viaggio come partecipante

**Exit Condition:** L'utente visualizza la pagina completa del viaggio

**Caso Eccezionale:** Il pagamento non viene concluso.





### 2.5.2.6 Segnalazione

**Attori:** Utente, SystemAdmin

**Entry Condition:** L'utente loggato ha visualizzato la pagina della segnalazione (che può essere un feedback o qualsiasi elemento di disturbo).

**Flusso di eventi**

-L'utente clicca sul link di segnalazione.

- Il sistema visualizza una form, dove l'utente può selezionare il tipo di contenuto da segnalare, una textbox dove l'utente può aggiungere ulteriori dettagli riguardo alla segnalazioni e un tasto per inviare la segnalazione.

-L'utente seleziona il contenuto critico, inserisce una descrizione e preme il tasto di invio

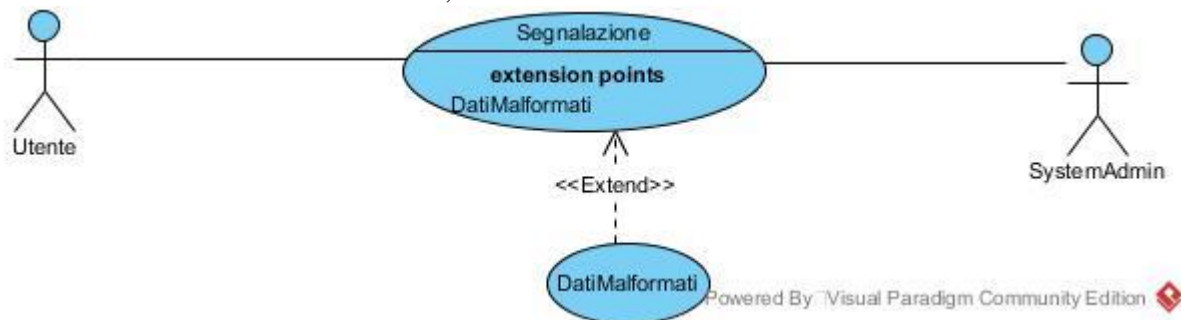
- Il sistema riceve la segnalazione e la memorizza nel database
- Il sistema inoltra la segnalazione al SystemAdmin

**Exit condition:**

**Utente:** Inoltra con successo la segnalazione e gli viene visualizzato un messaggio che lo avverte di ciò.

**SystemAdmin:** Visualizza la segnalazione.

**Casi d'errore:** Caduta connessione, L'utente inserisce dati mal formati



### 2.5.2.7 Ricerca luogo

**Attore:** Utente

**Entry Condition:** L'utente visualizza la pagina della ricerca

**Flusso di eventi:**

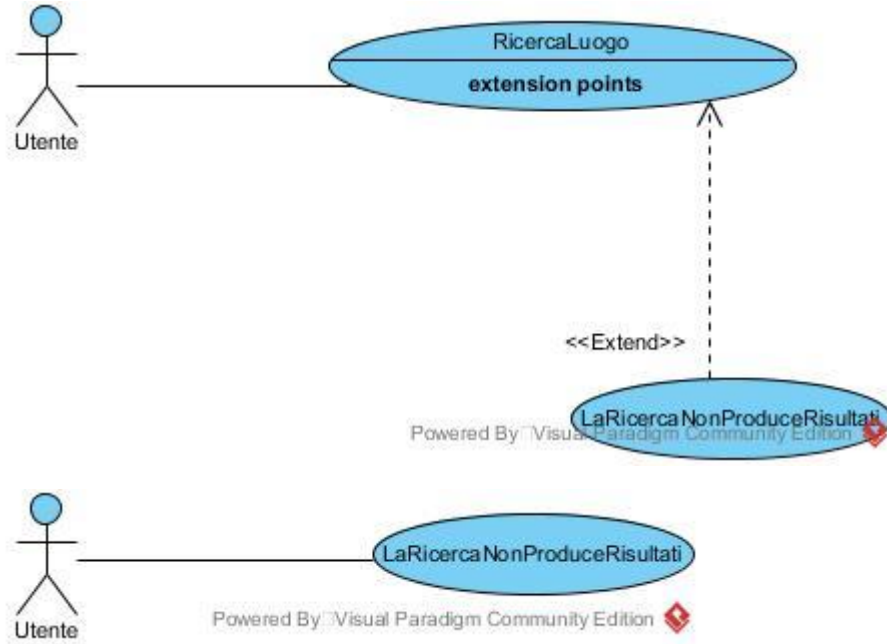
-L'utente inserisce il nome del luogo da ricercare

- Il sistema effettua la ricerca sul database, e restituisce una pagina in cui sono visualizzati i risultati della ricerca.

-L'utente seleziona il risultato corretto

**Exit condition:** Viene visualizzato il risultato selezionato.

**Casi d'errore:** Connessione d'errore, La ricerca non produce risultati



### 2.5.2.8 Ricerca itinerario per luogo

**Attore:** Utente

**Entry Condition:** L'utente visualizza la pagina della ricerca

**Flusso di eventi:**

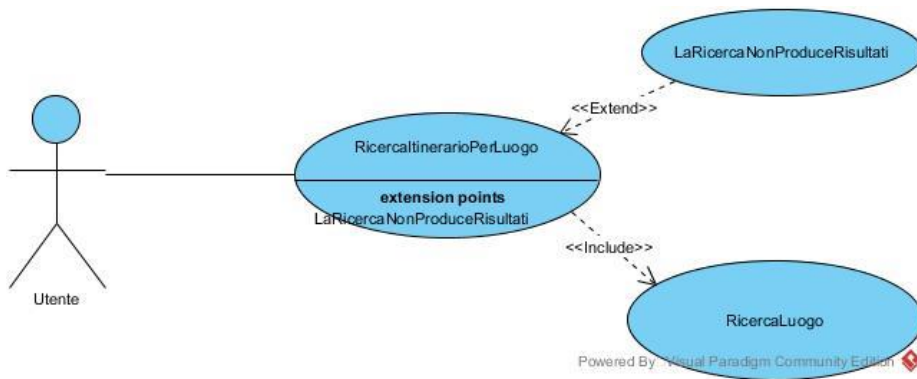
-L'utente inserisce il luogo che si desidera trovare all'interno di un itinerario.

- Il sistema effettua la ricerca sul database e restituisce una pagina in cui sono visualizzati i risultati della ricerca.

-L'utente seleziona il risultato corretto

**Exit condition:** Viene visualizzato il risultato selezionato.

**Casi d'errore:** Connessione d'errore, La ricerca non produce risultati



### 2.5.2.9 Eliminazione account

**Attore:** UtenteRegistrato

**Entry Condition:** L'utente visualizza la pagina del suo profilo

**Flusso di eventi**

- L'utente clicca sul tasto preposto all'eliminazione dell'utente.
  - Il sistema richiede una conferma.
- L'utente accetta.
  - Il sistema rimuove i dati dell'utente dal database.

**Exit Condition:** I dati vengono eliminati, l'utente viene informato della sua eliminazione.

**Casi eccezionali:** L'utente decide non accettare la conferma data dal sistema, la connessione cade, l'utente è già stato eliminato.



### 2.5.2.10 Ricerca utente

**Attore:** RegisteredUser

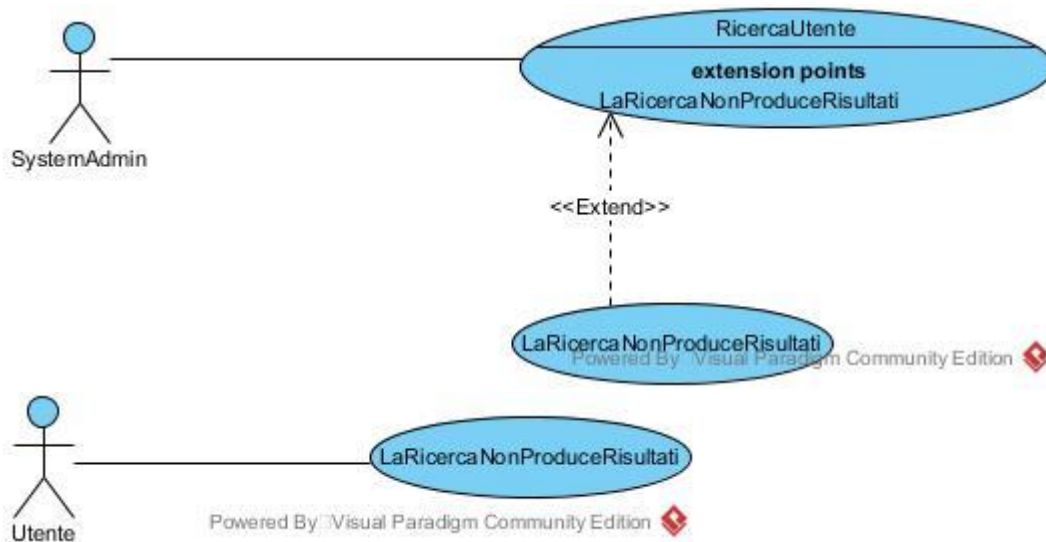
**Entry condition:** L'utente visualizza la pagina di ricerca

**Flusso di eventi:**

- L'utente inserisce nel form di ricerca lo username dell'utente da ricercare
  - Il sistema accede al database e ricerca l'utente dai dati inseriti nel form.
  - Il sistema prende i dati relativi all'utente e mostra la pagina del suo profilo.

**Exit condition:** L'utente visualizza la pagina dell'utente.

**Casi eccezionali:** La ricerca non produce risultati.



### 2.5.2.11 Eliminazione itinerario

**Attore:** SystemAdmin

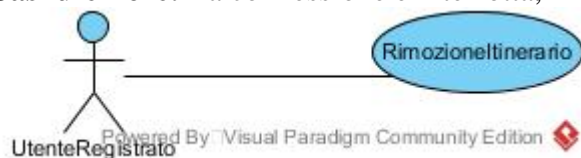
**Entry Condition:** L'amministratore accede alla piattaforma

**Flusso di eventi:**

- SystemAdmin effettua la ricerca dell'itinerario utilizzando le funzioni espresse nel caso d'uso relativo
- L'amministratore seleziona un itinerario da eliminare.
  - Il sistema visualizza la pagina dell'itinerario stesso, che, oltre alle normali opzioni disponibili a tutti gli utenti, visualizza anche delle operazioni specifiche per il SystemAdmin.
- Il SystemAdmin clicca sul tasto dedicato all'eliminazione dell'itinerario
  - Il sistema rimuove l'itinerario scelto dal database

**Exit Condition:** L'itinerario è eliminato correttamente

**Casi d'errore:** La connessione è interrotta, l'itinerario è già stato eliminato da un altro admin



### 2.5.2.12 Eliminazione luogo

**Attore:** SystemAdmin

**Entry Condition:** L'amministratore accede alla piattaforma

**Flusso di eventi:**

- SystemAdmin effettua la ricerca del luogo utilizzando le funzioni espresse nel caso d'uso relativo



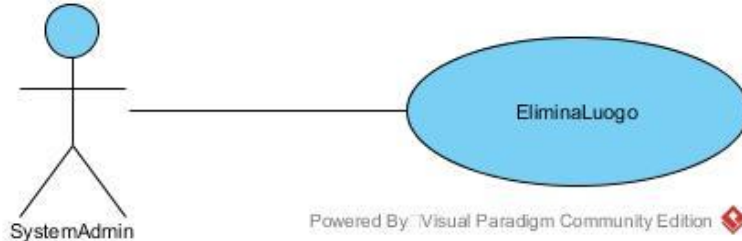
-L'amministratore seleziona un luogo da eliminare.

- Il sistema visualizza la pagina del luogo stesso, che, oltre alle normali opzioni disponibili a tutti gli utenti, visualizza anche delle operazioni specifiche per il SystemAdmin.
- Il SystemAdmin clicca sul tasto dedicato all'eliminazione del luogo

-Il sistema rimuove il luogo scelto dal database

**Exit Condition:** Il luogo è eliminato correttamente

**Casi d'errore:** La connessione è interrotta, il luogo è già stato eliminato da un altro admin



### 2.5.2.13 Eliminazione feedback itinerario

**Attore:** SystemAdmin

**Entry Condition:** Il SystemAdmin accede alla piattaforma.

**Flusso di eventi:**

-Il SystemAdmin effettua una ricerca all'itinerario che contiene il feedback da eliminare, con l'apposito caso d'uso.

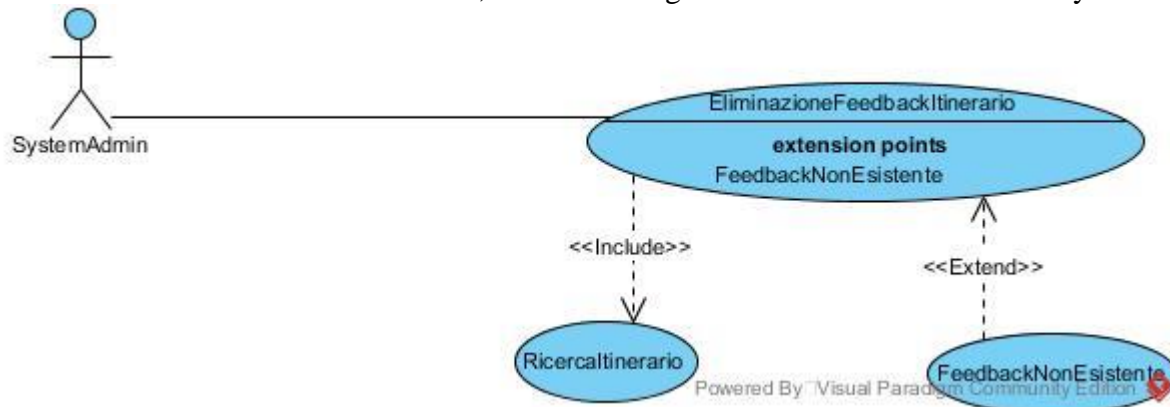
- Il sistema visualizza la pagina dell'itinerario scelto, che, oltre alle normali opzioni disponibili a tutti gli utenti, visualizza anche delle operazioni specifiche per il SystemAdmin.

-Il SystemAdmin seleziona il feedback da eliminare, e clicca sul tasto di eliminazione

- Il sistema cancella dal database il feedback.

**Exit Condition:** L'eliminazione ha avuto successo e viene visualizzato un messaggio di conferma.

**Casi d'errore:** Connessione interrotta, il feedback è già stato eliminato da un altro SystemAdmin.





### 2.5.2.14 Eliminazione feedback luogo

**Attore:** SystemAdmin

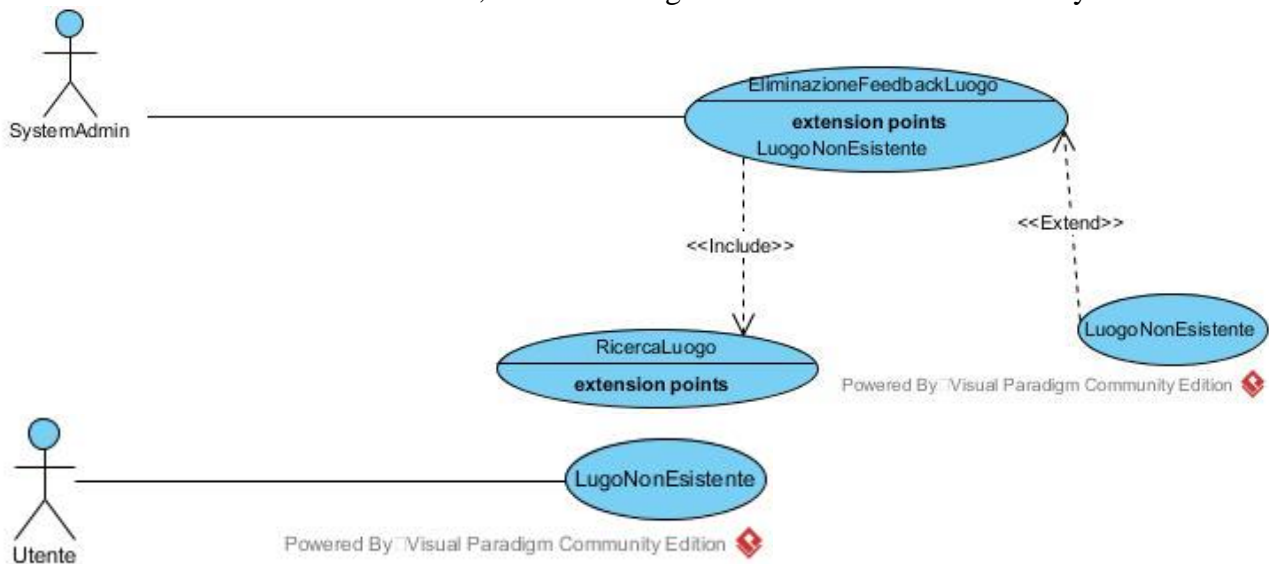
**Entry Condition:** Il SystemAdmin accede alla piattaforma.

**Flusso di eventi:**

- Il SystemAdmin effettua una ricerca al luogo che contiene il feedback da eliminare, con l'apposito caso d'uso.
  - Il sistema visualizza la pagina del luogo scelto, che, oltre alle normali opzioni disponibili a tutti gli utenti, visualizza anche delle operazioni specifiche per il SystemAdmin.
- Il SystemAdmin seleziona il feedback da eliminare, e clicca sul tasto di eliminazione
  - Il sistema cancella dal database il feedback.

**Exit Condition:** L'eliminazione ha avuto successo e viene visualizzato un messaggio di conferma.

**Casi d'errore:** Connessione interrotta, il feedback è già stato eliminato da un altro SystemAdmin.



### 2.5.2.15 Eliminazione feedback utente

**Attore:** SystemAdmin

**Entry Condition:** Il SystemAdmin accede alla piattaforma.

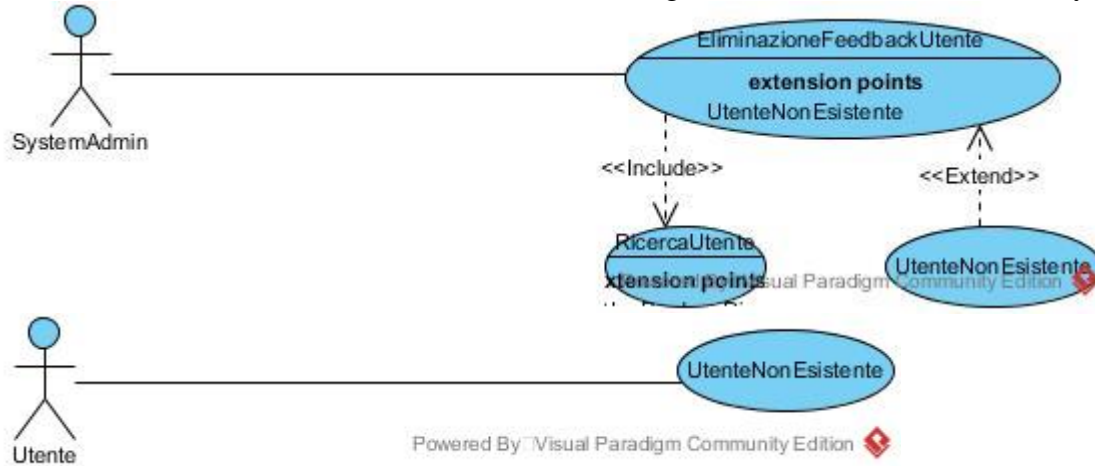
**Flusso di eventi:**

- Il SystemAdmin effettua una ricerca alla pagina dell'utente che contiene il feedback da eliminare, con l'apposito caso d'uso.

- Il sistema visualizza la pagina dell'utente scelto, che, oltre alle normali opzioni disponibili a tutti gli utenti, visualizza anche delle operazioni specifiche per il SystemAdmin.
- Il SystemAdmin seleziona il feedback da eliminare, e clicca sul tasto di eliminazione
  - Il sistema cancella dal database il feedback.

**Exit Condition:** L'eliminazione ha avuto successo e viene visualizzato un messaggio di conferma.

**Casi d'errore:** Connessione interrotta, il feedback è già stato eliminato da un altro SystemAdmin.



## 2.5.2.16 Creazione viaggio - Utente

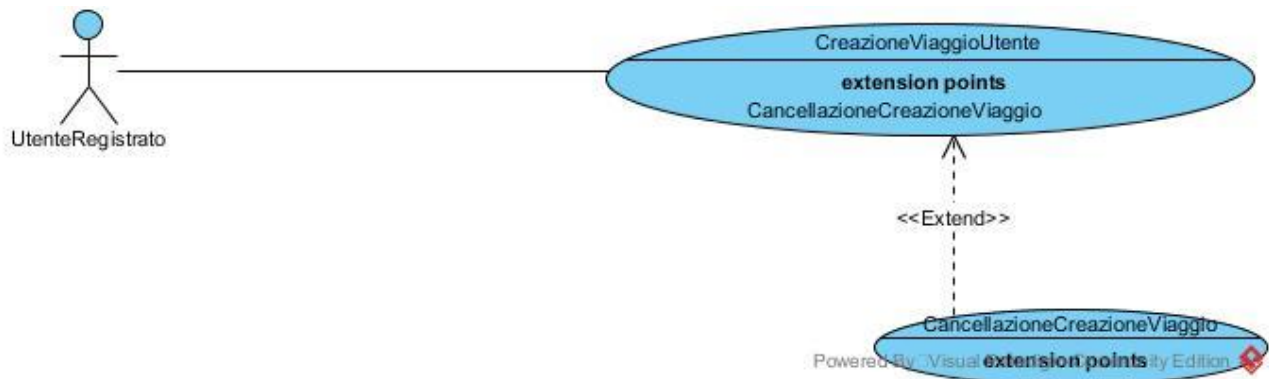
**Attori partecipanti:** Utente Registrato:

**Condizione d'entrata:** L'Utente Registrato ha selezionato l'attività di creazione itinerario

**Flusso di eventi:**

- Il sistema visualizza il form di creazione itinerario, che comprende campi per definire le date in cui si terrà il viaggio, un nome, etc...
- L'utente compila i vari campi del form
- L'utente aggiunge un itinerario al viaggio
- L'utente preme il tasto di conferma creazione
- Il sistema riceve i dati, e crea un nuovo itinerario legato all'utente, con le impostazioni scelte, e mostra il messaggio di conferma della creazione.

**Condizione di uscita:** L'utente registrato visualizza il messaggio di conferma



### 2.5.2.17 Cancellazione viaggio

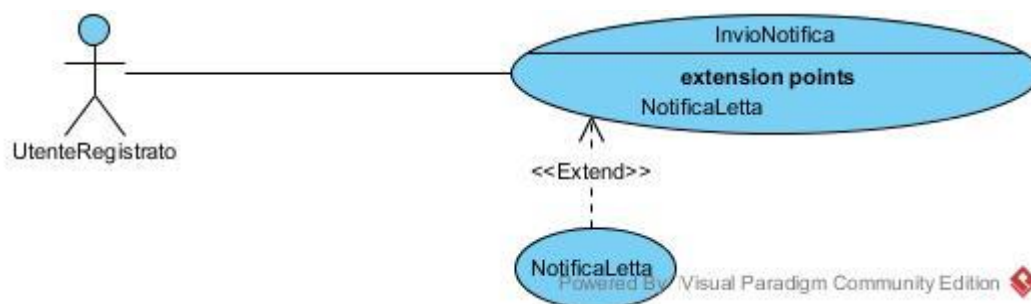
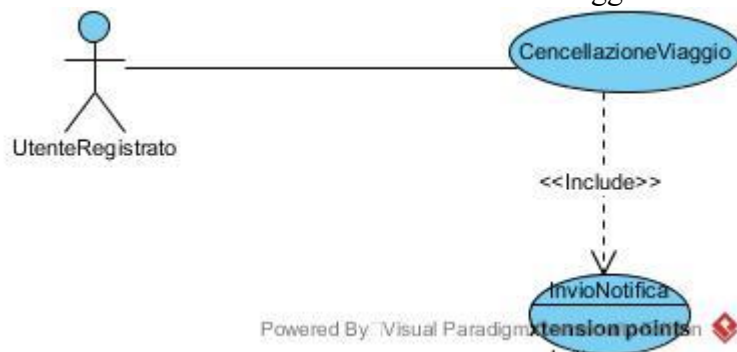
**Attori partecipanti:** Utente Registrato:

**Condizione d'entrata:** L'Utente Registrato ha selezionato un itinerario

**Flusso di eventi:**

- L'utente preme sul tasto di cancellazione viaggio
  - Il sistema rimuove i dati relativi al viaggio dal database
  - Il sistema manda una notifica a tutti i partecipanti avvisandoli della cancellazione del viaggio. (Caso d'uso: Invio notifica)
  - Il sistema, a cancellazione completata, invia un messaggio di conferma all'utente

**Condizione di uscita:** L'utente riceve il messaggio di conferma



### 2.5.2.18 Cancellazione creazione viaggio

**Attore:** Utente Registrato

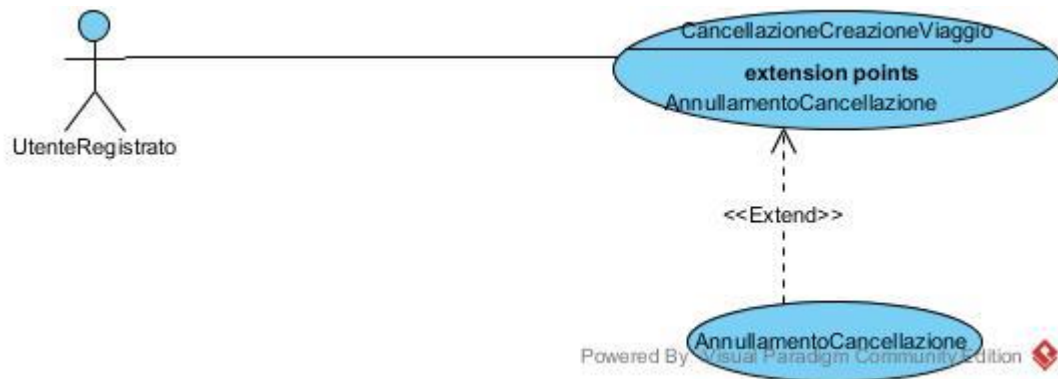
**Entry condition:** L'utente preme il tasto di annullamento della creazione del viaggio

**Flusso di eventi:**

- Il sistema mostra un messaggio all'utente per chiedergli se è sicuro di questa operazione.
- L'utente clicca sul tasto di conferma.
- Il sistema rilascia tutte le risorse legate alla creazione e mostra all'utente un messaggio di cancellazione avvenuta.

**Exit condition:** L'utente visualizza il messaggio

**Caso eccezionale:** Annullamento cancellazione



### 2.5.2.19 Annullamento Cancellazione creazione viaggio

**Attore:** Utente Registrato

**Entry condition:** L'utente non conferma la cancellazione del viaggio

**Flusso di eventi:**

- L'utente clicca sul tasto di rifiuto
- Il sistema riprende la creazione del viaggio dal punto in cui era stato lasciato (riprende il caso d'uso di creazione)

**Exit condition:** L'utente visualizza nuovamente il form di creazione.



### 2.5.2.20 Conferma viaggio

**Attori:** UtenteRegistrato

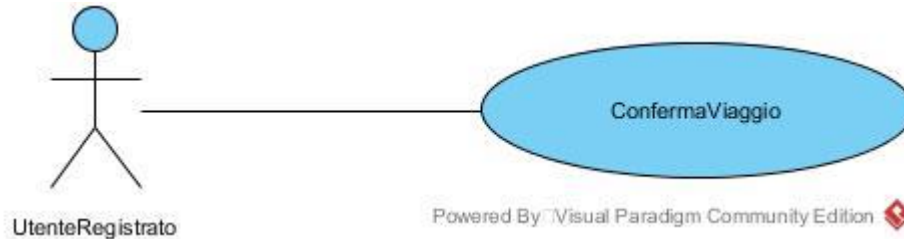
**Entry condition:**

- L'utente creatore del viaggio, preme il pulsante Conferma Viaggio

**Flusso di eventi:**

- Il sistema riceve la richiesta, e imposta lo stato del viaggio come “Confermato”
- Il sistema invia automaticamente una notifica a tutti i partecipanti al viaggio

**Exit condition:** Il viaggio viene confermato e non è più modificabile



### 2.5.2.21 Creazione itinerario

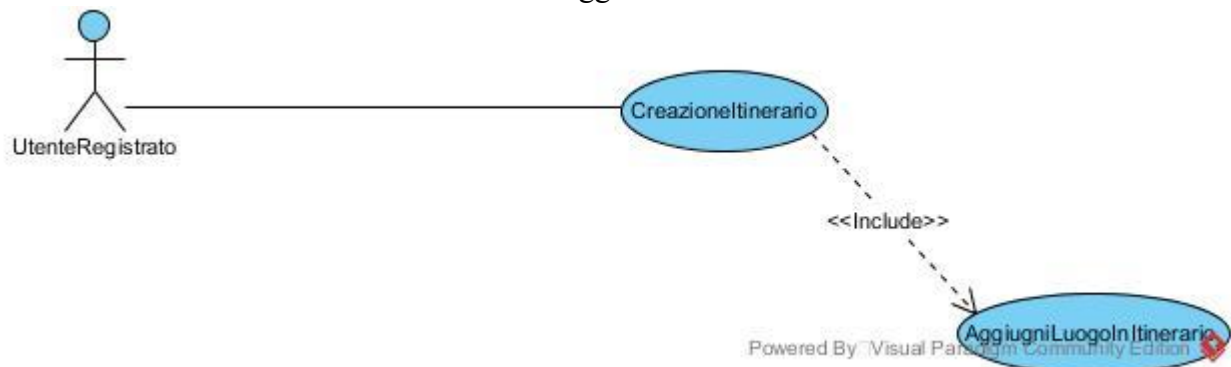
**Attori:** Utente Registrato

**Entry Condition:** L'utente clicca sul tasto di creazione itinerario

**Flusso di eventi:**

- Il sistema visualizza un form di ricerca ed un tasto per aggiungere un luogo
- L'utente inserisce un nome per l'itinerario
- L'utente ricerca dei luoghi con l'apposito caso d'uso e clicca sul tasto aggiungi
- L'utente clicca sul tasto conferma
- Il sistema crea un itinerario contenente i luoghi scelti dall'utente
- Il sistema invia un messaggio di conferma all'utente

**Exit Condition:** L'utente visualizza un messaggio di conferma



### 2.5.2.22 Aggiunta luogo ad un itinerario già creato

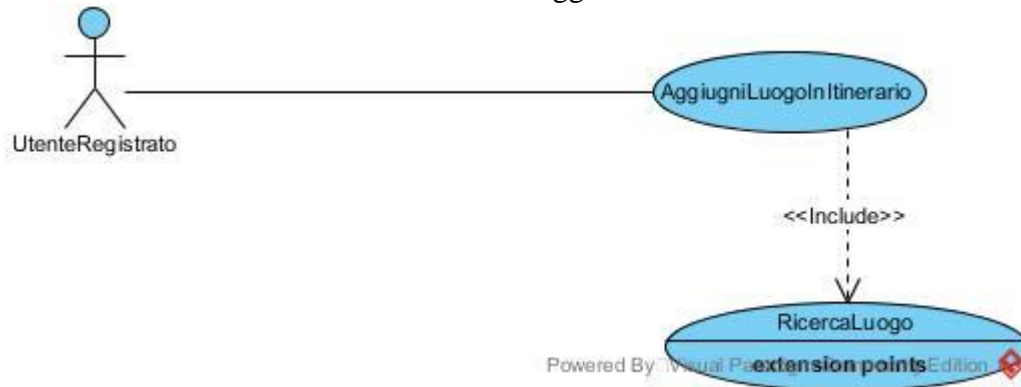
**Attori:** UtenteRegistrato

**Flusso di eventi:**

- L'utente clicca sul tasto di aggiunta di un nuovo luogo all'itinerario

- Il sistema visualizza un form di ricerca e un tasto per aggiungere il luogo
- L'utente ricerca dei luoghi con l'apposito cas d'uso e clicca sul tasto aggiungi
- Il sistema mostra una richiesta di confermato
- L'utente clicca sul tasto conferma
- Il sistema aggiunge all'itinerario il nuovo luogo selezionato
- Il sistema invia un messaggio di conferma all'utente

**Exit condition:** L'utente visualizza il messaggio di confermato



### 2.5.2.23 Rimuovi luogo da un itinerario

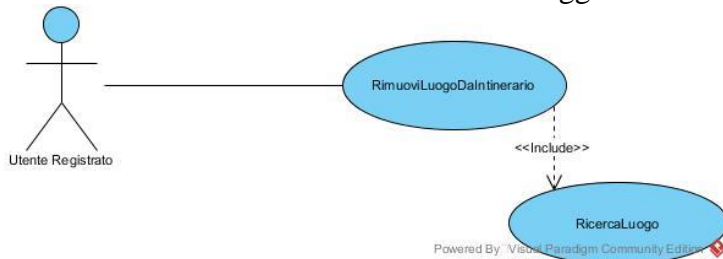
**Attore:** UtenteRegistrato

**Entry Condition:** L'utente visualizza un itinerario creato

**Flusso di eventi:**

- L'utente clicca sul tasto di rimozione di un luogo da un itinerario
- Il sistema rimuove dall'itinerario il luogo selezionato
- Il sistema mostra una notifica

**Exit condition:** L'utente visualizza il messaggio di conferma



### 2.5.2.24 Creazione sondaggio per aggiungere un luogo ad un viaggio

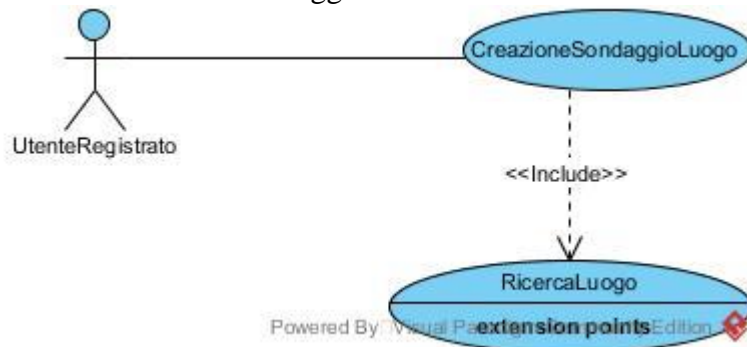
**Attori:** UtenteRegistrato

**Entry condition:** L'utente visualizza l'opzione di creazione sondaggio

**Flusso di eventi:**

- L'utente compila il form del sondaggio, inserendo il nome del luogo che vorrebbe aggiungere all'itinerario
  - Il sistema riceve la richiesta e crea un nuovo sondaggio legato al viaggio

**Exit Condition:** Il sondaggio viene creato



### 2.5.2.25 Votazione sondaggio

**Attori:** UtenteRegistrato

**Entry Condition:** L'utente visualizza un sondaggio in uno dei viaggi a cui partecipa

**Flusso di eventi:**

- L'utente seleziona una delle opzioni disponibili nella form del sondaggio corrente
- L'utente preme conferma, ed invia il suo voto
  - Il sistema riceve la richiesta e registra la scelta dell'utente

**Exit Condition:** Il voto viene visualizzato



### 2.5.2.26 Dare feedback ad un luogo

**Attore:** Utente Registrato

**Condizione d'entrata:** L'Utente Registrato accede alla pagina feedback di un luogo

**Flusso di eventi:**

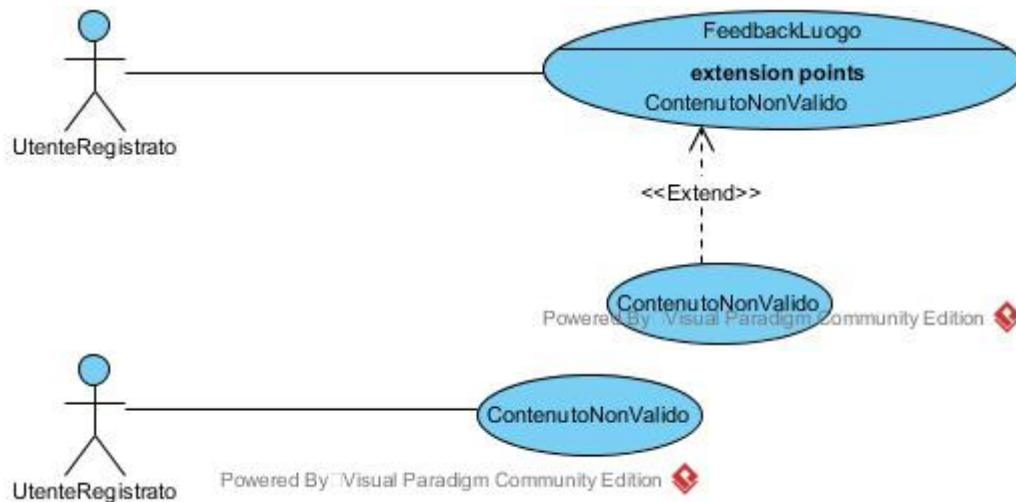
- L'Utente Registrato seleziona la funzione di rilascio feedback, e compila il form relativo inserendo il proprio feedback
  - Il sistema controlla che l'utente abbia visitato almeno una volta il luogo selezionato durante uno dei suoi viaggi
  - Il sistema salva il feedback e lo fa diventare visibile sulla pagina del luogo.

**Condizione di uscita:** L'utente registrato visualizza il feedback pubblicato.

**Requisiti speciali:** Il sistema si occupa di gestire il feedback e di registrarlo sulla pagina del luogo in questione.

**Casi eccezionali:** L'Utente Registrato inserisce un contenuto non valido.





### 2.5.2.27 Dare feedback ad un utente

**Attori partecipanti:** UtenteRegistrato

**Condizione d'entrata:** L'Utente Registrato visualizza il profilo di un utente

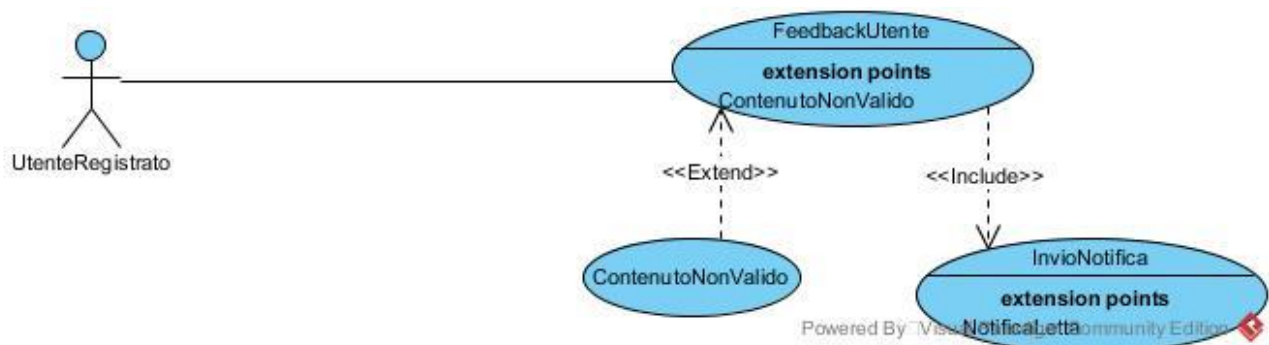
**Flusso di eventi:**

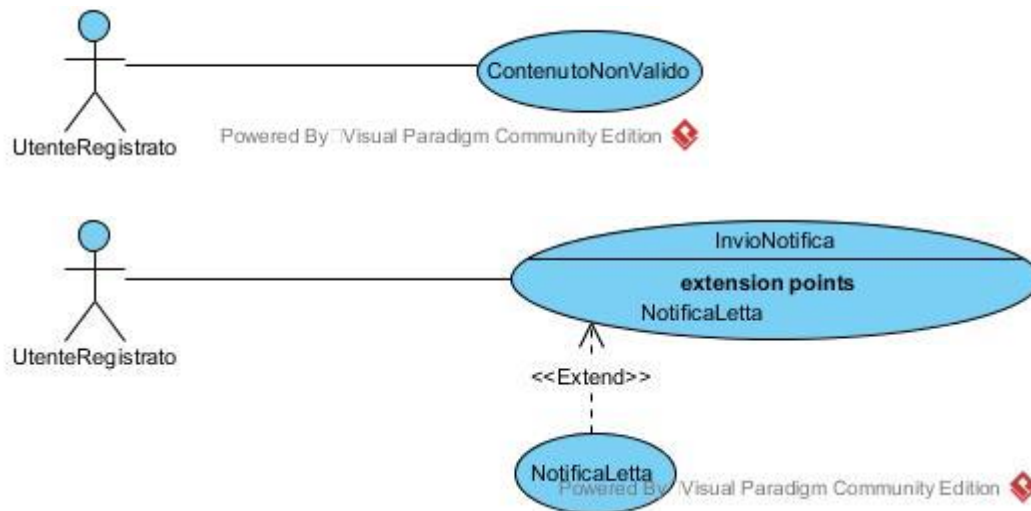
- L'UtenteRegistrato seleziona la funzione di rilascio feedback, e compila il form relativo inserendo il proprio feedback
- Il sistema salva il feedback e lo fa diventare visibile sul profilo dell'utente
- Il sistema notifica l'utente a cui è rivolto il feedback con l'apposito caso d'uso.

**Condizione di uscita:** Il feedback rilasciato viene pubblicato sulla pagina del profilo del secondo utente.

**Requisiti speciali:** Il sistema si occupa di gestire il feedback e di registrarlo sul profilo del secondo utente.

**Casi eccezionali:** L'Utente Registrato inserisce un contenuto non valido.





### 2.5.2.28 Dare feedback ad un itinerario

**Attori partecipanti:** UtenteRegistrato

**Condizione d'entrata:** L'Utente Registrato accede alla pagina un itinerario

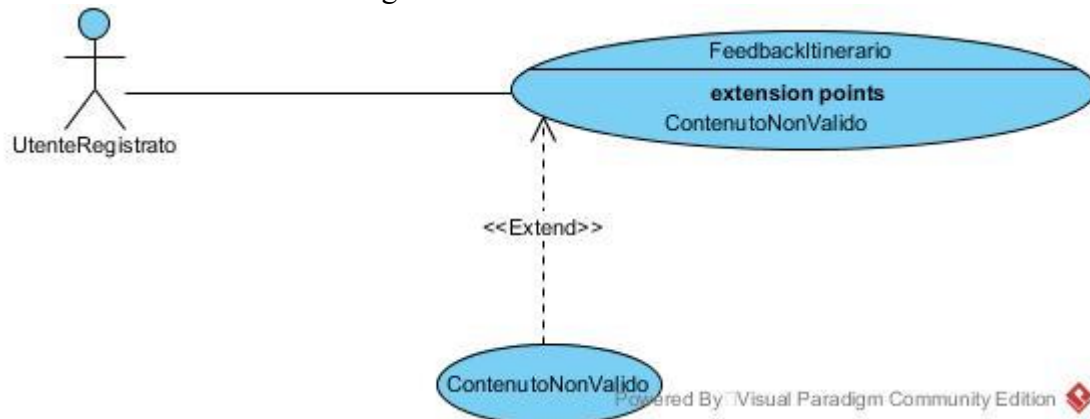
**Flusso di eventi:**

- L'Utente Registrato seleziona la funzione di rilascio feedback, e compila il form relativo inserendo il proprio feedback
  - Il sistema controlla che l'utente abbia seguito almeno una volta l'itinerario scelto
  - Il sistema salva il feedback e lo fa diventare visibile sulla pagina dell'itinerario.

**Condizione di uscita:** Il feedback viene pubblicato sulla pagina dedicata all'itinerario.

**Requisiti speciali:** Il sistema si occupa di gestire il feedback e di registrarlo sulla pagina dell'itinerario. Se l'itinerario è stato creato da un Tour Operator, viene aggiornata la valutazione legata a quest'ultimo.

**Casi eccezionali:** L'Utente Registrato inserisce un contenuto non valido.





### 2.5.2.29 Filtra viaggi per data

**Attori partecipanti:** Utente Registrato

**Condizione d'entrata:** L'Utente seleziona il filtro per periodo

**Flusso di eventi:**

- L'utente inserisce il periodo desiderato
  - Il sistema effettua il sort dei risultati della ricerca precedente in base alla data
  - Il sistema visualizza gli elementi sulla pagina del risultato

**Condizione di uscita:** L'utente visualizza gli elementi ordinati

**Requisiti speciali:** L'utente non visualizza fra gli itinerari proposti quelli privati; L'utente, se non registrato, non potrà compiere ulteriore operazione sugli itinerari



### 2.5.2.30 Filtra viaggi per luogo

**Attori partecipanti:** Utente Registrato

**Condizione d'entrata:** L'Utente seleziona il filtro per luogo

**Flusso di eventi:**

- L'utente inserisce il luogo di interesse
  - Il sistema effettua il sort dei risultati della ricerca precedente in base al luogo
  - Il sistema visualizza gli elementi che includono il luogo sulla pagina del risultato

**Condizione di uscita:** L'utente visualizza gli elementi ordinati

**Requisiti speciali:** L'utente non visualizza fra gli itinerari proposti quelli privati; L'utente, se non registrato, non potrà compiere ulteriore operazione sugli itinerari



### 2.5.2.31 Gestione notifiche

**Attori partecipanti:** UtenteRegistrato

**Condizione d'entrata:** L'Utente Registrato riceve una notifica

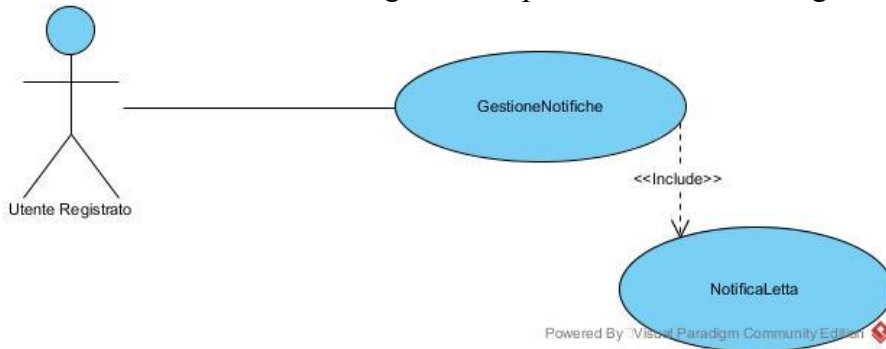
**Flusso di eventi:**

- L'Utente Registrato accede al proprio profilo, e poi alla sezione notifiche
  - Il sistema recupera tutte le notifiche relative all'Utente Registrato e le visualizza nella schermata
- L'Utente Registrato clicca su un'area che rappresenta l'anteprima della notifica
  - Il sistema visualizza la notifica completa, e la imposta come "letta"

**Condizione di uscita:** L'utente esce dalla sezione notifiche

**Requisiti speciali:** Il sistema registra l'avvenuta lettura della notifica e la segna come letta.

**Casi eccezionali:** L'Utente Registrato imposta la notifica come già letta.



### 2.5.2.32 Imposta notifica come già letta

**Attore:** Utente

**Entry condition:** L'utente clicca sulla sezione notifiche del suo profilo.

**Flusso di eventi:**

- L'utente seleziona delle notifiche
- Clicca sul pulsante per segnare le notifiche come già lette
  - Il sistema modifica lo stato delle notifiche ricevute

**Exit condition:** L'utente vede le notifiche come già lette



### 2.5.2.33 Login

**Attori:** UtenteRegistrato.

**Entry condition:** L'utente accede alla pagina iniziale della piattaforma

**Flusso di eventi:**

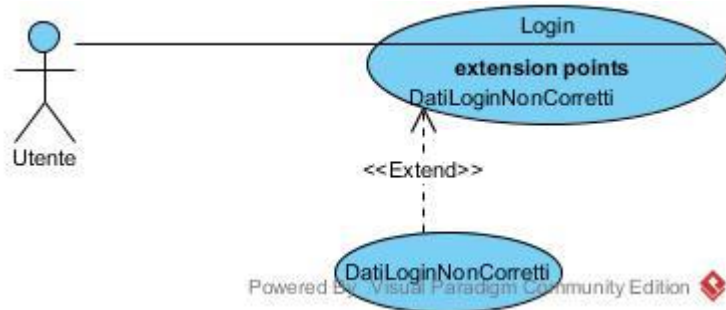
Il sistema visualizza una form con nome utente, password e un pulsante che indica la relativa operazione.

- L'utente compila il form del login e clicca sul tasto per la login.

- Il sistema controlla che i dati sono corretti, se lo sono crea una nuova sessione per l'utente
- Il sistema mostra la homepage della piattaforma

**Exit condition:** L'utente visualizza la homepage.

**Caso eccezionale:** Dati inseriti non corretti.



## 2.5.2.34 Logout

**Attore:** UtenteRegistrato

**Entry condition:** L'utente clicca sul tasto di logout.

**Flusso di eventi:**

- Il sistema controlla la validità della sessione dell'utente che richiede il logout
- Il sistema chiude la sessione relativa a questo utente sulla piattaforma.

**Exit condition:** L'utente visualizza l'homepage.

**Caso eccezionale:** La sessione dell'utente è già scaduta.



## 2.5.2.35 Partecipazione ad un viaggio

**Attori partecipanti:** Utente Registrato

**Condizione d'entrata:** L'Utente visualizza un viaggio

**Flusso di eventi:**

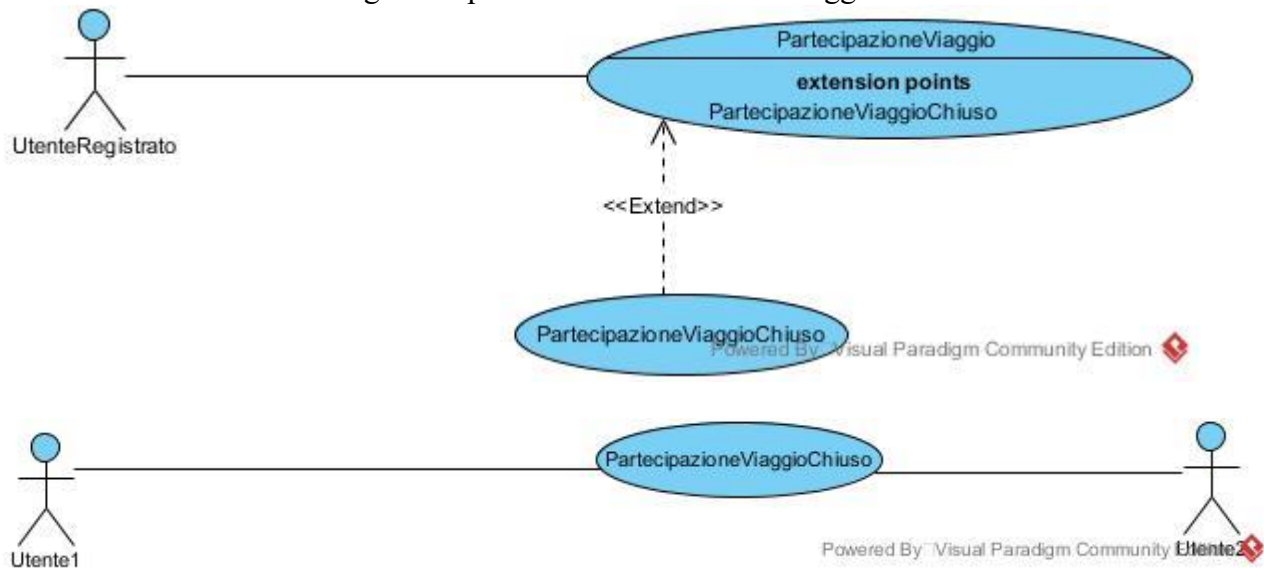
- L'utente clicca sul tasto di partecipazione al viaggio
  - Il sistema effettua la registrazione, aggiungendo i dati del nuovo utente al viaggio

- Il sistema conferma l'iscrizione all'utente e mostra un messaggio di operazione riuscita

**Exit condition:** L'utente visualizza il messaggio

**Requisiti speciali:** La partecipazione viene registrata

**Casi eccezionali:** L'utente registrato prova ad accedere ad un viaggio che è ormai concluso.



### 2.5.2.36 Password dimenticata

**Attori:** Utente

**Entry condition:** L'utente clicca sul link della pagina dedicata alla password dimenticata.

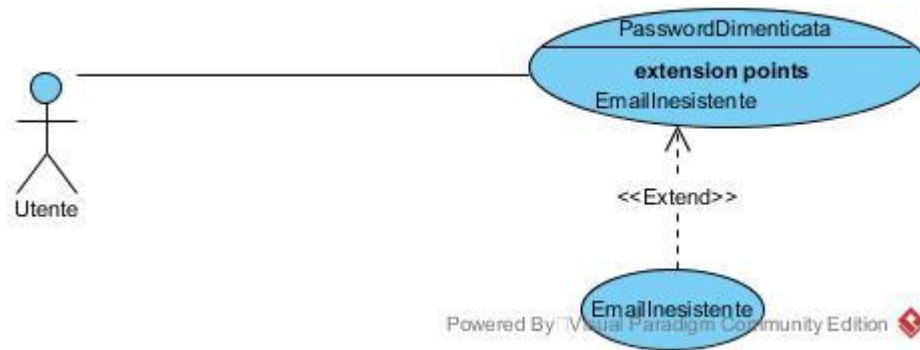
**Flusso di eventi:**

- Il sistema mostra la pagina relativa al recupero di una password con un form per inserire la mail relativa all'account di cui si vuole recuperare la password
- L'utente inserisce la sua mail nel form e preme invio
  - Il sistema controlla che la mail appartenga ad un account registrato sul sistema
  - Il sistema manda via email all'utente un link per poter aggiornare la password
- L'utente accede tramite il link alla schermata di modifica password
  - Il sistema visualizza la schermata con un campo per inserire la password
- L'utente inserisce la nuova password e clicca su accetta
  - Il sistema verifica la nuova password e la associa all'account dell'utente

**Exit condition:** Viene modificata la password dell'utente

**Caso eccezionale:** L'utente inserisce una email che non corrisponde ad un profilo utente registrato.

Allora il sistema gli comunica che l'email inserita non è valida.



### 2.5.2.37 Registrazione

**Attori:** Utente

**Entry condition:** L'utente accede all'area registrazione.

**Flusso di eventi:**

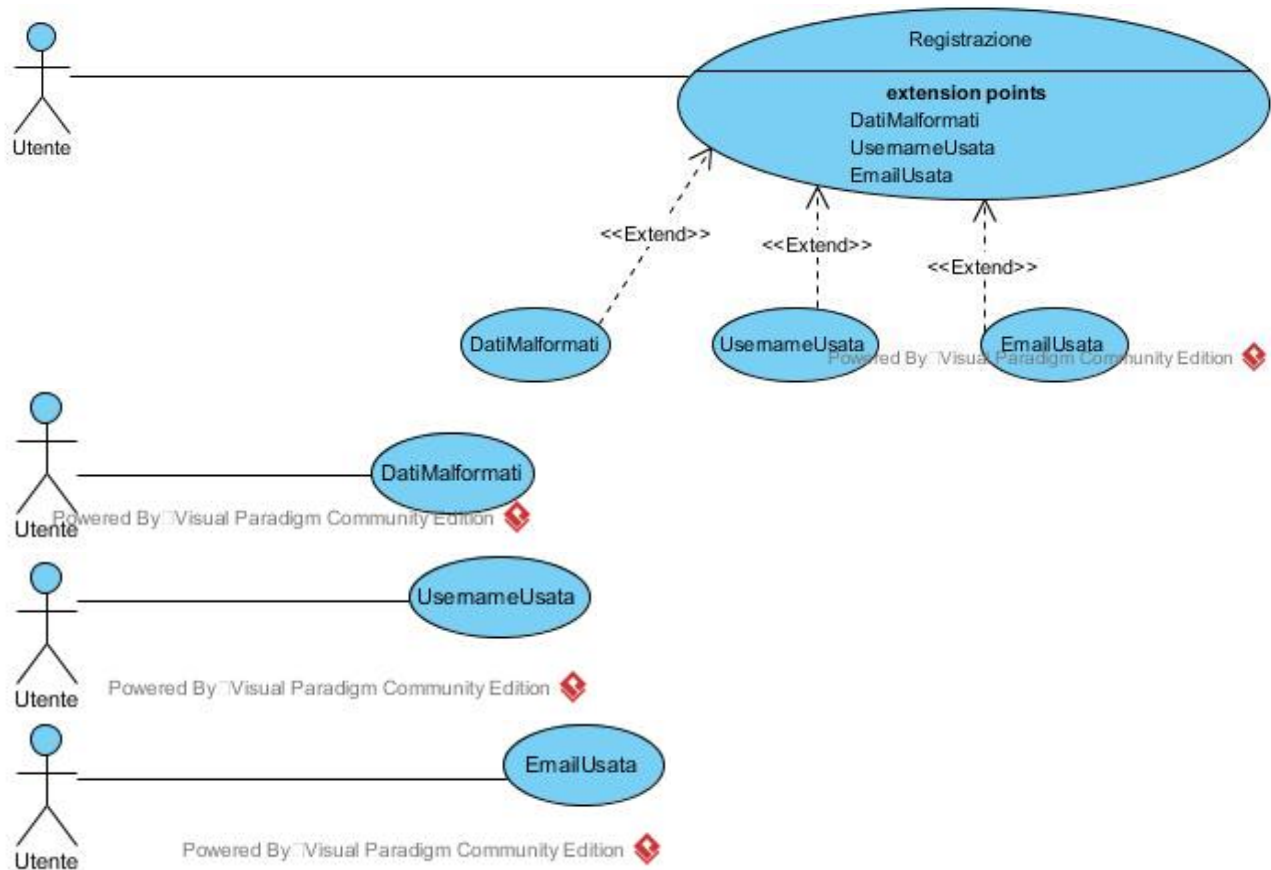
- Il sistema visualizza un form di registrazione, comprensivo, tra gli altri, di campi per la mail di registrazione, nome, cognome, password, etc...
- L'utente inserisce vari dati, tra cui quelli anagrafici e quelli identificativi sulla piattaforma (email, username, password)
- L'utente clicca sul pulsante di conferma registrazione
  - Il sistema riceve la richiesta di registrazione
  - Il sistema crea un nuovo utente utilizzando i dati raccolti dal form
  - Il sistema reindirizza l'utente appena registrato alla propria pagina personale

**Exit condition:** L'utente visualizza la sua pagina personale

**Caso eccezionale:** Dati malformati. Username già in uso. Email già in uso.

**Special requirements:** L'operazione deve assicurare all'utente sicurezza nel momento del salvataggio di dati sensibili sul database.





### 2.5.2.38 Rinnovo Pacchetto Tour Operator

**Attori:** TourOperator, SistemaPagamento

**Entry Condition:** Un TourOperator accede alla funzione di rinnovo del proprio pacchetto

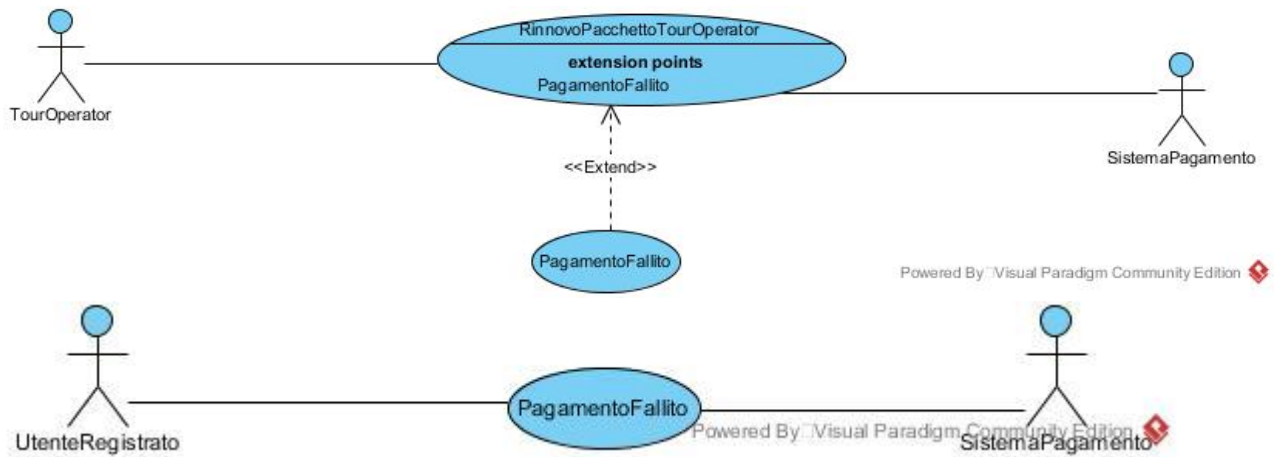
**Flusso di eventi:**

- Il sistema visualizza i dati correnti relativi all'opzione scelta dal TourOperator
- L'utente inserisce i dati per il pagamento e da la conferma
  - Il sistema effettua le operazioni relative al pagamento tramite il SistemaPagamento
  - Il sistema riceve conferma da SistemaPagamento. A pagamento effettuato, il sistema aggiorna i dati dell'utente, fornendo i privilegi da Tour Operator

**Exit Condition:** L'utente diventa Tour Operator

**Caso Eccezionale:** Il pagamento non viene concluso.





### 2.5.2.39 Aggiungere un luogo al database

**Attore:** SystemAdmin

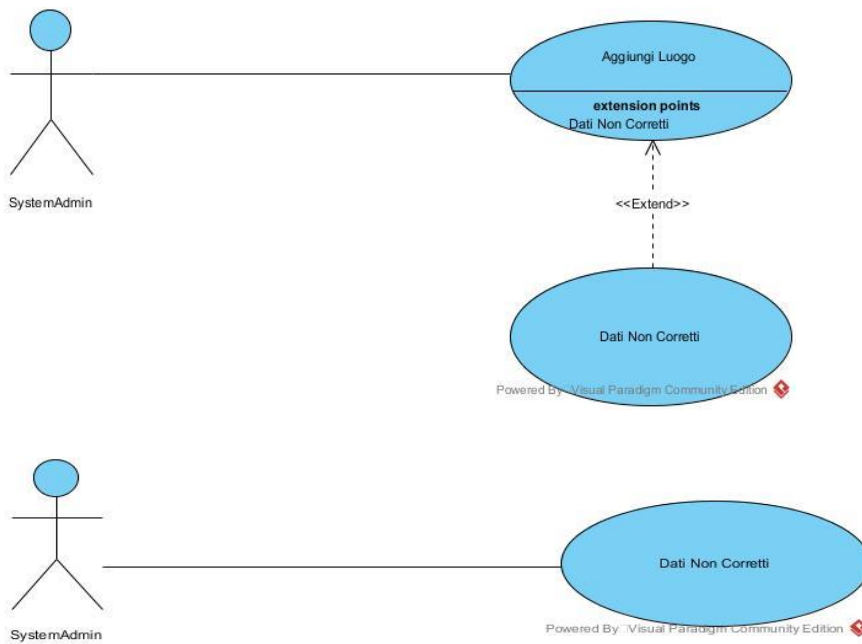
**Entry Condition:** Il SystemAdmin accede alla piattaforma

**Flusso di eventi:**

- Il SystemAdmin seleziona il nuovo luogo da aggiungere.
- Il SystemAdmin aggiunge un nuovo luogo al database.

**Exit Condition:** Il nuovo luogo viene aggiunto al database

**Caso Eccezionale:** Il luogo da registrare ha dati non validi.



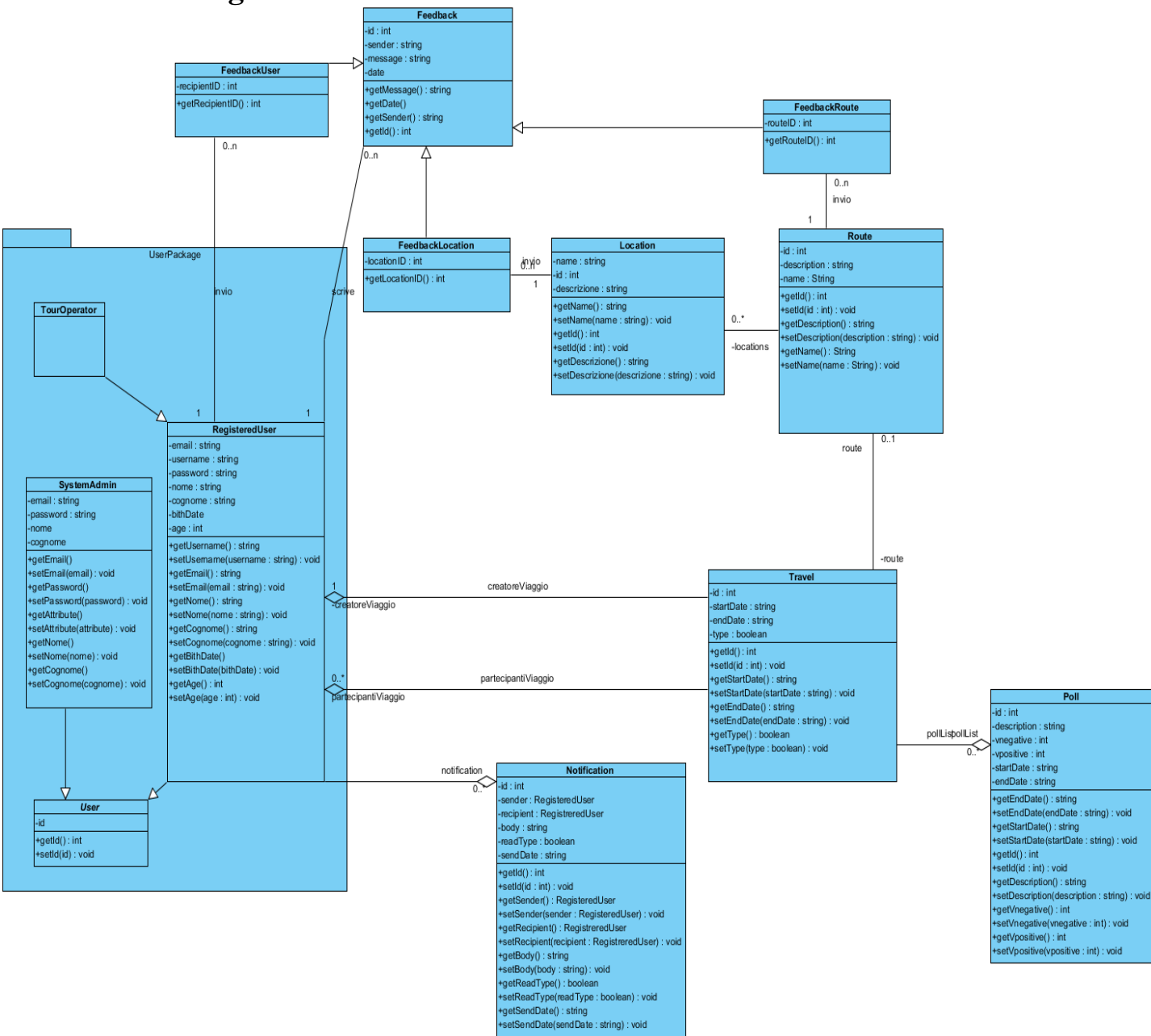
## 2.5.3 Object Model

### 2.5.3.1 Data dictionary

Classi	Parametri
RegisteredUser	id (Long) P.K. username (String) UNIQUE name (String) lastName (String) email (String) UNIQUE password (String) birthDate (String) age (Short)
SystemAdmin	id (Long) P.K. email (String) UNIQUE password (String)
TourOperator	id (Long) P.K. package (String) earnings (Double) purchaseDate (String) expired (Boolean)
Travel	id (Long) P.K. startDate (String) endDate (String) routeID (Long) F.K. TO Route destination (String) creatorID (Long) F.K. TO RegisteredUser Type (Boolean)
UserTravelMatch	(participantID (Long) F.K. TO RegisteredUser, travelID (Long) F.K. TO Travel) P.K. entryDate (String)
Route	id (Long) P.K. description (String) name (String)
Location	id (Long) P.K. description (String) name (String)

RouteLocationMatch	(locationID (Long) F.K. TO Location, routeID (Long) F.K. TO Route) P.K. entryDate (String)
Poll	id (Long) P.K. location (String) description (String) positive (Integer) negative (Integer) startDate (String) endDate (String) idTravel (Long) F.K. TO Travel
FeedbackUser	id (Long) P.K. senderID (Long) F.K. TO RegisteredUser recipientID (Long) F.K. TO RegisteredUser message (String) sendDate (String)
FeedbackRoute	id (Long) P.K. senderID (Long) F.K. TO RegisteredUser routeID (Long) F.K. TO Route message (String) sendDate (String)
FeedbackLocation	id (Long) P.K. senderID (Long) F.K. TO RegisteredUser locationID (Long) F.K. TO Location message (String) sendDate (String)
Notification	(id (Long), senderID (Long) F.K. TO RegisteredUser recipient (Long) F.K. TO RegisteredUser) P.K. body (String) read (Boolean) sendDate (String)

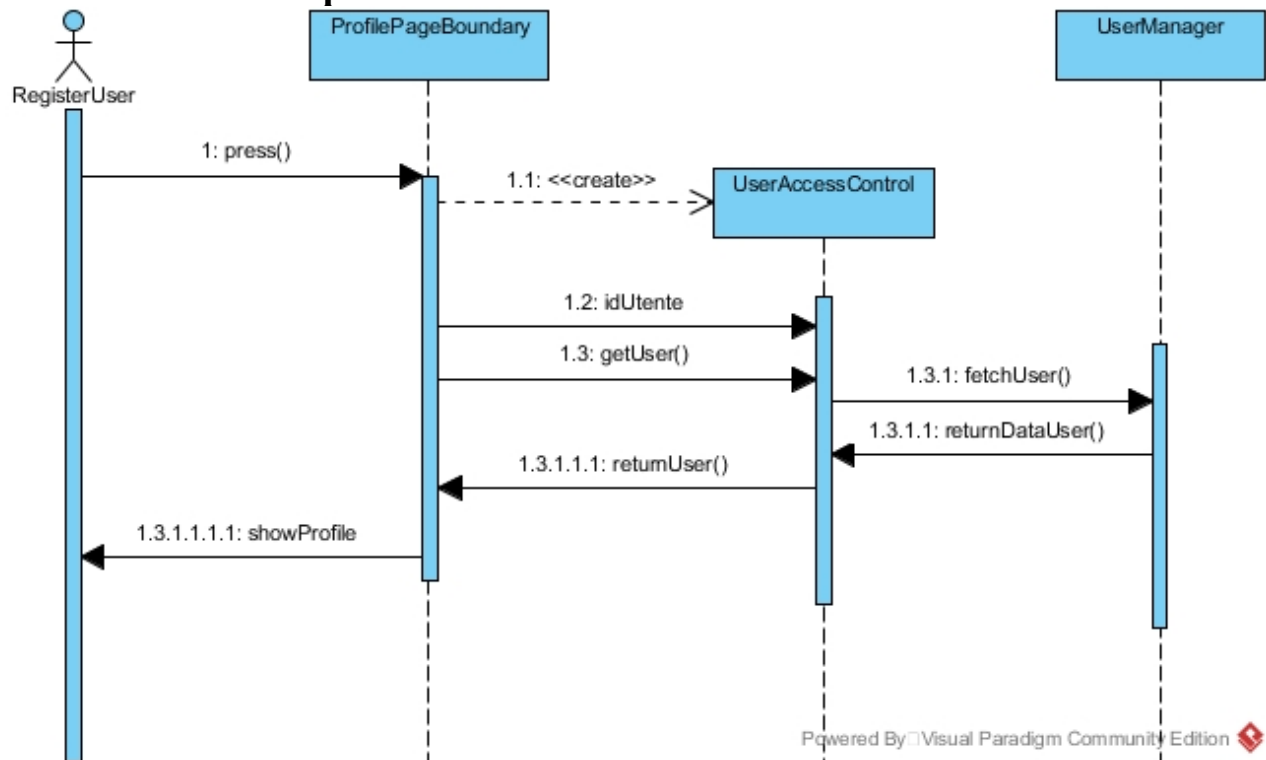
### 2.5.3.2 Class Diagram



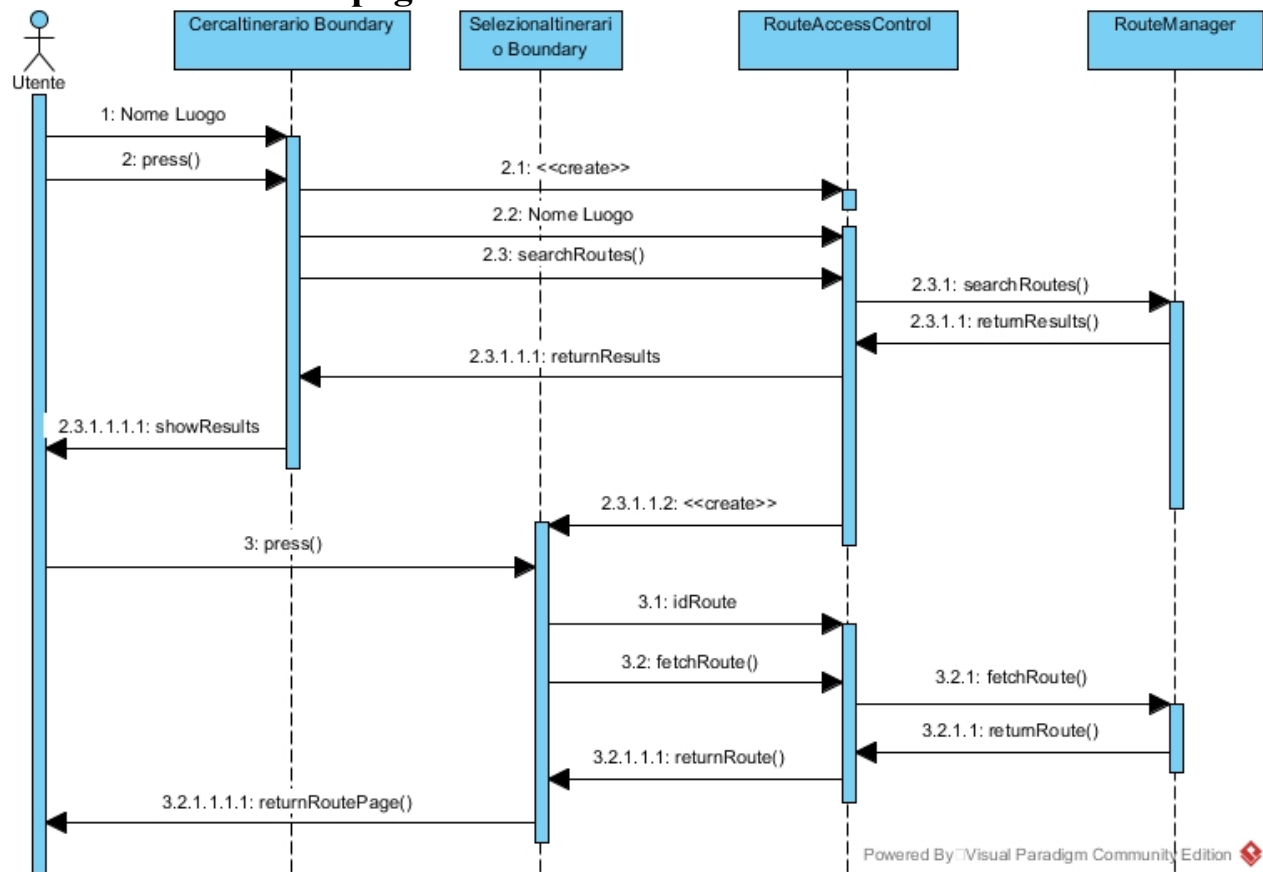
## 2.5.4 Dynamic Models

### 2.5.4.1 Sequence diagrams

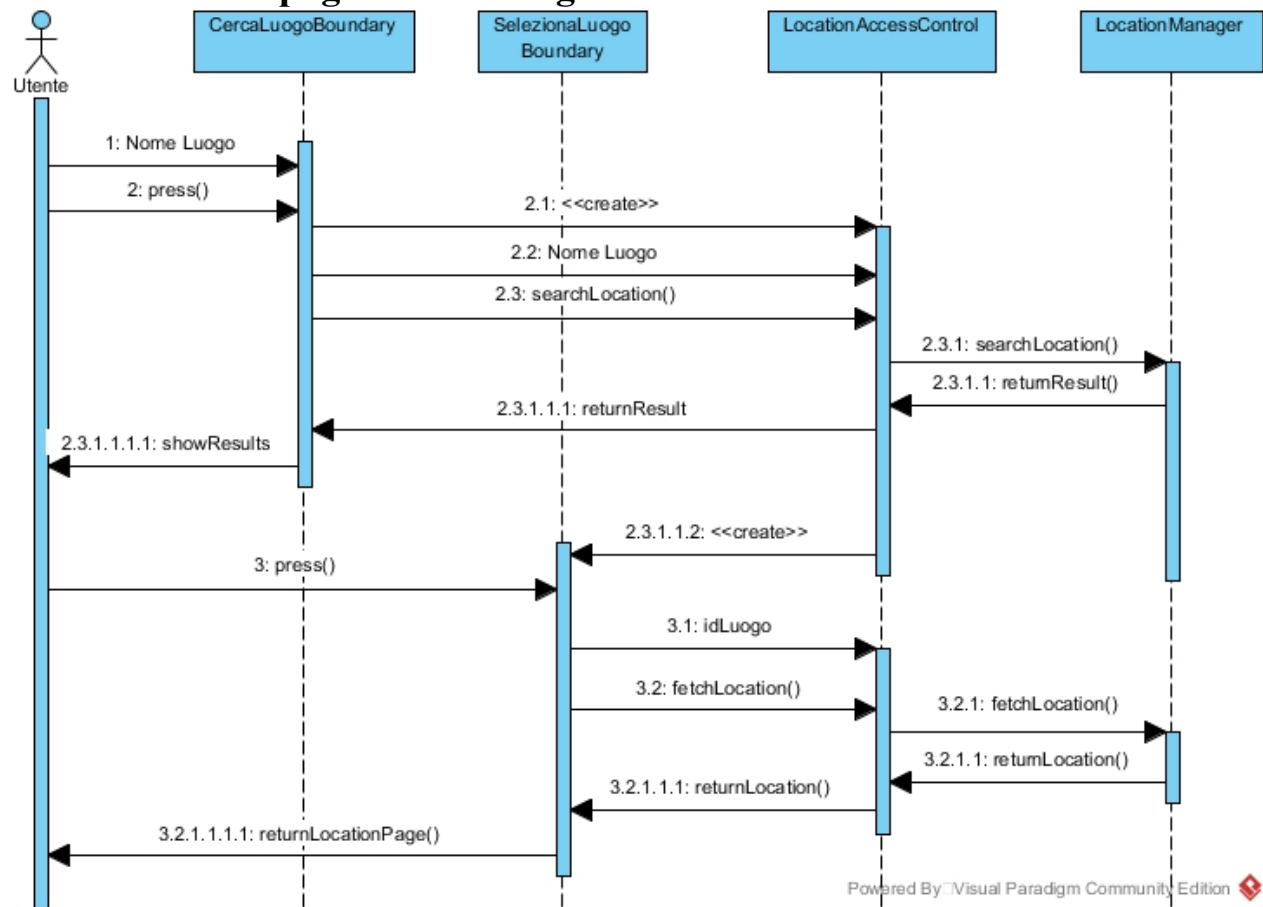
#### 2.5.4.1.1 Accesso al profilo utente



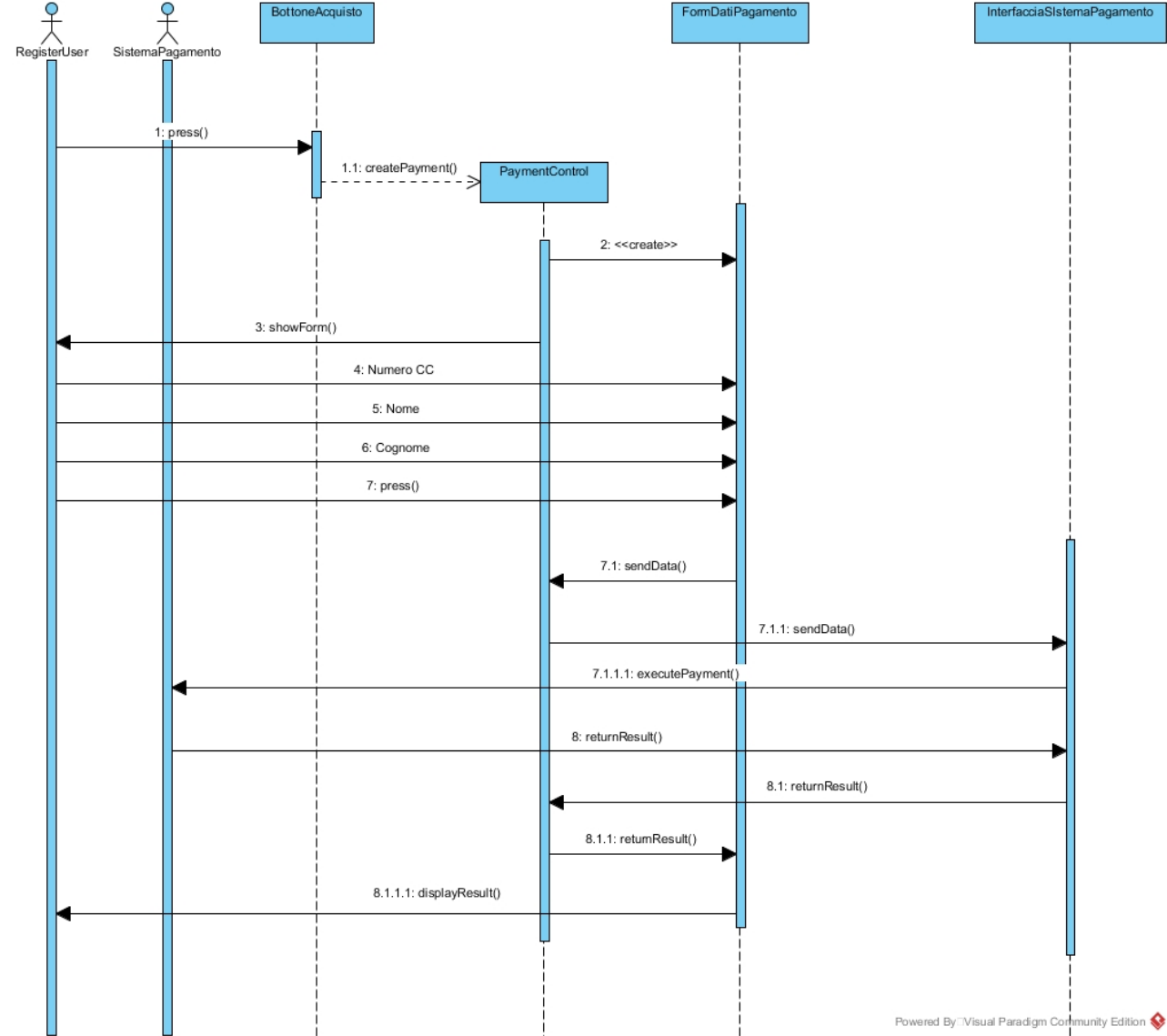
2.5.4.1.2 Accesso alla pagina di un itinerario



### 2.5.4.1.3 Accesso pagina di un luogo

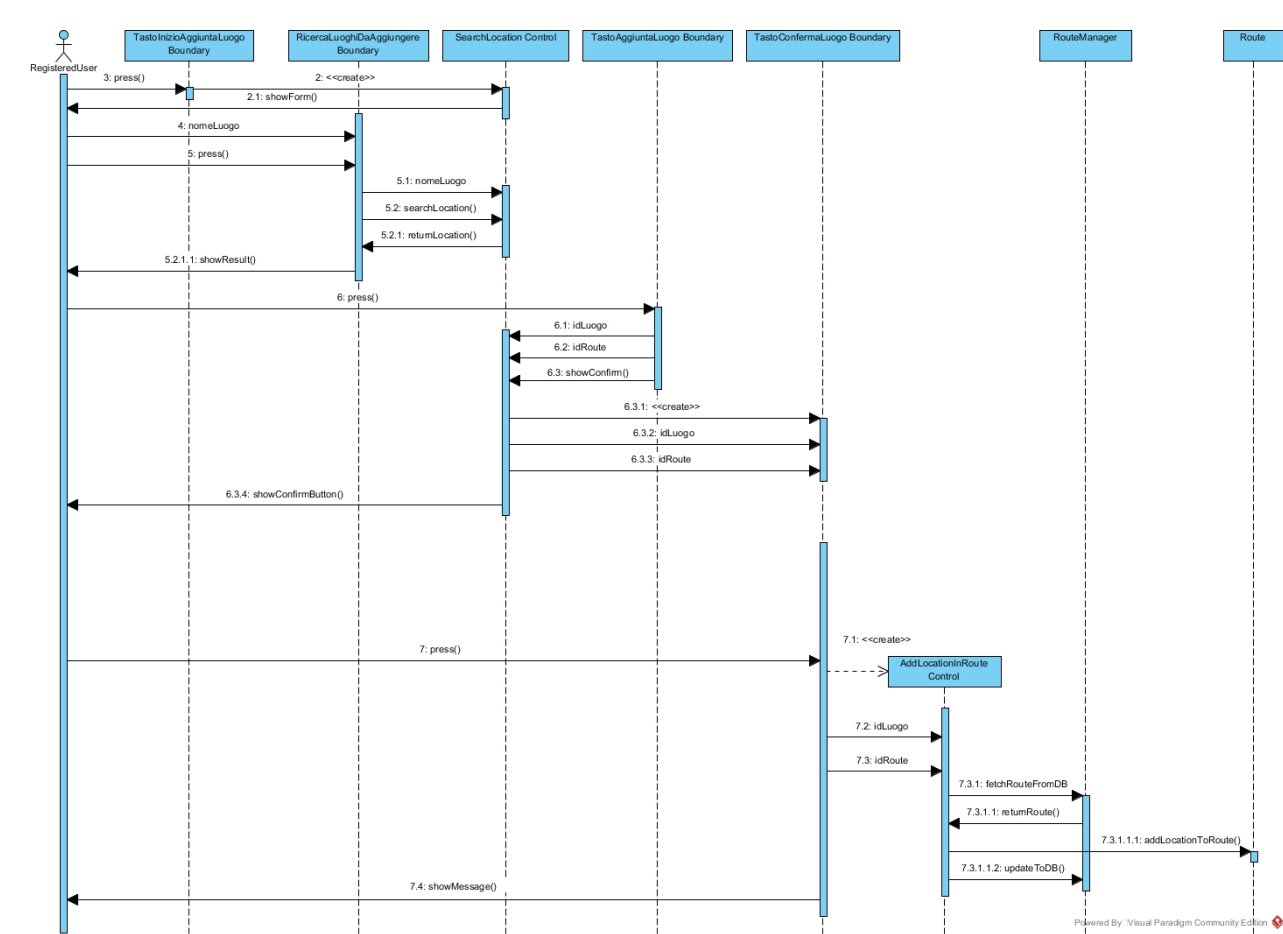


# 2.5.4.1.4 Acquisto viaggio da TourOperator

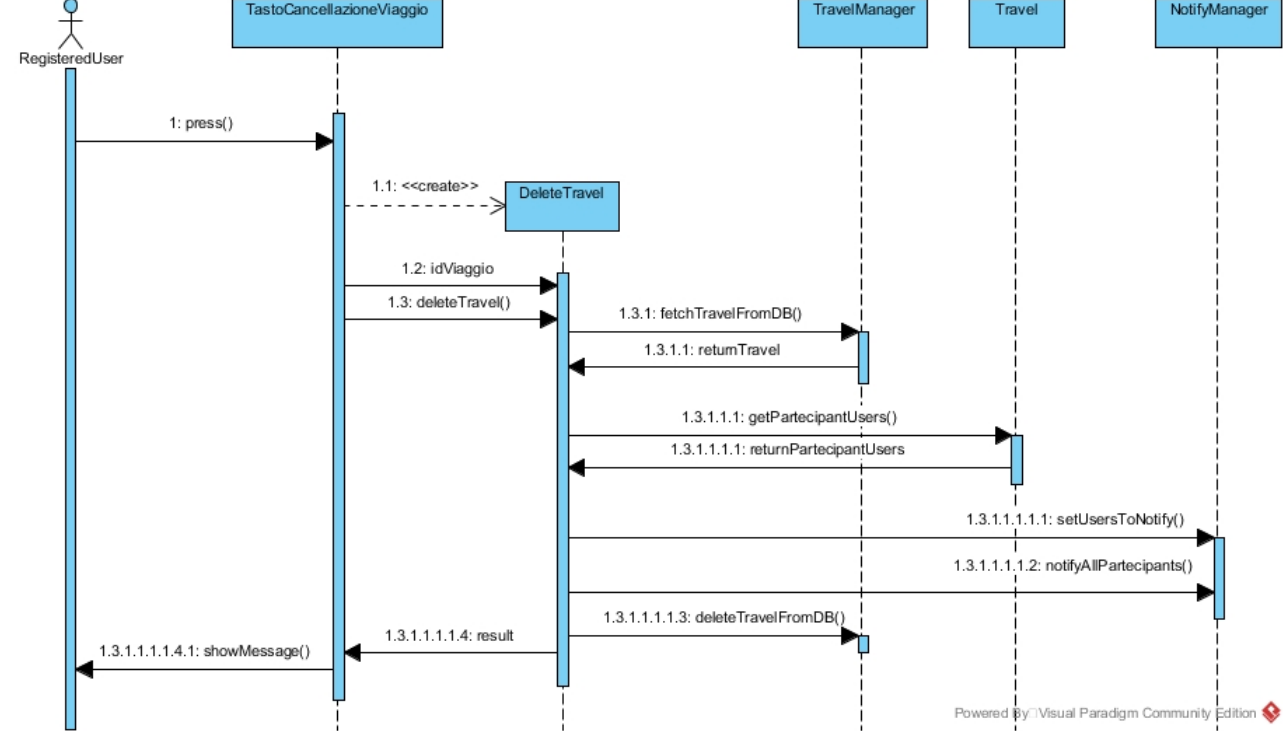




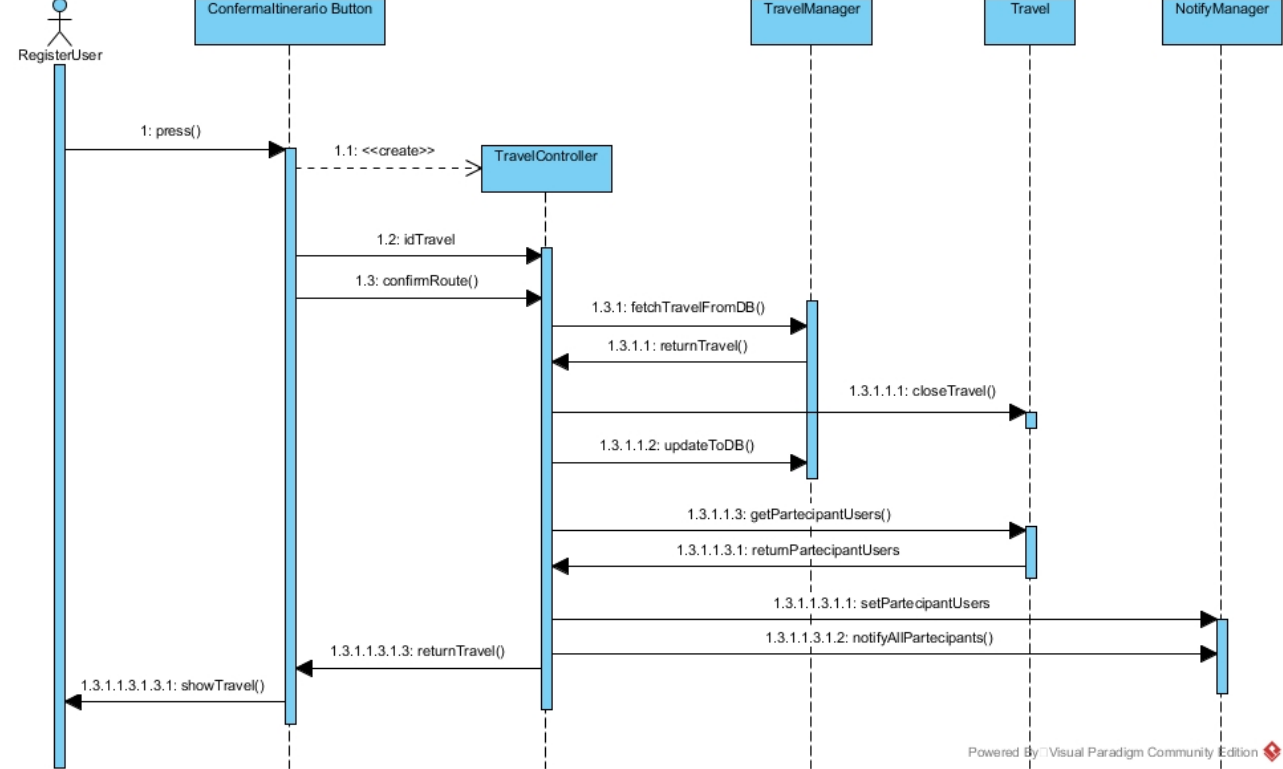
# 2.5.4.1.5 Aggiunta Luogo In Un Itinerario Già Creato



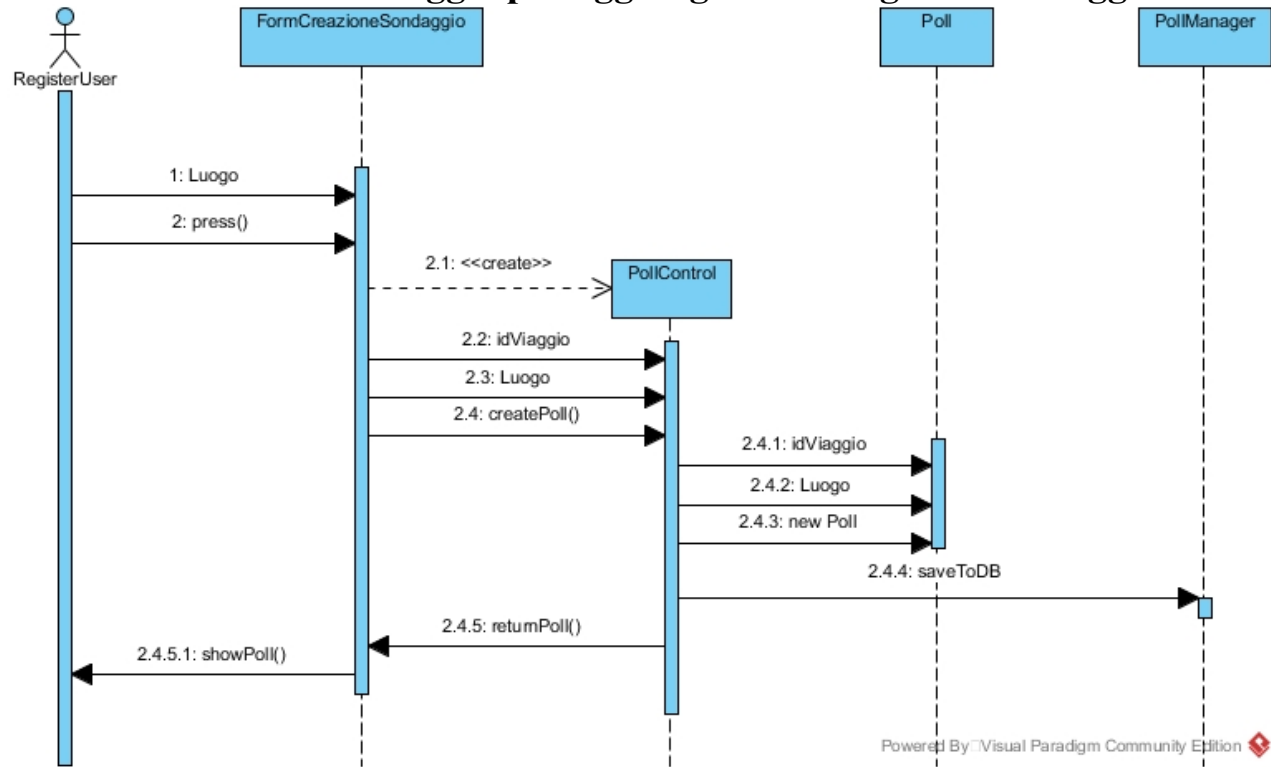
# 2.5.4.1.6 Cancellazione Viaggio



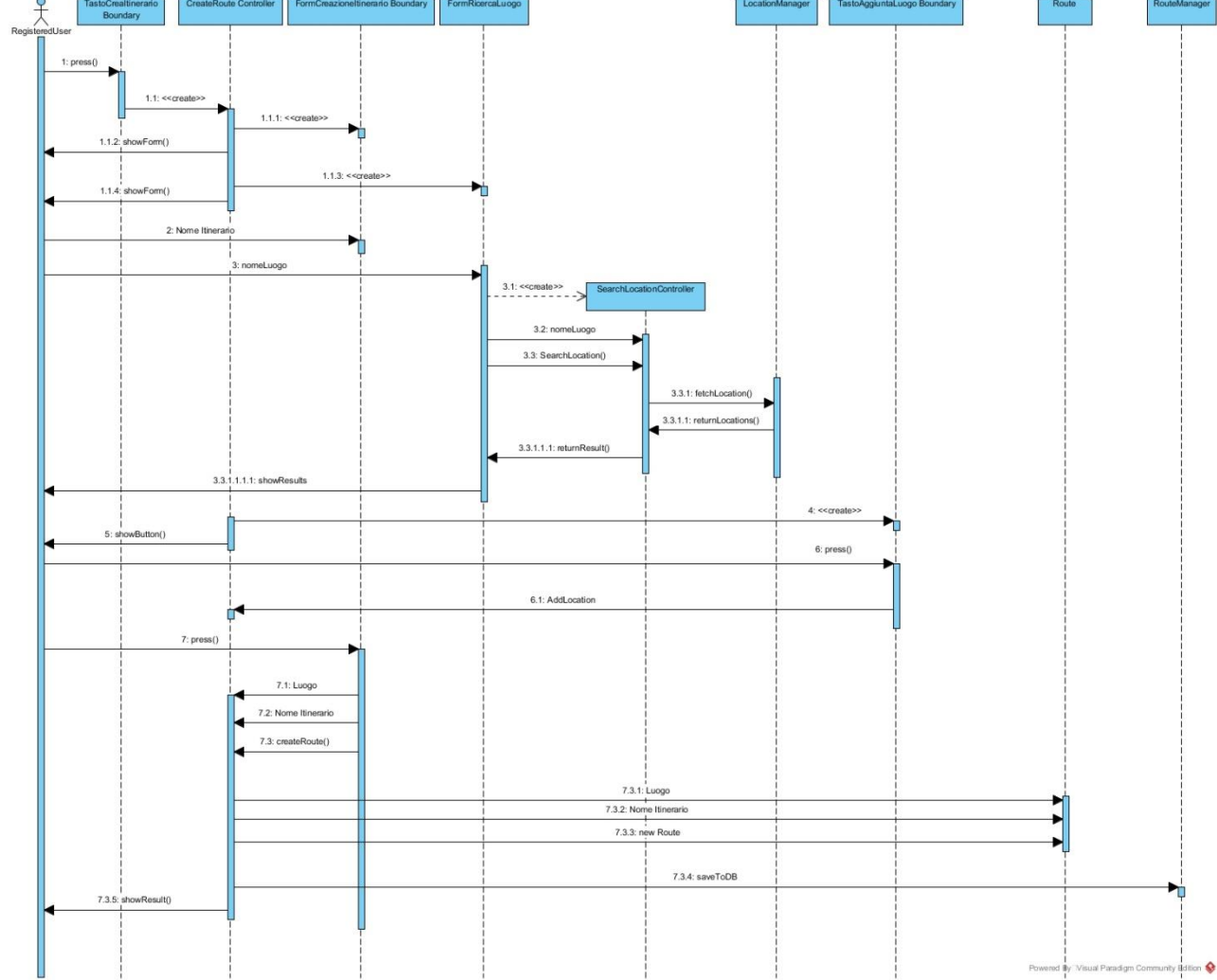
# 2.5.4.1.7 Conferma Viaggio



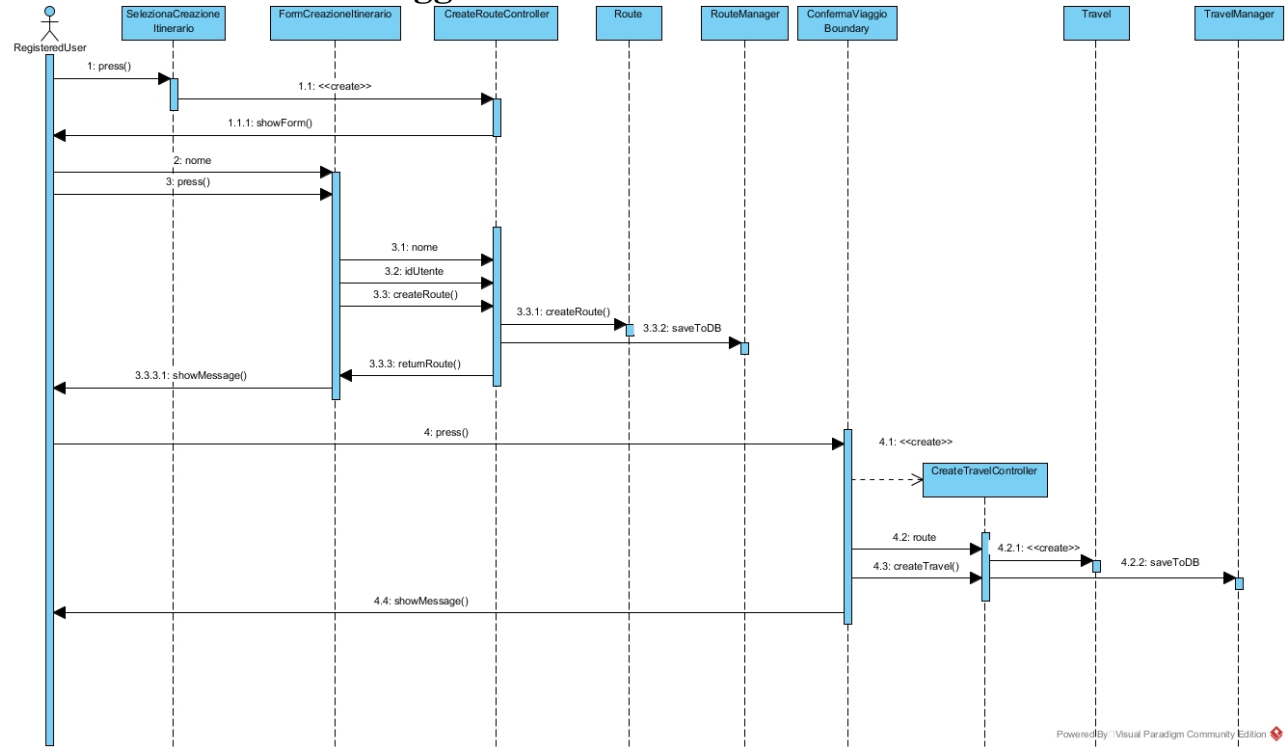
### 2.5.4.1.8 Creazione sondaggio per aggiungere un luogo ad un viaggio



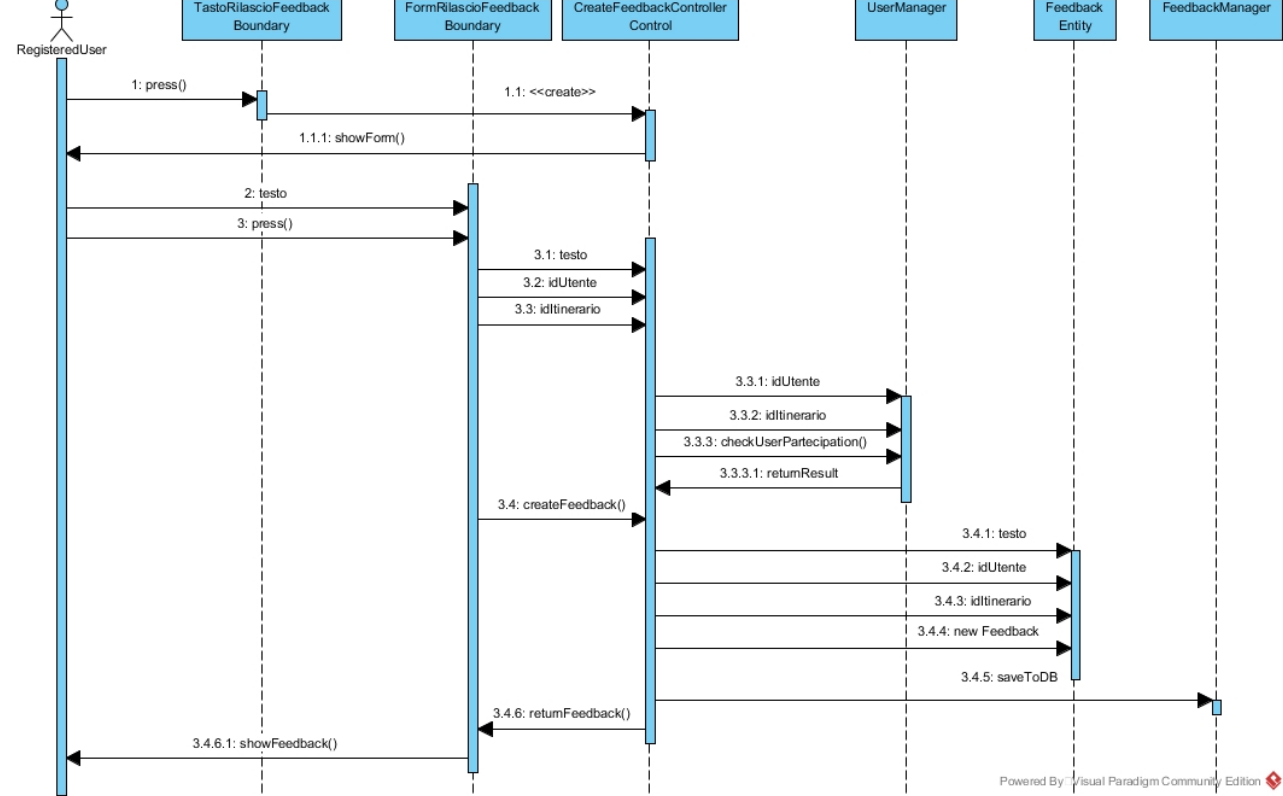
# 2.5.4.1.9 Creazione Itinerario



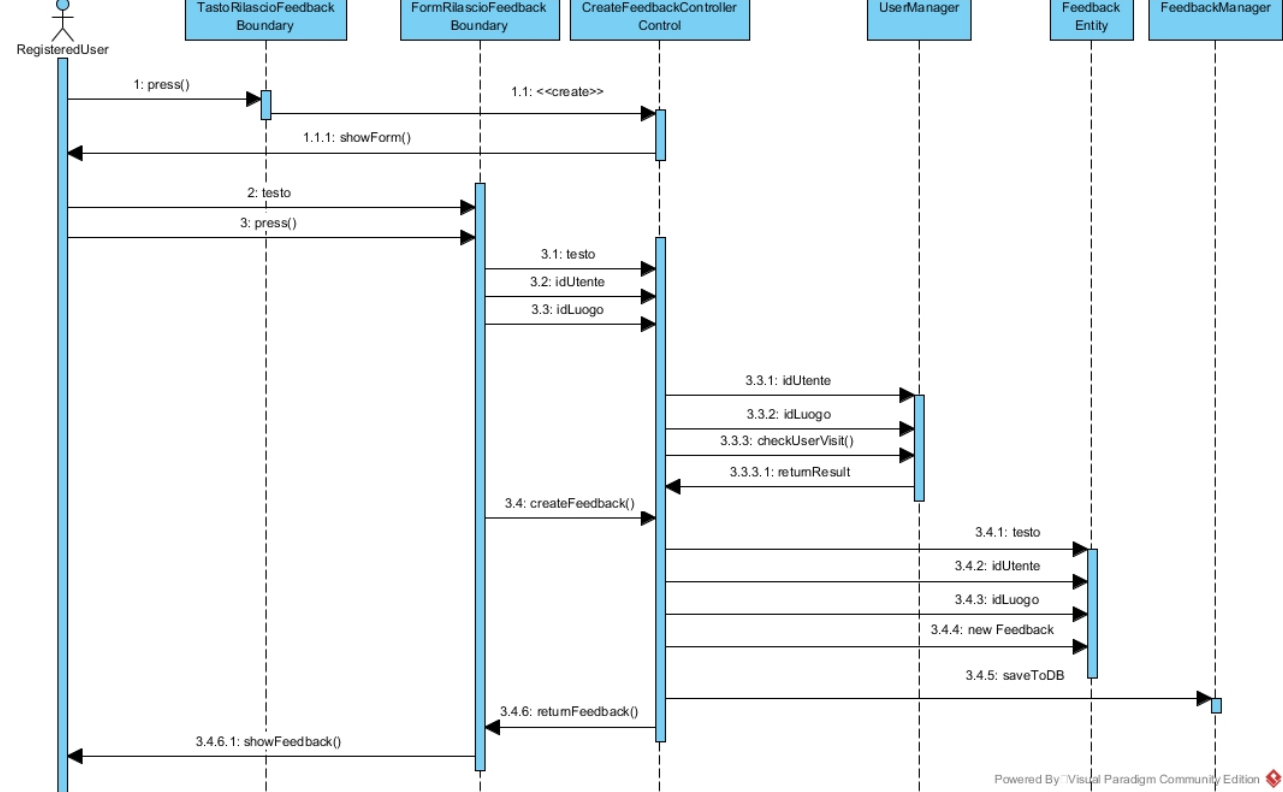
# 2.5.4.1.10 Creazione Viaggio



# 2.5.4.1.11 Dare Feedback Ad Un Itinerario

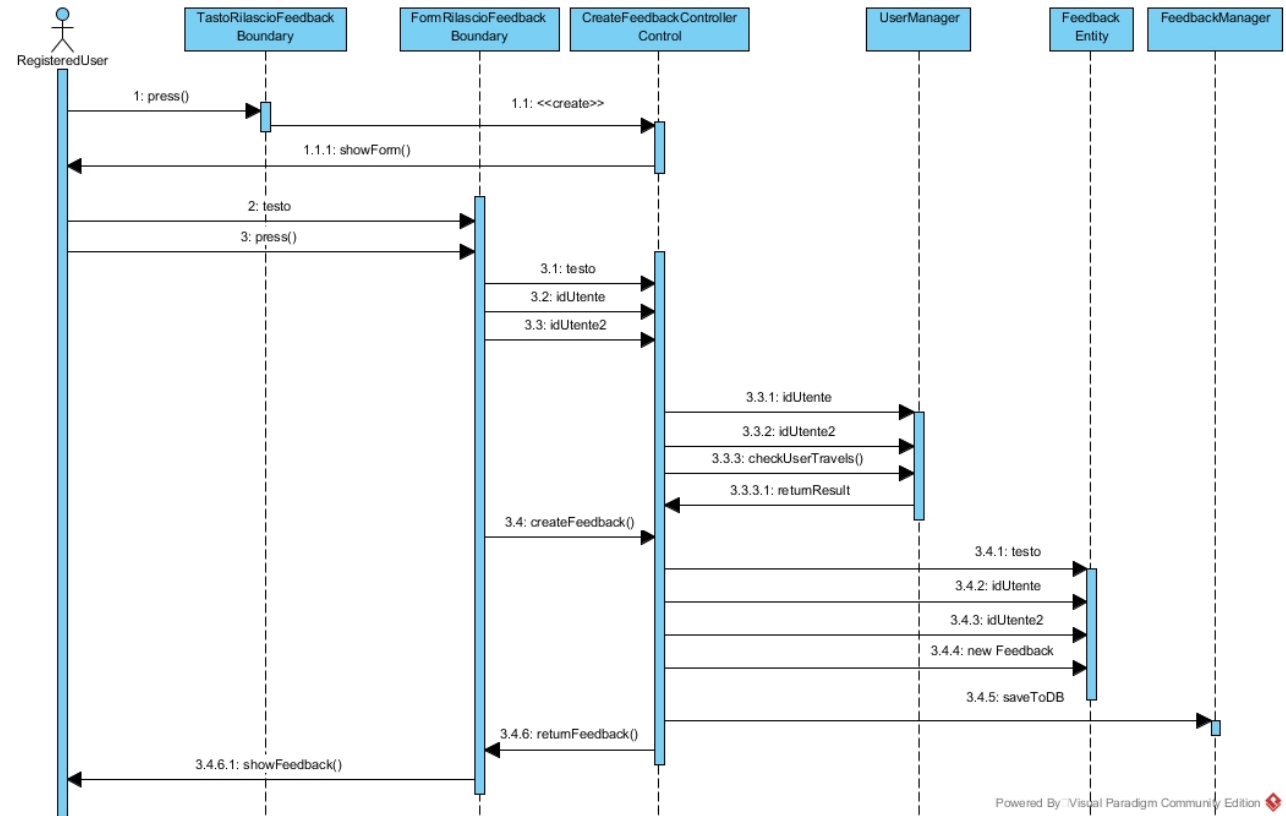


# 2.5.4.1.12 Dare Feedback Ad Un Luogo

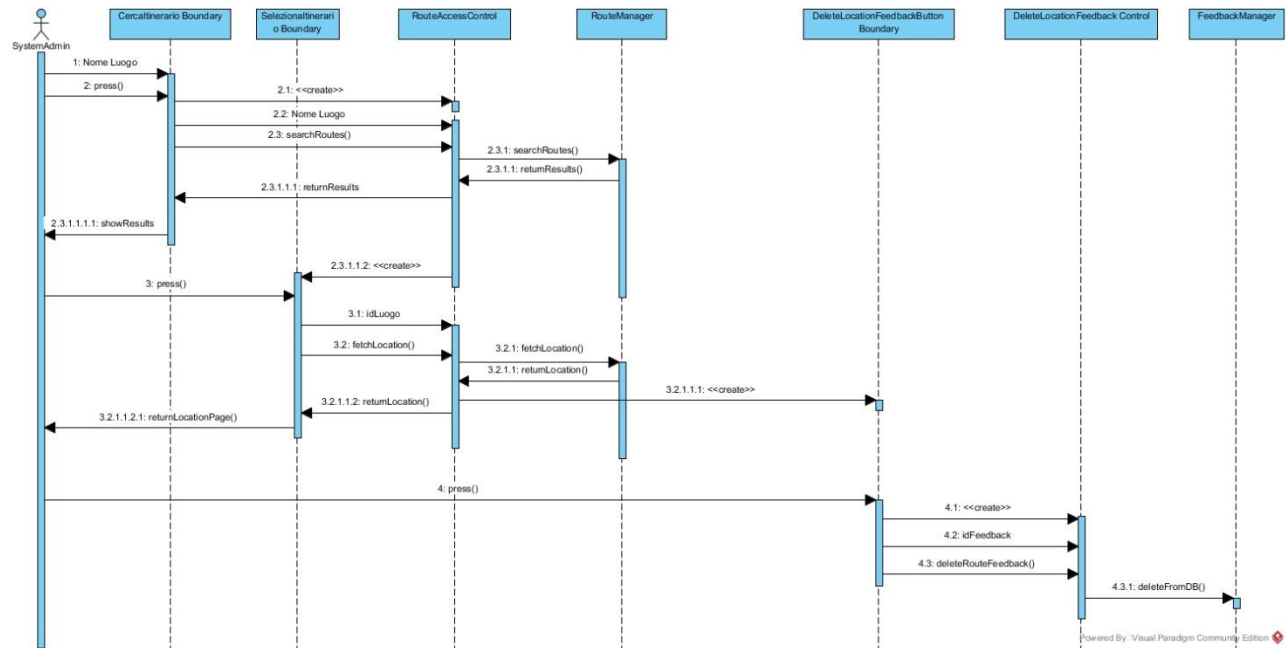




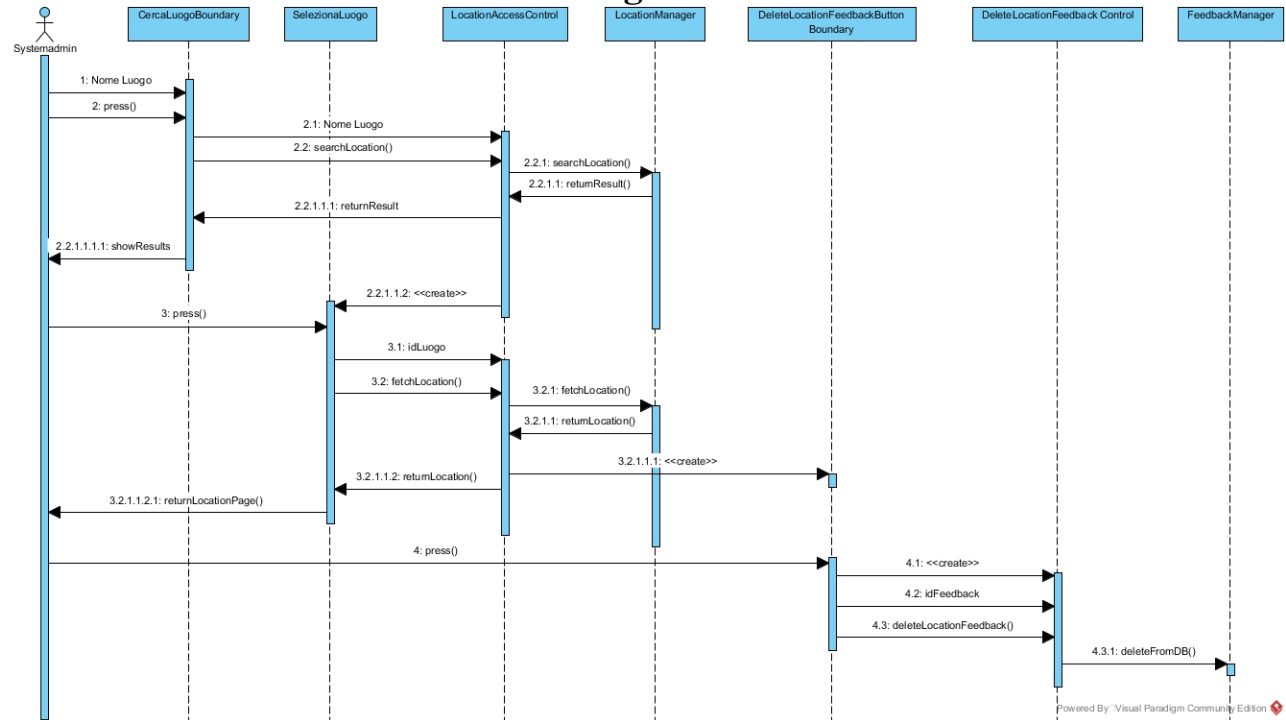
# 2.5.4.1.13 Dare Feedback Ad Un Utente



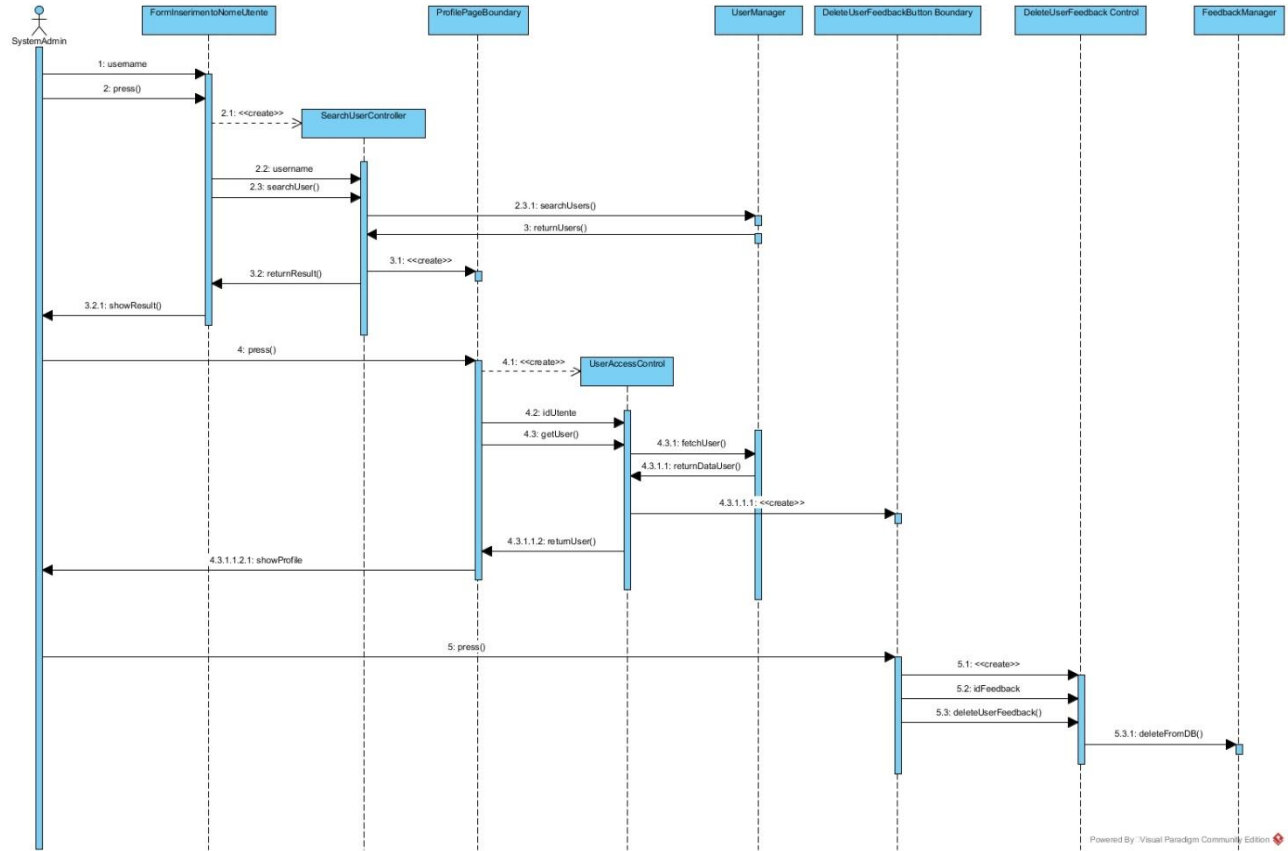
# 2.5.4.1.14 Eliminazione feedback itinerario



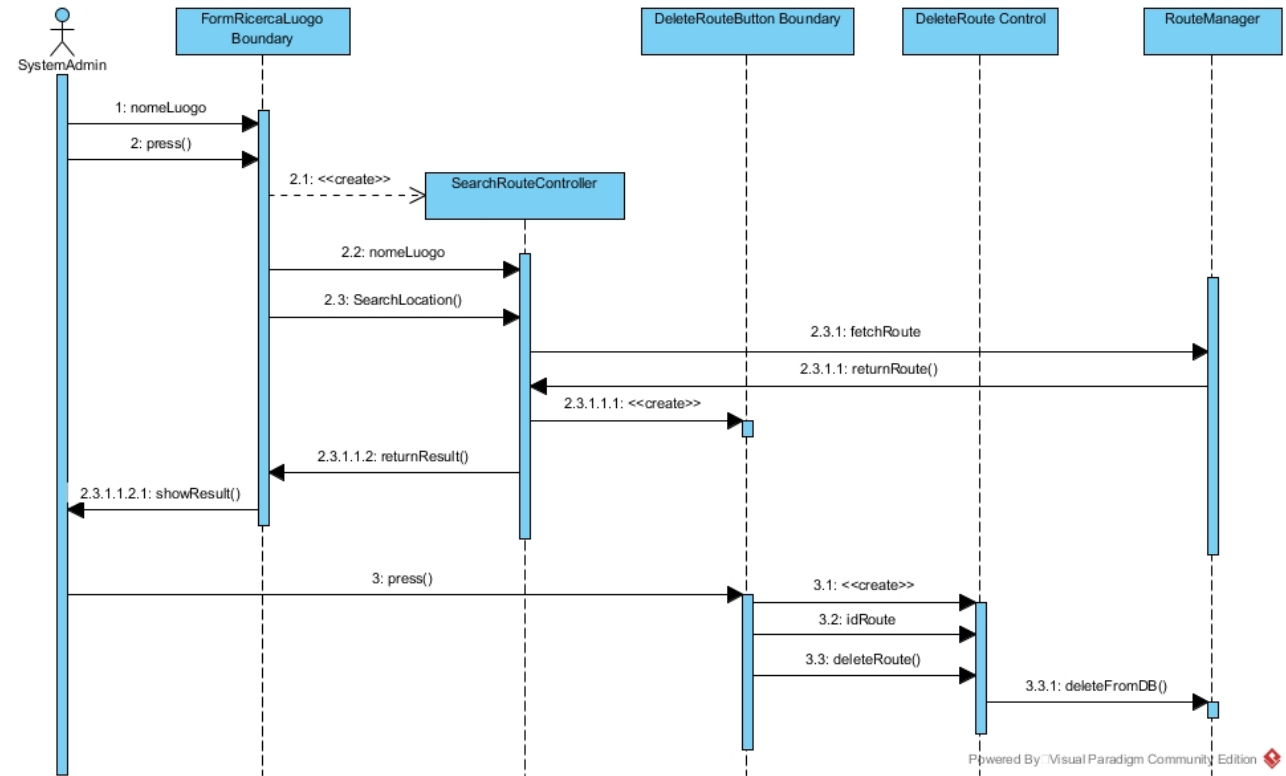
# 2.5.4.1.15 Eliminazione feedback luogo



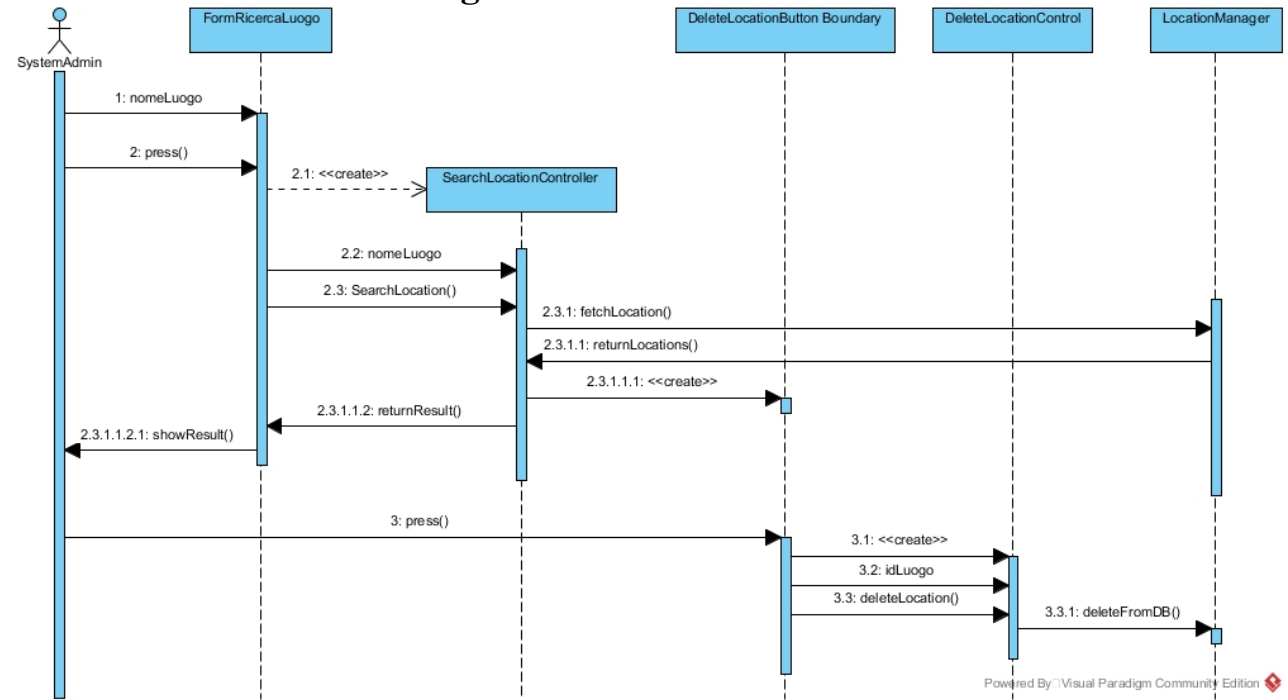
# 2.5.4.1.16 Eliminazione feedback utente



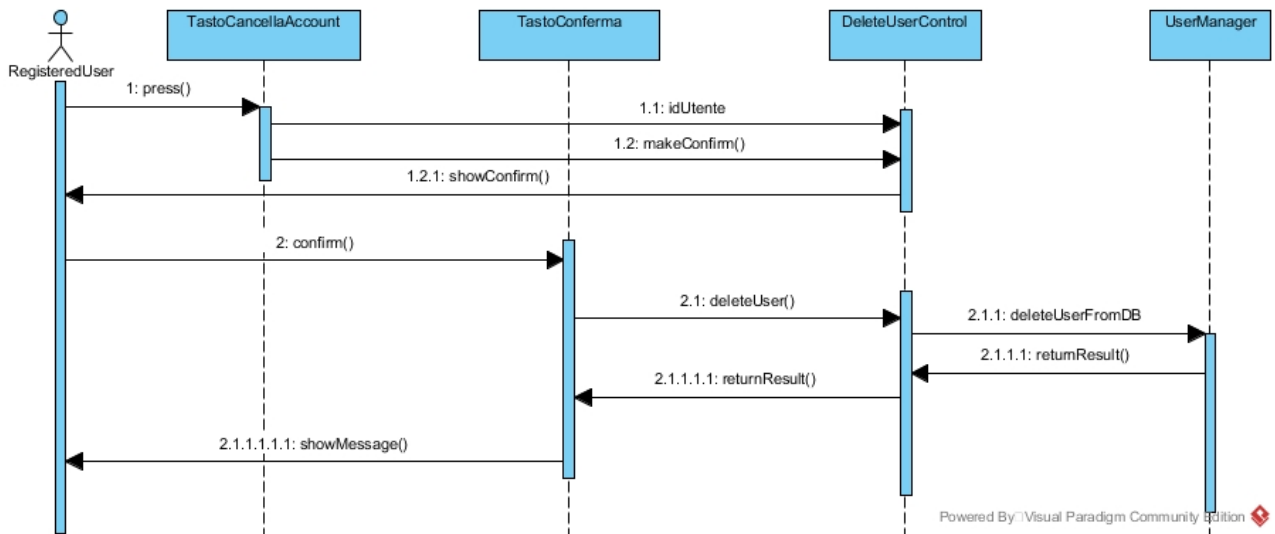
# 2.5.4.1.17 Eliminazione itinerario



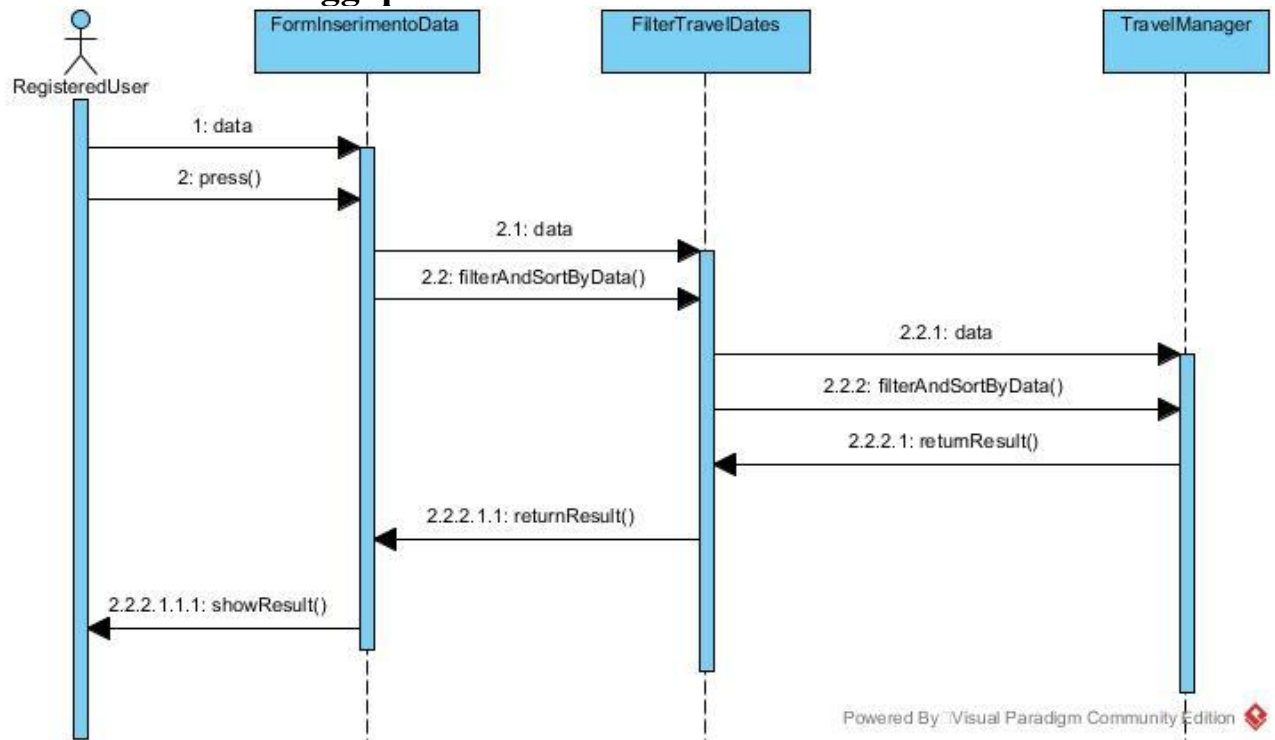
# 2.5.4.1.18 Eliminazione luogo



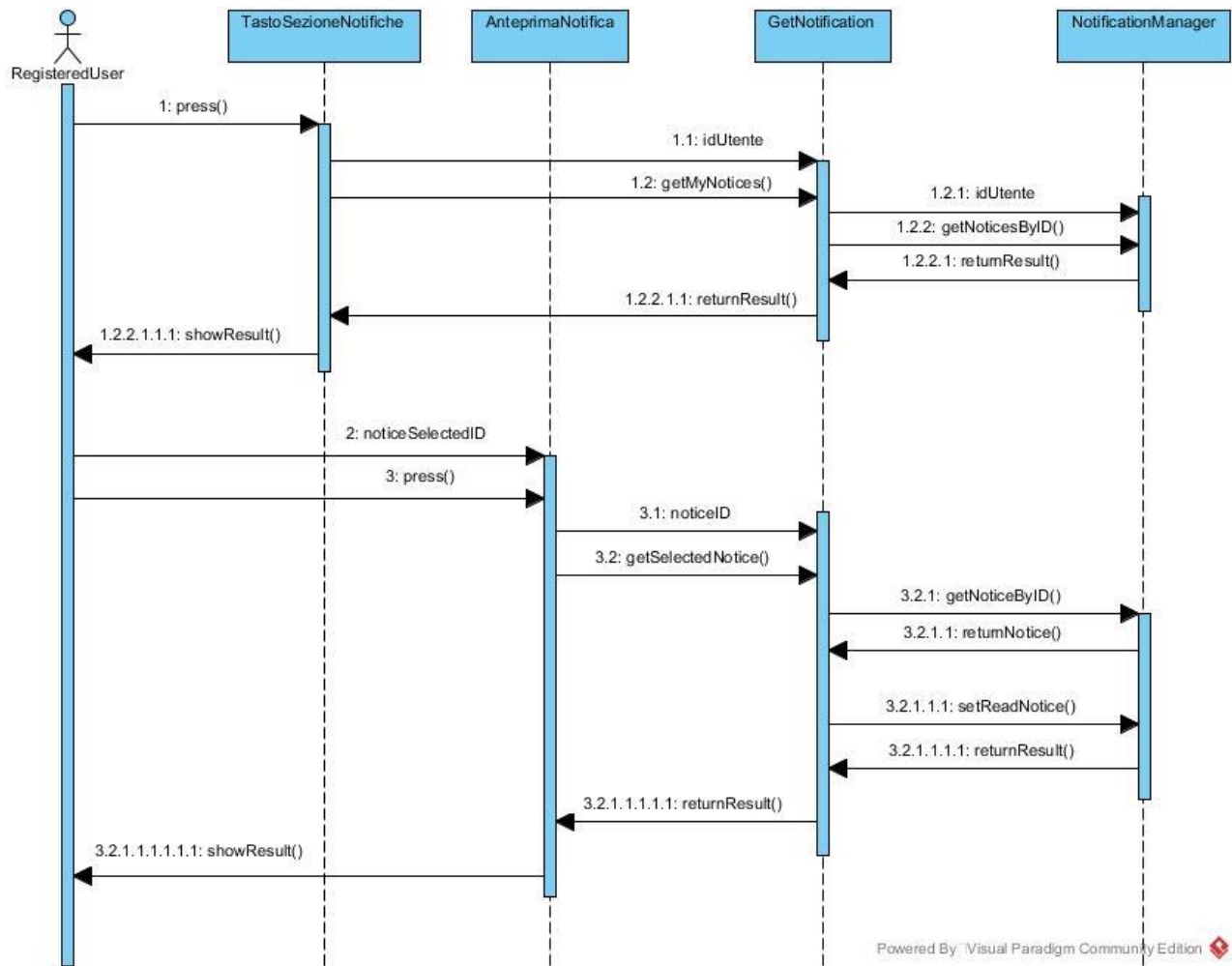
### 2.5.4.1.19 Eliminazione Account



### 2.5.4.1.20 Filtra viaggi per data

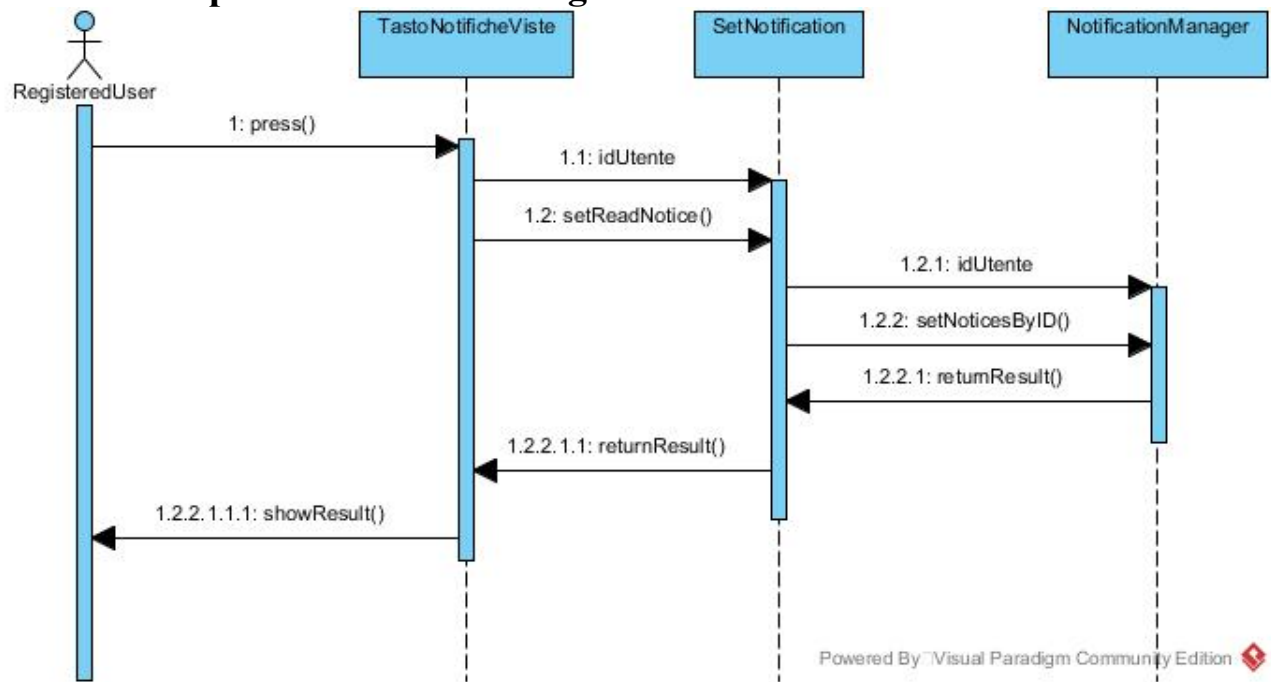


### 2.5.4.1.21 Gestione notifiche

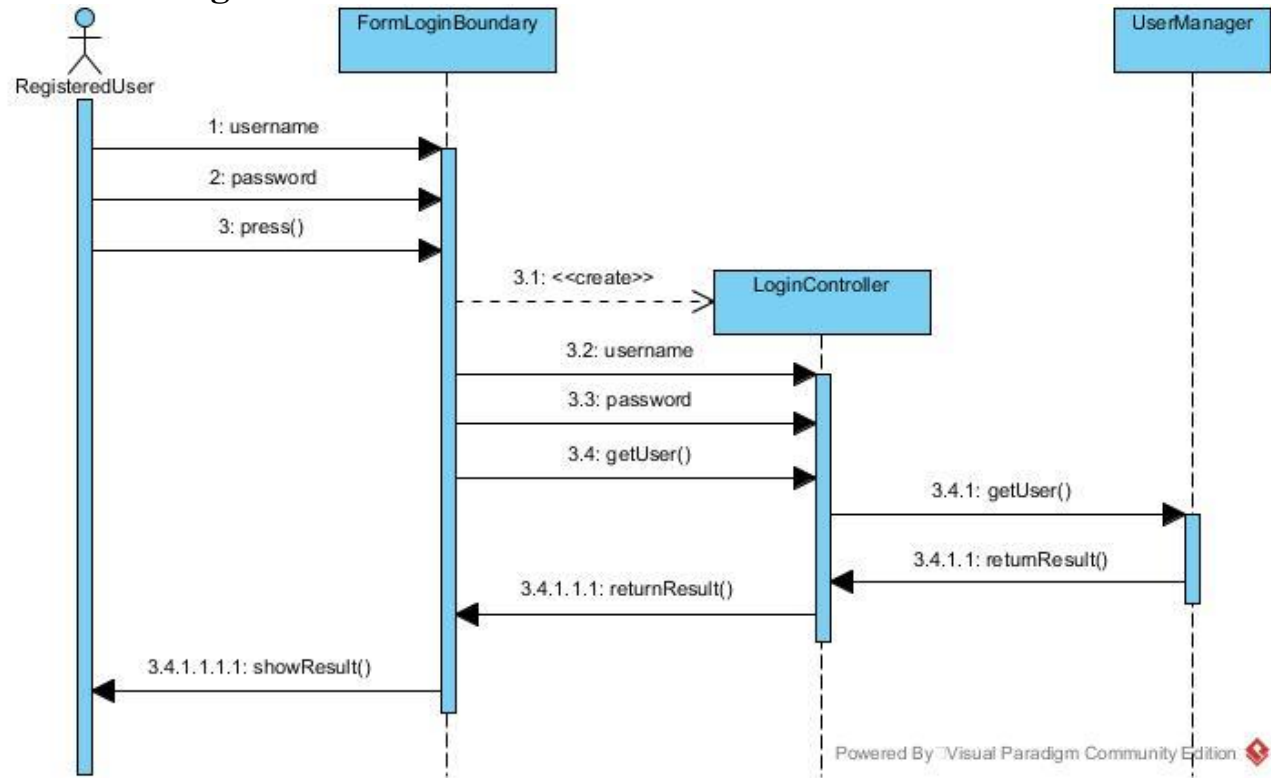




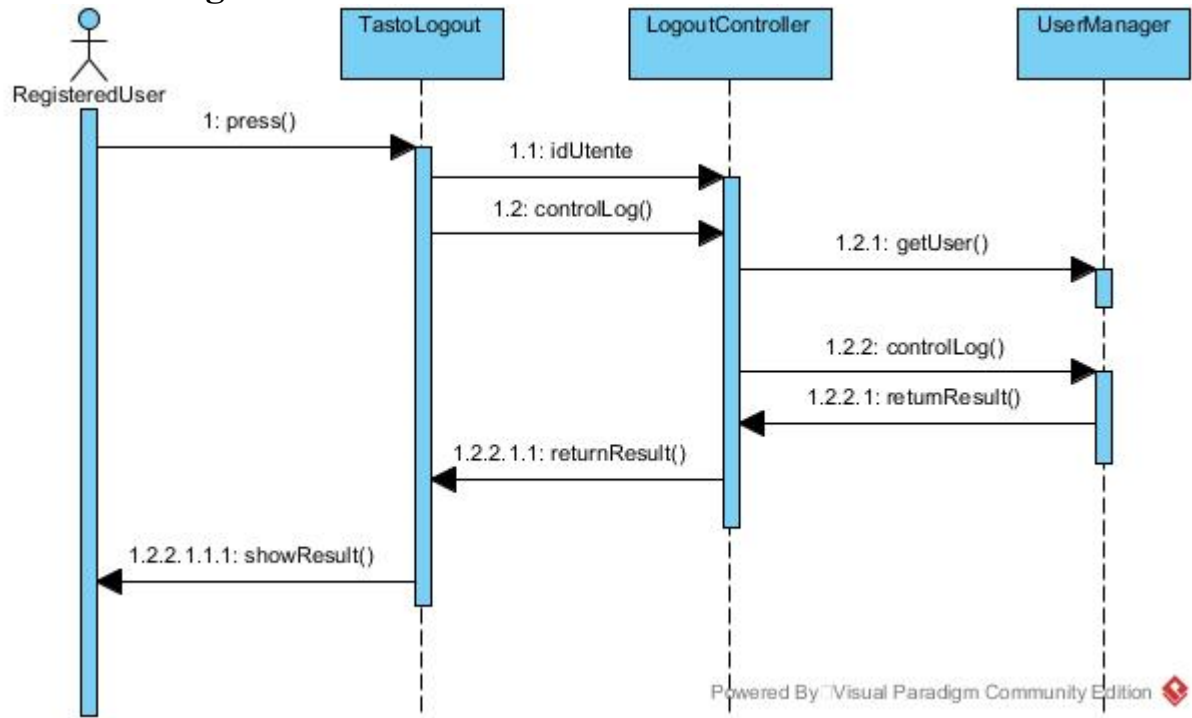
### 2.5.4.1.22 Imposta notifiche come già lette



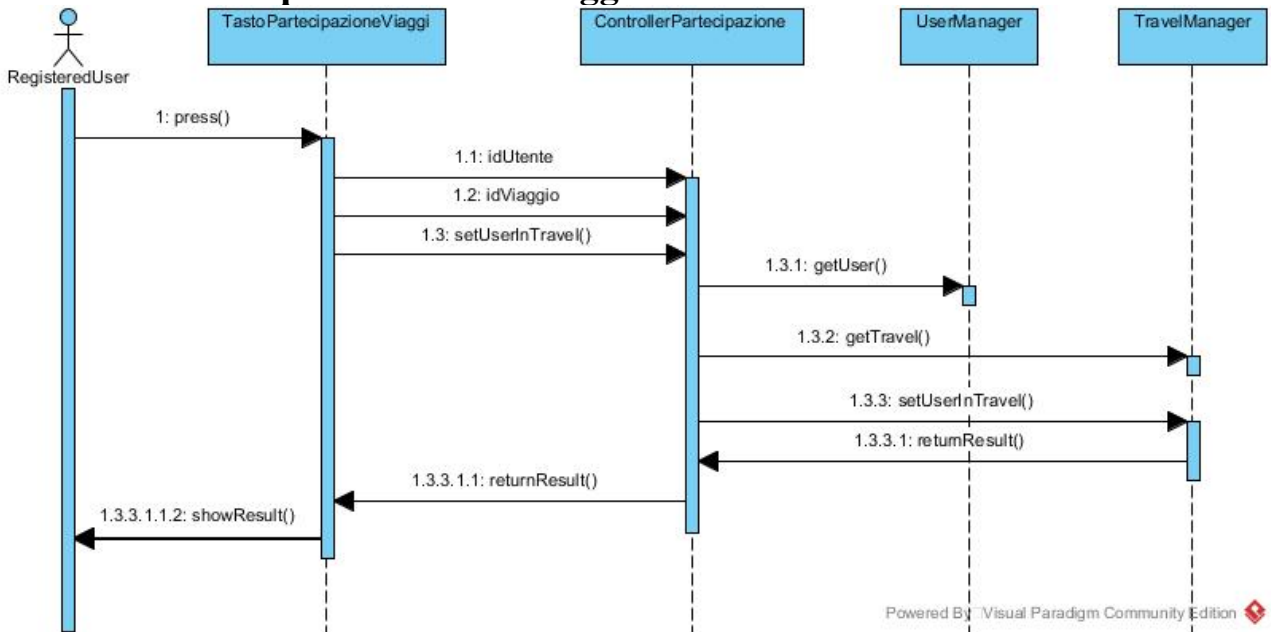
### 2.5.4.1.23 Login



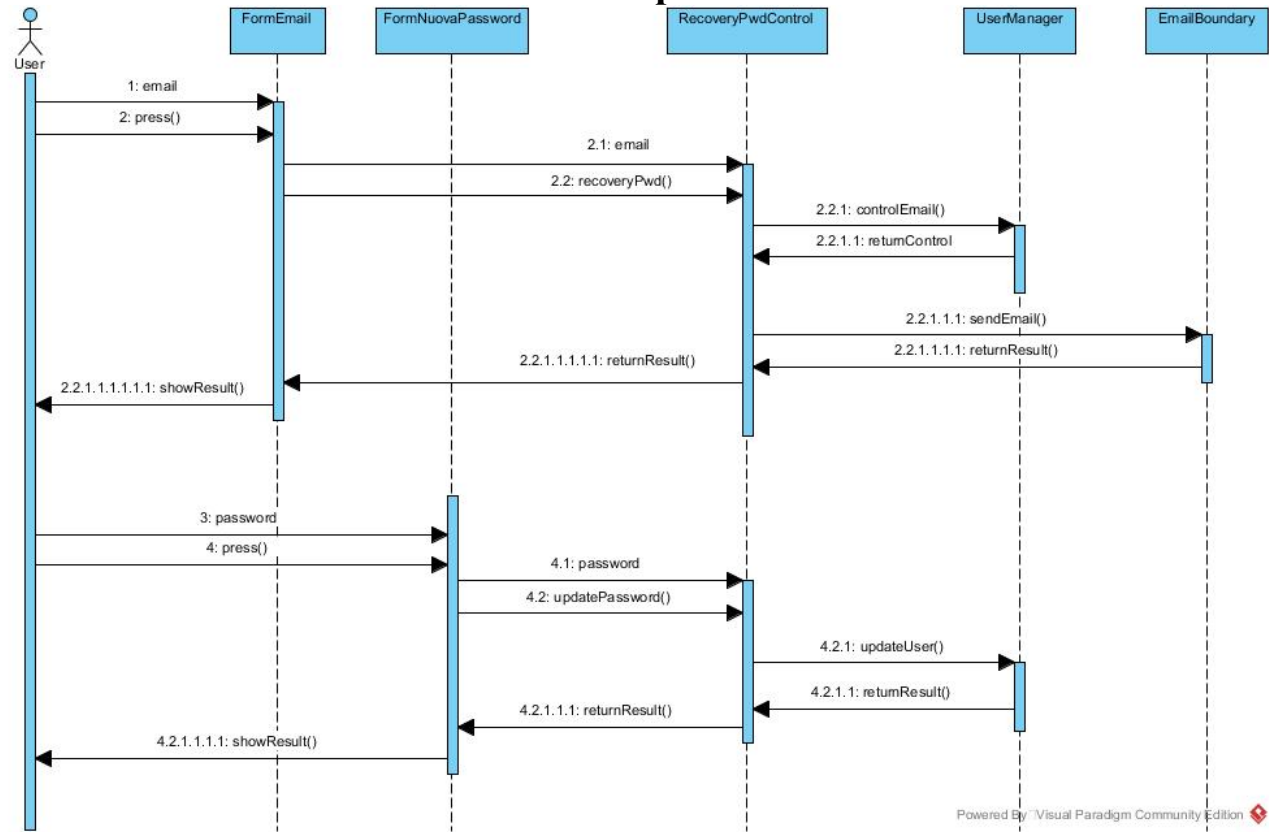
### 2.5.4.1.24 Logout



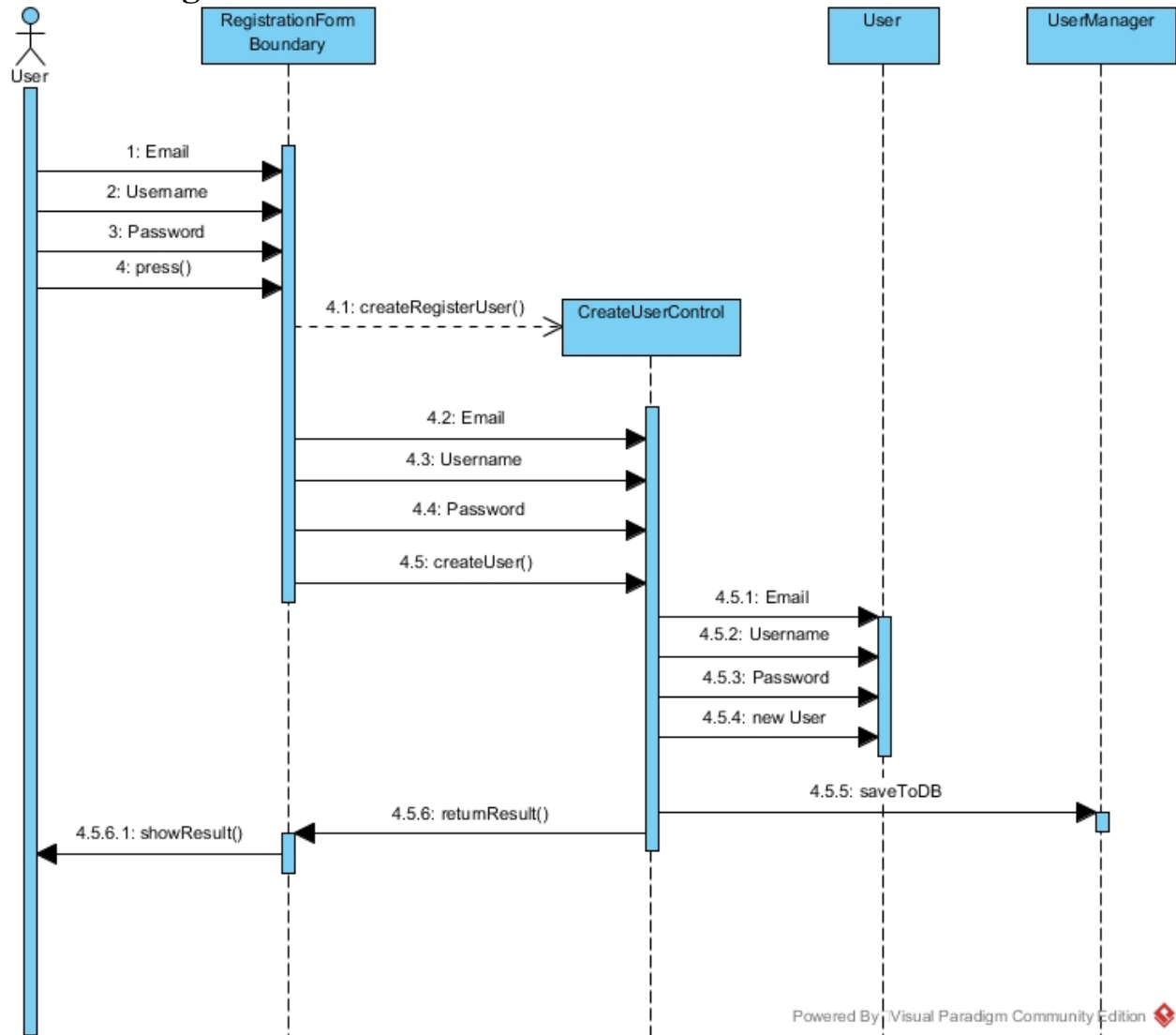
### 2.5.4.1.25 Partecipazione ad un viaggio



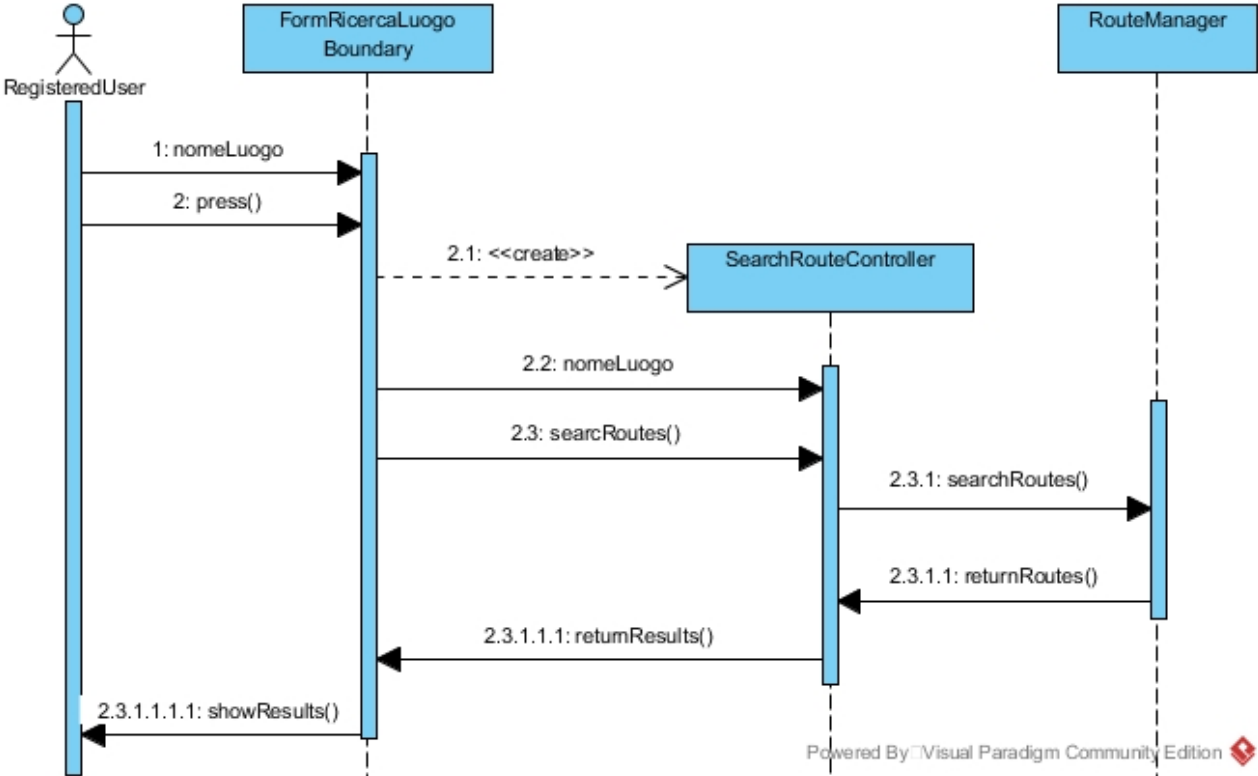
# 2.5.4.1.26 Password dimenticata - Recupero Password



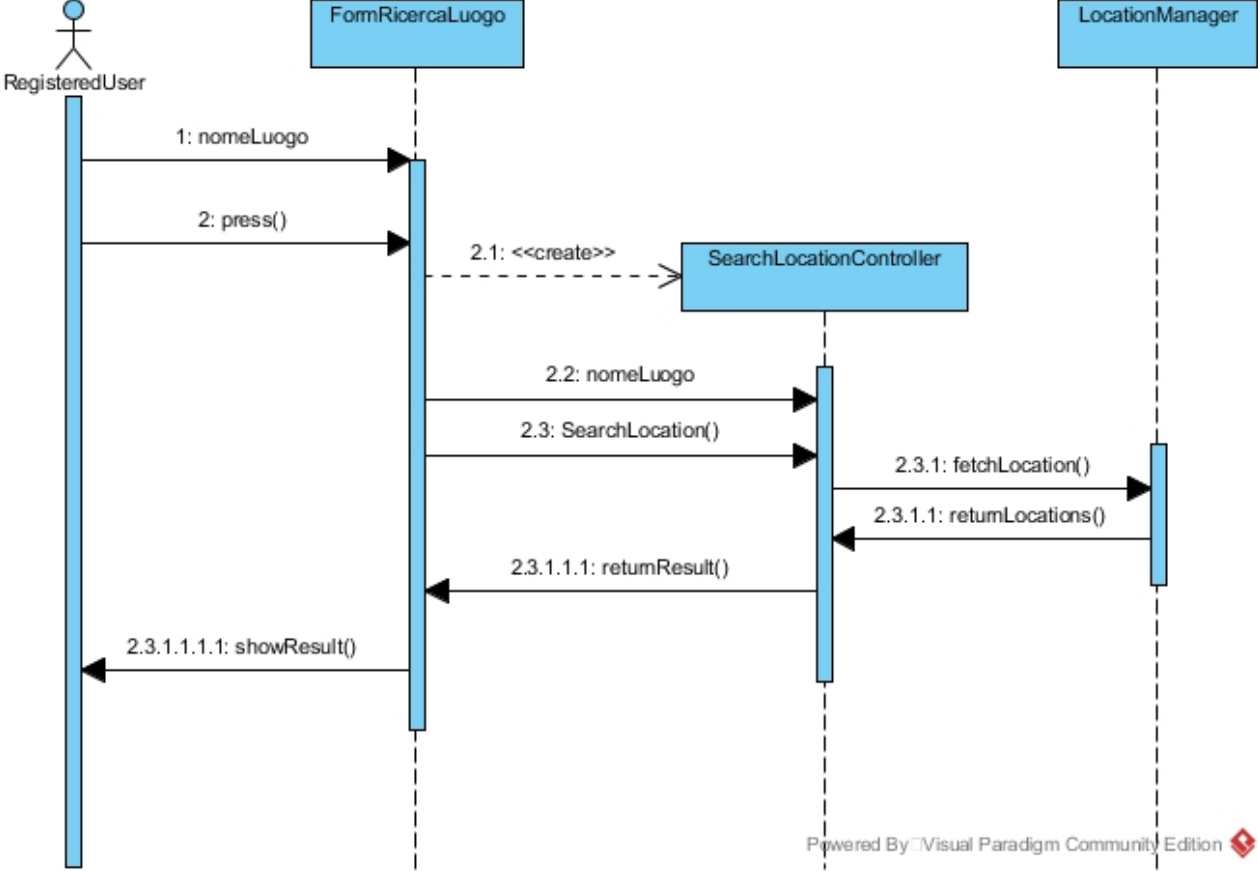
2.5.4.1.27 Registrazione



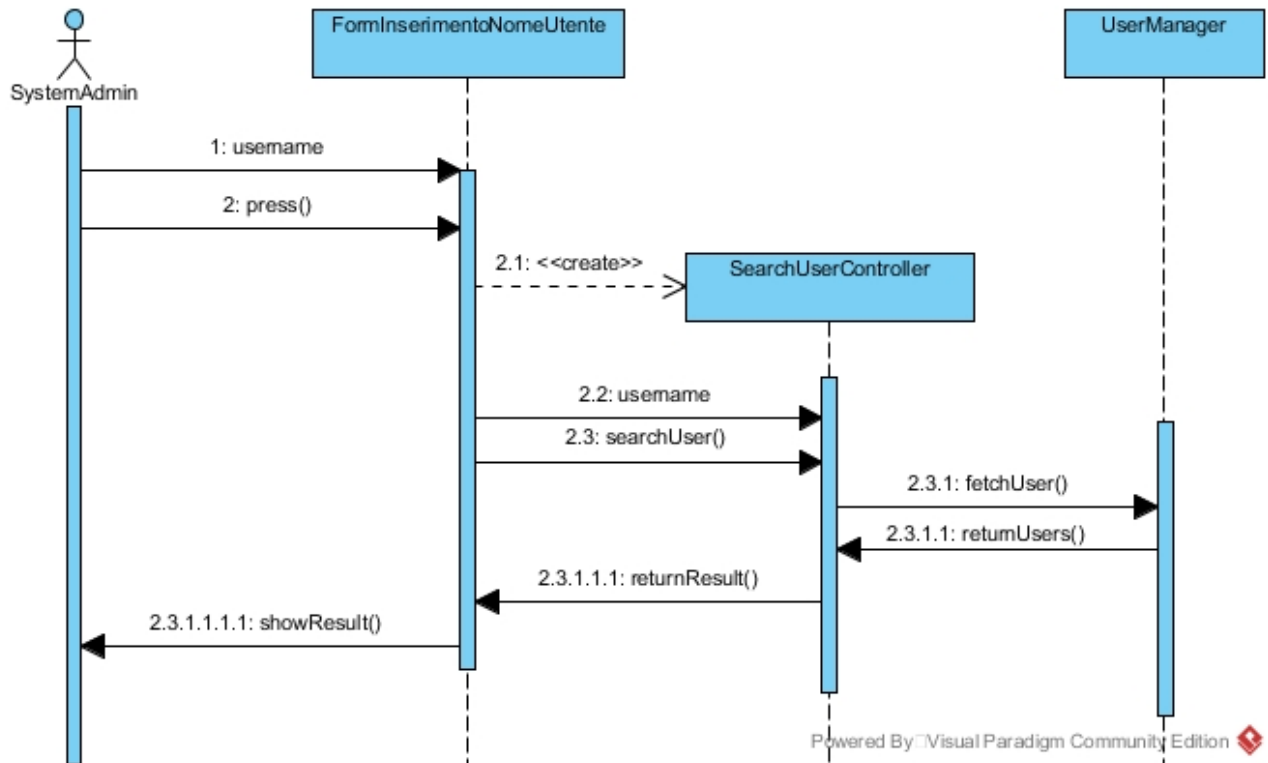
2.5.4.1.28 Ricerca Itinerario In Base Al Luogo



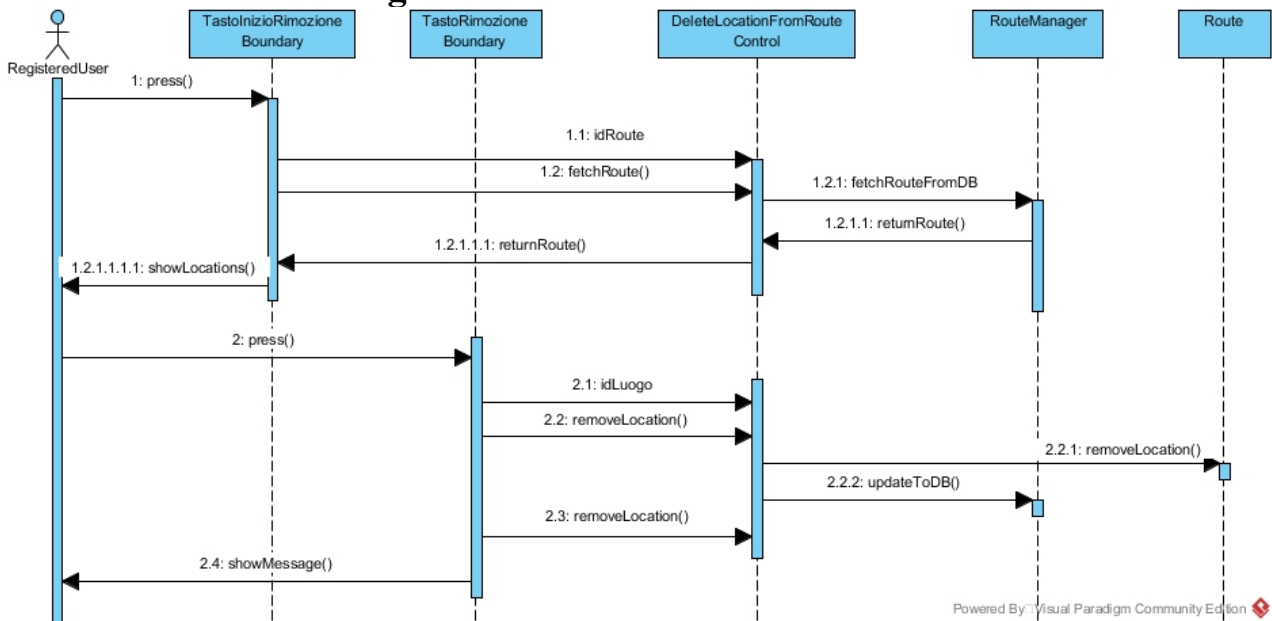
2.5.4.1.29 Ricerca Luogo



### 2.5.4.1.30 Ricerca Utente

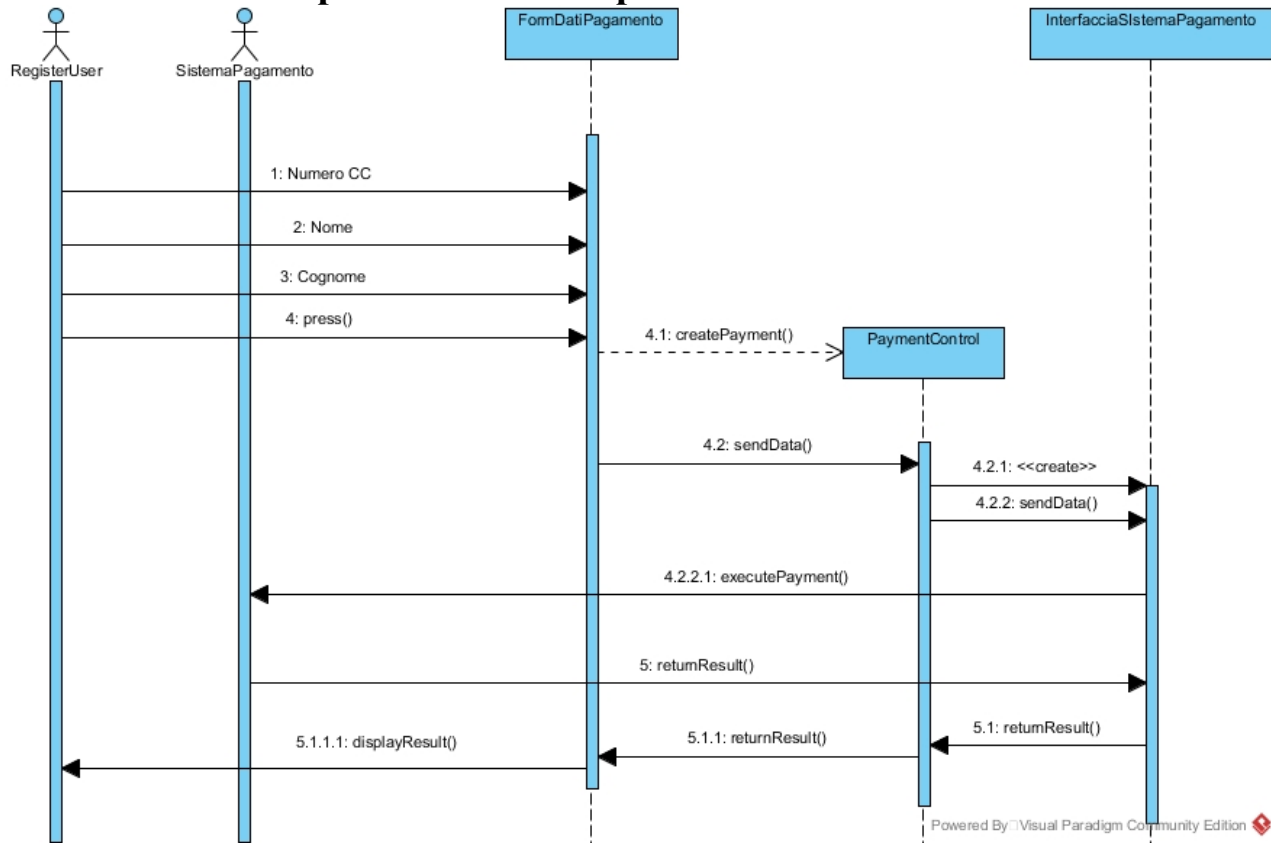


### 2.5.4.1.31 Rimuovi Luogo Da Itinerario

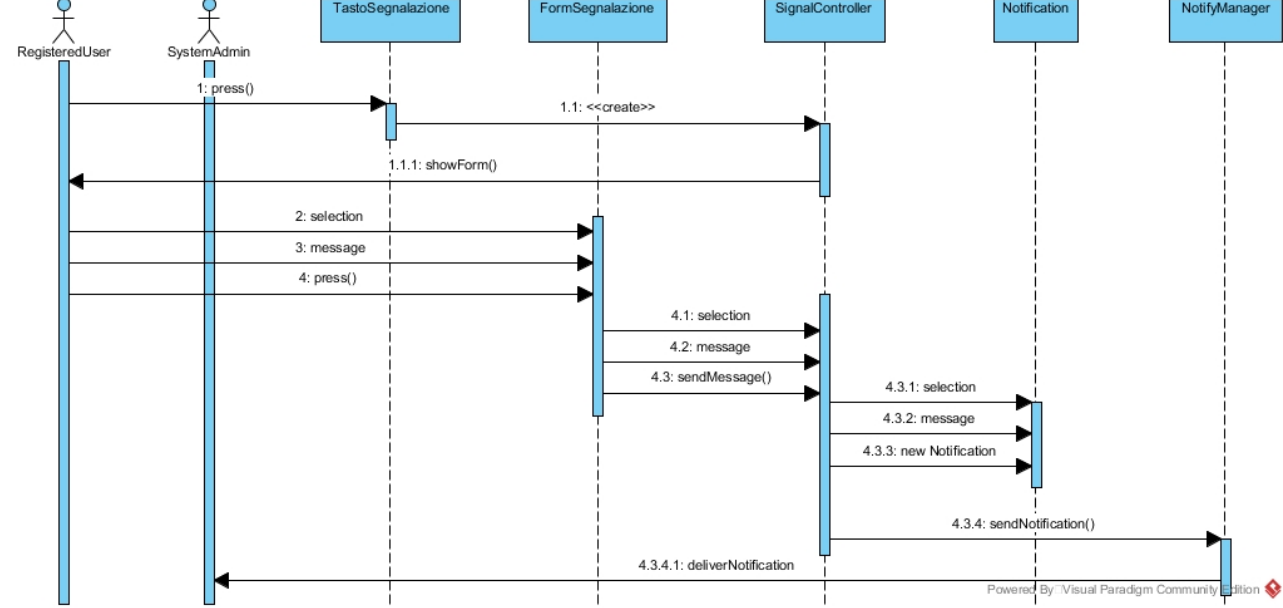




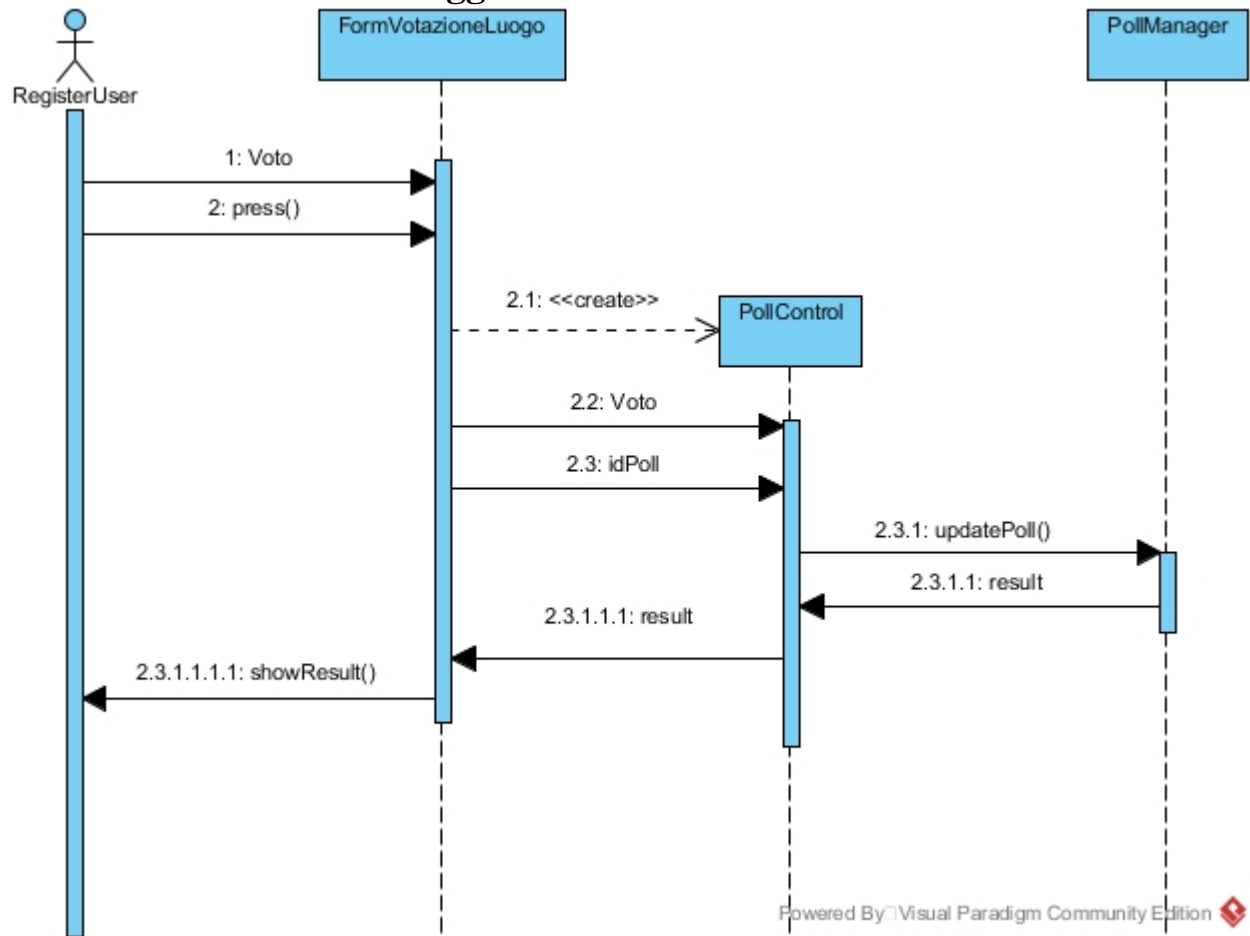
2.5.4.1.32 Rinnovo pacchetto Tour Operator



# 2.5.4.1.33 Segnalazione

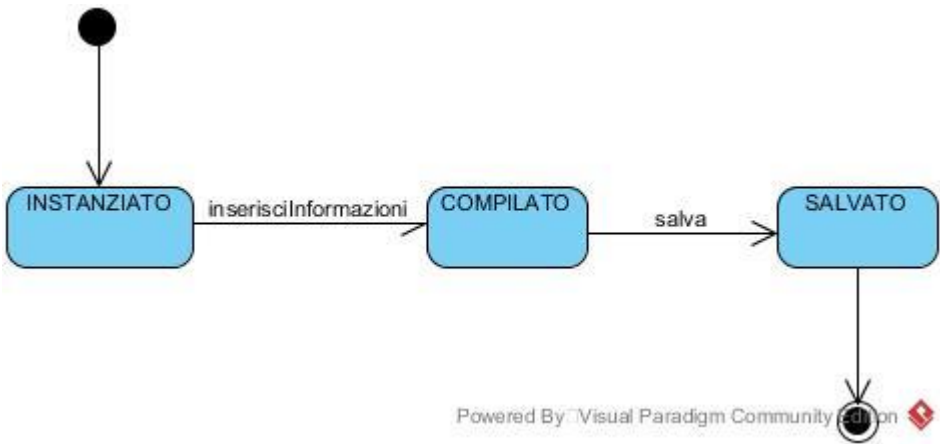


2.5.4.1.34 Votazione Sondaggio

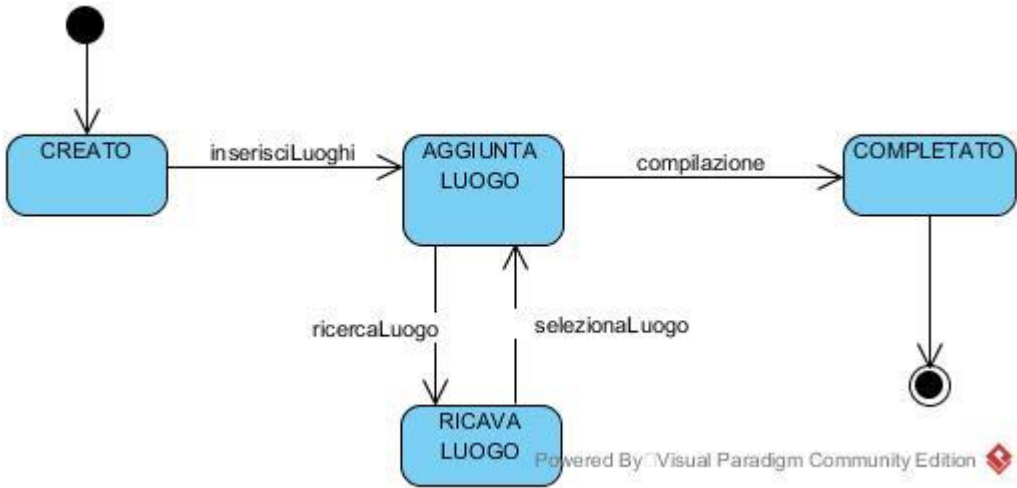


2.5.4.2 Statechart Diagrams

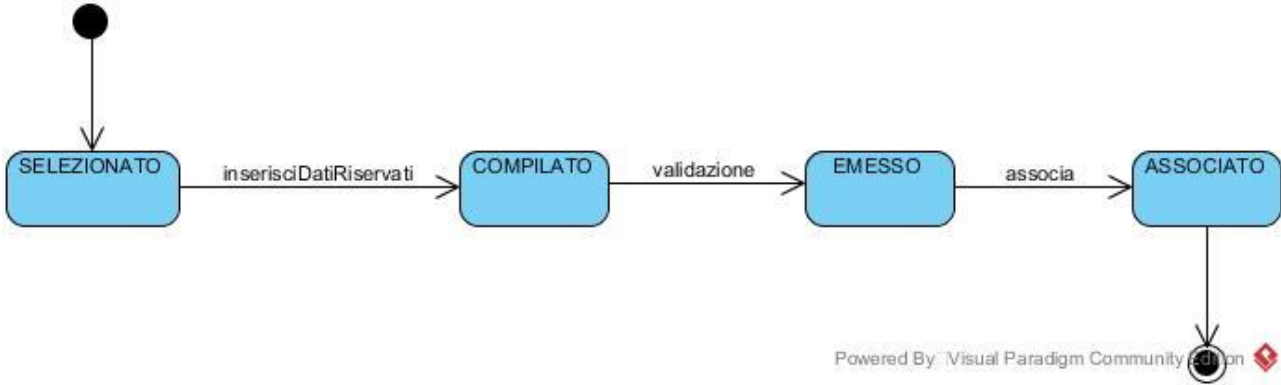
2.5.4.2.1 Utente



2.5.4.2.2 Route



2.5.4.2.3 Pacchetto Tour Operator



2.5.4.2.4 Feedback Luoghi

