

Università degli Studi di Salerno

Corso di Ingegneria del Software

Football League Manager

Requirement Analysis Document

Versione 1.14



Data: 26/01/2017

Partecipanti

Nome	Matricola
Nocerino Christian Giuseppe	0512103250
Manzo Gerardo	0512102974
Sarubbi Raffaele	0512102934
Capaldo Giovanni	0512102980
Iannuzzi Serena	0512103216

Scritto da	Nocerino Christian Giuseppe
-------------------	-----------------------------

Revision History

Data	Versione	Descrizione	Autore
11/10/2016	1.0	Aggiunte sezioni: 1. Introduction (1.1, 1.3, 1.5, 1.6) 2. Proposed system (3.1, 3.2)	Christian Giuseppe Nocerino
12/10/2016	1.1	Aggiunte sezioni: 1. Proposed system (3.3)	1. Giovanni Capaldo 2. Gerardo Manzo
14/10/2016	1.2	Aggiunte sezioni: 1. Proposed system (3.4.1, 3.4.2, 3.4.5) Correzioni	1. Raffaele Sarubbi 2. Serena Iannuzzi
18/10/2016	1.3	Correzioni	Gerardo Manzo
21/10/2016	1.4	Aggiornate sezioni: 1. Introduction (1.1) 2. Proposed system (3.2)	Giovanni Capaldo
24/10/2016	1.5	Aggiornate sezioni: 1. Proposed system (3.4.1, 3.4.2 e 3.4.5)	1. Christian Giuseppe Nocerino 2. Raffaele Sarubbi
27/10/2016	1.6	Aggiornate sezioni: 1. Introduction (1.1) 2. Current system	Giovanni Capaldo
29/10/2016	1.7	Aggiornate sezioni: Proposed system (3.1, 3.2, 3.3, 3.4)	Gerardo Manzo

01/11/2016	1.8	Aggiornate sezioni: 1. Introduction (1.6)	Serena Iannuzzi
03/11/2016	1.9	Aggiornate sezioni: 1. Proposed system (3.4.2, 3.4.5) Aggiunte sezioni: 1. Proposed system (3.4.3, 3.4.4)	1. Christian Giuseppe Nocerino 2. Gerardo Manzo
09/11/2016	1.10	Aggiunte sezioni: 1. Introduction (1.2, 1.4) 2. Glossary	Serena Iannuzzi Raffaele Sarubbi
16/11/2016	1.11	Aggiornate sezioni: 1. Proposed system (3.4.2, 3.4.3)	1. Gerardo Manzo 2. Giovanni Capaldo
20/11/2016	1.12	Aggiornate sezioni: 1. Proposed system (3.4.4, 3.4.5)	1. Raffaele Sarubbi 2. Serena Iannuzzi 3. Christian Giuseppe Nocerino
13/01/2017	1.13	Aggiornate sezioni: 1. Functional requirements (3.2) 2. Proposed system (3.4.2, 3.4.3, 3.4.4, 3.4.5)	1. Giovanni Capaldo 2. Gerardo Manzo 3. Christian Giuseppe Nocerino
26/01/2017	1.14	Aggiornate sezioni: 1. Functional requirements (3.2) 2. Proposed system (3.4.2, 3.4.4, 3.4.5)	1. Gerardo Manzo 2. Christian Giuseppe Nocerino 3. Serena Iannuzzi

Indice

1. Introduction	6
1.1. Purpose of the system.....	6
1.2. Scope of the system	6
1.3. Objectives and success criteria of the project.....	7
1.4. Definitions, acronyms, and abbreviations.....	7
1.5. References	7
1.6. Overview.....	7
2. Current System.....	8
3. Proposed system	8
3.1. Overview.....	8
3.2. Functional requirements.....	8
3.3. Nonfunctional requirements	10
3.3.1. Usability	10
3.3.2. Reliability	10
3.3.3. Performance.....	10
3.3.4. Supportability	10
3.3.5. Implementation	10
3.3.6. Interface	10
3.3.7. Packaging	11
3.3.8. Legal	11

3.4. System models.....	11
3.4.1. Scenari	11
3.4.2. Use case model	12
3.4.3. Object model.....	23
3.4.4. Dynamic model	23
3.4.5. User interface navigational paths and screen mock-ups	32
4. Glossary	44

1. Introduction

1.1. Purpose of the system

Un centro sportivo ha richiesto un sistema informatico per la gestione dei campionati di calcio a 5 organizzati all'interno della propria struttura. Il numero di squadre iscritte varia in base al campionato.

Il sistema viene utilizzato da 3 tipologie di utenti:

- il proprietario del centro che amministra il sistema (amministratore);
- gli allenatori che gestiscono la propria squadra;
- gli arbitri che inseriscono i referti delle partite.

Al momento dell'iscrizione la squadra deve essere composta da 8 giocatori. L'allenatore dovrà pagare la quota d'iscrizione per rendere effettiva l'iscrizione della sua squadra. Durante il campionato è possibile modificare la rosa.

Un calendario definisce l'ordine con il quale le squadre si affrontano. Ad ogni partita ciascuna squadra deve partecipare con almeno 5 giocatori, pena la sconfitta a tavolino. Non è posto alcun limite sul numero di sostituzioni.

Un utente non iscritto al sistema potrà soltanto aggiornarsi sugli sviluppi dei campionati. Non potrà quindi usufruire dei benefici di un utente registrato.

1.2. Scope of the system

Il sistema è mirato al miglioramento della gestione delle informazioni, dei servizi e delle altre comuni operazioni inerenti l'ambito gestionale di un campionato.

Il sistema, quindi, si prefigge i seguenti obiettivi:

- Mostrare agli utenti informazioni inerenti al campionato (come quelli riguardanti le partite, le classifiche, ecc.
- Facilitare gli utenti nella registrazione nel campionato
- Consultare informazioni sulle partite del campionato

1.3. Objectives and success criteria of the project

Il sistema si propone come obiettivo:

- Gestire le partite e la classifica di ogni campionato;
- Tenere traccia dei risultati delle squadre;
- Tenere traccia delle statistiche (goal realizzati, cartellini ricevuti etc...) dei calciatori

1.4. Definitions, acronyms, and abbreviations

Definizioni:

- assistman: Nel linguaggio sportivo, giocatore che contribuisce al buon esito di un'azione favorendo un suo compagno di squadra.
- marcatore: Nel linguaggio sportivo, il giocatore che realizza goal per la propria squadra.

1.5. References

- B. Bruegge, A.H. Dutoit, *Object Oriented Software Engineering – Using UML, Patterns and Java*, Prentice Hall, 3rd edition, 2009.

1.6. Overview

Il sistema offre la possibilità di iscriversi ai campionati organizzati in un centro sportivo. Il proprietario si occupa di aprire un nuovo campionato. Ad inizio campionato gli allenatori possono iscrivere le proprie squadre. Inoltre hanno la possibilità di visualizzare la rosa della propria squadra, il calendario e la classifica del campionato. Gli arbitri poi hanno il compito di inserire i referti delle partite, indicando il risultato, il numero di goal e assist di ogni giocatore e eventuali cartellini assegnati a questi ultimi.

Gli utenti non iscritti al sistema possono comunque tenersi aggiornati sull'andamento dei vari campionati.

2. Current System

Al momento il centro sportivo non dispone di un sistema informatico per la gestione dei campionati, per cui il sistema proposto dovrà essere sviluppato ex novo.

3. Proposed system

3.1. Overview

In questa sezione viene fornita una panoramica dei requisiti funzionali e di quelli non funzionali del sistema proposto. Vengono quindi descritti gli scenari più significativi e i casi d'uso.

3.2. Functional requirements

Il sistema deve offrire agli allenatori la possibilità di registrarsi. La registrazione degli arbitri è affidata al proprietario del centro sportivo che ha quindi il ruolo di amministratore.

L'amministratore ha il compito di:

- Creare un campionato e aprirne le iscrizioni (la chiusura delle iscrizioni avviene in automatico quando è stato raggiunto il numero di squadre prestabilito);
- Iscrivere un arbitro al sistema;
- Confermare l'iscrizione di una squadra.
- Assegnare un arbitro ad una partita.
- Iscrivere un giocatore al sistema.
- Modificare le informazioni di un giocatore.
- Chiudere un campionato.

Gli allenatori devono poter:

- Creare una squadra;
- Iscrivere la propria squadra ad un campionato;
- Visualizzare la rosa di ogni squadra;
- Sostituire un giocatore;

- Inserire giocatore alla rosa
- Visualizzare il calendario;
- Visualizzare la classifica del campionato e le statistiche di ogni partita;

Gli arbitri hanno il compito di inserire il referto di ogni partita. Questo include:

- Il risultato della partita, il numero di goal e assist di ogni giocatore;
- I cartellini con relativa motivazione e nel caso di espulsione il numero di giornate di squalifica.

Le sostituzioni effettuate non vengono riportate in quanto non è posto alcun limite sul numero.

3.3. Nonfunctional requirements

3.3.1. Usability

Le interfacce utente sono composte principalmente da messaggi testuali e/o moduli. Tutte le operazioni sono gestite mediante la compilazione e la sottomissione di tali moduli.

3.3.2. Reliability

Il sistema deve essere disponibile per la maggior parte del tempo. È previsto un server di backup in caso di malfunzionamento del sistema. Se il sistema risulta fuori servizio, esse dev'essere reso nuovamente disponibile entro 2 ore. In tal caso è comunque possibile usare altri metodi per usufruire dei servizi del sistema (email o moduli cartacei, da recapitare all'amministratore). L'amministratore dovrà registrare eventuali modifiche non appena il sistema ritorna disponibile.

Gli utenti devono registrarsi al sistema. Per eseguire le operazioni è richiesto l'accesso al sistema.

3.3.3. Performance

Il sistema deve rispondere agli input in tempo reale.

3.3.4. Supportability

È previsto lo sviluppo di un'app per cellulari che offre la possibilità di consultare le statistiche.

3.3.5. Implementation

Il sistema deve essere implementato in linguaggio Java.

3.3.6. Interface

Non applicabile al sistema proposto.

3.3.7. Packaging

Il sistema viene sviluppato come applicazione web. Potranno quindi usufruire del sistema tutti gli utenti che possiedono una connessione ad Internet e un web browser.

3.3.8. Legal

Non è richiesta alcuna licenza per l'utilizzo del sistema.

3.4. System models

3.4.1. Scenari

Creazione campionato "Holly & Benji"

Francesco, il proprietario del sistema Football League Manager, ha deciso di organizzare un nuovo campionato dal nome "Holly & Benji". Per crearlo dovrà effettuare il login all'interno del sito inserendo la propria email e password. Una volta entrato, avrà a disposizione una pagina in cui solo lui può accedere; qui potrà svolgere le seguenti operazioni:

1. inserire il nome del campionato
2. decidere il numero delle squadre

Iscrizione squadra "Molossi"

Jhonny vuole partecipare al campionato "Holly & Benji", per fare ciò dovrà accedere al sistema fornendo le credenziali (e-mail, password). A questo punto entrerà nella pagina dedicata al campionato dove potrà creare e registrare la propria squadra che verrà salvata e confermata da Francesco. Le uniche informazioni relative alla squadra saranno il nome della stessa e i dati anagrafici dei giocatori.

Report delle statistiche per la partita Molossi - Tigers

Kumar (arbitro) avrà il compito di pubblicare le statistiche (risultato, goal e assist e cartellini) della partita da lui arbitrata sulla pagina del campionato. Il report avviene attraverso una sezione in cui vi saranno dei campi da compilare con le relative statistiche della partita.

Fine del campionato “Holly & Benji”

Una volta terminate tutte le partite, alla squadra vincitrice verrà consegnato il premio “Golden Ball” (consiste nell’ammontare di tutte le quote raccolte dai vari partecipanti) che sarà consegnato direttamente da Francesco.

3.4.2. Use case model

ID	UC1
Name	Creazione campionato
Participating actors	Amministratore
Entry condition	Il caso d’uso inizia quando l’amministratore seleziona dal menù l’opzione “Crea nuovo campionato”.
Exit condition	Il campionato è stato creato e le iscrizioni sono aperte.
Event flow	<ol style="list-style-type: none">1. L’amministratore inserisce il nome del campionato.2. L’amministratore inserisce il numero di squadre che potranno iscriversi.3. L’amministratore conferma i dati inseriti.
References	figura 3.4.4.1, figura 3.4.5.2

ID	UC2
Name	Assegnazione arbitro
Participating actors	Amministratore
Entry condition	Il caso d'uso inizia quando l'amministratore seleziona dal menù l'opzione "Assegna arbitro".
Exit condition	L'arbitro è stato assegnato alla partita scelta.
Event flow	<ol style="list-style-type: none"> 1. L'amministratore seleziona una partita. 2. L'amministratore seleziona un arbitro. 3. L'amministratore conferma la scelta.
References	figura 3.4.4.2, figura 3.4.5.3

ID	UC3
Name	Registrazione allenatore
Participating actors	Allenatore
Entry condition	Il caso d'uso inizia quando l'allenatore seleziona l'opzione "Registrati".
Exit condition	L'allenatore è registrato al sistema.
Event flow	<ol style="list-style-type: none"> 1. L'allenatore inserisce una email ed una password. 2. L'allenatore inserisce le proprie generalità(Nome,Cognome) 3. L'allenatore conferma i dati inseriti.
References	figura 3.4.4.3, figura 3.4.5.1, figura 3.4.5.4

ID	UC4
Name	Creazione squadra
Participating actors	Allenatore
Entry condition	Il caso d'uso inizia quando l'allenatore seleziona l'opzione "Crea squadra".
Exit condition	La squadra è stata creata.
Event flow	<ol style="list-style-type: none"> 1. L'allenatore inserisce il nome della squadra. 2. L'allenatore conferma i dati inseriti.
References	Figura 3.4.4.4, figura 3.4.5.5

ID	UC5
Name	Registrazione arbitro
Participating actors	Amministratore
Entry condition	Il caso d'uso inizia quando l'amministratore seleziona l'opzione "Registra arbitro".
Exit condition	L'arbitro è registrato al sistema.
Event flow	<ol style="list-style-type: none"> 1. L'amministratore inserisce una email ed una password per l'arbitro. 2. L'amministratore inserisce le generalità dell'arbitro. 3. L'allenatore conferma i dati inseriti.
References	figura 3.4.4.5, figura 3.4.5.6

ID	UC6
Name	Registrazione referto
Participating actors	Arbitro
Entry condition	Il caso d'uso inizia quando l'arbitro seleziona l'opzione "Registra referto".
Exit condition	Il referto è registrato sul sistema.
Event flow	<ol style="list-style-type: none"> 1. L'arbitro seleziona la partita. 2. L'arbitro inserisce la data della partita. 3. L'arbitro indica per ogni giocatore numero di goal, numero di assist, cartellini, motivazione e durata squalifica. 4. L'arbitro conferma i dati inseriti.
References	Figura 3.4.4.6, figura 3.4.5.7.1, figura 3.4.5.7.2

ID	UC7
Name	Visualizzazione rosa
Participating actors	Allenatore
Entry condition	Il caso d'uso inizia quando l'allenatore seleziona l'opzione "Visualizza rosa".
Exit condition	La rosa viene mostrata all'allenatore.
Event flow	<ol style="list-style-type: none"> 1. L'allenatore seleziona la squadra. 2. L'allenatore conferma la squadra selezionata.
References	figura 3.4.4.7, figura 3.4.5.8.1, figura 3.4.5.8.2

ID	UC8
Name	Modifica giocatore
Participating actors	Amministratore
Entry condition	Il caso d'uso inizia quando l'amministratore seleziona l'opzione "Modifica giocatore".
Exit condition	Le informazioni del giocatore sono state modificate.
Event flow	<ol style="list-style-type: none"> 1. L'amministratore seleziona il giocatore di cui intende aggiornare le informazioni. 2. L'amministratore inserisce le nuove informazioni del giocatore. 3. L'amministratore conferma i dati inseriti.
References	Figura 3.4.4.8, figura 3.4.5.9

ID	UC9
Name	Sostituisci giocatore
Participating actors	Allenatore
Entry condition	Il caso d'uso inizia quando l'allenatore seleziona l'opzione "Sostituisci giocatore".
Exit condition	Il giocatore è stato sostituito
Event flow	<ol style="list-style-type: none"> 1. L'allenatore seleziona la squadra. 2. L'allenatore seleziona il giocatore che intende sostituire dalla squadra. 3. L'allenatore seleziona il giocatore che intende aggiungere alla squadra. 4. L'allenatore conferma i dati inseriti.
References	Figura 3.4.4.9, figura 3.4.5.10.1

ID	UC10
Name	Visualizzazione classifica
Participating actors	Allenatore
Entry condition	Il caso d'uso inizia quando l'allenatore seleziona l'opzione "Visualizza classifica".
Exit condition	La classifica viene mostrata all'allenatore.
Event flow	<ol style="list-style-type: none"> 1. L'allenatore seleziona il campionato. 2. L'allenatore conferma il campionato selezionato.
References	figura 3.4.4.10, figura 3.4.5.11.1, figura 3.4.5.11.2

ID	UC11
Name	Visualizzazione calendario
Participating actors	Allenatore
Entry condition	Il caso d'uso inizia quando l'allenatore seleziona l'opzione "Visualizza calendario".
Exit condition	Il calendario viene mostrato all'allenatore.
Event flow	<ol style="list-style-type: none"> 1. L'allenatore seleziona il campionato. 2. L'allenatore conferma il campionato selezionato.
References	figura 3.4.4.11, figura 3.4.5.12.1, figura 3.4.5.12.2

ID	UC12
Name	Chiusura campionato
Participating actors	Amministratore
Entry condition	Il caso d'uso inizia quando tutte le partite sono state disputate.
Exit condition	Il campionato è stato eliminato dal sistema.
Event flow	<ol style="list-style-type: none"> 1. L'amministratore seleziona il campionato. 2. L'amministratore conferma il campionato selezionato. 3. L'amministratore elimina il campionato dal sistema.
References	figura 3.4.4.12, figura 3.4.5.14

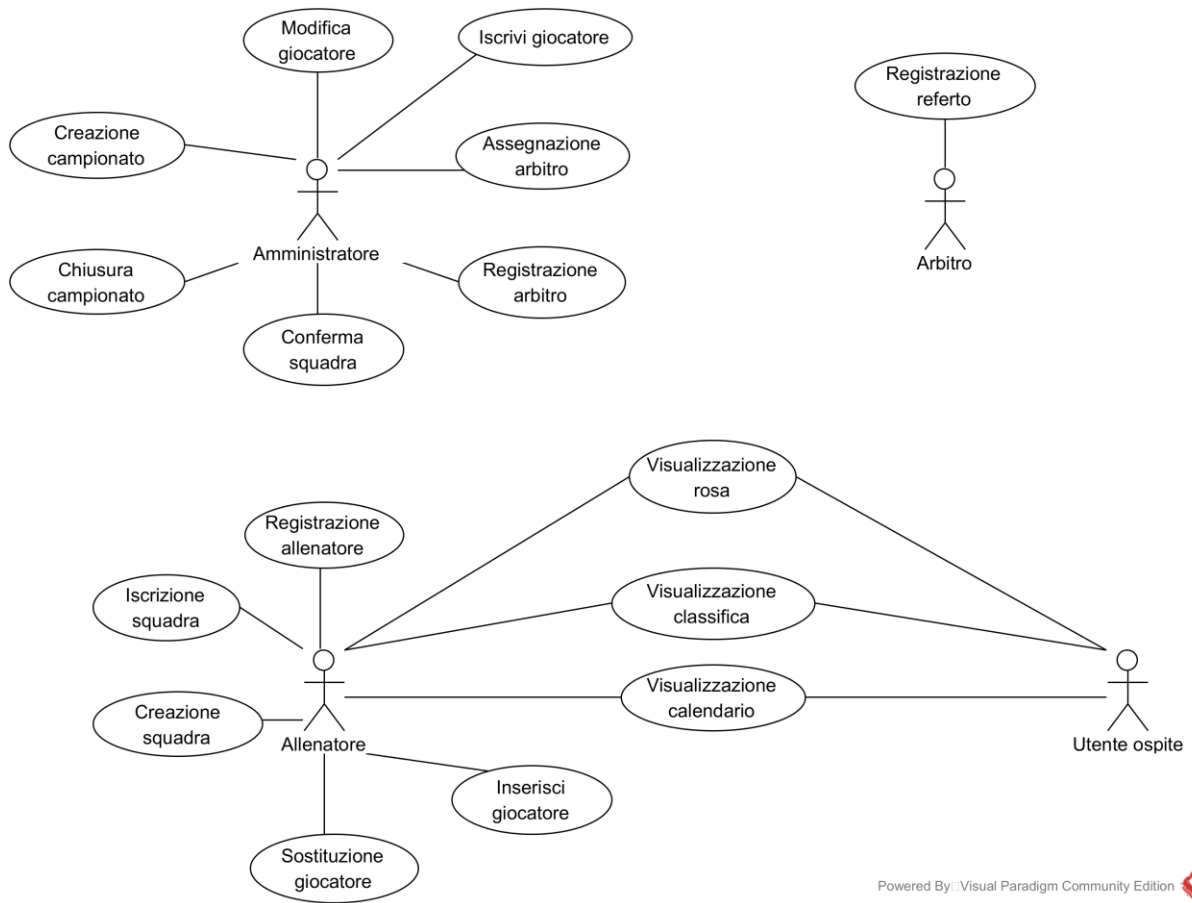
ID	UC13
Name	Conferma squadra
Participating actors	Amministratore
Entry condition	Il caso d'uso inizia quando l'amministratore seleziona "Conferma squadra".
Exit condition	La squadra è pronta per l'iscrizione.
Event flow	<ol style="list-style-type: none"> 1. L'amministratore sceglie la squadra da confermare. 2. L'amministratore conferma la scelta.
References	figura 3.4.4.13, figura 3.4.5.13

ID	UC14
Name	Iscrivi giocatore
Participating actors	Amministratore
Entry condition	Il caso d'uso inizia quando l'amministratore seleziona "Iscrivi giocatore"
Exit condition	Il giocatore è iscritto al sistema.
Event flow	<ol style="list-style-type: none"> 1. L'amministratore inserisce i dati anagrafici del giocatore (nome e cognome). 2. L'amministratore conferma i dati inseriti.
References	Figura 3.4.4.15, figura 3.4.5.15

ID	UC15
Name	Iscrivi squadra
Participating actors	Allenatore
Entry condition	Il caso d'uso inizia quando l'allenatore seleziona "Iscrivi squadra"
Exit condition	La squadra è in attesa di conferma ⁹ .
Event flow	<ol style="list-style-type: none"> 1. L'allenatore seleziona la squadra da iscrivere. 2. L'allenatore seleziona il campionato a cui partecipare. 3. L'allenatore conferma i dati inseriti.
References	Figura 3.4.4.14, figura 3.4.5.16

ID	UC16
Name	Inserisci Giocatore
Participating actors	Allenatore
Entry condition	Il caso d'uso inizia quando l'allenatore seleziona "Inserisci giocatore"
Exit condition	Il giocatore è inserito in squadra.
Event flow	<ol style="list-style-type: none"> 1. L'allenatore inserisce i dati anagrafici del giocatore (nome e cognome). 2. L'allenatore conferma i dati inseriti.
References	Figura 3.4.4.16, figura 3.4.5.10.2

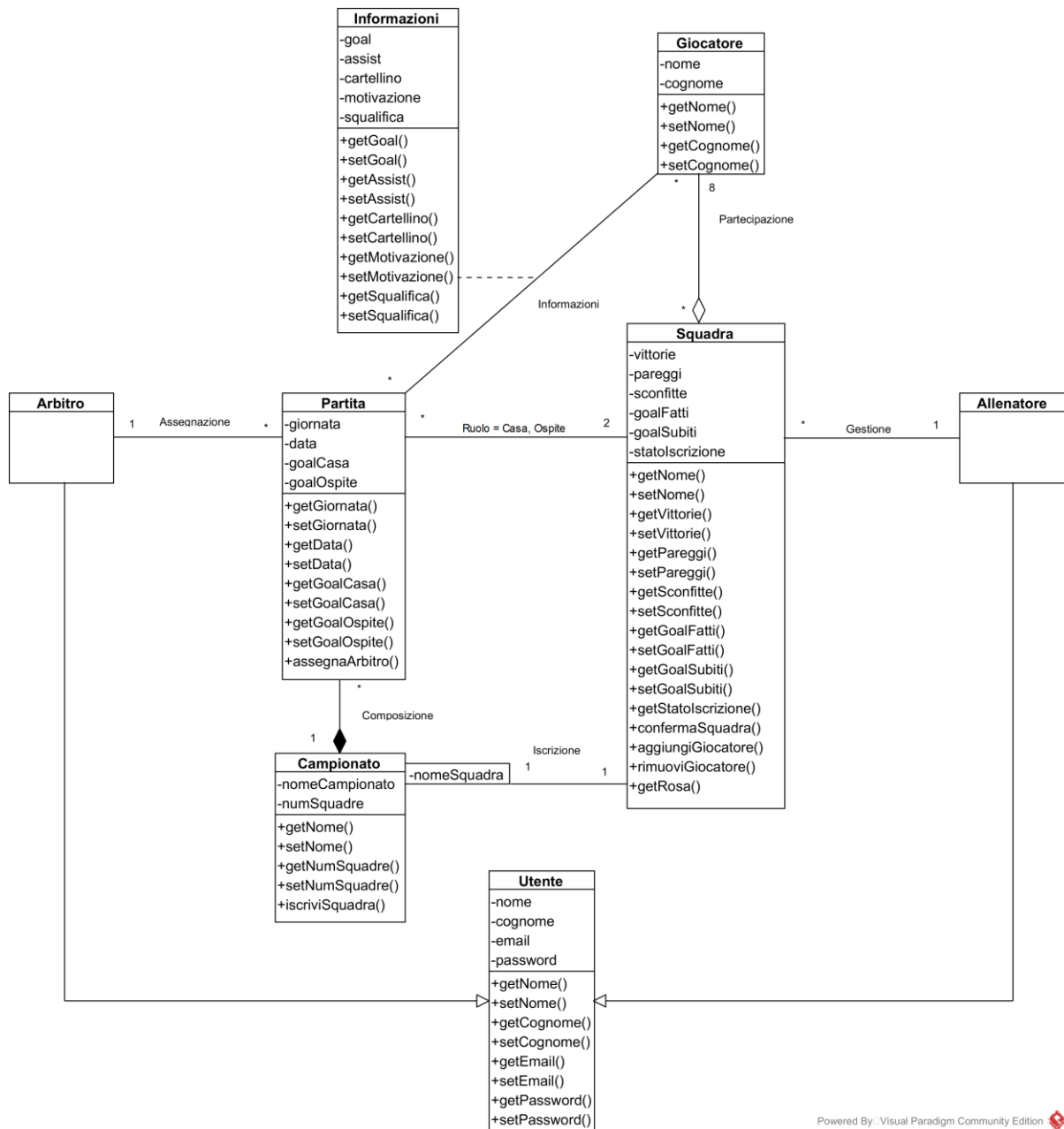
Use case diagram



Powered By: Visual Paradigm Community Edition

3.4.3. Object model

Class Diagram



3.4.4. Dynamic model

Sequence Diagrams

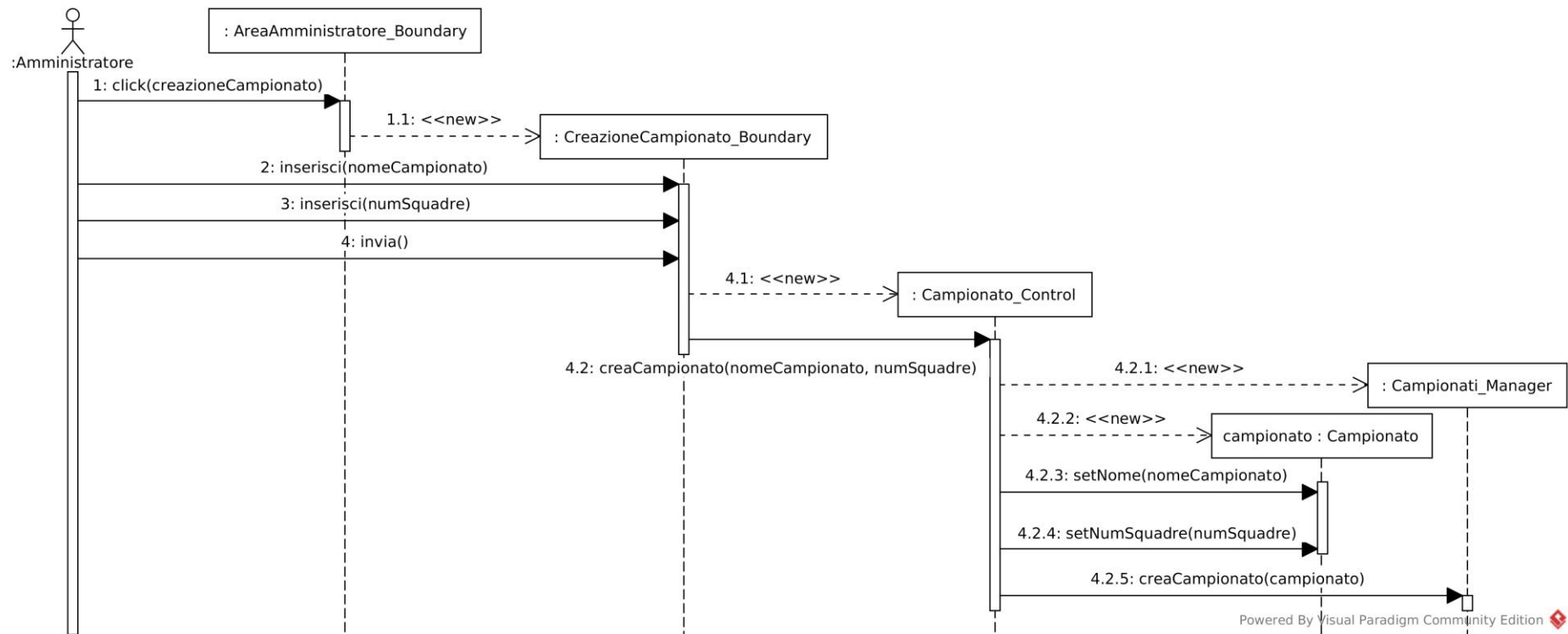


figura 3.4.4.1

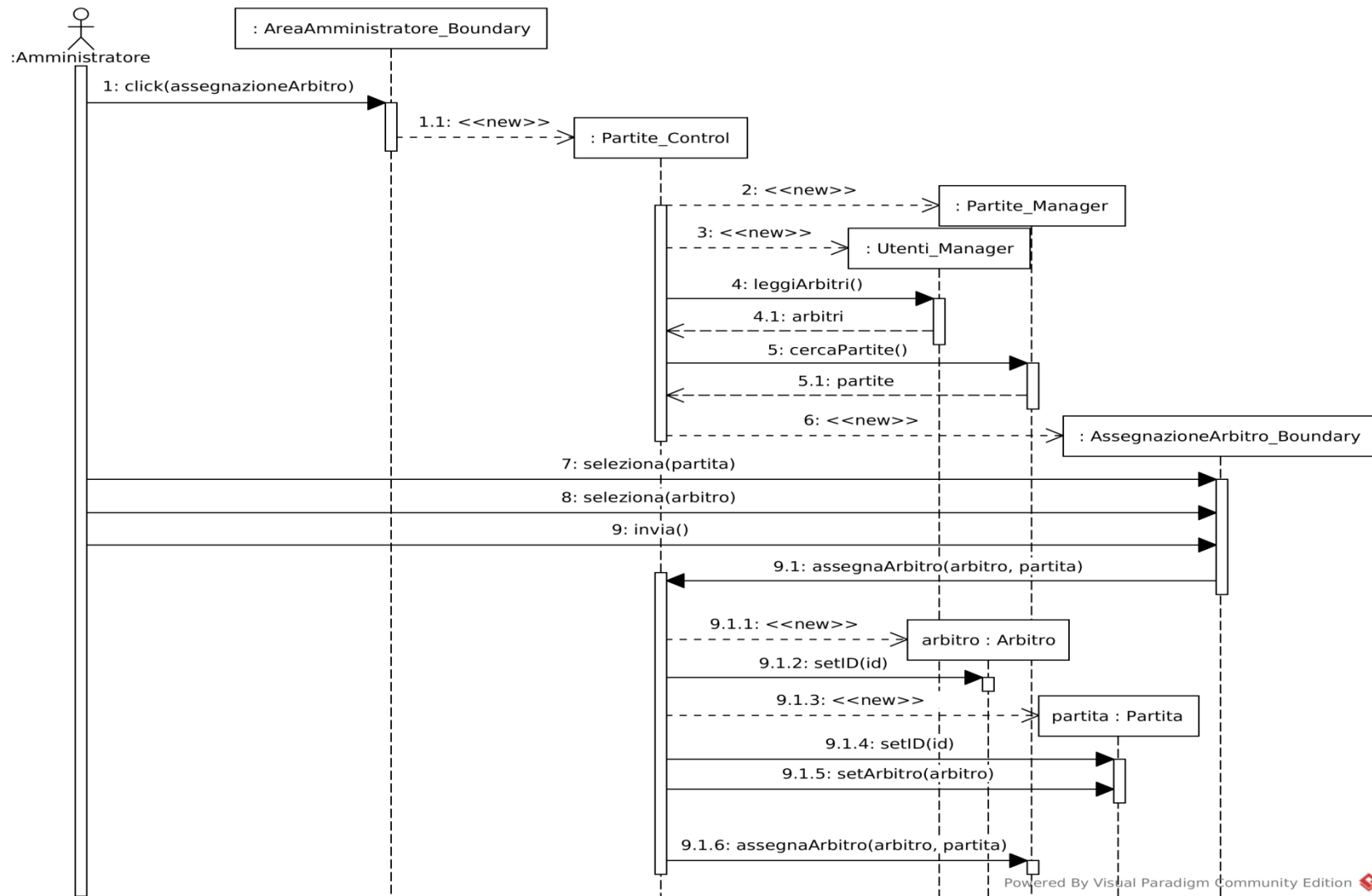


figura 3.4.4.2

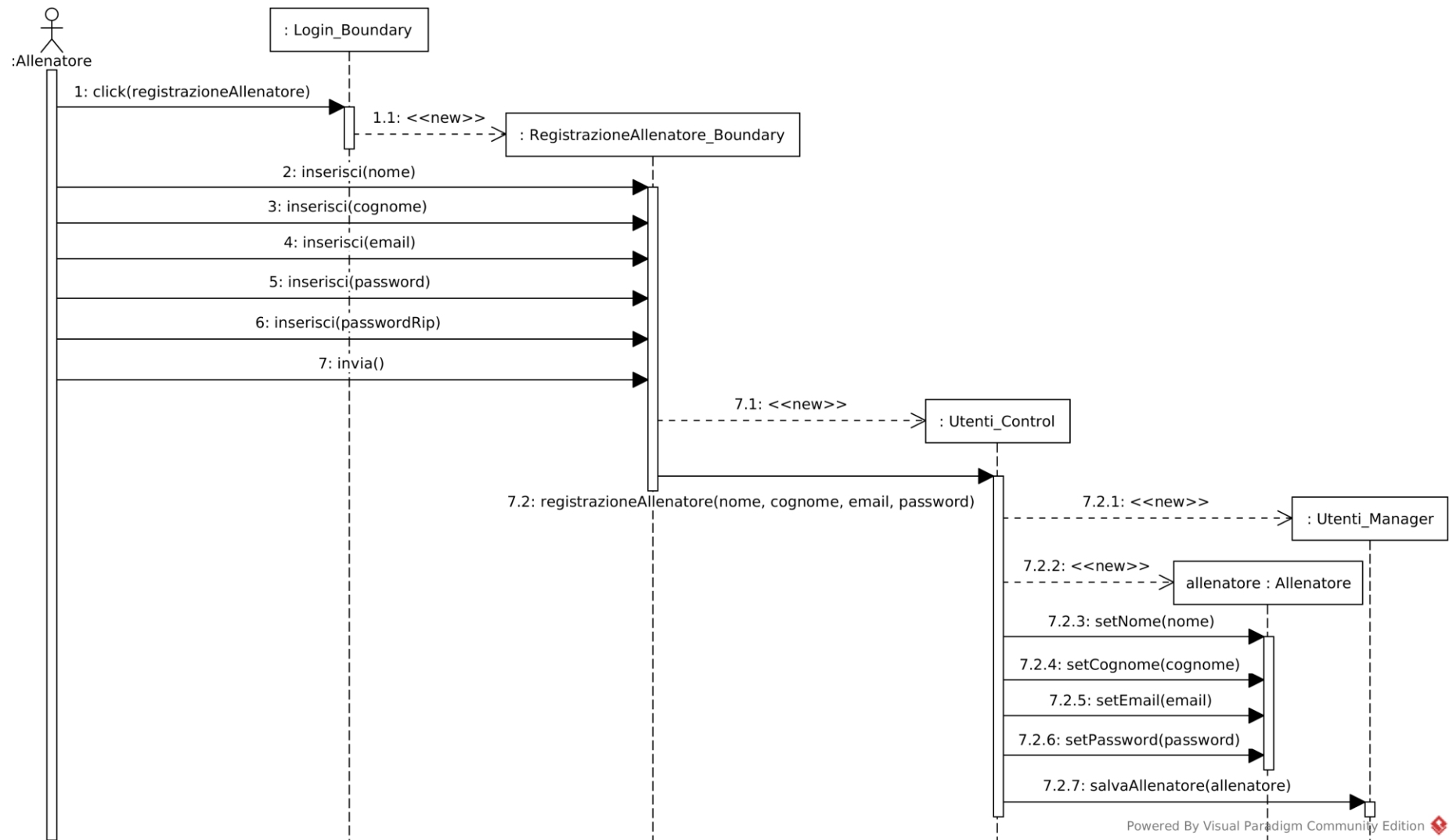


figura 3.4.4.3

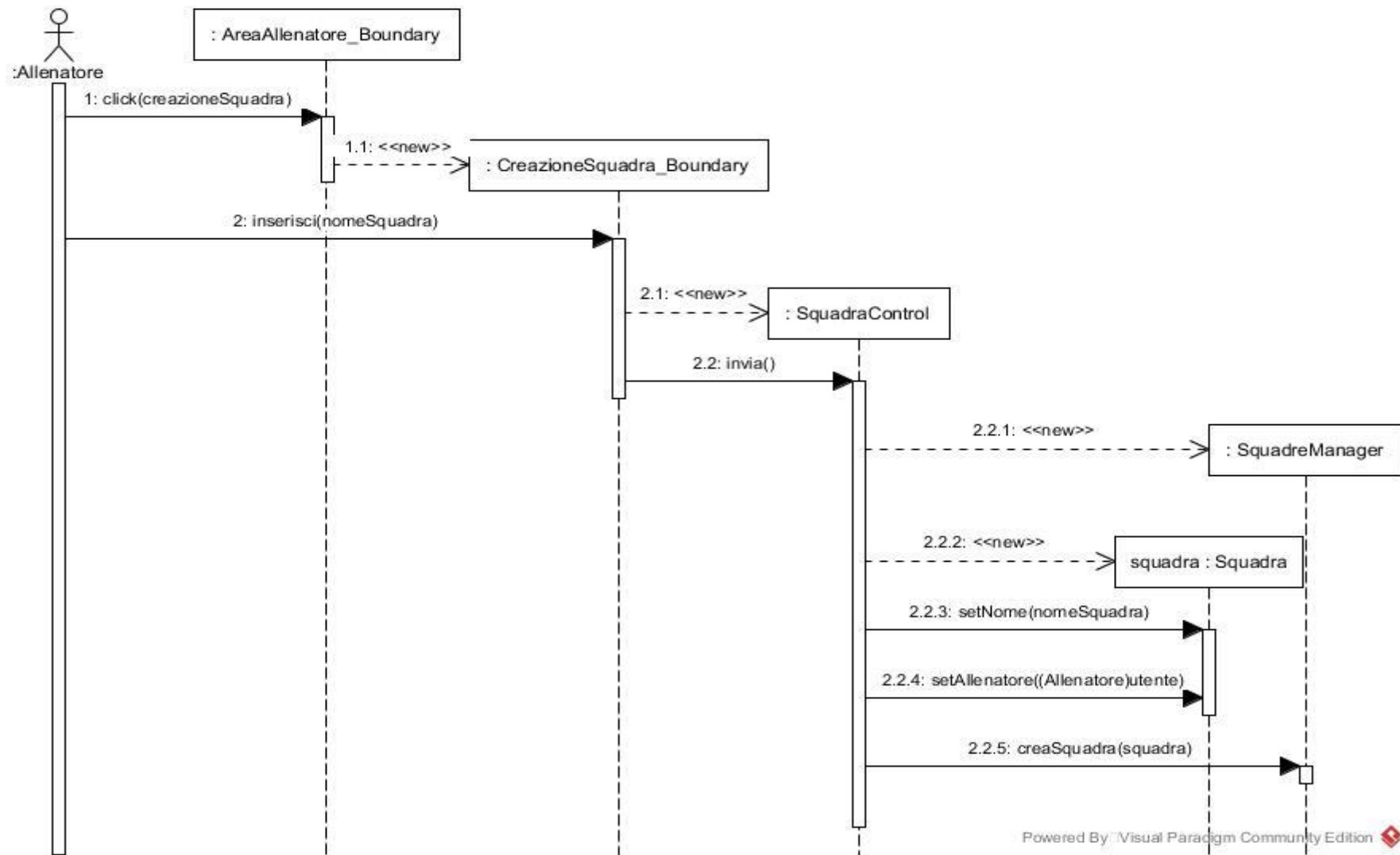


figura 3.4.4.4

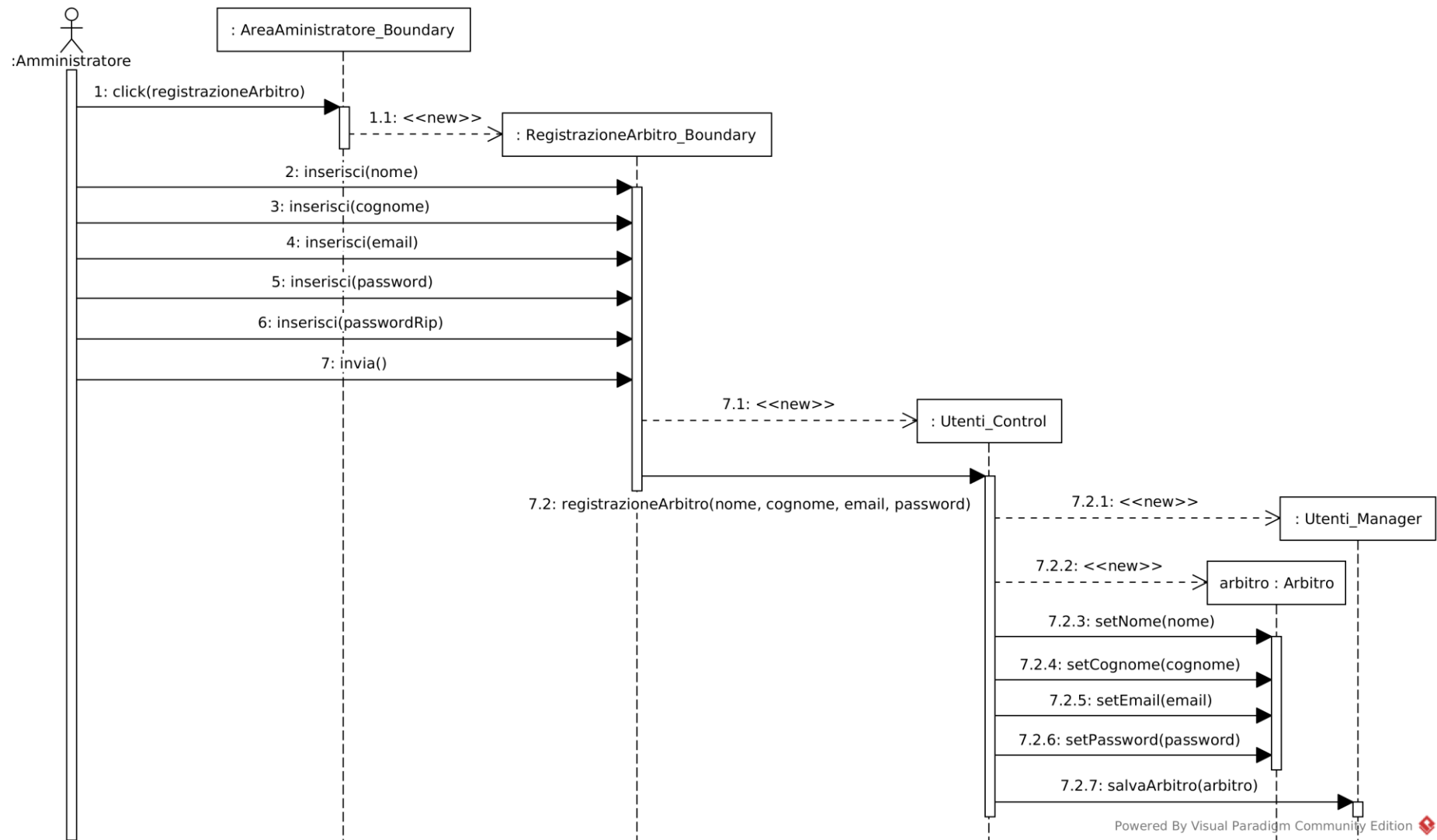


figura 3.4.4.5

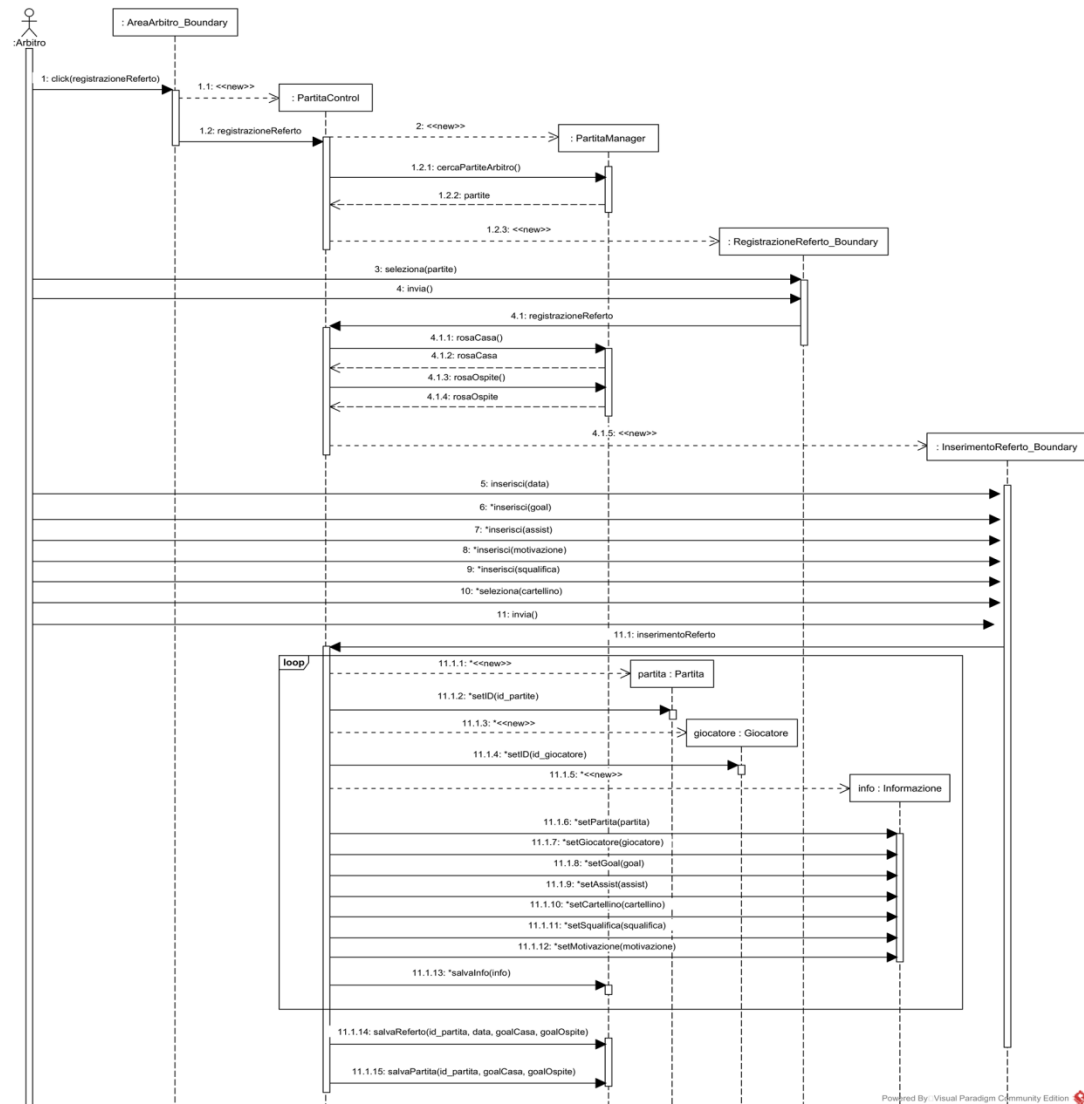


figura 3.4.4.6

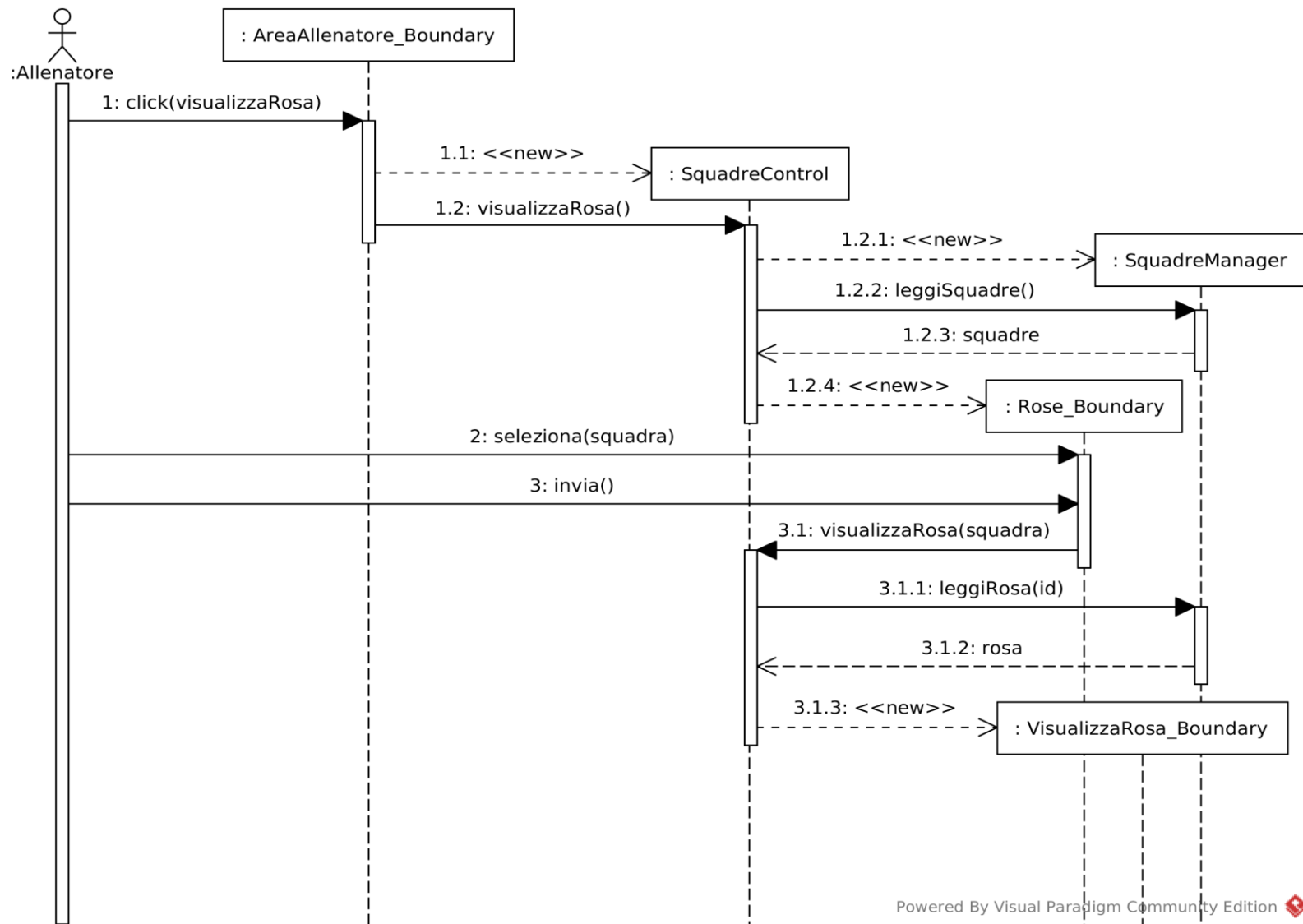
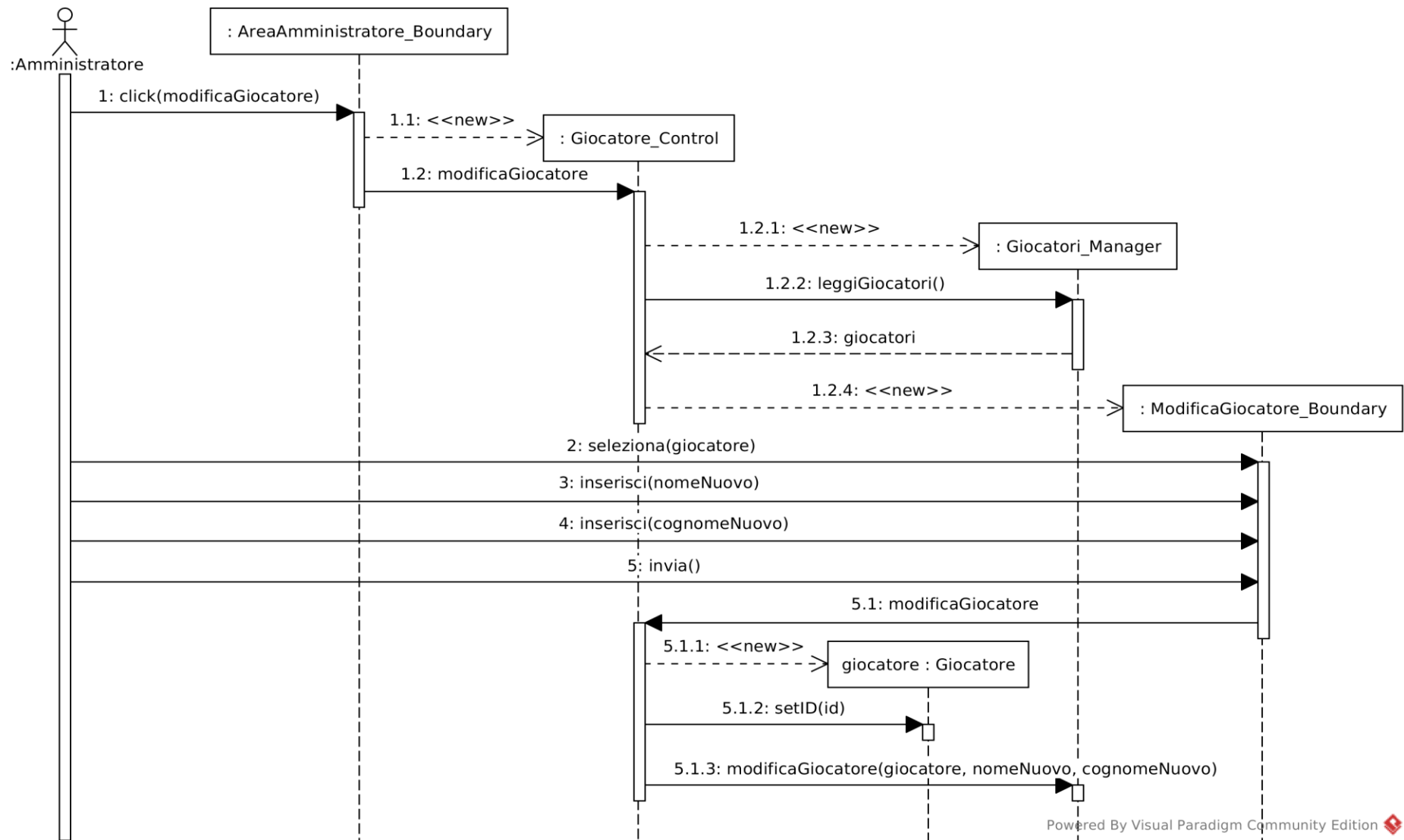


figura 3.4.4.7



Powered By Visual Paradigm Community Edition

figura 3.4.4.8

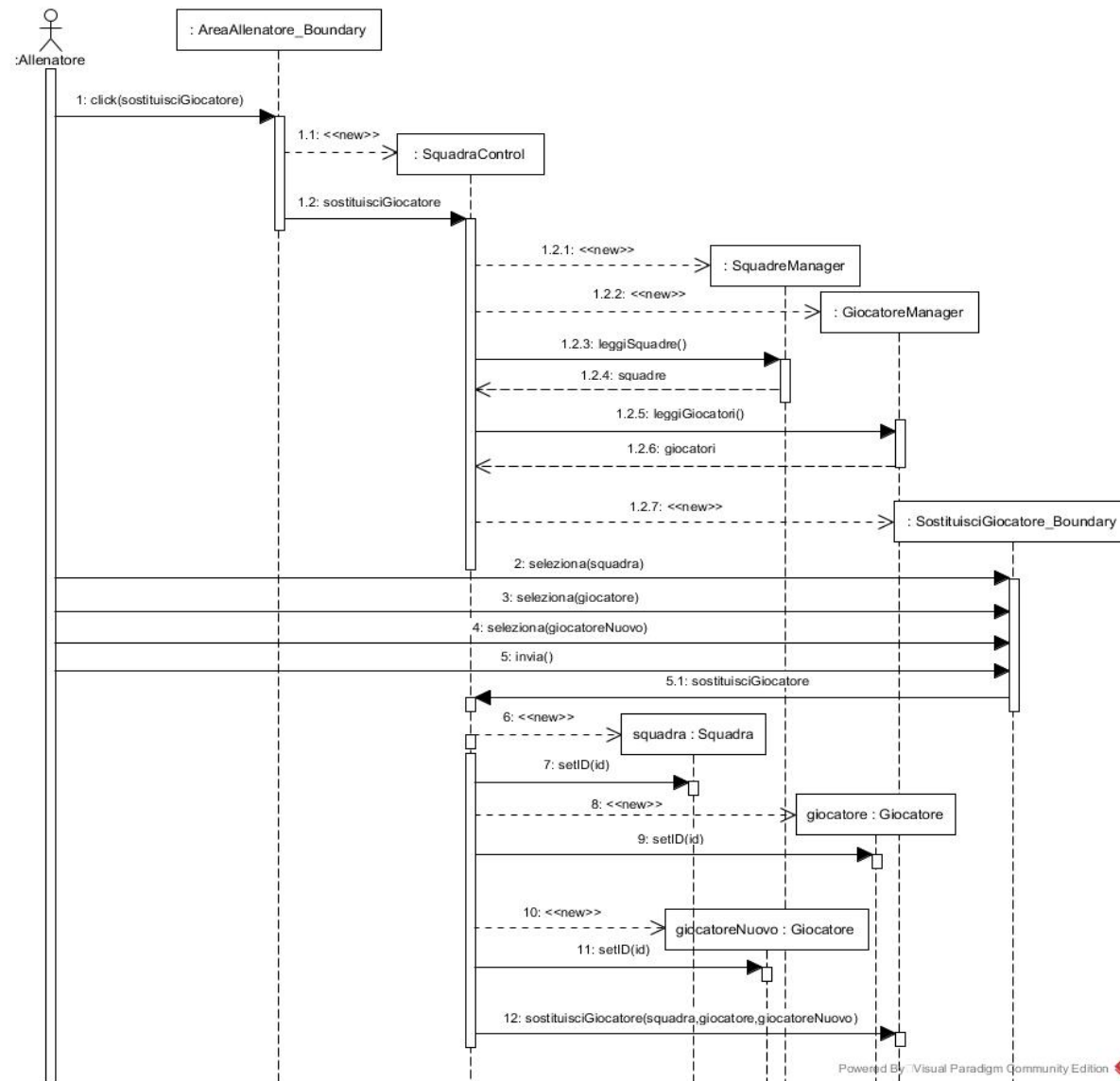


figura 3.4.4.9

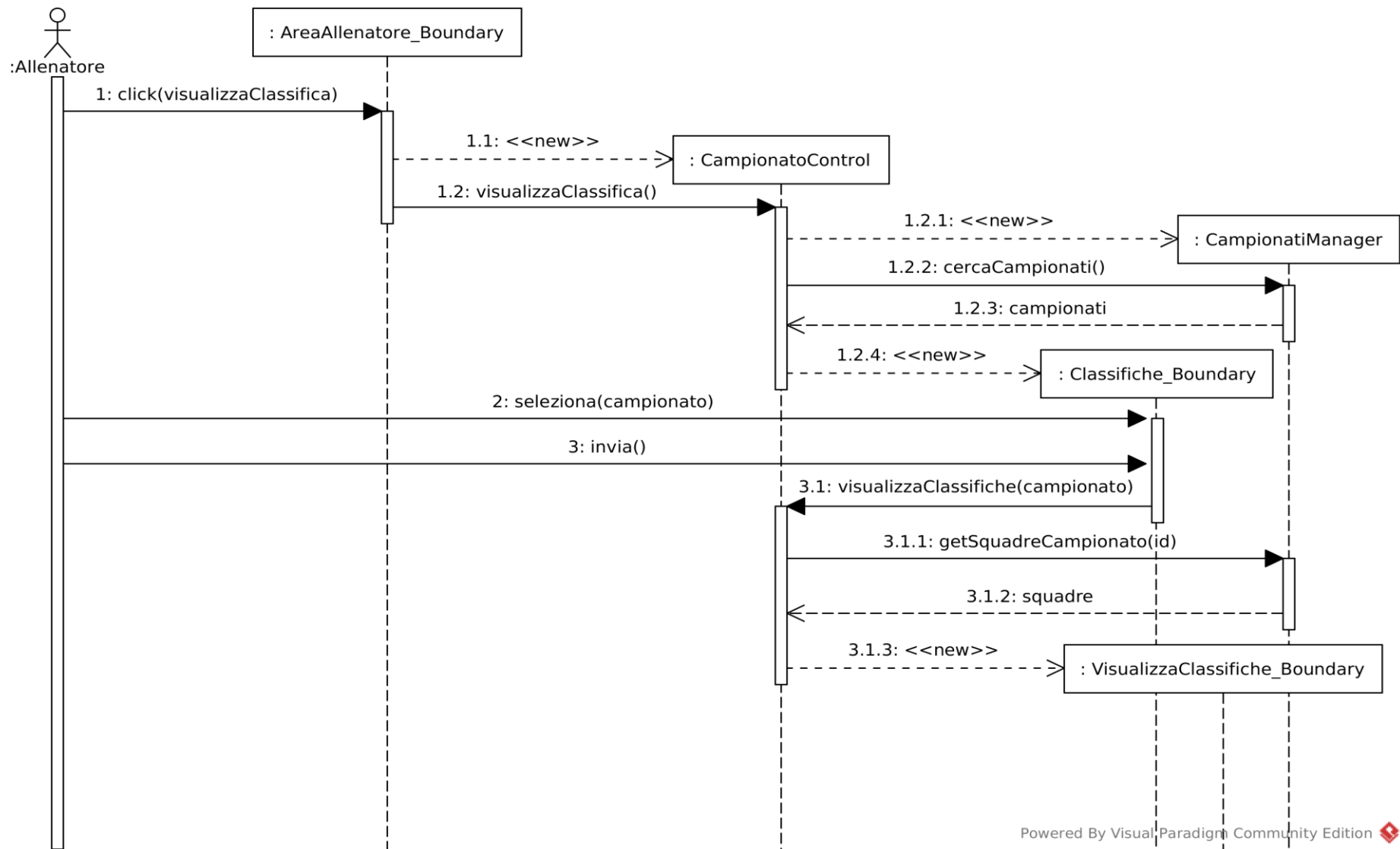


figura 3.4.4.10

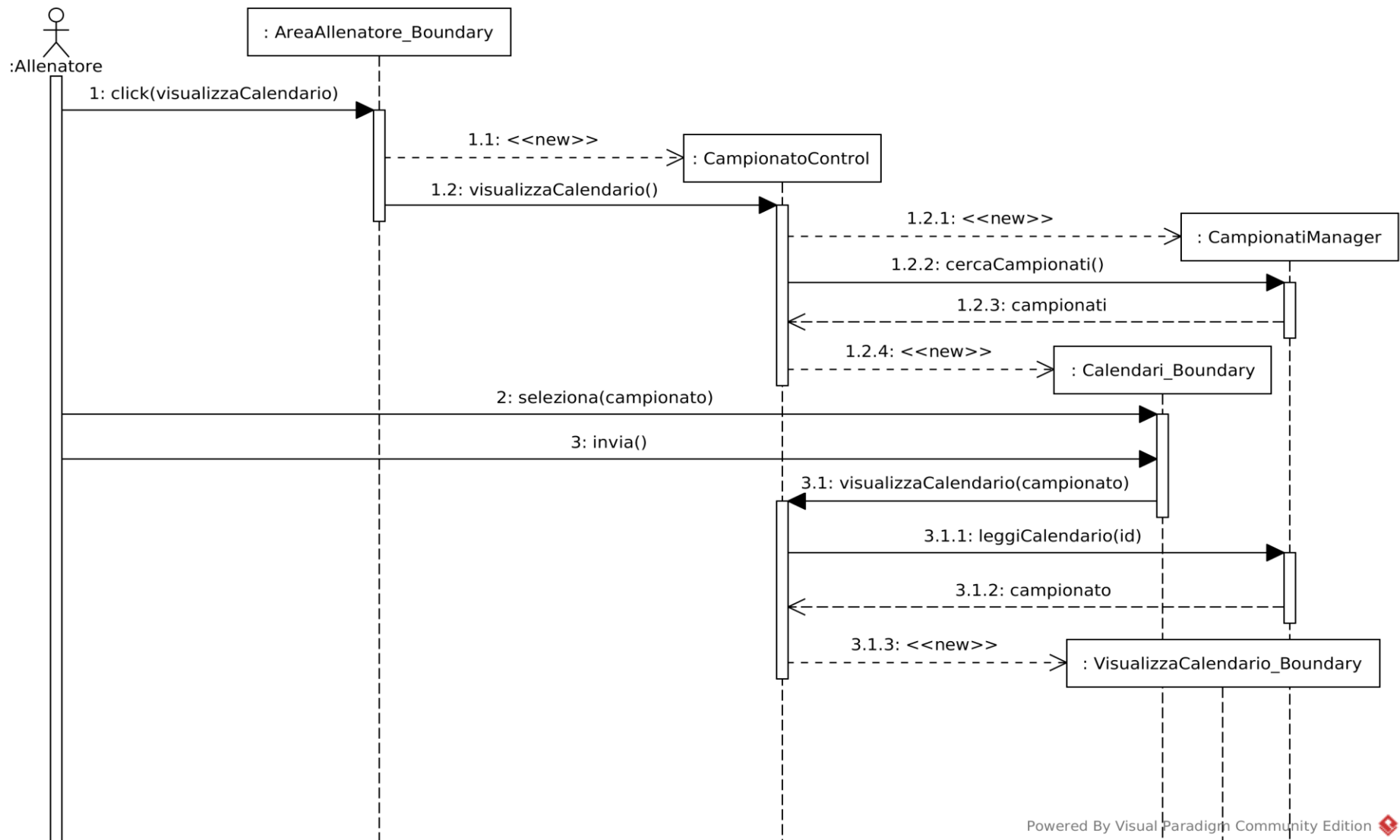


figura 3.4.4.11

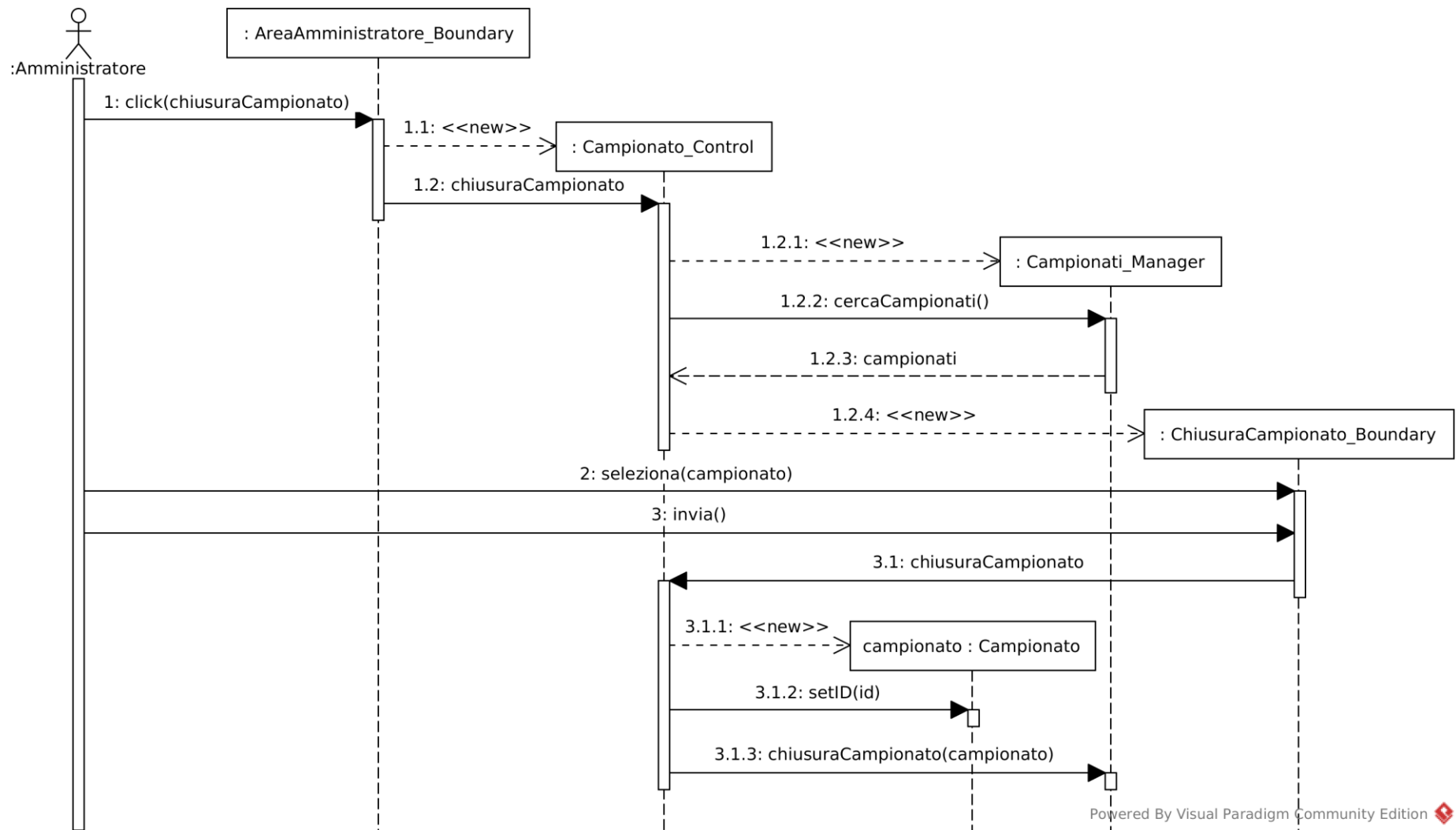
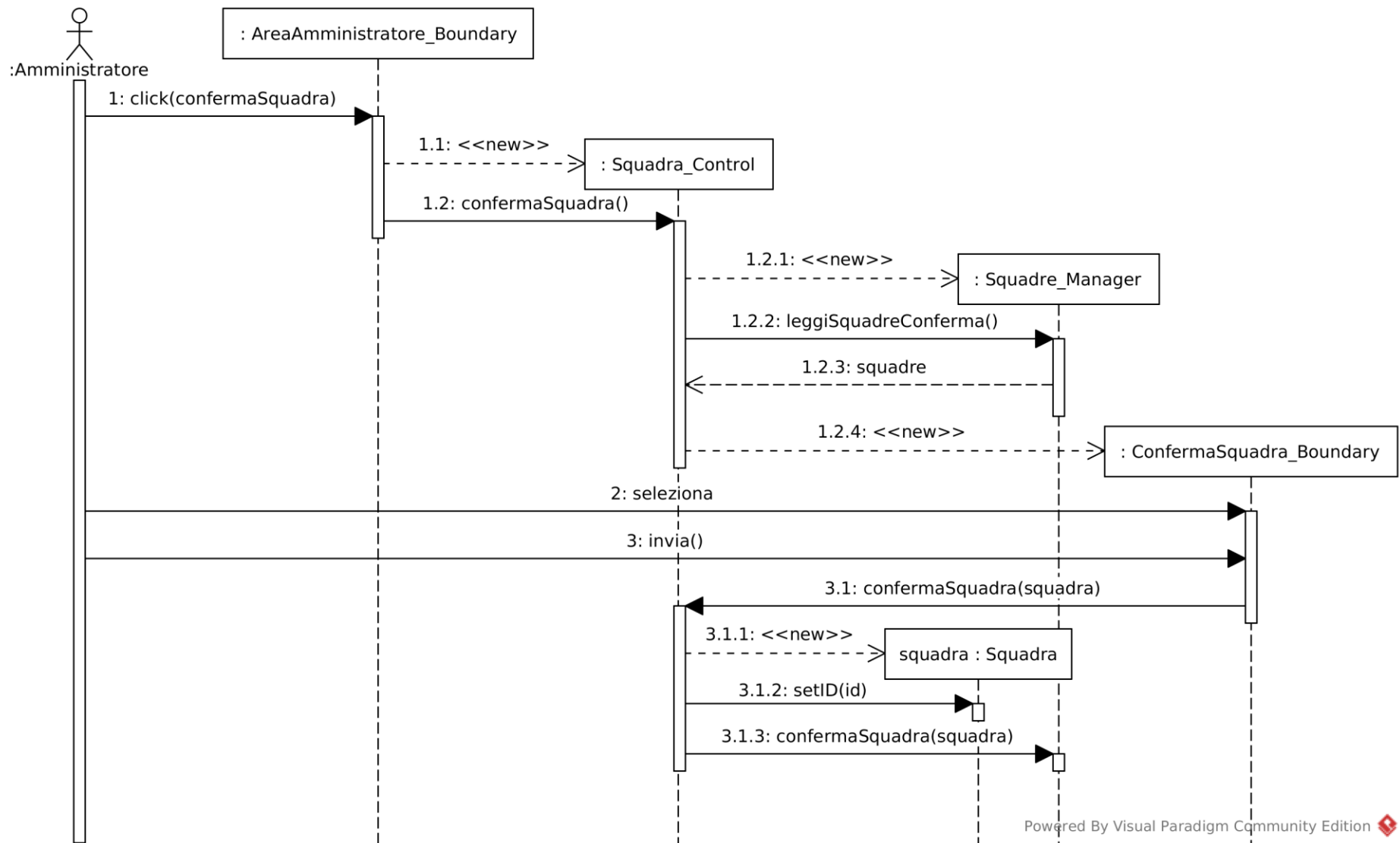
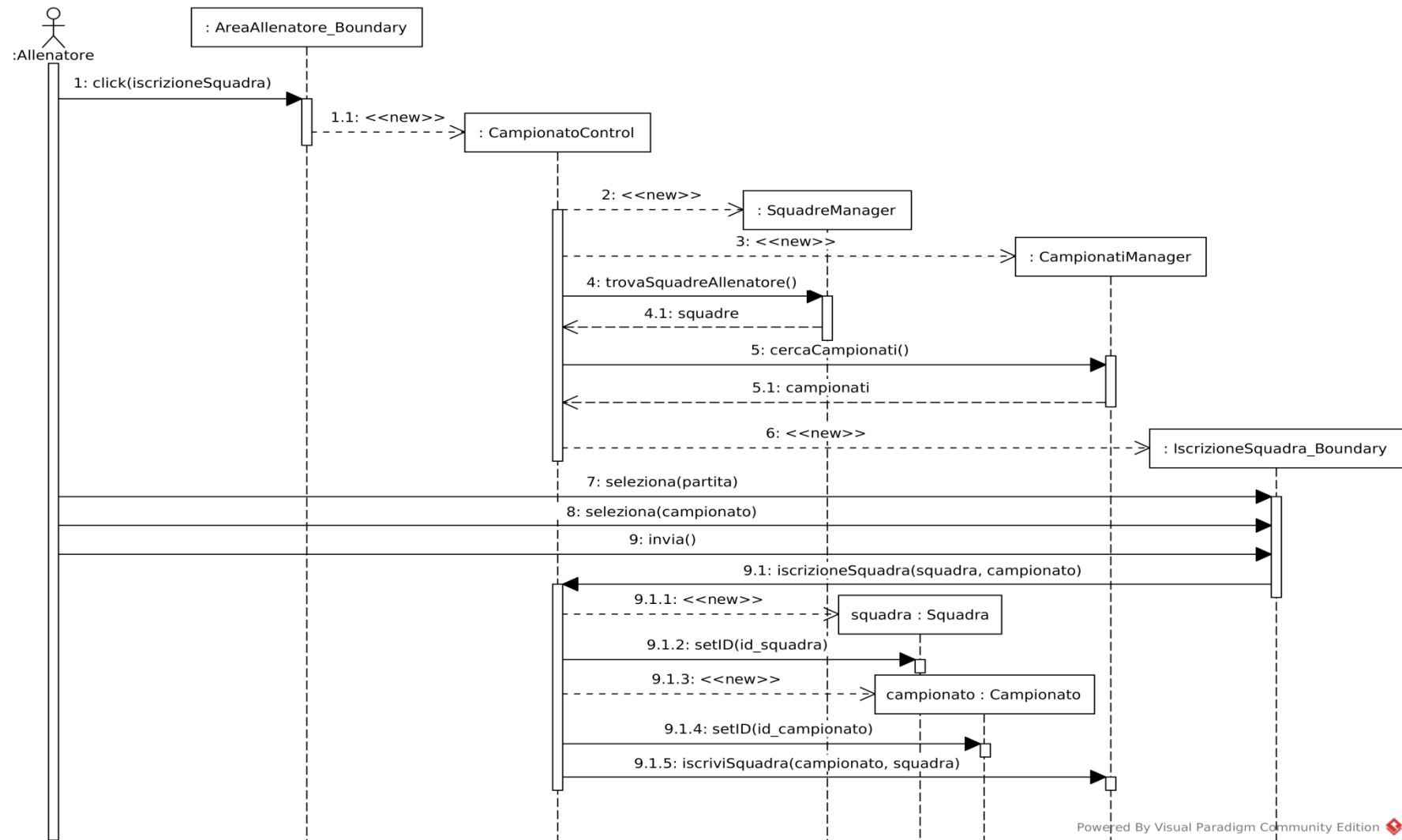


figura 3.4.4.12



Powered By Visual Paradigm Community Edition

figura 3.4.4.13



Powered By Visual Paradigm Community Edition

figura 3.4.4.14

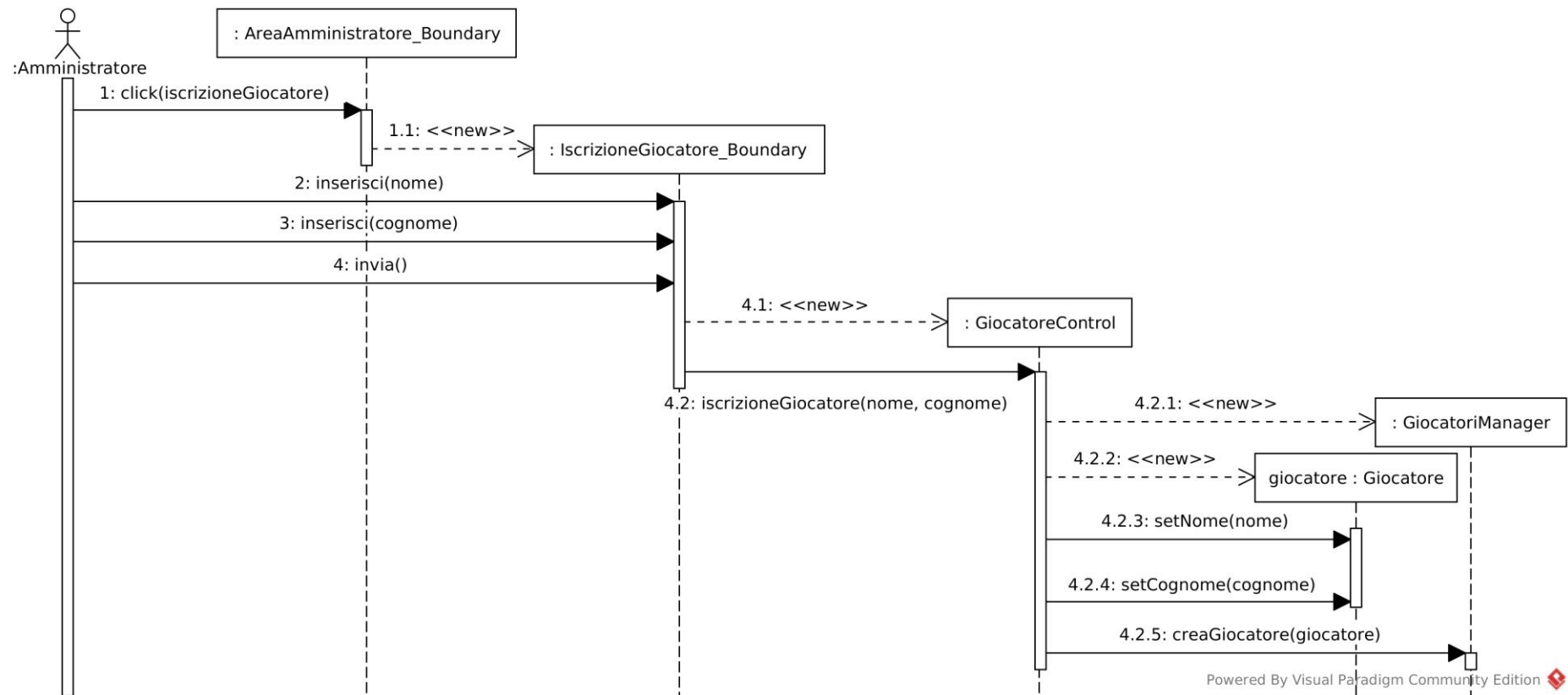
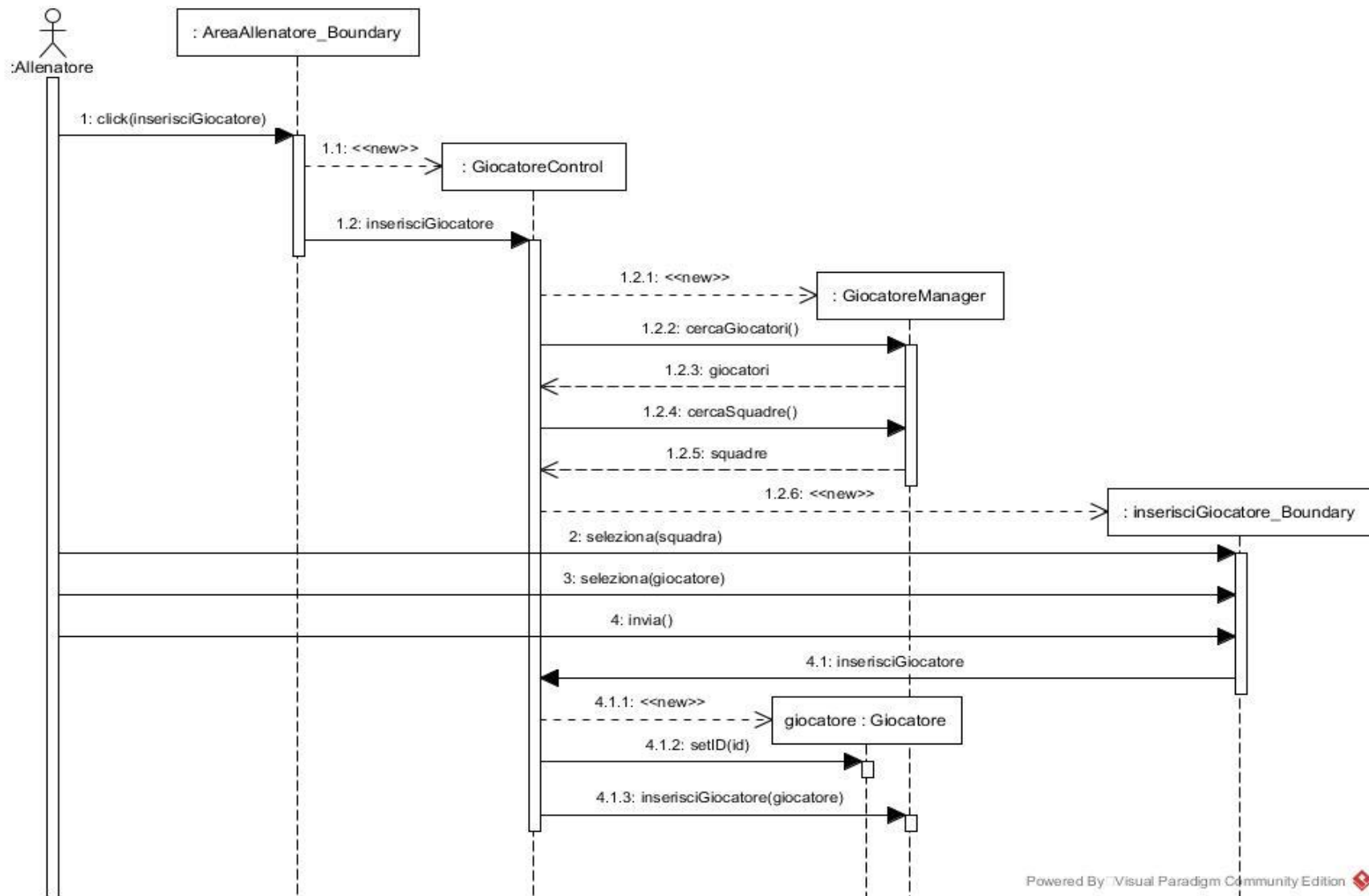


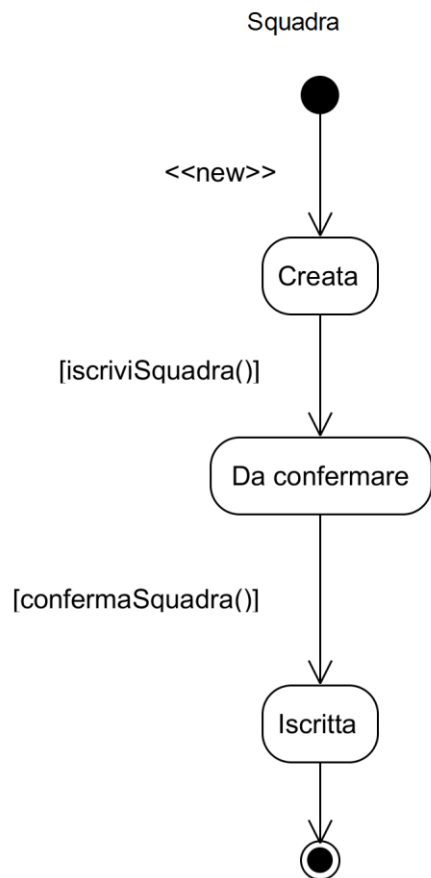
figura 3.4.4.15



Powered By: Visual Paradigm Community Edition

figura 3.4.4.16

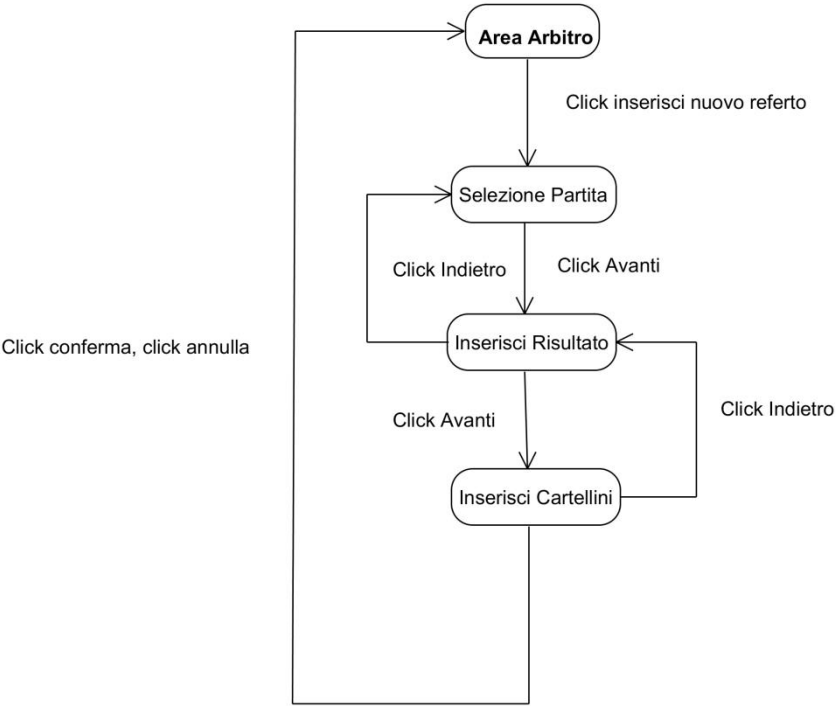
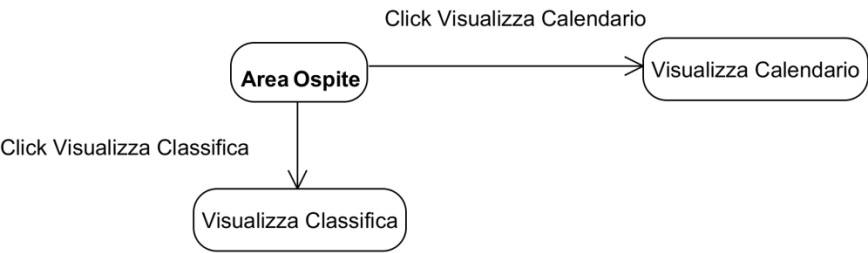
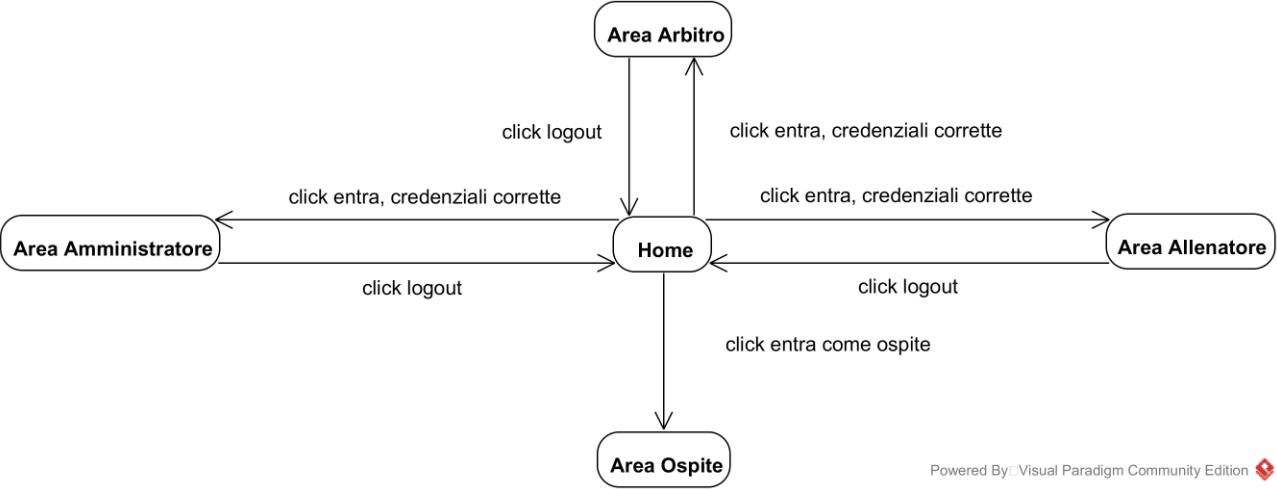
Activity Diagram

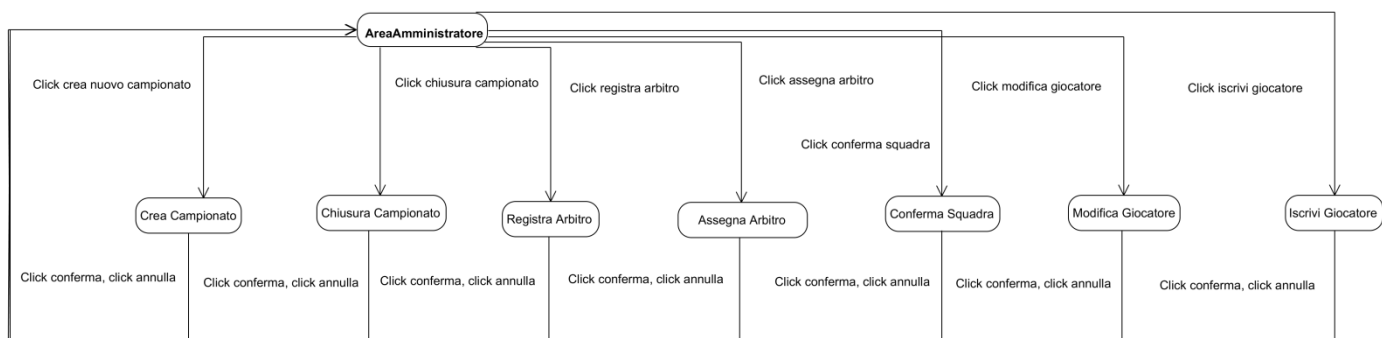
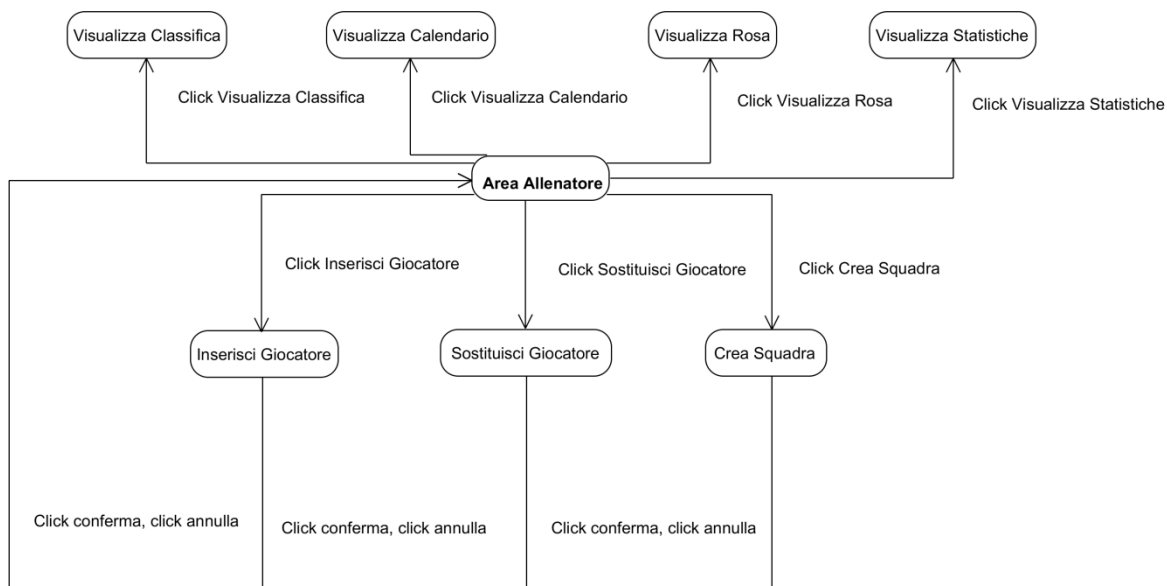


Creata: statolscrizione = NESSUNA_ISCRIZIONE
Da confermare: statolscrizione = ATTESA_CONFERMA
Iscritta: statolscrizione = SQUADRA_ISCRITTA

3.4.5. User interface navigational paths and screen mock-ups

Navigational paths





Mock-ups

Football League Manager

Accedi a FLM

E-mail

Password

Accedi

Accedi come ospite

Registra allenatore

figura 3.4.5.1

Creazione Campionato

Nome campionato

Squadre partecipanti

Crea campionato

figura 3.4.5.2

Assegnazione Arbitro

Nocerino Christian ▼

Senior B | Juventus - Milan ▼

Conferma scelta

figura 3.4.5.3

Registrazione Allenatore

Nome

Cognome

E-mail

Password

Ripeti password

Registrati

figura 3.4.5.4

Creazione Squadra

Nome squadra

Crea squadra

This screenshot shows a web form titled "Creazione Squadra". It features a single text input field labeled "Nome squadra" and a blue button labeled "Crea squadra". The form is centered on a white background, which is itself set against a larger green background.

figura 3.4.5.5

Registrazione Arbitro

Nome

Cognome

E-mail

Password

Ripeti password

Registrati

This screenshot shows a web form titled "Registrazione Arbitro". It contains five text input fields: "Nome", "Cognome", "E-mail" (with an email icon), "Password" (with a password icon), and "Ripeti password". A blue button labeled "Registrati" is positioned below the fields. The form is centered on a white background, which is itself set against a larger green background.

figura 3.4.5.6

Registrazione Referto

Senior B | Juventus - Napoli

Conferma scelta

figura 3.4.5.7.1

Registrazione Referto

Data partita

D'Amico Giuseppe

Manzo Domenico

Esposito Dante

Pepe Francesco

Bisogno Felice

De Rossi Gennaro

Di Michele Marco

Alighieri Matteo

Rossi Paolo

Negro Giuseppe

Verdi Antonio

figura 3.4.5.7.2

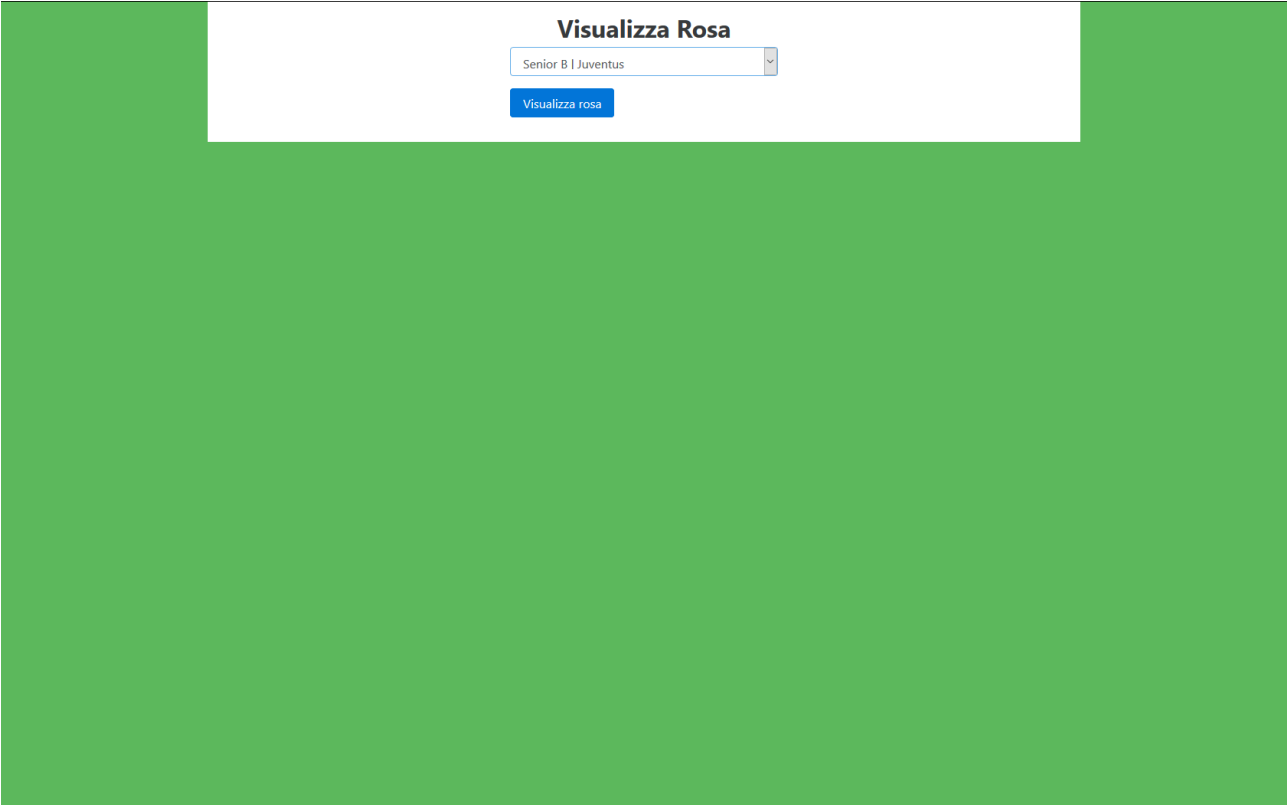


figura 3.4.5.8.1

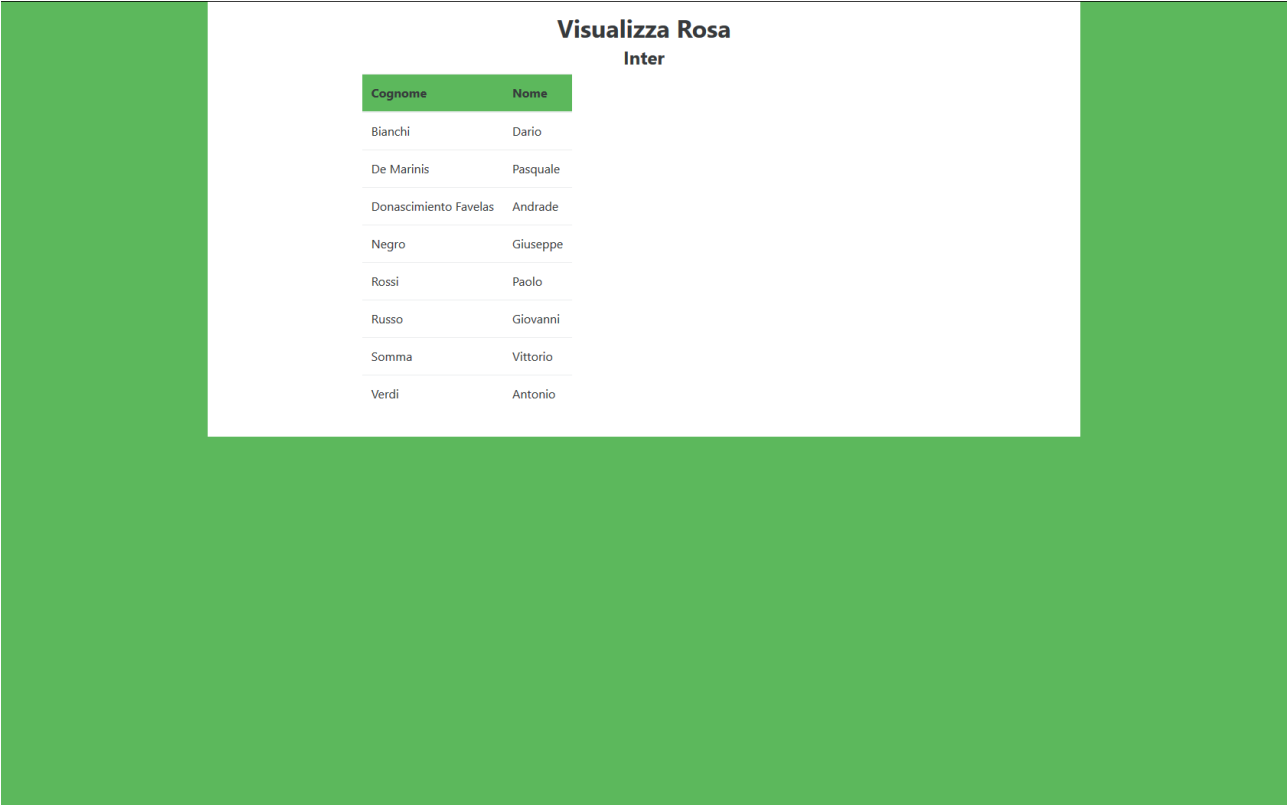
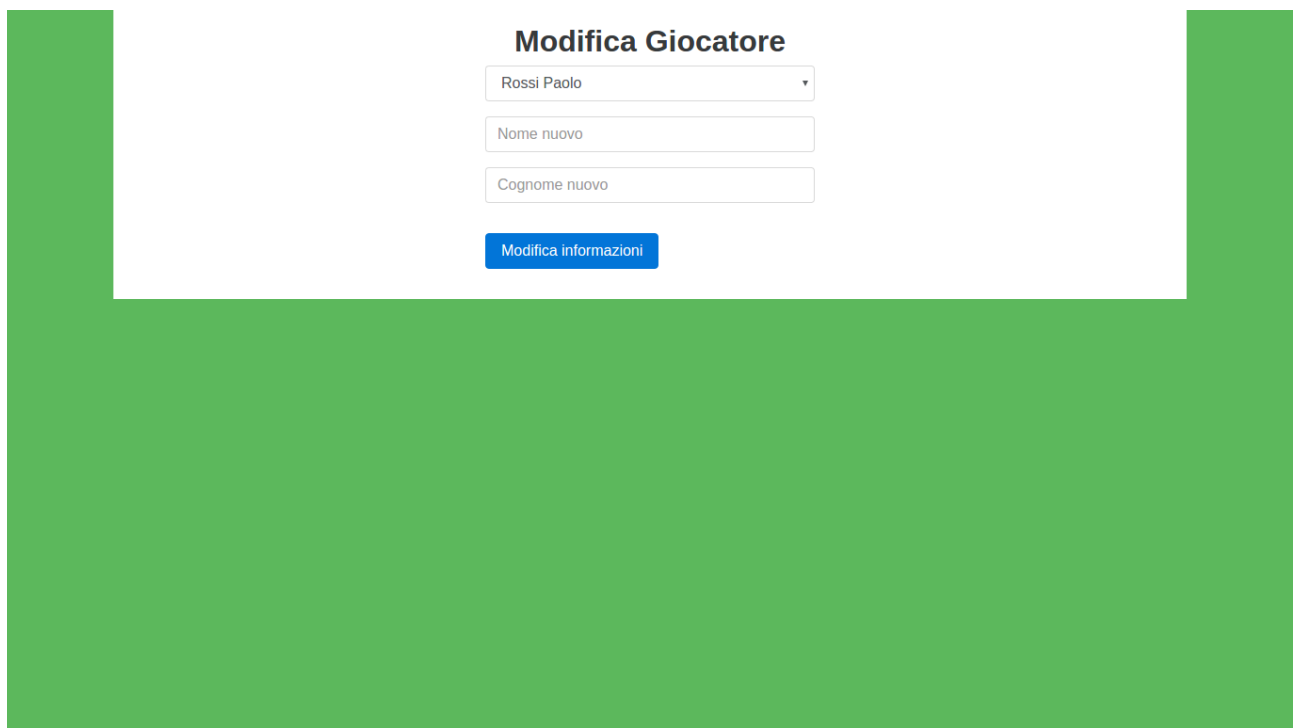


figura 3.4.5.8.2



Modifica Giocatore

Rossi Paolo ▼

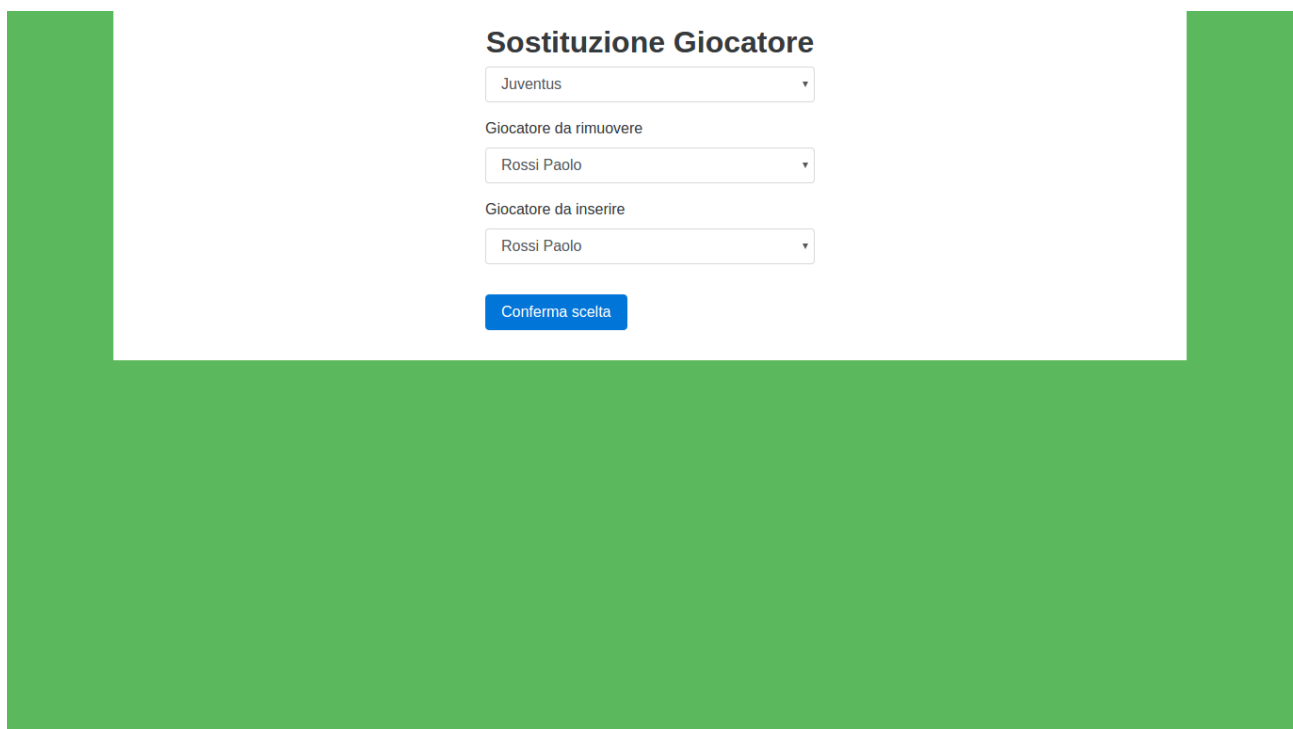
Nome nuovo

Cognome nuovo

Modifica informazioni

This screenshot shows a web form titled "Modifica Giocatore". It features a dropdown menu with "Rossi Paolo" selected, two text input fields labeled "Nome nuovo" and "Cognome nuovo", and a blue button labeled "Modifica informazioni". The form is set against a light gray background with a large green rectangular area below it.

figura 3.4.5.9



Sostituzione Giocatore

Juventus ▼

Giocatore da rimuovere

Rossi Paolo ▼

Giocatore da inserire

Rossi Paolo ▼

Conferma scelta

This screenshot shows a web form titled "Sostituzione Giocatore". It includes a dropdown menu with "Juventus" selected, followed by two sections: "Giocatore da rimuovere" with a dropdown showing "Rossi Paolo", and "Giocatore da inserire" with a dropdown also showing "Rossi Paolo". A blue button labeled "Conferma scelta" is at the bottom. The form is set against a light gray background with a large green rectangular area below it.

figura 3.4.5.10.1

Inserimento Giocatore

Juventus

▼

Rossi Paolo

▼

Conferma scelta

figura 3.4.5.10.2

Visualizza Classifica

Junior

▼

Visualizza classifica

figura 3.4.5.11.1

Visualizza Classifica

Classifica

Posizione	Nome Squadra	Partite giocate	Punti	Vittorie	Pareggi	Sconfitte	Goal fatti	Goal subiti
1	Roma	3	6	2	0	1	9	7
2	Juventus	3	5	1	2	0	8	5
3	Milan	3	4	1	1	1	5	5
4	Napoli	1	2	0	2	1	4	9

figura 3.4.5.11.2

Visualizza Calendario

Senior B

Visualizza calendario

figura 3.4.5.12.1

Visualizza Calendario				
Senior B				
Giornata	Data	Squadra Casa	Squadra Ospite	Risultato
1	2017-01-15	Milan	Juventus	4-4
1	2017-01-23	Inter	Napoli	11-9
2	2017-01-18	Juventus	Inter	5-2
2	2017-01-20	Napoli	Milan	6-3
3	2017-01-19	Napoli	Juventus	2-5
3	2017-01-18	Milan	Inter	4-2

figura 3.4.5.12.2

Conferma Squadra

Inter

Conferma squadra

figura 3.4.5.13

Chiusura Campionato

Junior ▼

Conferma scelta

figura 3.4.5.14

Iscrizione Giocatore

Nome giocatore

Cognome giocatore

Iscrivi

figura 3.4.5.15


Iscrizione Squadra

Milan


Junior

Iscrizione squadra

figura 3.4.5.16



Football League Manager



Crea campionato

Crea un nuovo campionato e apre le iscrizioni.

Crea!

Chiudi campionato

Chiudi un campionato e assegna il premio al vincitore.

Chiudi!

Registra arbitro

Registra un nuovo arbitro al sistema.

Registra!

Assegna arbitro

Assegna un arbitro ad una partita.

Assegna!

Conferma squadra

Conferma l'iscrizione di una squadra.

Conferma!

Iscrivi giocatore

Iscrivi un giocatore al sistema

Iscrivi!

Modifica giocatore

Aggiorna le informazioni di un giocatore.

Modifica!

figura 3.4.5.17

45

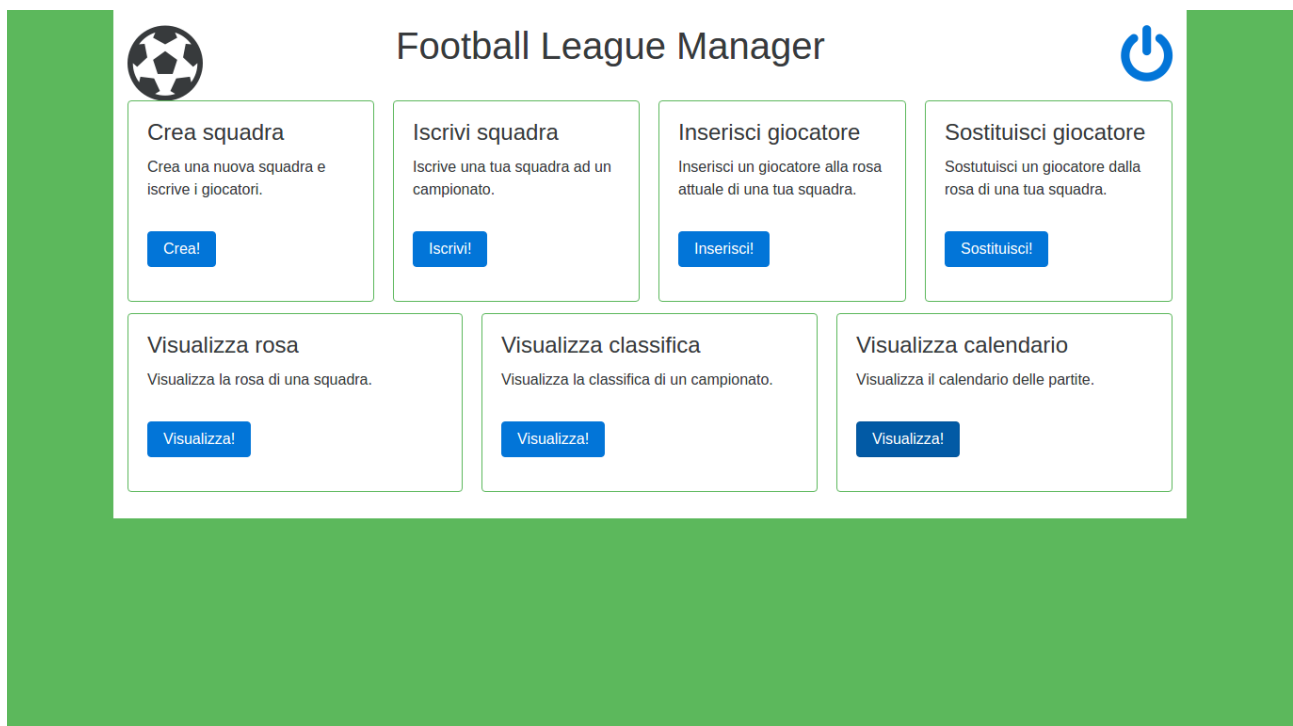


figura 3.4.5.18

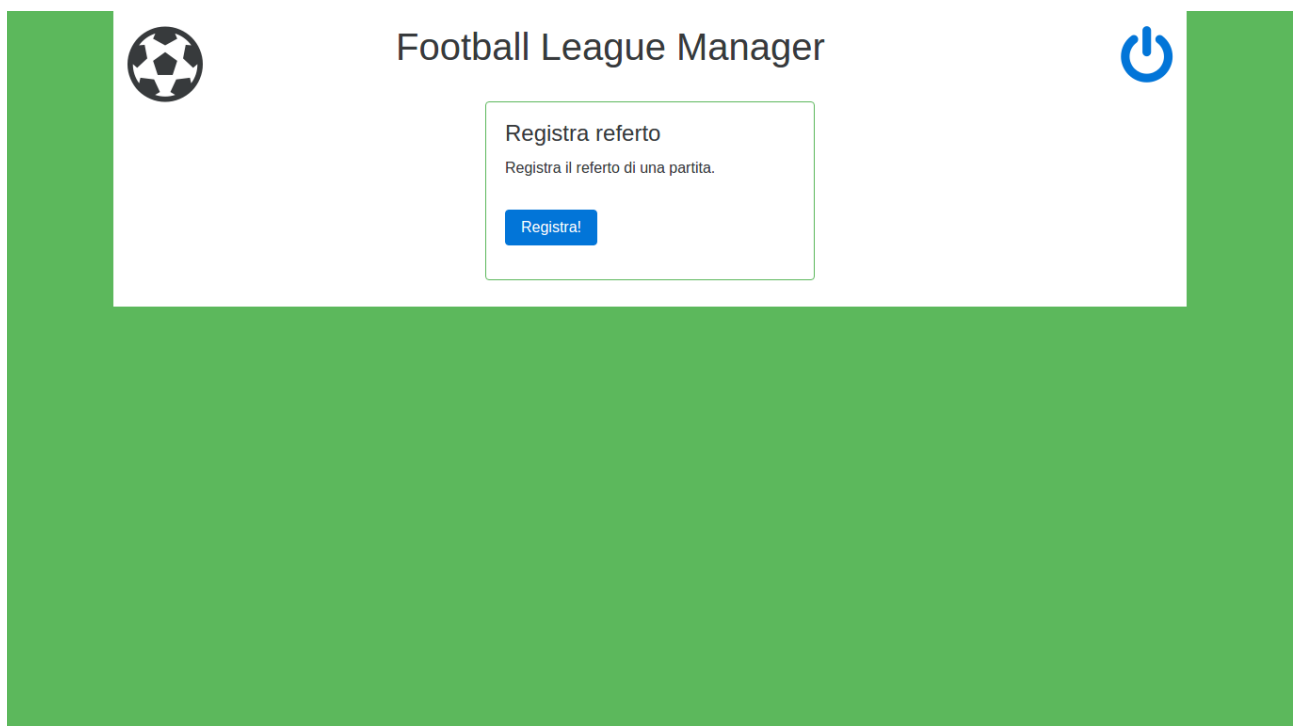


figura 3.4.5.19



Football League Manager

Visualizza rosa

Visualizza la rosa di una squadra.

Visualizza!

Visualizza classifica

Visualizza la classifica di un campionato.

Visualizza!

Visualizza calendario

Visualizza il calendario delle partite.

Visualizza!

figura 3.4.5.20

4. Glossary

A

- Attore: è un'entità fuori dal sistema che deve essere modellata e che interagisce con il sistema.

B

C

- Class Diagram: Rappresenta le classi del sistema con i loro attributi e operazioni. Inoltre mostra le relazioni tra le classi (associazioni, relazioni e gerarchie di specializzazione/generalizzazione); può essere utilizzato a diversi livelli di dettaglio (nell'analisi e nel design del sistema).
- Entry condition: E' un vincolo che deve essere soddisfatto quando viene invocata per la prima volta un'operazione.
- Exit condition: E' un vincolo che deve essere soddisfatto quando termina l'operazione.

D

- Diagram: Rappresentazione grafica di una collezione di elementi del modello. UML supporta i seguenti diagrammi: dei casi d'uso, di sequenza, di collaborazione, di attività, di stato, delle classi, degli oggetti, dei comportamenti e di dispiegamento.

E

- Extend: Relazione tra un caso d'uso estendente e uno base, che specifica come il comportamento definito dallo use case estendente è incorporato nel comportamento del caso d'uso base. Extends viene usato anche per definire dei casi eccezionali di un caso d'uso.

F

- FLM: Football League Manager, sistema che permette la creazione e la gestione di un campionato.

G

H

I

- Include: E' un tipo di relazione che serve a fattorizzare una parte del comportamento di un caso d'uso e metterlo in un altro caso d'uso.

L

- Login: La login è un'informazione che serve per accedere ad un sistema, è necessaria ma non sufficiente; ad essa è sempre associata ad una password che è rappresentata da una stringa alfanumerica.
- Logout: Rappresenta l'uscita dal sistema corrente che impone come prossima operazione un nuovo accesso al sistema con un account differente.

M

- Modello: E' un concetto astratto che descrive un sottoinsieme del sistema.
- Mock-ups: Produzione completa dell'interfaccia utente.

N

O

- Object Model: Modello di tipo UML che chiarisce le relazioni tra vari oggetti identificati per lo sviluppo del sistema software. Comprende anche le molteplicità, i versi e gli attributi delle relazioni tra gli oggetti identificati in fase di progettazione.

P

Q

R

- Requisiti non funzionali: Requisito che specifica proprietà richieste al sistema, come vincoli ambientali e di sviluppo, prestazioni, dipendenze dalla piattaforma, di manutenibilità, estensibilità, sicurezza e affidabilità. Requisito che sancisce vincoli di carattere fisico relativi ai requisiti funzionali.
- Requirements analysis: Insieme di dati utili a chiarire al cliente tutto ciò che concerne le decisioni iniziali, i requisiti, gli obiettivi e i modelli relativi alla progettazione del sistema software da sviluppare.

S

- Scenario: Un'istanza di un caso d'uso.
- Sequence Diagram: È utilizzato per definire la logica di uno scenario (specifica sequenza di eventi) di un caso d'uso. È uno dei principali input per l'implementazione dello scenario; mostra gli oggetti coinvolti specificando la sequenza temporale dei messaggi che gli oggetti si scambiano.
- Statechart Diagram: È normalmente utilizzato per modellare il ciclo di vita degli oggetti di una singola classe; mostra gli eventi che causano la transizione da uno stato all'altro, le azioni eseguite a fronte di un determinato evento.

T

U

- UML: È uno standard usato per modellare software orientato agli oggetti.
- Use Case: Rappresenta un possibile modo di utilizzo del sistema e descrive le interazioni tra esso e gli attori.
- Use Case Diagram: Diagramma che serve a descrivere il comportamento funzionale del sistema da come è percepito dall'utente.
- Utente non registrato: Colui che interagisce con il sistema per visualizzare le partite e le classifiche di un campionato.
- Utente Registrato: Colui che interagisce col sistema per usufruire dei servizi che il sistema offre.

V

Z