



Università Politecnica delle Marche

Corso di Laurea Triennale in Ingegneria Informatica
e dell'Automazione

Progetto per l'esame di Ingegneria del Software
Software per la gestione di un circolo tennis

*Ivan Pacenti
Nicola Picciafuoco
Giada Remedìa*

Sommario

Sommario	2
Descrizione in linguaggio naturale	4
1.1 Struttura del circolo tennis	4
1.2 Struttura del software	5
1.2.1 Gestione prenotazioni	5
1.2.2 Gestione utente	5
1.2.3 Gestione tesseramento	5
1.2.4 Gestione organizzazione	6
1.3 Glossario dei termini.....	6
Analisi dei requisiti	7
2.1 Requisiti del sistema	7
2.1.1 Requisiti funzionali	8
2.1.2 Requisiti non funzionali.....	10
2.2 Diagrammi dei casi d'uso.....	11
2.2.1 Gestione Prenotazione	11
2.2.2 Gestione partita	14
2.2.3 Gestione Utenti	17
2.2.4 Gestione organizzazione	19
2.3 Matrice di mapping dei requisiti	21
Diagrammi di analisi.....	22
3.1 Diagramma delle classi di analisi	23
3.2 Diagrammi di sequenza.....	24
3.2.1 backup	24
3.2.2 CUDPartita	25
3.2.3 CUDPrenotazione	26
3.2.4 Tesseramento	27
3.2.5 InserisciPunteggio	28
3.2.6 RicercaPartita	29
3.3 Diagrammi di attività.....	30
3.3.1 Backup	30
3.3.2 CUDPartita	31
3.3.3 GestionediTessera	32
3.3.4 RicercaPartita	33
Diagrammi di progettazione	34
4.1 Classi di gestione.....	35
4.2 Classi Generali	36
4.3 Macchine a stati.....	37
4.3.1 Campo	37
4.3.2 Partita	38
4.3.3 Prenotazione	38
4.3.4 Tesseramento	39
4.4 Diagramma dei componenti	40

<i>Implementazione</i>	41
5.1 Diagramma di deployment	41
<i>Mockup</i>	42
6.1 Home	43
6.2 Login	44
6.3 Conferma login	45
6.4 Conferma logout	45
6.5 Registrazione	46
6.6 Conferma iscrizione	46
6.7 Gestione tesseramento	47
6.8 Conferma tesseramento	48
6.9 Gestione organizzazione	48
6.10 Visualizza lista soci	49
6.11 Visualizza lista utenti	49
6.12 Modifica dati registrazione	50
6.13 Gestione prenotazioni	50
6.14 Aggiungi partita	51
6.15 Conferma modifica iscrizione	51
6.16 Elimina prenotazione	52
<i>Unit testing</i>	53

Descrizione in linguaggio naturale

Il progetto proposto consiste nella realizzazione di un sistema informativo che si occupa della gestione di un circolo tennis.

Il particolare, il gestionale ricopre tre ambiti:

- Prenotazione dei campi
- Gestione delle statistiche
- Gestione dei soci

1.1 Struttura del circolo tennis

Il circolo dispone di due campi da tennis in terra rossa battuta, uno da tennis in erba sintetica e un campo da paddle. Un campo in terra battuta dispone di una copertura fissa. È possibile, quindi, giocare tutto l'anno al coperto. Gli altri due campi sono all'aperto durante la stagione estiva e al chiuso durante quella invernale. Il campo da paddle è all'aperto.

Tutti i campi sono provvisti di luci per la possibilità di giocare in notturna. Il circolo mette a disposizione, all'interno delle coperture, il riscaldamento per la stagione invernale.

Le prenotazioni avvengono tramite un applicativo. Per poter prenotare è necessaria la registrazione all'applicazione. Al momento della prenotazione è necessario scegliere il giorno, la data e il campo.

È possibile giocare con le racchette da paddle del circolo tennis gratuitamente.

Alla fine dell'ora i giocatori sono tenuti a sistemare il campo.

Il pagamento avviene in loco. Il custode si occupa di riscuotere la somma dovuta.

I campi da tennis sono disponibili 12 mesi all'anno dalle ore 8.00 alle ore 23:00.

È prevista la possibilità di tesserarsi al circolo tennis.

Esistono tre tipologie di tessere:

- tessera per giocare a tennis;
- tessera per giocare a paddle
- tessera per giocare sia a tennis che a paddle.

Il costo del campo varia dal numero di giocatori, tipologia di campo e presenza della tessera.

1.2 Struttura del software

Le interviste effettuate agli stakeholder hanno reso possibile l'individuazione degli ambiti che il software deve coprire. Questi requisiti possono essere suddivisi in quattro macrocategorie:

- Gestione prenotazioni
- Gestione utente
- Gestione tesseramento
- Gestione organizzazione

Per la salvaguardia dei dati degli utenti è necessario creare un servizio di backup.

1.2.1 Gestione prenotazioni

Per quanto riguarda la gestione prenotazioni, le funzionalità riguardano la gestione delle prenotazioni dei campi e la gestione delle partite.

Un utente può aggiungere ed eliminare una prenotazione scegliendo il giorno, l'orario ed il campo di gioco.

Un utente può inserire il punteggio di una partita, indicando numero di set vinti e persi (fino ad un massimo di tre) ed avversario. Può inoltre vedere la lista delle partite giocate ed i rispettivi punteggi. Un utente può visualizzare le statistiche, ovvero la percentuale di vittorie sul numero totale di partite; può quindi verificare qual è la sua percentuale di vittorie sulle partite totali.

L'amministratore può aggiungere ed eliminare una prenotazione ed una partita scegliendo l'utente, il giorno, l'orario ed il campo di gioco.

Il custode ed il giocatore possono prenotare i campi, aggiungere una partita propria con il relativo punteggio, visualizzare la lista delle partite disputate e visualizzare la percentuale.

1.2.2 Gestione utente

Per quanto riguarda la gestione utente, le funzionalità riguardano il login e la registrazione degli utenti. Un utente per poter utilizzare il software deve registrarsi e poi accedere. La registrazione può avvenire solo come giocatore.

Un utente è registrato con nome, cognome, nome utente, password, cellulare e data di nascita. Dopo aver effettuato il login è possibile modificare il proprio account.

Un giocatore tesserato ha la possibilità di mantenere il proprio tesseramento se decidesse di modificare il suo account.

1.2.3 Gestione tesseramento

Per quanto concerne la gestione tesseramento, la funzionalità richiesta è solo la gestione dei tesseramenti.

Un giocatore può tesserarsi, l'amministratore ed il custode non possono effettuare tesseramenti. I campi necessari per il tesseramento sono: codice fiscale, e-mail e tipologia tesseramento (tennis, paddle, tennis e paddle)

1.2.4 Gestione organizzazione

Per quanto concerne la gestione organizzazione, le funzionalità richieste sono la gestione degli utenti e dei soci.

Un giocatore non ha accesso alla gestione organizzazione. L'amministratore ed il custode possono visualizzare la lista degli utenti e la lista dei soci.

1.3 Glossario dei termini

Spieghiamo nel linguaggio naturale i termini utilizzati nel paragrafo precedente che potrebbero generare ambiguità. Per ognuno dei quali forniamo una descrizione, specifichiamo di che tipo è (sportivo o tecnico), e ne indichiamo i sinonimi.

Termine	Descrizione	Tipo	Sinonimi
Circolo tennis	Impianto sportivo dotato di campo da tennis	Sportivo	Nessuno
Custode	Dipendente addetto alla custodia e sorveglianza	Sportivo	Nessuno
Paddle	Sport con la palla di derivazione tennistica	Sportivo	Nessuno
Amministratore	Persona o gruppo di persone che amministrano il circolo tennis	Sportivo	Nessuno
Racchetta	Attrezzo utilizzato nel gioco del tennis per colpire la palla	Sportivo	Nessuno
Campo in terra rossa battuta	Campo standard per giocare a tennis	Sportivo	Nessuno
Campi da paddle	Campo per giocare a paddle	Sportivo	Nessuno
Campi in erba sintetica	Variante di campo da tennis	Sportivo	Nessuno
Copertura fissa	Struttura che ripara il campo dagli agenti esterni	Sportivo	Nessuno
Prenotazione	Atto mediante il quale vengono riservati uno o più servizi ad un socio per un certo tempo.	Tecnico	Nessuno
Tesseramento	Atto mediante il quale ci si iscrive ad un circolo	Tecnico	Nessuno
Socio	Membro del circolo	Tecnico	Nessuno
Partita	Scontrarsi per stabilire il vincitore	Tecnico	Incontro

Analisi dei requisiti

Uno studio approfondito su quello che ha richiesto il circolo ci ha reso possibile stabilire una sequenza di requisiti per il software.

2.1 Requisiti del sistema

I requisiti sono stati successivamente suddivisi in requisiti funzionali e requisiti non funzionali. I requisiti funzionali si presentano come elenchi di funzionalità o servizi che il sistema deve fornire. I requisiti non funzionali, invece, rappresentano i vincoli, le proprietà e le caratteristiche relative al sistema.

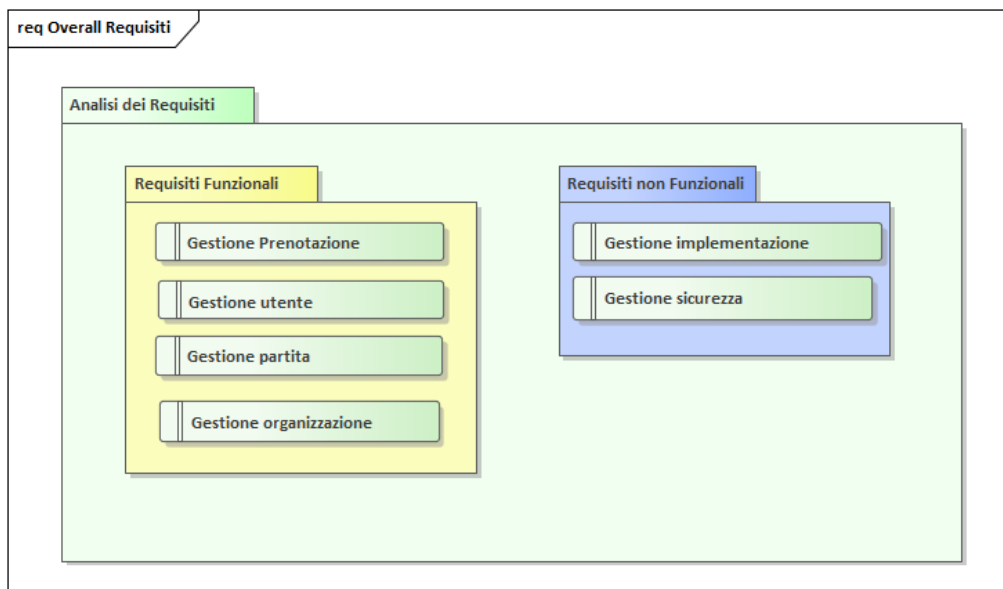


Figura 1: Suddivisione dei requisiti

2.1.1 Requisiti funzionali

I requisiti funzionali sono stati suddivisi in quattro macrocategorie: gestione prenotazione, gestione utente, gestione partite, gestione organizzazione. Ogni macrocategoria è rappresentata da un package che contiene i propri requisiti funzionali.

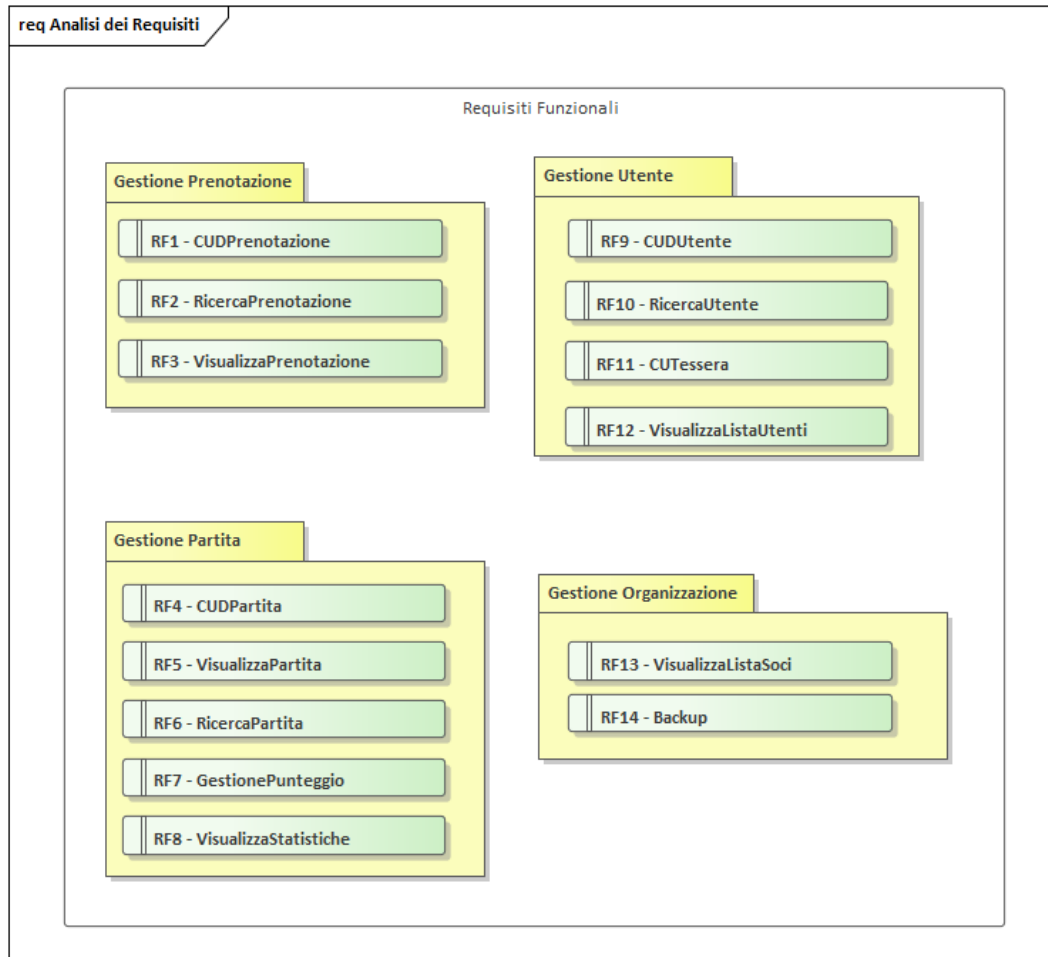


Figura 2: Suddivisione dei requisiti funzionali

Gestione Prenotazione

- **RF1 – CUDPrenotazione:** il sistema dovrà gestire le attività CUD della prenotazione
- **RF2 – RicercaPrenotazione:** il sistema dovrà gestire la ricerca di una prenotazione
- **RF3 – VisualizzaPrenotazione:** il sistema dovrà gestire la visualizzazione delle prenotazioni

Gestione Partita

- **RF4 – CUDPartita:** il sistema dovrà gestire il servizio CUD della partita.
- **RF5 – VisualizzaPartita:** il sistema dovrà gestire la visualizzazione di una partita
- **RF6 – RicercaPartita:** il sistema dovrà gestire la ricerca di una partita
- **RF7 – GestionePunteggio:** il sistema dovrà gestire l'inserimento del punteggio di una partita
- **RF8 – VisualizzaStatistiche:** il sistema dovrà gestire la visualizzazione delle statistiche, ovvero la percentuale di vittorie di un giocatore in un anno

Gestione Utente

- **RF9 – CUDUtente:** il sistema dovrà gestire le attività CUD dell'utente
- **RF10 – RicercaUtente:** il sistema dovrà gestire la ricerca di un utente
- **RF11 – CUTessera:** il sistema dovrà occuparsi della gestione della tessera
- **RF12 – VisualizzaListaUtenti:** il sistema dovrà gestire la visualizzazione della lista degli utenti

Gestione Organizzazione

- **RF13 - VisualizzaListaSoci:** il sistema dovrà gestire la visualizzazione dei soci
- **RF14 – Backup:** il sistema dovrà eseguire il backup dei dati

2.1.2 Requisiti non funzionali

I requisiti funzionali sono stati suddivisi in due: requisiti non funzionali d'implementazione e requisiti non funzionali che gestiscono la sicurezza.

Ogni macrocategoria è rappresentata da un package che contiene i propri requisiti non funzionali.

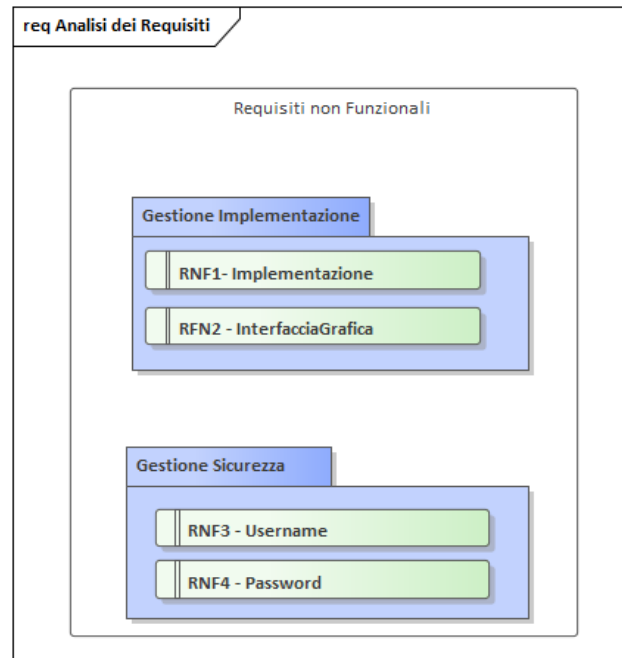


Figura 3: Suddivisione requisiti non funzionali

Gestione Implementazione

- **RNF1 - Implementazione:** Il sistema dovrà essere implementato utilizzando Python 3.
- **RNF2 - Interfaccia Grafica:** Il sistema dovrà prevedere un'interfaccia grafica

Gestione Sicurezza

- **RNF3 – Username:** Il sistema dovrà gestire l'username
- **RNF4 – Password:** Il sistema dovrà gestire la password

2.2 Diagrammi dei casi d'uso

Anche i casi d'uso del sistema sono stati suddivisi in quattro macrocategorie.

2.2.1 Gestione Prenotazione

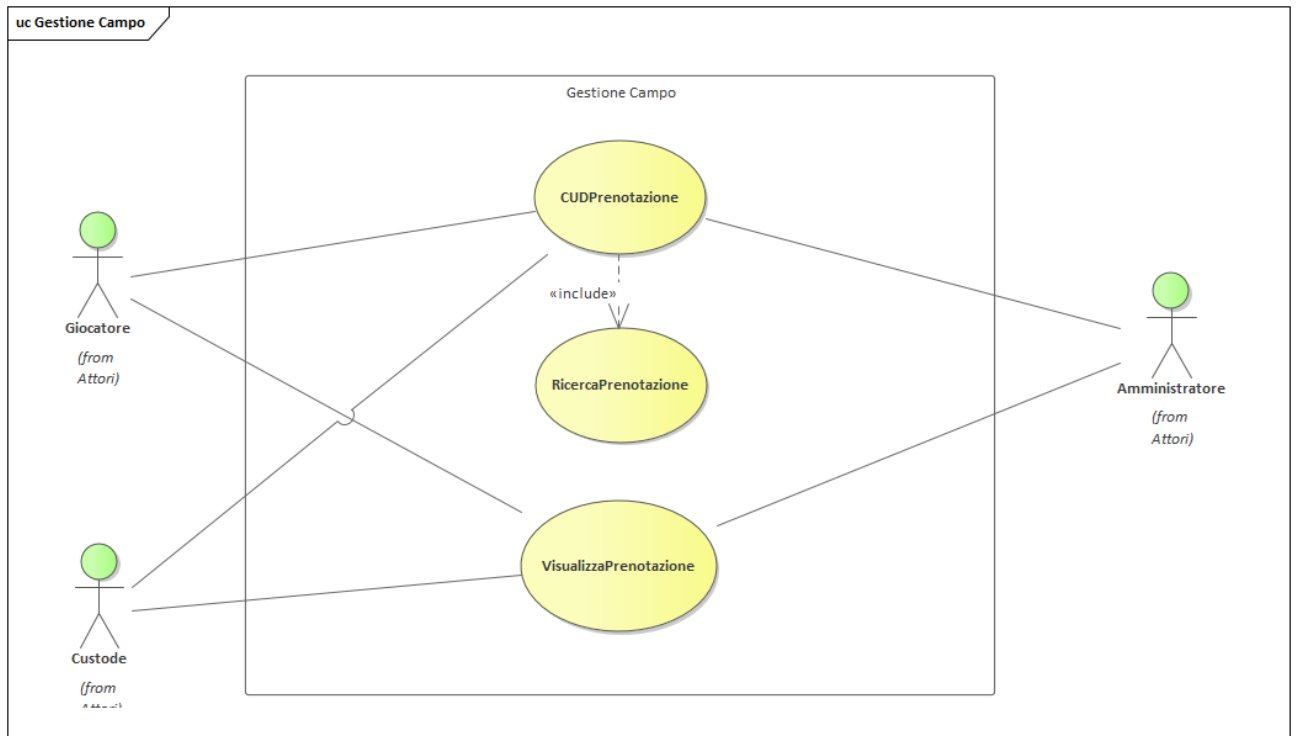


Figura 4: Diagramma dei casi d'uso Gestione campo

Il diagramma dei casi d'uso di gestione campo (Figura 4) comprende tre casi d'uso:

- CUDPrenotazione
- RicercaPrenotazione
- VisualizzaPrenotazione

Gli attori che si rapportano con questo caso d'uso sono il custode, il giocatore e l'amministratore.

CUDPrenotazione

Il caso d'uso CUDPrenotazione consente l'inserimento, la modifica o la rimozione di dati relativi ad una prenotazione.

Attori primari: Amministratore, Giocatore, Custode.

Attori secondari: Nessuno.

Precondizioni: 1. L'attore deve disporre di un account utente.

Postcondizioni: Nessuna.

Sequenza degli eventi principale:

1. Il caso d'uso inizia quando l'attore primario vuole effettuare un'operazione CUD relativa ad una prenotazione.
2. *if* l'attore vuole inserire una nuova prenotazione:
 - 2.1. L'attore inserisce tutti i dati relativi alla prenotazione.
 - 2.2. *include* (RicercaPrenotazione)
 - 2.3. *if* la prenotazione è già esistente:
 - 2.3.1. Il sistema informa che non è possibile prenotare il campo.
 - 2.4. *else*:
 - 2.4.1. Il sistema provvede ad aggiungere la prenotazione.
 - 2.4.2. Il sistema mostra un messaggio di conferma prenotazione.
3. *else if* l'attore vuole rimuovere una prenotazione:
 - 3.1. L'attore inserisce dei parametri per ricercare la prenotazione da rimuovere.
 - 3.2. *include* (RicercaPrenotazione)
 - 3.3. Il sistema provvede a eliminare la prenotazione.
4. Il sistema mostra un messaggio di conferma eliminazione.

Sequenza degli eventi alternativa: Nessuna.

RicercaPrenotazione

Il caso d'uso RicercaPrenotazione consente di ricercare delle prenotazioni inserendo dei determinati parametri di ricerca.

Attori primari: Amministratore, Giocatore, Custode.

Attori secondari: Nessuno.

Precondizioni:

1. L'attore deve disporre di un account utente.

2. Devono esistere delle prenotazioni

Postcondizioni: Nessuna.

Sequenza degli eventi principale:

1. Il caso d'uso inizia quando l'attore primario o il sistema vuole ricercare una prenotazione.
2. Il sistema preleva i parametri di ricerca di una prenotazione.
3. *for each* prenotazione nel database:
 - 3.1. Il sistema confronta i parametri di ricerca con i parametri della prenotazione.
 - 3.2. *if* i parametri corrispondono:
 - 3.2.1. Il sistema aggiunge la prenotazione alla lista del risultato della ricerca.
4. *if* il sistema ha trovato almeno una prenotazione corrispondente ai parametri di ricerca:
 - 4.1. Il sistema fornisce tutti i dati relativi alle prenotazioni trovate.
5. *else*:
 - 5.1. Il sistema manda un messaggio dicendo che non ha trovato prenotazioni corrispondenti a parametri

Sequenza degli eventi alternativa: Nessuna.

VisualizzaPrenotazione

Il caso d'uso VisualizzaPrenotazione consente di visualizzare la lista delle prenotazioni.

Attori primari: Amministratore, Giocatore, Custode.

Attori secondari: Nessuno.

Precondizioni: 1. L'attore deve disporre di un account utente.

Postcondizioni: Nessuna.

Sequenza degli eventi principale:

1. Il caso d'uso inizia quando l'attore vuole visualizzare la lista delle prenotazioni.
2. Il sistema mostra a schermo la lista delle prenotazioni.

Sequenza degli eventi alternativa: Nessuna.

2.2.2 Gestione partita

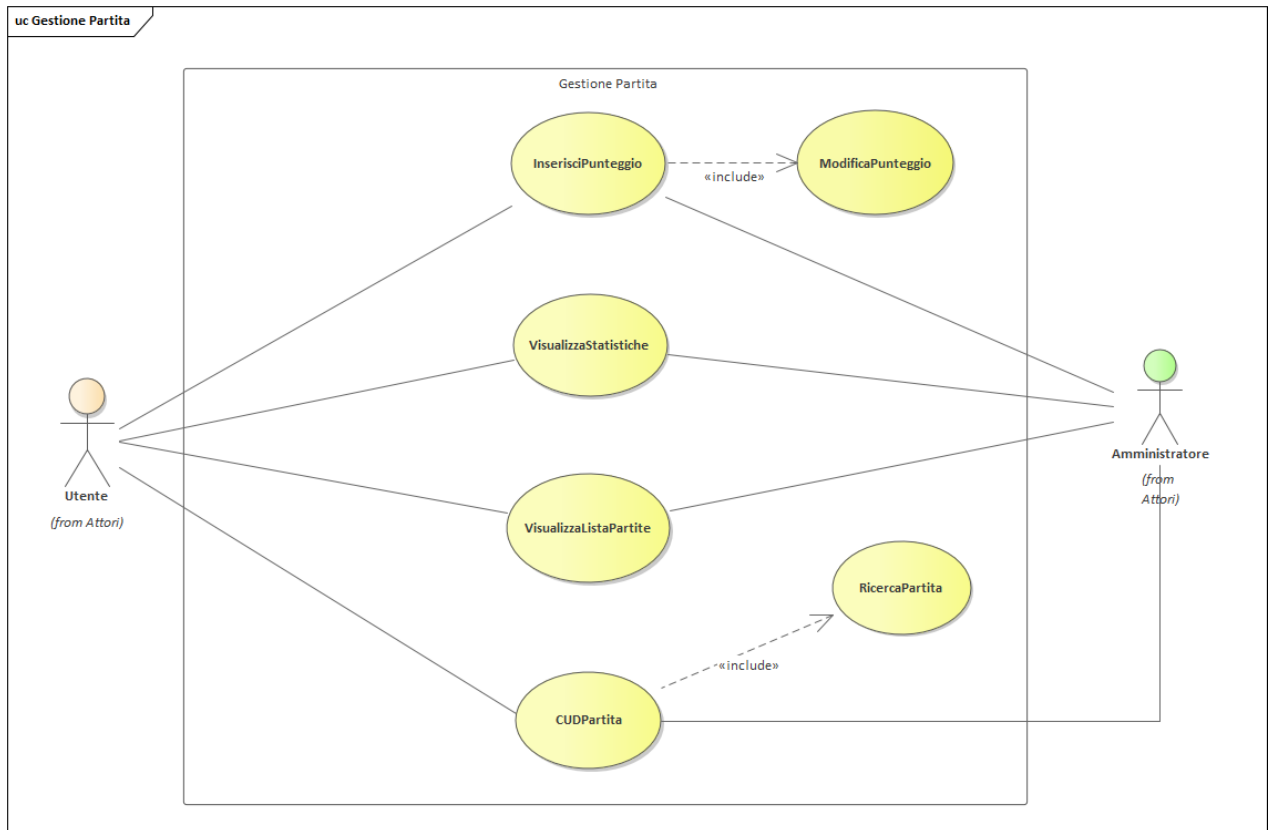


Figura 5: Diagramma dei casi d'uso gestione partita

Il diagramma dei casi d'uso di gestione campo (Figura 5) comprende sei casi d'uso:

- InserisciPunteggio
- ModificaPunteggio
- VisualizzaStatistiche
- VisualizzaListaPartite
- CUDPartita
- RicercaPartita

Gli attori che si rapportano con questo caso d'uso sono il custode, il giocatore e l'amministratore e il custode.

InserisciPunteggio

Il caso d'uso InserisciPunteggio consente l'inserimento di un punteggio di una partita.

Attori primari: Amministratore, Giocatore, Custode.

Attori secondari: Nessuno.

Precondizioni:

1. L'attore deve disporre di un account utente.
2. L'attore deve aver disputato almeno una partita

Postcondizioni: Nessuna.

Sequenza degli eventi principale:

1. Il caso d'uso inizia quando l'attore primario vuole inserire un punteggio di una partita.
2. **i f** l'attore vuole inserire un nuovo punteggio:
 - 2.1. L'attore sceglie la partita a cui aggiungere il punteggio.
 - 2.2. L'attore inserisce il punteggio.
 - 2.3. Il sistema provvede ad aggiungere il punteggio.
3. Il sistema aggiunge il punteggio alle partite.

VisualizzaStatistiche

Il caso d'uso VisualizzaStatistiche consente di visualizzare le informazioni sulle statistiche.

Attori primari: Amministratore, Giocatore, Custode.

Attori secondari: Nessuno.

Precondizioni:

1. L'attore deve disporre di un account utente.
2. L'attore deve aver disputato almeno una partita

Postcondizioni: Nessuna.

Sequenza degli eventi principale:

1. Il caso d'uso inizia quando l'attore vuole visualizzare le statistiche.
2. Il sistema mostra le statistiche.

Sequenza degli eventi alternativa: Nessuna.

CUDPartita

Il caso d'uso CUDPartita consente l'inserimento, la modifica o la rimozione di dati relativi ad una partita.

Attori primari: Amministratore, Giocatore, Custode.

Attori secondari: Nessuno.

Precondizioni: 1. L'attore deve disporre di un account utente.

Postcondizioni: Nessuna.

Sequenza degli eventi principale:

1. Il caso d'uso inizia quando l'attore primario vuole effettuare un'operazione CUD relativa ad una partita.
2. `if` l'attore vuole inserire una nuova partita:
 - 2.1. L'attore inserisce tutti i dati relativi alla partita.
 - 2.2. `include`(RicercaPartita)
 - 2.3. `if` la partita è già esistente:
 - 2.3.1. Il sistema mostra un messaggio in cui informa che la partita già esiste.
 - 2.4. `else`:
 - 2.4.1. Il sistema provvede ad aggiungere la partita.
 - 2.4.2. Il sistema mostra un messaggio di successo.
3. `else if` l'attore vuole modificare una partita già esistente:
 - 3.1. L'attore inserisce dei parametri per ricercare la partita da modificare.
 - 3.2. `include`(RicercaPartita)
 - 3.3. Il sistema mostra la partita trovata.
 - 3.4. L'attore modifica i dati desiderati della partita.
 - 3.5. Il sistema mostra un messaggio di conferma modifica.
4. `else if` l'attore vuole rimuovere una partita:
 - 4.1. L'attore inserisce dei parametri per ricercare la partita da rimuovere.
 - 4.2. `include`(RicercaPartita)
 - 4.3. Il sistema provvede a eliminare la partita.
 - 4.4. Il sistema mostra un messaggio di conferma eliminazione.

Sequenza degli eventi alternativa: Nessuna.

RicercaPartita

Il caso d'uso RicercaPartita consente di ricercare una partita inserendo dei determinati parametri di ricerca.

Attori primari: Amministratore, Custode, Giocatore.

Attori secondari: Nessuno.

Precondizioni:

1. L'attore deve disporre di un account utente.

Postcondizioni: Nessuna.

Sequenza degli eventi principale:

1. Il caso d'uso inizia quando l'attore primario o il sistema vuole ricercare una partita.
2. Il sistema preleva i parametri di ricerca di una partita.
3. `for each` partite nel database:
 - 3.1. Il sistema confronta i parametri di ricerca con i parametri della partita.

- 3.2. *if* i parametri corrispondono:
 - 3.3. Il sistema aggiunge la partita alla lista del risultato della ricerca.
 4. *if* il sistema ha trovato almeno una partita corrispondente ai parametri di ricerca:
 - 4.1. Il sistema fornisce la partita.
 5. *else*:
 - 5.1. Il sistema manda un messaggio dicendo che non ha trovato partite corrispondenti ai parametri
- Sequenza degli eventi alternativa:** Nessuna.

2.2.3 Gestione Utenti

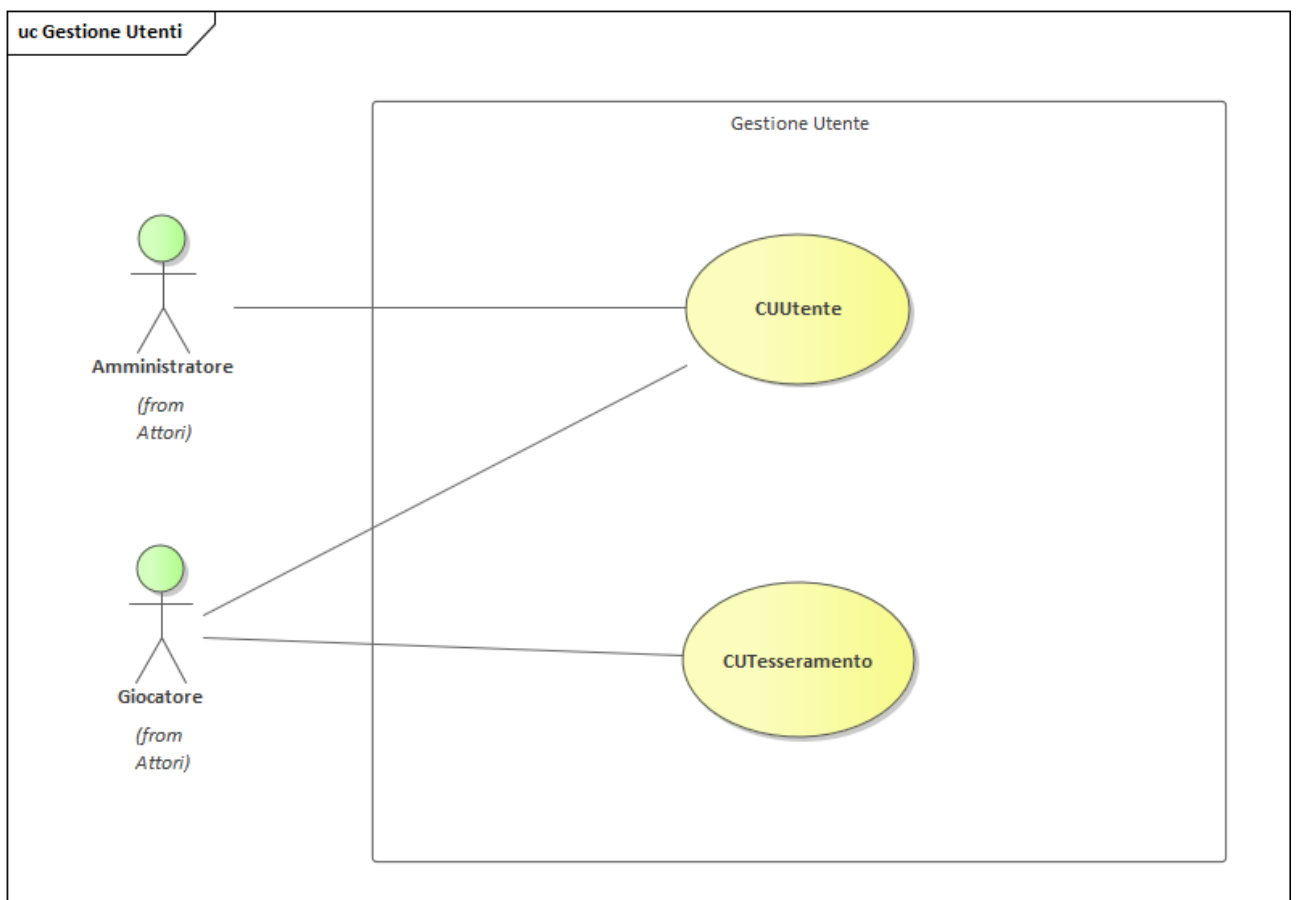


Figura 6: Diagramma dei casi d'uso Gestione utenti

Il diagramma dei casi d'uso di gestione utenti (Figura 6) comprende sei casi d'uso:

- CUUtente
- CUTesseramento

Gli attori che si rapportano con questo caso d'uso sono il custode, il giocatore e l'amministratore.

CUUtente

Il caso d'uso CUUtente consente l'inserimento e la modifica di dati relativi ad una prenotazione.

Attori primari: Amministratore, Giocatore.

Attori secondari: Nessuno.

Precondizioni:

1. L'attore deve disporre di un account utente se vuole effettuare operazioni di modifica

Postcondizioni: Nessuna.

Sequenza degli eventi principale:

1. Il caso d'uso inizia quando l'attore primario vuole effettuare un'operazione CUD relativa ad un utente.
2. *if* l'attore vuole inserire un nuovo utente:
 - 2.1. L'attore inserisce tutti i dati relativi all'utente.
 - 2.2 Il sistema provvede ad aggiungere l'utente.
 - 2.3 Il sistema mostra un messaggio di conferma aggiunta.
3. *else if* l'attore vuole modificare un utente già esistente:
 - 3.1. L'attore modifica i dati desiderati dell'utente.
 - 3.2. Il sistema mostra un messaggio di conferma modifica.

Sequenza degli eventi alternativa: Nessuna.

CUTesseramento

Il caso d'uso tesseramento consente l'inserimento e la modifica di dati relativi ad un tesseramento.

Attori primari: Giocatore.

Attori secondari: Nessuno.

Precondizioni: 1. L'attore deve disporre di un account utente.

Postcondizioni: Nessuna.

Sequenza degli eventi principale:

1. Il caso d'uso inizia quando l'attore primario vuole effettuare un'operazione relativa ad un tesseramento.
2. *if* l'attore vuole inserire una nuova tessera:
 - 2.1. L'attore inserisce tutti i dati relativi alla tessera.
 - 2.3 Il sistema provvede ad aggiungere la tessera.
 - 2.4. Il sistema mostra un messaggio di conferma aggiunta.
3. *else if* l'attore vuole modificare una tessera già esistente:
 - 3.1. L'attore modifica i dati desiderati della tessera.
 - 3.2. Il sistema mostra un messaggio di conferma modifica.

Sequenza degli eventi alternativa: Nessuna.

2.2.4 Gestione organizzazione

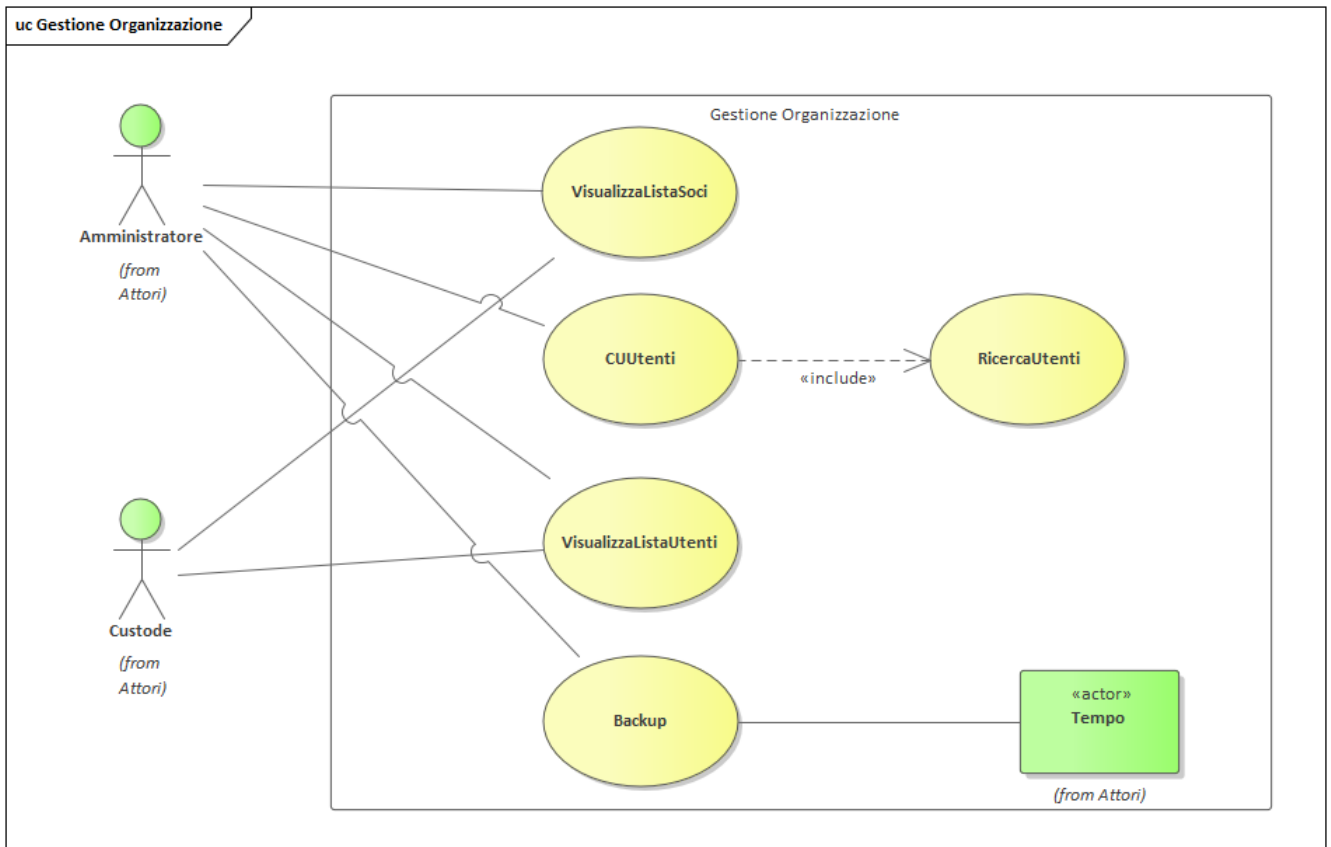


Figura 7: Diagramma dei casi d'uso gestione organizzazione

Il diagramma dei casi d'uso di gestione organizzazione (Figura 7) comprende sette casi d'uso:

- Backup
- VisualizzaListaSoci
- CUUtenti
- RicercaUtenti
- VisualizzaListaUtenti

Gli attori che si rapportano con questo caso d'uso sono il custode, il giocatore, l'amministratore e il tempo.

Backup

Il caso d'uso Backup consente di fare un backup dei dati quando lo decide l'amministratore.

Attori primari: Tempo, Amministratore.

Attori secondari: Nessuno.

Precondizioni: Nessuna

Postcondizioni:

1. I dati sono stati salvati.

Sequenza degli eventi principale:

1. Il caso d'uso inizia quando l'amministratore decide di fare il backup o alle ore 2

2. Il sistema preleva i dati delle prenotazioni.

3. Il sistema copia i dati delle prenotazioni sul disco.

4. Il sistema preleva i dati degli utenti.

5. Il sistema copia i dati degli utenti sul disco.

Sequenza degli eventi alternativa: Nessuna.

2.3 Matrice di mapping dei requisiti

	Requisiti::RF1 - CUDPrenotazione	Requisiti::RF10 - RicercaUtente	Requisiti::RF11 - CUTesseramento	Requisiti::RF12 - VisualizzaListaUtenti	Requisiti::RF13 - VisualizzaListaSoci	Requisiti::RF14 - Backup	Requisiti::RF2 - RicercaPrenotazione	Requisiti::RF3 - VisualizzaPrenotazione	Requisiti::RF4 - CUDPartita	Requisiti::RF5 - VisualizzaPartita	Requisiti::RF6 - RicercaPartita	Requisiti::RF7 - GestionePunteggio	Requisiti::RF8 - VisualizzaStatistiche	Requisiti::RF9 - CUDUtente
Gestione Organizzazione::Backup						↑								
Gestione Organizzazione::CUUtenti				↑										↑
Gestione Organizzazione::RicercaUtente		↑												↑
Gestione Organizzazione::RicercaUtenti		↑		↑										
Gestione Organizzazione::VisualizzaListaSoci					↑									
Gestione Organizzazione::VisualizzaListaUtenti				↑										
Gestione Partita::CUDPartita							↑		↑					
Gestione Partita::InserisciPunteggio												↑		
Gestione Partita::ModificaPunteggio												↑		
Gestione Partita::RicercaPartita										↑	↑			
Gestione Partita::VisualizzaListaPartite									↑	↑	↑			
Gestione Partita::VisualizzaStatistiche													↑	
Gestione Prenotazioni::CUDPrenotazione	↑						↑	↑						
Gestione Prenotazioni::RicercaPrenotazione	↑						↑	↑						
Gestione Prenotazioni::VisualizzaPrenotazione	↑						↑	↑						
Gestione Utenti::CUTesseramento			↑											
Gestione Utenti::CUUtente		↑												↑

Figura 8: Matrice di mapping dei requisiti

Diagrammi di analisi

Dopo aver realizzato i casi d'uso abbiamo implementato, a partire da essi, i diagrammi delle classi che sono composti da diagrammi di sequenza e diagrammi di attività.

I diagrammi di analisi rappresentano le mansioni che il software deve ricoprire.

3.1 Diagramma delle classi di analisi

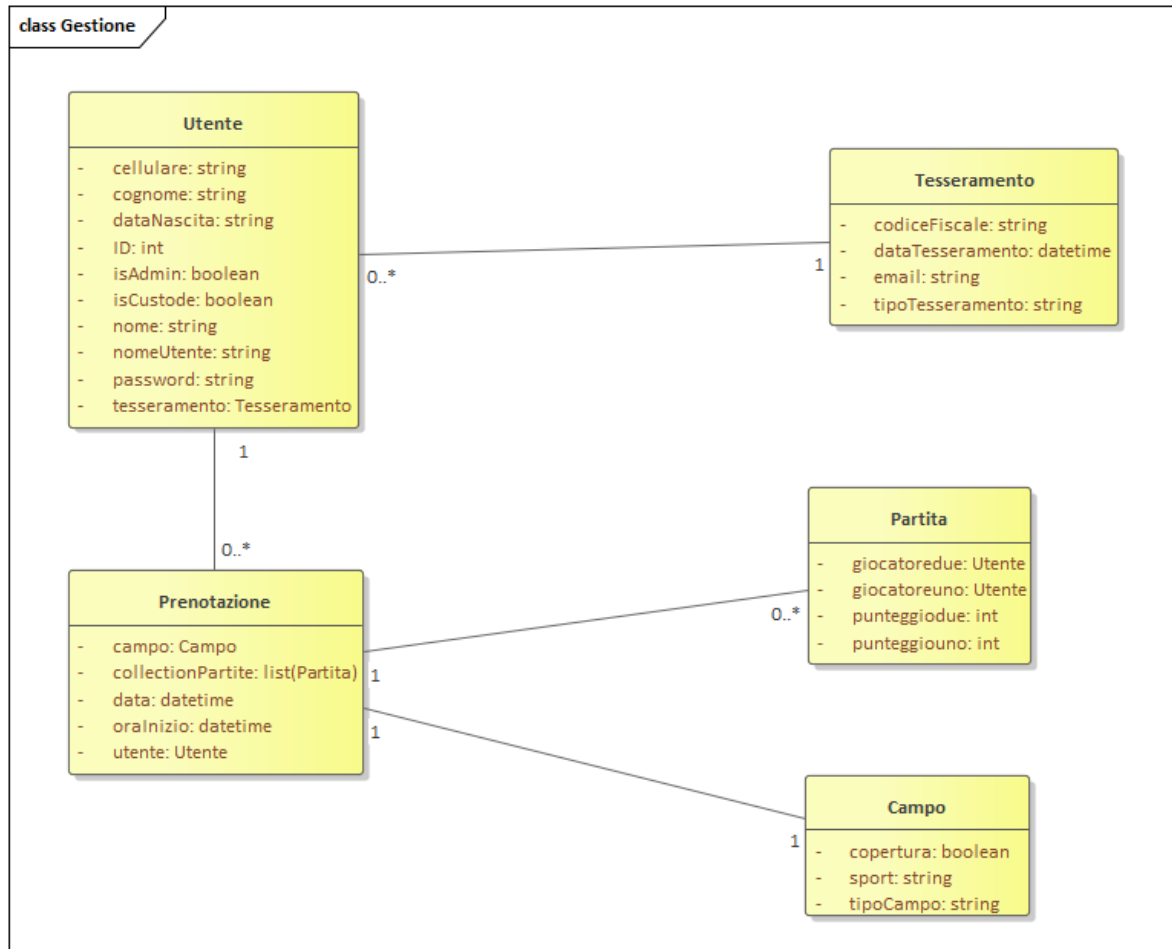


Figura 9: Diagramma delle classi di analisi

3.2 Diagrammi di sequenza

3.2.1 backup

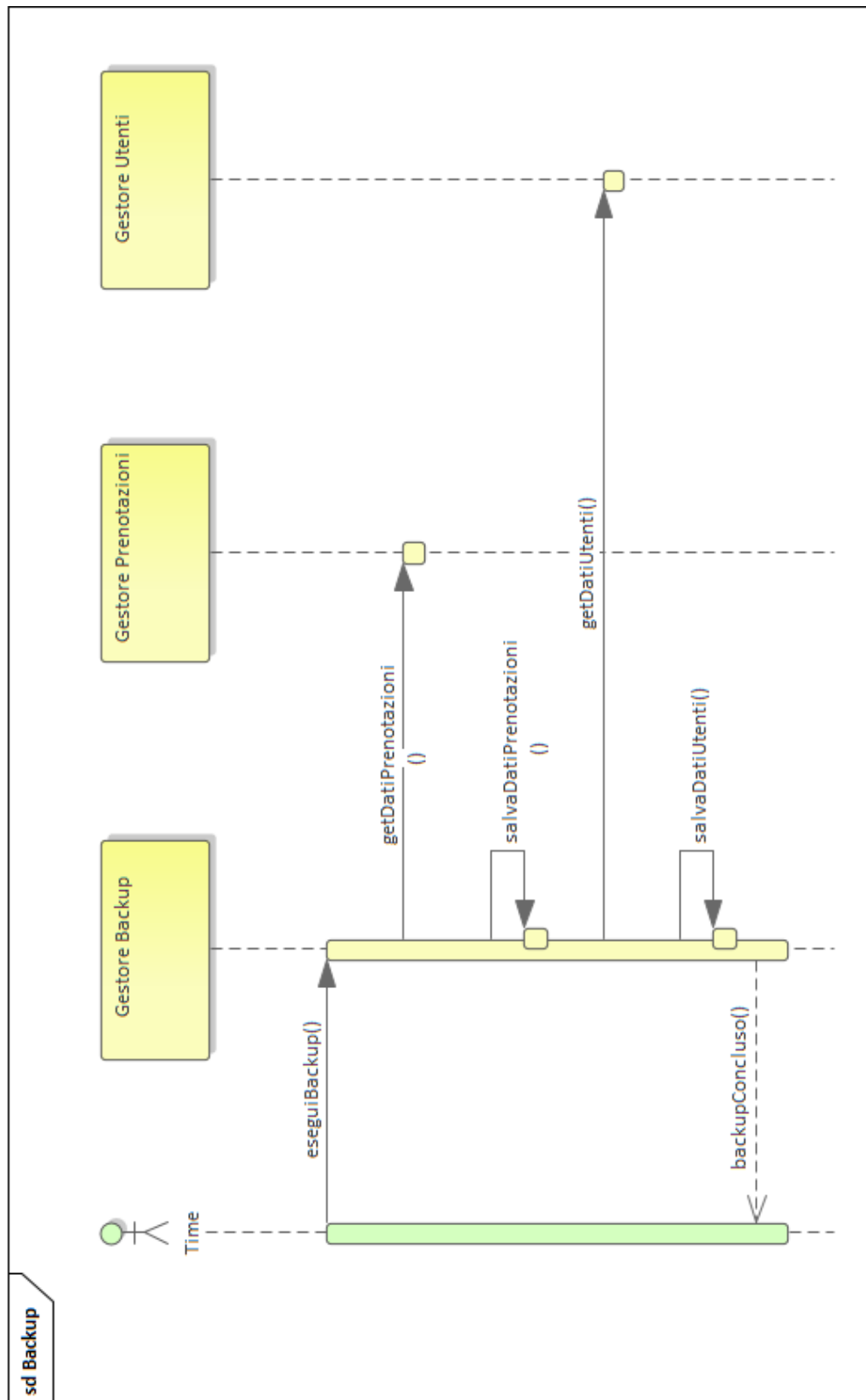


Figura 10: Diagramma di sequenza backup

3.2.2 CUDPartita

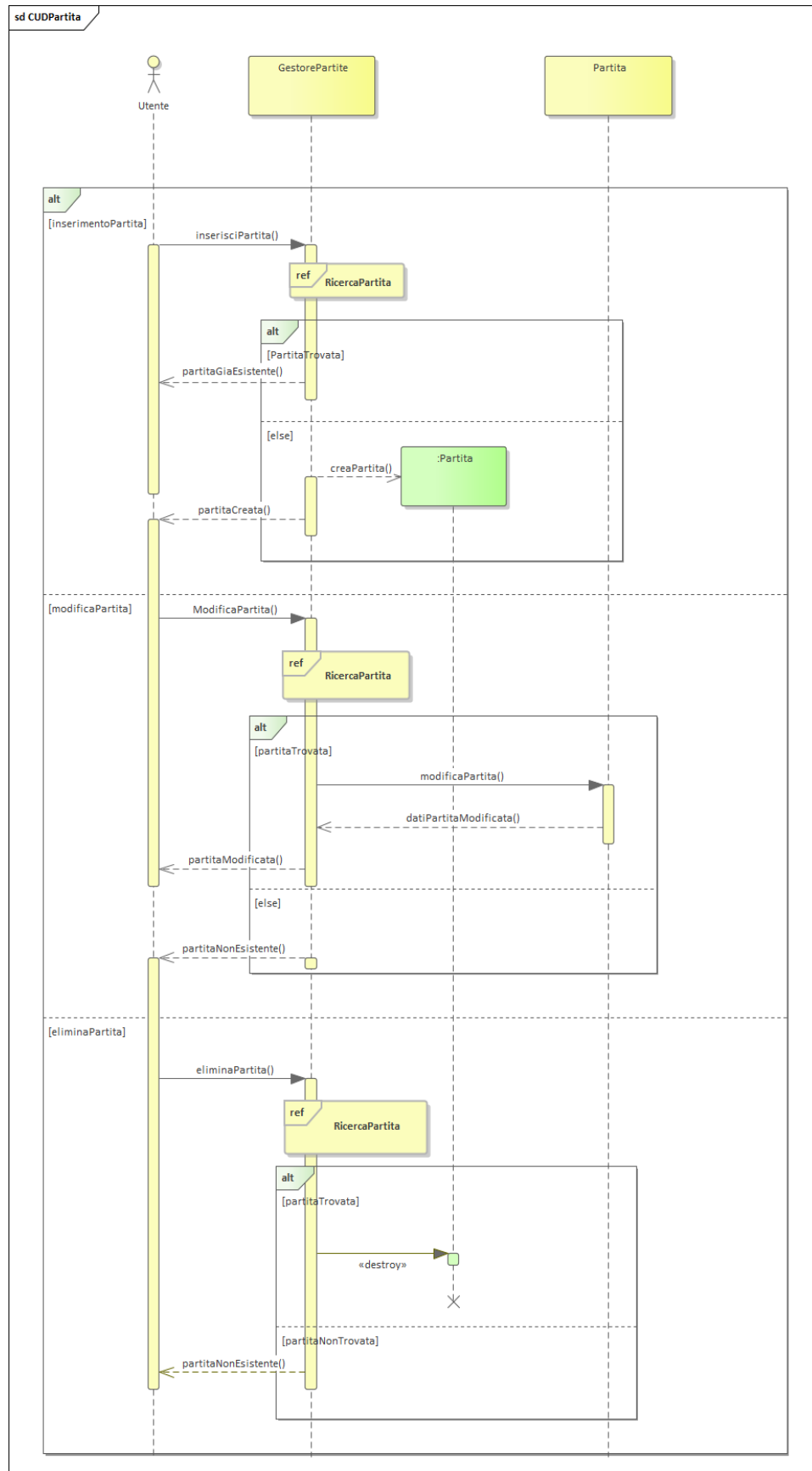


Figura 11: Diagramma di sequenza CUDPartita

3.2.3 CUDPrenotazione

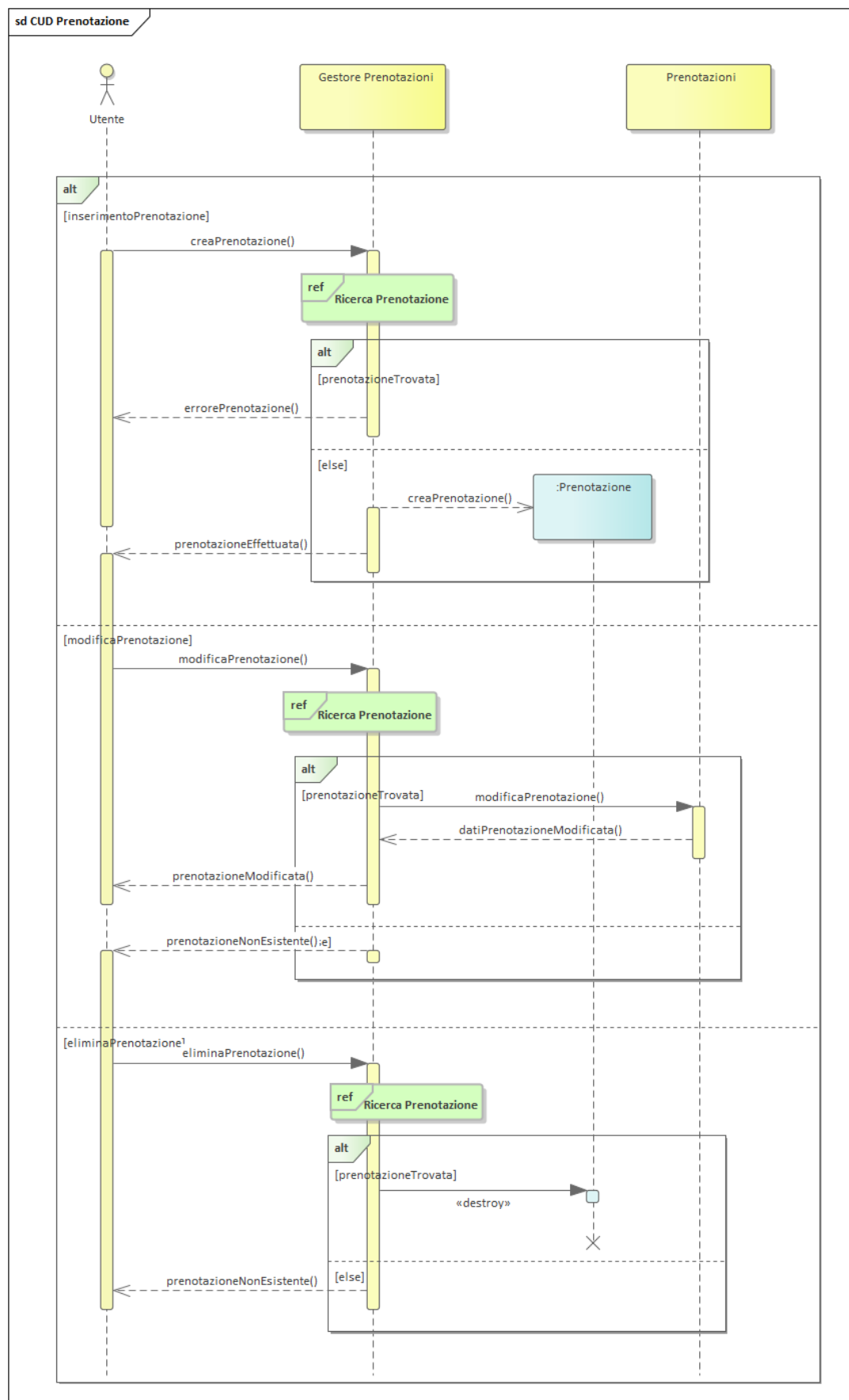


Figura 12: Diagramma di sequenza CUDPrenotazione

3.2.4 Tesseramento

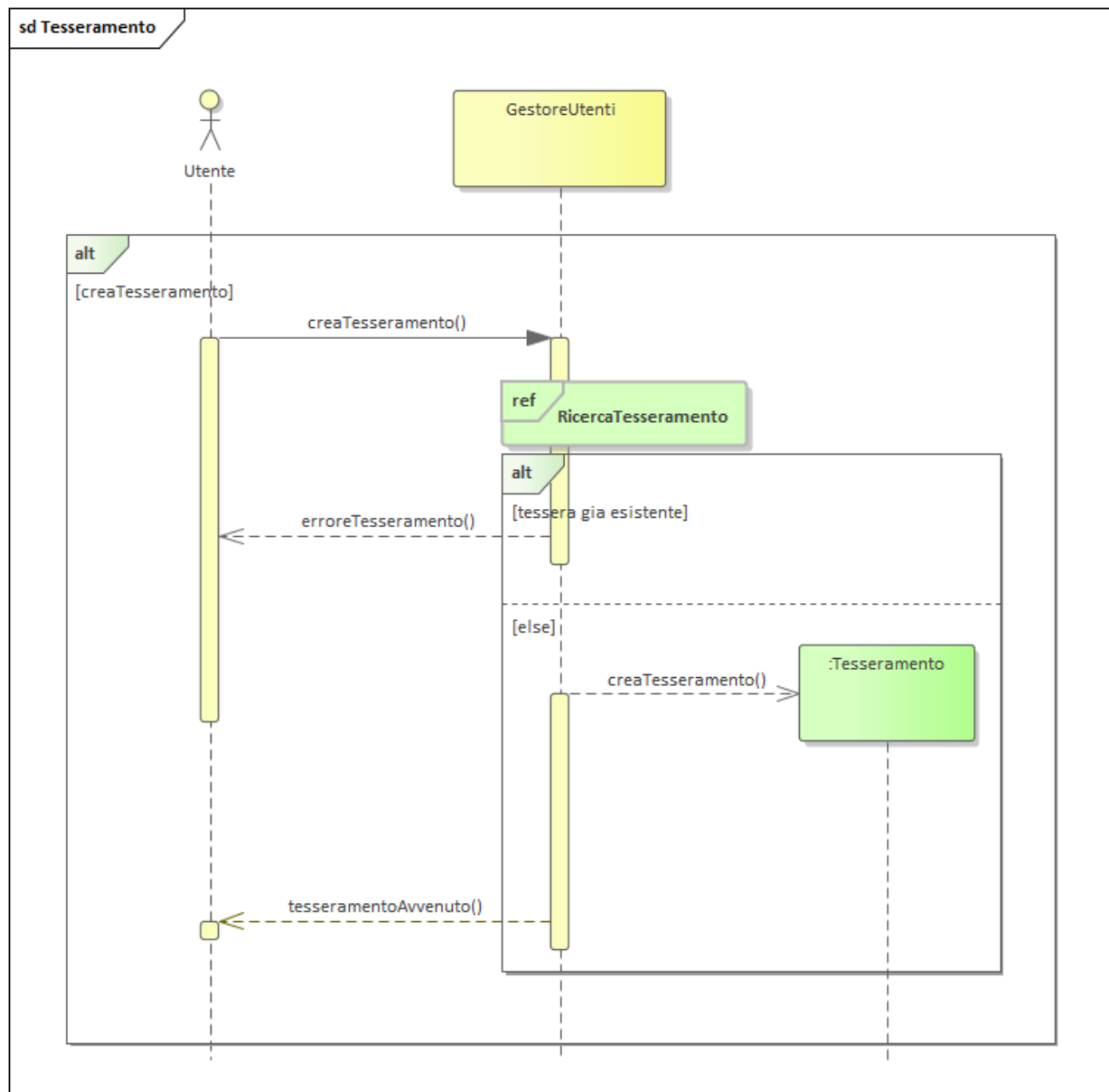


Figura 13: Diagramma di sequenza tesseramento

3.2.5 InserisciPunteggio

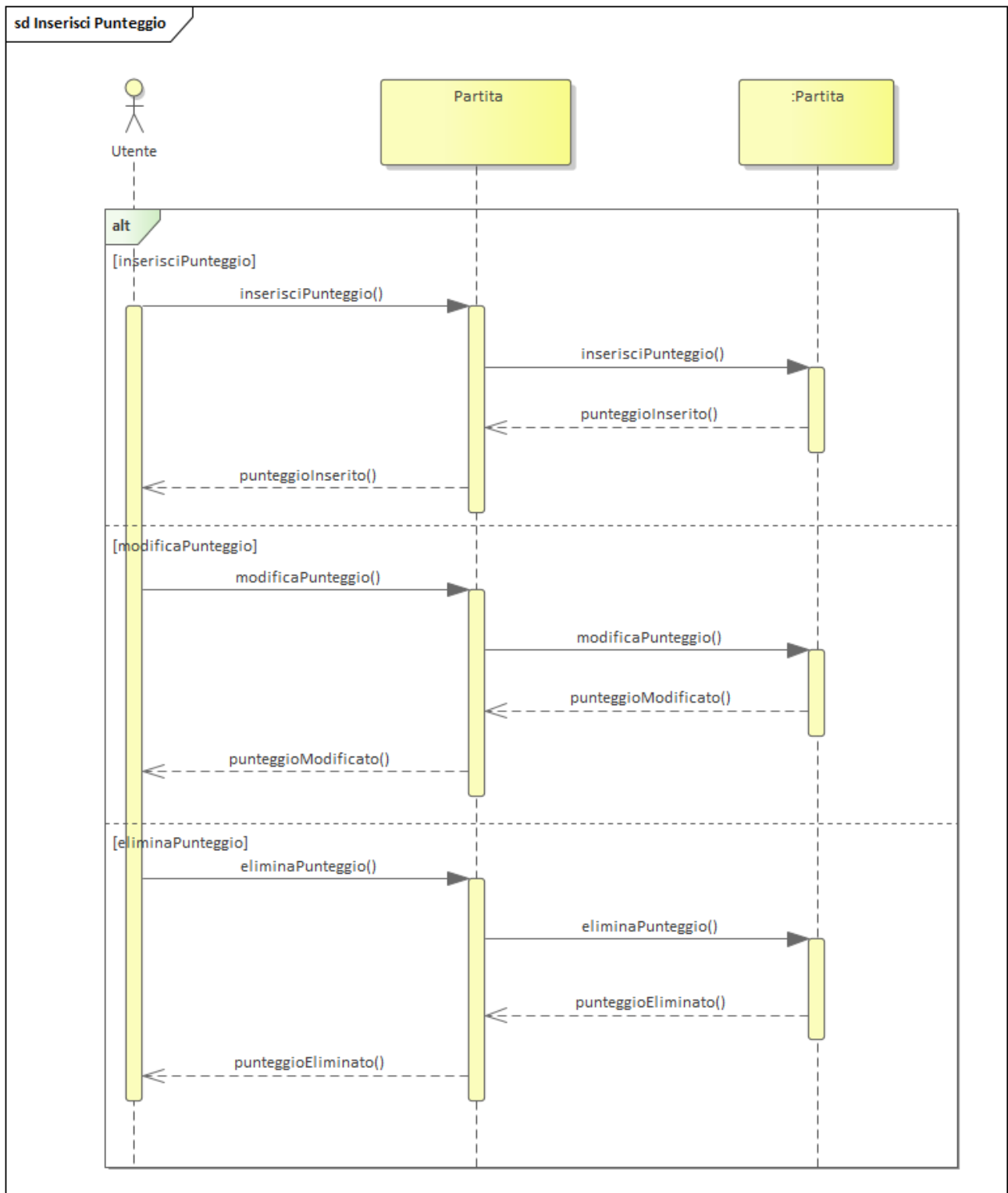


Figura 14: Diagramma di sequenza InserisciPunteggio

3.2.6 RicercaPartita

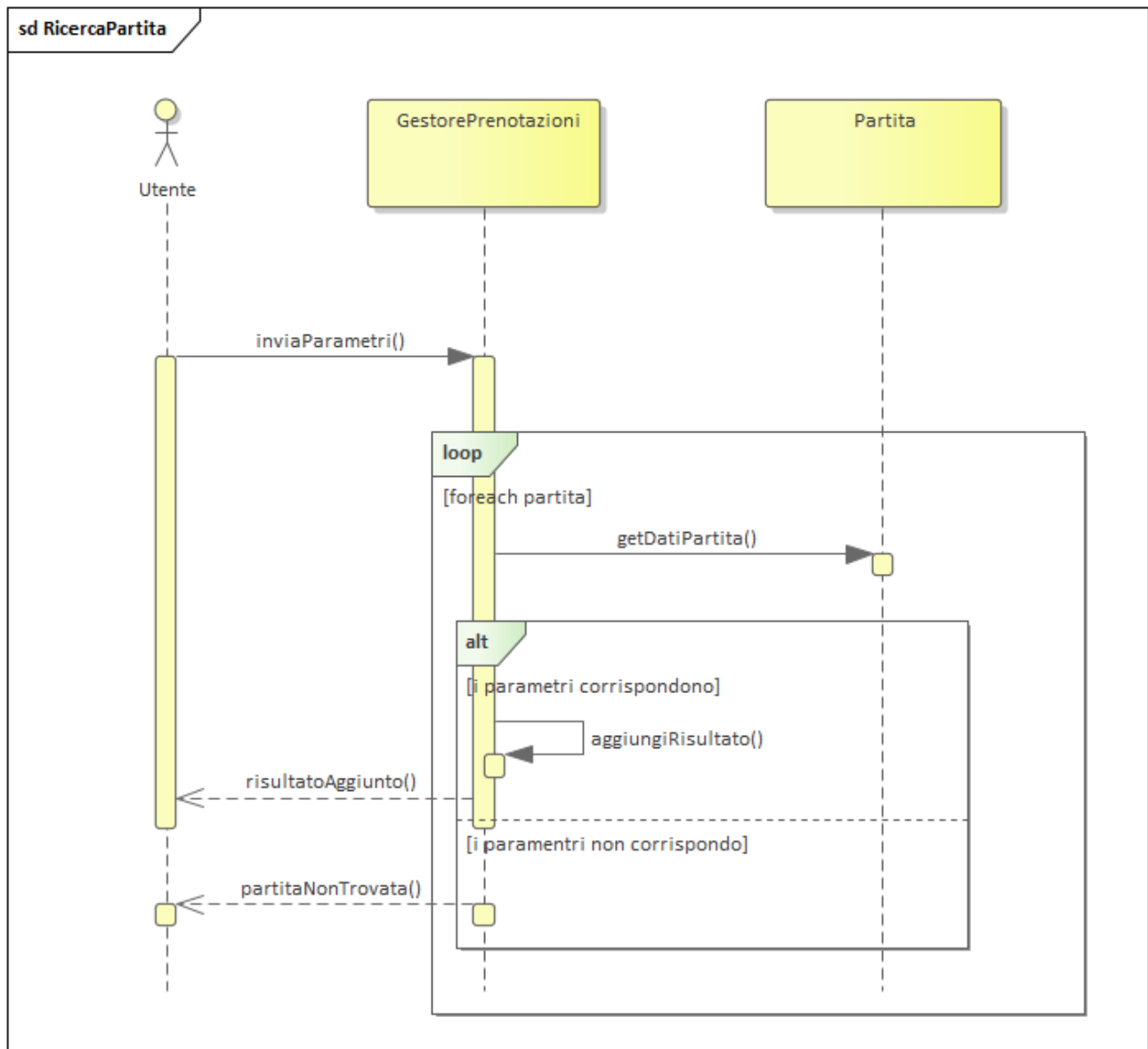


Figura 15: Diagramma di sequenza ricercaPartita

3.3 Diagrammi di attività

3.3.1 Backup

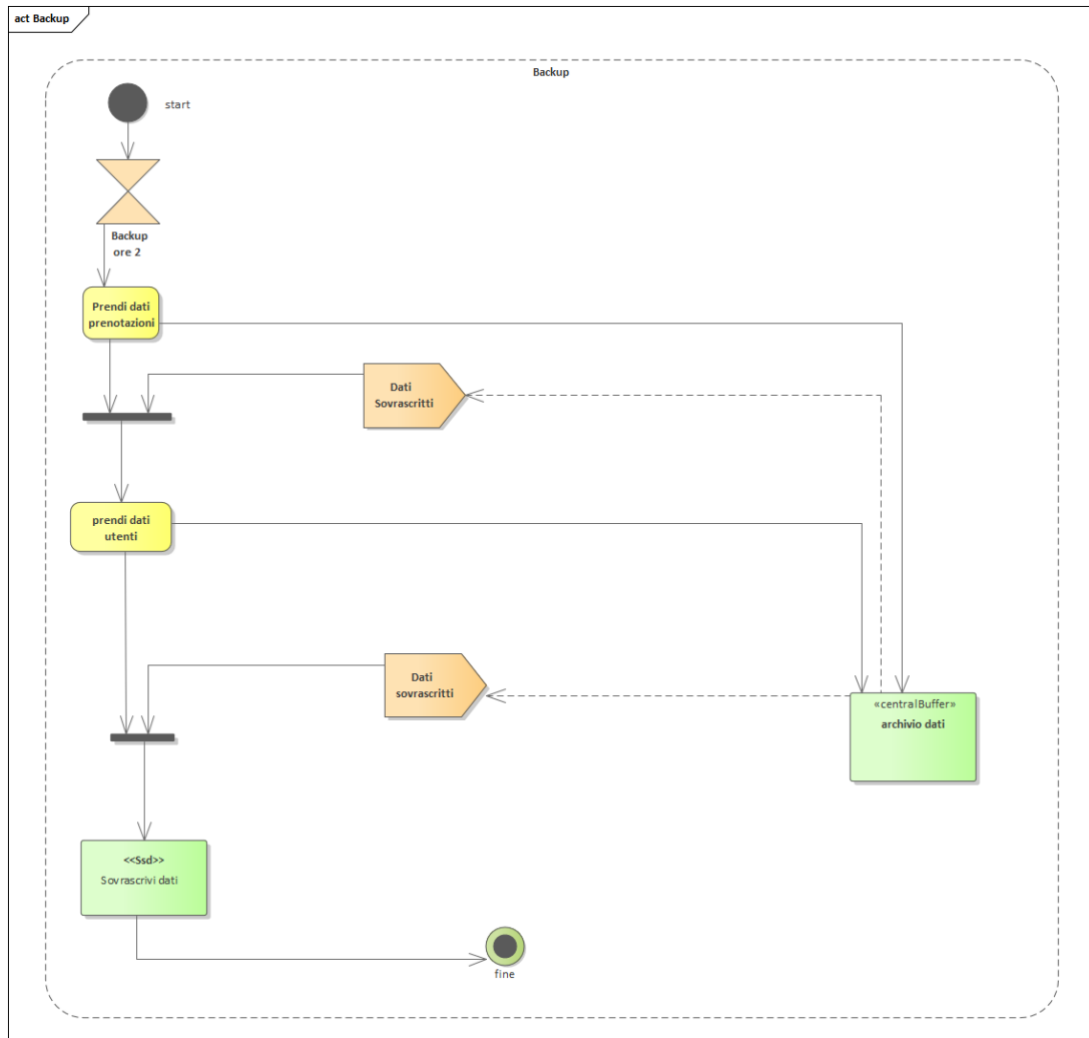


Figura 16: Diagramma di attività backup

3.3.2 CUDPartita

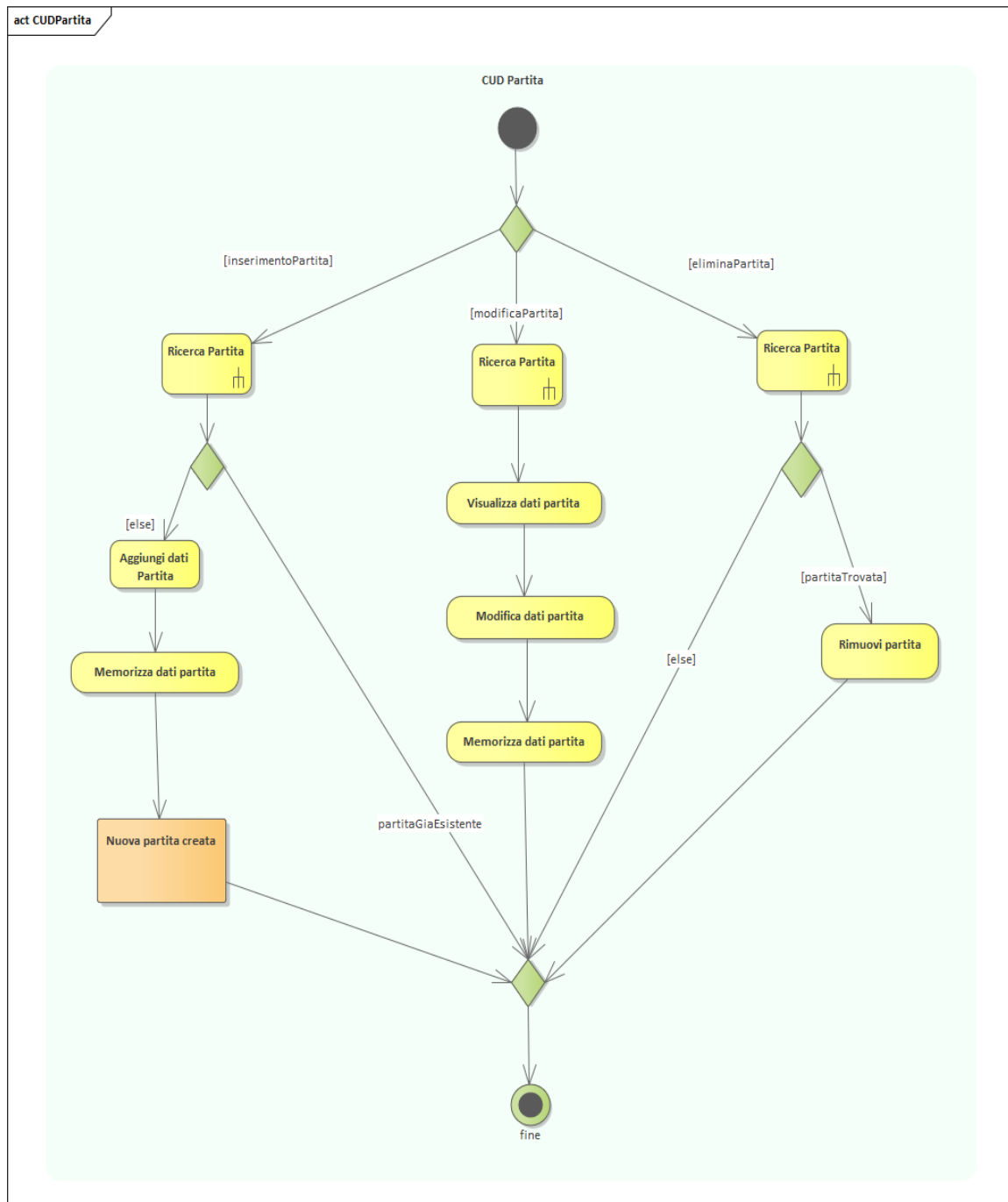


Figura 17: Diagramma di attività CUDPartita

3.3.3 GestionediTessera

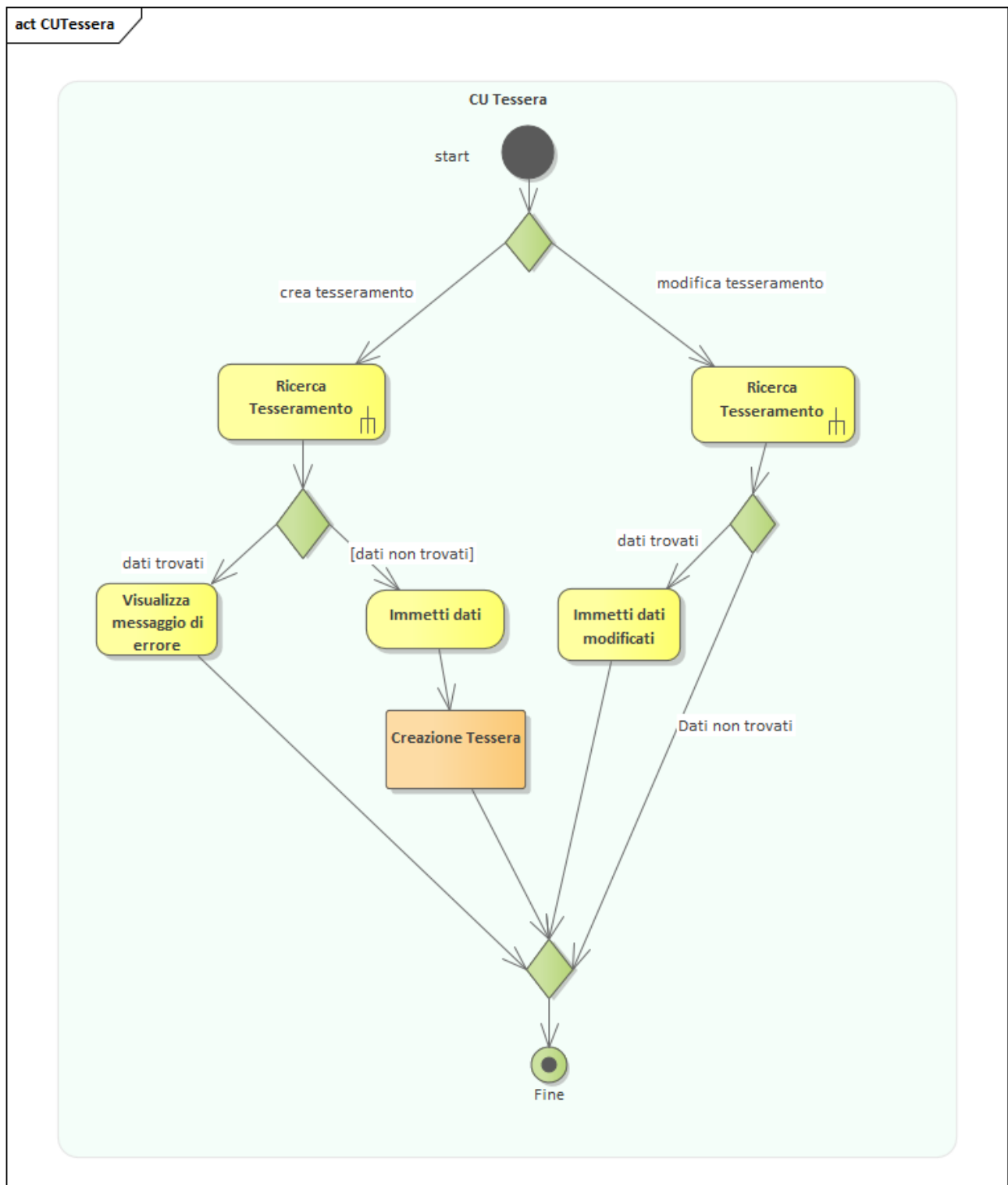


Figura 18: Diagramma di attività CUTessera

3.3.4 RicercaPartita

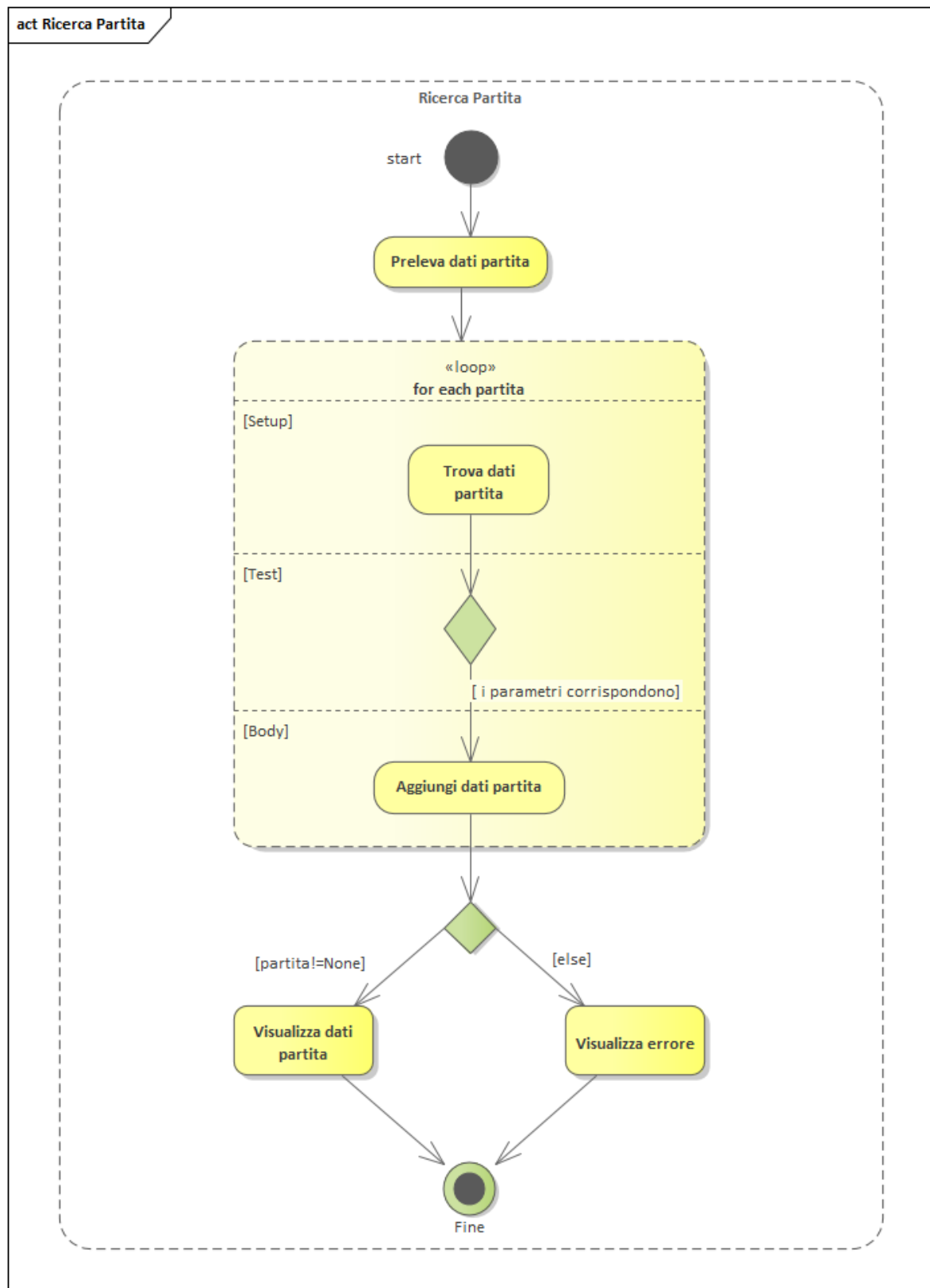


Figura 19: Diagramma di attività RicercaPartita

Diagrammi di progettazione

Per quanto riguarda la progettazione abbiamo creato le classi di progettazione partendo dal diagramma delle classi e dal diagramma di analisi, per poi realizzare quattro macchine a stati ed il diagramma dei componenti.

Abbiamo suddiviso le classi di progettazione in classi di gestione e classi generali.

Le macchine a stati riguardano le classi: campo, partita, prenotazione e tesseramento.

4.1 Classi di gestione

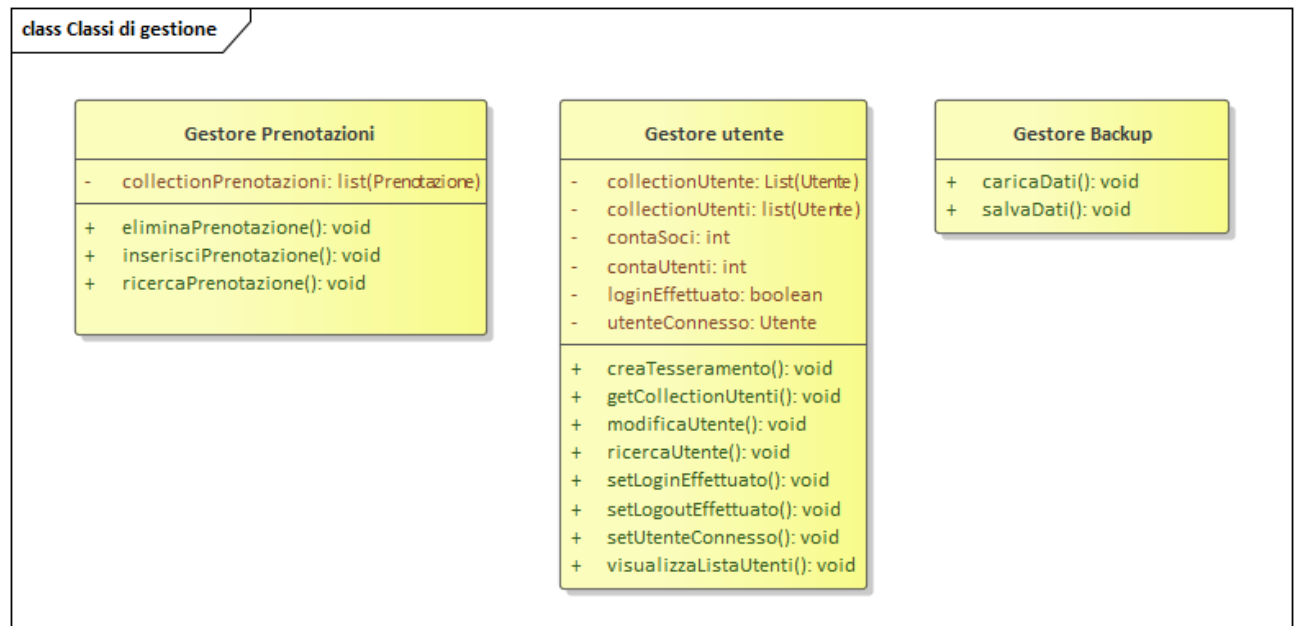


Figura 20: Classi di gestione

4.2 Classi Generali

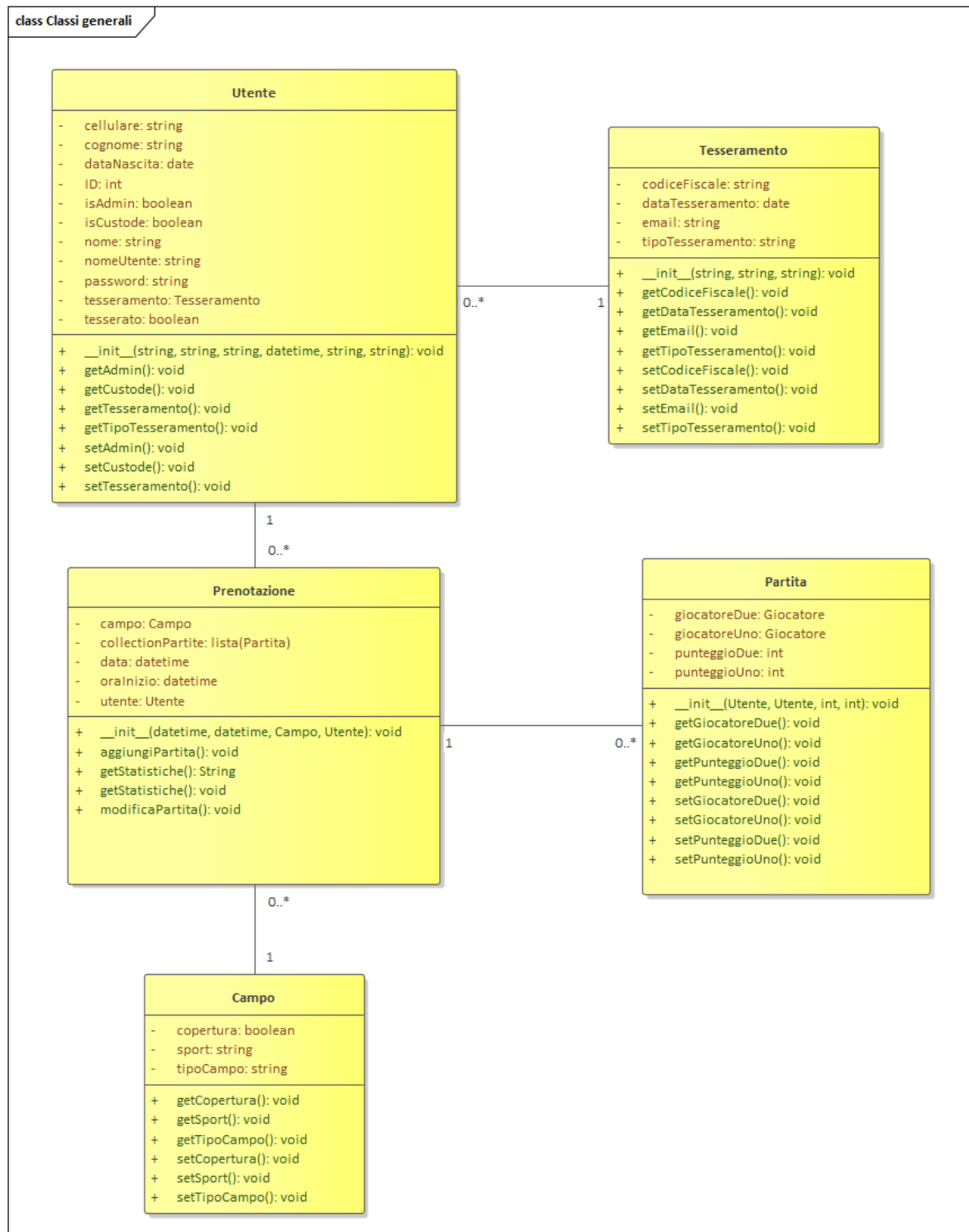


Figura 21: Classi generali

4.3 Macchine a stati

4.3.1 Campo

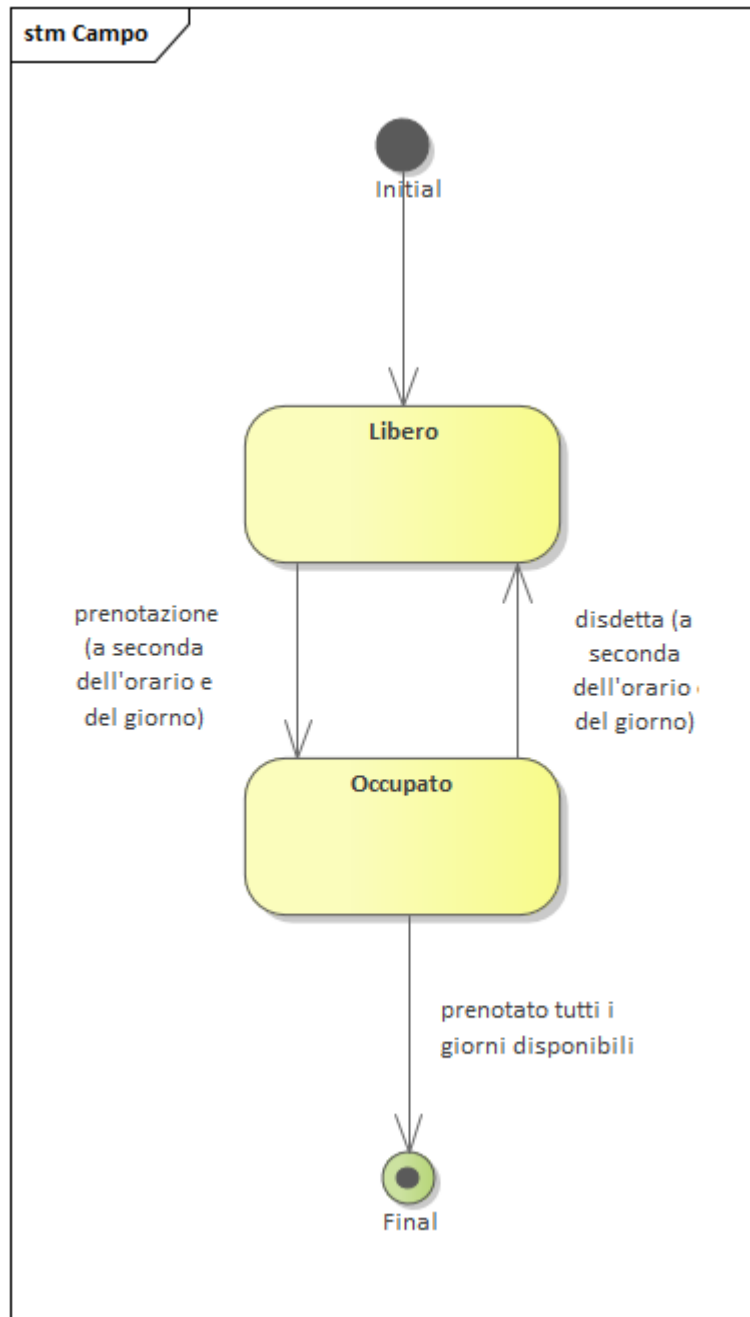


Figura 22: Macchina a stato campo

4.3.2 Partita

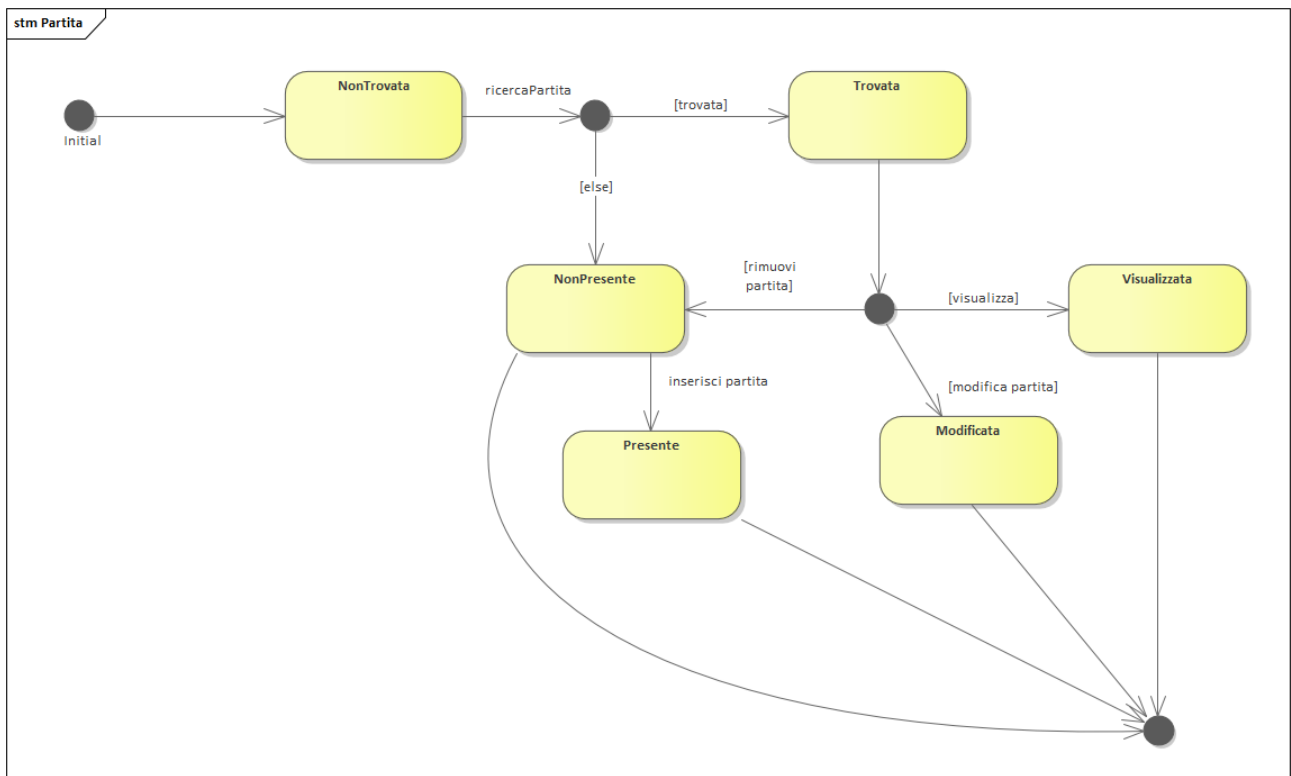


Figura 23: Macchina a stati partita

4.3.3 Prenotazione

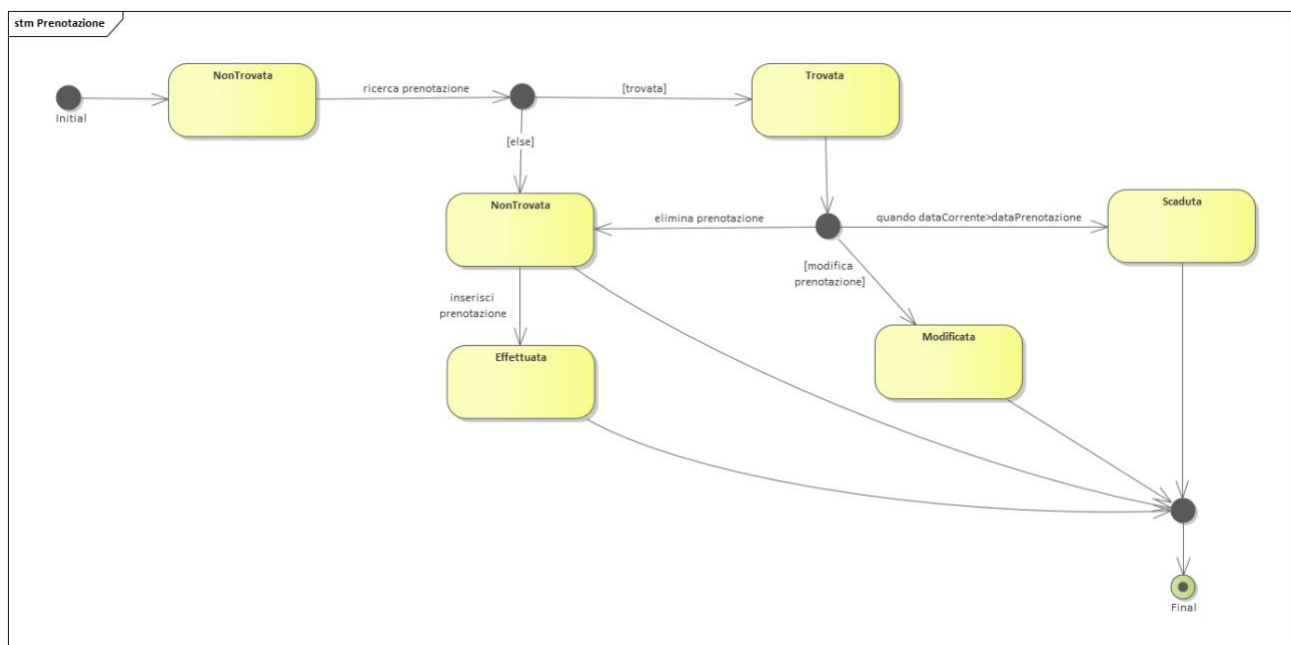


Figura 24: Macchina a stati prenotazione

4.3.4 Tesseramento

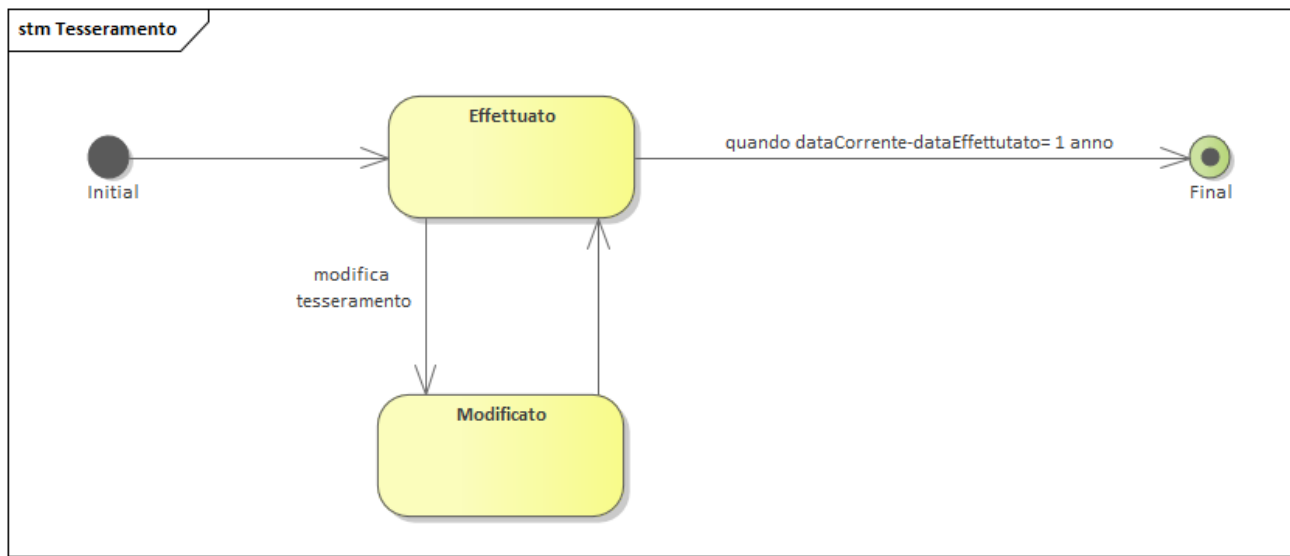


Figura 25: Macchina a stati tesseramento

4.4 Diagramma dei componenti

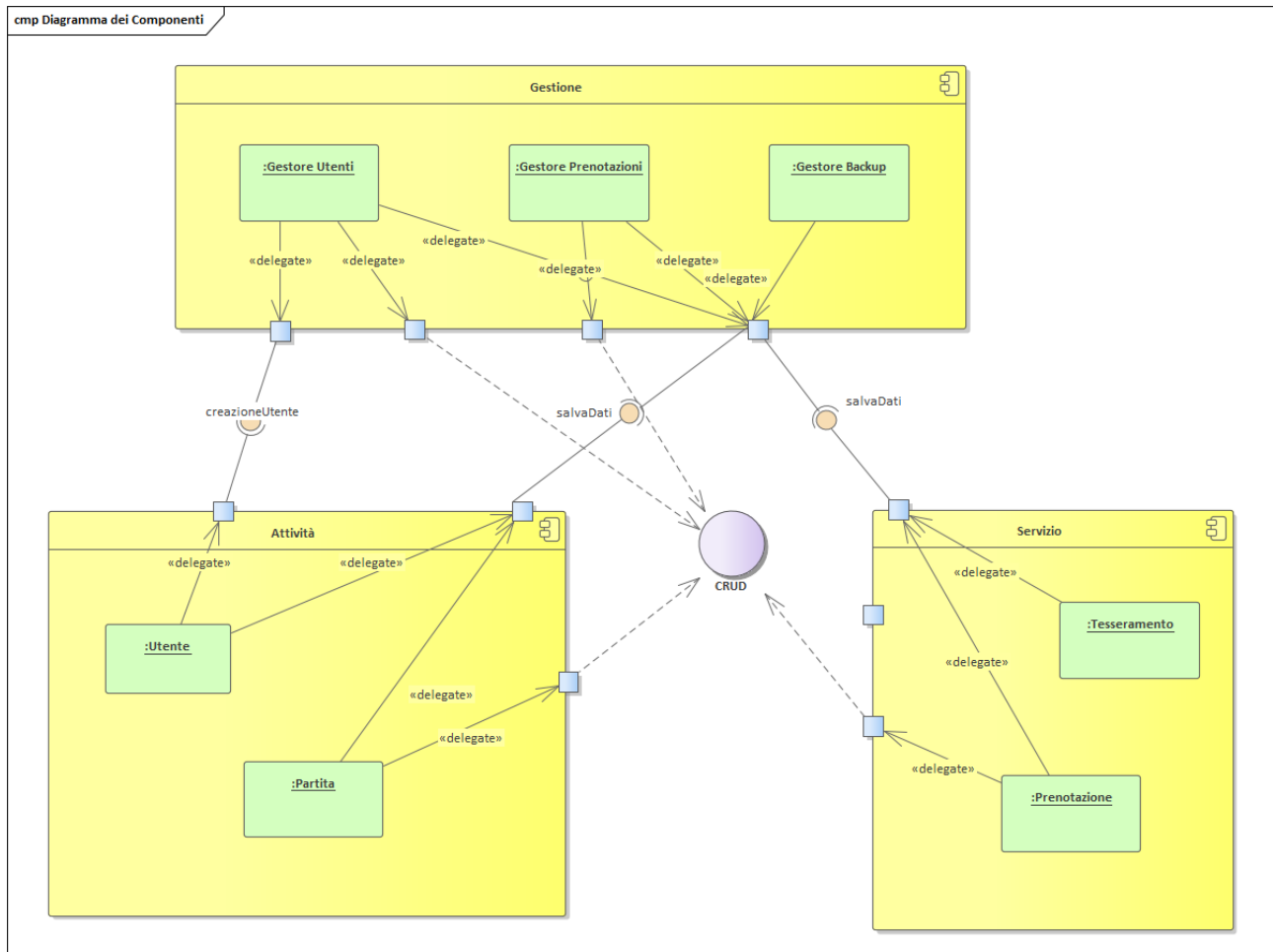


Figura 26: Diagramma dei componenti

Implementazione

L'implementazione l'abbiamo progettata tramite il diagramma di deployment.
Il diagramma deployment mostra in che modo il software viene distribuito nell'hardware.

5.1 Diagramma di deployment

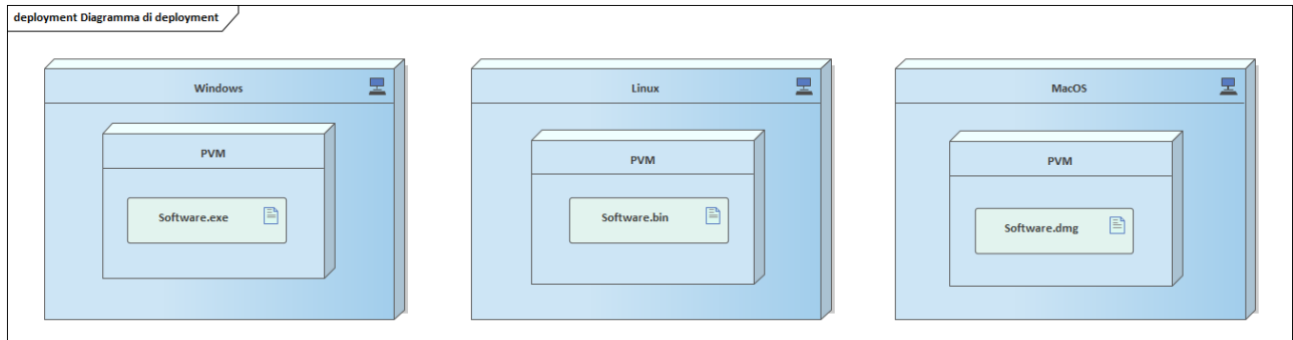


Figura 27: Diagramma di deployment

Mockup

Prima di realizzare le interfacce grafiche le abbiamo progettate, abbiamo quindi utilizzato dei mockup.

Per realizzare tali mockup abbiamo utilizzato il software Qt Designer.

6.1 Home

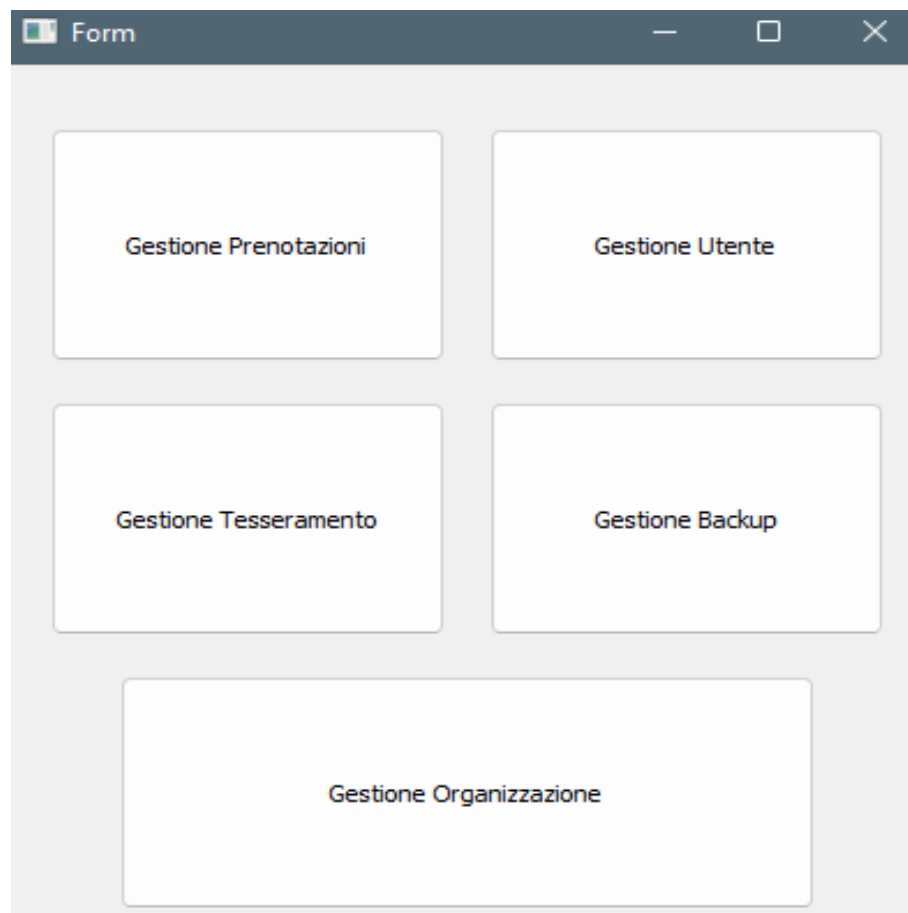
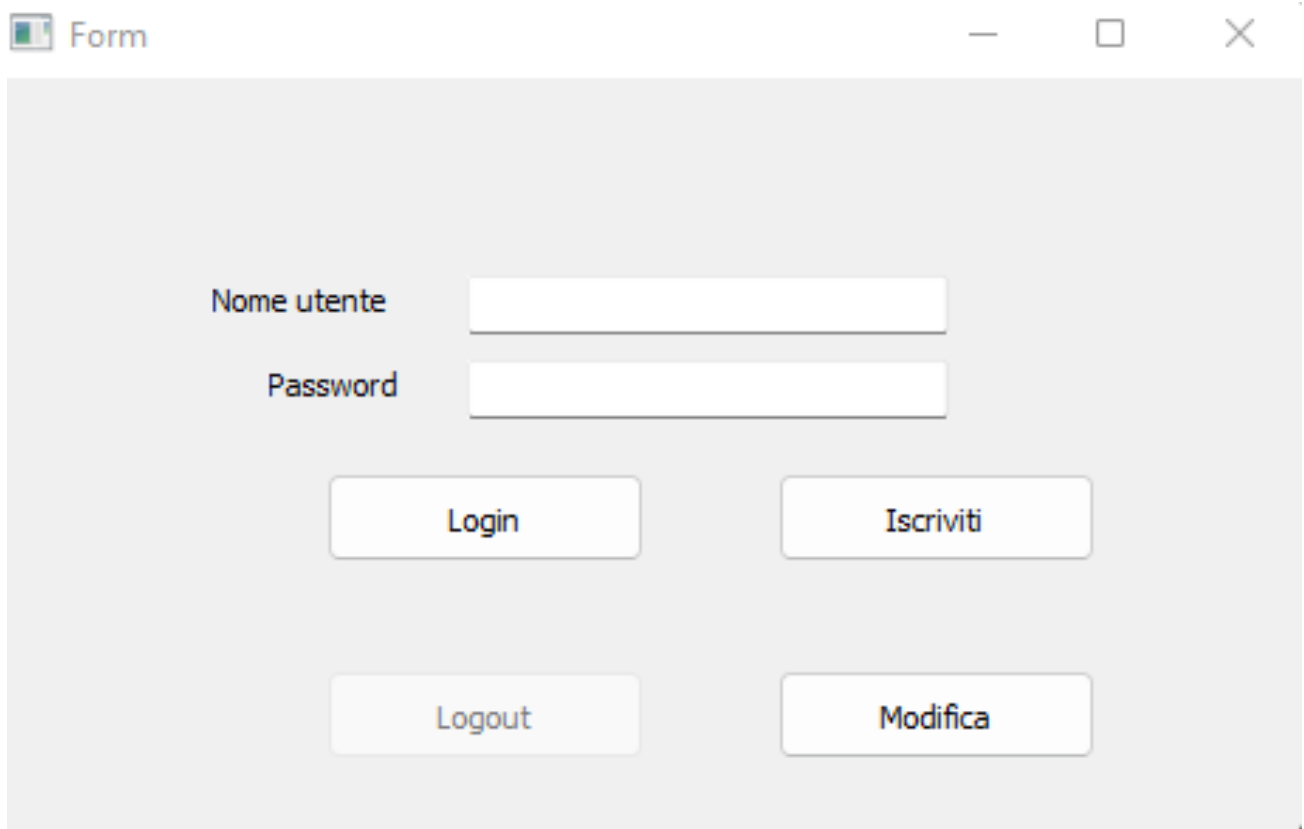


Figura 28: Vista home

6.2 Login



The image shows a screenshot of a web application window titled "Form". The window has a light gray background and contains the following elements:

- Two input fields: "Nome utente" (Username) and "Password".
- Four buttons: "Login", "Iscriviti" (Sign up), "Logout", and "Modifica" (Edit).

The "Nome utente" and "Password" labels are positioned to the left of their respective input fields. The "Login" and "Iscriviti" buttons are arranged horizontally below the input fields. The "Logout" and "Modifica" buttons are arranged horizontally below the "Login" and "Iscriviti" buttons.

Figura 29: Vista login

6.3 Conferma login

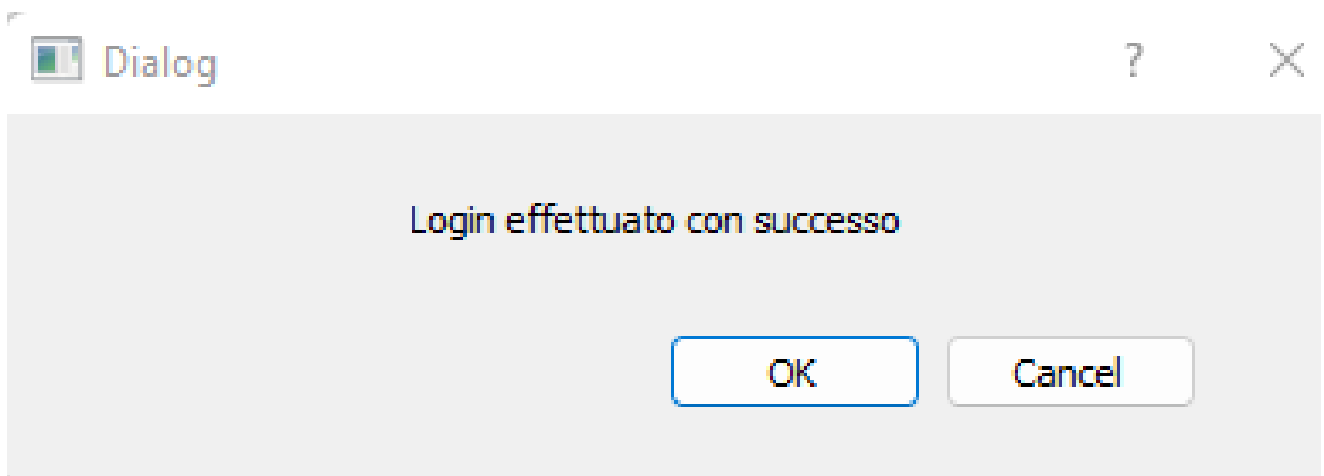


Figura 30: Vista conferma login

6.4 Conferma logout

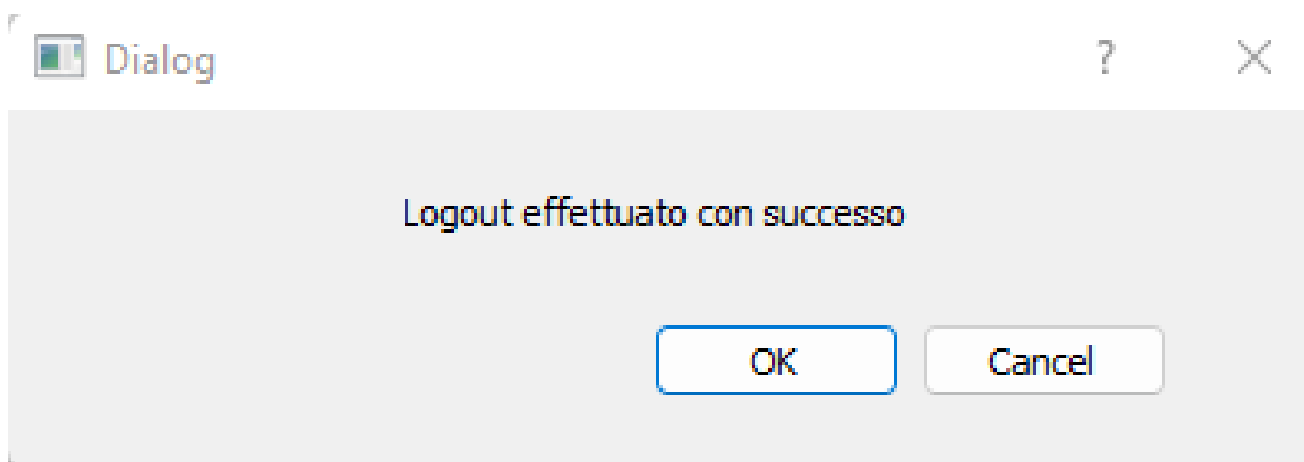
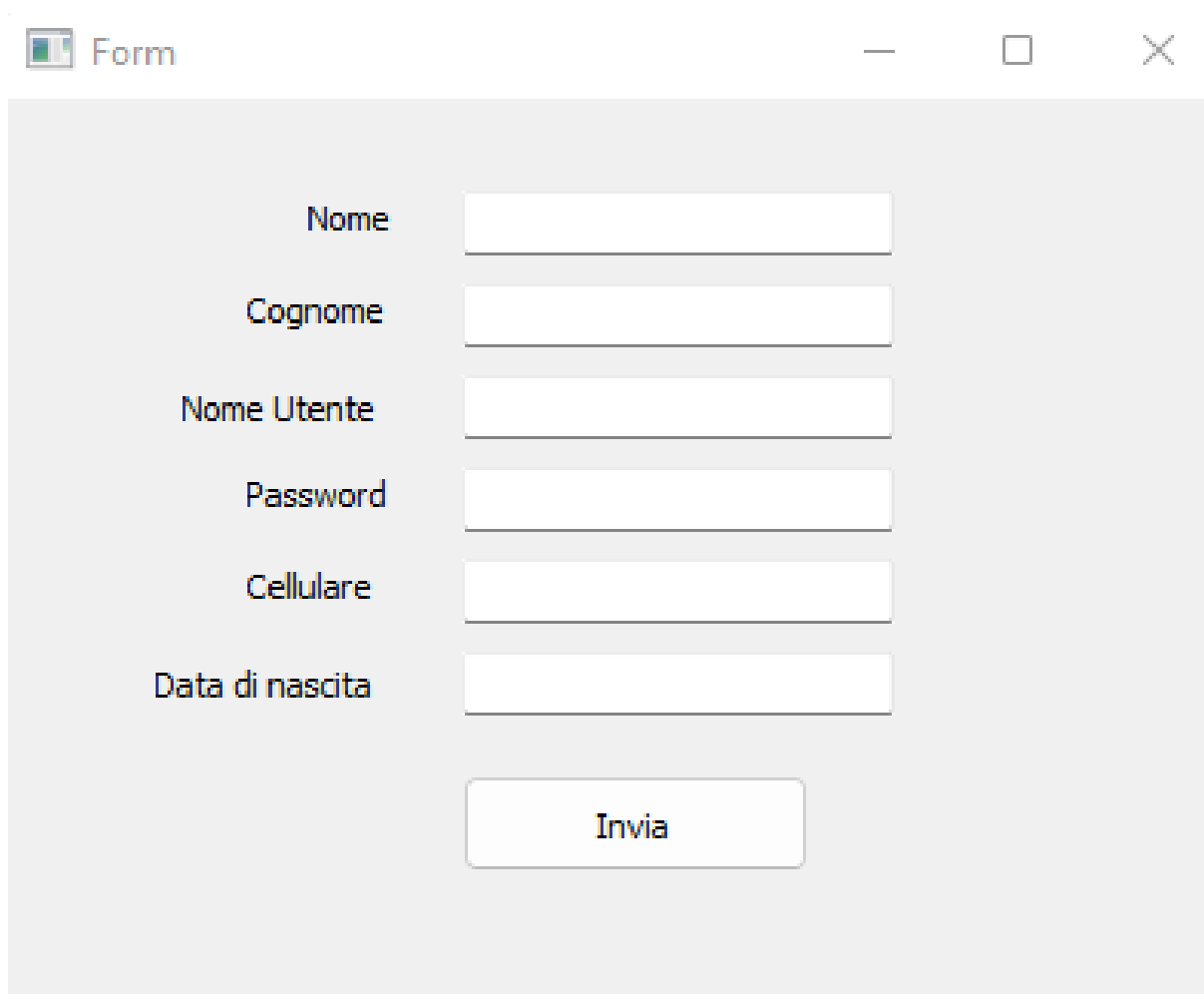


Figura 31: Vista conferma logout

6.5 Registrazione



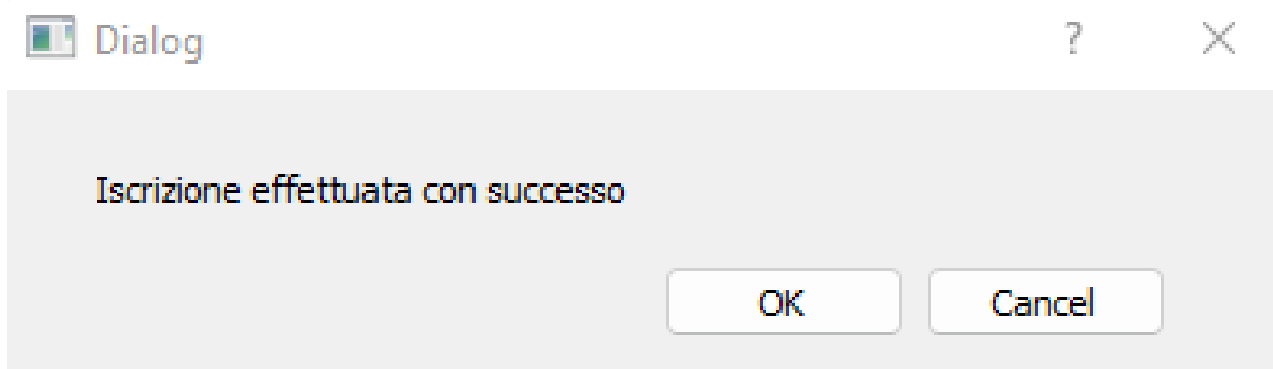
The screenshot shows a window titled "Form" with a standard Windows title bar (minimize, maximize, close buttons). The window contains a registration form with the following fields and labels:

- Nome
- Cognome
- Nome Utente
- Password
- Cellulare
- Data di nascita

Below the input fields is a button labeled "Invia".

Figura 32: Vista iscrizione

6.6 Conferma iscrizione



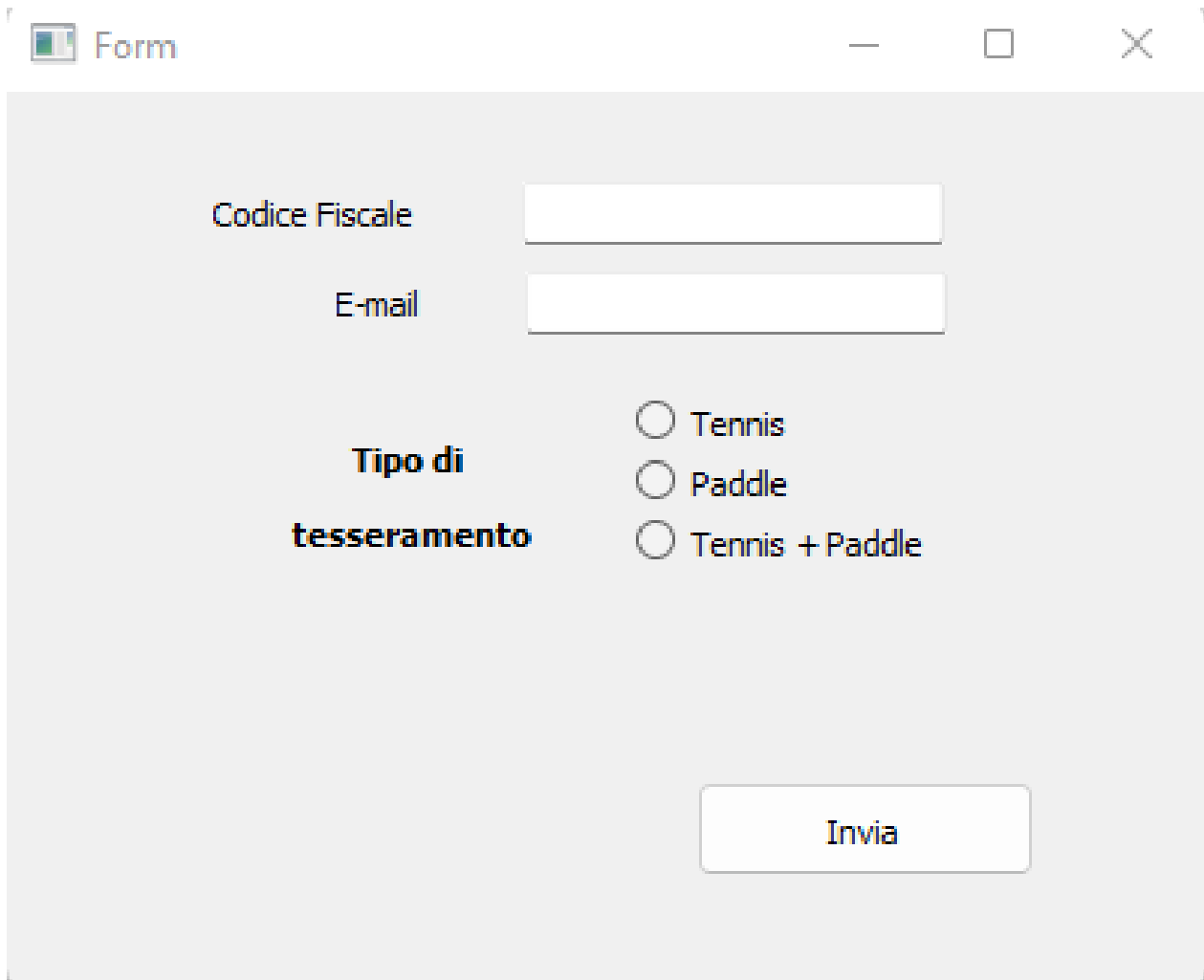
The screenshot shows a dialog box titled "Dialog" with a standard Windows title bar (help, close buttons). The dialog contains the following text and buttons:

Iscrizione effettuata con successo

At the bottom right, there are two buttons: "OK" and "Cancel".

Figura 33: Vista conferma iscrizione

6.7 Gestione tesseramento



Form

Codice Fiscale

E-mail

Tipo di tesseramento

☐ Tennis

☐ Paddle

☐ Tennis + Paddle

Figura 34: Vista gestione tesseramento

6.8 Conferma tesseramento

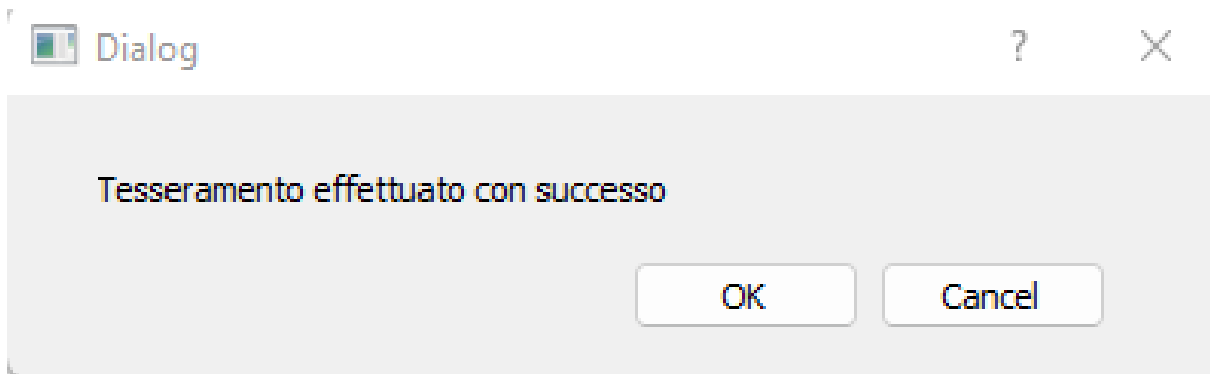


Figura 35: Vista conferma tesseramento

6.9 Gestione organizzazione

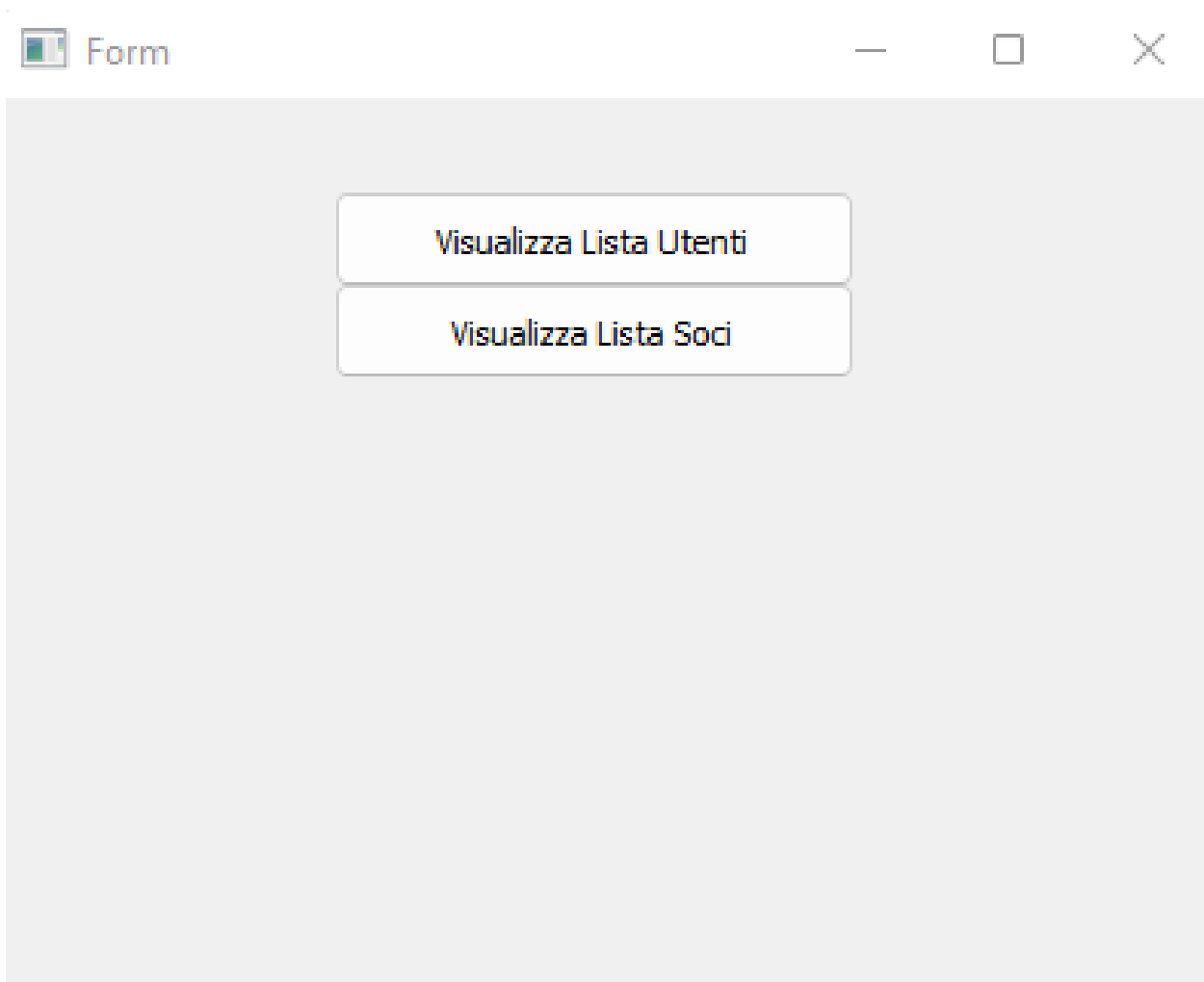


Figura 36: Vista gestione organizzazione

6.10 Visualizza lista soci

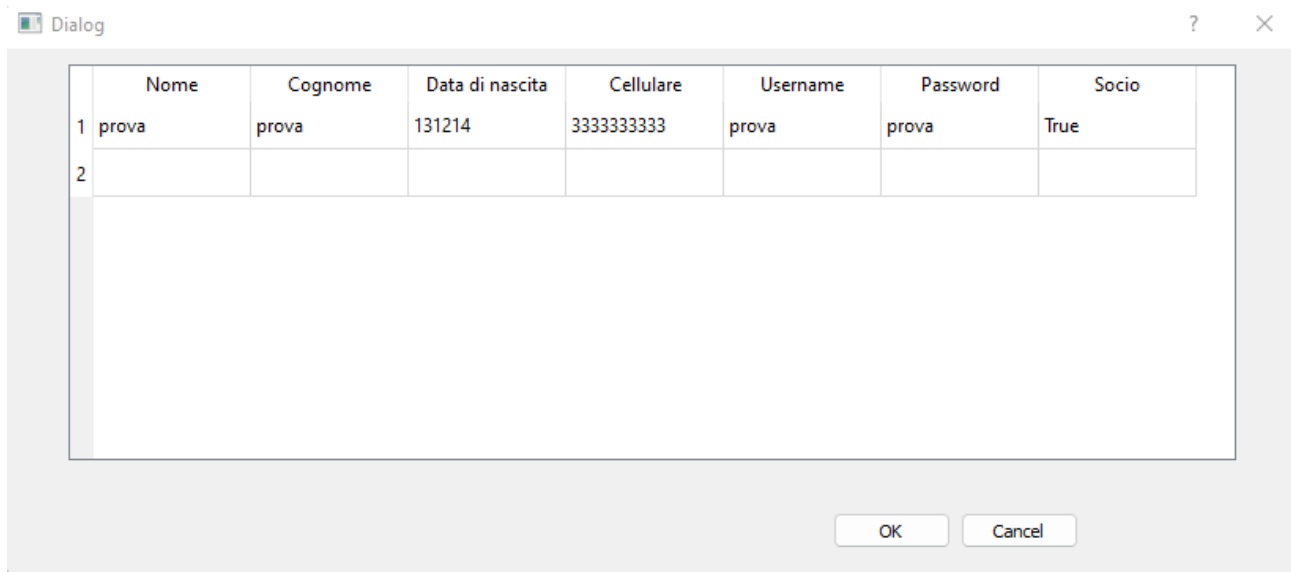


Figura 37: Vista lista soci

6.11 Visualizza lista utenti

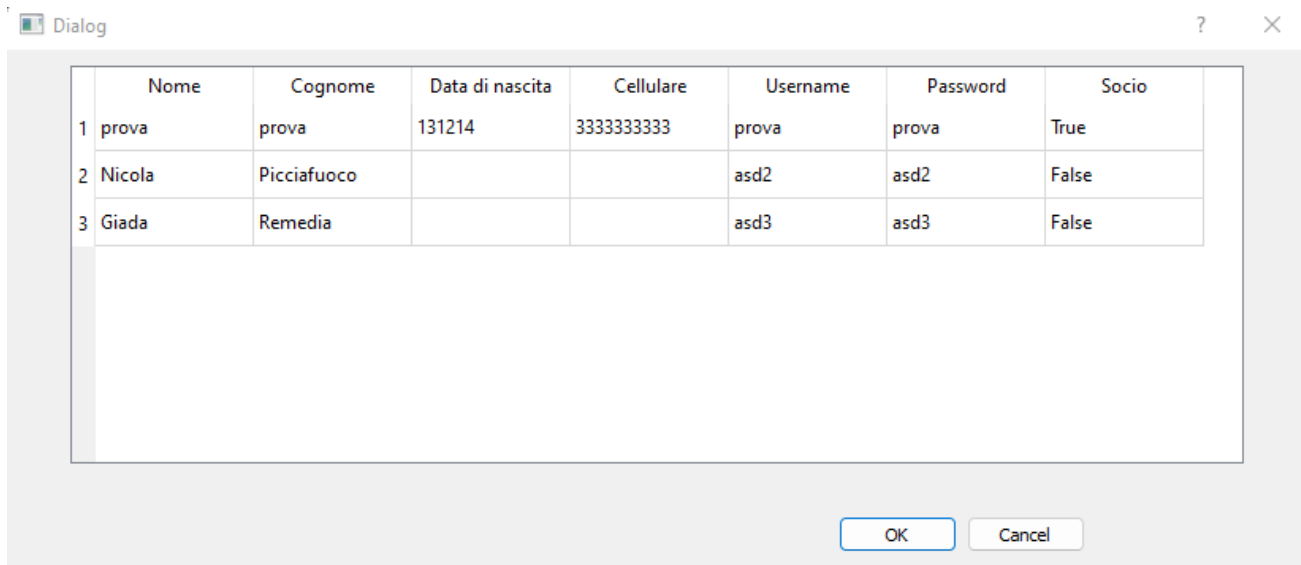


Figura 38: Vista utenti

6.12 Modifica dati registrazione

The screenshot shows a web form titled "Form" with the following fields and values:

Nome	xxx
Cognome	xxx
Nome Utente	asia
Password	••••
Cellulare	3333333333
Data di nascita	05052006

At the bottom of the form is a button labeled "Invia".

Figura 39: Vista cambia registrazione

6.13 Gestione prenotazioni

The screenshot shows a booking management interface titled "Form". It features a calendar for July 2022 on the left and a table of bookings on the right.

Calendar (luglio, 2022):

dom	lun	mar	mer	gio	ven	sab
26	27	28	29	30	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

Below the calendar are buttons: "Partite", "Inserisci", and "Elimina".

Booking Table:

	Terra Rossa Coperto	Terra Rossa	Erba Sintetica	Paddle
8-9				
9-10				
10-11				
11-12				
12-13				
13-14				
14-15				
15-16				
16-17				
17-18				
18-19				
19-20				
20-21				

Figura 40: Vista gestione prenotazioni

6.14 Aggiungi partita

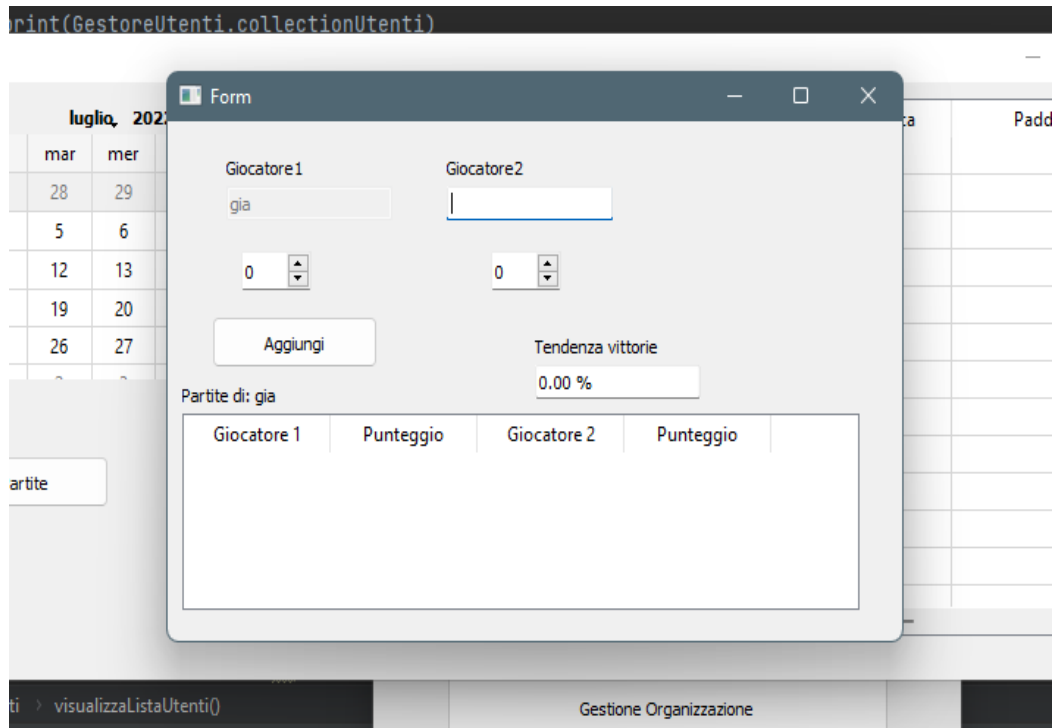


Figura 41: Vista aggiungi partita

6.15 Conferma modifica iscrizione

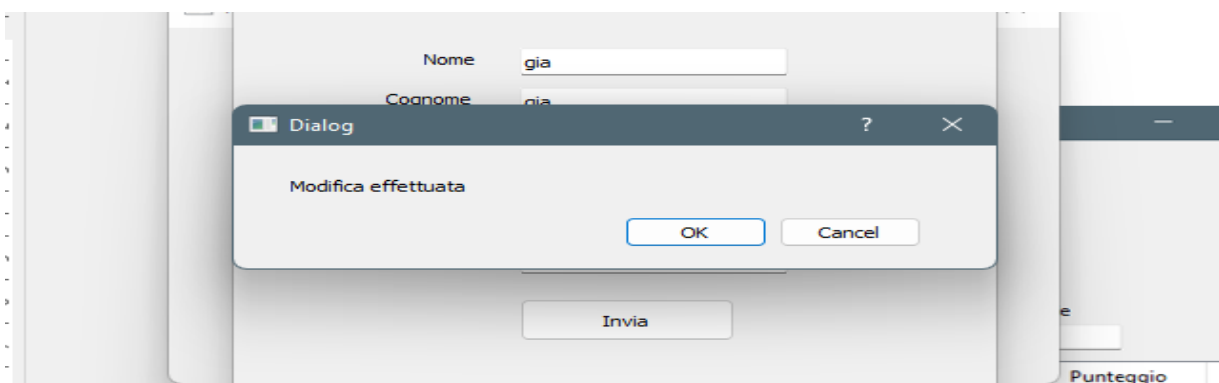


Figura 42: Vista conferma modifica iscrizione

6.16 Elimina prenotazione

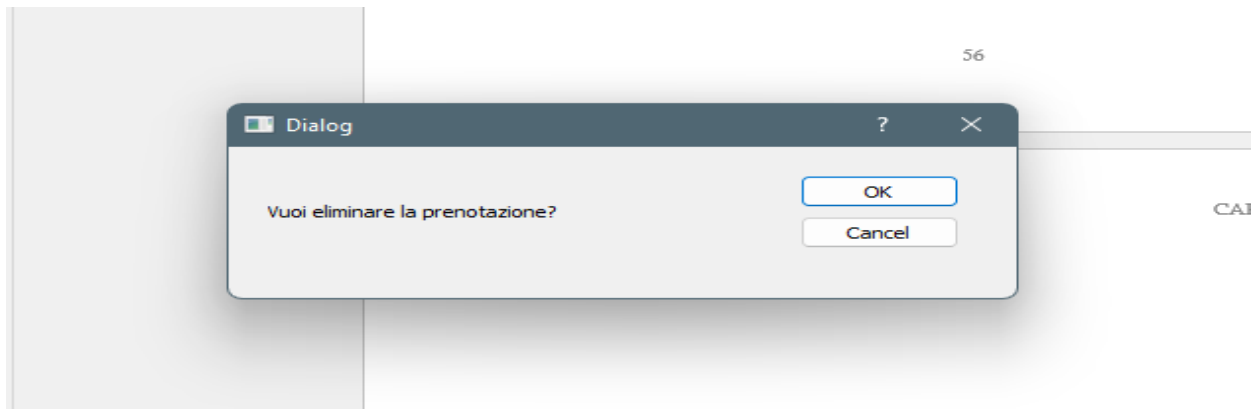


Figura 43: Vista conferma elimina prenotazione

Unit testing

In fine, abbiamo svolto degli unit testing per verificare che il codice funzionasse correttamente e per identificare possibili bug.

```

1  def testPrenotazione(self):
2      self.campo = GestorePrenotazioni.paddle
3      self.utente = Utente("Ivan", "prova", "08/06/1991", "333222556", "password", "asd")
4      self.prenotazioneuno = Prenotazione("2021 ,7 ,8", "10:00", self.campo, self.utente)
5      GestorePrenotazioni.collectionPrenotazioni.append(self.prenotazioneuno)
6      GestoreUtenti.utenteConnesso = self.utente
7      self.assertEqual(GestorePrenotazioni.cercaPrenotazione("2021 ,7 ,8", 2, 3), self.prenotazioneuno)

```

```

1  def testCopiaTesseramento(self):
2      self.utente = Utente("Ivan", "prova", "08/06/1991", "333222556", "password", "asd")
3      self.utente.setTesseramento("prova@google.com", "PCNVNI91H08I608S", "Paddle")
4      GestoreUtenti.collectionUtenti.append(self.utente)
5      GestoreUtenti.utenteConnesso = self.utente
6      self.utentemodificato = Utente("Mario", "Franchi", "08/06/1991", "333222556", "password", "asd")
7      GestoreUtenti.modificaUtente(self.utente, self.utentemodificato)
8      self.assertEqual(self.utente.getTesseramento(), self.utentemodificato.getTesseramento())

```

```

1  def testLogin(self):
2      self.utente = Utente("Ivan", "prova", "08/06/1991", "333222556", "password", "asd")
3      GestoreUtenti.setUtenteConnesso(self.utente)
4      self.assertEqual(GestoreUtenti.utenteConnesso.isAdmin, False)

```

Figura 44: Unit testing