

# **Centro Sportivo UNIVPM**

Francesco Maria Mosca, Enrico Piergallini,  
Niccolò Balloni, Francesco Concetti

23/07/2022



Progetto di ingegneria del software  
Anno Accademico 2020/2021

# Indice

<b>1 Descrizione del contesto</b>	<b>3</b>
<b>2 Intervista</b>	<b>4</b>
<b>3 Descrizione del linguaggio naturale</b>	<b>5</b>
3.1 Glossario dei termini . . . . .	6
<b>4 Analisi dei requisiti</b>	<b>7</b>
4.1 Requisiti funzionali . . . . .	7
<b>5 Design del sistema</b>	<b>11</b>
5.1 Diagramma dei casi d'uso . . . . .	11
5.2 Casi d'uso: Attori . . . . .	11
5.3 Casi d'uso: Gestione Dipendenti . . . . .	12
5.4 Casi d'uso: Sicurezza . . . . .	14
5.5 Casi d'uso: Gestioni Soci . . . . .	16
5.6 Casi d'uso: Gestione Liquidità . . . . .	19
5.7 Casi d'uso: Prenotazine Campi . . . . .	21
5.8 Gestioni Fornitori . . . . .	23
5.9 Matrice di mapping . . . . .	26
5.10 Diagrammi delle classi . . . . .	27
5.11 Diagramma Entità-Relazioni . . . . .	30
5.11.1 Schema ER . . . . .	30
5.11.2 Traduzione del modello ER nel modello relazionale . . . . .	31
5.12 Diagramma di Sequenza . . . . .	32
5.13 Diagrammi di attività . . . . .	40
<b>6 Dettagli Implementativi</b>	<b>59</b>
6.1 Mappa dell'architettura . . . . .	60
6.2 Graphic User Interface . . . . .	63
6.2.1 Login . . . . .	63
6.2.2 Homepage . . . . .	63
6.2.3 Gestione Dipendenti . . . . .	64
6.2.4 Sicurezza . . . . .	65
6.2.5 Soci . . . . .	66
6.2.6 Liquidità . . . . .	67
6.2.7 Campi . . . . .	68
6.2.8 Fornitori . . . . .	69
6.3 Test del software . . . . .	70

## 1 Descrizione del contesto

La nozione di centro sportivo è usata per riferirsi a ciò che è legato a un multiplot di sport. Può essere usato come additivo o come supporto a seconda del contesto. Si chiama centro sportivo il luogo dove ci sono diverse strutture e infrastrutture per la pratica delle discipline sportive. Di solito è un'area chiusa, sebbene spesso siano incluse aree esterne. Potrebbe essere un posto che ti permette di esercitarti pallacanestro, pallamano, pallavolo e ginnastica, ma può comprendere anche campi da tennis e da calcio, circuiti di atletica e piscine. Così, nella stessa zona, si concentra una grande varietà di attività sportive. Come per qualsiasi altro spazio dedicato all'attività umana, per aprire e mantenere un centro sportivo è obbligatorio rispettare una serie di regole ben definite dalle leggi in vigore in ogni paese e città, che vanno ben oltre gli obiettivi accademici della società. Infatti, l'adempimento di tutti i requisiti legali è essenziale per il raggiungimento di questi obiettivi in condizioni di sicurezza e benessere.

Le funzioni di un centro sportivo si raggruppano in due ambiti:

- Funzioni orientate verso la manutenzione (manutenzione della struttura e assunzione personale)
- Funzioni orientate verso i clienti (prenotazioni e abbonamenti)

Dal punto di vista organizzativo, il rapporto del centro sportivo con i clienti si configura come un servizio. Ogni centro Sportivo rispetta un orario di apertura, che tende a conciliarsi con quello del tempo libero, deve garantire un corretto funzionamento del servizio di prenotazione e deve ricordare al socio la data di scadenza del proprio abbonamento così da evitare malintesi al suo termine.



## 2 Intervista

PROGRAMMATORE: "Salve, di cosa ha bisogno?"

RICHIEDENTE: "Avrei bisogno di un programma che mi permetta di gestire in maniera semplice ed efficiente il mio centro sportivo."

PROGRAMMATORE: "Cosa deve fare il software?"

RICHIEDENTE: "Il Software deve permettere di poter gestire la prenotazione dei campi attraverso un calendario, inoltre deve includere la gestione della contabilità e delle risorse del mio centro sportivo. Infine, desidero che alcuni processi vengano automatizzati. Inoltre, necessito di una schermata di login che mi permetta di accedere al servizio."

PROGRAMMATORE: "Chi userà il software?"

RICHIEDENTE: "Io e i miei dipendenti dovremmo avere accesso al software attraverso delle credenziali di login?"

PROGRAMMATORE: "Quanti e quali campi saranno prenotabili?"

RICHIEDENTE: "Ci sono 2 campi da Tennis, entrambi in terra rossa, 2 campi da Padel ed 1 campo da calcio a 5 in erba."

PROGRAMMATORE: "Ok, come deve essere strutturato il sistema di prenotazioni?"

RICHIEDENTE: "Il sistema deve permettere di far prenotare i campi inserendo l'ora, nome e cognome dei giocatori per tennis e paddle, nel caso del calcetto basta solo un referente; dovrà essere anche possibile disdire la prenotazione. Il software dovrà inoltre controllare se i nomi inseriti nelle prenotazioni di tennis e padel figurano nella lista soci del centro sportivo."

PROGRAMMATORE: "In cosa consiste la Lista Soci?"

RICHIEDENTE: "E' una lista nella quale figurano i dati di coloro che, avendo versato una quota sociale annuale, hanno diritto a dei benefici; questa lista potrà essere aggiornata."

PROGRAMMATORE: "Come sarà gestita la Liquidità?"

RICHIEDENTE: "Il software dovrà includere una sezione nella quale sarà possibile tenere traccia dei movimenti (entrate ed uscite) relativi al Centro Sportivo."

PROGRAMMATORE: "Come deve essere strutturata l'interfaccia del sito?"

RICHIEDENTE: "Innanzitutto, io e i miei dipendenti vorremmo effettuare un login per accedere a tutte le funzionalità del sito. Una volta dentro, attraverso i bottoni presenti nel programma si potrà accedere a tutte le funzionalità implementate."

PROGRAMMATORE: "Mi perdoni, la sicurezza cosa dovrebbe fare?"

RICHIEDENTE: "E' la sezione che permette di creare gli account che verranno poi forniti ai dipendenti"

PROGRAMMATORE: "I fornitori invece?"

RICHIEDENTE: "In maniera analoga alla Lista Soci, è una lista contenente le informazioni relative ai vari fornitori, quali ID del fornitore, nome dell'azienda, sede, recapito telefonico email e del settore di cui si occupa."

PROGRAMMATORE: "Va bene, le farò sapere il prima possibile!"

RICHIEDENTE: "La ringrazio, arrivederci!"

### **3 Descrizione del linguaggio naturale**

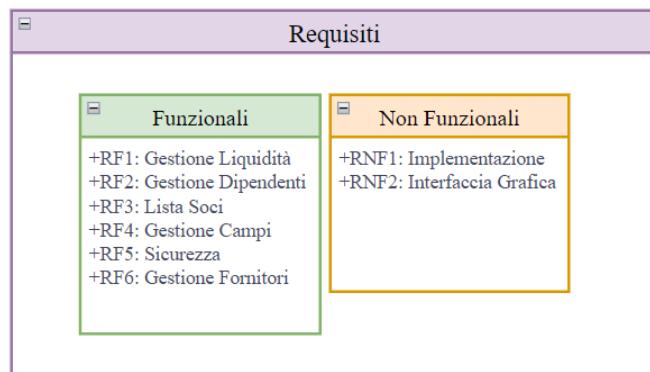
Il progetto proposto consiste nella realizzazione di un sistema informativo per la gestione di un centro sportivo. L'accesso al servizio è protetto da una schermata di login. Sarà possibile creare un nuovo utente cliccando sulla voce “crea utente” che si trova nella sezione “sicurezza”; Gli utenti che avranno accesso al software saranno i dipendenti del Centro Sportivo, i quali potranno scegliere un nome utente e una password. Una sezione del programma sarà dedicata alla gestione dei dipendenti , sarà possibile inserirli , modificarli ed eliminarli. Il software permetterà inoltre all'Utente di effettuare o disdire la prenotazione di un campo; L'Utente potrà infine interfacciarsi con la “Lista Soci” potendo aggiungere , modificare o rimuovere un membro. Vi sarà un funzionamento analogo per la lista dei fornitori del Centro. L'ultima sezione del programma è dedicata alla gestione della Liquidità, nella quale l'Utente inserirà le entrate e le uscite relative al Centro Sportivo.

### 3.1 Glossario dei termini

<u>Termine</u>	<u>Descrizione</u>	<u>Tipo</u>	<u>Sinonimi</u>
<i>Centro sportivo</i>	Complesso adibito allo svolgimento delle attività sportive.	Business	Nessuno
<i>Servizio</i>	Complesso di strutture offerte ai clienti: campo da tennis, campo da padel, campo da calcetto	Business	Nessuno
<i>Gestore</i>	Colui che amministra beni, servizi e finanze del centro sportivo	Business	Amministratore
<i>Cliente</i>	Colui che usufruisce dei servizi offerti dal centro sportivo.	Business	Nessuno
<i>Dipendente</i>	Dipendente del centro sportivo, utilizza il software e lavora presso la struttura.	Business	Lavoratore
<i>Socio</i>	Colui che pagando una quota annuale, ha diritto ad alcuni benefici.	Business	Nessuno
<i>Lista Soci</i>	Lista aggiornabile, nella quale figurano i nomi dei soci.	Business	Albo Sociale
<i>Campo</i>	Struttura adibita alla pratica dell'attività sportiva.	Business	Nessuno
<i>Prenotazione</i>	Atto mediante il quale viene riservato un campo ad uno o più clienti (o soci) per un certo tempo.	Tecnico	Nessuno
<i>Account</i>	Mezzo attraverso il quale il dipendente usufruisce dei servizi del software.	Tecnico	Nessuno

Glossario dei termini

## 4 Analisi dei requisiti



Schema generale dei requisiti

### 4.1 Requisiti funzionali

Requisiti Funzionali		
Gestione Liquidità	Gestione Dipendenti	Gestione Soci
+ RF1.1 Visualizza Movimenti + RF1.2 Inserimento Movimento + RF1.3 Elimina Movimento	+ RF2.1 Visualizza Dipendenti + RF2.2 Inserimento Dipendente + RF2.3 Elimina Dipendente + RF2.4 Modifica Dipendente	+ RF3.1 Visualizza Soci + RF3.2 Inserimento Socio + RF3.3 Modifica Socio + RF3.4 Elimina Socio
Prenotazione Campi	Sicurezza	Gestione Fornitori
+ RF4.1 Visualizza Campi + RF4.2 Prenotazione Campo + RF4.3 Elimina Prenotazione	+ RF5.1 Visualizza Utenti + RF5.2 Creazione Utente + RF5.3 Elimina Utente	+ RF6.1 Visualizza Fornitori + RF6.2 Inserimento Fornitore + RF6.3 Elimina Fornitore + RF6.4 Modifica Fornitore

Requisiti Funzionali

### **RF1: Gestione Liquidità**

Requisito	Descrizione
<b>RF1.1 Visualizza Movimenti</b>	il sistema dovrà consentire di visualizzare i movimenti effettuati
<b>RF1.2 Inserimento Movimento</b>	il sistema dovrà consentire l'inserimento di nuovi movimenti
<b>RF1.3 Elimina Movimento</b>	il sistema dovrà consentire eliminare un movimento esistente

### **RF2: Gestione Dipendenti**

Requisito	Descrizione
<b>RF2.1 Visualizza Dipendenti</b>	il sistema dovrà consentire di visualizzare i dipendenti esistenti
<b>RF2.2 Inserimento Dipendente</b>	il sistema dovrà consentire l'inserimento di un nuovo dipendente
<b>RF2.3 Elimina Dipendente</b>	il sistema dovrà consentire di rimuovere un dipendente
<b>RF2.4 Modifica Dipendente</b>	il sistema dovrà consentire di modificare i dati di un dipendente

### **RF3: Lista Soci**

Requisito	Descrizione
<b>RF3.1 Visualizza Soci</b>	il sistema dovrà consentire di visualizzare i soci esistenti
<b>RF3.2 Inserimento Socio</b>	il sistema dovrà consentire l'inserimento di un nuovo socio nella lista
<b>RF3.3 Modifica Socio</b>	il sistema dovrà consentire di modificare i dati di un socio presente nella lista
<b>RF3.4 Elimina Socio</b>	il sistema dovrà consentire di rimuovere un socio dalla lista

#### **RF4: Gestione Campi**

Requisito	Descrizione
<b>RF4.1 Visualizza Campi</b>	il sistema dovrà consentire di visualizzare i campi e le relative prenotazioni di un dato giorno
<b>RF4.2 Prenotazione Campo</b>	il sistema dovrà consentire l'inserimento di una prenotazione per un dato campo
<b>RF4.3 Elimina Prenotazione</b>	il sistema dovrà consentire di eliminare una data prenotazione

#### **RF5: Sicurezza**

Requisito	Descrizione
<b>RF5.1 Visualizza Utenti</b>	il sistema dovrà consentire di visualizzare gli utenti esistenti
<b>RF5.2 Creazione Utente</b>	il sistema dovrà consentire di creare un nuovo utente
<b>RF5.3 Elimina Utente</b>	il sistema dovrà consentire di rimuovere un utente esistente

#### **RF6: Gestione Fornitori**

Requisito	Descrizione
<b>RF6.1 Visualizza Fornitori</b>	il sistema dovrà consentire di visualizzare gli fornitori esistenti
<b>RF6.2 Inserimento Fornitore</b>	il sistema dovrà consentire l'inserimento di un nuovo Fornitore nella lista
<b>RF6.3 Modifica Fornitore</b>	il sistema dovrà consentire di modificare i dati di un Fornitore presente nella lista
<b>RF6.4 Elimina Fornitore</b>	il sistema dovrà consentire di rimuovere un Fornitore dalla lista



Descrizione dei Requisiti non funzionali

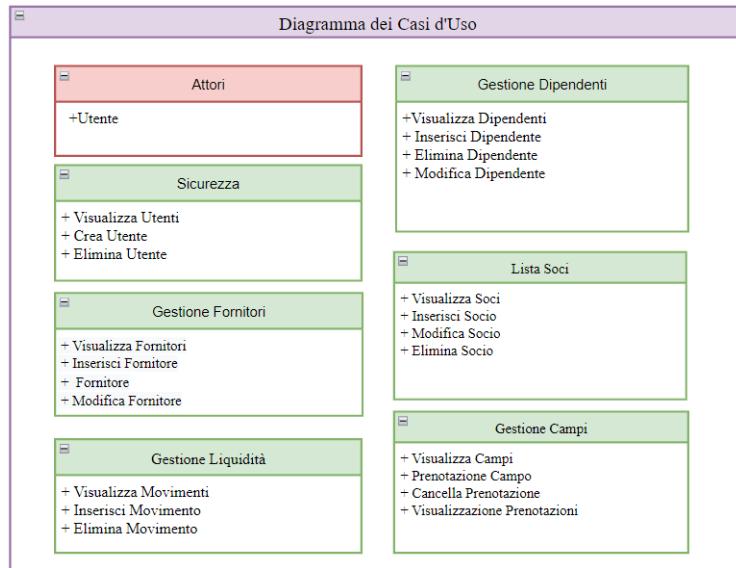
Requisito	Descrizione
<b>RNF1 Implementazione</b>	il sistema dovrà essere realizzato con la tecnologia Python 3
<b>RNF2 Interfaccia Grafica</b>	il sistema dovrà essere dotato di un'interfaccia grafica

Schema dei requisiti non funzionali

## 5 Design del sistema

### 5.1 Diagramma dei casi d'uso

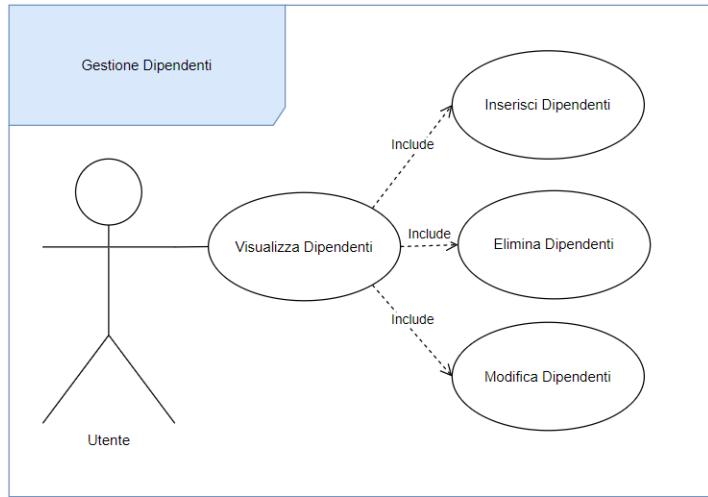
Una volta definiti e spiegati per bene i requisiti del nostro sistema, andiamo a definire i casi d'uso, ovvero gli scenari possibili che si potranno verificare durante l'utilizzo del nostro software da parte degli utenti.



### 5.2 Casi d'uso: Attori

Nel nostro software abbiamo un solo attore che interagisce con il sistema, chiamato "Utente". L'utente non è altro che un dipendente del centro sportivo, il quale dispone di un username e di una password da lui scelti ed è in grado di accedere ad ogni funzione del software.

### 5.3 Casi d'uso: Gestione Dipendenti



#### 1) Visualizza Dipendente

Attori	Utente
Pre-Condizioni	L'utente dovrà essere correttamente loggato.
Post-Condizioni	Nel caso si scelga di inserire , modificare o di eliminare un Dipendente si passa al relativo caso d'uso
Sequenza degli eventi principale	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando l'utente correttamente loggato vuole visualizzare la lista dei Dipendenti</li> <li>2. si entra nella sezione Gestione Dipendenti e il sistema mostra a schermo la lista</li> </ol>
Sequenza alternativa	Nessuna

2) Inserisci Dipendente

<b>Attori</b>	Utente
<b>Pre-Condizioni</b>	L'utente dovrà essere correttamente loggato. Il Dipendente non deve essere già presente
<b>Post-Condizioni</b>	il Nuovo Dipendente è inserito nell'elenco
<b>Sequenza degli eventi principale</b>	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando l'utente correttamente loggato vuole inserire un nuovo Dipendente</li> <li>2. Vengono Richiesti i dati del Dipendente</li> <li>3. il sistema controlla che il Dipendente non sia già presente a sistema</li> <li>4. Il Dipendente viene aggiunto all'elenco</li> </ol>
<b>Sequenza alternativa</b>	Da 3) se non è possibile inserire il nuovo dipendente , dopo una notifica , si torna al punto 1)

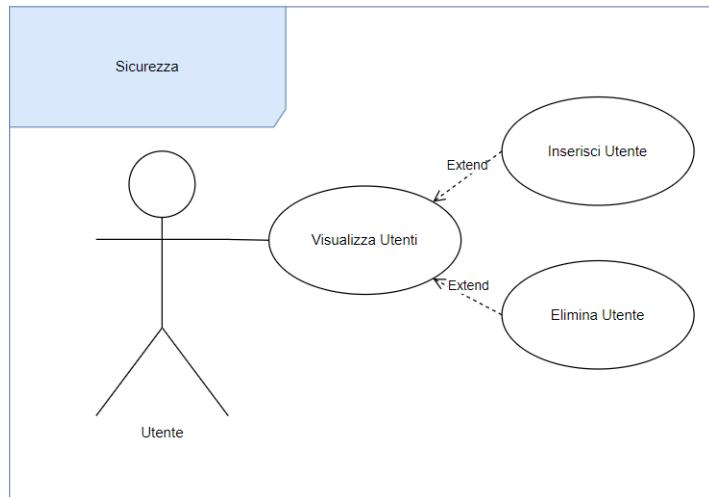
3) Elimina Dipendente

<b>Attori</b>	Utente
<b>Pre-Condizioni</b>	L'utente dovrà essere correttamente loggato. Il Dipendente deve esistere a sistema
<b>Post-Condizioni</b>	il Dipendente è rimosso nell'elenco
<b>Sequenza degli eventi principale</b>	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando l'utente correttamente loggato vuole eliminare una Dipendente esistente</li> <li>2. Viene richiesto il codice fiscale del Dipendente</li> <li>3. il sistema controlla che il Dipendente sia già presente a sistema</li> <li>4. Il Dipendente viene eliminato dall'elenco</li> </ol>
<b>Sequenza alternativa</b>	Da 3) se non vi è riscontro con un Dipendente , dopo una notifica , si torna al punto 1)

#### 4) Modifica Dipendente

Attori	Utente
<i>Pre-Condizioni</i>	L'utente dovrà essere correttamente loggato. Il Dipendente deve esistere a sistema
<i>Post-Condizioni</i>	i Dati del Dipendente nell'elenco vengono modificati
<i>Sequenza degli eventi principale</i>	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando l'utente correttamente loggato vuole modificare i dati di un Dipendente esistente</li> <li>2. Viene richiesto il codice fiscale del Dipendente</li> <li>3. il sistema controlla che il Dipendente sia già presente a sistema</li> <li>4. i dati del Dipendente vengono visualizzati e sono modificabili</li> <li>5. i dati del Dipendente vengono Aggiornati a sistema</li> </ol>
<i>Sequenza alternativa</i>	Da 3) se non vi è riscontro con un Dipendente , dopo una notifica , si torna al punto 1)

#### 5.4 Casi d'uso: Sicurezza



## 1) Visualizza Utenti

Attori	Utente
Pre-Condizioni	L'utente dovrà essere correttamente loggato.
Post-Condizioni	Nel caso si scelga di inserire o di eliminare un'utente si passa al relativo caso d'uso
Sequenza degli eventi principale	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando l'utente correttamente loggato vuole visualizzare la lista Utenti</li> <li>2. si entra nella sezione Sicurezza e il sistema mostra a schermo la lista</li> </ol>
Sequenza alternativa	Nessuna

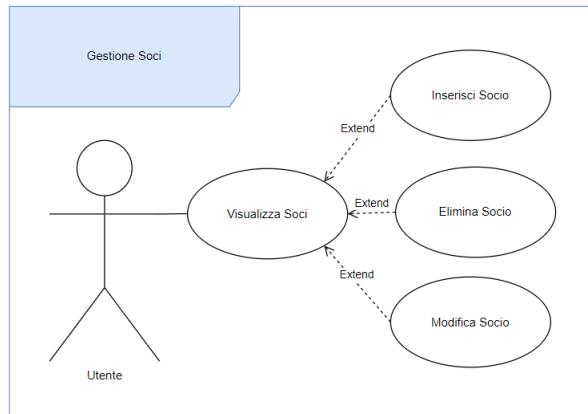
## 2) Inserisci Utente

Attori	Utente
Pre-Condizioni	L'utente dovrà essere correttamente loggato.
Post-Condizioni	il nuovo Utente è aggiunto a sistema
Sequenza degli eventi principale	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando l'utente correttamente loggato , entrato nella sezione della sicurezza vuole aggiungere un nuovo utente</li> <li>2. Si inserisci il CF dell'Utente</li> <li>3. Il sistema controlla che il CF sia di un dipendente e non già presente come utente.</li> <li>4. Si inseriscono i Dati dell'utente (username,password)</li> <li>5. il sistema controlla che l'utente non sia già presente.</li> <li>6. il nuovo utente è inserito a sistema.</li> </ol>
Sequenza alternativa	<p>Da 3) se il controllo è negativo , dopo una notifica , si torna al punto 1).</p> <p>da 5) se non è possibile creare l'utente si torna al punto 1)</p>

### 3) Elimina Utente

Attori	Utente
<i>Pre-Condizioni</i>	L'utente dovrà essere correttamente loggato e dovrà aver inserito correttamente la chiave di accesso alla sezione.
<i>Post-Condizioni</i>	l'Utente selezionato viene rimosso dal sistema
<i>Sequenza degli eventi principale</i>	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando l'utente correttamente loggato , entrato nella sezione della sicurezza vuole rimuovere un utente esistente</li> <li>2. Viene richiesto l'username dell'utente</li> <li>3. il sistema controlla che il Utente sia esistente a sistema</li> <li>4. l'Utente viene viene eliminato e ora non è più autorizzato a usare il sistema</li> </ol>
<i>Sequenza alternativa</i>	Da 3) se l'Utente non viene trovato a sistema, dopo una notifica , si torna al punto 1)

## 5.5 Casi d'uso: Gestioni Soci



1) Visualizza Soci

Attori	Utente
Pre-Condizioni	L'utente dovrà essere correttamente loggato.
Post-Condizioni	Nel caso si scelga di inserire , modificare o di eliminare un Socio si passa al relativo caso d'uso
Sequenza degli eventi principale	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando l'utente correttamente loggato vuole visualizzare la lista Soci</li> <li>2. si entra nella sezione Gestione Soci e il sistema mostra a schermo la lista</li> </ol>
Sequenza alternativa	Nessuna

2) Inserisci Socio

Attori	Utente
Pre-Condizioni	L'utente dovrà essere correttamente loggato. Il Socio non deve essere già presente a sistema
Post-Condizioni	Il Nuovo Socio è inserito nell'elenco
Sequenza degli eventi principale	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando l'utente correttamente loggato vuole inserire un nuovo Socio.</li> <li>2. Vengono Richiesti i dati del Socio.</li> <li>3. Il sistema controlla che il Socio non sia già presente a sistema.</li> <li>4. Il Socio viene aggiunto all'elenco.</li> </ol>
Sequenza alternativa	Da 3) se non è possibile inserire il nuovo Socio , dopo una notifica , si torna al punto 1)

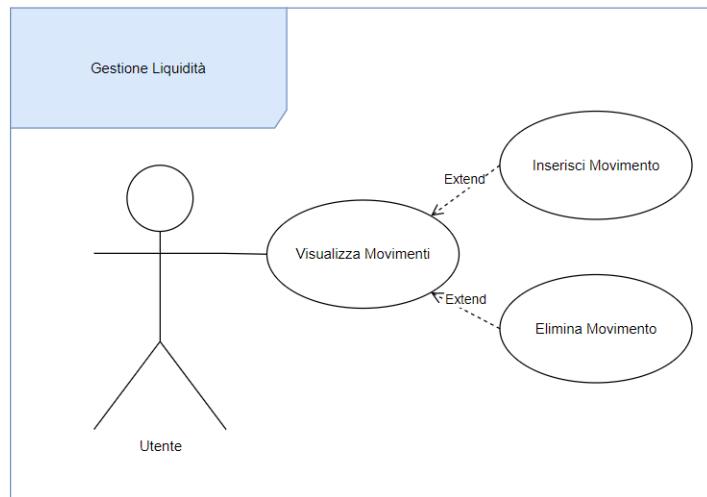
3) Elimina Socio

<b>Attori</b>	Utente
<b>Pre-Condizioni</b>	L'utente dovrà essere correttamente loggato. Il Socio deve esistere a sistema
<b>Post-Condizioni</b>	Il Dipendente è rimosso nell'elenco
<b>Sequenza degli eventi principale</b>	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando l'utente correttamente loggato vuole eliminare una Socio esistente</li> <li>2. Viene richiesto il codice fiscale del Socio</li> <li>3. il sistema controlla che il Socio sia già presente a sistema</li> <li>4. Il Socio viene eliminato dall'elenco</li> </ol>
<b>Sequenza alternativa</b>	Da 3) se non vi è riscontro con un Socio , dopo una notifica , si torna al punto 1)

4) Modifica Socio

<b>Attori</b>	Utente
<b>Pre-Condizioni</b>	L'utente dovrà essere correttamente loggato. Il Socio deve esistere a sistema
<b>Post-Condizioni</b>	I Dati del Socio nell'elenco vengono modificati
<b>Sequenza degli eventi principale</b>	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando l'utente correttamente loggato vuole modificare i dati di un Socio esistente</li> <li>2. Viene richiesto l'ID del Socio</li> <li>3. il sistema controlla che il Socio sia già presente a sistema</li> <li>4. I Dati del Socio vengono modificati</li> </ol>
<b>Sequenza alternativa</b>	Da 3) se non vi è riscontro con un Socio , dopo una notifica , si torna al punto 1)

## 5.6 Casi d'uso: Gestione Liquidità



### 1) Visualizza Movimenti

Attori	Utente
Pre-Condizioni	L'utente dovrà essere correttamente loggato.
Post-Condizioni	Nel caso si scelga di inserire o di eliminare un movimento si passa al relativo caso d'uso
Sequenza degli eventi principale	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando l'utente correttamente loggato vuole visualizzare la lista dei Movimenti</li> <li>2. si entra nella sezione Gestione liquidità e il sistema mostra a schermo la lista</li> </ol>
Sequenza alternativa	Nessuna

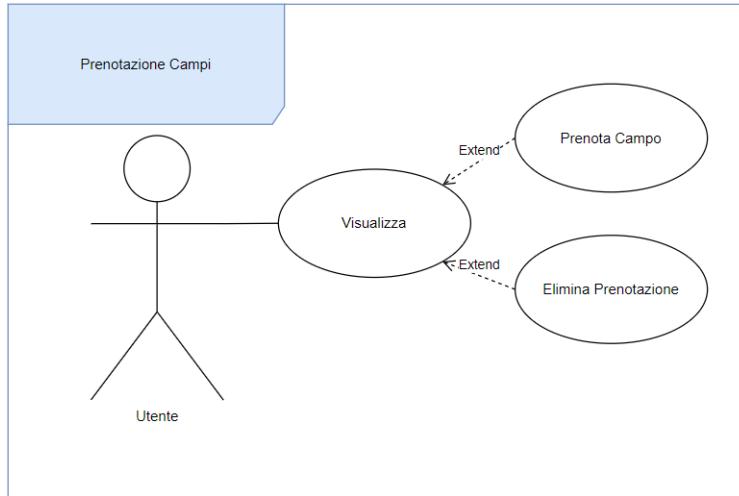
2) Inserisci Movimento

<b>Attori</b>	Utente
<b>Pre-Condizioni</b>	L'utente dovrà essere correttamente loggato
<b>Post-Condizioni</b>	Il Movimento è salvato a sistema
<b>Sequenza degli eventi principale</b>	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando l'utente correttamente loggato vuole inserire un nuovo movimento.</li> <li>2. Si inseriscono i dati del movimento.</li> <li>3. Il sistema controlla che i dati siano accettabili.</li> <li>4. Il movimento viene salvato a sistema</li> </ol>
<b>Sequenza alternativa</b>	Da 3) se i dati non risultano accettabili si torna al punto 1)

3) Elimina Movimento

<b>Attori</b>	Utente
<b>Pre-Condizioni</b>	L'utente dovrà essere correttamente loggato
<b>Post-Condizioni</b>	Il Movimento è Eliminato a sistema
<b>Sequenza degli eventi principale</b>	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando l'utente correttamente loggato vuole eliminare un movimento esistente.</li> <li>2. l'utente inserisce l'ID del movimento che vuole eliminare.</li> <li>3. il sistema verifica che l'ID sia di un movimento esistente.</li> <li>4. il Movimento è eliminato a sistema</li> </ol>
<b>Sequenza alternativa</b>	Da 3) se l'ID non risultano esistente si torna al punto 1)

## 5.7 Casi d'uso: Prenotazione Campi



### 1) Visualizza Campi

Attori	Utente
Pre-Condizioni	L'utente dovrà essere correttamente loggato.
Post-Condizioni	Nel caso si sceglie di effettuare una prenotazione o di eliminarla si passa al relativo caso d'uso
Sequenza degli eventi principale	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando l'utente correttamente loggato vuole visualizzare le prenotazioni di un dato giorno</li> <li>2. Si sceglie la data</li> <li>3. Il sistema mostra a schermo le ore con relative prenotazioni del giorno stabilito</li> </ol>
Sequenza alternativa	Nessuna

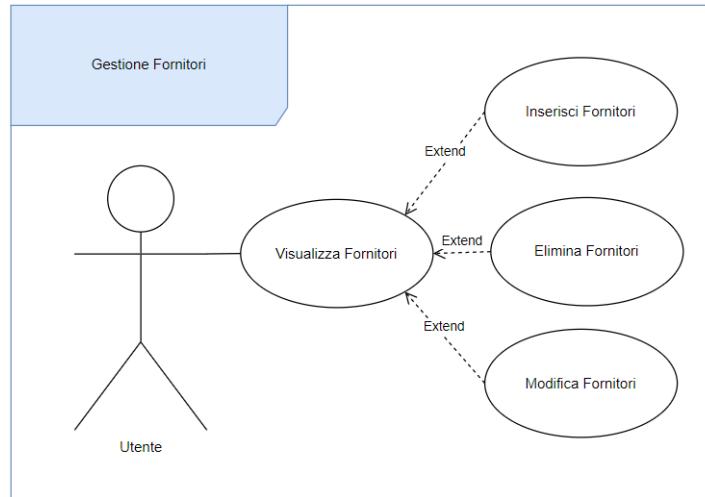
## 2) Prenota Campo

Attori	Utente
Pre-Condizioni	L'utente dovrà essere correttamente loggato. l'ora desiderata deve essere Prenotabile
Post-Condizioni	il Campo risulta prenotato all'ora selezionata
Sequenza degli eventi principale	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando l'utente correttamente loggato vuole Prenotare un campo</li> <li>2. Si sceglie il giorno.</li> <li>3. Visualizzati a schermo i campi del giorno si sceglie Campo ed Orario.</li> <li>4. Il sistema controlla che il campo scelto sia prenotabile all'ora scelta.</li> <li>5. vengono richiesti i Dati dei giocatori.</li> <li>6. Se è presente un Socio il sistema invia l'email di conferma prenotazione</li> <li>7. la prenotazione è salvata a sistema</li> </ol>
Sequenza alternativa	da 4) se il campo scelto risulta già prenotato , in questo caso dopo una notifica d'errore si torna al punto 3)

## 3) Elimina Prenotazione

Attori	Utente
Pre-Condizioni	L'utente dovrà essere correttamente loggato. l'ora desiderata deve essere Prenotata
Post-Condizioni	il Campo risulterà non prenotato all'ora selezionata
Sequenza degli eventi principale	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando l'utente correttamente loggato vuole eliminare la Prenotazione di un campo</li> <li>2. Si sceglie il giorno</li> <li>3. Visualizzati a schermo i campi del giorno si sceglie Campo ed Orario.</li> <li>4. La prenotazione è rimossa a sistema.</li> </ol>
Sequenza alternativa	Nessuna

## 5.8 Gestioni Fornitori



### 1) Visualizza Fornitori

Attori	Utente
Pre-Condizioni	L'utente dovrà essere correttamente loggato.
Post-Condizioni	Nel caso si scelga di inserire , modificare o di eliminare un Fornitore si passa al relativo caso d'uso
Sequenza degli eventi principale	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando l'utente correttamente loggato vuole visualizzare la lista dei Fornitori</li> <li>2. si entra nella sezione Gestione Fornitori e il sistema mostra a schermo la lista</li> </ol>
Sequenza alternativa	Nessuna

2) Inserisci Fornitore

<b>Attori</b>	Utente
<b>Pre-Condizioni</b>	L'utente dovrà essere correttamente loggato. il Fornitore non deve essere già presente
<b>Post-Condizioni</b>	il Nuovo Fornitore è inserito nell'elenco
<b>Sequenza degli eventi principale</b>	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando l'utente correttamente loggato vuole inserire un nuovo Fornitore</li> <li>2. Vengono Richiesti i dati del Fornitore</li> <li>3. il sistema controlla che il Fornitore non sia già presente a sistema</li> <li>4. Il Fornitore viene aggiunto all'elenco</li> </ol>
<b>Sequenza alternativa</b>	Da 3) se non è possibile inserire il nuovo Fornitore, dopo una notifica , si torna al punto 1)

3) Elimina Fornitore

<b>Attori</b>	Utente
<b>Pre-Condizioni</b>	L'utente dovrà essere correttamente loggato. Il Fornitore deve esistere a sistema
<b>Post-Condizioni</b>	il Fornitore è rimosso nell'elenco
<b>Sequenza degli eventi principale</b>	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando l'utente correttamente loggato vuole eliminare una Fornitore esistente</li> <li>2. Viene richiesto il codice fiscale del Fornitore</li> <li>3. il sistema controlla che il Fornitore sia già presente a sistema</li> <li>4. Il Fornitore viene eliminato dall'elenco</li> </ol>
<b>Sequenza alternativa</b>	Da 3) se non vi è riscontro con un Fornitore , dopo una notifica , si torna al punto 1)

#### 4) Modifica Fornitore

<b>Attori</b>	<b>Utente</b>
<b>Pre-Condizioni</b>	L'utente dovrà essere correttamente loggato. Il Fornitore deve esistere a sistema
<b>Post-Condizioni</b>	i Dati del Fornitore nell'elenco vengono modificati
<b>Sequenza degli eventi principale</b>	<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando l'utente correttamente loggato vuole modificare i dati di un Fornitore esistente</li> <li>2. Viene richiesto l'ID del Fornitore</li> <li>3. Il sistema controlla che il Fornitore sia già presente a sistema</li> <li>4. i dati del Fornitore vengono modificati a sistema</li> </ol>
<b>Sequenza alternativa</b>	Da 3) se non vi è riscontro con un Fornitore , dopo una notifica , si torna al punto 1)

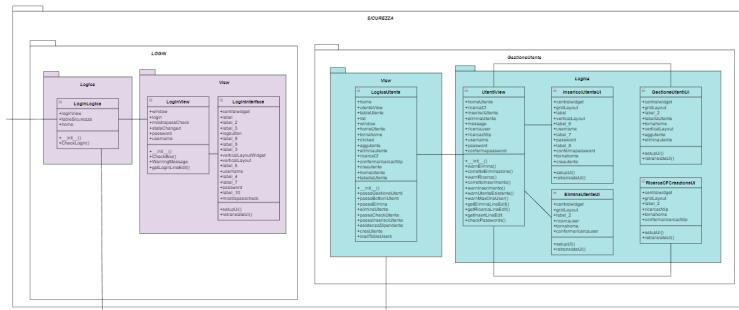
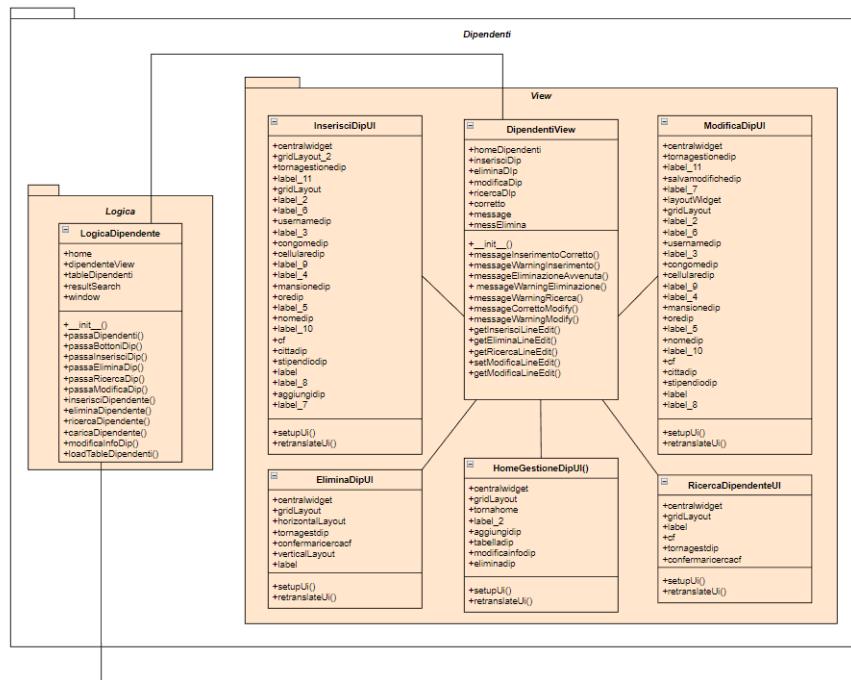
## 5.9 Matrice di mapping

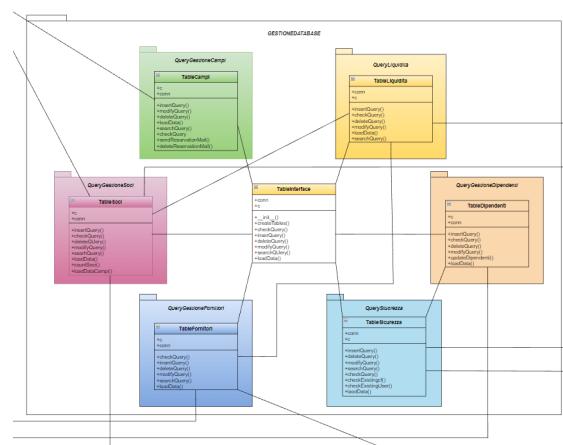
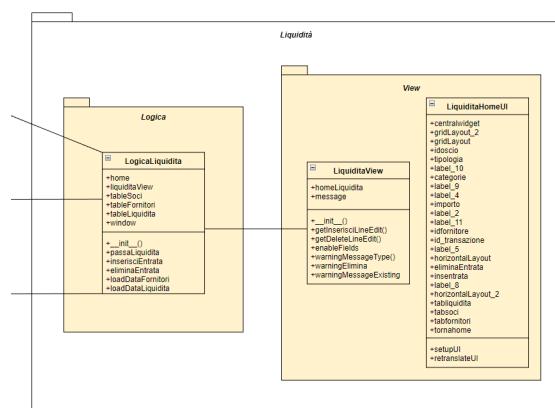
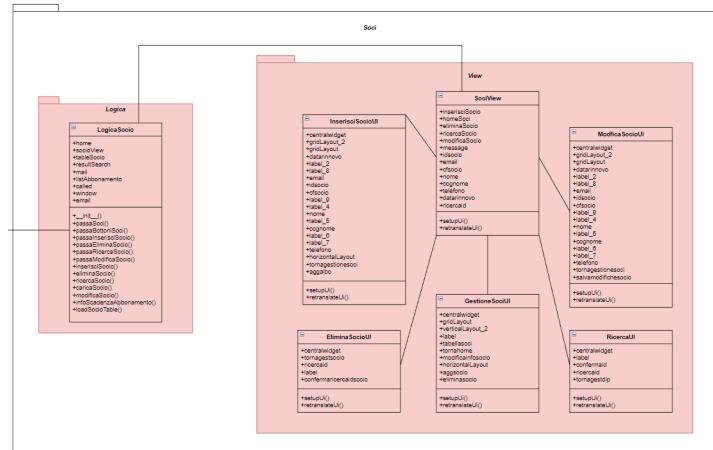
Di seguito è riportata la matrice di mapping tra requisiti funzionali e casi d'uso

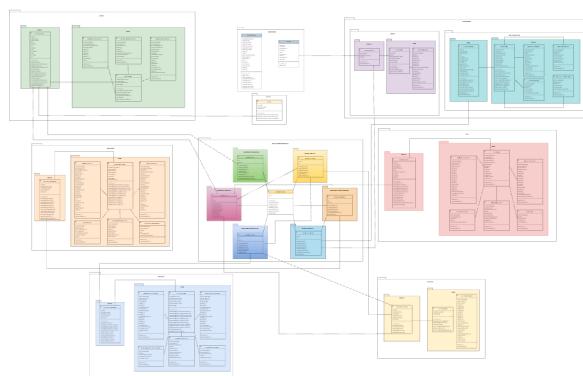
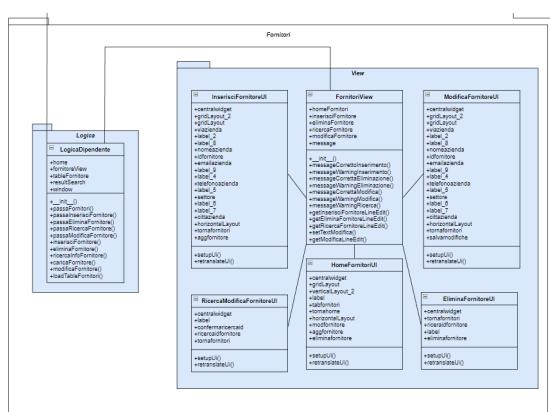
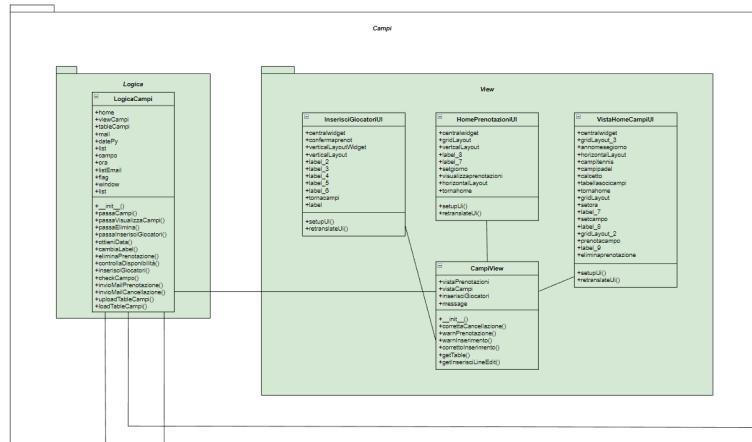
	RF1.1 Visualizza Movimenti	RF1.2 Inserimento Movimenti	RF1.3 Elimina Movimenti	RF2.1 Visualizza Dipendenze	RF2.2 Inserimento Dipendenze	RF2.3 Elimina Dipendenze	RF2.4 Modifica Dipendenze	RF3.1 Visualizza Sessi	RF3.2 Inserimento Sessi	RF3.3 Elimina Sessi	RF3.4 Visualizza Campi	RF4.1 Visualizza Campi	RF4.2 Modifica Campi	RF5.1 Visualizza Utenti	RF5.2 Creazione Utente	RF5.3 Elimina Utente	RF6.1 Inserimento Femmine	RF6.2 Elimina Femmine	RF6.3 Elimina Femmine	RF6.4 Modifica Femmine	
Visualizza Movimenti	X	X	X																		
Inserisci Movimento				X	X	X	X														
Elimina Movimento								X	X	X	X	X	X	X	X	X	X	X	X	X	X
Visualizza Dipendenze																					
Inserisci Dipendenze																					
Elimina Dipendente																					
Modifica Dipendente																					
Visualizza Sessi																					
Inserisci Sesso																					
Modifica Sesso																					
Elimina Sesso																					
Visualizza Campi																					
Inserisci Campo																					
Elimina Pretestosse																					
Visualizza Utenti																					
Crea Utente																					
Elimina Utente																					
Visualizza Femmine																					
Inserisci Femmine																					
Modifica Femmine																					

## 5.10 Diagrammi delle classi

Il diagramma delle classi mostra le differenti classi coinvolte nel software e come esse sono collegate. Per ogni classe sono mostrati gli attributi e i metodi principali.





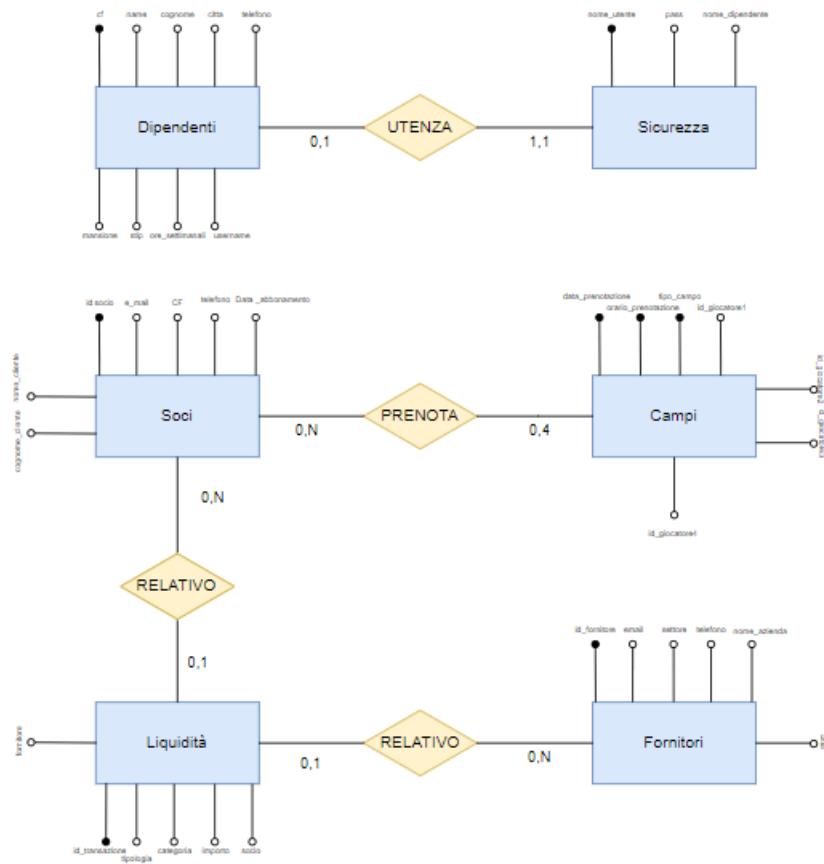


## Diagramma delle classi complessivo

## 5.11 Diagramma Entità-Relazioni

### 5.11.1 Schema ER

Lo schema ER mostra le entità coinvolte nel sistema e le relazioni fra loro intercorrenti.

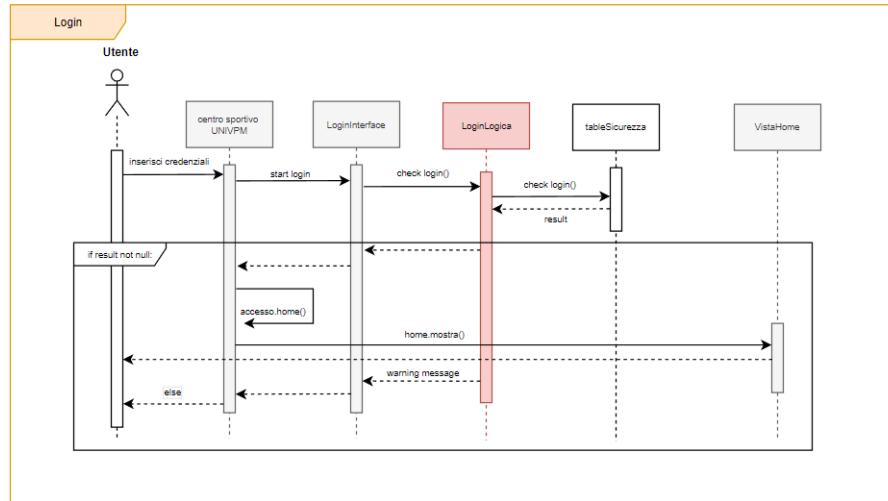


### 5.11.2 Traduzione del modello ER nel modello relazionale

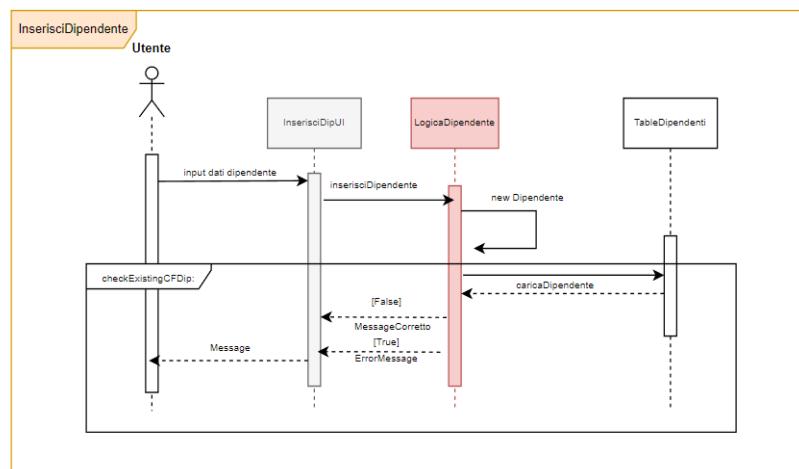
<i>Entità_Relazione</i>	<i>Traduzione</i>	<i>Vincoli di Riferimento</i>
<i>Dipendenti</i>	Dipendenti( <i>cf</i> , name, cognome, citta, telefono, mansione, ore_settimanali, stip, username)	username sicurezza.nome_utente
<i>Sicurezza</i>	Sicurezza( <i>nome_utente</i> , pass, nome_dipendente)	nome_dipendente dipendenti.cf
<i>Fornitori</i>	Fornitori( <i>id_fornitore</i> , nome_azienza, email, telefono, settore, citta, via)	*
<i>Soci</i>	Soci( <i>id_socio</i> , e_mail, CF, nome_cliente, cognome_cliente, telefono, Data_abbonamento)	*
<i>Campi</i>	Campi( <i>data_prenotazione</i> , <i>orario_prenotazione</i> , <i>tipo_campo</i> , <i>id_giocatore1</i> , <i>id_giocatore2</i> , <i>id_giocatore3</i> , <i>id_giocatore4</i> )	*
<i>Liquidità</i>	Liquidità( <i>id_transazione</i> , tipologia, categoria, importo, socio, fornitore)	socio socio.id_socio fornitori fornitori.id_fornitore

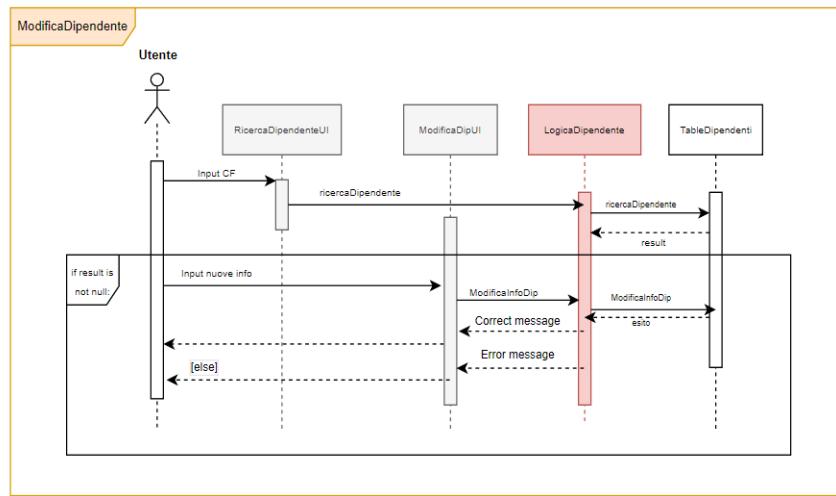
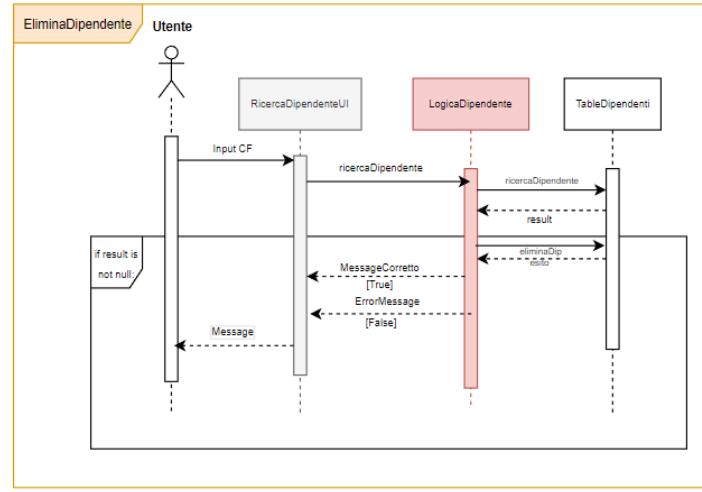
## 5.12 Diagramma di Sequenza

Diagramma di sequenza relativo all'autenticazione.

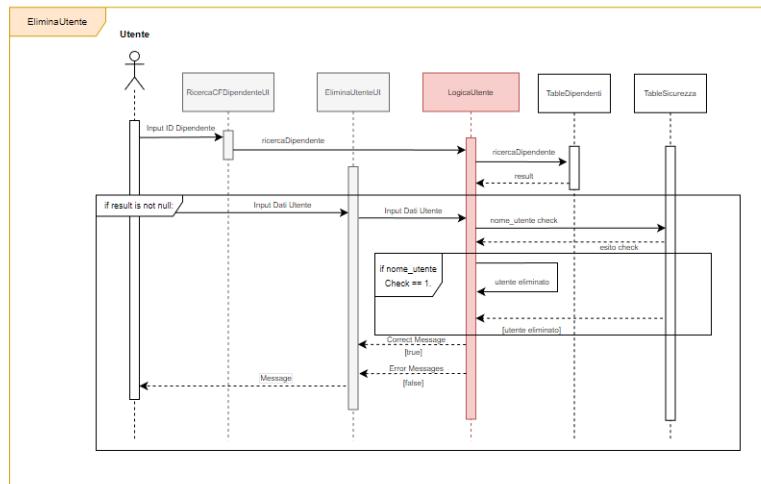
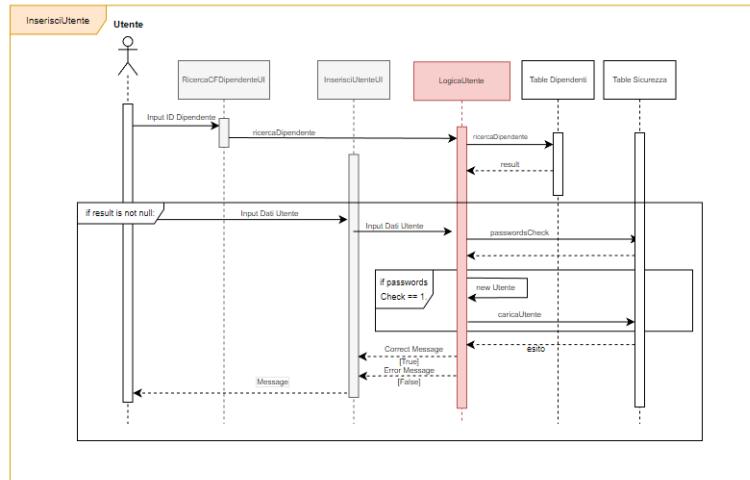


Diagrammo di sequenza relativi alla gestione dei dipendenti.

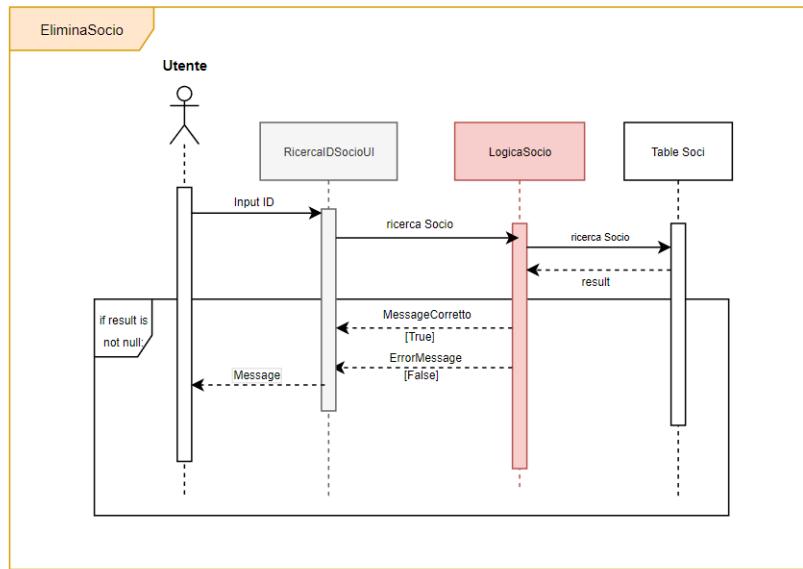
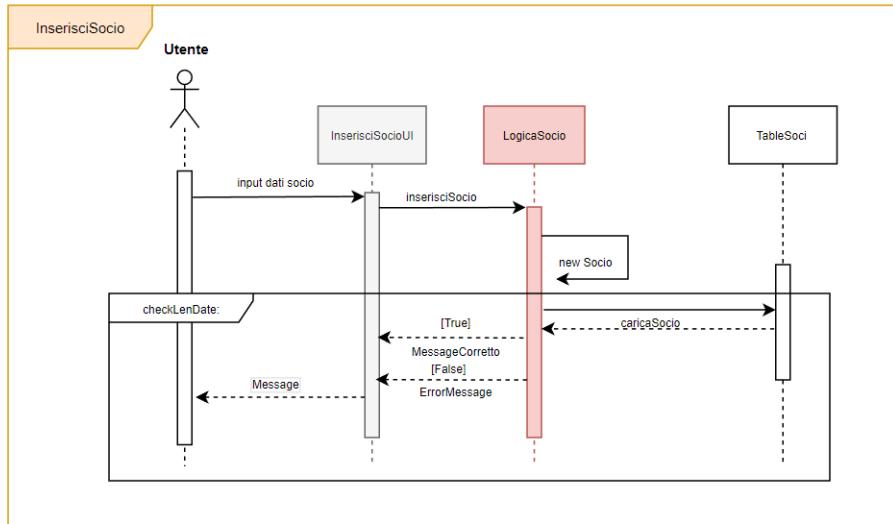


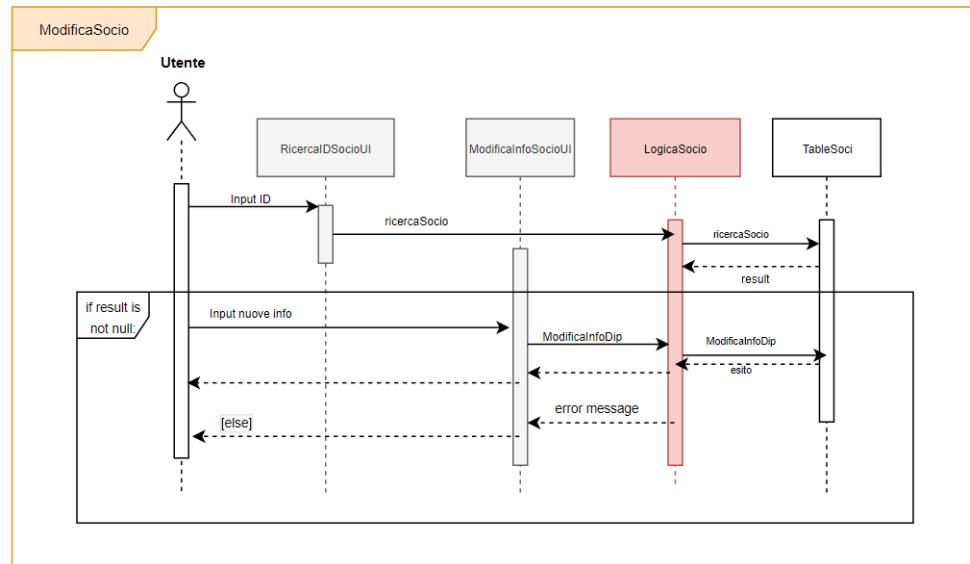


## Diagrammi della sequenza relativa alla Sicurezza

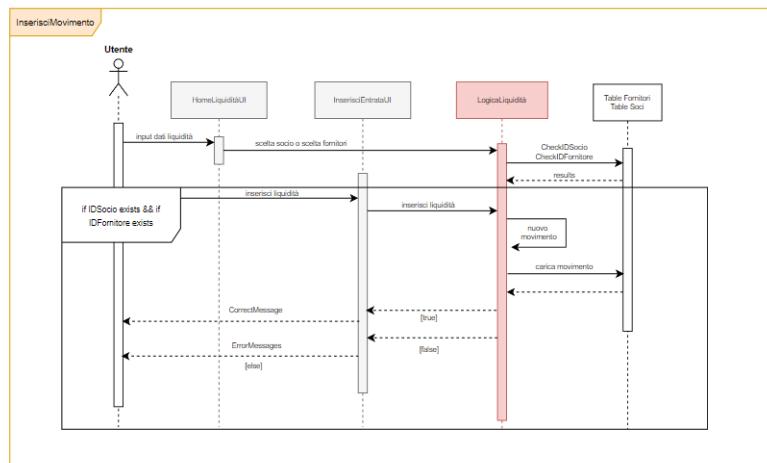


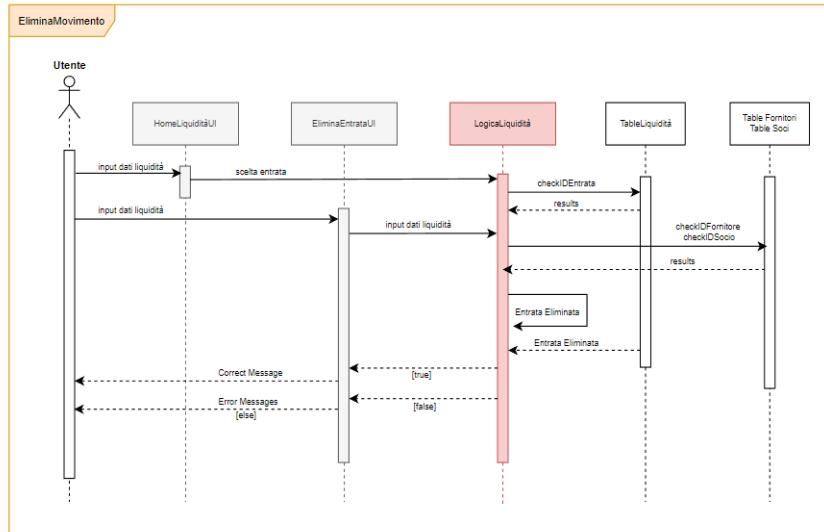
Diagrammi della sequenza relativa alla lista dei soci



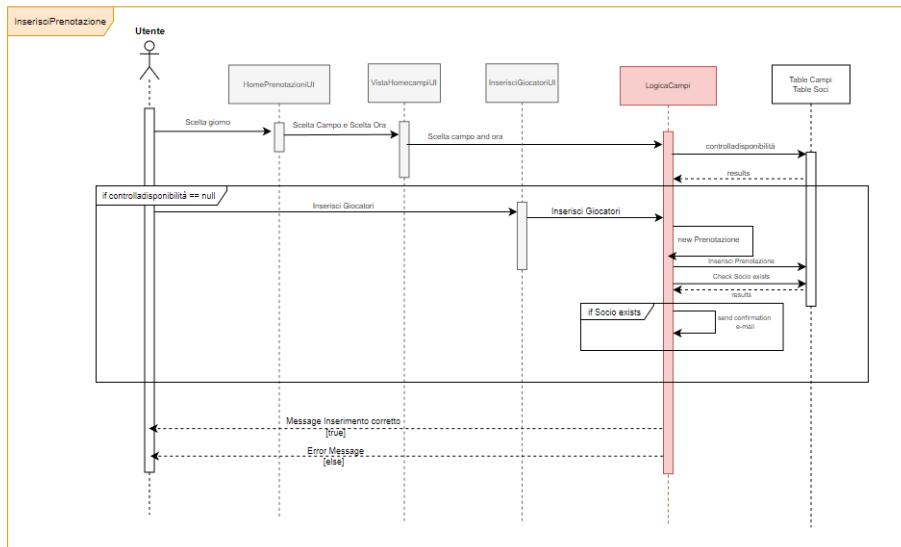


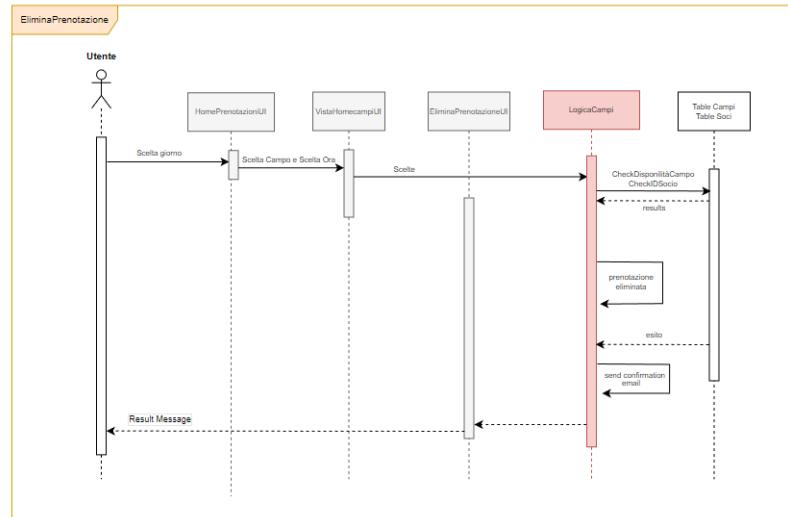
Diagrammi della sequenza relativi alla gestione della liquidità



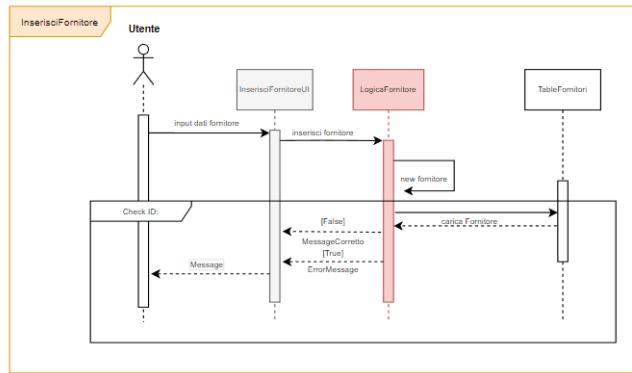


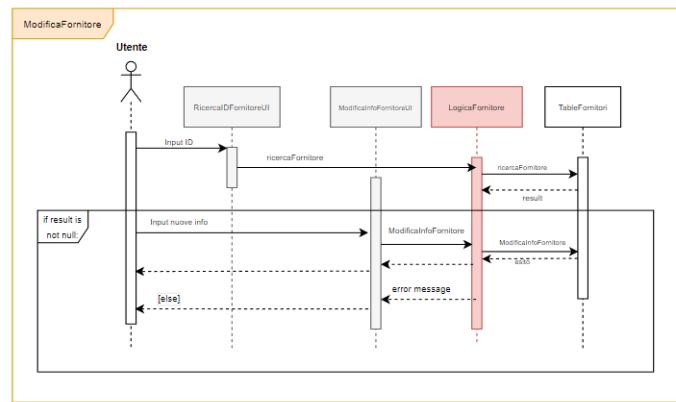
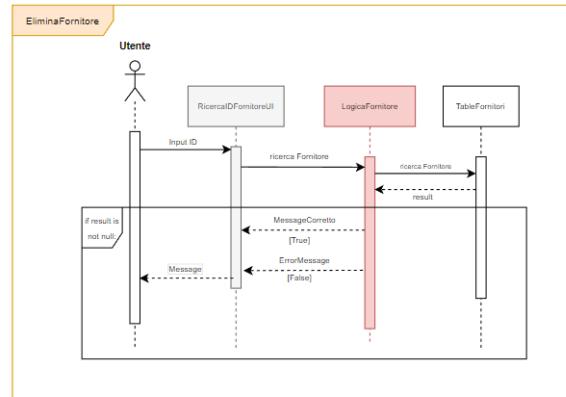
Diagrammi delle sequenze relativi alla gestione dei campi





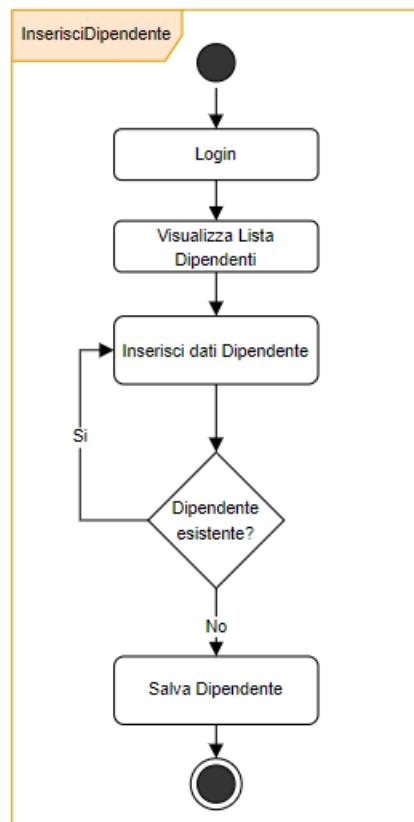
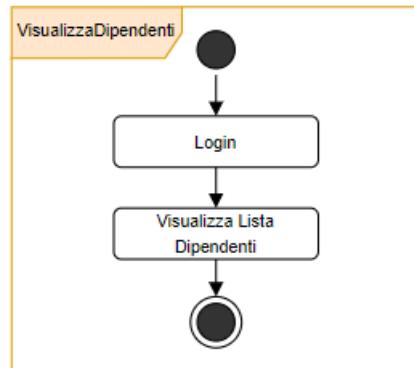
Diagrammi della sequanza relativi alla gestione dei fornitori

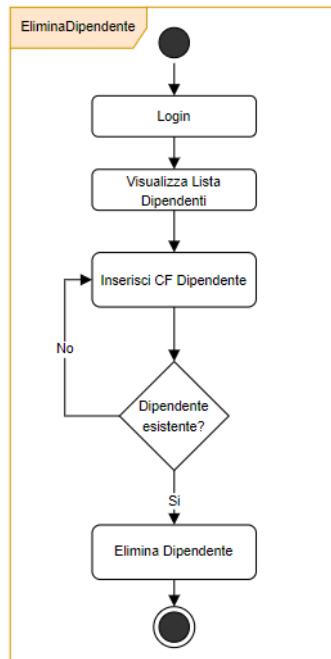


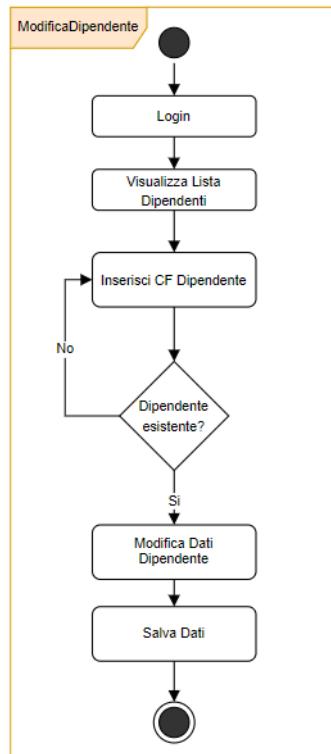


## 5.13 Diagrammi di attività

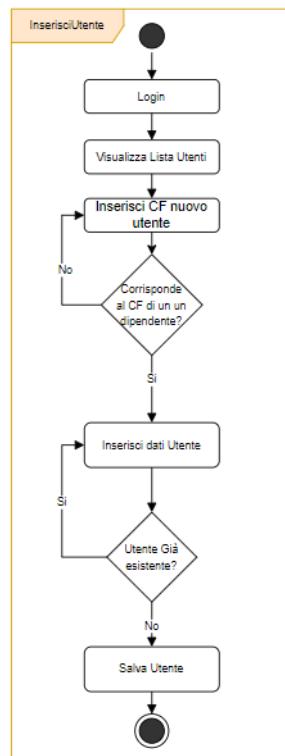
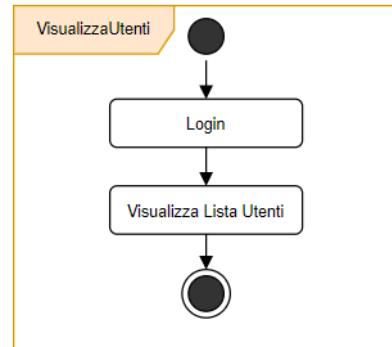
Attività: Gestione Dipendenti

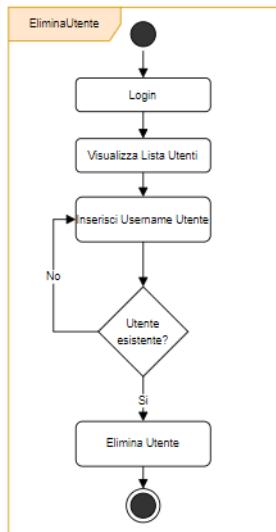




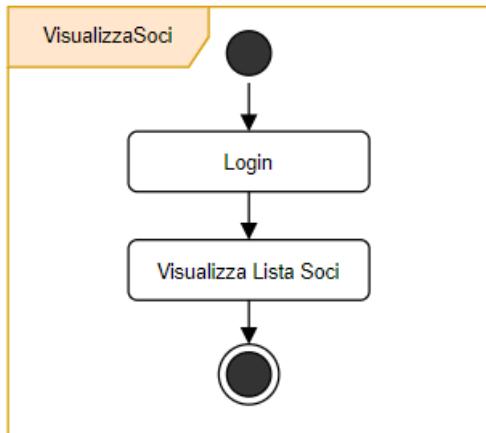


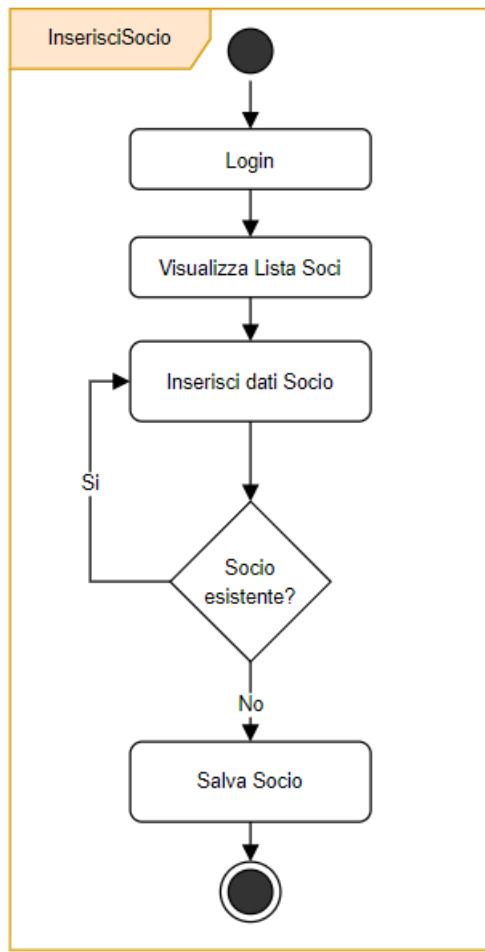
## Attività: Sicurezza

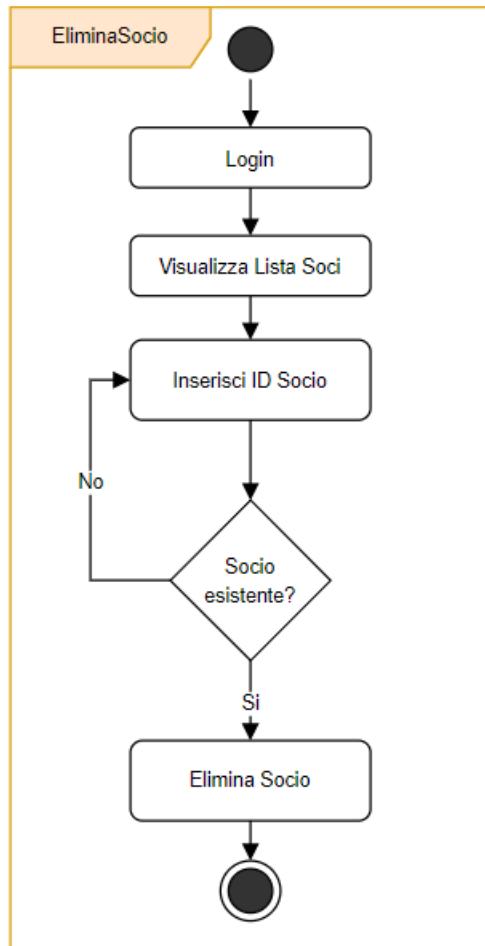


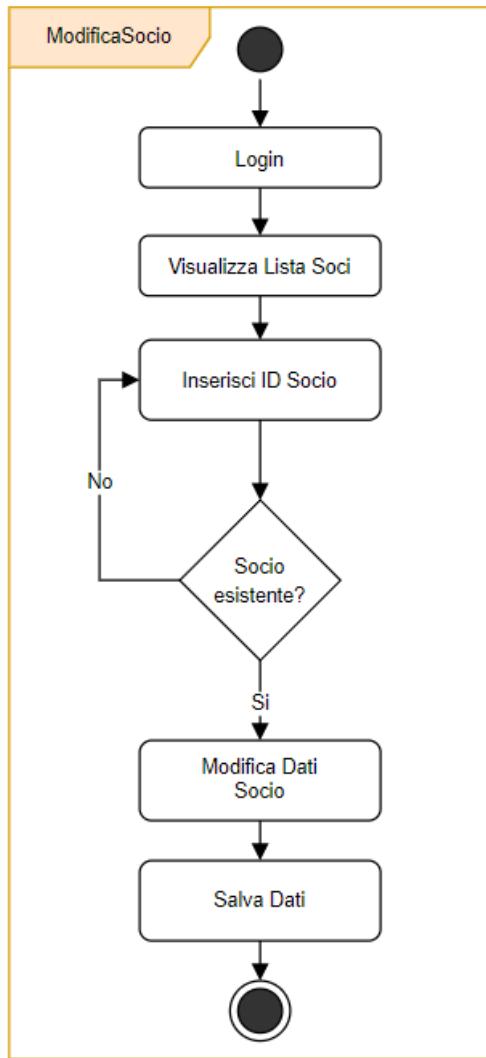


Attività: Lista Soci

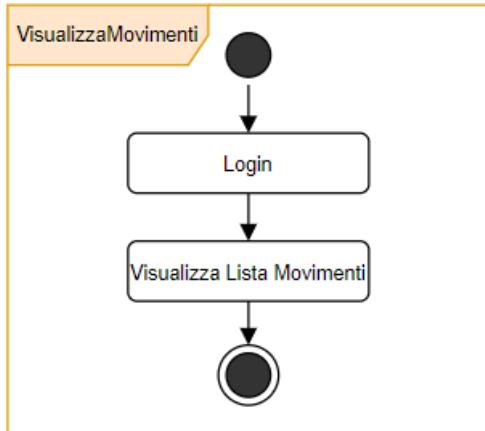


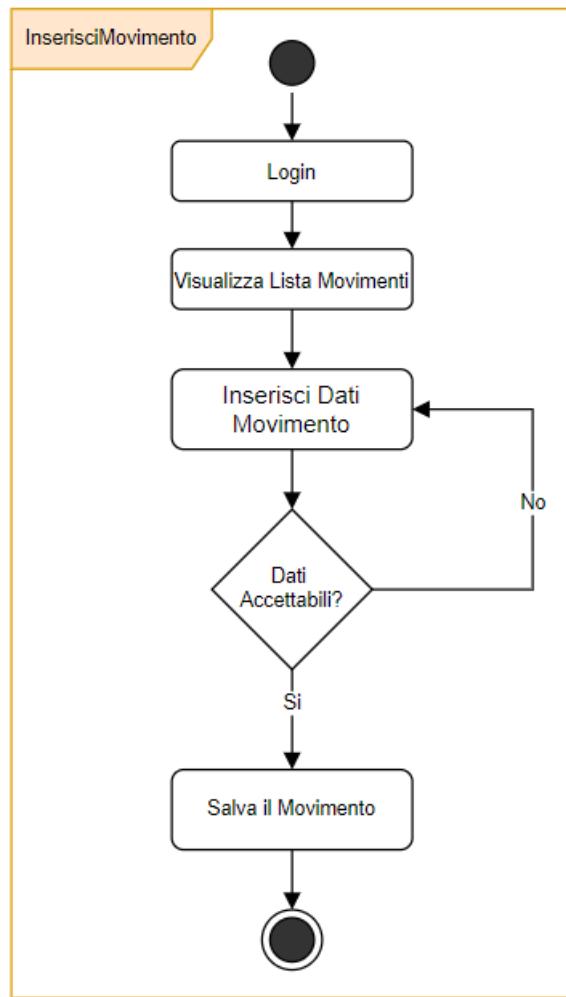


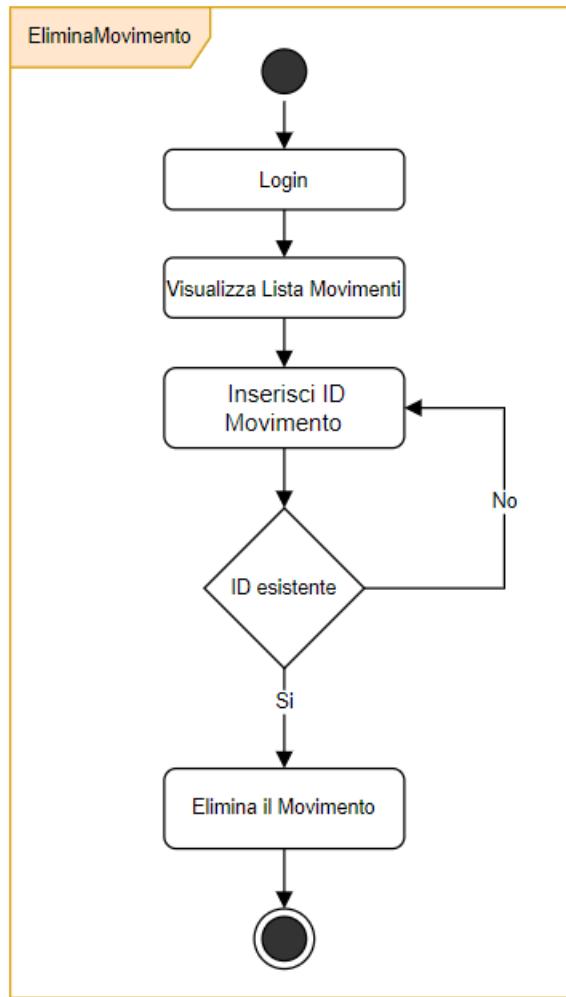




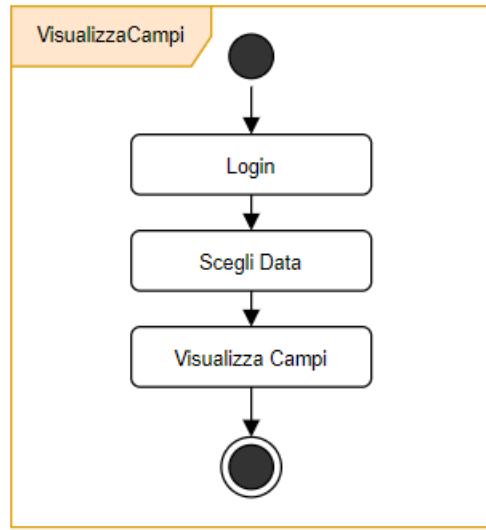
Attività: Gestione Liquidità

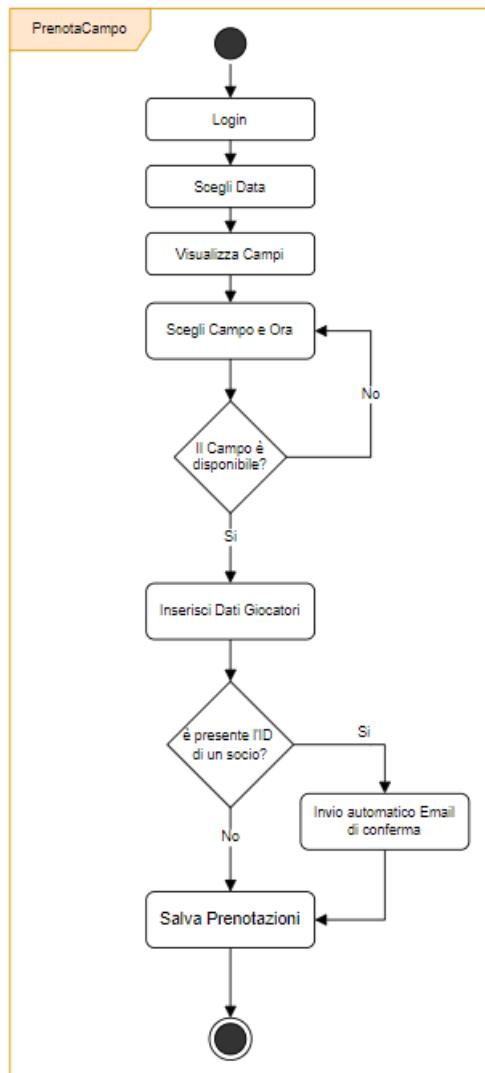


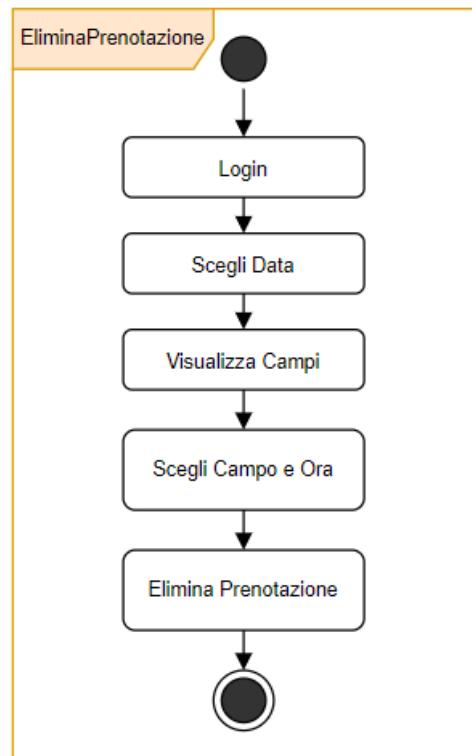




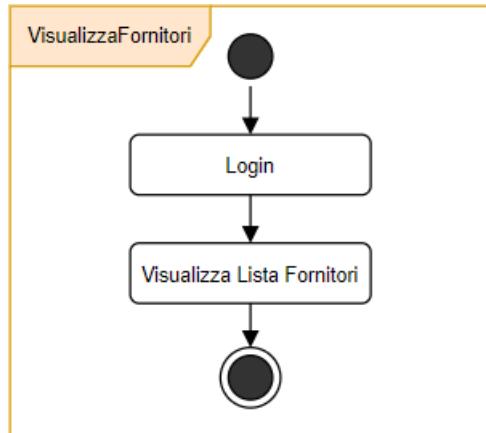
Attività: Prenotazione Campi

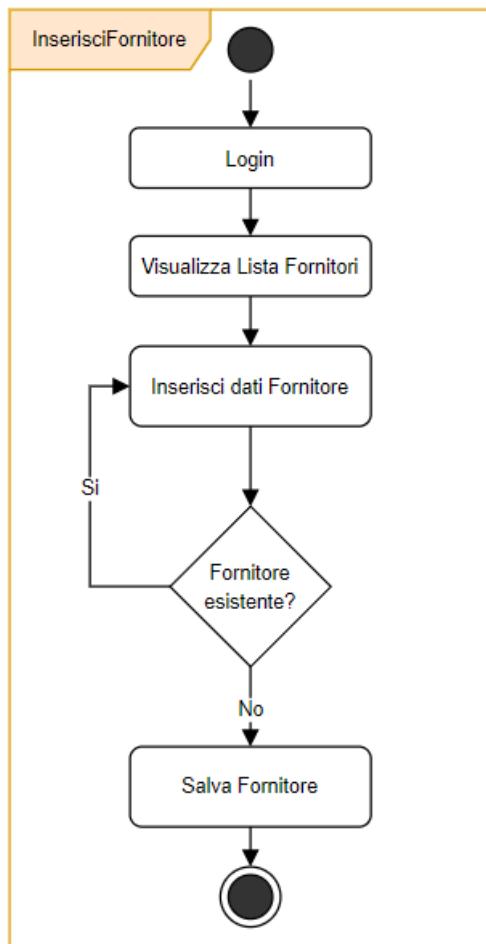


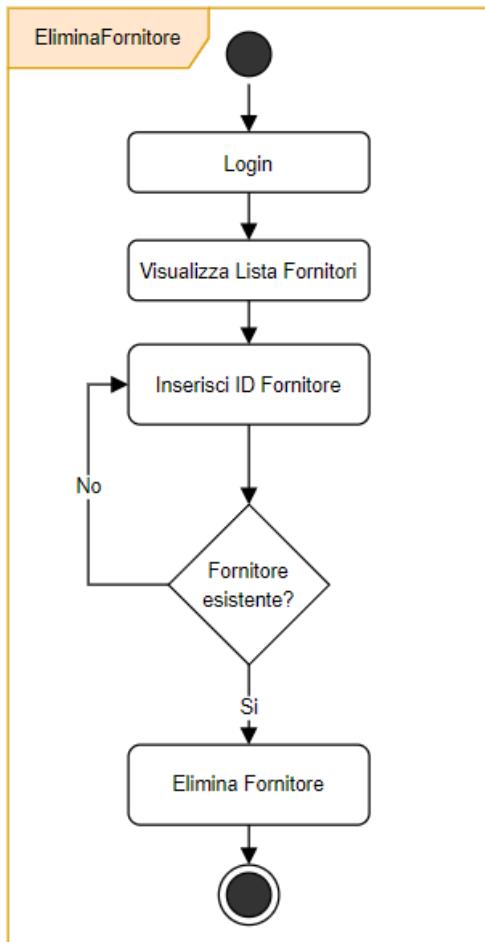


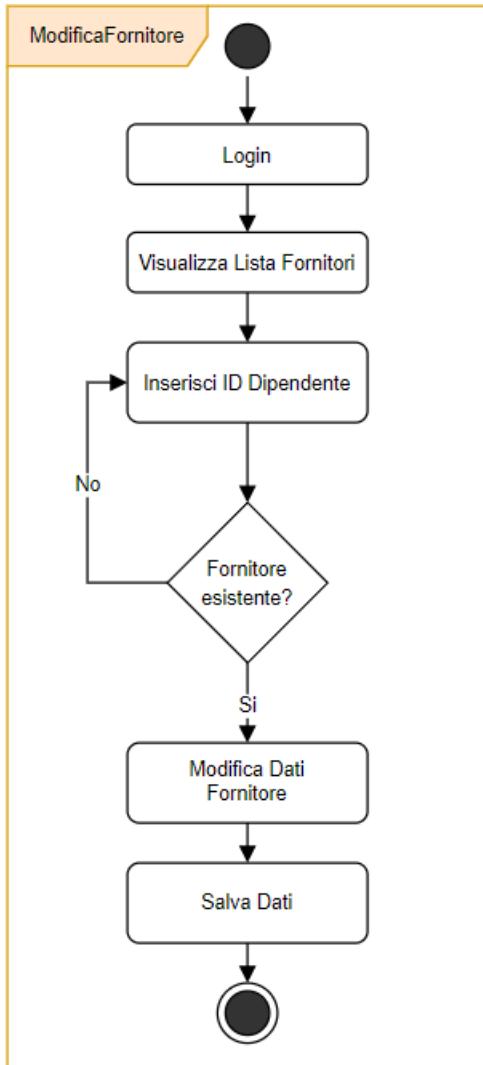


Attività: Gestione Fornitori









## 6 Dettagli Implementativi

Il linguaggio scelto per l'implementazione del software è Python 3, l'IDE di riferimento per lo sviluppo è PyCharm.

L'interfaccia grafica è sviluppata tramite PyQt, generata tramite il software QTDesigner.

Per quanto riguarda la gestione dei dati, si utilizza la libreria sqlite3 che implementa un DBMS incorporabile all'interno dell'applicazione.

Per la nostra applicazione abbiamo optato per un'architettura che si basa su 3 principali componenti:

- GestioneDatabase
- View
- Logica

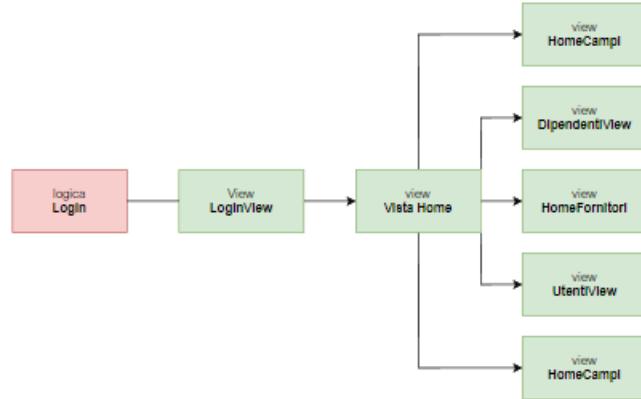
Nella directory "GestioneDatabase" è presente una classe astratta TableInterface che: crea le tabelle del DB, crea il cursore per effettuare le query ed implementa una serie di metodi astratti. Sono presenti le sottoDirectory e, all'interno di ogni sottoDirectory, c'è una classe "Table+*#nomesezione*", una per ogni tabella, che implementano i metodi astratti ed effettuano le query sul DB (quando chiamate dalla logica).

Nella directory "View" si trova la classe "#nomesezione+View" al cui interno sono importate tutte le classi relative alla vista, sono presenti le funzioni che, al trigger di eventi, stampano a video i messaggi (ad esempio di corretto inserimento) ed una serie di metodi getter per ottenere i valori delle variabili che poi verranno passate alle funzioni nella logica.

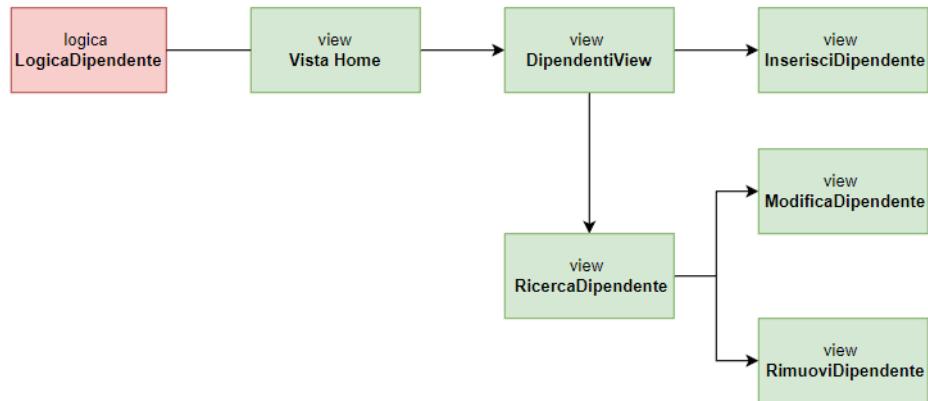
Nella directory "Logica" è presente la classe "Logica+*#nomesezione*" che gestisce i cambi di interfaccia per quella sezione, importando la relativa vista, e, al trigger di determinati eventi, scatena le relative funzioni da effettuare sul DB, importando la relativa tabella all'interno della directory GestioneDatabase.

## 6.1 Mappa dell'architettura

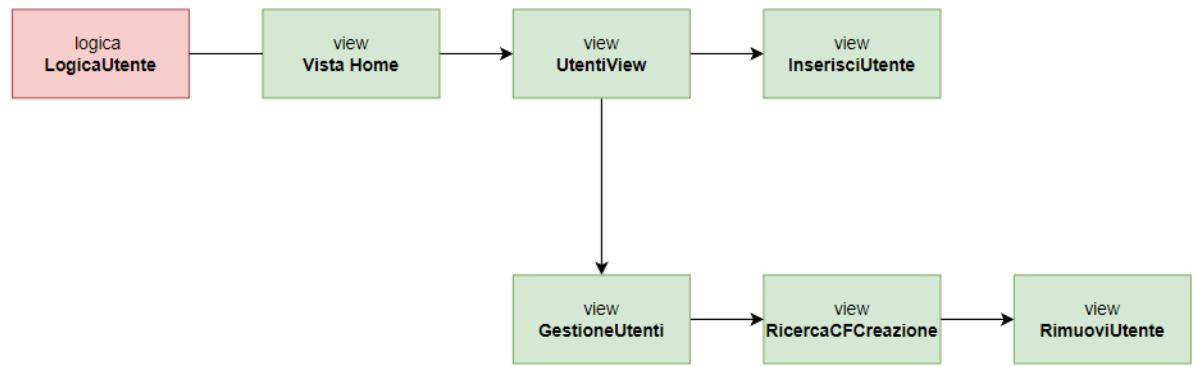
Di seguito riportiamo la mappa dell'architettura



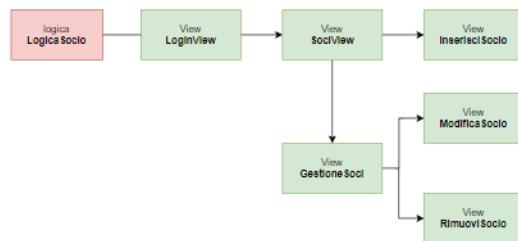
Autenticazione



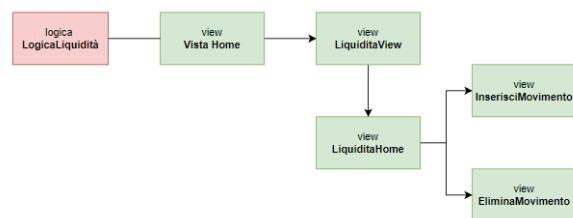
Dipendenti



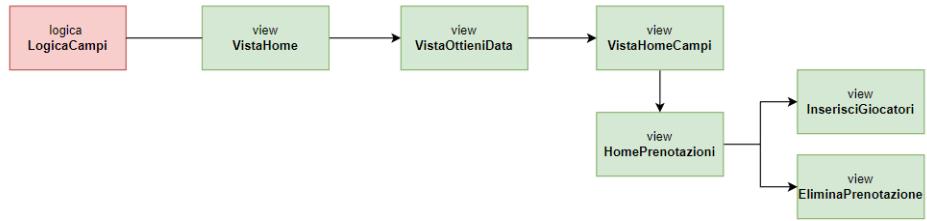
Sicurezza



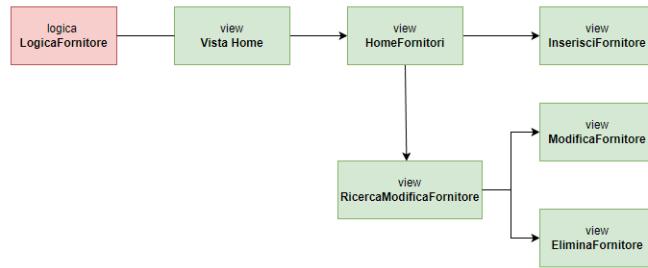
Soci



Liquidità



Campi

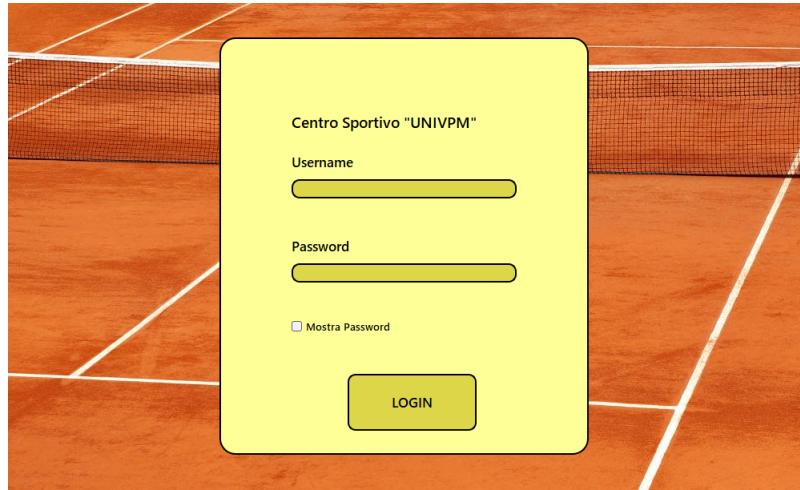


Fornitori

## 6.2 Graphic User Interface

### 6.2.1 Login

Ciascun utente è fornito di un username e di un password



### 6.2.2 Homepage

Dopo aver effettuato il login con successo, all'utente viene presentata l'homepage del sistema.



### 6.2.3 Gestione Dipendenti

La sezione sottostante mostra l'interfaccia relativa alla gestione dei dipendenti

Lista Dipendenti:								
	CF	Nome	Cognome	Città	Telefono	Mansione	Ore Sett.	€
1	PGI	Enrico	Piergallini	sbt	3339451947	Giardiniere	121	10000 €
2	CENSO1	Gianluca	Censo	Grottammare	66666666		100	1000 €
3	NICOLA	Nicola	Mochi	Loro Piceno	343434343	ciaoaoa	1000	1200 €
4								
5								
6								
7								
8								
9								
10								
11								
12								
*								

[Aggiungi Dipendente](#)      [Modifica Info Dipendente](#)  
[→ Torna alla Home](#)      [Elimina Dipendente](#)

CF	Nome
<input type="text"/>	<input type="text"/>
Cognome	Città di residenza
<input type="text"/>	<input type="text"/>
Telefono	Mansione
<input type="text"/>	<input type="text"/>
Ore settimanali	Stipendio
<input type="text"/>	<input type="text"/> €
Username	<input type="text"/>
<a href="#">→ Torna a Gestione dipendenti</a> <a href="#">Aggiungi Dipendente</a>	

CF	Nome
<input type="text"/>	<input type="text"/>
Cognome	Città di residenza
<input type="text"/>	<input type="text"/>
Telefono	Mansione
<input type="text"/>	<input type="text"/>
Ore settimanali	Stipendio
<input type="text"/>	<input type="text"/> €
Username	<input type="text"/>
<a href="#">→ Torna a Gestione dipendenti</a> <a href="#">Salva Modifiche</a>	

(a) Aggiungi Dipendente

(b) Modifica Dipendente

Inserisci il C.F. del dipendente da rimuovere:

[→ Torna a Gestione dipendenti](#)

[CONFERMA](#)

(c) Elimina Dipendente

#### 6.2.4 Sicurezza

La sezione sottostante mostra l'interfaccia relativa alla Sicurezza

Lista Utenti:	
CF Dipendente	Username
1 PGI	Enrico
2 NICOLA	Nicola
3	
4	
5	
6	
7	
8	
9	

[Aggiungi Utente](#)  
[Elimina Utente](#)

Inserisci un nuovo Utente:

Inserisci il CF del dipendente a cui vuoi associare un nuovo utente:

[Torna alla Home](#) [Conferma](#) [Torna alla Home](#) [Conferma e crea l'Utente](#)

(a) Aggiungi Utente

Inserisci l'username del dipendente da eliminare:

[Torna alla Home](#) [Conferma Eliminazione](#)

(b) Elimina Utente

### 6.2.5 Soci

ALBO SOCIALE						
ID Socio	e-mail	CF	Nome	Cognome	Telefono	Data Rinnovo
1	francescomaria.botti@gmail.com	FRCM85SC	Francesco Maria	Mosca	3467895412	2021-03-21
2	\$1094615@studenti.univpm.it	DDG	Davide	De Grazia	3456453217	2022-03-25
3	niccolol30@gmail.com	NCCBLN	Niccolò	Balconi	3339451947	2021-03-27
4						
5						
6						
7						
8						
9						
10						

[→ Torna alla Home](#) [+ Aggiungi Socio](#) [- Elimina Socio](#) [Modifica Info Socio](#)

ID Socio	e-mail
Codice Fiscale	Nome
Cognome	Telefono
Data rinnovo quota sociale (yyyy-mm-dd)	Inserisci l'ID del socio da eliminare:

[→ Torna a Gestione soci](#) [Aggiungi all'Albo Sociale](#) [→ Torna a Gestione Soci](#) [CONFERMA](#)

(a) Aggiungi Socio

(b) Elimina Socio

ID Socio	e-mail
2	\$1094615@studenti.univpm.it
Codice Fiscale	Nome
DDG	Davide
Cognome	Telefono
De Grazia	3456453217
Data rinnovo quota sociale	2022-03-25

[→ Torna a Gestione soci](#) [Salva Modifiche](#)

(c) Modifica Socio

## 6.2.6 Liquidità

Gestione Liquidità									
Categoria	€	Tipologia	ID Socio	ID Fornitore	ID	ID Socio	Nome	ID Fornitore	Nome F.
1			1	1	1	1	Chicco	1	MP Pesa
2			2	2	2	2	Davide	2	Dunlop
3			3	3	3	3	Nicolo	3	MAPEI
4			4	4	4	4	Enrico	4	
5			5	5	5	5	Matteo	5	
6			6	6	6	6	Mario	6	
7			7	8	7	8	Leonard	7	
8			8	9	8	9	Leonardo	8	
9			9	10	9	10	Fabio	9	
10			10	11	10	11	Nicola	10	
11			11	12	11	12	Lorenzo	11	
12									

Categoria  
Quota Sociale

ID Fornitore

Tipologia  
ENTRATA

ID Socio

Importo

ID Movimento:

Torna alla Home      Elimina Movimento      Inserisci nella Liquidità

### 6.2.7 Campi

Riguardo la prenotazione di un campo, una prima interfaccia(?) permette di scegliere la data tramite un piccolo calendario, mentre la seconda permette di effettuare la prenotazione o di poterla cancellare.

**Sezione Prenotazioni**

Selezione un giorno:  
20/07/2022

**Visualizza Campi**

[Torna alla Home](#)

(a) Scelta Giorno



(b) Calendario

2022-07-20

	Campo Centrale	Campo 2	Padel 1	Padel 2	Calcetto	ID Socio	Nome Cognome
9-10						1	Chicco Mosca
10-11						2	Davide De Grazia
11-12						3	Niccolò Balloni
12-13						4	Enrico Piergallini
13-14						5	Matteo Montesi
14-15						6	Mario da Montorio
15-16						7	Leonardo Mosca
16-17						8	Leonardo Lupidii
17-18						9	Fabio Baldonieri
18-19						10	Nicola Mochi
19-20						11	Lorenzo Medori
20-21						12	
21-22						13	

Torna alla Home    Scegli il campo:    Scegli la fascia oraria:    Selezione l'azione:

Campo CENTRALE (Terra Rossa) -> 9-10    Prenota    Elimina Prenotazione

(c) Home Prenota Campi

Inoltre, una volta effettuata la prenotazione, il software manda in automatico una mail se ad aver prenotato il campo è un socio

Prenotazione avvenuta con successo Posta in arrivo x

centrosportivounivpm@gmail.com  
a me <

Caro Niccolò Balloni,  
La sua prenotazione per il campo: Campo CENTRALE (Terra Rossa) in data: 2022-07-20 alle ore: 9-10 è avvenuta con successo!  
Buona giornata da tutto lo staff del Centro Sportivo!

### 6.2.8 Fornitori

Lista Fornitori					
ID Fornitore	Nome Azienda	e-mail	Telefono	Settore	
1	MP Pescia	mppesca@gmail.com	33394591947	Pesce	
2	Dunlop	dunlopita@gmail.com	3453434345	Attrezzatura Tennis	
3	MAPEI	mapeiinfo@gmail.com	3526377288	Superfici di Gioco	
4					
5					
6					
7					
8					

[→ Torna alla Home](#) [Modifica Info Fornitore](#) [Aggiungi Fornitore](#) [Elimina Fornitore](#)

ID Fornitore	Nome Azienda
e-mail	Telefono
Settore	Città
Via	
<a href="#">→ Torna a Fornitori</a> <a href="#">Inserisci Fornitore</a>	

(a) Aggiungi Fornitore

Inserisci l'ID del fornitore da eliminare

[→ Torna a Fornitori](#)

[Elimina](#)

(b) Elimina Fornitore

Inserisci l'ID del fornitore da modifica

[→ Torna a Fornitori](#)

[Conferma](#)

(c) Modifica Fornitore

### 6.3 Test del software

Il completamento del software è raggiunto dopo che esso è stato accuratamente testato. Per eseguire i test delle componenti del sistema è stato utilizzato il framework UnitTest, una libreria che permette di verificare se metodi e classi si comportino come previsto.

Le componenti testate riguardano...

I relativi test sono raccolti in un package a sé chiamato Tests. Vediamo di seguito un esempio di Unit Test riguardante

```
class TestSoftware(unittest.TestCase):

    def setUp(self):
        conn = sqlite3.connect('CentroSportivo.db')
        c = conn.cursor()
        c.execute("CREATE TABLE IF NOT EXISTS sicurezza(nome_utente text PRIMARY KEY, pass text NOT NULL, nome_dipendente text NOT NULL)")
        queryinseriscutente = "INSERT INTO sicurezza VALUES ('usertest','pass','enmetest')"
        c.execute(queryinseriscutente)
        conn.commit()

        c.execute("CREATE TABLE IF NOT EXISTS Soci(id_socio INTEGER PRIMARY KEY , e_mail text, CF text NOT NULL, nome_cliente text NOT NULL, cognome_cliente text NOT NULL, telefono text, Data_abbonamento text NOT NULL")
        queryinseriscisoci = "INSERT INTO Soci VALUES ('', 'test@socio.com','CFTEST', 'testname', 'testcognome', '5302728', '18-07-2022')"
        c.execute(queryinseriscisoci)
        conn.commit()
```

```
def tearDown(self):
    conn = sqlite3.connect('CentroSportivo.db')
    c = conn.cursor()
    queryeliminautente = "DELETE FROM sicurezza WHERE nome_utente = 'usertest'"
    queryelimasocio = "DELETE FROM Soci WHERE id_socio = '8'"
    c.execute(queryeliminautente)
    c.execute(queryelimasocio)
    conn.commit()
```

I metodi setUp() e tearDown() consentono di definire le istruzioni che verranno eseguite prima e dopo ogni metodo di test.

```
# testo che se sbaglio la password non effettuo il login
def test_logincorretto(self):
    class LoginViewMockUp:
        def getLoginlineEdit(self):
            return 'usertest', 'pass_sbagliata'
        def WarningMessage(self):
            pass

    class LoginLogicaMockUp(LoginLogica):
        def __init__(self):
            self.loginView = LoginViewMockUp()
            self.tableSicurezza = TableSicurezza()
    loginlogica = LoginLogicaMockUp()
    checkresult = loginlogica.CheckLogin()
    self.assertEqual(checkresult,0)
```

Test Login

```

#testo la modifica dei dati di un socio
def test_modificaCredenzialiSocio(self):

    class SocioViewMockUp:
        def __init__(self):
            class DataRinnovoMockUp:
                def __text__(self):
                    return '04-03-2020'
            class ModificaSocioMockUp:
                datarinnovo = DataRinnovoMockUp()
                self.modificaSocio = ModificaSocioMockUp()
        def getModificaLineEdit(self):
            return '8', 'newtest@socio.com', 'NEWCFTEST', 'newtestnome', 'newtestcognome', '86346445', '04-03-2013'
        def sociModificaCorretto(self):
            pass

    class LogicaSocioMockUp(LogicaSocio):
        def init_view(self):
            self.socioView = SocioViewMockUp()

    logicasocio = LogicaSocioMockUp(None)
    checkmodifica = logicasocio.modificaSocio()

    self.assertEqual(checkmodifica, 1)

```

### Test Modifica Socio

```

C:\Users\Utente\AppData\Local\Microsoft\WindowsApps\python3.9.exe "C:\Program Files\JetBrains\PyCharm 2021.2.3\plugins\python\helpers\pycharm\_jb_unittest_runner.py" --path C:/Users/Utente/Desktop/CentroSportivoUNIVPM\Tests
Testing started at 10:06 ...
Launching unitests with arguments python -m unittest C:/Users/Utente/Desktop/CentroSportivoUNIVPM/Tests/unittest.py in C:/Users/Utente/Desktop/CentroSportivoUNIVPM/Tests

Ran 2 tests in 0.068s
OK
Process finished with exit code 0

```

### Esito del test

Dopo aver accertato il funzionamento del software, esso può ritenersi concluso.

**Grazie per l'attenzione!**