

Progetto di Basi di Dati

- Piattaforma di scommesse calcistiche -

1. Raccolta delle specifiche della realtà d'interesse

Descrizione:

Si vuole progettare una base di dati per la gestione di un sito di scommesse calcistiche, con l'intento di offrire un'esperienza utente intuitiva e informativa. Al centro dell'idea c'è la volontà di soddisfare le esigenze degli appassionati, offrendo loro la possibilità di scommettere su una grande varietà di incontri calcistici in un ambiente affidabile e ricco di dati.

La piattaforma è principalmente destinata a una categoria di utenti: gli scommettitori. Gli scommettitori avranno accesso ad un sistema completo per monitorare le loro attività di scommessa, includendo la registrazione di importi scommessi, le quote e un bilancio dettagliato delle loro transazioni finanziarie. Il sistema permetterà inoltre agli utenti di gestire i loro depositi e prelievi in modo efficiente.

La struttura del database, necessario per realizzare questa visione, è progettata per supportare funzionalità fondamentali quali la registrazione e gestione degli account utente, deposito e prelievo di fondi, piazzamento di scommesse, e la visualizzazione di cronologie dettagliate sulle previsioni passate. Ogni account, includerà dati anagrafici essenziali e informazioni per l'accesso, assicurando così efficienza e sicurezza nel trattamento dei dati personali.

Il cuore del database sono i dati sulle partite di calcio, che includono informazioni aggiornate in tempo reale su squadre, giocatori e classifiche. Le quote di scommessa ed i risultati degli incontri, arricchiti da statistiche dettagliate, sono anch'essi tenuti in considerazione, per permettere agli utenti di avere sempre a disposizione le informazioni più aggiornate e pertinenti.

Il successo della piattaforma dipenderà dalla sua capacità di offrire dati precisi e aggiornati, un'interfaccia utente facile da usare e un'esperienza di scommessa personalizzabile. Per questo motivo è necessaria una base di dati che permetta di effettuare sui dati tutte le operazioni necessarie.

Specifiche della realtà d'interesse:

La piattaforma online di scommesse calcistiche è progettata per offrire un'esperienza completa agli utenti, sottolineando funzionalità chiave che richiedono un solido supporto da parte del nostro database.

Gli utenti, al momento della registrazione e della creazione del proprio account, ottengono automaticamente accesso alle funzionalità cruciali della piattaforma e vengono dotati di un conto personale, gestibile attraverso transazioni finanziarie da conti di credito esterni. Ad ogni utente viene

permesso di navigare attraverso una vasta gamma di partite e di piazzare scommesse con facilità. Il database svolge un ruolo essenziale nella registrazione di tutte le operazioni, garantendo precisione e tempestività sia nell'inserimento delle scommesse che nelle transazioni finanziarie.

Le scommesse sono gestite attraverso la classica formula delle schedine. Gli utenti possono scegliere tra varie partite di calcio, anche da differenti campionati, selezionando, come opzione di scommessa offerta dalla piattaforma, l'esito degli incontri: Vittoria squadra di casa/Pareggio/Vittoria squadra fuori casa, indicati come 1/X/2.

Dopo aver selezionato le opzioni, l'utente crea la schedina. Le quote degli incontri scommessi vengono moltiplicate per determinare la quota combinata della schedina, fornendo un'indicazione della potenziale vincita. Dopo la conferma, la schedina viene piazzata e diventa irremovibile.

Gli scommettitori seguono gli eventi inclusi nella schedina e, in caso di corrette previsioni su tutti gli esiti, ricevono un pagamento in base alle regole della piattaforma. Le scommesse in schedine offrono la possibilità di vincite considerevoli, ma richiedono una corretta previsione di più risultati, introducendo un elemento di sfida nel processo di scommessa. Nel caso si desideri, è anche possibile creare una schedina focalizzata su una singola partita, la vincita della quale sarà determinata dalla singola quota dell'incontro. Il database cataloga le partite in due categorie: quelle concluse e quelle future. Viene offerta la possibilità di visualizzare i risultati delle partite passate ma, ovviamente, le scommesse saranno consentite esclusivamente per gli incontri in programma.

Per ottimizzare l'esperienza utente, le statistiche dettagliate su squadre, giocatori e performance storiche sono facilmente accessibili. La visualizzazione del numero di vittorie di una squadra, la lista completa delle partite che ha disputato e l'aggiornamento in tempo reale della posizione nel campionato richiedono un costante aggiornamento del database in relazione a risultati sportivi e classifiche.

Le operazioni di visualizzazione dello storico schedine di un utente, delle sue transazioni e del saldo costituiscono un pilastro essenziale della piattaforma. Il database deve essere in grado di gestire efficacemente queste richieste, fornendo un quadro completo e dettagliato delle attività passate di ciascun utente, inclusi importi scommessi, quote, e dettagli finanziari.

L'inserimento di nuovi utenti e la gestione delle loro credenziali sono operazioni fondamentali per la crescita della piattaforma, analogamente, l'inserimento di transazioni e schedine relative agli utenti richiede una gestione accurata del flusso finanziario e delle scommesse, con il database che deve tenere traccia di ogni dettaglio per garantire la coerenza delle informazioni.

In sintesi, il nostro database è progettato per supportare una serie di operazioni cruciali che vanno dalla gestione degli utenti e delle loro attività di scommessa, al fornire informazioni accurate sulle performance delle squadre sulle quali gli utenti possono scommettere. La sua robustezza e efficienza sono fondamentali per realizzare la visione di piattaforma di scommesse online sicura, intuitiva e informativa.

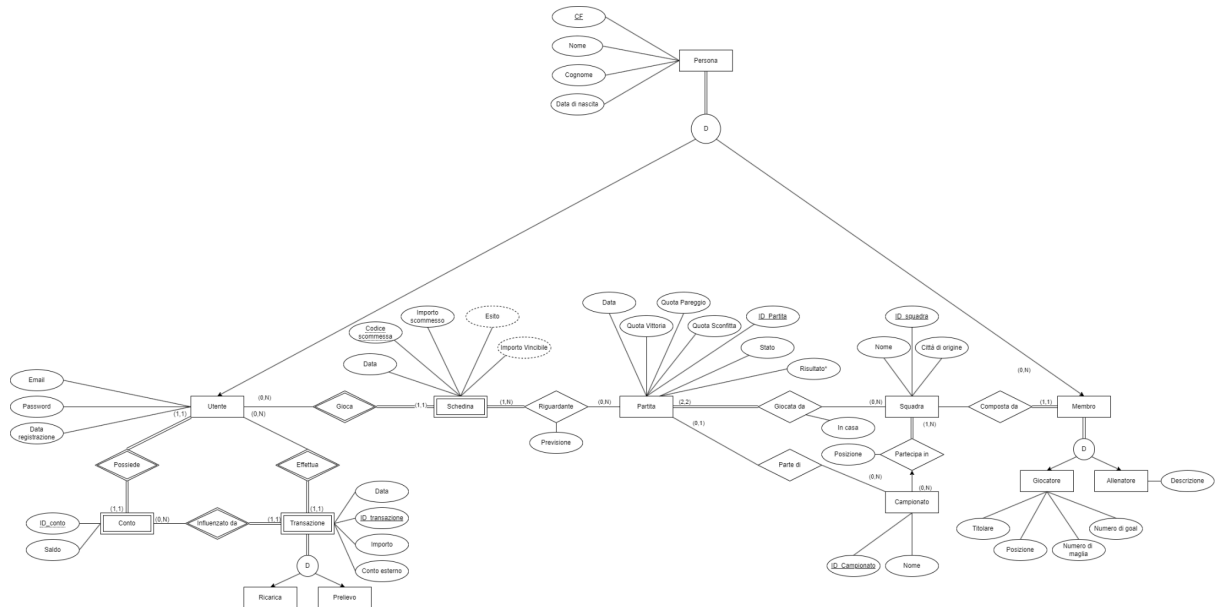
Glossario dei termini

Termine	Definizione
Squadra	Gruppo di giocatori che partecipa a competizioni sportive
Campionato	Torneo in cui le squadre di una lega calcistica si affrontano in partite organizzate competendo per il titolo di campione
Esito	Risultato di una partita di calcio, che può essere la vittoria di una squadra, un pareggio o una sconfitta
Scommessa	Azione attraverso la quale gli utenti della piattaforma selezionano e puntano su possibili risultati di eventi sportivi, in questo caso le partite di calcio. Le scommesse possono variare in termini di importo e tipo di previsione.
Scommettitore	Individuo registrato sulla piattaforma che partecipa attivamente piazzando scommesse
Quota	Valore numerico associato a ciascun possibile esito di un incontro, indicante la moltiplicazione del denaro scommesso per calcolare la vincita potenziale
Schedina	Lista di scommesse selezionate dagli scommettitori, con le quote moltiplicate tra loro per determinare la quota combinata

2. Progettazione concettuale della base di dati

Schema EER

Procedendo con la progettazione concettuale della base di dati, si ottiene il seguente schema EER:



Dizionario delle entità

Legenda: sotto-entità, attributo multivalore, attributo ridondante, entità debole, chiave candidata

Entità	Descrizione	Attributi	Identificatore
Persona	Generica persona da salvare nel database	C.F. Nome Cognome Data di Nascita	C.F
Utente	Utilizzatore registrato del sito	E-Mail Password Data di Registrazione	
Membro	Non utilizzatore del sito legato ad una squadra		
Giocatore	Giocatore di una squadra	Titolare Posizione Numero di maglia Numero di goal	
Allenatore	Allenatore di una squadra	Descrizione	
Conto	Deposito di fondi finanziari univocamente associato ad un Utente	ID_Conto Saldo	ID_Conto
Transazione	Operazione finanziaria effettuata su un conto	ID_transazione Importo Conto esterno Data	ID_Transazione
Ricarica	Operazione finanziaria consistente nello spostamento di soldi verso un conto utente		
Prelievo	Operazione finanziaria consistente nello spostamento di soldi da un conto		

Entità	Descrizione	Attributi	Identificatore
Persona	Generica persona da salvare nel database	C.F. Nome Cognome Data di Nascita	C.F
Utente	Utilizzatore registrato del sito	E-Mail Password Data di Registrazione	
Membro	Non utilizzatore del sito legato ad una squadra		
Giocatore	Giocatore di una squadra	Titolare Posizione Numero di maglia Numero di goal	
Allenatore	Allenatore di una squadra	Descrizione	
	utente		
Schedina	Lista di scommesse scelte da un Utente	Codice scommessa Importo scommesso Esito Data Importo Vincibile	Codice scommessa
Partita	Incontro calcistico dall'esito incerto al quale partecipano due squadre	Data Quota Vittoria Quota Pareggio Quota Sconfitta ID_Partita Stato Risultato	ID_Partita
Squadra	Contendente che partecipa ad una o più partite	Nome ID_Squadra Città di origine	ID_Squadra
Campionato	Torneo composto da più partite al quale prendono parte numerose squadre	ID_Campionato Nome	ID_Campionato

Dizionario delle relazioni

Relazione	Descrizione	Entità Coinvolte	Attributi
Possiede	Ogni utente possiede un conto	Utente(1,1) Conto(1,1)	
Effettua	Un utente può effettuare una transazione	Utente(0,N) Transazione(1,1)	
Gioca	Un utente può giocare una schedina	Utente(0,N) Schedina(1,1)	
Riguardante	Una schedina riguarda una o più partite	Schedina(1,N) Partita(0,N)	Previsione
Giocata da	Una partita è giocata da due squadre	Partita(2,2) Squadra(0,N)	In casa
Parte di	Una partita è parte di un campionato	Partita(0,1) Campionato(0,N)	
Partecipa in	Una squadra partecipa in un campionato	Squadra(1,N) Campionato(0,N)	Posizione
Composta da	Una squadra è composta da membri	Membro(1,1) Squadra(12,19)	

Vincoli non esprimibili nello schema

Oltre ciò che è deducibile dallo schema EER, si tenga conto dei seguenti vincoli:

- Ogni volta che in un attributo bisogna salvare una data essa viene scritta in formato Anno/Mese/Giorno.
- L'attributo "Previsione" della relazione "Riguardante" può assumere uno tra tre valori distinti: 1 rappresenta la vittoria della squadra di casa, X il pareggio e 2 la vittoria della squadra ospite. Lo stesso si applica all'attributo "Risultato" dell'entità "Partita"

- L'attributo "In casa" della relazione "Giocata da" può assumere uno tra due valori distinti: 0 se la squadra coinvolta nella relazione è la squadra ospite, 1 altrimenti

3. Definizione delle procedure per la gestione della base di dati

Tavola dei volumi

Definiamo di seguito la tavola dei volumi della base di dati.

Concetto	Tipo	Carico Applicativo
Utente	E	100
Conto	E	100
Transazioni	E	500
Schedina	E	200
Partite	E	100
Squadre	E	20
Campionati	E	3
Membri	E	380
Giocatore	E	360
Allenatore	E	20
Persona	E	460
Possiede	R	100
Influenzato da	R	500
Gioca	R	200
Riguardante	R	800
Giocata da	R	200
Partecipa in	R	40
Parte di	R	80
Composta da	R	380
Effettua	R	500

Tavola delle operazioni

Definiamo di seguito la tavola delle operazioni per la gestione dei dati memorizzati nella base di dati.

	Operazione	Tipo	Frequenza
1	Inserimento di un nuovo utente	I	20 volte al giorno
2	Visualizzare lo storico schedine di un'utente	B	40 volte al giorno
3	Visualizzare le transazioni di un'utente	B	40 volte al giorno
4	Visualizzare le credenziali di un'utente	B	30 volte al giorno
5	Visualizzare il saldo di un'utente	I	100 volte al giorno
6	Inserimento di una transazione relativa ad un'utente	I	70 volte al giorno
7	Inserimento di una schedina relativa ad un'utente	I	40 volte al giorno
8	Visualizzazione di una schedina	I	70 volte al giorno
9	Inserimento di una partita	I	3 volte al giorno
10	Inserimento di una partita riguardante un campionato.	I	8 volte al giorno
11	Conclusione di una partita e inserimento del risultato	I	33 volte al giorno
12	Inserimento di un campionato	I	0,5 volte al giorno
13	Visualizzare le squadre e le partite relative ad un campionato	I	10 volte al giorno
14	Registrazione di una squadra e inserimento nel relativo campionato	I	5 volte al giorno
15	Visualizzare tutte le partite di una squadra	I	10 volte al giorno

	Operazione	Tipo	Frequenza
16	Aggiornamento della posizione di una squadra in un campionato	I	33 volte al giorno
17	Visualizzare una partita e le squadre che la giocano	I	60 volte al giorno
18	Inserimento di un giocatore in una squadra	I	20 volte al giorno
19	inserimento di un'allenatore in una squadra	i	2 volte al giorno
20	Visualizzazione dei membri di una squadra	I	10 volte al giorno

4. Progettazione logica

Analisi delle ridondanze

I dati ridondanti sono gli attributi “Esito” e “Importo vincibile” della Schedina, infatti sarebbe possibile calcolarli partendo dalle informazioni delle partite che la riguardano. Supponendo che esito abbia un peso di 1 byte e che importo vincibile abbia un peso di 3 byte, se si decidesse di tenerli entrambi, avendo 200 come volume di Schedina, il dato andrebbe ad occupare uno spazio di circa 800 byte. Per decidere se mantenere o meno il dato ridondante è necessario calcolare, per le operazioni che lo coinvolgono, la differenza nel numero di accessi con e senza di esso.

Tavola degli accessi

Operazione 2:

Visualizzare lo storico schedine di un'utente.(40 volte al giorno)

Calcolo con ridondanza di entrambe				Calcolo senza ridondanza di entrambe			
	Tipo	Accessi	Tipo Accessi		Tipo	Accessi	Tipo Accessi
Utente	E	1	L	Utente	E	1	L
Gioca	R	2	L	Gioca	R	2	L
Schedina	E	2	L	Schedina	E	2	L
				Riguardante	R	8	L
				Partita	E	8	L
TOTALE	$40 \cdot (1+2+2)=200$			TOTALE	$40 \cdot (1+2+2+8+8)=840$		

Calcolo con ridondanza di esito				Calcolo con ridondanza di importo vincibile			
	Tipo	Accessi	Tipo Accessi		Tipo	Accessi	Tipo Accessi
Utente	E	1	L	Utente	E	1	L
Gioca	R	2	L	Gioca	R	2	L
Schedina	E	2	L	Schedina	E	2	L
Riguardante	R	8	L	Riguardante	R	8	L
Partita	E	8	L	Partita	E	8	L
TOTALE	$40 \cdot (1+2+2+8+8)=840$			TOTALE	$40 \cdot (1+2+2+8+8)=840$		

Operazione 7:

-Inserimento di una schedina relativa ad un'utente.(40 volte al giorno).

Calcolo con ridondanza di entrambe				Calcolo senza ridondanza di entrambe			
	Tipo	Accessi	Tipo Accessi		Tipo	Accessi	Tipo Accessi
Schedina	E	1	S	Schedina	E	1	S
Gioca	R	1	S	Gioca	R	1	S
Riguardante	R	4	S	Riguardante	R	4	S
Partita	E	4	L				
TOTALE	$40 \cdot (4 + (1 + 1 + 4) \cdot 2) = 640$			TOTALE	$40 \cdot ((1 + 1 + 4) \cdot 2) = 480$		

Calcolo con ridondanza di esito				Calcolo con ridondanza di importo vincibile			
	Tipo	Accessi	Tipo Accessi		Tipo	Accessi	Tipo Accessi
Schedina	E	1	S	Schedina	E	1	S
Gioca	R	1	S	Gioca	R	1	S
Riguardante	R	4	S	Riguardante	R	4	S
Partita	E	4	L	Partita	E	4	L
TOTALE	$40 \cdot (4 + (1 + 1 + 4) \cdot 2) = 640$			TOTALE	$40 \cdot (4 + (1 + 1 + 4) \cdot 2) = 640$		

Operazione 8:

-Visualizzazione di una schedina.(70 volte al giorno)

Calcolo con ridondanza di entrambe				Calcolo senza ridondanza di entrambe			
	Tipo	Accessi	Tipo Accessi		Tipo	Accessi	Tipo Accessi
Schedina	E	1	L	Schedina	E	1	L
				Riguardante	R	4	L
				Partita	E	4	L
TOTALE	70*1=70			TOTALE	70*9=630		

Calcolo con ridondanza di esito				Calcolo con ridondanza di importo vincibile			
	Tipo	Accessi	Tipo Accessi		Tipo	Accessi	Tipo Accessi
Schedina	E	1	L	Schedina	E	1	L
Riguardante	R	4	L	Riguardante	R	4	L
Partita	E	4	L	Partita	E	4	L
TOTALE	70*9=630			TOTALE	70*9=630		

Operazione 11:

-Modifica stato di una partita.(33 volte al giorno)[impatto solo su esito].

Calcolo con ridondanza di esito				Calcolo senza ridondanza di esito			
	Tipo	Accessi	Tipo Accessi		Tipo	Accessi	Tipo Accessi
Partita	E	1	L	Partita	E	1	L
Partita	E	1	S	Partita	E	1	S
Riguardante	R	8	L				
Schedina	E	8	L				
Riguardante	R	32	L				
Partita	E	32	L				
Schedina	E	8	S				
TOTALE	$33 \cdot (1 + 8 + 8 + 32 + 32 + 2 \cdot (1 + 8)) = 3267$			TOTALE	$33 \cdot (1 + 1 \cdot 2) = 99$		

Con importo e esito ridondante
200
640
70
3267
TOTALE: 4177

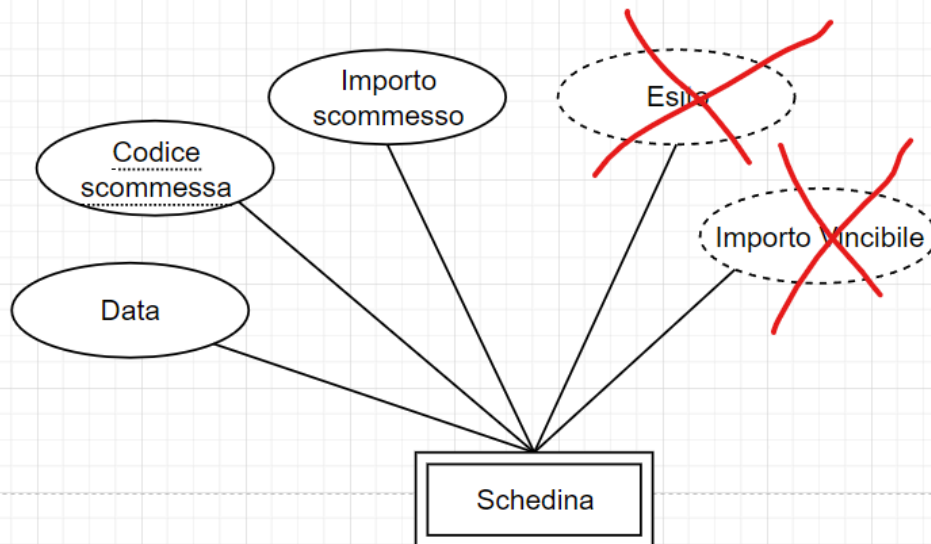
Con esito ridondante e senza importo ridondante
840
640
630
3267
TOTALE: 5377

Con importo ridondante e senza esito ridondante
840
640
630
99
TOTALE: 2209

Senza importo ridondante e senza esito ridondante
840
480
630
99

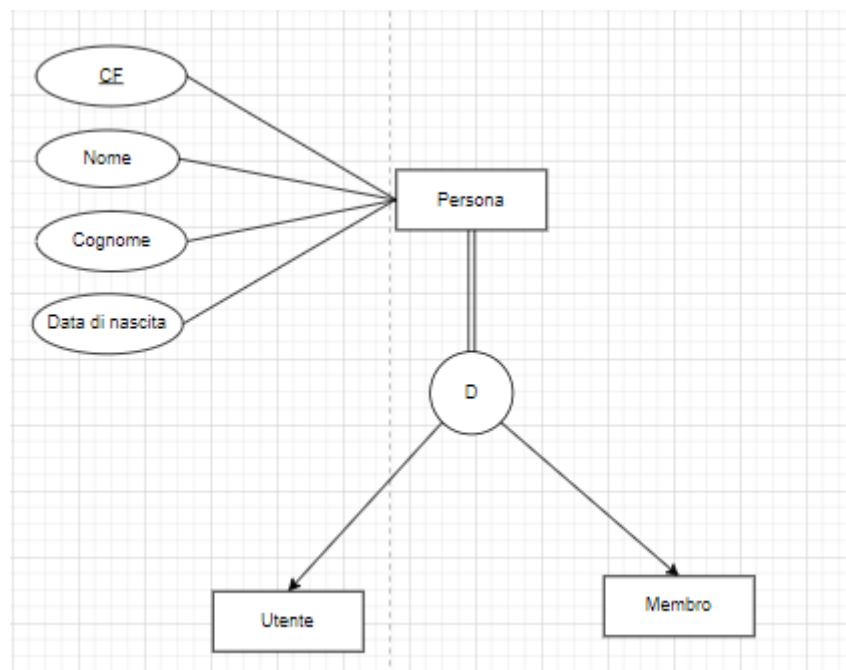
TOTALE: 2049

Dato il minor numero di accessi e lo spreco di spazio pari a 40 byte, è più efficiente scegliere di **non mantenere** nessuno dei due dati ridondanti.



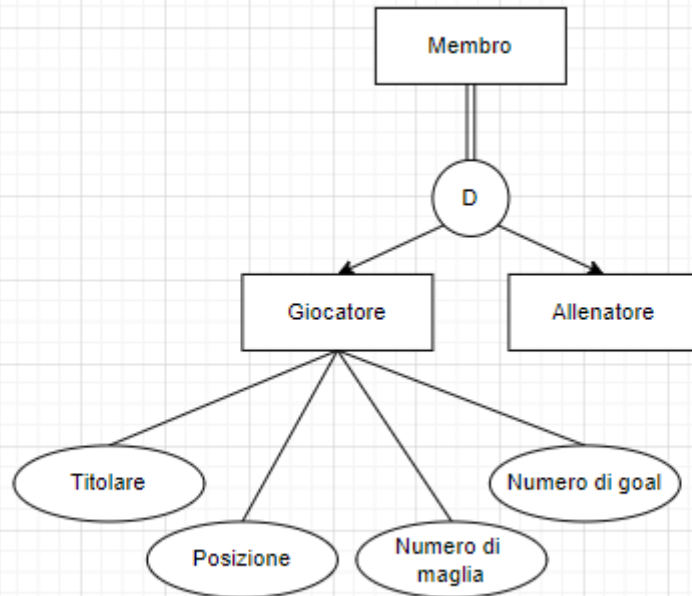
Eliminazione delle gerarchie

Nello schema inizialmente elaborato, è presente la seguente specializzazione dell'entità "Persona":



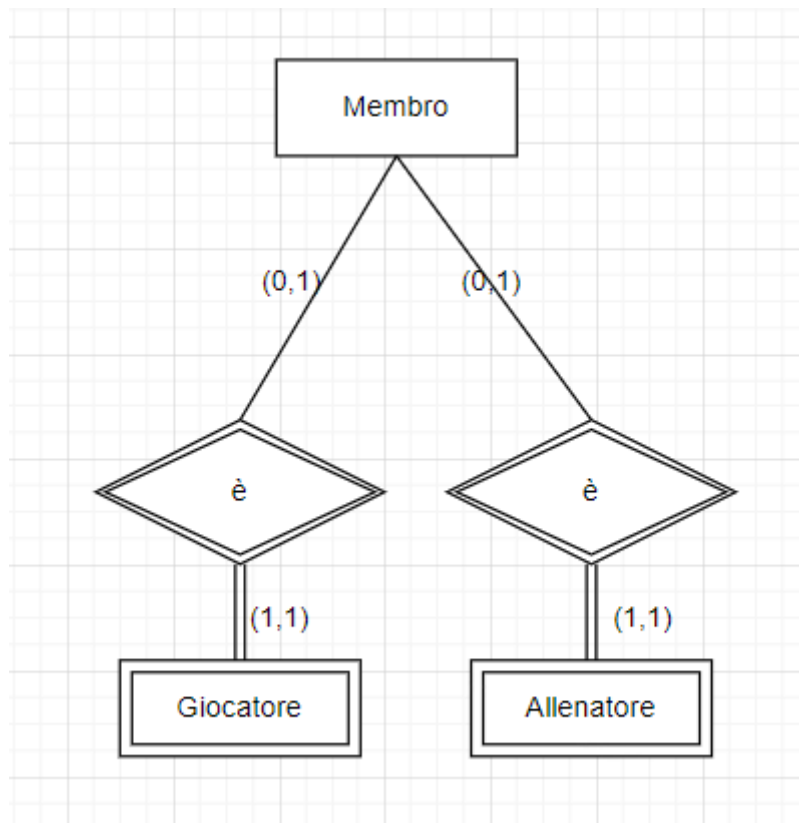
Poichè l'entità persona non partecipa direttamente a nessuna relazione scegliamo di eliminarla e di far ereditare i suoi attributi ad Utente e Membro.

Nella schema iniziale è inoltre presente un'altra specializzazione ovvero quella di "Membro":

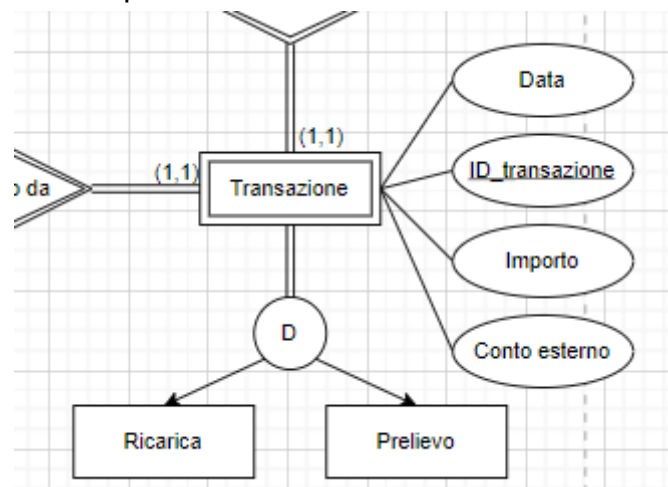


in questo caso la scelta effettuata è la creazione di due nuove relazioni, che rendono le entità figlie due nuove entità deboli. Ciò è necessario affinché si preservi la separazione dei due concetti e si evitino attributi NULL.

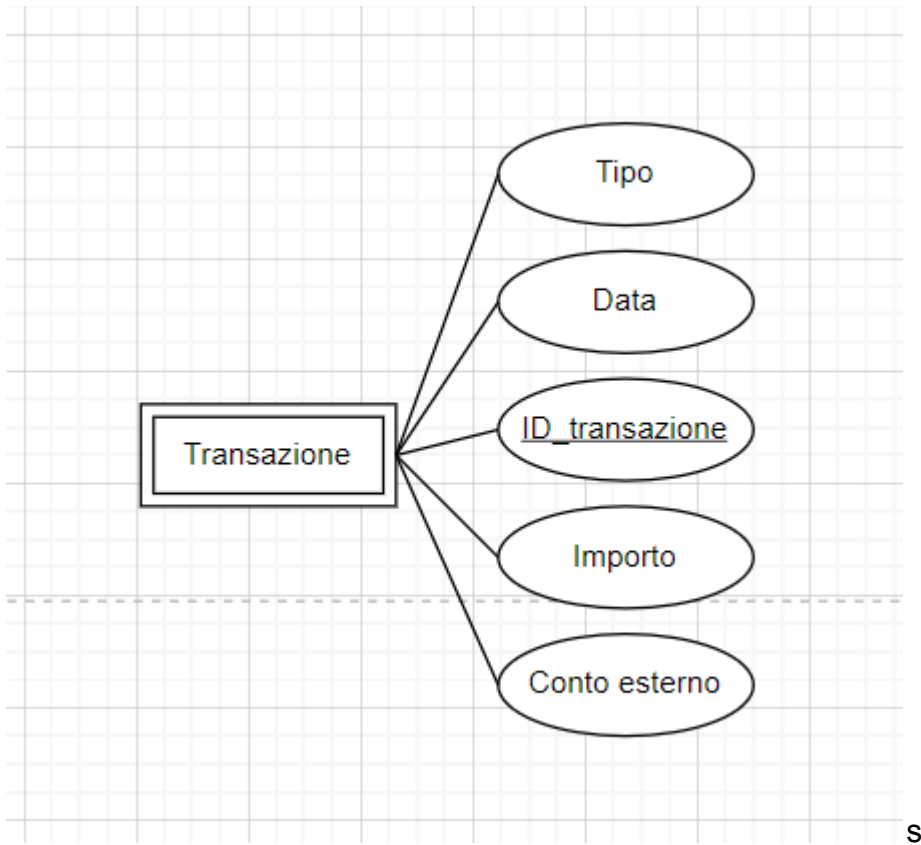
il risultato di questa ristrutturazione è il seguente:



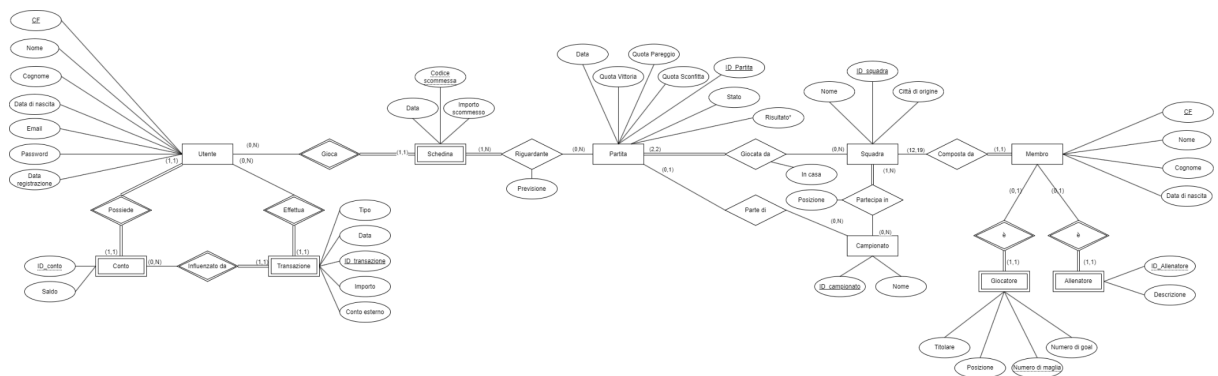
Infine è presente un'altra specializzazione su "Transazione":



La seguente specializzazione viene modificata come segue poichè non è presente nessun attributo nelle sotto entità di transazione ed inoltre non partecipano a nessuna relazione in maniera diretta.



Schema EER ristrutturato



Schema relazionale

Si procede al mapping della base di dati:

Membro(CF, Nome, Cognome, Data_di_nascita, Squadra.ID_Squadra↑,)
Giocatore(Membro.CF↑, Numero_di_maglia, Numero_di_goal, Posizione, Titolare)
Allenatore(ID_Allenatore, Membro.CF↑, Descrizione)
Squadra(ID_Squadra, Nome, Città_di_Origine)
Campionato(ID_Campionato, Nome)
Partecipa_in(ID_Squadra↑, ID_Campionato↑, posizione)
Giocata_da(ID_Partita↑, ID_squadra↑, in_casa)
Partita(ID_Partita, Data, Quota_vittoria, Quota_pareggio, Quota_sconfitta, stato, Risultato*)
Parte_di(ID_Partita↑, ID_Campionato↑)
Utente(CF, Nome, Cognome, Data_di_nascita, Email, Password, Data_registrazione)
Conto(ID_Conto, Utente.CF↑, Saldo)
Transazione(Conto.ID_conto↑, Conto.Utente.CF↑, Utente.CF↑, ID_Transazione, Data, Importo, Conto_esterno, Tipo)
Schedina(Codice_Scommessa, Utente.CF↑, Data, Importo_scommesso)
Riguardante(Schedina.Codice_Scommessa↑, Partita.ID_Partita↑, Previsione)

Normalizzazione

Il database si presenta già normalizzato. È infatti in 1NF in quanto tutti gli attributi sono atomici dopo la ristrutturazione. È in 2NF perché, oltre ad essere già in 1NF, quando è presente una chiave primaria composta da più attributi tutte le dipendenze funzionali che la riguardano sono piene e non parziali. È in 3NF perché, oltre ad essere già in 2NF, in tutte le tabelle non sono presenti dipendenze transitive fra attributi non chiave e la chiave primaria.

5. Realizzazione della base di dati con MySQL

Di seguito il listato delle istruzioni MySQL per la creazione della base di dati:

```
CREATE TABLE Membro (  
    CF VARCHAR(16) PRIMARY KEY,  
    Nome VARCHAR(100) NOT NULL,  
    Cognome VARCHAR(100) NOT NULL,  
    Data_di_nascita DATE NOT NULL,  
    Squadra_ID INT  
);
```

```
CREATE TABLE Giocatore (  
    Membro_CF VARCHAR(16) NOT NULL,  
    Numero_di_maglia INT NOT NULL,  
    Numero_di_goal INT NOT NULL,  
    Posizione VARCHAR(50) NOT NULL,  
    Titolare BOOLEAN NOT NULL,  
    PRIMARY KEY (Membro_CF, Numero_di_maglia),  
    FOREIGN KEY (Membro_CF) REFERENCES Membro(CF)  
);
```

```
CREATE TABLE Allenatore (  
    ID_Allenatore INT NOT NULL,  
    Membro_CF VARCHAR(16) NOT NULL,  
    Descrizione TEXT NOT NULL,  
    PRIMARY KEY (ID_Allenatore, Membro_CF),  
    FOREIGN KEY (Membro_CF) REFERENCES Membro(CF)  
);
```

```
CREATE TABLE Squadra (  
    ID_Squadra INT PRIMARY KEY,  
    Nome VARCHAR(100) NOT NULL,  
    Citta_di_Origine VARCHAR(100) NOT NULL  
);
```

```
CREATE TABLE Campionato (  
    ID_Campionato INT PRIMARY KEY,  
    Nome VARCHAR(100) NOT NULL  
);
```

```
CREATE TABLE Partecipa_in (  
    ID_Squadra INT NOT NULL,  
    ID_Campionato INT NOT NULL,  
    posizione INT NOT NULL,  
    PRIMARY KEY (ID_Squadra, ID_Campionato),  
    FOREIGN KEY (ID_Squadra) REFERENCES Squadra(ID_Squadra),  
    FOREIGN KEY (ID_Campionato) REFERENCES Campionato(ID_Campionato)  
);
```

```
CREATE TABLE Giocata_da (  
    ID_Partita INT NOT NULL,  
    ID_squadra INT NOT NULL,  
    in_casa BOOLEAN NOT NULL,  
    PRIMARY KEY (ID_Partita, ID_squadra),  
    FOREIGN KEY (ID_squadra) REFERENCES Squadra(ID_Squadra)  
);
```

```
CREATE TABLE Partita (  
    ID_Partita INT PRIMARY KEY,  
    Data DATE NOT NULL,  
    Quota_vittoria DECIMAL(5,2) NOT NULL,  
    Quota_pareggio DECIMAL(5,2) NOT NULL,  
    Quota_sconfitta DECIMAL(5,2) NOT NULL,  
    stato VARCHAR(50) NOT NULL,  
    Risultato VARCHAR(50)  
);
```

```
CREATE TABLE Parte_di (  
    ID_Partita INT NOT NULL,  
    ID_Campionato INT NOT NULL,  
    PRIMARY KEY (ID_Partita, ID_Campionato),  
    FOREIGN KEY (ID_Partita) REFERENCES Partita(ID_Partita),  
    FOREIGN KEY (ID_Campionato) REFERENCES Campionato(ID_Campionato)  
);
```

```
CREATE TABLE Utente (  
    CF VARCHAR(16) PRIMARY KEY,  
    Nome VARCHAR(100) NOT NULL,  
    Cognome VARCHAR(100) NOT NULL,  
    Data_di_nascita DATE NOT NULL,  
    Email VARCHAR(100) NOT NULL,  
    Password VARCHAR(100) NOT NULL,  
    Data_registrazione DATE NOT NULL  
);
```

```
CREATE TABLE Conto (  
    ID_Conto INT PRIMARY KEY,  
    Utente_CF VARCHAR(16) NOT NULL,  
    Saldo DECIMAL(10,2) NOT NULL,  
    FOREIGN KEY (Utente_CF) REFERENCES Utente(CF)  
);
```

```

CREATE TABLE Transazione (
    ID_Transazione INT PRIMARY KEY,
    Conto_ID_conto INT NOT NULL,
    Conto_Utente_CF VARCHAR(16) NOT NULL,
    Utente_CF VARCHAR(16) NOT NULL,
    Data DATE NOT NULL,
    Importo DECIMAL(10,2) NOT NULL,
    Conto_esterno VARCHAR(50) NOT NULL,
    Tipo VARCHAR(50) NOT NULL,
    FOREIGN KEY (Conto_ID_conto) REFERENCES Conto(ID_Conto),
    FOREIGN KEY (Conto_Utente_CF) REFERENCES Conto(Utente_CF),
    FOREIGN KEY (Utente_CF) REFERENCES Utente(CF)
);

CREATE TABLE Schedina (
    Codice_Scommessa INT NOT NULL,
    Utente_CF VARCHAR(16) NOT NULL,
    Data DATE NOT NULL,
    Importo_scommesso DECIMAL(10,2) NOT NULL,
    PRIMARY KEY(Codice_Scommessa, Utente_CF),
    FOREIGN KEY (Utente_CF) REFERENCES Utente(CF)
);

CREATE TABLE Riguardante (
    Codice_Scommessa INT NOT NULL,
    Utente_CF VARCHAR(16) NOT NULL,
    ID_Partita INT NOT NULL,
    Previsione VARCHAR(50) NOT NULL,
    PRIMARY KEY (Codice_Scommessa, Utente_CF, ID_Partita),
    FOREIGN KEY (Codice_Scommessa, Utente_CF) REFERENCES Schedina(Codice_Scommessa, Utente_CF),
    FOREIGN KEY (ID_Partita) REFERENCES Partita(ID_Partita)
);

```

6. Implementazione query SQL

1. Inserimento di un nuovo utente

INSERT INTO Utente (CF, Nome, Cognome, Data_di_nascita, Email, Password, Data_registrazione)

VALUES ('[codice fiscale]', '[nome]', '[cognome]', '[data di nascita]', '[email]', '[password]', CURDATE());

INSERT INTO Conto (ID_Conto, Utente_CF, Saldo)

VALUES ('[id conto]', '[codice fiscale]', 0.00);

2. Visualizzare lo storico schedine di un utente

```
SELECT * FROM Schedina  
WHERE Utente_CF = '[codice fiscale utente]';
```

3. Visualizzare le transazioni di un utente

```
SELECT * FROM Transazione  
WHERE Utente_CF = '[codice fiscale utente]';
```

4. Visualizzare le credenziali di un utente

```
SELECT CF, Nome, Cognome, Email, Password FROM Utente  
WHERE CF = '[codice fiscale utente]';
```

5. Visualizzare il saldo di un utente

```
SELECT Saldo FROM Conto  
WHERE Utente_CF = '[codice fiscale utente]';
```

6. Inserimento di una transazione relativa ad un utente

```
INSERT INTO Transazione (ID_Transazione, Conto_ID_conto, Conto_Utente_CF,  
Utente_CF, Data, Importo, Conto_esterno, Tipo)  
VALUES ('[id transazione]', '[id conto]', '[conto utente cf]', '[utente cf]', CURDATE(),  
[importo], '[conto esterno]', '[tipo]');
```

```
UPDATE Conto SET Saldo = Saldo - ? WHERE ID_Conto = ?
```

```
UPDATE Conto SET Saldo = Saldo + ? WHERE ID_Conto = ?
```


7. Inserimento di una schedina relativa ad un utente

Poichè l'operazione non era traducibile in una sola operazione sql è stata divisa in due

Creazione di una schedina:

```
INSERT INTO Schedina (Codice_Scommessa, Utente_CF, Data,  
Importo_scommesso)  
VALUES ('[codice scommessa]', '[utente cf]', CURDATE(), [importo scommesso]);
```

Inserimento delle partite all'interno di una schedina:

```
INSERT INTO Riguardante (Codice_Scommessa, Utente_CF, ID_Partita,  
Previsione)  
VALUES ('[codice scommessa]', '[codice fiscale utente]', '[id partita]', '[previsione]');
```

8. Visualizzazione di una schedina

```
SELECT  
S.Codice_Scommessa, S.Utente_CF, S.Data, S.Importo_scommesso,  
P.ID_Partita, P.Data as Data_Partita, P.Risultato,  
R.Previsione,  
SQ1.Nome as Squadra_Casa, SQ2.Nome as Squadra_Ospite  
FROM  
Schedina S  
JOIN  
Riguardante R ON S.Codice_Scommessa = R.Codice_Scommessa  
JOIN  
Partita P ON R.ID_Partita = P.ID_Partita  
JOIN  
Giocata_da G1 ON P.ID_Partita = G1.ID_Partita AND G1.in_casa = TRUE  
JOIN  
Giocata_da G2 ON P.ID_Partita = G2.ID_Partita AND G2.in_casa = FALSE  
JOIN  
Squadra SQ1 ON G1.ID_squadra = SQ1.ID_Squadra  
JOIN  
Squadra SQ2 ON G2.ID_squadra = SQ2.ID_Squadra  
WHERE  
S.Codice_Scommessa = '[codice scommessa]';
```

9. Inserimento di una partita

```
INSERT INTO Partita (ID_Partita, Data, Quota_vittoria, Quota_pareggio,  
Quota_sconfitta, stato)  
VALUES ('[id partita]', '[data]', [quota vittoria], [quota pareggio], [quota sconfitta],  
'[stato]');
```

```
INSERT INTO Giocata_da (ID_Partita, ID_squadra, in_casa)  
VALUES ('[id partita]', '[id squadra 1]', [true o false]);
```

```
INSERT INTO Giocata_da (ID_Partita, ID_squadra, in_casa)  
VALUES ('[id partita]', '[id squadra 2]', [true o false]);
```

10. Inserimento di una partita riguardante un campionato

```
INSERT INTO Partita (ID_Partita, Data, Quota_vittoria, Quota_pareggio,  
Quota_sconfitta, stato)  
VALUES ('[id partita]', '[data]', [quota vittoria], [quota pareggio], [quota sconfitta],  
'[stato]');
```

```
INSERT INTO Parte_di (ID_Partita, ID_Campionato)  
VALUES ('[id partita]', '[id campionato]');
```

Inserire le due squadre nella tabella Giocata_da:

```
INSERT INTO Giocata_da (ID_Partita, ID_squadra, in_casa)  
VALUES ('[id partita]', '[id squadra 1]', [true o false]);
```

```
INSERT INTO Giocata_da (ID_Partita, ID_squadra, in_casa)  
VALUES ('[id partita]', '[id squadra 2]', [true o false]);
```

11. Conclusione di una partita e inserimento del risultato

```
UPDATE Partita  
SET stato = 'conclusa'  
SET risultato = '[risultato]'  
WHERE ID_Partita = '[id partita]';
```

12. Inserimento di un campionato

```
INSERT INTO Campionato (ID_Campionato, Nome)
VALUES ('[id campionato]', '[nome]');
```

13. Visualizzare le squadre relative ad un campionato

```
SELECT s.ID_Squadra, s.Nome, s.Citta_di_Origine, p.posizione
FROM Squadra s
JOIN Partecipa_in p ON s.ID_Squadra = p.ID_Squadra
WHERE p.ID_Campionato = '[id campionato]';
```

14. Registrazione di una squadra ed inserimento nel relativo campionato

```
INSERT INTO Squadra (ID_Squadra, Nome, Citta_di_Origine)
VALUES ('[id squadra]', '[nome]', '[citt  di origine]');
INSERT INTO Partecipa_in (ID_Squadra, ID_Campionato, posizione)
VALUES ('[id squadra]', '[id campionato]', [posizione]);
```

15. Visualizzare tutte le partite di una squadra

```
SELECT Partita.*
FROM Partita
JOIN Giocata_da ON Partita.ID_Partita = Giocata_da.ID_Partita
WHERE Giocata_da.ID_squadra = '[id squadra]';
```

16. Aggiornamento della posizione di una squadra in un campionato

```
UPDATE Partecipa_in
SET posizione = [nuova posizione]
WHERE ID_Squadra = '[id squadra]' AND ID_Campionato = '[id campionato]';
```

17. Visualizzare una partita e le squadre che la giocano

```
SELECT Partita.*, Squadra.Nome
FROM Partita
JOIN Giocata_da ON Partita.ID_Partita = Giocata_da.ID_Partita
JOIN Squadra ON Giocata_da.ID_squadra = Squadra.ID_Squadra
```

WHERE Partita.ID_Partita = '[id partita]';

18. Inserimento di un giocatore in una squadra

```
INSERT INTO Membro (CF, Nome, Cognome, Data_di_nascita, Squadra_ID)
VALUES ('[cf]', '[nome]', '[cognome]', '[data di nascita]', '[squadra id]');
```

```
INSERT INTO Giocatore (Membro_CF, Numero_di_maglia, Numero_di_goal,
Posizione, Titolare)
VALUES ('[cf]', [numero di maglia], [numero di goal], '[posizione]', [titolare]);
```

19. Inserimento di un Allenatore in una squadra

```
INSERT INTO Membro (CF, Nome, Cognome, Data_di_nascita, Squadra_ID)
VALUES ('[cf]', '[nome]', '[cognome]', '[data di nascita]', '[squadra id]');
```

```
INSERT INTO Allenatore (ID_Allenatore, Membro_CF, Descrizione)
VALUES ('[id allenatore]', '[cf]', '[descrizione]');
```

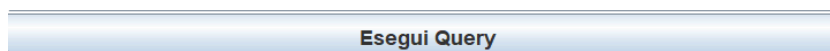
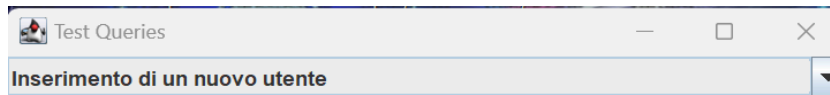
20. Visualizzazione dei membri di una squadra

```
SELECT
    M.CF, M.Nome, M.Cognome, M.Data_di_nascita, M.Squadra_ID,
    G.Numero_di_maglia, G.Numero_di_goal, G.Posizione, G.Titolare,
    A.ID_Allenatore, A.Descrizione
FROM
    Membro M
LEFT JOIN
    Giocatore G ON M.CF = G.Membro_CF
LEFT JOIN
    Allenatore A ON M.CF = A.Membro_CF
WHERE
    M.Squadra_ID = [id squadra];
```

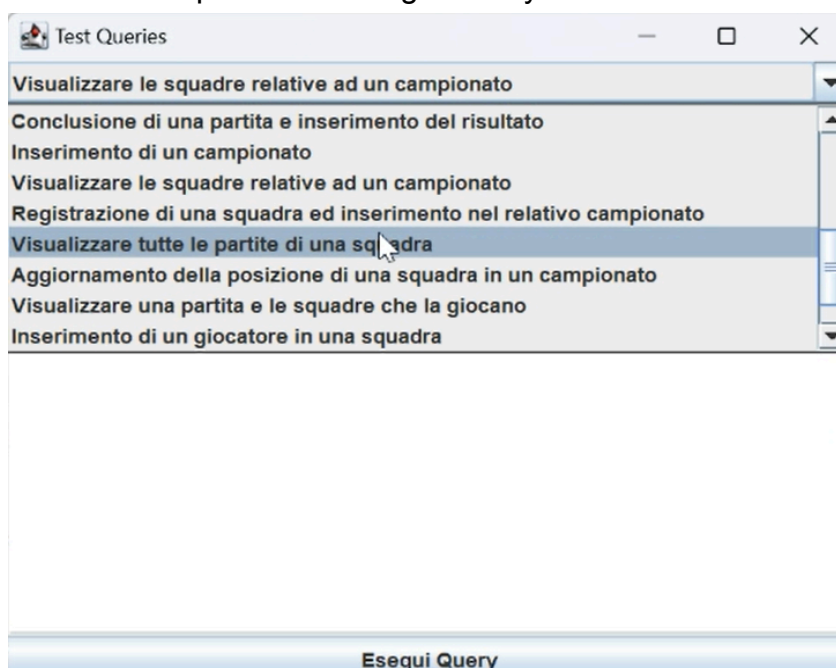
7. Test dell'applicazione Java

L'applicazione è pensata per essere utilizzata dai DBD (Database Designer) o dall'Amministratore per testare le funzionalità della query prima di implementarle nella piattaforma finale

All'avvio dell'applicazione, si ha l'immediata possibilità di provare il funzionamento delle varie query sul Database creato in MySQL.



Tramite un menù a tendina si potrà scegliere quella che interessa, eseguendola cliccando sul pulsante: "Esegui Query".



Verranno chiesti in input i dati relativi alla query scelta e nella casella di testo centrale verrà visualizzato il risultato.

