

DMI CUP

PROGETTO DATABASE

VINCENZO VILLANOVA 1000031067



Università
di Catania



Indice

○ <u>Descrizione del progetto</u>	<u>2</u>
○ <u>Descrizione degli scenari</u>	<u>2</u>
○ <u>Glossario dei termini</u>	<u>3</u>
○ <u>Progettazione concettuale</u>	<u>4</u>
○ <u>Schema scheletro</u>	<u>4</u>
○ <u>Schema scheletro completo</u>	<u>6</u>
○ <u>Vincoli non esprimibili sullo schema E-R</u>	<u>7</u>
○ <u>Porzione del dizionario dati – Entità</u>	<u>8</u>
○ <u>Porzione del dizionario dati – Relazioni</u>	<u>8</u>
○ <u>Ristrutturazione dello schema E-R</u>	<u>9</u>
○ <u>Operazioni</u>	<u>9</u>
○ <u>Tabella dei volumi</u>	<u>10</u>
○ <u>Tabella delle frequenze</u>	<u>10</u>
○ <u>Traduzione verso il modello relazionale</u>	<u>11</u>
○ <u>Progettazione logica</u>	<u>11</u>
○ <u>Tabella di Peso delle Istanze</u>	<u>12</u>
○ <u>Progettazione fisica</u>	<u>14</u>
○ <u>Creazione delle tabelle in SQL</u>	<u>14</u>
○ <u>Trigger ed interrogazioni SQL</u>	<u>16</u>
○ <u>Conclusioni e ringraziamenti</u>	<u>19</u>

NB: **Attraverso** l'impiego dei **collegamenti ipertestuali**, è possibile esplorare la relazione **cliccando** direttamente su un **paragrafo** ben specificato nell'**indice**.

Descrizione del progetto

Si vuole realizzare il **database** relativo alla gestione di un torneo di **calcio a cinque** con lo scopo di gestire le informazioni relative alle **partite**, inclusi i **giocatori**, le **squadre**, gli **arbitri** e gli **stadi** coinvolti.

Descrizione degli scenari

Per garantire la realizzazione ottimizzata e coerente del **database**, è essenziale considerare attentamente gli scenari delle persone coinvolte nel **progetto**. Tale approccio permette di creare un sistema di gestione dei dati coerente con gli obiettivi del progetto, facilitando l'organizzazione, l'analisi e l'utilizzo delle informazioni rilevanti per le diverse parti interessate.

Giocatore

Questa tabella contiene informazioni sui **giocatori** che prendono parte ad una **squadra**. Ogni riga rappresenta un giocatore e contiene dettagli come l'ID del giocatore, il nome, il ruolo, la data di nascita, la nazionalità e l'ID della squadra a cui il giocatore appartiene.

Squadra

Questa tabella raccoglie i dettagli sulle **squadre** coinvolte nelle **partite**. Ogni riga rappresenta una squadra e contiene dati come l'ID della squadra, il nome, l'anno di fondazione, l'allenatore principale e altri dettagli specifici della squadra.

Partita

Questa tabella registra le informazioni relative alle **partite**. Ogni riga rappresenta una singola partita e contiene dati come l'ID della partita, la data, il punteggio finale, l'ID dello **stadio** in cui si è svolta la partita e l'ID dell'**arbitro** che ha diretto l'incontro.

Arbitro

Questa tabella tiene traccia delle informazioni sugli **arbitri** che dirigono le **partite**. Ogni riga rappresenta un arbitro e contiene dettagli come l'ID dell'arbitro, il nome, la data di nascita, la nazionalità.

Allenatore

Questa tabella è dedicata alla registrazione dei dettagli sugli **allenatori** che dirigono le **squadre** di calcio. Ogni riga in questa tabella rappresenta un allenatore e contiene informazioni essenziali come l'ID univoco dell'allenatore, il nome completo, la data di nascita, la nazionalità.

Stadio

Questa tabella registra i dettagli sugli **stadi** in cui si svolgono le **partite**. Ogni riga rappresenta uno stadio e contiene informazioni come l'ID dello stadio, il nome, la capacità, la località.

Glossario dei termini

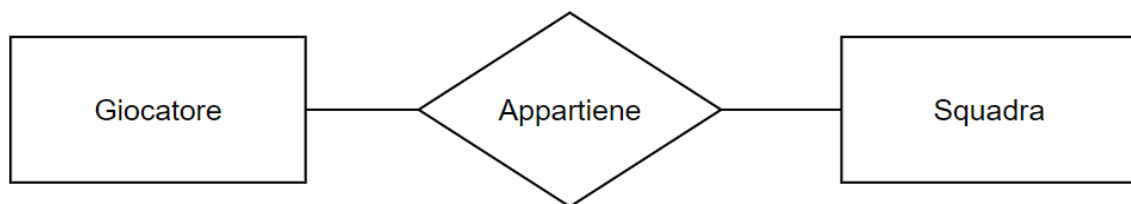
Entità	Descrizione	Sinonimo	Relazioni
Giocatore	Componente di una squadra	Atleta	Giocatore appartiene a Squadra
Squadra	Insieme di giocatori	Team	Squadra gioca Partita
Partita	Incontro tra due squadre	Match	Partita viene giocata da Squadre
Arbitro	Direttore di gara	Giudice di gara	Ogni Partita viene arbitrata da un Arbitro
Allenatore	Colui che guida una squadra	Mister	Ogni Squadra viene allenata da un Allenatore
Stadio	Struttura dedicata allo svolgimento delle partite	Campo di gioco	Ogni Partita viene giocata in uno Stadio

Progettazione concettuale

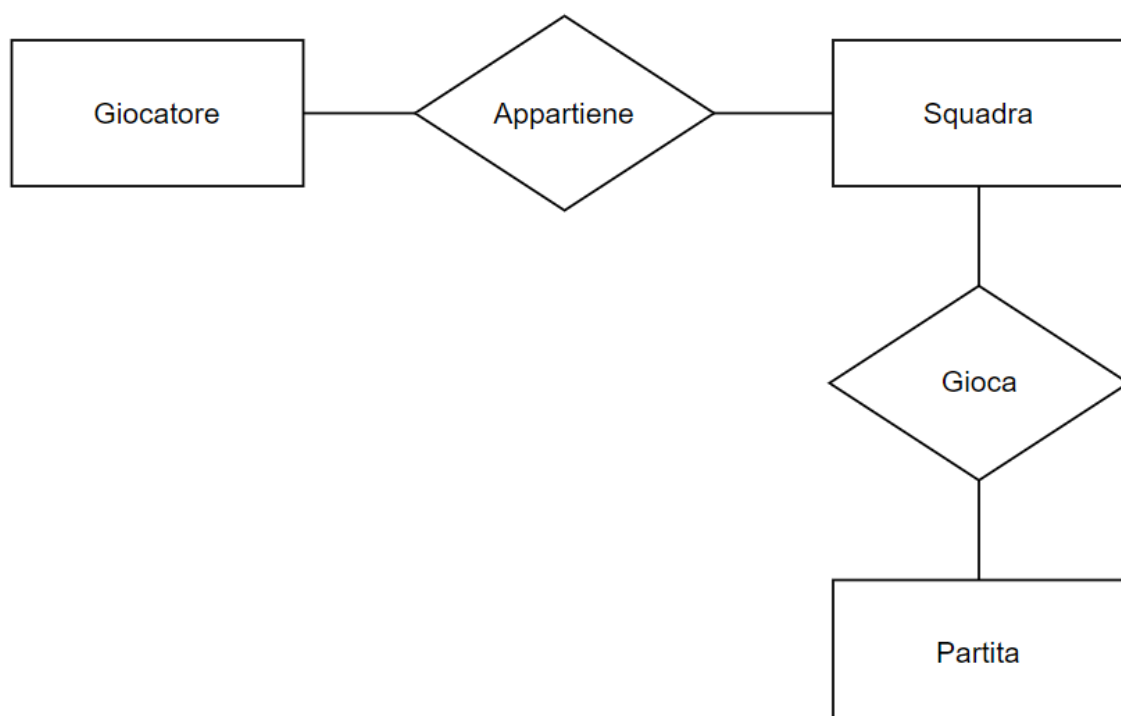
Per la strategia implementativa si opta per una strategia **top-down**. Le primitive di trasformazione **top-down** sono regole che operano su un singolo concetto dello schema e lo trasformano in una struttura più complessa che descrive il concetto con maggiore dettaglio. Si redige in breve tempo il progetto concettuale della base dati **senza** preoccuparsi dei **dettagli** descrivendone in modo astratto e generale la realtà di interesse. In questo modo si ha una **visione generale** delle componenti **dell'intero sistema**.

Schema scheletro

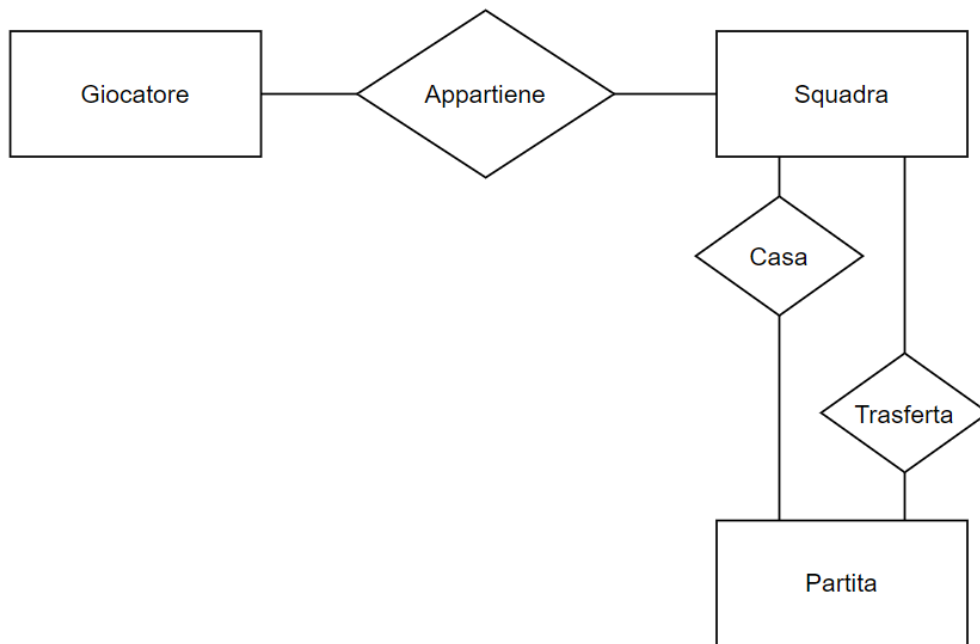
Si individua l'entità della **squadra** e si intende rappresentare la partecipazione di un **giocatore** ad essa.



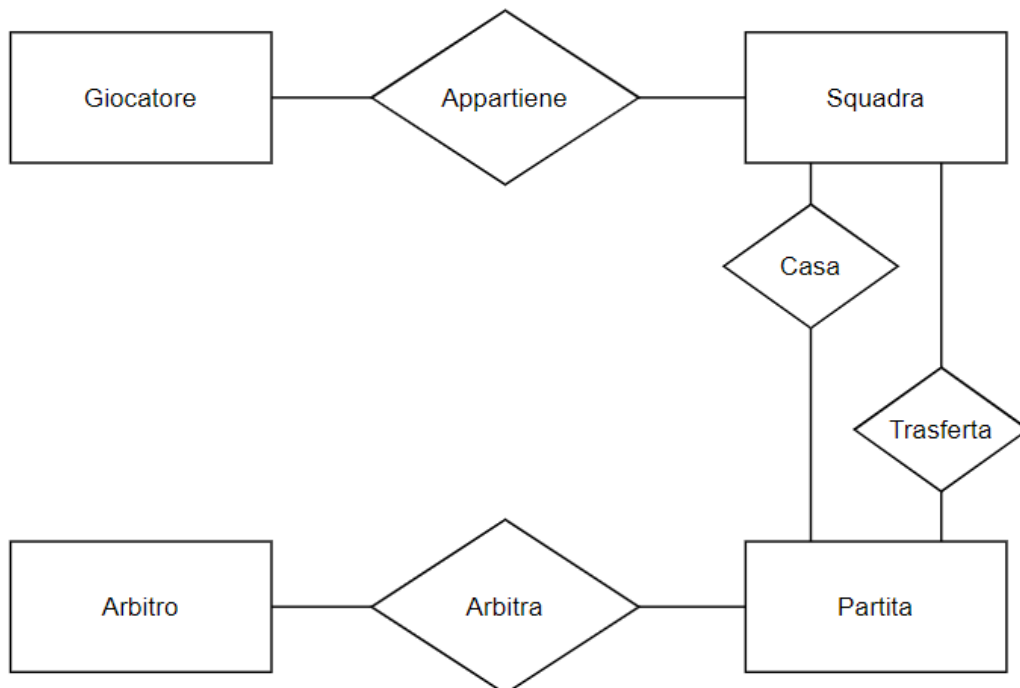
Si necessita di introdurre l'entità **partita** per tenere traccia dei risultati di una **squadra**.



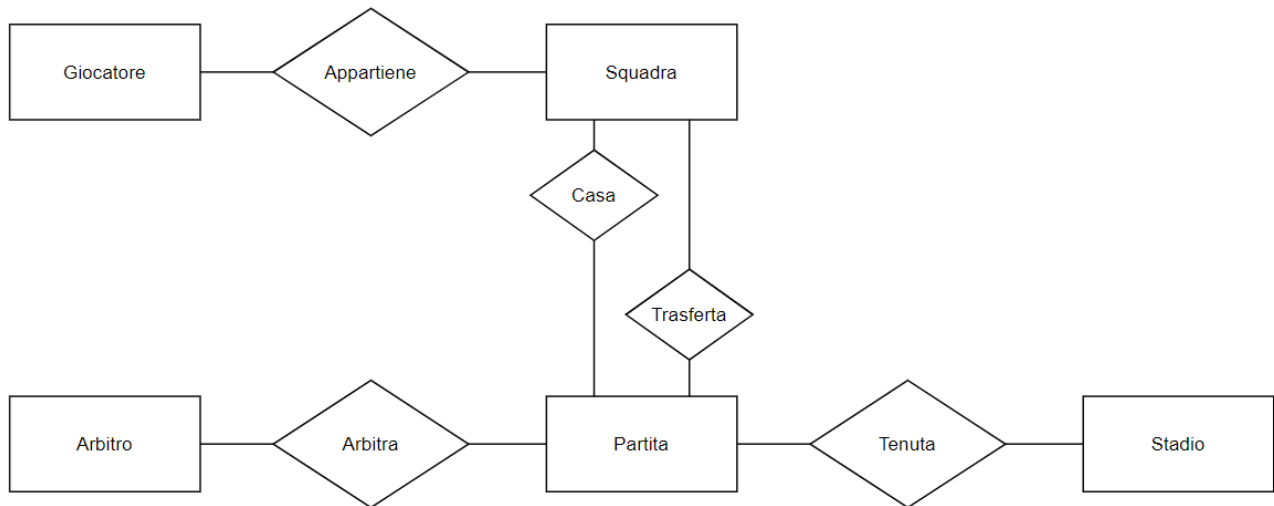
Si nota che per ogni singola **partita** giocano due **squadre** e non una sola.



Abbiamo la necessità di immettere l'entità **arbitro** che avrà il ruolo di dirigere la **partita** affinché i regolamenti previsti vengano rispettati.

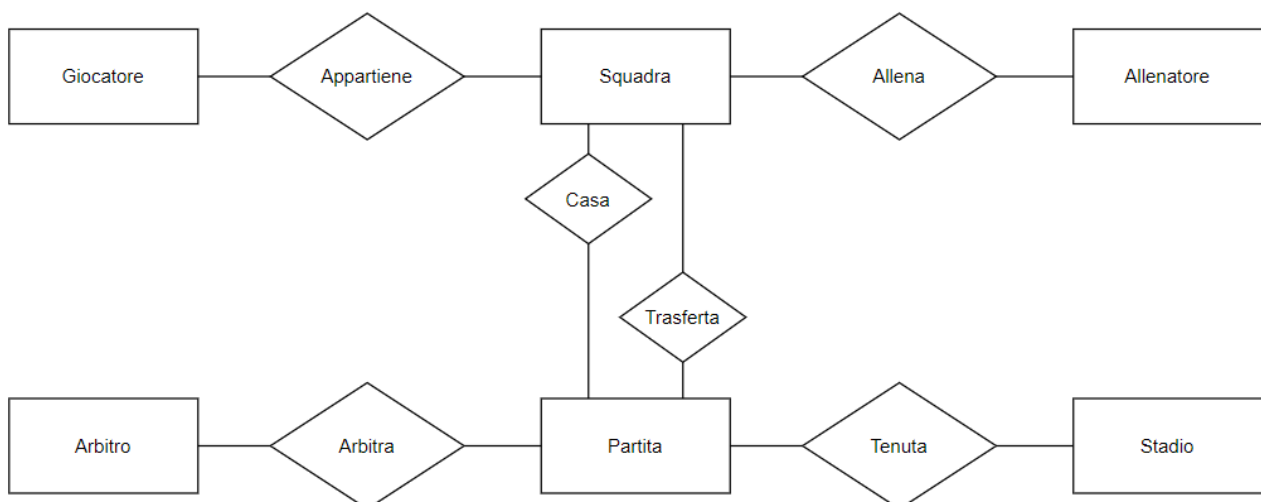


Introduciamo l'entità **stadio** per avere ulteriori informazioni sulla località in cui si è disputata la **partita**.



Schema scheletro completo

Per avere maggiori informazioni sulle squadre, è necessario aggiungere l'entità **Allenatore** al database. Questa aggiunta consente di associare gli allenatori alle singole **squadre** e di gestire le informazioni in modo più completo.



Vincoli non esprimibili sullo schema E-R

Vincoli sui giocatori

- Ogni **giocatore** può partecipare esclusivamente in una singola **squadra**
- Ogni **giocatore** ha un **ruolo** specifico assegnato (portiere, difensore, centrocampista o attaccante).

Vincoli sulle squadre

- Ogni **squadra** può giocare una **partita** alla settimana
- Ogni entità **squadra** possiede un attributo punti che viene aggiornato automaticamente tramite un **trigger** ogni volta che la squadra disputa una **partita**:
 - ⚽ Si aggiungono 0 punti in caso di **sconfitta**.
 - ⚽ Si aggiungono 1 punto in caso di **pareggio**.
 - ⚽ Si aggiungono 3 punti in caso di **vittoria**.

Vincoli sugli allenatori

- Ad ogni **allenatore** è consentito allenare una sola **squadra**.

Vincoli sulle partite

- Ogni **partita** richiede la partecipazione simultanea di due **squadre** diverse.

Vincoli sugli arbitri

- Un **arbitro** può arbitrare al massimo una **partita** alla settimana

Vincoli sugli stadi

- Non è possibile disputare più **partite** contemporaneamente nello stesso **stadio**.

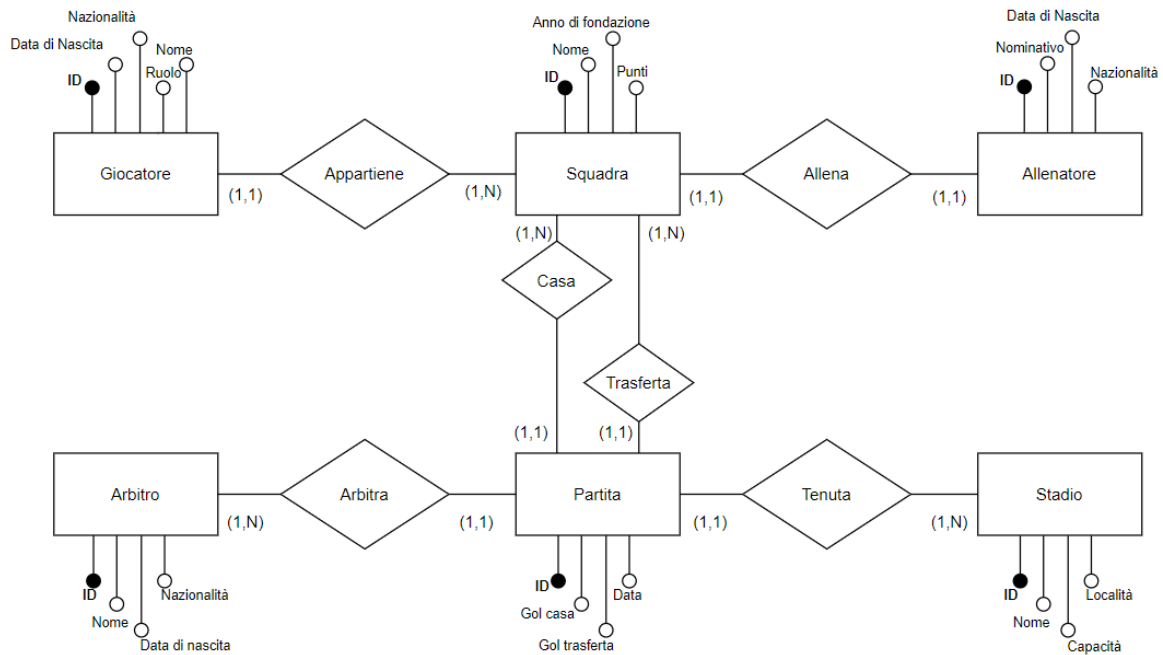
Porzione del dizionario dati – Entità

Entità	Descrizione	Attributi	Identificatore
Giocatore	Componente di una squadra	ID giocatore, nome, ruolo, data di nascita, nazionalità	ID giocatore
Squadra	Insieme di giocatori	ID squadra, nome, anno di fondazione, punti	ID squadra
Partita	Incontro tra due squadre	ID partita, data, gol squadra casa, gol squadra ospite	ID partita
Arbitro	Direttore di gara	ID arbitro, nome, data di nascita, nazionalità	ID arbitro
Allenatore	Colui che guida una squadra	ID allenatore, nominativo, data di nascita, nazionalità	ID allenatore
Stadio	Struttura dedicata allo svolgimento delle partite	ID stadio, nome, capacità, località	ID stadio

Porzione del dizionario dati – Relazioni

Relazione	Entità partecipanti	Descrizione	Attributi
Appartiene	Giocatore, Squadra	Ogni giocatore fa parte di una squadra	-
Allena	Squadra, Allenatore	Ogni squadra viene allenata da un allenatore	-
Arbitra	Arbitro, Partita	Ogni arbitro arbitra una partita	-
Tenuta	Partita, Stadio	Ogni partita viene tenuta in uno stadio	-
Casa	Partita, Squadra	Una partita viene disputata da due squadre: una che gioca in casa e l'altra come ospite .	-
Ospite	Partita, Squadra		-

Ristrutturazione dello schema E-R



Operazioni

1. Inserire una nuova partita
2. Visualizzare la classifica generale
3. Visualizzare i giocatori della squadra vincente
4. Visualizzare l'allenatore di una squadra vincente
5. Visualizzare tutte le partite giocate da una squadra specifica
6. Visualizzare le squadre con più gol fatti in casa
7. Visualizzare le squadre con più gol fatti fuori casa
8. Visualizzare le squadre con più gol effettuati

Tabella dei volumi

Concetto	Tipo	Volume
Giocatore	E	60
Squadra	E	10
Partita	E	25
Arbitro	E	8
Allenatore	E	10
Stadio	E	8
Appartiene	R	1
Allena	R	1
Arbitra	R	1
Tenuta	R	1
Casa	R	1
Ospite	R	1

Tabella delle frequenze

Operazione	Descrizione	Frequenza	Tipo
O1	Inserire una nuova partita	3/settimana	I
O2	Visualizzare la classifica generale	2/giorno	I
O3	Visualizzare i giocatori della squadra vincente	1/settimana	I
O4	Visualizzare l'allenatore della squadra vincente	1/anno	I
O5	Visualizzare tutte le partite giocate da una squadra specifica	4/anno	I
O6	Visualizzare le squadre con più gol fatti in casa	1/anno	I
O7	Visualizzare le squadre con più gol fatti fuori casa	1/anno	I
O8	Visualizzare le squadre con più gol effettuati	1/anno	I

Traduzione verso il modello relazionale

Nella traduzione a modello relazionale sono state eliminate le relazioni:

Appartiene → ID **squadra** nell'entità **giocatore**

Allena → ID **allenatore** nell'entità **squadra**

Arbitra → ID **arbitro** nell'entità **partita**

Tenuta → ID **stadio** nell'entità **partita**

Casa → ID **casa** nell'entità **partita**

Ospite → ID **ospite** nell'entità **partita**

→ = Viene sostituita da

Progettazione Logica

Schema Logico

Completata la Progettazione Concettuale, definiamo lo Schema Logico

- ⚽ **Giocatore** (ID, nome, ruolo, data di nascita, nazionalità, ID_squadra)
- ⚽ **Squadra** (ID, nome, anno di fondazione, punti, IDAllenatore)
- ⚽ **Allenatore** (ID, nominativo, data di nascita, nazionalità)
- ⚽ **Arbitro** (ID, nome, data di nascita, nazionalità)
- ⚽ **Partita** (ID, ID_casa, ID_trasferta, gol casa, gol trasferta, data, ID_stadio, ID_arbitro)
- ⚽ **Stadio** (ID, nome, capacità, località)

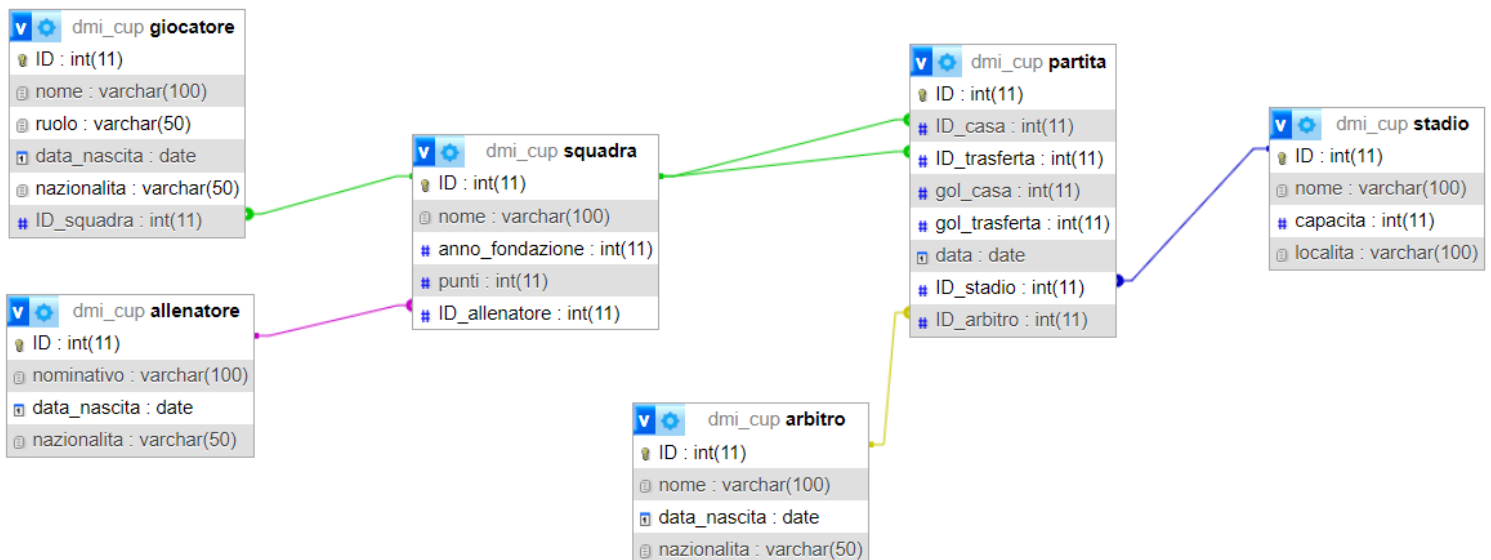
Tabella di Peso delle Istanze

Tabella	Attributi	Byte	Peso totale
Giocatore	ID	4	211
	nome	100	
	ruolo	50	
	data_nascita	3	
	nazionalità	50	
	ID_squadra	4	
Squadra	ID	4	116
	nome	100	
	Anno_fondazione	4	
	punti	4	
	IDAllenatore	4	
Allenatore	ID	4	157
	nominativo	100	
	Data_nascita	3	
	nazionalità	50	

Tabella	Attributi	Byte	Peso totale
Arbitro	ID	4	157
	nome	100	
	data_nascita	3	
	nazionalità	50	
Partita	ID	4	31
	ID_casa	4	
	ID_trasferta	4	
	gol_casa	4	
	gol_trasferta	4	
	data	3	
	ID_stadio	4	
	ID_arbitro	4	
Stadio	ID	4	208
	nome	100	
	capacità	4	
	località	100	

Progettazione Fisica

Viene allegata l'implementazione fisica dello schema successivamente definito in file.



Creazione delle tabelle – Implementazione DB in SQL

Giocatore

```
CREATE TABLE `giocatore` (  
  `ID` int(11) NOT NULL,  
  `nome` varchar(100) DEFAULT NULL,  
  `ruolo` varchar(50) DEFAULT NULL,  
  `data_nascita` date DEFAULT NULL,  
  `nazionalita` varchar(50) DEFAULT NULL,  
  `ID_squadra` int(11) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
```

Squadra

```
CREATE TABLE `squadra` (  
  `ID` int(11) NOT NULL,  
  `nome` varchar(100) DEFAULT NULL,  
  `anno_fondazione` int(11) DEFAULT NULL,  
  `punti` int(11) DEFAULT NULL,  
  `IDAllenatore` int(11) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
```

Allenatore

```
CREATE TABLE `allenatore` (  
  `ID` int(11) NOT NULL,  
  `nominativo` varchar(100) DEFAULT NULL,  
  `data_nascita` date DEFAULT NULL,  
  `nazionalita` varchar(50) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
```

Arbitro

```
CREATE TABLE `arbitro` (  
  `ID` int(11) NOT NULL,  
  `nome` varchar(100) DEFAULT NULL,  
  `data_nascita` date DEFAULT NULL,  
  `nazionalita` varchar(50) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
```

Partita

```
CREATE TABLE `partita` (  
  `ID` int(11) NOT NULL,  
  `ID_casa` int(11) DEFAULT NULL,  
  `ID_trasferta` int(11) DEFAULT NULL,  
  `gol_casa` int(11) DEFAULT NULL,  
  `gol_trasferta` int(11) DEFAULT NULL,  
  `data` date DEFAULT NULL,  
  `ID_stadio` int(11) DEFAULT NULL,  
  `ID_arbitro` int(11) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
```


Stadio

```
CREATE TABLE `stadio` (  
  `ID` int(11) NOT NULL,  
  `nome` varchar(100) DEFAULT NULL,  
  `capacita` int(11) DEFAULT NULL,  
  `localita` varchar(100) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
```

Per vedere anche l'inserimento dei dati consultare il file: **dmi_cup.sql**

Trigger ed interrogazioni in SQL

È stato definito un trigger per l'aggiornamento dell'attributo **punti** presente nell'entità squadre dopo aver effettuato un'operazione di inserimento nella tabella **partite**.

```
CREATE TRIGGER `aggiorna_punti_squadra` AFTER INSERT ON `partita` FOR EACH ROW  
BEGIN  
  DECLARE casa_punti INT;  
  DECLARE trasferta_punti INT;  
  
  IF NEW.gol_casa > NEW.gol_trasferta THEN  
    SET casa_punti = 3;  
    SET trasferta_punti = 0;  
  ELSEIF NEW.gol_casa = NEW.gol_trasferta THEN  
    SET casa_punti = 1;  
    SET trasferta_punti = 1;  
  ELSE  
    SET casa_punti = 0;  
    SET trasferta_punti = 3;  
  END IF;  
  
  UPDATE Squadra  
  SET punti = punti + casa_punti  
  WHERE ID = NEW.ID_casa;  
  
  UPDATE Squadra  
  SET punti = punti + trasferta_punti  
  WHERE ID = NEW.ID_trasferta;  
END
```

Query per visualizzare la classifica del campionato

Questa **query** seleziona i **nomi** delle **squadre** e i loro **punti totali** dalla tabella "Squadra" e li **ordina** in modo **decrescente** in base al numero di **punti**.

```
SELECT nome AS NomeSquadra,punti AS PuntiTotali
FROM squadra
ORDER BY punti DESC;
```

Query per visualizzare la rosa che ha vinto il torneo

Questa **query** seleziona i **dettagli** dei **giocatori** dalla tabella "giocatore" che appartengono alla **squadra** con il **punteggio massimo**.

```
SELECT G.ID, G.nome, G.ruolo, G.data_nascita, G.nazionalita
FROM giocatore G
JOIN squadra S ON G.ID_squadra = S.ID
WHERE S.punti = (
    SELECT MAX(punti)
    FROM squadra
);
```

Query per visualizzare l'allenatore della squadra vincente

Questa **query** seleziona i **dettagli** degli **allenatori** dalla tabella "allenatore" che sono associati alla **squadra** con il **punteggio massimo**.

```
SELECT A.ID, A.nominativo, A.data_nascita, A.nazionalita
FROM allenatore A
JOIN squadra S ON A.ID = S.IDAllenatore
WHERE S.punti = (
    SELECT MAX(punti)
    FROM squadra
);
```

Query per visualizzare tutte le partite giocate da squadra

Questa **query** è stata progettata per mostrare **tutte** le **partite** giocate da una **squadra** con un ID specifico.

```
SELECT P.ID, C.nome AS squadra_casa, T.nome AS squadra_trasferta, P.gol_casa,
P.gol_trasferta, P.data, P.ID_stadio, P.ID_arbitro
FROM Partita P
JOIN Squadra C ON P.ID_casa = C.ID
JOIN Squadra T ON P.ID_trasferta = T.ID
WHERE C.ID = <ID_squadra> OR T.ID = <ID_squadra>;
```

<ID_squadra>→id squadra della quale vogliamo visualizzarne le partite

Query per visualizzare le squadre con più gol fatti in casa

Questa **query calcola** il numero totale di **gol** segnati in **casa** da ciascuna **squadra** e ordina i risultati in base ai gol segnati in casa in modo **decrescente**.

```
SELECT S.nome AS NomeSquadra, SUM(P.gol_casa) AS gol_fatti_in_casa
FROM squadra S
JOIN partita P ON S.ID = P.ID_casa
GROUP BY S.nome
ORDER BY gol_fatti_in_casa DESC;
```

Query per visualizzare le squadre con più gol in trasferta

Questa **query calcola** il numero totale di **gol** segnati **fuori casa** da ciascuna **squadra** e ordina i risultati in base ai gol segnati **fuori casa** in modo **decrescente**.

```
SELECT S.nome AS NomeSquadra, SUM(P.gol_trasferta) AS gol_fatti_fuori_casa
FROM squadra S
JOIN partita P ON S.ID = P.ID_trasferta
GROUP BY S.nome
ORDER BY gol_fatti_fuori_casa DESC;
```

Query per visualizzare le squadre con più gol in totale

Questa **query calcola** il numero totale di **gol** segnati da ciascuna **squadra**, sia in **casa** che in **trasferta**, e ordina i risultati in base al numero totale di gol segnati in modo **decrescente**.

```
SELECT squadra.nome, sum(gol) AS gol_totali
FROM (
    SELECT id_casa AS id_squadra, gol_casa AS gol
    FROM partita
    UNION ALL
    SELECT id_trasferta, gol_trasferta
    FROM partita
) AS gol
JOIN squadra ON squadra.id = gol.id_squadra
GROUP BY squadra.id, squadra.nome
ORDER BY gol_totali DESC;
```

Tutte le query elencate sono presenti nel file SQL: **query.sql**

Riepilogo Organizzazione file SQL

- **Creazione** Database e tabelle (**dmi_cup.sql**)
- **Dati** relativi alle tabelle (**dmi_cup.sql**)
- **Trigger** `aggiorna_punti_squadra` (**dmi_cup.sql**)
- Per visualizzare tutte le **query** SQL (**query.sql**)

Conclusioni e ringraziamenti

Con il presente lavoro si è modellata una **base di dati** (sia da un punto di vista **concettuale** che **logico**) per la gestione di un **torneo di calcio a 5**. La **progettazione** è stata realizzata seguendo la metodologia studiata a **lezione** e sono state incluse diverse **query** in base alle specifiche operazionali identificate durante l'analisi dei requisiti.

Desidero esprimere il mio sincero **ringraziamento** al **Professor Alaimo** per avermi concesso l'opportunità di realizzare questo **progetto**. La preziosa esperienza acquisita sarà un valore aggiunto nel mio **percorso futuro**.



Data

03/08/2023

Nominativo

Vincenzo Villanova

Matricola: 1000031067