

CHAMPION HUB

Elaborato per il corso di basi di dati



Saponaro Mattia matr 0001071328

Brighi Federico matr 0001070887

STRUTTURA DELL'ELABORATO

Questo documento tratta nel dettaglio la progettazione e l'implementazione dell'elaborato "Champion Hub" di Brighi Federico e Saponaro Mattia. Il documento è strutturato nel seguente modo:

- 1) Analisi dei requisiti
 - a) Intervista
 - b) Estrazione dei concetti principali
- 2) Progettazione concettuale
 - a) Schema scheletro
 - b) Schema concettuale finale
- 3) Progettazione logica
 - a) Stima del volume dei dati
 - b) Descrizione delle operazioni principali con stima della loro frequenza
 - c) Schemi di navigazione e tabelle degli accessi
 - d) Raffinamento dello schema
 - e) Analisi delle ridondanze
 - f) Traduzione di entità e associazioni in relazioni
 - g) Schema relazionale finale
 - h) Traduzione delle operazioni in query SQL
- 4) Progettazione dell'applicazione
 - a) Interfaccia grafica

ANALISI DEI REQUISITI

L'obiettivo di questo progetto è realizzare un sistema di supporto ad un ente che gestisce un campionato di calcio. La base di dati dovrà immagazzinare informazioni relative: alle squadre, ai loro rispettivi giocatori e alle partite giocate.

Intervista

Ogni squadra può iscriversi al campionato inserendo le proprie informazioni: ogni squadra è posseduta da un presidente, il quale si affida ad un direttore sportivo. Con i fondi, fornitogli dal patrimonio del presidente e dai suoi sponsor, si occupa di scambiare, tramite il mercato sportivo più giocatori, un allenatore e qualche elemento dello staff (richiestogli dall'allenatore stesso).

Ogni squadra potrà affrontare le altre, iscritte allo stesso campionato, in varie partite (squadra casa vs squadra ospite) le quali verranno arbitrate da un arbitro tra quelli del campionato. Queste partite si svolgeranno in uno stadio, il quale richiede un affitto per poter essere usato, e produrranno un tabellino con le statistiche riguardanti la partita stessa.

Il campionato verrà sponsorizzato da varie aziende e potrà essere trasmesso su diversi canali, i quali porteranno fondi per l'organizzazione assieme alla vendita dei biglietti per vedere la partita.

Appena ci si iscrive al campionato, alla squadra viene assegnata una quota N di punti. Ogni sconfitta apporta alla squadra $-M$ punti e ogni vittoria $+M$ punti. Una partita non può concludersi con un pareggio. Superata negativamente una quota di L punti si viene retrocessi e quindi esclusi dal campionato. All'arrivo di Q ($Q > N$) punti si vince il campionato e si ottiene un premio in denaro. Le funzionalità offerte dalla piattaforma agli utenti sono le seguenti:

- 1) Iscrizione al campionato della propria squadra.
- 2) Proporre scambi di giocatori tra i propri giocatori e quelli delle altre squadre:
 - a) Registrazione degli scambi e dei trasferimenti tra le squadre.
- 3) Registrare gli infortuni dei giocatori durante le partite o gli allenamenti, con relativa durata prevista di assenza e stato di recupero.

- 4) Gestione degli sponsor delle squadre, registrando i contratti di sponsorizzazione e le relative condizioni.

Inoltre l'utente potrà consultare informazioni aggregate come:

- 1) Visualizzazione della classifica del campionato in tempo reale.
- 2) Visualizzazione dei giocatori infortunati.
- 3) Visualizzazione del proprio registro di transazioni finanziarie: per acquisti di giocatori, allenatori, staff e altre spese.
- 4) Visualizzazione del calendario (storico) del campionato.

Infine l'amministratore del sito potrà:

- 1) Inserire o togliere giocatori delle squadre in caso di trasferimento.
- 2) Togliere dal campionato le squadre con meno di L punti in caso di retrocessione.
- 3) Bloccare/sbloccare un calciatore espulso in una partita.
- 4) Gestire le spese.
- 5) Visualizzare un resoconto dei ricavi dai vari campionati (il calcolo avverrà automaticamente).
- 6) Organizzazione di partite tra le squadre:
 - a) Calcolo automatico dei punti di ogni squadra per ogni partita giocata.

Estrazione dei concetti principali

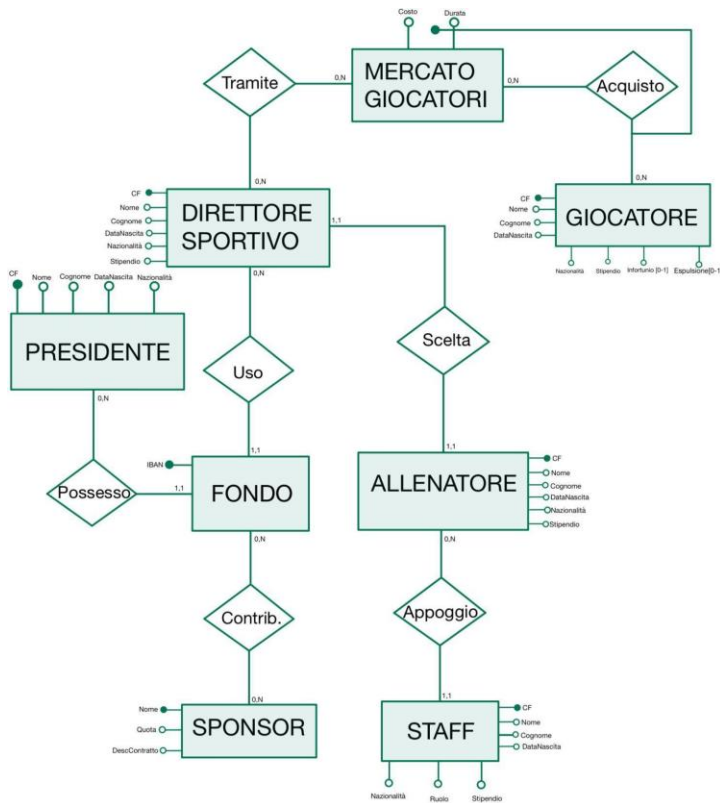
GIOCATORE	Calciatore che gioca in una determinata squadra
SQUADRA	Squadra che partecipa ad un campionato
PARTITA	Incontro tra due squadre del campionato
INFO TORNEO	Campionato in cui giocano le varie squadre
CALENDARIO	Insieme delle date delle varie partite
DIRETTORE SPORTIVO	Gestore della squadra a cui è associato
TABELLINO STATISTICHE	Report delle statistiche di ogni partita

A seguito dell'esposizione dei concetti principali, si riporta un testo che li riassume, eliminando possibili ambiguità:

Per ogni **squadra** iscritta al **torneo** vengono memorizzati: nome, anno di fondazione, città di riferimento, numero di trofei vinti, quota di iscrizione, punteggio, posizione in classifica. Ogni squadra è composta da più **giocatori** di cui si memorizzano: codice fiscale, nome, cognome, data di nascita, nazionalità, stipendio, numero di maglia, ruolo ed eventuali stati di infortunio o espulsione. Questi sono acquistati dal **direttore sportivo**, caratterizzato da: codice fiscale, nome, cognome, data di nascita, nazionalità, stipendio. Seguendo il **calendario**, le squadre si scontrano in una **partita**, dove vengono memorizzati: codice identificativo, prezzo dei biglietti, squadra vincitrice e risultato finale. Dopo ogni partita viene prodotto un **tabellino delle statistiche** di cui si memorizzano: codice identificativo, data della partita, eventuali marcatori ed infortuni, numero di goal, pali e cartellini.

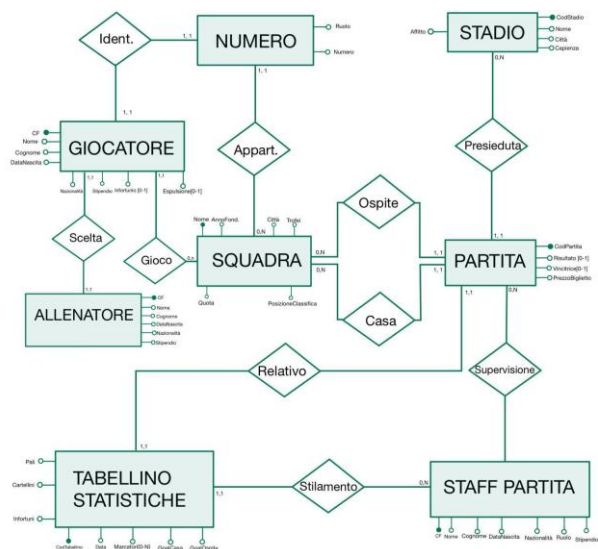
PROGETTAZIONE CONCETTUALE

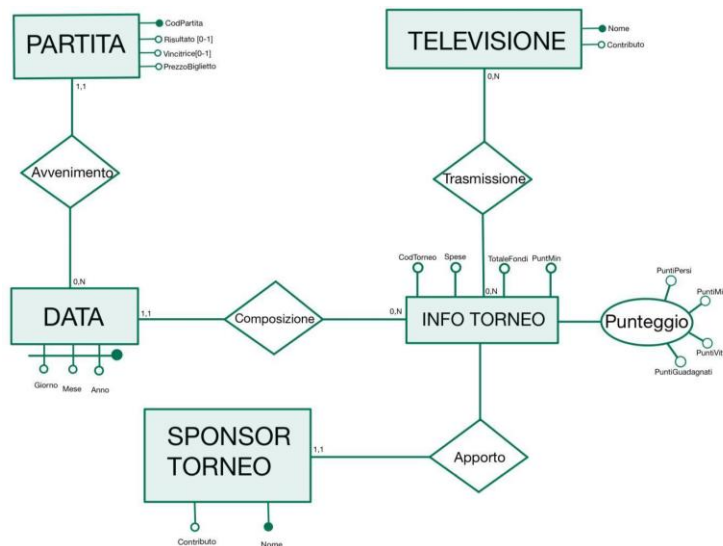
Schema scheletro



Il presidente possiede un patrimonio che farà parte dei fondi usati dal direttore sportivo per creare la squadra. In particolare ha il compito di scegliere: giocatori e l'allenatore. I loro stipendi verranno detratti dai fondi della squadra. L'allenatore si appoggerà a dello staff per gestire i giocatori. Il presidente è riconosciuto da un codice fiscale, i fondi sono rappresentati da un'entità per tenere traccia del totale di questi. Il mercato ha come chiave la sua durata e il giocatore acquistato, in modo tale che uno stesso giocatore possa essere acquistato in anni diversi ma mai nello stesso anno. Il giocatore è

univocamente riconosciuto con il codice fiscale. Ogni giocatore gioca in una squadra, con un numero univoco all'interno di essa. La squadra, identificata dal nome, può giocare come ospite o come casa una partita, il cui risultato verrà aggiornato in seguito ad essa. La partita è supervisionata da uno staff (stipendiato), che produce un tabellino delle statistiche. La partita si tiene in uno stadio di cui bisogna pagare un affitto, e nell'associazione si tiene conto di quanti biglietti sono stati venduti per quella partita. Il loro numero dovrà essere minore o uguale della capienza massima dello stadio.

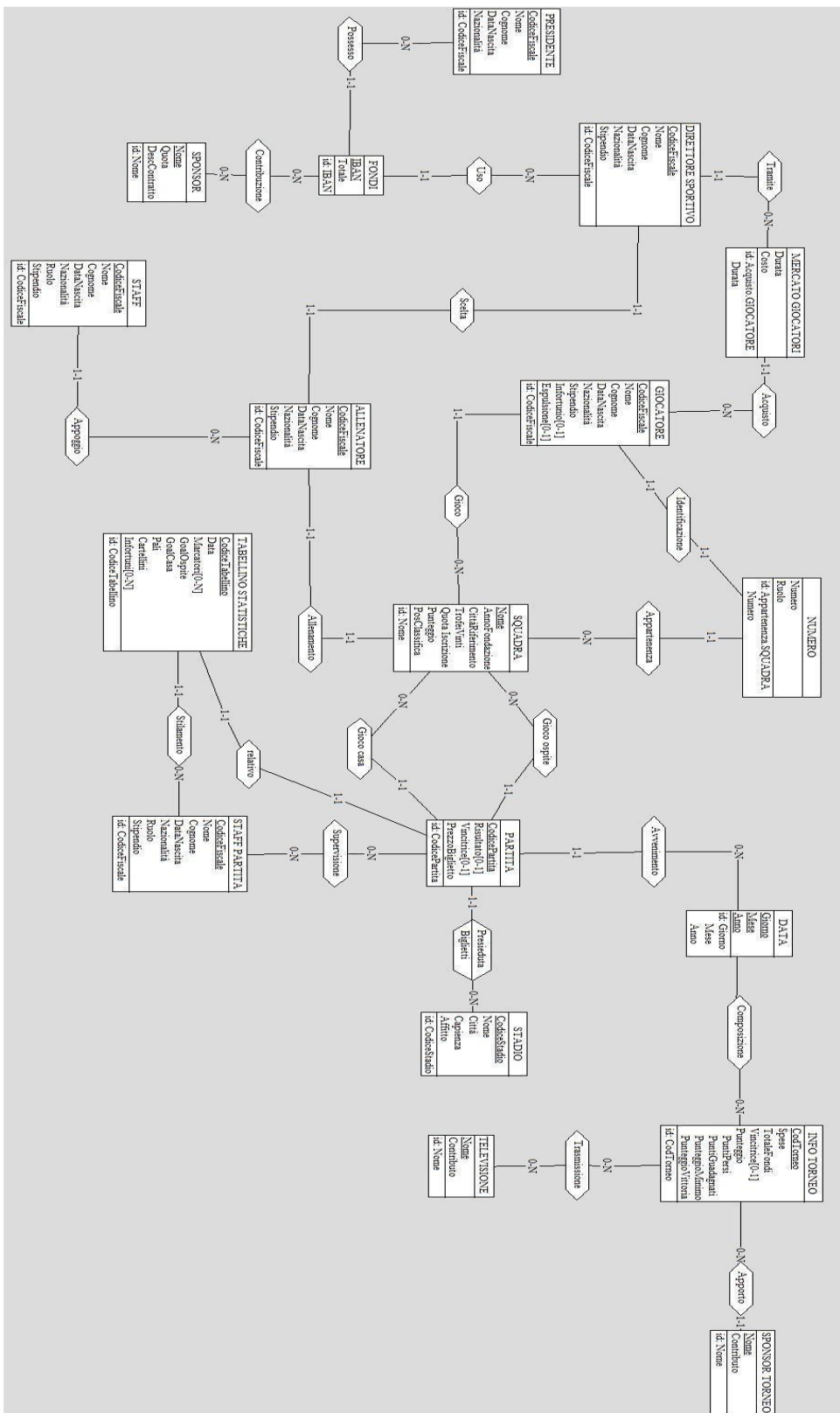




Ogni partita viene giocata in una determinata data del calendario. La data è univocamente identificata dalla tripla (giorno, mese, anno). Il torneo è identificato da un codice e si vuole tenere traccia delle spese organizzative, che verranno coperte dai fondi portati dai vari sponsor e dalle televisioni che vorranno trasmettere le partite. Tra gli

attributi del torneo viene anche precisato il sistema di punti e alla fine verrà aggiunta la squadra vincitrice.

Schema concettuale finale



PROGETTAZIONE LOGICA

Stima del volume dei dati

CONCETTO	COSTRUTTO	VOLUME
Presidente	E	10
Direttore Sportivo	E	10
Fondi	E	10
Giocatore	E	200
Allenatore	E	10
Staff	E	50
Squadra	E	10
Numero	E	90
Mercato	E	200
Partita	E	50
Tabellino	E	50
Stadio	E	10
Torneo	E	1
Televisione	E	5
Data	E	50
Calendario	E	1
Staff Partita	E	100
Sponsor	E	50
Sponsor Torneo	E	80
Possesso	R	10
Uso	R	10
Contributo	R	50
Tramite	R	200
Acquisto	R	200
Gioco	R	100
Scelta	R	10
Appoggio	R	70
Allenamento	R	10
Gioco Ospite	R	25
Gioco Casa	R	25
Supervisione	R	50
Stilamento	R	50
Presieduta	R	50
Avvenimento	R	50
Composizione	R	50
Formazione	R	1
Trasmissione	R	5
Apporto	R	80
Identifica	R	100
Appartenenza	R	90
Relativo	R	50

Descrizione delle operazioni principali e stima della loro frequenza

CODICE	OPERAZIONE	FREQUENZA(op/gg)
1	Iscrivere una nuova squadra	2
2	Proporre scambi di giocatori	10
3	Inserire/togliere giocatori dalle squadre	10
4	Togliere dal torneo le squadre retrocesse	1
5	Bloccare/Sbloccare un giocatore espulso	2
6	Organizzazione partite tra le squadre	1
7	Calcolo dei punti di ogni squadra	2
8	Registrazione degli infortuni	2
9	Gestione sponsor delle squadre	3
10	Gestione Sponsor del Torneo	1
11	Visualizzazione della classifica	1
12	Visualizzazione giocatori infortunati	1
13	Visualizzazione del calendario del campionato	5
14	Visualizzare un resoconto dei ricavi del torneo	1
15	Visualizzazione ricavi del Presidente	1
16	Visualizzazione dei giocatori espulsi	2

Schemi di navigazione e tabelle degli accessi

Riportiamo in seguito le tabelle degli accessi delle operazioni sopra riportate; inoltre, dove non risulti banale, inseriamo i relativi schemi di navigazione. Al fine del calcolo dei costi, consideriamo di peso doppio gli aspetti in scrittura rispetto a quelli in lettura.

OP1 - Iscrivere una nuova squadra

CONCETTO	CoSTRUTTO	ACCESSI	TIPO
SQUADRA	E Costo: 4 acc/gg	1	S

OP2 - Proporre scambi di giocatori

CONCETTO	CoSTRUTTO	ACCESSI	TIPO
DIRETTORE	E	1	L
MERCATO GIOCATORI	E	1	S
GIOCATORE	E	1	L
ACQUISTO	R Costo: 60 acc/gg	1	S

OP3 - Inserire/togliere giocatori dalle squadre

CONCETTO	COSTRUTTO	ACCESSI	TIPO
GIOCATORE	E	1	L
GIOCO	R	1	S
SQUADRA	E	1	L
Costo: 40 acc/gg			

OP4 - Togliere dal torneo squadre retrocesse

CONCETTO	COSTRUTTO	ACCESSI	TIPO
SQUADRA	E	1	L
INFO TORNEO	E	1	L
SQUADRA	E	1	S
Costo: 4 acc/gg			

OP5 - Bloccare/Sbloccare un giocatore espulso

CONCETTO	COSTRUTTO	ACCESSI	TIPO
GIOCATORE	E	1	S
Costo: 4 acc/gg			

OP6 - Organizzazione partita tra le squadre

CONCETTO	COSTRUTTO	ACCESSI	TIPO
SQUADRA	E	2	L
PARTITA	E	1	S
GIOCO OSPITE	R	1	S
GIOCO CASA	R	1	S
AVVENIMENTO	R	1	S
DATA	E	1	S
Costo: 12 acc/gg			

OP7 - Calcolo dei punti di ogni squadra

CONCETTO	COSTRUTTO	ACCESSI	TIPO
SQUADRA	E	1	L
INFO TORNEO	E	1	L
SQUADRA	E	1	S
Costo: 8 acc/gg			

OP8 - Registrazione degli infortuni

CONCETTO	COSTRUTTO	ACCESSI	TIPO
TABELLINO	E	1	L
STATISTICHE			
GIOCATORE	E	1	S
Costo: 6 acc/gg			

OP9 - Gestione Sponsor delle squadre

CONCETTO	COSTRUTTO	ACCESSI	TIPO
SPONSOR	E	1	S
FONDI	E	1	L
CONTRIBUZIONE	R	1	S
FONDI	E	1	S
Costo: 21 acc/gg			

OP10 - Gestione Sponsor del Torneo

CONCETTO	COSTRUTTO	ACCESSI	TIPO
SPONSOR	E	1	S
TORNEO			
APPORTO	A	1	S
INFO TORNEO	E	1	L
INFO TORNEO	E	1	S
Costo: 7 acc/gg			

OP11 - Visualizzazione classifica

CONCETTO	COSTRUTTO	ACCESSI	TIPO
SQUADRA	E	10	L
Costo: 10 acc/gg			

OP12 - Visualizzazione giocatori infortunati

CONCETTO	COSTRUTTO	ACCESSI	TIPO
GIOCATORI	E	200	L
Costo: 200 acc/gg			

OP13 - Visualizzazione del calendario del campionato

CONCETTO	COSTRUTTO	ACCESSI	TIPO
CALENDARIO	E	1	L
COMPOSIZIONE	R	50	L
DATA	E	50	L
AVVENIMENTO	R	50	L
PARTITA	E	50	L
Costo: 1005 acc/gg			

OP14 - Visualizzare un resoconto dei ricavi del torneo

CONCETTO	COSTRUTTO	ACCESSI	TIPO
INFO TORNEO	E	1	L
SPONSOR	E	80	L
TORNEO			
TELEVISIONE	E	5	L
Costo: 86 acc/gg			

OP15 - Visualizzare i ricavi del presidente

CONCETTO	COSTRUTTO	ACCESSI	TIPO
FONDI	E Costo: 1 acc/gg	1	L

OP16 - Visualizzazione giocatori espulsi

CONCETTO	COSTRUTTO	ACCESSI	TIPO
GIOCATORE	E Costo: 800 acc/gg	200	L

Analisi delle ridondanze

Nel nostro schema sono presenti due attributi ridondanti, riguardanti rispettivamente il totale dei fondi del torneo e del presidente. Segue una trattazione per capire se sia conveniente o meno, in funzione del numero di accessi al giorno, mantenere queste ridondanze.

FONDI - Totale

Le operazioni che riguardano questa ridondanza sono OP9 e OP16

OP9 con ridondanza

Come già analizzato in precedenza, ha un costo pari a 21 accessi al giorno, in quanto in seguito alla rimozione o all'aggiunta di uno sponsor è necessario aggiornare il Totale dei fondi.

OP9 senza ridondanza

CONCETTO	COSTRUTTO	ACCESSI	TIPO
SPONSOR	E	1	S
CONTRIBUZIONE	R	1	S
FONDI	E	1	L
Costo: 15 acc/gg			

OP15 con ridondanza

Come già analizzato in precedenza, ha un costo pari a 1 accesso al giorno in quanto è sufficiente visualizzare il Totale dei fondi in seguito alla detrazione dei vari stipendi dei giocatori

CONCETTO	COSTRUTTO	ACCESSI	TIPO
FONDI	E	1	L
SPONSOR	E	5	L
DIRETTORE SPORTIVO	E	1	L
MERCATO	E	20	L
ALLENATORE	E	1	L
STAFF	E	5	L
GIOCATORE	E	20	L
SQUADRA	E	1	L
Costo: 54 acc/gg			

OP15 senza ridondanza

CONCLUSIONE

Per l'attributo FONDI - Totale è conveniente mantenere la ridondanza

INFO TORNEO - TotaleFondi

Le operazioni che riguardano questa ridondanza sono OP11 e OP16

OP10 con ridondanza

Come già analizzato in precedenza, ha un costo pari a 7 accessi al giorno, in quanto in seguito alla rimozione o all'aggiunta di uno sponsor è necessario aggiornare il Totale dei fondi.

OP10 senza ridondanza

CONCETTO	COSTRUTTO	ACCESSI	TIPO
SPONSOR TORNEO	E	1	S
APPORTO	A	1	S
INFO TORNEO	E	1	L
Costo 5 acc/gg			

OP14 con ridondanza

Come già analizzato in precedenza, ha un costo pari a 1 accesso al giorno in quanto è sufficiente visualizzare il Totale dei fondi in seguito alla detrazione dei vari costi.

OP14 senza ridondanza

CONCETTO	COSTRUTTO	ACCESSI	TIPO
SPONSOR TORNEO	E	80	L
TELEVISIONE	E	5	
INFO TORNEO	E	1	L
STADI	E	10	L
PARTITA	E	50	L
STAFF PARTITA	E	100	L
SQUADRE	E	10	L
Costo: 256 acc/gg			

CONCLUSIONE

Per l'attributo INFO TORNEO - TotaleFondi è conveniente mantenere la ridondanza.

Raffinamento dello schema

Eliminazione delle gerarchie

Nel nostro schema non sono presenti gerarchie, pertanto non è stato necessario intervenire.

Eliminazione degli attributi composti

Nell'entità "INFO TORNEO" che è stato diviso nei suoi sotto-attributi. Sarà necessario assicurarsi che, a livello applicazione, questi saranno impostati ad un valore coerente.

Eliminazione degli attributi multivalore

Nell'entità "TABELLINO STATISTICHE" sono presenti due attributi multivalore, ovvero i marcatori e i giocatori infortunati. Questi sono stati trasformati in due entità in fase di progettazione logica.

Scelta delle chiavi primarie

Nel nostro schema sono già evidenziate senza ambiguità tutte le chiavi primarie per tutte le entità presenti.

Eliminazione degli identificatori esterni

Sono state eliminate le seguenti relazioni:

- Possesso, importando CodiceFiscale in FONDI
- Contribuzione, creando l'entità CONTRIBUZIONI e importando IBAN e Nome
- Uso, importando CodiceFiscale in FONDI
- Tramite, creando l'entità TRAMITE e importando le chiavi di DIRETTORE SPORTIVO e MERCATO GIOCATORI
- Acquisto, importando CodiceFiscale in MERCATO GIOCATORI
- Appartenenza, importando Nome in NUMERO
- Gioco, importando Nome in GIOCATORE
- Gioco ospite/casa, importando Nome in PARTITA
- Supervisione, creando l'entità SUPERVISIONI e importando CodiceFiscale e CodicePartita
- Stilamento, importando CodiceFiscale in TABELLINO STATISTICHE
- Presieduta, importando CodiceStadio in PARTITA e aggiungendo l'attributo monovalore Biglietti
- Avvenimento, importando Giorno, Mese, Anno in PARTITA
- Composizione, importando CodTorneo in DATA
- Apporto, importando CodTorneo in SPONSOR TORNEO
- Trasmissione creando l'entità TRASMISSIONI e importando CodTorneo e Nome

Traduzione delle entità e associazioni in relazioni

PRESIDENTI (CodiceFiscale, Nome, Cognome, DataNascita, Nazionalità)

DIRETTORI_SPORTIVI (CodiceFiscale, Nome, Cognome, DataNascita, Nazionalità, Stipendio)

FONDI (IBAN, Totale, CodPresidente:PRESIDENTI, CodDir:DIRETTORI_SPORTIVI)

SPONSOR (Nome, Quota, DescContratto)

CONTRIBUZIONI (IBAN:FONDI, Nome:SPONSOR)

TRAMITE (CodDir:DIRETTORI_SPORTIVI, (CodGiocatore, Durata):MERCATO_GIOCATORI)

MERCATO_GIOCATORI (Durata, CodGiocatore:GIOCATORI, Costo)

GIOCATORI (CodiceFiscale, Nome, Cognome, DataNascita, Nazionalità, Stipendio, Infortunio*, Espulsione*, NomeSquadra:SQUADRE)

ALLENATORI (CodiceFiscale, Nome, Cognome, DataNascita, Nazionalità, Stipendio, CodDir:DIRETTORI_SPORTIVI, NomeSquadra:SQUADRE)

STAFF (CodiceFiscale, Nome, Cognome, DataNascita, Nazionalità, Ruolo, Stipendio, CodiceAll:ALLENATORI)

SQUADRE (NomeSquadra, AnnoFondazione, CittàRiferimento, TrofeiVinti, Quotaliscrizione, Punteggio, PosClassifica)

NUMERI (NomeSquadra:SQUADRE, Numero, CodGio:GIOCATORI, Ruolo)

PARTITE (CodPartita, Risultato*, Vincitrice*, Biglietti, PrezzoBiglietto, CodiceTab:TABELLINI, NomeOspite:SQUADRE, NomeCasa:SQUADRE, CodStadio:STADI, (Giorno, Mese, Anno):DATE)

STADI (CodStadio, Nome, Città, Capienza, Affitto)

SUPERVISIONE (CodStaff:STAFF_PARTITE, CodPartita:PARTITE)

STAFF_PARTITE (CodFiscale, Nome, Cognome, DataNascita, Stipendio, Ruolo)

TABELLINI_STATISTICHE (CodTabellino, Data, GoalOspite, GoalCasa, Pali, Cartellini, CodStaff:STAFF_PARTITE)

MARCATORI (CodTabellino:TABELLINI_STATISTICHE, Marcatori)

INFORTUNI (CodTabellino:TABELLINI_STATISTICHE, Infortuni)

DATE(Giorno, Mese, Anno, CodTorneo:TORNEI)

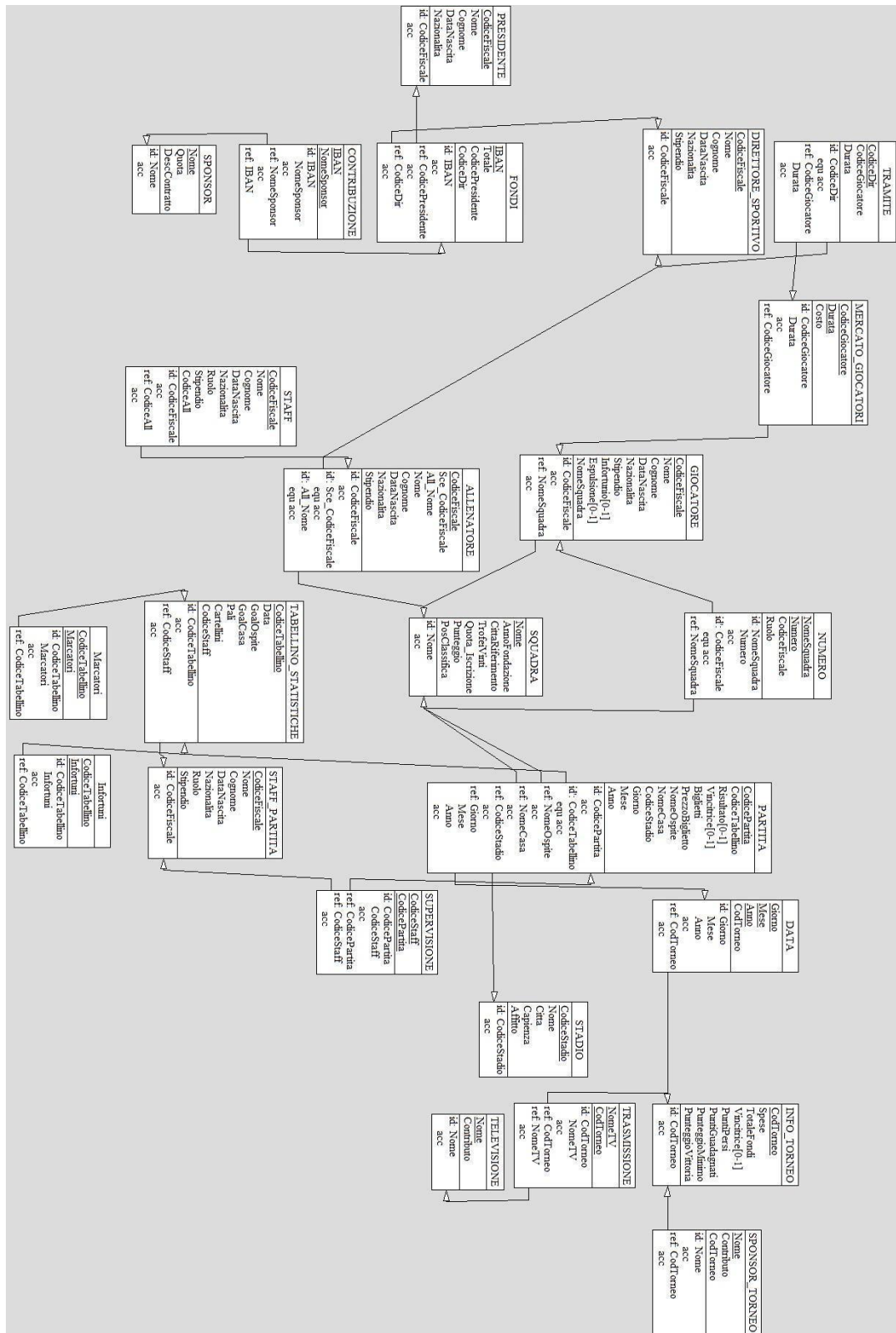
INFO_TORNEI (CodTorneo, TotaleFondi, Vincitrice*, PuntiPersi, PuntiGuadagnati, PunteggioMinimo, PunteggioVittoria)

SPONSOR_TORNEI (Nome, Contributo, CodTorneo:TORNEI)

TELEVISIONI (Nome, Contributo)

TRASMISSIONI (CodTorneo:TORNEI, NomeTV:TELEVISIONI)

Schema relazionale finale



Traduzione delle operazioni in query SQL

OP1 - Inserire una nuova squadra

```
INSERT INTO squadra (Nome, AnnoFondazione, CittaRiferimento, Quota_Iscrizione,  
TrofeiVinti, Punteggio, PosClassifica)  
VALUES (?, ?, ?, ?, ?);
```

OP2 - Scambiare giocatori tra due squadre

```
SELECT COUNT(*) FROM giocatore  
WHERE CodiceFiscale = ?  
AND NomeSquadra = ?;  
UPDATE giocatore SET NomeSquadra = ?  
WHERE CodiceFiscale = ?;  
UPDATE giocatore SET NomeSquadra = ?  
WHERE CodiceFiscale = ?;  
UPDATE numero SET NomeSquadra = ?  
WHERE CodiceFiscale = ?;  
UPDATE numero SET NomeSquadra = ?  
WHERE CodiceFiscale = ?;
```

OP3 - Inserire/togliere giocatori dalle squadre

```
INSERT INTO giocatore (CodiceFiscale, Nome, Cognome, DataNascita, Nazionalita,  
Stipendio, Infortunio, Espulsione, NomeSquadra)  
VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?);  
INSERT INTO numero (CodiceFiscale, NomeSquadra, Numero, Ruolo)  
VALUES (?, ?, ?, ?);
```

```
DELETE FROM giocatore  
WHERE CodiceFiscale = ?;  
DELETE FROM numero  
WHERE CodiceFiscale = ?;
```

OP4 - Togliere dal torneo squadre retrocesse

```
SELECT Nome FROM squadra WHERE Punteggio < 4
DELETE FROM tramite WHERE CodiceGiocatore = ?
DELETE FROM mercato_giocatori WHERE CodiceGiocatore = ?
DELETE FROM numero WHERE CodiceFiscale = ?
DELETE FROM giocatore WHERE CodiceFiscale = ?
SELECT CodiceFiscale FROM allenatore WHERE All_Nome = ?
DELETE FROM staff WHERE CodiceAll = ?
DELETE FROM allenatore WHERE CodiceFiscale = ?
SELECT CodicePartita FROM partita WHERE NomeCasa = ? OR NomeOspite = ?
DELETE FROM supervisione WHERE CodicePartita = ?
DELETE FROM partita WHERE NomeCasa = ?
DELETE FROM partita WHERE NomeOspite = ?
DELETE FROM squadra WHERE Nome = ?
```

OP5 - Bloccare/Sbloccare un giocatore espulso

```
UPDATE giocatore SET Espulsione = ? WHERE CodiceFiscale = ?
```

OP6 - Organizzazione delle partite

```
INSERT INTO tabellino_statistiche (CodiceTabellino, DataTab, GoalCasa, GoalOspite,
Pali, Cartellini, CodiceStaff)
VALUES (?, ?, ?, ?, ?, ?, ?);
INSERT INTO data (Giorno, Mese, Anno, CodTorneo)
VALUES (?, ?, ?, ?)
ON DUPLICATE KEY UPDATE Giorno=VALUES(Giorno), Mese=VALUES(Mese),
Anno=VALUES(Anno), CodTorneo=VALUES(codTorneo);
INSERT INTO partita (CodicePartita, CodiceTabellino, Giorno, Mese, Anno, NomeCasa,
NomeOspite, Risultato, CodiceStadio, Biglietti, PrezzoBiglietto)
VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?);
```

OP7 - Aggiornare i punti delle squadre

```
UPDATE squadra
SET Punteggio = Punteggio + ?
WHERE Nome = ?
```

OP8 - Registrazione degli infortuni

```
INSERT INTO infortuni (id_giocatore, descrizione, durata_giorni)  
VALUES (?, ?, ?);
```

```
UPDATE giocatore SET stato = 'Infortunato'  
WHERE codice_fiscale = ?;
```

OP9 Gestione degli sponsor delle squadre

OP10 - Gestione sponsor del torneo

```
INSERT INTO sponsor_torneo (nome, contributo, CodTorneo)  
VALUES (?, ?, ?)
```

OP11 - Visualizzazione della classifica

```
SELECT Nome, Punteggio, PosClassifica  
FROM squadr  
ORDER BY Punteggio DESC, PosClassifica ASC
```

OP12 - Visualizzazione dei giocatori infortunati

```
SELECT Nome, NomeSquadra, Infortunio FROM giocatore  
WHERE Infortunio = 'Infortunato'
```

OP13 - Visualizzazione del calendario del campionato

```
SELECT p.*, sc.nome AS squadra_casa_nome, so.nome AS squadra_ospite_nome  
FROM partita p  
JOIN squadra sc ON p.squadra_casa = sc.id  
JOIN squadra so ON p.squadra_ospite = so.id  
ORDER BY p.data_partita;
```

OP14 - Visualizzazione del resoconto dei ricavi del torneo

```
SELECT TotaleFondi, Spese  
FROM info_torneo  
WHERE CodTorneo = ?
```

OP15 - Visualizzazione dei ricavi del presidente

```
SELECT SUM(f.Totale + s.Quota - COALESCE(g.Stipendio, 0) -  
COALESCE(a.Stipendio, 0) - COALESCE(ds.Stipendio, 0)) AS FondiTotali  
FROM presidente p  
JOIN fondi f ON p.CodiceFiscale = f.CodicePresidente  
JOIN direttore_sportivo ds ON f.CodiceDir = ds.CodiceFiscale  
JOIN allenatore a ON ds.CodiceFiscale = a.Sce_CodiceFiscale  
JOIN squadra sq ON a.All_Nome = sq.Nome  
LEFT JOIN giocatore g ON sq.Nome = g.NomeSquadra  
JOIN contribuzione c ON f.IBAN = c.IBAN  
JOIN sponsor s ON c.NomeSponsor = s.Nome  
WHERE p.Nome = %s AND p.Cognome = %s
```

OP16 - Visualizzazione giocatori espulsi

```
SELECT Nome,Cognome, NomeSquadra, Espulsione  
FROM giocatore  
WHERE Espulsione = 'Espulso'
```

PROGETTAZIONE DELL'APPLICAZIONE

Interfaccia grafica



L'applicazione per interfacciarsi al database è stata realizzata in Python, sfruttando la libreria MySQL Connector per le connessioni al database. Il database risiede in locale e il DBMS utilizzato è MySQL. L'applicazione è una semplice applicazione Tkinter che fa uso di vari file python associati ad un proprio modulo, il quale ha il compito di portare a termine le

query appoggiandosi alla classe DatabaseConnection.

Il progetto si trova seguendo il percorso di directory PYTHON/app. Qui ci si ritrova a quella che è la struttura portante del progetto, la quale segue il pattern MVC:

- **File GUI**

(app/gui): Ogni file Python in questa cartella definisce una finestra

dell'interfaccia grafica utilizzando Tkinter. Ognuna ha un proprio modulo che gestisce le operazioni specifiche.

- **File di Model (app/models):** Ogni file Python in questa cartella definisce le classi che corrispondono alle tabelle del database relative alle entità principali.
- **File di Servizio (app/services):** Ogni file Python in questa cartella definisce le operazioni e le query verso il database. Questi file sono utilizzati dai moduli della GUI per interagire con il database.

Passiamo poi ai file fondamentali che permettono la corretta connessione al database e la possibilità di visualizzare a video la nostra applicazione:



- **main.py:** La finestra principale dell'applicazione Tkinter che contiene i pulsanti per accedere alle varie funzionalità dell'applicazione.
- **db.py (DatabaseConnection):** Gestisce la connessione al database MySQL utilizzando la libreria MySQL Connector. Contiene metodi per eseguire query e recuperare risultati.
- **config.py (Authentication):** Contiene i dati di autenticazione al Database MySQL tra cui HOST ,USER, PASSWORD ,DBNAME.

Per accedere all'applicazione sarà necessario aver installato alcune librerie python che permettono la realizzazione della GUI e ci si dovrà poi spostare su PYTHON/app.

Una volta qui, si dovrà lanciare il comando "python main.py" e l'applicazione verrà lanciata con successo. Si aprirà quindi l'interfaccia grafica con il nostro logo annesso che dà il benvenuto all'utente e mostra le funzionalità del nostro progetto. L'applicazione fornisce nel complesso le

funzionalità richieste evidenziate nella fase di progettazione.