

PROGETTO – BASE DI DATI

Ingegneria e Scienze Informatiche, Cesena

Base di dati per la gestione di un'associazione di basket

Luca Bighini 0000977083 luca.bighini@studio.unibo.it

ANALISI DEI REQUISITI

La base di dati è volta a organizzare e automatizzare la gestione di una società sportiva di pallacanestro. Si occuperà quindi, di raccogliere dati e informazioni sugli allenatori e giocatori, i vari corsi, i campi, gli allenamenti, le palestre e le gare a cui essi parteciperanno. Dovrà inoltre occuparsi della vendita del materiale sportivo e dei corsi offerti dalla società.

INTERVISTA

L'International Basket Imola è una società di pallacanestro con sede sul territorio imolese. La società è formata da giocatori e allenatori, essi sono identificati dal proprio codice fiscale, nome, cognome, età, telefono, e-mail e identificativo. Una volta che il giocatore pagherà una quota di 250 euro gli verrà fornita la divisa d'allenamento e potrà scegliere il suo numero di maglia.

Di ogni giocatore vengono inoltre memorizzate le statistiche relative ad ogni partita giocata (percentuali di: tiri liberi, canestri da 2 punti, canestri da 3 punti, falli commessi, rimbalzi, palle perse e recuperate).

Gli allenatori allenano una squadra formata da un massimo di 15 giocatori. Ogni squadra è caratterizzata da un nome e da una fascia d'età.

Ogni squadra della società può iscriversi ad uno o più campionati di livello differente, giocando così le partite.

Le squadre si allenano settimanalmente in 5 palestre messe a disposizione dal comune: Cavina, Ravaglia, Volta, Pedagna, Paolini. Ogni palestra ha varie attrezzature per gli allenamenti (cerchi, palloni, coni, ecc.).

Nelle palestre durante la stagione estiva, nelle settimane di luglio, gli allenatori organizzano dei campi estivi per i giocatori e sono compresi nel prezzo della quota.

Ogni sede vende del materiale aggiuntivo come borse, cappellini, zaini, maglie, sopra maglie, calzini, ecc. La società vuole mantenere traccia degli acquisti effettuati memorizzando la data e l'ora, così come per le quote.

Ad oggi la società ha una sola sede nella città di Imola ma è in progetto un'espansione sul territorio coinvolgendo le città di Castel Guelfo e Massa Lombarda, che andranno a costituire nuove sedi.

RILEVAMENTO DELLE AMBIGUITÀ E CORREZIONI PROPOSTE

Dato che l'interesse della nostra base di dati riguarda solo le squadre della società di Basket di Imola, la memorizzazione dei campionati non viene tenuto conto delle squadre avversarie delle altre società. Ecco perché la relazione che lega il campionato alla squadra è 1-1, ovvero esiste una sola squadra della società che partecipa ad un

campionato. Le informazioni relative al campionato vengono quindi integrate negli attributi della squadra. Il campionato diventa quindi un attributo composto di partita.

Inoltre riteniamo utile tenere conto di tutti i movimenti che la società opera (per il momento solo l'incasso delle quote e gli acquisti di materiale) e rendere estendibile questa entità nel caso ci sia necessità di registrare nuovi movimenti (eventuali sponsor, donazioni, spese, tasse ecc.). In quest'ottica si è deciso di creare l'entità movimenti che tiene traccia delle entrate e delle uscite ed è legata ad una sede.

DEFINIZIONE DELLE SPECIFICHE IN LINGUAGGIO NATURALE ED ESTRAZIONE DEI CONCETTI PRINCIPALI

TERMINE	DESCRIZIONE
GIOCATORE	Colui che gioca in una squadra
ALLENATORE	Colui che allena una squadra
STATISTICA GIOCATORE	Informazioni relative ad un giocatore
SQUADRA	Insieme di giocatori
CAMPIONATO	Insieme di partite di uno stesso livello
PARTITA	Competizione tra due squadre
PALESTRA	Luogo dove si allena la squadra
ATTREZZATURA	Oggetti sportivi presenti in palestra
CAMPI ESTIVI	Evento estivo settimanale
QUOTA	Pagamento per iscriversi ad una squadra e/o campo estivo
MATERIALE	Oggetti venduti dalla società
VENDITA	Informazione sulla vendita di materiale (o ACQUISTO)
SEDE	Città nella quale opera un ramo della società.

Giocatori e allenatori sono persone e sono identificate dal proprio codice fiscale, nome, cognome, età, telefono, e-mail e identificativo. Un giocatore inoltre pagherà una **quota** e avrà una divisa d'allenamento con un numero di maglia.

Le **statistiche** sono relative ai giocatori e ad ogni **partita** giocata. Sono in percentuali e riguardano i tiri liberi, canestri da 2 punti, canestri da 3 punti, falli commessi, rimbalzi, palle perse e recuperate.

Ogni **squadra** è formata da giocatori ed è caratterizzata da un nome e da una fascia d'età. Ogni squadra della società può iscriversi ad uno o più **campionati** di livello differente, giocando così le partite. Le squadre si allenano in varie **palestre** con differenti **attrezzature**.

Nelle palestre gli allenatori organizzano dei **campi estivi**.

In fine la società vende ai giocatori del **materiale** aggiuntivo. La società vuole mantenere traccia degli **acquisti**.

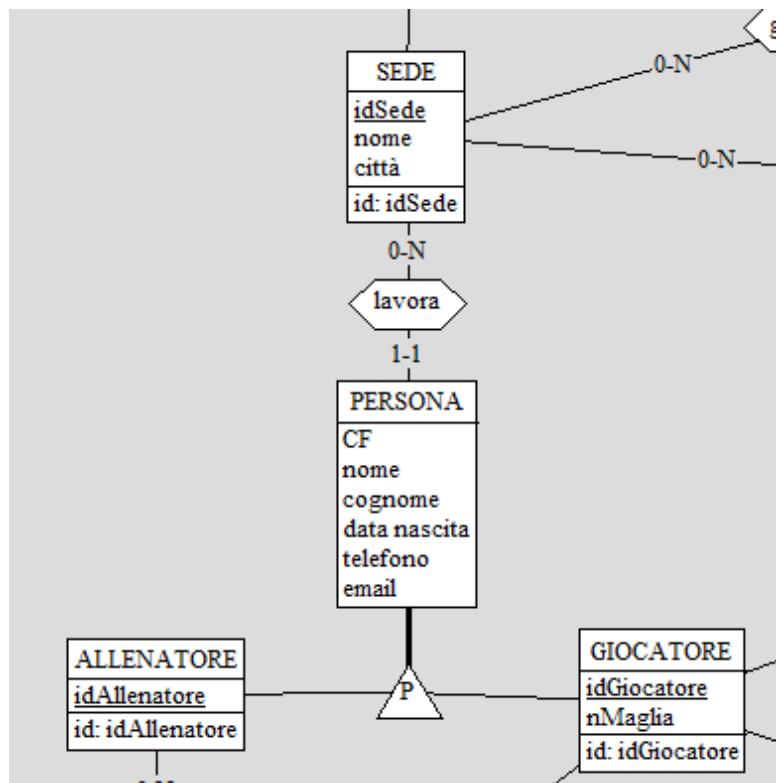
Ad oggi la società ha una sola **sede**. È in progetto un'espansione sul territorio che andrà a costituire nuove **sedi**.

POSSIBILI OPERAZIONI RICHIESTE:

1. Inserire allenatore
2. Inserire giocatore assegnandogli una squadra
3. Visualizzare il numero totale di persone della società
4. Creare una squadra
5. Modificare annata di una squadra
6. Rimuovere un giocatore ad una squadra
7. Assegnare un allenatore ad una squadra
8. Visualizzare i giocatori di una squadra
9. Assegnare un giocatore ad un campo estivo
10. Assegnare un allenatore ad un campo estivo
11. Visualizzare i giocatori di un campo estivo
12. Visualizzare le attrezzature di una palestra
13. Visualizzare le partite di una squadra
14. Visualizzare il campionato di ogni squadra
15. Visualizzare gli allenamenti di una squadra
16. Acquistare x paia di calzini
17. Visualizzare i materiali disponibili
18. Visualizzare il guadagno totale
19. Visualizzare numero di giocatori in una squadra
20. Visualizzare un estratto conto dei movimenti

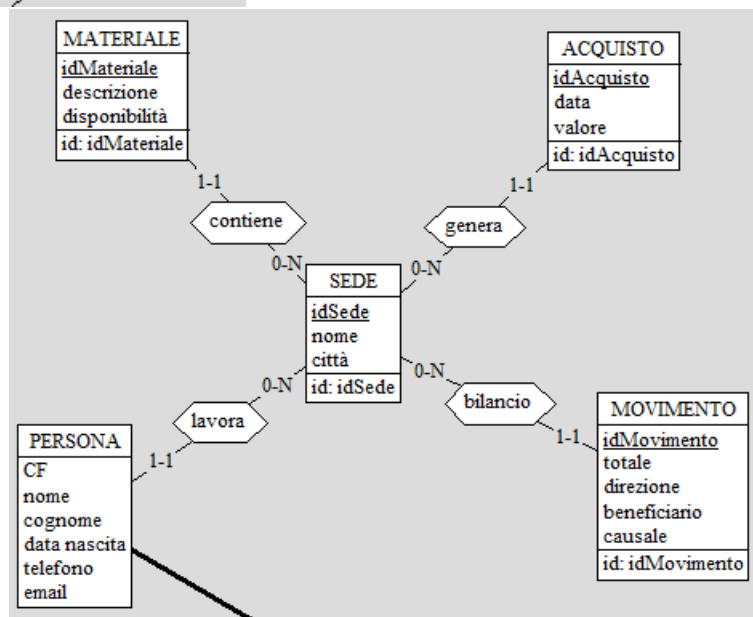
PROGETTAZIONE CONCETTUALE

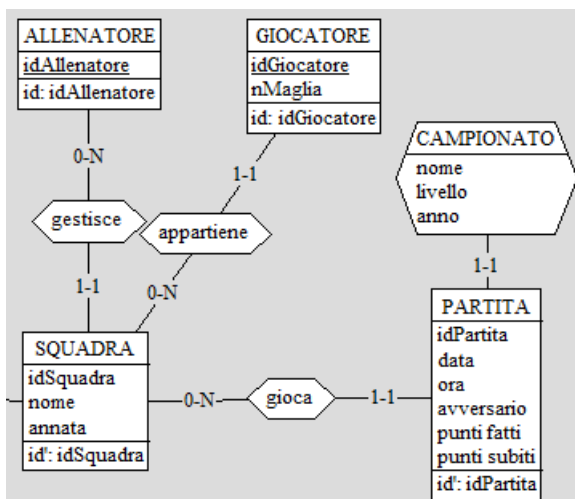
SCHEMA SCHELETRO



Per la rappresentazione delle entità allenatore e giocatore abbiamo scelto di usare la gerarchia PERSONA per generalizzare gli attributi in comune. Le due entità vengono poi distinte tramite un identificativo, rispettivamente idAllenatore e idGiocatore. Il giocatore ha inoltre il numero di maglia. Tutte le persone fanno parte di una sola sede. Nello schema logico si è optato per un collasso verso il basso essendo la copertura della specializzazione totale ed esclusiva.

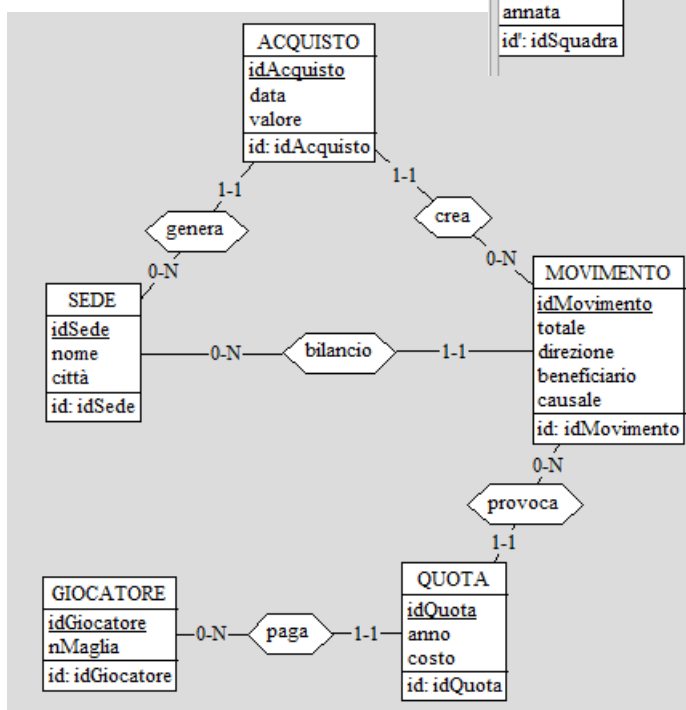
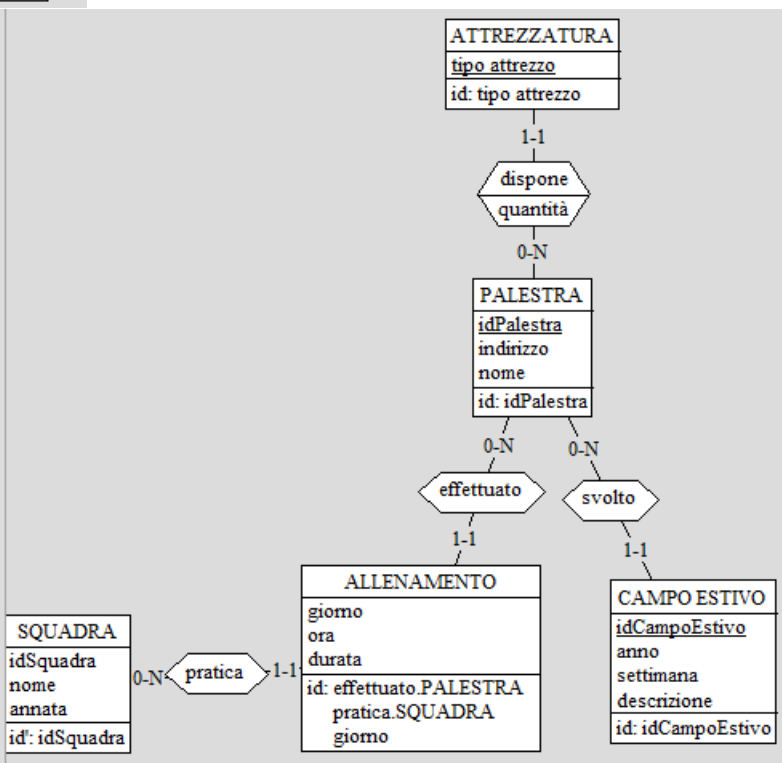
La società è organizzata in base alla sede. Ogni sede si trova in una città e mette in vendita, tramite un acquisto, il materiale in base alla disponibilità che essa contiene. Inoltre tiene conto di tutti i movimenti e infine è costituita da persone, che come abbiamo visto prima si dividono in giocatori e allenatori.





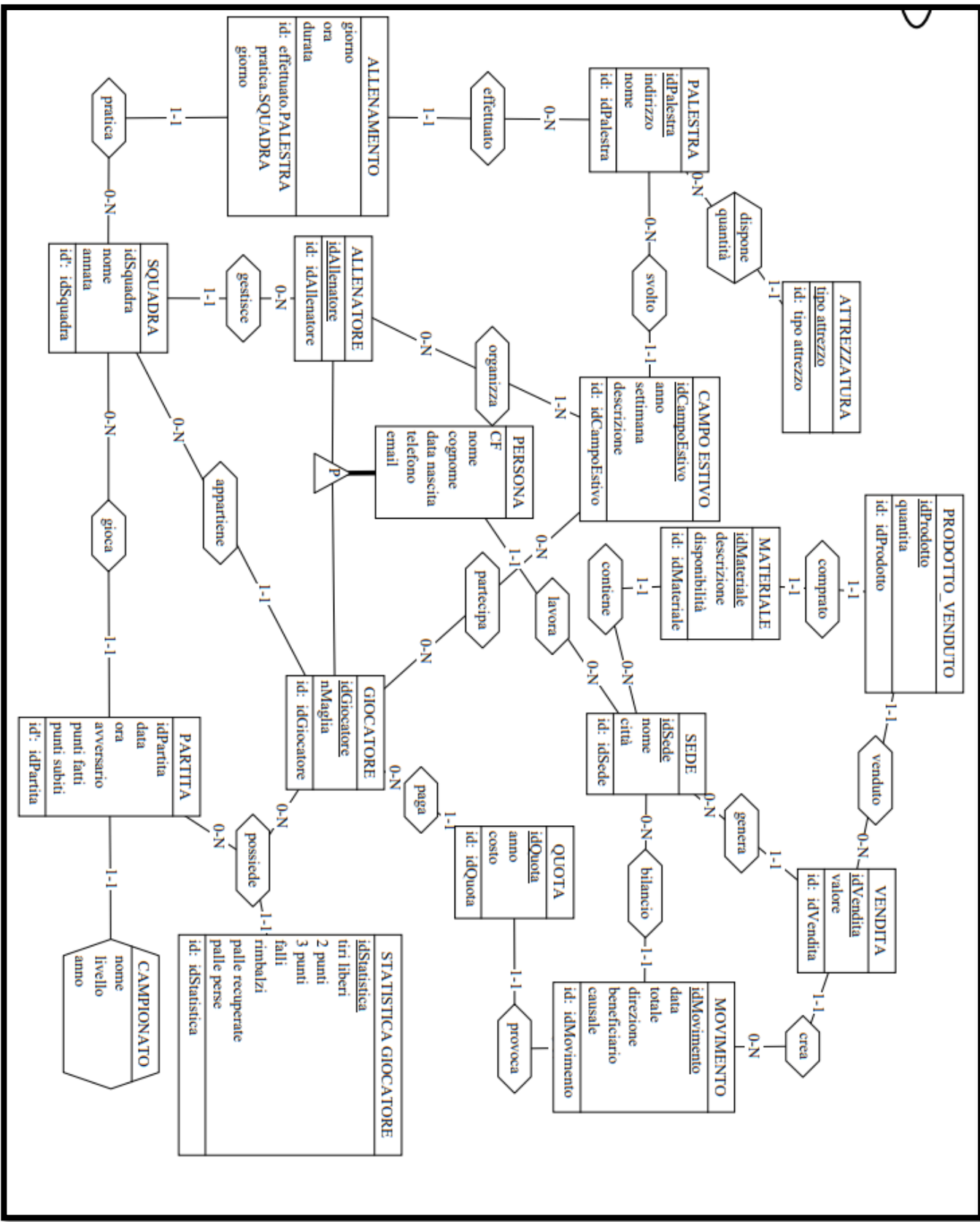
Tutti i giocatori della società fanno parte o devono entrare in una squadra. Ogni squadra è allenata da un allenatore. Le quadre, formate da un massimo di 15 giocatori, giocano le partite. Ogni partita possiede anche le relative informazioni sul campionato, lasciando libertà ad una squadra di poter partecipare anche a più campionati (anche negli anni).

Le squadre si allenano seguendo un programma di allenamenti. Degli allenamenti viene memorizzato l'orario e la periodicità settimanale (ovvero il giorno della settimana in cui l'allenamento viene svolto). Gli allenamenti e i campi estivi (organizzati dagli allenatori e ai quali partecipano i giocatori), vengono fatti nelle palestre. Ogni palestra infine dispone dell'attrezzatura, necessaria alle attività svolte, in quantità differenti.



La sede tiene traccia dei movimenti effettuati. I movimenti sono generati dalle quote pagate dai giocatori e dagli acquisti di materiale.

SCHEMA CONCETTUALE E/R



PROGETTAZIONE LOGICA

STIMA DEI VOLUMI

CONCETTO	COSTRUTTO	VOLUME
SEDE	E	1
ALLENATORE	E	10
Gestisce	R	40
Organizza	R	20
GIOCATORE	E	600
Partecipa	R	400
Appartiene	R	600
VENDITA	E	1 200
PRODOTTO_VEDNUTO	E	2000
Venduto	R	1200
Comprato	R	200
Genera	E	1200
Possiede	R	400 000
SQUADRA	E	40
ALLENAMENTO	E	120
Pratica	R	120
Effettuato	R	120
Iscritta	R	40
CAMPO ESTIVO	E	4
Svolto	R	10
PALESTRA	E	5
ATTREZZATURA	E	25
Dispone	R	25
CAMPIONATO	E	40
Costituito	R	800
PARTITA	E	800
Conta	R	460 000
STATISTICHE	E	480 000
QUOTA	E	1 000
Paga	R	750
MATERIALE	E	8
Contiene	R	8
MOVIMENTO	E	2000
Bilancio	R	2000
Provoca	R	1000
Crea	R	1200

DESCRIZIONE DELLE OPERAZIONI PRINCIPALI E STIMA DELLA LORO FREQUENZA

NUMERO	OPERAZIONE	FREQUENZA
1	Inserire allenatore	3 all'anno
2	Inserire giocatore assegnandogli una squadra	30 all'anno
3	Visualizzare il numero totale di persone della società	2 all'anno
4	Creare una squadra	2 all'anno
5	Modificare annata di una squadra	40 all'anno
6	Rimuovere un giocatore da una squadra	10 all'anno
7	Assegnare un allenatore ad una squadra	5 all'anno
8	Visualizzare i giocatori di una squadra	800 all'anno
9	Assegnare un giocatore ad un campo estivo	320 all'anno
10	Assegnare un allenatore ad un campo estivo	20 all'anno
11	Visualizzare i giocatori di un campo estivo	24 all'anno
12	Visualizzare le attrezzature di una palestra	5 al mese
13	Visualizzare le partite in un determinato periodo	40 a settimana
14	Visualizzare in ordine decrescente i campionati che ha giocato una squadra nel tempo.	120 all'anno
15	Visualizzare gli allenamenti di una squadra e in che palestra	45 a settimana
16	Acquistare x paia di calzini	3 al mese
17	Visualizzare i materiali disponibili	1 a settimana
18	Visualizzare numero vendite fatte nelle sedi	1 al mese
19	Visualizzare numero di giocatori in una squadra	800 all'anno
20	Visualizzare un estratto conto dei movimenti	1 al mese

TABELLE DEGLI ACCESSI

OP 1 – INSERIRE ALENATORE

CONCETTO	COSTRUTTO	ACCESSI	TIPO
Allenatore	E	1	S

Tot: 1S

Frequenza: 3 all'anno

Costo totale: $3 \times (2 \times 1) = 6$ all'anno

OP2 – CREARE GIOCATORE ASSEGNARGLI UNA SQUADRA

CONCETTO	COSTRUTTO	ACCESSI	TIPO
Giocatore	E	1	S
Appartiene	R	1	S

Tot: 2S

Frequenza: 30 all'anno

Costo totale: $30 \times (2 \times 2) = 120$ all'anno

OP3 - VISUALIZZARE IL NUMERO TOTALE DI PERSONE DELLA SOCIETÀ

CONCETTO	COSTRUTTO	ACCESSI	TIPO
Allenatore	E	10	L
Giocatore	E	600	L

Tot: 610L

Frequenza: 2 all'anno

Costo totale: $2 \times 610 = 1220$ all'anno

OP4 - CREARE UNA SQUADRA

CONCETTO	COSTRUTTO	ACCESSI	TIPO
Squadra	E	1	S

Tot: 1S

Frequenza: 2 all'anno

Costo totale: $2 \times (1 \times 2) = 4$ all'anno

OP5 – MODIFICARE ANNATA DI SQUADRA

CONCETTO	COSTRUTTO	ACCESSI	TIPO
Squadra	E	1	L
Squadra	E	1	S

Tot: 1L + 1S

Frequenza: 40 all'anno

Costo totale: $40 \times (1 + 1 \times 2) = 120$ all'anno

OP6 – RIMUOVERE UN GIOCATORE DA UNA SQUADRA

Dato che il numero massimo di giocatori in una squadra è di 15 prendo come valore medio 8.

CONCETTO	COSTRUTTO	ACCESSI	TIPO
Appartiene	R	8	L
Appartiene	R	1	S

Tot: 8L + 1S

Frequenza: 10 all'anno

Costo totale: $10 \times (8+1 \times 2) = 100$ all'anno

OP7 - ASSEGNARE UN ALLENATORE AD UNA SQUADRA

CONCETTO	COSTRUTTO	ACCESSI	TIPO
Gestisce	R	1	L
Gestisce	R	1	S

Tot: 1L + 1S

Frequenza: 20 all'anno

Costo totale: $20 \times (1+1 \times 2) = 60$ all'anno

OP8 - VISUALIZZARE I GIOCATORI DI UNA SQUADRA

Anche in questo caso prendo un valore medio di giocatori della squadra. Per esperienza mediamente una squadra possiede almeno 10 giocatori.

CONCETTO	COSTRUTTO	ACCESSI	TIPO
Appartiene	R	10	L
Giocatore	E	10	L

Tot: 20L

Frequenza: 800 all'anno

Costo totale: $800 \times 20 = 1600$ all'anno

OP9 - ASSEGNARE UN GIOCATORE AD UN CAMPO ESTIVO

Il numero medio stimato di giocatori in un campo estivo è 80. Un giocatore per partecipare ad un Campo Estivo deve aver già pagato la quota. Mediamente un giocatore nella sua carriera paga 4 quote.

CONCETTO	COSTRUTTO	ACCESSI	TIPO
Partecipa	R	80	L
Paga	R	4	L
Quota	E	4	L
Partecipa	R	1	S

Tot: 88L + 1S

Frequenza: 320 all'anno

Costo totale: $320 \times (88+1 \times 2) = 28\,800$ all'anno

OP10 - ASSEGNARE UN ALLENATORE AD UN CAMPO ESTIVO

CONCETTO	COSTRUTTO	ACCESSI	TIPO
Organizza	R	1	S

Tot: 1S

Frequenza: 40 all'anno

Costo totale: $40 \times (1 \times 2) = 80$ all'anno

OP11 – VISUALIZZARE I GIOCATORI DI UN CAMPO ESTIVO

Mediamente i giocatori in un campo estivo sono 80.

CONCETTO	COSTRUTTO	ACCESSI	TIPO
Partecipa	R	80	L
Giocatori	E	80	L

Tot: 160L

Frequenza: 24 all'anno

Costo totale: $24 \times 160 = 3\,840$ all'anno

OP12 – VISUALIZZARE LE ATTREZZATURE DI UNA PALESTRA

Dato che l'attrezzatura totale è formata da 25 oggetti divisi nelle palestre e in quantità differenti, mediamente per ogni palestra ci saranno 5 attrezzi.

CONCETTO	COSTRUTTO	ACCESSI	TIPO
Dispone	R	5	L
Attrezzatura	E	5	L

Tot: 10L

Frequenza: 5 al mese

Costo totale: $5 \times 10 = 50$ al mese

OP13 – VISUALIZZARE LE PARTITE DI UNA SQUADRA

Una squadra partecipa di solito ad un campionato all'anno per 14 anni. Uso come valore medio 7. In un campionato invece ci sono 20 partite.

CONCETTO	COSTRUTTO	ACCESSI	TIPO
Iscritta	R	7	L
Campionato	E	1	L
Costituito	R	20	L
Partite	E	20	L

Tot: 48L

Frequenza: 40 a settimana

Costo totale: $40 \times 48 = 1\,920$ a settimana

OP14 – VISUALIZZARE IL CAMPIONATO CORRENTE DI OGNI SQUADRA

Una squadra partecipa di solito ad un campionato all'anno per 14 anni. Uso come valore medio 7.

CONCETTO	COSTRUTTO	ACCESSI	TIPO
Iscritta	R	7	L
Campionato	E	1	L

Tot: 8L

Frequenza: 120 all'anno

Costo totale: $120 \times 8 = 960$ all'anno

OP15 – VISUALIZZARE GLI ALLENAMENTI DI UNA SQUADRA INCLUSA LA PALESTRA

Mediamente una squadra fa 3 allenamenti a settimana

CONCETTO	COSTRUTTO	ACCESSI	TIPO
Pratica	R	3	L
Allenamento	E	3	L
Effettuato	R	3	L
Palestra	E	3	L

Tot: 12L

Frequenza: 45 a settimana

Costo totale: $45 \times 12 = 540$ a settimana

OP16 – ACQUISTARE X PAIA DI CALZINI

Nell'operazione è necessario cercare il materiale indicato e valutarne la disponibilità. Eseguito l'acquisto sarà poi necessario aggiornare la disponibilità del materiale e generare uno movimento. I materiali in tutto sono 8, mediamente saranno richiesti 4 accessi per trovare il materiale desiderato.

CONCETTO	COSTRUTTO	ACCESSI	TIPO
Materiale	E	4	L
Vendita	R	1	S
Prodotto_Venduto	E	1	s
Materiale	E	1	L
Materiale	E	1	S
Crea	R	1	S
Movimento	E	1	S

Tot: 5L + 5S

Frequenza: 3 al mese

Costo totale: $3 \times (5 + 10) = 45$ al mese

OP17 – VISUALIZZARE I MATERIALI DISPONIBILI

Materiali in tutto sono 8, mediamente saranno richiesti 4 accessi per trovare il materiale desiderato.

CONCETTO	COSTRUTTO	ACCESSI	TIPO
Materiale	E	4	L

Tot: 4L

Frequenza: 1 a settimana

Costo totale: $1 \times 4 = 4$ a settimana

OP18 – VISUALIZZARE IL GUADAGNO TOTALE

CONCETTO	COSTRUTTO	ACCESSI	TIPO
Scontrino	E	75	L

Tot: 75L

Frequenza: 1 al mese

Costo totale: $1 \times 75 = 75$ al mese

OP19 - VISUALIZZARE NUMERO DI GIOCATORI IN UNA SQUADRA

CONCETTO	COSTRUTTO	ACCESSI	TIPO
Squadra	E	1	L

Tot: 1L

Frequenza: 800 all'anno

Costo totale: $800 \times 1 = 800$ all'anno

OP20 - VISUALIZZARE UN ESTRATTO CONTO DEI MOVIMENTI

CONCETTO	COSTRUTTO	ACCESSI	TIPO
Squadra	E	1	L

Tot: 75L

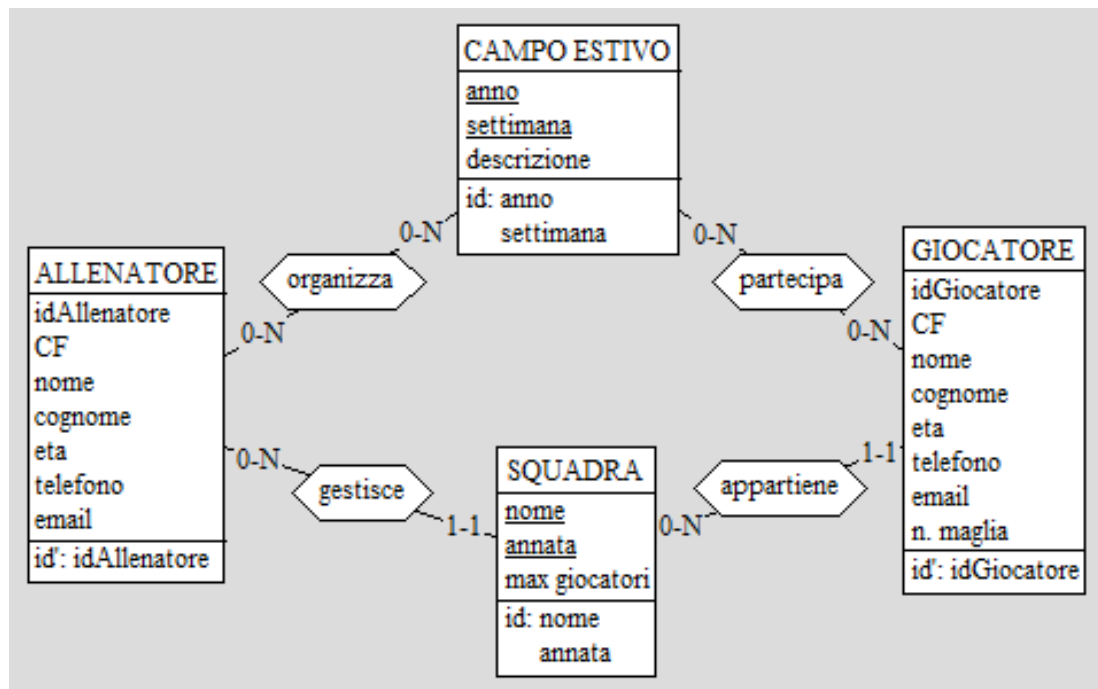
Frequenza: 1 al mese

Costo totale: $1 \times 75 = 75$ al mese

RAFFINAMENTO DELLO SCHEMA

Eliminazione delle gerarchie

Per l'eliminazione della gerarchia PERSONA (t, e), abbiamo deciso di adottare come soluzione il collasso verso il basso, copiando gli attributi in istruttore e iscritto. La scelta è motivata dal fatto che la gerarchia è totale ed esclusiva e rappresenta meglio la società di pallacanestro mantenendo le entità ben distinte e con ruoli diversi. Inoltre la cardinalità della relazione che esiste tra allenatore e squadra è ben diversa da quella che c'è fra giocatore e squadra.



Scelta delle chiavi

Nello schema si è preferito scegliere un identificativo per ogni entità che è stata creata in modo da facilitare lo sviluppo implementativo. In generale nelle tabelle sono presenti numerose altre chiavi candidate che potrebbero rimpiazzare l'id, un esempio può essere il codice fiscale nelle specializzazioni di persona (giocatore e allenatore). Anche le statistiche dei giocatori potrebbero essere definite dalle chiavi esterne di giocatore e partita che importano, ma preferiamo comunque attribuirgli un id specifico.

Eliminazione identificativi esterni

Dallo schema E/R sono state eliminate le seguenti relazioni:

- Relazione "effettuato" tra PALESTRA e ALLENAMENTO, importando la chiave esterna idPalestra in ALLENAMENTO
- Relazione "pratica" tra SQUADRA e ALLENAMENTO, importando la chiave esterna idSquadra in ALLENAMENTO
- Relazione "gestisce" tra SQUADRA e ALLENATORE, importando la chiave esterna idAllenatore in SQUADRA

- Relazione “appartiene” tra GIOCATORE e SQUADRA, importando la chiave esterna idGiocatore in SQUADRA
- Relazione “organizza” tra CAMPO ESTIVO e ALLENATORE, importando la chiave esterna idAllenatore in CAMPO ESTIVO
- Relazione “partecipa” tra GIOCATORE e CAMPO ESTIVO, importando la chiave esterna idGiocatore in CAMPO ESTIVO
- Relazione “possiede” tra GIOCATORE, STATISTICA GIOCATORE e PARTITA, importando la chiave esterna idGiocatore e idPartita in STATISTICA GIOCATORE
- Relazione “paga” tra QUOTA e GIOCATORE, importando la chiave esterna idGiocatore in QUOTA
- Relazione “dispone” tra PALESTRA e ATTREZZATURA, importando la chiave esterna idPalestra in ATTREZZATURA
- L’attributo composto CAMPIONATO viene importato in PARTITA in quanto relazione 1-1
- Relazione “svolto” tra CAMPO ESTIVO e PALESTRA, importando la chiave esterna idPalestra in CAMPO ESTIVO
- Relazione “acquista” tra MATERIALE e GIOCATORE, importando la chiave esterna idGiocatore in MATERIALE
- Relazione “lavora” tra ALLENATORE, GIOCATORE e SEDE, importando la chiave esterna idSede in ALLENATORE, GIOCATORE
- Relazione “provoca” tra QUOTA e MOVIMENTO, importando la chiave esterna idMovimento in QUOTA
- Relazione “bilancio” tra SEDE e MOVIMENTO, importando la chiave esterna idSede in MOVIMENTO
- Relazione “crea” tra VENDITA e MOVIMENTO, importando la chiave esterna idMovimento in VENDITA
- Relazione “genera” tra SEDE e VENDITA, importando la chiave esterna idSede in VENDITA
- Relazione “contiene” tra SEDE e MATERIALE, importando la chiave esterna idSede in MATERIALE

Analisi delle ridondanze

La ridondanza sarebbe introdotta con l'aggiunta dell'attributo nGiocatori all'interno di SQUADRA

OP2 - INSERIRE GIOCATORE ASSEGNANDOGLI UNA SQUADRA

Caso con ridondanza: dovrei aggiornare ogni volta che inserisco un giocatore anche la squadra

CONCETTO	COSTRUTTO	ACCESSI	TIPO
Giocatore	E	1	S
Appartiene	R	1	S
Squadra	E	1	L
Squadra	E	1	S

Tot: 1L + 3S

Aggiornamento = 1L + 1S

Frequenza: 30 all'anno

Costo totale: $30 \times (1 + 3 \times 2) = 210$ all'anno

Caso senza ridondanza:

CONCETTO	COSTRUTTO	ACCESSI	TIPO
Giocatore	E	1	S
Appartiene	R	1	S

Tot: 2S

Frequenza: 30 all'anno

Costo totale: $30 \times (2 \times 2) = 120$ all'anno

OP19 – VISUALIZZARE NUMERO GIOCATORI IN UNA SQUADRA

Caso con ridondanza:

CONCETTO	COSTRUTTO	ACCESSI	TIPO
Squadra	E	1	L

Tot: 1L

Frequenza: 800 all'anno

Costo totale: $800 \times 1 = 800$ all'anno

Caso senza ridondanza:

CONCETTO	COSTRUTTO	ACCESSI	TIPO
Giocatore	E	1	L
Appartiene	R	15	L

Tot: 16L

Frequenza: 800 all'anno

Costo totale: $800 \times 16 = 12\,800$ all'anno

Si decide pertanto di mantenere la ridondanza, privilegiando l'efficienza.

Un'altra apparente ridondanza potrebbe essere quella data dagli attributi totale e direzione di movimento. La direzione potrebbe essere data infatti dal segno del totale, ma preferiamo considerare tale attributo come una quantità di denaro sempre positiva che può entrare o uscire dalla società.

TRADUZIONE DI ENTITÀ E ASSOCIAZIONI IN RELAZIONI

MOVIMENTO (idMovimento, totale, direzione, beneficiario, causale, idSede)

FK: idSede REFERENCES SEDE

ACQUISTO (idAcquisto, data, valore, idMovimento, idSede)

FK: idMovimento REFERENCES MOVIMENTO

FK: idSede REFERENCES SEDE

GIOCATORE (idGiocatore, CF, nome, cognome, eta, telefono, email, nMaglia)

FK: idSede REFERENCES SEDE

ALLENATORE (idAllenatore, CF, nome, cognome, eta, telefono, email, idSede)

FK: idSede REFERENCES SEDE

SQUADRA (idSquadra, nome, annata, nGiocatori, idGiocatore, idAllenatore)

appartiene (id, idSquadra, idGiocatore)

FK: idGiocatore REFERENCES GIOCATORE

FK: idSquadra REFERENCES SQUADRA

gestisce (id, idAllenatore, idSquadra)

FK: idAllenatore REFERENCES ALLENATORE

FK: idSquadra REFERENCES SQUADRA

PARTITA (idPartita, data, ora, avversario, punti_fatti, punti_subiti, campionato_livello, campionato_nome, campionato_anno, idSquadra)

FK: idSquadra REFERENCES SQUADRA

STATISTICHE GIOCATORE (idGiocatore, idPartita, tiri_liberi, due_punti, tre_punti, falli, rimbalzi, palle recuperate, palle perse)

FK: idGiocatore REFERENCES GIOCATORE

FK: idPartita REFERENCES PARTITA

ALLENAMENTO (idSquadra, giorno, ora, durata, idPalestra)

FK: idSquadra REFERENCES SQUADRA

FK: idPalestra REFERENCES PALESTRA

PALESTRA (idPalestra, indirizzo, nome)

ATTREZZATURA (idAttrezzatura, quantità, descrizione, idPalestra)

FK: idPalestra REFERENCES PALESTRA

CAMPO_ESTIVO (idCampoEstivo, anno, settimana, descrizione, idPalestra)

FK: idPalestra REFERENCES PALESTRA

partecipa (id, idGiocatore, idCampoEstivo)

FK: idGiocatore REFERENCES GIOCATORE

FK: idCampoEstivo REFERENCES CAMPO_ESTIVO

organizza (id, idAllenatore, idCampoEstivo)

FK: idAllenatore REFERENCES ALLENATORE

FK: idCampoEstivo REFERENCES CAMPO_ESTIVO

QUOTA (idQuota, data, costo, idGiocatore, idMovimento)

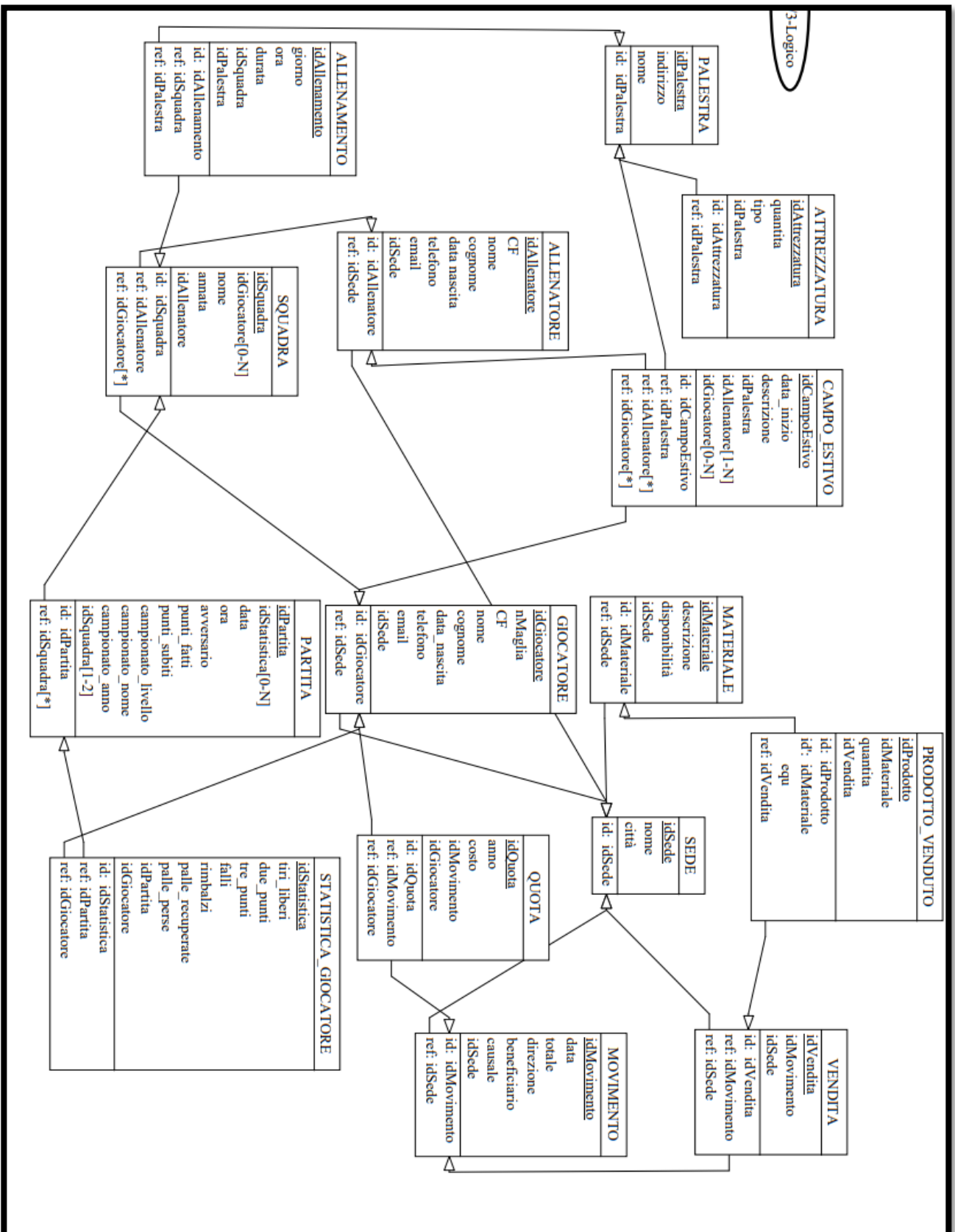
FK: idGiocatore REFERENCES GIOCATORE

FK: idMovimento REFERENCES MOVIMENTO

MATERIALE (idMateriale, descrizione, disponibilità, prezzo, quantità)

FK: idSede REFERENCES SEDE

SCHEMA RELAZIONALE



CREAZIONE DELLE QUERY E SVILUPPO APPLICATIVO

TRADUZIONE DELLE OPERAZIONI IN QUERY SQL

Lo schema è stato generato automaticamente da DB-Main e rielaborato, sistemando i tipi di dato degli attributi e creando, ove necessario nuove tabelle per trattare meglio, a livello implementativo, alcune relazioni molti a molti o gestire indipendentemente alcune entità da altre e poi creare relazioni se necessario.

```
create table MOVIMENTO (  
    id integer not null,  
    data date not null,  
    totale integer not null,  
    direzione char(32) not null,  
    beneficiario char(256) not null,  
    causale char(256) not null,  
    idSede integer not null,  
    constraint ID primary key (id),  
    foreign key (idSede) references SEDE);  
  
create table VENDITA (  
    id integer not null,  
    idMovimento integer not null,  
    idSede integer not null,  
    constraint ID primary key (id),  
    foreign key (idMovimento) references MOVIMENTO,  
    foreign key (idSede) references SEDE);  
  
create table PRODOTTO_VENDUTO (  
    id integer not null,  
    id_vendita integer not null,  
    id_prodotto integer not null,  
    quantita integer not null,  
    constraint id primary key (id),  
    foreign key (id_vendita) references VENDITA,  
    foreign key (id_prodotto) references MATERIALE);  
  
create table SEDE (  
    id integer not null,  
    nome char(64),  
    citta char(64),  
    constraint ID primary key (id));  
  
create table ALLENAMENTO (  
    id integer not null,  
    giorno_settimanale char(32) not null,  
    ora_inizio integer not null,
```

```
ora_fine integer not null,  
idSquadra integer not null,  
idPalestra integer not null,  
constraint ID primary key (id),  
foreign key (idSquadra) references SQUADRA,  
foreign key (idPalestra) references PALESTRA);
```

```
create table ALLENATORE (  
    id integer not null,  
    CF char(32) not null,  
    nome char(32) not null,  
    cognome char(64) not null,  
    data_nascita date not null,  
    telefono integer not null,  
    email char(128) not null,  
    idSede integer not null,  
    constraint ID primary key (id),  
    foreign key (idSede) references SEDE);
```

```
create table ATTREZZATURA (  
    id integer not null,  
    quantita integer not null,  
    tipo char(64) not null,  
    idPalestra integer not null,  
    constraint ID primary key (id),  
    foreign key (idPalestra) references PALESTRA);
```

```
create table CAMPO_ESTIVO (  
    id integer not null,  
    data_inizio date not null,  
    descrizione char(256) not null,  
    idPalestra integer,  
    constraint ID primary key (id),  
    foreign key (idPalestra) references PALESTRA);
```

```
create table GIOCATORE (  
    id integer not null,  
    nMaglia integer not null,  
    CF char(64) not null,  
    nome char(64) not null,  
    cognome char(128) not null,  
    data_nascita date not null,  
    telefono integer not null,  
    email char(128) not null,  
    idSede integer not null,  
    constraint ID primary key (id)  
    foreign key (idSede) references SEDE);
```

```
create table MATERIALE (  
    id integer not null,  
    descrizione char(256) not null,  
    disponibilità integer not null,  
    prezzo integer not null,  
    idSede integer not null,  
    constraint ID primary key (id)  
    foreign key (idSede) references SEDE);  
  
create table PALESTRA (  
    id integer not null,  
    indirizzo char(256) not null,  
    nome char(64) not null,  
    constraint ID primary key (id));  
  
create table PARTITA (  
    id integer not null,  
    idStatistica integer,  
    data date not null,  
    ora integer not null,  
    avversario char(128) not null,  
    punti_fatti integer not null,  
    punti_subiti integer not null,  
    campionato_livello char(32) not null,  
    campionato_nome char(64) not null,  
    campionato_anno integer not null,  
    idSquadra integer not null,  
    constraint ID primary key (id),  
    foreign key (idSquadra) references SQUADRA);  
  
create table QUOTA (  
    id integer not null,  
    anno integer not null,  
    costo integer not null,  
    idGiocatore integer not null,  
    idMovimento integer not null,  
    constraint ID primary key (id),  
    foreign key (idGiocatore) references GIOCATORE,  
    foreign key (idMovimento) references MOVIMENTO);  
  
create table SQUADRA (  
    id integer not null,  
    nome char(32) not null,  
    annata integer not null,  
    constraint ID primary key (id));
```

```
create table STATISTICA_GIOCATORE (  
    id integer not null,  
    tiri_liberi integer not null,  
    due_punti integer not null,  
    tre_punti integer not null,  
    falli char(1) not null,  
    rimbalzi integer not null,  
    palle_recuperate integer not null,  
    palle_perse integer not null,  
    idPartita integer not null,  
    idGiocatore integer not null,  
    constraint ID primary key (id),  
    foreign key (idPartita) references PARTITA,  
    foreign key (idGiocatore) references GIOCATORE);  
  
create table appartiene (  
    id integer primary key not null,  
    idSquadra integer not null,  
    idGiocatore integer not null);  
  
create table gestisce (  
    id integer primary key not null,  
    idAllenatore integer not null,  
    idSquadra integer not null);  
  
create table partecipa (  
    id integer primary key not null,  
    idGiocatore integer not null,  
    idCampoEstivo integer not null);  
  
create table organizza (  
    id integer primary key not null,  
    idAllenatore integer not null,  
    idCampoEstivo integer not null);
```


QUERY

Le Queries sono state scritte sulla base delle 20 operazioni scelte come più frequenti.

```
-- OP1 Inserire allenatore
INSERT INTO ALLENATORE (CF, nome, cognome, eta, telefono, email)
VALUES (?, ?, ?, ?, ?, ?);

-- OP2 Inserire giocatore assegnandogli una squadra
INSERT INTO GIOCATORE (CF, nome, cognome, eta, telefono, email, nMaglia)
VALUES (?, ?, ?, ?, ?, ?, ?);
INSERT INTO appartiene (idGiocatore, idSquadra)
VALUES (?, ?);

-- OP3 Visualizzare il numero totale di persone della società
SELECT
(SELECT COUNT(*) FROM ALLENATORE) +
(SELECT COUNT(*) FROM GIOCATORE);

-- OP4 Creare una squadra
INSERT INTO SQUADRA (nome, annata, nGiocatori)
VALUES (?, ?, ?);

-- OP5 Modificare annata di una squadra
UPDATE SQUADRA
WHERE ID = ?
SET annata = ?;

-- OP6 Rimuovere un giocatore da una squadra
DELETE FROM appartiene
WHERE GIOCATORE.id = ?;

-- OP7 Assegnare un allenatore ad una squadra
INSERT INTO gestisce (idAllenatore, idSquadra)
VALUES(?, ?);

-- OP8 Visualizzare i giocatori di una squadra
SELECT GIOCATORE
FROM appartiene A, GIOCATORE G
WHERE A.idGiocatore = G.idGiocatore AND A.idSquadra = ?;

-- OP9 Assegnare un giocatore ad un campo estivo
INSERT INTO partecipa (idGiocatore, idCampoEstivo)
VALUES (?, ?);

-- OP10 Assegnare un allenatore ad un campo estivo
INSERT INTO organizza (idAllenatore, idCampoEstivo)
VALUES (?, ?);
```

```

-- OP11 Visualizzare i giocatori di un campo estivo
SELECT GIOCATORE
FROM partecipa P
WHERE P.idCampoEstivo = ?;

-- OP12 Visualizzare le attrezzature di una palestra
SELECT ATTREZZATURA
FROM ATTREZZATURA A
WHERE A.idPalestra = ?;

-- OP13 Visualizzare le partite di una squadra tra due date
SELECT PARTITA
FROM PARTITA P
WHERE P.idSquadra = ?
AND P.data BETWEEN ? AND ?;

-- OP14 Visualizzare i campionati giocati da una squadra negli anni (decrescente)
SELECT PARTITA.campionato_anno, PARTITA.campionato_nome, PARTITA.campionato_livello
FROM PARTITA P
WHERE P.idSquadra = ?
ORDER BY P.campionato_anno DESC;

-- OP15 Visualizzare gli allenamenti di una squadra e in che palestra
SELECT ALLENAMENTO, PALESTRA
FROM ALLENAMENTO A, PALESTRA P
WHERE A.idPalestra = P.idPalestra
AND A.idSquadra = ?;
-- alternativa con join
SELECT *
FROM ALLENAMENTO
JOIN PALESTRA
ON A.idPalestra = P.idPalestra;
WHERE A.idSquadra = ?;

-- OP-16 Acquistare x paia di calzini
-- Innanzitutto, bisogna controllare che gli oggetti desiderati siano disponibili.
SELECT disponibilita -- Se ritorna un valore allora OK
FROM MATERIALE
WHERE idMateriale = ?
AND disponibilita >= ?;
-- Creo un Movimento
INSERT INTO MOVIMENTO (id, data, totale, direzione, beneficiario, causale, idSede)
VALUES (?, GETDATE(), ?, ?, ?, ?, ?);
-- Creo un nuovo acquisto
INSERT INTO VENDITA (id, valore, idMovimento, idSede)
VALUES (?, ?, ?, ?);
--creo record del prodotto venduto
INSERT INTO PRODOTTO_VENDUTO(id, id_vendita, id_prodotto, quantità)
VALUES (?, ?, ?, ?) --id prodotto è quello dell aprima query

```

```

-- A questo punto bisogna diminuire la quantità dell'oggetto.
UPDATE MATERIALE
SET quantita = quantita - ?
WHERE idMateriale = ?;

-- OP-17 Visualizzare i materiali disponibili
SELECT MATERIALI;

-- OP18 Visualizzare numero di vendite fatte nelle varie sedi
SELECT SEDE, COUNT(*) AS N_VENDITE
FROM ACQUISTO A
WHERE A.idSede = S.idSede
GROUP BY SEDE.idSede;

-- OP-19 Visualizzare numero di giocatori in ogni squadra
SELECT SQUARA, COUNT(GIOCATORE) AS Giocatori
GROUP BY SQUADRA.idSquadra;

-- OP20 Visualizzare i movimenti e il guadagno totale in un periodo di una sede
SELECT *
FROM MOVIMENTO
AND MOVIMENTO.idSede = ?
WHERE data BETWEEN ? AND ?

SELECT( SELECT SUM(MOVIMENTO.totale)
        FROM MOVIMENTO
        WHERE MOVIMENTO.direzione = "entrata"
        AND MOVIMENTO.idSede = ?
        AND MOVIMENTO.data BETWEEN ? And ?;
        -
        SELECT SUM(MOVIMENTO.totale)
        FROM MOVIMENTO
        WHERE MOVIMENTO.direzione = "uscita"
        AND MOVIMENTO.idSede = ?
        AND MOVIMENTO.data BETWEEN ? And ?;
        );

```

SVILUPPO DELL'APPLICATIVO

Abbiamo deciso di sviluppare l'applicativo su Python e con un interfaccia web in HTML tramite Flask. L'accesso al database tramite l'applicativo in Python avviene grazie alla libreria SQLite che grazie ad istruzioni in SQL ci ha permesso di creare il database e accedervi tramite query SQL.

Per lanciare l'applicativo è sufficiente avere installato python e la relativa libreria di flask. Istruzioni più dettagliate sono contenute nel file Readme.txt. Segue uno screen su i comandi eseguiti su bash.

```
tella@LAPTOP-NMM1QUQ8: /mnt/c/Users/tella_tasat48/Workspace/DataBaseBasket/flask$ flask --help
Usage: flask [OPTIONS] COMMAND [ARGS]...

A general utility script for Flask applications.

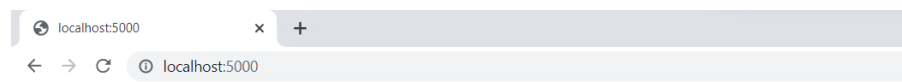
Provides commands from Flask, extensions, and the application. Loads the
application defined in the FLASK_APP environment variable, or from a wsgi.py
file. Setting the FLASK_ENV environment variable to 'development' will
enable debug mode.

$ export FLASK_APP=hello.py
$ export FLASK_ENV=development
$ flask run

Options:
  --version  Show the flask version
  --help     Show this message and exit.

Commands:
  routes  Show the routes for the app.
  run     Run a development server.
  shell   Run a shell in the app context.
tella@LAPTOP-NMM1QUQ8: /mnt/c/Users/tella_tasat48/Workspace/DataBaseBasket/flask$ export FLASK_APP=app.py
tella@LAPTOP-NMM1QUQ8: /mnt/c/Users/tella_tasat48/Workspace/DataBaseBasket/flask$ flask run
```

Una volta eseguiti i comandi da prompt si apre una pagina internet con indirizzo localhost:5000. Qui sarà reperibile la homepage del database, con i link alle varie pagine delle operazioni e le viste delle tabelle.



Basket Imola International

- [Elenco giocatori](#)
- [Registra giocatore](#)
- [Registra pagamento quota](#)
- [Elenco allenatori](#)
- [Registra allenatore](#)
- [Squadre](#)
- [Crea squadra](#)
- [Magazzino prodotti](#)
- [Registra vendita](#)

PAGINE PLAYERS E COACHES

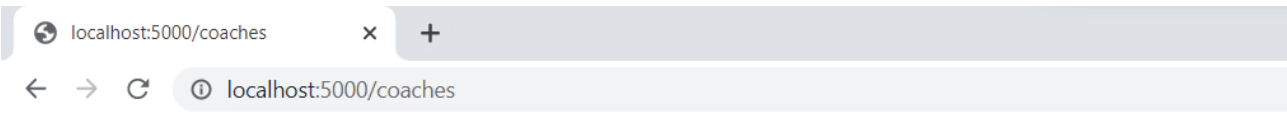
La prima pagina fornisce un elenco dei giocatori della società. Per ogni giocatore è poi possibile effettuare una modifica. Analoga è la quarta pagina che fornisce una vista degli allenatori.



Giocatori

Seleziona una sede -- Scegli una sede -- ▼

id	nMaglia	CF	nome	cognome	data_nascita	telefono	email	idSede	
1	4114	bbbb	genova	lacerenza	1974-04-01	2415456	genova@lacerenza.com	1	Modifica
2	9443	SQDLES40Z55S468B	dolorata	marino	2012-02-21	3064423	dolorata@marino.com	1	Modifica
3	5642	EOMKRB72Q41M555L	santin	tisba	1972-11-06	7633722	santin@tisba.com	1	Modifica
4	4836	MVQFXZ55V37Y6280	ferrandina	ciarla	1907-01-30	9963755	ferrandina@ciarla.com	1	Modifica
5	4835	BJHYKI63D910928Z	enerio	robbio	1940-06-30	1115352	enerio@robbio.com	1	Modifica
6	6020	AROUUT13E02Z8400	carmelito	colongi	2019-11-09	6911499	carmelito@colongi.com	1	Modifica
7	1529	PTNXFA11K79T494V	aristarco	armaroli	1982-03-11	4311366	aristarco@armaroli.com	1	Modifica
8	6665	WAMYGU27Z85Q007D	fioldisa	mugione	1904-10-05	7293165	fioldisa@mugione.com	2	Modifica



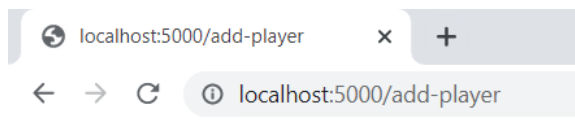
allenatori

Seleziona una sede -- Scegli una sede -- ▼

id	CF	nome	cognome	data_nascita	telefono	email	idSede	
1	DVZVSF40H00U875S	erminda	zolnowski	1977-03-26	1188352	erminda@zolnowski.com	1	Modifica
2	QILYOX70V29F824M	odette	tronconi	1928-10-02	4855981	odette@tronconi.com	1	Modifica
3	QETIXB90G55Z718E	agnello	conti	1976-04-04	8106431	agnello@conti.com	1	Modifica
4	TCVKZJ66B62X0730	gisberta	polidori	1953-05-19	2914657	gisberta@polidori.com	1	Modifica
5	HNWTEL34L80M214D	dolorice	benedetti	1950-03-19	5296230	dolorice@benedetti.com	1	Modifica
6	JHWTAJ50I17S584D	mohamed	quartarone	2007-07-30	9586231	mohamed@quartarone.com	1	Modifica
7	SLFUNV00N37M877L	ultima	bruni	1907-04-01	8633575	ultima@bruni.com	1	Modifica
8	MJVVYAL33W57U922K	danilla	palma	1975-10-29	6084893	danilla@palma.com	1	Modifica

PAGINE DI INSERIMENTO ADD-PLAYER e ADD-COACH

La seconda pagina invece serve ad inserire un nuovo giocatore all'interno della società.




Aggiungi giocatore

Nome

Cognome

Codice Fiscale

Data-nascita


 

Numero-maglia

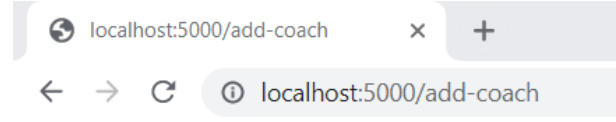
Telefono

Email

Sede

Registra




Aggiungi allenatore

Nome

Cognome

Codice Fiscale


Data-nascita

Telefono

Email

Sede

Registra

Altre pagine analoghe sono state sviluppate, come quelle per la registrazione della quota, la modifica di tabelle, acquisto di prodotti in base alla disponibilità del magazzino e possibilità di visualizzare i prodotti di magazzino.