

Relazione per il progetto di
“Basi di Dati”

Linda Fabbri,
Federico Raffoni,
Simone Rega

20 giugno 2021

Indice

1	Introduzione	3
2	Analisi dei Requisiti	4
2.1	Requisiti in linguaggio naturale	4
2.2	Estrazione dei concetti fondamentali	5
3	Progettazione Concettuale	7
3.1	Anteprima sviluppo delle "Persone"	7
3.2	Anteprima sviluppo dei "Videogiochi"	7
3.3	Anteprima sviluppo delle "Partite"	8
3.4	Anteprima sviluppo dei "Tornei"	10
3.5	Schema Generale	11
4	Progettazione Logica	13
4.1	Stima del volume dei dati	14
4.2	Descrizione delle operazioni principali e stima della loro frequenza	15
4.3	Schemi di navigazione e tabelle degli accessi	15
4.4	Raffinamento dello schema	17
4.5	Analisi delle ridondanze	18
4.6	Traduzione di entità e associazioni in relazioni	19
4.7	Schema relazionale finale	20
5	Progettazione Fisica	21
5.1	Traduzione in SQL	21
6	Progettazione dell'Applicazione	23
6.1	Descrizione della scelta del linguaggio e del DBMS	23
6.2	Descrizione dell'architettura	23
6.3	Interfaccia Utente	23
6.3.1	Amministratore Torneo	23

6.3.2	Giocatore	23
-------	---------------------	----

Capitolo 1

Introduzione

Il progetto consiste nella realizzazione di un sistema database che funga da supporto alla creazione di Tornei Internazionali di Videogiochi. Il database ha l'obiettivo principale di immagazzinare le informazioni relative a: videogiochi, giocatori e partite. L'applicazione permetterà la creazione di vari tornei in tutto il mondo consultando statistiche dei giocatori nei vari videogiochi e cercando il luogo migliore in cui ospitarli, ovvero con strutture adeguatamente attrezzate e tenendo conto dell'audience e sponsor locali.

Capitolo 2

Analisi dei Requisiti

La seguente descrizione riporta in linguaggio naturale i requisiti per il nostro sistema informativo, per poi poterne estrarre i principali concetti fondamentali:

2.1 Requisiti in linguaggio naturale

”Jeff Kaplan, prima di lasciare le redini del videogioco Overwatch, ha deciso di commissionare un sistema informativo di supporto per la gestione di tornei internazionali di cui finanzierà i premi. Si vuole creare una applicazione che dia la possibilità ai giocatori di iscriversi ad un torneo o venir reclutati in una squadra. Ogni giocatore può partecipare a più squadre contemporaneamente di giochi diversi, così come un Coach può allenare più squadre contemporaneamente. Ogni Squadra è allenata da un Coach e ha un numero massimo di Player che possono aderire ad essa e gioca ad un singolo Videogioco; il numero di adesioni è determinato dal Videogioco in questione. Si vuole tenere traccia dei Player iscritti, memorizzando di ognuno il nome, cognome, nickname, codice fiscale, stato in cui risiede, mail e statistiche di gioco (per statistiche si intendono il numero di partite vinte e giocate). Una Squadra può partecipare a uno o più Tornei purché il Videogioco su cui esso si basa è lo stesso giocato dalla Squadra. Per quanto riguarda i Tornei si vuole memorizzare: Stato, Città e Arena in cui si svolge, numero di squadre totali, videogioco per cui si disputa il torneo in questione e lo Sponsor che finanzierà il torneo stesso. Di ogni Videogioco si vuole tener traccia del Nome, della data di creazione, della sua azienda produttrice, della tipologia di gioco e del numero di componenti di ogni squadra. In ogni Arena possono assistere alle Partite un numero massimo di Spettatori, i quali per poter assistere dovranno

no pagare un Biglietto nominativo; ogni partita sarà visionata da un Arbitro e commentata da uno Speaker.

2.2 Estrazione dei concetti fondamentali

Soggetto	Descrizione	Sinonimi
Player	Colui che gioca ad almeno un videogioco e si iscrive ad un Torneo previ adesione ad una Squadra	Videogiocatore
Squadra	Gruppo di persone che giocano allo stesso videogioco, la sua grandezza è determinata dal videogioco stesso	Team
Coach	Colui che allena la Squadra	Allenatore
Arbitro	Colui che regola e visiona le partite del torneo	-
Speaker	Colui che commenta in tempo reale le partite del torneo	Commentatore
Spettatore	Colui che compra un biglietto per assistere ad una partita di un torneo	-
Biglietto	Ticket univoco e nominativo che permette la visualizzazione di una partita di un torneo ad uno spettatore in una precisa data	Ticket
Videogioco	Software videoludico a cui giocano i player e su cui si basano i tornei	Videogame
Azienda Videogioco	Software house che sviluppa il videogioco	Casa Produttrice
Sponsor	Aziende o compagnie che sponsorizzano il torneo e lo finanziano	-
Arena	Luogo fisico dove si svolgono tutte le partite di un determinato torneo	Stadio
Partita	Insieme di scontri virtuali tra due squadre	Game, Match

A seguito della lettura e comprensione dei requisiti, si procede redigendo un testo che ne riassume tutti i concetti e in particolare ne estragga quelli principali eliminando le ambiguità sopra rilevate:

Per ogni **Player** si memorizzano: Codice Fiscale, nickname, nome e cognome, genere, mail e data di nascita. Ogni Player può partecipare ad una sola squadra per volta. Ogni **Player** può giocare a più Videogiochi.

Per ogni **Videogame** si memorizzano: nome, data di creazione.

Per ogni **Squadra** si memorizzano: IdSquadra, nome e data di creazione. Una specifica squadra può giocare a più videogiochi. La **Squadra** è composta da 5 **Player** ed *eventualmente* 1 **Coach**. La **Squadra** può iscriversi a più **Tornei** contemporaneamente.

Per ogni **Torneo** si memorizzano : la data di inizio, la data di fine e il numero massimo di iscrizioni. Il **Torneo** si svolge interamente in una singola **Arena** e può essere finanziato da uno **Sponsor**. Il **Torneo** inoltre riguarda un singolo **Videogioco** e prevede diverse **Partite**. Ad ogni **Torneo** possono partecipare un numero variabile di squadre (che non superano mai le 15 squadre) che si sfidano tutti contro tutti e la squadra con più partite vinte vincerà il torneo.

Per ogni **Partita** si memorizzano: le due squadre che si sfidano e la data dell'incontro.

Per ogni **Biglietto** si memorizzano: il costo, la Partita e l'Arena in cui si disputa.

Per ogni **Spettatore** si vuole memorizzare: Codice Fiscale, nome e cognome, genere, mail e data di nascita. Ogni **Spettatore** può comprare un solo **Biglietto** per una determinata **Partita**.

Segue un elenco delle principali azioni richieste:

- Aggiunta di un nuovo Player
- Aggiunta Videogioco giocato da un Player
- Aggiunta di un nuovo Spettatore
- Creazione di una Squadra
- Aggiunta di un Player ad una Squadra
- Creazione di un Torneo
- Iscrizione di una Squadra ad un Torneo
- Creazione di nuove Partite
- Acquisto di Biglietti

Capitolo 3

Progettazione Concettuale

3.1 Anteprima sviluppo delle "Persone"

In questa sezione verrà modellato l'aspetto di **Persona**. Si decide di dividere i vari aspetti delle varie Persone che saranno presenti nel Database in più entità : **Player**, **Coach**, **Spettatore**, **Arbitro** e **Speaker**.

Ognuno di queste entità citate sopra sarà figlia della classe padre **Persona**, che contiene gli attributi comuni a tutte le entità figlie.

Si è deciso di rendere la gerarchia *totale* e *sovrapposta* poiché le persone fisiche di cui abbiamo la necessità di salvarci i dati sono esattamente e solamente le classi figlie di **Persona** citate prima; inoltre è possibile che una classe figlia di **Persona** possa ricoprire più ruoli, ad esempio un **Player** potrebbe essere uno **Spettatore** come potrebbe essere un **Coach**.

Di **Persona** si vogliono conoscere le caratteristiche base utili alla gestione delle entità figlie, quindi: Codice Fiscale (il quale identifica la persona), Nome, Cognome, Mail, Data di nascita e Genere. Se necessario ogni entità figlia potrà aggiungere agli attributi del padre i propri, ad esempio **Player** avrà un Nickname, **Arbitro** un counter di PartiteArbitrate e lo **Speaker** un Nickname e un counter di PartiteCommentate.

3.2 Anteprima sviluppo dei "Videogiochi"

In questa sezione verrà modellato l'aspetto di **Videogioco**. Il **Videogioco** ha un Nome, dal quale viene identificato, e una data di creazione. Esiste una relazione con una **Azienda di Videogiochi** la quale può creare più Videogiochi.

Ogni Videogioco ha la propria **Tipologia** , ad esempio Sparatutto o Giochi di Carte.

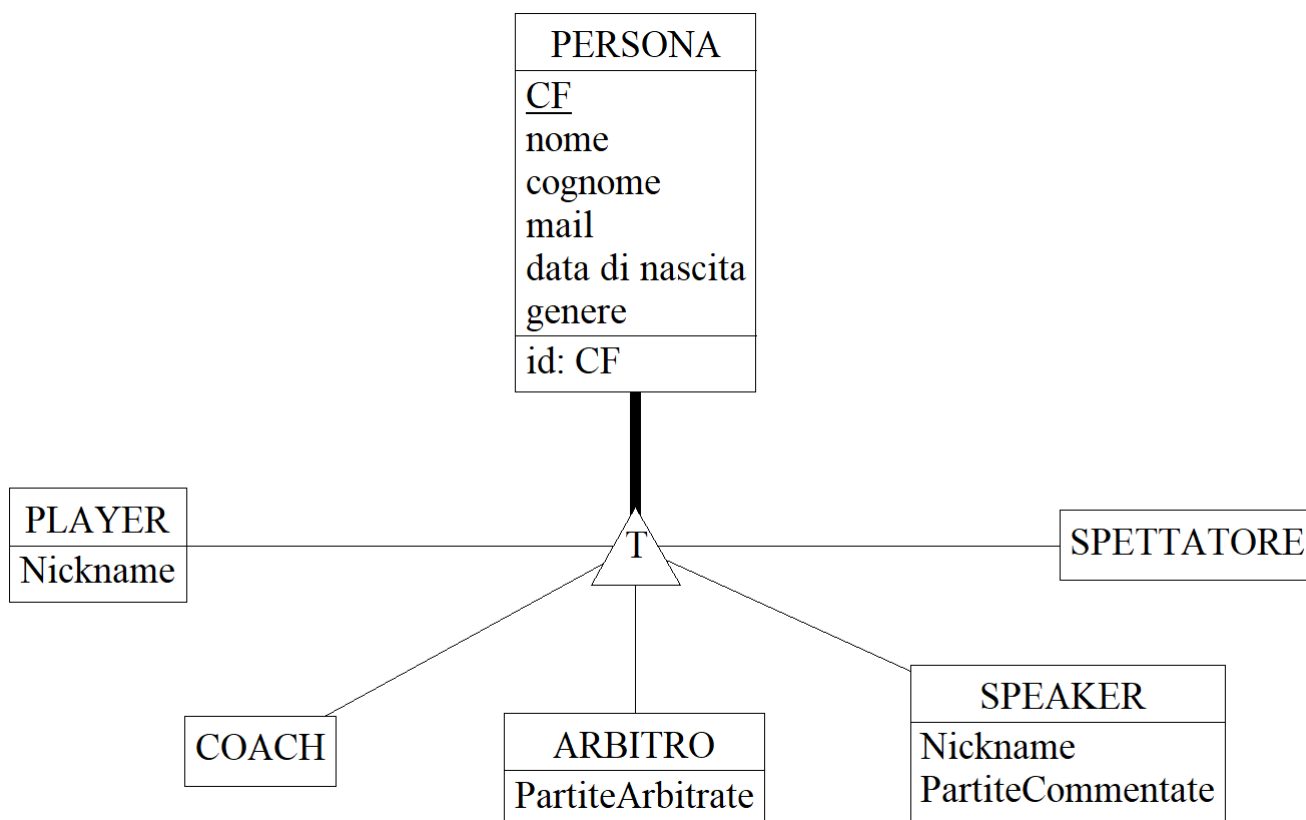


Figura 3.1: Schema ER che espone le principali caratteristiche delle Persone

Abbiamo la necessità di salvarci ogni **Player** a quale e quanti Videogiochi **Gioca**, memorizzando nel frattempo le sue Partite Vinte e le Partite Giocate (*Nota Bene* : questi due attributi si riferisco a statistiche personali relative all'avanzamento nel gioco, non sono collegate in nessun modo all'entità dell'E-R chiamata "Partita", quest'ultima si riferisce esclusivamente ad un match all'interno di un Torneo).

3.3 Anteprima sviluppo delle "Partite"

In questa sezione verrà modellato l'aspetto di **Partita**. L'entità **Partita** è probabilmente uno degli elementi più importanti e centrali di tutto lo schema E-R. Ogni **Partita** è identificata univocamente dalle due **Squadre** che parteciperanno all'incontro e dalla Data e Ora in cui si svolgerà.

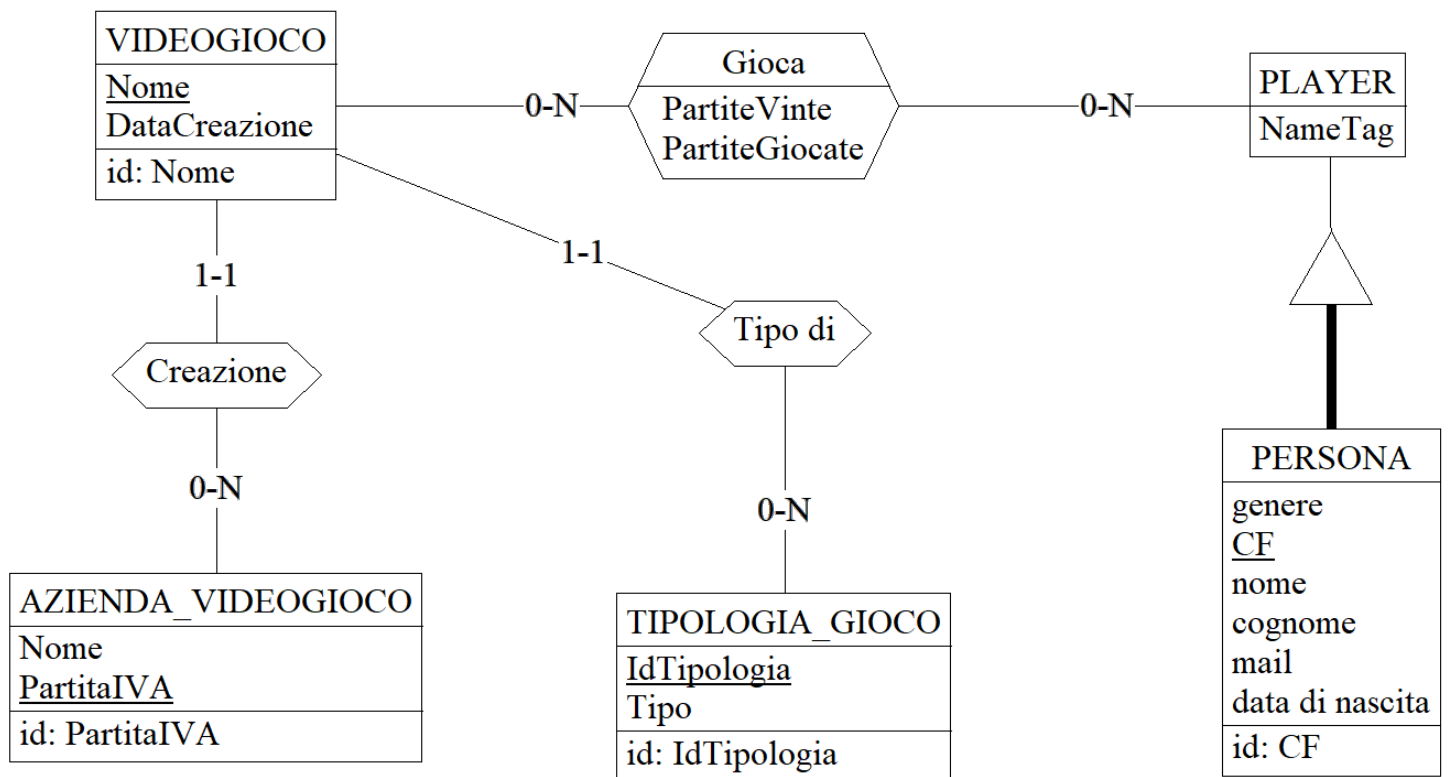


Figura 3.2: Schema ER che espone le principali caratteristiche dei Videogiochi

Di ogni partita vogliamo memorizzarci ovviamente l'id delle due squadre che vi partecipano e anche l'id della **Squadra** che vincerà effettivamente lo scontro.

Ad ogni Partita inoltre partecipano due figure professionali, figlie della classe **Persona**: l'**Arbitro** che regolerà l'incontro e uno **Speaker** che lo commenterà.

Inoltre ogni **Partita** potrà avere degli Spettatori previo acquisto di un **Biglietto**. L'entità **Biglietto** è un template generale che associa ad una **Partita** un costo; lo **Spettatore** invece comprerà un biglietto effettivamente acquistabile, ovvero **AcquistoBiglietto**, che associa un **Biglietto** ad un unico **Spettatore**.

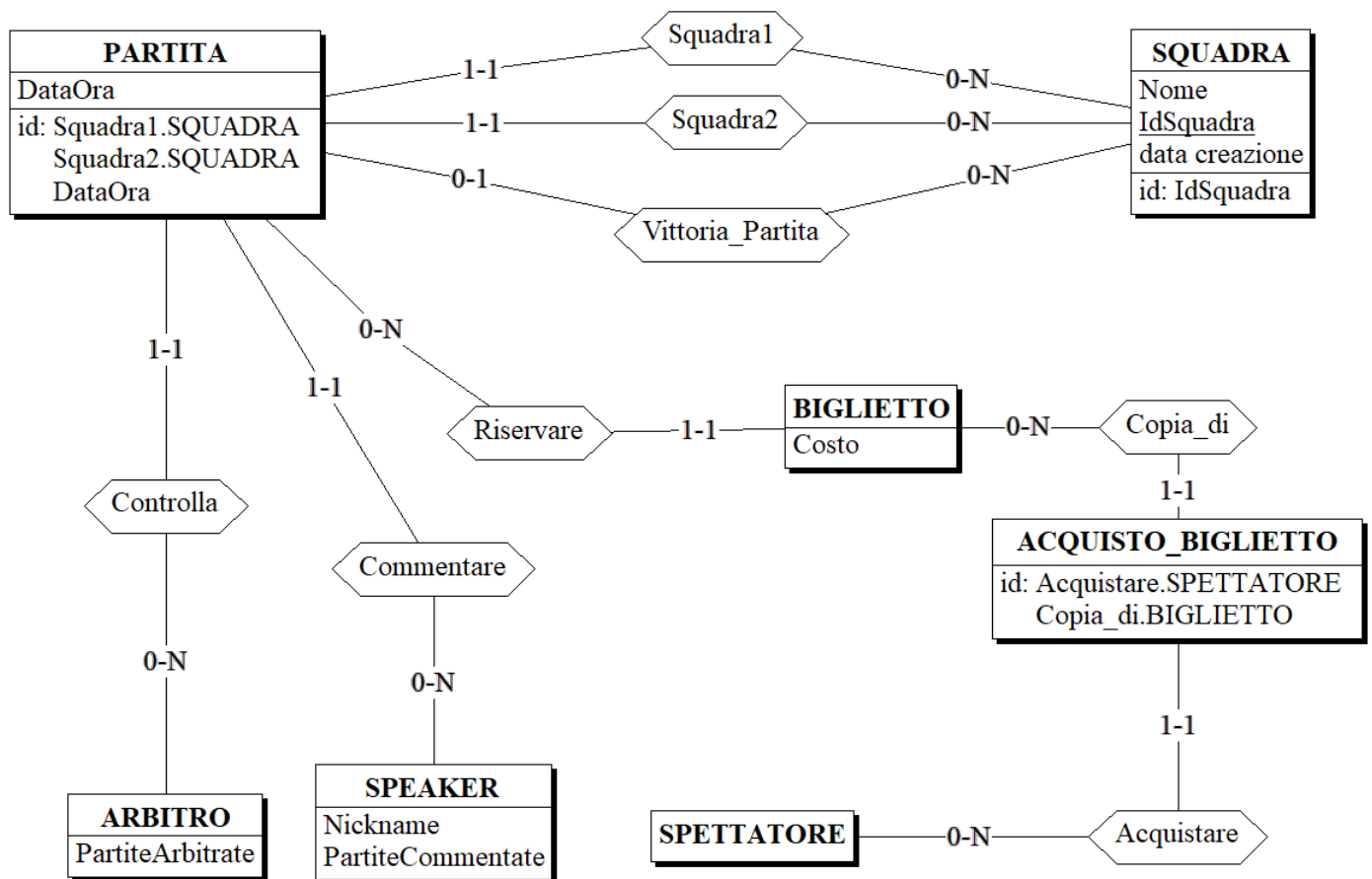


Figura 3.3: Schema ER che espone le principali caratteristiche delle Partite

3.4 Anteprima sviluppo dei "Tornei"

In questa sezione verrà modellato l'aspetto di **Torneo**. L'entità **Torneo** è il fulcro di tutto il nostro sistema informativo. Tutte le altre entità si collegano al Torneo in modo diretto o indiretto.

Ogni **Torneo** è identificato da un numero progressivo e si vuole memorizzare: Data di inizio, Data di fine e il numero massimo di squadre che si possono iscrivere al torneo; vogliamo anche sapere su che **Videogioco** si baserà il Torneo (l'interno **Torneo** si baserà interamente su un unico Videogioco).

Ogni **Torneo** si svolge in una **Arena** situata in una città e può prevedere il finanziamento da parte di una **Sponsor**.

Ad ogni **Torneo** possono iscriversi più **Squadre** e si vuole memorizzare in particolare quale tra le due vincerà.

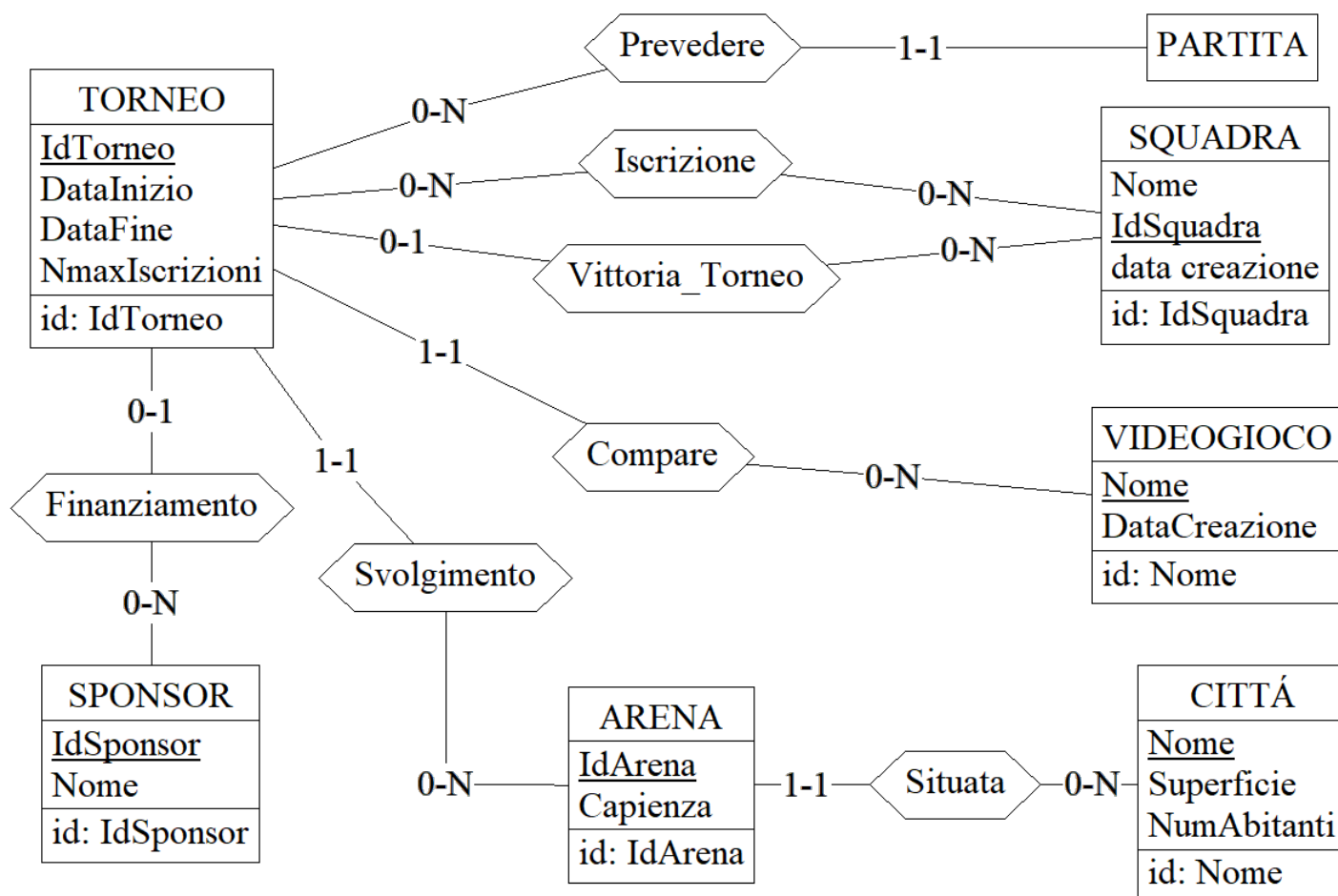


Figura 3.4: Schema ER che espone le principali caratteristiche dei Tornei

3.5 Schema Generale

Di seguito verrà riportato lo schema concettuale generale, contenente tutte le entità e associazioni prima citate nelle varie sezioni superiori con l'aggiunta di entità secondarie di minor importanza.

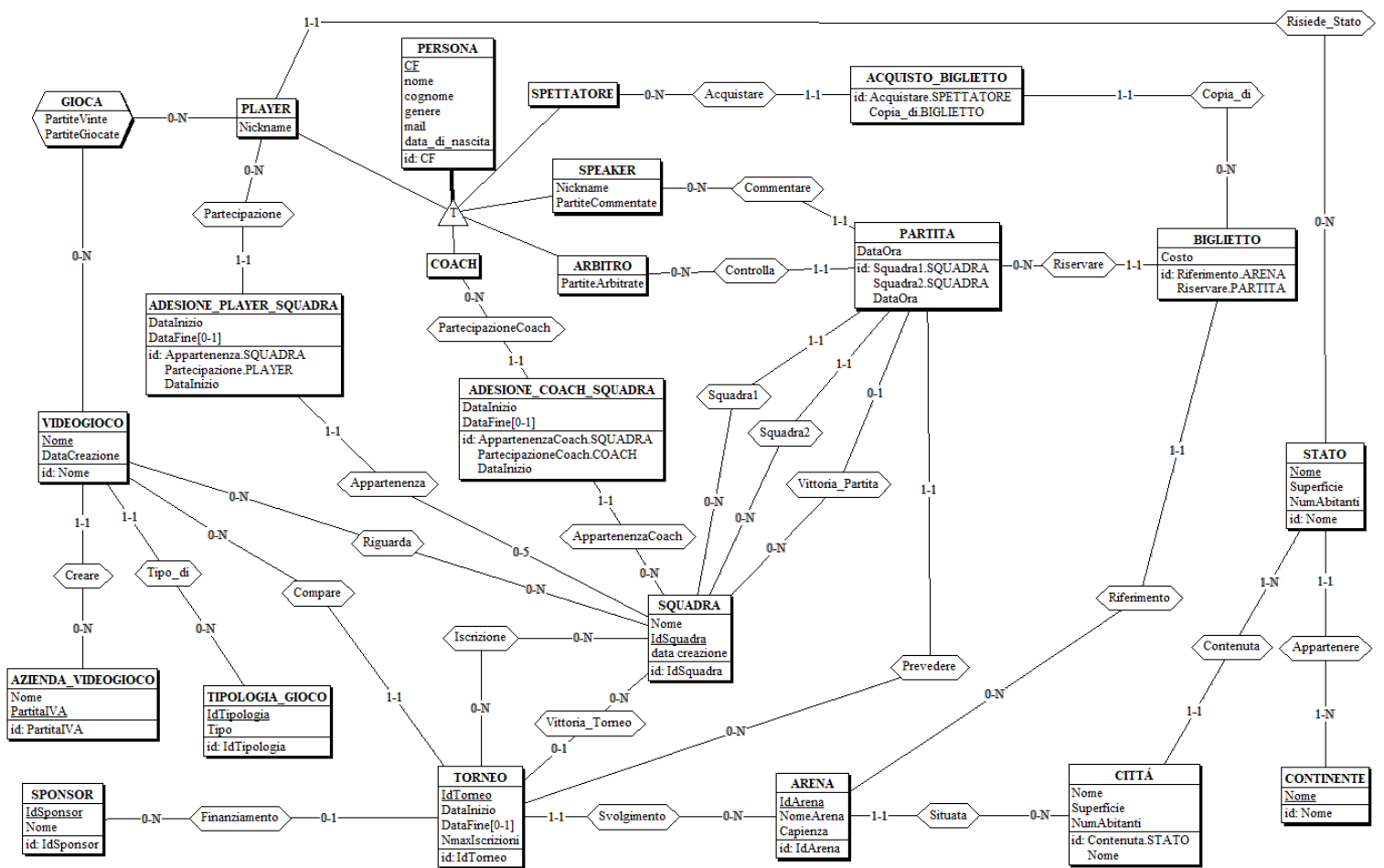


Figura 3.5: Schema ER che espone lo schema concettuale finale

Capitolo 4

Progettazione Logica

4.1 Stima del volume dei dati

Soggetto	Tipo	Volume	Soggetto	Tipo	Volume
Player	E	500.000	Speaker	E	1.000
Videogiochi	E	10	Arbitro	E	1.000
Tipologia Videogioco	E	5	Spettatori	E	200.000.000
Gioca	A	1.000.000	Acquisto Biglietto	E	200.000.000
Azienda Videogioco	E	5	Sponsor	E	35
Squadra Riguarda Videogioco	A	250.000			
Squadra	E	100.000	Tornei	E	10.000
Coach	E	50.000	Partite	E	450.000
Adesione Player Squadra	E	600.000	Previste	A	450.000
Adesione Coach Squadra	E	75.000	Iscrizioni Torneo	A	100.000
			Continente	E	6
			Stati	E	30
			Città	E	60
			Arene	E	70

4.2 Descrizione delle operazioni principali e stima della loro frequenza

Le operazioni da effettuare sono quelle già elencate nella fase di analisi. Segue una tabella riportante la loro descrizione e relativa frequenza:

Codice	Descrizione Operazione	Frequenza
1	Aggiunta di un nuovo Player	50 al giorno
2	Aggiunta Videogioco giocato da un Player	400 a settimana
3	Aggiunta di un nuovo Spettatore	30.000 a settimana
4	Creazione di una Squadra	100 al mese
5	Aggiunta di un Player ad una Squadra	1.500 al mese
6	Creazione di un Torneo	1 a settimana
7	Iscrizione di una Squadra ad un Torneo	10 a settimana
8	Creazione di nuove Partite in un Torneo	45 a settimana
9	Acquisto di un nuovo Biglietto	30.000 a settimana

4.3 Schemi di navigazione e tabelle degli accessi

Sono riportate in seguito le tabelle degli accessi delle operazioni sopra riportate; inoltre, ove non risulti banale, sono stati inseriti i relativi schemi di navigazione. Al fine del calcolo degli costi, si considerano di peso doppio gli accessi in scrittura rispetto a quelli in lettura.

(1) Aggiunta di un nuovo Player

Concetto	Costrutto	Accessi	Tipo
Player	E	1	S
Totale: 1S * 50 al giorno → 100 al giorno			

(2) Aggiunta di un videogioco giocato da un Player

Concetto	Costrutto	Accessi	Tipo
Gioca	R	1	S
Totale: $1S * 400$ a settimana $\rightarrow 800$ a settimana			

(3) Aggiunta di un nuovo Spettatore

Concetto	Costrutto	Accessi	Tipo
Spettatore	E	1	S
Totale: $1S * 30.000$ a settimana $\rightarrow 60.000$ a settimana			

(4) Creazione di una Squadra

Chi crea la squadra di solito è un player quindi quando crea una squadra automaticamente si iscrive ad essa

Concetto	Costrutto	Accessi	Tipo
Squadra	E	1	S
Adesione Player	A	1	S
Totale: $(1S + 1S) * 100$ al mese $\rightarrow 400$ al mese			

(5) Aggiunta di un Player ad una Squadra

controllo la squadra se già piena e ce lo caccio dentro

Concetto	Costrutto	Accessi	Tipo
Squadra	E	1	L
Adesione Player	A	1	S
Totale: $(1L + 2S) * 150.000$ al giorno $\rightarrow 450.000$ al giorno			

(6) Creazione di un Torneo

Concetto	Costrutto	Accessi	Tipo
Torneo	E	1	S
Partite Previste	A	45	S
Iscrizione Squadra	A	10	S
Totale: $(1S + 45S + 10S) \rightarrow 72$ alla settimana			

(7) Iscrizione di una Squadra ad un Torneo

Concetto	Costrutto	Accessi	Tipo
Iscrizione Squadra	A	1	S
Totale: $1S * 8$ a settimana $\rightarrow 16$ a settimana			

(8) Creazione di nuove Partite in un Torneo

Concetto	Costrutto	Accessi	Tipo
Partita	E	45	S
Partita Prevista	A	45	S
Totale: $(45S + 45S) \rightarrow 180$ a settimana			

(9) Acquisto di un nuovo Biglietto

devo leggere il costo

Concetto	Costrutto	Accessi	Tipo
Acquistare	A	1	S
Biglietto	E	1	L
Totale: $1L + 2S \rightarrow 150.000$			

4.4 Raffinamento dello schema

Eliminazione Gerarchie

Eliminazione delle gerarchie Per l'eliminazione della gerarchia persona si è scelto di adottare l'approccio del collasso verso il basso, replicando così gli

attributi della Persona nelle seguenti entità: Player, Coach, Arbitro, Spettatore. Si è adottata questa strategia in quanto si deve interagire con i clienti molto più spesso che con gli istruttori, e non si ha la necessità che l'identificatore per tali entità sia globalmente univoco.

Scelta delle Chiavi

Sin dall'inizio abbiamo scelto accuratamente tutte le chiavi per ogni entità; queste sono evidenziate senza ambiguità nello schema E-R.

4.5 Analisi delle ridondanze

Senza Ridondanza

senza la ridondanza sul numero di biglietti venduti, leggo il costo del biglietto; in media ho 500 letture su acquisto-biglietto per vedere quanti sono stati venduti; ho una lettura su arena per verificarne la capienza.

Concetto	Costrutto	Accessi	Tipo
Acquistare	A	1	S
Biglietto	E	1	L
AcquistoBiglietto	E	500	L
Arena	E	1	L

Totale: $(502L + 2S) * 30.000 \rightarrow 15.180.000$ ogni settimana

Con la Ridondanza

Concetto	Costrutto	Accessi	Tipo
Acquistare	A	1	S
Biglietto	E	1	L

Totale: $1L + 2S \rightarrow 150.000$ ogni settimana

4.6 Traduzione di entità e associazioni in relazioni

ACQUISTO_BIGLIETTO((IdArena, IdSquadra1, IdSquadra2, DataOra) :
BIGLIETTO, CF_Spettatore)

ADESIONE_COACH_SQUADRA(IdSquadra: SQUADRA, CF_Coach: COACH,
DataInizio, DataFine*)

ADESIONE_PLAYER_SQUADRA(IdSquadra: SQUADRA, CF_Player: PLAYER,
DataInizio, DataFine*)

ARBITRO(CF, nome, cognome, genere, mail, data_di_nascita, PartiteArbi-
trate)

ARENA(IdArena, NomeArena, Capienza, (NomeStato, NomeCitta): CITTÀ)

AZIENDA_VIDEOGIOCO(nome, partitaIVA)

BIGLIETTO(IdArena : ARENA, (IdSquadra1, IdSquadra2, DataOra) : PAR-
TITA, Costo)

CITTÀ(NomeStato: STATO, Nome, Superficie, NumAbitanti)

COACH(CF, nome, cognome, genere, mail, data_di_nascita)

CONTINENTE(Nome)

GIOCA(NomeVideogioco : VIDEOGIOCO, CF_Player : PLAYER, Partite-
Vinte, PartiteGiocate)

ISCRIZIONE(IdTorneo : TORNEO, IdSquadra : SQUADRA)

PARTITA((IdSquadra2, IdSquadra1) : SQUADRA, DataOra, CF_Arbitro :
ARBITRO, CF_Speaker : SPEAKER, IdSquadraVincitrice* : SQUADRA,
IdTorneo : TORNEO)

PLAYER(CF, nome, cognome, genere, mail, data_di_nascita, Nickname, No-
meStato: STATO)

RIGUARDA(IdSquadra : SQUADRA, NomeVideogioco : VIDEOGIOCO)

SPEAKER(CF, nome, cognome, genere, mail, data_di_nascita, Nickname, PartiteCommentate)

SPETTATORE(CF, nome, cognome, genere, mail, data_di_nascita)

SPONSOR(IdSponsor, Nome)

SQUADRA(nome, IdSquadra, data_creazione)

STATO(Nome, Superficie, NumAbitanti, NomeContinente : CONTINENTE)

TIPOLOGIA_GIOCO(IdTipologia, Tipo)

TORNEO(IdTorneo, DataInizio, DataFine*, NMaxIscrizioni, IdSponsor* : SPONSOR, NomeVideoGioco : VIDEOGIOCO, IdArena : ARENA, IdSquadraVincitrice* : SQUADRA)

VIDEOGIOCO(Nome, DataCreazione, TipologiaGioco : TIPOLOGIA_GIOCO, PartitaIVAAzienda : AZIENDA_VIDEOGIOCO)

4.7 Schema relazionale finale

Capitolo 5

Progettazione Fisica

5.1 Traduzione in SQL

(1) Aggiunta di un nuovo Player

`INSERT INTO` player (CF, nome, cognomi, genere, mail, data_di_nascita, Nickname, Nome_Stato) `VALUES` (?, ?, ?, ?, ?, ?, ?, ?);

(2) Aggiunta di un videogioco giocato da un Player

`INSERT INTO` gioca(NomeVideogioco, CF_Player, PartiteVinte, PartiteGiocate) `VALUES` (?, ?, ?, ?);

(3) Aggiunta di un nuovo Spettatore

`INSERT INTO` spettatore(CF, nome, cognomi, genere, mail, data_di_nascita) `VALUES` (?, ?, ?, ?, ?, ?);

(4) Creazione di una Squadra

`INSERT INTO` squadra(Nome, IdSquadra, data_creazione) `VALUES` (?, IdSquadra, ?);

(5) Aggiunta di un Player ad una Squadra

`INSERT INTO` adesione_player_squadra(IdSquadra, CF_Player, DataInizio, DataFine) `VALUES` (?, ?, ?, null);

(6) Creazione di un Torneo

`INSERT INTO` torneo(IdTorneo, DataInizio, DataFine, NmaxIscrizioni, Id-Sponsor, NomeVideogioco,IdArena,IdSquadraVincitrice) `VALUES` (IdTorneo,?,?,?,?);

(7) Iscrizione di una Squadra ad un Torneo

`INSERT INTO` iscrizione(IdTorneo, IdSquadra) `VALUES` (?,?);

(8) Creazione di nuove Partite in un Torneo

`INSERT INTO` partita(IdSquadra1, IdSquadra2, DataOra, CF_Arbitro,CF_Speaker, IdSquadraVincitrice, IdTorneo) `VALUES` (?,?,?,?,?);

(9) Acquisto di un nuovo Biglietto

`INSERT INTO` acquisto_biglietto(IdArena, IdSquadra1, IdSquadra2, DataO-ra, CF_Spettatore) `VALUES` (?,?,?,?,?);

Capitolo 6

Progettazione dell'Applicazione

6.1 Descrizione della scelta del linguaggio e del DBMS

6.2 Descrizione dell'architettura

6.3 Interfaccia Utente

6.3.1 Amministratore Torneo

6.3.2 Giocatore