



gamingsport

De Laurentis Arianna Pia 2008077

Fabio Pantaleo 2009124

1 ABSTRACT

Si vuole gestire l'evento **internazionale GamingSport** che prevede la partecipazione di team professionali i quali si affronteranno in diversi giochi.

Al **GamingSport**, si può partecipare all'evento come giocatore o come spettatore, registrandosi tramite la propria email.

Ogni giocatore può far parte di un **solo** team durante tutto l'evento. Un team sarà formato da 3 giocatori, che avranno vari ruoli ("CapoSquadra", "Supporter", "Berserker"); se il team non è completo non partecipa all'evento. Inoltre ogni team sarà caratterizzato da una propria icona originale.

Ogni team potrà partecipare alle varie partite dei giochi scelti per il GamingSport. Ogni partita si terrà all'interno di un'arena, dove quest'ultima può ospitare **più partite** durante la stessa giornata.

Le varie squadre si sfideranno tramite delle partite ai giochi che sono stati scelti per il torneo. I giochi vengono prodotti da una game company, che avrà sede in un indirizzo specifico. Ai team potrà essere assegnato un premio qualora lo vincessero. L'assegnazione avverrà in un secondo momento dalla fine della partita. Il valore del premio è pari al numero di biglietti che sono stati acquistati dagli spettatori per poter assistere a quella partita specifica.

Lo spettatore per poter entrare e guardare le partite del GamingSport dovrà acquistare un biglietto che però varrà **solo** per il gioco corrispondente alla partita, è possibile che uno spettatore possa acquistare **più biglietti** per la stessa partita e nel corso della stessa giornata.

Del biglietto è infatti fondamentale sapere il prezzo (per il calcolo del premio di ogni partita) che sarà di un valore prefissato per tutte le partite e anche il codice della partita con cui può essere utilizzato.

Ogni arena è caratterizzata da **più stand**, ognuno dei quali ha scopi diversi, ci saranno degli stand dedicati al cibo, altri saranno dedicati alla vendita di gadget di varie tipologie e altri stand verranno utilizzati per svolgere le partite.

2 ANALISI DEI REQUISITI

2.1 Descrizione testuale

Nel nostro progetto avremo bisogno di diverse entità in modo da avere risposte dettagliate ed esaustive ad ogni richiesta.

Nella base di dati sono presenti i dati relativi ai **Partecipanti** dell'evento, che comprendono:

- Nome
- Cognome

- Email
- Sesso
- Età

Indirizzo, che rappresenta l'indirizzo di ogni partecipante oppure l'indirizzo della game company, e conterrà al suo interno:

- Nazione
- Città
- Via
- Numero Civico

I partecipanti si divideranno in due tipologie: **Giocatore** e **Spettatore**.

Ogni **Spettatore** dovrà acquistare un biglietto per ogni partita alla quale vuole assistere e avrà diritto ad uno sconto del 20% nel momento in cui questi ha un'età inferiore o uguale di 12 anni.

Ogni **Biglietto** sarà unico, e potrà essere utilizzato solo nella partita dal codice corrispondente (non potrà essere utilizzato per altre partite durante la giornata):

- Id
- Prezzo
- Sconto

Ogni **Giocatore**, che assumerà un ruolo unico all'interno del proprio team, prevede informazioni quali:

- Nickname
- Ruolo

Ogni giocatore farà parte di un **Team**, che per poter partecipare all'evento dovrà essere composto da 3 giocatori. Durante il corso dell'evento un team può partecipare a più partite.

Il **team** avrà:

- Nome
- Sponsor
- Icona_team

Ogni gioco avrà diverse sessioni di gioco, cioè varie **partite**, durante le quali si possono sfidare più team e alle quali possono assistere più spettatori.

Ogni **Partita** comprende:

- Id
- Numero spettatori

Ogni **Gioco** che è stato scelto per GamingSport potrà essere giocato dai vari team presenti e comprenderà:

- Nome
- Anno
- Genere

- Sviluppatore
- Rating

Ogni gioco viene prodotto da una **Game Company**, ognuna delle quali avrà sede ad un indirizzo diverso e sarà gestita da un CEO, l'entità sarà caratterizzata da:

- Identificativo
- nome
- CEO

Dove con anno si intende l'anno di rilascio del gioco.

Le partite si svolgono all'interno di arene. Ogni arena è adibita a poter ospitare più partite.

L'entità **Arena** sarà caratterizzata da:

- Codice Arena
- Area
- Altezza
- Nome arena

Un'arena è caratterizzata da **Stand**, la cui entità è caratterizzata da:

- Codice stand
- Nome stand
- posizione
- Tipo stand

dove con posizione si intende la locazione in termini di nord, sud, est, ovest all'interno dell'arena in cui è posizionato, e con tipo si indica lo scopo per cui è usato lo stand (cibo, gadget, gioco).

Al termine di ogni partita si possono ottenere dei premi, il cui valore corrisponde al prezzo di tutti i biglietti che sono stati venduti per quel match.

Con premio si vuole indicare quali premi il giocatore ha vinto durante il corso degli anni;

Premio che contiene il codice identificativo del premio e una piccola descrizione riguardo ad esso:

- Id premio
- Descrizione

Relativamente ad un premio si possono distinguere due concetti diversi:

Premio Ottenuto, che indica i premi che sono già stati vinti dal team:

- Codice premio

Premio Partita, con cui si intende il premio non ancora vinto per cui si sta giocando una partita e di cui sarà di interesse il valore in denaro:

- valore

2.2 GLOSSARIO DEI TERMINI

Termine	Descrizione	Collegamenti
Partecipante	Persona che partecipa all'evento GamingSport	Indirizzo,Spettatore, Giocatore
Indirizzo	Luogo di abitazione dei partecipanti, o indirizzo della game company	Partecipante, game company
Spettatore	Partecipante che guarda le partite dei vari giochi	Entità figlia di Partecipante, Biglietto,Partita
Adulto	Spettatore che acquisterà un biglietto non ridotto	Entità figlia di Spettatore
Bambino	Spettatore che avrà diritto ad uno sconto nell'acquisto del biglietto	Entità figlia di Spettatore
Biglietto	Biglietto acquistato dallo spettatore per guardare le partite	Partita, Spettatore
Giocatore	Partecipante che fa parte di un team e gioca	Entità figlia di Partecipante, Team
Game Company	Rappresenta la casa produttrice del gioco	Gioco, indirizzo
Premio	Sono i premi che hanno vinto i team nelle partite	Team, Partita
Team	È il gruppo di giocatori che partecipa alle partite	Giocatore,Partita,Premio
Partita	Rappresenta la singola partita di un gioco	Team ,Arena, Gioco, Biglietto, Premio_Partita
Gioco	Rappresenta un gioco che sarà nell'evento	Partita, Game_Company
Arena	Luogo in cui si svolge una partita	Partita, Stand
Stand	Indica la suddivisione di un'arena	Arena

2.3 OPERAZIONE TIPICHE

Registrazioni di un utente all'evento	1000 al giorno
Ricerca dei giochi che ci sono nell'evento	500 durante l'evento
Calcolo del valore del premio di una partita	100 al giorno
Acquisto di un biglietto	500 al giorno
Controllo valore premio	1000 al giorno
Visualizza biglietti comprati per partita	50 al giorno
Visualizza Costo Biglietto	5000 al giorno

3 PROGETTAZIONE CONCETTUALE

3.1 TABELLA ENTITA'

PARTECIPANTE: Persona che partecipa al GamingSport		
<u>Email</u>	varchar (30) primary key	Email usata per partecipare all'evento
Nome	varchar (30)	Nome del partecipante
Cognome	varchar (30)	Cognome del partecipante
Età	int	Informazione necessaria per stabilire, nel caso in cui fosse spettatore, il diritto ad uno sconto
Sesso	char (1)	valore che può essere F o M

INDIRIZZO: Luogo in cui abita il partecipante		
<u>Num Civico</u>	int primary key	
<u>Città</u>	varchar (30) primary key	
<u>Via</u>	varchar (30) primary key	
Nazione	varchar (30)	La nazione del partecipante

Partecipante si specializza in due entità: **Spettatore** e **Giocatore**

SPETTATORE: Partecipante che guarda le partite

ADULTO: Spettatore con età > 12 anni

BAMBINO: Spettatore con età <= 12 anni

GIOCATORE: Partecipante che fa parte di un team e gioca

Nickname	varchar (16) UNIQUE	Il nickname del giocatore
Ruolo	varchar (16)	Attributo che comprende 3 valori {CapoSquadra, Supporter, Berserker}

BIGLIETTO: Biglietto acquistato per vedere le partite

<u>id</u>	int primary key	identificativo del biglietto
Prezzo	double	Può avere due valori:10 euro o 20 euro.
Sconto	bool	se l'età<=12 anni allora si ha un diritto allo sconto

GAME_COMPANY: Casa produttrice di un gioco

<u>Id_Gc</u>	varchar (30) primary key	Codice che identifica la casa produttrice
CEO	varchar (30)	nome del CEO
Nome_Gc	varchar (30)	Nome della Game Company

PREMIO: Premi vinti dal team

<u>Id-Premio</u>	int primary key	Codice del premio. (È una parte della chiave.)
Descrizione	varchar (50)	Descrizione del premio

PREMIO_OTTENUTO: Premio vinto dal team

PREMIO_PARTITA: Premio per cui si sta giocando la partita

Valore	int	valore del premio in denaro
--------	-----	-----------------------------

TEAM: Gruppo di giocatori che giocano assieme (3 giocatori)

<u>Nome Team</u>	varchar (30) primary key	Nome del team
Sponsor	varchar (30)	Lo sponsor affiliato al team
Icona_Team	varchar (30)	Simbolo rappresentativo del team

PARTITA: Singola partita di un gioco

<u>Id-Partita</u>	Int primary key	Codice che identifica la partita
NumSpettatori	int	Il numero di persone che guardano la partita

GIOCO: Gioco che può essere giocato all'evento

<u>Nome</u>	varchar (30) primary key	
Anno	Date	Anno della pubblicazione del gioco
Genere	varchar (16)	Tipologia di gioco
Sviluppatore	varchar (30)	Nome dello sviluppatore del gioco
Rating	int	Rating del gioco in una scala 1 a 5 (dove 1 è il minimo)

ARENA: Luogo dove si svolge una partita

<u>Codice Arena</u>	int primary key	Codice identificativo dell'arena
Dimensione	Int	Ampiezza dell'arena
Nome_arena	varchar (16)	Nome dell'arena
Altezza	int	Altezza dell'arena in metri
Area	int	Area dell'arena in metri quadrati

STAND: Stand in cui si divide l'arena		
<u>Codice_stand</u>	Int primary key	Codice che identifica lo stand
Posizione	varchar (1)	Posizione dello stand all'interno dell'arena
Tipo_stand	varchar (16)	Tipo di stand i cui valori possono essere {cibo, gadget, giocattolo}
Nome_stand	varchar (30)	Nome dato allo stand

3.2 LISTA DELLE RELAZIONI

RELAZIONI	ENTITÀ e CARDINALITÀ	DESCRIZIONE	ATTRIBUTI
Vivere	Indirizzo (1,1) Partecipante (1,1)	Un partecipante vive "in un solo indirizzo. In un indirizzo può vivere una sola persona.	
Vincere	Team(0,N) Premio_Ottenuto (1,1)	Un team può aver vinto 0 o più premi. Un premio è vinto da un unico team	Data_Vincita
Avere sede	Indirizzo (1,1) Game Company (1,1)	Una game Company ha sede in un solo indirizzo, e un solo indirizzo può ospitare una sola Game Company	
Far Parte	Giocatore (1,1) Team (3,3)	Un giocatore fa parte di un solo team e il team deve comprendere necessariamente 3 giocatori	
Acquistare	Spettatore (1,N) Biglietto (1,1)	Uno spettatore può acquistare più biglietti. Il singolo biglietto può essere acquistato da una sola persona	
Usato Per	Biglietto (1,1) Partita (1,N)	Un biglietto può essere usato ad una sola partita, ad una partita possono essere usati più biglietti	
Partecipare	Team (1,N) Partita (1,N)	Un team gioca a più partite e in una partita ci sono più team che giocano, anche uno solo.	

4 PROGETTAZIONE LOGICA

4.1 RISTRUTTURAZIONE

4.1.1 ANALISI DELLE RIDONDANZE

L'attributo "num_spettatori" dell'entità "PARTITA" che rappresenta il totale delle persone che partecipano ad un singolo match, è ridondante in quanto può essere calcolato come somma di tutti i biglietti (attraverso il loro identificativo ID), che sono stati acquistati per poter partecipare a quella specifica partita (anch'essa identificata da un proprio id_partita).

Questo attributo viene utilizzato in un'unica operazione che è quella di calcolo del valore del premio per ogni partita.

In media si acquistano 500 biglietti al giorno, che dovranno essere aggiunti a quelli già acquistati precedentemente.

CONCETTO	TIPO	VOLUME
Premio	E	1000
Partita	E	5000
Biglietto	E	300000

Operazione 1: Calcolo valore del premio

Operazione 2: Visualizza Valore Premio

CON RIDONDANZA

Una partita in media ha $30000/5000 = 60$ biglietti = 60 spettatori

Operazione 1:

CONCETTO	TIPO	ACCESSI	TIPO	
Premio	E	1	S	X50
Partita	E	1	S	X500
Partita	E	60	L	X50
Biglietto	E	1	L	X5000

Operazione 2:

CONCETTO	TIPO	ACCESSI	TIPO	
Premio	E	1	L	X1000

Costo = $50*2 + 500*2 + 50*60 + 5000 + 1000 = 7400$

SENZA RIDONDANZA

Operazione 1:

CONCETTO	TIPO	ACCESSI	TIPO	
Premio	E	1	S	X50
Partita	E	1	L	X50
Biglietto	E	60	L	X5000

Operazione 2:

CONCETTO	TIPO	ACCESSI	TIPO	
Premio	E	1	L	X1000

Costo = $50 \cdot 2 + 60 \cdot 5000 + 50 + 1000 = 30150$

In conclusione l'attributo num_spettatori presente in partita ci semplifica il costo computazionale richiesto dalle operazioni di Calcolo Valore Premio e Visualizza Valore Premio. Quindi la ridondanza viene lasciata e il valore del num_spettatori verrà aggiornato a ogni acquisto di un biglietto per la partita.

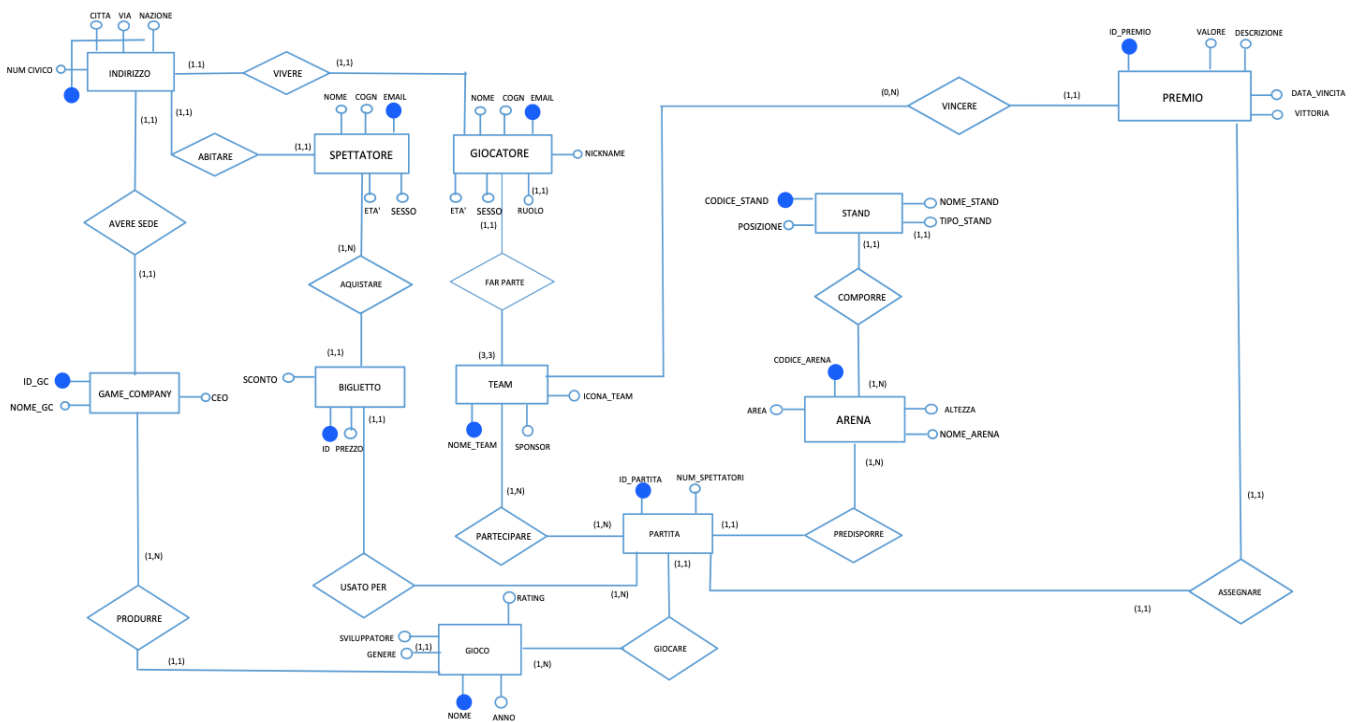
4.1.2 ELIMINAZIONE DELLE GENERALIZZAZIONI

GENERALIZZAZIONE	RISOLUZIONE
PARTECIPANTE <- GIOCATORE, SPETTATORE	Avendo una generalizzazione totale e essendo che gli accessi alle entità figlie non sono contestuali, si preferisce eliminare l'entità Padre "PARTECIPANTE", considerando solo "giocatore" e "spettatore" ora indipendenti. Entrambe ereditano gli attributi email, nome e cognome e verranno connesse direttamente alle relazioni "vivere".
PREMIO <- PREMIO_OTTENUTO, PREMIO_PARTITA	Essendo che le due entità figlie rappresentano lo stesso concetto, possono essere accorpate entrambe all'interno di PREMIO. Il fatto che un premio possa essere già stato ottenuto da un team in partite passate, sarà identificato da un valore di "data_vincita" (che potrà essere NULL se il premio non è stato ancora vinto) e anche da un nuovo valore dell'entità "PREMIO": <ul style="list-style-type: none">• Vittoria: bool, NOT NULL Successivamente l'entità risultante verrà connessa solamente all'entità "PARTITA".
SPETTATORE<-BAMBINO, ADULTO	In questo caso, le due entità figlie verranno incorporate all'interno dell'entità padre in quanto il diritto allo sconto può essere calcolato controllando l'attributo età di spettatore.

4.1.3 SCELTA DEGLI IDENTIFICATORI PRIMARI

Per quanto riguarda la scelta degli identificatori primari, dopo aver eliminato le generalizzazioni questi rimangono invariati, ovvero l'identificatore sia per l'entità giocatore che per spettatore ricade sull'attributo email, e per premio rimane sempre Id_Premio.

SCHEMA E-R RISTRUTTURATO



4.2 Creazione delle tabelle e vincoli di integrità referenziale

Creiamo le tabelle del nostro E-R ristrutturato con le relative chiavi esterne. Gli attributi con # davanti potranno essere NULL.

Spettatore (Email, Nome, Cognome, Sesso, Età, Luogo->Indirizzo.Città, Luogo->Indirizzo.Via, Luogo->Indirizzo.Num_Civico)

Giocatore (Email, Nome, Cognome, Sesso, Età, Nickname, Ruolo, Luogo->Indirizzo.Città, Luogo->Indirizzo.Via, Luogo->Indirizzo.Num_Civico, Gruppo->Team.Nome_Team)

Indirizzo (Città, Via, Numero Civico, Nazione)

Biglietto (Id, Prezzo, Sconto, Proprietario->Spettatore.Email, Sessione->Partita.Id_Partita)

Team (Nome_Team, Icona_Team, #Sponsor)

Partita (Id_Partita, Num_Spettatori, Game->Gioco.Nome_Gioco, Palazzetto->Arena.Codice_Arena)

Partecipare (Gruppo->Team.Nome_Team, Sessione->Partita.Id_Partita)

Gioco (Nome_Gioco, Sviluppatore, Anno, Genere, Rating, Casa_Produttrice->GameCompany.Id_Cg)

GameCompany (Id_Gc, Nome_Gc, CEO, Luogo->Indirizzo.Città, Luogo->Indirizzo.Via, Luogo->Indirizzo.Num_Civico)

Arena (Codice_Arena, Altezza, Area, Nome_Arena)

Stand (Codice_Stand, Posizione, Tipo_Stand, Nome_Stand, Palazzetto->Arena.Codice_Arena)

Premio (Id_Premio, Valore, #Data_Vincita, Vittoria, Descrizione, #Gruppo -> Team.Nome_Team, Sessione->Partita.Id_Partita)

5 QUERY E INDICI

5.1 QUERY

1) Mostrare quali Team, restituendo nome e icona e partite totali, ad aver giocato più partite durante il torneo.

```
drop view if exists PartiteTeam;
CREATE VIEW PartiteTeam(NomeTeam, Icona, npartite) AS
SELECT Nome_Team, Icona_Team, COUNT(*)
FROM Team AS T, Partecipare AS K
WHERE T.Nome_Team = K.Gruppo
GROUP BY Nome_Team;

SELECT NomeTeam, Icona, npartite
FROM PartiteTeam
WHERE npartite = (SELECT MAX(npartite) FROM PartiteTeam)
```

nometeam	icona	npartite
NiP	Pistola	28
Virtus	Orso	28
BMF	Croce	28

2) Calcolare la quantità di biglietti ridotti, quindi destinati ad un target bambino, che sono stati venduti per poter partecipare come spettatore alla partita con codice 5.

```
SELECT sessione, COUNT(*)
FROM Biglietto AS B, Spettatore AS S
WHERE S.email=B.Possessore AND Sessione= 15 AND S.Età<=12
GROUP BY Sessione
```

sessione	count
15	3

3) Mostrare il numero stand dello stesso tipo che sono presenti nell'arena 9.

```
SELECT Tipo, COUNT(*)
FROM Stand AS S, Arena AS A
WHERE S.Palazzetto=A.Codice_Arena
GROUP BY Tipo, A.Codice_Arena
HAVING A.Codice_Arena= 9
```

tipo	count
Cucina	1
Gadget	5
Giocattolo	2

4) Si chiede di visualizzare tutte le arene (codice e nome relativo) dove sono state svolte le partite del gioco best rated (quelli con la miglior valutazione).

```
drop view if exists bestRated_Match;
CREATE VIEW bestRated_Match (Codice_partita) AS
SELECT P.Id_Partita
FROM Gioco AS G, Partita AS P
WHERE P.Game=G.Nome_Gioco
GROUP BY Id_Partita, G.Rating
HAVING G.Rating=(SELECT DISTINCT MAX(G.Rating)
                  FROM Gioco);
```

```
SELECT A.Codice_Area, A.Nome_Arena
FROM bestRated_Match AS B, Arena AS A
WHERE B.Codice_Partita=A.Codice_Arena
```

codice_arena	nome_arena
10	Voltsillam
9	Flexidy
5	Treeflex
8	Flexidy
1	Gembucket
4	Aerified
7	Wrapsafe
2	Zontrax
3	Hatity
6	Stronghold

5) Mostrare il nome del gioco dove è stato vinto il premio di maggior valore con la relativa partita e il team vincitore del premio

```
drop view if exists higher_price CASCADE;
CREATE VIEW higher_price (NomeTeam, valorepremio) AS
SELECT P.Gruppo, MAX(P.Valore)
FROM Team AS T, Premio AS P
WHERE P.Gruppo = T.Nome_Team AND Vittoria= TRUE
GROUP BY P.Gruppo;
```

```
SELECT Id_partita, P.Valore, H.NomeTeam, A.Game
FROM higher_price AS H, Partita AS A, Premio AS P
WHERE A.Id_Partita=P.Sessione AND P.Gruppo=H.NomeTeam AND
      P.Valore= (SELECT MAX (valorepremio) FROM higher_price);
```

id_partita	valore	nometeam	game
14	80	Virtus	Twolife

6) Per un sondaggio relativo agli spettatori si chiede di mostrare quante, tra le persone che partecipano come tali all'International GamingSport, sono di sesso femminile e quanti di sesso maschile con relativa età media.

```
drop view if exists gender_count;  
drop view if exists avg_age;  
CREATE VIEW gender_count (Genere, npartecipanti) AS  
SELECT Sesso, COUNT(*)  
FROM Spettatore  
GROUP BY Sesso
```

```
CREATE VIEW avg_age (Genere, etàmedia) AS  
SELECT Sesso, AVG(età)::REAL  
FROM Spettatore  
GROUP BY Sesso
```

```
SELECT G.Genere, G.npartecipanti, A.etàmedia  
FROM gender_count AS G, avg_age AS A  
WHERE A.Genere=G.Genere
```

genere	npartecipanti	etàmedia
M	24	41.25
F	16	35.25

5.2 INDICI

5.2.1 Indici per la ricerca

Analisi e Obiettivi

All'interno di un Database che gestisce un evento internazionale come il GamingSport è di importanza fondamentale avere un una ricerca efficiente e veloce dei team che partecipano alle varie partite, in modo da poter controllare come queste squadre procedono nel corso dell'evento e soprattutto se la loro composizione rispetta i canoni per la partecipazione (obbligo di 3 giocatori per team). Una modalità tramite cui implementare questa ricerca è l'uso di parole chiave.

Implementazione

Si procede a creare un indice della tabella TEAM, tabella che può raggiungere dimensioni particolarmente elevate, in quanto contiene tutte le informazioni necessarie per poter individuare un team, i giocatori che ne fanno parte, e le vittorie che sono state conseguite con relativi premi.

```
CREATE INDEX idx_NomeTeam ON TEAM(Nome_Team);
```

6. CODICE

È stato creato un codice in c++ per la connessione al database su sistema operativo **Ubuntu**. Nel file è stato incluso l'header libpq-fe.h per i comandi SQL per la connessione al database e per l'esecuzione delle query. Quindi si sono stabiliti degli attributi quali Host, User, Password, Port e Nome del database, utilizzati nella funzione **connect ()**. Una volta connessi al database si possono scegliere le query di esempio che si trovano disponibili. L'esecuzione del codice verrà eseguita da terminale, quindi si andrà a creare il file .exe attraverso il comando "**g++ Codice.cpp -o Codice -l /usr / include / postgresql -lpq**" e poi si eseguirà il file generato con il comando "**./Codice**".