Marco Calcaterra #748236
Salvatore Mariano Librici #748240
Emanuele Buggin #748676
Davide Buggin #749715

Documentazione sviluppo base di dati

Tabella Users

Per salvare i dati di tutti gli utenti registrati, è necessaria una tabella che chiamiamo “Users”, che ha i seguenti attributi:

- UserID TEXT
- UserName TEXT
- UserPassword TEXT
- UserEmail TEXT
- UserLastUse DATE
- UserPlaylist TEXT[]

Dove “UserID” è la PRIMARY KEY e ogni attributo ha un vincolo di NOT NULL, essendo questi campi tutti obbligatori, ad eccezione di “UserPlaylist”, che può essere NULL nel momento in cui (per esempio) l’utente si è appena registrato e non ha ancora creato nessuna playlist.

Inoltre, l’attributo “UserPlaylist” è un attributo multivalore, dato che non c’è un limite al numero di playlist che un utente può creare.

Tabella Playlists

Anche le playlist che vengono create sono definite attraverso una tabella che chiamiamo “Playlists”, che ha i seguenti attributi:

- PlaylistID TEXT
- CreationDate DATE
- PlaylistName TEXT
- UserID TEXT
- SongList TEXT[]

Dove “PlaylistID” è la PRIMARY KEY e ogni attributo ha un vincolo di NOT NULL, essendo questi campi tutti obbligatori.

L’attributo “SongList” è un attributo multivalore, dal momento che per ogni playlist che viene creata ci sarà una lista di più canzoni.

Inoltre, l’attributo “UserID” è in comune con la tabella “Users”: questo è dato dal fatto che è chiave esterna sulla tabella “Playlist”, infatti, come vedremo successivamente nello Schema Logico, le due tabelle sono associate.

Tabella Songs

Tutto il repository di canzoni disponibili per l'utilizzo dell'applicazione è rappresentato da una tabella che chiamiamo "Songs", che ha i seguenti attributi:

- SongID TEXT
- Author TEXT
- Year BIGINT
- Title TEXT
- ReviewTotal BIGINT
- Amazement BIGINT
- Tenderness BIGINT
- Nostalgia BIGINT
- Calmness BIGINT
- Solemnity BIGINT
- Power BIGINT
- Tension BIGINT
- Sadness BIGINT
- Joy BIGINT

Dove "SongID" è la PRIMARY KEY e ogni attributo ha un vincolo di NOT NULL, essendo tutti questi campi obbligatori.

Tutti gli attributi relativi alle emozioni hanno un valore di default di 0, dato che alcune canzoni è possibile che non siano ancora mai state recensite. Il valore si aggiornerà in automatico nel momento in cui un utente lascerà una recensione e inserirà gli score di ciascuna emozione.

-Nota: È stata presa la decisione di rappresentare ogni singola emozione come attributo semplice invece di utilizzare strutture più articolate (es: attributi composti). Questa scelta progettuale è stata presa per motivi di semplicità strutturale dell'entità e dello schema ER.

Tabella AdminInfo

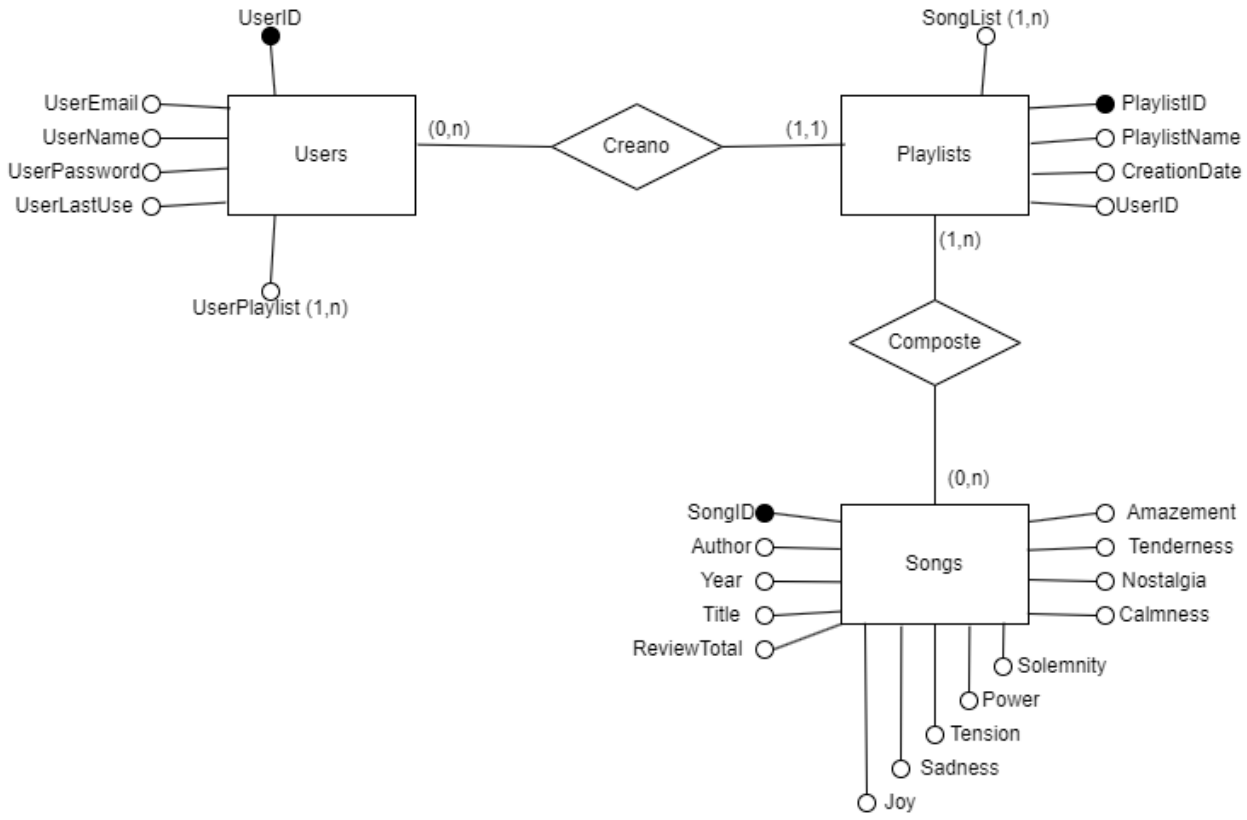
Per salvare i dati di tutti gli admin, è necessaria una tabella che chiamiamo "AdminInfo", che ha i seguenti attributi:

- ID BIGINT
- AdminUserName TEXT
- Password TEXT
- Email TEXT

Dove "ID" è la PRIMARY KEY e ogni attributo ha un vincolo di NOT NULL, essendo tutti questi campi obbligatori.

-Nota: questa tabella non sarà presente nello schema ER, perché essendo quella degli admin ha dei metodi esclusivi, che non c'entrano con quelli standard dell'app.

Schema ER



Scelte progettuali per la costruzione dello schema

Per costruire lo schema ER è stata adottata una strategia mista, partendo da uno “schema scheletro” dove ogni tabella è stata definita come entità.

Successivamente ogni tabella è stata integrata con attributi e identificatori, mentre le associazioni sono state integrate con i vincoli di cardinalità.

Sulla base di questo schema ER, è stato utilizzato PostgreSQL per realizzare il Database contenente le tabelle che l’app Emotional Songs utilizza per la gestione dei dati.

Associazioni e vincoli di cardinalità

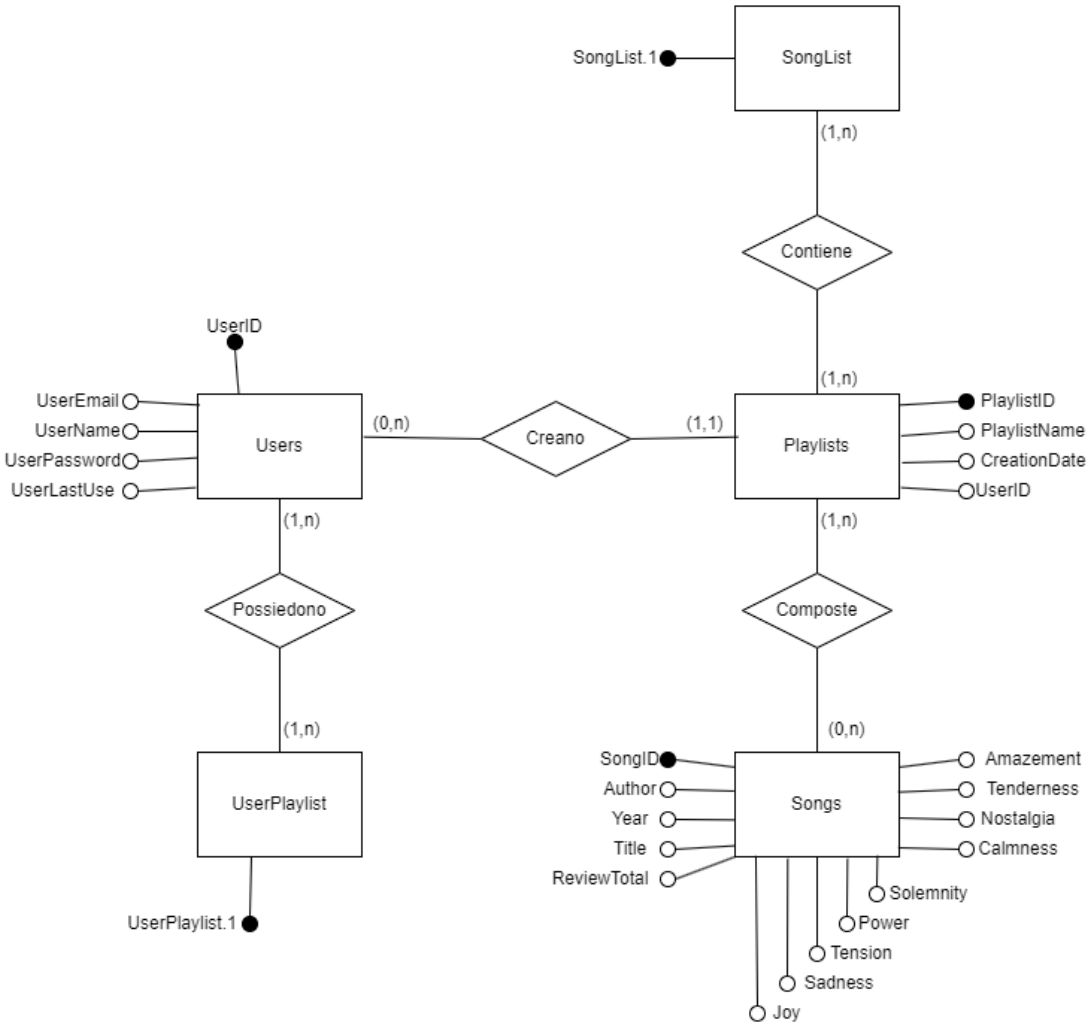
Nello schema sono presenti due associazioni che mettono in comunicazione le tre entità.

1. Creano: “Gli Users CREANO le Playlists”.
   È un’associazione **uno a molti** perché una playlist può essere stata creata da uno e un solo User, e uno User può creare più playlists.
   La cardinalità minima di Users è 0 perché può anche non aver creato nessuna playlist, mentre la cardinalità minima di Playlists è 1 perché se fosse 0 vorrebbe dire che non ha un creatore, cioè che non è stata creata e di conseguenza non potrebbe esistere.

2. Composte: “Le Playlists sono COMPOSTE dalle Songs”.
   È un’associazione **molti a molti** perché una playlist può contenere più canzoni, e una canzone può trovarsi in più playlist.
   La cardinalità minima di Playlists è uno perché non può esistere una playlist vuota (senza canzoni), mentre la cardinalità minima di Songs è 0 perché una canzone può non essere presente in nessuna playlist.

Schema ER ristrutturato

È stata effettuata una ristrutturazione dello schema originale, in quanto sono presenti degli attributi multivalore (che sono strutture complesse), con l'obiettivo di ottenere uno schema logico più chiaro e semplice.



Ristrutturazione schema ER

Sono stati sostituiti gli attributi multivalore con delle associazioni **molti a molti** che mettono in comunicazione le entità di partenza con due nuove entità omonime agli attributi multivalore e autoidentificate tramite un identificatore, che corrisponde al nome dell'unico attributo dell'entità.

Traduzione in schema logico

Sulla base dello schema ER ristrutturato è stato ricavato lo schema logico:

- Definendo prima tutte le entità con tutti i loro attributi e identificatori.
- Definendo tutte le associazioni tramite le chiavi esterne delle entità.

Come si può vedere nello schema logico, le PRIMARY KEY vengono rappresentate tramite una sottolineatura; le chiavi esterne vengono rappresentate con all'apice il nome della loro entità referente.