

Documentazione Database

Indice

1. Analisi del sistema

1.1 Obiettivi e scopo del prodotto

2. Schema ER

2.1 Metodologia di sviluppo dello schema ER

2.2 Schema ER

3. Analisi Entità e Associazioni dello schema relazionale

3.1 Dizionario delle Entità

3.2 Dizionario delle Associazioni

3.3 Dizionario delle Generalizzazioni

3.4 Vincoli di integrità dello schema relazionale

3.5 Motivazioni decisionali della progettazione dello schema concettuale

4. Schema logico

4.1 Schema logico

5. Analisi schema logico

5.1 Fase di ristrutturazione

5.2 Fase di traduzione

5.3 Motivazioni decisionali della traduzione dello schema concettuale

6. Query significative

6.1 Registrazione utente

6.2 Login utente

6.3 getEmozioniFromBrano 1

6.4 getEmozioniFromBrano 2

6.5 Filtra per titolo

6.6 Filtra per autore anno

6.7 getCanzoniForPlaylist

6.8 createPlaylist 1

6.9 createPlaylist 2

6.10 getPlaylist

6.11 getCanzoniFromPlaylist

6.12 inserisciEmozione

Documentazione Database

1. Analisi del sistema

1.1 Obiettivi e scopo del prodotto

Il progetto descritto in questo documento ha come scopo la realizzazione di un sistema di annotazione di emozioni percepite durante l'ascolto di brani musicali. L'utente deve avere la possibilità di registrarsi al sistema, effettuare l'accesso al sistema tramite il quale sarà possibile:

- Creare playlist.**
- Inserire canzoni all'interno delle playlist.**
- Filtrare le canzoni per titolo**
- Filtrare le canzoni per autore e anno**
- Inserire un punteggio relativo alle emozioni.**
- Inserire note opzionali relative alle emozioni.**

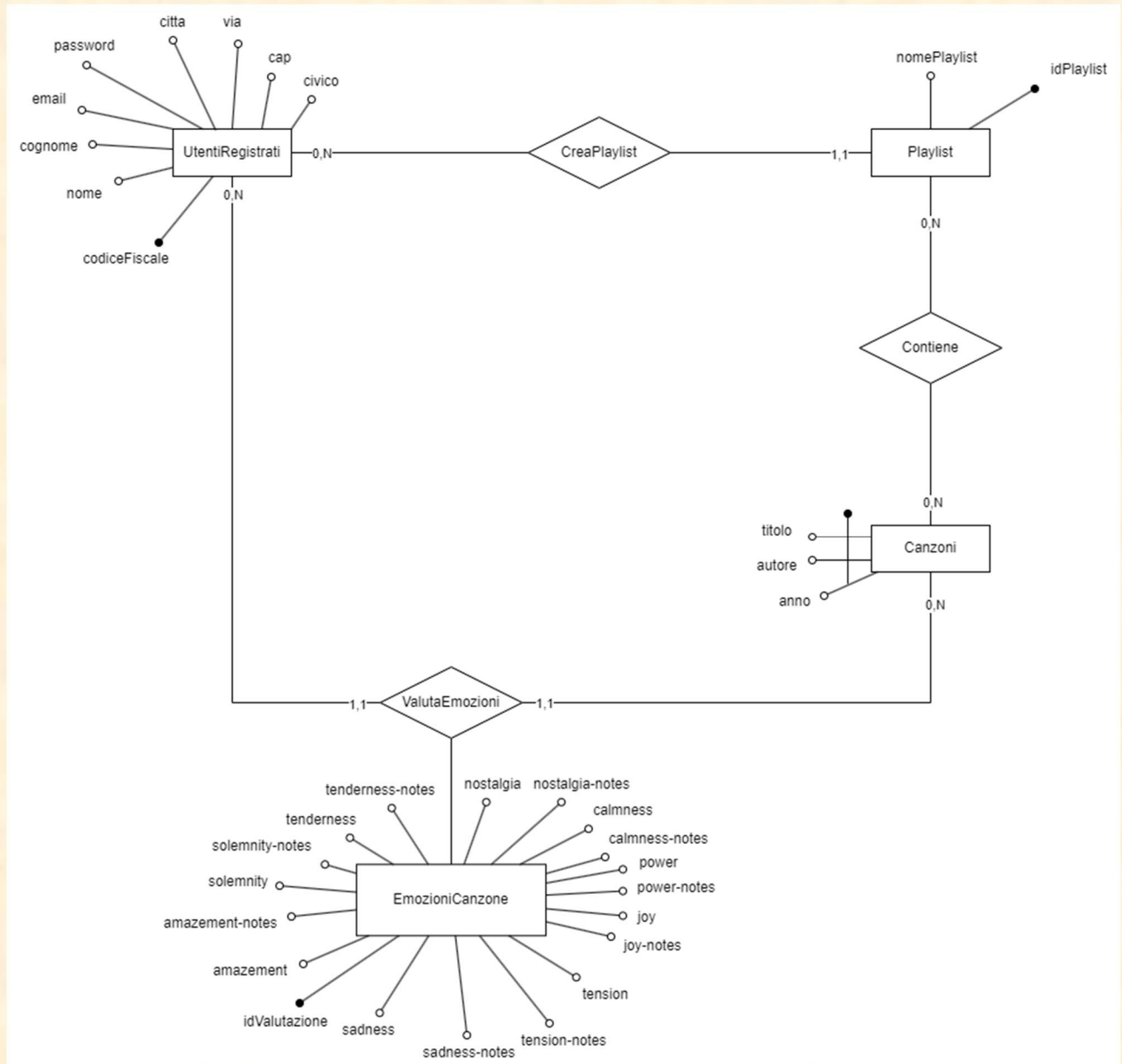
Il sistema dev'essere realizzato in modo tale da poter supportare l'esecuzione concorrente in ambito distribuito.

1. Schema ER

2.1 Metodologia di sviluppo dello schema ER

Lo sviluppo dello schema ER ha seguito un approccio top-down, dovuto soprattutto dal file con le richieste e le specifiche di sistema, il quale aumenta la propria specificità in seguito a letture successive.

2.2



2. Analisi Entità e Associazioni dello schema relazionale

3.1 Dizionario Entità

<u>Nome</u>	<u>Descrizione</u>	<u>Attributi</u>	<u>Identificatori</u>
Canzoni	Canzoni utilizzate per l'ascolto da parte dell'utente.	Titolo, Autore, Anno	Titolo, Autore, Anno
UtentiRegistrati	Utenti che usufriscono dei servizi offerti dal sistema.	Nome, Cognome, CodiceFiscale, Email, Password, cap, citta, civico, via	CodiceFiscale
Playlist	Playlist contenenti le	idPlaylist, NomePlaylist	idPlaylist

	canzoni inserite dall'utente.		
EmozioniCanzoni	Emozioni suscitate dall'utente dopo l'ascolto di un brano specifico.	idValutazione, Amazement, Amazement-Notes, Solemnity, Solemnity-Notes, Tenderness, Tenderness Notes, Nostalgia, Nostalgia-Notes, Calmness, Calmness-Notes, Power, Power- Notes, Joy, Joy-Notes, Tension, Tension-Notes, Sadness, Sadness- Notes.	idValutazione

3.2 Dizionario delle Associazioni

<u>Nome</u>	<u>Descrizione</u>	<u>Attributi</u>	<u>Entità collegate</u>
CreaPlaylist	Rappresenta la creazione della playlist da parte dell'utente.		UtentiRegistrati, Playlist
Contiene	Rappresenta l'inserimento delle canzoni nella playlist da parte dell'utente.		Playlist, Canzoni
ValutaEmozioni	Rappresenta la valutazione da parte dell'utente delle canzoni ascoltate.		UtentiRegistrati, Canzoni, EmozioniCanzone

3.4 Vincoli di integrità dello schema relazionale

3.4.1 Vincoli di integrità di chiave primaria:

- **CodiceFiscale** chiave primaria per l'entità **UtentiRegistrati**
- **IdPlaylist** chiave primaria per l'entità **Playlist**
- **Titolo, Autore, Anno** chiave primaria composta per l'entità **Canzoni**
- **IdValutazione** chiave primaria per l'entità **EmozioniCanzoni**

3.4.2 Vincoli di integrità referenziale

- Una playlist non può esistere senza una canzone associata.
- Una playlist non può esistere senza un utente associato.
- Una valutazione delle emozioni non può esistere senza un brano associato.
- Una valutazione delle emozioni non può esistere senza un utente associato.

3.4.3 Vincoli di integrità unique

- vincolo di integrità unique sulla tabella **Playlist** per cui un utente non può avere due playlist con stesso nome.

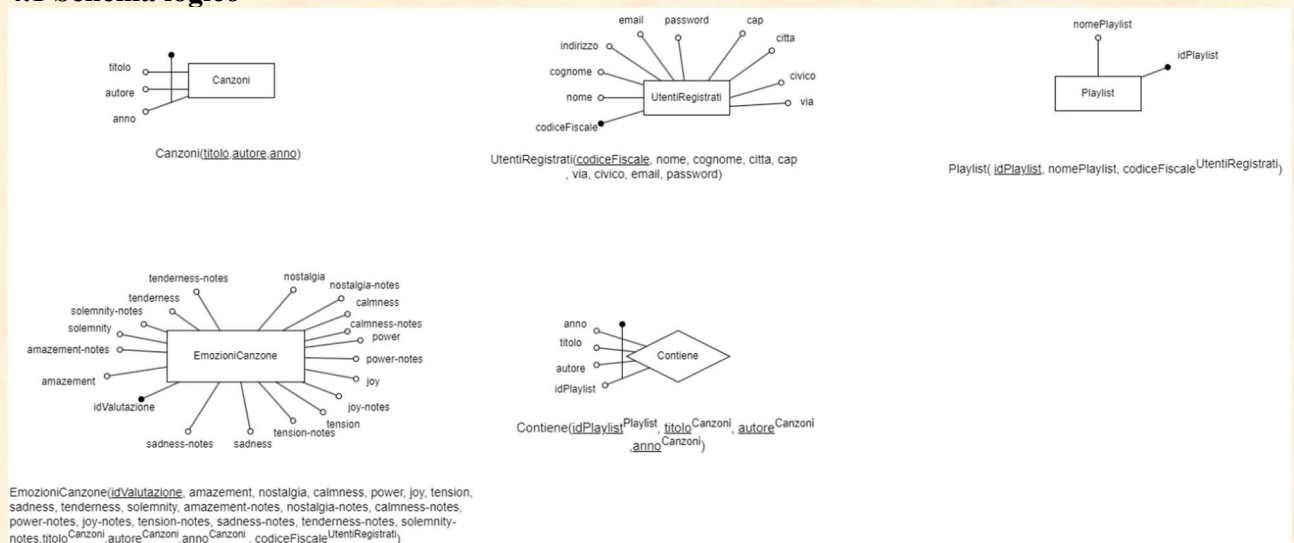
3.5 Motivazioni decisionali della progettazione dello schema concettuale

Durante la progettazione dello schema E/R si sono prese una serie di scelte giustificate dalle seguenti motivazioni:

- L'entità Canzoni possiede un identificatore composto da 3 attributi, Titolo, Autore e Anno, in quanto è possibile che differenti autori pubblichino canzoni con lo stesso titolo ma anche che lo stesso autore pubblichi la stessa canzone in anni differenti (e.g. remix); non è però possibile che lo stesso autore pubblichi due canzoni identiche nello stesso anno.
- L'entità UtentiRegistrati possiede come identificatore codiceFiscale e non email in quanto il codice fiscale rappresenta un'informazione maggiormente valida ed efficiente in termini di recupero di dati anagrafici dell'utente.
- L'entità Playlist possiede come identificatore un identificativo generato automaticamente dal DBMS in quanto è consentito allo stesso utente di possedere due playlist omonime.
- L'entità EmozioniCanzone possiede come identificatore un identificativo generato automaticamente dal DBMS, in modo da permettere, da parte dello stesso utente, di rilasciare più recensioni sulla stessa canzone.

3. Schema logico

4.1 Schema logico



4. Analisi schema logico

5.1 Fase di ristrutturazione

La fase di ristrutturazione dello schema E/R ha prodotto il medesimo schema concettuale a causa dell'assenza di gerarchie di generalizzazione, attributi composti e attributi multi valore.

5.2 Fase di traduzione

- Traduzione dell'entità Playlist: Playlist(idPlaylist, nomePlaylist, codiceFiscale^{UtentiRegistrati})
- Traduzione dell'entità Canzoni: Canzoni(titolo, autore, anno)

- Traduzione dell'entità UtentiRegistrati: UtentiRegistrati(codiceFiscale, nome, cognome, citta, cap, via, civico , email, password)
- Traduzione dell'entità EmozioniCanzone: EmozioniCanzone(idValutazione, amazement, nostalgia, calmness, power, joy, tension, sadness, tenderness, solemnity, amazement-notes, nostalgia-notes, calmness-notes, power-notes, joy-notes, tension-notes, sadness-notes, tenderness-notes, solemnity-notes, titolo^{Canzoni}, autore^{Canzoni}, anno^{Canzoni}, codiceFiscale^{UtentiRegistrati})

5.3 Motivazioni decisionali della traduzione dello schema concettuale

Non vi sono motivazioni legate alla fase di traduzione dello schema concettuale in quanto lo schema E/R fornito in input non presenta ambiguità.

5. Query significative

5.1. Registrazione utente

```
1. Registrazione utente
INSERT INTO utentiregistrati VALUES ('CCCMTT01P04I577M', 'Matteo', 'Cacciarino', 'Meda', 20821, 'Luigi Rho', 27, 'matcacciarino@gmail.com', 'ciao')
```

5.2. Login utente

```
2. Login utente (Su codice controlliamo se la password dell'utente inserita sia corretta)

SELECT nome, cognome, password FROM utentiregistrati WHERE LOWER(codicefiscale) = LOWER('CCCMTT01P04I577M')
```

5.3. getEmozioniFromBrano 1 – Restituisce le emozioni associate al brano specificato da un utente

```
3. getEmozioniFromBrano 1 - Selezione singole emozioni da EmozioniCanzone

SELECT idvalutazione, amazement, amazement_notes, nostalgia, nostalgia_notes, calmness, calmness_notes, power, power_notes, joy, joy_notes, tension, tension_notes, sadness, sadness_notes, tenderness, tenderness_notes, solemnity, solemnity_notes
FROM emozionicanzone
WHERE LOWER(titolo) = LOWER('walk right in') AND LOWER-autore) = LOWER('cannon's jug stompers') AND anno = 1929
```

5.4. getEmozioniFromBrano 2 – Restituisce le emozioni associate al brano specificato da un utente

```
4. getEmozioniFromBrano 2 - Ritorna la media delle emozioni da titolo, autore, anno

SELECT AVG(amazement) AS avg_amazement, AVG(nostalgia) AS avg_nostalgia, AVG(calmness) AS avg_calmness, AVG(power) AS avg_power, AVG(joy) AS avg_joy, AVG(tension) AS avg_tension, AVG(sadness) AS avg_sadness, AVG(tenderness) AS avg_tenderness, AVG(solemnity) AS avg_solemnity
FROM emozionicanzone
WHERE LOWER(titolo) = LOWER('walk right in') AND LOWER-autore) = LOWER('cannon's jug stompers') AND anno = 1929
```


5.5. Filtra per titolo – Restituisce le canzoni filtrate per il titolo specificato



```
5. Filtra per titolo 1 - ricerco in canzoni la canzone (do tutto)
SELECT * FROM canzoni WHERE LOWER(titolo) LIKE LOWER('dale%')
```

5.6. Filtra per autore anno – Restituisce le canzoni filtrate per autore e anno specificato



```
6. Filtra per autore anno - ricerco in canzoni la canzone (do tutto)
SELECT * FROM canzoni WHERE LOWER(autore) LIKE LOWER('ja%') AND anno = 2001
```

5.7. getCanzoniForPlaylist – Restituisce tutte le canzoni presenti nel database



```
7. getCanzoniForPlaylist - ritorno tutte le canzoni contenute all'interno della tabella canzoni, da
scegliere per la playlist
SELECT * FROM canzoni
```

5.8. createPlaylist 1 - Creazione della playlist richiesta dall'utente



```
8. createPlaylist 1 - creazione della playlist all'interno della tabella playlist
INSERT INTO playlist (nomeplaylist, codicefiscale) VALUES ('ProvaPlaylist', 'CCCMTT01P04I577M')
```


5.9. createPlaylist 2 - Inserimento delle canzoni all'interno della playlist tramite inserimento nella relazione “contiene”

```
9. createPlaylist 2 - Inserimento delle canzoni all'interno della playlist tramite inserimento nella tabella contiene

INSERT INTO contiene VALUES ((SELECT idplaylist FROM playlist WHERE LOWER(nomeplaylist) = LOWER('ProvaPlaylist') AND LOWER(codicefiscale)= LOWER('CCCMTT01P04I577M')), 'walk right in','cannon''s jug stompers',1929)
```

5.10. getPlaylist - Restituisce le playlist associate al codice fiscale di un utente

```
10. getPlaylist - ritorno le playlist associate al codice fiscale di un utente

SELECT * FROM playlist WHERE LOWER(codicefiscale) = LOWER('CCCMTT01P04I577M')
```

5.11. getCanzoniFromPlaylist – Restituisce le canzoni da inserire nella playlist

```
11. getCanzoniFromPlaylist - versione con nome della playlist (si può fare anche dando direttamente l id)

SELECT titolo, autore, anno FROM contiene WHERE idplaylist = (SELECT idplaylist
FROM playlist
WHERE LOWER(nomeplaylist) = LOWER('nome') AND
LOWER(codicefiscale)= LOWER('CCCMTT01P04I577M'))
```

5.12. inserisciEmozione - Inserimento emozioni

```
12. inserisciEmozione - inserimento emozioni

INSERT INTO emozionicanzone
(titolo,autore,codicefiscale,amazement,nostalgia,calmness,power,joy,tension,sadness,
tenderness,solemnity,amazement_notes,nostalgia_notes,calmness_notes,power_notes,joy_notes,
tension_notes,sadness_notes,tenderness_notes,solemnity_notes,anno)
VALUES ('walk right in','cannon''s jug
stompers','CCCMTT01P04I577M',5,5,5,5,5,5,5,5,5,'nota',
'nota','nota','nota','nota','nota','nota','nota',1929)
```