



UNIVERSITÀ DEGLI STUDI DI NAPOLI FEDERICO II

PROGETTO BASI DI DATI 22/23

Bisogna inserire cose più fighe tipo aggiungere un sommario o che
ne so

Adolfo Torcicollo
Francesca Pugliese

26 febbraio 2022

Part I

Descrizione del progetto

1 Analisi del problema

Si progetterà e svilupperà una base di dati di tipo relazionale per la gestione di progetti di un'azienda.

2 Approccio al problema

A fronte di un'analisi, è stato scelto di gestire la piattaforma che fa uso della base di dati in questione in modo totalmente indipendente rispetto agli strumenti utilizzati per la gestione di altre funzionalità di un app musicale.

Inoltre, non si tratta di un programma pubblico e aperto; pertanto, nella progettazione della base di dati è stato scelto di omettere informazioni non indispensabili.

3 Informazioni da analizzare

La base di dati deve garantire alla piattaforma che ne fa uso i seguenti servizi: ¹ Il database è stato pensato in stretta relazione con l'applicativo; pertanto, in fase di progettazione si è deciso di ripartire il carico di operazioni e vincoli tra entrambe le parti.

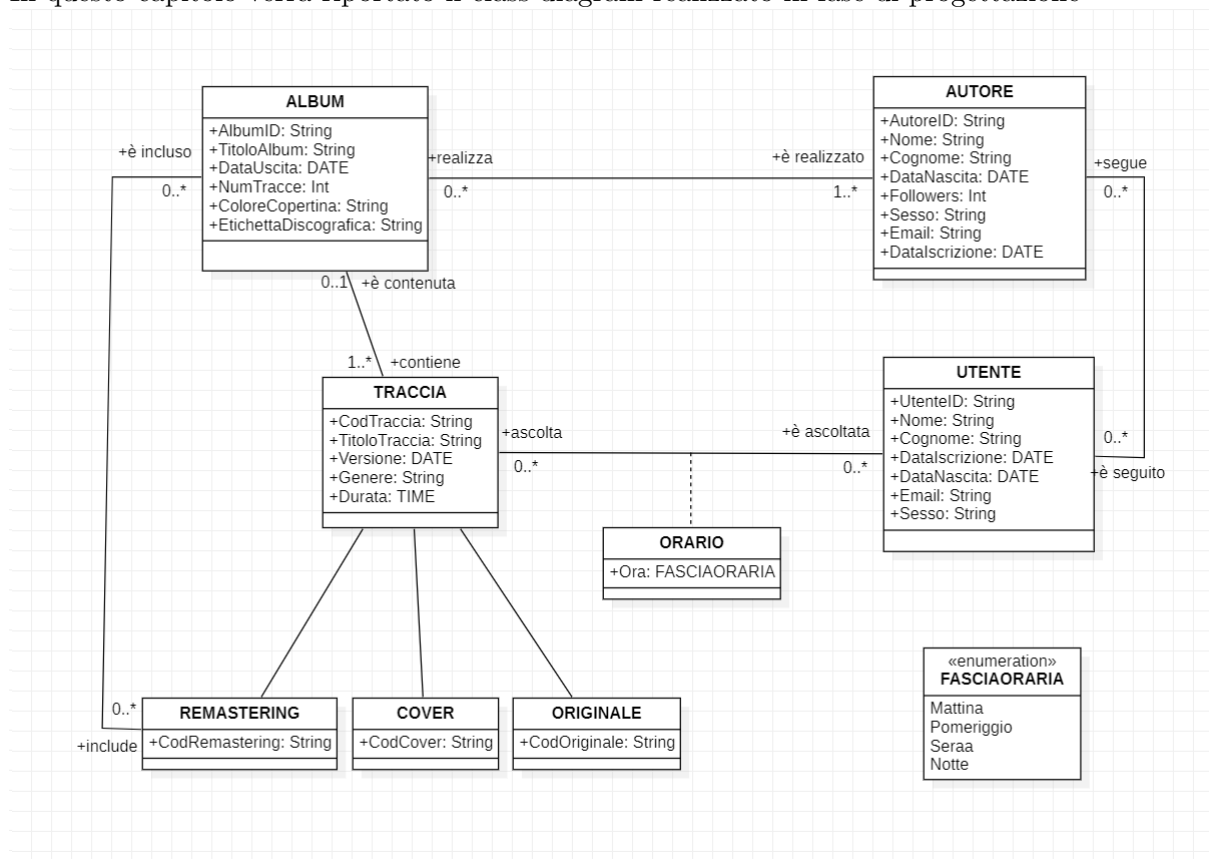
¹INSERIRE ROBA QUI, dove leggi il numero 1

Part II

Progettazione concettuale

4 Class Diagram

In questo capitolo verrà riportato il class diagram realizzato in fase di progettazione

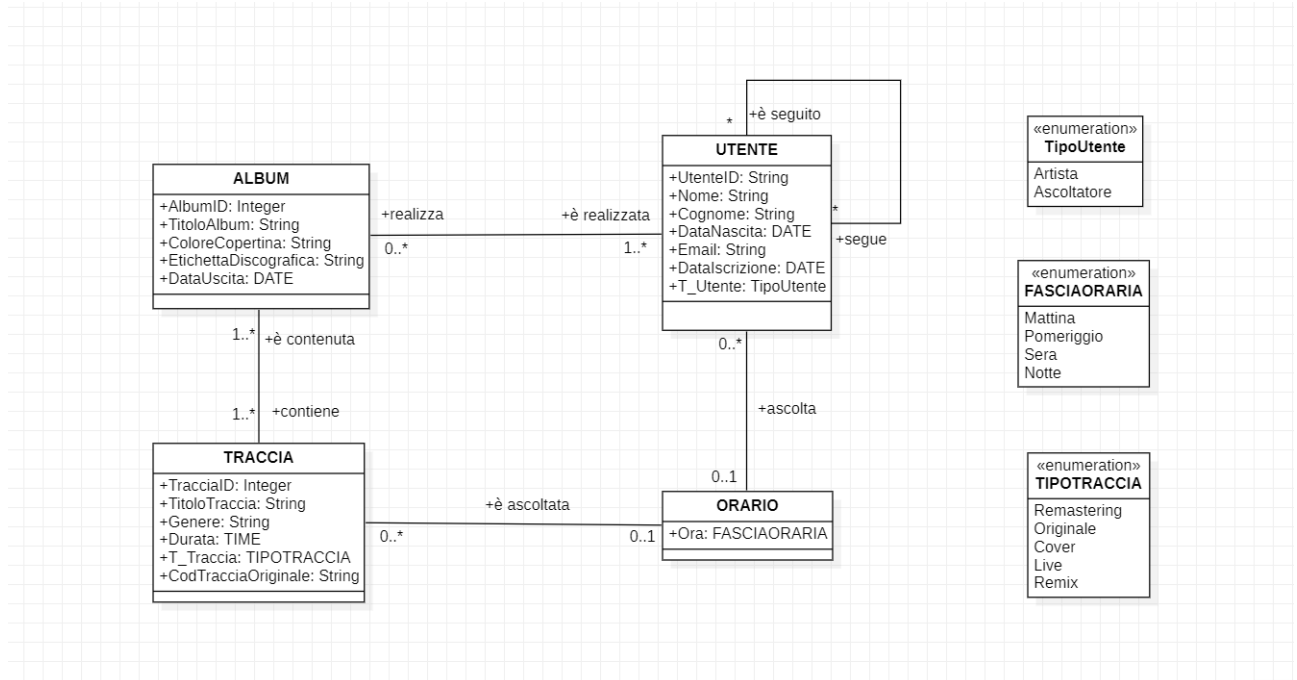


5 Class Diagram Ristrutturato

Per rendere il class diagram più vicino alla progettazione logica e alla realizzazione di tabelle sul database, si è provveduto a ristrutturarlo.

In particolare, come prima fase si è analizzato il contenuto delle classi per cercare eventuali ridondanze.

In una fase successiva sono state analizzate le associazioni, in particolare le gerarchie di specializzazione, sono state rese associazioni.



In Particolare:

ELIMINAZIONE DELLA SPECIALIZZAZIONE Una traccia può essere un 'Remastering', una cover, o un originale. Eliminiamo la specializzazione e inseriamo all'interno della classe **TRACCIA** un attributo discriminante, ossia **TIPOTRACCIA** che potrà avere valore uguale a 'Remastering', 'Live', 'Remix', 'Cover' oppure 'Originale'. Differenziamo il caso in cui **TIPOTRACCIA** avesse valore uguale a 'Cover', 'Remastering', 'Live', 'Remix'; introduciamo l'attributo **CodTracciaOriginale** per poter risalire alla traccia originale

ELIMINAZIONE DELLE RITONDANZE Nel class diagram originale apparivano attributi che possono essere "calcolati" tramite una query, come NumTracce nella classe **ALBUM** e Followers di **UTENTE**.

Inoltre, le classi **AUTORE** e **UTENTE** risultano essere costituite da attributi pressoché simili; fondiamo le due classi nella classe **UTENTE** e definiamo l'associazione following come ricorsiva. Per distinguere il caso di Autore e Ascoltatore abbiamo introdotto un nuovo attributo discriminante: **TIPOUTENTE**. Nella classe Utente è stato eliminato l'attributo Followers in quando tale informazione è recuperabile tramite la classe **FOLLOWERS**.

6 Dizionari

6.1 Dizionario delle Classi

CLASSE	DESCRIZIONE	ATTRIBUTI
ALBUM	Descrittore del supporto discografico, contenente una raccolta di tracce musicali di uno stesso autore/artista o più autori/artisti differenti	ALBUMID(integer): identificativo dell'album TitoloAlbum(string): nome dell'album N.Tracce(Integer): numero di tracce contenuteDataUscita(Date): riferimento temporale dell'album ColoreCopertina(String): colore scelto per la copertina dell'album EtichettaDiscografica(String): Marchio commerciale creato dalla casa discografica che ha lavorato sull'album
TRACCIA	Unità che compone un album discografico. E' un brano musicale.	TitoloTraccia(String): nome/titolo associato alla traccia che consente di scegliere cosa ascoltare. Genere(String): genere musicale a cui appartiene la traccia Durata(Time): tempo necessario per l'ascolto completo della traccia TracciaID(integer): Identificatore univoco della traccia Versione(date): anno della traccia originale se essa è una remastering T.Traccia(tipotraccia): ci da informazioni sulla tipologia (cover,remastering,originale,live,remix) CodTracciaOriginale(String):in caso di cover/remastering/live/remix ci da il riferimento alla traccia originale
UTENTE	Identità virtuale che usufruisce della piattaforma musicale	UtenteID (String): Identificativo univoco scelto dall'utente che comparirà al pubblico sulla piattaforma Nome(String): Nome ufficiale dell'utente Cognome(String): Cognome ufficiale dell'utente Email(String): email personale utilizzata dall'utente per accedere alla piattaforma DataIscrizione(String): Data dell'iscrizione alla piattaforma T.Utente(TipoUtente) : Ci dice se l'utente è un autore o un ascoltatore
ORA	Ci fornisce un riferimento temporale Fascia oraria in cui viene ascoltata una traccia da un utente	Ora(FasciaOrariaFASCIAORARIA): indica se una determinata traccia è stata ascoltata da quel determinato utente di mattina,pomeriggio,sera o notte un momento della giornata (mattina,pomeriggio,sera,notte).

6.2 Dizionario delle Associazioni

NOME	DESCRIZIONE	CLASSI COINVOLTE
REALIZZAZIONE	Esprime la relazione tra chi ha progettato/realizzato l'album e l'album stesso	Autore[0...*] ruolo "realizza": indica che l'autore può realizzare nessuno o molti album. Album[1...*] ruolo "è realizzato": indica che un album può essere creato da uno o più autori
SEGUITO	Descrive la connessione tra gli autori e gli utenti della piattaforma tra gli utenti come autori e utenti come ascoltatori	Autore[0...*] ruolo "è seguito": indica che l'autore può essere seguito da nessuno o da più utenti Utente[0...*] ruolo "segue": indica che l'utente può seguire nessuno/più utenti
ASCOLTO	Ci da informazioni sul momento in cui un utente ascolta una traccia	Utente[0...*] ruolo "ascolta": indica che l'utente può ascoltare nessuna o più tracce Traccia[0...*] ruolo "è ascoltata" Orario[0...*] ruolo "è trascorsa": indica che un momento della giornata può essere utilizzato o meno da un utente sulla piattaforma
RIPRODUZIONE	Ci da informazioni sul quando una traccia è ascoltata	Traccia[0...*] ruolo "è riprodotta": indica che una traccia può essere riprodotta o meno in una fascia oraria. Orario[0...*] ruolo "è utilizzata" : indica che un certo momento della giornata può essere o meno utilizzato per ascoltare una traccia sulla piattaforma
CONTIENE	Descrive l'appartenenza di tracce negli album	Album [1...*] ruolo "contiene": indica che un album contiene almeno una traccia. Traccia[1...*] ruolo "è contenuta": indica che una traccia deve essere contenuta in almeno un album.

6.3 Dizionario dei Vincoli

NOME VINCOLO	DESCRIZIONE
CorrectEmail	Gli indirizzi e-mail degli utenti devono essere indirizzi email di forma legittima, ovvero contenere almeno un carattere prima della @, almeno un carattere tra essa e il punto e almeno due caratteri nella parte finale.
CorrectBirth	La DataNascita di un utente deve essere precedente a DataIscrizione.
UnicitàEmail	Una stessa email non può essere associata a utenti diversi.
UnicitàUtenteID	Uno stesso UtenteID non può corrispondere ad utenti diversi.
UnicitàAlbumID	Non possono esistere album diversi con AlbumID uguali.
UnicitàTracciaID	Tracce diverse non possono avere TracciaID uguali.
LimitiDurataTraccia	Durata deve avere un valore compreso tra 00:00:05 secondi e 01:00:00 ora .
ParzialitàCodTracciaOriginale	L'attributo CodTracciaOriginale ha valore associato solo se T_Traccia è Remastering, Cover, Live, o Remix.
VincoloTipologiaTraccia	Una traccia con T_Traccia di valore 'Remastering' ha titolo uguali autore ma deve avere versione e titoloAlbum diversi. Le restanti tipologie di traccia non possono appartenere ad album diversi.

Part III

Progettazione Logica

7 Schema logico

Di seguito, la rappresentazione della progettazione logica derivata dal class diagram revisionato e che va a costituire la struttura tabellare del database relazionale. Gli attributi sottolineati rappresentano le chiavi primarie, mentre quelli con doppia sottolineatura indicano le chiavi esterne.

UTENTE (UtenteID, Nome, Cognome, DataNascita, DataIscrizione, Email, T_Utente)

ALBUM (AlbumID, TitoloAlbum, ArtistaID, ColoreCopertina, EtichettaDiscografica, DataUscita)

TRACCIA(TracciaID, AlbumID, TitoloTraccia, Durata, T_Traccia, CodTracciaOriginale)

SEGUITO(Segue, Seguito_da)

ASCOLTI (AscoltatoreID, TracciaID, ora)