



## Applicazione per la gestione di librerie musicali di un gruppo di utenti

Francesco Orlando    Alfredo Valerij Laino  
N86003618    N86003776  
18 febbraio 2022

# Indice

<b>1</b>	<b>Descrizione del progetto</b>	<b>3</b>
1.1	Descrizione e analisi del progetto . . . . .	3
1.2	Requisiti identificati . . . . .	3
<b>2</b>	<b>Progettazione</b>	<b>5</b>
2.1	Diagramma delle classi . . . . .	5
2.2	Diagramma della soluzione . . . . .	7
2.3	Sequence Diagram . . . . .	9
2.3.1	Login . . . . .	9
2.3.2	Ascolto . . . . .	10
<b>3</b>	<b>Manuale d'uso</b>	<b>11</b>

# Capitolo 1

## Descrizione del progetto

### 1.1 Descrizione e analisi del progetto

In questo progetto si sviluppa un programma utile per la gestione delle librerie musicali di un gruppo di utenti. Ogni utente potrà ascoltare diverse canzoni, creare delle playlist, pubbliche o private, ed aggiungere delle canzoni alla playlist brani preferiti. Viene data anche la possibilità di creare un profilo premium per poter accedere ad una qualità delle tracce migliori e poter dare un voto da 1 a 10 ad ogni traccia, quindi in base alla media dei voti ogni album ed ogni artista avrà un indice di gradimento. Viene poi data la possibilità ad un admin di monitorare gli ascolti delle tracce in determinate fasce orarie. Ogni traccia può avere la versione rimasterizzata o la cover creata da un altro artista. Gli utenti o altre a cercare delle tracce, degli artisti o degli album, potranno vedere i profili di altri utenti e visualizzare le playlist pubbliche di quest'ultimi.

### 1.2 Requisiti identificati

Leggendo attentamente la traccia, abbiamo individuato le seguenti classi:

- Per gestire gli utenti abbiamo una classe **Utente** da cui ereditano:
  - **UtentePremium**: un tipo di utente che ha la possibilità di ascoltare tracce di qualità migliore e votarle.
  - **Admin**: un utente che può monitorare gli ascolti delle tracce ordinate per fasce orarie.
- **Playlist**: ogni utente può creare una playlist con i brani che vuole e decidere se rendere la playlist pubblica o privata. Viene fornita di default una playlist “Brani preferiti” dove l’utente potrà inserire le tracce che preferisce.
- Abbiamo una classe **Traccia** che è padre di:

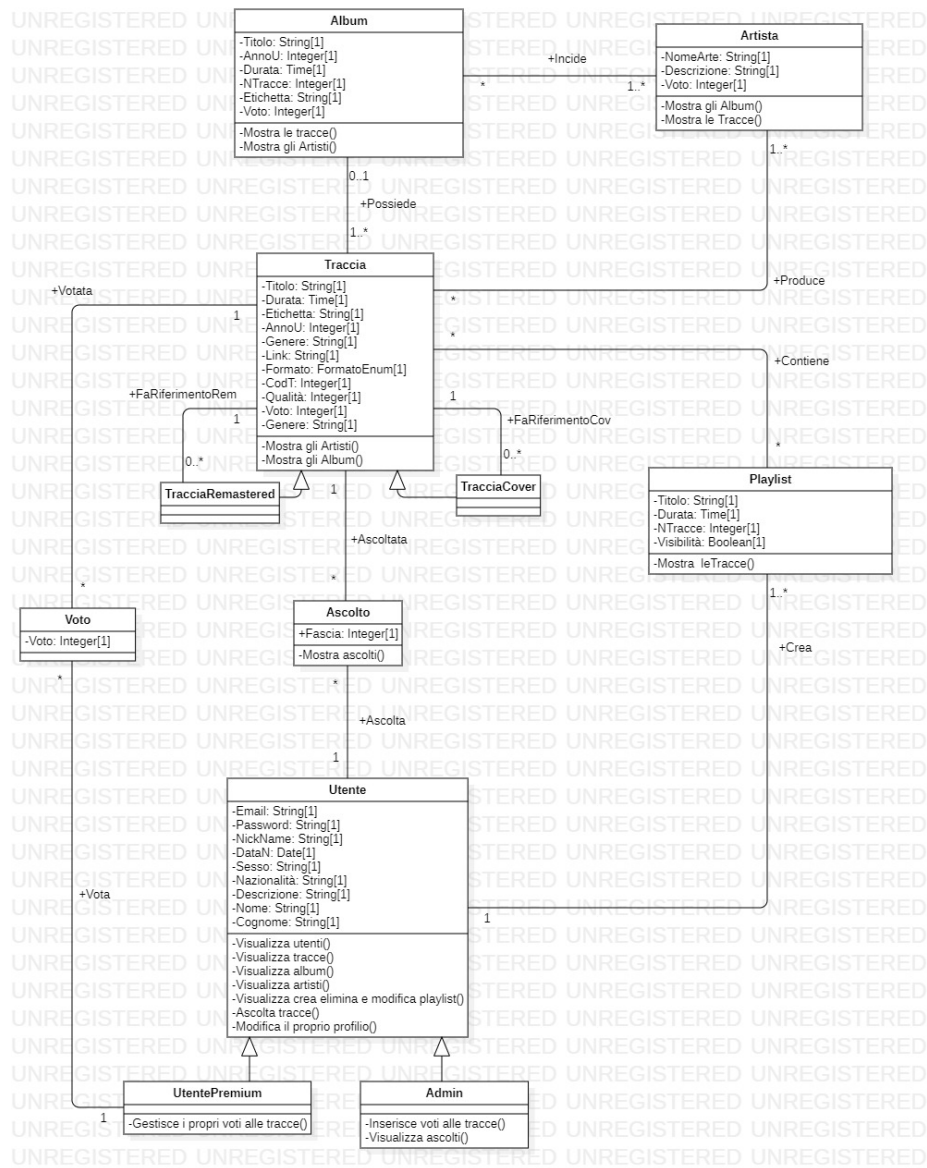
- **TracciaRemastered:** è la remastered di una traccia ed avrà un formato dell'audio migliore rispetto all'originale.
- **TracciaCover:** un artista può fare una cover di una traccia di un altro artista.
- **Album:** una traccia può appartenere ad un album oppure può essere un singolo.
- **Artista:** una traccia può essere prodotta da uno o più artisti e lo stesso vale anche per album.
- **Ascolto:** questa classe serve per gestire gli ascolti delle tracce.
- **Voto:** gli utenti premium possono votare le tracce quindi abbiamo bisogno di una classe voto per poterli gestire.

## Capitolo 2

# Progettazione

### 2.1 Diagramma delle classi

In questa sezione mostriamo il diagramma delle classi del problema in base ai requisiti che abbiamo individuato.



## 2.2 Diagramma della soluzione

Vediamo ora il diagramma della soluzione:

- **Model:** in questo package vi sono le classi che abbiamo individuato, sono state apportate alcune modifiche rispetto al diagramma delle classi per poterle adattare al database.
- **DAO:** qui abbiamo le interfacce per poter ricevere e inviare dati al database.
- **ImplementazioneDao:** ogni classe del package ImplementazioneDao implementa le interfacce DAO per il database postgres.
- **ConnessioneDatabase:** qui abbiamo la classe connessione che si occupa della comunicazione con il database.
- **Controller:** qui vi è la classe controller, al suo interno saranno istanziati gli oggetti delle classi di ImplementazioneDao, e si potranno chiamare i loro metodi. La GUI userà il controller come tramite, poiché è l'unica classe che può comunicare con le ImplementazioniDao e quindi ricevere e inviare dati al database.
- **GUI:** in questo package ci sono tutti i JFrame per l'interfaccia grafica.



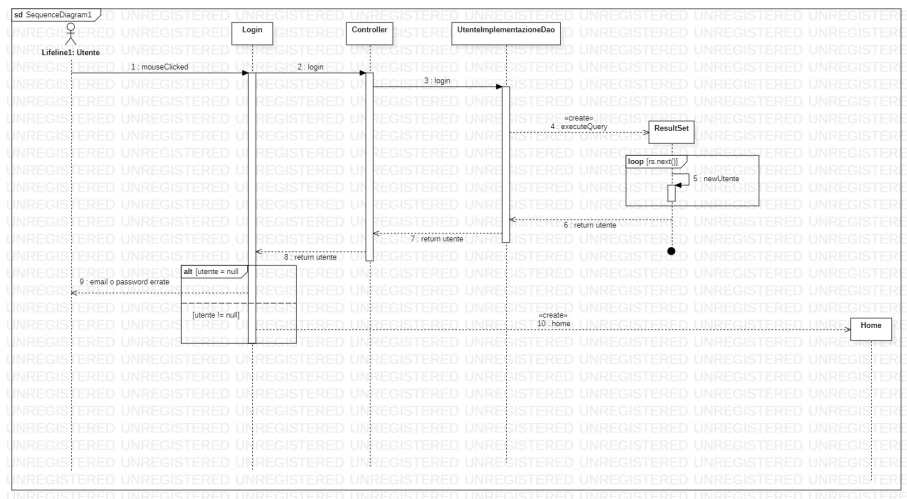


## 2.3 Sequence Diagram

Mostriamo il sequence diagram del login e quello dell'ascolto di una traccia.

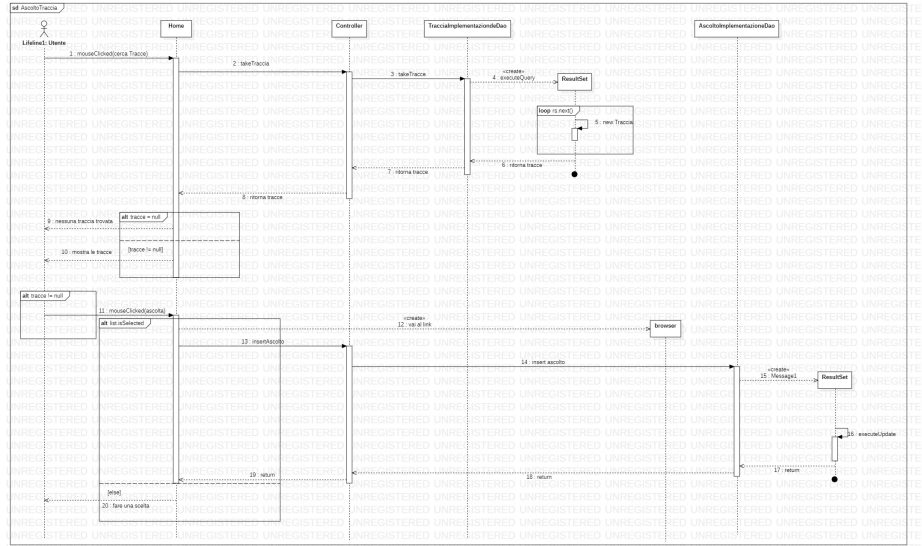
### 2.3.1 Login

Quando un utente ha fornito le credenziali del suo account cliccherà sul pulsante sign in, la GUI allora chiamerà il metodo login dal controller che a sua volta chiamerà il metodo login da UtenteImplementazioneDao. Da qui verrà eseguita la query per poter recuperare l'utente e istanziarlo come oggetto che verrà poi ritornato fino alla GUI di Login. Se l'oggetto utente è null allora vuol dire che le credenziali fornite non sono corrette, se invece è tutto corretto allora verrà aperta la schermata di home.



### 2.3.2 Ascolto

Quando un utente scrive il titolo di una traccia nella barra di ricerca e preme sul pulsante cerca traccia la GUI chiamerà il metodo takeTraccia dal controller, che a sua volta chiamerà da TracciaImplementazioneDao il metodo takeTraccia. Da qui verrà eseguita la query per poter recuperare le tracce che hanno il titolo inserito dall'utente, quindi verrà restituito ad home un arrayList di tracce. Se questo arrayList è vuoto allora vuol dire che non esistono tracce con il titolo inserito, altrimenti verranno mostrate. Se l'arrayList non è vuoto allora l'utente sceglie una traccia da ascoltare e preme il pulsante ascolta. La GUI quindi aprirà dal browser la pagina puntata dal link della traccia e chiamerà il metodo insertAscolto del controller, che a sua volta chiamerà da AscoltoImplementazioneDao il metodo insertAscolto. Quindi verrà eseguito un executeUpdate per inserire l'ascolto nel database.



## Capitolo 3

# Manuale d'uso

Da [questo link](#) è possibile aprire la repository di GitHub dove è possibile vedere l'intero codice e altre informazioni. Per poter accedere come utente admin inserire le seguenti credenziali:

- **email:** porfirio.tramontata@unina.it
- **password:** ForzaNapoli

Per poter accedere come utente solo premium inserire le seguenti credenziali:

- **email:** francesco.orlando3@studenti.unina.it
- **password:** y6t5r4e3w2q1

Infine se si vuole accedere come utente base queste sono le credenziali:

- **email:** loso@ahokok.com
- **password:** Miscusi

In ogni caso è possibile creare un nuovo utente. Il database è online quindi è possibile eseguire il programma senza doverlo creare in locale, ma proprio per questo motivo alcune ricerche saranno più lente di altre.

Dalla schermata home se si lascia vuota la barra di ricerca e si clicca su uno dei pulsanti di ricerca, ad esempio cerca traccia, verranno mostrate tutte le tracce presenti nel database.