



**UNIVERSITÀ  
DI PARMA**

Facoltà di Ingegneria - Ingegneria informatica, elettronica e delle  
telecomunicazioni

Basi di Dati e Web 2021-2022

**Lyrics**

**Realazione Progetto BDW**

Mamone Maximiliano 308214 Valenti Martina 308044

15/06/2022

# INDICE

1.	Introduzione .....	1
1.1.	Specifiche del progetto.....	1
1.2.	Tecnologie utilizzate .....	1
2.	Schema concettuale.....	2
3.	Schema logico.....	3
4.	Query .....	4
4.1.	Registrazione.....	4
4.2.	Login .....	4
4.3.	Ricerca lyrics.....	5
4.4.	Pagina lyrics .....	5
4.5.	Pagina artista .....	6
4.6.	Aggiunta artista .....	6
4.7.	Aggiunta lyrics .....	6

# **1. Introduzione**

## **1.1. Specifiche del progetto**

Il progetto consiste nella creazione di un sito web dinamico tramite il quale è possibile gestire una collezione di canzoni associate al loro testo e agli artisti che le hanno prodotte e cantate.

È possibile registrarsi e accedere al sito tramite una pagina di login, inserendo username e password.

Una volta effettuato il login, si accede alla homepage tramite la quale l'utente può usufruire di tutte le principali funzionalità offerte dal sito. È possibile effettuare una ricerca inserendo il nome di un artista, il titolo di una canzone oppure il testo (o una porzione di testo) di una canzone e visualizzare gli eventuali risultati corrispondenti.

Per ogni canzone è possibile visualizzare una pagina ad-hoc dalla quale si trovano le informazioni relative alla canzone (titolo e artista) e il suo testo.

Anche ogni artista possiede una pagina ad-hoc contenente un elenco di tutte le canzoni realizzate da tale artista e le relative informazioni.

Ciascun utente può aggiungere nuove canzoni al database e creare una pagina per un nuovo artista.

## **1.2. Tecnologie utilizzate**

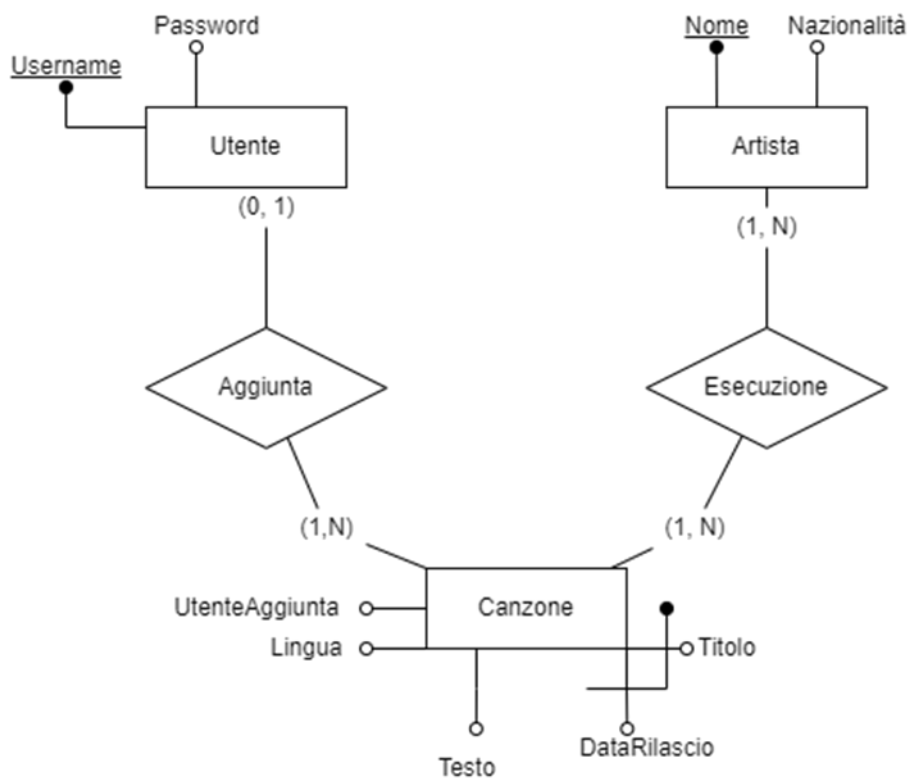
HTML, CSS e JavaScript, usufruendo delle librerie Bootstrap 5 e JQuery, per la gestione dell'interfaccia grafica e delle interazioni utente con il sito.

PHP per la gestione delle query e la generazione dinamica di elementi HTML in base ai risultati ottenuti dal database.

Un Database MySQL tramite XAMPP contenente tutte le informazioni necessarie al sito.

## 2. Schema concettuale

Fornisce una rappresentazione semplificata della realtà d'interesse in termini di un modello formale, ad alto livello, indipendente dall'implementazione del database. Permette di identificazione le entità che faranno parte del database e le relazioni tra di esse.



### 3. Schema logico

Schema logico che descrive la struttura dell'intero database:

Entità: User, Song, Artist

Relazioni: Performance (rappresenta il collegamento tra Song e Artist)

**User** (Username, Password)

**Song** (Title, ReleaseDate, Language, Text, UserAdd)

**Performance** (SongTitle, SongReleaseDate, ArtistName)

**Artist** (Name, Nationality)

Vincoli di integrità referenziale

**Performance.SongTitle** -> **Song.Title**

**Performance.SongReleaseDate** -> **Song.ReleaseDate**

**Performance.ArtistName** -> **Artist.Name**

**Song.UserAdd** -> **User.Username**

## 4. Query

### 4.1. Registrazione

Controlla se il nome utente è già utilizzato:

```
SELECT COUNT(*) AS cntUser FROM user u WHERE  
u.username='".$username."'
```

Aggiunge il nuovo utente nella tabella user nel caso in cui l'username scelto sia disponibile:

```
INSERT INTO `user` (`username`, `password`) VALUES ('".$  
$username."', '".$password."')
```

### 4.2. Login

Controlla se esiste un utente registrato le cui credenziali corrispondono con quelle inserite per effettuare il login:

```
SELECT COUNT(*) AS cntUser FROM user u WHERE  
u.username='".$username."' AND u.password='".$password."'
```

### 4.3. Ricerca lyrics

Cerca tutte le canzoni il cui titolo contiene la stringa di ricerca o il cui testo contiene la stringa di ricerca o il cui artista corrisponde alla stringa di ricerca:

```
SELECT s.Title, s.ReleaseDate, s.Language, p.ArtistName,
s.userAdd FROM song s JOIN performance p ON s.Title =
p.SongTitle AND s.ReleaseDate = p.SongReleaseDate WHERE
s.Title LIKE '%".$search.%' OR s.Text LIKE '%".$search.%'
OR p.SongTitle IN

( SELECT s.Title FROM song s JOIN performance p ON
s.Title = p.SongTitle WHERE p.ArtistName LIKE
'".$search.%' )

OR YEAR(s.ReleaseDate) = '$search'
GROUP BY s.Title, s.ReleaseDate, s.Language, p.ArtistName,
s.userAdd
```

### 4.4. Pagina lyrics

Cerca la canzone da mostrare e gli artisti che la hanno realizzata:

```
SELECT * FROM song s JOIN performance p ON s.ReleaseDate =
p.SongReleaseDate AND s.Title = p.SongTitle WHERE s.Title
LIKE '".$title."' AND s.ReleaseDate LIKE '".$date."'
```

## 4.5. Pagina artista

Cerca tutte le canzoni realizzate da un dato artista:

```
SELECT s.Title, s.Language, s.ReleaseDate, s.userAdd FROM
song s JOIN performance p ON s.ReleaseDate =
p.SongReleaseDate AND s.Title = p.SongTitle WHERE
p.ArtistName LIKE ' ".$name."
```

## 4.6. Aggiunta artista

Controlla che l'artista non esista già:

```
SELECT COUNT(*) AS cntArtist FROM artist a WHERE a.Name =
' ".$name."
```

Aggiunge l'artista alla tabella artist:

```
INSERT INTO `artist` (`Name`, `Nationality`) VALUES
(' ".$name."', ' ".$nationality."')
```

## 4.7. Aggiunta lyrics

Controlla che la canzone non esista già:

```
SELECT COUNT(*) AS cntSongs FROM song s WHERE s.Title = ' ".$title."'
AND s.ReleaseDate = ' ".$relDate."'
```



Aggiunge la canzone alla tabella song:

```
INSERT INTO `song` (`Title`, `ReleaseDate`, `Text`,  
`Language`, `userAdd`) VALUES ('$title', '$relDate',  
'$text', '$language', '".$_SESSION["username"]."')
```

Aggiunge la canzone alla tabella performance:

```
INSERT INTO `performance` (`SongTitle`, `SongReleaseDate`,  
`ArtistName`) VALUES ('$title', '$relDate', '$item')
```