

RELAZIONE DI PROGETTO

- Tecnologie Web -

Cortosi Vittorio,

Matricola: 884816

Tema del Sito

Il sito replica le funzionalità di una piattaforma di streaming musicale Online.

Il sito è composto da due pagine principali:

1. La pagina di introduzione, che corrisponde all'indice (index) del sito, è quella in cui l'utente è abilitato solo alla creazione di un account oppure ad effettuare l'accesso con i propri dati, se già registrato in precedenza.
2. La seconda pagina contiene tutto il contenuto della piattaforma (Libreria Musicale, Sezione per esplorare nuove tracce da aggiungere alla libreria, ...).

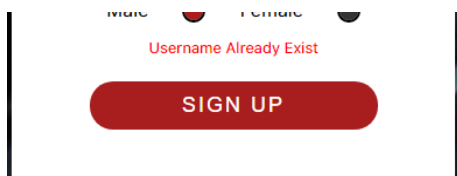
La parte di responsabilità è stata trattata solo nella pagina principale del sito.

Pagina principale

La pagina principale mostra in primo piano una frase di introduzione per invogliare l'utente a registrarsi e quindi ad accedere ai contenuti del sito. Se l'utente decide di registrarsi al sito, dovrà cliccare il button in alto a destra dell'interfaccia. Una volta apparsa la finestra di registrazione, l'utente dovrà inserire 4 parametri fondamentali ai fini della registrazione:

1. Username
2. E-mail
3. Password
4. Sesso

I controlli sull'input vengono effettuati sia lato client che server. Per il client ci sono controlli sul tag `<input>`, come la lunghezza massima dello username (con l'attributo `size`) ed il controllo del pattern della mail. Se l'input è corretto, verrà interrogata la pagina "sign-up.php" che effettuerà ulteriori controlli lato server per verificare che l'input sia corretto come per esempio, il metodo `strlen($username)` per verificare che la lunghezza dello username sia corretta e il metodo `filter_var($email, FILTER_VALIDATE_EMAIL)`, per verificare che la mail abbia un pattern corretto. Se uno dei controlli lato server dovesse dare riscontri negativi, l'utente sarà informato con il relativo messaggio d'errore.



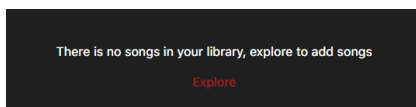
Se l'utente decide di effettuare l'accesso dovrà, come nella registrazione, cliccare un button situato sempre in alto a destra dell'interfaccia. Una volta aperta la finestra di login, l'utente potrà inserire le credenziali per autenticarsi. A differenza della registrazione, i controlli lato server vengono fatti solo per quanto riguarda la validità delle credenziali (se esiste lo username e se la password sia corretta, senza controlli sui pattern o lunghezze dell'input). Come nella registrazione se i controlli falliranno, l'utente sarà informato con un messaggio sopra il bottone di Login

Per entrambi i casi, le richieste al server vengono effettuate in maniera Asincrona, tramite metodo \$.get() offerto da jQuery. Per esempio, la chiamata asincrona per la registrazione è la seguente:

```
$.get("./php/signup.php", 'username=' + $('#signup_username').val() +
"&email=" + $('#signup_email').val() +
"&password=" + $('#signup_password').val() +
"&gender=" + $(".signup_input_fields input[type='radio']:checked").val(),
function(data) {
    switch (data) {
        case 'USERNAME_EXIST':
            $('#signup_error').addClass('error_show').text("Username Already Exist");
            break;
        case 'EMAIL_EXIST':
            $('#signup_error').addClass('error_show').text("Email already used");
            break;
        case 'REGISTERED':
            window.location = "./php/signup_success.php";
            break;
        default:
            $('#signup_error').addClass('error_show').text(data);
    }
});
```

Libreria Musicale

Se l'utente dovesse scegliere di accedere al sito, inserendo correttamente i dati, verrà indirizzato alla propria libreria musicale. Se l'utente è nuovo la libreria sarà ovviamente vuota e verrà invitato ad aggiungere nuovi brani alla libreria con un messaggio apposito



L'utente sarà invitato a cliccare sul button "Explore" che rappresenta la sezione in cui l'utente sfoglia i brani messi a disposizione dalla piattaforma e che può scegliere di aggiungere alla propria libreria.

Sezione Esplora

Nella sezione "Explore", l'utente potrà decidere di aggiungere alla propria libreria musicale sia i singoli brani che interi album.

Libreria Personale


Quando l'utente avrà aggiunto almeno un brano alla libreria personale, le sezioni (Songs, Albums, Artists) verranno popolate in base ai brani scelti dall'utente. L'utente potrà scegliere di riprodurre i propri brani, creare una coda di riproduzione, creare una playlist oppure aggiungere un brano ad una playlist o rimuovere un brano dalla propria libreria musicale.

Queue

La coda di riproduzione è gestita in Javascript, utilizzando come strutture dati: Un array per contenere i brani in coda, e un oggetto che rappresenta un brano in questo formato:


```
1. {
2.   'name': $songname,
3.   'artist': $songartist,
4.   'album': $songalbum }
```

Riproduzione

È possibile attivare la riproduzione casuale dalla barra dei controlli in alto dell'interfaccia . Questo è gestito secondo un metodo chiamato `shuffle_queue()` che applica un algoritmo di shuffling alla coda di riproduzione. È possibile attivare la riproduzione casuale sia prima di aggiungere brani alla coda, che mentre sono presenti (al momento del click sul bottone per la riproduzione casuale, l'algoritmo di shuffling verrà applicato ai brani in coda).

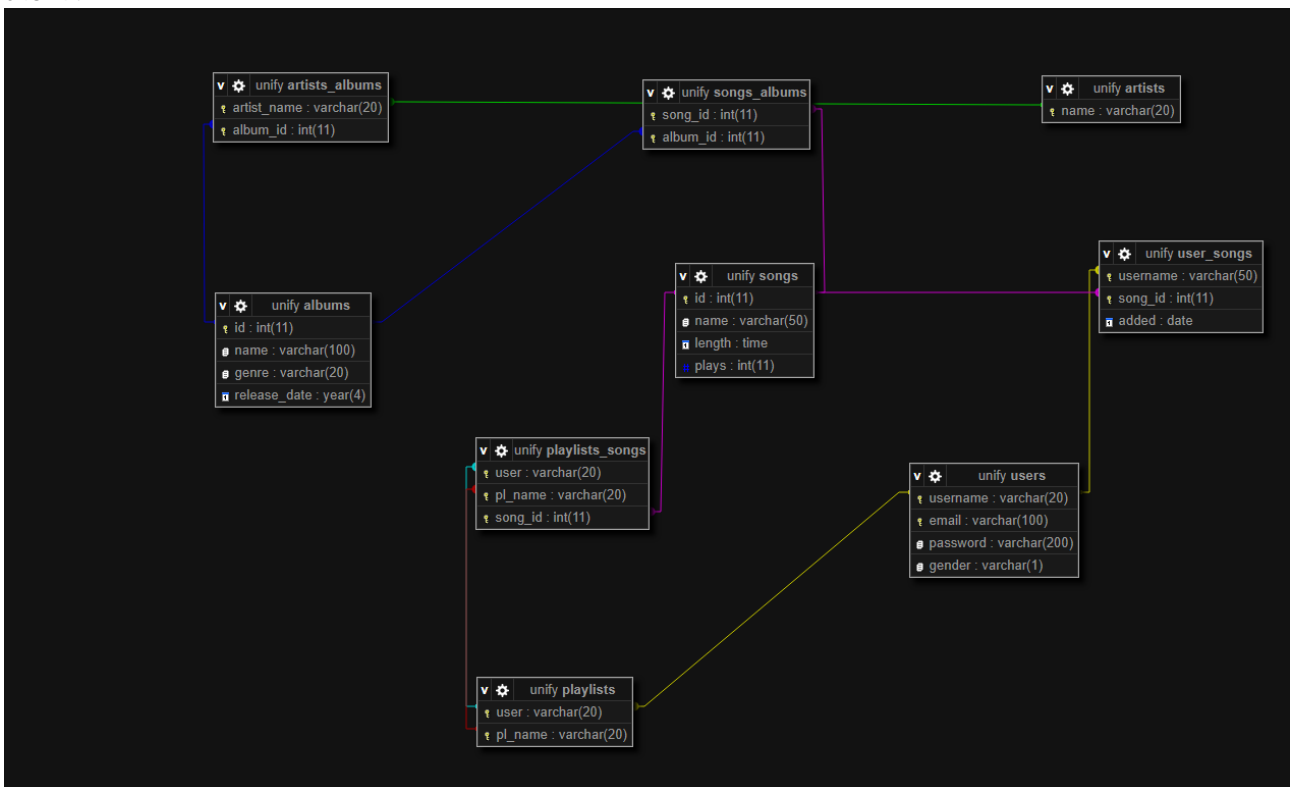
È possibile anche attivare la funzione di "repeat" per ripetere il brano attuale ininterrottamente.

La riproduzione è gestita con un oggetto **Audio**, manipolato in base a ciò che vuole riprodurre l'utente.

I metodi che gestiscono il suo comportamento sono: `play_this(songname, artist, album)`, che riproduce il brano che rispecchia i parametri formali, `play_album(songalbum, songartist, all = "NULL")` che riproduce un intero album, `play_next()` che riproduce, se presente, il prossimo brano in coda, `play_pause()` che gestisce l'avvio e la pausa della riproduzione, `change_volume()` che gestisce l'audio della traccia (si tratta di un handler, che viene richiamato da un listener in particolare): `$(".volume_slider").change(change_volume);`, lo slider è un semplice slider del volume , è un input type "range" riadattato in css per avere uno stile adeguato per la pagina.

Schema Relazionale

Il sito si appoggia ad un database relazionale per immagazzinare le informazioni sui brani, artisti, album ed utenti.



La password degli utenti è criptata secondo algoritmo SHA1.

Funzionalità Lato Server

Tutte le richieste al server vengono effettuate in Asincrono tramite richieste Ajax. Il file server-side addetto alla gestione delle richieste Ajax è chiamato "ajax_requests.php". Il server interroga il database in base alla richiesta del client e inoltra la risposta (JSON o solo testo) sempre in base alla richiesta. Per capire che tipo di richiesta inoltrare, il server ha bisogno di un parametro, che il client passerà attraverso parametri GET,

chiamato "type". In base al valore che assume questo parametro durante la richiesta, il server risponderà di conseguenza.

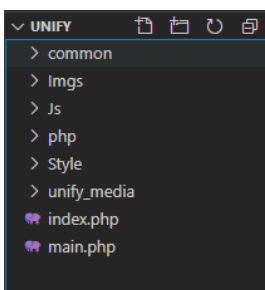
```
1. {
2.   "songs": [
3.     {
4.       "name"      : <songname>,
5.       "artist"    : <songartist>,
6.       "album"     : <album>
7.     },
8.     {
9.       "name"      : <songname>,
10.      "artist"     : <songartist>,
11.      "album"      : <album>
12.    },
13.    .....
14.    .....
15.  ]
16. }
```

Questa è la struttura del file "ajax_requests.php" collassato, per dare un'idea della struttura.

```
<?php
@include './connection.php';
session_start();
if (isset($_SESSION['username']) && $_SERVER['REQUEST_METHOD'] == 'GET' && isset($_GET['type'])) {
    switch ($_GET['type']) {
        case 'explore_songs': ...
        case 'explore_album': ...
        case 'song_list': ...
        case 'album_list': ...
        case 'user_playlists': ...
        case 'user_artists': ...
        case 'check_pl_exist': ...
        case 'insert_new_pl': ...
        case 'songs_into_pl': ...
        case 'insert_song_into_pl': ...
        case 'album_songs': ...
        case 'album_artist_per_user': ...
        case 'check_owned_song': ...
        case 'add_song_to_lib': ...
        case 'add_album_to_library': ...
        case 'check_owned_album': ...
        case 'remove_from_library': ...
    }
} else {
    exit('invalid request');
}
```

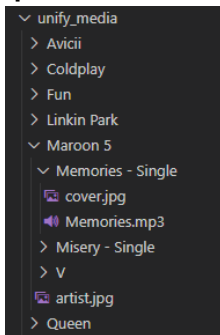
Front End

Il progetto è strutturato in subdirectory



- **Common:** files che non appartengono a nessuna delle categorie delle directory
- **Imgs:** Immagini e icone del sito
- **Js:** Files javascript per ogni pagina del sito
- **Php:** files php
- **Style:** files css per ogni pagina del sito

- **Unify_media:** contiene tutti i brani, immagini degli album ed immagini degli artisti, organizzato in questo modo:



Stile

Lo stile del sito è basato principalmente sull'uso di CSS GRIDS e FLEXBOX layout. Le animazioni della pagina principale come il fade-in della scritta di introduzione sono state realizzate con semplici keyframes animation che effettuano un cambiamento di opacità sull'oggetto.

```
.intro {
  font-size: 40px;
  text-align: center;
  animation-name: fade-in;
  animation-duration: 3s;
  animation-fill-mode: forwards;
  line-height: 70px;
}

@keyframes fade-in {
  from {
    opacity: 0;
  }
  to {
    opacity: 1;
  }
}
```

L'animazione sull'apertura delle finestre di Log-in e Sign-up, sfruttano una proprietà CSS chiamata Clip-path.

```
.login_panel,
.signup_panel {
  height: 100vh;
  width: 100vw;
  position: absolute;
  clip-path: circle(0px at 73% 7vh);
  transition: 0.5s all ease-out;
  background: url("../Imgs/index_background.jpg") center/cover;
}

.login_panel_open {
  clip-path: circle(2048px);
}
```

La proprietà trasforma l'oggetto a cui si applica, in un cerchio di dimensione (in questo caso) 0px. Dopo "at" bisogna specificare la posizione del clip-path (in questo caso, 73% da left e 7vh da top). Al momento del click per aprire la finestra di login/signup, verrà applicata la classe "panel_open" che espande il cerchio per tutta la viewport e dona un effetto di espansione.