

Progetto Tecnologie Web a.a. 2021/2022

Agostino Messina (913813)

Gennaio 2022

Indice

1	Tema del sito	2
1.1	Struttura	2
2	Pagina di introduzione (index.php)	3
3	Pagina di registrazione (signup.php)	3
4	Pagina principale (home.php)	5
4.1	Home / Esplora	6
4.2	Libreria personale	6
4.3	Brani salvati	6
4.4	Playlist	6
4.5	Riproduzione	6
4.5.1	Coda di riproduzione	7
4.5.2	Riproduzione casuale	7
4.5.3	Ripetizione brano	7
5	Ruoli utente	7
5.1	Esempio	7
6	Database	8
6.1	Schema	8
7	Funzionalità lato server	9
7.1	Esempio	9
8	Stile	10

1 Tema del sito

Il sito replica le funzionalità di una piattaforma di streaming musicale online. Le funzionalità principali offerte dal sito sono:

- Login e registrazione degli utenti;
- Riproduzione di artisti, album, brani e playlist;
- Creazione di playlist;
- Aggiunta di brani alle playlist o alla propria libreria musicale tramite drag and drop;
- Controlli sullo streaming musicale (play, pausa, avanti, indietro, riproduzione casuale, ripetizione brano).

1.1 Struttura

Il progetto è strutturato in subdirectory

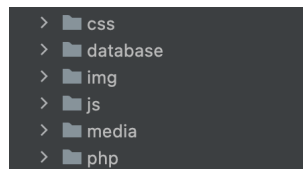


Figura 1: Struttura del progetto

- Css: files css per ogni pagina del sito;
- Imgs: Immagini, icone e gif;
- Js: Files javascript per ogni pagina del sito;
- Php: files php;
- Media: contiene tutti i brani, immagini degli album ed immagini degli artisti, organizzato in questo modo:

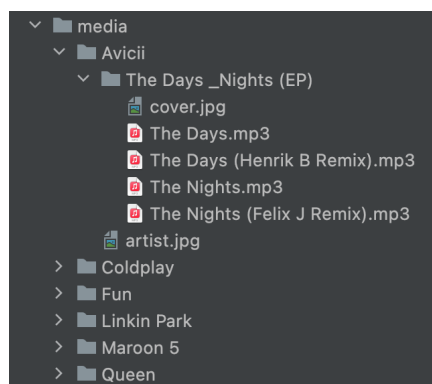


Figura 2: Struttura media files

Il sito è composto da tre pagine principali:

- La pagina di introduzione, che corrisponde all'indice (index) del sito, è quella in cui l'utente è abilitato all'accesso con un account registrato in precedenza;
- La pagina di signup volta ad effettuare la registrazione con i propri dati personali;
- L'ultima pagina contiene tutto il contenuto della piattaforma (libreria musicale, home per esplorare nuove tracce e artisti da aggiungere alla libreria, ...).

2 Pagina di introduzione (index.php)

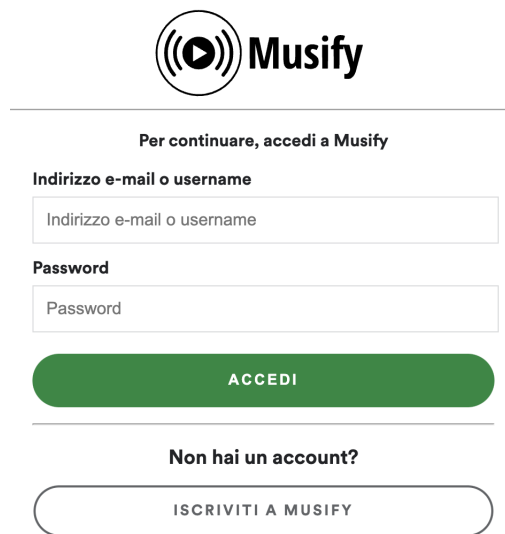


Figura 3: Login alla pagina index.php

La pagina di introduzione mostra in primo piano un form per invogliare l'utente ad accedere ai contenuti del sito specificando username (o e-mail) e password. Se l'utente decide di registrarsi, dovrà cliccare il button in basso per caricare la pagina dedicata.

3 Pagina di registrazione (signup.php)

Una volta caricata la pagina di registrazione, l'utente si troverà dinnanzi un form in cui dovrà inserire alcuni dati personali fondamentali ai fini della registrazione:

- E-mail;
- Password;
- Username;
- Data di nascita;
- 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 (con l'attributo maxlength) ed il controllo del pattern. Ulteriori controlli vengono effettuati tramite javascript nelle funzioni assegnate agli ascoltatori di determinati eventi.



Iscriviti gratuitamente per iniziare ad ascoltare.

Qual è il tuo indirizzo e-mail?

Conferma la tua e-mail

Crea una password

Come ti dobbiamo chiamare?

Qual è la data del tuo compleanno?

Anno

Mese

Giorno

Di che sesso sei?

☐ Maschio ☐ Femmina ☐ Non binario

ISCRIVITI

Figura 4: Registrazione alla pagina signup.php

Se l'input è corretto, verrà interrogata la pagina "signup_call.php" che effettuerà ulteriori controlli lato server, come il metodo "strlen(username)" per verificare la lunghezza dello username e il metodo "filter_var(email, FILTER_VALIDATE_EMAIL)" per verificare che la mail rispetti il pattern corretto. Se uno dei controlli lato server dovesse dare riscontri negativi, l'utente sarà informato con il relativo messaggio d'errore.

Crea una password

La password deve essere di almeno 8 caratteri

Come ti dobbiamo chiamare?

Questo username è già in uso

Figura 5: Errori sull'input

Le richieste al server vengono effettuate in maniera asincrona, tramite metodi "\$.get()" e "\$.post()" offerti da jQuery. Per esempio, la chiamata asincrona per la registrazione è la seguente:

```
$.ajax({
  type: 'POST',
  url: './signup_call.php',
  data: {
    "email": $('#email').val().trim(),
    "password": $('#password').val().trim(),
    "username": $('#username').val().trim(),
    "birth_date": $('#year').val() + $('#month').val() + $('#day').val(),
    "gender": $('form input[type=radio]:checked').val(),
  },
  success: function (data) {
    switch (data) {
      case 'USERNAME_EXIST':
        error_username.text("Questo username è già in uso");
        error_username.show();
        break;
      case 'EMAIL_EXIST':
        error_email.text("Questa e-mail è già in uso");
        error_email.show();
        break;
      case 'REGISTERED':
        window.location = "./signup_success.php";
        break;
      default:
        error_gender.text(data);
        error_gender.show();
    }
  }
});
```

Figura 6: Chiamata POST da ajax per la registrazione

4 Pagina principale (home.php)

Se l'utente dovesse accedere al sito, inserendo correttamente i dati richiesti, verrà indirizzato alla pagina principale contenente i contenuti musicali. Questa si presenta con un nav bar sulla sinistra, contenente tutte le sezioni principali del sito:

- Home / Esplora;
- Libreria;
- Brani salvati;
- Playlist.

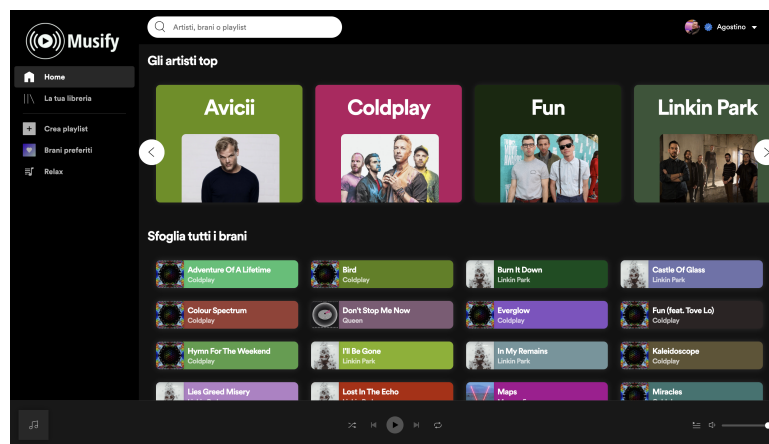


Figura 7: Schermata principale home.php

4.1 Home / Esplora

Nella sezione Home, l'utente potrà scoprire gli artisti e i brani presenti nello streaming musicale e decidere di aggiungerli alla propria libreria personale, riprodurli o trascinarli (tramite drag and drop) su una delle playlist per aggiungerli.

4.2 Libreria personale

La libreria personale contiene nella barra superiore un nav bar per poter scegliere la sotto sezione desiderata, appositamente popolata con i brani salvati, tra cui:

- Artisti;
- Brani;
- Playlist.

Se l'utente non ha ancora nessun brano salvato, la libreria personale sarà vuota e apparirà sia un apposito messaggio per invitare l'utente ad aggiungere nuovi brani che un link che porta alla sezione [Home / Esplora](#).

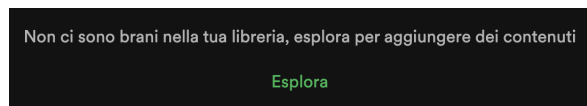


Figura 8: Sezione vuota

L'utente potrà inoltre scegliere di riprodurre questi brani, creare una coda di riproduzione, creare una playlist, aggiungere un brano ad una playlist o rimuoverlo dalla propria libreria musicale.

4.3 Brani salvati

La sezione relativa ai brani salvati consiste in una pagina che contiene tutte le tracce preferite dall'utente.

4.4 Playlist

L'ultima sezione è composta da tutte le playlist create dall'utente, cliccando su una di queste si aprirà nel pannello centrale una schermata con l'icona della playlist, il suo nome, il numero di brani e la lista di questi con le relative informazioni e opzioni.

4.5 Riproduzione

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(song, artist, album)`, che riproduce il brano che rispetta i parametri formali;
- `play_album(songalbum, songartist, all)` 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, mentre lo slider è un input type "range" riadattato in css per avere uno stile adeguato per la pagina.

4.5.1 Coda di riproduzione

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:

```
{
  'name': item.name,
  'artist': item.artist,
  'album': item.album
}
```

Figura 9: Rappresentazione di un brano

4.5.2 Riproduzione casuale

È possibile attivare la riproduzione casuale dalla barra dei controlli in basso nella pagina, dentro la sezione dei controlli sulla riproduzione.

Questa è gestita 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).

4.5.3 Ripetizione brano

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

5 Ruoli utente

La web application dispone di base di due ruoli utente:

- utente base;
- amministratore.

L'amministratore ha in più la possibilità di nascondere o mostrare determinati brani dal resto degli utenti base, in modo da avere il pieno controllo sullo streaming e sui suoi contenuti (anche nei migliori streaming musicali può capitare la rimozione dei brani per violazioni o licenze revocate).

5.1 Esempio

Di seguito viene messo a confronto la vista dell'amministratore da quella di un utente base:

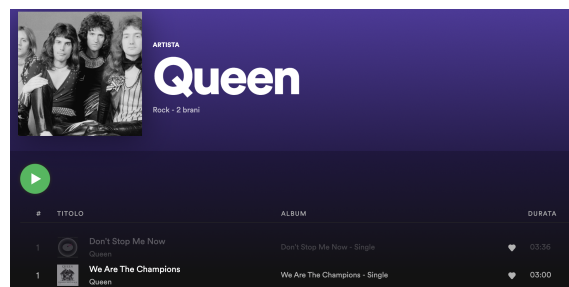


Figura 10: Vista dell'amministratore

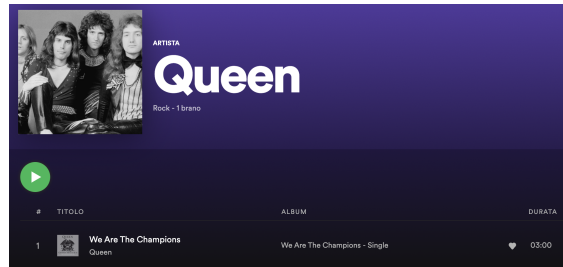


Figura 11: Vista dell'utente base

6 Database

Il sito si appoggia ad un database relazionale per immagazzinare le informazioni sui brani, artisti, album, playlist ed utenti. La password degli utenti viene criptata secondo algoritmo SHA1.

6.1 Schema

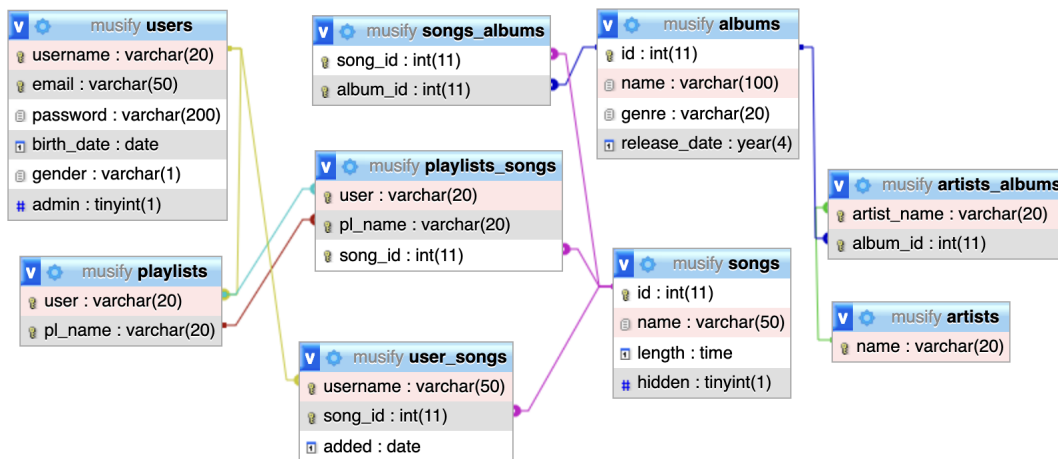


Figura 12: Schema del database

7 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 “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 distinguere la richiesta da inoltrare, il server ha bisogno di un parametro chiamato “type” che il client passerà come parametri GET.

```
case 'all_songs':...
case 'all_artists':...
case 'user_songs':...
case 'user_albums':...
case 'user_playlists':...
case 'user_artists':...
case 'songs_into_pl':...
case 'all_album_songs':...
case 'user_album_songs':...
case 'user_artist_songs':...
case 'all_artist_songs':...
case 'check_owned_song':...
case 'add_song_to_lib':...
case 'remove_from_library':...
case 'remove_from_pl':...
case 'check_pl_exist':...
case 'insert_new_pl':...
case 'insert_song_into_pl':...
case 'check_hidden_song':...
case 'hide_song':...
case 'show_song':...
```

Figura 13: Tipi di richieste

7.1 Esempio

Viene riportato un esempio che mette in luce la comunicazione tra front-end e back-end. Eseguendo questa chiamata GET:

```
Request URL: http://localhost/Progetto-TWEB/php/requests.php?username=Agostino&artist=Avicii&type=all_artist_songs
Request method: GET
Status code: 200 OK
```

Figura 14: Chiamata get

Si ottiene la seguente risposta:

```
▼ albums: [{albumname: "The Days _Nights (EP)", genre: "Dance",...}]
  ▼ 0: {albumname: "The Days _Nights (EP)", genre: "Dance",...}
    albumname: "The Days _Nights (EP)"
    genre: "Dance"
  ▼ songs: [{songname: "The Days", length: "00:04:38"},...]
    ▼ 0: {songname: "The Days", length: "00:04:38"}
      length: "00:04:38"
      songname: "The Days"
    ▼ 1: {songname: "The Days (Henrik B Remix)", length: "00:03:56"}
      length: "00:03:56"
      songname: "The Days (Henrik B Remix)"
    ▼ 2: {songname: "The Nights", length: "00:02:56"}
      length: "00:02:56"
      songname: "The Nights"
    ▼ 3: {songname: "The Nights (Felix J Remix)", length: "00:03:20"}
      length: "00:03:20"
      songname: "The Nights (Felix J Remix)"
  nsongs: 4
```

Figura 15: Risultato della chiamata

8 Stile

Lo stile del sito è basato principalmente sull'uso di CSS GRIDS e FLEXBOX layout. Le animazioni della pagina principale come l'highlight dei link alle sezioni nel nav sulla sinistra sono state realizzate nel modo seguente:

```
.li-item, .link-item, .link-section {  
  color: #b3b3b3;  
  transition-duration: .2s;  
  transition-property: color;  
  transition-timing-function: linear  
}
```

Figura 16: Transition per la creazione di animazioni

Altre animazioni grafiche vengono effettuate tramite JQuery utilizzando gli appositi metodi "fadeIn", "fadeOut", "show", "hide".

È stato anche realizzato lo scroll di un div contenente gli artisti nella homepage:

```
let the #scroll div slide to the specified direction  
function slide(direction) {  
  var container = document.getElementById( 'scroll');  
  scrollCompleted = 0;  
  var slideVar = setInterval( handler: function () {  
    if (direction == 'Indietro') {  
      container.scrollLeft -= 100;  
    } else {  
      container.scrollLeft += 100;  
    }  
    scrollCompleted += 100;  
    if (scrollCompleted >= 500) {  
      window.clearInterval(slideVar);  
    }  
  }, timeout: 50);  
}
```

Figura 17: Funzione di scroll