



Relazione progetto Tec. e Linguaggi per il Web

Giacomo Lucca 02678A

[Contenuto del file](#)

Struttura, presentazione del sito web e scelte implementative

1. Come avviare il server	2
2. Presentazione del sito	3
0.9 Registrati	12
01. Home	3
02. Playlist di Spotify	4
03. Album	5
04. Artista	6
05. Cerca su Spotify	7
06. Cercare internamente al sito	8
07. Playlist interne	9
08. Login	11
10. Crea playlist interna	12
11. Modifica playlist	13
12. Cambiare le preferenze del profilo	14
13. Aggiorna profilo	15
3. SWAGGER	15
4. Impostazione delle risorse	17
1. Database	17
2. Back-end	18
3. CSS	18
4. Componenti EJS	18
5. Front-end	19
6. Struttura finale	19
5. Note implementative	20
6. Prove di funzionamento	21
1. Eliminazione del primo brano di una playlist	21
2. Cambio username con uno già presente	22

Come avviare il server

Avere installato sul PC:

- Node.js per la parte server del progetto.
- I vari pacchetti richiesti per il corretto funzionamento dell'applicazione ovvero:
 - o cors
 - o path
 - o express
 - o ejs
 - o mongodb
 - o dotenv
 - o swagger-ui-express
 - o swagger-autogen

L'installazione può avvenire attraverso l'esecuzione del comando: *npm install*

- Creare un file di nome .env in cui andrà contenuto:

```
PORT= NUMERO_PORTA
CLIENT_ID="client_id_spotify"
CLIENT_SECRET="client_secret_spotify"

ACCESS_TOKEN=""
URI="mongodb_server"
```

- Per eseguire il server aprire il file server.js con Nodjs: *node ./server.js*
- Il server successivamente sarà raggiungibile attraverso la porta indicata nel file .env all'indirizzo 127.0.0.1 .

Indirizzi web utili:

- *127.0.0.1:PORTA* per accedere alla pagina iniziale.
- *127.0.0.1:PORTA/api-docs* per accedere alla documentazione delle API generato con swagger.

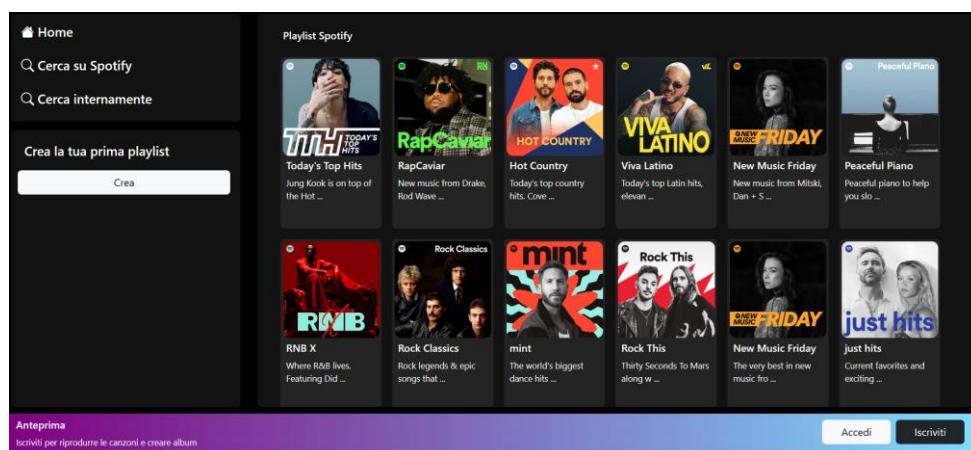
Presentazione del sito

1. Home

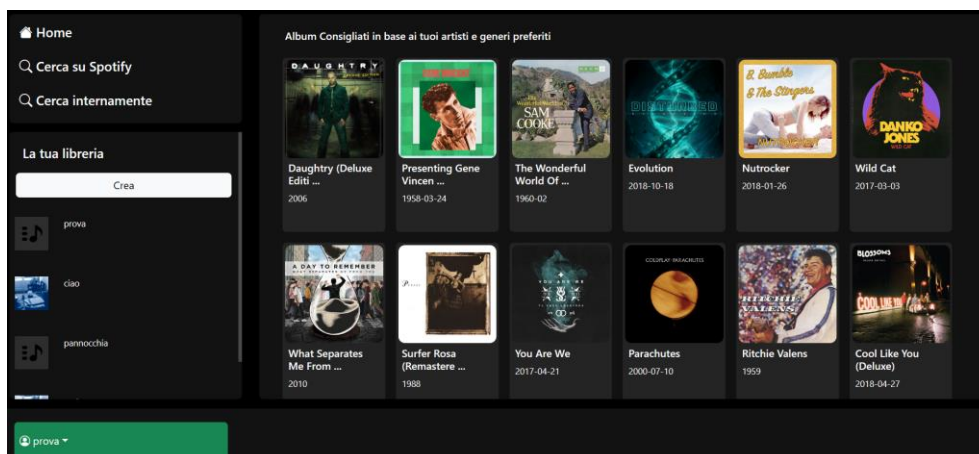
Dopo essersi connessi alla pagina 127.0.0.1 verrà mostrato la home page formata da:

- Nav bar laterale per andare nelle varie sezione del sito.
- Barra laterale dove verranno salvati le varie playlist che si segue oppure si è proprietario.
- Schermata principale che mostra i vari dati.
- Footer bar che mostra o il profilo dell'utente autenticato oppure i tasti per accedere o iscriversi al sito.

La schermata principale mostrerà contenuti diversi in base se autenticati o meno. Se non si autenticati oppure si è autenticati ma non si ha ancora dichiarato il genere preferito e l'artista preferito, verranno mostrate le prime 20 playlist dello user di Spotify "Spotify".

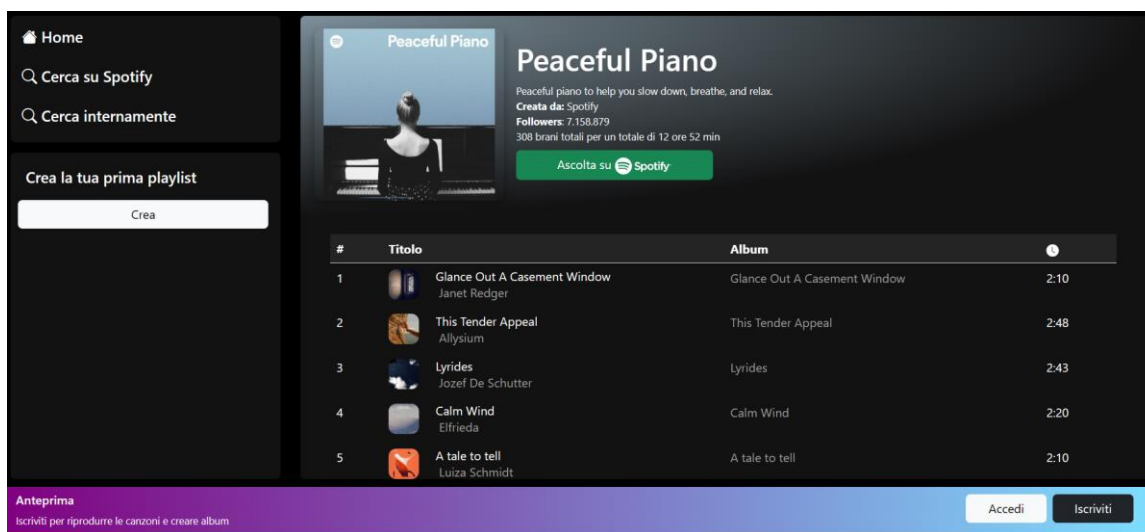


Altrimenti mostrerà questo:

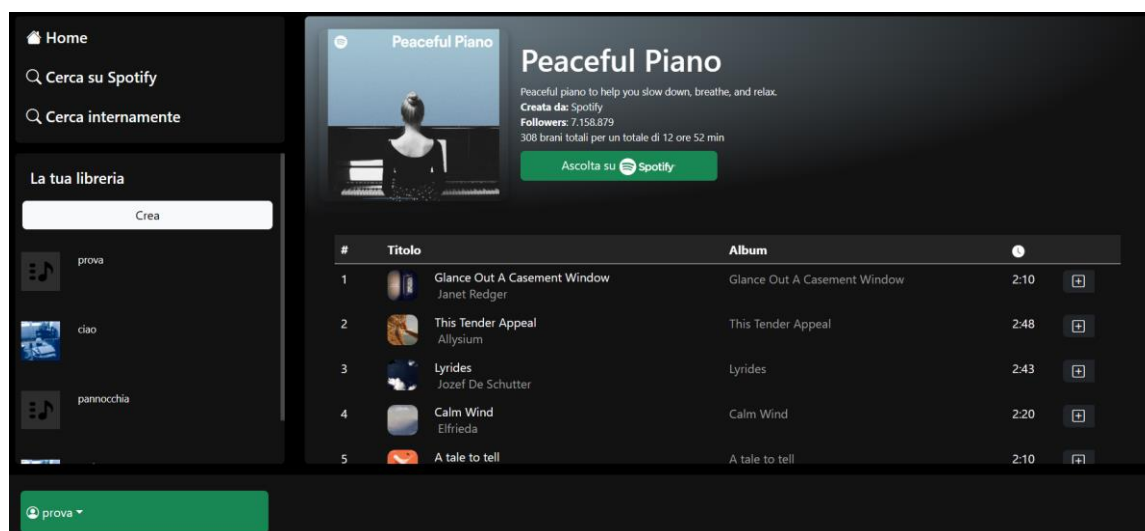


2. Playlist di Spotify

Cliccando sulla singola playlist verrà mostrata la playlist di Spotify in dettaglio



Se si è autenticati, mostrerà questo:



Verrà mostrata una nuova voce con il simbolo + che indica la possibilità di aggiungere il brano alla playlist di cui siamo proprietari.

Durante l'aggiunta del brano alla playlist interna del sito, se non ci sono ancora brani, aggiungerà come immagine copertina la prima canzone che è presente nella playlist. MongoDB ordina i dati in base all'ordine di inserimento, quindi se successivamente eliminiamo il primo brano contenuto nella playlist, l'immagine della copertina cambierà con quella successiva, se non è presente nessun brano, ne imposterà una di default.

3. Album

Stessa cosa vale per gli album.

Home

Cerca su Spotify

Cerca internamente

Crea la tua prima playlist

Crea

40 Most Beautiful PIANO CLASSICS

Rilasciato il 2011-06-28

©A Warner Classics/Teldec release, © 2011 Warner Music UK Limited
©A Warner Classics/Teldec release, © 2011 Warner Music UK Limited
40 brani totali per un totale di 2 ore 58 min

Ascolta su Spotify

#	Titolo	
1	Beethoven: Piano Sonata No. 14 in C-Shar... Ludwig van Beethoven,	4:49
2	Grieg: Peer Gynt Suite No. 1, Op. 46: I... Edvard Grieg, Cyprien Katsaris	4:08
3	Debussy: Suite bergamasque, CD 82, L. 75... Claude Debussy, Monique Haas	5:27
4	Ravel: Pavane pour une infante défunte, M. 19 Maurice Ravel, Monique Haas	7:07
5	Schumann: Kinderszenen, Op. 15: No. 7, T...	2:56

Anteprima

Iscriviti per riprodurre le canzoni e creare album

Accedi

Iscriviti

Per chi è autenticato:

Home

Cerca su Spotify

Cerca internamente

La tua libreria

Crea

40 Most Beautiful PIANO CLASSICS

Rilasciato il 2011-06-28

©A Warner Classics/Teldec release, © 2011 Warner Music UK Limited
©A Warner Classics/Teldec release, © 2011 Warner Music UK Limited
40 brani totali per un totale di 2 ore 58 min

Ascolta su Spotify

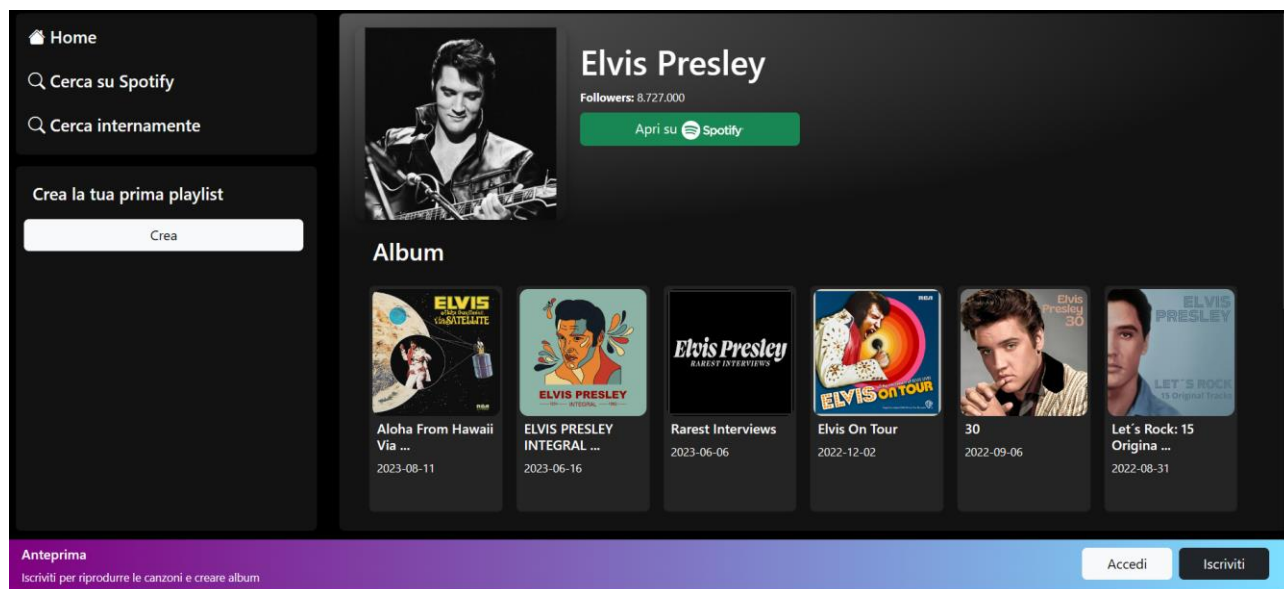
#	Titolo		
1	Beethoven: Piano Sonata No. 14 in C-Shar... Ludwig van Beethoven,	4:49	+
2	Grieg: Peer Gynt Suite No. 1, Op. 46: I... Edvard Grieg, Cyprien Katsaris	4:08	+
3	Debussy: Suite bergamasque, CD 82, L. 75... Claude Debussy, Monique Haas	5:27	+
4	Ravel: Pavane pour une infante défunte, M. 19 Maurice Ravel, Monique Haas	7:07	+

prova

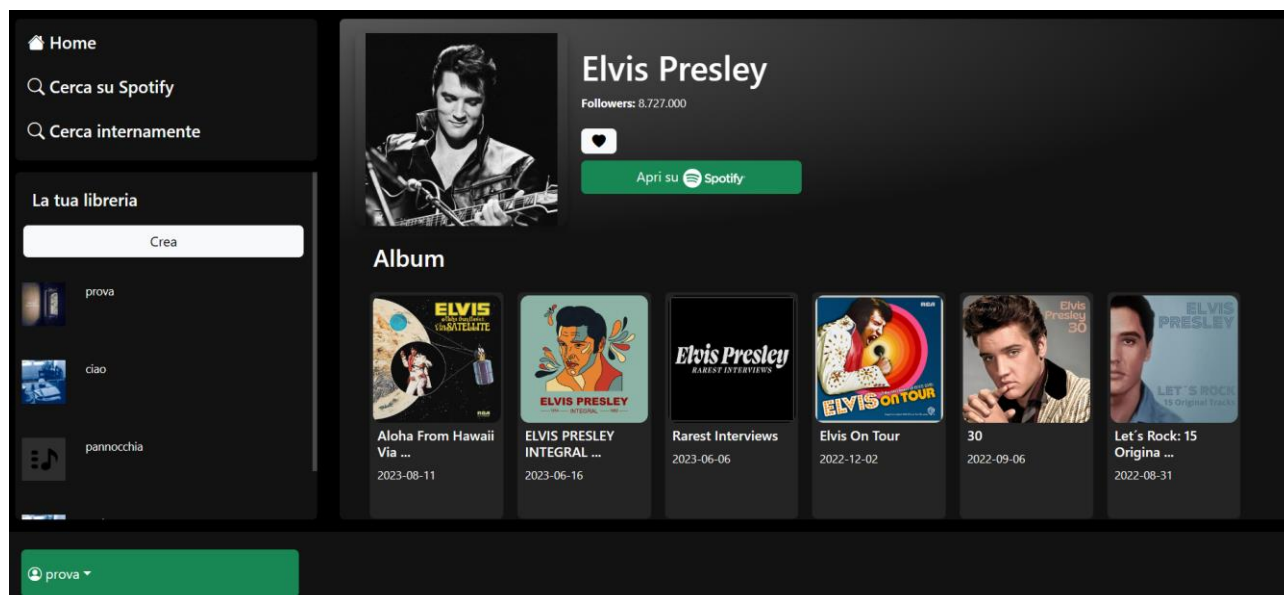
In queste due sezioni, ovvero playlist ed album, sarà presente un pulsante che permetterà di aprire il suo corrispettivo (album o playlist) all'interno di Spotify.

4. Artista

Cliccando su un artista verrà mostrata la sua pagina in cui saranno contenuti gli ultimi 20 album che ha pubblicato e i suoi ultimi 20 singoli.



Per chi è autenticato verrà mostrato un tasto extra. Questo tasto permette di inserire l'artista nella lista degli artisti preferiti reperibile nella sezione "Profilo"

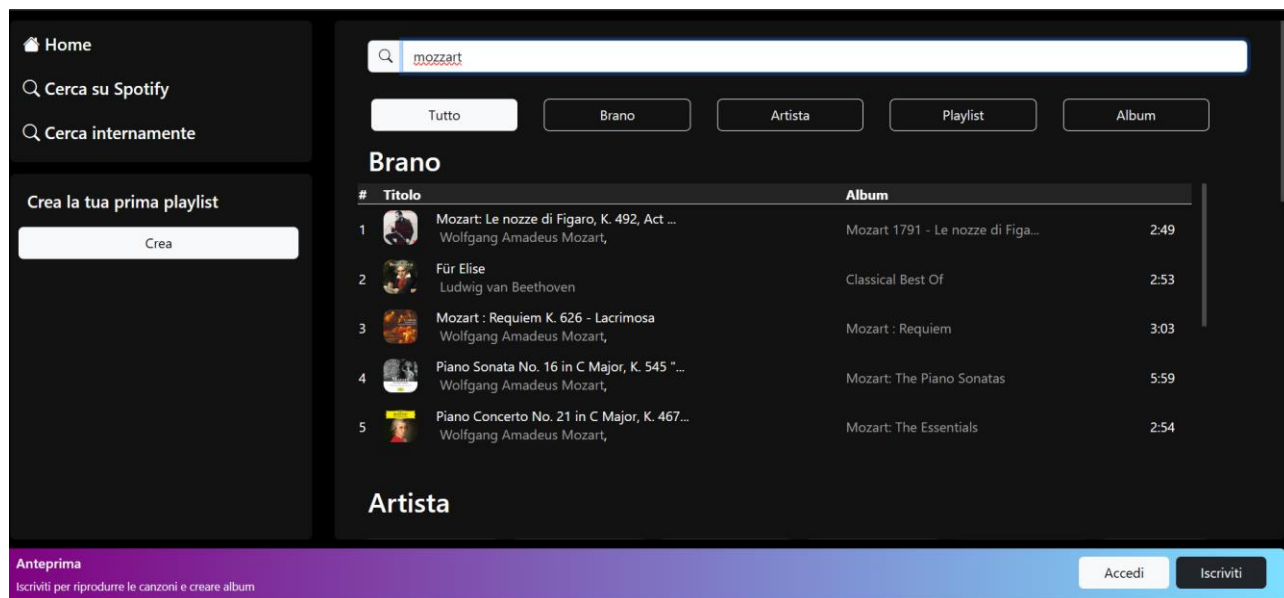


Se il bottone contiene un cuore pieno, significa che abbiamo già inserito l'artista tra i nostri preferiti, altrimenti il cuore risulterà non pieno se non lo è ancora.

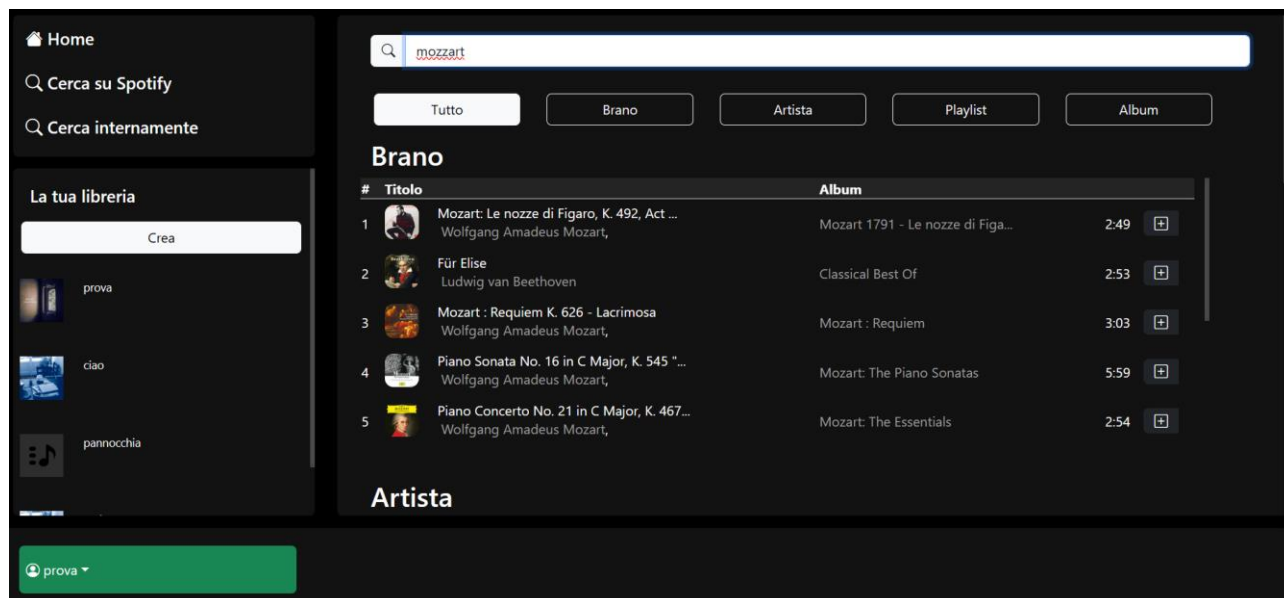
Dalla questa pagina, le informazioni inerenti agli album e singoli rilasciati, verranno mostrati all'interno di una scrollbar orizzontale. In poche parole è come se fosse un div di lunghezza infinita.

5. Cerca su Spotify

Pagina adibita alla ricerca dei vari contenuti all'interno di Spotify. Verranno ricercati i brani, album, artista e playlist che corrispondono al valore inserito nella barra di ricerca. Possiamo anche scegliere di far visualizzare solamente una di queste componenti cliccando sull'apposito tasto.



Per chi è autenticato verrà mostrato, come in album e playlist, il tasto per aggiungere il brano alla propria playlist.

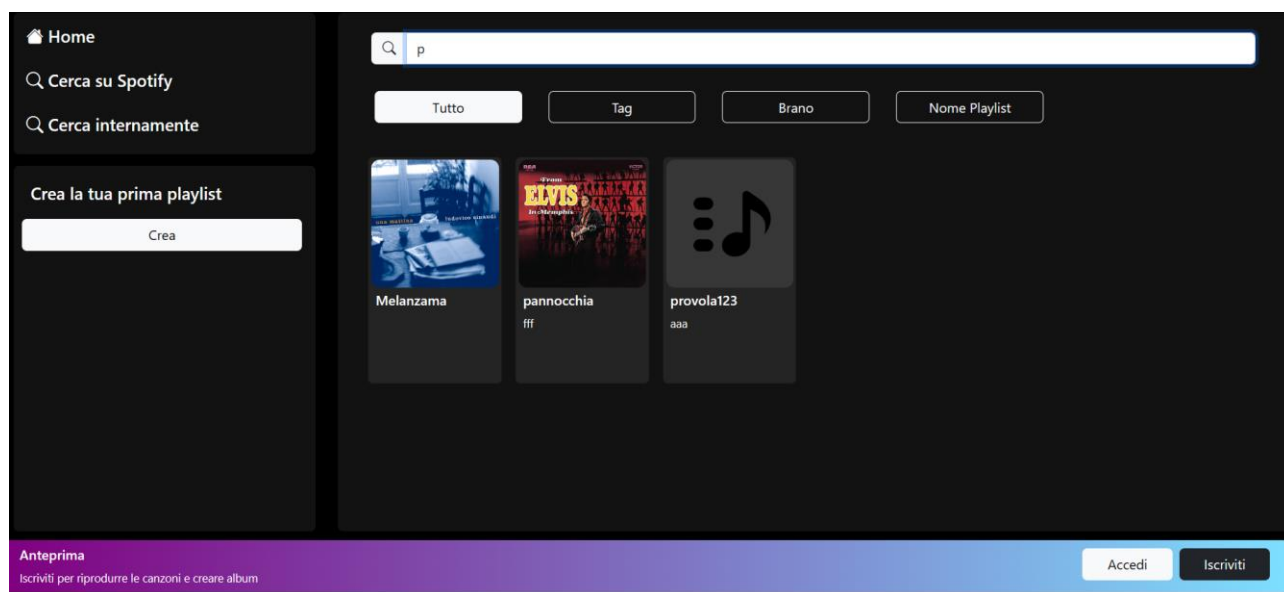


Le richieste di ricerca vengono effettuate attraverso AJAX.

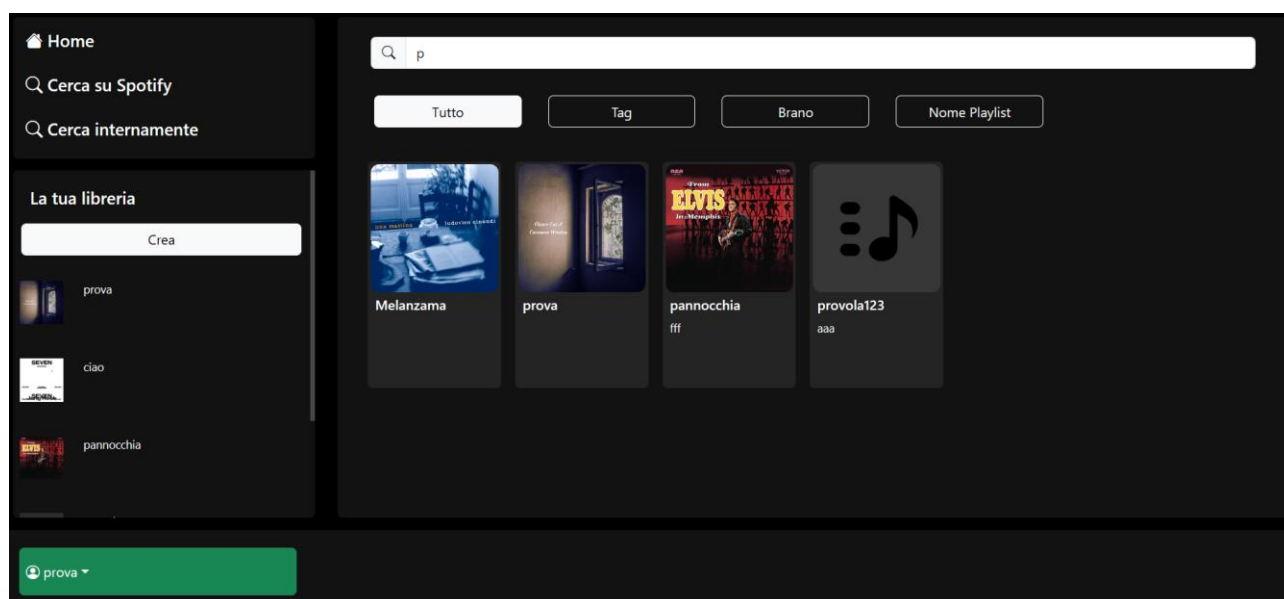
6. Cercare internamente al sito

Questa pagina è come la precedente. L'unica differenza è che la ricerca è adibita solo ed esclusivamente alle playlist interne al sito. Le opzioni che per cui posso eseguire la ricerca sono: Tag, brano e il nome della playlist. Ovviamente possiamo anche cercare in base a tutto. Verranno mostrati unicamente le playlist interne pubbliche, se è privata la playlist, sarà mostrata unicamente dal proprietario della playlist nella ricerca.

Cosa visualizza un utente non autenticato:



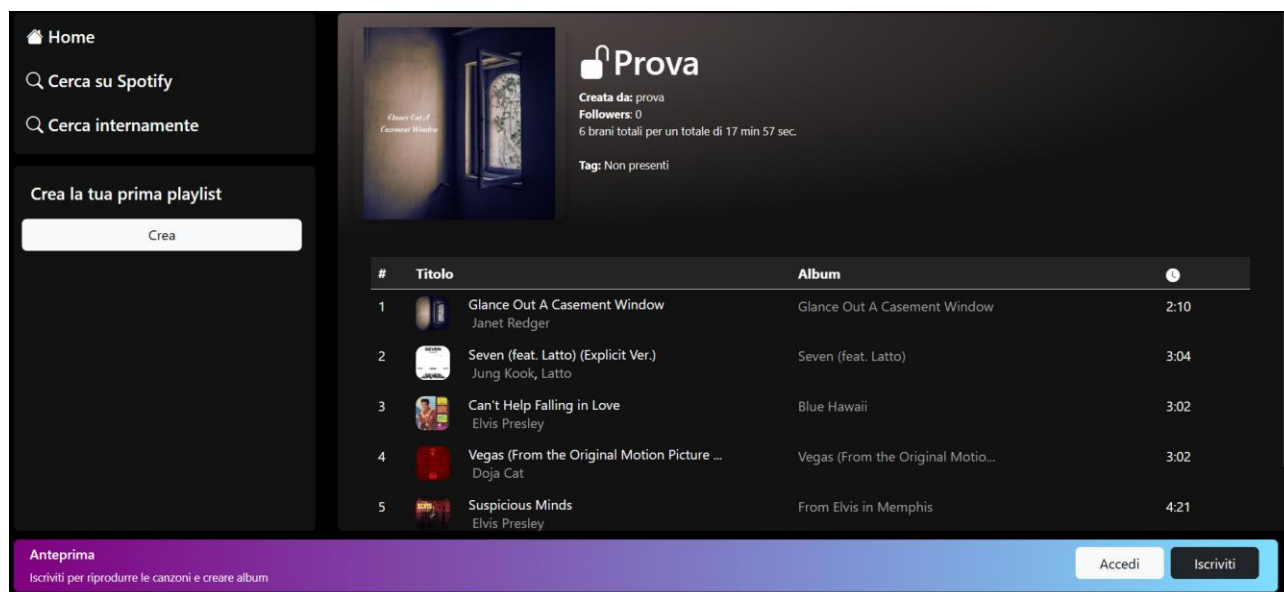
Cosa visualizza l'utente che è proprietario della risorsa privata:



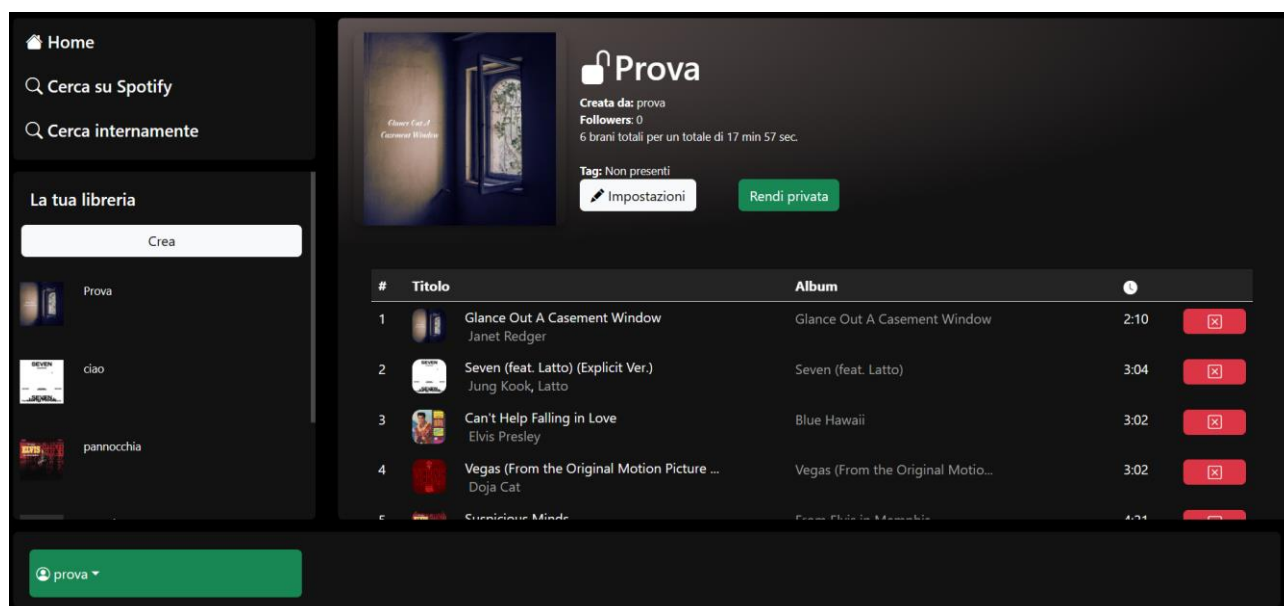
Ricerca sempre con AJAX.

7. Playlist interne

Pagina che mostre le playlist interne. È come la pagina Playlist di Spotify.



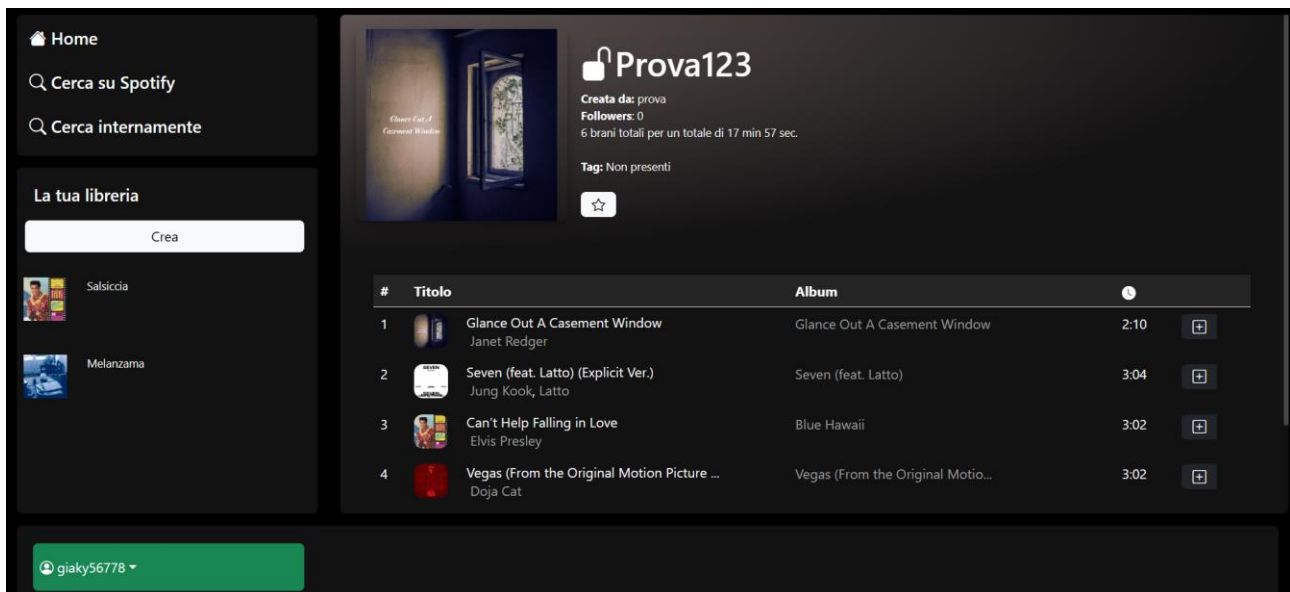
Per gli utenti autenticati, se è il proprietario chi interagisce la playlist verrà mostrata questa schermata dove è possibile cambiare le varie impostazione della playlist.



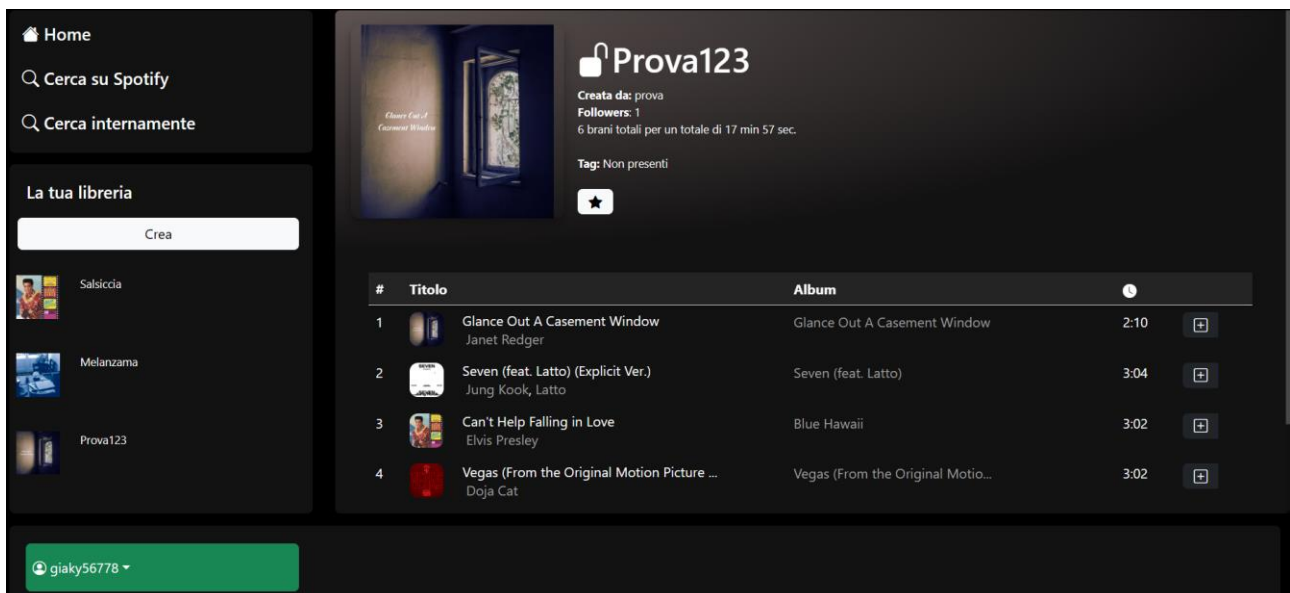
Il tasto rendi privata consente di modificare la playlist da pubblica a privata. Cliccandoci sopra, eventuali followers che seguono la playlist verranno cancellati e non gli verrà visualizzata la playlist nella barra laterale sinistra. I pulsanti in rosso sulle tracce, indicano se vogliamo tenere ancora la traccia nella playlist. Premendola, il brano verrà cancellato e la playlist ricaricata. Se il brano eliminato era il brano che

dava l'immagine alla copertina, la copertina cambierà con il primo brano disponibile alla prima posizione della playlist.

Se l'utente è diverso dal proprietario, verrà mostrata questa schermata



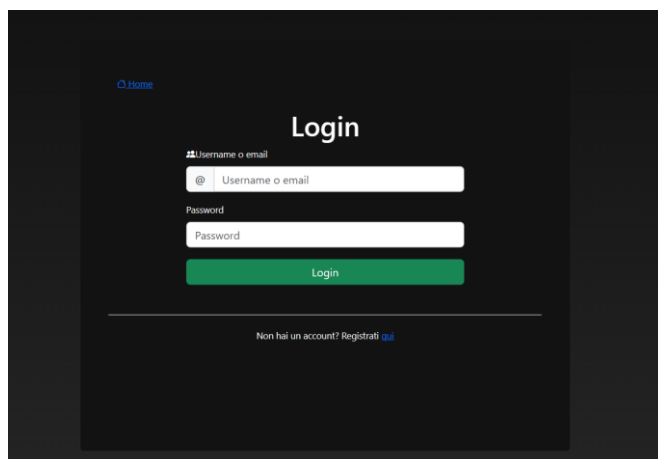
La stellina ha la stessa proprietà del cuore dei followers. In questo caso, se la clicco, verrò segnato come un followers e mi comparirà tra la barra laterale sinistra come si può vedere in questa foto.



Se rischiaccio il pulsante, verrò tolto dai followers e non mi comparirà tra la mia libreria.

8. Login

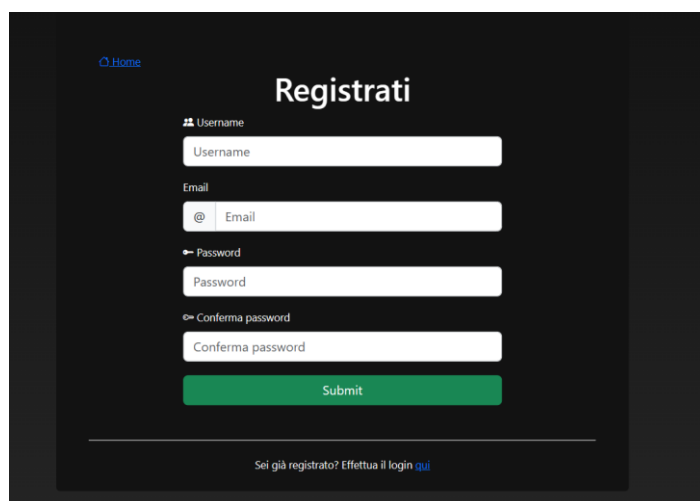
Pagina adibita al login nel sito. Posso autenticarmi o con l'username o con l'email data in fase di registrazione.



The screenshot shows a dark-themed login page. At the top left, there is a 'Home' link with a house icon. The title 'Login' is centered. Below it, there is a label 'Username o email' with a user icon, followed by a text input field containing '@ Username o email'. Below that is a 'Password' label and a text input field containing 'Password'. A green 'Login' button is positioned below the password field. At the bottom, a link says 'Non hai un account? Registrati qui'.

9. Registrati

Pagina adibita alla registrazione dell'account. Verranno fatti dei controlli per verificare se la password e la password di conferma valgono, se l'email è un email (controllata con una espressione regolare) e se l'email e lo username sono univoci all'interno del database. Se non si verifica una di queste condizioni verrà mostrato un errore, e non si registrerà l'account.



The screenshot shows a dark-themed registration page. At the top left, there is a 'Home' link with a house icon. The title 'Registrati' is centered. Below it, there are four input fields: 'Username' (with a user icon), 'Email' (with an '@' icon), 'Password' (with a key icon), and 'Conferma password' (with a double key icon). Each field has its label above it. A green 'Submit' button is located below the 'Conferma password' field. At the bottom, a link says 'Sei già registrato? Effettua il login qui'.

10. Crea playlist interna

Pagina adibita alla creazione di una playlist. È obbligatorio unicamente il nome della playlist. Per l'aggiungere i tag bisogna: scrivere il tag nell'apposito spazio e premere il pulsante aggiungi tag. Sotto al pulsante crea, compariranno i vari tag da inserire all'interno della playlist. Per implementare l'aggiunta dei tag prima della creazione della playlist, i tag verranno salvati in un array dinamico. Se non si desidera più il tag, lo si può rimuovere premendo l'icona "cestino" vicino al tag.

The screenshot shows a web interface for creating a playlist. On the left is a sidebar with a 'Home' link, search options ('Cerca su Spotify', 'Cerca internamente'), and a section titled 'La tua libreria' containing a 'Crea' button and three items: 'Prova123', 'ciao', and 'pannocchia'. The main area is titled 'Playlist' and contains a form with three input fields: 'Nome playlist', 'Descrizione', and a tag input field with a '#' prefix and an 'Aggiungi tag' button. Below the form is a large green 'Crea' button. At the bottom of the sidebar is a green button with a user icon and the text 'prova'.

11. Modifica playlist

Rende possibile modificare la mia playlist. Contrariamente da *crea playlist*, qui i tag che verranno aggiunti o tolti verranno aggiunti e tolti direttamente nel database, non passando da un array temporaneo come in *crea playlist*. La modifica dei parametri comporterà anche l'aggiornamento della barra laterale sinistra.

Home

Cerca su Spotify

Cerca internamente

La tua libreria

Crea

Prova123

ciao

pannocchia

prova

Playlist

Nome playlist

Prova123

Descrizione

Descrizione

Tag

Tag

Aggiungi tag

Modifica

Cancella

12. Cambiare le preferenze del profilo

Pagina abita alla modifica delle preferenze del profilo. Posso cambiare e visualizzare le preferenze che ho messo. Posso anche aggiungerne e toglierle.

Home

Cerca su Spotify

Cerca internamente

La tua libreria

Crea

Prova123

ciao

pannocchia

prova

Artista preferito

Aaron May

Wiz Khalifa

Artista

Genere preferito

rock

black-metal

Genere

Informazioni Personali

Username: prova

Email: prova@prova.it

Cambia dati

Cancella account

Posso anche cancellare il profilo cliccando sull'apposito tasto. Se lo si clicca, verrà invocata una funzione che cancellerà le varie playlist create dall'utente e rimuoverà i follow dalle mie playlist. Operazione necessaria perché il follow della playlist non viene salvato dentro la tabella della playlist, ma dentro al singolo utente, quindi necessario che vengano cancellate, altrimenti risulterà che sto seguendo una playlist inesistente.

13. Aggiorna profilo

Pagina adibita alla modifica dei dati del profilo. Aggiornando o lo username o la password, verrà invocata una funzione che controllerà se l'username o email è disponibile, se disponibile l'aggiorna, altrimenti restituirà errore. La password viene controllata come nella fase di registrazione, se non corrisponde *password* con *conferma password* restituirà errore.

The screenshot displays the 'Aggiorna profilo' (Update profile) interface. The sidebar on the left includes a 'Home' button, search options ('Cerca su Spotify', 'Cerca internamente'), and a 'La tua libreria' (Your library) section with a 'Crea' button and a list of items: 'Prova123', 'ciao', and 'pannocchia'. The main content area is titled 'Aggiorna profilo' and contains the following form fields:

- Username:** Input field with the value 'prova'.
- Email:** Input field with the value 'prova@prova.it'.
- Password:** Input field with the label 'Password'.
- Conferma password:** Input field with the label 'Conferma password'.

A green 'Submit' button is located below the password fields. At the bottom of the page, a green bar displays the current user 'prova' with a dropdown arrow.

SWAGGER

The screenshot displays the Swagger UI for an API named "My API" (version 1.0.0). The interface is organized into several expandable sections, each containing a list of API endpoints with their respective HTTP methods and descriptions. The sections are:

- Principale**:
 - POST `/api/get_san`: Richiede a Spotify
 - GET `/`: Restituisce la home del sito web
- Playlist**:
 - GET `/playlist/{id}`: Cerca e restituisce la pagina delle playlist di Spotify
 - GET `/my_playlist/{id}`: Cerca e restituisce la pagina delle playlist interne di Spotify
 - GET `/crea_playlist`: Restituisce la pagina per la creazione di playlist interne
 - GET `/change_my_playlist/{id}`: Restituisce la pagina per la modifica delle mie playlist interne
 - PUT `/api/create_my_playlist`: Crea una nuova playlist (Autenticazione richiesta)
 - POST `/api/view_my_playlist`: Restituisce le informazioni delle playlist che segue l'ha creato l'account (Autenticazione richiesta)
 - POST `/api/modifica_playlist`: Modifica la playlist interna (Autenticazione richiesta)
 - DELETE `/api/delete_playlist`: Rimuove la playlist interna (Autenticazione richiesta)
 - POST `/api/modifica_privato`: Modifica la playlist interna e la rende o privata o pubblica (Autenticazione richiesta)
 - PUT `/api/add_track`: Aggiunge i brani alla playlist interna (Autenticazione richiesta)
 - DELETE `/api/remove_track`: Rimuove il brano dalla playlist interna (Autenticazione richiesta)
 - POST `/api/visualizza_playlist_follow`: Visualizza la playlist che segue
 - PUT `/api/aggiungi_follow`: Aggiunge un nuovo follower alla playlist interna (Autenticazione richiesta)
 - DELETE `/api/rimuovi_follow`: Rimuove il follow dalla playlist interna (Autenticazione richiesta)
 - POST `/api/count_follower`: Visualizza il numero di followers di una playlist pubblica interna
 - POST `/api/check_follow`: Visualizza il numero di playlist che segue (Autenticazione richiesta)
 - POST `/api/my_playlists`: Visualizza i dati delle playlist pubbliche interne che segue (Autenticazione richiesta)
 - POST `/api/my_playlists_add_track`: Restituisce la playlist che ha creato (Autenticazione richiesta)
- Album**:
 - GET `/album/{id}`: Cerca e restituisce la pagina dell'album di Spotify
- Artista**:
 - GET `/artist/{id}`: Cerca e restituisce la pagina dell'artista di Spotify
 - PUT `/api/add_favorite_artist`: Aggiunge l'artista ai preferiti per quell'account (Autenticazione richiesta)
 - POST `/api/view_favorite_artist`: Guarda gli artisti preferiti per quell'account (Autenticazione richiesta)
 - DELETE `/api/delete_favorite_artist`: Rimuove l'artista ai preferiti per quell'account (Autenticazione richiesta)
- Cerca**:
 - GET `/search`: Cerca e restituisce la pagina dell'artista di Spotify
 - GET `/search-internal`: Restituisce la pagina per la ricerca interna delle playlist
 - POST `/api/search_genre`: Cerca le brani ai generi di Spotify con quello dato
 - POST `/api/cerca_interna`: Effettua una ricerca attraverso: Tag, nome playlist, brani oppure su tutto sulle playlist interne
- User**:
 - GET `/change_profile`: Restituisce la pagina per la modifica delle preferenze dell'utente
 - GET `/change_data`: Restituisce la pagina per la modifica dei dati dell'utente
- Autenticazione**:
 - PUT `/api/register`: Registra l'utente dentro
 - POST `/api/login`: Fase di controllo per il login
- Genere**:
 - PUT `/api/add_genre`: Aggiunge generi ai preferiti (Autenticazione richiesta)
 - POST `/api/view_favorite_genre`: Visualizza i generi preferiti (Autenticazione richiesta)
 - DELETE `/api/delete_favorite_genre`: Cancella il genere preferito (Autenticazione richiesta)
- Account**:
 - POST `/api/account_info`: Restituisce le informazioni dell'account (Autenticazione richiesta)
 - DELETE `/api/remove_account`: Rimuove l'account (Autenticazione richiesta)
 - POST `/api/change_profile`: Cambia i dati dell'account (Autenticazione richiesta)
- Home**:
 - POST `/api/view_suggestion`: Restituisce i brani correlati ai generi e artisti preferiti (Autenticazione richiesta)
- Tag**:
 - DELETE `/api/delete_tag`: Rimuove i tag dalla playlist interna (Autenticazione richiesta)
 - PUT `/api/add_tag`: Aggiunge i tag alla playlist interna (Autenticazione richiesta)

La documentazione del back-end può essere visualizzata all'indirizzo:

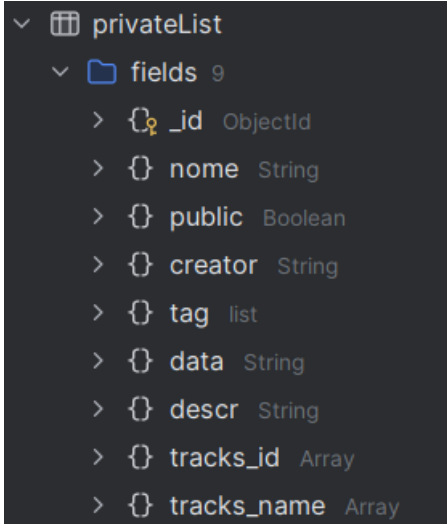
127.0.0.1:PORTA/api-docs

Le varie API sono raggruppate per cosa fanno. Per esempio le API inerenti a playlist servono esclusivamente a gestire l'aspetto delle playlist (sia interne che esterne).

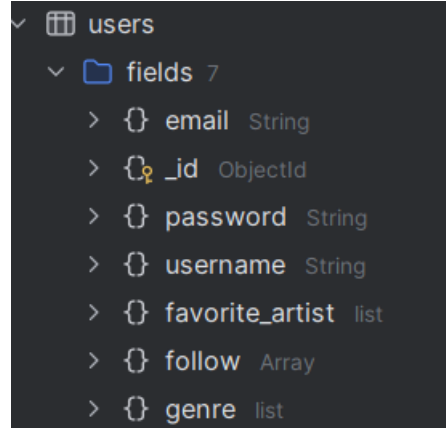
Impostazione delle risorse

1. Database

Gestione delle playlist privata



Gestione degli utenti



privateList:

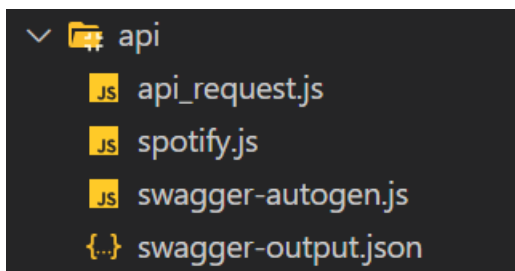
- **nome:** nome della playlist
- **public:** identifica se è pubblica
- **creator:** _id dello user che l'ha creata
- **nome_tracks:** nome delle tracce contenute dalla playlist (utile per velocizzare la ricerca in base al contenuto della playlist)
- **tag:** tag assegnati alla playlist
- **data:** utilizzata unicamente in fase di creazione della playlist. Dopo averla creata, ci sposta in automatico nella pagina della playlist generata. Non salvare questa informazione ci pone davanti al problema di riconoscere qual è l'ultima playlist che abbiamo generato.
- **Tracks:** contiene gli id delle canzoni di Spotify
- **descr:** descrizione della playlist

users:

- **email:** contiene l'email con cui l'utente si è registrato. Deve essere unica.
- **password:** contiene la password in formato MD5 dell'utente.
- **username:** contiene il nome dell'utente. Deve essere unico.

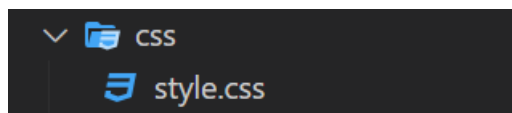
- **favorite_artist:** contiene gli id di Spotify degli artisti preferiti, gli elementi per ogni array devono essere unici, ovvero un utente non può avere lo stesso artista ripetuto più volte.
- **genre:** contiene gli id di Spotify degli artisti preferiti. Vale la stessa proprietà di favorite_artist.
- **follow:** contiene gli _id delle playlist contenute dentro a privateList delle playlist che seguo.

2. Back-end



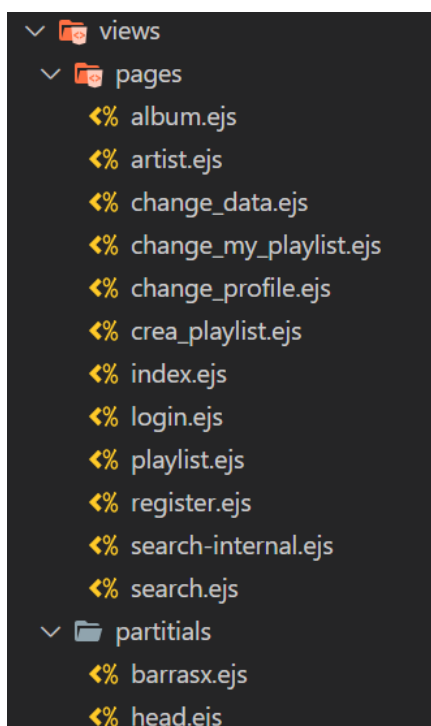
- **api_request.js:** contiene i metodi che interagiscono con il database.
- **spotify.js:** contiene i metodi che interagiscono con le api di Spotify.
- **swagger-autogen.js:** serve a generare le API per tutto il back-end

3. CSS



- **style.css:** contiene il css personalizzato del sito web. La maggior parte dello stile deriva da Bootstrap.

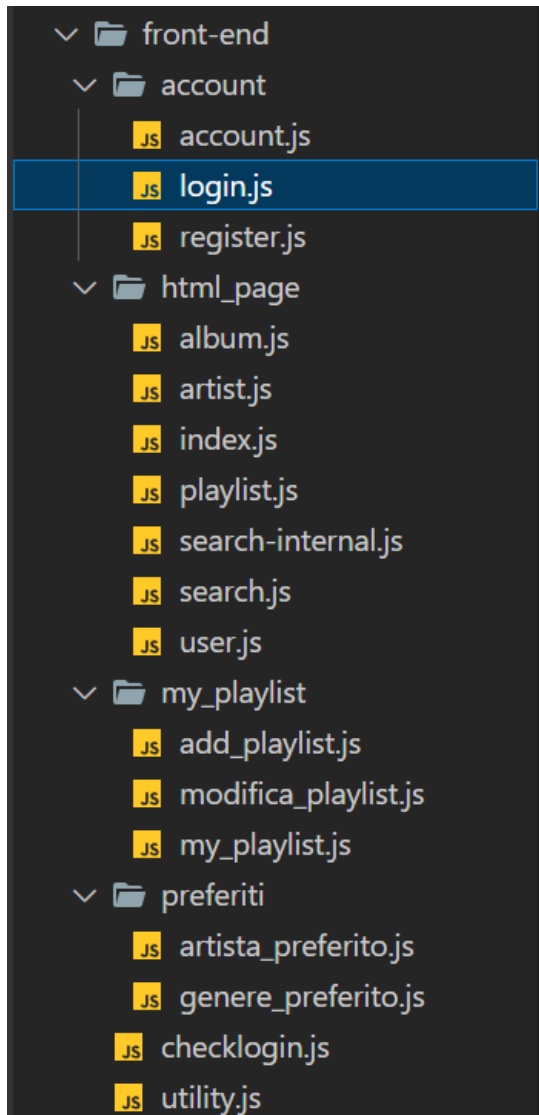
4. Componenti EJS



Contiene i vari template delle varie pagine con i componenti comuni in *partials*.

Ogni file rappresenta una pagina distinta tranne che per *playlist* che verrà mostrato lo stesso layout per 2 pagine diversi, ovvero Playlist di Spotify e Playlist interne.

5. Front-end

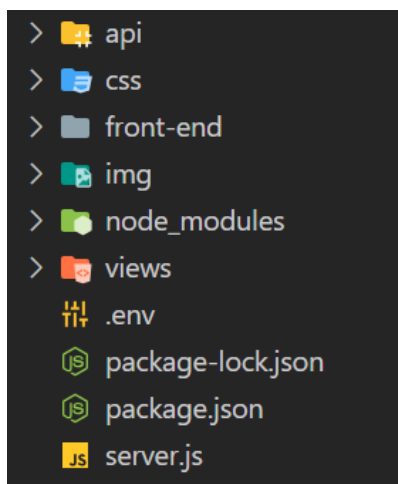


Contiene i vari file raggruppati per categoria.

N.B: i file non sono ordinati per pagina, ma sono ordinati per funzionalità. Esempio: se la pagina per la ricerca richiede di visualizzare i brani ma la visualizzazione è uguale a quella fatta per le playlist, utilizzerò quel metodo per rappresentarlo.

- **account:** contiene i vari file inerenti alla gestione dell'account.
- **html_page:** contiene i file inerenti alla gestione delle varie pagine.
- **my_playlist:** contiene i file inerenti alla gestione delle playlist interne della piattaforma.
- **preferiti:** gestisce le preferenze dei generi e degli artisti di un utente.
- **utility.js:** contiene delle funzioni base per la gestione globale della piattaforma. Viene incluso in tutti le pagine del sito.
- **checklogin.js:** controlla (in solo alcune pagine) se un utente è autenticato o meno. Se non lo è, mostrerà un errore.

6. Struttura finale



Il file principale è *server.js* che contiene le funzioni base per far inizializzare il server NodeJS con l'ausilio del pacchetto Express.

Note implementative

Nelle pagine: Album, Playlist, Playlist interne e Artista lo sfondo viene calcolato a partire dall'immagine attraverso la [fast-average-color](#) che permette di calcolare il colore medio a partire da un'immagine che gli si passa. Lo sfondo verrà rappresentato con un *radial-gradient*.

Il sito web è implementato con un misto tra EJS e DOM. EJS si preoccupa di creare il template della pagina web e viene renderizzato dal server. Il DOM viene utilizzato per visualizzare a schermo i dati che verranno elaborati da JavaScript. Si è scelto di fare in questo modo perché effettuando delle fetch al nostro back-end, il server ci impiegherà diversi secondi a rispondere. Se avessimo generato il sito esclusivamente con EJS, le pagine ci avrebbero impiegato diversi secondi ad essere visualizzate. Nel modo implementato per questo progetto, il caricamento della pagina web non sembrerà bloccato, ma sembrerà più reattivo (anche se le richieste per visualizzare i dati ci impiegheranno diversi secondi a essere completate e visualizzate).

Ogni pagina effettuerà delle fetch al back-end. Se le informazioni richiedono Spotify, verranno utilizzati i metodi dentro al file *spotify.js* che si occupa di instaurare connessione con le API di Spotify generando un token valido e salvato dentro una variabile d'ambiente. Si è scelto di gestire la connessione con Spotify lato server perché altrimenti avremmo esposto le chiavi segrete necessarie per generare un token valido.

Non tutte le richieste necessitano di interagire con Spotify, quelle che non richiedono la sua interazione, interagiscono con il database MongoDB dove saranno contenuti i vari utenti e le varie playlist contenuto nel file *api_request.js*.

Per verificare se un utente è autenticato faremo solamente un controllo se la variabile *account* di *localStorage* è impostata. Se lo è, si ritiene che l'utente è autenticato correttamente. La variabile *account* non è altro che la chiave primaria del record dell'utente dentro MongoDB. È stata scelta questa variabile perché anche se dovessimo cambiare username o email, la chiave primaria non cambia.

Prove di funzionamento

Oltre alle schermate esposte nelle prime pagine di questa relazione, eccone altre di esempi di funzionamento

1. Eliminazione del primo brano di una playlist

Prima:

The screenshot shows the Spotify app interface. On the left, there's a sidebar with 'Home', 'Cerca su Spotify', 'Cerca internamente', and 'La tua libreria'. The main area displays the 'Prova123' playlist, created by 'prova'. It has 5 songs and a total duration of 14 min 52 sec. The first song, 'Glance Out A Casement Window' by Janet Redger, is highlighted. The playlist table is as follows:

#	Titolo	Album		
1	Glance Out A Casement Window Janet Redger	Glance Out A Casement Window	2:10	[X]
2	Can't Help Falling in Love Elvis Presley	Blue Hawaii	3:02	[X]
3	Vegas (From the Original Motion Picture ... Doja Cat	Vegas (From the Original Motio...	3:02	[X]
4	Suspicious Minds Elvis Presley	From Elvis in Memphis	4:21	[X]
5	Hound Dog Elvis Presley	Elvis' Golden Records	2:16	[X]

Dopo:

The screenshot shows the same Spotify app interface, but the first song has been removed. The playlist 'Prova123' now has 4 songs and a total duration of 12 min 42 sec. The playlist table is as follows:

#	Titolo	Album		
1	Can't Help Falling in Love Elvis Presley	Blue Hawaii	3:02	[X]
2	Vegas (From the Original Motion Picture ... Doja Cat	Vegas (From the Original Motio...	3:02	[X]
3	Suspicious Minds Elvis Presley	From Elvis in Memphis	4:21	[X]
4	Hound Dog Elvis Presley	Elvis' Golden Records	2:16	[X]

2. Cambio username con uno già presente

The screenshot displays a web application interface with a dark theme. On the left, a sidebar contains a 'Home' link, search options for Spotify and internal search, and a 'La tua libreria' (Your library) section with a 'Crea' button and three items: 'Prova123', 'ciao', and 'pannocchia'. The main content area is titled 'Indietro' (Back) and features a red error banner at the top stating 'ERRORE, email o utente duplicato' (Error, email or user duplicated). Below this, the 'Aggiorna profilo' (Update profile) form is shown. The form includes fields for 'Username' (giaky56778), 'Email' (prova@prova.it), 'Password', and 'Conferma password'. A green 'Submit' button is at the bottom of the form. A green bar at the very bottom of the page shows a user profile icon and the name 'prova'.

Home

Indietro

ERRORE, email o utente duplicato

Aggiorna profilo

Username
giaky56778

Email
prova@prova.it

Password
Password

Conferma password
Conferma password

Submit

prova