

Opis projektu LunaChord

Jakub Wiczyński, Borys Żurek 4C

1. Główne strony:

Witryna opiera się na trzech głównych stronach:

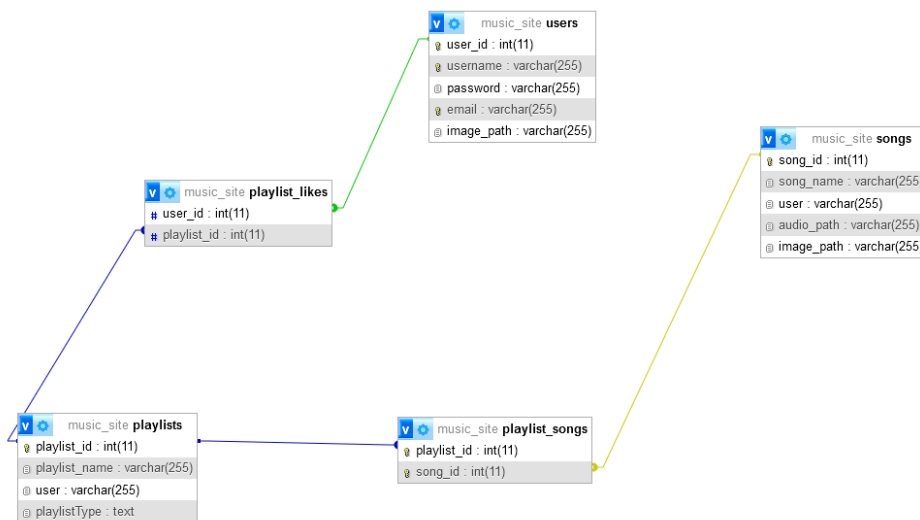
- index.php → odpowiada za wyświetlanie playlist i piosenek
- account.php → odpowiada za wyświetlanie profili użytkowników, ich piosenek oraz playlist
- search.php → odpowiada za wyświetlanie wyników wyszukiwania piosenek, playlist lub użytkownika

Za rejestrację użytkownika oraz logowanie odpowiadają strony register.php oraz login.php

Dodatkowo na każdej stronie zawarte są części:

- bottomTab.html → zawiera interfejs audio
- leftTab.php → zawiera pola na dodawanie playlist oraz piosenek
- rightTab.php → wyświetla utworzone oraz polubione przez użytkownika playlisty
- navbar.php → pasek nawigacyjny służący do przemieszczania się między stronami

2. Struktura bazy danych



Baza danych o nazwie music_site.sql składa się z 5 tabel:

2.1. playlists

Przechowuje informacje o

playlistach

Składa się z pól:

- playlist_id → identyfikator playlisty
- playlist_name → nazwa playlisty
- user → użytkownik, do którego należy playlista
- playlistType → typ playlisty

Playlist może mieć 4 typy:

- Public → widoczna dla innych użytkowników
- Private → widoczna tylko do użytkownika, który ją stworzył
- Local → pozwala użytkownikowi na umieszczenie w niej dowolnych plików audio ze swojego dysku, nie publikując ich na swoim profilu. Także widoczna jest tylko do użytkownika, który ją stworzył
- User → playlista przechowująca opublikowane przez użytkownika piosenki
- Liked → playlista przechowująca polubione przez użytkownika piosenki

2.2 playlist_likes

Służy jako tabela relacyjna, przypisująca playliste (playlist_id) do użytkownika (user_id).

2.3 playlist_songs

Służy jako tabela relacyjna, przypisująca playliste (playlist_id) do piosenki (song_id).

2.4 songs

Przechowuje informacje o piosenkach

Składa się z 5 pól:

- song_id → identyfikator piosenki
- song_name → nazwa piosenki
- user → użytkownik, do którego należy piosenka
- audio_path → ścieżka do pliku audio piosenki (folder audio)
- image_path → ścieżka do obrazka przypisanego do piosenki (folder images/songImages). Jeżeli obrazek nie został podany, do piosenki przypisywany jest obrazek default-song-image.png

2.5 users

Przechowuje informacje o użytkownikach

Składa się z 5 pól:

- user_id → identyfikator użytkownika
- username → nazwa użytkownika
- password → hasło użytkownika (zahashowane)
- email → adres email użytkownika
- image_path → ikona użytkownika. Jeżeli nie została podana, do użytkownika przypisywany jest obrazek default-user-icon.png

3. Opis działania strony.

Kiedy użytkownik zostaje zarejestrowany, automatycznie tworzon jest przypisana do niego playlista typu "Liked" oraz "User", aby użytkownik mógł publikować piosenki oraz umieszczać piosenki innych użytkowników w polubionych utworach. Po zalogowaniu użytkownik ma zarówno opcje

tworzenia playlist, zmieniania ich nazw oraz ich usuwania, jak i umieszczania piosenek swój profil lub do lokalnych playlist, zmieniania ich nazw, dodawania oraz usuwania ich z playlist.

Użytkownik ma również opcję usunięcia swojego konta. Odtwarzanie piosenek możliwe jest za pomocą JavaScript (audioPlayer.js). Timer odliczający czas piosenki działa za pomocą biblioteki easyTimer.js (<https://albert-gonzalez.github.io/easytimer.js/>). Aby zapewnić płynne odtwarzanie piosenek między stronami oraz playlistami, witryna korzysta z technologii htmx (folder htmx). W bazie danych utworzone zostało także konto administratora (login: admin, hasło: admin), które ma uprawnienia zarówno do usuwania użytkowników, jak i do usuwania oraz zmiany nazw piosenek.

Za logikę, na podstawie której funkcjonuje witryna odpowiada plik db.php. Utworzone w nim zostały 3 klasy:

- **Klasa abstrakcyjna Db_Connection** zawierająca metody połączenia z bazą danych
- **Klasa SQL_Functions** zawierająca metody interakcji z bazą danych za pomocą SQL (SELECT, INSERT, DELETE, ALTER, UPDATE)
- **Klasa HTML_Display_Functions** zawierająca metody wyświetlania wyników kwerend w HTML

4. Podział pracy

- Jakub Wiczyński → układ stron w HTML, podstawowy desingn witryny w CSS, logika PHP, na której opiera się działanie witryny, implementacja technologii htmx, logika JS, na której opiera się odtwarzanie plików audio (z wyjątkiem plików simpleTimer.js),
- Borys Żurek → znaczne rozwinięcie css zarówno dla strony głównej jak i pomniejszych jak logowania czy rejestracji (pomimo, że znacznej części zmian nie ma w commitach, ponieważ zostały one wysłane do Jakuba by commitować je wraz z innymi zmianami), polepszenie funkcjonalności kontroli audio (poziom głośności, slider do kontrolowania momentu odtwarzania i przyciski)