



CS 353 DATABASE SYSTEMS

2017-2018 SPRING SEMESTER

## **MUSICHOLICS**

PROJECT FINAL REPORT

### GROUP 17

Esra Nur AYAZ

Ezgi ÇAKIR

Metehan KAYA

Miraç Vuslat Başaran

## **1. Project Description**

Musicholics is a web-based music service allowing users to stream and buy music online anywhere. Musicholics aims to be a breath of fresh air to the field by combining music streaming and social life. Musicholics users can be friend of each other, post on others' wall, and gift songs to their loved ones. Unlike currently available music services and social media sites, Musicholics brings them together and provides an innovative way to have fun.

To use Musicholics, first, a person has to sign up by entering some typical information such as username, password, email address, etc. If she registered in the past, she needs to sign in. After registration or sign in, the user may start listening to music. She needs to add money to her budget, then purchase tracks with her budget. She can search for her friends and be friends with them. Moreover, she can view their playlists and follow, rate, or comment on them. She is able to post something of her friends' profile wall. If she wants she can block a user, also.

A user can have a premium account which enables the user to listen to music even if she is offline. Upgrading the account to premium is optional and have a cost. If a user wants to add track and other people to listen, she can upgrade to artist user type. It has a cost, too.

Also, the user can buy a track or send it to a friend as a gift.

A user has an opportunity to create a new playlist which is composed of purchased tracks. This saves the user's time from searching music and adding them to a new playlist.

Musicholics should provide access to detailed information of tracks, artist, albums, and publishers.

### **Contribution of Group Members**

All pages are distributed evenly to each group of members. They were responsible of all implementation of those given pages.

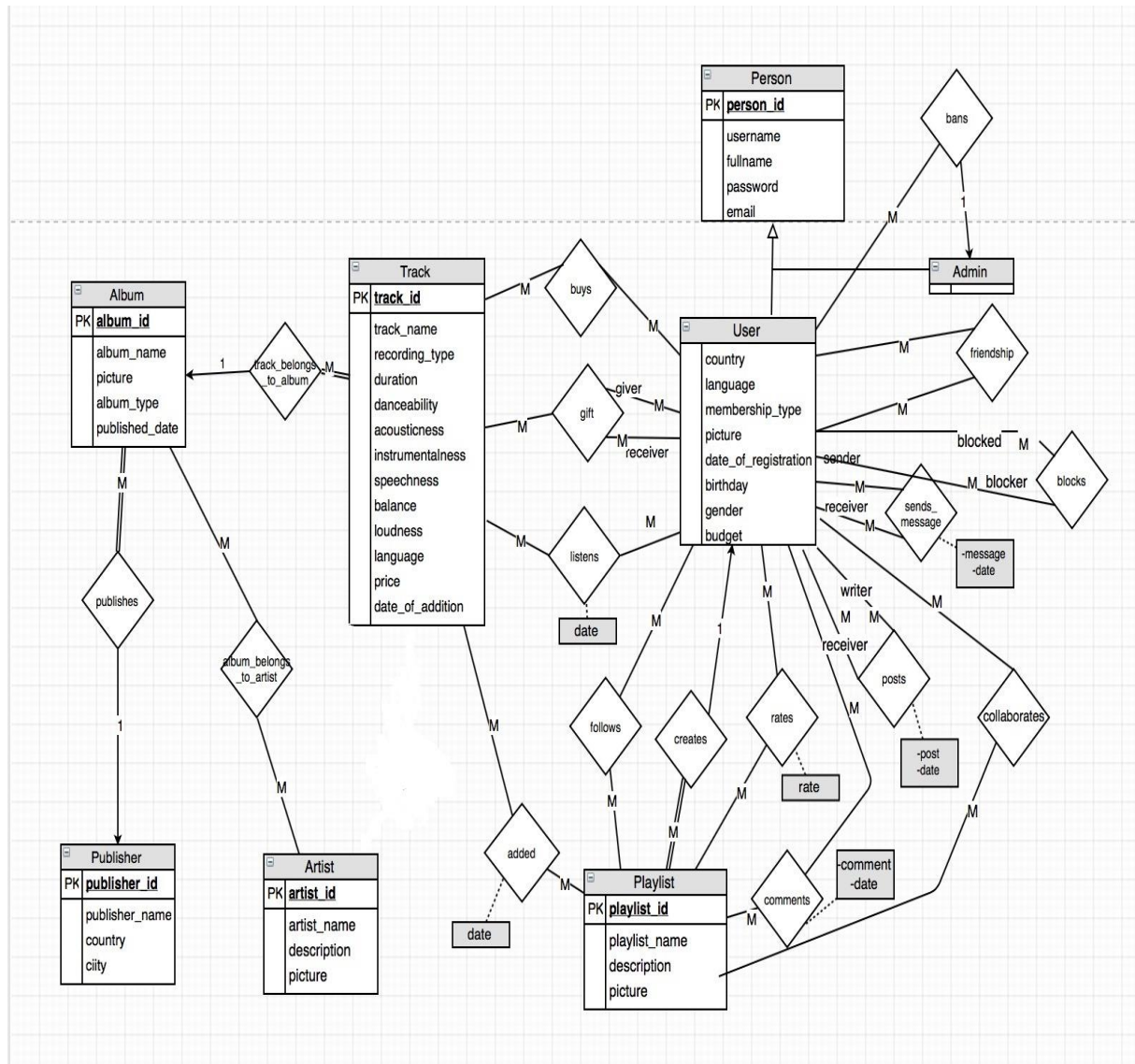
Ezgi implemented homepage, profile settings and change password screens for user, all profile views of any user such as a user's own profile, friends profile, non-friends and non-blocked profiles, a user's profile that is blocked by logged user, and user's profile viewed by admin pages. Additionally, she implemented all purchase screens such as purchased budget with credit card, upgrading to artist or premium with budget, and purchase tracks with budget screens.

Esra implemented sign in and sign up screens, all search screens such as general search, detailed search for user, artist, track, album, and playlist, friends list view of a user, and admin panel.

Metehan implemented views of tracks, playlists in detailed, view friends' playlists, those playlists in detailed, message screen of a user, send message screens and modify playlist screen.

Miraç Vuslat implemented tracks view by admin and modify track, publisher, and artists, screen, view screens of publisher, artists, album. Additionally, he implemented stored procedures and triggers.

## 2. E/R diagram



We deleted “Track\_Belongs\_To\_Artist” relation from the Design Report because it unnecessarily complicated insertions and deletions of tracks.

### 3. Tables

#### 3.1. Person

```
CREATE TABLE IF NOT EXISTS Person (  
    person_id INT NOT NULL AUTO_INCREMENT,  
    username VARCHAR(45) NOT NULL,  
    fullname VARCHAR(45),  
    password VARCHAR(45) NOT NULL,  
    email VARCHAR(128) NOT NULL,  
    PRIMARY KEY(person_id),  
    UNIQUE(username),  
    UNIQUE(email) )
```

#### 3.2. Admin

```
CREATE TABLE IF NOT EXISTS Admin (  
    admin_id INT NOT NULL,  
    PRIMARY KEY(admin_id),  
    FOREIGN KEY(admin_id) REFERENCES Person(person_id) )
```

#### 3.3. User

```
CREATE TABLE IF NOT EXISTS User (  
    user_id INT NOT NULL,  
    country VARCHAR(45),  
    language VARCHAR(45),  
    picture VARCHAR(1024),  
    date_of_registration DATE,  
    membership_type VARCHAR(30),  
    birthday DATE,  
    gender VARCHAR(10),  
    budget FLOAT(8, 3),  
    PRIMARY KEY(user_id),  
    FOREIGN KEY(user_id) REFERENCES Person(person_id) )
```

### 3.4. Publisher

```
CREATE TABLE IF NOT EXISTS Publisher (  
    publisher_id INT NOT NULL AUTO_INCREMENT,  
    publisher_name VARCHAR(45) NOT NULL UNIQUE,  
    country VARCHAR(45) NULL,  
    city VARCHAR(45) NULL,  
    PRIMARY KEY(publisher_id) )
```

### 3.5. Album

```
CREATE TABLE IF NOT EXISTS Album (  
    album_id INT NOT NULL AUTO_INCREMENT,  
    album_name VARCHAR(45) NOT NULL,  
    picture VARCHAR(1024),  
    album_type VARCHAR(40) ,  
    published_date DATE NOT NULL,  
    publisher_id INT NOT NULL,  
    PRIMARY KEY(album_id),  
    FOREIGN KEY(publisher_id) REFERENCES publisher(publisher_id) )
```

### 3.6. Artist

```
CREATE TABLE IF NOT EXISTS Artist (  
    artist_id INT NOT NULL AUTO_INCREMENT,  
    artist_name VARCHAR(45) NOT NULL,  
    description VARCHAR(2048),  
    picture VARCHAR(1024),  
    PRIMARY KEY(artist_id) )
```

### 3.7. Track

```
CREATE TABLE IF NOT EXISTS Track (  
    track_id INT NOT NULL AUTO_INCREMENT,  
    track_name VARCHAR(45) NOT NULL,  
    recording_type VARCHAR(30),  
    duration TIME NOT NULL,  
    danceability FLOAT(4,3),
```

```

acousticness FLOAT(4,3),
instrumentalness FLOAT(4,3),
speechness FLOAT(4,3),
balance FLOAT(4,3),
loudness FLOAT(4,3),
language VARCHAR(45),
price FLOAT (6,3) NOT NULL,
date_of_addition DATE NOT NULL,
album_id INT NOT NULL,
PRIMARY KEY(track_id),
FOREIGN KEY(album_id) REFERENCES album(album_id) )

```

### **3.8. Album\_Belongs\_To\_Artist**

```

CREATE TABLE IF NOT EXISTS Album_Belongs_To_Artist (
    artist_id INT NOT NULL,
    album_id INT NOT NULL,
    PRIMARY KEY(artist_id, album_id),
    FOREIGN KEY(artist_id) REFERENCES artist(artist_id),
    FOREIGN KEY(album_id) REFERENCES album(album_id) )

```

### **3.9. Playlist**

```

CREATE TABLE IF NOT EXISTS Playlist (
    playlist_id INT NOT NULL AUTO_INCREMENT,
    playlist_name VARCHAR(45) NOT NULL,
    description VARCHAR(2048),
    picture VARCHAR(1024),
    creator_id INT NOT NULL,
    date DATE NOT NULL,
    PRIMARY KEY(playlist_id),
    FOREIGN KEY(creator_id) REFERENCES user(user_id))

```

### **3.10. Added**

```

CREATE TABLE IF NOT EXISTS Added (
    playlist_id INT NOT NULL,

```

```

track_id INT NOT NULL,

date DATE NOT NULL,

PRIMARY KEY(playlist_id, track_id),

FOREIGN KEY(track_id) REFERENCES track(track_id),

FOREIGN KEY(playlist_id) REFERENCES playlist(playlist_id) )

```

### **3.11. Buys**

```

CREATE TABLE IF NOT EXISTS Buys (

    user_id INT NOT NULL,

    track_id INT NOT NULL,

    PRIMARY KEY(user_id, track_id),

    FOREIGN KEY(user_id) REFERENCES User(user_id),

    FOREIGN KEY(track_id) REFERENCES Track(track_id) )

```

### **3.12. Gift**

```

CREATE TABLE IF NOT EXISTS Gift (

    giver_id INT NOT NULL,

    receiver_id INT NOT NULL,

    track_id INT NOT NULL,

    PRIMARY KEY(giver_id, receiver_id, track_id),

    FOREIGN KEY(giver_id) REFERENCES User(user_id),

    FOREIGN KEY(receiver_id) REFERENCES User(user_id),

    FOREIGN KEY(track_id) REFERENCES Track(track_id) )

```

### **3.13. Listens**

```

CREATE TABLE IF NOT EXISTS Listens (

    user_id INT NOT NULL,

    track_id INT NOT NULL,

    date TIMESTAMP NOT NULL,

    PRIMARY KEY(user_id, track_id, date),

    FOREIGN KEY(user_id) REFERENCES User(user_id),

    FOREIGN KEY(track_id) REFERENCES Track(track_id) )

```

### **3.14. Bans**

```

CREATE TABLE IF NOT EXISTS Bans (

```

```

user_id INT NOT NULL,
admin_id INT NOT NULL,
PRIMARY KEY(user_id),
FOREIGN KEY(user_id) REFERENCES User(user_id),
FOREIGN KEY(admin_id) REFERENCES Admin(admin_id) )

```

### **3.15. Follows**

```

CREATE TABLE IF NOT EXISTS Follows (
    user_id INT NOT NULL,
    playlist_id INT NOT NULL,
    PRIMARY KEY(user_id, playlist_id) ,
    FOREIGN KEY(user_id) REFERENCES User(user_id),
    FOREIGN KEY(playlist_id) REFERENCES Playlist(playlist_id) )

```

### **3.16. Collaborates**

```

CREATE TABLE IF NOT EXISTS Collaborates (
    user_id INT NOT NULL,
    playlist_id INT NOT NULL,
    PRIMARY KEY(user_id, playlist_id) ,
    FOREIGN KEY(user_id) REFERENCES User(user_id),
    FOREIGN KEY(playlist_id) REFERENCES Playlist(playlist_id) )

```

### **3.17. Rates**

```

CREATE TABLE IF NOT EXISTS Rates (
    user_id INT NOT NULL,
    playlist_id INT NOT NULL,
    rate INT NOT NULL,
    PRIMARY KEY(user_id, playlist_id),
    FOREIGN KEY(user_id) REFERENCES user(user_id),
    FOREIGN KEY(playlist_id) REFERENCES playlist(playlist_id) )

```

### **3.18. Comments**

```

CREATE TABLE IF NOT EXISTS Comments (
    comment_id INT NOT NULL AUTO_INCREMENT,
    user_id INT NOT NULL,

```



```

playlist_id INT NOT NULL,
comment VARCHAR(2048) NOT NULL,
date TIMESTAMP NOT NULL,
PRIMARY KEY(comment_id) ,
FOREIGN KEY(user_id) REFERENCES user(user_id),
FOREIGN KEY(playlist_id) REFERENCES playlist(playlist_id) )

```

### 3.19. Friendship

```

CREATE TABLE IF NOT EXISTS Friendship (
    user1_id INT NOT NULL,
    user2_id INT NOT NULL,
    PRIMARY KEY(user1_id, user2_id) ,
    FOREIGN KEY(user1_id) REFERENCES user(user_id),
    FOREIGN KEY(user2_id) REFERENCES user(user_id) )

```

### 3.20. Blocks

```

CREATE TABLE IF NOT EXISTS Blocks (
    blocker_id INT NOT NULL,
    blocked_id INT NOT NULL,
    PRIMARY KEY(blocker_id, blocked_id) ,
    FOREIGN KEY(blocker_id) REFERENCES user(user_id),
    FOREIGN KEY(blocked_id) REFERENCES user(user_id) )

```

### 3.21. Sends\_Message

```

CREATE TABLE IF NOT EXISTS Sends_Message (
    message_id INT NOT NULL AUTO_INCREMENT,
    sender_id INT NOT NULL,
    receiver_id INT NOT NULL,
    date TIMESTAMP NOT NULL,
    message VARCHAR(2048) NOT NULL,
    PRIMARY KEY(message_id) ,
    FOREIGN KEY(sender_id) REFERENCES user(user_id),
    FOREIGN KEY(receiver_id) REFERENCES user(user_id) )

```

### 3.22. Posts

```
CREATE TABLE IF NOT EXISTS Posts (  
    post_id INT NOT NULL AUTO_INCREMENT,  
    writer_id INT NOT NULL,  
    receiver_id INT NOT NULL,  
    date TIMESTAMP NOT NULL,  
    post VARCHAR(2048) NOT NULL,  
    PRIMARY KEY(post_id) ,  
    FOREIGN KEY(writer_id) REFERENCES user(user_id),  
    FOREIGN KEY(receiver_id) REFERENCES user(user_id) )
```

## 4. Implementation Details

Our project, Musicholics, needed a database, and website. We implemented our database system using MySQL on XAMPP. HTML is used for front-end of Musicholics. Graphical User Interface was set up by HTML elements. We did not use CSS. Some JavaScript script was used for alerting users of errors. PHP is used to establish communication between our database and front-end part of our system. A git repository was used for code development.

The only problem we have face is image upload. We could not manage to upload images to the database. It is most likely to be a permission issue.

## 5. Advanced DB Features

### Triggers

We used to a trigger to insert an entry to “buys” relation after a user gifts a song to someone.

```
CREATE TRIGGER gift_to_buy  
AFTER INSERT ON Gift FOR EACH ROW  
BEGIN  
    INSERT INTO Buys (user_id, track_id)  
        VALUES(NEW.receiver_id, NEW.track_id);  
END
```

### Stored Procedures

We defined a stored procedure to delete a track with a given track\_id.

```
CREATE PROCEDURE DeleteTrack(IN trackId INT)  
BEGIN
```

```

DELETE FROM Added WHERE track_id = trackId;

DELETE FROM Buys WHERE track_id = trackId;

DELETE FROM Gift WHERE track_id = trackId;

DELETE FROM Listens WHERE track_id = trackId;

DELETE FROM Track WHERE track_id = trackId;

END

```

We defined a stored procedure to delete an album, which uses the DeleteTrack() procedure we defined earlier.

```

CREATE PROCEDURE DeleteAlbum(IN albumId INT)
BEGIN
    DECLARE done INT DEFAULT FALSE;
    DECLARE i INT;
    DECLARE cur_tracks CURSOR FOR SELECT track_id FROM TRACK WHERE album_id = albumId;
    DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = TRUE;

    DELETE FROM Album_Belongs_To_Artist WHERE album_id = albumId;

    OPEN cur_tracks;
    track_loop: LOOP
        FETCH cur_tracks INTO i;
        IF done THEN
            LEAVE track_loop;
        END IF;
        CALL DeleteTrack(i);
    END LOOP;
    CLOSE cur_tracks;
    DELETE FROM Album WHERE album_id = albumId;

END

```

We defined a DeleteAlbumFromArtist Procedure to delete an album from an artist's albums. The difference between this procedure and DeleteAlbum() is that if the album belongs to more than one artists, DeleteAlbumFromArtist Procedure doesn't delete the album, it only removes the album from artist's albums.

```

CREATE PROCEDURE DeleteAlbumFromArtist(IN albumId INT, IN artistId INT)
BEGIN
    DECLARE numElements INT;
    DELETE FROM Album_Belongs_To_Artist WHERE album_id = albumId AND artist_id = artistId;
    SET numElements = (SELECT COUNT(*) FROM Album_Belongs_To_Artist WHERE album_id = albumId);
    IF numElements = 0 THEN CALL DeleteAlbum(albumId);
    END IF;

```

END

We defined a procedure to delete an artist, which uses DeleteAlbumFromArtist Procedure from before.

```
CREATE PROCEDURE DeleteArtist(IN artistId INT)
```

```
BEGIN
```

```
    DECLARE done INT DEFAULT FALSE;
```

```
    DECLARE i INT;
```

```
    DECLARE cur_albums CURSOR FOR SELECT album_id FROM Album_Belongs_To_Artist WHERE artist_id = artistId;
```

```
    DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = TRUE;
```

```
    OPEN cur_albums;
```

```
    album_loop: LOOP
```

```
        FETCH cur_albums INTO i;
```

```
        IF done THEN
```

```
            LEAVE album_loop;
```

```
        END IF;
```

```
        CALL DeleteAlbumFromArtist(i, artistId);
```

```
    END LOOP;
```

```
    CLOSE cur_albums;
```

```
    DELETE FROM Artist WHERE artist_id = artistId;
```

```
END
```

Finally, we defined a procedure to delete a publisher, which uses the DeleteAlbum Procedure from before.

```
CREATE PROCEDURE DeletePublisher(IN publisherId INT)
```

```
BEGIN
```

```
    DECLARE done INT DEFAULT FALSE;
```

```
    DECLARE i INT;
```

```
    DECLARE cur_albums CURSOR FOR SELECT album_id FROM Album WHERE publisher_id = publisherId;
```

```
    DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = TRUE;
```

```
    OPEN cur_albums;
```

```
    album_loop: LOOP
```

```
        FETCH cur_albums INTO i;
```

```
        IF done THEN
```

```
            LEAVE album_loop;
```

```
        END IF;
```

```
        CALL DeleteAlbum(i);
```

```
    END LOOP;
```

```
    CLOSE cur_albums;
```

```
DELETE FROM Publisher WHERE publisher_id = publisherId;
```

END

## Reports

Report for displaying the user who listens to maximum number of songs:

```
$max_listener_num_listens := SELECT MAX(num) AS num_listens FROM ((SELECT user_id, COUNT(*) AS num FROM Listens GROUP BY user_id) AS temp);
```

```
SELECT user_id, num FROM ((SELECT user_id, COUNT(*) AS num FROM Listens GROUP BY user_id) AS temp) WHERE num = $max_listener_num_listens;
```

Output: 4 15

Report for displaying the user who purchases songs the most:

```
$max_buyer_num_buys := SELECT MAX(num) AS num_listens FROM ((SELECT user_id, COUNT(*) AS num FROM Buys GROUP BY user_id) AS temp);
```

```
SELECT user_id, num FROM ((SELECT user_id, COUNT(*) AS num FROM Buys GROUP BY user_id) AS temp) WHERE num = $max_buyer_num_buys;
```

Output: 3 10

Report for displaying the richest user:

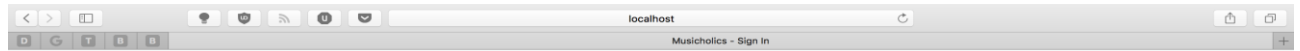
```
$richest_budget := SELECT MAX(budget) AS budget FROM USER;
```

```
SELECT user_id, budget FROM USER WHERE budget = $richest_budget;
```

Output: 750.000

## 6. User Manual

### Login



### Login

One must be a registered user to use Musicholics. You will use your username and password you have chosen. If you are not a registered user or username or password you typed is invalid, you will get an error message.

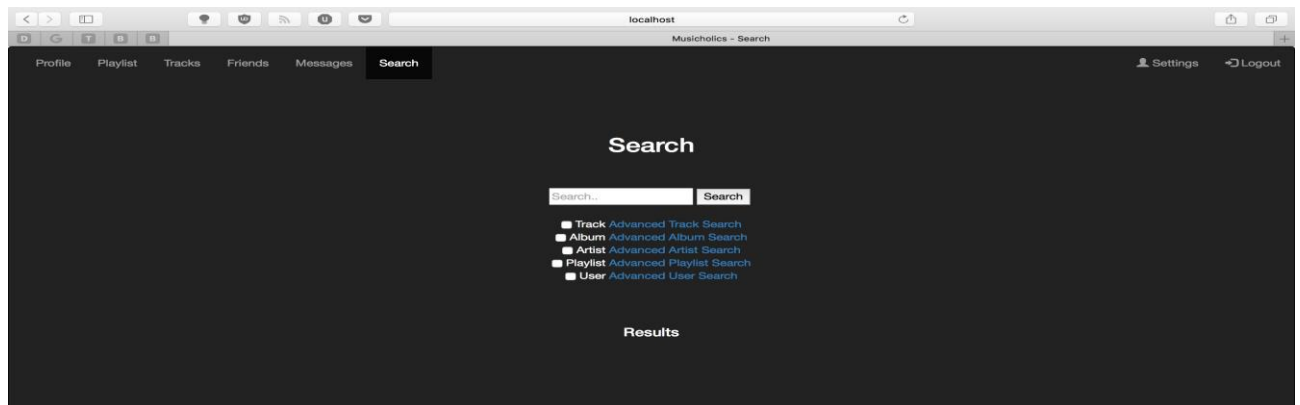
### Register



### Register

Musicholics needs your full name, username, a password, e-mail address, country, language, birthday, and gender. Empty entries will cause an error message.

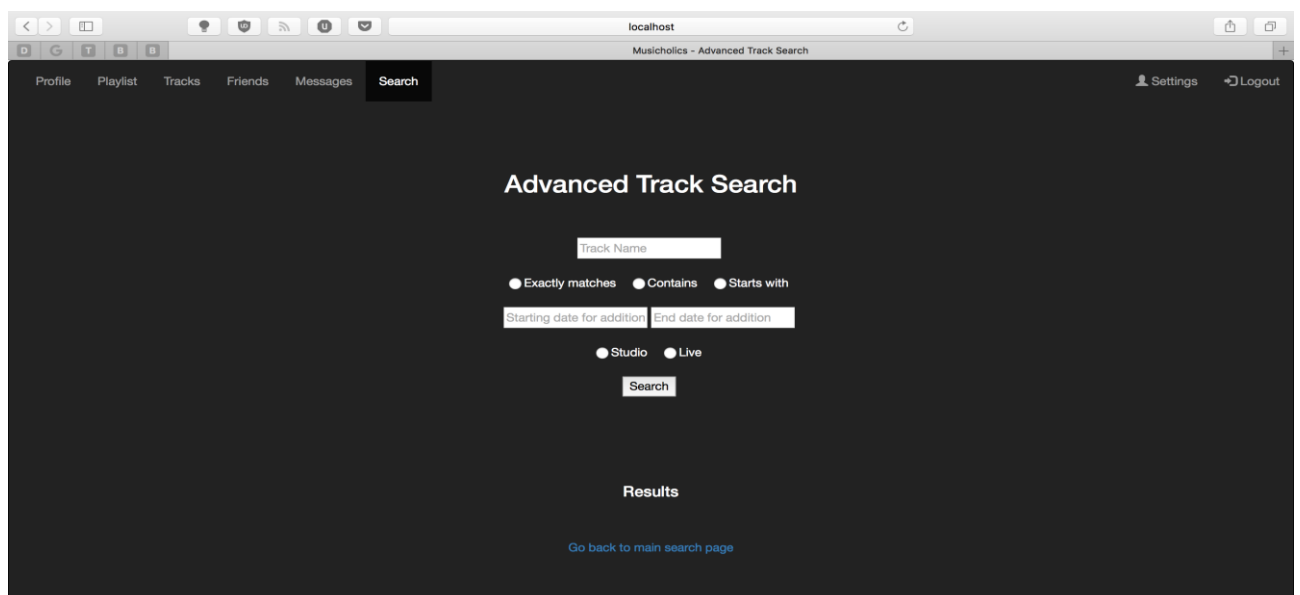
## Search screen for users



Tastamam (00:00:00 )

You can search for tracks, albums, artist, playlists, and users. It is possible to search for only one option or any number of options. For user search, you will not see people who you have blocked. If you wish, you can go to advanced search pages and search for detailed results.

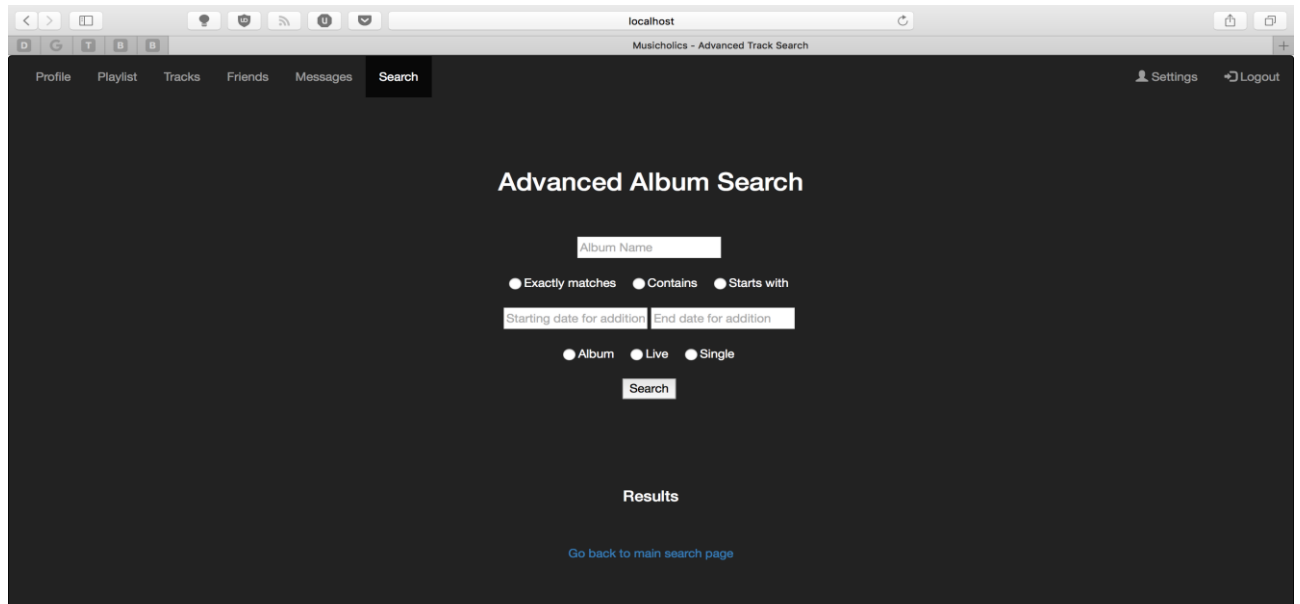
## Advanced track search for users



Tastamam (00:00:00 )

You can search for tracks using track name, date of addition, and track type. You can go back to main search page using the link at the bottom of the page.

### Advanced album search for users

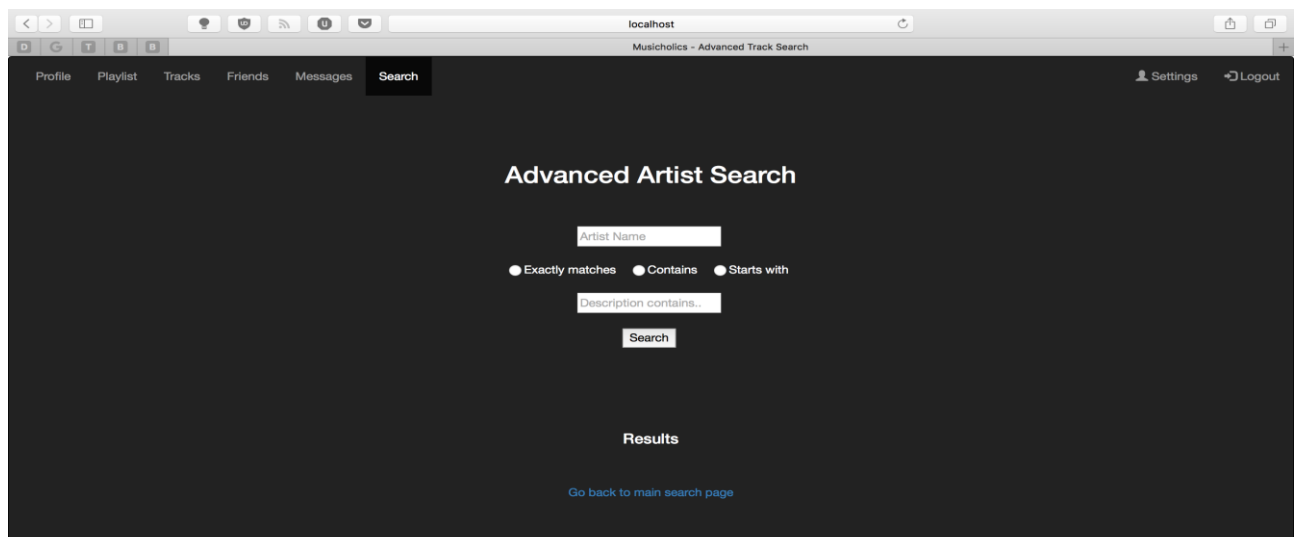


The screenshot shows a web browser window with the address bar set to localhost. The page title is "Musicholics - Advanced Track Search". The navigation bar includes links for Profile, Playlist, Tracks, Friends, Messages, and Search (which is highlighted). In the top right corner, there are links for Settings and Logout. The main content area is titled "Advanced Album Search". It features a text input field for "Album Name", three radio buttons for "Exactly matches", "Contains", and "Starts with", two text input fields for "Starting date for addition" and "End date for addition", and three radio buttons for "Album", "Live", and "Single". A "Search" button is located below these options. Under the heading "Results", there is a link that says "Go back to main search page".

Tastamam (00:00:00 )

You can search for albums using album name, date of addition, and album type. You can go back to main search page using the link at the bottom of the page.

### Advanced artist search for users



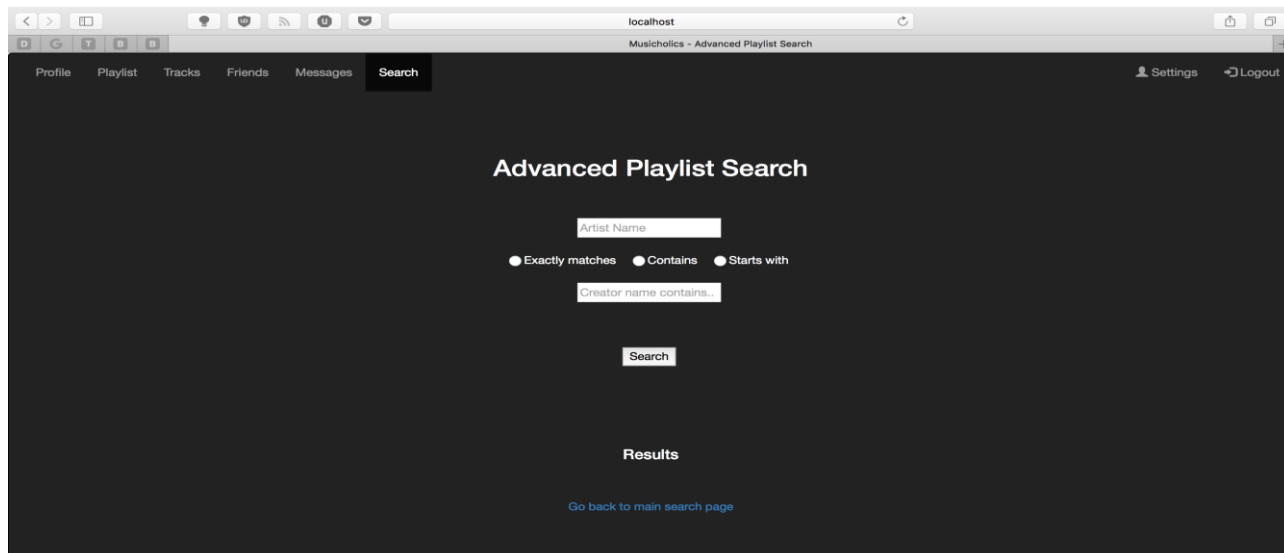
The screenshot shows a web browser window with the address bar set to localhost. The page title is "Musicholics - Advanced Track Search". The navigation bar includes links for Profile, Playlist, Tracks, Friends, Messages, and Search (which is highlighted). In the top right corner, there are links for Settings and Logout. The main content area is titled "Advanced Artist Search". It features a text input field for "Artist Name", three radio buttons for "Exactly matches", "Contains", and "Starts with", and a text input field for "Description contains..". A "Search" button is located below these options. Under the heading "Results", there is a link that says "Go back to main search page".

Tastamam (00:00:00 )



You can search for artists using artist name, and description. You can go back to main search page using the link at the bottom of the page.

## Advanced playlist search for users

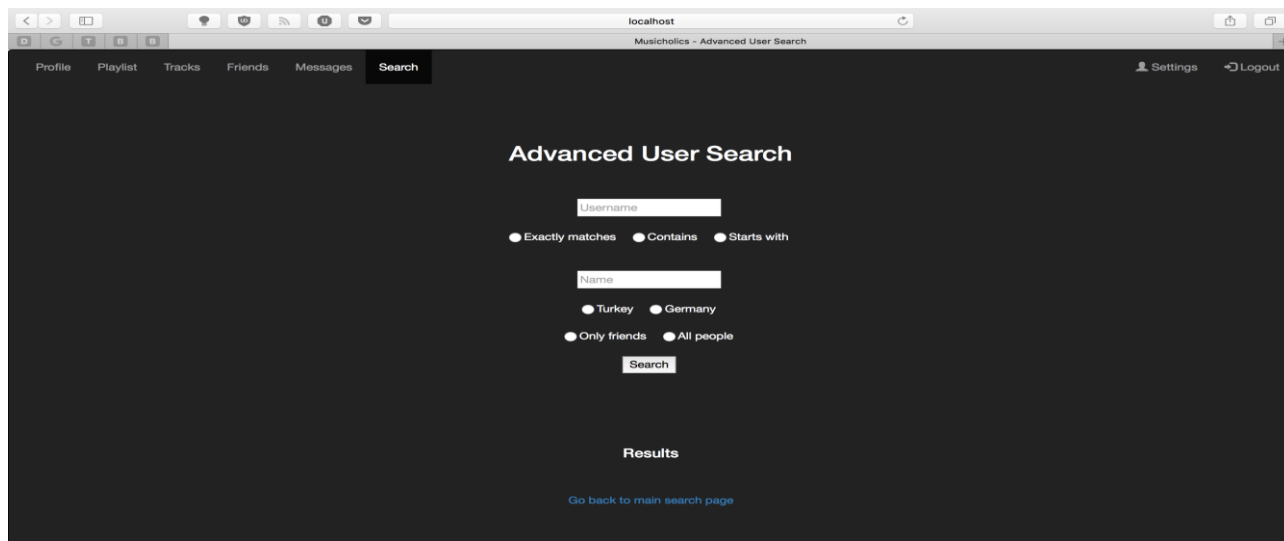


The screenshot shows a web browser window with the address bar set to 'localhost'. The page title is 'Musicholics - Advanced Playlist Search'. The navigation bar includes 'Profile', 'Playlist', 'Tracks', 'Friends', 'Messages', and 'Search' (which is highlighted). In the top right corner, there are links for 'Settings' and 'Logout'. The main content area is titled 'Advanced Playlist Search' and contains a search form. The form has a text input field labeled 'Artist Name'. Below it are three radio buttons: 'Exactly matches', 'Contains', and 'Starts with'. There is another text input field labeled 'Creator name contains:'. A 'Search' button is positioned below these fields. Underneath the search button, the word 'Results' is displayed. At the bottom of the form, there is a link that says 'Go back to main search page'.

Tastamam (00:00:00 )

You can search for playlists using playlist name, and creator name. You can go back to main search page using the link at the bottom of the page.

## Advanced user search for users

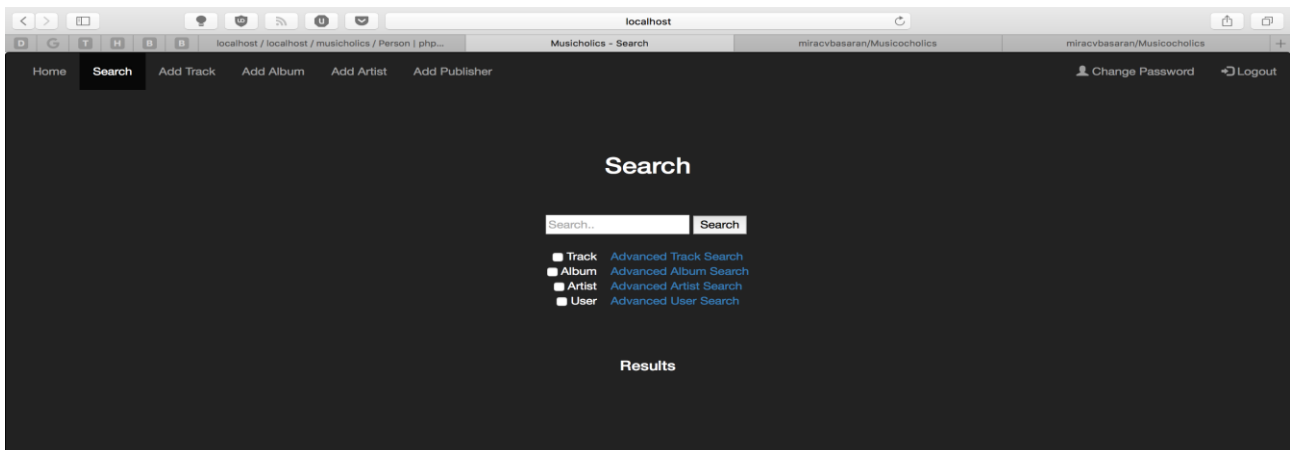


The screenshot shows a web browser window with the address bar set to 'localhost'. The page title is 'Musicholics - Advanced User Search'. The navigation bar includes 'Profile', 'Playlist', 'Tracks', 'Friends', 'Messages', and 'Search' (which is highlighted). In the top right corner, there are links for 'Settings' and 'Logout'. The main content area is titled 'Advanced User Search' and contains a search form. The form has a text input field labeled 'Username'. Below it are three radio buttons: 'Exactly matches', 'Contains', and 'Starts with'. There is another text input field labeled 'Name'. Below that are two radio buttons: 'Turkey' and 'Germany'. There are also two radio buttons: 'Only friends' and 'All people'. A 'Search' button is positioned below these fields. Underneath the search button, the word 'Results' is displayed. At the bottom of the form, there is a link that says 'Go back to main search page'.

Tastamam (00:00:00 )

You can search for users using username, name, country. You can also search amongst your friends or search for any person. You will not see people who you have blocked. You can go back to main search page using the link at the bottom of the page.

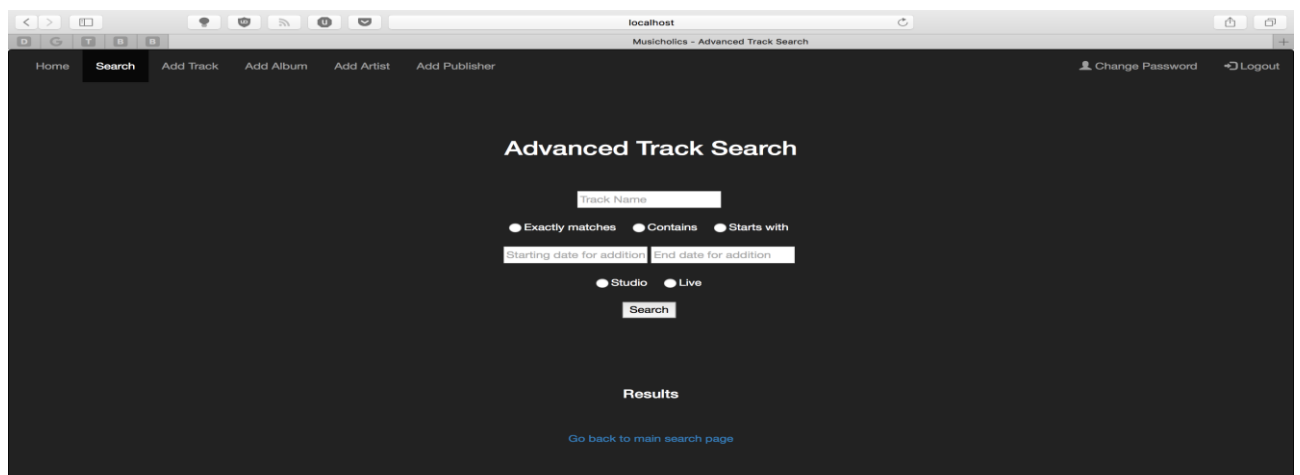
## Search screen for admins



The screenshot shows a web browser window with the URL `localhost / localhost / musiccholics / Person | php...`. The page title is "Musiccholics - Search". The navigation bar includes "Home", "Search" (active), "Add Track", "Add Album", "Add Artist", and "Add Publisher". On the right, there are links for "Change Password" and "Logout". The main content area has a "Search" heading, a search input field with the placeholder "Search...", and a "Search" button. Below the input field, there are four search options: "Track" (with a radio button), "Album" (with a radio button), "Artist" (with a radio button), and "User" (with a radio button). Each option has a corresponding "Advanced" search link: "Advanced Track Search", "Advanced Album Search", "Advanced Artist Search", and "Advanced User Search". At the bottom, there is a "Results" heading.

You can search for tracks, albums, artist, and users. It is possible to search for only one option or any number of options. If you wish, you can go to advanced search pages and search for detailed results.

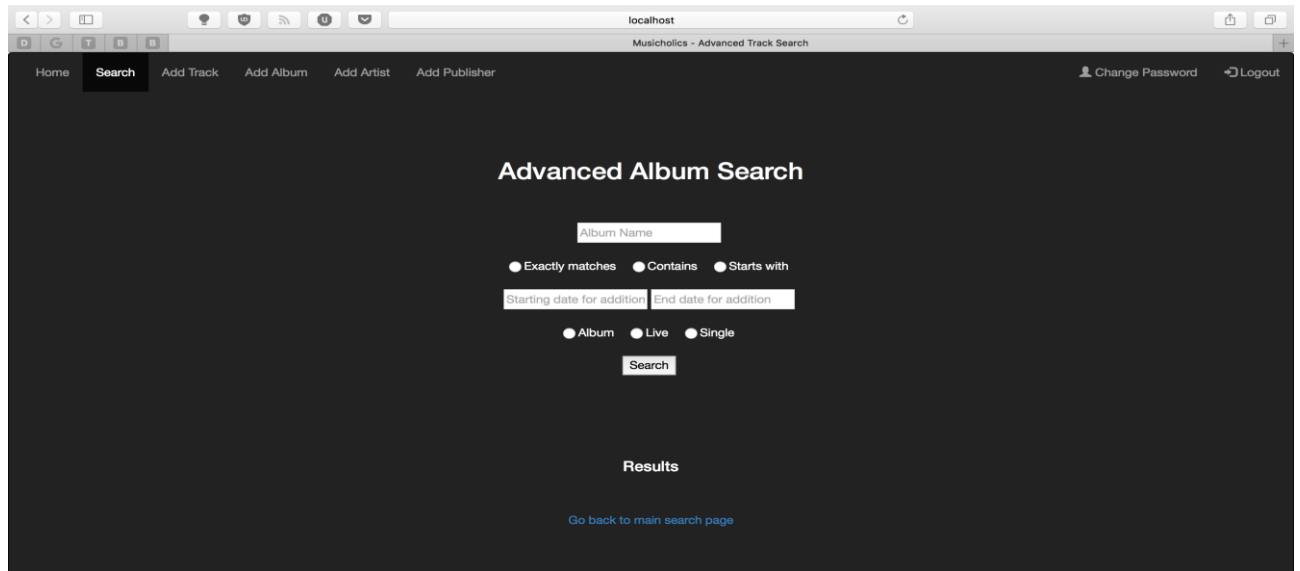
## Advanced track search for admins



The screenshot shows a web browser window with the URL `localhost / localhost / musiccholics / Advanced Track Search`. The page title is "Musiccholics - Advanced Track Search". The navigation bar is the same as the previous page. The main content area has an "Advanced Track Search" heading. Below the heading, there is a "Track Name" input field. Underneath, there are three radio buttons: "Exactly matches", "Contains", and "Starts with". Below these, there are two input fields: "Starting date for addition" and "End date for addition". Underneath these, there are two radio buttons: "Studio" and "Live". At the bottom, there is a "Search" button. Below the button, there is a "Results" heading. At the very bottom, there is a link: "Go back to main search page".

You can search for tracks using track name, date of addition, and track type. You can go back to main search page using the link at the bottom of the page.

### Advanced album search for admins

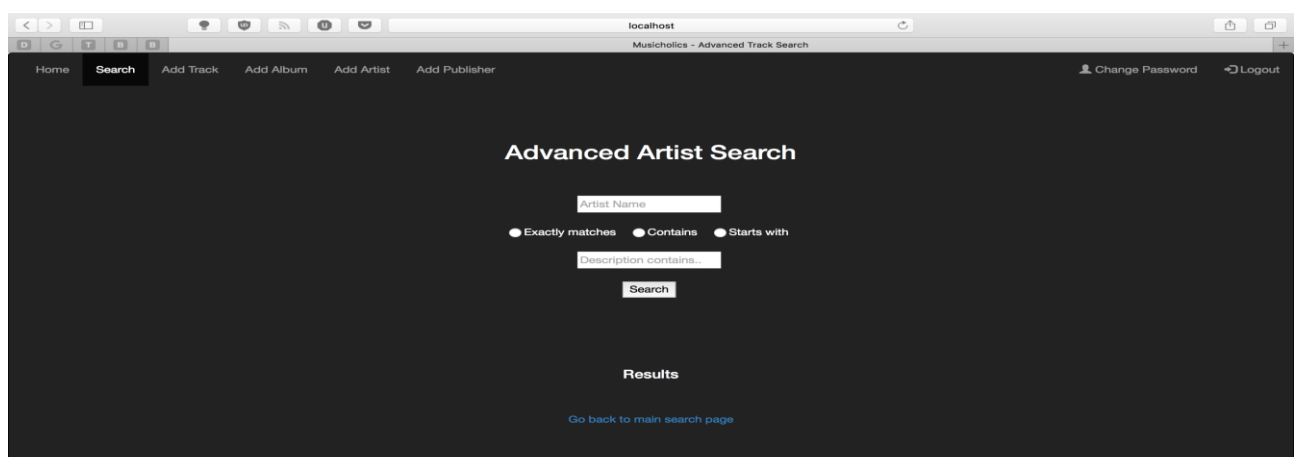


The screenshot shows a web browser window with the URL 'localhost' and the page title 'Musicholics - Advanced Track Search'. The navigation bar includes 'Home', 'Search' (highlighted), 'Add Track', 'Add Album', 'Add Artist', and 'Add Publisher'. On the right, there are links for 'Change Password' and 'Logout'. The main content area is titled 'Advanced Album Search' and contains the following form elements:

- An input field for 'Album Name'.
- Three radio buttons for search criteria: 'Exactly matches', 'Contains', and 'Starts with'.
- Two input fields for 'Starting date for addition' and 'End date for addition'.
- Three radio buttons for album type: 'Album', 'Live', and 'Single'.
- A 'Search' button.
- A 'Results' section.
- A link at the bottom: 'Go back to main search page'.

You can search for albums using album name, date of addition, and album type. You can go back to main search page using the link at the bottom of the page.

### Advanced artist for admins



The screenshot shows a web browser window with the URL 'localhost' and the page title 'Musicholics - Advanced Track Search'. The navigation bar is identical to the previous screenshot. The main content area is titled 'Advanced Artist Search' and contains the following form elements:

- An input field for 'Artist Name'.
- Three radio buttons for search criteria: 'Exactly matches', 'Contains', and 'Starts with'.
- An input field for 'Description contains...'.
- A 'Search' button.
- A 'Results' section.
- A link at the bottom: 'Go back to main search page'.

You can search for artists using artist name, and description. You can go back to main search page using the link at the bottom of the page.

## Advanced user search for admins

The screenshot shows a web browser window with the URL 'localhost'. The page title is 'Musicholics - Advanced User Search'. The navigation bar includes 'Home', 'Search' (highlighted), 'Add Track', 'Add Album', 'Add Artist', and 'Add Publisher'. On the right, there are links for 'Change Password' and 'Logout'. The main content area is titled 'Advanced User Search' and contains a search form with the following elements:

- A text input field labeled 'Username'.
- Three radio buttons: 'Exactly matches', 'Contains', and 'Starts with'.
- A text input field labeled 'Name contains...'.
- Two radio buttons: 'Turkey' and 'Germany'.
- A 'Search' button.

Below the search form, the word 'Results' is displayed. At the bottom, there is a link that says 'Go back to main search page'.

You can search for users using username, name, country. You can go back to main search page using the link at the bottom of the page.

## Friends

The screenshot shows a web browser window with the URL 'localhost'. The page title is 'Musicholics - Friends'. The navigation bar includes 'Profile', 'Playlist', 'Tracks', 'Friends' (highlighted), 'Messages', and 'Search'. On the right, there are links for 'Settings' and 'Logout'. The main content area is titled 'Friends' and contains a table with the following data:

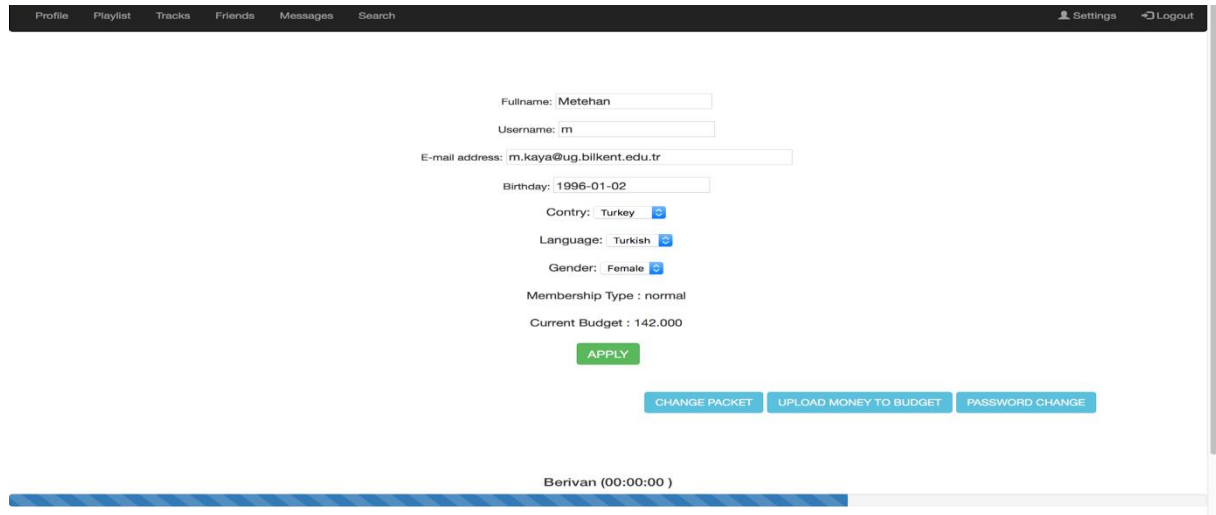
Username	Fullname
mvbasaran	Mirac Vuslat Basaran

At the bottom of the page, there is a progress bar with the text 'Tastamam (00:00:00)'.

You can view your friends on Friends section.

## Profile Settings

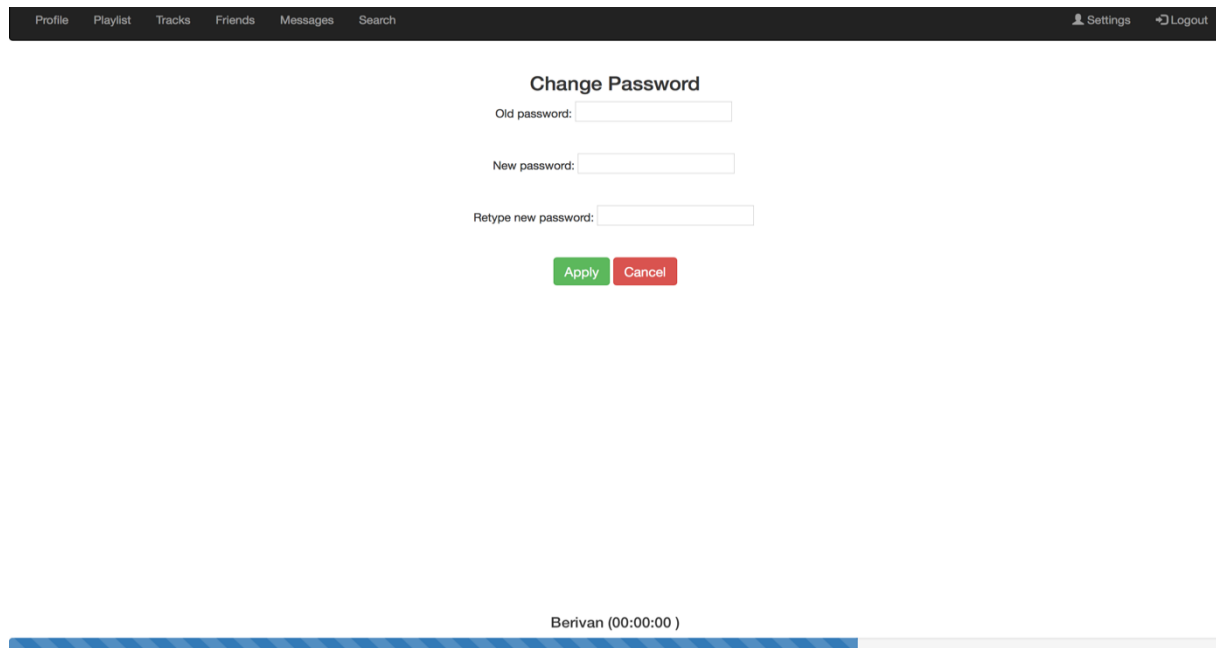
You can change details of her profile. Also you can add budget with credit card and upgrade to premium or artist by clicking on “change packet”. You can change her password by clicking on “change password”.



The screenshot shows the 'Profile Settings' page of a web application. At the top, there is a dark navigation bar with links: Profile, Playlist, Tracks, Friends, Messages, Search, Settings, and Logout. The main content area displays the user's profile information in a form-like layout. The fields are: Fullname: Metehan, Username: m, E-mail address: m.kaya@ug.bilkent.edu.tr, Birthday: 1996-01-02, Contry: Turkey (with a dropdown arrow), Language: Turkish (with a dropdown arrow), Gender: Female (with a dropdown arrow), Membership Type: normal, and Current Budget: 142.000. Below these fields is a green 'APPLY' button. At the bottom of the form, there are three blue buttons: 'CHANGE PACKET', 'UPLOAD MONEY TO BUDGET', and 'PASSWORD CHANGE'. At the very bottom of the page, there is a blue progress bar and the text 'Berivan (00:00:00)'.

## Change Password

You can change her password on this screen.



The screenshot shows the 'Change Password' page of a web application. At the top, there is a dark navigation bar with links: Profile, Playlist, Tracks, Friends, Messages, Search, Settings, and Logout. The main content area displays the 'Change Password' form. The fields are: Old password:, New password:, and Retype new password:. Below these fields are two buttons: a green 'Apply' button and a red 'Cancel' button. At the very bottom of the page, there is a blue progress bar and the text 'Berivan (00:00:00)'.

## Own Profile

You can view her own profile by clicking “Profile” section. Profile details and posts are seen and currently listened song is seen at the bottom of the page.

[Profile](#) [Playlist](#) [Tracks](#) [Friends](#) [Messages](#) [Search](#) [Settings](#) [Logout](#)

## Hello, Metehan

Username: mkaya

Fullname: Metehan

E-mail address: m.kaya@ug.bilkent.edu.tr

Country: Turkey

Gender: Female

Language: Turkish

Birthday: 1996-01-02

Budget: 142.000 \$

Date of registration: 2018-05-01

Membership type: normal

### POSTS

mvbasaran ( 2018-05-14 01:06:56 ):  
I like your photograph.

Berivan (00:00:00 )

ecakir ( 2017-10-08 02:03:04 ):

Budget: 142.000 \$

Date of registration: 2018-05-01

Membership type: normal

### POSTS

mvbasaran ( 2018-05-14 01:06:56 ):  
I like your photograph.

ecakir ( 2017-10-08 02:03:04 ):  
Are you an artist?

Berivan (00:00:00 )

## Complete Profile

Users' detailed profile information are seen by admin. Admin can view her playlist by clicking on "View Playlist" or ban the user by clicking on "Ban User".

This is,  
Ezgi Cakir  
Username: ecakir  
Fullname: Ezgi Cakir  
E-mail address: e.cakir@ug.bilkent.edu.tr  
Country: Turkey  
Gender: female  
Language: Turkish  
Birthday: 1996-01-04  
Budget: 30.000 \$  
Date of registration: 2018-05-03  
Membership type: artist

View Playlists Ban User

## POSTS

mvbasaran ( 2012-03-11 00:00:01 ):  
Please add you songs.

## Friend's Profile

You can view some information of her friend's profile and view all posts. You can view friend's playlists by clicking on "View Playlist" or block the user by clicking on "Block", unfriend him by clicking "Unfriend", and send message by clicking on "Send Message".

This is ecakir  
E-mail address: e.cakir@ug.bilkent.edu.tr  
Country: Turkey  
Gender: female  
Language: Turkish  
Birthday: 1996-01-04

View Playlists Send Message  
Block Unfriend

## POSTS

mvbasaran ( 2012-03-11 00:00:01 ):  
Please add you songs.

Berivan (00:00:00 )

## Nonfriend's Profile

You can view limited profile information of a nonfriend user. You can add the user as a friend by clicking on the button.

[Profile](#) [Playlist](#) [Tracks](#) [Friends](#) [Messages](#) [Search](#) [Settings](#) [Logout](#)

**This is, okarakas**

Fullname: Omer Karakas

Country: Germany

Gender: male

[ADD AS FRIEND](#)

Berivan (00:00:00 )

## Blocked Profile

If you block someone, you cannot view details of the profile. You can unblock the user by clicking on "Unblock".

[Profile](#) [Playlist](#) [Tracks](#) [Friends](#) [Messages](#) [Search](#) [Settings](#) [Logout](#)

**This is ecakir**

[UNBLOCK](#)

Berivan (00:00:00 )



## Purchase Track With Budget

A detailed information of user's budget and tracks is seen. You can purchase the track by clicking on "Purchase".

[Profile](#) [Playlist](#) [Tracks](#) [Friends](#) [Messages](#) [Search](#) [Settings](#) [Logout](#)

Purchase **Veda Busesi**

Price of Veda Busesi: \$ 1.150

Your budget: \$142.000

Purchase Cancel

Berivan (00:00:00 )

## Purchase Artist or Premium Packets with Budget

You can upgrade to premium and/or artist by clicking on respective buttons.

[Profile](#) [Playlist](#) [Tracks](#) [Friends](#) [Messages](#) [Search](#) [Settings](#) [Logout](#)

Your budget: \$ 142.000

Your current membership type : normal

### UPGRADE TO PREMIUM

Premium membership fee: 9.99

Upgrade To Premium

### UPGRADE TO ARTIST

Artist membership fee: 4.99

Upgrade To Artist

Berivan (00:00:00 )

## Adding Money to Budget With Credit Card

You can upload money to her budget with credit card only. You need to submit necessary information of credit card.

[Profile](#) [Playlist](#) [Tracks](#) [Friends](#) [Messages](#) [Search](#) [Settings](#) [Logout](#)

Amount: \$

Cardholder's Name:

Credit Card Number:

Expires on:

Credit Card Security Code:

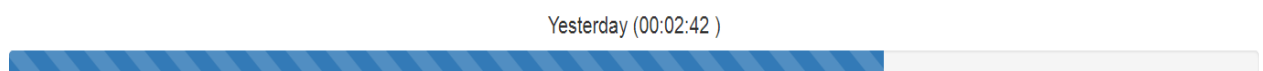


## View Track

[Profile](#) [Playlist](#) [Tracks](#) [Friends](#) [Messages](#) [Search](#) [Settings](#) [Logout](#)

### Berivan in Haziranda Olmek Zor

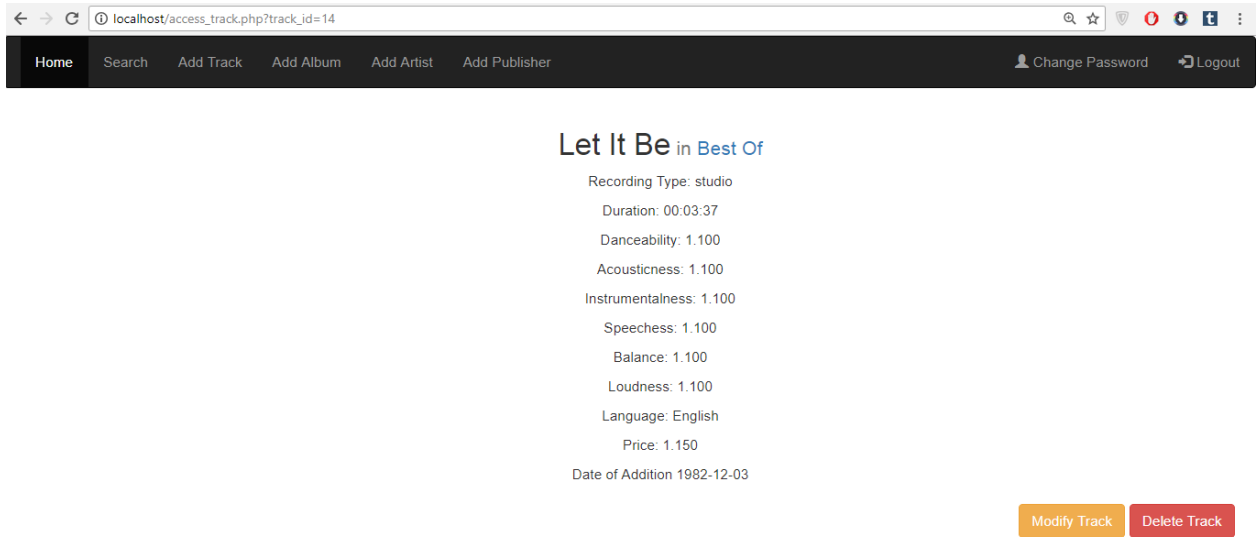
Recording Type: studio  
Duration: 00:03:12  
Danceability: 1.100  
Acousticness: 1.100  
Instrumentalness: 1.100  
Speechiness: 1.100  
Balance: 1.100  
Loudness: 1.100  
Language: Turkish  
Price: 1.150  
Date of Addition 2007-12-03



This is the view page a user is displayed when he clicks a track. He can view information about the track and he can perform three operations: He can push the purchase track button to go to purchase track

page where he can buy the track into his library; he can push the gift track button to go to gift track page where he can gift the track to another user; or he can play the track. Note that if the user does not have a premium or artist type membership, he can only listen to the songs he does not own at most 5 times a day. If he tries to listen to any more, he will get an error.

## Access Track



localhost/access\_track.php?track\_id=14

Home Search Add Track Add Album Add Artist Add Publisher Change Password Logout

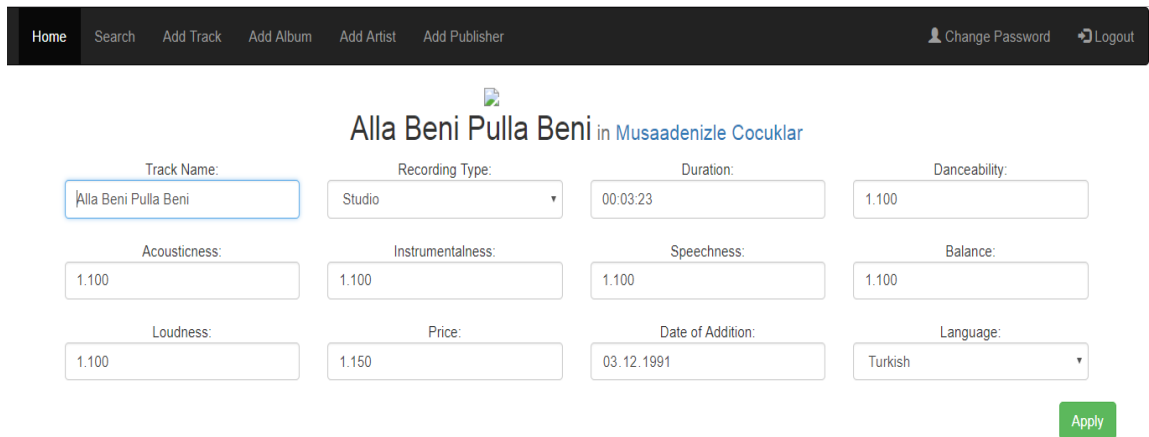
Let It Be in Best Of

Recording Type: studio  
Duration: 00:03:37  
Danceability: 1.100  
Acousticness: 1.100  
Instrumentalness: 1.100  
Speechiness: 1.100  
Balance: 1.100  
Loudness: 1.100  
Language: English  
Price: 1.150  
Date of Addition: 1982-12-03

Modify Track Delete Track

This is the access track page an admin encounters when he clicks on a track. Here, he can see information about the track and perform two operations: He can delete the track with “Delete Track” button or he can go to modify track page with the “Modify Track” button.

## Modify Track



localhost/access\_track.php?track\_id=14

Home Search Add Track Add Album Add Artist Add Publisher Change Password Logout

Alla Beni Pulla Beni in Musaadenizle Cocuklar

Track Name: Recording Type: Duration: Danceability:  
Alla Beni Pulla Beni Studio 00:03:23 1.100

Acousticness: Instrumentalness: Speechiness: Balance:  
1.100 1.100 1.100 1.100


Loudness: Price: Date of Addition: Language:  
1.100 1.150 03.12.1991 Turkish

Apply

This is the modify track page where an admin can modify the values of a track such as name, recording type, language etc. in the database. Changes take place when he clicks “Apply.”

## View Album

[Profile](#) [Playlist](#) [Tracks](#) [Friends](#) [Messages](#) [Search](#) [Settings](#) [Logout](#)



### Musaadenizle Cocuklar by [Baris Manco](#)

From Publisher [pubAnka](#)


Song Name	Length	Price
Alla Beni Pulla Beni	00:03:23	1.150
Hal Hal	00:03:25	1.150

Berivan (00:03:12 )

This is the view page a user is displayed when he clicks on an album. He can see the name, artist and publisher of the album and he can click on the links to go to them. He can also see the tracks that are in this album and he can click on the tracks to go to those tracks view page.

## Access Album

[Home](#) [Search](#) [Add Track](#) [Add Album](#) [Add Artist](#) [Add Publisher](#) [Change Password](#) [Logout](#)



### Musaadenizle Cocuklar by [Baris Manco](#)

From Publisher [pubAnka](#)


[Modify Album](#) [Delete Album](#)

Song Name	Length	Price
Alla Beni Pulla Beni	00:03:23	1.150
Hal Hal	00:03:25	1.150

This is the access page an admin is displayed when he clicks an album. He can see information about the album and access the tracks, the artist and the publisher. Here, he can delete the album by clicking “Delete Album” button or go to Modify Album page by clicking “Modify Album” button.

## Modify Album

[Home](#) [Search](#) [Add Track](#) [Add Album](#) [Add Artist](#) [Add Publisher](#) [Change Password](#) [Logout](#)



Dosya Seç

Dosya seçilmedi

Update

Musaadenizle Cocuklar by Baris Manco

Album Name:

Album Type:

Date:

Musaadenizle Cocuklar

Album

03.10.1995

Apply

Song Name	Length	Price	
Alla Beni Pulla Beni	00:03:23	1.150	<input type="checkbox"/>
Hal Hal	00:03:25	1.150	<input type="checkbox"/>

Delete Tracks

Add Track

Track Name:

Duration:


Price (in \$):

Add Track

This is the Modify Album page where an admin can change the values of an album such as name or date. Changes take place when he clicks “Apply.” He can also select one or more tracks in the album to be deleted. Selected tracks are deleted when he clicks “Delete Tracks” button. He can also add tracks to the album by specify information about the track and clicking “Add Track” button. This would redirect to modify track page where he can then modify information about the new track.

## View Artist

[Profile](#) [Playlist](#) [Tracks](#) [Friends](#) [Messages](#) [Search](#) [Settings](#) [Logout](#)



Group Yorum

Listen carefully


Album Name	Album Type	Publish Date
Haziranda Olmek Zor	album	1994-10-03

Barivan (00:03:12 )

This is the view page of an artist a user is displayed when he clicks on an artist. Here, he can see information about the artist and see and access the albums the artist has published.

## Access Artist

[Home](#) [Search](#) [Add Track](#) [Add Album](#) [Add Artist](#) [Add Publisher](#) [Change Password](#) [Logout](#)

**Baris Manco**  
Incredible Musician


[Modify Artist](#) [Delete Artist](#)

Album Name	Album Type	Publish Date
Musaadenizle Cocuklar	album	1995-10-03

This is the access page an admin encounters when he clicks on an artist. Here, he can see information about the artist and access his albums. He can click on “Delete Artist” button to delete the artist or he can click on “Modify Artist” button to go to Modify Artist page.

## Modify Artist

[Home](#) [Search](#) [Add Track](#) [Add Album](#) [Add Artist](#) [Add Publisher](#) [Change Password](#) [Logout](#)

 Dosya Seç Dosya seçilmedi

Update

[Apply](#)

Album Name	Album Type	Publish Date
<a href="#">Kainat Sustu</a>	album	2016-08-06

[Delete](#)

### Add Album

Album Name:

Album Publisher:

Album Type:  
Album

Publish Date:  
gg.aa.yyyy

Add Album


This is the Modify Page where and admin can change the date of an artist in the database. He can change the artist’s name, description etc. Changes take place when he clicks “Apply.”

He can select one or more albums and then click “Delete” button to delete the selected albums from the artist.

He can enter information about a new album and can click on “Add Album” button to add a new album to the artist. After clicking the button, system will redirect to “Modify Album” Page where the admin can then change information about this new album.

## View Publisher

[Profile](#) [Playlist](#) [Tracks](#) [Friends](#) [Messages](#) [Search](#) [Settings](#) [Logout](#)



### pubAnka

Country: Turkey  
City: Ankara


Album Name	Album Type	Publish Date
Haziranda Olmek Zor	album	1994-10-03
Musaadenizle Cocuklar	album	1995-10-03
Kainat Sustu	album	2016-08-06

Berivan (00:03:12 )

This is the view page of a publisher a user encounters when clicks on a publisher. Here, he can see information about the publisher and access the albums the publisher published. Clicking on an album will redirect to that album's view page.

## Access Publisher

[Home](#) [Search](#) [Add Track](#) [Add Album](#) [Add Artist](#) [Add Publisher](#) [Change Password](#) [Logout](#)



### pubBer

Country: Germany  
City: Berlin


[Modify Publisher](#) [Delete Publisher](#)

Album Name	Album Type	Publish Date
So Real	single	1994-05-21
Evolve	album	2017-10-03

This is the access page an admin encounters when he clicks on a publisher. Here, he can view information about the publisher and access the albums published by the publisher. He can click on "Modify Publisher" button to go to modify page of the publisher and he can click on "Delete Publisher" button to delete the publisher.

## Modify Publisher

[Home](#) [Search](#) [Add Track](#) [Add Album](#) [Add Artist](#) [Add Publisher](#) [Change Password](#) [Logout](#)

 **Publisher pubBer**

Name:

Country:

City:

Album Name	Album Type	Publish Date	
So Real	single	1994-05-21	<input type="checkbox"/>
Evolve	album	2017-10-03	<input type="checkbox"/>

This is the modify page of a publisher where an admin can change the data of a publisher. He can change name, country and city of the publisher. The changes take place when he clicks on "Apply." He can select one or more albums to be deleted. After clicking "Delete," selected albums will be deleted from the database.

## Message List

[Profile](#) [Playlist](#) [Tracks](#) [Friends](#) [Messages](#) [Search](#) [Settings](#) [Logout](#)

**Messages**

Name	Message
Mirac Vuslat Basaran	It is not you business.

Veda Busesi (00:02:34 )

The user can view his messages and he is enabled to send message via a button.



## Send Direct Message

ProfilePlaylistTracksFriendsMessagesSearch

SettingsLogout

Send Message

mvbasaran

Message:

Did you finish the project?

Send Message

Veda Busesi (00:02:34 )

The user can send a message by entering text.

## Send Optional Message

ProfilePlaylistTracksFriendsMessagesSearch

SettingsLogout

Send Message

Receiver Username:

Message:

hi, how are you?

Send Message

Veda Busesi (00:02:34 )

The user can send a message by entering username of another user, and text.

## View Own Playlists

ProfilePlaylistTracksFriendsMessagesSearch

SettingsLogout

Playlists

Playlist Name:

Description:

Create Playlist

Name	Description
golbiciler	mixed list
fav_playlist	Favourite tracks

Veda Busesi (00:02:34 )

The user can view his playlists.

## View Own Playlist

[Profile](#) [Playlist](#) [Tracks](#) [Friends](#) [Messages](#) [Search](#) [Settings](#) [Logout](#)

Playlist: **golibiciler**

by m

mixed list

Rate: 2.0000

Listen Playlist Add Tracks Delete Playlist

Song name	Length	Price
Veda Busesi	00:02:34	1.150
Hal Hal	00:03:25	1.150
Just A Boy	00:03:14	1.150
Demons	00:03:30	1.150

Username	Comment
ecakir	I DONT LIKED
eeyaz	Thanks for this list.

Veda Busesi (00:02:34 )

The user can listen his playlist if the limit conditions are satisfied. He can delete the playlist, and also add tracks to it. Also, he can view the comments to this playlist.

## View Another User's Playlists

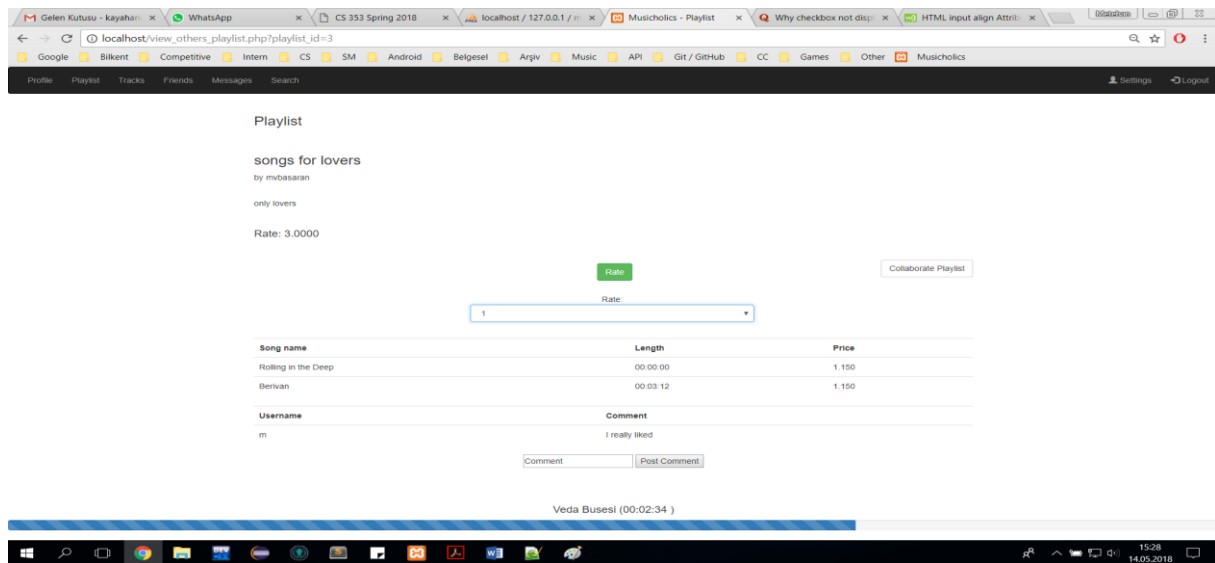
[Profile](#) [Playlist](#) [Tracks](#) [Friends](#) [Messages](#) [Search](#) [Settings](#) [Logout](#)

Playlists

Name	Description
songs for lovers	only lovers

The user can view another user's playlists.

## View Another User's Playlist



Playlist

songs for lovers  
by mubasaran  
only lovers  
Rate: 3.0000

[Rate](#) [Collaborate Playlist](#)

Rate: 1

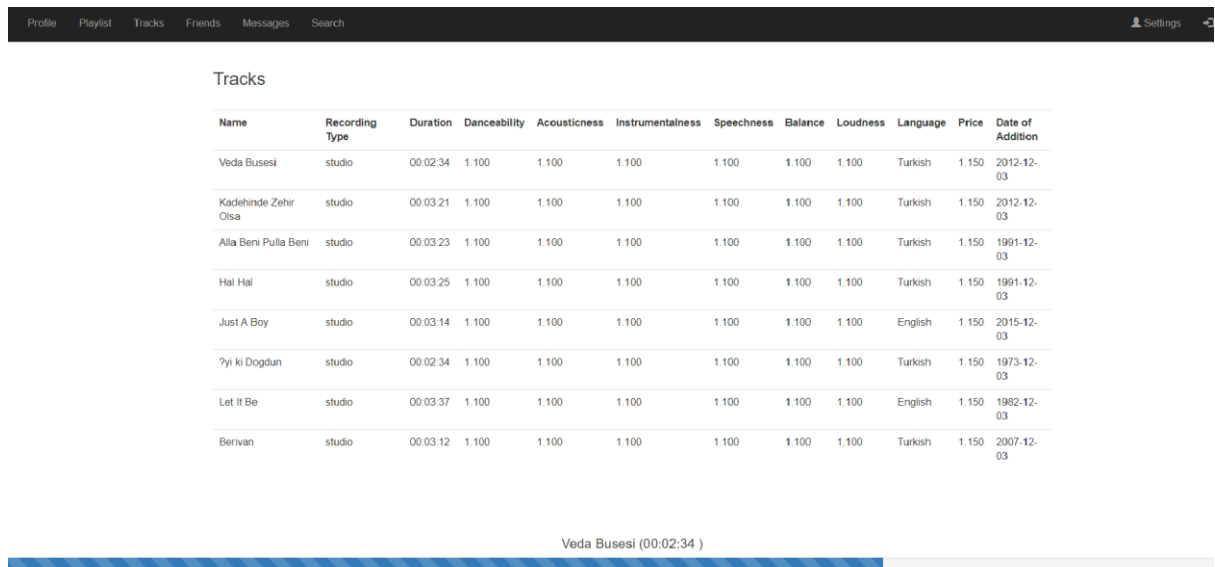
Song name	Length	Price
Rolling in the Deep	00:00:00	1.150
Berivan	00:03:12	1.150

Username	Comment
m	I really liked

Veda Busesi (00:02:34 )

The user is able to view another user's playlist. The user can rate another user's playlist and see its average rate. Also, the user can comment on the playlist.

## View Tracks



Tracks

Name	Recording Type	Duration	Danceability	Acousticness	Instrumentalness	Speechiness	Balance	Loudness	Language	Price	Date of Addition
Veda Busesi	studio	00:02:34	1.100	1.100	1.100	1.100	1.100	1.100	Turkish	1.150	2012-12-03
Kadehinde Zehir Olisa	studio	00:03:21	1.100	1.100	1.100	1.100	1.100	1.100	Turkish	1.150	2012-12-03
Alia Beni Pulla Beni	studio	00:03:23	1.100	1.100	1.100	1.100	1.100	1.100	Turkish	1.150	1991-12-03
Hal Hal	studio	00:03:25	1.100	1.100	1.100	1.100	1.100	1.100	Turkish	1.150	1991-12-03
Just A Boy	studio	00:03:14	1.100	1.100	1.100	1.100	1.100	1.100	English	1.150	2015-12-03
Y'i ki Dogdun	studio	00:02:34	1.100	1.100	1.100	1.100	1.100	1.100	Turkish	1.150	1973-12-03
Let It Be	studio	00:03:37	1.100	1.100	1.100	1.100	1.100	1.100	English	1.150	1982-12-03
Berivan	studio	00:03:12	1.100	1.100	1.100	1.100	1.100	1.100	Turkish	1.150	2007-12-03

Veda Busesi (00:02:34 )

User is able to view the tracks that he bought and gifted. Links for tracks are enabled.

## Modify Playlist

[Profile](#) [Playlist](#) [Tracks](#) [Friends](#) [Messages](#) [Search](#) [Settings](#) [Log](#)

Tracks

Add Tracks

Name	Recording Type	Duration	Danceability	Acousticness	Instrumentalness	Speechness	Balance	Loudness	Language	Price	Date of Addition
Veda Busesi	studio	00:02:34	1.100	1.100	1.100	1.100	1.100	1.100	Turkish	1.150	2012-12-03
Kadehinde Zehir Olsa	studio	00:03:21	1.100	1.100	1.100	1.100	1.100	1.100	Turkish	1.150	2012-12-03
Alla Beni Pulla Beni	studio	00:03:23	1.100	1.100	1.100	1.100	1.100	1.100	Turkish	1.150	1991-12-03
Hai Hai	studio	00:03:25	1.100	1.100	1.100	1.100	1.100	1.100	Turkish	1.150	1991-12-03
Just A Boy	studio	00:03:14	1.100	1.100	1.100	1.100	1.100	1.100	English	1.150	2015-12-03
?yi ki Dogdun	studio	00:02:34	1.100	1.100	1.100	1.100	1.100	1.100	Turkish	1.150	1973-12-03
Let It Be	studio	00:03:37	1.100	1.100	1.100	1.100	1.100	1.100	English	1.150	1982-12-03
Berivan	studio	00:03:12	1.100	1.100	1.100	1.100	1.100	1.100	Turkish	1.150	2007-12-03

Veda Busesi (00:02:34 )

User is able to add tracks that he bought and gifted to a playlist. Links for tracks are enabled.