*MusicHub*

*Система за организиране на музика*

Автор: Димитър Колев

Ръководител: Венета Тончева

Съдържание

[**I.** **Кратко описание на системата** 3](#_Toc71127609)

[**II.** **Цели на платформата** 3](#_Toc71127610)

[**III.** **Приложение на системата** 3](#_Toc71127611)

[**IV.** **Използвани технологии** 3](#_Toc71127612)

[**V.** **Сорс код** 4](#_Toc71127613)

[**VI.** **Тестови акаунти в системата** 4](#_Toc71127614)

[**VII.** **Реализация** 4](#_Toc71127615)

[**VIII.** **Архитектура на проекта** 5](#_Toc71127616)

[**IX.** **Валидации, сигурност, качество на кода** 6](#_Toc71127617)

[**X.** **Добри практики** 7](#_Toc71127618)

[**XI.** **Функционалности на приложението и роли в системата** 7](#_Toc71127619)

[1. Зареждане на приложение 7](#_Toc71127620)

[2. Налични playlist 8](#_Toc71127621)

[3. Регистриране на потребители 8](#_Toc71127622)

[4. Вход в системата 9](#_Toc71127623)

[5. Създаване на категория 9](#_Toc71127624)

[6. Преглед на наличните музикални категории 10](#_Toc71127625)

[7. Създаване на лиценз 10](#_Toc71127626)

[8. Одобрение на лиценз 10](#_Toc71127627)

[9. Добавяне на нова песен 11](#_Toc71127628)

[10. Моите песни 11](#_Toc71127629)

[11. Редактиране на песен 12](#_Toc71127630)

[12. Изтриване на песен 12](#_Toc71127631)

[13. Моите плейлисти 13](#_Toc71127632)

[14. Създаване на нов playlist 13](#_Toc71127633)

[15. Слушане на песен 13](#_Toc71127634)

[16. Харесване/не харесване на песен 14](#_Toc71127635)

[17. Създаване на коментар на песен 14](#_Toc71127636)

[18. Редактиране на коментар 15](#_Toc71127637)

[19. Изтриване на коментар 15](#_Toc71127638)

[20. Реагиране на коментар 15](#_Toc71127639)

[21. Подобни песни на слушаната 15](#_Toc71127640)

[22. Добавяне на песен към playlist 16](#_Toc71127641)

[23. Слушане на playlist 16](#_Toc71127642)

[24. Премахване на песен от playlist 17](#_Toc71127643)

[25. Следване на потребител 17](#_Toc71127644)

[26. Профил на потребител 17](#_Toc71127645)

[27. Редактиране на профил 18](#_Toc71127646)

[28. Смяна на парола 18](#_Toc71127647)

[29. Профил на други потребители 18](#_Toc71127648)

[30. Последвани потребители 19](#_Toc71127649)

[31. Потребители, които следват даден потребител 19](#_Toc71127650)

[32. Бърз достъп до последваните потребители 20](#_Toc71127651)

[33. Слушани песни 20](#_Toc71127652)

[34. Харесани песни 20](#_Toc71127653)

[**XII.** **Заключение** 21](#_Toc71127654)

# **Кратко описание на системата**

Много хора се забавляват като слушат музика. Затова аз реших да им помогна като направя уеб сайт, в който потребителите да могат да качват своите песни, както да слушат песните на другите потребители и да ги оценяват. Системата е лесна за използване, с интуитивен интерфейс и лесно разбираеми функции. Резултатът за крайния клиентът е по-чести приятни моменти чрез слушането на хубава музика, която може да бъде организирана в разнообразни плейлисти и категории.

# **Цели на платформата**

Платформата цели да:

* Предлага на потребителя приятни моменти чрез слушане на музика;
* Отрази мнението на потребителите за даден музикален хит;
* На база на потребителската активност да предложи на слушателя песни, които биха предизвикали интереса му;
* Да мотивира младите талантливи изпълнители да представят своята музика на аудиторията.

# **Приложение на системата**

Може да се използва от всеки един човек както като гост, разглеждащ сайта, така и като регистриран потребител. Това предполага масово използване на приложението като конкурент на YouTube Music, Spotify и SoundCloud. Гостите могат да слушат музика, но нямат право да добавят нова такава, да коментират, да създават плейлисти и други функции, характерни само за регистрирани потребители.

# **Използвани технологии**

* Visual Studio 2019 - среда за разработка
* Microsoft SQL Server, Entity Framework Core 2.2 – за работа с бази данни
* C# - основен програмен език
* .Net Core 3.1, ASP.NET CORE 3.1 – основни технологии
* MyTested.AspNetCore.Mvc, xUnit – за тестване на кода
* Automapper
* SendGrid
* Angular & Bootstrap, HTML, CSS

# **Сорс код**

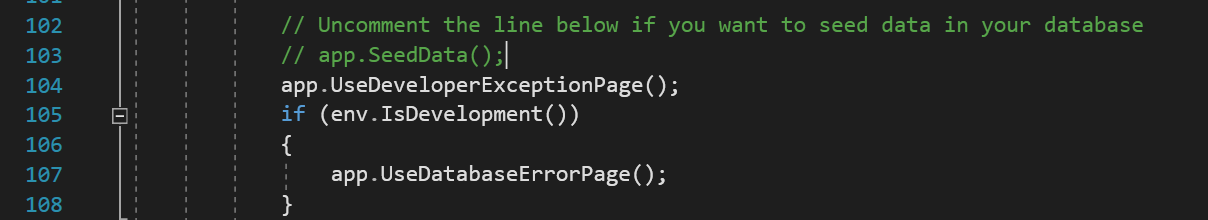
Линк към сорс кода: https://dev.azure.com/dimkolev2002/MusicHub

# **Тестови акаунти в системата**

За да тествате системата, трябва да имате инсталирани:

* .Net 3.1
* Наличен SQL Server

Преди да стартирате приложението, трябва да промените connection string-а, който служи за връзка с базата данни. Това действие може да направите във файла, намиращ се в Project\Web\MusicHub.Web с име appsettings.json. Ако в базата данни да има примерна информация, може да откоментирате ред 103 във файла Startup.cs в директорията Project\Web\MusicHub.Web.



Потребител

* Потребителско име: ***rootuser***
* Парола: ***rootpass***

Администратор

* Потребителско име: ***rootadmin***
* Парола: ***rootpass***

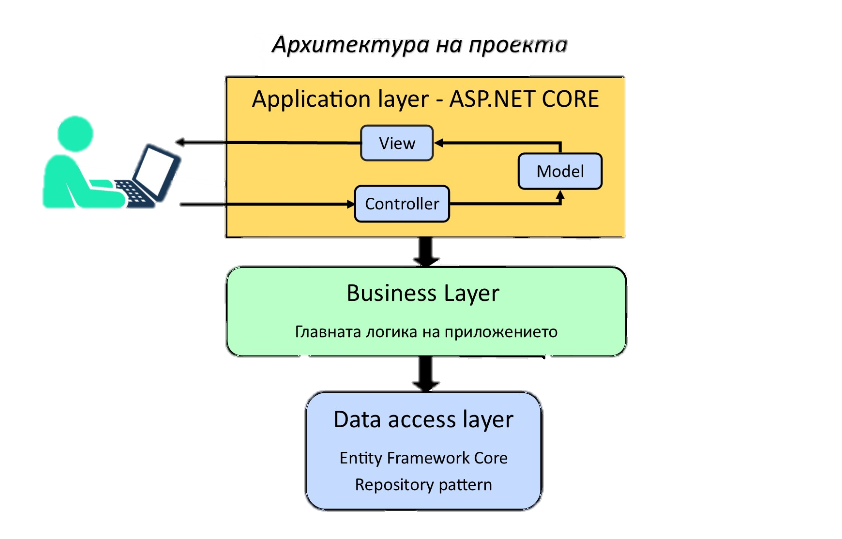
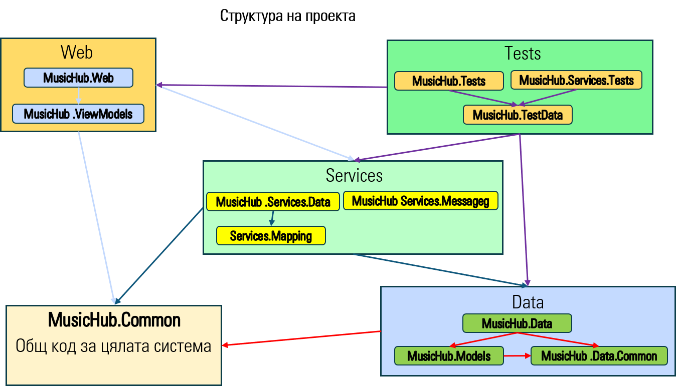
# **Реализация**

Стремя се да използвам най-модерните и качествени технологии.

* C#
* Модерен, бързо развиващ се, приятен за писане
* Типизиран
* Cross-platform
* ASP.NET Core
* Изключително бърз и удобен framework
* Разработен и се поддържан от Microsoft, което го прави надежден
* Cross-platform
* Лесен за поддръжка
* SQL
* Релационна база данни, която е по-подходяща за проекта
* Лесно се комбинира с ASP.NET Core
* Подходяща за по-мащабни проекти
* Angular
* Спестява много писане на код
* Прави приложението по-динамично
* Можем да променяме потребителския интерфейс без да презареждаме страницата
* Bootstrap
  + Спестява много време
  + Потребителския интерфейс става по-качествен и приятен
  + Responsive

# **Архитектура на проекта**

Приложението използва „MVC - Model, View and Controller“ модела и е разделено на няколко основни слоя.



* ***“Data access layer”*** – това е слоя, който работи с базата данни. Този слой е самостоятелен и може да се преизползва в някое друго приложение. Кода за този слой се намира в папката „Data“, като той се разделя на други два слоя.
* ***„Domain Layer“*** – който се грижи за моделите в базата данни. Кода за този слой е в папката „Data/MusicHub.Data.Models“. Той се състои от класове, които представляват дадена таблица в базата данни.
* ***„Persistence Layer“*** – това е слоя, който се грижи за съхраняването, извличането и обработката на данните в базата, в него се намират всички конфигурации свързани с базата данни, миграциите и логиката по началното добавяне на данни в базата - data seeding. Кода за този слой се намира в папката “Data/ MusicHub.Data“. Тук е имплементиран “Repository pattern“, което ни дава възможност, ако имаме бизнес изискване да използваме друг вид бази данни, например нерелационни – „NoSQL“, да го направим много бързо и лесно и с много малко промени по кода.
* ***“Business Layer”*** – съдържа цялата логика на приложението. Зависи само от „Data access layer“ и може да се преизползва в различни приложения. Ако искаме да направим мобилна версия на системата, кода от този слой ще се преизползва, тоест няма да се налага да се правят промени. Кода за този слой се намира в папката „Services/ MusicHub.Services“.
* ***“Application Layer”*** – състои се от елементи, които са специфични за приложението. Зависи от „Bisuness Layer“. Осъществява връзката между приложението и „Bisuness Layer“. Зависи от конкретни технологии. Зависи дали разработваме мобилно, интернет или друг вид приложение. В този случай се използват технологии и концепции като: ASP.NET Core, Midldlewares и др. Работата му е да приеме заявката (request), да я обработи и да върне отговор (response). Кода за този слой е в папка „Web/ MusicHub.Web“.
* ***“Presentation Layer”*** – тук се съдържа логиката по показване на съдържанието на потребителя. Приложението използва framework „Angular“ – презентира резултатите от действията на потребителя в браузера. Тук се използват и други технологии като: HTML, Bootstrap, CSS, TS. Логиката за този слой се намира в папките „ClientApp“ на папката “Web/ MusicHub.Web”
* ***“Common Layer”*** – съдържа логика, която се използва във всички останали слоеве. В него се намират елементи като: глобални константи, custom exceptions, extension methods. Кода за този слой се намира в папката “MusicHub.Common”
* ***„Tests“*** - системата е тествана с няколко автоматизирани тестове (unit tests и integration tests), които се намират в папката „Tests“.

# **Валидации, сигурност, качество на кода**

* ***Качество на кода*** - при писането на кода се спазват всички принципи за качествен програмен код. Спазени са всички „SOLID“ принципи. Използва се .editorconfig файл, в който се дефинират различни конвенции и стилове за писането на кода, за да може кода на проекта да е консистентен и да е форматиран добре.
* ***Сигурност и валидации*** – при всички функционалности, които изискват от потребителя да въведе някакви данни, данните се валидират и ако не са коректни се връща описателни съобщения с грешката за информация на потребителя. Приложението е защитено от атаки като: SQL Injection, Cross-site scripting (XSS), Cross Site Request Forgery(CSRF), Parameter tampering. Сигурността на приложението се базира на използваните технологии и е направена добре, за да се гарантира стабилността му пред крайния потребител.

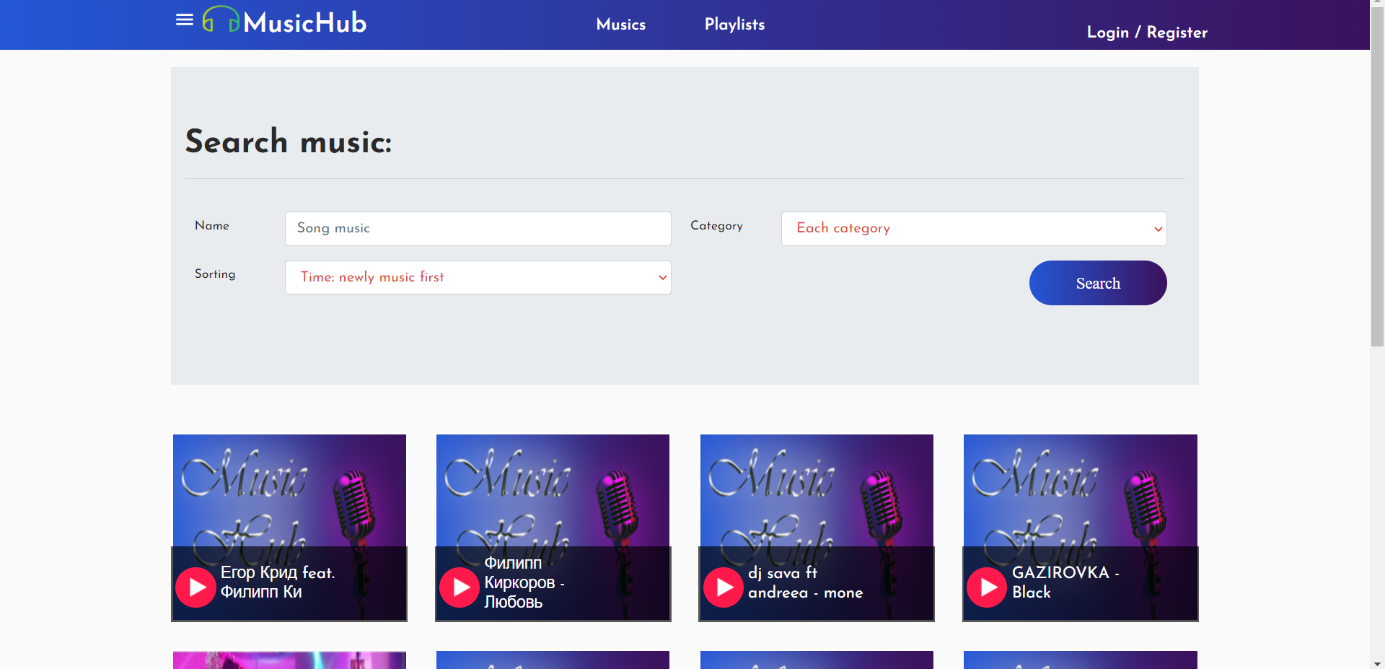
# **Добри практики**

* Използвам Azure, където може да се проследи разработката на проекта. Линк към Azure: <https://dev.azure.com/dimkolev2002/MusicHub>. Спазвам най-добрите практики като използвам branches и pull requests. В master branch винаги има стабилна версия на проекта. Другият важен branch е develop, който се използва по време на разработката. Когато е нужно да се разработи нова функционалност се създава нов branch и след като се имплементира се прави pull request към develop. За да се качат новите промени на develop e нужно да се прегледат от поне един човек. Само ако си администратор на проекта, можеш да добавяш директно промените. Master и develop са защитени и към тях не може да се прави директно push. Както и да управлявам задачите си по проекта.
* Code reviews – винаги преглеждам кода преди да направя commit или merge.
* Асинхронно програмиране
* Continuous integration, continuous delivery
* Различни среди – development environment и production environment
* Спазвам добрите практики при разработка на софтуер

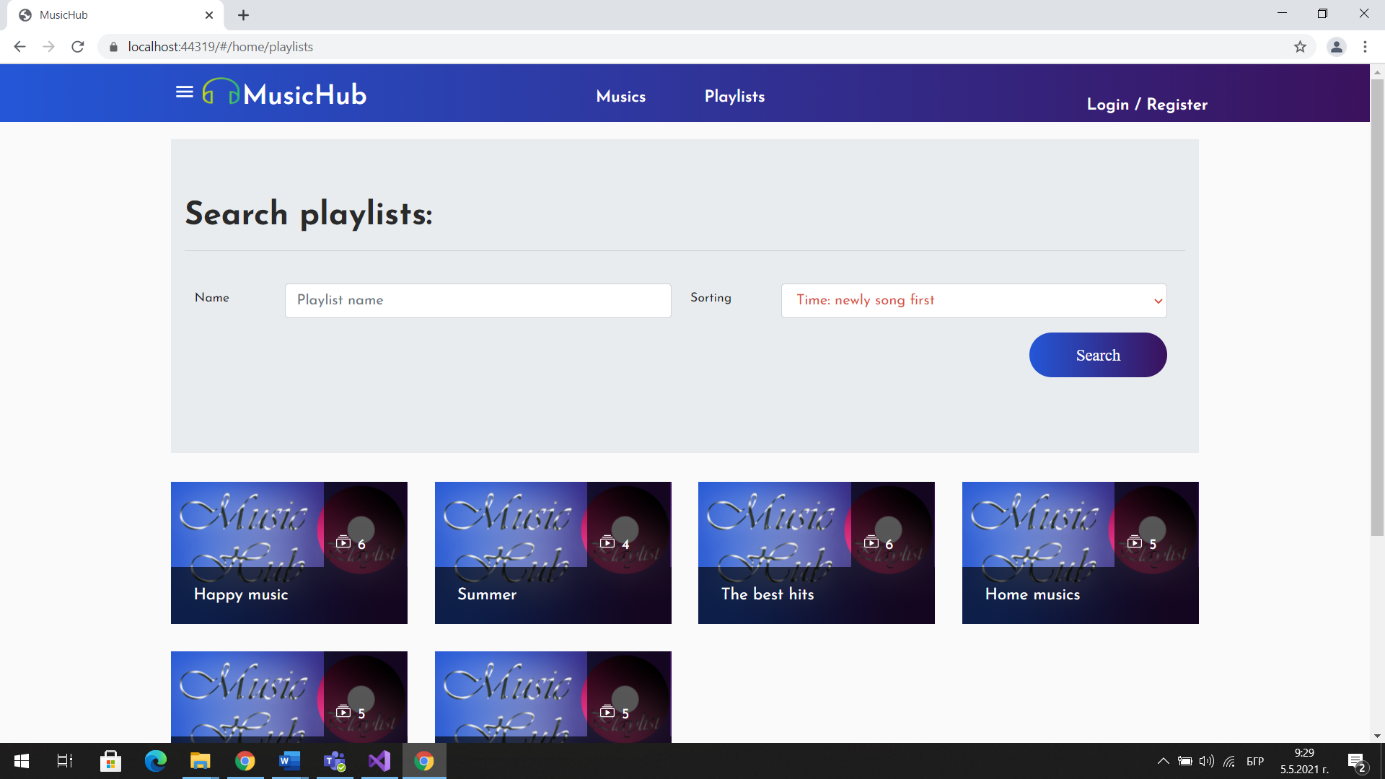
# **Функционалности на приложението и роли в системата**

Потребителите на приложението могат да бъдат в следните роли: „Посетител“, „Потребител“ и „Администратор“, като ако сте във всяка следваща роля в списъка, ще имате повече права в системата. Всеки ново регистрирал се автоматично придобива роля „Потребител“.

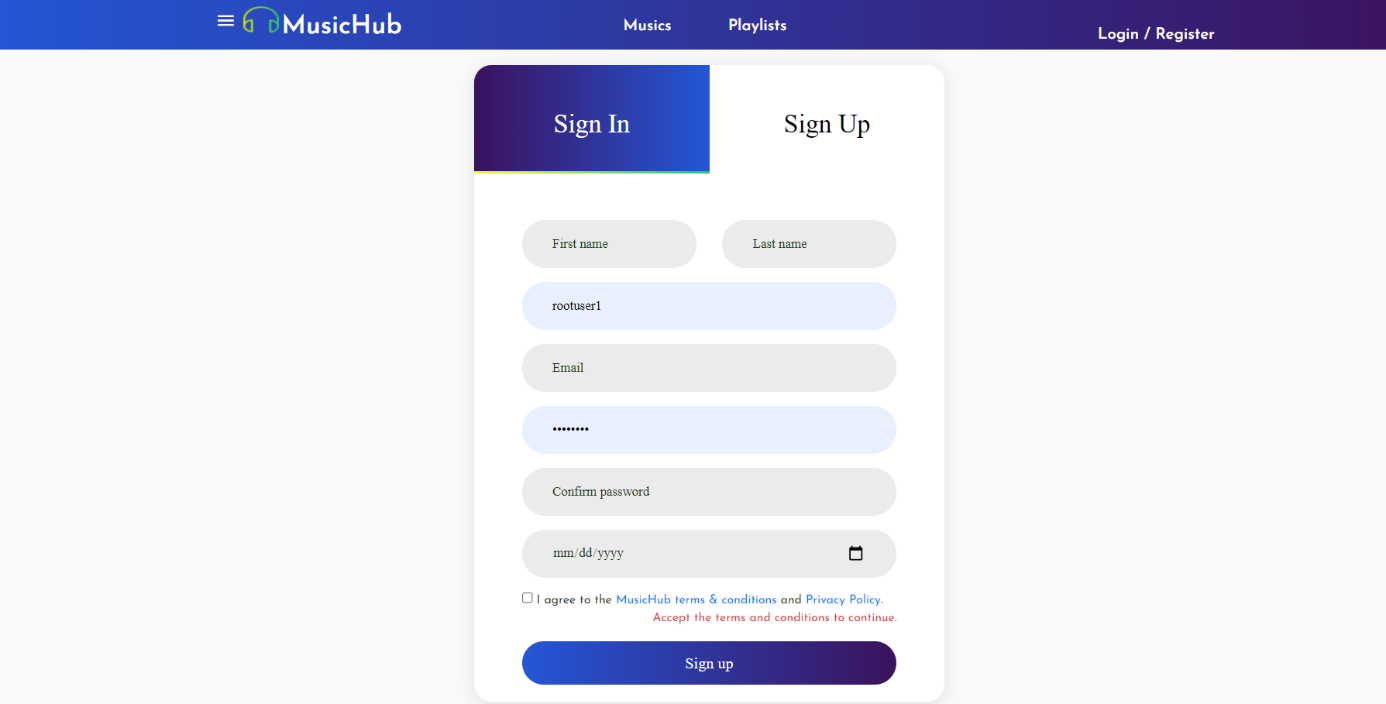
1. Зареждане на приложение

След като приложението зареди, Вие ще попаднете на страницата със всички налични песни в системата. Може да търсете песен по дадени критерии. За по-бързо зареждане на информацията се взимат 12 песни от базата данни. Когато потребителят доближи края на страницата, системата извършва ново взимане на информация от базата данни.

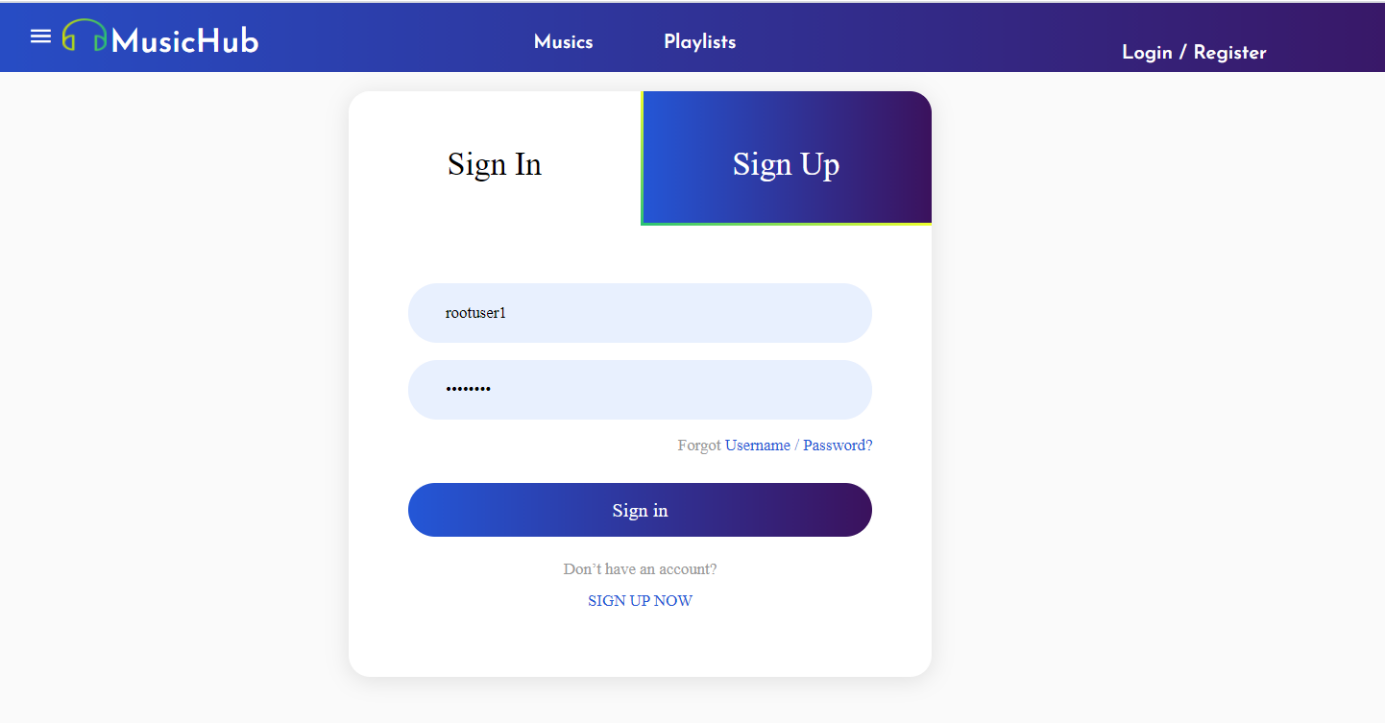
1. Налични playlist

Когато натиснете Playlists от навигационната лента, ще бъдете препратен на страница, където може да видите всички налични playlist. Може да търсете playlist по дадени критерии. За по-бързо зареждане на информацията се взимат 12 песни от базата данни. Когато потребителят доближи края на страницата, системата извършва ново взимане на информация от базата данни. Показват се само тези playlist-ти, които имат поне една песен в себе си.

1. Регистриране на потребители

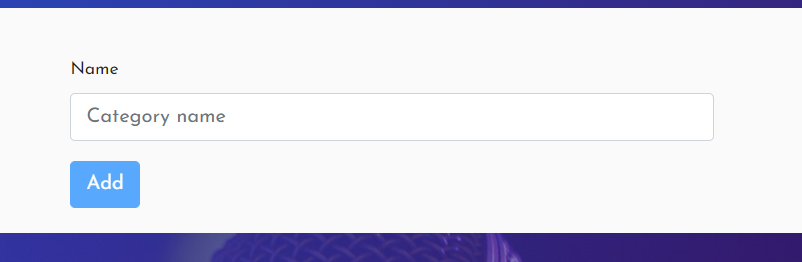
******Натиснете върху „Register“ и попълнете формата с вашите данни. Ако сте попълнили всичко вярно, ще можете да си създадете акаунт в системата. ***Важно: Ще получите email с попълнените от Вас данни!*** ***Когато се регистрирате трябва да се съгласите с политиката на MusicHub teams, което е гарантира законовите права на приложението.***

1. Вход в системата

Натиснете върху „Login“ в навигационния панел и попълнете Вашите потребителско име и парола:

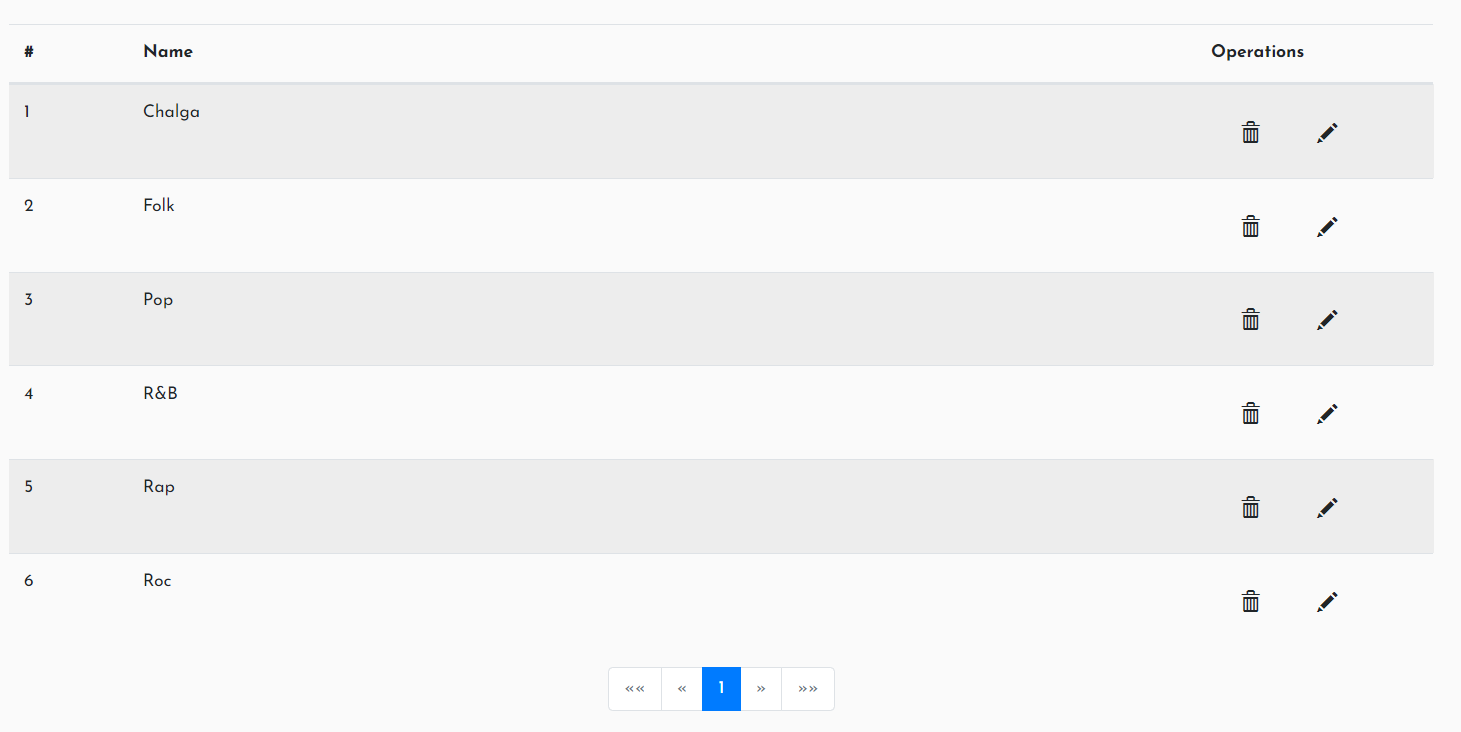
1. Създаване на категория

Всеки потребител с роля на администратор може да извършва основни действия – създаване, редактиране, изтриване на категория. За да бъде добавена дадена музикална категория към приложението, администраторът трябва да попълни формата за създаване на нова категория.

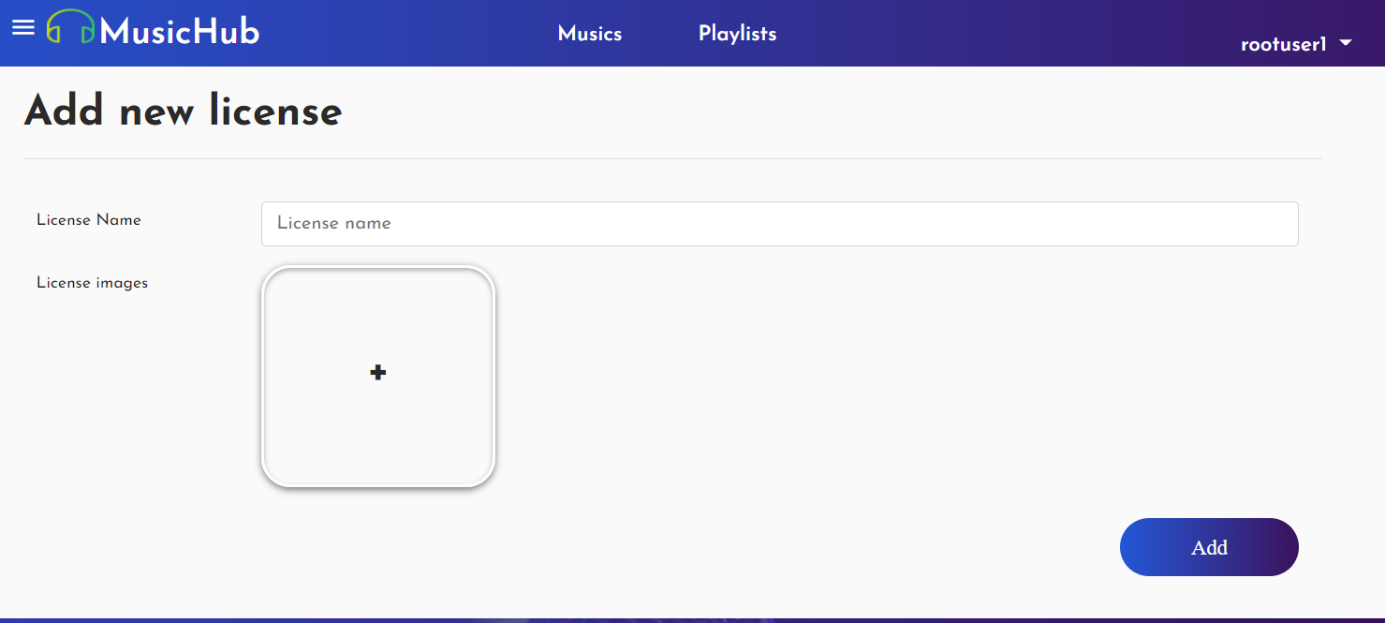


1. Преглед на наличните музикални категории

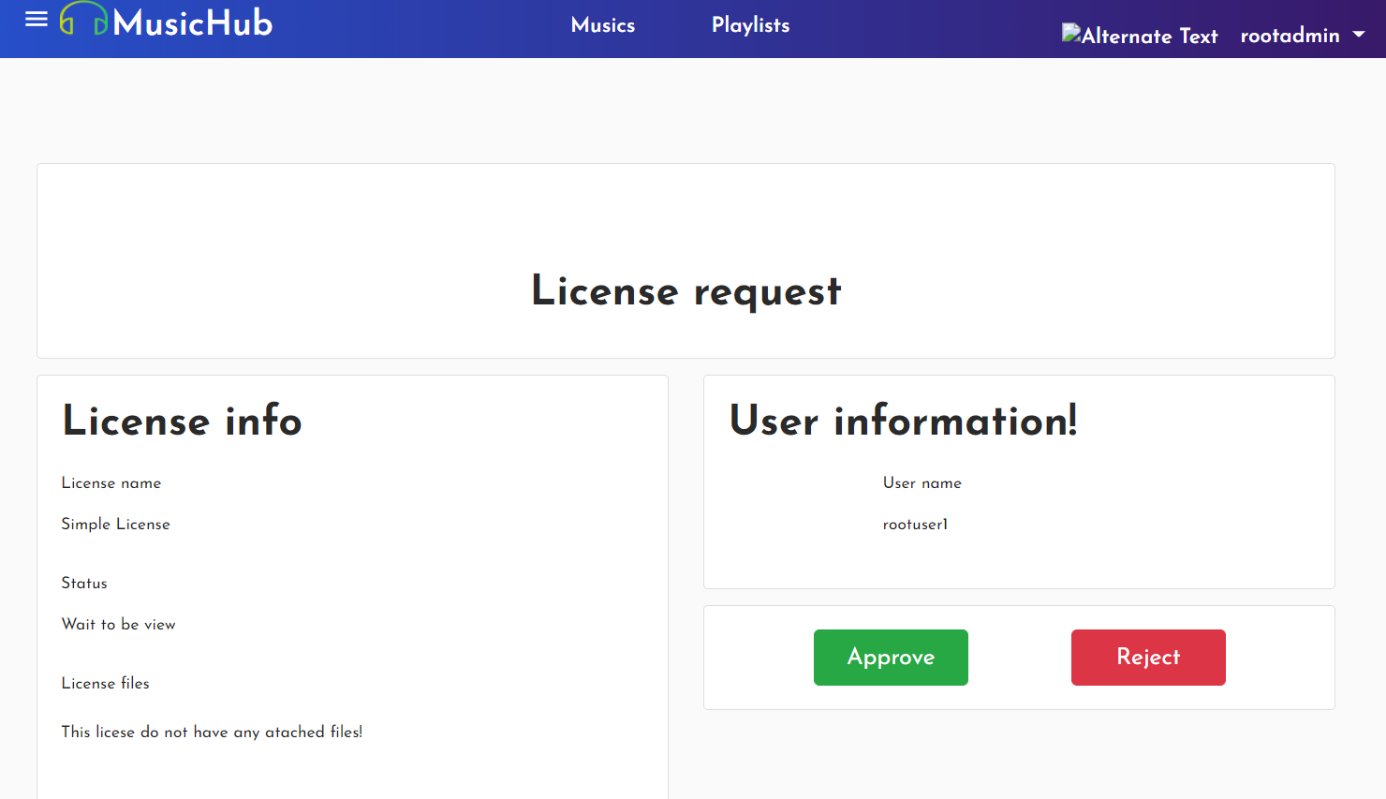
Администраторът има достъп до страница, където може да редактира и изтрива дадена категория. Бутона кошче изтрива категорията, а моливчето – изтрива я.



1. Създаване на лиценз

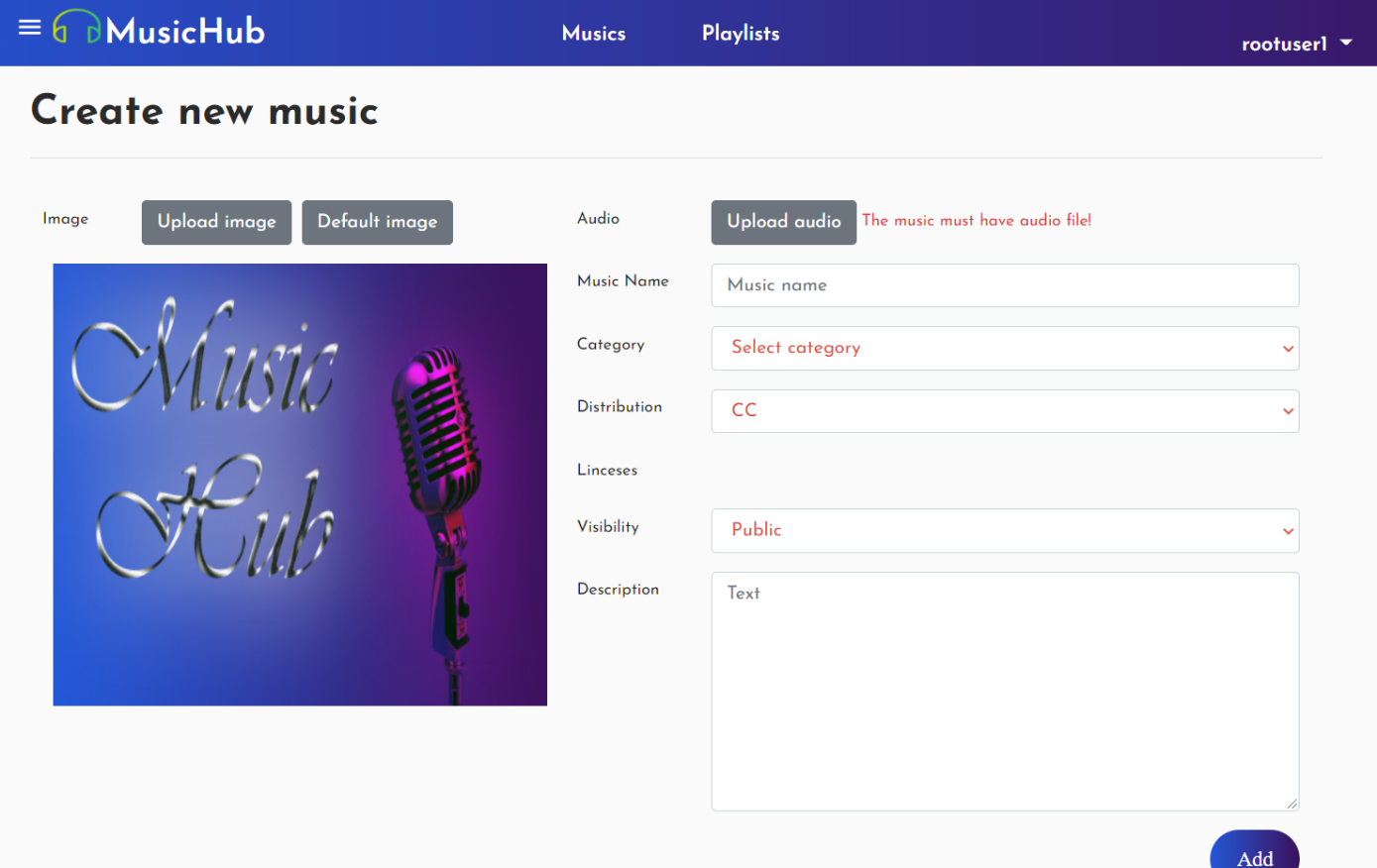
Когато потребител иска да добави музикален лиценз той трябва да избере от страничното меню „My licenses“. Страницата ще се презареди и потребителя ще види всички добавени от него лицензи. За да добави нов лиценз, потребителя трябва да избере бутона „New license“. Ще се покаже форма, чрез която може да се добави нов лиценз. Потребителя трябва да зададе име на лиценза, както и да добави документ, узаконяващ съответния лиценз.

1. Одобрение на лиценз

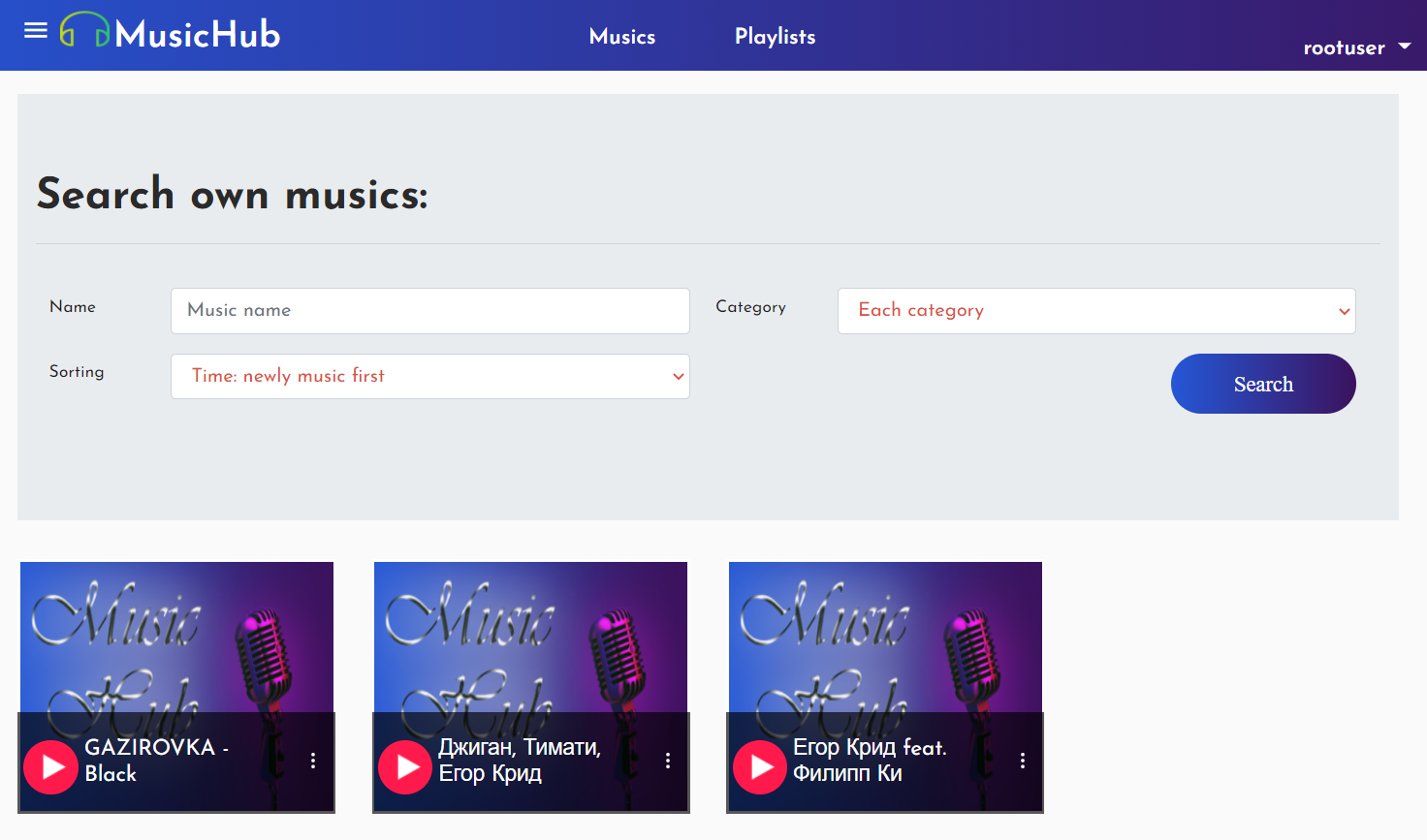
Всеки добавен лиценз в системата минава през администратор, за да го одобри. По този начин се избягва създаването на фалшиви лицензи.

1. Добавяне на нова песен

След като е влезнал в системата, потребител с роля на потребител може да добавя песни към своята колекция от песни. За да направи това действие, той трябва да избере “Add music” от падащото меню. След което ще може да попълни форма за създаване на песен, която изглежда така:

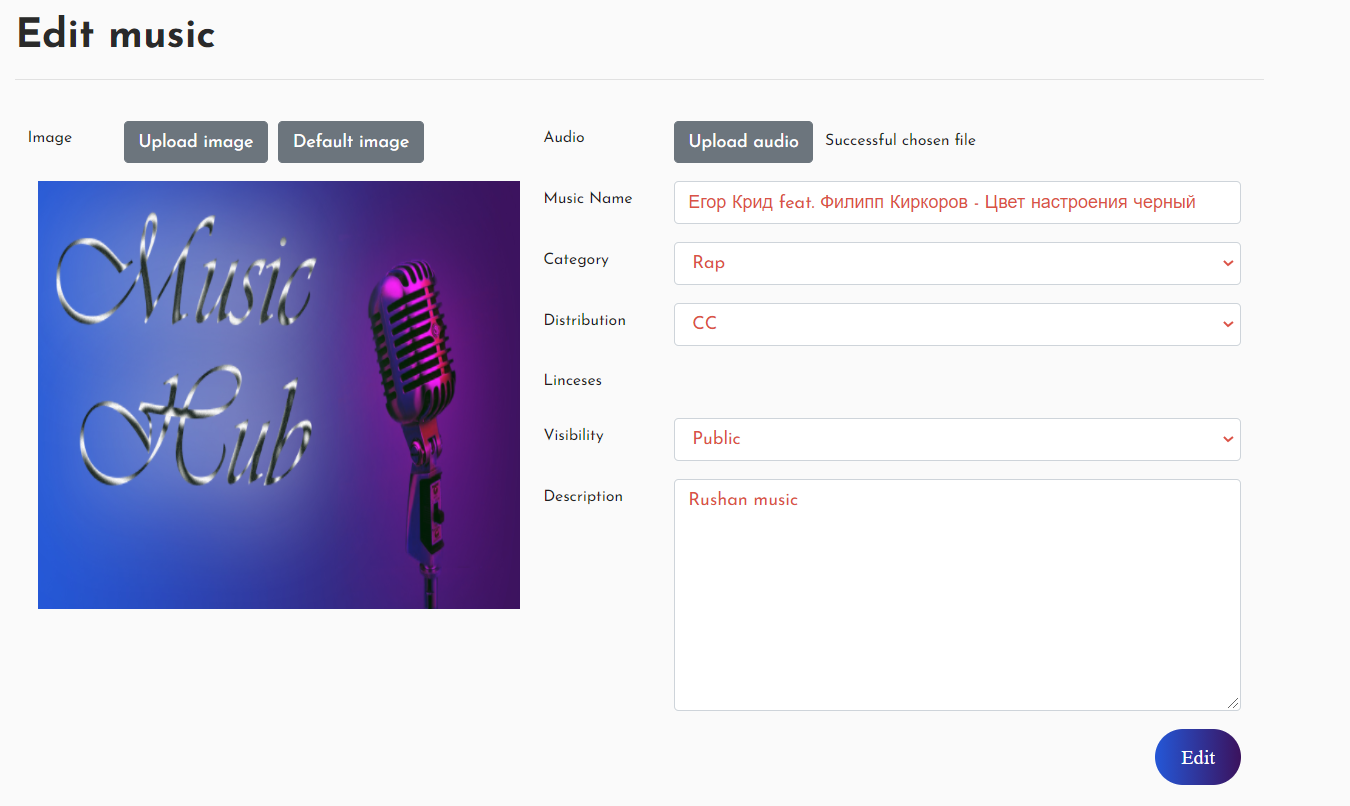


1. Моите песни

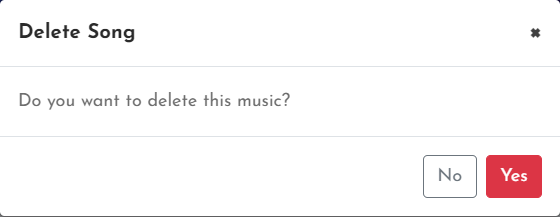
На страницата със всички лични песни, потребителя може да види какви песни има в неговата колекция. Тук потребителя може да търси песен, която е качил в системата. 

1. Редактиране на песен

Всяка песен може да бъде редактирана – да бъде сменено нейното име, категорията ѝ, снимката и друга информация за песента.

1. Изтриване на песен

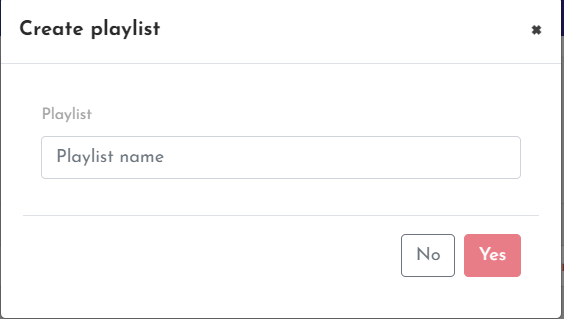
За да изтрие песен, потребителят, който я е качил, трябва да я намери в своите песни и от многоточието до песента да избере Delete. Ще се отвори изкачащ прозорец, чрез който потребителят потвърждава своето действие.



1. Моите плейлисти

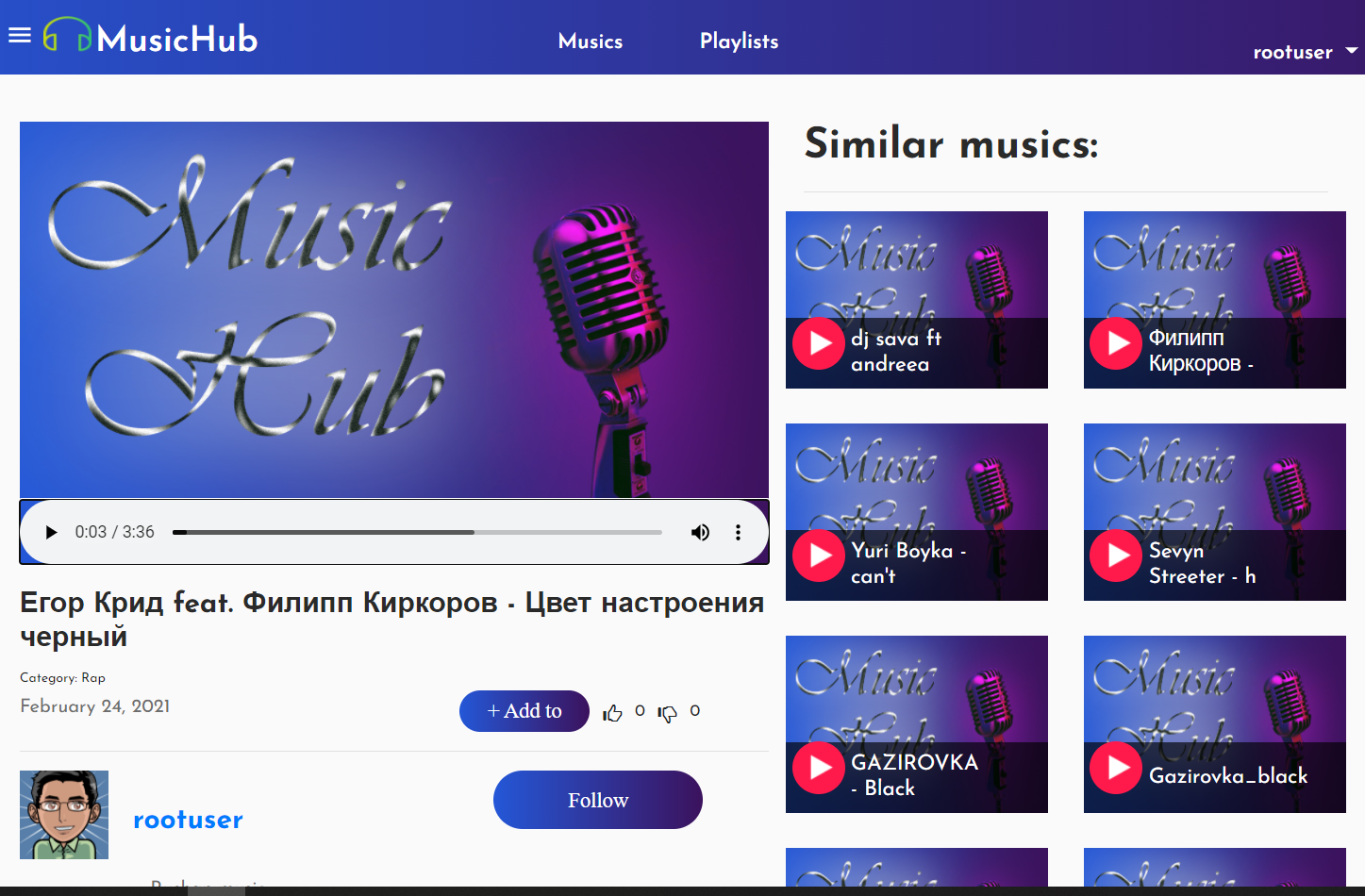
Тук потребителя има достъп до всички направени плейлисти от него, които имат повече от една песен в тях. За по-бързо намиране на любимия плейлист, потребителят може да го намери като напише неговото име в панела за търсене на плейлист. За по-бързо зареждане на информацията се взимат 12 песни от базата данни. Когато потребителят доближи края на страницата, системата извършва ново взимане на информация от базата данни. Показват се само тези playlist-ти, които имат поне една песен в себе си.

1. Създаване на нов playlist

****Когато потребител иска да направи нов playlist, той трябва да натисне бутона „New playlist“. Изкача прозорец, чрез който потребителят може да направи новия playlist.

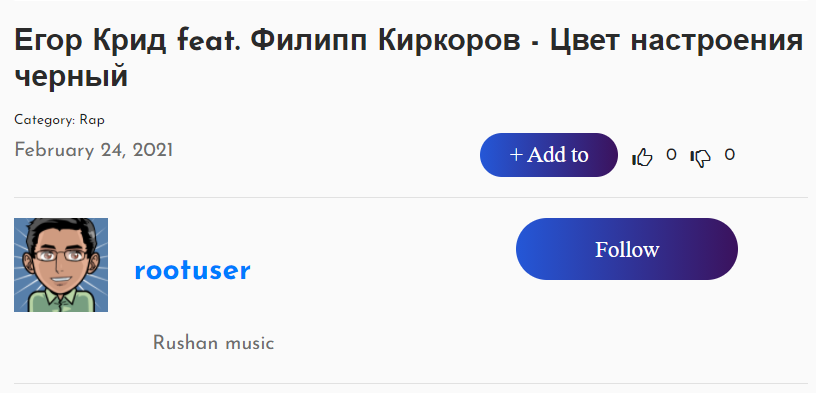
***ВАЖНО: За да бъде видим един playlist, то рой трябва да съдържа поне една песен в себе си!***

1. Слушане на песен

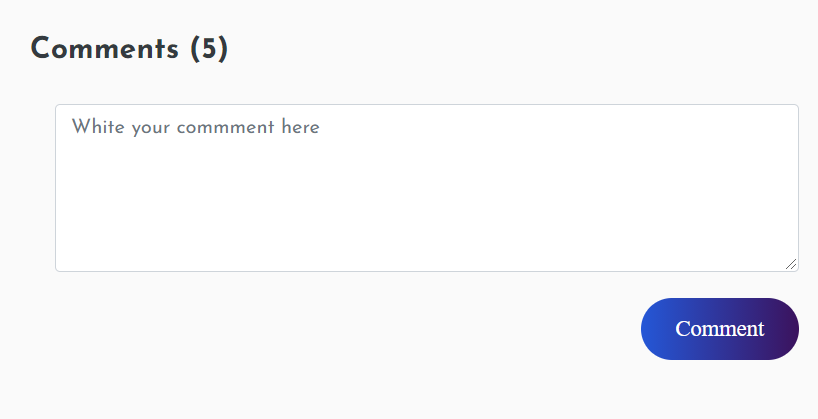
На страницата за слушане на музика, потребителя може да изслуша песента. Тя се стартира автоматично. Показва се името на песента, категорията ѝ, кога и от кого е създадена, както и допълнителна информация за нея.

1. Харесване/не харесване на песен

Всеки, влезнал с профила си потребител, може да даде изкаже своето вижда дали песента е добра или не. Това действие може да бъде направено чрез бутоните палец нагоре/надолу.

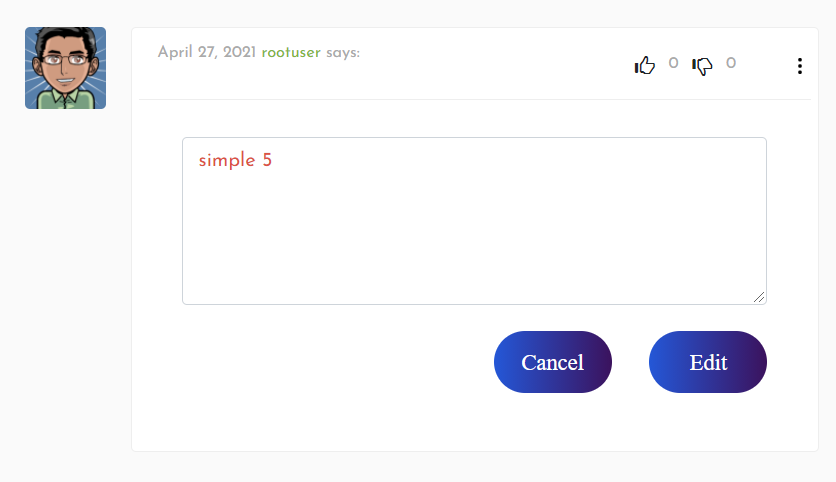


1. Създаване на коментар на песен

Всеки, влезнал с профила си потребител, може да даде своето впечатление относно музикалното изпълнение. За да го направи, той трябва да изкаже своето впечатление в кутийката за текст под песента и да натисне бутона „Comment“

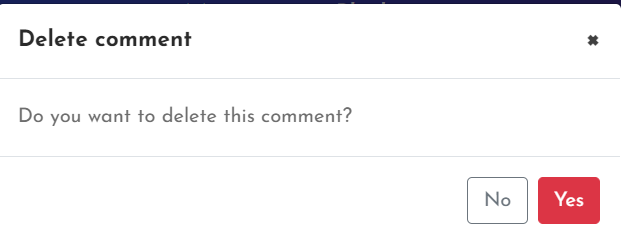
1. Редактиране на коментар

Ако потребителя е допуснал грешка или иска да редактира своя коментар, това може да бъде направено чрез многоточието до коментара на съответния потребите. Коментарът ще може да се редактира.



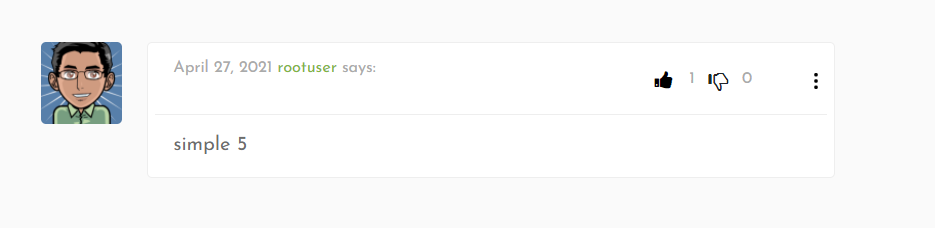
1. Изтриване на коментар

Всеки направен коментар може да бъде изтрит от неговия създател. След като потребителя натисне на трите точки и Delete, изкача прозорец за потвърждаване на действието.

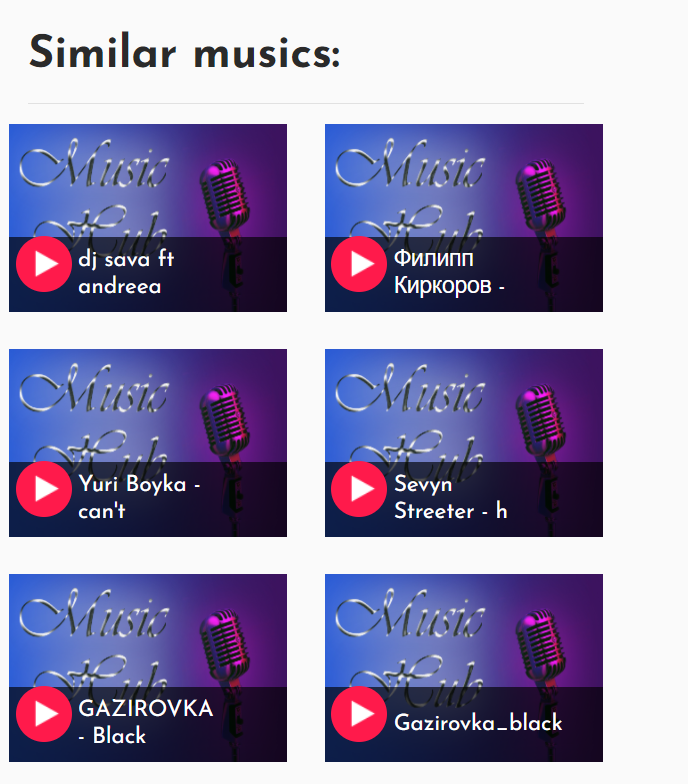


1. Реагиране на коментар

Както песните, така и коментарите могат да бъдат оценявани чрез палец нагоре/надолу.

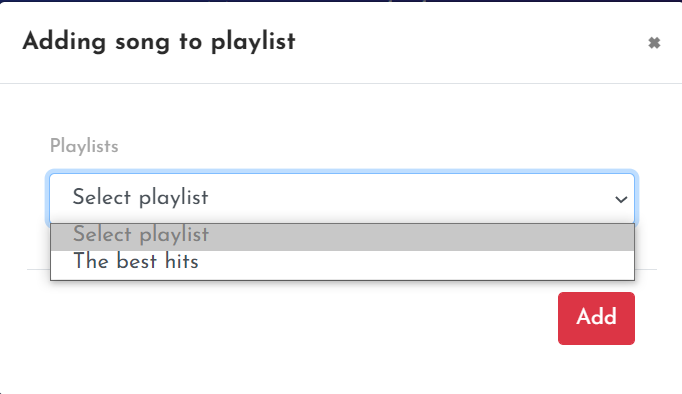


1. Подобни песни на слушаната

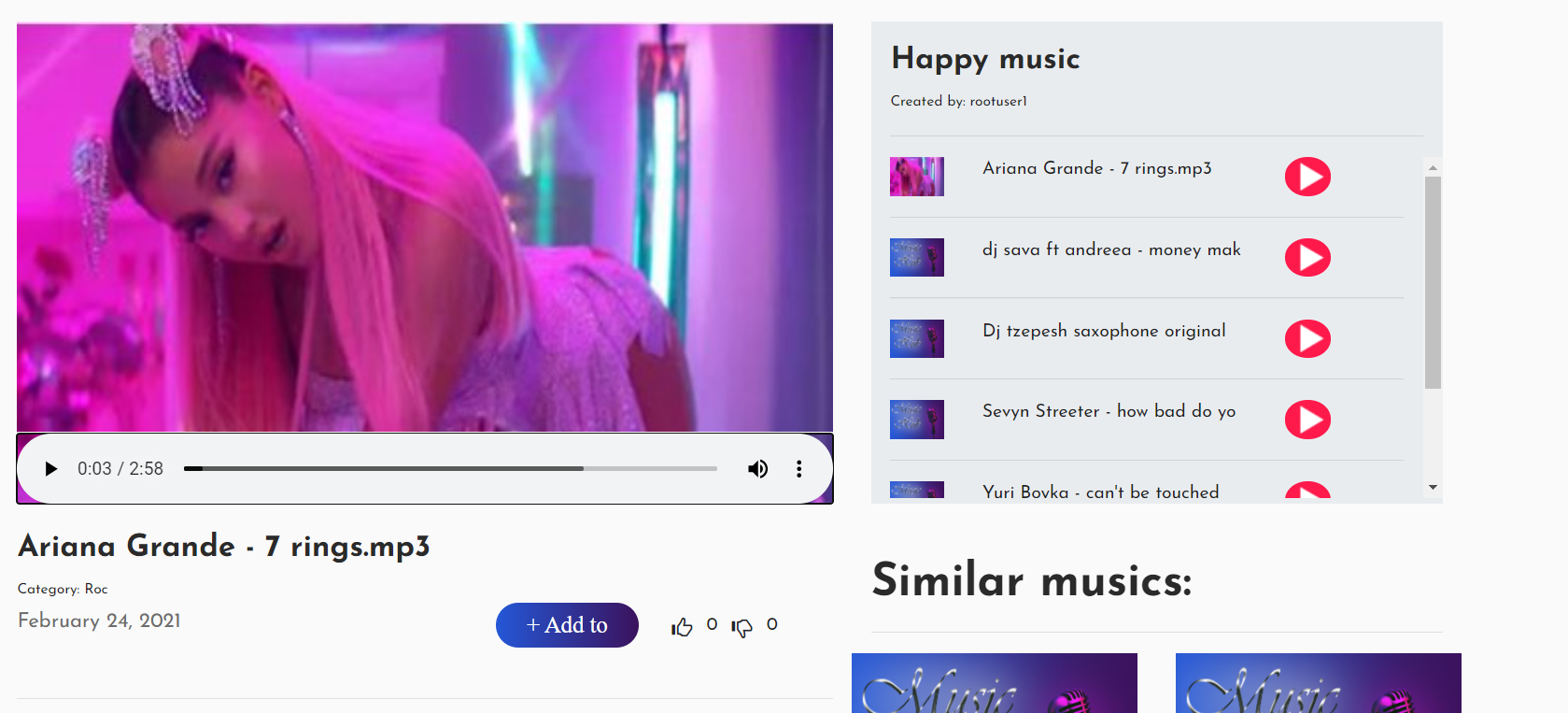
В дясно от слушаната песен потребителят може да види подобни песни на слушаната. Те са подредени според категорията, кога са качени, колко често са слушани, какви оценки са давали потребителите за тях и други критерии.

1. Добавяне на песен към playlist

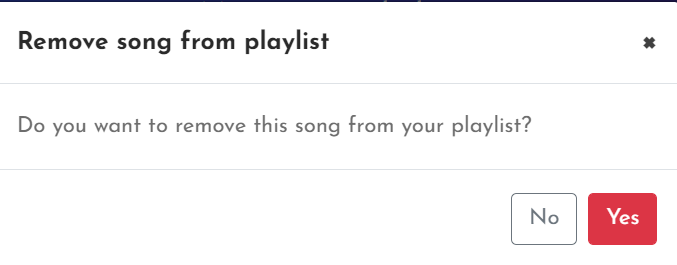
За да добави песен към playlist, потребителят трябва да натисне бутона „+Add to“. Ще се отвори прозорец, чрез който може да избере към кой playlist да бъде добавена песента. Показват се само тези playlist-и, които не съдържат песента.



1. Слушане на playlist

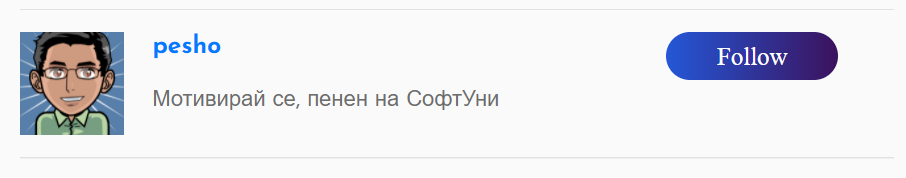
Когато потребител слуша playlist. В дясно от песента има списък на всички песни в playlist.

1. Премахване на песен от playlist

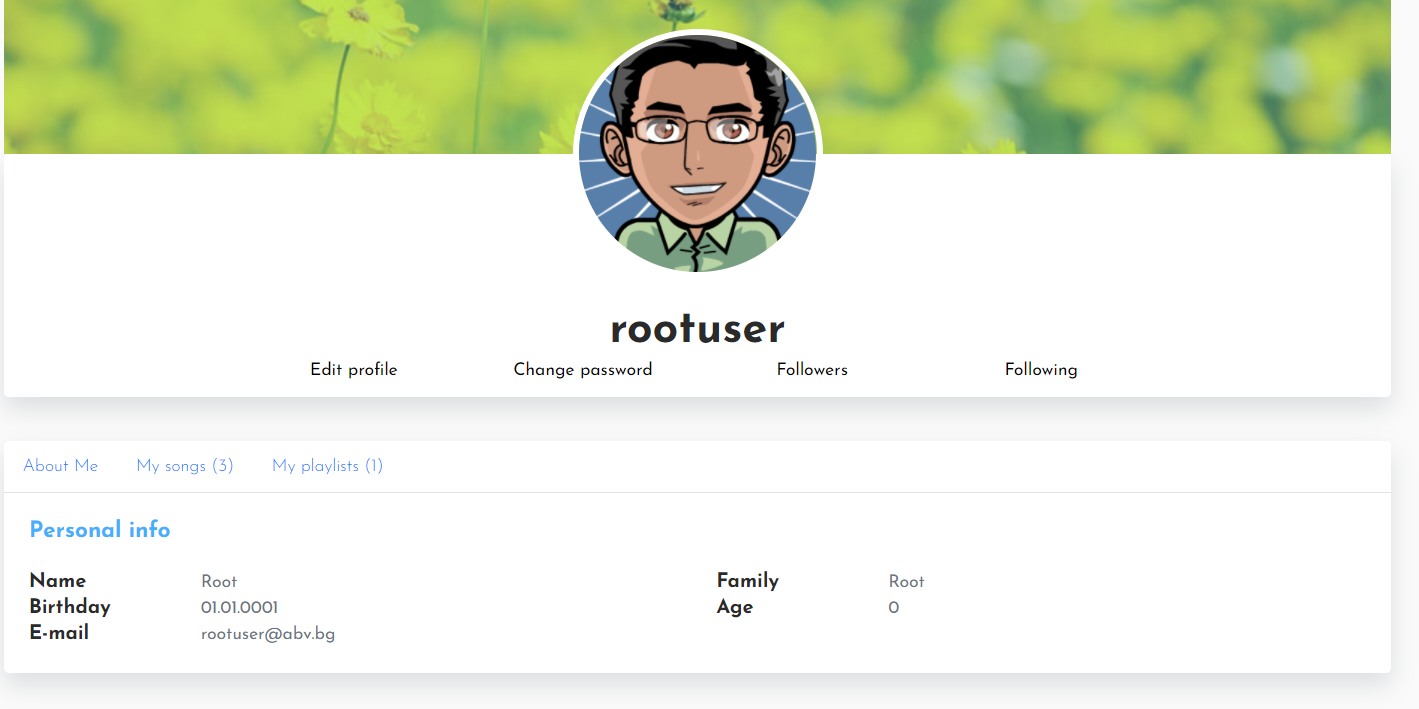
Ако playlist-а, който слуша потребителя в момента е негов, то той може да премахва песен от него чрез точките до съответната песен в playlist-а. Изисква се потвърждаване на действието. 

1. Следване на потребител

Потребителите могат да се следват чрез бутона „Follow“.

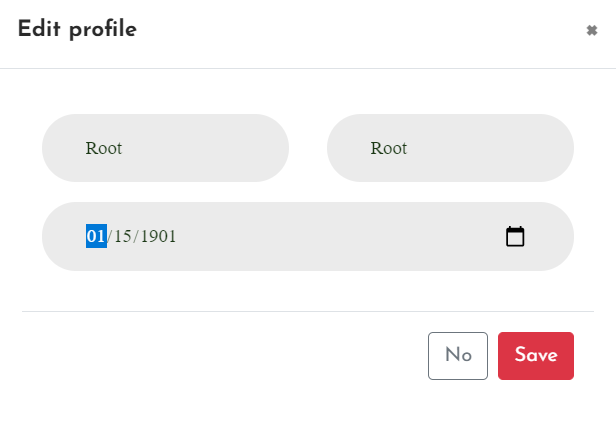


1. Профил на потребител

Всеки потребител може да разгледа личния си профил. Тук може да види персонална информация, кои потребители следва, от кого е последван, да види собствените си песни и playlist-и.

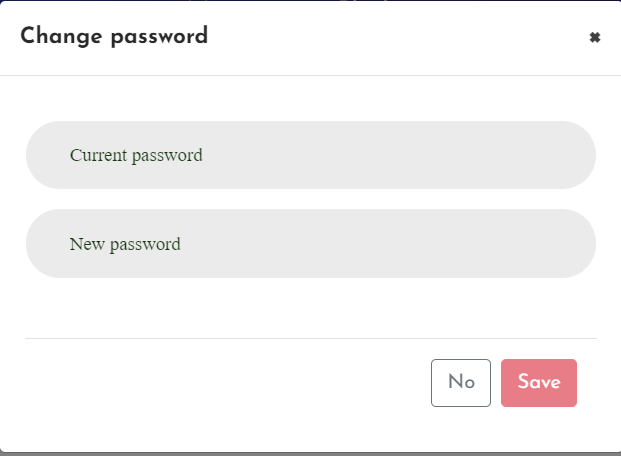
1. Редактиране на профил

Всеки потребител може да смени своите данни от прозореца Edit Profile



1. Смяна на парола

Паролата да бъде сменена. За да направи това действие потребителя трябва да натисне върху Change Password. След което да зададе нова парола. При натискане на Save, паролата ще бъде сменена.



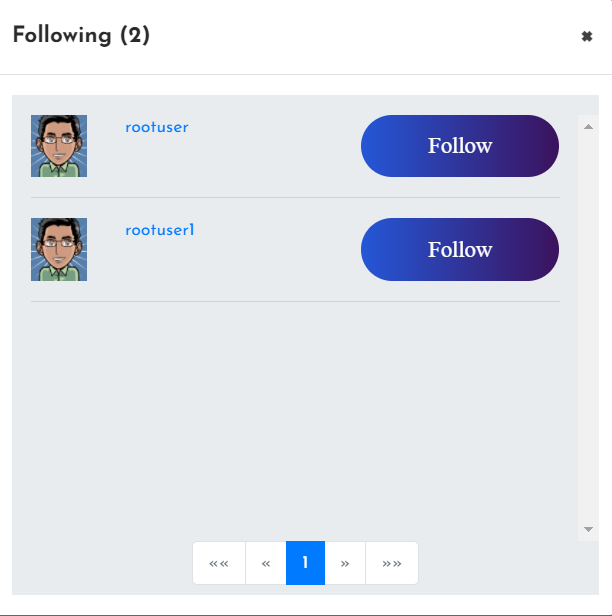
1. Профил на други потребители

Потребителите могат да разглеждат профилните страници на други потребители, като могат да видят кака музика и какви playlist-и са създали, кой е последвал даден потребител и кого следва той.



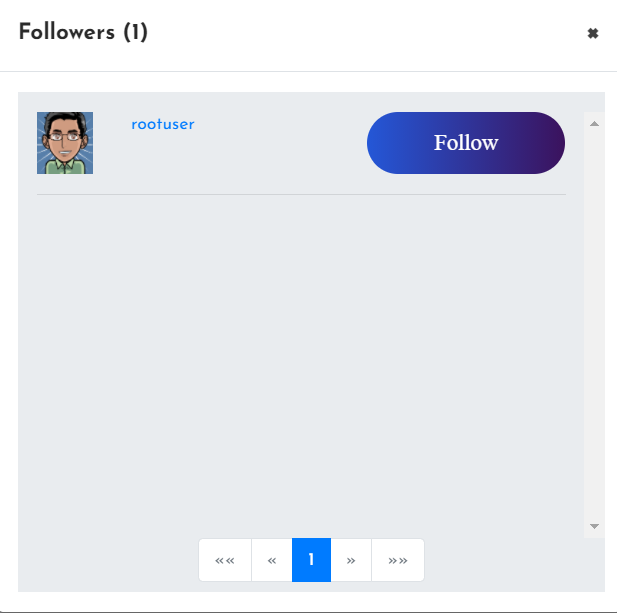
1. Последвани потребители

Изкачащият прозорец показва кои потребители са следвани от даден потребите.



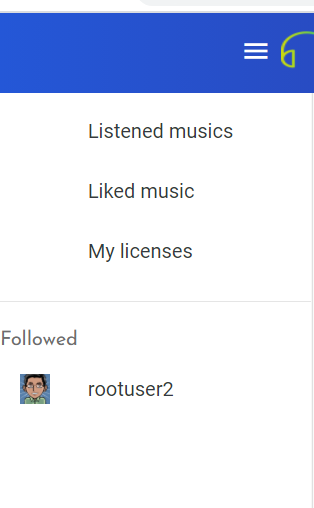
1. Потребители, които следват даден потребител

Изкачащият прозорец показва кои потребители следват даден потребите.



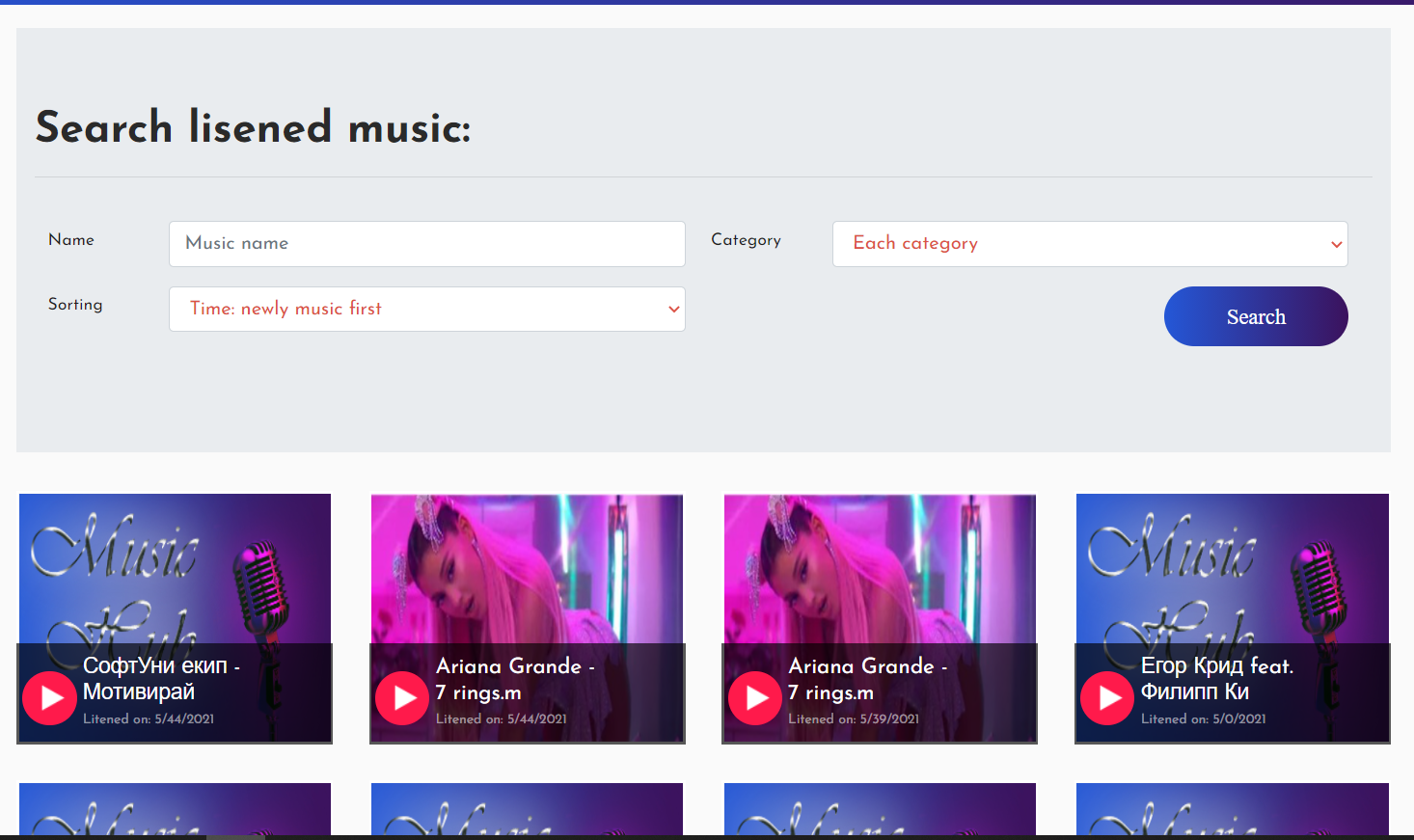
1. Бърз достъп до последваните потребители

В страничната лента всеки потребител има бърз достъп до потребителите, които следва. По този начин той бързо ще слуша харесваните от него изпълнители.



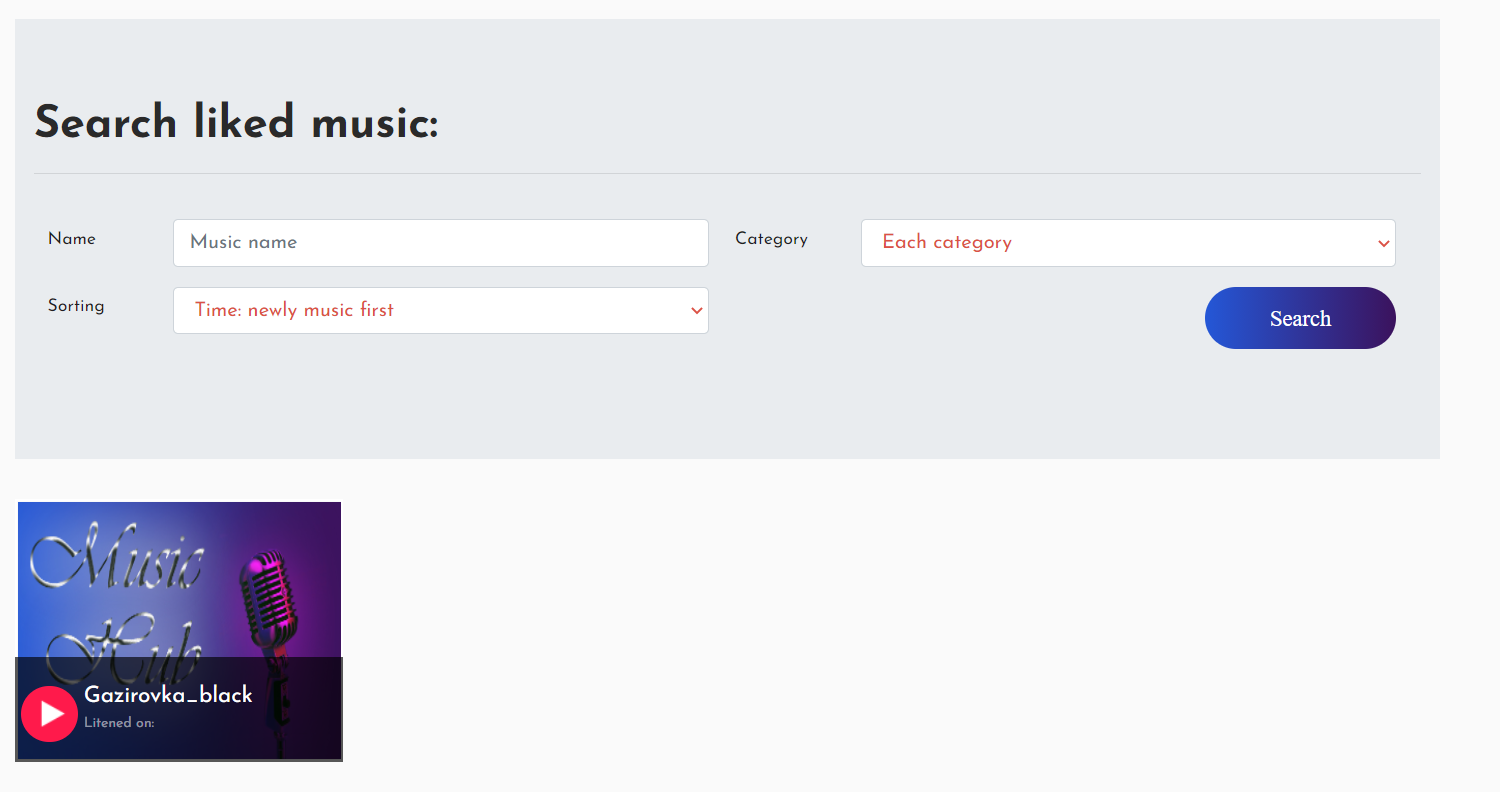
1. Слушани песни

На тази страница потребителите могат да видят кои песни и кога са ги слушали.



1. Харесани песни

Всеки потребител има страница с песните, на които е дал палец нагоре и ги е оценил като добри за него самият.



# **Заключение**

Приложението може да се използва в сегашния си вид. С цел потребителят да изживее повече добри моменти с платформата, съм планирал да добавя:

* Subscribe – слушателите да имат възможност да следват други потребители и да имат бърз достъп до техните песни
* 2D, 4D, 8D… filters – при качване музиката да има възможност да стане 2D, 4D, 8D и други. Когато е слушан на слушалки, песента да звучи сякаш обикаля около слушателя;
* AI filter – Чрез AI музиката да си променя стила, ако потребителят иска. Например песен с рок звучене да може да звучи като поп музика; песен с поп звучене да може да звучи като джаз песен.
* Merch shop – Един мой приятел ме научи на много нови неща. Затова аз реших да предам своите познания в училище - на съучениците ми. Показвам им как биха могли да си направят приложение, като има за цел да реализира онлайн магазин за продажба на стоки към приложение “Music hub”, а те да си направят собствени сайтове. Целта ми е да подам ръка и да дам нови знания на учителката си и на приятелите си, защото така се чувствам успял и ми е по-добре.

**Приятно слушане на музика!**