TALLINNA UNIVERSITY OF TECHNOLOGY

School of Information Technologies

Music discovery and sharing platform

Author: Marc Täll 232696IADB

Instructor: Andres Käver

Tallinn 2025

Contents

1 Project description 3

2 Project idea 4

3 Project features 5

3.1 Home page 6

3.2 Discovery page 7

4 ERD model 8

5 Project architecture 9

Summary 10

Sources 11

1 Project description

This web application aims to create a music discovery platform for independent musicians to gather listeners and feedback on their music.

A big problem today is that it is difficult for new musicians and producers to gain popularity. This is mainly because there are so many already popular creators out there and the competition is very big. The main ways of releasing music today are mainly mainstream platforms such as YouTube, Spotify and SoundCloud, which rely on algorithms that favour already popular artists. This means that there are many artists, who stay unrecognized and are overshadowed by big names and as a result fail to grow an audience.

According to different sources, 103 500 new tracks are uploaded every day to music streaming services such as Spotify, YouTube Music etc. When you also include SoundCloud, where uploading tracks are free for all accounts, the number increases by 120 000.

According to Luminate's 2023 data:

- Out of all the existing tracks in their catalogue, there were 158.6 million (around 86%) tracks, which received fewer than 1 000 streams over the entire year.
- 45.6 million tracks received 0 streams.
- Top 0.1% of the most popular tracks receive 40% of all listens.
- Top 1% receive 75-80% of all listens.
- The rest of the 99% of tracks receive 20-25% of all listens.

This means that most new tracks and artists remain practically invisible.

2 Project idea

This project aims to create a similar music platform, that instead focuses on less known artists and producers, where popularity doesn't matter. This gives every artist a fair chance to be heard. Artists can create their profile, upload their music by their musician's name and provide links to redirect traffic to their own platforms.

The second problem that the author has faced many times is finding new songs and music to enjoy, that depends on his mood. He finds it difficult and tedious to find new songs and music to his liking, because mainstream platforms promote popular music, which is not what he is usually looking for. This is also confirmed by many articles, where people are frustrated with popular streaming services showing them music, that they do not want to listen to instead of personalized music.

This project would aim to solve two different problems with one solution: it would help new artists grow by helping their songs reach new audiences and on the other hand it would help people discover new genres and types of music that they are looking for.

The motivation for this project comes from the authors own experience with producing music. The author plays the guitar and produces music in Fl Studio (music producing DAW). Although the author does this just as a hobby, not to gain recognition, he has noticed these problems with the music industry and prefers smaller communities to share their creations.

3 Project features

The web app features a home page and a discovery page. The front page would include a monthly featured artist, who has received the most ratings, comments, saves and voted by the community. This feature isn't to promote bigger artists, but rather to help artists every month to gather even more of an audience and inspire people to also try and get featured (everybody has a fair chance). The front page would only show only one monthly artist, to make users prioritize the discovery page, instead of looking through popular tracks, which would defeat the whole purpose of this web app.

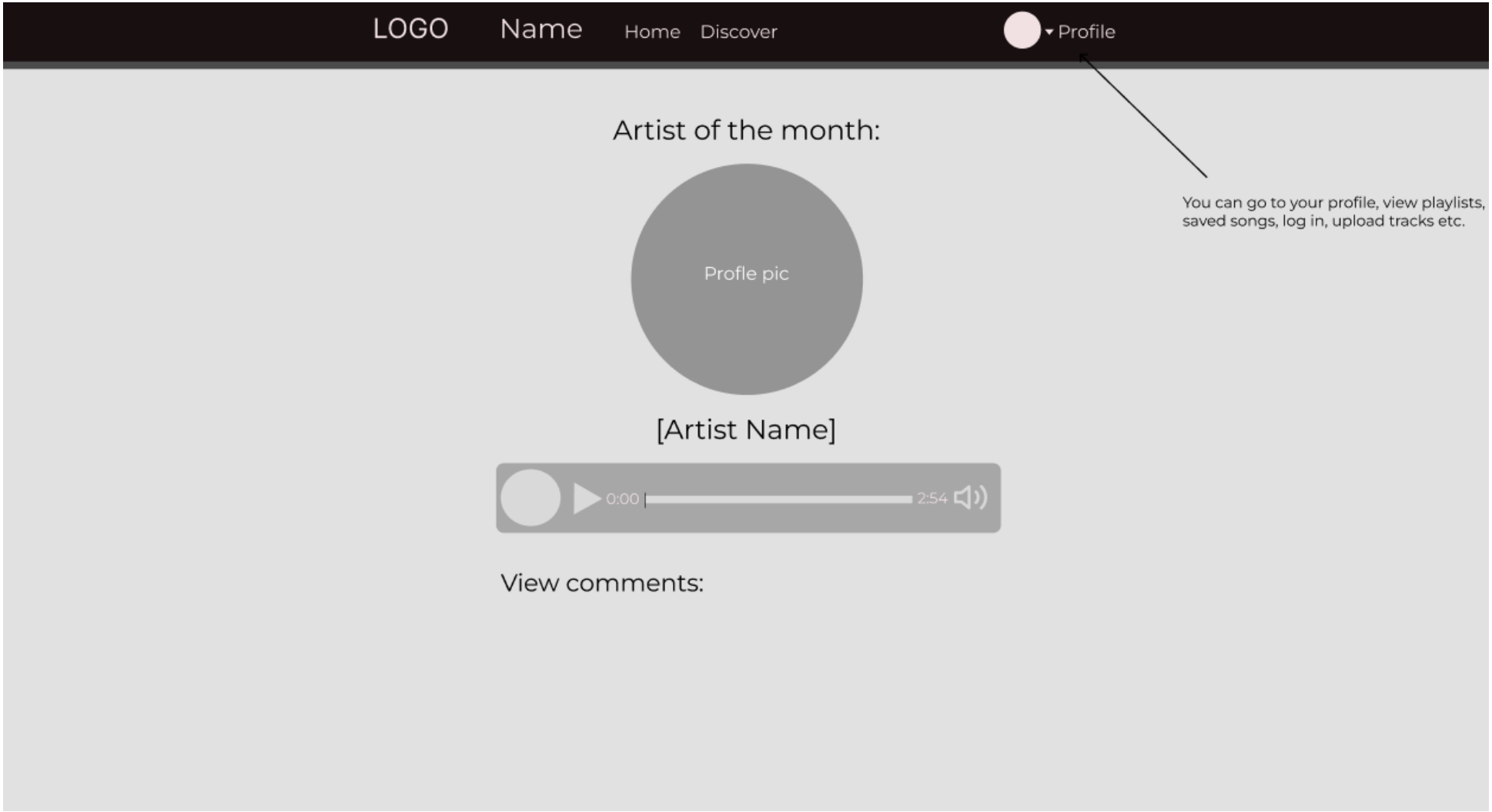
The discovery page is where you can discover new songs and music. You will be able to set your preferred tags and mood. Then based on your preferences you can play/scroll through random uploaded tracks, and you can rate them, leave feedback, save them and skip them. The "likes" on a track should only be visible to the artist himself, to avoid popularity bias.

In your profile view you can see and edit your profile, view saved/liked songs, create playlists, upload tracks and log into the website.

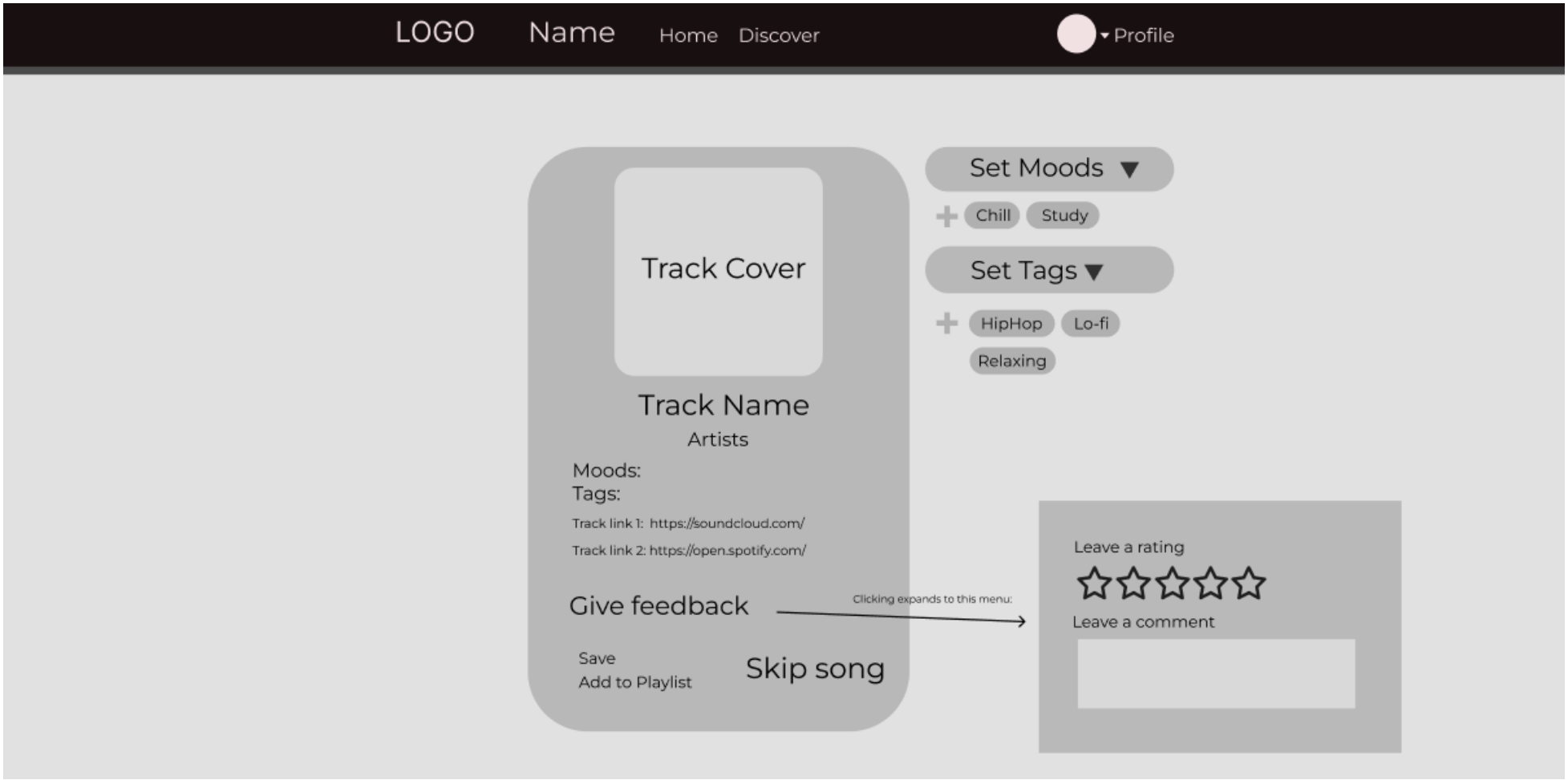
The web app will have a very simplistic, modern and functional UI. The style will be sleek and appealing, to engage users. The UI sketches are a rough representation of what the website will look like.

The UI sketches are done in Figma.com due to the author having previous experience with it. The ERD is made with Vertabelo.com for the same reason.

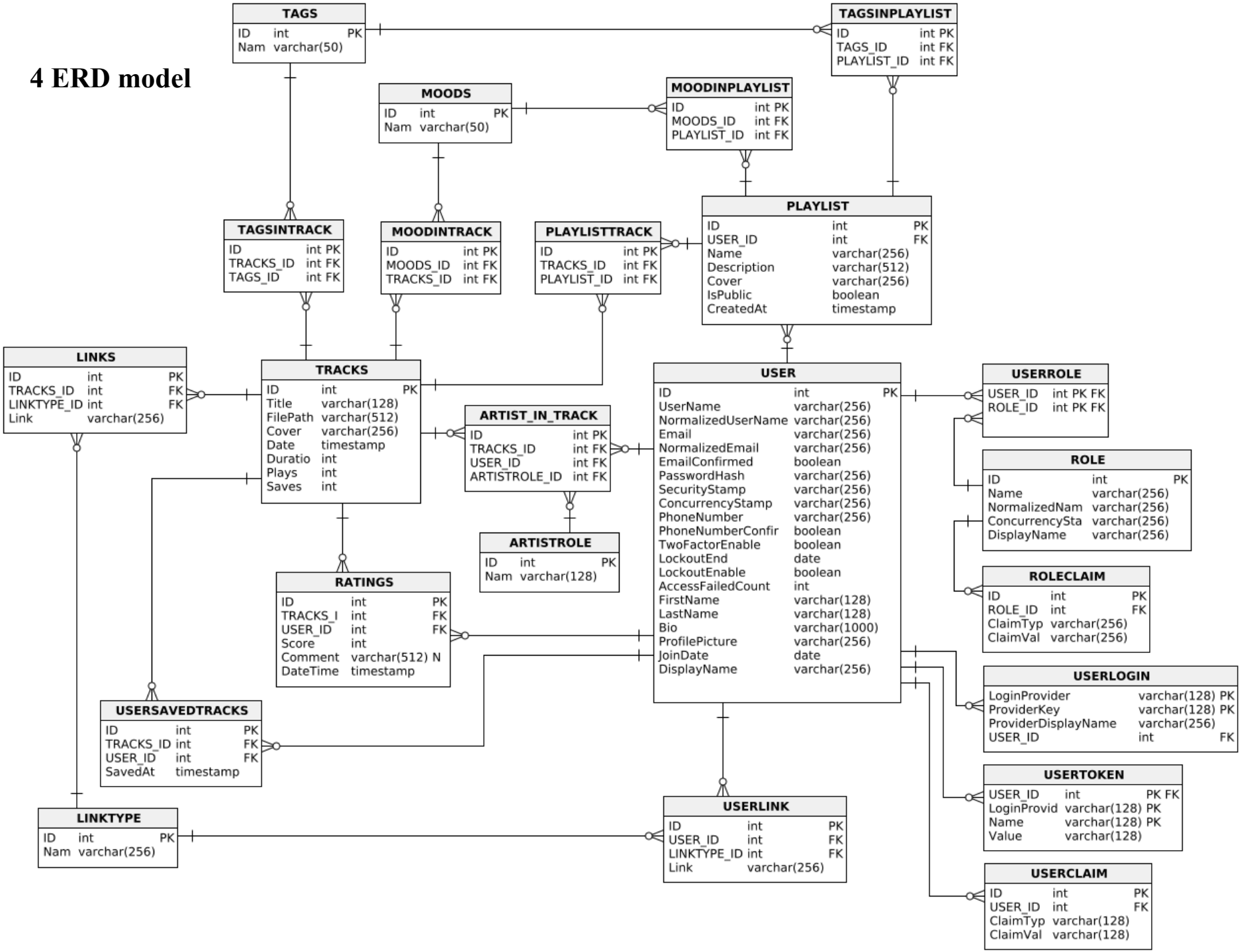
3.1 Home page


LOGO
Name
Home
Discover
Profile
You can go to your profile, view playlists, saved songs, log in, upload tracks etc.
Artist of the month:
Profle pic
[Artist Name]
0:00
2:54
View comments:

3.2 Discovery page



4 ERD model



5 Project architecture

The project follows many different separation of concerns ideas and different layers. This allows for increased scalability for the website in the future and keeps everything organized and readable.

The project includes the following **layers** in order from database to client:

1. **Domain layer** – this layer is used only for creating a Postgres database and contains domain models for each database entity. This also contains data seeding, which automatically creates certain needed entries in the database on creation, such as different tags and an admin user.
2. **Entity Framework Data Access Layer (DAL)** – this layer is responsible for creating the datasets into the Entity Framework, so they can be used in .NET context. DAL communicates with the domain layer and accesses the database.
3. **Unit Of Work (UOW)** – this layer is responsible keeping the projects repositories, which make queries to the database and passes it along to the next layer and vice versa (used for saving EF entities to the database).
4. **Business Logic Layer (BLL)** – this layer is responsible for keeping services, which communicate with UOW and the outer layer. The services perform logic-based calculations and doesn't make queries. The queries are passed on from the UOW layer.
5. **API** – this layer is used for displaying everything to the client including creating entities, which are then passed down to the lower layers and vice versa. This layer contains controllers.

Between every mentioned layer, there are DTO mappers, which map entities from one layer into the other. The project features a frontend hosted in VUE, which uses REST and an admin interface with MVC controllers. The account and identity use EF identity system with roles and uses JWT and refresh token-based approach. The final backend is hosted on docker and the frontend can be found on the authors *enos* page. Resources are user specific, so only correct users can access their information.

Finally, the project includes unit and integration testing. Unit testing focuses on the BLL layer and is used for testing different service methods. Integration testing covers user flow through the website (logging in, opening the front page, discovering songs, creating songs etc.)

Summary

It is very hard for new producers and musicians to get recognition and listeners due to mainstream platforms promoting popular creators. This project aims to create a music sharing platform, where artists can upload their music and have a fair chance to be heard. The web page will let people discover new songs by their preferred tags and mood, without popularity bias, as all songs played to them will be random. This is a great way of gaining feedback also.

The author had a lot of fun and gained new experience when creating the project. The work turned out to be a lot harder than anticipated, but he is proud of the effort he put in. Due to time constraints, the author couldn't implement playlists with their own tags and moods, so for the time being the client can only save their tracks and view them from their saved tracks page without sorting them into playlists. The author focused on making the webapp functional, and the playlists are not necessary for the initial website to function and can be added in the future, if the author decides to continue the project in his free time.

In summary, the application user can:

- view the monthly featured artist.
- explore other users' pages and view their songs.
- use the discovery page to discover new music by filtering with their preferred tags and moods, then skip or save the songs.
- leave feedback on tracks with ratings.
- view their profile information.
- view their tracks and track information – times saved, plays, comments and ratings.

Sources

https://courses.taltech.akaver.com/web-applications-with-csharp/assignments/home-assignments/project-proposal

https://my.vertabelo.com/

https://www.figma.com/

https://www.musicbusinessworldwide.com/158-million-tracks-1000-plays-on-streamingservices/#:~:text=streaming%20services%20per%20month%20last,year

https://www.hypebot.com/hypebot/2023/02/oops-100000-tracks-a-day-are-not-uploaded-to-spotify-apple-amazon-music.html#:~:text=It%20turns%20out%20that%2C%20while,an%20average%20day%20last%20year

https://www.gov.uk/government/publications/equitable-remuneration-er-in-the-streamingage/the-potential-economic-impact-of-er-on-performers-and-the-music-market-in-theuk#:~:text=Creators%20Earnings%20report%20found%20that,footnote%201

https://crackmagazine.net/article/long-reads/quitting-music-streaming-platformsspotify/#:~:text=From%20poor%20pay%20for%20artists,to%20quit%20them%20for%20go od

https://www.newyorker.com/culture/infinite-scroll/why-i-finally-quitspotify#:~:text=technology%20create%20issues%20with%20the,know%20what%20you%20 really%20do