Track ID

An application that allows users to download audio tracks based on a given title and play them directly in the browser. The project aims to simplify access to music and automate the download process.

Target Group

# Users looking for a simple tool to download audio tracks

# Users who want to play downloaded tracks directly in the browser

# People without advanced technical knowledge

## Main Features

## MVP

## Input form for the track title

## Download audio file to the user's local computer

## Play the downloaded track in the browser

## Log of downloaded files (title, download time)

Extended Version (Optional)

# Download tracks in various formats (mp3, wav)

# Support for multiple download sources (e.g. YouTube, SoundCloud)

# Download history with the option to browse and replay

# Download progress preview and notifications upon completion

# Playlist support (automatic download of multiple tracks)

# Generation of a report with downloaded tracks

# Technologies

# Backend: Python (Flask/FastAPI)

# Frontend: React

# Database (optional): SQLite

# Other: Docker (containerization), Python helper scripts

Future Development Opportunities

* Integration with user accounts
* Expansion of download history and statistics (data analysis)
* Playlist support and automatic downloading of multiple tracks
* Integration with external APIs (e.g. Spotify, YouTube Data API)
* Mobile version expansion

Architecture (MVC Pattern)

# Model: TrackInfo class with track data

# Service: DownloadService and StreamingService handling downloads and playback

# View/Interface: React interface with a form, buttons, and player

# Controller: APIController handling requests from the frontend

### Backend Classes

### TrackInfo

Description: Contains track information

title: str

stream\_url: str

download\_url: str

source: str

### DownloadService

download\_track(title: str) -> TrackInfo – returns a link for download and streaming

### StreamingService

get\_stream\_url(title: str) -> str – searches for a streaming link

### APIController

POST /search – accepts a track title and returns TrackInfo