



# Projektbericht Softwaredesign

**Audioerkennungsoftware BA-MECH-22-SWD**

**Abschlussprojekt Softwaredesign**

**Bachelorstudium - Mechatronik, Design and Innovation**

**3. Semester**

**Lektor: Dr.rer.pol. Huber Julian**

**Gruppe: BA-MECH-22**

**Autor: Andreas Thöni, Hannes Unterhuber, Peer Samuel**

**27. Februar 2024**

# Inhaltsverzeichnis

<b>1 Aufgabenstellung</b>	<b>1</b>
<b>2 Installationsanleitung</b>	<b>1</b>
<b>3 Programmfunktionen / Erweiterungen</b>	<b>1</b>
3.1 Minimalanforderung . . . . .	2
3.2 Teachen via YouTube-Link . . . . .	2
3.3 Erkennen mit Mikrofon . . . . .	2
3.4 Album Cover . . . . .	2
3.5 History . . . . .	2
3.6 Lyrics . . . . .	3
3.7 Deployment auf Linux-Server . . . . .	3
<b>4 UML-Diagramme</b>	<b>4</b>
<b>5 Quellen</b>	<b>5</b>
<b>Abbildungsverzeichnis</b>	<b>III</b>
<b>Literaturverzeichnis</b>	<b>IV</b>

# 1 Aufgabenstellung

Als abschließendes Projekt im Fach Softwaredesign soll ein Programm zur Audioerkennung, nach dem Vorbild von „Shazam“ und „Soundcloud“, auf Basis einer Web-Applikation erstellt werden. Das Projekt sollte mit objektorientierter Programmierung in Python umgesetzt und mithilfe von „streamlit“ über eine Web-Oberfläche bedienbar gemacht werden. Die Datenbank stützt sich auf „tinydb“. Was die Funktionalität betrifft, sollen Songs eingelesen und erkannt werden können [1].

## 2 Installationsanleitung

Um Harmonee auf Ihrem Rechner zu installieren, bedarf es folgender drei Schritte:

1. Inhalte des Git-Repository kopieren. Dazu kann dieser Link verwendet werden: "Harmonee"
2. Nun müssen die Inhalte des "requirements.txt" File mithilfe des Befehls "pip install -r requirements.txt" installiert werden.
3. Jetzt kann in der Konsole mithilfe des Befehls "streamlit run main.py" Harmonee gestartet werden.

Es öffnet sich sodann ein Browserfenster, auf welchem Harmonee genutzt werden kann.

## 3 Programmfunktionen / Erweiterungen

Es wurden außer den gestellten Minimalanforderungen, nämlich das Einlernen von Liedern, die Möglichkeit zum Abspielen des geladenen Liedes und das Erkennen von Liedern aus der Datenbank, einige zusätzliche Funktionen implementiert, die im folgenden Abschnitt näher erläutert werden. Ein Schnappschuss der Web-Oberfläche ist in Abbildung 3.1 ersichtlich.

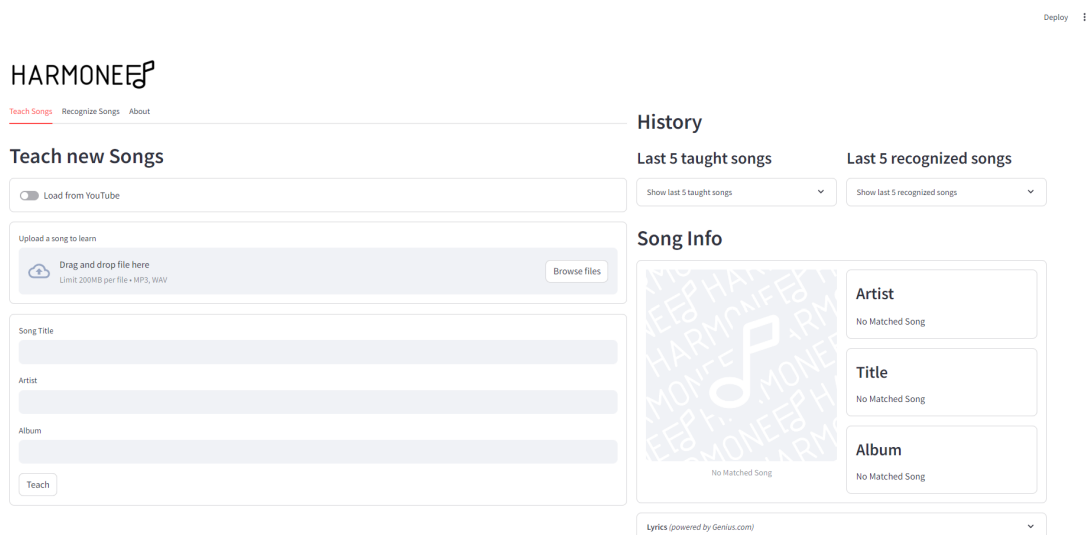


Abbildung 3.1: Teach-Tab, Song-Info vor Erkennen eines Liedes

### 3.1 MINIMALANFORDERUNG

Im Tab „Teach Songs“ können neue Lieder eingelernt werden, die dann in der Datenbank gespeichert werden. Über den File-browser können diese vom jeweiligen Rechner hochgeladen werden und über den erscheinenden Play-button abgespielt werden. Im Tab „Recognize Songs“ können mit der gleichen Vorgehensweise Schnipsel hochgeladen werden, die nach dem Betätigen des Buttons „Recognize“ erkannt werden.

### 3.2 TEACHEN VIA YOUTUBE-LINK

Die erste Zusatzfunktion ist das Anlernen von Liedern über einen Youtube-Link. Dazu muss der Schieber „Load from YouTube“ aktiviert werden. Nun erscheint eine Text-input-Zeile, in die der kopierte Link eingefügt werden kann. Nach dem Drücken des „Load“-Buttons wird das dem YouTube-Link zugehörige mp3-file heruntergeladen und zwischengespeichert. Darunter kann jederzeit die zuletzt geladene Audio-Datei abgespielt werden.

### 3.3 ERKENNEN MIT MIKROFON

Es besteht außerdem die Möglichkeit, Lieder mithilfe des Mikrofons zu erkennen. Dazu muss im Container „Recognize from Microphone“ der Button „Start Recording“ gedrückt werden. Es folgt eine 10-sekündige Aufnahme, die direkt mit der Datenbank abgeglichen wird. Ein etwaiges Ergebnis wird dann angezeigt.

### 3.4 ALBUM COVER

Nach dem erfolgreichen Erkennen eines Liedes wird mithilfe der DuckDuckGo-Search das Internet nach einem Album cover durchsucht. Der Bildersuche übergeben wird der Prompt: „title artist album cover“. Das erste aufscheinende Bild wird auf der rechten Seite unter „Song Info“ neben Name, Interpret und Album angezeigt.

### 3.5 HISTORY

Im Container „History“ werden nach Öffnen des jeweiligen Drop-Down-Menüs die 5 zuletzt gelernten oder erkannten Lieder inklusive Details angezeigt.

The screenshot displays the HARMONEE application interface. At the top, there's a navigation bar with 'Teach Songs', 'Recognize Songs' (active), and 'About'. The main content area is divided into two columns. The left column features the 'Recognize Songs from Snippet' section, which includes a file upload area with a 'Browse files' button, a video player showing a snippet of 'Rick Astley - Never Gonna Give You Up', and a 'Recognize' button. Below this is the 'Recognize from Microphone' section with a 'Start Recording' button. The right column contains the 'History' section with two dropdown menus for 'Last 5 taught songs' and 'Last 5 recognized songs'. Below the history is the 'Song Info' section, which displays the album cover for 'Never Gonna Give You Up' by Rick Astley, along with the artist's name, the title, and the album name. At the bottom of the right column, there's a 'Lyrics' section powered by Genius.com, showing the lyrics for 'Never Gonna Give You Up'.

Abbildung 3.2: Recognize-Tab, Song-Info nach Erkennen eines Liedes und Lyrics

## 3.6 LYRICS

Wurde ein Song erkannt wird außerdem mithilfe der Api von "Genius.com" deren Datenbank nach dem erkannten Lied durchsucht. Werden entsprechende Lyrics gefunden, so werden diese im Drop-Down-Menü „Lyrics“ unterhalb des „Song-Info“-Containers ausgegeben.

## 3.7 DEPLOYMENT AUF LINUX-SERVER

Zu guter Letzt wurde noch versucht, die Applikation auf einer kostenlosen AWS-EC2-Instanz auf Basis von Amazon-Linux 2023 zum Laufen zu bringen. Dazu wurde ein Dockerfile erstellt, welches Anweisungen zum Erstellen eines Docker-Containers enthält und einen Port, in unserem Fall Nummer 96, freigibt. Es wurde außerdem sichergestellt, dass auch die EC2-Instanz Datenaustausch über diesen Port zulässt und die „docker-engine“ auf dem Server installiert ist.

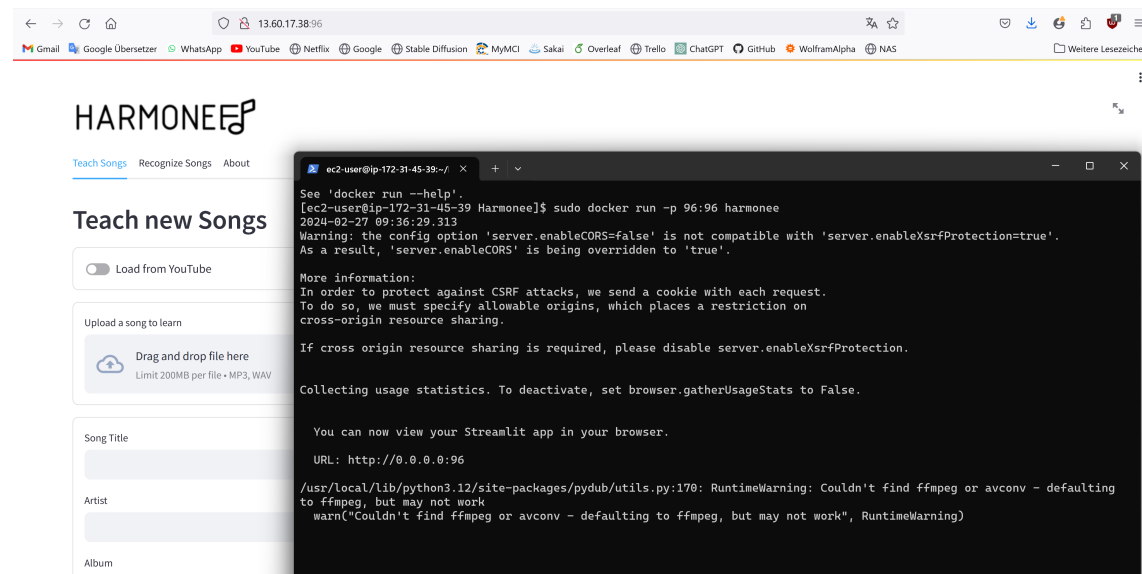


Abbildung 3.3: Erfolgreiches Deployment auf AWS-EC2-Instanz

Es gelang nach dem Überwinden einiger Schwierigkeiten, unter anderem mit der Python-Bibliothek „pyaudio“, ein Docker-Image zu erstellen und daraus einen Container, der die Web-Applikation startet. Aus Transparenzgründen sollte darauf hingewiesen werden, dass es einige Schwierigkeiten mit der „abracadabra“-Bibliothek gab, auf der der Fingerprinting-Code basiert. Es ergaben sich einige Errors, die sich nicht von uns beseitigen ließen, weshalb der Container zurzeit nicht mehr läuft. Auf die Webseite konnte von überall aus zugegriffen werden, wie in Abbildung 3.3 ersichtlich, es konnten jedoch keine Lieder gelernt oder erkannt werden. Das Dockerfile wurde der Vollständigkeit halber trotzdem zum Github-Repository hinzugefügt.

## 4 UML-Diagramme

In Abbildung 4.1 wurden noch 2 UML-Diagramme, die den Programmablauf visualisieren sollen, dargestellt. Das Lernen und Erkennen von Liedern wird als zwei voneinander unabhängige Prozesse dargestellt. Einzige Schnittstelle ist die Datenbank.

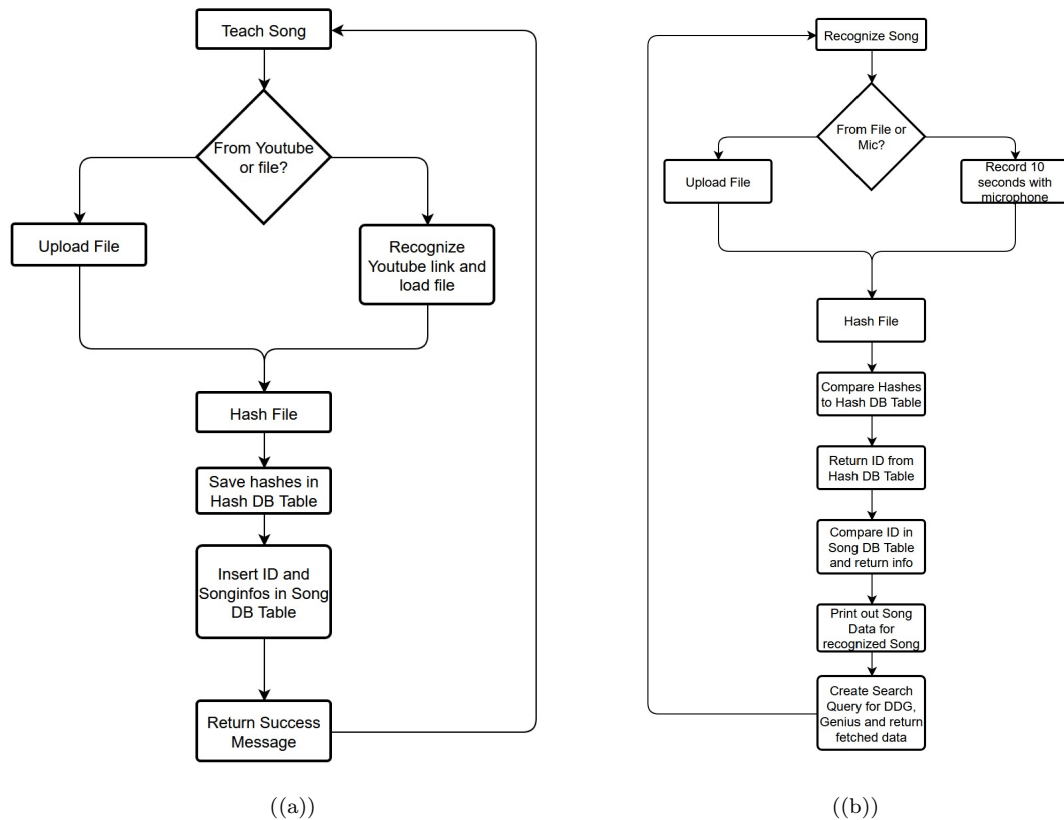


Abbildung 4.1: UML-Diagramme des Programmablaufs

## 5 Quellen

Zur Umsetzung wurden verschiedenste Bibliotheken verwendet, die im folgenden Abschnitt kurz aufgeschlüsselt werden.

- Streamlit (<https://github.com/streamlit/streamlit>) (Zugriff am 27.02.2024): Framework zur schnellen Umsetzung der Web-Umgebung
- Abracadabra (<https://github.com/notexactlyawe/abracadabra>) (Zugriff am 27.02.2024): Grundlage des Fingerprinting-Algorithmus
- TinyDB (<https://github.com/msiemens/tinydb>) (Zugriff am 27.02.2024): Umsetzung der Datenbankstruktur
- PyAudio (<https://github.com/CristiFati/pyaudio>) (Zugriff am 27.02.2024): Bibliothek zur Umsetzung der Audioaufnahme
- PyTube (<https://github.com/pytube/pytube>) (Zugriff am 27.02.2024): Bibliothek zum Herunterladen von mp3-Dateien von YouTube
- LyricsGenius (<https://github.com/johnwmillr/LyricsGenius>) (Zugriff am 27.02.2024): Bibliothek zum einfachen Verbinden mit der Genius.com-API.
- DuckDuckGo-Search ([https://github.com/deedy5/duckduckgo\\_search](https://github.com/deedy5/duckduckgo_search)) (Zugriff am 27.02.2024): Bibliothek zum Suchen von Bildern auf Basis von DuckDuckGo

Das Harmoniee-Logo sowie das dargestellte Default-Album-Cover, welches vor dem Erkennen eines Liedes dargestellt wird, wurde mithilfe von Blender selbst erstellt.

# Abbildungsverzeichnis

3.1	Teach-Tab, Song-Info vor Erkennen eines Liedes . . . . .	1
3.2	Recognize-Tab, Song-Info nach Erkennen eines Liedes und Lyrics . . . . .	2
3.3	Erfolgreiches Deployment auf AWS-EC2-Instanz . . . . .	3
4.1	UML-Diagramme des Programmablaufs . . . . .	4



# Literaturverzeichnis

[1] J.Huber, M.Panny, "Abschlussprojekt," MCI, 2023.