Identification of auditory information by utilising audio fingerprinting

By: Zander Boonzaaier



BSC.IT-HONS 2020

School of Computer Science and Information Systems

Mr. Dirk Snyman

North-West University

Potchefstroom

2020/11/23

# Abstract

This project is meant to act as a dissertation for the degree of Computer Science and Information Technology. This document contains extensive explorations of the possible areas of utilisation in which audio functions as an asset. Within this document, auditory information is considered a resource that can be utilised in a plethora of ways and as such the project had aimed to answer the research question: *How can auditory information be captured, translated into a machine-processable format, interpreted and presented to users in the form of metadata information?* The artefact that had been created in order to answer this research question, consisted of an IoT (Internet of Things) implementation that utilised an artificial intelligence process referred to as audio fingerprinting. The created artefact is able to capture auditory information by utilising an input device and my utilise pre-recorded auditory samples that are present within the user’s machine as input, transform audio samples into a machine-processable format by utilising spectrographs, interoperate these spectrographs by utilising audio fingerprinting and finally present users with metadata concerning said audio samples by utilising a graphical user interface. The artefact had been put through numerous phases, iterations and rigorous tests in order to establish its robustness and quality. The research that had been performed within this document would suggest that such an implementation could function properly within most scenarios.

# Abstrak

Hierdie projek is bedoel om te funksioneer as 'n verhandeling vir die graad Rekenaarwetenskap en Inligtingstegnologie. Hierdie dokumentasie bevat uitgebreide ondersoeke na die moontlike gebruiksgebiede waarin klank as bate kan funksioneer. In hierdie dokument word ouditiewe inligting beskou as 'n hulpbron wat op verskeie maniere gebruik kan word. Sodanig het die projek die doel gehad om die volgende navorsingsvraag te beantwoord: Hoe kan ouditiewe inligting vasgelê word, vertaal word in 'n masjienverwerkbare formaat, geïnterpreteer en uitendelik in die vorm van metadata-inligting aangebied word aan gebruikers? Die artefak wat geskep is om hierdie navorsingsvraag te beantwoord bestaan uit 'n IoT (Internet of Things) -implementasie, wat gebruik maak van 'n kunsmatige intelligensieproses wat na verwys word as “audio fingerprinting”. Die geskepte artefak is in staat om ouditiewe inligting op te neem deur gebruik te maak van 'n invoerapparaat. Dit kan selfs gebruik maak van vooraf opgeneemde ouditiewe inligting wat in die gebruiker se masjien bestaan as invoer, die ouditiewe inligting in 'n masjienverwerkbare formaat te transformeer deur spektrograwe te gebruik, “audio fingerprints” van die spektograwe te maak en uiteindelik metadata rakende die ouditiewe inligting aan gebruikers bied deur gebruik te maak van 'n grafiese gebruikerskoppelvlak. Die artefak is deur talle fases, herhalings en streng toetse gesit om die robuustheid en kwaliteit daarvan vas te stel. Die navorsing wat binne hierdie dokument uitgevoer is, dui daarop dat so 'n implementasie binne die meeste scenario's behoorlik kan funksioneer.

Contents

[Abstract i](#_Toc57059010)

[Abstrak ii](#_Toc57059011)

[1 Chapter - Introduction 1](#_Toc57059012)

[1.1 Project description 1](#_Toc57059013)

[1.2 Problem description and background 1](#_Toc57059014)

[1.3 Aims and objectives of the project 3](#_Toc57059015)

[1.4 Procedures and methods that will be used 4](#_Toc57059016)

[1.5 Approach to project management and project plan 10](#_Toc57059017)

[1.6 Description of development platform, resources, and environments that will be used 12](#_Toc57059018)

[1.7 Ethical and legal implications and dealing with these 14](#_Toc57059019)

[1.8 Provisional chapter division 15](#_Toc57059020)

[2 Chapter – Literature review 16](#_Toc57059021)

[2.1 Introduction to literature review 16](#_Toc57059022)

[2.2 Key theories, concepts and ideas 16](#_Toc57059023)

[2.3 Content-based Audio Identification Framework explanation 18](#_Toc57059024)

[2.4 Implementations of audio fingerprints 20](#_Toc57059025)

[2.5 Summary of Literature review 27](#_Toc57059026)

[3 Chapter – Development of artefact 28](#_Toc57059027)

[3.1 Description of artefact 28](#_Toc57059028)

[3.2 The life cycle followed and its different phases 38](#_Toc57059029)

[3.3 Description of the development of the artefact 43](#_Toc57059030)

[3.4 Conclusion of the development 49](#_Toc57059031)

[4 Results 50](#_Toc57059032)

[4.1 The data set used to perform the tests 50](#_Toc57059033)

[4.2 The minimum duration of an audio sample needed to effectively identify a track 53](#_Toc57059034)

[4.3 Identification accuracy at different levels of background noise 54](#_Toc57059035)

[5 Chapter – Reflection 57](#_Toc57059036)

[5.1 Lessons learned throughout the project life cycle. 57](#_Toc57059037)

[5.2 Evaluation of decisions made within the project 57](#_Toc57059038)

[5.3 Analysis of the strong and weak points of the product and the process that had been followed. 59](#_Toc57059039)

[5.4 What could have been done differently? 59](#_Toc57059040)

[5.5 Had the objectives set in this project been achieved? 60](#_Toc57059041)

[5.6 How successful had the project been managed in meeting target dates? 60](#_Toc57059042)

[5.7 Critical evaluation of the process followed 60](#_Toc57059043)

[6 Bibliography 61](#_Toc57059044)

[7 Appendices 64](#_Toc57059045)

[7.1 Research proposal according to the prescribed template 64](#_Toc57059046)

List of Tables

Table 1 Applications of audio fingerprinting 21

Table 2 Denotation of controls used within the main form 36

Table 3 Detailed descriptions of each implemented phase within the software development lifecycle. 44

Table 4 The 30 most streamed songs on Spotify 50

List of Figures

Figure 1 Spectrogram of speech recorded in a non-noisy environment with peaks identified (Chirp, 2018) 6

Figure 2 Spectrogram of speech recorded in a noisy environment with peaks identified (Chirp, 2018) 6

Figure 3 Constellation map of auditory information spectrogram 7

Figure 4 How audio fingerprinting works. (mufin.com, 2020) 8

Figure 5 Project Plan Gantt Chart 10

Figure 6 ACRCloud Branding Requirement Logo 14

Figure 7 Content-based Audio Identification Framework (Cano et al., 2002) 18

Figure 8 Spectrogram of speech recorded in a non-noisy environment (Chirp, 2018) 24

Figure 9 Spectrogram of speech recorded in a noisy environment (Chirp, 2018) 25

Figure 10 Spectrogram of speech recorded in a non-noisy environment with peaks identified (Chirp, 2018) 26

Figure 11 Spectrogram of speech recorded in a noisy environment with peaks identified (Chirp, 2018) 26

Figure 12 Constellation map of auditory information spectrogram 27

Figure 13 Example Of Metadata 30

Figure 14 Example Of Refined Metadata 31

Figure 15 Initial artefact form 32

Figure 16 Record Audio From Device form evolution 33

Figure 17 Select File From Computer form evolution 34

Figure 18 Unrecognised or un-utilisable file form evolution 35

Figure 19 Identified audio piece 36

Figure 20 Waterfall Model Life Cycle (Balaji & Murugaiyan, 2012) 40

Figure 21 Agile Model Life Cycle (Balaji & Murugaiyan, 2012) 42

Figure 22 The Agile methodology versus the Waterfall methodology 43

Figure 23 Accuracy rating in terms of sample duration 53

Figure 24 Accuracy in environment with no background noise 54

Figure 25 Accuracy in environments with a small amount of background noise 55

Figure 26 Accuracy in environments with a medium amount of background noise 55

Figure 27 Accuracy in environments with a large amount of background noise 56

# Chapter - Introduction

## Project description

Statistics over the years suggest a sharp increase in the number of different bodies of media that exist today. Every day the influence, scope, variety and sheer quantity of different media entities increase exponentially (Mohsin, 2020). One of the more prominent aspects of these different types of media is auditory information in the form of music and speech. There are various characteristics of auditory information that influences different aspects of the daily lives of people (and animals) around the world. These aspects include but are not limited to communication, development, emotion, interpretation, performance and behaviour (Scherer & Zentner, 2001). Auditory information thus represents a recourse that could be utilised to achieve a number of different goals. These utilisations include, but are not limited to, medicinal applications, general performance enhancement and behavioural manipulation. In accordance with this, this project aims to answer the research question:

*How can auditory information be* *captured, translated into a machine-processable format, interpreted and presented to users in the form of metadata information?*

To answer this question, an IoT-based system will be developed to attempt to capture, processes and interoperate auditory information. This system will be utilising an artificial intelligence process referred to as audio fingerprinting to map various instances of auditory information. Audio fingerprints are defined as follows: “An audio fingerprint is a unique and compact digest derived from perceptually relevant aspects of a recording. Fingerprinting technologies allow the monitoring of audio content without the need of metadata or watermark embedding.”(Cano *et al.*, 2005a)

## Problem description and background

One of the more prominent forms of auditory information in today’s environment is music. Music is a concept that has been part of society from the earliest evolutionary milestone and to this day not a single culture throughout the entire course of human history has not incorporated music in some way (David M. Greenberg, 2016). In accordance with this, the project will mainly focus on the interpretation of auditory information that has been classified as music.

Music is defined as an art of sound in time that expresses ideas and emotions in significant forms through the elements of melody, harmony, rhythm, and intensity.

The value of music in today’s civilisation depends on the areas in which we incorporate it (North *et al.*, 2004). In the society we live in today, it is almost impossible to avoid music, as it is incompatible into almost all aspects of our everyday life. (Green, 2006)

### Music in the entertainment industry.

Music is an ever-evolving concept in the world of entertainment. It has been implemented into theatre, film, television, and video games with further implementations being made daily (Kania, 2007). Music presents an intimate medium for affecting multiple aspects of an entity’s perception by inducing a certain tone, setting a certain mode or even assisting in delivering a certain message (Weakly, 2017). Significant efforts have been made to induct and improve music in the entertainment industry (Kania, 2007).

### Music in the medical industry.

The field of medicine is also no stranger to the implementation of music. Multiple experiments and successful studies proclaim the usefulness of auditory stimulus in medical practices (Aldridge, 1993). Some examples of areas in which music therapy has been successfully implemented include:

* Manipulating heart rate and blood pressure;
* Relieving anxiety and stress in cancer patients;
* Aiding in pain management and hospice care;
* Rehabilitating people with neurological problems; and
* Diagnosing cognitive and motor impairment in the elderly.

As time goes on different music implementations continue to be incorporated in different ways across multiple medicinal fields (Aldridge, 1993).

### Music in terms of behavioural manipulation.

Evidence suggests that music can have positive effects concerning behaviour and performance (Moreno, 2009). Multiple studies have been done to determine the effects of different auditory information on the behavioural patterns on different aspects of our behaviour. ”This approach involves the creation, selection, and improvisation of music idiosyncratic to the specific necessities of dealing with shaping the behaviour of each individual patient or client.” (Madsen *et al.*, 1968).

Music or musically classified auditory information could very easily be classified as a resource or tool that could be utilised in many different fields of application, as discussed above.

When studying and/or implementing different bodies of auditory information, utilising readily available metadata concerning them, would prove invaluable. In accordance with this, this project aims to answer the proposed research question.

The expected outcome of this research is that more information concerning the different components of auditory information will be available more readably available, thus making it more utilisable for future projects that might utilise auditory information.

## Aims and objectives of the project

The aim of this project is to attempt to answer the proposed research question by developing an IoT (Internet of Things) based system and/or software with the goal of effectively capturing, processing and interoperating auditory information.

### Objectives

In this section, four categories of objectives have been identified namely research, hardware, software and code-based objectives.

#### Estimation of possible research and text-based objectives

During the lifespan of the project, a series of documentation will be produced to aid in researching the proposed research question. These documentations may include but are not limited to, literature reviews, educational posters, potential user reviews, and possible experimentation reports.

#### Acquisition and general preparation of hardware

The first step in creating a functional system is to source most of the hardware components, in order to ensure low-level functionality.

1. Acquire a programmable computer that offers enough processing power to perform the necessary calculations and algorithms. A Raspberry-pi 3B+ would be a readably available, cost-effective solution as opposed to other microcomputers such as an Arduino implementation.
2. Source an efficient method of displaying information to the user. Any form of screen, built-in or otherwise would suffice.
3. Create a method of powering the mobile device. The solution must be reusable and cost-efficient. A small power-bank or rechargeable battery could be sufficient.
4. Acquire a method of receiving audio input from the environment for the purpose of analysing it. The recording device must be sufficient in terms of recording quality and cost-effectiveness.

#### Potential software objectives

There are a plethora of useful recourses available that could be utilised in the answering of the project question. It is thus, imperative that the project only utilise the necessary services and components that are sufficient in making it as functional as possible.

1. Source and/or create a base platform that will function as an audio fingerprint generation module.
2. Source and/or create an appropriate audio fingerprint data storage module that provides easy and accurate access to required metadata.
3. Source and/or create an audio fingerprint processing module.
4. Source and/or create a correlated database of metadata concerning the audio fingerprints provided.

#### Potential code-based objectives

1. Create a module that can record auditory information.
2. Annalise the acquired audio file and transcribe it into a spectrogram.
3. Construct an algorithm or utilise a pre-existing algorithm to analyse spectrograph data efficiently and create an audio fingerprint.
4. Implement a communication method between a database consisting of audio fingerprint data and metadata and the main user interface.
5. Retrieve the metadata provided by the database (if any).
6. Display the appropriate information to the end-users in a user-friendly manner utilising a user interface.

## Procedures and methods that will be used

Within this section of the document, potential methodologies, paradigms and methods that may be utilised during the lifecycle of the project are identified and briefly explained.

### Research paradigm that will be utilised.

The project mainly focuses on the statistical and quantitative data of audio fingerprints. All literature and reference material that will be utilised in this project will consist of impersonal, formal and rule-based media and text. During this project, a deductive approach will we followed in order to more thoroughly describe the relationships between variables, measure concepts quantitatively and generalize research findings to a certain extent. Because of this, the project will utilise an ontological-positivistic research paradigm in answering the proposed research question (Crowther & Lancaster, 2012).

Positivism is a philosophical system that states that any and all concepts can be verified by utilising experimentation, observation and mathematical/logical proof and that most knowledge stems from human experience (Collins, 2018). The positivistic philosophy also states that the only knowledge that is utilisable in a study, is gained through observation and measurements (Feigl, 2019). Ontology is classified as a branch of philosophy that is concerned with providing a definitive and exhaustive explanation, classification and information concerning different entities (Smith, 2012).

The project will utilise a quantitative-correlational research method. This method is a non-experimental method that focuses on the observation and documentation of the relationships between different variables using statistical analyses (Johnson, 2001).

### Audio fingerprint creation

The proposed system will make use of an artificial intelligence process referred to as audio fingerprinting. Audio fingerprinting will be utilised for the recording, extraction, storage, and interpretation of auditory information. “Audio fingerprinting is best known for its ability to link unlabelled audio to corresponding meta-data (e.g. artist and song name), regardless of the audio format.”(Cano *et al.*, 2005b)

Audio fingerprints are considered to be digital summarisations of auditory information. One can think of audio fingerprints in much the same way as human fingerprints. Just as human fingerprints are unique to a single person, audio fingerprints are unique to an instance of auditory information (Haitsma & Kalker, 2002). Most audio fingerprinting concepts and software utilise a time-frequency representation of auditory information to create fingerprints. These representations are called spectrographs.

Audio information that has been recorded with the minimum amount of background noise and sounds produce the most sufficient spectrograms for identification of frequency content, volume and peaks within the auditory information recording. When utilising a spectrogram that contains background noise, concepts such as frequency content and volume is considerably more difficult to identify and utilise. However, the different peaks of the auditory information recording are still moderately decipherable. This is what will be used to construct the audio fingerprints. (Wang, 2003)

In most implementations of audio fingerprinting, a logarithmic frequency scale is utilised to plot the different peaks of an auditory recording, as seen in Figure 1 and Figure 2. This is why the grouping of the peaks (denoted with the black dots) are more grouped together at the higher frequencies (Wang, 2003).

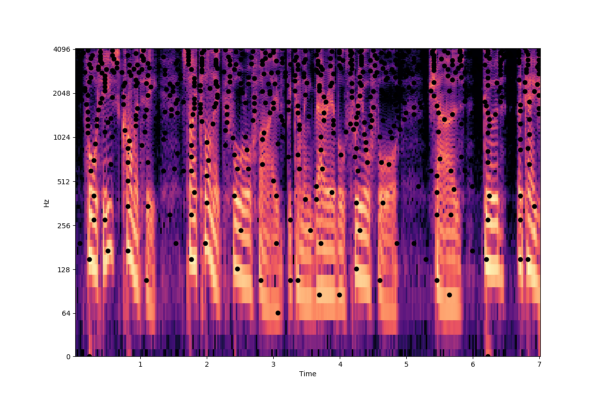


Figure 1 Spectrogram of speech recorded in a non-noisy environment with peaks identified (Chirp, 2018)

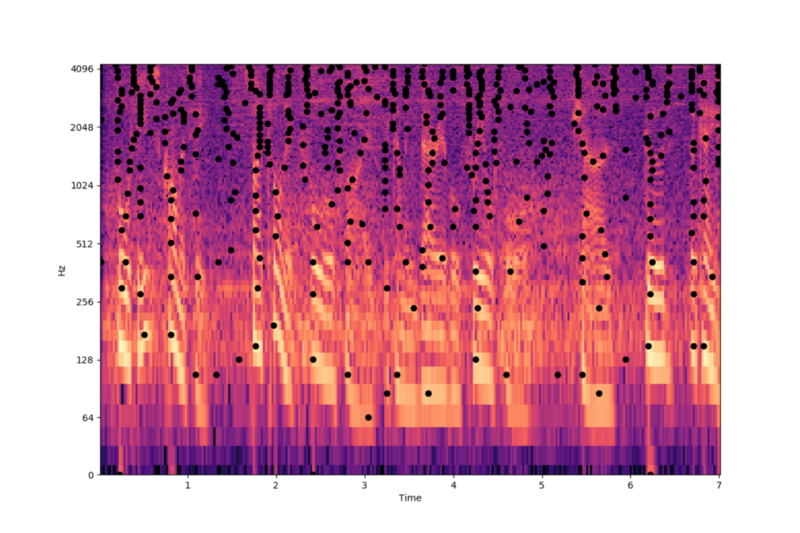


Figure 2 Spectrogram of speech recorded in a noisy environment with peaks identified (Chirp, 2018)

Once all of the peaks have been identified, we have the basis of an audio fingerprint. The peaks will then be mapped onto a constellation map for hashing purposes. Pairs of time-frequency points or peaks are combinatorically associated into anchor points and target zones to create different hashes. Multiple hashes of a spectrogram function as a fingerprint (Wang, 2003).

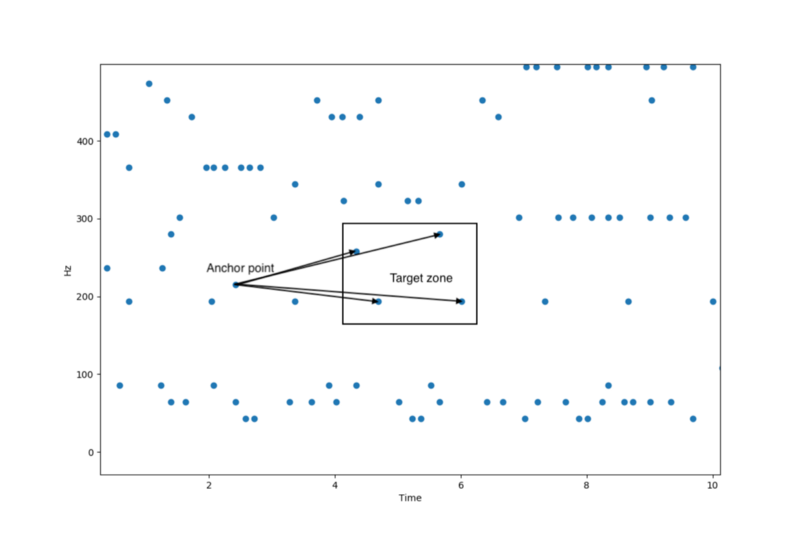


Figure 3 Constellation map of auditory information spectrogram

These hashes and any metadata regarding them is then stored in a database. Once an appropriate database had been established, we can begin comparing new fingerprints with ones already present in the database and attempt to identify different auditory information recordings (Wang, 2003).

### Audio fingerprint storage

Figure 4 depicts the basic concept of audio fingerprint extraction, storage and querying. For the project to function properly, a large database of high-quality audio fingerprints is required. (mufin.com, 2020).

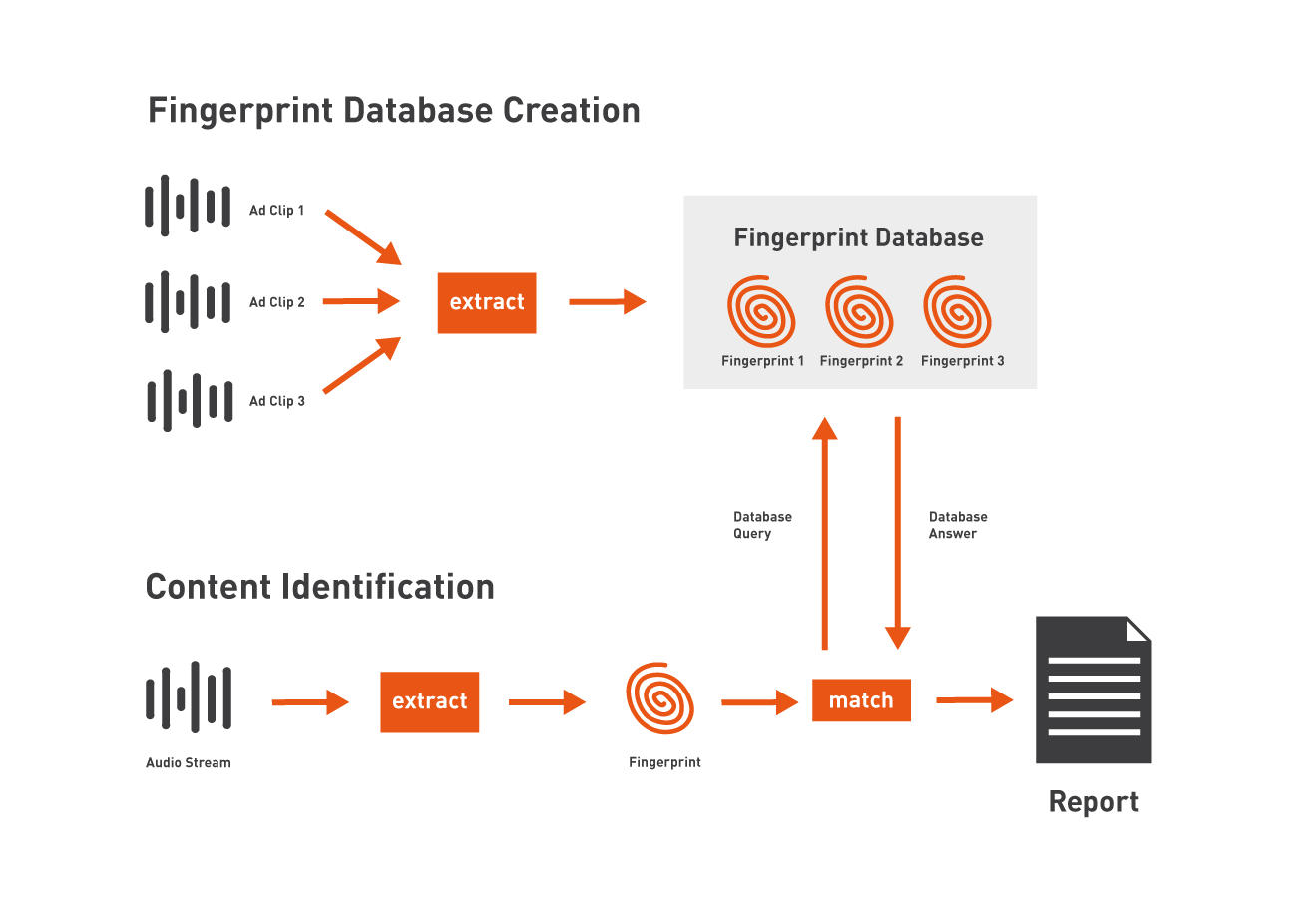


Figure 4 How audio fingerprinting works. (mufin.com, 2020)

To achieve this, two relevant options present themselves:

* Create a new audio fingerprint database; and
* Utilise a pre-existing or third-party audio fingerprint database, and program an integration module.

#### Creating a new audio fingerprint database

If a new database were to be created, a high-quality audio file would first need to be acquired. Afterwards, the file would need to be converted to an audio fingerprint format by following the abovementioned method.

That fingerprint would then need to be correlated with the original audio’s applicable metadata (such as author, album, artwork, URLs and a plethora of other information) and stored within the new database. This database would need to be updated with new audio fingerprints regularly, as to match the development of new and relevant auditory information.

When an attempt at identifying new auditory information is made, said auditory information is transcribed into an audio fingerprint format utilising the abovementioned method and compared to the audio fingerprints that are documented within the database. If a matching fingerprint is found, the relevant metadata is retrieved and delivered. If, however, a match is not found, no metadata will be retrieved or delivered.

#### Utilising an already pre-existing/third-party audio fingerprint database

Utilising a pre-existing database, there is a multitude of pre-existing audio fingerprint information and metadata within the database, thus decreasing database creation time significantly. This database would also need to be updated with new audio fingerprints regularly, as to match the development of new and relevant auditory information.

Utilising an appropriate third-party database, there is a multitude of pre-existing audio fingerprint information and metadata within the database and depending on the operational standards of the third-party entity, the database will automatically be updated with new audio fingerprints. This further decreases database creation and upkeep time significantly.

When an attempt at identifying new auditory information is made, said auditory information is transcribed into an audio fingerprint format utilising the abovementioned method and compared to the audio fingerprints that are documented within the database. If a matching fingerprint is found, the relevant metadata is retrieved and delivered. If, however, a match is not found, no metadata will be retrieved or delivered.

An artefact that is capable of creating, processing, storing and analysing different auditory information by utilising audio fingerprint technology will be created. Taking into consideration of all the above-mentioned options, the project is going to utilise both third-party pre-existing databases, as well as new localised databases for the integration of custom audio fingerprints and offline utilisation.

## Approach to project management and project plan

### Project Plan Gantt Chart

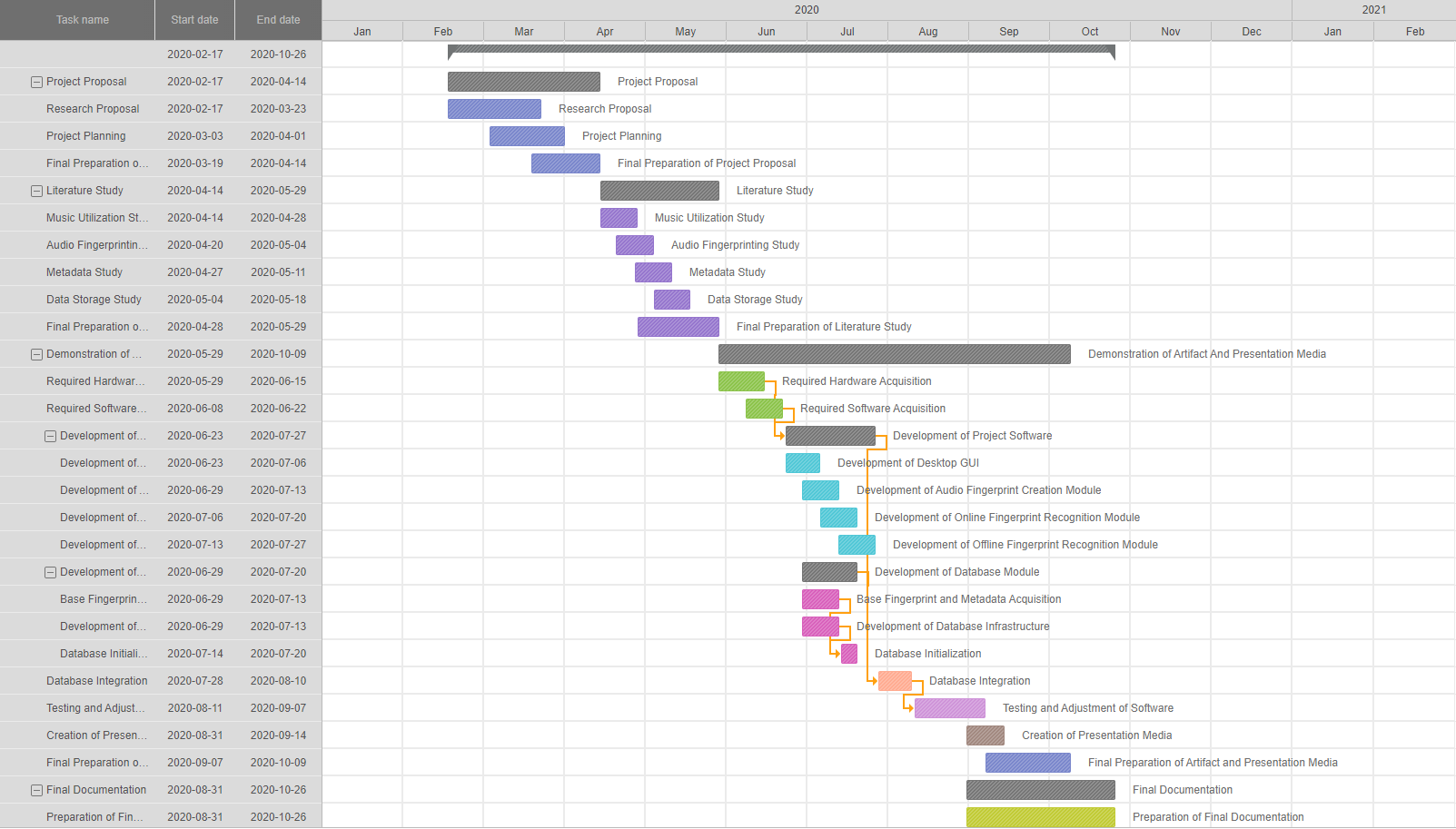


Figure 5 Project Plan Gantt Chart

### Project Scope

The project aims at delivering an artefact(s) in the form of software and/or hardware capable of completing the following actions:

1. Receive auditory information as input.
2. Transcribe auditory information into a machine-interpretable format, such as an audio fingerprint.
3. Store information concerning audio fingerprints and their applicable metadata in an offline and/or online database structure.
4. Utilise stored audio fingerprint data to attempt to identify auditory information.
5. Present relevant conclusions (and possibly corresponding metadata) concerning a piece of auditory information to a user.

### Project Limitations

Most of the limitations involving the project are concerned with the quality and content of the prospective auditory information that it is presented with. This includes:

* Auditory information that contains too much background noise. – The project will not be able to accurately identify a piece of auditory information if it contains too much background noise
* Auditory information that contains more than one clear piece of identifiable fingerprint data. – The project will not be able to accurately identify a piece of auditory information if it is interfered/mixed with other auditory information.
* Inadequate auditory information recording quality. – If the recorded audio file is of too poor quality, the project might not be able to accurately identify it.
* Insufficient audio fingerprint database data. – If the database of different audio fingerprints does not contain the relevant data concerning a piece of auditory information, it cannot present the user with data concerning it.
* Corrupted auditory information. – If the recorded audio file is corrupted in any way, the project might not be able to accurately identify it.

### Project Risks

Due to the volatile nature of copyrighted material and fair use law, one of the possible risks concerning the project is the utilisation of metadata and musical content, however, this is a slight problem that is easily circumventable by obtaining all of the necessary. It should also be noted that utilising audio fingerprints in any way does not cause any copyright infringements and as such, this is a small concern.

Another risk that the project may face it whether or not the Raspberry-pi 3B+ model would have enough processing power to create utilisable fingerprints and process the different data presented to it. One possible answer for this risk is to base as much data and do as much processing in the cloud instead of on the device itself.

It should be noted that Zander Boonzaaier (Student number: 28749995) will be the sole official contributing researcher of this project and as such, all responsibility befalls unto them.

## Description of development platform, resources, and environments that will be used

In this section discussion is presented in terms of the different development platforms, resources and applicable environments of implementation concerning the development and integration of the project are discussed.

### The development platform that may be utilised.

The base programming languages that may be implemented within this project are C# and Python. The project will be created using the Visual Studio platform. This option was chosen because it boasts a modicum of support and possible reference material. It also provides a platform for the development of Graphical User Interfaces (GUIs) and boasts powerful third-party support and implementation (such as GitHub).

### Resources that will be utilised

In this subsection, the different hardware and software requirements that may be utilised in the creation of the proposed system.

#### Hardware

The following will be used to create the artefact.

* A Raspberry-pi 3B+ or another main processing machine;
* A screen to be utilised in displaying information to, and interacting with, the user;
* A microphone to be used in audio recording; and
* A small speaker or another audio output device.

#### Software

Visual Studio will be used to develop most of the project’s software requirements. This is utilised because it is a powerful platform that offers a modicum of support and developmental options. Different SDK’s may also be implemented because they present a method of communication between projects and third-party entities.

### Applicable environments of implementation.

This project will be developed to be utilisable within any reasonable environment. The project will function more efficiently when used in an environment where it has an adequate connection to the internet, however, when this is not available the project may make use of a locally stored database or store the requested fingerprint until an adequate connection can be made.

### Audio fingerprinting services that are considered.

There are a plethora of audio fingerprinting services to choose from, however, 4 of which stand out. These are:

* Acoustid;
* Gracenote MusicID ;
* ACRCloud; and
* Rovi Media Recognition.

Each of these implementations boasts a wide variety of different benefits and requirements as described below.

Acoustid grants access to a large, crowdsourced database of audio fingerprints. Because Acoustid is crowd-sourced, incorrect identifications are more prominent. Acoustid provides a free service for non-commercial use and is heavily linked to online music encyclopaedia MusicBrainz and utilises MusicBrainz identifiers (MBIDs) in order to identify recordings. Acoustid, however, does not function well when utilising audio files containing moderate background noise.

Gracenote MusicID grants access to an extremely large and detailed database of audio fingerprints but is highly and non-transparently priced. However, Gracenote MusicID does offer free plans for non-commercial projects.

ACRCloud grants access to a large database of audio fingerprints (40 million+) as well as a large number of different implementable SDKs written in different languages, that can be used to access the service in multiple ways.

This service also presents multiple options for pricing which includes a free service for non-commercial use and a paid service for commercial use, guaranteeing service levels. ACRCloud also provides metadata links from multiple third-party identifiers, such as YouTube, Apple Music, and Spotify.

Rovi Media Recognition grants access to an extremely large and detailed database of audio fingerprints but is highly-priced and does not offer free plans for non-commercial projects. See the work of (Gravell, 2016) for a summary on the aforementioned audio fingerprinting services.

Taking into consideration the abovementioned options, the project will utilise the services of ACRCloud because it offers the widest variety of resources and services, even when not utilising a commercial licence.

## Ethical and legal implications and dealing with these

In this section, the required information concerning the utilisation of the third-party software ACRCloud is presented, along with the company’s mandatory ACRCloud service description.

### ACRCloud Branding Requirement

During the development of this project, services of ACRCloud will be utilised to provide audio fingerprinting code segments as well as a large database of pre-registered audio fingerprints. The following is a service description as provided by the ACRCloud company.

“ACRCloud utilises patented Automatic Content Recognition (ACR) technology to enable the generation of a unique real-time fingerprint to identify in a matter of seconds, the content being played via an audio or video source which is typically a first screen, in order to trigger an action. With proprietary audio identification technology, ACRCloud’s is able to identify millions of hours of content in both a manageable and efficient manner that is highly relevant for advertisers, broadcasters, video streaming providers, music services, consumer electronics manufacturers and app developers. ACR’s fingerprint library includes 40 million audio tracks, which is one of the biggest in the world.” (ACRCloud, 2019 )



Figure 6 ACRCloud Branding Requirement Logo

## Provisional chapter division

The final product of this research would be in the form of a dissertation as well as an accompanying artefact in the form of an IoT software and hardware implementation. The aforementioned dissertation will be in the following format:

### Chapter 1 – Introduction and problem statement

This chapter will contain an introduction to the study in terms of background information concerning the project, a relevant research question as well as information concerning the research methodology that will be implemented to conduct the study.

### Chapter 2 – Literature review

Within this chapter, the relevant information that was collected during the literature study of the project will be presented.

### Chapter 3 – Development of the artefact

Within this chapter, the different strategies, methodologies, platforms, artefact forms and other relevant information concerning the development of the IoT artefact will be presented.

### Chapter 4 –Results

Within this chapter, a revision of the collected data and general performance of the project within different environments and under various circumstances will be presented and discussed.

### Chapter 5 –Conclusion and reflection

The last chapter of the document will conclude the study by providing a summary of the project and its findings. Also contained within this chapter would be a revision of the decisions and developments that were made during the project development.

# Chapter – Literature review

## Introduction to literature review

People have been utilising the concept of fingerprints for thousands of years (Haitsma & Kalker, 2002). The first recorded example of fingerprints being utilised as a form of identification had been approximately 2000 years ago when different Chinese emperors used thumbprints in order to sign important documents and decrees.

In 1893 Sir Francis Galton, a well-known English Victorian era statistician, proved that no two human fingerprints were alike. In 1901, Scotland Yard had incorporated a system for identifying and matching the fingerprints of different people, as designed by Sir Edward Henry. This system still forms the basis of all human fingerprint identification techniques being used today (Haitsma & Kalker, 2002).

In recent years, the concept of identifying different multimedia trough utilisation of a fingerprint-based system has seen an abundance of study and implementation. Within this section of the document, these different studies and applicable findings are discussed.

## Key theories, concepts and ideas

### The goal of fingerprinting

According to Haitsma & Kalker (2002), the primary objective of multimedia fingerprinting is to function as an efficient mechanism to be used in order to test for equality between two multimedia objects. They further state that this is to be done, not by comparing the objects themselves, but by comparing their associated fingerprints.

By comparing only the applicable fingerprints instead of the entire objects, the following advantages are achieved:

* Memory and storage requirements are significantly reduced as fingerprints are considerably smaller than the actual files that they are based on;
* Slight signal deviations would have been removed from the fingerprints’ information, thus making comparisons more efficient; and
* Because the searched dataset is smaller, searching operations are more efficient.

### Parameters of an affective audio fingerprinting system

Haitsma & Kalker (2002) propose a set of five parameters to witch an audio fingerprinting system must adhere. These parameters are as follows:

* Fingerprint size: How much storage is needed in order to store a fingerprint and it’s applicable metadata? In order to promote faster database searching, fingerprints are typically stored in the RAM (Random Access Memory) of a device. Because of this, fingerprint size (usually expressed in bits per second or bits per song) determines the type of memory resources required to adequately host a fingerprint database server (Haitsma & Kalker, 2002).
* Granularity: How many seconds of audio is needed in order to effectively identify an audio clip? Granularity is a parameter that may be dependent on the fingerprinting system itself. For instance, in some applications, the entire piece of audio may be used for identification, and in others, only a short excerpt of audio is required.
* Reliability: How often are different media object identified incorrectly. The rate at which incorrect identifications occur is will be referred to as the false-positive rate.
* Robustness: Is it still possible for different media objects to be accurately identified after they have experienced signal degradation? In order to achieve high robustness, the created fingerprint would ideally be based on perceptual features that are mostly unaffected by different types of signal degradation. Ideally, a severely degraded audio sample must still produce a similar fingerprint when compared to an audio sample of good quality. The concept of false negatives should be utilised in order to express robustness. A false negative occurs when the fingerprints of perceptually similar audio clips are too different to lead to a positive match.
* Search speed and scalability: How much time would it take the system in order to find a fingerprint in a fingerprint database? Search speed and scalability are crucial criteria that are used to determine the effectiveness of an audio fingerprinting system.

### Fundamental processes of audio fingerprinting applications

There exist multiple audio fingerprinting methodologies that each utilises different processes and techniques to produce and process audio fingerprints (Patil, 2019).

Good examples hereof would be the popular audio fingerprint implementations: “Shazam” and “MusicBrainz Picard”. Both implementations may be utilised to identify audio samples, however, they utilise different processes, techniques and implementations to do so.

Cano *et al* (2002) state that whilst different audio fingerprinting implementations utilises different processes and techniques, they all consist of two main fundamental processes, namely: the fingerprint extraction algorithm and the fingerprint matching.

These two fundamental processes can be further expanded to a universal audio fingerprint identification framework. The framework is graphically represented in Figure 7:

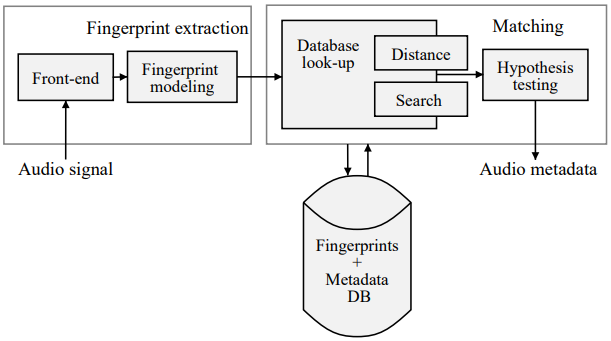


Figure 7 Content-based Audio Identification Framework (Cano et al., 2002)

## Content-based Audio Identification Framework explanation

As seen in Figure 7, there are two major parts to audio fingerprinting, namely fingerprint extraction and fingerprint matching. The purpose of the fingerprint extraction part is to compute different sets of characteristic features from the provided audio signal. These sets of characteristic features are called fingerprints (Liu, 2017).

After fingerprint extraction, the different fingerprints of the provided audio signal are utilised by a matching algorithm in order to locate the most probable match (if there are any) through searching a large database containing pre-acquired audio fingerprints. Within this part of the document, the Content-based Audio Identification Framework is further expanded upon.

### Front-end

The front-end block of an audio fingerprinting system is utilised in order to compute characteristic features sets from the provided audio signal. After the sets have been computed, they are sent to the fingerprinting modelling block for further processing.

The front-end generally consists of 5 steps (Liu, 2017), namely:

1. Pre-processing: Within this step, the provided audio signal is digitized and quantized. If it is necessary, the provided audio may also be converted into a mono signal. This may be done by averaging it’s input channels. If the audio signal’s sampling rate differs from the target rate, it is resampled.
2. Framing: Within this step, the audio signal is divided into different frames consisting of equal lengths. This is done by utilising a window function (Liu, 2017).
3. Transformation: Within this step, the sets of frames created within the framing step are transformed into new sets of features, in order to reduce the level of redundancy. The most popular transformation is the transformation from time domain to frequency domain (Liu, 2017).
4. Feature Extraction: Within this step, the final acoustic features are extracted from the time-frequency representation produced within the previous step. This is done in order to reduce the dimensionality and increase the robustness to distortions. There are several schemes that are ideal for this purpose, such as Spectral Flatness, Band representative vectors and Measure Mel-Frequency Cepstrum Coefficients (MFCC) (Liu, 2017).
5. Post-processing: Within this step, higher-order time derivatives are determined if it is necessary to capture the different temporal variations concerning the provided audio signal.

### Fingerprint Modelling

The fingerprint modelling block of an audio fingerprinting system is used in order to compute the final fingerprint. This is done by utilising the sequences of feature vectors that had been extracted by the front-end block.

### Distance and Search

After the fingerprints have been extracted from the initial audio signal, a database containing different audio fingerprints and metadata must be searched in order to find a matching audio sample.

The similarity between two fingerprints is a measurement of how much they are alike. This similarity variable is described in terms of distance (Liu, 2017). This means that a small distance represents a high degree of similarity, whilst a large distance represents a low degree of similarity.

Some popular distance measurements include:

* Accumulated approximation error;
* Euclidean distance;
* Exponential Pseudo Norm; and
* Manhattan distance

However, the ideal technique to be used to measure distance relies heavily on the design of the audio fingerprinting system (Liu, 2017).

Because a typical fingerprint database consists of millions of fingerprints, it would not be sufficient to compare each individual fingerprint to the query fingerprint. Because of this most audio fingerprinting implementations utilise an index data structure to decrease the number of distance calculations that need to be performed in order to obtain a possible match (Liu, 2017).

### Hypothesis Testing

The final stage of the Content-based Audio Identification Framework is used to determine whether or not a matching fingerprint has been located within the database. This is done by comparing each fingerprint’s distance score in relation to that of the query fingerprint. If there exist a fingerprint of fingerprints that are above a certain distance score threshold, they are returned to the user as possible matches (Liu, 2017). Depending on the design of the fingerprinting application and database, the user may ideally also be presented with accompanying metadata.

The performance of an audio fingerprint implementation is calculated as a fraction of the amount of correctly matched queries out of all of the queries used to test the system (Liu, 2017).

## Implementations of audio fingerprints

As audio fingerprinting is a powerful tool that may be utilised to identify a multitude of different auditory objects, it has multiple applications and possible areas of implementation.

The following table is used to list and describe the different applications of audio fingerprinting.

Table 1 Applications of audio fingerprinting

|  |  |
| --- | --- |
| **Application** | **Description** |
| Broadcast monitoring | Broadcast monitoring is possibly the most widespread implementation of audio fingerprinting. Broadcast monitoring refers to the automatic identification generation of content used in radio, television or web-based broadcasts. This application is mainly present in royalty collection, program verification, advertisement verification and social engineering campaigns (Haitsma & Kalker, 2002).  A newer implementation of this application is the utilisation of automated audio identification systems in discotheques (Sonnleitner *et al.*, 2016). The goal of this implementation is to monitor and control revenue distribution for DJ performances played within these establishments. Audio fingerprinting is ideal for this implementation, as it can still accurately detect different signal modifications introduced by the different performers. Some examples of these modifications include pitch shifting, tempo modifications, cross-fading and beat-matching. |
| Connected audio | Connected audio refers to more general-purpose applications for identifying auditory information that is connected to additional and supporting information. For example, utilising a mobile application to identify a specific song and the year in which it was released. This application of audio fingerprinting is supported by a number of different companies, such as “Shazam” and “SoundHound”.  This application is presented with a number of challenges regarding the accurate identification of auditory information due to audio quality and signal degradation (Haitsma & Kalker, 2002). For example, radio stations may apply some processing in order to broadcast song, or the quality of the recording device may be less than adequate.  This may greatly affect the chances of accurately identifying a sample of auditory information and as such, an audio fingerprint implementation must take these possibilities into account.  Another example of this application is the implementation of automatic content recognition performed by modern-day cars, that would allow the user to be presented with additional information of the song and its creators, or even to purchase the album of a song that is currently being played across the radio. |
| Filtering technology for file sharing | Filtering refers to the active prevention of the unlawful acquirement of auditory intellectual property (Haitsma & Kalker, 2002). This application may be implemented in many environments, from subscription-based audio sharing services to legal file-sharing authorisation. One of the benefits of this implementation is the possible utilisation of a consistent metadata and organisation system. By utilising this application, you also have a system that may identify any file that you had downloaded based on its auditory information.  It should be noted that whilst some civilian users might view this application as negative or restrictive because it prevents them from acquiring media files without paying, it is much more ethical to implement this system in order to combat intellectual property fraud. |
| Automatic Music Library Organization | Today, many personal computer users possess music libraries that consist of hundreds or even thousands of music and media files. These different media files are usually acquired through various sources, such as downloading them from an online source or ripping them from a media storage device.  These media files are generally stored on the user’s hard drive and are usually not well organised. In most cases, the metadata of the files are either inconsistent, incomplete and sometimes even incorrect (Haitsma & Kalker, 2002).  By utilising audio fingerprinting, these problems may be easily resolved by identifying the different media files and updating them with the application metadata.  Examples of these types of applications are “MusicBrainz Picard” and “ID3Man”. |
| Duplicate detection | As stated above, numerous personal computer users possess music libraries that consist of hundreds or even thousands of music and media files. Because of the sheer size of the different libraries, the chances of duplicated media files are very large (Burges *et al.*, 2005).  Audio fingerprinting may help in combatting this, by identifying duplicate files based on auditory data. Audio fingerprinting is especially useful in this scenario, as it allows for the detection of similar files that might have undergone some form of degradation. |
| Audio thumbnail generation | With the large number of media that is being created on a daily basis, it is difficult to explore all released content in a predetermined amount of time. To answer this, audio fingerprinting is utilised to determine a short section of audio that may act as a representative summarization of the content concerning a piece of media.  “Audio thumbnails can help improve audio browsing, either in simple plain list interfaces or in more complex multidimensional ones” (Burges *et al.*, 2005). |

### Audio fingerprint creation

Audio fingerprinting forms the main cornerstone of this project and as such it is imperative to have a relatively good understanding concerning how audio fingerprints are created, stored and utilised. Audio fingerprinting is mostly utilised in scenarios where unidentified auditory information is linked to corresponding meta-data (e.g. artist and song name), regardless of the audio format (Cano *et al.*, 2005b).

Audio fingerprints are considered to be digital summarisations of auditory information. One can think of audio fingerprints in much the same way as human fingerprints. Just as human fingerprints are unique to a single person, audio fingerprints are unique to an instance of auditory information (Haitsma & Kalker, 2002). Most audio fingerprinting concepts and software utilise a time-frequency representation of auditory information to create fingerprints. These representations are called spectrograms.

A spectrogram is a graphical representation of a spectrum of different audio frequencies that are present within a media file and how these different frequencies vary with time (Siva Sankaran, 2017). It should be noted that the utilisation of spectrograms and frequency analysis is ideal for audio fingerprinting, as research shows that these techniques are robust, scalable, and retain the different frequency characteristics that prominently define an auditory signal (Siva Sankaran, 2017).

Figure 8 shows the spectrogram of a recorded conversation. This conversation was recorded in a perfect environment, with little to no background noise, and as such presents the ideal subject for audio fingerprinting. However, this is an unrealistic standard as most recordings would consist of moderate background noise. Figure 9 shows a spectrogram of the same conversation but in a noisy environment. (Chirp, 2018)

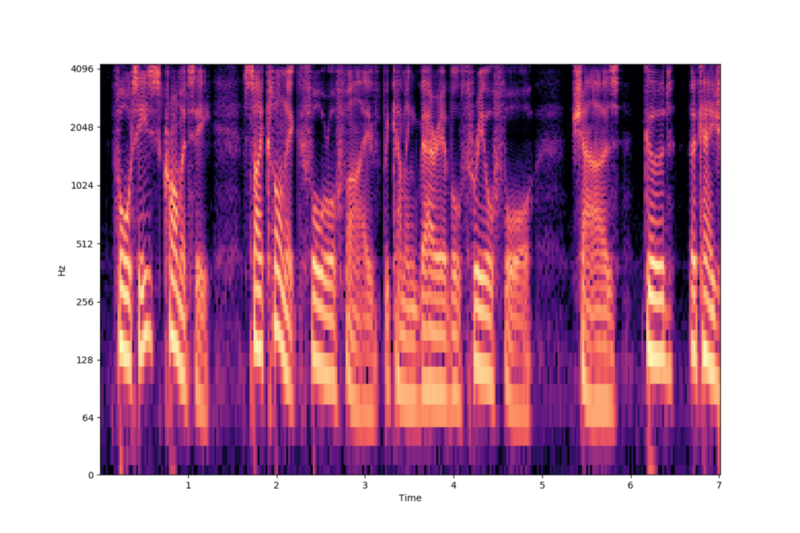


Figure 8 Spectrogram of speech recorded in a non-noisy environment (Chirp, 2018)

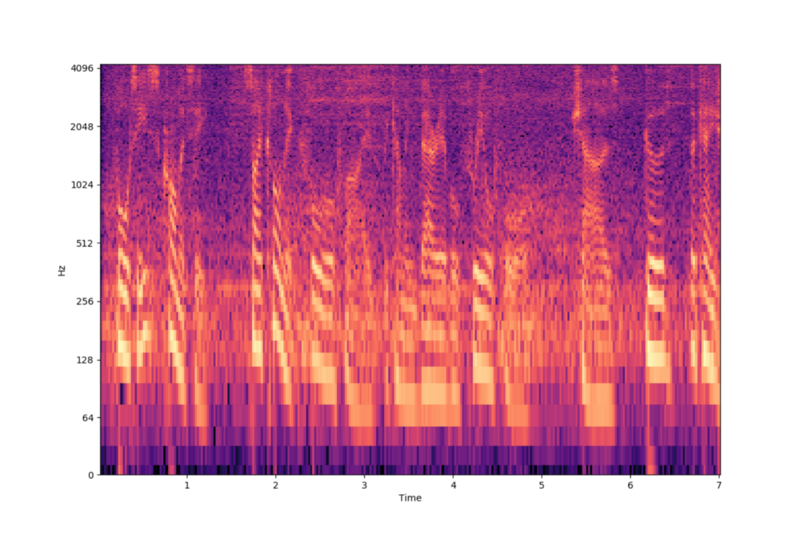


Figure 9 Spectrogram of speech recorded in a noisy environment (Chirp, 2018)

Ideal spectrograms, such as the case in Figure 8, with the minimum amount of background noise, are extremely efficient at identifying frequency content, volume and peaks within the auditory information recording. When there is a modicum of background noise present within a recording, the spectrogram is more likely to look like Figure 9.

When utilising a spectrogram that contains background noise, concepts such as frequency content and volume is considerably more difficult to identify and utilise. However, the different peaks of the auditory information recording are still moderately decipherable. This is what will be used to construct the audio fingerprints. (Wang, 2003)

Each spectrogram can be divided into two-dimensional windows which will be analysed. The objective here is to identify salient peaks in each region of a spectrogram which is not a result of background noise.

In Figures 10 and 11, the different peaks of the different auditory information recordings are depicted with black dots. In most implementations of audio fingerprinting, a logarithmic frequency scale is utilised to plot the different peaks. This is why the grouping of the peaks (denoted with the black dots) are more grouped together at the higher frequencies. (Wang, 2003)

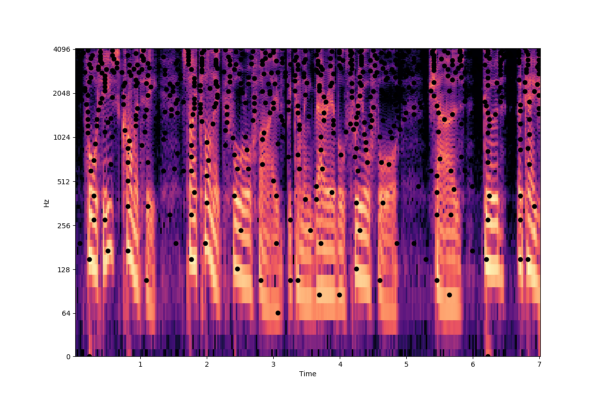


Figure 10 Spectrogram of speech recorded in a non-noisy environment with peaks identified (Chirp, 2018)

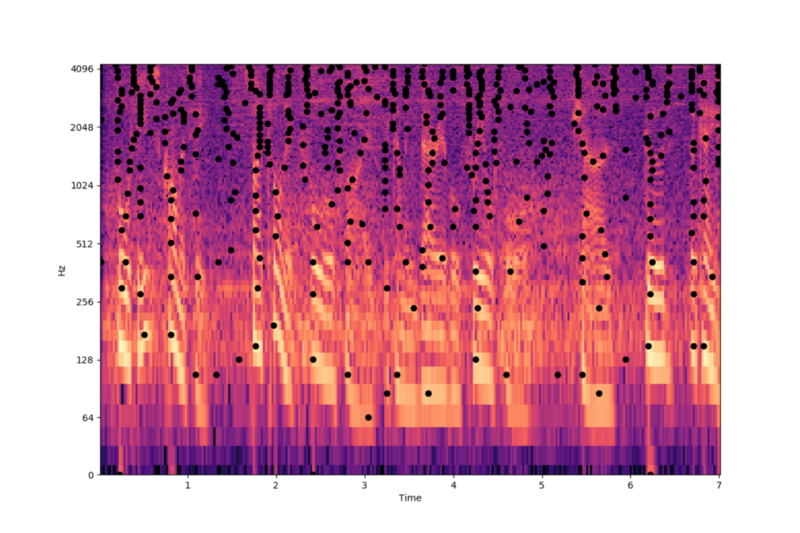


Figure 11 Spectrogram of speech recorded in a noisy environment with peaks identified (Chirp, 2018)

Once all of the peaks have been identified, the basis of an audio fingerprint has been developed. The peaks will then be mapped onto a “Constellation map” for “hashing” purposes. Fingerprint hashes are created by combinatorically associating different pairs of time-frequency points on the constellation map. Anchor points are then chosen and associated with different target zones. Each anchor point is sequentially paired with points within its target zone, which yields two frequency components as well as the time difference between the different points (Wang, 2003).

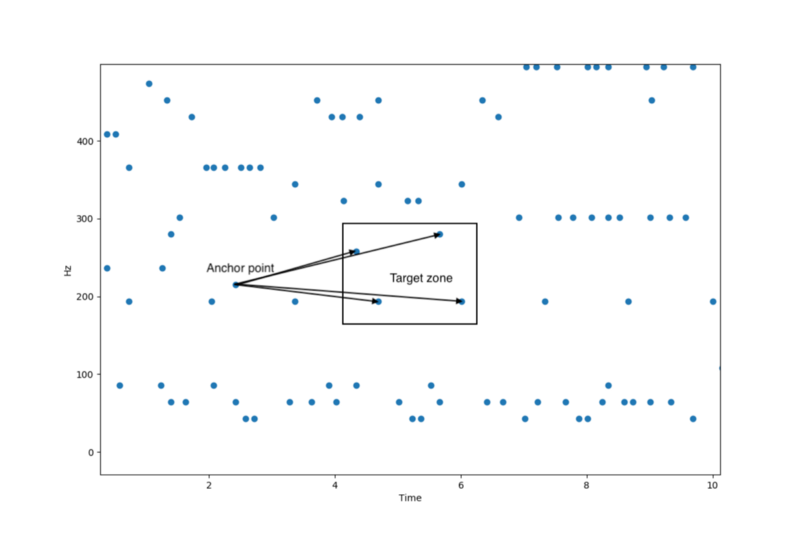


Figure 12 Constellation map of auditory information spectrogram

The hashes, along with any applicable metadata, is then stored within a database. Once an appropriate database had been established, the new fingerprints can be compared to the ones already present in the database, in an attempt to identify the different auditory information recordings.

## Summary of Literature review

With the information presented above, it is clear that audio fingerprinting is a complex concept that may be utilised in a plethora of different applications ranging from broadcast monitoring to file share filtering. With more and more aspects of our daily lives becoming digitised, implementations such as audio fingerprinting present a valuable asset, not only for corporations concerned with intellectual property laws but for civilian users as well. Because of this, audio fingerprinting has become a distinctive point of interest for researchers and developers.

# Chapter – Development of artefact

Audio fingerprinting is a powerful tool that can be utilised to capture auditory information, translate into a machine-processable format and present interpreted information to users in the form of metadata information (Cano *et al.*, 2005b). Audio fingerprints are defined as follows:

“*An audio fingerprint is a unique and compact digest derived from perceptually relevant aspects of a recording. Fingerprinting technologies allow the monitoring of audio content without the need of metadata or watermark embedding*”(Cano *et al.*, 2005a).

## Description of artefact

In this section of the document, a broad overview of the machinations of the project is provided, followed by a description of the different processes and components that form the complete artefact.

The goal in developing this artefact was to gain a more advanced understanding of the mechanics and possible implementations of audio fingerprinting in regard to automatic content recognition. Within its current state, the artefact takes the form of a desktop application designed to identify different audio samples as provided by its user.

### A broad overview of the artefact’s capabilities

Currently, the artefact may be utilised to achieve two different goals, namely, to attempt to identify any pre-recorded pieces of musical audio that the user would have on their device and to record any audio from a user’s microphone and attempt to identify said audio.

This is achieved by implementing the following steps:

1. Receive an audio sample as input;
2. Translate this audio sample or signal into a machine-interpretable format, namely a spectrograph;
3. Utilise this spectrograph to create a corresponding audio fingerprint;
4. Utilise this audio fingerprint and compare it to a database of other audio fingerprints, in an attempt to identify the audio sample and derive metadata from it;
5. Retrieve the corresponding metadata of the audio sample or signal had been successfully identified; and
6. Refine the retrieved metadata and present it to the user within an appealing graphical user interface (GUI).

As the abovementioned steps function as an extremely broad definition of the functions of the artefact, these steps will be further explained below.

**Receive an audio sample as input**

When first initialising the program the user is presented with two options, namely to select a pre-existing audio piece from their device to be identified, or to take a new recording of their environment, by utilising their microphone, in an attempt to identify any piece of musical auditory data that may be present near the user.

It should be noted that the artefact had been designed to be compatible with most types of audio formats. Formats that are supported by the artefact include M4A, FLAC, MP3, MP4, WAV, WMA and AAC.

**Transcribing audio samples or signals into a utilisable format, namely a spectrograph**

Once audio had been received as input, it is mapped to a time-frequency representation of auditory information. These representations are referred to as spectrographs and had been created by utilising a collection of software packages as provided by ACRCloud, an automatic content recognition organisation, and BASS.net, a .Net wrapper for the BASS audio library and all of its applicable add-ons.

**Utilise this spectrograph to create a corresponding audio fingerprint**

After a spectrograph had been created, said spectrograph is utilised in creating an audio fingerprint. This is done by utilising an auditory fingerprinting implementation as provided by ACRCloud.

**Utilise this audio fingerprint and compare it to a database of other audio fingerprints, in an attempt to identify auditory information**

Once the audio fingerprint had been created and stored, it is then compared to a plethora of other audio fingerprints that are stored on a cloud server. Each fingerprint contained within the database will be compared with the created fingerprint, with each one being assigned an audio “score”. This score is a representation of how well a fingerprint resembles the recorded audio. If a fingerprint is awarded a score of 100, then the comparisons are stopped, as the comparison algorithm had potentially found a perfect match.

Any fingerprints with a similarity of less than 60% are not considered to be eligible for identification, in order to minimise server resource requirements and promote artefact processing speed and accuracy.

**Retrieve the corresponding metadata if the audio sample or signal had been successfully identified**

Once the comparison algorithm had concluded, the fingerprint with the highest score (or a score of 100) will be chosen as the fingerprinting result. A second database which contains all of the metadata of the fingerprints contained within the first database will then be queried and the applicable metadata JavaScript object notation (JSON) file retrieved and sent back to the locally hosted instance of the artefact.

In the event that a satisfactory fingerprint comparison could not be found that the artefact would not be presented with any metadata and would simply inform the user that the audio could not be identified.

**Refine the retrieved metadata and present it to the user within an appealing graphical user interface (GUI)**

The artefact would receive the metadata from the server in the form of a JavaScript object notation text. The following is an example of a possible identification as provided by the server:

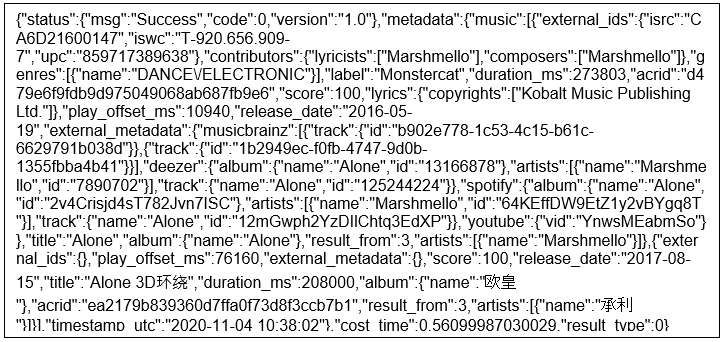


Figure 13 Example Of Metadata

As seen in Figure 13, the JavaScript object notation text received from the server contains a plethora of utilisable information but is presented in such a way that users that are not familiar with JavaScript object notation text may find it difficult to derive useful information from it.

In order to answer this, the artefact would then perform an initial interpretation and formulation of the provided JavaScript object notation text, resulting in a more readable and mouldable set of data. An example of a processed text may be seen in Figure 14:

{

"external\_ids": {

"isrc": "CA6D21600147",

"iswc": "T-920.656.909-7",

"upc": "859717389638"

},

"contributors": {

"lyricists": [

"Marshmello"

],

"composers": [

"Marshmello"

]

},

"genres": [

{

"name": "DANCE/ELECTRONIC"

}

],

Figure 14 Example Of Refined Metadata

This resulting text would then be further refined to provide the user with applicable information regarding the identified track. The information provided to the user is as follows:

* ID Score: This is the score represents the accuracy of an audio track’s identification. A score of 100 would mean that the track has been identified with complete certainty.
* Title: This is the title of the identified audio track, as it had been granted by its original composer.
* Artist(s): This is a list of all artist that had contributed to the production of an audio track.
* Genre(s): This is a subjective classification of the genre into which the audio track is divided.
* Original track's duration: This is a denotation of the original track’s duration, as denoted in terms of hours, minutes, seconds and milliseconds.
* Release Date: This is a denotation of the date when the original track had been released for public consumption and utilisation.

The abovementioned information is then presented to the user by means of a colourful and easily understandable GUI (Graphical User Interface).

### Artefact main form

The artefact mainly consists of a single “window” or “form” that evolves and changes based on the actions of the user. Figure 15 is an example of the main form that the user would initially be presented with:

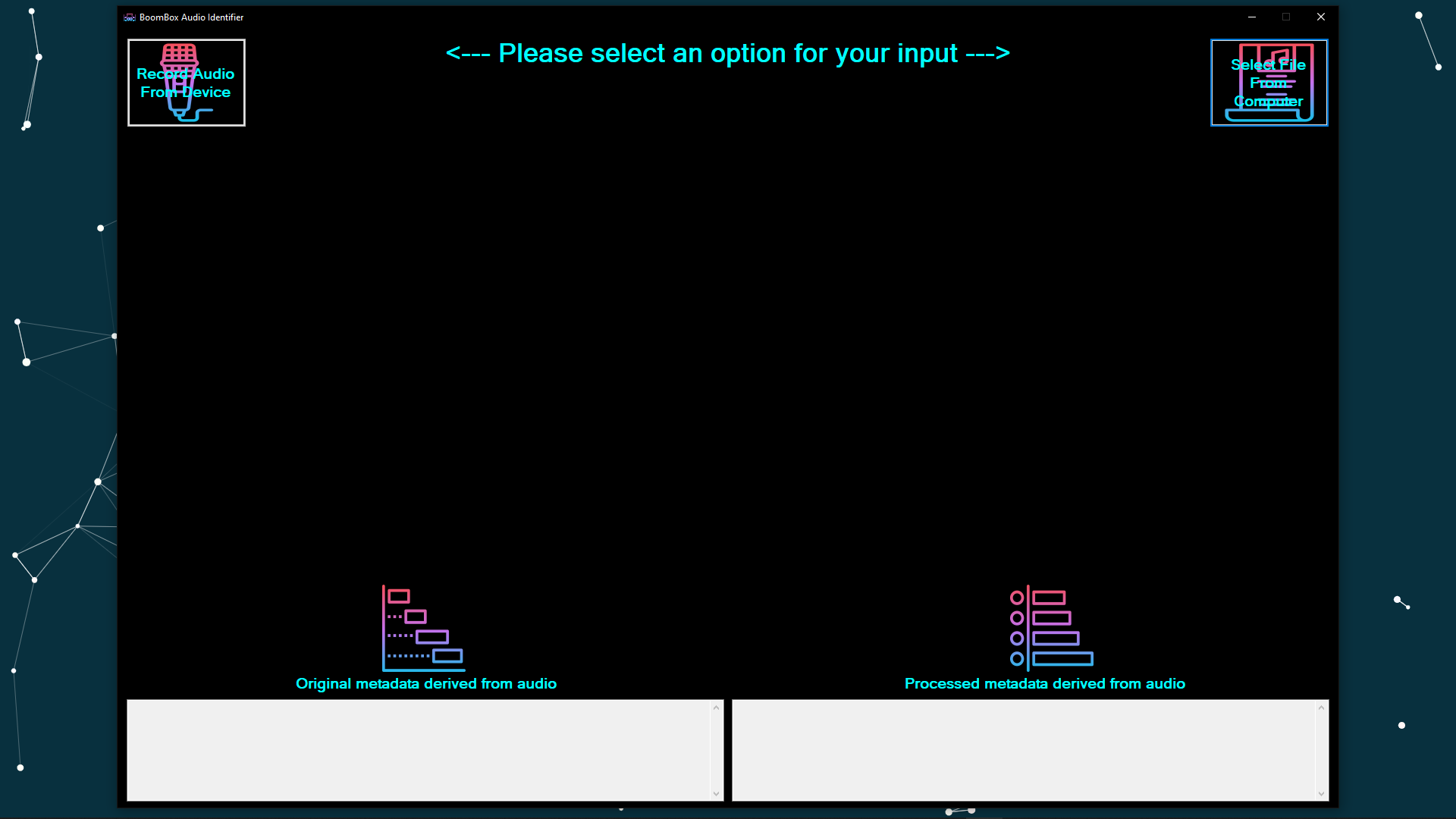


Figure 15 Initial artefact form

The user is initially presented with two options:

* A “Record Audio from Device” button – which allows the user to utilise their microphone to attempt to input any audio data.
* A “Select File from Computer” button – which allows the user to select any audio file on their device to function as auditory

### Recording audio from the device

The “Record Audio From Device” button would initialise a “listener” algorithm which records any input from the user’s primary microphone or audio input device by utilising an audio stream. This audio stream is then buffered in order to record the input stream.

After the “Record Audio from Device” button had been pressed, The button would become hidden and a third button, the “Stop Recording and Send Request” button, would be made visible and active. An example of the changed form may be seen in Figure 16:

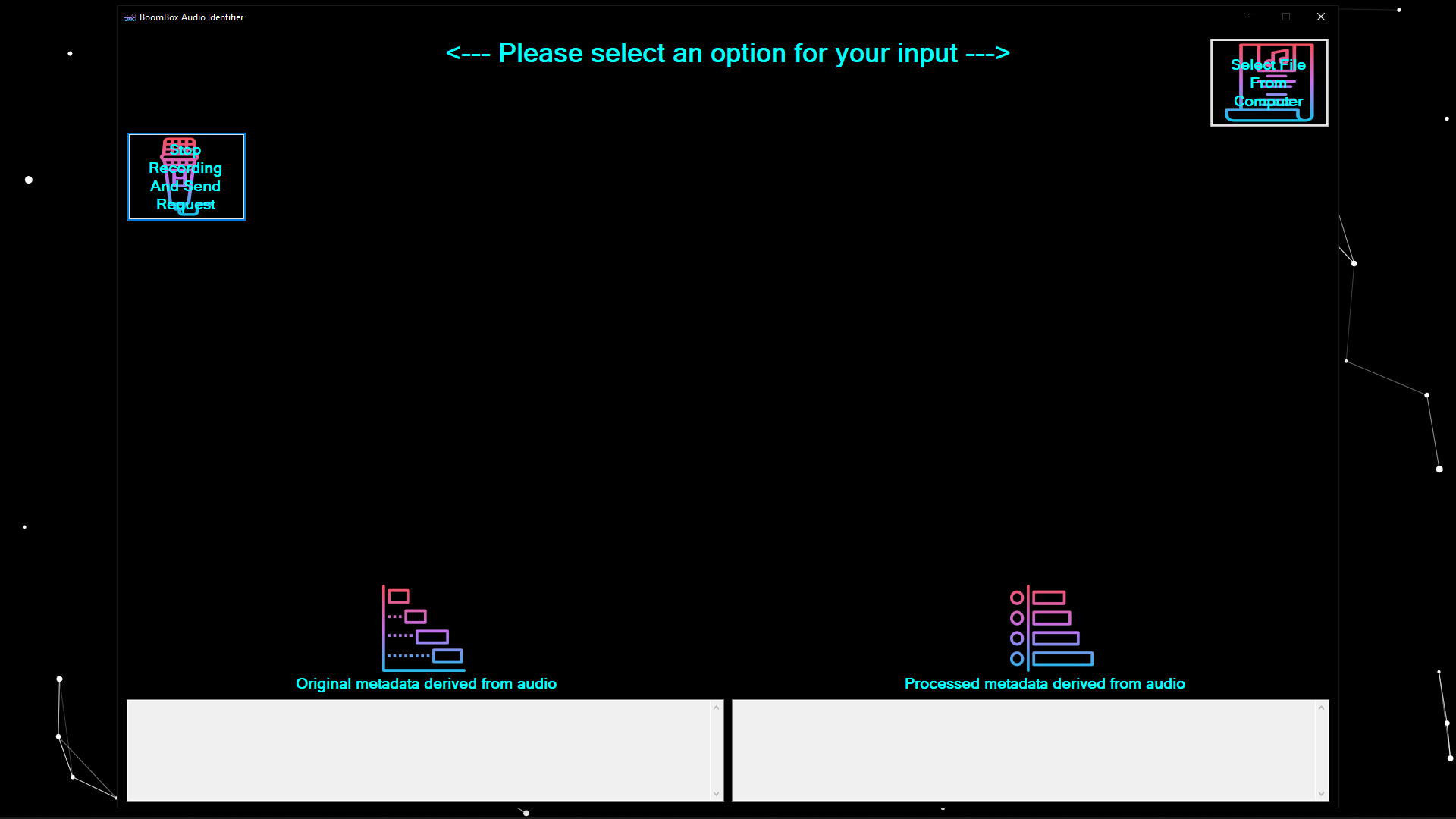


Figure 16 Record Audio From Device form evolution

When the “Stop Recording and Send Request” button is pressed the aforementioned input stream will be closed, and the recorded audio transformed into a locally stored WAV file. This WAV file will then undergo the fingerprinting process in which it is first translated to a spectrograph. This spectrograph is then utilised in creating an audio fingerprint. Said fingerprint is then used in an attempt to identify the user-provided audio.

### Selecting a pre-recorded file

The “Select File from Computer” button would open a file dialogue window that would allow the user to select a file which they would like to attempt to identify. An example of the updated form may be seen in Figure 17:

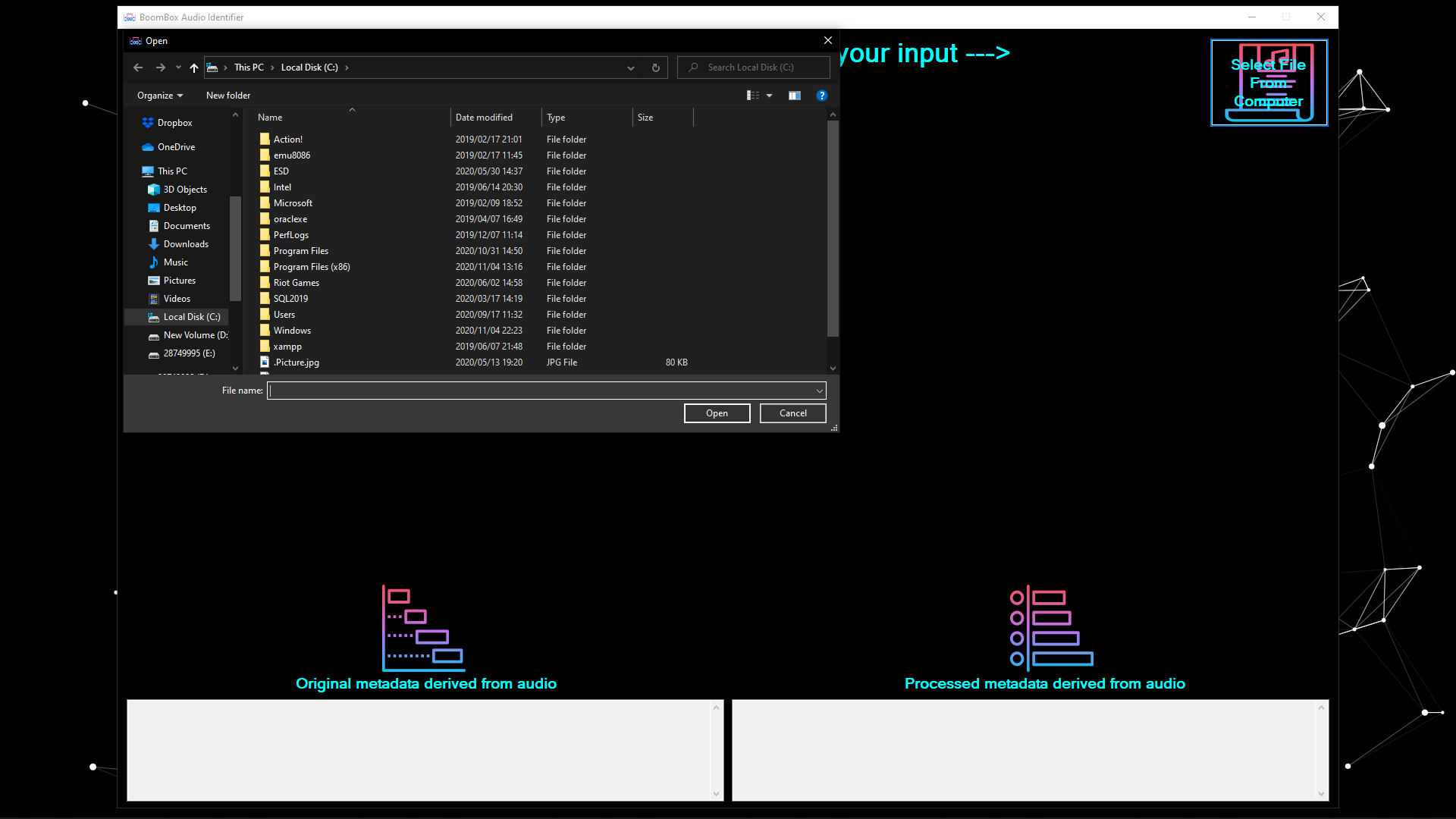


Figure 17 Select File From Computer form evolution

After the user had selected which file they would like to attempt to identify, the file is first tested to see if it can be converted to a spectrograph. File formats that are supported by the artefact include M4A, FLAC, MP3, MP4, WAV, WMA and AAC.

In the event that a file cannot be converted into a spectrograph, the artefact would abandon the fingerprinting attempt, and the user would be informed that no results had been found, that no metadata had been available and that the audio sample could not be identified.

### Unrecognised audio or unsupported file

In the event that the artefact is not able to create an audio fingerprint of the selected file or provided audio, or if the created fingerprint cannot be matched with a fingerprint within the database, the user will be informed by means of a message that reads: “Audio Not Identified”. An example of the evolved form may be seen in Figure 18:

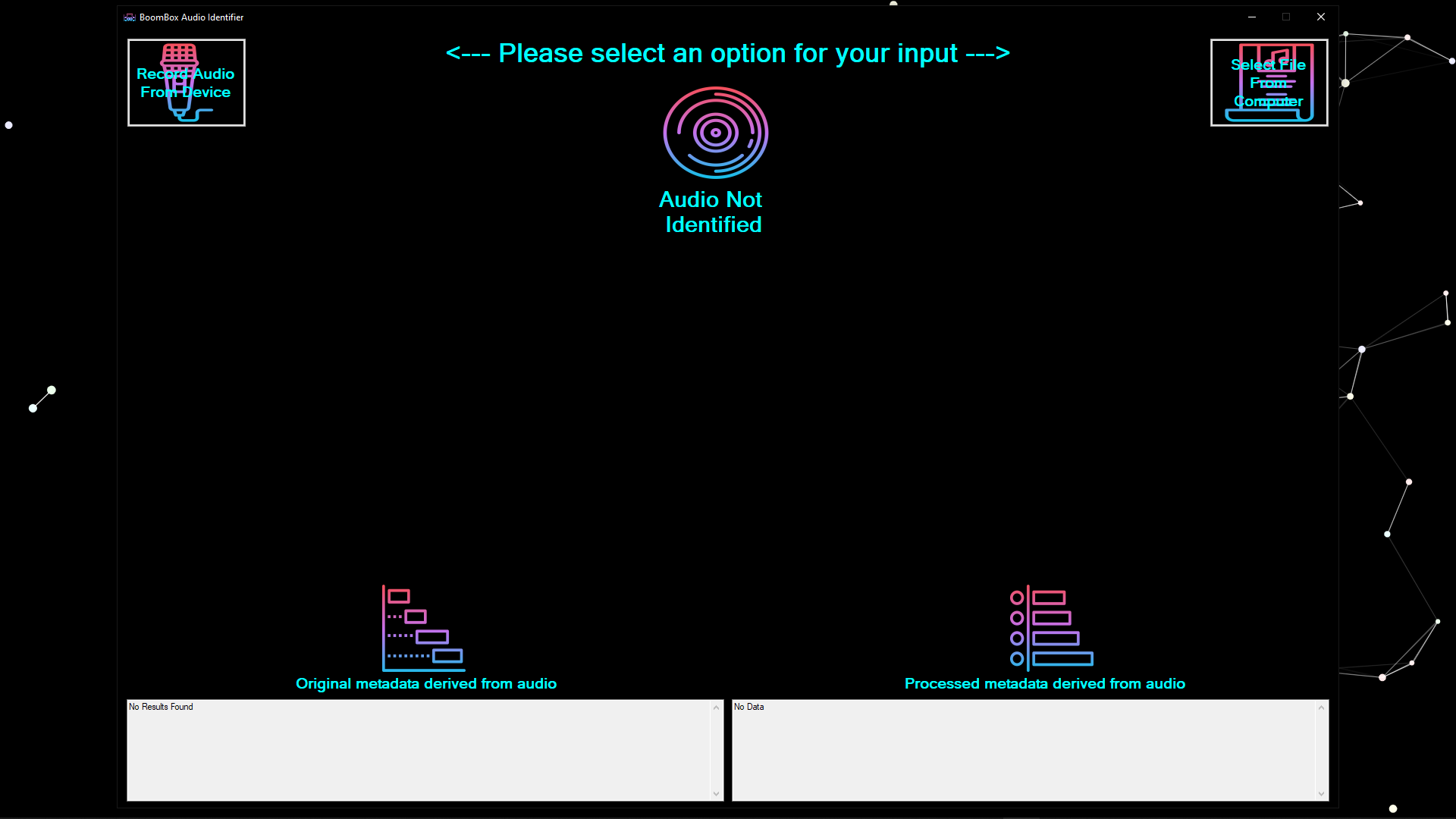


Figure 18 Unrecognised or un-utilisable file form evolution

The user would then be able to re-initialise a test by either re-recording an audio sample, by selecting another pre-recorded file from their device.

### Identified audio piece

In the event that the artefact does identify an audio sample, the user will be presented with an interface resembling the example shown in Figure 19:

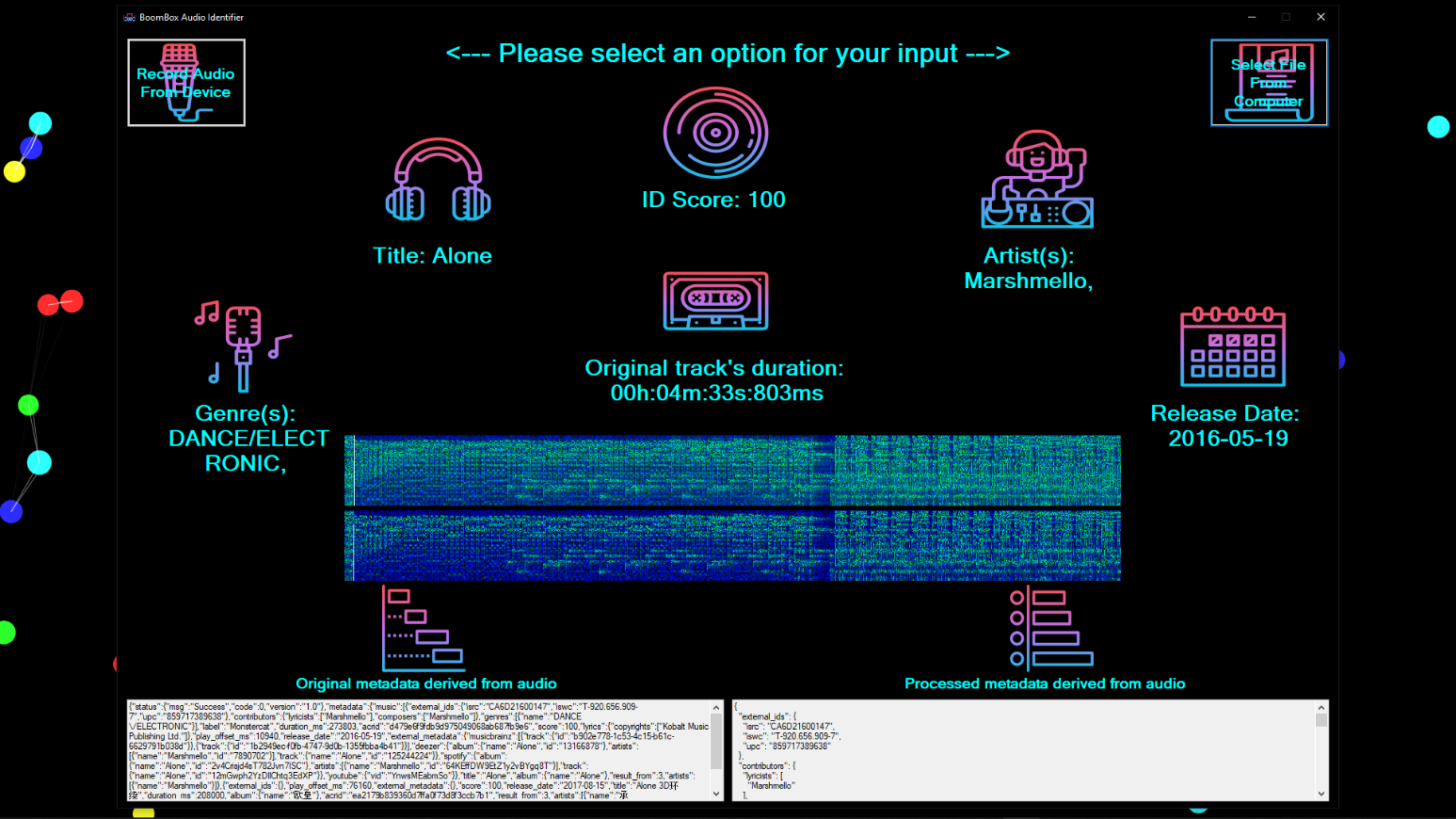


Figure 19 Identified audio piece

The user will be presented with relevant information regarding their submitted audio file within this colourful, easily readable and visually appealing form. The different controls, as well as their functions, are as follows:

Table 2 Denotation of controls used within the main form

|  |  |
| --- | --- |
| ***Control*** | Purpose of control |
|  | This control denotes a score out of 100 and represents the percentage of certainty of the artefact’s identification and classifications of the provided audio piece.  In the event that an audio file cannot be identified, the text of the control will simply read “Audio Not Identified” |
|  | This control denotes the title of the identified audio track, as it had been granted by its original composer. |
|  | This control denotes a list of all artist that had contributed in the production of an audio track and had been denoted as such within the original documentation of the audio track. |
|  | This control denotes a subjective classification of the genre into which the audio track is identified. |
|  | This control denotes the original track’s duration in terms of hours, minutes, seconds and milliseconds. |
|  | This control denotes the date when the original track had been released for public consumption and utilisation. |
|  | This control denotes an example spectrograph that is utilised within the creation of the track’s audio fingerprint. This control functions as a graphic which shows the creation of the spectrograph as the audio sample are being played back to the user in real-time. |
|  | This control denotes the original metadata text received from the server in an unprocessed or “raw” form |
|  | This control denotes metadata that had gone through the first set of interpretation, formulation and regularisation. |

## The life cycle followed and its different phases

Within this section of the document, the different processes and practices that had been implemented during the development lifecycle of the project are explored and explained.

### Background of the chosen development methodologies

There are a plethora of different ways to organise and optimise the process of developing and implementing software. The term “Software Development Methodology” is used to refer to the practices and processes followed during the planning, creating, testing and deployment of a project/software implementation (Lynn, 2018).

The regulations and implementations of software development methodologies are continuously changing to better reflect the current technological paradigm in circulation. This has resulted in the creation of

The driving concept behind traditional software development methodologies had been to assist in the development of different information system components by following deliberate, structured and methodical processes which required every stage of an implementation’s life cycle (meaning from the point of conception of the point of delivery of the final system) to be carried out rigidly and sequentially within the context of the framework being applied (Elliott, 2004).

Less traditional and more modernised frameworks range from a plethora of different specialisations. One of the more popular implementations includes the utilisation of flexible frameworks which enables developers to generate their own custom set of steps tailored to the needs of a specific project (Qureshi, 2012).

In 2019 the GoodFirms organisation, an organisation aimed at functioning as a research and review platform that helps software buyers and service seekers to opt for the best software or firm, had performed a series of industry surveys and research. This research had consulted and investigated of a collection of over 150 software development companies as well as utilised detailed analyses provided by industry experts in order to determine the industry essentials, identify the latest trends, and produce market insights (Raymond, 2019).

According to research performed, the Software Development Methodology utilisation statistics resemble the following:

* 61.5% of organisations utilise the Agile Software Development Methodology;
* 23.1% of organisations utilise the Scrum Software Development Methodology;
* 9.6% of organisations utilise the Waterfall Software Development Methodology; and
* 5.8% of organisations utilise other Software Development Methodologies.

Based on this research, it had been decided that this project would either incorporate the Agile development methodology, or the Waterfall development methodology. The following sections consist of considerations and observations made concerning both methodologies.

### The Waterfall methodology

The Waterfall development methodology is a Sequential development model in which the requirements of the project must be clear before proceeding to the next phase of the development lifecycle (Balaji & Murugaiyan, 2012). The following is a list of aspects concerning the Waterfall methodology:

* Each phase of development is completed in order without any overlapping occurring.
* A rigid schedule must be followed and each module of the project must be completed within a set period of time.
* Each module is frozen until the previous module is fully developed. This means that the requirements phase is frozen before the design phase commences, and once the design phase is frozen the coding phase would start.
* Testing is only carried out once the code has been fully developed.
* Documentation and testing occur at the end of each phase, which promotes high-quality documentation and also assists in maintaining the quality of the project.

An example of a typical Waterfall model life cycle can be seen in Figure 20:

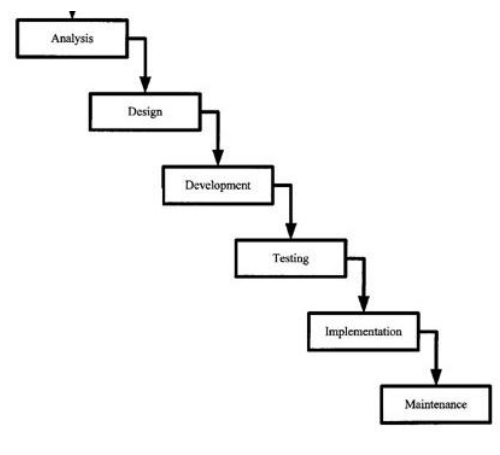


Figure 20 Waterfall Model Life Cycle (Balaji & Murugaiyan, 2012)

The most prominent advantages of implementing this methodology are as follows:

* The requirements of a project are clear before development starts.
* Each phase of a project is completed within a specified amount of time, after which the project moves on to the next phase.
* Because it is a linear model, it is easy to implement.
* A minimal amount of resources are required in order to implement this methodology.
* Each phase of the project is properly documented, thus ensuring the quality of the development cycle.

The most prominent disadvantages associated with the waterfall methodology are as follows:

* Some problems that occur within one phase of development are often never solved completely during that phase and may become more problematic after the phase is signed off, resulting in a badly structured system.
* In the event that any requirements need to be changed, these changes will not be implemented within the current development process, granting the Waterfall methodology a limited amount of adaptability.

### The Agile methodology

In the spring of 2000, a group of 17 software developers, including the likes of Bob Mart, Jeff Sutherland, Jim Highsmith, Jon Kern, Ken Schwabe and Martin Fowler had met in Oregon in order to deliberate how they could speed up the development times of projects in order bring new software to the market at a faster rate (Beck *et al.*, 2001b).

Within this meeting, the group had decided on two key opportunities that would allow them to speed up development times. These opportunities are as follows:

1. Shortening the delay of benefits to users in order to resolve the product-market fit and development graveyard problems
2. Getting feedback from users quickly to confirm the usefulness of new software and continue to improve on it accordingly.

This meeting had consequently resulted in the creation of the Agile Manifesto, and the Agile methodology.

The Agile software development methodology is based on an iterative and incremental approach which allows the requirements of a project to change over time. When implemented efficiently, the Agile methodology even accommodates changing requirements late in development.

The Agile methodology prescribes twelve principles that are aimed at guiding project teams to effectively and efficiently execute project development. These twelve principles are as follows:

1. Businesspeople and developers must collaborate daily throughout the project lifecycle.
2. During the lifecycle of a project, regular meetings would be held wherein the applicable developmental entities would reflect on how to become more effective. They would then adapt their behaviour accordingly.
3. Fully functional software must be frequently delivered, from a couple of weeks to a couple of months, with preference to the shorter timescale.
4. In order to enhance agility, continuous attention must be placed on technical excellence and good design.
5. Processes based on the Agile methodology are developed to place higher importance on sustainable development. This means that the applicable sponsors, developers, and users would be able to maintain a constant pace of development, indefinitely.
6. Projects should be built around motivated individuals. These individuals must be presented with the environment and support they need and must be trusted to get the job done.
7. Simplicity -- the art of maximizing the amount of work not done -- is essential.
8. The best architectures, requirements, and designs emerge from self-organizing teams.
9. The highest priority of the Agile methodology is to satisfy the customer through early and continuous delivery of valuable software.
10. The most efficient and effective method of communication is face-to-face conversation.
11. Welcome changing requirements, even late in development. Agile processes harness change for the customer’s competitive advantage.
12. Working software is the primary measure of progress.

The abovementioned twelve principles of the Agile Software Development Methodology had been derived from the official Agile Manifesto (Beck *et al.*, 2001a).

The most important principles within the Agile methodology is customer satisfaction through the rapid and continuous delivery of small and useful software (Balaji & Murugaiyan, 2012). An example of a typical Agile model life cycle can be seen in Figure 21:

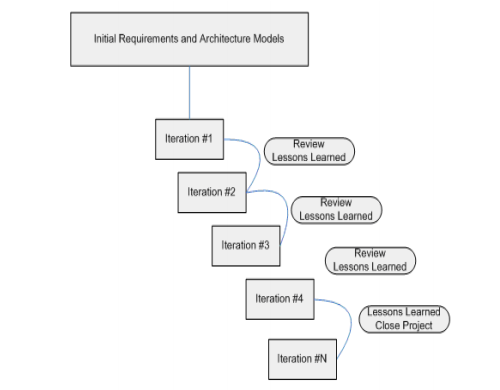


Figure 21 Agile Model Life Cycle (Balaji & Murugaiyan, 2012)

The most prominent advantages of implementing the Agile methodology is its ability to respond to the changing requirements of a project adequately and efficiently.

The most prominent disadvantages associated with the Agile methodology is that the agile methodology is most profitable when being utilised in smaller projects, but if it is implemented in a larger project, then it becomes difficult to judge the efforts and the time required for the project in the software development life cycle.

### Choosing which methodology to implement

Both methodologies present their own set of advantages and disadvantages. However, through further consideration and research, it had been deduced that the Agile methodology would function much more efficiently than the Waterfall methodology.

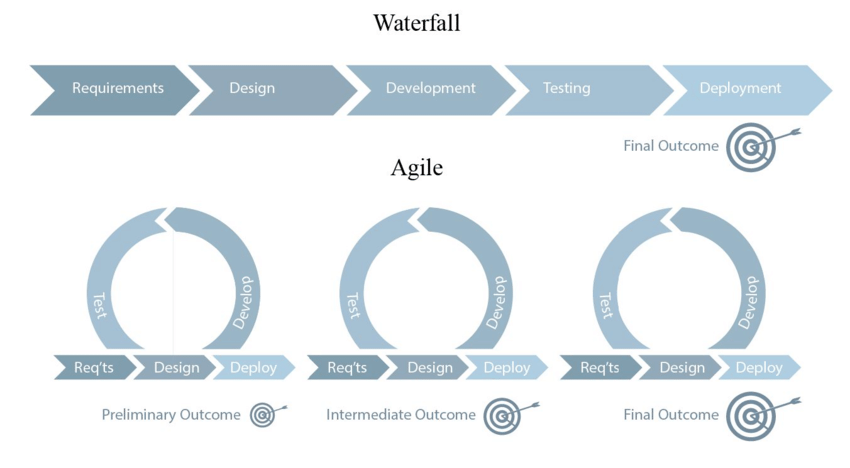


Figure 22 The Agile methodology versus the Waterfall methodology

It can be seen in Figure 22 that the Agile methodology can be thought of as an iterative extension of the Waterfall methodology. The Agile methodology consists of a number of sprints similar to the Waterfall methodology and also encourages an ongoing cycle of improvement and adaptation. Adaptability had been deemed an extremely important aspect of the project and because of this, a version of the Agile software development methodology had been selected to be implemented during the lifecycle of the project.

## Description of the development of the artefact

Within the project’s lifecycle, the Agile methodology had been implemented in order to effectively and efficiently create, adapt and test the final project artefact. Within this part of the document, a detailed explanation of the processes and practices followed during the project lifecycle are explored.

Throughout the lifecycle of the project, the artefact had gone through a number of different iterations. Each iteration had been the focus of a single sprint and the different iterations of the project had been focussed on adding new functionality to the artefact.

The following table is a representation of the different iterations and sprints that had concluded in order to create the final artefact.

Table 3 Detailed descriptions of each implemented phase within the software development lifecycle.

|  |  |  |
| --- | --- | --- |
| **Phase** | **Description** | **Tasks** |
| **Planning** | | |
| 1 | Project conception | The first step had been to contemplate and deliberate the ideal frameworks, architectures and components to utilise within the project.  The following had been utilised:   * Front-end development:   + Visual studio form designer * Back-end development   + C#   + Windows Forms Application Architecture * Audio Fingerprint creation:   + ACRCloud C# Audio Fingerprinting algorithm * Metadata Database:   + ACRCloud metadata database |
| 2 | Obtaining resources | After the necessary components had been identified, the next step had been to acquire the necessary applications and permissions. The front end development and back end development components had been acquired through the Visual Studio programming platform. Permissions for utilising the ACRCloud Audio fingerprinting framework and ACRCloud metadata database had been acquired after communicating and developing an agreement with the ACRCloud software development team. |
| 3 | First sprint planning | Once the appropriate permissions and components had been acquired, the following step was to design the ergonomics of the first development sprint.  This sprint entailed the general preparation and initial setup of all of the acquired components. |
| **Development** | | |
| **Sprint 1 – Initial setup of components** | | |
| 4 | Set up of development environment | The first task of the first sprint had been to prepare the development environment. A new project had been started by utilising the visual studio development platform. This had created a basic form unto which to work. |
| 5 | Set up of ACRCloud requirements | Next, the ACRCloud framework had to be set up within the back end of the project. This required that an account be created with the ACRCloud organisation and that a specialised development key be inputted every time the artefact executed. This key had been obtained during phase 3. |
| 6 | Testing and calibrating the connection between the ACRCloud server and the artefact. | After the set up of the ACRCloud requirements, both the artefact and ACRCloud server that had been utilised had to be calibrated to communicate with one another, transfer the appropriate data and test for the appropriate content. |
| **Sprint 2 – Developing the first iteration of the artefact** | | |
| 7 | Development of an interface for debugging purposes. | Once the setup had been completed, a basic user interface had been developed, which had consisted of a single button and four text boxes. This interface had been utilised for debugging purposes within the early stages of the artefact’s development. |
| 8 | Creating the first iteration of the fingerprinting algorithm. | With the interface in place, the first iteration of the fingerprinting algorithm had been developed.  This iteration would create a spectrograph of a specific file within the program’s “bin” folder, utilise said spectrograph to create an audio fingerprint and send said audio fingerprint to the aforementioned ACRCloud server for identification. The ACRCloud server would then return metadata in the form of a JavaScript Object Notation string. |
| 9 | Attempting an identification within a controlled environment | A pre-identified audio file would be utilised with the first iteration of the fingerprinting algorithm in an attempt to test if the algorithm functioned. The metadata that had been retrieved would then be stored in one of the aforementioned text boxes. |
| 10 | Developing the first instance of the metadata processing algorithm. | When first receiving the metadata from the server, it is relatively difficult to read. An example hereof can be seen in Figure 13. As such, the deciphering of this metadata had become a priority. This had led to the development of the metadata processing algorithm, which would transform the metadata into the example seen in Figure 14. |
| **Sprint 3 – Developing the second iteration of the artefact** | | |
| 11 | Creating the capability to identify any pre-recorded audio on a computer. | After the artefact had the capability to identify a specific audio sample in a controlled environment, the next objective had been to identify any pre-recorded audio file located anywhere on a computer. To achieve this an open file dialogue implementation had been utilised to provide the user with a graphic that resembles the default windows file explorer. The user would then be able to easily navigate to, and select, the file that they would like to attempt to identify. Another button had been added to the form in order to initialise this process. The same fingerprinting algorithm produced in phase 8 had been utilised here. |
| 12 | Creating the capability to identify any audio that the user had recorded through their microphone. | After the artefact had the capability to identify any pre-recorded audio sample that the user has stored on their computer, the next step had been to allow the user to record an audio track through their microphone in an attempt to identify said audio track. To achieve this the user’s default input device is detected from the system's settings and a bitstream utilised in order to serialise any input that is picked up from the user’s microphone. Two more buttons had been added to the form in order to initialise this process. Once the user had recorded the audio that they would like to attempt to identify, they would be able to press the other button in order to initialise the fingerprinting algorithm. The same fingerprinting algorithm produced in phase 8 had been utilised here as well. |
| **Sprint 4 – Developing the third iteration of the artefact** | | |
| 13 | Further developing the metadata processing algorithm | After achieving the fingerprinting capabilities of both pre-recorded and newly recorded audio samples, the next focus point had been to develop the final iteration of the metadata processing algorithm. This was achieved by utilising the first iteration of the metadata processing algorithm as mentioned in phase 10 and creating a second metadata processing algorithm that would read specific lines from the already processed metadata. This information had then been stored within different string variables within the artefact’s backend. |
| 14 | Further developing the user interface | Once most of the backend processing functioned properly, focus had been placed into further developing the graphical user interface. All of the text boxes had been removed, except for two, which would be used to display the unformatted and somewhat formatted metadata to the user. Six labels had then been added in order to display the final formatted metadata, derived in phase 13, to the user. |
| 15 | Developing human error failsafe systems | Within this phase, the program had been stress tested to see in which ways the user might be able to cause the artefact to malfunction. The artefact had then undergone a set of “fool-proofing” procedures and code had been added to ensure that the artefact would function properly in all foreseeable scenarios. |
| **Sprint 5 – Developing the final iteration of the artefact** | | |
| 16 | Rearranging and adding icons to the metadata labels | After the artefact had functioned as intended in terms of audio processing and identification, the next step had been to present the metadata to the user in a visually pleasing graphical user interface. To achieve this, the metadata labels that had been created in phase 14 had been formatted to be easily readable.  Additionally colourful and corresponding icons had been added to each label in order to make the program more visually pleasing. |
| 17 | Adding a live spectrograph generator to the form. | Once phase 16 had concluded, the program had been given to different application “testers” which had been asked for additional advice or preferences. One of the more prominent recommendations had been to add a live spectrograph generator to the artefact’s interface. This had been achieved by utilising a picture box control, converting it into a live graphic and utilising a visual studio library called “Un4seen.Bass”. Once this had been implemented, the artefact had been completed. |

As seen in the above table, the artefact had gone through five different sprints in which a myriad of different functionalities, controls and aspects had been added, adapted and removed from the artefact.

## Conclusion of the development

This section of the document had discussed the development methodology utilised during the lifecycle of the project, as well as the artefact lifecycle that had taken place.

A background of two of the most popular software development methodologies, namely the Agile methodology and the Waterfall methodology, had been presented. Accompanying this had been a discussion of the thought process utilised for choosing the most ideal software development methodology to be implemented within this project, namely Agile.

The Agile methodology had boasted a considerable amount of focus on the concept of adaptability. Based on the changing events and paradigm shifts that had occurred during the current year, it would stand to reason that the Agile methodology had been the ideal developmental methodology to implement and had benefitted the lifecycle of the project greatly.

# Results

Within this section of the document, a discussion of all of the data centred tests and their applicable results are presented.

## The data set used to perform the tests

In order to perform non-biased tests that accurately reflect the current auditory paradigm in effect, a dataset of audio samples had to be chosen that had not been selected by the researcher, but rather by popular demand across the world.

Currently, Spotify is considered to be the world’s largest on-demand music streaming organisation, with over 75 million active users listening to tens of millions of songs and other audio tracks (Jacobson *et al.*, 2016). The dataset that had been utilised to perform the test are the thirty most-streamed audio tracks contained within the Spotify database. Information concerning the thirty tracks that had been utilised can be seen within the following table:

Table 4 The 30 most streamed songs on Spotify

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Rank** | **Song** | **Artist(s)** | **Album** | **Streams (Millions)** | **Date published** |
| 1 | "Shape of You" | Ed Sheeran | ÷ | 2,645 | 6 January 2017 |
| 2 | "Rockstar" | Post Malone featuring 21 Savage | Beerbongs & Bentleys | 2,056 | 15 September 2017 |
| 3 | "Dance Monkey" | Tones and I | The Kids Are Coming | 1,945 | 10 May 2019 |
| 4 | "One Dance" | Drake featuring Wizkid and Kyla | Views | 1,939 | 5 April 2016 |
| 5 | "Closer" | The Chainsmokers featuring Halsey | Collage | 1,890 | 29 July 2016 |
| 6 | "Sunflower" | Post Malone and Swae Lee | Spider-Man: Into the Spider-Verse & Hollywood's Bleeding | 1,712 | 18 October 2018 |
| 7 | "Blinding Lights" | The Weeknd | After Hours | 1,662 | 29 November 2019 |
| 8 | "Thinking Out Loud" | Ed Sheeran | × | 1,647 | 20 June 2014 |
| 9 | "Señorita" | Shawn Mendes and Camila Cabello | Shawn Mendes & Romance | 1,647 | 21 June 2019 |
| 10 | "God's Plan" | Drake | Scorpion | 1,640 | 19 January 2018 |
| 11 | "Someone You Loved" | Lewis Capaldi | Breach & Divinely Uninspired to a Hellish Extent | 1,610 | 8 November 2018 |
| 12 | "Bad Guy" | Billie Eilish | When We All Fall Asleep, Where Do We Go? | 1,594 | 29 March 2019 |
| 13 | "Say You Won't Let Go" | James Arthur | Back from the Edge | 1,562 | 9 September 2016 |
| 14 | "Perfect" | Ed Sheeran | ÷ | 1,560 | 3 March 2017 |
| 15 | "Believer" | Imagine Dragons | Evolve | 1,534 | 1 February 2017 |
| 16 | "Havana" | Camila Cabello featuring Young Thug | Camila | 1,521 | 3 August 2017 |
| 17 | "Photograph" | Ed Sheeran | × | 1,482 | 20 June 2014 |
| 18 | "Love Yourself" | Justin Bieber | Purpose | 1,470 | 9 November 2015 |
| 19 | "Starboy" | The Weeknd featuring Daft Punk | Starboy | 1,456 | 21 September 2016 |
| 20 | "Lucid Dreams" | Juice WRLD | Goodbye & Good Riddance | 1,439 | 4 May 2018 |
| 21 | "Sad!" | XXXTentacion | ? | 1,436 | 1 March 2018 |
| 22 | "New Rules" | Dua Lipa | Dua Lipa | 1,436 | 2 June 2017 |
| 23 | "Something Just Like This" | The Chainsmokers and Coldplay | Memories...Do Not Open | 1,422 | 22 February 2017 |
| 24 | "Lean On" | Major Lazer and DJ Snake featuring MØ | Peace Is the Mission | 1,416 | 2 March 2015 |
| 25 | "Despacito (Remix)" | Luis Fonsi and Daddy Yankee featuring Justin Bieber | Vida | 1,400 | 17 April 2017 |
| 26 | "Thunder" | Imagine Dragons | Evolve | 1,393 | 27 April 2017 |
| 27 | "Sorry" | Justin Bieber | Purpose | 1,378 | 22 October 2015 |
| 28 | "XO Tour Llif3" | Lil Uzi Vert | Luv Is Rage 2 | 1,366 | 24 March 2017 |
| 29 | "Don't Let Me Down" | The Chainsmokers featuring Daya | Collage | 1,355 | 5 February 2016 |
| 30 | "Bohemian Rhapsody" | Queen | A Night at the Opera | 1,346 | 31 October 1975 |

The following are a set of tests conducted on the abovementioned thirty audio tracks.

## The minimum duration of an audio sample needed to effectively identify a track

As the artefact had been designed to be utilised in scenarios in which an entire audio sample may not be obtainable, this test had been implemented in order to determine the minimum duration of a sample needed to be able to identify an audio sample.

The test consisted of recording different duration sections from each of the thirty samples and sending them through the artefact. The results recorded are as follows:

Figure 23 Accuracy rating in terms of sample duration

After the test had been completed, it had been concluded that the optimal minimum duration of a sample must not be less than seven seconds, as this may result in the incorrect identification of a track, or the track not being identified at all. This assumption is supported by the drastic decrease in the number of correct identifications between the seven-second duration tests and the five-second duration tests.

Additionally, this test also supports the deceleration of robustness for the artefact, as it demonstrates the identificatory capabilities of the artefact, even when only presented with a few seconds of auditory data.

## Identification accuracy at different levels of background noise

The artefact had been developed with the capability to identify audio tracks within environments that contain background noise. Within this test, the artefact had been tasked with identifying the thirty different sample tracks within environments that contain varying levels/intensities of background noise.

The following charts represent the results of the identification attempts made within environments that contain no background noise, a small amount of background noise (such as people talking in the distance or in another room), a medium amount of background noise (such as conversations being held within the same room such as a coffee shop), and a large amount of background noise (intentionally loud conversations near the microphone).

Figure 24 Accuracy in environment with no background noise

The environment utilised within this scenario had been an empty room in which only the audio sample was being played. Within this environment, the artefact functioned ideally, and could accurately identify all thirty of the audio tracks.

Figure 25 Accuracy in environments with a small amount of background noise

The environment utilised within this test had been a public nature park. Within this environment, the artefact also functioned ideally and could accurately identify all thirty of the audio tracks.

Figure 26 Accuracy in environments with a medium amount of background noise

This test had been conducted within a simulated coffee shop environment, as the restrictions presented by the COVID-19 virus had still been in effect. Within this environment, the artefact had struggled to identify some of the samples, as the level of background noise had started to affect the spectrograph and fingerprinting algorithms.

Figure 27 Accuracy in environments with a large amount of background noise

This test had been conducted within a simulation of a crowded concert hall, as the restrictions presented by the COVID-19 virus had still been in effect. Within this environment, the artefact had struggled immensely to identify most of the audio samples accurately. It is believed that this occurs because the intensity of the background noise completely overtakes any meaningful data that could be derived from a spectrograph recorded within this environment.

It should be noted that some of the abovementioned tests had to be conducted within simulated environments, as the restrictions presented by the COVID-19 virus had rendered it unsafe, and in some cases illegal, to perform the tests within the actual venues or environments.

Based on the results of these tests, it had been deduced that the artefact would function ideally in most scenarios where background noise would not be considered excessive or overbearing. However, the evidence of the test of accuracy in environments with a medium amount of background noise supports the concept that the artefact would still function ideally within most scenarios that users would find themselves, only failing to perform adequately within situations of extreme background noise.

# Chapter – Reflection

Within this section of the document a brief reflection concerning a number of different topics, as provided by the “Addendum D” document, is provided.

## Lessons learned throughout the project life cycle.

During the lifecycle of this project, newer concepts such as the ergonomics and necessity of automatic content recognition had been explored and the general knowledge base of the researcher advanced. Concepts such as audio fingerprinting and its impressive capabilities had also been an unknown factor before commencing within this project.

Additionally, newer forms of data storage, processing and translation in the form of JavaScript Object Notation files had been thoroughly researched and experimented with during the lifecycle of the project.

Newer methods of information presentation had also been researched, which allowed for the implementation of live graphics within the project. This concept of continuously updating graphics had also proved to be an implementation of value and will be implemented within future projects.

Finally, due to the constraints and paradigmatic shifts that had occurred due to the COVID-19 global health pandemic, the advantages and necessity of the concept of adaptability had also been made abundantly clear. Accompanying this realisation, was the importance of ideally implementing an adequate software development methodology.

## Evaluation of decisions made within the project

A plethora of different developmental and implementational decisions had been made throughout the lifecycle of this project. The following table is a representation of the most prominent decisions made, accompanied by a brief evaluation of the decision made.

|  |  |
| --- | --- |
| **Decision made** | **Explanation and evaluation** |
| The utilisation of C# and Visual Studio as a programming platform and programming language. | C# and Visual Studio had been utilised within the project as they had been the platform and language that the researcher had been most familiar with. In reflection, the C# programming language had proved to be a powerful language capable of performing the necessary processing in an efficient and resource un-intensive manner, additionally, the Visual Studio platform had provided an easy to utilise platform to design and test user interfaces and software implementations. |
| The utilisation of the Agile development methodology. | The reasons that the Agile methodology had been chosen to be utilised are discussed within section 3.2.4. In reflection, the Agile methodology had proved to be an incredible asset during the development of the project, as it had provided a developmental framework that allowed and even encouraged change. Needless to say that with the events of the year 2020, change had been an inevitability, and as such the Agile methodology had been utilised to its utmost potential. |
| The utilisation of the ACRCloud services. | While doing research considering different metadata acquisition and storage methods, the ACRCloud organisation had been discovered. They had boasted an impressive metadata database with a moderate amount of support for utilising their services. In reflection, based on the results obtained throughout the testing and implementation of the artefact, ACRCloud had proved to be an adequate data source for the duration of the project. It should be noted that some of the records that are stored within their database are not complete and require further quality control in order to be up to standard in comparison with some of their more well-documented files. |

## Analysis of the strong and weak points of the product and the process that had been followed.

One of the strongpoints noted of the project is its capability and potential to function in a plethora of environments. During the testing phase of the artefact (documented in section 4.3), the artefact had presented the ability to function in almost any environment where the level background is not overbearing.

Another strong point of the project is its large database of audio fingerprints and auditory metadata. With over 40 million records of different audio tracks and their applicable metadata, the project has a higher possibility of correctly identifying most audio samples within circulation today.

One of the weak points of the project is that whilst the fingerprinting database that is used is extremely vast, some of its records are incomplete. This means that whilst an audio sample may be easily identified, not all metadata will be presented to the user. The most prominent example of this are samples which have atypical genre classifications. In these cases, the user will not be presented with the genre, which would simply display that the genre could not be identified.

## What could have been done differently?

Were there fewer constraints (as caused by the COVID-19 epidemic), it is believed that the project may have benefited from live-action tests and peer reviews. Additionally, where there more time to complete the project, creating a custom metadata database could possibly have provided the project with more relevant metadata, such as embedded links that can be utilised to listen to a sample of the audio, or even album cover art.

Additionally, had the researcher a broader understanding of coding principles concerning web development, the project would have been more widely available and made accessible to a larger group of individuals.

## Had the objectives set in this project been achieved?

This project had aimed to answer the research question: *How can auditory information be captured, translated into a machine-processable format, interpreted and presented to users in the form of metadata information?*

Based on this question the project consists of the following capabilities:

* The artefact is able to capture auditory information by utilising an input device and my utilise pre-recorded auditory samples that are present within the user’s machine.
* The artefact is able to transform audio samples into a machine-processable format by utilising spectrographs.
* The artefact is able to interoperate these spectrographs by utilising audio fingerprinting
* The artefact is able to present the users with metadata by utilising a graphical user interface.

Based on the abovementioned capabilities of the artefact, it is clear that the proposed research question had been answered. As such the objectives that had been set for the project had been achieved.

## How successful had the project been managed in meeting target dates?

When reflecting on the management of the project and the time provided to complete it, the project development had gone relatively smoothly, with some events being caused by the COVID-19 epidemic to alter some aspects of the project, such as live testing and user-based reviews. However, the project had come to a satisfactory conclusion and had been completed within the allotted time frame.

## Critical evaluation of the process followed

During the lifecycle of the project, an Agile methodology had been implemented. The Agile methodology had originally evolved from a collection of different lightweight software approaches in the 1990s and was originally meant to function as a response to the rigid and linear Waterfall methodology(Cockburn & Highsmith, 2001).

Based on the events that had occurred throughout the year, the Agile methodology’s focuses on flexibility, continuous improvement, and speed had proved to be a cornerstone during the development of the project. And through the effective utilisation of the Agile development methodology, a complete system that thoroughly answers the proposed research question, had been created.

# Bibliography

ACRCloud. 2019 Brand Guidelines. <https://www.acrcloud.com/branding/> Date of access: 2020/03/02.

Aldridge, D. 1993. The music of the body: music therapy in medical settings. *Advances*.

Balaji, S. & Murugaiyan, M.S. 2012. Waterfall vs. V-Model vs. Agile: A comparative study on SDLC. *International Journal of Information Technology and Business Management*, 2(1):26-30.

Beck, K., Beedle, M., Van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A. & Jeffries, R. 2001a. The agile manifesto: Feb.

Beck, K., Beedle, M., Van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A. & Jeffries, R. 2001b. Manifesto for agile software development.

Burges, C.J., Plastina, D., Platt, J.C., Renshaw, E. & Malvar, H.S. 2005. Using audio fingerprinting for duplicate detection and thumbnail generation*.* (*In*. Proceedings.(ICASSP'05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005. organised by: IEEE. p. iii/9-iii12 Vol. 13).

Cano, P., Batle, E., Kalker, T. & Haitsma, J. 2002. A review of algorithms for audio fingerprinting*.* (*In*. 2002 IEEE Workshop on Multimedia Signal Processing. organised by: IEEE. p. 169-173).

Cano, P., Batlle, E., Gómez, E., de CT Gomes, L. & Bonnet, M. 2005a. Audio fingerprinting: concepts and applications. Computational intelligence for modelling and prediction. Springer. p. 233-245).

Cano, P., Batlle, E., Kalker, T. & Haitsma, J. 2005b. A review of audio fingerprinting. *Journal of VLSI signal processing systems for signal, image and video technology*, 41(3):271-284.

Chirp. 2018. Audio fingerprinting — what is it and why is it useful? The Data Over Sound Blog: The Data Over Sound Blog.

Cockburn, A. & Highsmith, J. 2001. Agile software development, the people factor. *Computer*, 34(11):131-133.

Collins, H. 2018. Creative research: the theory and practice of research for the creative industries: Bloomsbury Publishing.

Crowther, D. & Lancaster, G. 2012. Research methods: Routledge.

David M. Greenberg, P.D. 2016. What Is Music…Exactly? *The Power of Music*. psychologytoday.com: psychologytoday.com.

Elliott, G. 2004. Global business information technology: an integrated systems approach: Pearson Education.

Feigl, H. 2019. Positivism. <https://www.britannica.com/topic/positivism> Date of access: 2020/04/16.

Gravell, D. 2016. Choosing an audio fingerprinting service. *Market*. <https://www.acrcloud.com/blog/choosing-audio-fingerprinting-service> Date of access: 2020/04/08.

Green, L. 2006. Popular music education in and for itself, and for ‘other’music: current research in the classroom. *International journal of music education*, 24(2):101-118.

Haitsma, J. & Kalker, T. 2002. A Highly Robust Audio Fingerprinting System*.* (*In* organised by.

Jacobson, K., Murali, V., Newett, E., Whitman, B. & Yon, R. 2016. Music personalization at Spotify*.* (*In*. Proceedings of the 10th ACM Conference on Recommender Systems organised by. p. 373-373).

Johnson, B. 2001. Toward a new classification of nonexperimental quantitative research. *Educational Researcher*, 30(2):3-13.

Kania, A. 2007. The philosophy of music.

Liu, F. 2017. Audio fingerprinting for speech reconstruction and recognition in noisy environments*.*

Lynn, R. 2018. The History of Agile. planview.com: Planview. Date of access: 2020-11-13.

Madsen, C.K., Cotter, V. & Madsen, C. 1968. A behavioral approach to music therapy. *Journal of Music Therapy*, 5(3):69-71.

Mohsin, M. 2020. 10 Social Media Statistics You Need to Know in 2020. <https://www.oberlo.com/blog/social-media-marketing-statistics> Date of access: 2020/04/02.

Moreno, S. 2009. Can music influence language and cognition? *Contemporary Music Review*, 28(3):329-345.

mufin.com. 2020. Audio Fingerprinting Technology. <https://www.mufin.com/company/technology/> Date of access: 2020/02/27.

North, A.C., Hargreaves, D.J. & Hargreaves, J.J. 2004. Uses of music in everyday life. *Music Perception: An Interdisciplinary Journal*, 22(1):41-77.

Patil, C.M. 2019. Music Identification based on Audio-Fingerprinting.

Qureshi, M.R.J. 2012. Agile software development methodology for medium and large projects. *IET software*, 6(4):358-363.

Raymond, M. 2019. Remarkably Useful Stats and Trends on Software Development *GoodFirms Research*. <https://www.goodfirms.co/resources/software-development-research> Date of access: 2020-11-13.

Scherer, K.R. & Zentner, M.R. 2001. Emotional effects of music: Production rules. *Music and emotion: Theory and research*, 361(2001):392.

Siva Sankaran, S.K. 2017. Analysis of audio fingerprinting techniques.

Smith, B. 2012. Ontology. The furniture of the world. Brill Rodopi. p. 47-68).

Sonnleitner, R., Arzt, A. & Widmer, G. 2016. Landmark-Based Audio Fingerprinting for DJ Mix Monitoring*.* (*In*. ISMIR organised by. p. 185-191).

Wang, A. 2003. An industrial strength audio search algorithm*.* (*In*. Ismir organised by: Washington, DC. p. 7-13).

Weakly, L. 2017. 7 Reasons Why Music Is Important To World Cultures *Music In World Cultures* <http://www.musicthinktank.com/blog/7-reasons-why-music-is-important-to-world-cultures.html> Date of access: 2020/04/04.

# Appendices

## Research proposal according to the prescribed template

**SUBJECT GROUP COMPUTER SCIENCE AND INFORMATION SYSTEMS**

**Research Proposal for the Honours project**

The student and the supervisor must consult the *Manual for Postgraduate Studies* prior to writing the research proposal. The *Manual for Postgraduate Studies* explains in detail what is expected at each of the subheadings below. The proposal should not be longer than 5 pages.

The Subject Group requires that the research proposal will be submitted through the use of this form and in the format below. Please complete using a computer.

**Student initials, surname and student number**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Initials | Z | Surname | Boonzaaier | Student number | 28749995 |

**Degree for which student is registered**

|  |
| --- |
| BSC.IT HONOURS |

**Name of supervisor**

|  |  |
| --- | --- |
| Initials and surname | Mr. D.P. Snyman |

**Proposed title**

|  |  |
| --- | --- |
| Title (preferably not more than 12 words) | Identification of auditory information by utilising audio fingerprinting. |

**Problem statement and substantiation**

Provide the theme and link with gaps in the literature and recent research in the area. Indicate the research question, its actuality and how the research will endeavour to answer the question.

|  |
| --- |
| Statistics over the years suggest a sharp increase in the number of different bodies of media that exist today. Every day the influence, scope, variety and sheer quantity of different media entities increase exponentially (Mohsin, 2020).  One of the more prominent aspects of these different types of media is auditory information in the form of music and speech. There are various characteristics of auditory information that influences different aspects of the daily lives of people (and animals) around the world. These aspects include but are not limited to communication, development, emotion, interpretation, performance and behaviour (Scherer & Zentner, 2001). Auditory information thus represents a recourse that could be utilised to achieve a number of different goals. These utilisations include, but are not limited to, medicinal applications, general performance enhancement and behavioural manipulation. In accordance with this, this project aims to answer the research question:  *How can auditory information be captured, translated into a machine-processable format, interpreted and presented to users in the form of metadata information?*  To answer this question, an IoT based system will be developed to attempt to capture, processes and interoperate auditory information. This system will be utilising an artificial intelligence process referred to as audio fingerprinting to map various instances of auditory information. Audio fingerprints are defined as follows: “An audio fingerprint is a unique and compact digest derived from perceptually relevant aspects of a recording. Fingerprinting technologies allow the monitoring of audio content without the need of metadata or watermark embedding.”(Cano *et al.*, 2005a) |

**Research aims and objectives**

Provide the different general as well as the specific aspects which will form part of the research.

|  |
| --- |
| The aim of this project is to attempt to answer the proposed research question by developing an IoT (Internet of Things) based system and/or software with the goal of effectively capturing, processing and interoperating auditory information.  Four categories of objectives have been identified as the main focus points of the project. These objectives are:   * research; * hardware; * software; and * code-based objectives.   During the lifespan of the project, a series of documentation will be produced to aid in researching the proposed research question. These documentations may include but are not limited to, literature reviews, educational posters, potential user reviews, and possible experimentation reports.  The first objective in creating the system, is to source most of the hardware components, in order to ensure low-level functionality. It would also be important to ensure that the different hardware components are compatible with one another.  The next objective would be to source and utilise the appropriate software for the development of the project. There are a plethora of useful recourses available that could be utilised in the answering of the project question. It is thus, imperative that the project only utilise the necessary services and components that are sufficient in making it as functional as possible.  The final objective would be to code and develop an IoT based system with the capabilities of capturing, processing and interoperating auditory information. Throughout the development of all of these objectives, it would be imperative that documentations concerning the lifecycle of the project be made and kept up to date. |

**Basic hypothesis (where applicable)**

|  |
| --- |
| The proposed hypothesis is that by utilising audio fingerprinting on audio samples that consist of a long enough duration and which consist of a high enough quality, that any audio sample could be recorded, the appropriate audio fingerprint created and stored and used to identify the same audio sample with corresponding metadata. |

**Literature study**

Provide an indication only of which literature will be used in the study with a few key references. A summary of the literature is not required here.

|  |
| --- |
| There are many literature pieces concerning the structure and effects of auditory information, especially concerning music. Some of the literature that will be utilised in this project are:  Audio fingerprinting: concepts and applications (Cano *et al.*, 2005a).  A review of audio fingerprinting(Cano *et al.*, 2005b).  A Highly Robust Audio Fingerprinting System (Haitsma & Kalker, 2002).  A behavioural approach to music therapy (Madsen *et al.*, 1968).  Uses of music in everyday life (North *et al.*, 2004). |

**Methods of investigation**

The proposed design, data acquisition, procedures, data processing, funding sources (but not a budget), mathematical methods, computer methods, etc.

|  |
| --- |
| The project mainly focuses on the statistical and quantitative data of audio fingerprints. During this project, a deductive approach will we followed in order to more thoroughly describe the relationships between variables, measure concepts quantitatively and generalize research findings to a certain extent. Because of this, the project will utilise an ontological-positivistic research paradigm in answering the proposed research question (Crowther & Lancaster, 2012). The project will utilise a quantitative-correlational research method. This method is a non-experimental method that focuses on the observation and documentation of the relationships between different variables using statistical analyses (Johnson, 2001).  The data that will be used during the lifecycle of this project will be sourced from pre-existing audio fingerprint databases. User-generated content may also be utilised for testing purposes.  Initially, the proposed program would be installed on a device with a microphone. The program may be presented with a piece of auditory information, either from a pre-existing file on the device or from a recording made by utilising the device’s microphone. This auditory information would then be transformed into a corresponding audio fingerprint.  This fingerprint would then be compared to a database of pre-existing audio fingerprints in an attempt to derive metadata from the presented auditory information. The user would then be informed concerning the outcome of the comparisons, and possibly be presented with corresponding metadata.  This project does not require the incorporating of any form of funding sources.  This project does not require the incorporation of any complex mathematical methods.  The project will make use of an artificial intelligence process referred to as audio fingerprinting. Audio fingerprinting will be utilised for the recording, extraction, storage, and interpretation of auditory information. “Audio fingerprinting is best known for its ability to link unlabelled audio to corresponding meta-data (e.g. artist and song name), regardless of the audio format.”(Cano *et al.*, 2005b) |

**Provisional chapter division**

Here it should be clear that there was a proper reflection on the appearance of the final product (mini-dissertation). Provide provisional titles of the various chapters, with a brief outline of the planned content of each.

|  |
| --- |
| **Chapter 1 – Introduction and problem statement**  This chapter will contain an introduction to the study in terms of background information concerning the project, a relevant research question as well as information concerning the research methodology that will be implemented to conduct the study.  **Chapter 2 – Literature review**  Within this chapter, the relevant information that was collected during the literature study of the project will be presented.  **Chapter 3 – Development of the artefact**  Within this chapter, the different strategies, methodologies, platforms, artefact forms and other relevant information concerning the development of the IoT artefact will be presented.  **Chapter 4 –Results**  Within this chapter, a revision of the collected data and general performance of the project within different environments and under various circumstances will be presented and discussed.  **Chapter 5 –Conclusion and reflection**  The last chapter of the document will conclude the study by providing a summary of the project and its findings. Also contained within this chapter would be a revision of the decisions and developments that were made during the project development. |

**Literature references**

Provide complete references to the literature referenced in this proposal only.

|  |
| --- |
| Cano, P., Batlle, E., Gómez, E., de CT Gomes, L. & Bonnet, M. 2005a. Audio fingerprinting: concepts and applications. Computational intelligence for modelling and prediction. Springer. p. 233-245).  Cano, P., Batlle, E., Kalker, T. & Haitsma, J. 2005b. A review of audio fingerprinting. *Journal of VLSI signal processing systems for signal, image and video technology*, 41(3):271-284.  Crowther, D. & Lancaster, G. 2012. Research methods: Routledge.  Haitsma, J. & Kalker, T. 2002. A Highly Robust Audio Fingerprinting System*.* (*In* organised by.  Johnson, B. 2001. Toward a new classification of nonexperimental quantitative research. *Educational Researcher*, 30(2):3-13.  Madsen, C.K., Cotter, V. & Madsen, C. 1968. A behavioral approach to music therapy. *Journal of Music Therapy*, 5(3):69-71.  Mohsin, M. 2020. 10 Social Media Statistics You Need to Know in 2020. <https://www.oberlo.com/blog/social-media-marketing-statistics> Date of access: 2020/04/02.  North, A.C., Hargreaves, D.J. & Hargreaves, J.J. 2004. Uses of music in everyday life. *Music Perception: An Interdisciplinary Journal*, 22(1):41-77.  Scherer, K.R. & Zentner, M.R. 2001. Emotional effects of music: Production rules. *Music and emotion: Theory and research*, 361(2001):392. |

**........................... ............................. .......................**

Student Supervisor Date