Music Visualiser

Project Definition Document (Initial Brief)

Purpose of this document

The purpose of this document is to define a proposed project with enough information for an informed decision to be made on whether to approve the project for production.
This document is the **responsibility of the Project Team.**

Project approval is dependent upon the document being completed and reviewed by the teacher of the project course.
- If the project is approved, it can then begin planning and execution,
- If the project is not approved, the **Project Team** can review the project and resubmit for reconsideration.

Table of Contents

1. Project Definition Information

Project Name:	Music Visualiser
Project Team Members:	Evelyn Gao
Mentoring Teacher:	Edwin Griffin

2. Project Idea

A music visualiser. It would take an audio file as input and play it back with some sort of visualisation paired with it (most likely a sine wave with changing amplitudes). It will do so using frequency/volume analysis.
Ideally, STFT (Short-time Fourier Transform) will be used as the algorithm/technique in doing so, however considering time constraints, the project will likely not show frequency, making STFT unfit for the current prototype. For this prototype, volume as the human perception of loudness of the song will be being displayed using RMS (Root-mean-square). RMS in the context of audio processing takes the samples over a time frame, and gets an average amplitude by squaring all the values and square-rooting the mean of the squared values. Involved in this technique are concepts such as time series and various other things used in audio processing, but the main algorithm thing is RMS.

3. Project Purpose

To provide information on the specifics (perceived volume) of a given song, and display it in an intuitive and interesting way. To investigate audio analysis and audio processing through the techniques used in them such as RMS, sampling, and STFT in the future.

4. Initial Scope of the Project

In Scope	Out of Scope
Any desired WAV file will be able to be played and represented	Actual loading screen sort of GUI to upload files for smoother user experience.
Visualisation displayed with a bar, rectangle, line, or similar.	Full frequency analysis and representation.
Basic pygame display window, command line interaction to start the program.	Abstract, artistic visuals. Aesthetically pleasing interface.
Program can quit at any time and quits at song completion.	Queueing system.

Basic file uploader after 'play' is pressed on the program file.	System to prevent the program from crashing when the user does not upload anything. Like it forces you to upload something or doesn't break

5. Constraints

Constraint	Impact on Project Success (High/Med/Low)
Limited to usage directly through Vscode.	Low
Real time analysis is not possible with Librosa as a library as it loads all the data of a file in one go and analysis done on data must be done on that preloaded array.	Med
Without a file upload, the wav file would have to be downloaded and also entered manually into the code (an upload file function would eliminate this).	Low

6. Initial Issues

Issue	Description
Shorter time to complete prototype than initially given.	As it took longer than expected to pick the idea and thus begin this planning stage, the timeframe for working on the actual project has been cut short.

7. Deliverables, Timeframes and Dependencies

Timeframe estimate: 3 weeks

Deliverable	Duration	Completion Date	Dependencies
Understanding of libraries needed and plan for approach (sort of pseudocode).	1 Week	End Week 3	

Real-time analysis of a file as it plays. (The whole project, essentially).	1 Week	End Week 4	Understanding of interplay and usage of a wide range of libraries.
Deepen knowledge and understanding of library, terminology, and other theory.	1 Week	End Week 5	(Ideally) completed project.
Curate presentation.	1 Week	End Week 5	Completed project and masterful understanding of underlying theory.

8. References

Code
https://www.youtube.com/watch?v=675teI6-_-g&ab_channel=AndingAnalytics
https://www.geeksforgeeks.org/python-find-the-closest-key-in-dictionary/
https://stackoverflow.com/questions/78262383/after-resampling-how-to-get-the-time-matrix-corresponding-to-rms-matrix-using-l

Some Presentation Information
https://librosa.org/doc/0.11.0/index.html
https://unison.audio/what-is-rms-in-audio/
https://ampedstudio.com/rms-in-audio/
https://majormixing.com/what-is-rms-in-audio-world/
https://en.wikipedia.org/wiki/Sampling_(signal_processing)