



**Faculty Of Engineering.**  
**Computer & Control Department.**

**Media Player with  
some effects on the  
Audio.**

**Submitted By:**

- 1 – Youssef Ahmed Hassan Elzedy.
- 2 – Youssef Ashraf Mohamed Megahed.
- 3 – Youssef Salah Saleh Abdelrehem.

**Supervised By:**

Dr.Eslam Abdelfatah Shaalan.

## ACKNOWLEDGEMENT

Primarily, I would like to praise and thank God, the Almighty, who has granted countless blessings, knowledge, and opportunity to this project, so that I have been finally able to accomplish the thesis.

Second, I would like to express my special thanks of gratitude to **Eslam Samy** as well as **Dr. Eslam Shaalan** who gave us the excellent opportunity to do this wonderful program on the topic (Media Player with some effects on the Audio), which also helped us in doing a lot of Research. We came to know about so many new things. We are thankful to them.

Third I would also like to thank my friends who helped me a lot in finalizing this project within the limited period, I am grateful for them.

# ABSTRACT

The audio file player program is a powerful tool for manipulating audio files. It offers a range of advanced features, including the ability to change the speed of the sound, adjust the pitch, and manipulate sound compression and expansion. Additionally, the program can generate white noise to create a soothing ambiance for the listener. These features provide significant advantages for musicians, audio engineers, and music enthusiasts alike. Speeding up or slowing down the sound allows for more efficient learning and analysis, while the compression and expansion features provide a more fantastic range of dynamic expressions. Overall, the audio file player program is a versatile and user-friendly tool that can significantly enhance the listening and editing experience for a variety of applications.

# TABLE OF CONTENT

ACKNOWLEDGEMENT .....	II
ABSTRACT .....	III
TABLE OF CONTENT .....	IV
INTRODUCTION .....	6
1.1 Introduction .....	6
1.2 Motivations .....	6
1.3 Project objective .....	6
1.4 Program Layout .....	7
1.5 Report outline .....	8
1.5.1 Project Libraries.....	8
1.5.2 The method of work .....	8
1.5.3 Advantages.....	8
1.5.4 Program idea .....	8
PROJECT LIBRARIES .....	9
2.1 Tinter .....	9
2.1.1 Simple application .....	9
2.2 Pygame.....	10
2.3 Librosa.....	11
2.4 Numpy.....	12
2.5 Sound files.....	13
2.6 Mixer .....	13

THE METHOD OF WORK.....	14
3.1 How Program Works .....	14
ADVANTAGES .....	16
4.1 White noise generation.....	16
4.2 Expansion and Compression audio.....	16
4.3 Increase or Decrease the speed audio. ....	16
4.4 Full control of the audio. ....	17
4.5 Reverse audio .....	17
4.6 Skip audio .....	17
PROGRAM IDEA .....	19
CONCLUSION .....	20
REFERENCES.....	21

# INTRODUCTION

## 1.1 Introduction

In recent years, the use of audio files has become increasingly prevalent in a variety of industries, including music, education, and entertainment. As a result, the demand for software that can effectively manipulate audio files has grown significantly. The audio file player program is a tool that has been developed to meet this demand, offering a range of advanced features that allow for efficient and effective audio editing and playback.

## 1.2 Motivations

We searched a lot to make this program, and this one was very fantastic, easy, and quite simple and easy to use, that is the thing that motivated us to work on this program also It was the solution to a problem we thought about.

## 1.3 Project objective

The objective of this project is to develop an audio file player program that offers a range of advanced features for efficient and effective audio editing and playback. The program will include the ability to change the sound of audio files, adjusting the pitch, tempo, and other aspects of the audio as needed. Additionally, the program will feature the ability to speed up or slow down the sound for efficient learning and analysis.

Furthermore, the program will include sound compression and expansion features, providing a greater range of dynamic expression for musicians and other audio professionals. Finally, the program will generate white noise to create a soothing ambiance for the listener, ideal for meditation or relaxation.

The program will be user-friendly and easy to navigate, with intuitive controls for all its features. The program will also be designed to work with a wide range of audio file formats, ensuring maximum compatibility with users' existing audio libraries.

Overall, the objective of this project is to create an audio file player program that provides a comprehensive range of advanced features for efficient and effective audio editing and playback, serving the needs of musicians, audio engineers, language learners, and anyone else who works with audio files.

## 1.4 Program Layout

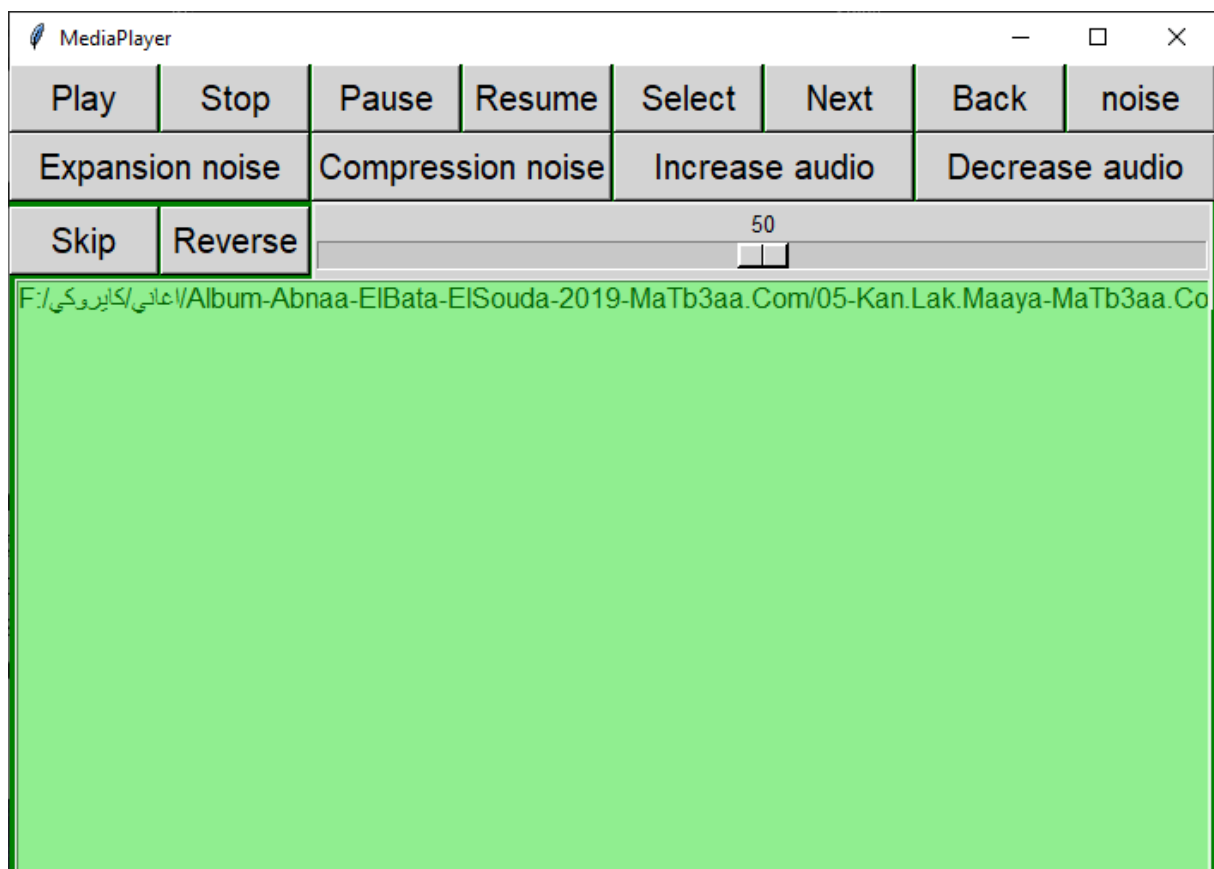


Figure 1.4.1: Layout program

## **1.5 Report outline**

### **1.5.1 Project Libraries**

What libraries are used and what are their benefits in the program?

### **1.5.2 The method of work**

The way the program works and the best way to use it to get the best performance.

### **1.5.3 Advantages**

Features of the program.

### **1.5.4 Program idea**

The idea of the work of the program in terms of use and method of work.



# PROJECT LIBRARIES

## 2.1 Tinter

Tkinter is a Python binding to the Tk GUI toolkit. It is the standard Python interface to the Tk GUI toolkit and is Python's de facto standard GUI. Tkinter is included with standard Linux, Microsoft Windows, and macOS installs of Python, the name Tkinter comes from Tk interface. Tkinter was written by Steen Lumholt and Guido van Rossum, then later revised by Fredrik Lundh. Tkinter is free software released under a Python license.

### 2.1.1 Simple application

```
import tkinter as tk

class Application(tk.Frame):

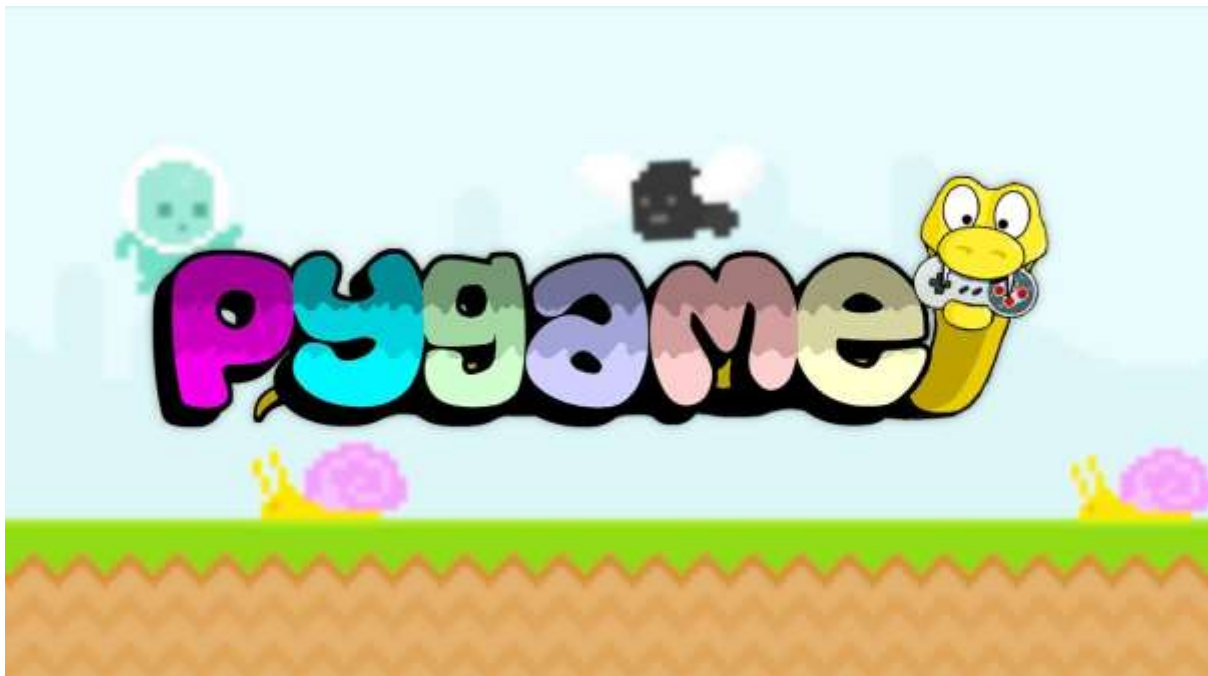
    def __init__(self, master=None):
        tk.Frame.__init__(self, master)
        self.grid()
        self.createWidgets()

app = Application()
app.master.title('Sample application')
app.mainloop()
```

## 2.2 Pygame

Pygame is a cross-platform set of Python modules designed for writing video games. It includes computer graphics and sound libraries designed to be used with the Python programming language.

Pygame was originally written by Pete Shinnars to replace PySDL after its development stalled. It has been a community project since 2000 and is released under the free software GNU Lesser General Public License (which "provides for Pygame to be distributed with open source and commercial software").



*Figure 2.2.1: Pygame*

## 2.3 Librosa

Librosa is a valuable Python music and sound investigation library that helps programming designers to fabricate applications for working with sound and music document designs utilizing Python. This Python bundle for music and sound examination is utilized when we work with sound information, like in the music age (utilizing Lstm's), Automatic Speech Recognition.

The library is not difficult to utilize and can deal with fundamental as well as cutting-edge errands connected with sound and music handling. It is open source and uninhibitedly accessible under the ISC License.

The library upholds a few elements connected with sound records handling and extraction like burden sound from a circle, register of different spectrogram portrayals, symphonious percussive source detachment, conventional spectrogram decay, stacks and translates the sound, Time-space sound handling, successive demonstrating, coordinating consonant percussive partition, beat-simultaneous and some more.



Figure 2.3.1: Librosa Python Library for Audio and Music Analysis

## 2.4 Numpy

NumPy (pronounced/'nʌmpaɪ/ (NUM-py) or sometimes /'nʌmpi/=(NUM-pee)) is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays. The predecessor of NumPy, Numeric, was originally created by Jim Hugunin with contributions from several other developers. In 2005, Travis Oliphant created NumPy by incorporating features of the competing Numarray into Numeric, with extensive modifications. NumPy is open-source software and has many contributors. NumPy is a NumFOCUS fiscally sponsored project.



Figure 2.4.1 NumPy

## 2.5 Sound files

The `soundfile` module can read and write sound files. File reading/writing is supported through `libsndfile`, which is a free, cross-platform, open-source (LGPL) library for reading and writing many different sampled sound file formats that runs on many platforms including Windows, OS X, and Unix. It is accessed through CFFI, which is a foreign function interface for Python calling C code. CFFI is supported for CPython 2.6+, 3.x and PyPy 2.0+. The `soundfile` module represents audio data as NumPy arrays.



*Figure 2.5.1: Soundfile*

## 2.6 Mixer

The Mixer is a helper to generate instances of Django or SQLAlchemy models. It is useful for testing and fixture replacement. Fast and convenient test-data generation.

# THE METHOD OF WORK

## 3.1 How Program Works

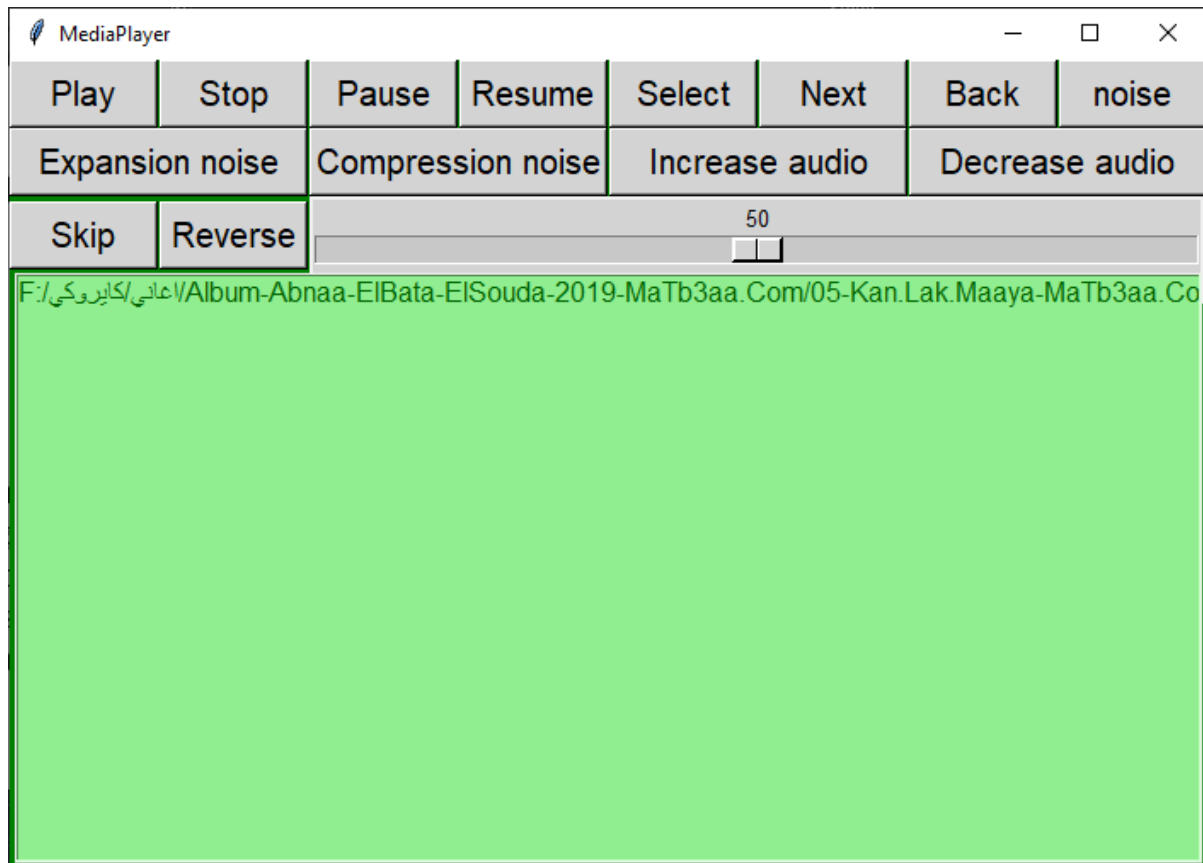


Figure 3.1.1: Layout program

A media player program works by using software to interpret digital audio and display them on a computer device. When a user selects a file to play, the media player program reads the file and decodes it into a format that can be played back.

The media player program typically includes a user interface that allows users to control playback and adjust various settings. This can include features like adjusting the volume and selecting different playback modes such as repeat.

In addition, a media player program may offer additional features such as the ability to create playlists, display lyrics, and access online music. some media player programs also include advanced features such as

equalizers, sound effects, and visualizers to enhance the listening experience.

The media player program uses a combination of software algorithms and codecs to decode and playback Mp3 type of digital media files. Codecs are used to compress and decompress digital media files, making them easier to store and transmit. The media player program needs to have the appropriate codec installed on the computer or device to play back the file.

Overall, the media player program is a software application that allows users to play back digital media files on a computer, with a range of features and settings to enhance the listening experience.

# ADVANTAGES

## 4.1 White noise generation.

The program offers a white noise feature that can create a distorted and blurred sound effect, like the sound of rain. While this feature can be useful for creating a relaxing atmosphere or for masking distracting sounds in the environment, it may also make it difficult to hear voices or distinguish strange sounds in the background. As such, users should be aware of this effect and use it judiciously. Overall, the white noise feature provides an additional level of customization for users, but it should be used appropriately to avoid any negative impact on the audio playback.

## 4.2 Expansion and Compression audio.

This program offers the ability to compress and expand audio, allowing users to modify the sound characteristics from a thin or compact sound to a larger or more spacious sound, and vice versa. This feature can be particularly useful for music producers or sound engineers who want to adjust the tonal balance and dynamic range of their audio recordings. With this tool, users can fine-tune the sound to their liking. Overall, the audio compression and expansion feature are a powerful tool that can help users achieve their desired sound output.

## 4.3 Increase or Decrease the speed audio.

This program provides professional-level functionality to increase or decrease the audio speed, with a range of speed options from x0.5 to x2. This feature can be useful for a variety of applications, such as adjusting the tempo of music recordings, slowing down speech for language learning or transcription, or speeding up narration for time-lapse videos. The ability to fine-tune the audio speed can also help users save time when listening to longer audio recordings, by increasing the playback speed without sacrificing intelligibility. Overall, the audio



speed control feature is a versatile tool that can be customized to meet the specific needs of users.

#### 4.4 Full control of the audio.

This program provides comprehensive sound control, enabling users to stop and start playback, adjust volume, and navigate seamlessly between audio files via a user-friendly list interface. With these features, users can tailor their listening experience to their liking, effortlessly managing their media library with ease. The program's intuitive design facilitates ease of use, allowing users to access and customize their audio content with minimal effort. Overall, this program offers a powerful set of sound control tools that are both flexible and accessible, enhancing the overall user experience.

#### 4.5 Reverse audio

**A reverse function** in a media player audio allows users to play the audio content in reverse, essentially reversing the playback direction. While implementing a reverse function can be technically challenging, especially for real-time audio playback, I'll outline a high-level approach to achieving this functionality:

1. **Audio Processing:** Reverse playback involves manipulating the audio data to reverse the playback direction. To achieve this, you'll need to process the audio samples in reverse order.
2. **Audio Buffering:** To facilitate reverse playback, you'll need to buffer a certain portion of the audio data to ensure smooth and uninterrupted playback. The size of the buffer will depend on the specific requirements of your media player.

#### 4.6 Skip audio

**A skip function** in a media player audio allows users to quickly jump forward or backward within a track or playlist. It provides convenience and flexibility for users to navigate through the audio content. Here's how you can implement a skip function in a media player:

1. **Determine the Skip Duration:** Decide on the duration of each skip. It could be a fixed value, such as 10 seconds, or you can provide options for users to choose from, like 5 seconds, 10 seconds, or 30 seconds.
2. **Design User Interface Controls:** Create user interface controls that enable users to initiate the skip function. This can be done through buttons, icons, or gestures. For example, you can have dedicated skip forward and skip backward buttons or use swipe gestures on the playback screen.

# PROGRAM IDEA

a media player is an innovative media player program for Windows that offers a range of advanced features to enhance the listening and viewing experience. With support for a wide range of audio and video file formats, users can enjoy their favorite songs and videos in high quality. A media player allows users to adjust the playback speed of media files, apply various sound effects and enhancements, and even add white noise or other background sounds to create a relaxing atmosphere. The program also offers features for creating and managing playlists of songs and videos, displaying lyrics in real time. Overall, a media player provides a comprehensive set of features that make it easy to fine-tune the playback of media files and discover recent music and videos in a personalized and enjoyable way.

# CONCLUSION

In conclusion, a media player is an innovative media player program for Windows that offers a range of advanced features to enhance the listening experience. With support for a wide range of audio Mp3 file formats, users can enjoy their favorite media files in high quality. The program's ability to adjust playback speed, apply sound effects and enhancements, and even add white noise or other background sounds makes it a standout among media player programs. Additionally, the program's features for creating and managing playlists, and displaying lyrics in real-time. The customizable user interface also allows users to personalize the app to their liking. Overall, media player is a versatile and innovative media player program that provides an exceptional experience for Windows users.

## REFERENCES

- [1] Python interface to Tcl/Tk, [tkinter — Python interface to Tcl/Tk — Python 3.11.3 documentation](#)
- [2] Librosa, [librosa — librosa 0.10.0 documentation](#)
- [3] Mixer, [mixer · PyPI](#)
- [4] Soundfile, [soundfile · PyPI](#)
- [5] NumPy, [NumPy Website](#)
- [6] NumPy Website, [NumPy](#)
- [7] Tkinter, [Tkinter official Website](#)
- [8] Librosa Library in Python, [Librosa Library in Python - Javatpoint](#)
- [9] Pygame ,softwareforcodeing.com, [Pygame](#)
- [10] Pygame Website, [www.pygame.org](#)
- [11] Sliders in Tkinter, [Sliders in Tkinter | Tkinter | python-course.eu](#)
- [12] pypi, [Pygame · PyPI](#)