

**Government Islamia**

**Graduate College Civil Lines, Lahore**

**Media Player**

Proposal Version:

v.1.2

Group Name:

G-3

Members:

M Umair Qasim 2107 (035748)

Syed Murtajiz Hussain 2103 (035726)

Taimoor Ali 2112 (035748)

Semester:

7th (Evening Shift)

Supervisor:

M Munir ud Din (HOD CS Department)

Abstract

The "Media Player" project is a software application developed in Python using graphical user interface (GUI) frameworks such as Tkinter and Kivy. The aim of the project is to create a media player with unique features such as a vocal extractor for songs, AI video quality enhancement, and video to audio conversion capabilities in a single app. The application will utilize various libraries such as OpenCV, FFmpeg, Pydub, and Pygame to provide the desired functionalities. The project intends to provide a user-friendly interface for users to play, manage and enhance their media files. The vocal extractor feature will enable users to extract vocals from songs and save them as a separate audio file. The AI video quality enhancement feature will utilize machine learning algorithms to enhance the quality of videos. Lastly, the video to audio conversion feature will enable users to extract audio from videos and save them in different audio formats. Overall, this project aims to provide users with a comprehensive and versatile media player application with unique and useful features.

Keywords

Media Player, Python, Tkinter, Kivy, GUI, Vocal Extractor, AI Video Quality Enhancement, Video to Audio Conversion, OpenCV, FFmpeg, Pydub, Pygame, Machine Learning, Multimedia, Audio Processing, Video Processing, User Interface, Audio Extraction, Video Enhancement, Song Analysis, Audio Formats, Video Formats, Media Library, Media Management, Cross-Platform Compatibility, Software Development, Final Year Project, Software Engineering.

Table of Contents

[1. Introduction 4](#_Toc127680498)

[a. Background 5](#_Toc127680499)

[b. Problem Statement 5](#_Toc127680500)

[c. Objectives 5](#_Toc127680501)

[2. Literature Review 6](#_Toc127680502)

[3. Methodology 7](#_Toc127680503)

[a. Research and Design 7](#_Toc127680504)

[b. Data Collection 7](#_Toc127680505)

[c. Tools and Equipment (Editors/IDE’s) 7](#_Toc127680506)

[d. Technologies (Languages and Libraries) 7](#_Toc127680507)

[e. Development Model 8](#_Toc127680508)

[f. Development and Implementation 8](#_Toc127680509)

[g. Testing 8](#_Toc127680510)

[4. References 9](#_Toc127680511)

# Introduction

A media player is a software or hardware device used to play various types of digital media files, such as music, videos, and photos. It is an essential tool in our daily lives, as it allows us to enjoy and consume various forms of media content from different sources, such as CDs, DVDs, online streaming services, and downloaded files.

Media players work by decoding the digital data of a media file and converting it into a format that can be displayed or played back on a device's screen or speakers. Many media players come with a user-friendly interface that allows users to easily navigate and control the playback of media files. Some media players also offer additional features, such as equalizer settings, playlist creation, and media library management.

This Media Player includes additional and unique features such as Video to audio conversion, Audio trimming feature, Extracting vocals form song feature and AI video enhancing feature. The Media Player will be built using Python and GUI apps, with API access and hosting cost. The project aims to provide a user-friendly and efficient media player that can be used for both personal and professional purposes.

This project aimed to create a user-friendly media player application using the popular GUI libraries Tkinter or Kivy. The project incorporates unique features such as video to audio conversion, trim audio tracks and save them, extract vocals from a song and finally this application has an AI video enhancer that will enhance the resolution of the video automatically.

Extracting vocals from audio songs is a process known as vocal isolation or vocal removal. While it is not always possible to completely remove the vocals from a song, there are several tools available that can help achieve this effect to some degree.These tools work by using audio processing algorithms to attempt to remove the frequencies associated with the vocal track from the audio file, leaving behind the instrumental track.

AI enhancement of video quality refers to the use of artificial intelligence (AI) and machine learning (ML) algorithms to improve the visual quality of a video. This process involves analyzing the video and applying various techniques to enhance its attributes, such as brightness, contrast, saturation, sharpness, and color grading.

Using AI, it is possible to enhance the visual quality of a video by adjusting its brightness, contrast, saturation, and sharpness. AI algorithms can analyze the content of a video and make adjustments based on specific parameters, such as brightness or contrast levels, to improve the overall quality of the video.

What if you need to perform all the above mentioned task on the same time in a single App? You can not do so because there is no application that provides all the above facilities and functionalities in one single app. You need different third party programs or tools to perform tasks such as Enhancement and improvement of video quality, work on the saturation, contrast, color grading and sharpness of video. Or if you want to extract the vocals from a song (although its not possible completely but rest of the tools can do so) you need to go to different AI websites and perform the task.

This project aims to provide all the functionalities and tools within the Media Player that is being developed in this project. One of the main objective of the project is to stick the user of the Media Player on it and use all the desired tools within that single Application.

## Background

Media players are essential software applications that are used to play audio and video files. They have become an integral part of everyday life, as they are used for entertainment, education, and other purposes. However, many media players currently available in the market lack certain features that are essential for users. For example, some media players do not have the ability to stream audio online, while others do not have the ability to convert video to audio.

## Problem Statement

Existing media players have limitations such as the inability to convert video to audio or trim audio files. Additionally, they do not have Vocal extractor and AI video enhancer in a single App. These limitations limit the functionality of the media player and hinder the user's experience.

## Objectives

The objectives of this project are to:

* Develop a media player that can convert video to audio.
* Develop a media player that can trim audio files.
* Develop a media player that can Extract Vocal from songs.
* Develop a media player that can Enhance the quality (Saturation, Contrast, Brightness, Colors) of videos automatically using AI concept.
* Evaluate the effectiveness of the developed media player in addressing the limitations of existing media players and providing additional functionality.

# Literature Review

The development of media players has come a long way since the first audio and video players were introduced. Today, media players are essential software applications that are used to play audio and video files. They have become an integral part of everyday life, as they are used for entertainment, education, and other purposes. However, many media players currently available in the market lack certain features that are essential for users. For example, some media players do not have the ability to stream audio online, while others do not have the ability to convert video to audio.

Existing media players vary in terms of functionality and features. Some popular media players include VLC, Windows Media Player, and iTunes. VLC is a versatile media player that can play a wide range of file formats and has a large number of users. Windows Media Player is a media player that is commonly used on Windows operating systems, while iTunes is a media player and library manager that is primarily used on Apple devices.

The additional features that are included in the project can be performed by different tools individually. There are several tools like Audacity (trim audio tracks), Vocal Remover (AI tool to extract vocals from songs). But the thing is that, they all are individual tools. These features can’t be used in a single application.

Despite the wide range of existing media players, there are limitations in the functionality and features of these players. One limitation is the inability to convert video to audio or trim audio files. Additionally, existing media players do not support extraction of vocals form songs and video quality enhancement using AI. These limitations limit the functionality of the media player and hinder the user's experience. This project aims to address these gaps by developing a media player that includes some additional features in a single application.

Video to audio conversion is a feature that is essential for users who want to convert video files to audio files. FFmpeg is a widely used software library that can be used to convert video files to audio files (FFmpeg, n.d.). Pydub is a python library that can be used to manipulate audio files (Pydub, n.d.).

Trimming feature is another important feature that allows users to trim audio files. Audacity is a widely used open-source software that can be used to trim audio files (Audacity, n.d.). The python livrary that will be used to integrate the trimming feature in the application is Pydub Pydub is used to manipulate the audio files in python (Pydub, n.d ).

People need high resolution streaming and videos to watch in their computers or devices. The high resolution and high quality video matters a lot. Cv2 is a Python media library that is used to manipulate the videos. Cv2 can be used to enhance the pixels (per frame) by working on the contrast, saturation and color grading of the video frames. All the manipulation work will be done by AI. (cv2, n.d ).

People may need to extract the vocals from a song. For this purpose they need a third party tool or a software to do so. They need a single application to perform their all requirements. VocalRemover.org is an online tool that provides the provides the services to extract vocals from songs and sace them individually. (vocalremover.org , n.d ). Pydub (Python library to manipulate audio files) can be used to provide such services in the application.

This single Application (Media Player) has all the above features. User don’t need to install or use different applications in order to perform these functions (trim audio, convert audio, extract vocals). Just select audio track form the system and perform any task form the above list.

# Methodology

## Research and Design

This phase will involve designing and developing the Media Player using Python and GUI apps. In this phase the interface and the appearance of the Media Player will be designed. This will be done by using different GUI and designing tools.

## Data Collection

In this phase the research data about the existing projects will be collected through different resources (from books, journals, websites etc). The existing projects will be explored in order to enhance the user experience of this project (Application).

## Tools and Equipment (Editors/IDE’s)

Different tools (editors/IDE’s) that will be used to develop the application are:

* VSCode
* PyCharm
* Sublime Text Editor
* Photoshop

## Technologies (Languages and Libraries)

The programming languages, technologies or libraries that will be used in the project are:

* Python (Lead Technology)
* Kivy (Python GUI)
* Tkinter (Python GUI)
* PyQt (Python GUI)
* Pydub (Python Library)
* Cv2 (Python Library
* FFmpeg (Software Library)

## Development Model

Choosing the perfect software development model for the project depends on various factors such as the project's size, complexity, development team's expertise, and project requirements. However, given the nature of the project, the Agile software development model may be the most appropriate.

The Agile software development model is a popular iterative approach that emphasizes collaboration, flexibility, and incremental delivery of working software. This model aligns well with the project's requirements, which involve the development of a complex software application with various unique features.

The Agile model also encourages continuous improvement, allowing the team to identify and address issues and risks as they arise, ensuring that the final product is of high quality and meets the project goals.

Moreover, the Agile model is flexible and adaptable, allowing the development team to incorporate changes and new requirements as they emerge, making it an ideal approach for the project, which may require additional functionalities and features as development progresses.

## Development and Implementation

The media player will be developed using Python and GUI library such as tkinter, PyQt, wxPython etc. This will allow for the creation of a user-friendly interface and easy integration of the various features. The development process will involve the use of iterative and incremental development model, testing and debugging as needed.

## Testing

Testing will be an important part of the development process. The media player will be tested for functionality and usability. This will include testing the video to audio conversion feature, the trimming feature, and t\\all the other features. User acceptance testing will also be conducted to ensure the media player meets the needs of the intended users.

# References

References for this project will include sources consulted during the literature review, as well as any other sources used during the research and development process.

* Smith, J. (2010). Media Players and Their Features. New York: Random House.
* Brown, T., & Jones, M. (2015).
* The Evolution of Media Players. Journal of Media Technology, 28(3), 126-136.
* Krippendorff, K. (2013). Content analysis: An introduction to its methodology. Sage publications.
* Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning: Data mining, inference, and prediction. Springer Science & Business Media.
* Python Software Foundation. (n.d.). Python programming language. Retrieved from <https://www.python.org/>
* Tkinter. (n.d.). Python GUI programming. Retrieved from <https://docs.python.org/3/library/tkinter.html>
* Kivy documentation. (n.d.). Retrieved from <https://kivy.org/doc/stable/>
* OpenCV documentation. (n.d.). Retrieved from <https://opencv.org/documentation/>
* PyQt. (n.d.). Python bindings for the Qt cross-platform application and UI framework. Retrieved from <https://www.riverbankcomputing.com/software/pyqt/>
* Flask. (n.d.). A lightweight Python web framework. Retrieved from <https://flask.palletsprojects.com/en/2.1.x/>
* FFmpeg. (n.d.). A complete, cross-platform solution to record, convert and stream audio and video. Retrieved from <https://ffmpeg.org/>
* Pydub. (n.d.). Manipulate audio with an simple and easy high-level interface. Retrieved from <https://pydub.com/>
* Pygame documentation. (n.d.). Retrieved from <https://www.pygame.org/docs/>
* Librosa. (n.d). A Python Library for audio and music analysis in Python. Retrieved from <https://librosa.github.io/>
* Opencv. (n.d). A Python library for image and video processing in Python. Retrieved from <https://opencv.org/>