DEPARTMENT OF COMPUTER SCIENCE ENGINEERING

Rajiv Gandhi University of Knowledge Technologies - Nuzvid,

Eluru, Andhra Pradesh – 521202.

YOUTUBE AUDIO AND VIDEO DOWNLOADER

A Project Progress Report

Submitted in partial fulfillment for the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

Submitted by

J. PAVAN SANKAR N180109

Under the Esteem Guidance of

Mrs. N. SWATHI

Assistant professor in Department of Computer Science & Engineering

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING

Rajiv Gandhi University of Knowledge Technologies – Nuzvid,

Eluru, Andhra Pradesh – 521202.

CERTIFICATE OF COMPLETION

This is to certify that the work entitled, **"YOUTUBE AUDIO AND VIDEO DOWNLOADER"** is the bonafied work of **J. PAVAN SANKAR (ID No: N180109)** carried out under my guidance and supervision for the 3rd year summer project of Bachelor of Technology in the department of Computer Science and Engineering under RGUKT IIIT, Nuzvid. This work is done during the academic session May 2023 – June 2023, under our guidance.

……..

Mrs. N. Swathi
Assistant Professor,
Department of CSE,
RGUKT - Nuzvid.

…….

Dr. Sadu Chiranjeevi
Assistant Professor,
Head of the Department,
Department of CSE,
RGUKT - Nuzvid.

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING

Rajiv Gandhi University of Knowledge Technologies – Nuzvid,

Eluru, Andhra Pradesh – 521202.

CERTIFICATE OF EXAMINATION

This is to certify that the work entitled, **"YOUTUBE AUDIO AND VIDEO DOWNLOADER"** is the bonafied work of **J. PAVAN SANKAR (ID No: N180109)** and hereby accord our approval of it as a study carried out and presented in a manner required for its acceptance in the 3rd year of Bachelor of Technology for which it has been submitted. This approval does not necessarily endorse or accept every statement made, opinion expressed or conclusion drawn, as recorded in this thesis. It only signifies the acceptance of this thesis for the purpose for which it has been submitted.

……..……

Mrs. N. Swathi

Assistant Professor,

Department of CSE,

RGUKT - Nuzvid.

..……

Project Examiner

RGUKT - Nuzvid.

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING

Rajiv Gandhi University of Knowledge Technologies – Nuzvid,

Eluru, Andhra Pradesh – 521202.

DECLARATION

I **J. PAVAN SANKAR (ID No: N180109)** hereby declare that the project report entitled **"YOUTUBE AUDIO AND VIDEO DOWNLOADER"** done by me under the guidance of Mrs. N. Swathi, Assistant Professor is submitted for the fulfillment of summer project during the academic session May 2023 - June 2023 at RGUKT - Nuzvid.

I also declare that this project is a result of my own effort and has not been copied or imitated from any source. Citations from any websites are mentioned in the references. The results embodied in this project report have not been submitted to any other university or institute for the award of any degree or diploma.

Date: 17-08-2023

Place: Nuzvid

J. PAVAN SANKAR N180109

ACKNOWLEDGEMENT

I would like to express my profound gratitude and deep regards to my guide **Mrs. N. Swathi** mam for her exemplary guidance, monitoring and constant encouragement to us throughout the B. Tech course. I shall always cherish the time spent with her during the course of this work.

I am extremely grateful for the confidence bestowed in me and entrusting my project entitled **"YOUTUBE AUDIO AND VIDEO DOWNLOADER"**.

I express my gratitude to Dr. Sadu Chiranjeevi (HOD of CSE) and other faculty members for being a source of inspiration and constant encouragement which helped us in completing the project successfully.

Finally, yet importantly, I would like to express my heartfelt thanks to our beloved God and parents for their blessings, my friends for their wishes for the successful completion of this project.

TABLE OF CONTENTS

YOUTUBE AUDIO AND VIDEO DOWNLOADER

ABSTRACT

The aim of this project is to develop a desktop application capable of downloading both YouTube videos and audios. Additionally, it can efficiently download entire YouTube playlists without requiring manual intervention for each individual video or audio download. While several online tools exist for downloading YouTube audio and video, they often lack the ability to download playlists seamlessly. Similarly, certain online playlist downloaders necessitate manual downloading of each video from the playlist. In response, I have created an application that combines the strengths of both functionalities. Upon receiving a provided link, the 20 application initiates a verification process to determine whether the link corresponds to a playlist or a standalone video. It then presents distinct download options based on the media type. This approach streamlines the download process, enabling users to effortlessly save their desired content. Through this effort, I have successfully developed a comprehensive desktop application for downloading various media from YouTube.

Keywords: Youtube's audio and video download, streams, ffmpeg, transcode

1. INTRODUCTION

In today's world, YouTube has evolved into a ubiquitous source of entertainment, education, and knowledge for individuals across the globe. It offers the flexibility to enjoy a wide range of content whenever an internet connection is available. However, there is a growing need to download YouTube videos for future reference, especially in cases where internet access might not be readily available. While various online downloaders cater to this demand, they are not without their limitations. Regarding downloaders, platforms like y2mate[1], ymate[2] and some others excel at downloading individual YouTube audio and video content. However, they fall short when it comes to downloading playlists. YouTube 4K Downloader[3] shares some similarities with my project, offering the capability to download single YouTube videos and audio, as well as playlists. Nonetheless, this website imposes restrictions on the number of videos that can be downloaded from a playlist and necessitates manual interaction for each video

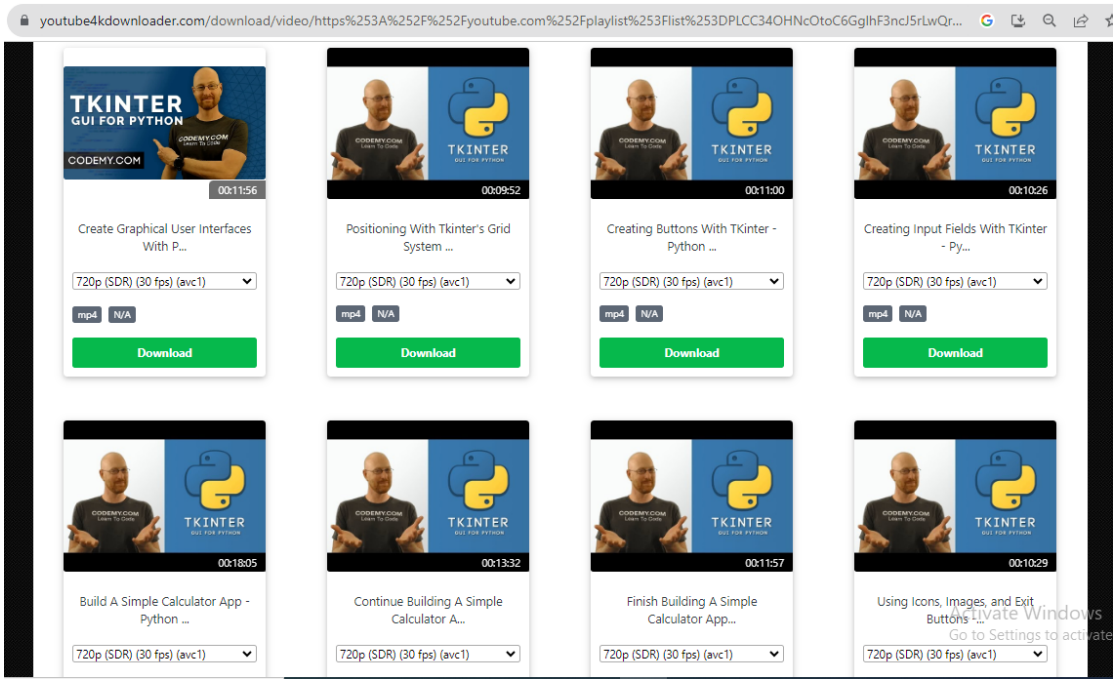
download, resulting in a time-intensive process. To bridge these gaps, I have developed the 'YouTube Audio and Video Downloader' application. This versatile tool efficiently downloads both audio and video content, as well as entire playlists. The application intelligently distinguishes between single videos and playlists based on the provided link. This functionality is made possible by leveraging the pytube module, which adeptly handles the downloading of videos and playlists from YouTube links. It offers access to available streams for a specific video, enabling users to select their preferred resolution for downloading. Additionally, the application supports audio downloads, albeit in a video container format. To address this, I implemented ffmpeg to seamlessly convert the downloaded audio into the widely compatible MP3 format. The user interface of the application is built using the tkinter and customtkinter libraries in Python, ensuring a seamless and engaging user experience. In conclusion, this application successfully addresses the challenge of downloading YouTube videos and audios, significantly streamlining the process of playlist downloads and enhancing overall user convenience.

2. EXISTING WORKS

YouTube 4K Downloader[3] can download audio, videos and also playlists

Limitations:

- Only limited number of videos can be downloaded but not the entire playlist
- We need to manually click on each video to download it
- Unwanted ads and pop-ups



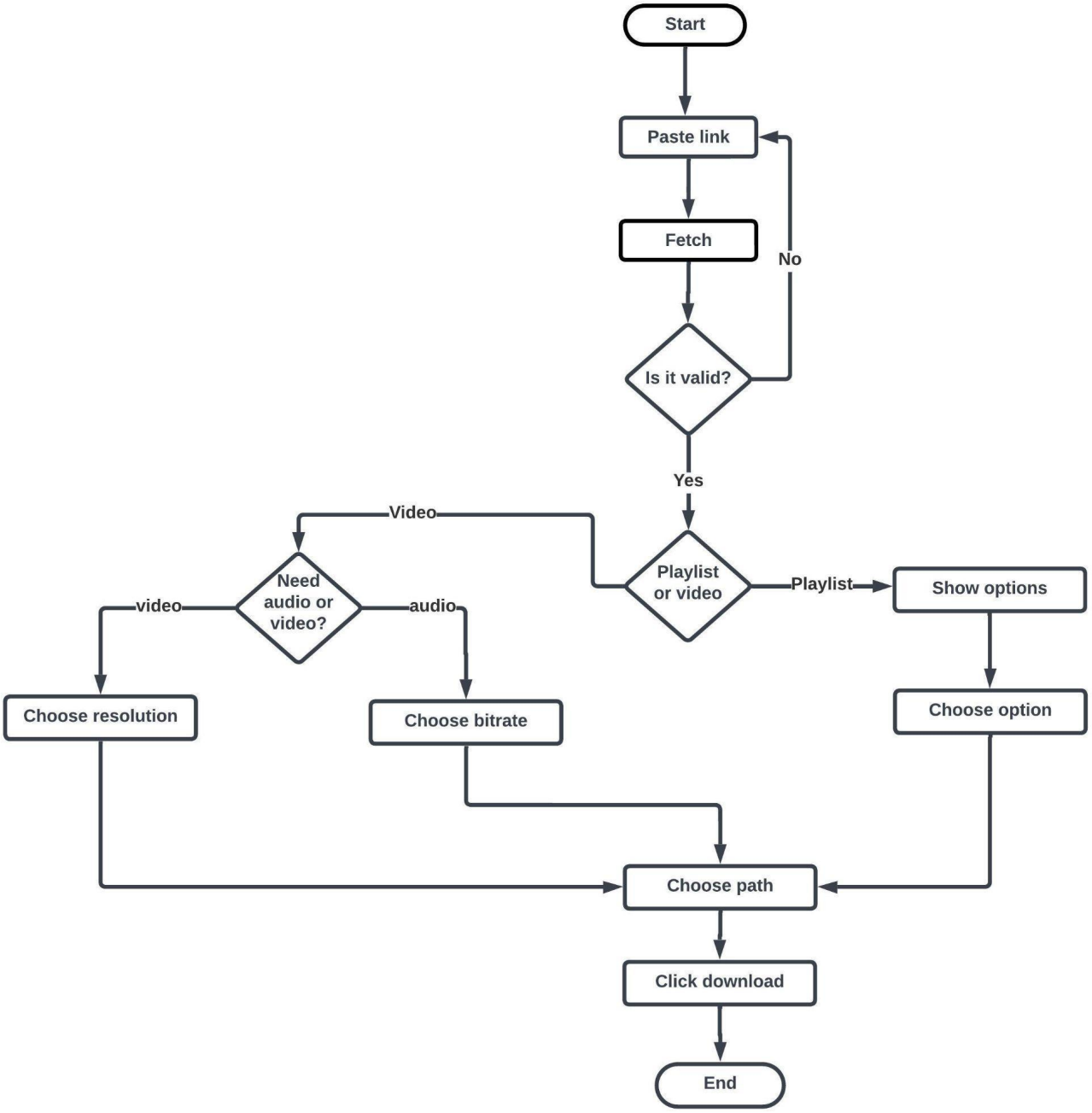
For a total of 225 videos it has given download option for 25 videos only

3. PROPOSED METHOD

I have proposed a desktop application designed to download audio or videos with customized bitrates and resolutions, catering to individual preferences and it also streamline the process of downloading YouTube playlists while minimizing user effort. Upon launching the application, a window appears, prompting the user to input the desired YouTube link for downloading audio or video content. After clicking the "Fetch" button, the application intelligently determines whether the input corresponds to a playlist or a single video. In cases where a single video is detected, the interface activates the audio and video buttons, granting access to a dropdown list of available bitrates and resolutions. Upon selecting the desired options, clicking the "Download" button initiates the download process, storing the video or audio in the specified file path. However, it's worth noting that the downloaded audio may initially be concealed, as it is acquired in the MP4 or webm container format commonly preferred by YouTube. Subsequently, the application employs the ffmpeg command-line tool to convert the audio into the universally supported MP3 format, making it visible and compatible with a wide array of systems. If the provided link leads to a playlist, the application presents users with various options. These options include the ability to download all high-resolution or low-resolution videos, exclusively download audio from the playlist, or analyze common video resolutions and audio bitrates shared among all playlist videos. Based on user preferences, the chosen settings are applied. Upon clicking the "Download" button, the application seamlessly downloads the complete playlist, saving the files in the designated path. This application effectively simplifies the process of downloading YouTube content, ensuring efficient playlist downloads and personalized audio/video downloads with minimal manual intervention.

3.1 Flowchart

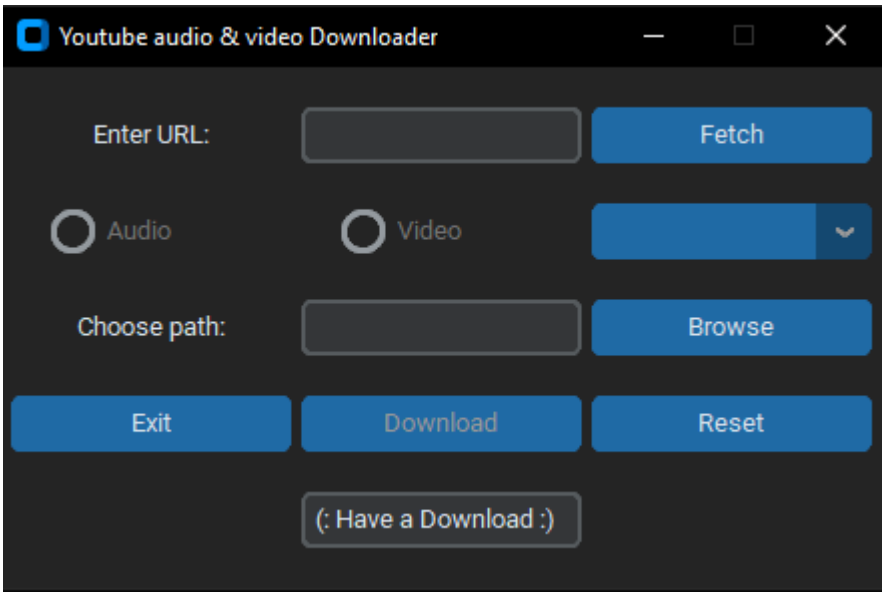
The following is the flowchart of my project.



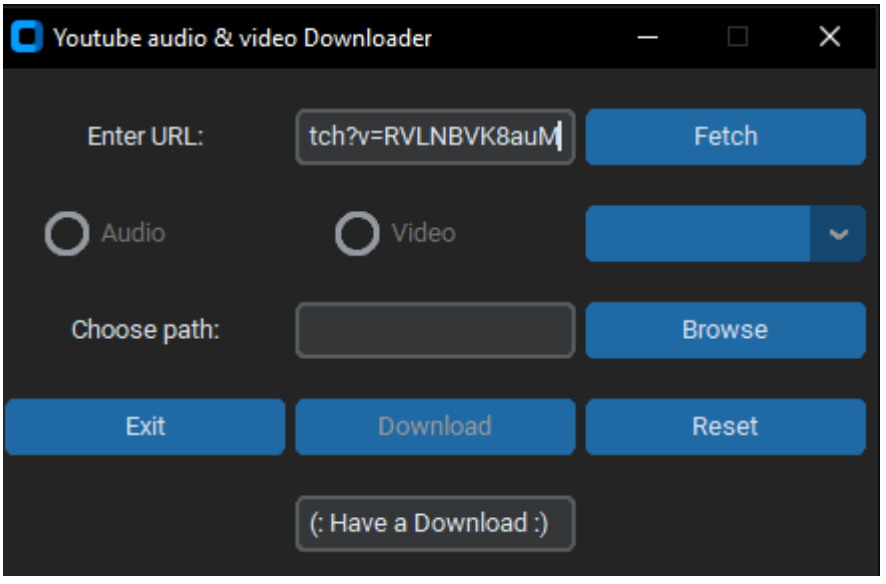
customtkinter:

In my project, I used CustomTkinter for the graphical user interface (GUI). CustomTkinter, a modified version of Tkinter, offers enhanced flexibility and control in designing the visual elements and layout of our GUI. With CustomTkinter, I have created interactive components such as widgets, windows, and top levels. This library has enabled me to deliver a visually appealing and user-friendly interface, enhancing the overall user experience.

Start application: Starting the application will display a window like below



Enter URL : Enter URL of the video you want to download

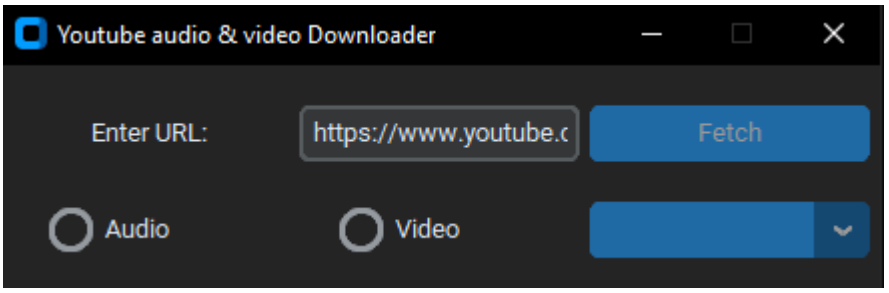


pytube:

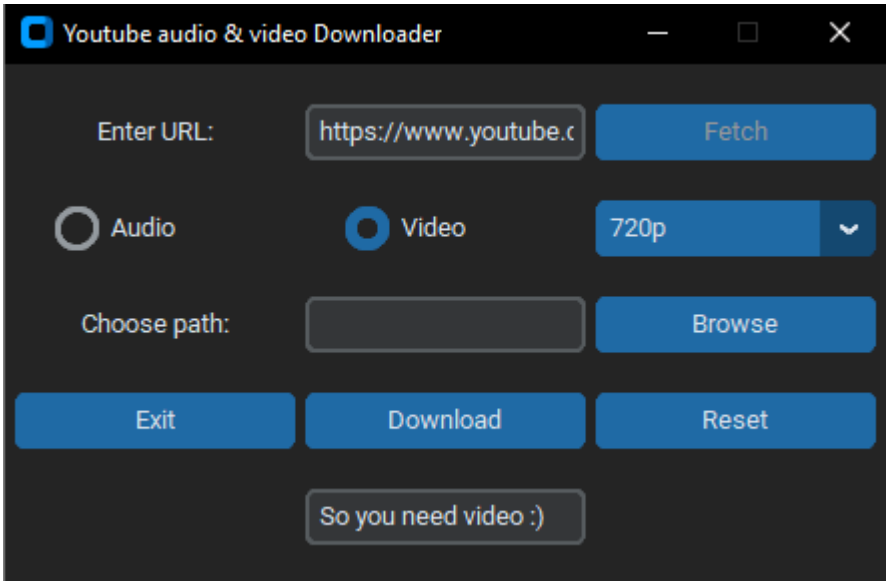
Pytube is a Python library that provides a convenient way to interact with YouTube videos. Pytube is particularly useful for tasks like downloading YouTube videos, extracting audio from videos, fetching video metadata, and more. We can specify the video URL, choose the desired video quality, and download the video to our local machine. Pytube allows us to retrieve detailed information about a YouTube video, such as its title, duration, views, likes, and more. It can also work with YouTube playlists, including downloading all videos in a playlist or fetching information about the videos in a playlist.

1. Giving youtube video link:

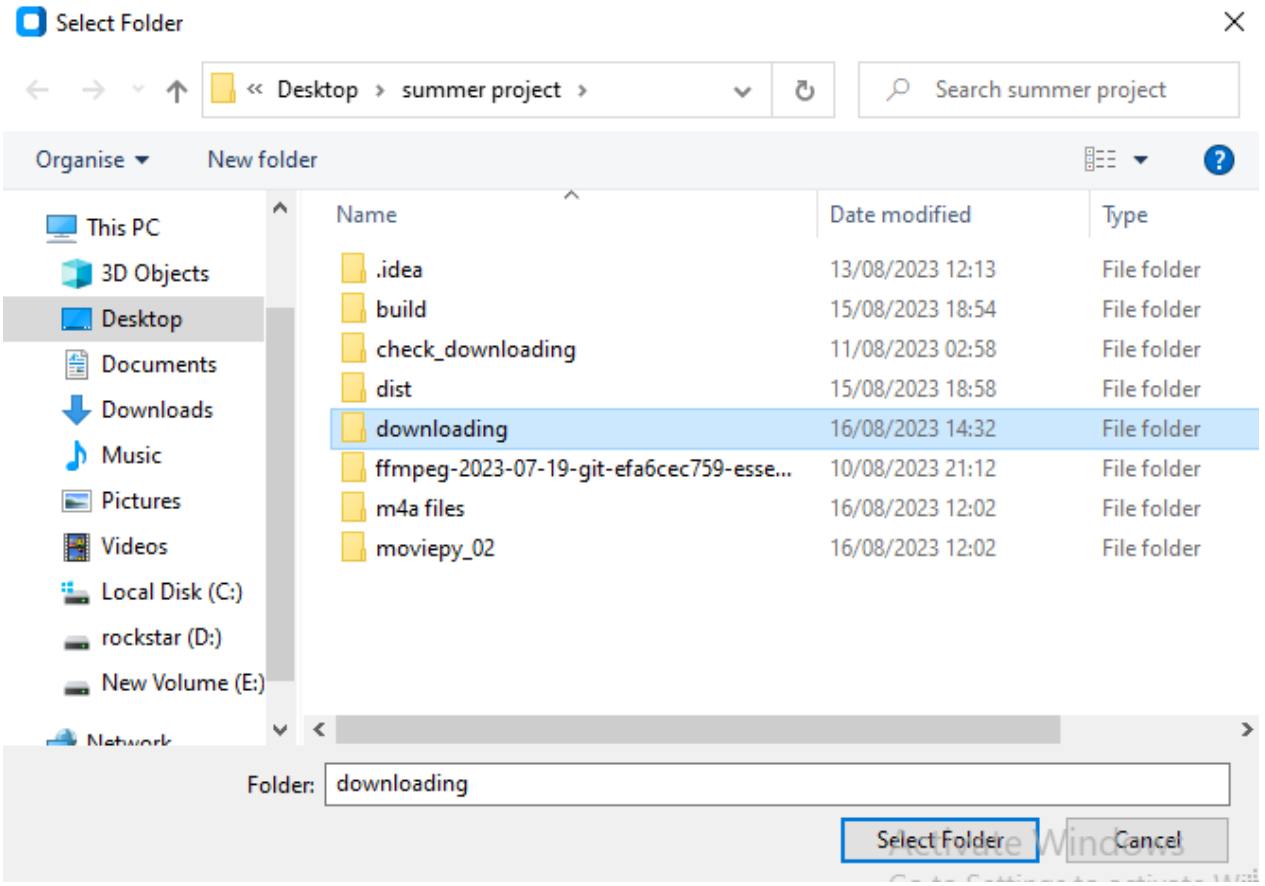
Fetching data: Clicking on fetch will retrieve the various streams available for a video



1.1 Choosing video: Selecting video and choosing the resolution



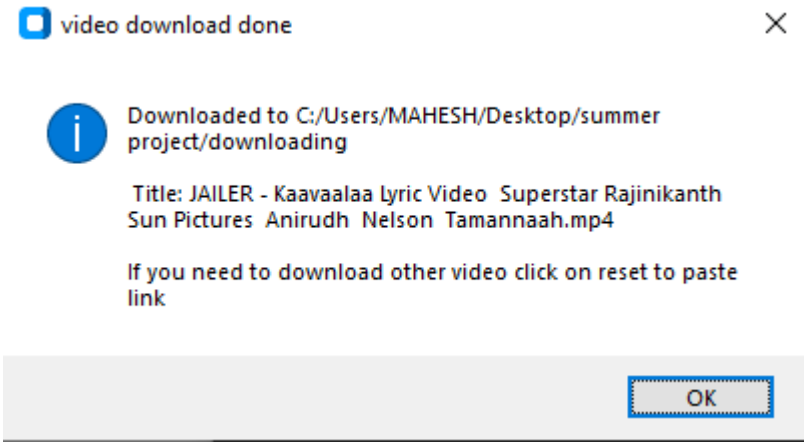
Choosing path: Clicked on browse to choose path

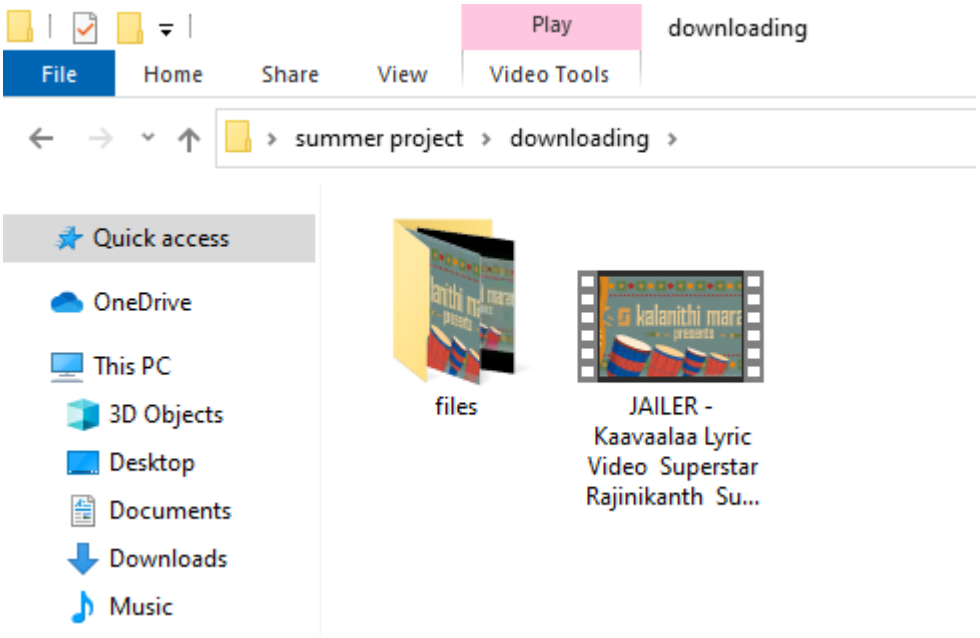


tkinter:

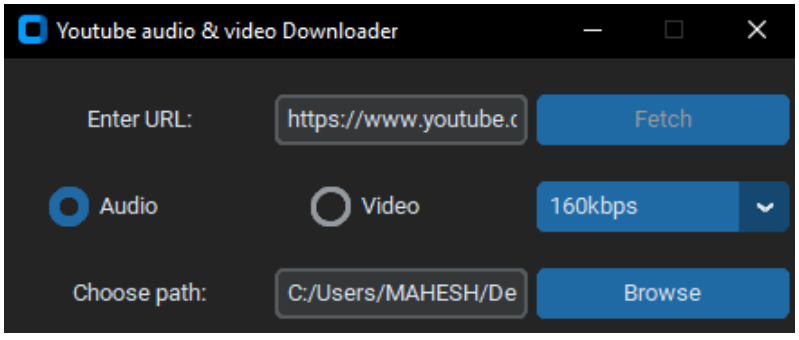
Tkinter is a built-in Python library used for creating graphical user interfaces (GUIs). It provides a set of widgets (buttons, labels, entry fields, etc.) and tools to develop desktop applications with interactive and visually appealing interfaces. Tkinter is cross-platform, meaning applications developed with Tkinter can run on various operating systems like Windows, macOS, and Linux. I used tkinter for notifications in my application.

Downloading: Clicking on download will download the video in the given path





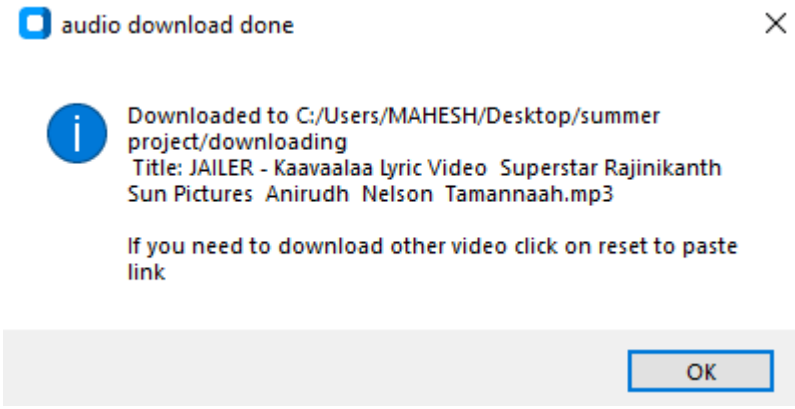
1.2 Choosing audio: Selecting audio and choosing the bitrate

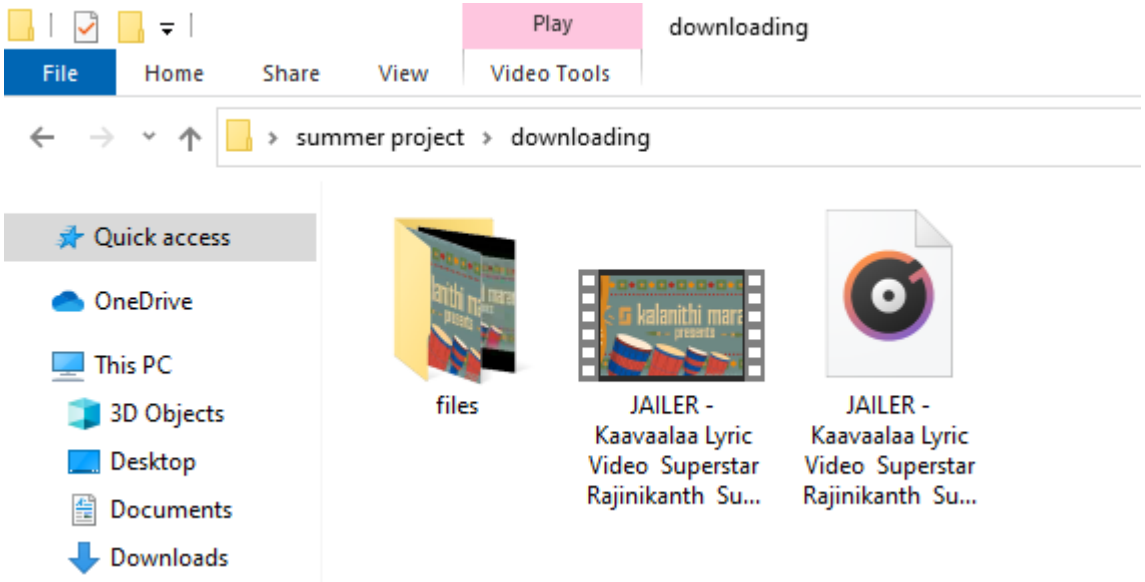


FFmpeg:

FFmpeg is a powerful multimedia framework that includes a command-line tool with a wide range of capabilities for working with audio and video files. The FFmpeg command-line tool allows you to perform various tasks related to audio and video processing, conversion, manipulation, and more.

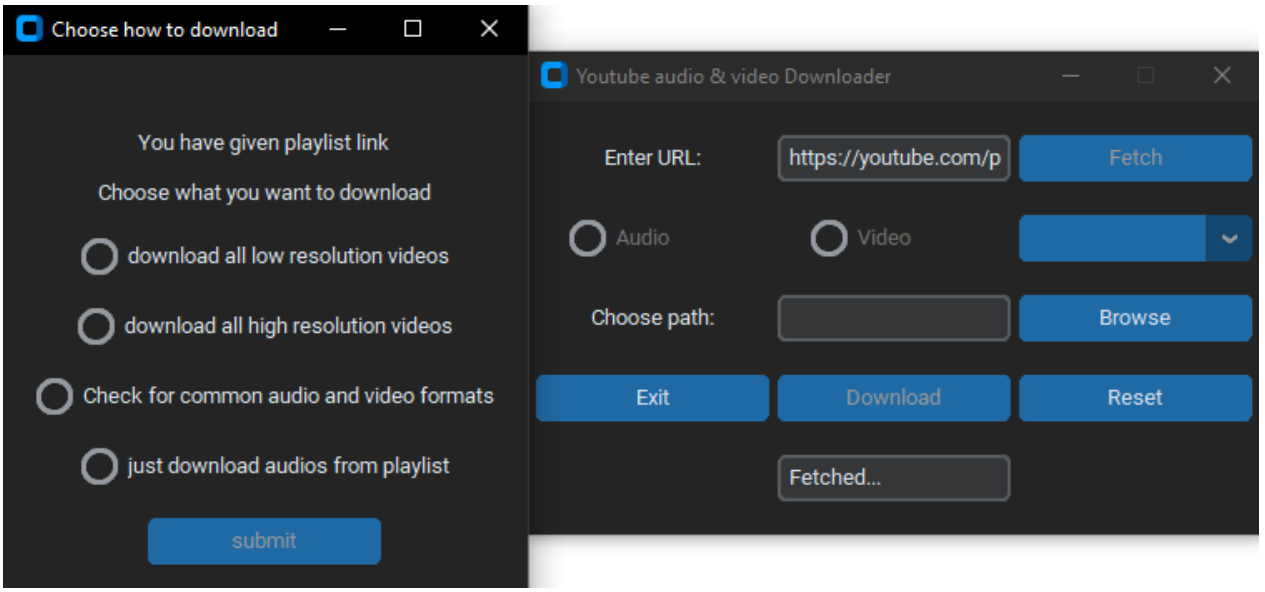
Downloading: Clicking on download will download the audio in the given path



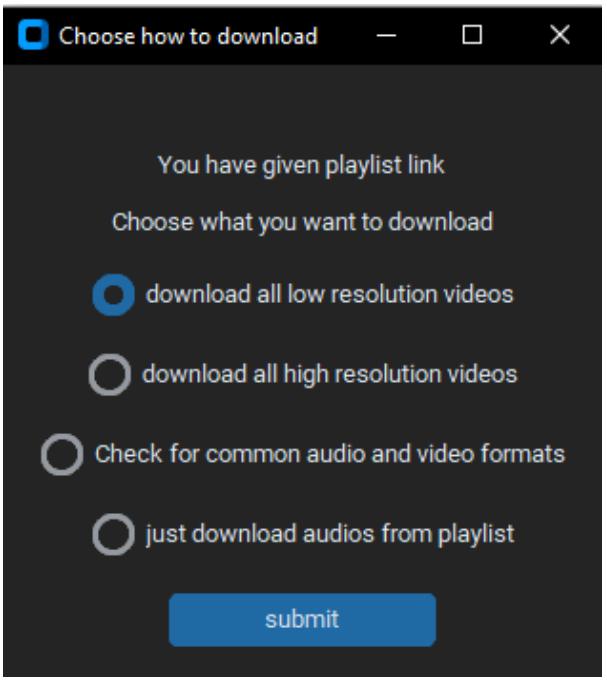


2. Giving playlist link:

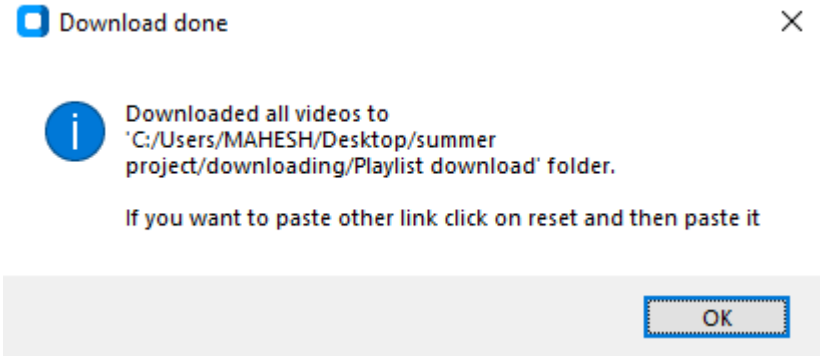
Fetching playlist data: Options shown for the given playlist

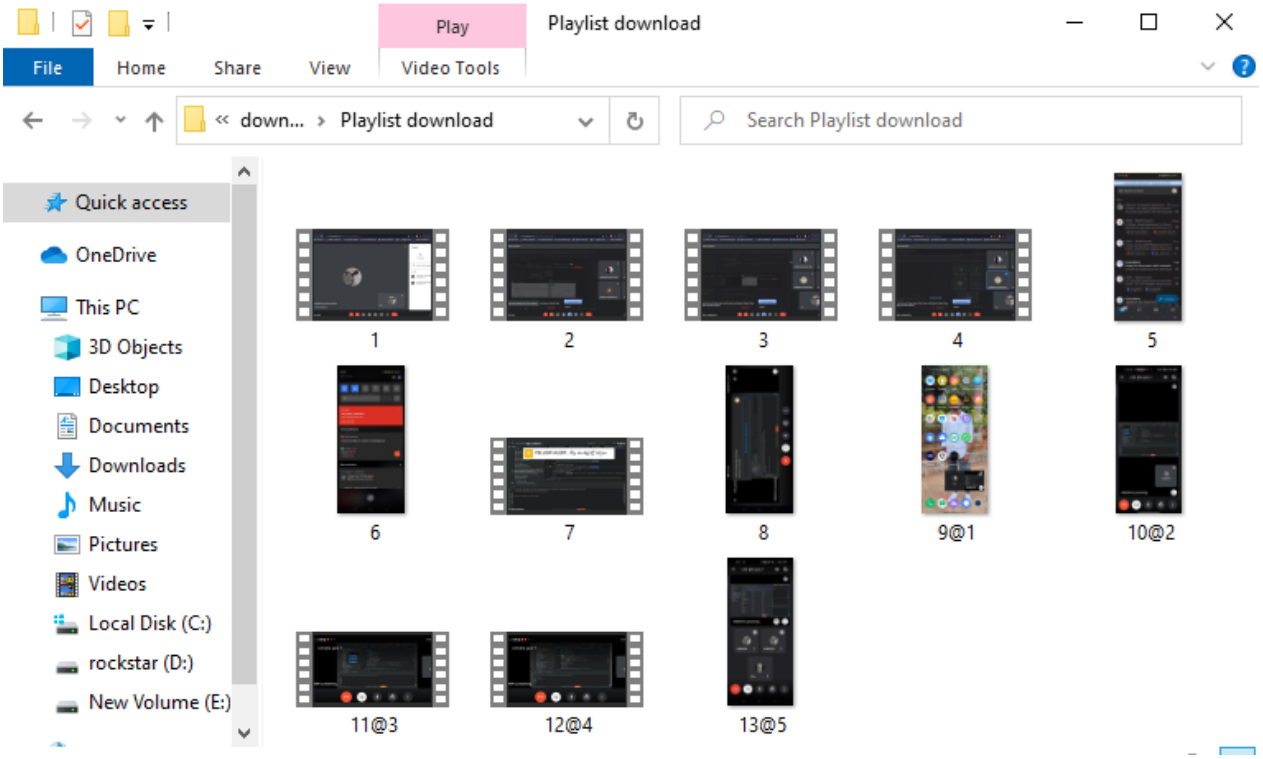


Choosing one option: Selecting one option among all



Downloading: Clicking on download will download the playlist in the given path





3.2 FFMPEG

ffmpeg is a powerful multimedia framework that includes a collection of libraries and tools for handling multimedia data, including audio and video conversion. When converting audio from one format to another (e.g., MP3) using ffmpeg, it typically follows these steps:

Audio Decoding: ffmpeg first decodes the audio data from the input file. It reads the encoded audio streams, extracts the audio data, and converts it into a raw format.
Audio Processing: ffmpeg applies any necessary processing on the audio data, such as resampling, audio filters, or channel manipulation, depending on the output format and desired settings.
Audio Encoding: After processing the audio, ffmpeg encodes the audio data into the output MP3 format. It compresses the raw audio data using the MP3 codec, which involves applying psychoacoustic models and bit-rate control to achieve the desired audio quality and file size.
Container Format: ffmpeg also takes care of wrapping the encoded audio data into the MP3 container format, which includes headers and metadata required to play the MP3 file correctly.

ffmpeg supports a wide range of audio codecs and formats, and the algorithms used for each depend on the specific codec's implementation.

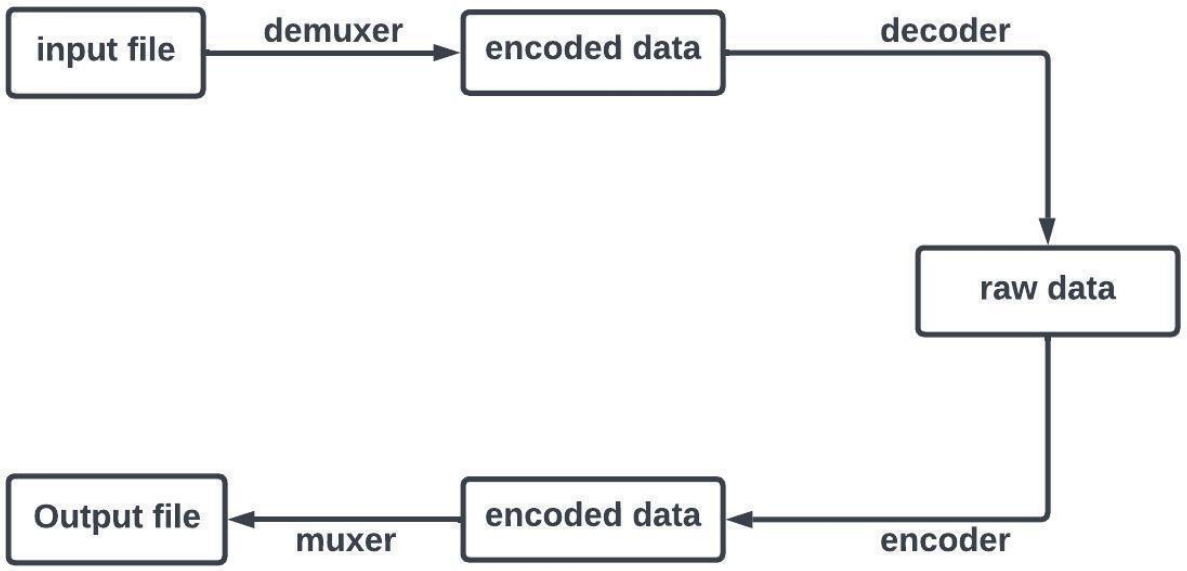
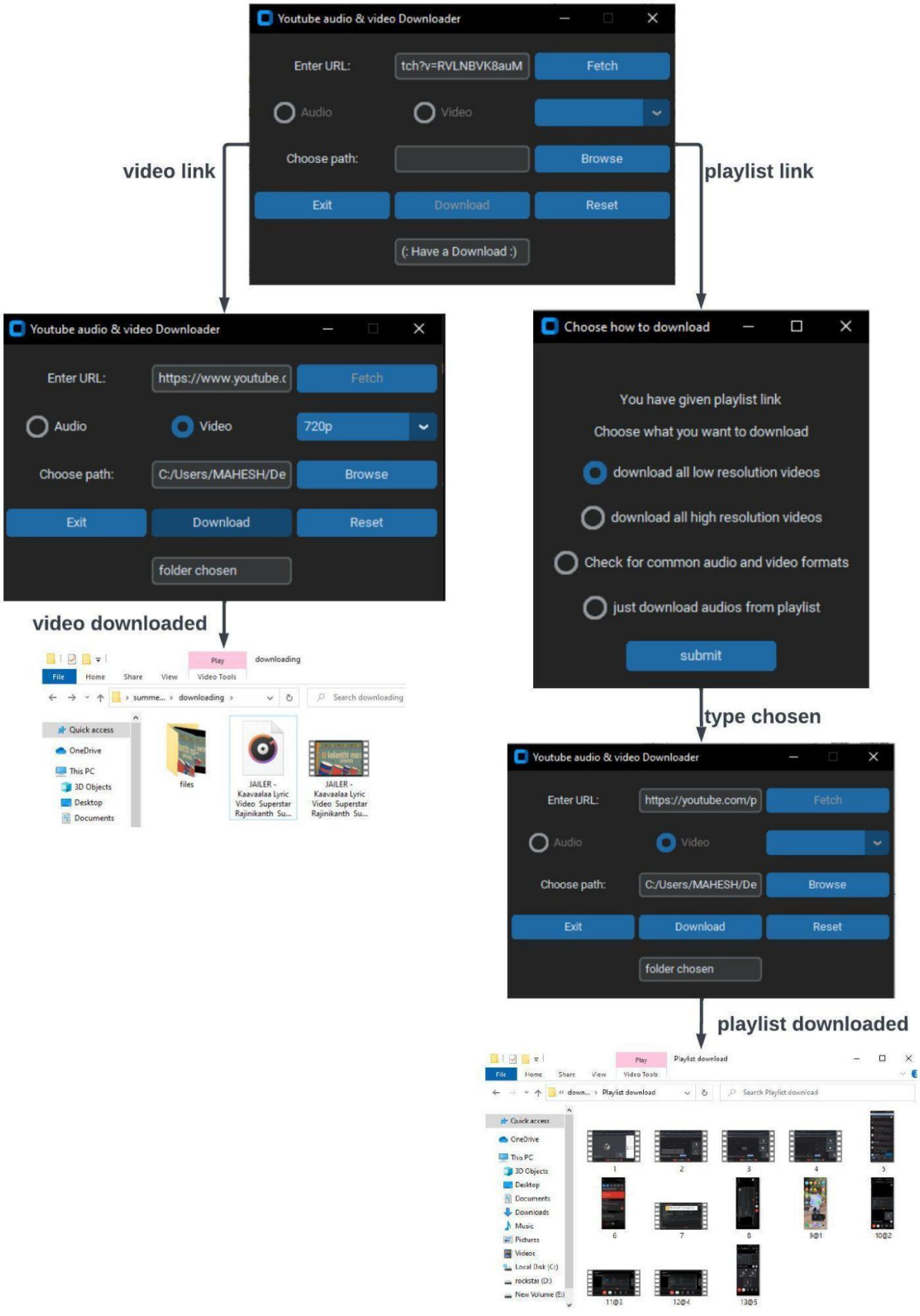


Fig: TRANSCODING PROCESS IN FFMPEG

4. EXPERIMENTAL RESULTS

The below one is the pictorial representation of flowchart of my project.



5. CONCLUSION AND FUTURE SCOPE

In conclusion, I have successfully implemented an application that harnesses the capabilities of various modules to facilitate the effortless download of YouTube audio, video, and playlists. This approach minimizes the need for extensive human interaction and eliminates the inconvenience of encountering unwanted ads often associated with online downloaders. By incorporating the pytube module, the application adeptly accesses diverse streams available for both video and audio content, allowing users to download according to their preferred resolutions and bitrates.Furthermore, the integration of the command-line tool ffmpeg simplifies the conversion process of downloaded audio container formats to the universally accepted MP3 format, ensuring compatibility with a wide range of systems. The user interface, meticulously crafted using customtkinter and tkinter, presents an appealing visual experience, enhancing accessibility for users, particularly students. In synergy, these components synergize to yield a robust and user-friendly YouTube audio and video downloader, offering a seamless and enjoyable user journey.

Future Scope:

- I will develop a web interface for this project and I will deploy it in the cloud.
- Inclusion of different audio formats for download
- Making high resolution videos available to download

6. REFERENCES

[1] https://www.y2mate.com/en807

[2] https://ymate.app/en69

[3] https://youtube4kdownloader.com/en84/convert-youtube-4k-video.html