

RECORDING USING PYTHON

Under the Guidance of :

Ms.D.Deepthi Mam
M.Tech, Assistant professor
Dept. Of CSE

BY:

P. Vanaja	-19BQ1A05I5
M. Ajay Kumar	-19BQ1A05E0
M. Akash	-19BQ1A05D1
K. Sivaiah	-20BQ5A0516

Problem Statement:

Recorder is an application to record the screen, audio and web cam and save them to the current working file. This provides user to watch the recorded videos and audios in offline.

We have developed this project using python and we have used several modules in python to build this application.

PROPOSED SOLUTION:

We have proposed the solution using different modules in python and we also included the Gui application to make our project more effective.

Advantages:

- This recorder enables the user to record audio, video, webcam individually.
- The recorded video or audio file is saved directly into the current working file.

Limitations:

- This cannot be accessed on mobiles.
- we cannot merge the audio and video file

Hardware And Software Requirements:

Hardware:

- 1) **Operating System:** Windows 7 or higher/ Mac OS X 10.11 or higher/ Linux: RHEL 6/7.
- 2) **RAM:** 4 GB or above.
- 3) **Storage:** 1 GB free disk space.

Software:

Python:

- Version-3.6 or above.
- Packages- (tkinter, pyautogui, cv2, numpy, win32api, sounddevice)

IDE:

pycharm

Screen Recording And Web Cam Recording

Screen recording enables us to create demonstration videos, record gaming achievements and create videos that can be shared online on social media.

Many industrial softwares exist out there that can help you do that very easily though.

In our project we will show you how to make your own simple screen recorder and web cam recorder in python that can further extend to your own needs.

Web cam recording is used in various applications like live streaming, traffic signals etc

Modules Used

1) PyAutoGui

2) NumPy

3) Win32api

4) Open cv

Pyautogui

PyAutoGUI lets your Python scripts control the mouse and keyboard to automate interactions with other applications. The API is designed to be as simple. **PyAutoGUI** works on Windows, macOS, and Linux, and runs on Python 2 and 3...

PyAutoGUI is a Python automation library used to click, drag, scroll, move, etc. It can be used to click at an exact position.

Open command prompt/ powershell and type:
`pip install pyautogui`

NumPy

- NumPy is a python library used for working with arrays.
- NumPy was created in 2005 by Trans Oliphant.

To install NumPy we have to use the statement

pip install numpy

NumPy Functions

- ❖ `Array()` → A NumPy array is a grid of values, all of the same type, and is indexed by a Non-Negative integers. We can store image related information in NumPy array.
- ❖ `Zeros()` → The `zeros()` function is used to get a new array of given shape and type, filled with zeroes.

WIN32API():

PYWIN32 is basically a very thin wrapper of python that allows us to interact with component object model(CMO) objects and automate windows applications with python

GetSystemMetrics() function:

The system metric or configuration setting to be retrived.

GetSystemMetrics(0)

returns the width of the window

GetSystemMetrics(1)

returns the height of the window

Open Cv

OpenCV is the huge open-source library for the computer vision, machine learning, and image processing.

By using it, one can process images and videos to identify objects, faces, or even handwriting of a human.

When it's integrated with various libraries, such as NumPy, python is capable of processing the OpenCV arraystructure for analysis.

OpenCV: To install OpenCV type the below command in the terminal.

`pip install open cv-python`

1. **cv2.imshow()**

As the name suggests this method is used for showing the images

Parameters: `cv2.imshow(window_name,image)`

2. **cv2.cvtColor()**

`cvtColor` can be interpreted as convert color and this method is used for converting the image from one color space to another color space (Find what is a color space here if you don't know about it)

Parameters: `cv2.cvtColor(src,code[, dst[, dstCn]])`

3. **cv2.waitKey()**

This function is very important, without this function `cv2.imshow()` won't work properly.

Parameters: `cv2.waitKey(wait time in milliseconds)`

4. **cv2.destroyAllWindows()**

This method destroys (in other words "closes") all the windows created using the opencv methods. If you want to close a specific window, then you can pass the window name as the argument within this function

Parameters: name of a window opened using opencv (not mandatory)

5. **cv2.VideoWriter(filename, fourcc, fps, frameSize)**

To save a video in OpenCV `cv2.VideoWriter()` method is used

Audio Recording

The `python-sounddevice` and `pyaudio` libraries provide ways to record audio with python.

Python sound device records to Numpy arrays and `pyaudio` records to bytes objects.

Both of these can be stored as WAV files using the `scipy` and `wave` libraries, respectively.

SOUND DEVICE

- This python module provides bindings for the portaudio library and a few convenience functions to play and record Numpy arrays containing audio signals. The sounddevice module is available for linux, macOS and Windows.
- `Sounddevice.rec()` : to record audio data from your sound device into a Numpy array

SCIPY.IO.WAVFILE.WRITE(FILENAME,RATE,DATA)

Parameters

Filename: string or open file handle

Output wave file

Rate : int

The sample rate(in samples/sec)

Data : ndarray

A 1-D or 2-D numpy array of either integer or float data type

GUI Application:

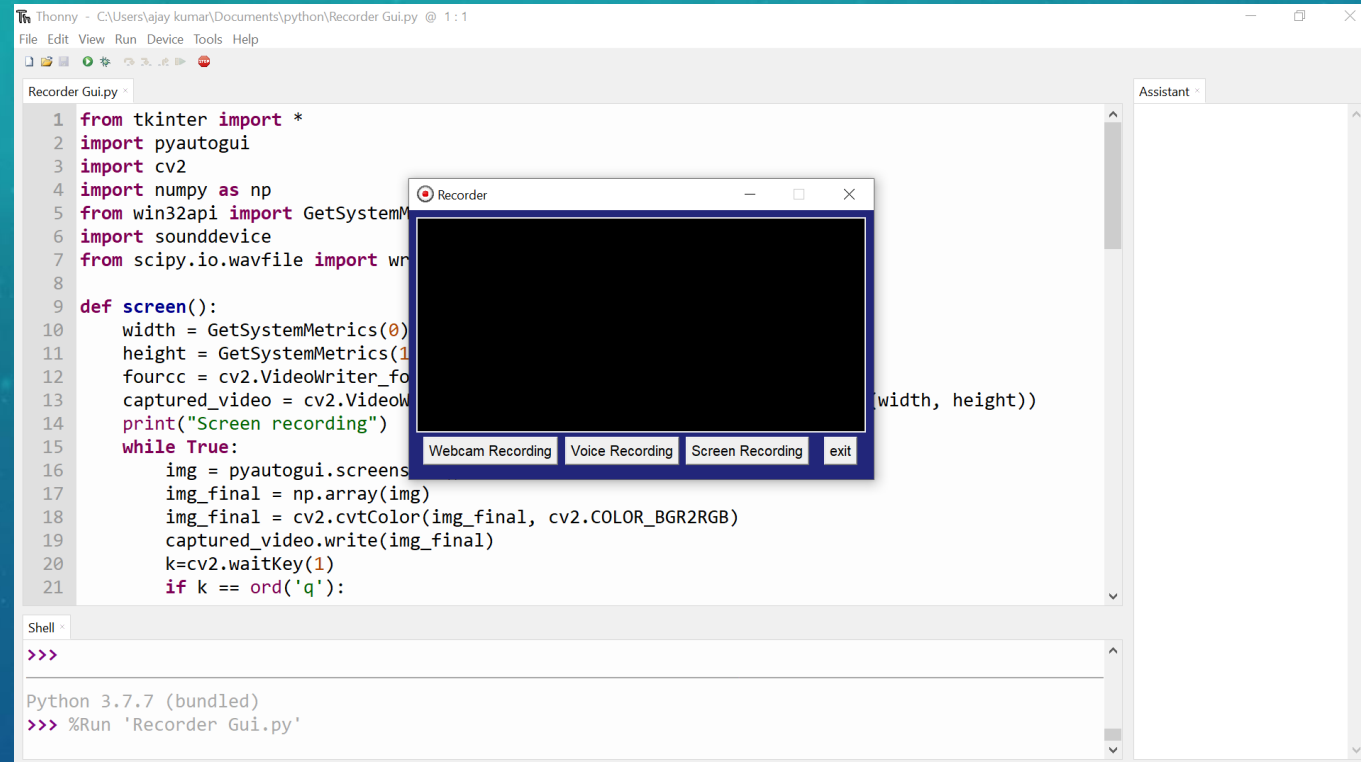
Tkinter:

Tkinter is the standard GUI library for Python. Python when combined with Tkinter provides a fast and easy way to create GUI applications. Tkinter provides a powerful object-oriented interface to the Tk GUI toolkit.

Creating GUI using Tkinter

- Import the *Tkinter* module.
- Create the GUI application main window.
- Add one or more of the above-mentioned widgets to the GUI application.
- Enter the main event loop to take action against each event triggered by the user.

GUI Application:

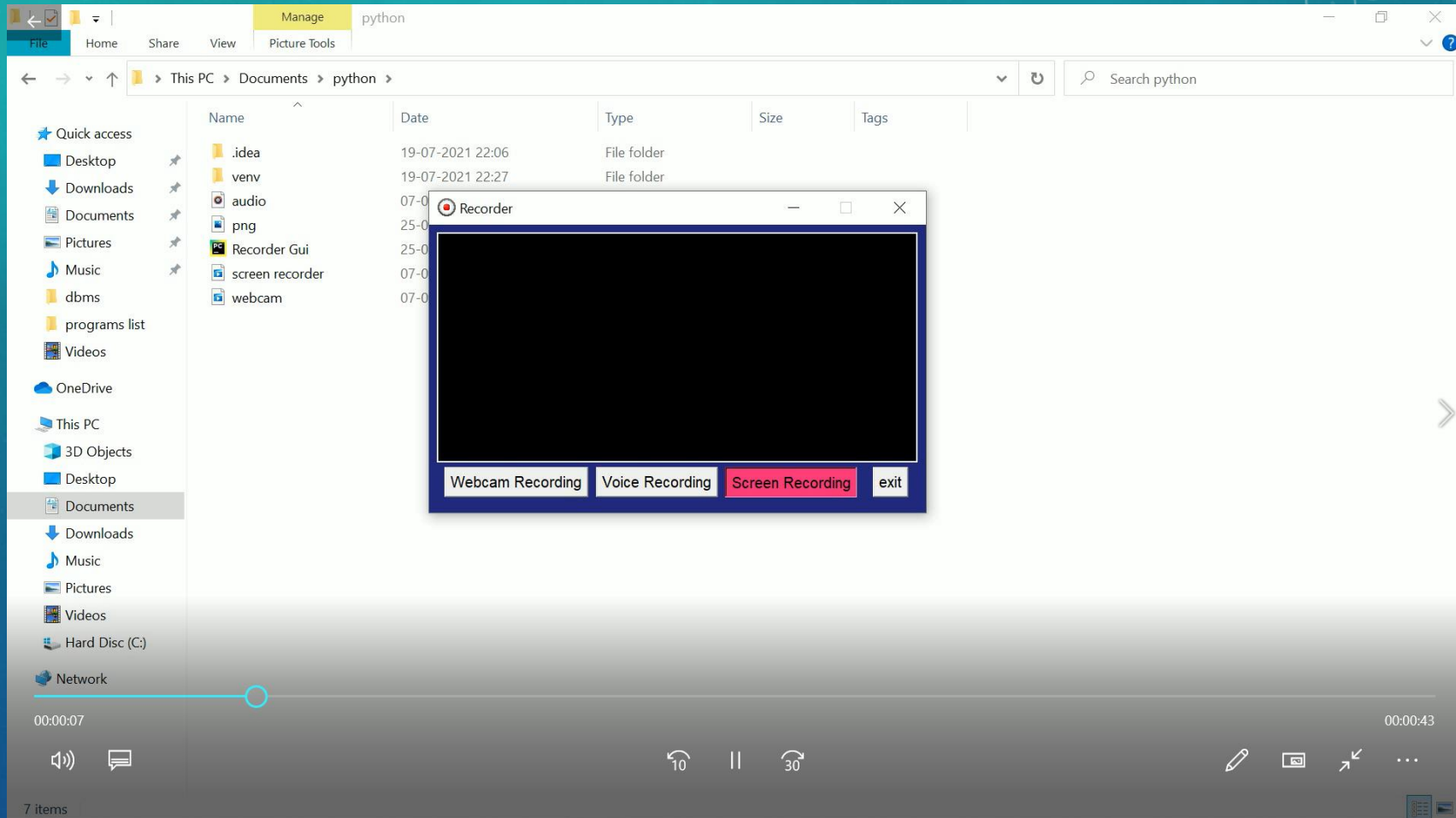


```
Thonny - C:\Users\ajay kumar\Documents\python\Recorder Gui.py @ 1:1
File Edit View Run Device Tools Help
Recorder Gui.py
Assistant

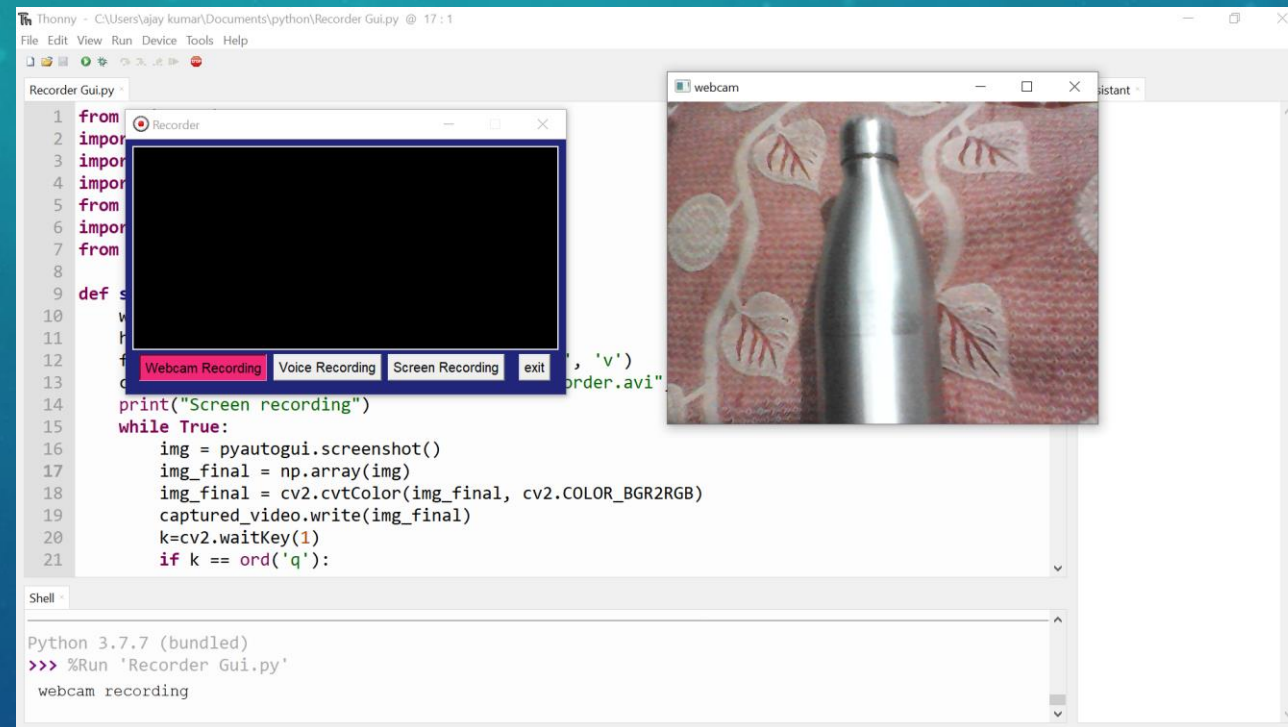
1 from tkinter import *
2 import pyautogui
3 import cv2
4 import numpy as np
5 from win32api import GetSystemMetrics
6 import sounddevice
7 from scipy.io.wavfile import write
8
9 def screen():
10     width = GetSystemMetrics(0)
11     height = GetSystemMetrics(1)
12     fourcc = cv2.VideoWriter_fourcc(*'mp4v')
13     captured_video = cv2.VideoWriter('recording.mp4', fourcc, 20, width, height)
14     print("Screen recording")
15     while True:
16         img = pyautogui.screenshot()
17         img_final = np.array(img)
18         img_final = cv2.cvtColor(img_final, cv2.COLOR_BGR2RGB)
19         captured_video.write(img_final)
20         k=cv2.waitKey(1)
21         if k == ord('q'):
22             break
23
24 if __name__ == '__main__':
25     screen()
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

Shell
Python 3.7.7 (bundled)
>>> %Run 'Recorder Gui.py'
```

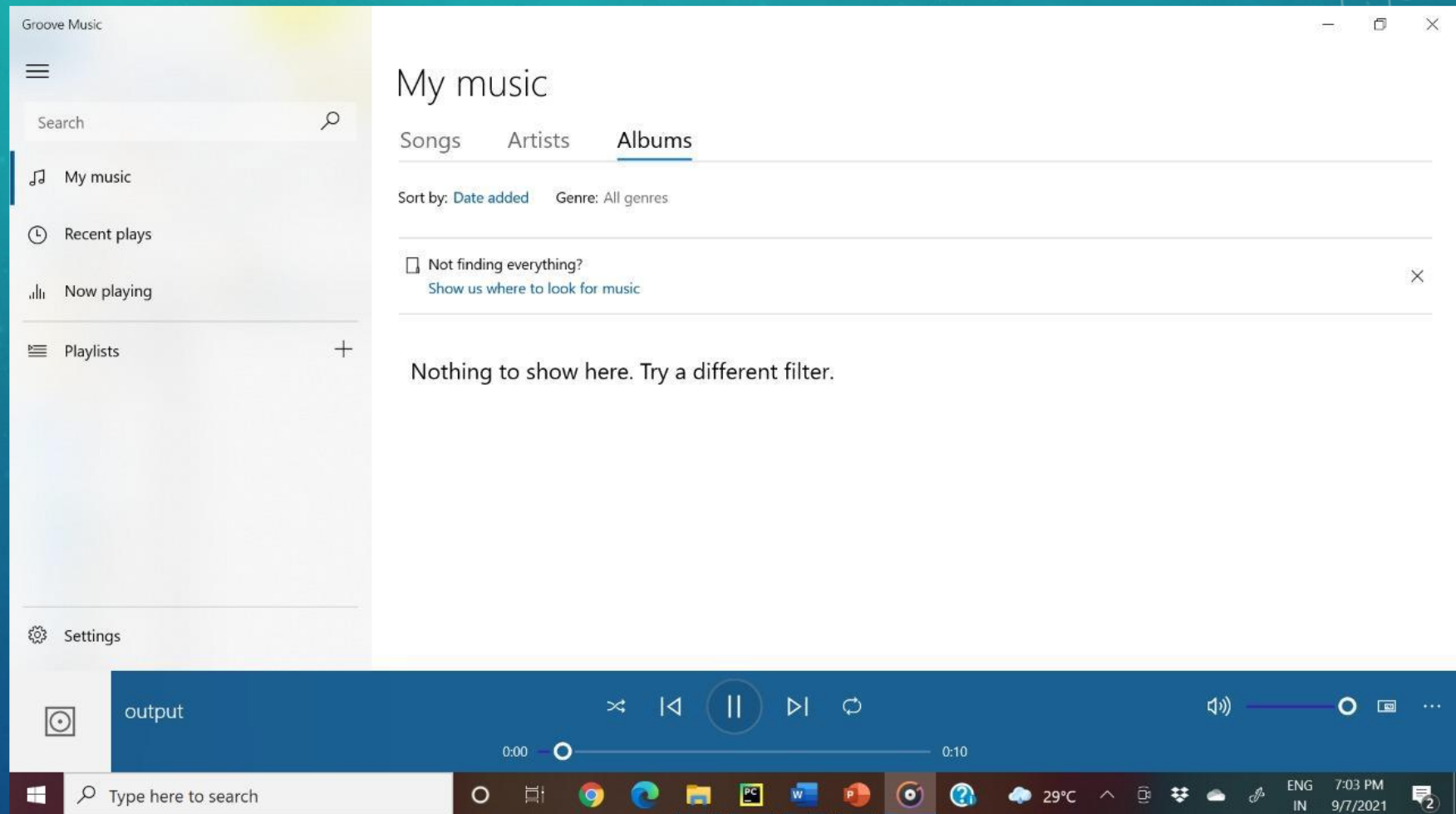

Screen Recording:



Web Cam Recording:



Audio Recording:





Thank You