# PROJECT TRAINING REPORT ON

**Voice command youtube downloader**

Submitted in fulfillment of the requirement for the awards of degree of

## Bachelors of Technology Computer Science & Engineering (Batch: 2017-21)

In



|  |  |
| --- | --- |
| **Under the Supervision of:** | **Submitted by** |
| Mr.Shravan Kumar | Mr.Sachin |
| Assistant Professor CSE, | CSE VIII Semester  Roll No. : |

-----------------------------------------------------------------------------------------------------

## KIIT College of Engineering, Gurgaon

**(Affiliated to MDU, Rohtak, Approved by AICTE, New Delhi)**

KIIT Campus, Gurgaon- Sohna Road, Maruti Kunj, Gurgaon-122102, Haryana June 2021

## STUDENT’s DECLARATION

I, **SCHIN** student of **B.Tech (CSE)-VIII Semester**, **Batch 2017-21** hereby declare that this project report on, **“Voice command youtube downloader”** which is being submitted in partial fulfillment for the programme in **B.Tech (CSE)**, is the record of authentic work carried out by me during the period from **April 16, 2021 to July 16, 2021** under the guidance of **Mr. Tushar.**

The matter embodied in this project report has not been submitted by me for any other degree or diploma.

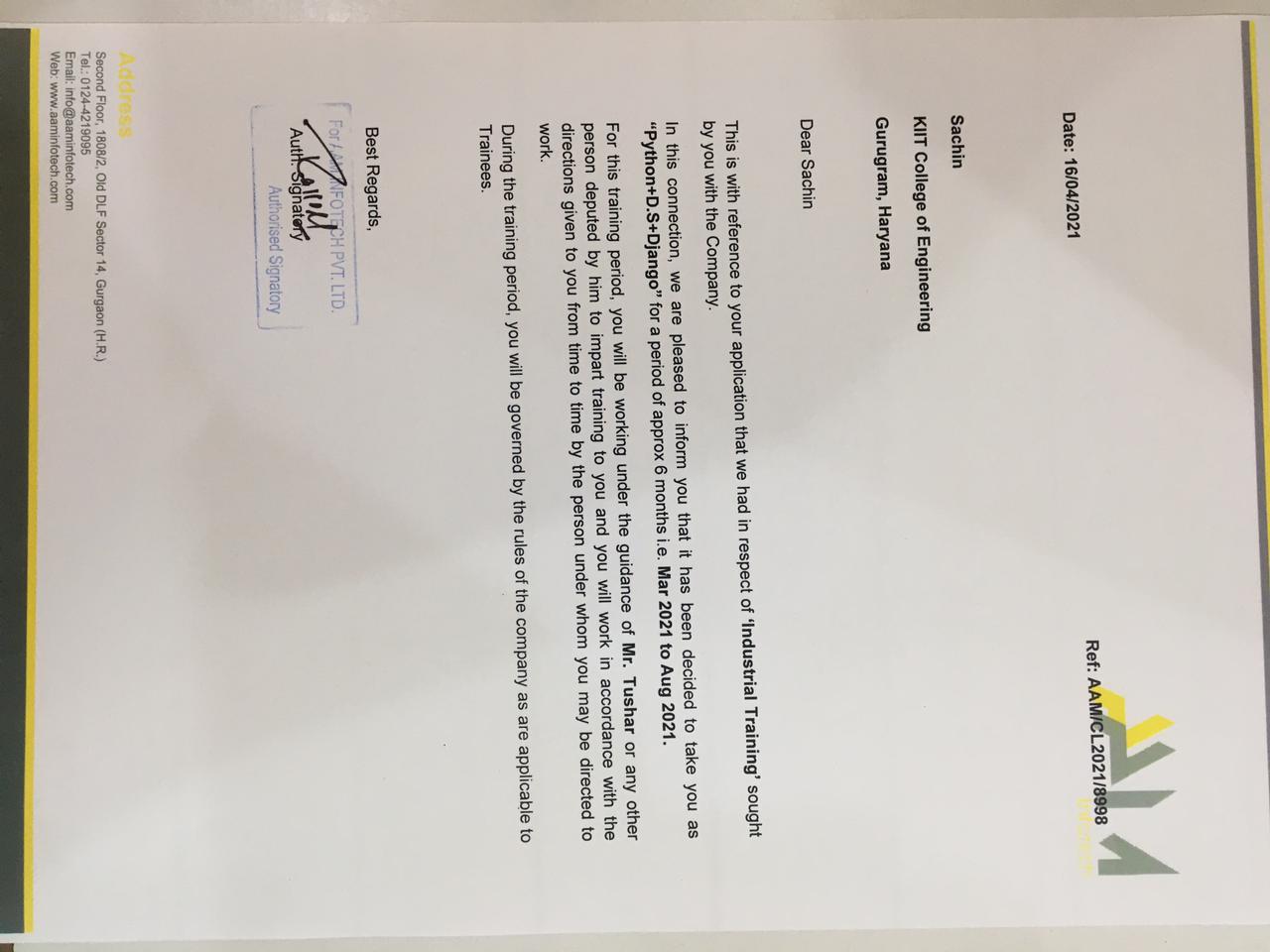
Signature of the Student

**Name of Student :Sachin**

**University Roll No :**

Date:

## CERTIFICATE FROM THE COMPANY/ORGANISATION



## CERTIFICATE FROM INTERNAL GUIDE

This is to certify that Sachin of CSE, 8th, Semester, 2017-2021, **KIIT College of**

**Engineering, Gurugram** has successfully completed the project work entitled **“voice command youtube downloader WASTE APPLICATION** in partial fulfillment of requirement for the completion of **Bachelor of Technology (Computer Science and Engineering)** as prescribed by the **Maharishi Dayanand University, Rohtak**.

This project report is the record of authentic work carried out by him/her during the period from march 16, 2021 to July 16, 2021.

He/She has worked under my guidance. The performance of the student is satisfactory.

Name of Internal Supervisor: Mr. Shravan Kumar

Designation: Assistant Professor

Date:

## ACKNOWLEDGEMENT

I take immense pleasure in thanking **Prof. (Dr.) S. S. AGGRAWAL**, **Director General (KIIT Group of Colleges) and Prof. (Dr.) S. K. AGGARWAL**, **Principal (KCE), Prof. (Dr.) VIKRAM SINGH (Vice-Principal (KCE), Mrs. REETA SAXENA TPO (KIIT)** and (**DR.) ATUL KUMAR** , our Project coordinator for having permitted me to carry out this project work.

I wish to express my deep sense of gratitude to our Project Supervisor, **Mr. SHARVAN KUMAR (Assistant Professor)** for his guidance and useful suggestions, which helped me in completing the project work, in time.

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

**Sachin**

**Batch: 2017-21**

**Roll No:**

## ABSTRACT

The project aims to develop a personal-assistant for window-based systems. Jarvis its inspiration from virtual assistants like Cortana for Windows, and Siri for iOS. It has been designed to provide a user-friendly interface for carrying out a variety of tasks by employing certain well-defined commands. Users can interact with the assistant either through voice commands or using keyboard input. As a personal assistant, Jarvis assists the end-user with day-to-day activities like general human conversation, searching queries in google, bing or yahoo, searching for videos, retrieving images, live weather conditions, word meanings, searching for medicine details, health recommendations based on symptoms and reminding the user about the scheduled events and tasks. The user statements/commands are analysed with the help of machine learning to give an optimal solution.

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **Chapter** | **Title** | **Page** |
| **No.** |  | **No.** |
|  |  |  |
|  | *Student’s declaration* | I |
|  | *Certificate from the company/organization* | II |
|  | *Certificate from internal guide* | III |
|  | *Acknowledgement* | IV |
|  | *Abstract* | V |
|  | *Table of content* | VI-VII |
|  | *List of figures* | VIII |
| Chapter 1 | Introduction  1.1 Aim  1.2 Scope  1.3 Advantages of voice command youtube downloader | 1-2 |
| Chapter 2 | System Requirement Analysis  2.1 Hardware  2.2 Software  2.2.A Operating system | 3-11 |
| Chapter 3 | SYSTEM DESIGN  3.1 Creating virtual env  3.2 **Managing Packages with pip**  **3.3 Source code** | 9-12 |

|  |  |  |
| --- | --- | --- |
| Chapter 4 | Systems Development and Implementation  4.1 Purpose  4.2 (a) Programme Development  4.2 (b) Implementation & Testing. | 13-20 |
| Chapter 5 | 5.1 Future use | 21-22 |
|  | References/Bibliography | 23 |

# CHAPTER 1

## Introduction

Voice command, assists the end-user with day-to-day activities like general human conversation, searching queries in various search engines like Google, Bing or Yahoo, searching for videos, retrieving images, live weather conditions, word meanings, searching for medicine details, health recommendations based on symptoms and reminding the user about the scheduled events and tasks. The user statements/commands are analysed with the help of machine learning to give an optimal solution. SCOPE Presently, Jarvis is being developed as an automation tool and virtual assistant. Among the Various roles played by Jarvis are

## 1.1Aim

It download the youtube video or audio by voice command

### 1.2 Scope

Download youtube video

Download youtube music in audio format

### 1.3 Advantages of voice command youtube downloader

It is easy to use.

It is neasy to maintain.

It uses less data.

It provide both audio and video.

# CHAPTER 2

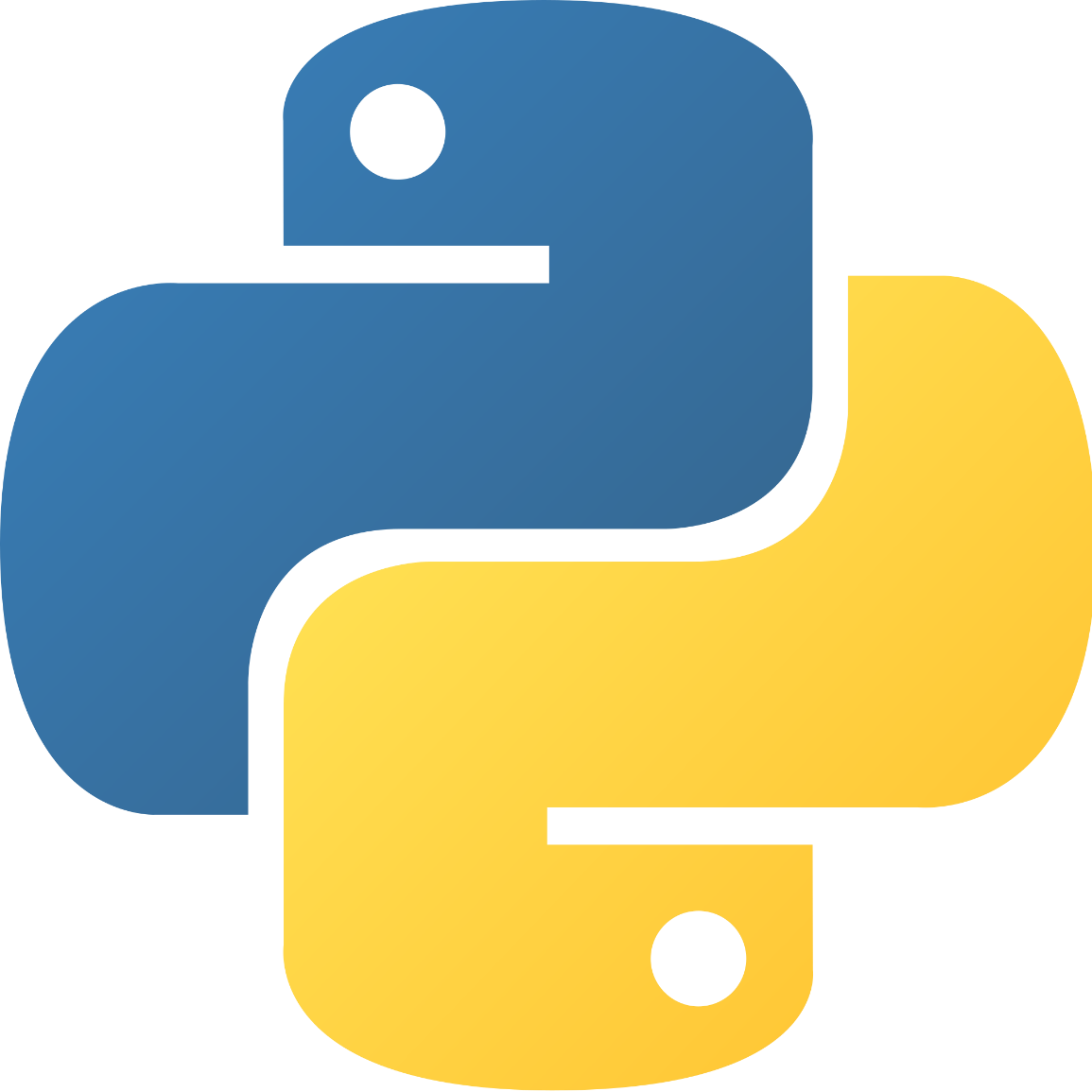
## SYSTEM REQUIREMENT ANALYSIS

The project is built using python

Python is a general-purpose, versatile, and powerful programming language. It’s a great first language because it’s concise and easy to read. Whatever you want to do, Python can do it. From web development to machine learning to data science, Python is the language for you.

Why we love it:

* Great first language
* Large programming community
* Excellent online documentation
* Endless libraries and packages
* World-wide popularity
* Powerful and flexible



### 2.1 Hardware Requirement

* 64-bit Microsoft® Windows® 8/10.
* x86\_64 CPU architecture; 2nd generation Intel Core or newer, or AMD CPU with support for a Windows Hypervisor.

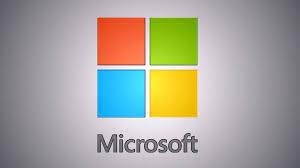
2 GB RAM or more.

* 8 GB of available disk space minimum (python+ **pip package**)  1280 x 800 minimum screen resolutions.

#### 2.2 Software requirement

**A. Operating System**

 Microsoft Windows 7/8/10 (32-bit or 64-bit)



# CHAPTER-3

**SYSTEM DESIGN**

**3.1**

## Creating Virtual Environments

The module used to create and manage virtual environments is called [venv](https://docs.python.org/3/library/venv.html#module-venv). [venv](https://docs.python.org/3/library/venv.html#module-venv) will usually install the most recent version of Python that you have available. If you have multiple versions of Python on your system, you can select a specific Python version by running python3 or whichever version you want.

To create a virtual environment, decide upon a directory where you want to place it, and run the [venv](https://docs.python.org/3/library/venv.html#module-venv) module as a script with the directory path:

python3 -m venv tutorial-env

This will create the tutorial-env directory if it doesn’t exist, and also create directories inside it containing a copy of the Python interpreter, the standard library, and various supporting files.

A common directory location for a virtual environment is .venv. This name keeps the directory typically hidden in your shell and thus out of the way while giving it a name that explains why the directory exists. It also prevents clashing with .env environment variable definition files that some tooling supports.

Once you’ve created a virtual environment, you may activate it.

On Windows, run:

tutorial-env\Scripts\activate.bat

On Unix or MacOS, run:

source tutorial-env/bin/activate

(This script is written for the bash shell. If you use the **csh** or **fish** shells, there are alternate activate.csh and activate.fish scripts you should use instead.)

Activating the virtual environment will change your shell’s prompt to show what virtual environment you’re using, and modify the environment so that running python will get you that particular version and installation of Python. For example:

$ source ~/envs/tutorial-env/bin/activate

(tutorial-env) $ python

Python 3.5.1 (default, May 6 2016, 10:59:36)

...

>>> import sys

>>> sys.path

['', '/usr/local/lib/python35.zip', ...,

'~/envs/tutorial-env/lib/python3.5/site-packages']

>>>

## Managing Packages with pip

You can install, upgrade, and remove packages using a program called **pip**. By default pip will install packages from the Python Package Index, <[https://pypi.org](https://pypi.org/)>. You can browse the Python Package Index by going to it in your web browser.

pip has a number of subcommands: “install”, “uninstall”, “freeze”, etc. (Consult the [Installing Python Modules](https://docs.python.org/3/installing/index.html#installing-index) guide for complete documentation for pip.)

You can install the latest version of a package by specifying a package’s name:

**SOURCE CODE**

**\*Acess media**

**"""**

**THIS MODULE DEALS WITH THE MEDIA FILES**

**"""**

**from iovoice.input import voiceinput**

**from iovoice.output import speak**

**from .downloadmedia import searchonline**

**import os**

**def searchandplay(term,/):**

**"""**

**looks for file containing the term in its name**

**"""**

**walk\_result = list(os.walk(**

**os.path.join(**

**os.getcwd(),**

**"downloads"**

**)**

**))**

**if walk\_result:**

**ch = voiceinput("No result found with given term. Would you like me to search it online?")**

**if 'yes' in ch:**

**searchonline(term)**

**else:**

**pass**

**DOWNLOAD\_PATH = os.path.join(**

**os.getcwd(),**

**"downloads"**

**)**

**# from bs4 import BeautifulSoup**

**# create the folder to contain the online downloaded media**

**try:**

**os.mkdir("downloads")**

**except:pass**

**def searchonline(term,/):**

**response = req.get(SEARCH\_URL.format(term.replace(" ","+"))).text**

**watch\_ids = re.findall(REGEX\_WATCH,response)**

**# print(watch\_ids)**

**# extract the information from each watch id**

**for x,wid in enumerate(watch\_ids):**

**temp = ytd.new(WATCH\_URL.format(wid))**

**print(f"{x} -> {temp.title}")**

**speak("Pick any one song or you pick multiple seperated using commas")**

**picks = input().split(",")**

**if picks[0].lower() == "all":**

**for x in watch\_ids:**

**save(x)**

**else:**

**for x in picks:**

**if int(x) < 0 or int(x) > len(watch\_ids):**

**break**

**else:**

**if len(picks) > 1:**

**for x in picks:**

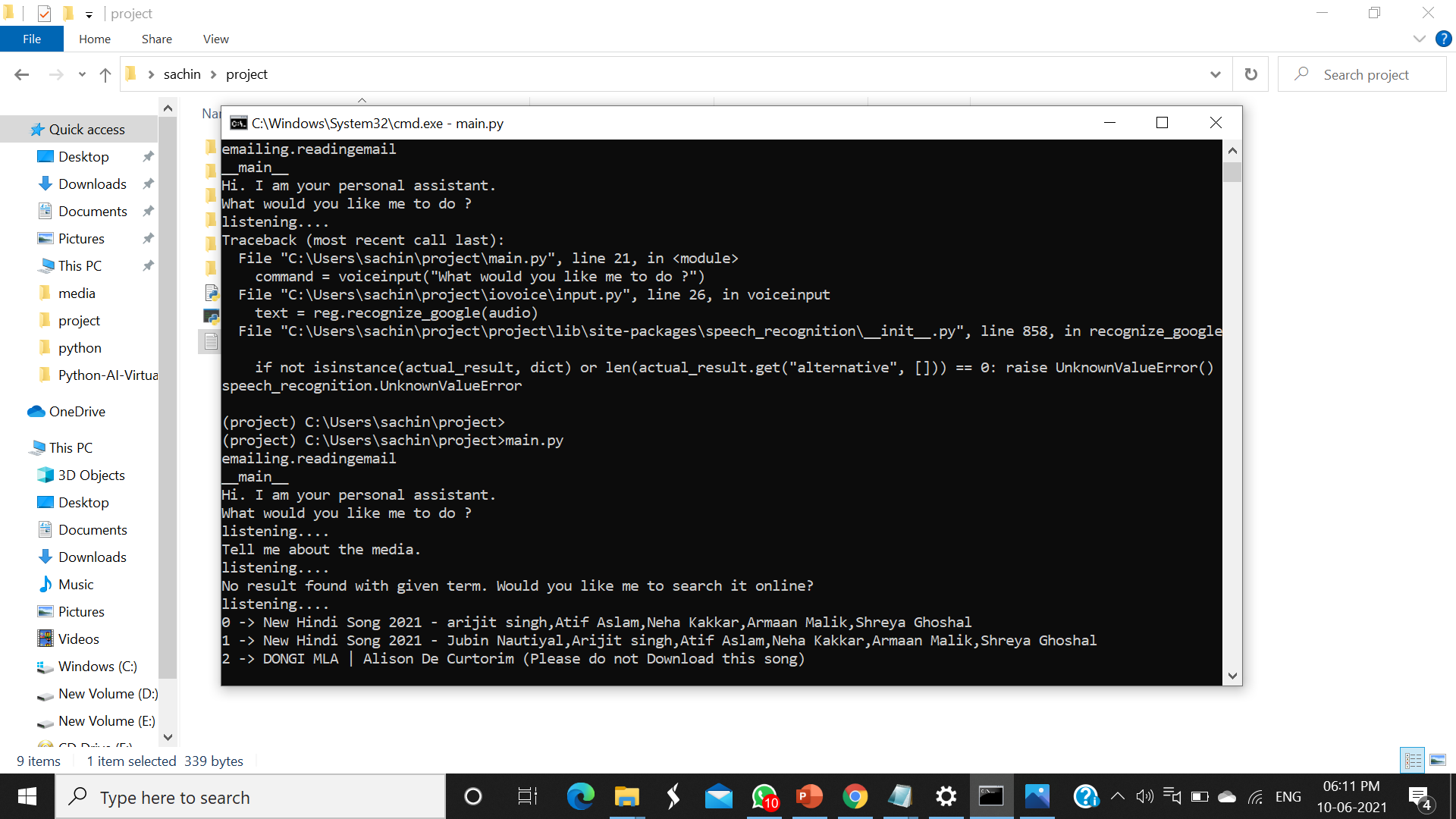
**save(watch\_ids[int(x)])**

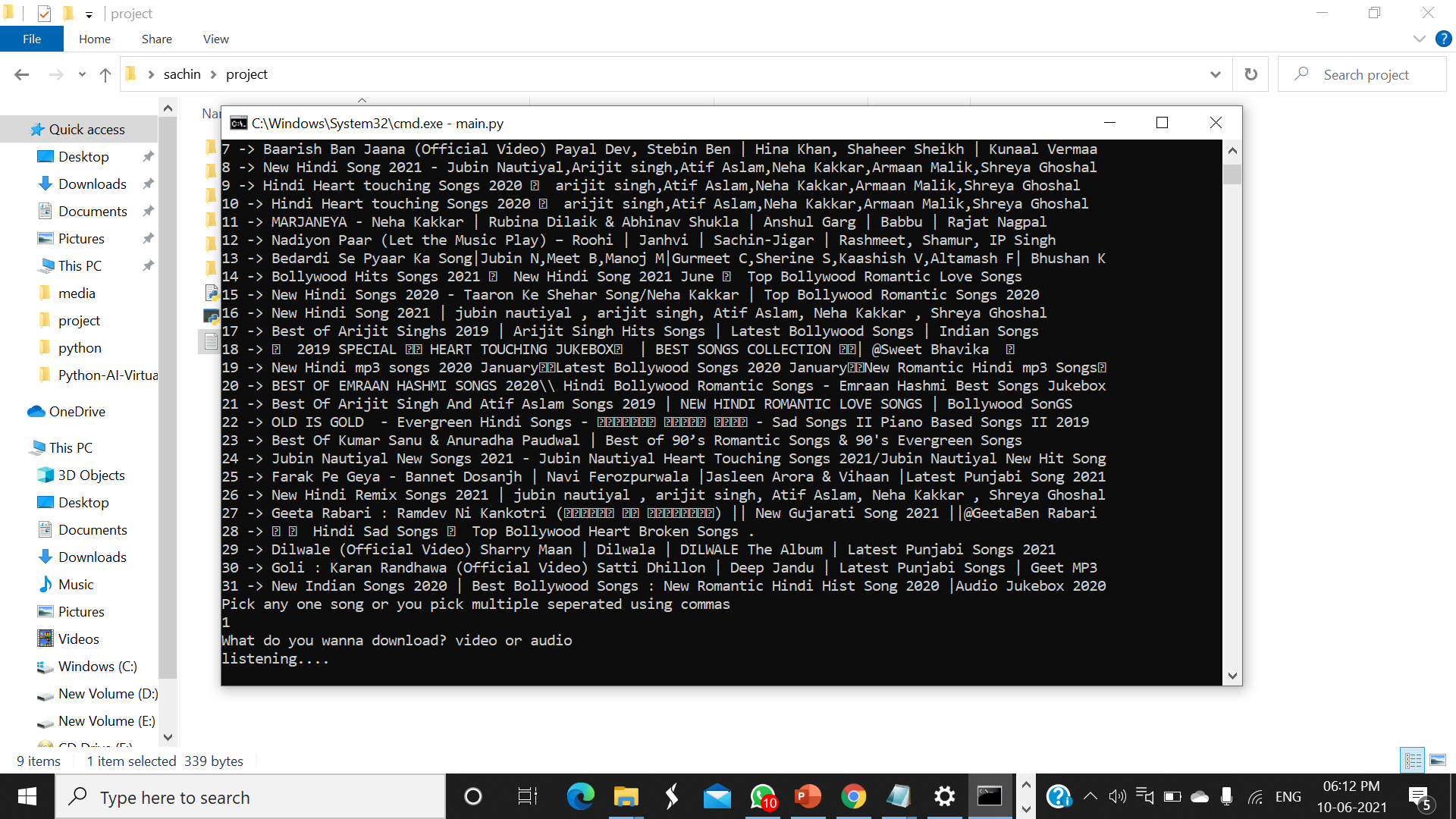
**elif len(picks) == 1:**

**save(watch\_ids[int(picks[0])])**

**else:**

**speak("Invalid Input")**

****

****

## CHAPTER-4

### SYSTEM DEVELOPMENT AND IMPLEMENTATION

hese are the following steps to build youtube video downloader project in python :

* Import libraries
* Create display window
* Create field to enter link
* Create function to start downloading

#### 1. Import Libraries

Start the project by importing the required modules.

from tkinter import \*

from pytube import YouTube

In this python project, we import Tkinter and [pytube](https://pypi.org/project/pytube/) modules.

#### 2. Create Display Window

root = Tk()

root.geometry('500x300')

root.resizable(0,0)

root.title("DataFlair-youtube video downloader")

* **Tk()** used to initialize tkinter to create display window
* **geometry()** used to set the window’s width and height
* **resizable(0,0)** set the fix size of window
* **title()** used to give the title of window

Label(root,text = 'Youtube Video Downloader', font ='arial 20 bold').pack()

* **Label()** widget use to display text that users can’t able to modify.
* **root** is the name of the window
* **text** which we display the title of the label
* **font** in which our text is written
* **pack** organized widget in block

#### 3. Create Field to Enter Link

link = StringVar()

Label(root, text = 'Paste Link Here:', font = 'arial 15 bold').place(x= 160 , y = 60)

link\_enter = Entry(root, width = 70,textvariable = link).place(x = 32, y = 90)

* **link** is a string type variable that stores the youtube video link that the user enters.
* **Entry()** widget is used when we want to create an input text field.
* **width** sets the width of entry widget
* **textvariable** used to retrieve the value of current text variable to the entry widget
* **place()** use to place the widget at a specific position

#### 4. Create Function to Start Downloading

def Downloader():

url =YouTube(str(link.get()))

video = url.streams.first()

video.download()

Label(root, text = 'DOWNLOADED', font = 'arial 15').place(x= 180 , y = 210)

Button(root,text = 'DOWNLOAD', font = 'arial 15 bold' ,bg = 'pale violet red', padx = 2, command = Downloader).place(x=180 ,y = 150)

root.mainloop()

Url variable gets the youtube link from the link variable by **get()** function and then **str()** will convert the link in string datatype.

The video is download in the first present stream of that video by **stream.first()** method.

**Button()** widget used to display button on the window.

* **text** which we display on the label
* **font** in which the text is written
* **bg** sets the background color
* **command** is used to call the function

**root.mainloop()** is a method that executes when we want to run the program.

Fig 4.2.3 Gradle Build is Running

**5.2 FUTURE USE**

1. Role based Authentication: There would be different routes and different role which can be there to access screen securely and information can be shared without any security issues.
2. Can be made a effective dashboard which shows information etc.
3. Event management system for the users can be made using this project.
4. The mailing system could be added that would send the information when the work done.
5. Social authentication can be made.
6. Can be made responsive using other styles or themes.
7. Components are reusable with little configuration.
8. Api could be used for different front end.

## BIBLIOGRAPHY

1. <https://stackoverflow.com/questions/56154908/onboarding-screen-animations>
2. https://www.tutorialspoint.com/android/index.htm[/](https://mongoosejs.com/docs/)
3. <https://www.youtube.com/results?search_query=Android+app>
4. 472587433962
5. https://stackoverflow.com/questions/3058919/text-to-speechtts-androi