VisionHack 2024

2347129-Mushabbar Ahmed
2347122-Justin ziv Patil

Attendance Management Using Face Detection

This project is a **Face Recognition-based Attendance Management System**. It captures the faces of known individuals, matches them in real-time, and marks attendance. The following Python libraries and frameworks are utilized to implement this system

Libraries Used

1. OpenCV (cv2): Captures live video, detects and marks faces, and displays video frames with recognized faces.
2. SimpleFacerec: Encodes known faces from images and matches them in live video for face recognition.
3. pyttsx3: Converts text to speech for audio feedback upon successful face recognition.
4. SpeechRecognition (sr): Processes speech input for potential voice command integration.
5. Tkinter: Builds the GUI for buttons, user management, and attendance retrieval.
6. Pillow (PIL): Resizes and displays images in the GUI.
7. Datetime: Logs and formats attendance dates and times.
8. OS: Handles file existence checks and directory management.
9. Shutil: Performs high-level file operations like copying or removing directories.

About the Project

Purpose:

- To simplify and automate the attendance process using face recognition technology.
- Manual attendance is time-consuming and prone to errors.
- Face recognition ensures accuracy and efficiency.

Key Features:

- Real-time face recognition using a webcam.
- Automated attendance marking into CSV files.
- GUI for managing attendance, adding users, and fetching records.

VisionHack 2024

2347129-Mushabbar Ahmed
2347122-Justin ziv Patil

- Audio feedback for recognized users.

How It Works:

- Encodes faces of known individuals stored in an "images" folder.
- Captures live video frames, detects faces, and matches them to pre-encoded images.
- If a match is found, attendance is logged with the name, time, and date.

Code:

```
import cv2
from simple_facerec import SimpleFacerec
import datetime
import pyttsx3  # pip install pyttsx3
import speech_recognition as sr
import os
import shutil
from tkinter import Tk
from tkinter.filedialog import askopenfilename
from tkinter import *
import tkinter.messagebox as t
from PIL import Image, ImageTk


def start():
    root = Tk()
    root.withdraw()
    root.update()

    present_date = datetime.datetime.now().strftime("%d%b%y")
    new_date = datetime.datetime.now().strftime("%d:%b:%y")
    print(new_date)
    strTime = datetime.datetime.now().strftime("%H:%M:%S")
    print(strTime)
    # Encode faces from a folder
    sfr = SimpleFacerec()
    sfr.load_encoding_images("images/")
    engine = pyttsx3.init('sapi5')
    voices = engine.getProperty('voices')
    engine.setProperty('voice', voices[1].id)

    def speak(audio):
        engine.say(audio)
        engine.runAndWait()
```

VisionHack 2024

2347129-Mushabbar Ahmed
2347122-Justin ziv Patil

```
    # Load Camera
    cap = cv2.VideoCapture(0)

    i1 = 0
    list2 = ["farhaan", "mushu", "Jeff Bezoz", "Ryan Reynolds", "Elon
Musk", "faizan", "junaid", "siddiq", "ali",
             "taheer", "tarun", "tenzing", "nishmitha", "gokul"]
    string_date = str(present_date)
    folder_loc = os.getcwd()
    attendance_data = f"attendance{string_date}.csv"
    directories = os.listdir()
    if attendance_data not in directories:
        f1 = open(attendance_data, "w+")
        st = "name,time1,date1"
        read = f1.readline()
        print(read)
        print("sasa")
        if read != st:
            print("gfg")
            f1.write(f'name,time1,date1')
        f1.close()
    print(folder_loc)



    def attendance(name):

        name_file = name + ".csv"
        folder_loc1 = os.getcwd()
        directories = os.listdir()
        if name_file not in directories:
            f11 = open(name_file, "w+")
            st1 = "name,time1,date1"
            read1 = f11.readline()
            print(read1)
            print("sasa")
            if read1 != st1:
                print("gfg")
                f11.write(f'name,time1,date1')
            f11.close()

        with open(name_file, "r+") as fn:
            data_fn = fn.readlines()
            namelist_fn = []
            datelist_fn = []
            for line1 in data_fn:
                entry = line1.split(",")
                namelist_fn.append(entry[0])
                datelist_fn.append(entry[2])
            if new_date not in datelist_fn:
                fn.write(f'\n{name},{strTime},{new_date}')
                fn.close()

        with open(attendance_data, "r+") as f:
            data = f.readlines()
            namelist = []
            datelist = []
```

VisionHack 2024

2347129-Mushabbar Ahmed
2347122-Justin ziv Patil

```
            for line in data:
                entry = line.split(",")
                namelist.append(entry[0])
                datelist.append(entry[2])
            if name not in namelist:
                f.write(f'\n{name},{strTime},{new_date}')
                f.close()

    while True:
        ret, frame = cap.read()
        # Detect Faces
        face_locations, face_names = sfr.detect_known_faces(frame)

        for face_loc, name in zip(face_locations, face_names):
            y1, x2, y2, x1 = face_loc[0], face_loc[1], face_loc[2],
face_loc[3]

            cv2.putText(frame, name, (x1, y1 - 10),
cv2.FONT_HERSHEY_DUPLEX, 1, (0, 0, 200), 2)
            cv2.rectangle(frame, (x1, y1), (x2, y2), (0, 0, 200), 4)

            for li in list2:

                if name == li:
                    with open(attendance_data, "r+") as f:
                        data = f.readlines()
                        namelist = []
                        for line in data:
                            entry = line.split(",")
                            namelist.append(entry[0])
                        if name not in namelist:
                            word = name, "has logged in at", strTime
                            speak(word)
                        attendance(name)

        cv2.imshow("Frame", frame)

        key = cv2.waitKey(1)
        if key == 27:
            break

    cap.release()
    cv2.destroyAllWindows()

    pass


def new_user_add():
    pass


def delete():

    pass


def get_attendance_bystudent1():
    from PIL import Image, ImageTk
```

VisionHack 2024

2347129-Mushabbar Ahmed
2347122-Justin ziv Patil

```
    def get_student1():
        student_search_name = student_name.get()
        student_search_file = f'{student_search_name}.csv'
        print(student_search_file)
        with open(student_search_file, "r+") as new:
            new_readline_list = []
            data1 = new.readlines()
            i = 0
            for dat in data1:
                i += 1
            i =  i
            print(i)
            i1.set(i)
            label5.update()
    root.destroy()
    root_new = Tk()
    fraem1 = Frame(root_new)
    fraem1.pack()
    root_new1 = Frame(root_new)
    root_new1.pack()
    enter_student = Label(fraem1, text="ENTER STUDENT NAME:")
    enter_student.grid(row=0, column=0)
    student_name = StringVar()
    entry_student = Entry(fraem1, textvariable=student_name)
    entry_student.grid(row=0, column=1)
    search_button = Button(fraem1, text="SEARCH", command=get_student1)
    search_button.grid(row=0, column=2)
    adarsh_college11 = Image.open("J.png")
    re111 = adarsh_college11.resize((200, 200))
    ada = ImageTk.PhotoImage(re111)
    picture21 = Label(root_new1, image=ada)
    picture21.grid(row=1, column=1)
    label1 = Label(root_new1, text="Name", borderwidth=6, relief=GROOVE)
    label1.grid(row=1, column=2)
    label2 = Label(root_new1, text="Course", borderwidth=6, relief=GROOVE)
    label2.grid(row=1, column=3)
    label3 = Label(root_new1, text="Sem", borderwidth=6, relief=GROOVE)
    label3.grid(row=1, column=4)
    label4 = Label(root_new1, text="Total days", borderwidth=6,
relief=GROOVE)
    label4.grid(row=1, column=5)
    i1 = StringVar()
    i1.set("no")
    label5 = Label(root_new1, textvariable=i1, borderwidth=6,
relief=GROOVE)
    label5.grid(row=1, column=6)
    root_new.mainloop()


root = Tk()
# root.configure(bg="#1E130C")
root.configure(bg="black")
root.title("AUTO ATTENDANCE ")
# root.geometry("382x250")
root.minsize(1000, 540)
root.maxsize(1000, 540)
main_frame_top = Frame(root, bg="grey", borderwidth=6)
```

VisionHack 2024

2347129-Mushabbar Ahmed
2347122-Justin ziv Patil

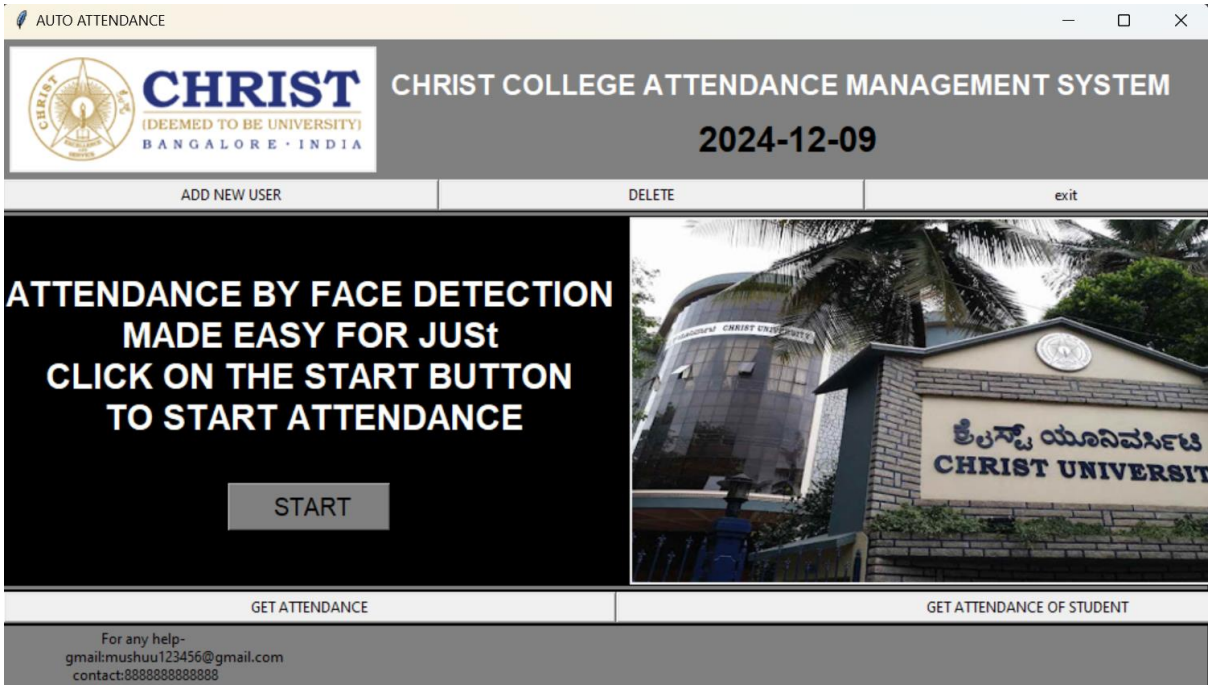
```
main_frame_top.pack(fill=X)
adarsh_logo = Image.open("christ_logo.png")
resize_logo = adarsh_logo.resize((300, 100))
adarsh_log = ImageTk.PhotoImage(resize_logo)
picture1 = Label(main_frame_top, image=adarsh_log)
picture1.grid(row=0, column=0)
main_frame_top2 = Frame(main_frame_top, bg="grey")
main_frame_top2.grid(row=0, column=1)
attendance_label = Label(main_frame_top2, text="CHRIST COLLEGE ATTENDANCE
MANAGEMENT SYSTEM  ", bg="grey", fg="white",
                         font="comicsansms 17 bold", pady=10, padx=10)

attendance_label.grid(row=0)
import datetime

date = datetime.datetime.now()
date12 = str(date).split(" ")
date2 = date12[0]

test = Label(main_frame_top2, text=date2, font="comicsansms 22 bold",
bg="grey")
test.grid(row=1, column=0)
frame_button_top = Frame(root)
frame_button_top.pack()
frame_button_top1 = Frame(root, borderwidth=1, relief='solid', bg="grey",
height=5)
frame_button_top1.pack(fill=X)

add_user = Button(frame_button_top, text="        ADD NEW USER  ",
command=new_user_add, padx=120)
add_user.grid(row=0, column=1)
delete_button = Button(frame_button_top, text="           DELETE
", command=delete, padx=120)
delete_button.grid(row=0, column=2)
exit_button = Button(frame_button_top, text="           exit           ",
command=quit, padx=120)
exit_button.grid(row=0, column=3)
middle_frame = Frame(root, bg="black")
middle_frame.pack(fill=X)
left_frame = Frame(middle_frame, bg="black")
left_frame.grid(row=0, column=0)
right_frame = Frame(middle_frame)
right_frame.grid(row=0, column=1, padx=10)
start_button = Button(left_frame, text="    START    ", command=start,
padx=7, bg="grey", font="bold")
start_button.grid(row=3, column=0)
text = Label(left_frame, text="ATTENDANCE BY FACE DETECTION\n"
                              "MADE EASY FOR JUSt\n"
                              "CLICK ON THE START BUTTON\n "
                              "TO START ATTENDANCE\n ",
             font="comicsansms 22 bold", fg="white", bg="black")
text.grid(row=0, column=0)
adarsh_college = Image.open("collge.jpg")
re = adarsh_college.resize((500, 300))
adarsh_college_image = ImageTk.PhotoImage(re)
picture2 = Label(right_frame, image=adarsh_college_image)

picture2.grid(row=0, column=4)
```

VisionHack 2024

2347129-Mushabbar Ahmed
2347122-Justin ziv Patil

```
frame_button_dowm1 = Frame(root, borderwidth=1, relief='solid', bg="grey",
height=5)
frame_button_dowm1.pack(fill=X)
frame_button_dowm = Frame(root, borderwidth=1, relief='solid', bg="black")
frame_button_dowm.pack(fill=X)
get_attedance = Button(frame_button_dowm, text="       GET ATTENDANCE
", command=quit, padx=180)
get_attedance.grid(row=0, column=0)
getstudentattandance = Button(frame_button_dowm, text="           GET
ATTENDANCE OF STUDENT           ",
                              command=get_attendance_bystudent1, padx=220)
getstudentattandance.grid(row=0, column=2)
frame_dowm1 = Frame(root, borderwidth=1, relief='solid', bg="grey")
frame_dowm1.pack(fill=X)
label1 = Label(frame_dowm1, text="For any help- \n"
                                 "\tgmail:mushuu123456@gmail.com\n"
                                 "contact:8888888888888"
               , bg="grey")
label1.grid(row=0, column=0)
root.mainloop()
```

Output:



Features of the Attendance Management System (GUI Output)

Navigation Buttons:

- **Add New User: To add new individuals for face recognition.**
- **Delete: To remove existing user data.**

VisionHack 2024

2347129-Mushabbar Ahmed
2347122-Justin ziv Patil

Exit: To close the application.

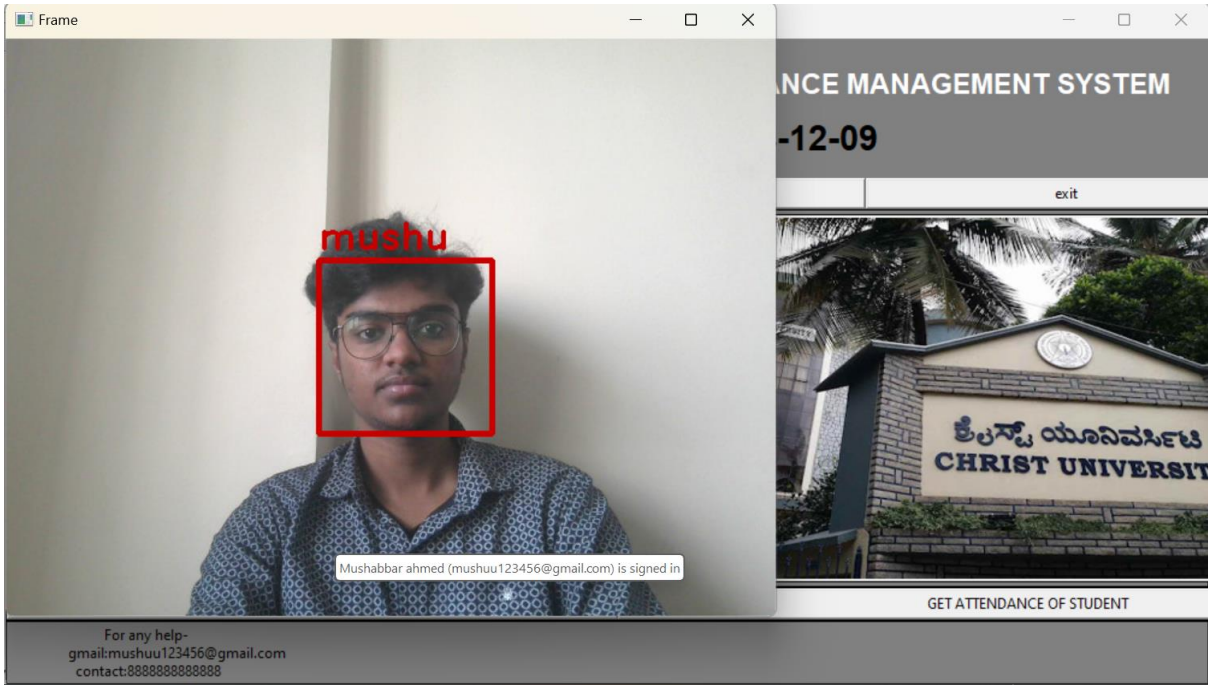
Main Display Area:

- **Instructions: Explains how to use the system—"Click on the START button to start attendance."**
- **Large START button for initiating the face recognition-based attendance process.**

Attendance Retrieval Buttons:

- **Get Attendance: Retrieve the recorded attendance list.**
- **Get Attendance of Student: Get attendance for a specific student.**

This interface is user-friendly, combining clear instructions, intuitive buttons

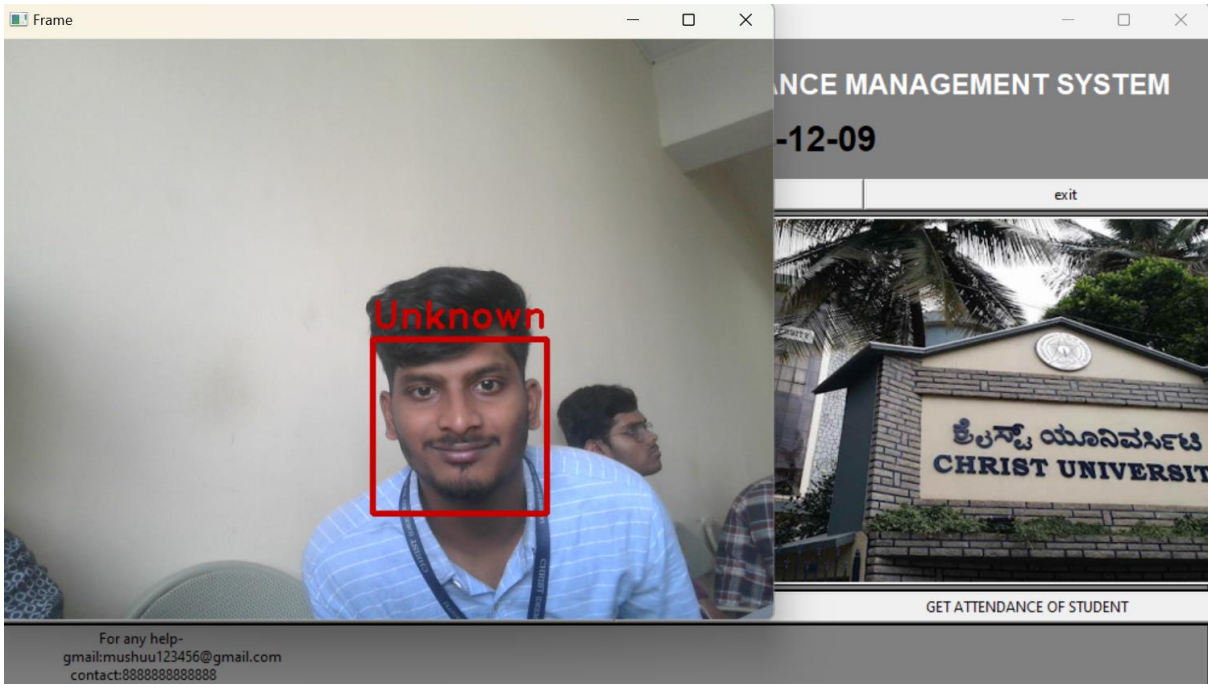


If student image is present in Images folder then it will detect the student face and mentions their name on screen along with it it records their entry in csv and every student csv is created so that it will be easy to search and have

2347129-Mushabbar Ahmed
2347122-Justin ziv Patil

records of only that student and if the student login after the time i.e now the time is set 9 am so before that the student are able to login but after it passes 9 thes students cannot login and it will not be recorded in their csv file data

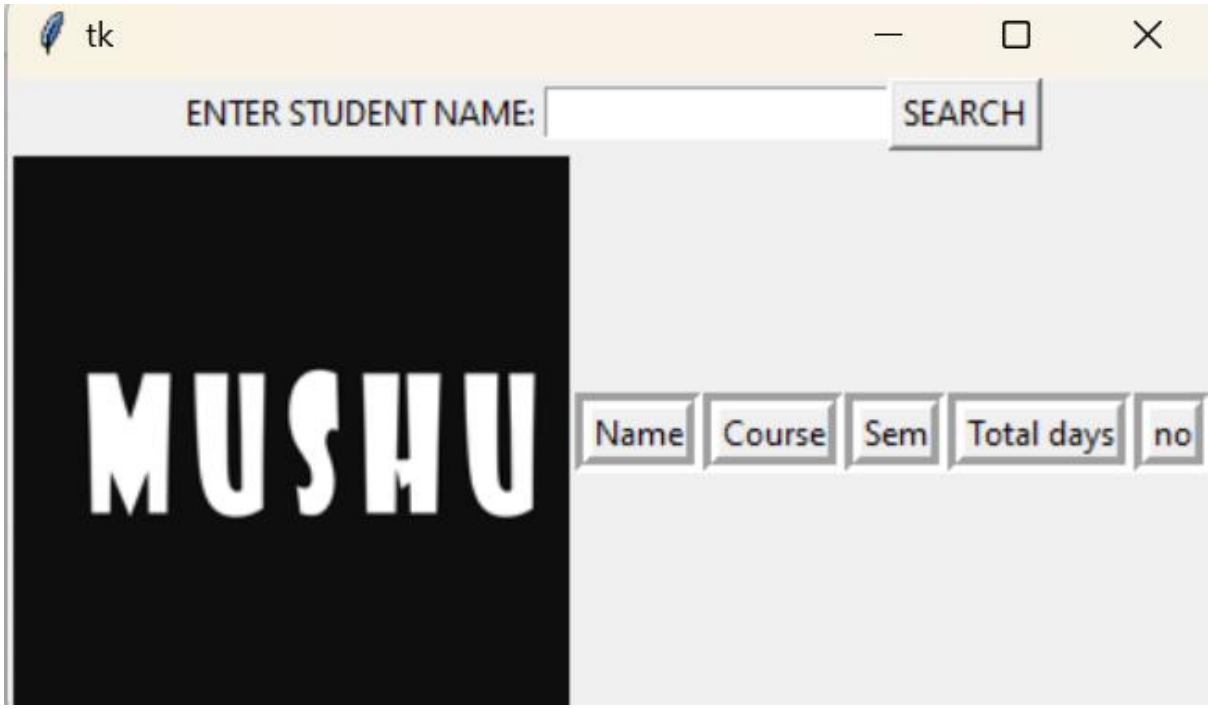
And if any unknown people try to login it will not add their record and will mention their name on screen as unkown

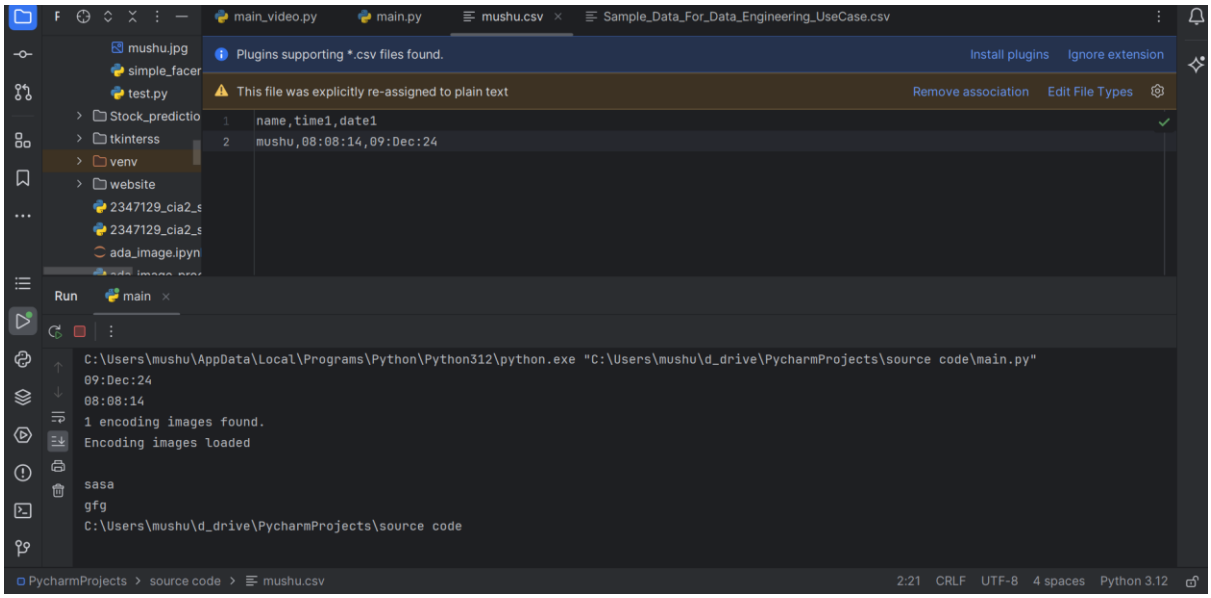


UI contains getting student detail using their names:

VisionHack 2024

2347129-Mushabbar Ahmed
2347122-Justin ziv Patil





Conclusion

The project effectively leverages the power of Python libraries to deliver an accurate, user-friendly, and efficient attendance system. It integrates real-time face detection with a robust backend for attendance logging. The project is

scalable for educational institutions, workplaces, and other settings, demonstrating the practical application of machine learning and computer vision.