**AI-Powered Automated Attendance System with Face Recognition for Students**

**Mohammedalmahdi Abdulhameed Majeed**

Department of Electronic and Communications Engineering,

Faculty of Engineering, University of Kufa, Iraq.

**Abstract**

This project implements a real-time face recognition-based attendance system using OpenCV, face\_recognition, and pandas libraries. It captures student images using a webcam, identifies their faces using face recognition, and updates their attendance records in an Excel spreadsheet.

**Keywords:** Face Detection, Face Recognition, OpenCV, cv2, Attendance system

**1. Introduction**

The attendance registration system is of great importance to people of this age and in various fields, the most important of which are the fields of work and learning, Where attendance is checked in different places and times.

There are two ways to register attendance:

1) Manual Attendance System (MAS).

2) Automated Attendance System (AAS).

***1.1 History of the Attendance Registration System***

In the previous century, the attendance registration system was done manually by writing the names of attendees in a special archive for the school, university, company, etc.

And this is the traditional way that people depend on to this day,

But this method has its advantages and disadvantages, one of its advantages is that it is a traditional and old method and depends on it a lot because it is a manual method that is not subject to the intervention of the machine and computer, but disadvantages are that it takes a lot of time and disperses the work of the attendees and bears a lot of errors and can easily manipulate the results of attendance because it is often manually recorded only once.

This system evolved in the late twentieth century and the beginning of the twenty-first century to become a system of recording attendance through fingerprints or fingers, but there are still those who follow the traditional manual method of recording attendance.

***1.2 Attendance Apps***

One of the most important applications for recording attendance in education areas such as primary and secondary schools and universities at various stages of studies.

It also occupies a large and most important space in workplaces such as companies and in the public and private sectors where the attendance registration system affects the economic factor of the company and thus the attendance registration system plays an important role.

***1.3 Automation of the Attendance Registration System***

In this paper, I will limit myself to attendance applications in universities and schools, and I will show you a new method, which is the method of recording students' attendance through the camera in the halls and calculating the time it took the student to attend.

Taking into consideration the time, date, and name of the current lecture and calculating all of that.

***1.4. Face Detection Systems for Attendance Management***

How to detect faces and record attendance through OpenCV and machine learning

Face detection technology is done through a special algorithm called the HOG brain.

The students' photos are stored in a specific file in addition to their names, and when the student comes to be opposite the camera, he determines the person's image and compares it with the image of him in the database, if the image matches the face, then the attendance of this student is recorded in a file in xlsx format.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S. No.** | **Existing System** | **Features** | **Benefits** | **Limitations** |
| 1. | Automated attendance management system using face recognition | Use Eigenfaces for Recognition | High accuracy | Multiple faces were not recognized. |
| 2. | Face recognition attendance system by nevon | Stores the faces that are detected and automatically marks attendance | Used for security purposes | Don't recognize properly in poor light. |
| 3. | Smart Attendance System using OpenCV based on Facial Recognition | Takes pictures through the webcam and creates a dataset for users using m images. Takes real-time images and marks attendance | Used for marking attendance in schools and colleges. | Cannot mark attendance of the student on a remote server database. |
| 4. | Smart Attendance Management System Using Face Recognition) | In this, the data is stored in a sorted manner so that it can easily be accessible | Required high-definition camera |  |
| 5. | Face Recognition - A Tool for Automated Attendance | Face detection, Pre-processing, Feature extraction, and Classification stages | High accuracy | The camera should be attached at a specific position |
| 6. | Smart Application For AMS Using Face Recognition/ Student Attendance System in Classroom Using Face Recognition Technique | Uses CCTV and Android mobile | A 3D face recognition algorithm is used | Android phone is expensive and detects one face at a time |
| 7. | Attendance System based on Face Recognition using Eigen face and PCA Algorithms [10] | In this Illumination invariant algorithm is used | The problem of light intensity problem and head pose was overcome. | Masked faces were not recognized. |
| 8. | Attendance System Using Face Recognition and Class Monitoring System | Open CV python library is used and Mysql is used for database | This method is fast and secure and has a low false positive rate. | The recognition rate is lower |
| 9. | Algorithm for Efficient Attendance Management: Face Recognition based approach2 | Multiple faces can be detected at a time and no special hardware is needed | Accuracy is low only 50% of faces were recognized |  |
| 10. | Median filter and skin classification is used |  |  |  |

Table (1) shows the advantages and limitations of face detection systems for attendance management

**2. Method**

***2.1 software requirements***

***2.1.1 IDE software and programming languages required***

This project is mainly based on the Python programming language

* You can download the Python programming language by visiting the marginalized site (2).
* And download the Anaconda program through the marginalized site (3).
* And download the free version of PyCharm through the marginalized site (4)
* And download the Microsoft Visual Studio Installer through the marginalized site (5)

***2.1.2 Required Libraries in Python***

Datatime, os, cv2,

numpy, face-recognition

, pandas, openpyxl,

dlib, cmake

you can install these libraries by opening cmd and writing the following command

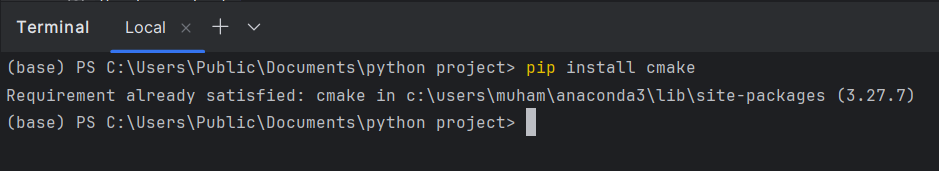
pip install ‘the name of the library’. 

fig.1 shows how to install libraries by Pycharm's terminal

Or you can install these packages from PyCharm by following these steps:  
1. Open PyCharm.

2. Go to the main menu or click on ‘Alt + \’.

3. Go to settings or click on ‘Ctrl + Alt + S’.

4. Go to ‘Project: python project’ then ‘Python Interpreter’.

5. Click on the plus bottom write the name of the libraries and install it.

Before installing cmake you should install the Microsoft Visual Studio installer and download Desktop development with C++, from Visual Studio Community for free.

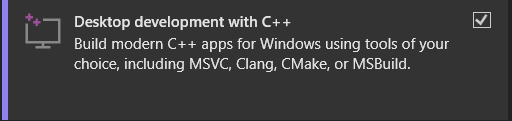


Fig.2 shows the requirements of the cmak library.

***2.1.3 Project file***

Create project file, This file consists of a file that contains clear pictures of students' faces, and these pictures are called IDs for each student that are not repeated, and the basic Python file,

As well as an Excel file containing the IDs of each student and the name of the student in front of his ID.

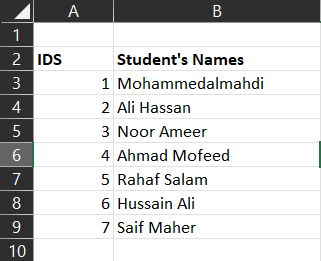
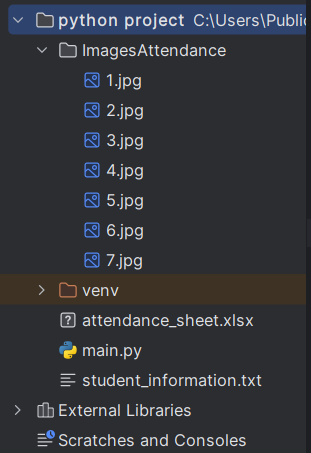


Fig.3 shows how directories should be. Fig.4 shows how the Excel sheet should be.

***2.1.4 flowchart of the coed***

Start

|

|--> Import necessary libraries

|

|--> Define functions:

| |--> read\_Images(path): Reads images from a specified path

| |--> student\_name(file\_path): Extracts student names from an Excel file

| |--> findEncodings(images): Encodes faces in a list of images

| |--> write\_time\_to\_excel(file\_path, column): Records the current time in an Excel file

| |--> write\_date\_to\_excel(file\_path, column): Records the current date in an Excel file

| |--> markAttendance(name, xlsx\_file): Marks attendance for a student in an Excel file

|

|--> Set path to image directory and load attendance sheet

|

|--> Read images from the image directory

|

|--> Encode faces in the read images

|

|--> Initialize video capture device

|

|--> Extract student names from the attendance sheet

|

|--> Loop:

| |--> Capture a frame from the video capture device

| |--> Resize the captured frame

| |--> Convert the resized frame to RGB color space

| |--> Detect faces in the resized frame

| |--> Encode faces in the detected faces

| |--> Record current time and date in the attendance sheet

| |--> Loop through each encoded face:

| |--> Compare the current encoded face with all known encoded faces

| |--> If a match is found:

| |--> Draw a rectangle around the matched face

| |--> Display the student's name on the frame

| |--> Mark attendance for the student

|

|--> Check for user input (press 'q') to exit

|

|--> Release video capture device

|

|--> Close all windows

|

End

***2.1.5 Functions Explanations***

* read\_Images(path)  
  This function reads all images from the images file directory and returns them as a list of NumPy list (the list represents the pixels of the image as a matrix of integers numbers).
* students\_name(file\_path)  
  This function reads the Excel sheet file using Pandas library and returns a list of student’s names.
* findEncodings(images)  
  This function takes the image pixels matrix list and returns the encoding of every image as a list of NumPy arrays by using the face\_recognition library.
* write\_time\_to\_excel(file\_path, column=3)  
  This function takes the Excel file sheet and the number of the column to start with, the default column is 3 and in the first row, then writes the current time in a specific cell in the Excel sheet, this function uses datatime and openpyxl libraries to work.
* write\_date\_to\_excel(file\_path, column=3)  
  This function takes the Excel file sheet and the number of the column to start with, the default column is 3 and the second row writes the current date in a specific cell in the Excel sheet, this function uses datatime and openpyxl libraries to work.
* markAttendance(name, xlsx\_file)  
  This function takes the names of the students as an input parameter, It marks attendance for the specific student in the Excel sheet finds the row corresponding to the student's name, and marks attendance for the current date in that row.

***2.1 Hardware requirements***

The hardware that has been used in the project is a computer and webcam, in another case camera with a computer or a camera with Raspberry Pi.

**3. Results and Discussion**

Each name has its own hands and each picture in the photo file is identical to the student's ID

As every student has an ID and a picture with the same hand

Thus, the student's ID will be taken and matched with the image that bears the same ID, so if the image is identical to what the student's face is during the operation of the program and the recognition of the face, attendance will be recorded with the code P on the student himself.

***3.1 Before taking attendance***

The image 1.jpg represent Mohammed’s face

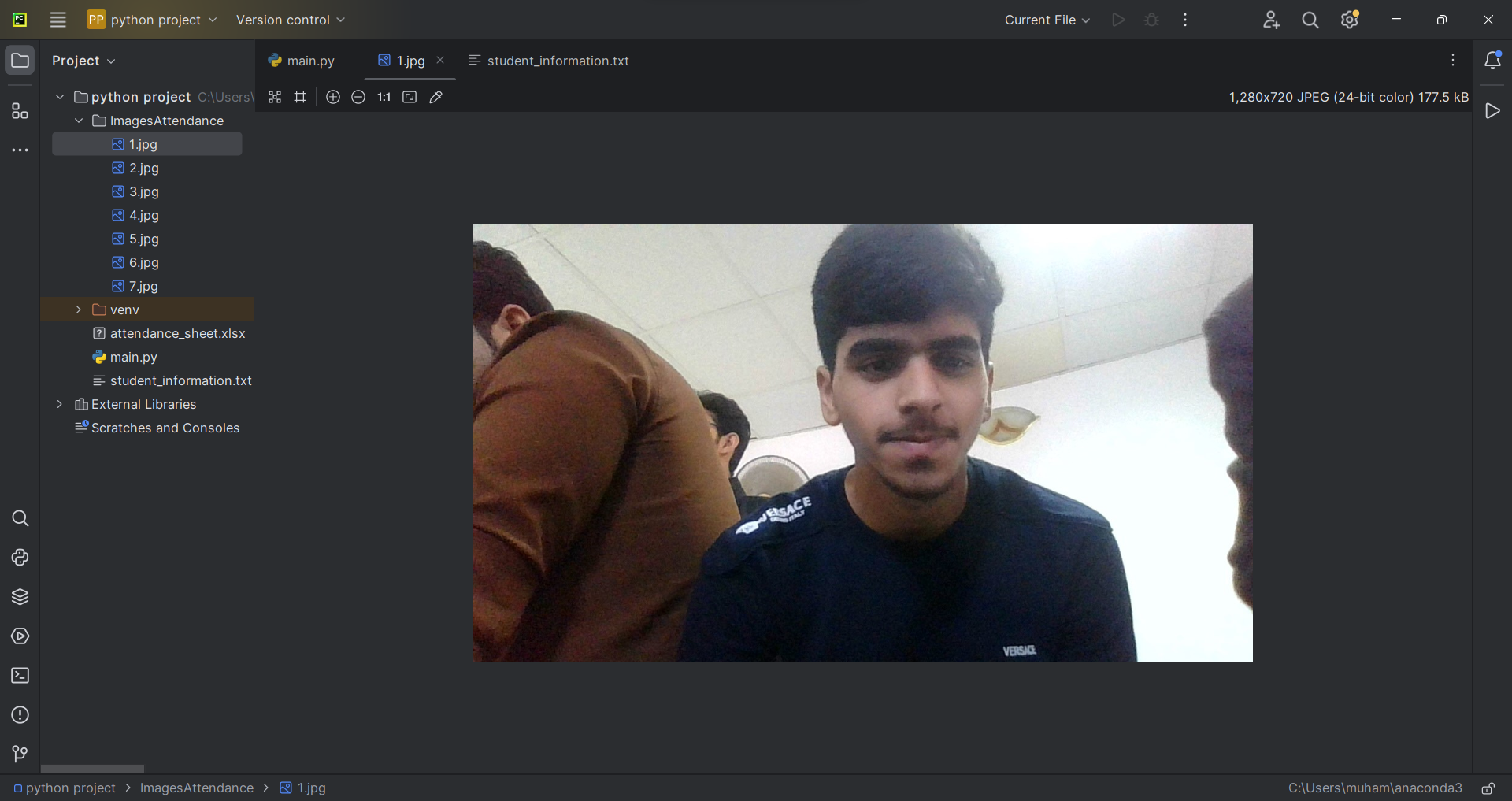
****

Fig.5 This is one of the stored feces images in the ImagesAttendance file.

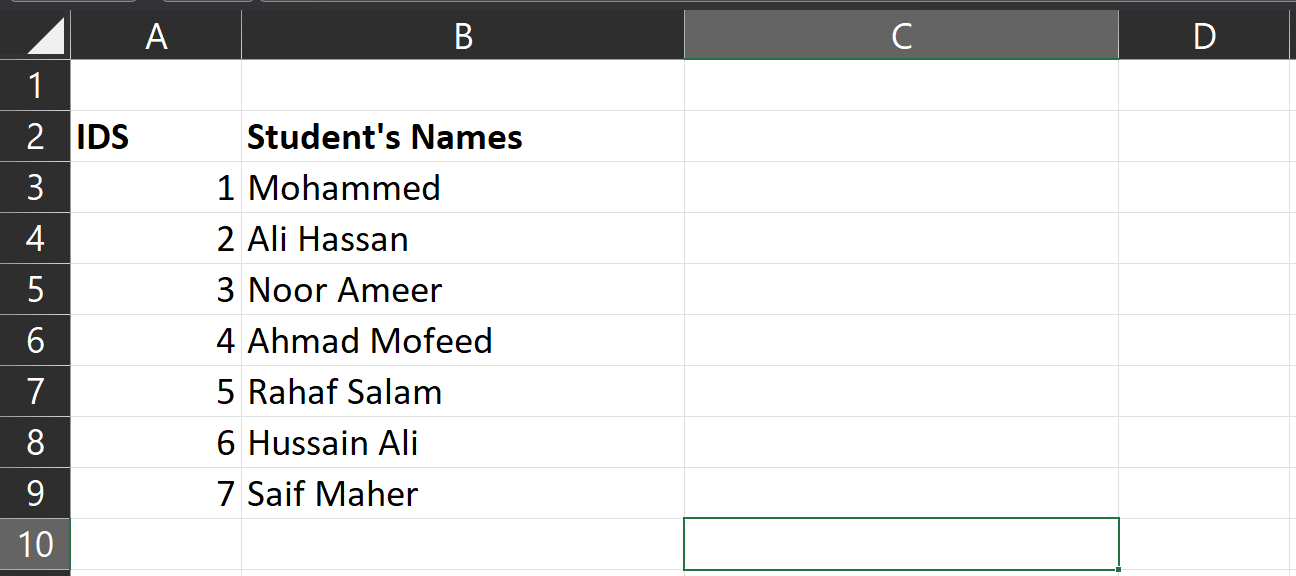
****

Fig.6 This is the Excel sheet before taking attendance.

***3.2 After taking attendance***

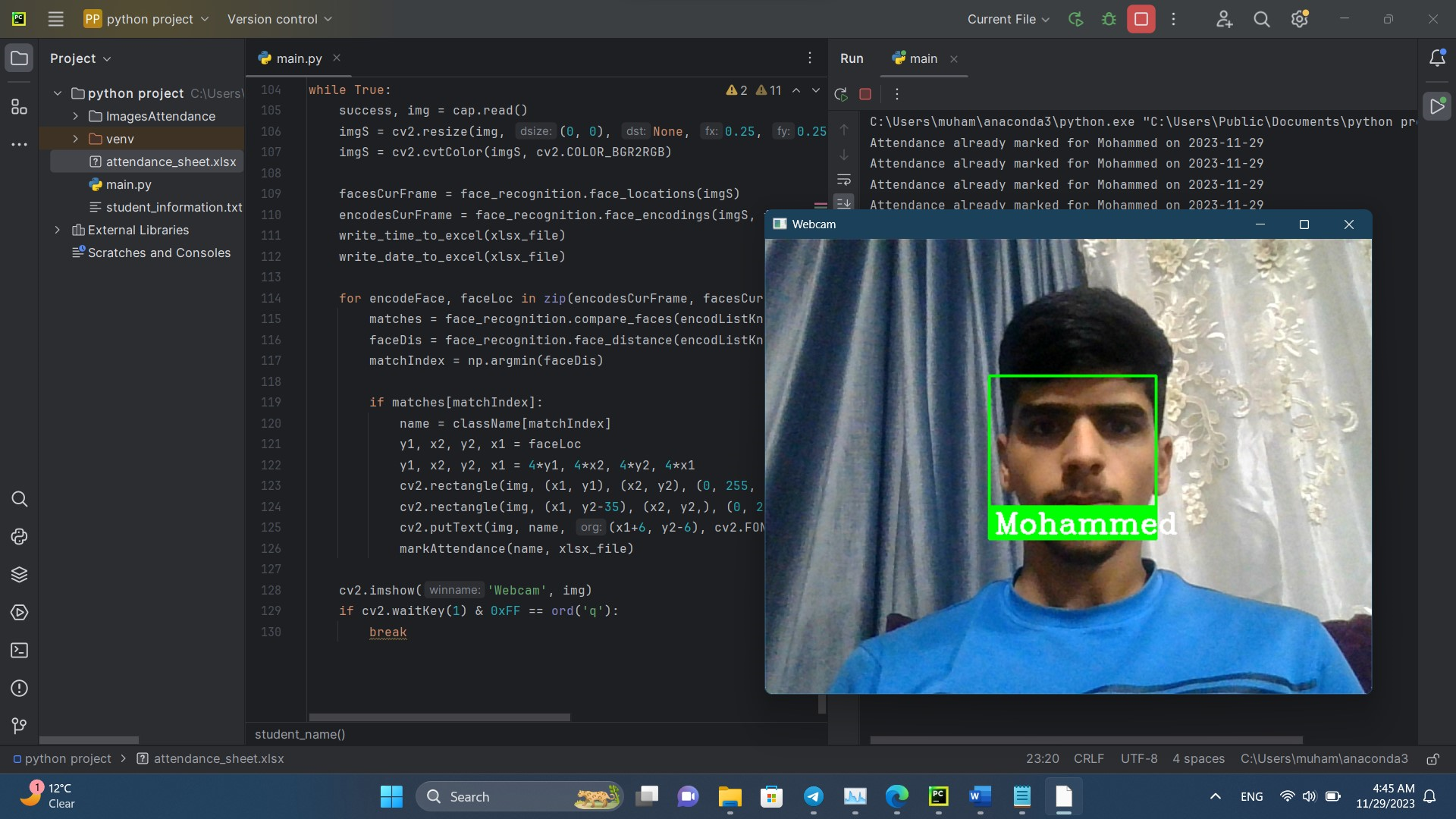


Fig.7 This is the Execution of the code and take attendance for Mohammed.

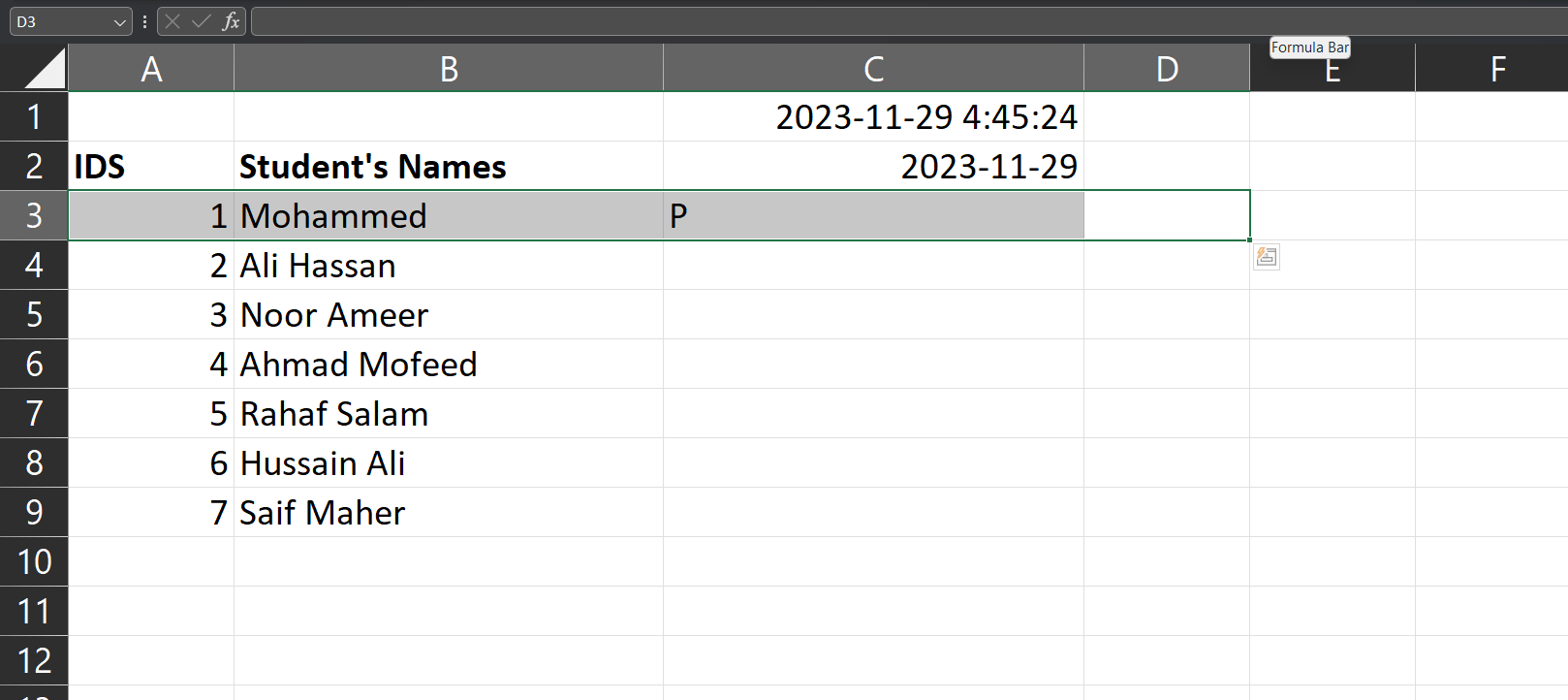
****

Fig.8 This is the Excel sheet after taking attendance for Mohammed.

**4. Conclusion**

The system that you made to register attendance is characterized by the fact that the entries such as the names of the students come from the Excel sheet file,

I have tried a lot to create a web page tool for the project, but it is difficult for developers who are not familiar with programming languages that deal with web pages and how to deal with them to create a web page, even if it is simply entering student information and uploading their photos.

At first, I created a web page only using Python and HTML, but it looks very simple and traditional and does not contain any features

Then I tried to bring a ready-made and free tool template from some sites that provide beautiful and wonderful web page templates, but I needed to know how to modify those templates to suit me, and for this, the developer must know programming languages such as HTML, CSS and JS  
To change those interfaces.

In general, this project depends on three basic variables, namely the algorithm used to recognize faces, the method of entering data and images, and finally the method of displaying the results and additions that can be added to the results.

I chose the Excel sheet file because it seemed the easiest and fastest to register attendance and it is the closest to use in colleges and schools.

**Appendix**

1. The code of the project:

import datetime  
import os  
import cv2  
import numpy as np  
import face\_recognition  
import pandas as pd  
import openpyxl  
  
  
def read\_Images(path):  
 images = [] *# empty list will contain all images in the images file as a matrix of pixels* myList = os.listdir(path) *# list of all directory files in the 'path'* for cl in myList:  
 curImg = cv2.imread(f'{path}/{cl}')  
 images.append(curImg)  
 return images  
  
  
def student\_name(file\_path):  
 df = pd.read\_excel(file\_path)  
 students = df.iloc[1:, 1].tolist()  
 return students  
  
  
def findEncodings(images):  
 *"""  
 :param images:  
 :return: list of the encoding for a list of images  
 """* encodeList = []  
 for img in images:  
 img = cv2.cvtColor(img, cv2.COLOR\_BGR2RGB)  
 encode = face\_recognition.face\_encodings(img)[0]  
 encodeList.append(encode)  
 return encodeList  
  
  
def write\_time\_to\_excel(file\_path, column=3):  
  
 *# Load the workbook* workbook = openpyxl.load\_workbook(file\_path)  
 sheet = workbook.active  
  
 *# Get the current time* current\_time = datetime.now()  
  
 *# Write the current time to the specified cell* sheet.cell(row=1, column=column, value=current\_time)  
 workbook.save(file\_path)  
  
  
def write\_date\_to\_excel(file\_path, column=3):  
  
 *# Load the workbook* workbook = openpyxl.load\_workbook(file\_path)  
 sheet = workbook.active  
  
 *# Get the current date* current\_date = datetime.now().date()  
  
 *# Write the current date to the specified cell* sheet.cell(row=2, column=column, value=current\_date)  
 workbook.save(file\_path)  
  
  
def markAttendance(name, xlsx\_file):  
 *# Load the workbook and sheet* workbook = openpyxl.load\_workbook(xlsx\_file)  
 sheet = workbook.active  
  
 *# Get the current date (assuming today's date)* current\_date = datetime.now().date()  
  
 *# Find the row corresponding to the student's name* for row in range(2, sheet.max\_row + 1):  
 student\_name\_cell = sheet.cell(row=row, column=2)  
 if student\_name\_cell.value == name:  
 break  
  
 *# If the student's name is found, mark attendance* if student\_name\_cell.value == name:  
 *# Check if attendance has already been marked for the current date* attendance\_cell = sheet.cell(row=row, column=current\_date.weekday() + 2)  
 if attendance\_cell.value is None:  
 *# Mark attendance by writing 'P' in the corresponding cell* attendance\_cell.value = 'P'  
 workbook.save(xlsx\_file)  
 print(f"Attendance marked for {name} on {current\_date}")  
 else:  
 print(f"Attendance already marked for {name} on {current\_date}")  
 else:  
 print(f"Student '{name}' not found in the attendance sheet")  
  
  
xlsx\_file = 'attendance\_sheet.xlsx'  
path = 'ImagesAttendance' *# the path of the file that contain the images*images = read\_Images(path)  
encodListKnown = findEncodings(images)  
cap = cv2.VideoCapture(0)  
className = student\_name(xlsx\_file)  
  
  
while True:  
 success, img = cap.read()  
 imgS = cv2.resize(img, (0, 0), None, 0.25, 0.25)  
 imgS = cv2.cvtColor(imgS, cv2.COLOR\_BGR2RGB)  
  
 facesCurFrame = face\_recognition.face\_locations(imgS)  
 encodesCurFrame = face\_recognition.face\_encodings(imgS, facesCurFrame)  
 write\_time\_to\_excel(xlsx\_file)  
 write\_date\_to\_excel(xlsx\_file)  
  
 for encodeFace, faceLoc in zip(encodesCurFrame, facesCurFrame):  
 matches = face\_recognition.compare\_faces(encodListKnown, encodeFace)  
 faceDis = face\_recognition.face\_distance(encodListKnown, encodeFace)  
 matchIndex = np.argmin(faceDis)  
  
 if matches[matchIndex]:  
 name = className[matchIndex]  
 y1, x2, y2, x1 = faceLoc  
 y1, x2, y2, x1 = 4\*y1, 4\*x2, 4\*y2, 4\*x1  
 cv2.rectangle(img, (x1, y1), (x2, y2), (0, 255, 0), 2)  
 cv2.rectangle(img, (x1, y2-35), (x2, y2,), (0, 255, 0), cv2.FILLED)  
 cv2.putText(img, name, (x1+6, y2-6), cv2.FONT\_HERSHEY\_COMPLEX, 1, (255, 255, 255), 2)  
 markAttendance(name, xlsx\_file)  
  
 cv2.imshow('Webcam', img)  
 if cv2.waitKey(1) & 0xFF == ord('q'):  
 break

**Acknowledgments**

This work was supported by the Electronic and Communications Department, Faculty of Engineering, University of Kufa, Iraq.

**References**

1. Ali, A., Mandhanya, R., Birla, S., Mandloi, U., & Jain, V. (2021). Automatic face recognition attendance system using Perl and OpenCV. GRD Journal for Engineering, 6(4), 49-55.
2. <https://www.python.org/downloads/>
3. https://www.anaconda.com/
4. <https://www.jetbrains.com/edu-products/download/>
5. <https://visualstudio.microsoft.com/thank-you-downloading-visual-studio/?sku=Community&channel=Release&version=VS2022&source=VSLandingPage&cid=2030&passive=false>
6. murtazahassan.com
7. datetime: https://docs.python.org/3/c-api/datetime.html
8. os: https://docs.python.org/3/library/functions.html
9. cv2: https://github.com/opencv/opencv
10. numpy: https://numpy.org/doc/stable/
11. face\_recognition: https://github.com/trygvea/face-recognition-dlib
12. pandas: https://pandas.pydata.org/docs/
13. openpyxl: https://openpyxl.readthedocs.io/en/3.0/