

A
Mini Project
On
ATTENDANCE REGISTRATION USING FACE RECOGNITION
(Submitted in partial fulfillment of the requirements for the award of Degree)

BACHELOR OF TECHNOLOGY
In
COMPUTER SCIENCE AND ENGINEERING
By
RACHAMALLA SINDHUJA(197R1A05P4)
KOTA VENKATA NIKHIL(197R1A05M1)
G.J MANIKANTA VARMA(197R1A05K8)

Under the Guidance of
B.P DEEPAK KUMAR
(Assistant Professor)



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
CMR TECHNICAL CAMPUS
UGC AUTONOMOUS

(Accredited by NAAC, NBA, Permanently Affiliated to JNTUH, Approved by AICTE, New Delhi) Recognized Under
Section 2(f) & 12(B) of the UGC Act, 1956, Kandlakoya (V), Medchal Road, Hyderabad-501401.
2019-2023

ABSTRACT



Attendance is a compulsory requirement of every organization. Maintaining the attendance register daily is a difficult and time-consuming task. This project provides an efficient and smart method for marking attendance.

As it is known that the primary identification for any human is their face. The subject of face recognition is as old as computer vision. Despite the fact that other methods of identification (such as fingerprints, or iris scans) can be more accurate, face recognition has always remained a major focus of research because of its noninvasive nature.

This system overcomes the ambiguities like fake attendance, high cost, and time consumption. This system uses a face recognizer library for facial recognition and storing attendance. The system will record the attendance automatically and it will provide the facilities to the absentee's supervisor or the faculty to access the information easily.

EXISTING SYSTEM



Manual entry.

- Here the attendance will be marked manually, in a physical register. This can be a tedious job to maintain the record for every individual manually. The human effort is more here.
- The retrieval of the information is also done manually as the records are maintained in the handwritten registers.
- The accuracy of the data collected is the biggest issue, as the attendance of a particular person can be taken by a third party without the realization of the institution or the organization.

Biometric scanners.

- Attendance is recorded by scanning the biometric features like the fingerprint or iris of an individual.
- This is an invasive method of marking attendance. (this system may fail to provide safety in a contagious environment)
- A special Biometric scanner needs to be installed on the campus.

DISADVANTAGES

- **Not User Friendly**
- **Difficulty in report generating**
- **Manual Control**
- **Time consuming**
- **Accessibility of the information by the legitimate concerned party**
- **Data can be lost**
- **There is a risk of human error**

PROPOSED SYSTEM

- The task of the proposed system is to capture the face of each person and to store it in the database.
- The face of the person will be captured in such a manner that all the features of the person's face will be detected.
- There is no need for the supervision of the third party to manually take attendance.
- The system takes in a live video stream as input and through further processing the faces are recognized and the attendance database is updated.
- This system is developed using the python **OpenCV library** and the **face_recognition()** package is used.

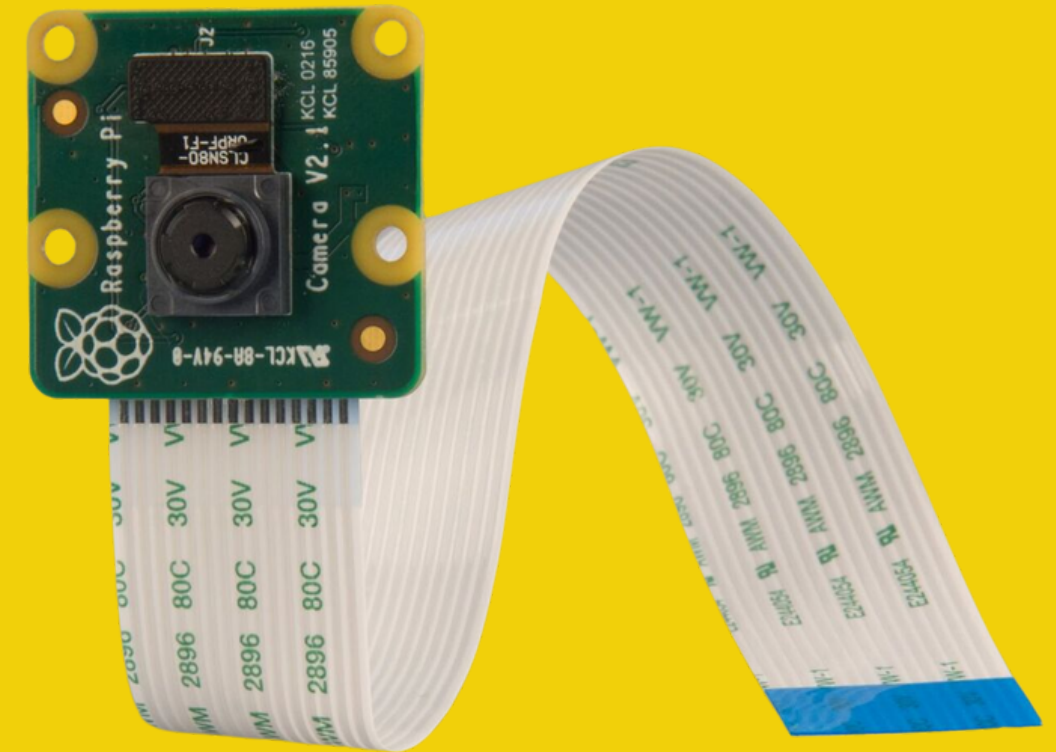


ADVANTAGES

- **The speed of the attendance recording process is faster than the previous system.**
- **Able to recognize the face of an individual accurately based on the already known faces database**
- **Provide a simple spreadsheet for admins to access the attendance database.**
- **The system is completely automated with minimum to no human intervention.**
- **Smart Integration where automatically, emails are sent to the absentees and/or the concerned supervisor or guardian.**

HARDWARE REQUIREMENTS:

- **Camera Module with good resolution**
- **Power Supply**



SOFTWARE REQUIREMENTS:

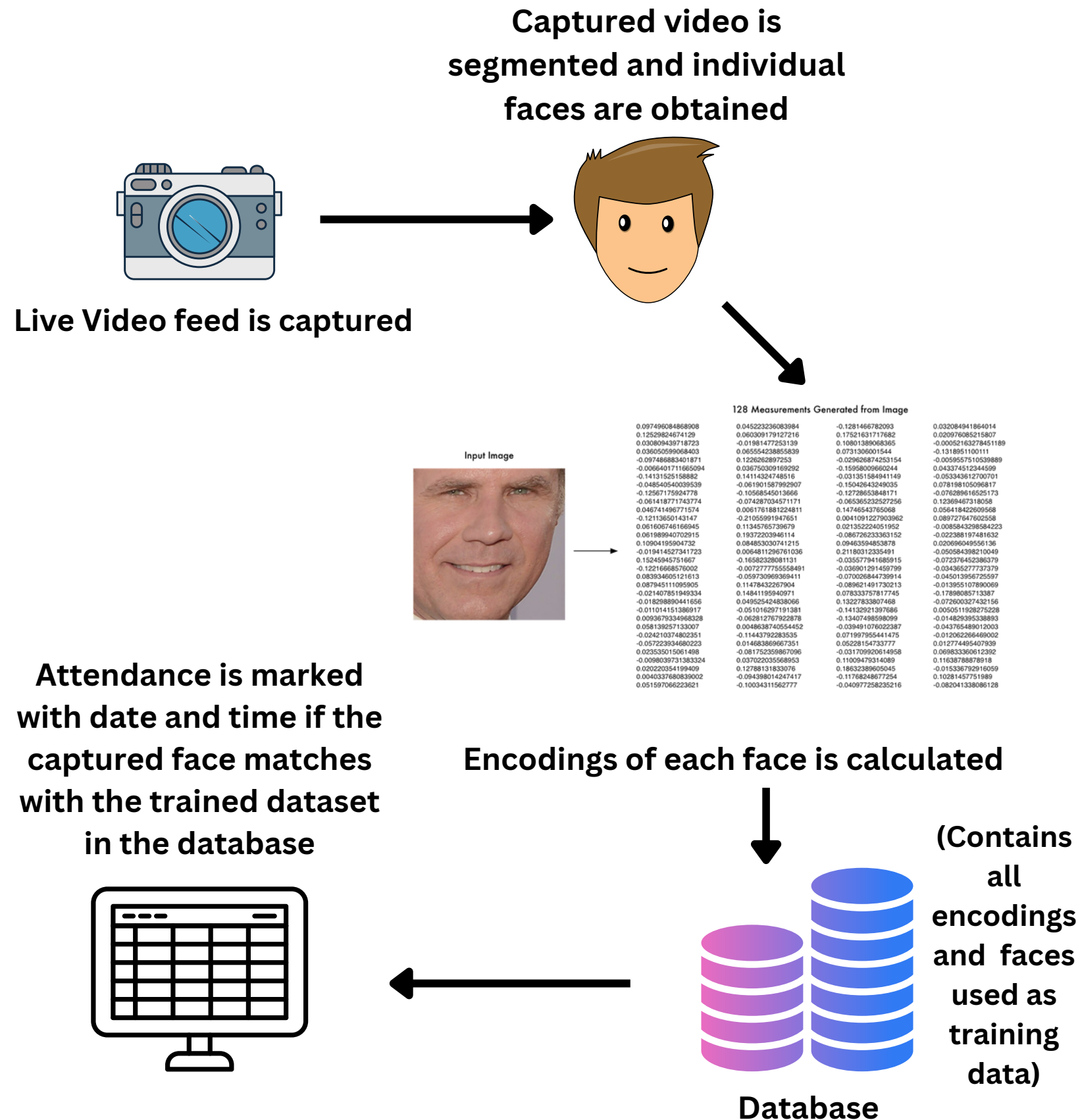


- **Python**
- **OpenCV (Open-source computer vision) is a library of programming functions mainly aimed at real-time computer vision**
- **Gcc (C++ compiler which is required by the dlib package)**
- **face_recognition() library**

NOVELTY:

The novelty of this project (Attendance Registration using Face Recognition) is that we are using the openCv library and face_recognition() package which provides the features such as finding faces, manipulating and identifying faces in the images with high accuracy. This project introduces all these features in real time allowing us to mark attendance. This system helps to build an attendance system which is based on face recognition technique. Here the face of an individual will be considered for marking attendance. In the domain of computer vision and image processing, face recognition is gaining more popularity and has been widely used and further developments are being made. We proposed a system which detects the faces of students from live streaming video of the classroom and attendance will be marked if the face detected is found in the recorded database.

ARCHITECTURE



The architecture for the proposed system has been designed to keep it pretty straightforward and easy to understand. The steps that must be undertaken to reach the final end step of the system is ensuring the attendance of the student is updated correctly and timely. The system can easily be accessed by anyone, and students' attendance can easily be checked and maintained by the faculty when required. An inbuilt camera module or any USB-based camera module can be used for capturing live video feeds of the class and simultaneously performing recognition for the students. OpenCV-Python will be used to access the camera and the face-recognition library will be used to obtain the encodings of the captured face. then recognition and matching of the captured faces against the stored faces available in the previously acquired data sets are done to finally update the attendance.



MODULES

Importing Libraries

```
1  import cv2
2  import numpy as np
3  import face_recognition
4  import os
5  from datetime import datetime
```



MODULES

Finding the no. of Unique classes

```
9 path = 'Training_images'
10 images = []
11 classNames = []
12 myList = os.listdir(path)
13 print(myList)
14 for cl in myList:
15     curImg = cv2.imread(f'{path}/{cl}')
16     images.append(curImg)
17     classNames.append(os.path.splitext(cl)[0])
18 print(classNames)
```



MODULES

Access camera and prepare images

```
52 encodeListKnown = findEncodings(images)
53 print('Encoding Complete')
54
55 cap = cv2.VideoCapture(0)
56
57 while True:
58     success, img = cap.read()
59     # img = captureScreen()
60     imgS = cv2.resize(img, (0, 0), None, 0.25, 0.25)
61     imgS = cv2.cvtColor(imgS, cv2.COLOR_BGR2RGB)
62
63     facesCurFrame = face_recognition.face_locations(imgS)
64     encodesCurFrame = face_recognition.face_encodings(imgS, facesCurFrame)
```



MODULES

Find Encodings

```
21 def findEncodings(images):  
22     encodeList = []  
23  
24     for img in images:  
25         img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)  
26         encode = face_recognition.face_encodings(img)[0]  
27         encodeList.append(encode)  
28     return encodeList
```

MODULES

Match detected faces with Training Images

```
65     for encodeFace, faceLoc in zip(encodesCurFrame, facesCurFrame):
66         matches = face_recognition.compare_faces(encodeListKnown, encodeFace)
67         faceDis = face_recognition.face_distance(encodeListKnown, encodeFace)
68
69         matchIndex = np.argmin(faceDis)
70
71         if matches[matchIndex]:
72             name = classNames[matchIndex].upper()
73
74             y1, x2, y2, x1 = faceLoc
75             y1, x2, y2, x1 = y1 * 4, x2 * 4, y2 * 4, x1 * 4
76             cv2.rectangle(img, (x1, y1), (x2, y2), (0, 255, 0), 2)
77             cv2.rectangle(img, (x1, y2 - 35), (x2, y2), (0, 255, 0), cv2.FILLED)
78             cv2.putText(img, name, (x1 + 6, y2 - 6), cv2.FONT_HERSHEY_COMPLEX, 1, (255, 255, 255), 2)
79             markAttendance(name)
80
81     cv2.imshow('Webcam', img)
82     cv2.waitKey(1)
```

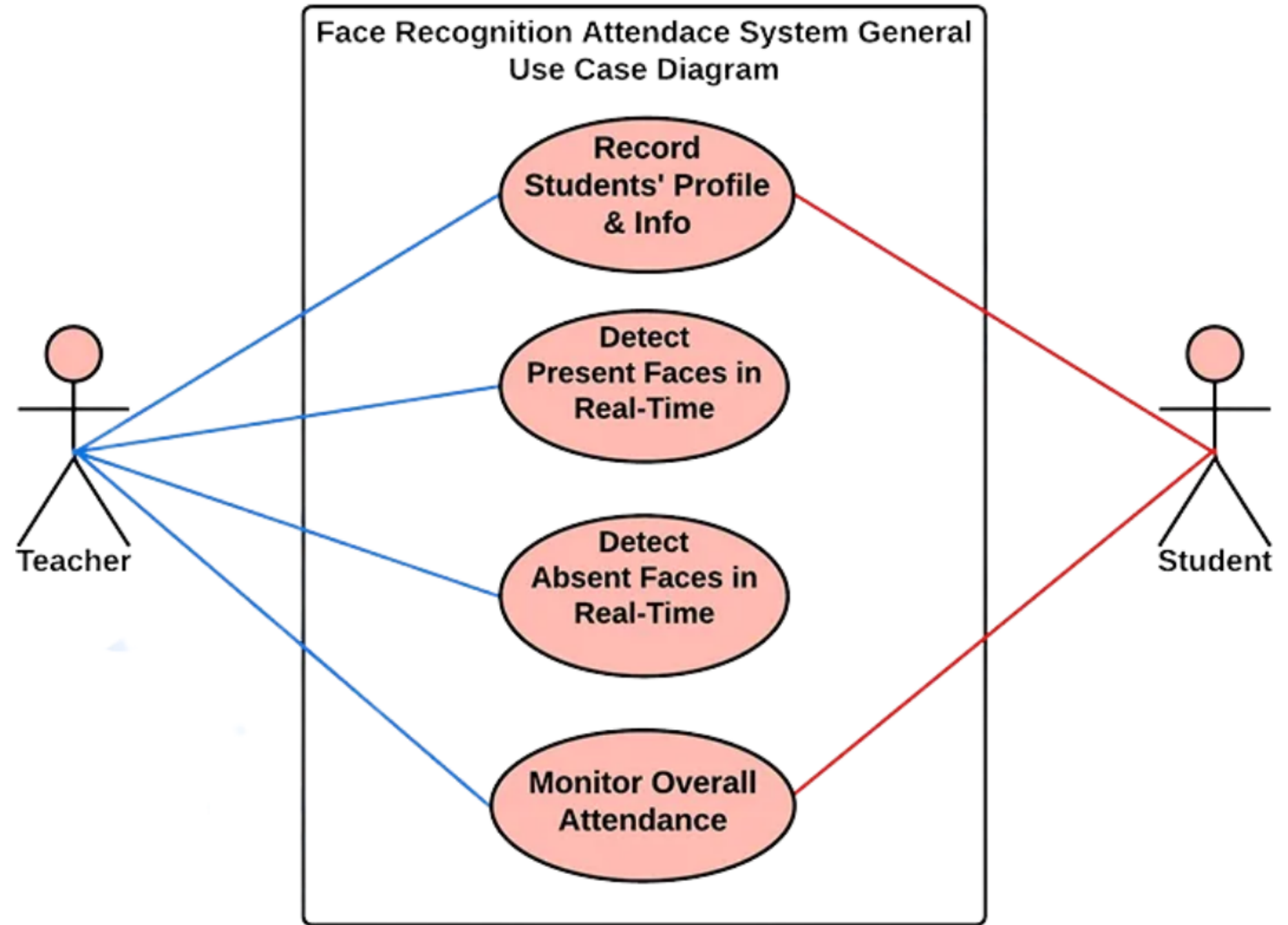



MODULES

Mark Attendance

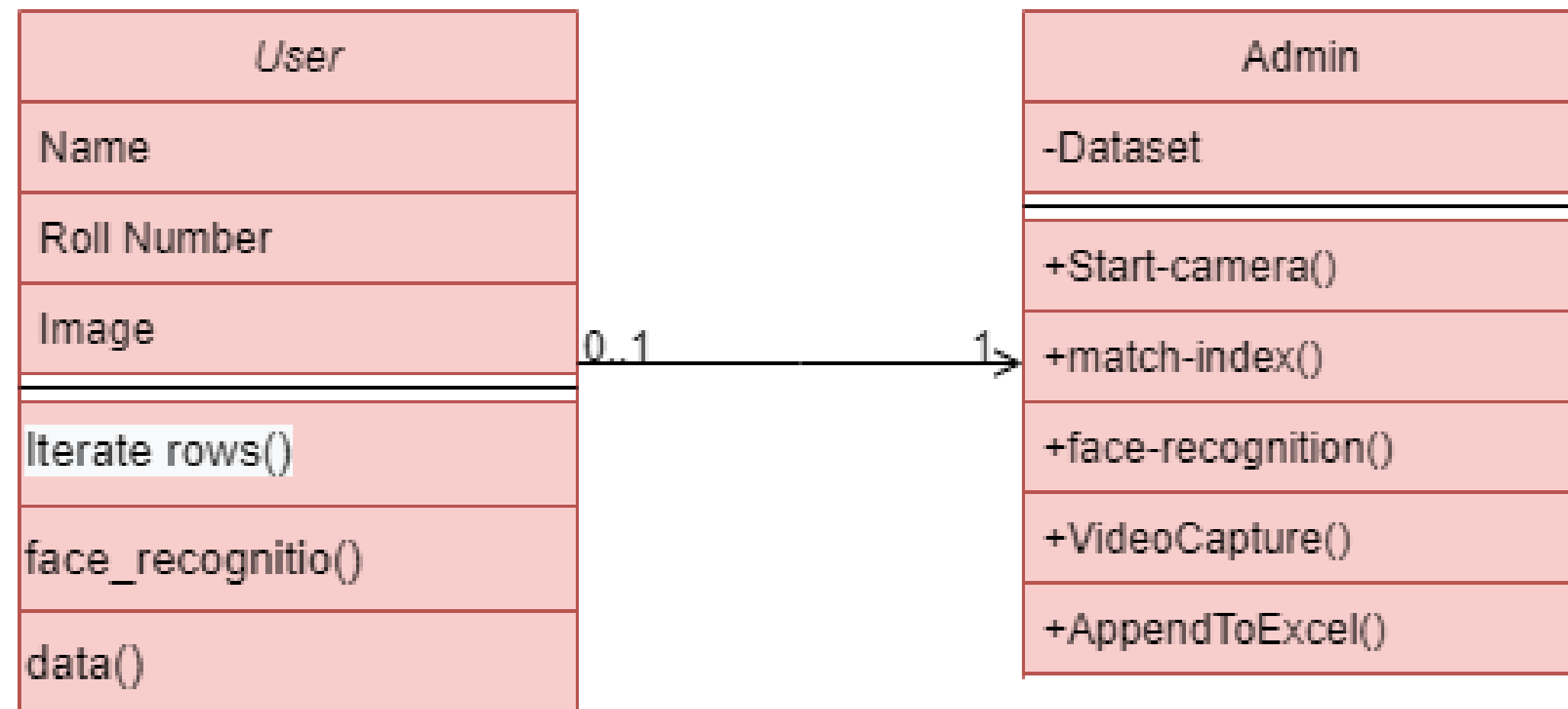
```
31 def markAttendance(name):
32     with open('Attendance.csv', 'r+') as f:
33         myDataList = f.readlines()
34
35         nameList = []
36         for line in myDataList:
37             entry = line.split(',')
38             nameList.append(entry[0])
39             if name not in nameList:
40                 now = datetime.now()
41                 dtString = now.strftime('%H:%M:%S')
42                 f.writelines(f'\n{name},{dtString}')
```

UML DIAGRAMS:

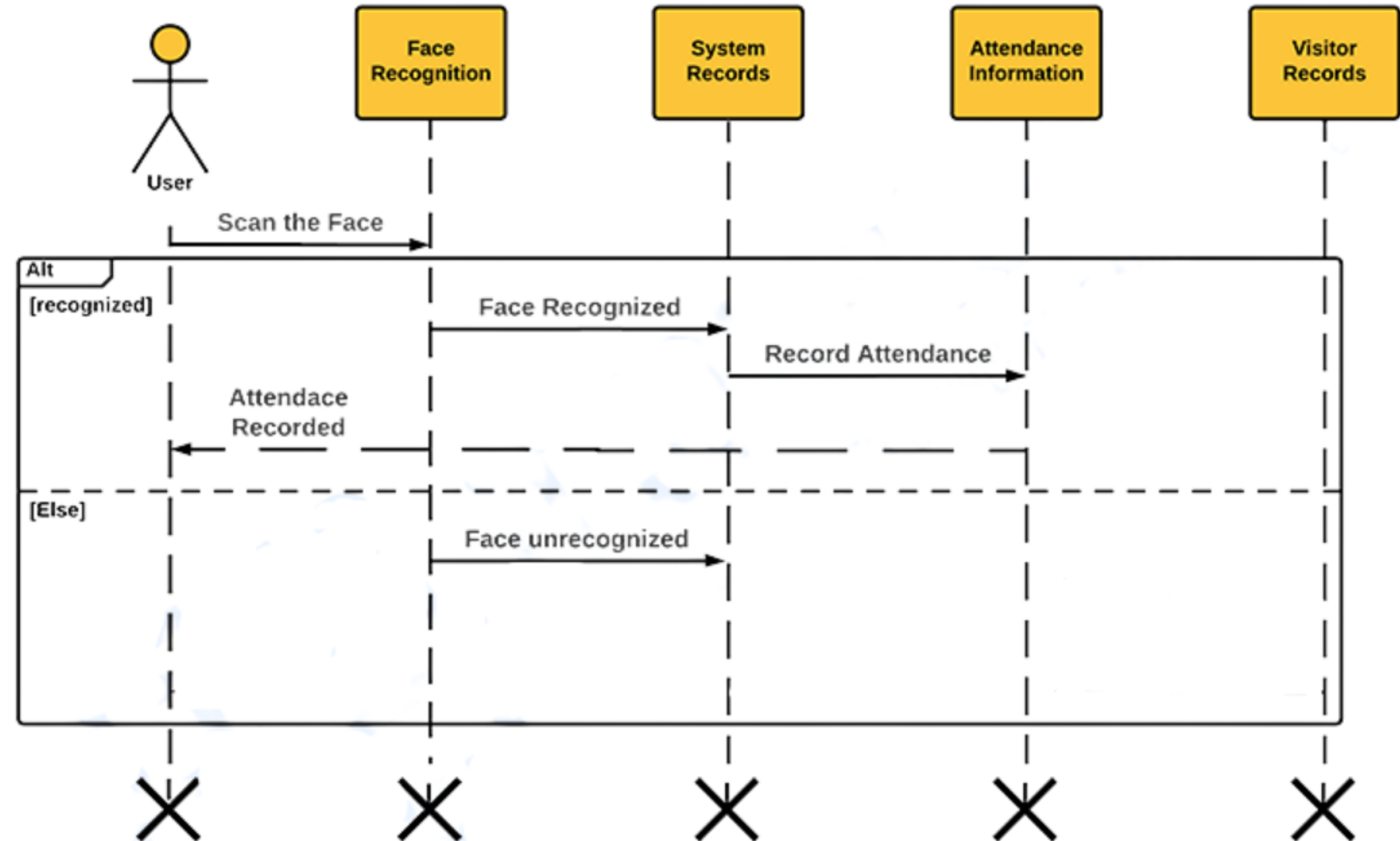


USE CASE DIAGRAM

UML DIAGRAMS:

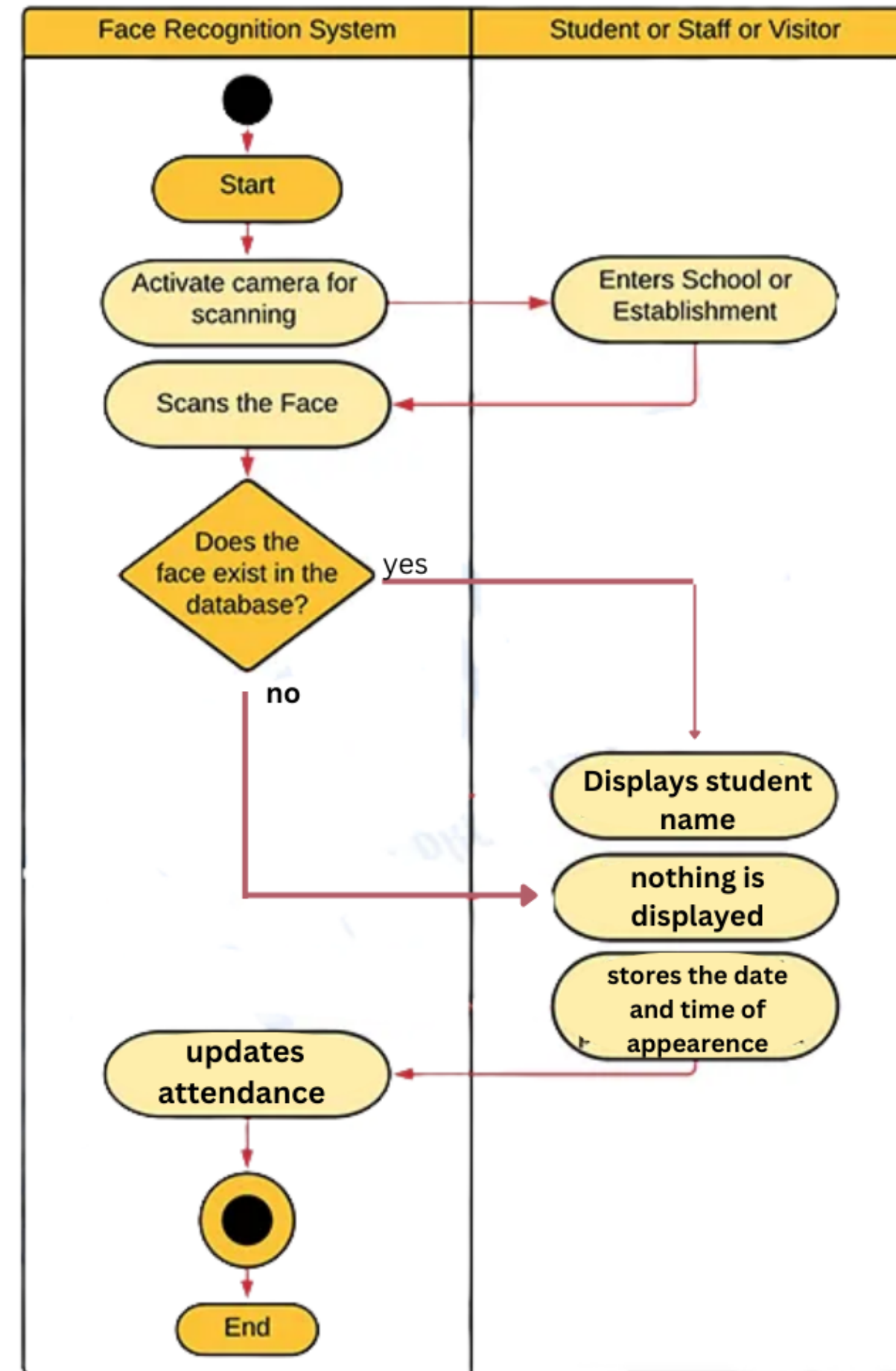


UML DIAGRAMS:



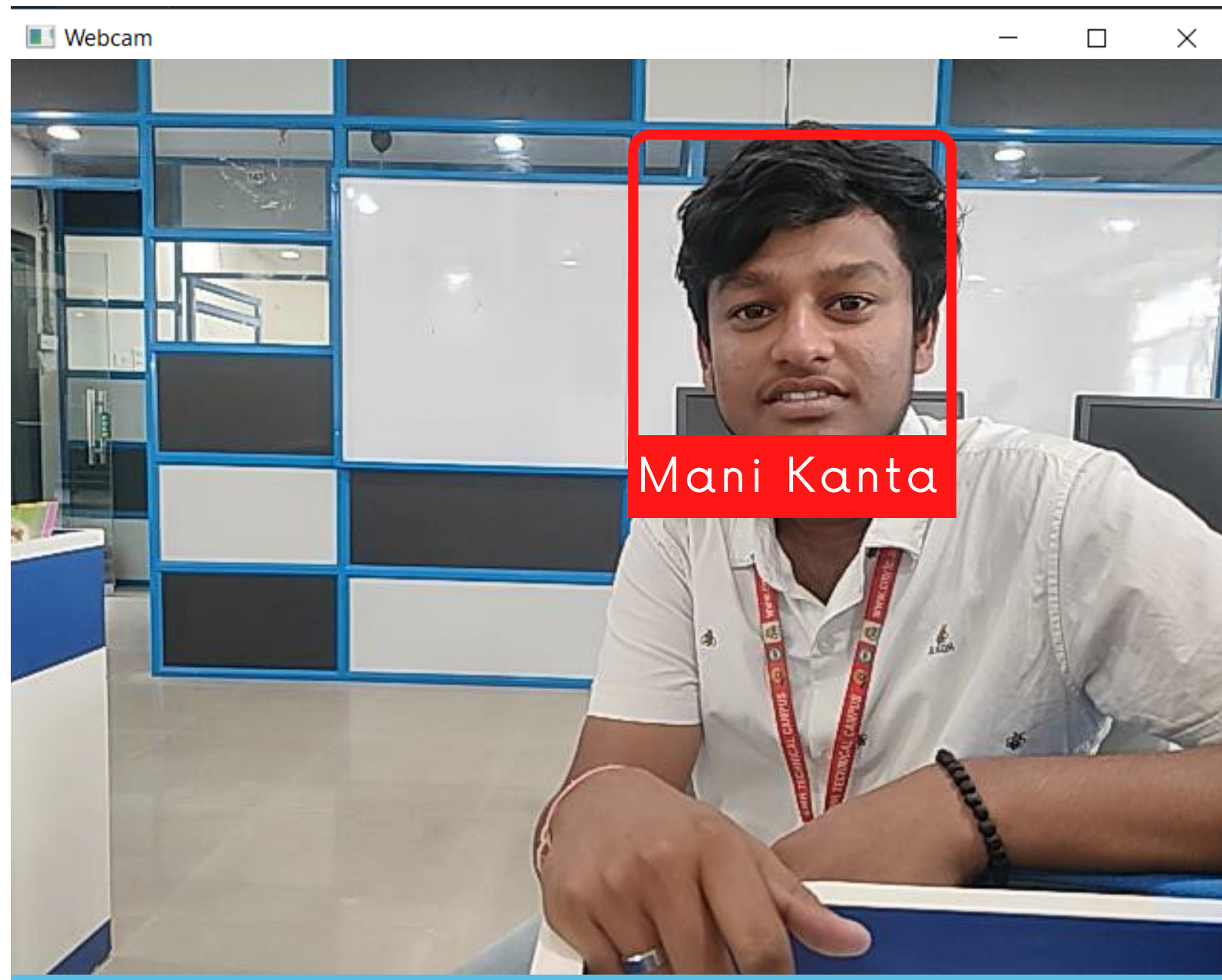
SEQUENCE DIAGRAM

UML DIAGRAMS:



ACTIVITY DIAGRAM

OUTPUT



Face Recognition

	A	B	
1	Name	Time	
2			
3	SAI PRANAV	09:19:24	
4	ANANYA	09:20:38	
5	GAURAV	09:20:13	
6	NIKHIL	09:21:24	
7	SINDHUJA	09:21:38	
8	MANI KANTA	09:43:13	
9			
10			

.csv Sheet



CONCLUSION

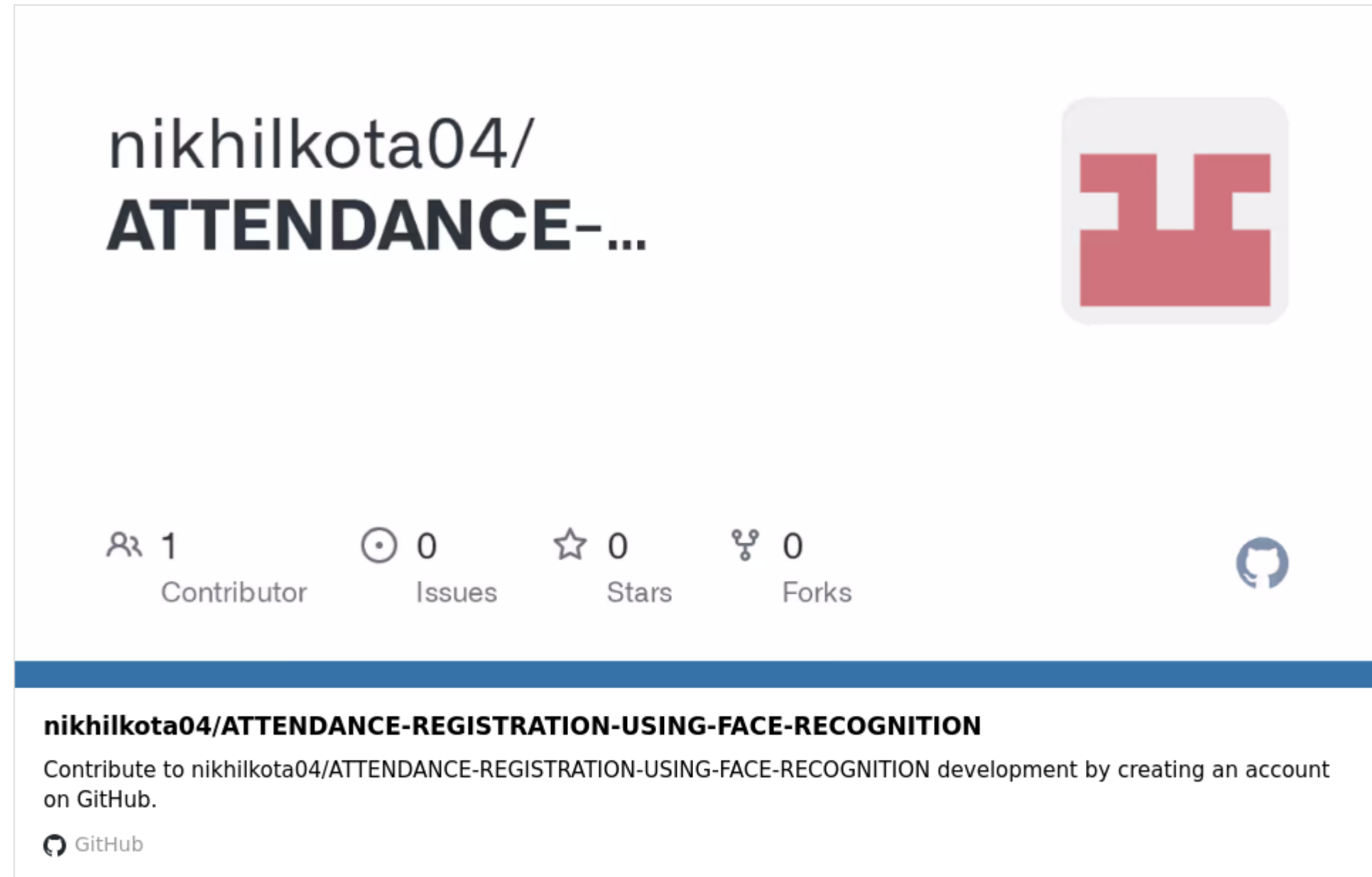
This system aims to build an effective attendance system using face recognition techniques. The proposed system will be able to mark the attendance via face Id. It will detect faces with help of a camera module and then recognize the faces. After recognition, it will mark the attendance of the recognized person and update the attendance record. Using technology to conquer the defects cannot merely save resources but also reduces human intervention in the whole process by handling all the complicated tasks to the machine. In the end, the system not only resolves troubles that exist in the old model but also provides convenience to the user to access the information collected by mailing the attendance sheet to their respective supervisor or guardian.



FUTURE ENHANCEMENTS

Practically all academic institutions require attendance records of students and maintaining attendance physically can be a hectic as well as time-consuming task. Hence maintaining attendance automatically with the help of face recognition will be exceptionally useful and less prone to mistakes or errors as compared to the manual procedure. This will also reduce the manipulation of attendance records done by students and reduces time consumption too. The future extent of the proposed work can be, catching numerous definite pictures of the students and utilizing any cloud innovation to store these pictures so that the attendance sheets can be accessed anywhere. This framework can be designed and utilized in ATM machines to identify frauds, Retailing, and Shopping malls. Also, the framework can be utilized at the time of elections where the voters can be distinguished by perceiving the face.

GitHub link



<https://github.com/nikhilkota04/ATTENDANCE-REGISTRATION-USING-FACE-RECOGNITION.git>



.....> **THANK YOU** <.....