Mini Project 1 B Report

Submitted in partial fulfillment of the requirement of University of Mumbai

For the Degree of

(Computer Engineering)

By

1) “Yash Phuke”	**“ID No: TU3F2122117”**
2) “Kiran Mishra”	**“ID No: TU3F2122105”**

Under the Guidance of

Prof. Vishwajit Gaikwad

Department of Computer Engineering

TERNA ENGINEERING COLLEGE

Plot no.12, Sector-22, Opp. Nerul Railway station,

Phase-11, Nerul (w), Navi Mumbai 400706

UNIVERSITY OF MUMBAI

TERNA ENGINEERING COLLEGE, NERUL,

NAVI MUMBAI

Department of Computer Engineering

Academic Year 2022-23

CERTIFICATE

This is to certify that the mini project 1B entitles “**Face Recognition Attendance System**” is a bonafide work of

1)“Yash Phuke”	**“ID No: TU3F2122117”**
2) “Kiran Mishra”	**“ID No: TU3F2122105”**

submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the Bachelor of Engineering (Computer Engineering).

Guide **Head of Department** **Principal**

Project Report Approval

This Mini Project 1B Report – “**Face Recognition Attendance System**” by following students is approved for the degree of ***S.E. in "Computer Engineering"***.

Submitted by:

1) “Yash Phuke”	**“ID No: TU3F2122117”**
2) “Kiran Mishra”	**“ID No: TU3F2122105”**

Examiners Name & Signature:

1.---

2.---

Date: --------------------------------

Place: --------------------------------

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

1) "Yash Phuke"	**"ID No: TU3F2122117"**
2) "Kiran Mishra"	**"ID No: TU3F2122105"**

Date: ______________________

Place: ______________________

Acknowledgement

We would like to express our sincere gratitude towards our guide **Prof. Vishwajit Gaikwad**, Mini Project Coordinators **Prof. Mini Project Coordinator Saima Sayyed** for their help, guidance and encouragement, they provided during the project development. This work would have not been possible without their valuable time, patience and motivation. We thank them for making our stint thoroughly pleasant and enriching. It was great learning and an honor being their student.

We are deeply thankful to **Dr. Seema Bidey (H.O.D Computer Department)** and entire team in the Computer Department. They supported us with scientific guidance, advice and encouragement, they were always helpful and enthusiastic and this inspired us in our work.

We take the privilege to express our sincere thanks to **Dr. L. K. Ragha** our Principal for providing the encouragement and much support throughout our work.

1) "Yash Phuke"	**"ID No: TU3F2122117"**
2) "Kiran Mishra"	**"ID No: TU3F2122105"**

Date: ______________________

Place: ______________________

Index

Abstract

The face detection & Attendance marking system is a computer vision project which refers to the detection of face automatically by computerized systems by taking a look at face .

It utilizes OpenCV, Firebase, and facial recognition algorithms. The objective of this project is to create an automated attendance system that can capture an image or video stream, detect faces using OpenCV, match the detected faces with the faces stored in Firebase's real-time database, and finally, mark attendance for recognized faces.. It is a popular feature used in bio metrics, digital cameras and social tagging.

The project demonstrates how pre-trained models can be used to perform face recognition in Python using the face_recognition library.

This project has various potential applications in the education and corporate sector, where attendance management is a critical task. The automated attendance marking system can help reduce the time and effort required for manual attendance marking, improve accuracy, and provide real-time insights into attendance patterns.

Chapter 1

Introduction

- The traditional method of taking attendance in schools, colleges, and organizations involves calling out the names and marking attendance manually, which is not only time-consuming but also prone to errors. With the advancement in technology, automated attendance systems have become increasingly popular, and one such system is the face detection attendance system.
- The face Recognition Attendance system is a modern-day solution that leverages computer vision and artificial intelligence (AI) algorithms to automatically capture attendance data by recognizing the faces of individuals. The system is designed to streamline the attendance-taking process, minimize human intervention, and eliminate errors.
- Overall, this report aims to provide insights into the development of a face detection attendance system and its potential to revolutionize the attendance-taking process.

Chapter 2

Literature Survey

Company Name	Key Features :	Critical Parts & Applications
Amazon Rekognition	-High accuracy and speed - Can recognize faces in images and videos - Supports facial analysis (e.g. emotion, gender, age)	Security and surveillance, customer engagement, and user verification
Microsoft Face API	-Can detect and recognize faces in images and videos -Supports facial analysis (e.g. emotion, age, gender) -Can group faces with similar features	Security and surveillance, access control, and digital marketing
IBM Watson Visual Recognition	-Image and facial analysis, -Face detection and recognition, -Age and gender estimation	Security and surveillance, retail and e-commerce, and healthcare
Google Cloud Vision	Face detection -Face recognition & tracking	Security and surveillance, access control, and Digital marketing
Kairos	-Can recognize faces in images and videos -Offers an API for easy integration -Supports facial analysis (e.g. emotion, age, gender)	Cloud Computing, Security and surveillance.

Chapter 3

3.1 Objective

1. To design a face detection attendance system that can accurately and efficiently capture attendance data.
2. To develop an algorithm that can recognize faces in various lighting conditions and orientations.
3. To integrate the face detection algorithm with a database to enable real-time attendance recording and monitoring.

3.2 Problem statement

- Traditional attendance taking methods involve manual recording of attendance, which is time-consuming and prone to errors. Furthermore, it can be difficult to identify students or employees who are absent or tardy, which can lead to issues with accountability and productivity. To overcome these challenges, there is a need for an automated attendance system that can accurately and efficiently record attendance.

3.3 Hardware & Software Requirements

Software :

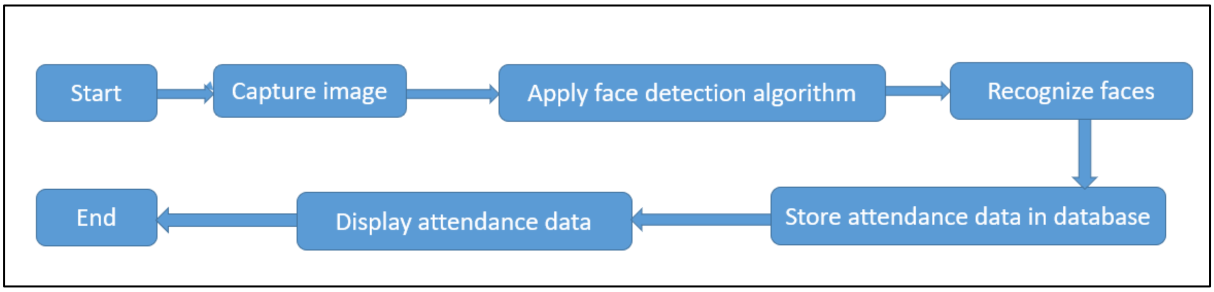
1. Firebase
2. Pycharm Community Edition 2022.3.3

Hardware :

1. Intel core i5
2. Minimum 8GB of RAM
3. Minimum 256GB of Hard disk Space

Chapter 4

Architecture :


Start
Capture image
Apply face detection algorithm
Recognize faces
Store attendance data in database
Display attendance data
End

Project Design & Implementation:

1.Face Registration: First the user registers himself .In the Registration process , we add the data of the student and his images are taken , later on stored in a specified path.We use capture.py file to capture the image the user whose registration is to be done.Videocapture() is the method used to capture the images , cap.read() reads each frame of the video and returns the image in form of numpy array .

2.Adding Info : Further ,the details of the students are added. The details includes his or her name , ID , Course , Admission year.His Id Is saved as the file name which will be used to refer which file was matched with the current face while taking attendance.The information are updated in the Firebase. The Firebase has a systematic records of each and every information stored.

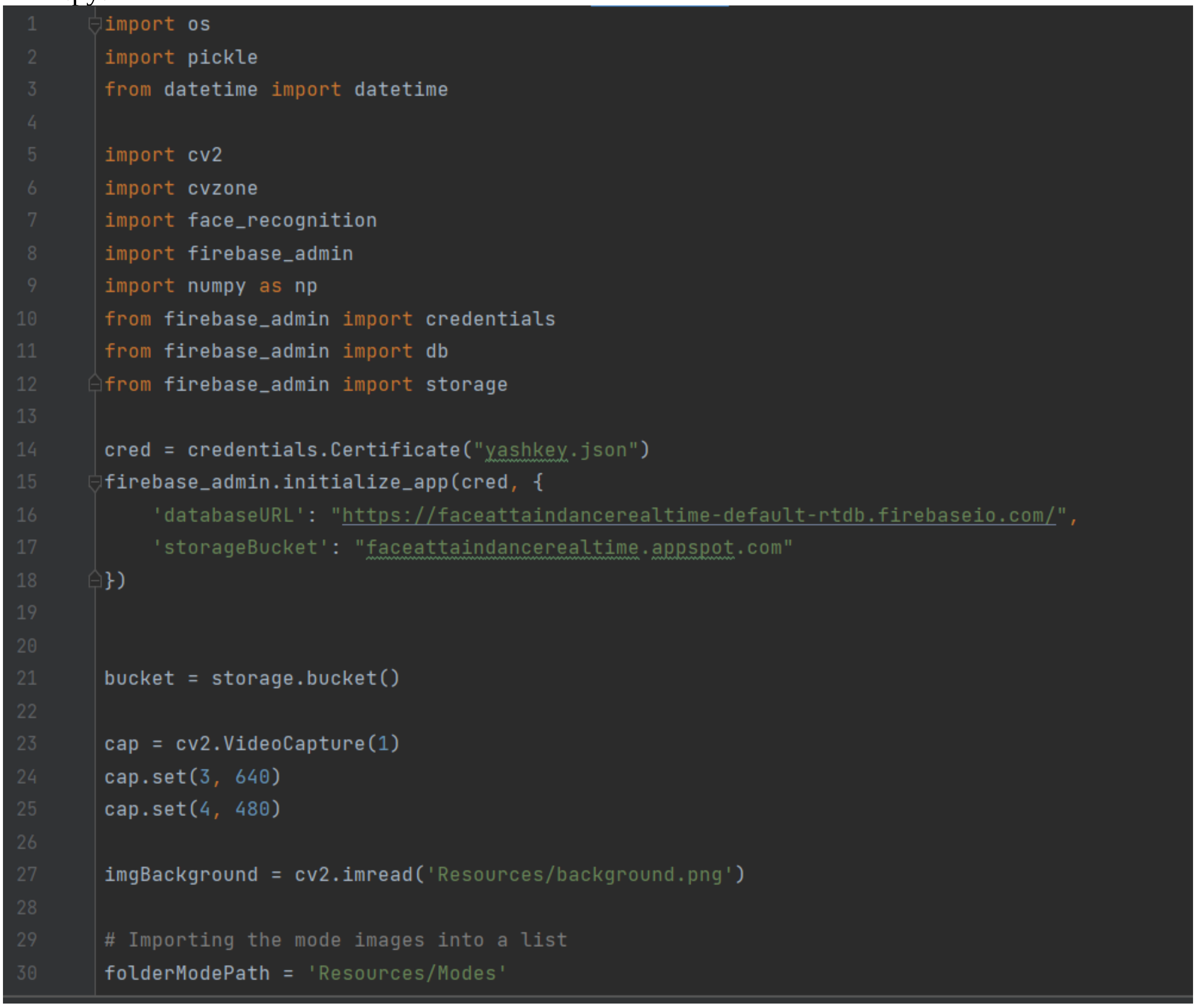
3.Taking Attendance: For taking Attendance , webcam opens , captures images , generated encodes. And tries to match with already available encodes , if matches , face is detected.When Attendance is being taken , each face image has record of its ID , Which is also assigned an index . When the algorithm matches the face , it goes to that index , and fetches its ID.

4.Updating Records :

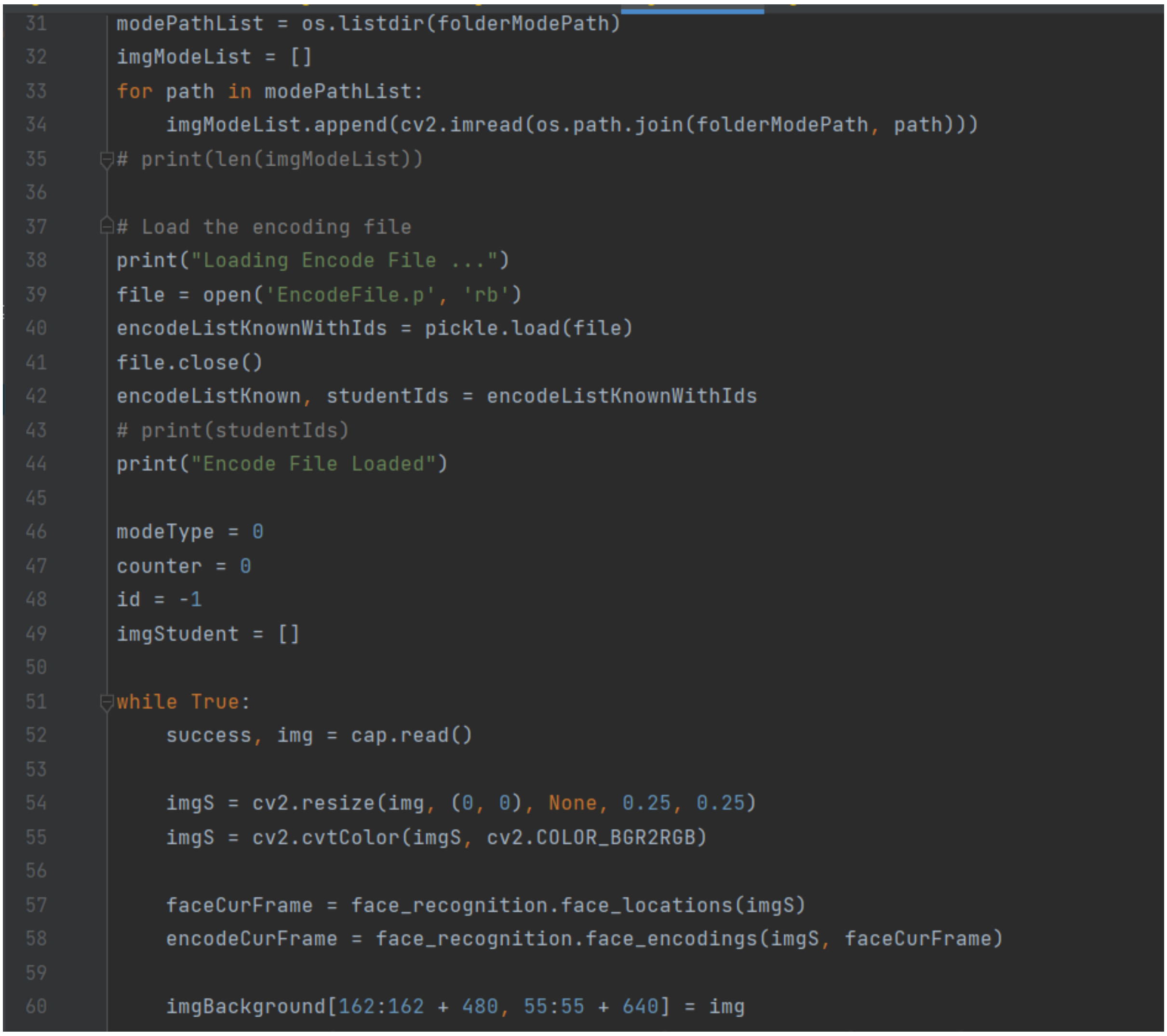
After face is recognized , attendance is updated in the firebase .The records will be useful for future information , so that the admin can find out the records anytime needed.

Code:

Main.py:



```
import os
import pickle
from datetime import datetime

import cv2
import cvzone
import face_recognition
import firebase_admin
import numpy as np
from firebase_admin import credentials
from firebase_admin import db
from firebase_admin import storage

cred = credentials.Certificate("yashkey.json")
firebase_admin.initialize_app(cred, {
    'databaseURL': "https://faceattaindancerealtime-default-rtdb.firebaseio.com/",
    'storageBucket': "faceattaindancerealtime.appspot.com"
})


bucket = storage.bucket()

cap = cv2.VideoCapture(1)
cap.set(3, 640)
cap.set(4, 480)

imgBackground = cv2.imread('Resources/background.png')

# Importing the mode images into a list
folderModePath = 'Resources/Modes'
```

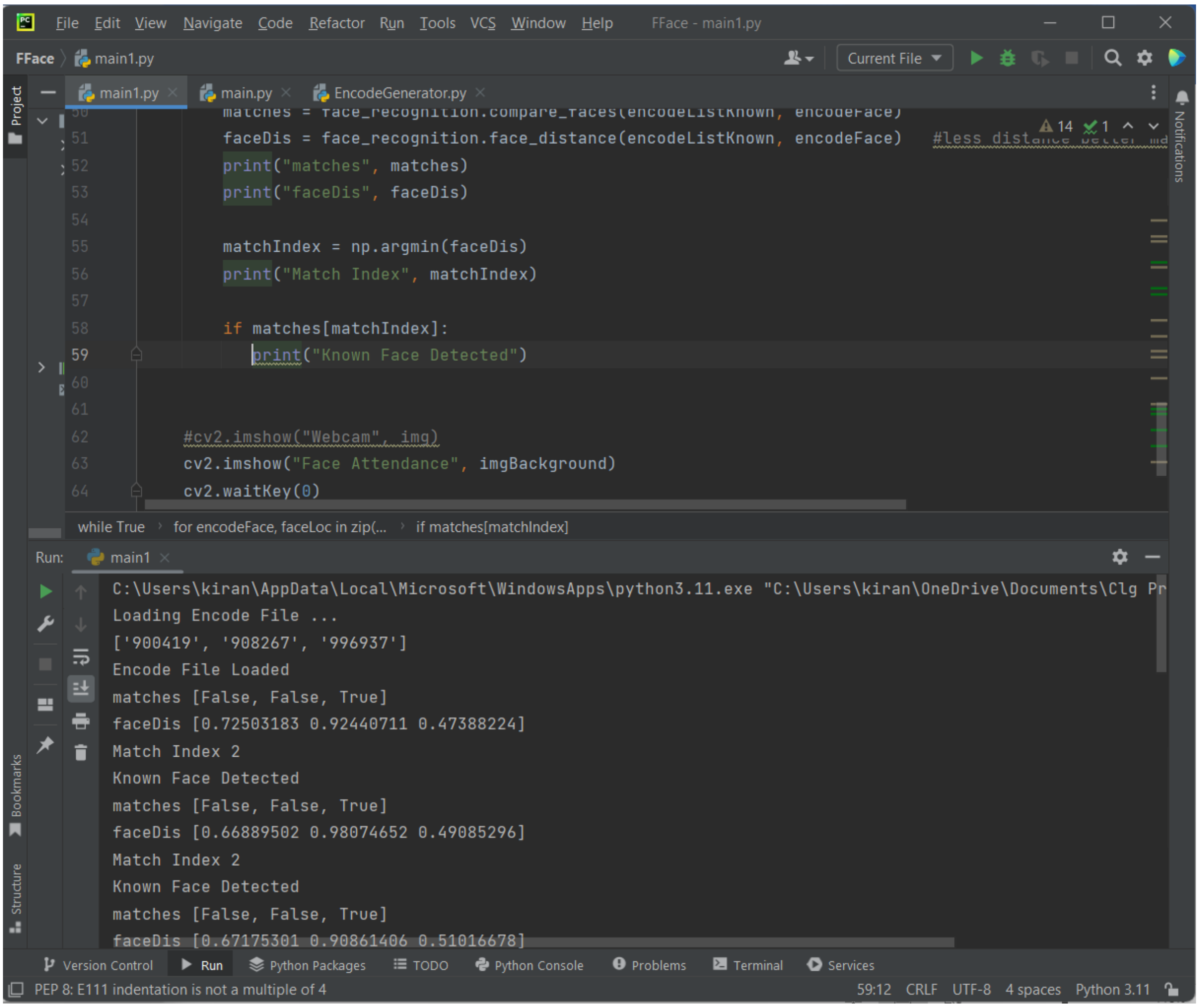


```python
modePathList = os.listdir(folderModePath)
imgModeList = []
for path in modePathList:
    imgModeList.append(cv2.imread(os.path.join(folderModePath, path)))
# print(len(imgModeList))

# Load the encoding file
print("Loading Encode File ...")
file = open('EncodeFile.p', 'rb')
encodeListKnownWithIds = pickle.load(file)
file.close()
encodeListKnown, studentIds = encodeListKnownWithIds
# print(studentIds)
print("Encode File Loaded")

modeType = 0
counter = 0
id = -1
imgStudent = []

while True:
    success, img = cap.read()

    imgS = cv2.resize(img, (0, 0), None, 0.25, 0.25)
    imgS = cv2.cvtColor(imgS, cv2.COLOR_BGR2RGB)

    faceCurFrame = face_recognition.face_locations(imgS)
    encodeCurFrame = face_recognition.face_encodings(imgS, faceCurFrame)

    imgBackground[162:162 + 480, 55:55 + 640] = img
```

Implementation :

When it was at its early , it was able to detect faces and returned True , to indicate which images was matching.

Matching Index No.



File Edit View Navigate Code Refactor Run Tools VCS Window Help FFace - main1.py
FFace › main1.py
Current File
main1.py
main.py
EncodeGenerator.py
Project
Notifications
while True › for encodeFace, faceLoc in zip(... › if matches[matchIndex]
Run: main1
Bookmarks
Structure
Version Control
Run
Python Packages
TODO
Python Console
Problems
Terminal
Services
PEP 8: E111 indentation is not a multiple of 4
59:12 CRLF UTF-8 4 spaces Python 3.11
14 1

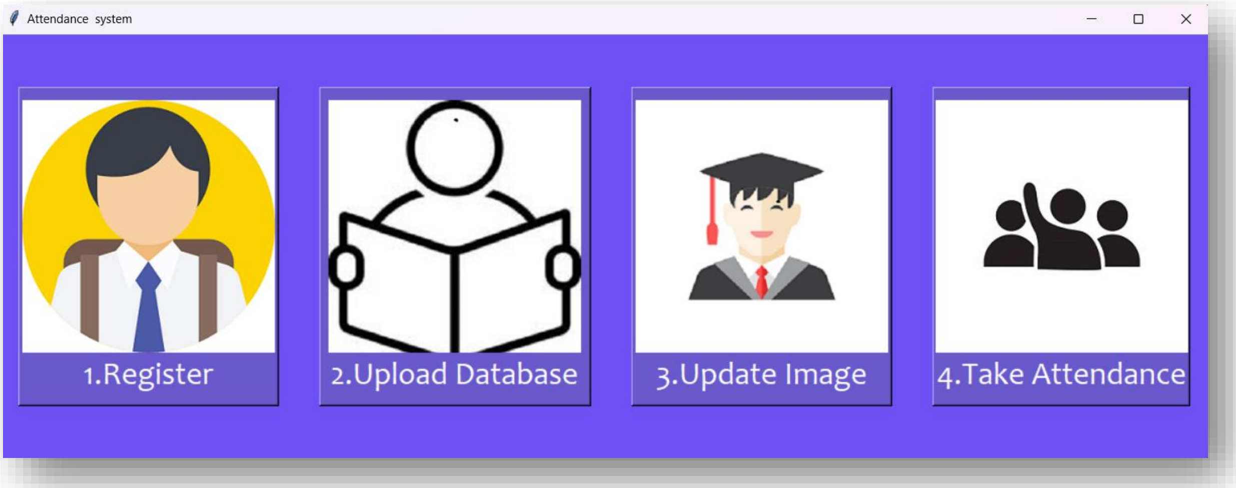
```
        matches = face_recognition.compare_faces(encodeListKnown, encodeFace)
        faceDis = face_recognition.face_distance(encodeListKnown, encodeFace)   #less distance better ma
        print("matches", matches)
        print("faceDis", faceDis)

        matchIndex = np.argmin(faceDis)
        print("Match Index", matchIndex)

        if matches[matchIndex]:
           print("Known Face Detected")


    #cv2.imshow("Webcam", img)
    cv2.imshow("Face Attendance", imgBackground)
    cv2.waitKey(0)
```

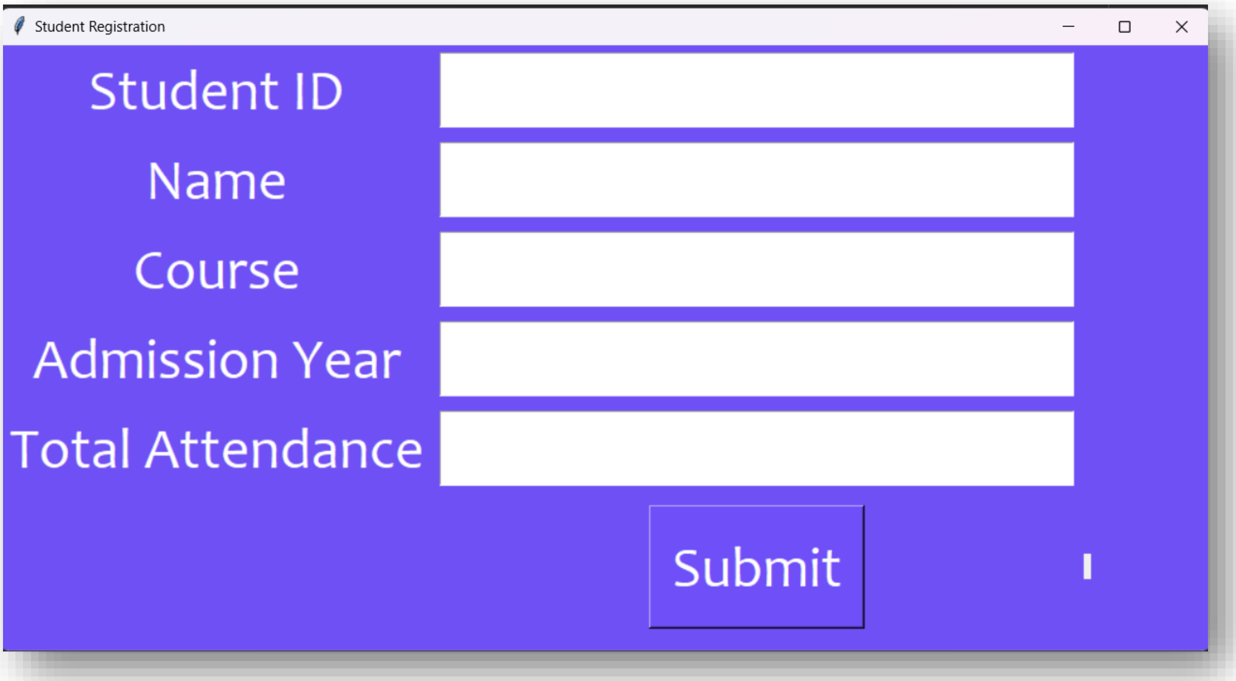
```
C:\Users\kiran\AppData\Local\Microsoft\WindowsApps\python3.11.exe "C:\Users\kiran\OneDrive\Documents\Clg Pr
Loading Encode File ...
['900419', '908267', '996937']
Encode File Loaded
matches [False, False, True]
faceDis [0.72503183 0.92440711 0.47388224]
Match Index 2
Known Face Detected
matches [False, False, True]
faceDis [0.66889502 0.98074652 0.49085296]
Match Index 2
Known Face Detected
matches [False, False, True]
faceDis [0.67175301 0.90861406 0.51016678]
```

Chapter 5

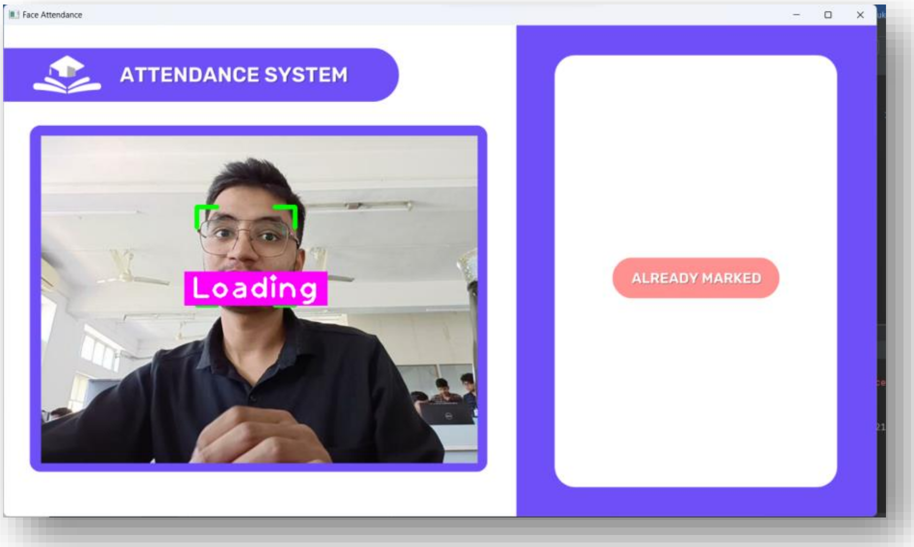
Results:

Registration Process:

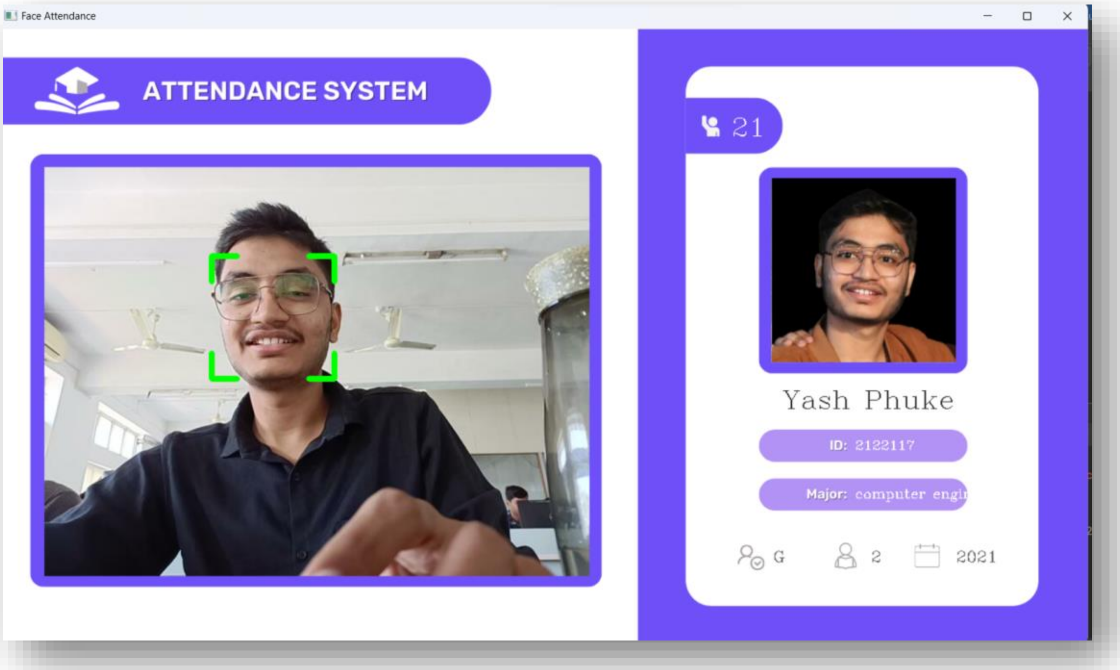




The process of Marking Attendance.



Displaying the details of Student after Recognization.



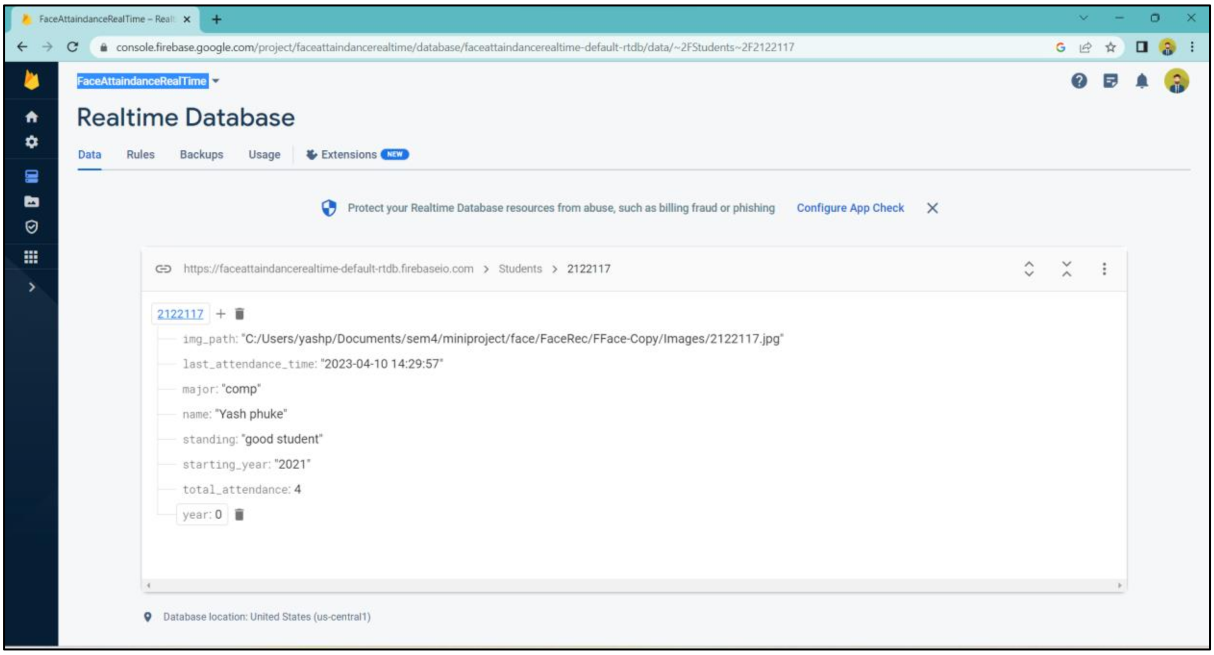
Yash Phuke

ID: 2122117

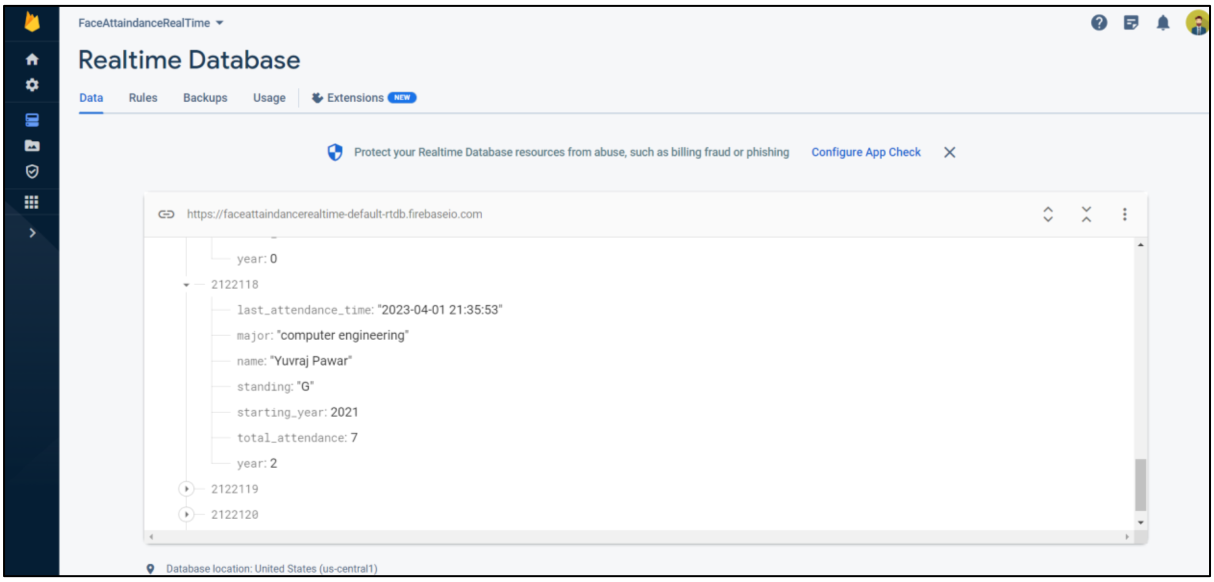
Major: computer engi

G 2 2021

The information regarding the student , the name, major , total-attendance, admission year,the path in which it is stored is stored in the Realtime Database section of firebase As an when , attendance is taken ,records are updated here.

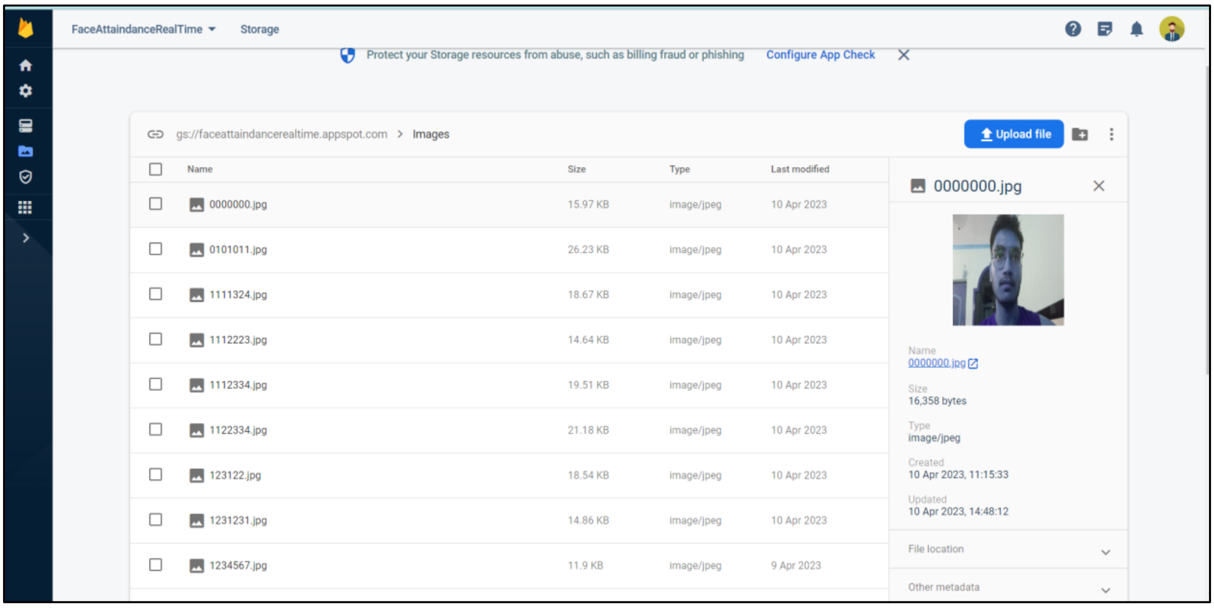

FaceAttaindanceRealTime – Reali
console.firebase.google.com/project/faceattaindancerealtime/database/faceattaindancerealtime-default-rtdb/data/~2FStudents~2F2122117
FaceAttaindanceRealTime
Realtime Database
Data
Rules
Backups
Usage
Extensions NEW
Protect your Realtime Database resources from abuse, such as billing fraud or phishing
Configure App Check
https://faceattaindancerealtime-default-rtdb.firebaseio.com > Students > 2122117
Database location: United States (us-central1)

```
2122117
  img_path: "C:/Users/yashp/Documents/sem4/miniproject/face/FaceRec/FFace-Copy/Images/2122117.jpg"
  last_attendance_time: "2023-04-10 14:29:57"
  major: "comp"
  name: "Yash phuke"
  standing: "good student"
  starting_year: "2021"
  total_attendance: 4
  year: 0
```


FaceAttaindanceRealTime
Realtime Database
Data
Rules
Backups
Usage
Extensions NEW
Protect your Realtime Database resources from abuse, such as billing fraud or phishing
Configure App Check
https://faceattaindancerealtime-default-rtdb.firebaseio.com
Database location: United States (us-central1)

```
    year: 0
2122118
    last_attendance_time: "2023-04-01 21:35:53"
    major: "computer engineering"
    name: "Yuvraj Pawar"
    standing: "G"
    starting_year: 2021
    total_attendance: 7
    year: 2
2122119
2122120
```

The images which it captures for face recognition are stored in the Storage section in the firebase.


FaceAttaindanceRealTime
Storage
Protect your Storage resources from abuse, such as billing fraud or phishing
Configure App Check
gs://faceattaindancerealtime.appspot.com > Images
Upload file

☐	Name	Size	Type	Last modified
☐	0000000.jpg	15.97 KB	image/jpeg	10 Apr 2023
☐	0101011.jpg	26.23 KB	image/jpeg	10 Apr 2023
☐	1111324.jpg	18.67 KB	image/jpeg	10 Apr 2023
☐	1112223.jpg	14.64 KB	image/jpeg	10 Apr 2023
☐	1112334.jpg	19.51 KB	image/jpeg	10 Apr 2023
☐	1122334.jpg	21.18 KB	image/jpeg	10 Apr 2023
☐	123122.jpg	18.54 KB	image/jpeg	10 Apr 2023
☐	1231231.jpg	14.86 KB	image/jpeg	10 Apr 2023
☐	1234567.jpg	11.9 KB	image/jpeg	9 Apr 2023

0000000.jpg

Name
0000000.jpg

Size
16,358 bytes

Type
image/jpeg

Created
10 Apr 2023, 11:15:33

Updated
10 Apr 2023, 14:48:12

File location

Other metadata

Chapter 6

Conclusion

- This system is an efficient and accurate solution for attendance-taking that leverages computer vision and AI algorithms. The system automates the attendance-taking process by recognizing individuals' faces and storing attendance data in a database system. The user interface allows users to monitor attendance in real-time and generate reports, making it an ideal solution for various domains, including education, healthcare, and corporate settings.
- The system's implementation requires a camera capable of capturing high-quality images, a face detection algorithm capable of recognizing faces in various lighting conditions, and a database management system capable of storing attendance data. The system can be customized to meet the specific requirements of different organizations, making it a versatile solution for attendance management.

Chapter 7

Future Scope

- The face Recognition attendance system can be further enhanced in several ways to improve its accuracy, efficiency, and usability. Some of the future scope areas include:

1. Integration with other biometric authentication methods such as fingerprint, iris, or voice recognition to increase security.
2. Integration with machine learning algorithms to improve face detection accuracy and reduce false positives and false negatives.
3. Adding Featues Of Online Video Conferencing & Chatbox.
4. Development of a mobile application that can access attendance data and provide real-time notifications to students, teachers, or administrators.
5. Integration with blockchain technology to increase data security and transparency.

Chapter 8

References

1. https://ieeexplore.ieee.org/document/8974493
2. https://www.researchgate.net/publication/323390774_Face_detection_and_Recognition_A_review/
3. OpenCV: Face Detection. Retrieved from https://docs.opencv.org/master/d7/d8b/tutorial_py_face_detection.html
4. https://medium.com/@ageitgey/machine-learning-is-fun-part-4-modern-face-recognition-with-deep-learning-c3cffc121d78
5. https://www.youtube.com/watch?v=6qQZr9h8qL0/
6. Ingle, R., & Bhardwaj, R. K. (2017). Development of attendance management system using face recognition. International Journal of Advanced Research in Computer Engineering & Technology, 6(11), 249-251.