Jaypee Institute of Information Technology, Noida

Major Project-1

Odd Semester, 2021

Progress Report-3

1. Group No.: 17

2. Group Members Name and Enrollment Number:

1. Rishabh Rajpurohit (9919103096)
2. P. Paul Jonathan (9919103121)
3. Yash Raj Gairola (9919103150)

3. Supervisor: Dr. Neeraj Jain

4. Title: Smart Attendance System

5. Objectives in Points:

- To create a csv file based database which can store attendance of different classes in a combined file in a structured way.
- Our attendance record shall contain subject name, faculty name, class type, class scheduled time, enrollment of students with their attendance timestamps.

6. Work done:

- We have implemented a functionality in which a daily attendance record csv file is created with the filename - Attendance-<current Date>.csv in the Attendance directory of our project. Whenever we run Recognition Mode (2) in our System, the system asks us to input details like Subject Name, Class Type and Timings and the Faculty name. Then it starts the webcam and looks for students who are registered or stored in the numpy database with their enrollment number as their labels. It then checks if the header is inserted in the Attendance record and if yes then it starts to store enrollment number of recognized students in the database in serial order subject and class wise.

```
recognized__students__list = list()
date_today_compressed = date.today().strftime("%a-%d-%m-%y")
date_today_detailed = date.today().strftime("%a-%d-%B-%Y")

attendance__dir = "C:\\Users\\Rishabh Rajpurohit\\Documents\\majorP\\Attendance"

sub__name = "default subject"
cls__type = "lecture"
faculty__name = "default faculty"
cls__time = "9am"
s__no = 1

if f'Attendance-{date_today_detailed}.csv' not in os.listdir(attendance__dir):
    with open(f'Attendance/Attendance-{date_today_detailed}.csv','w') as f:
        f.write('S-No,Enrollment-No,Time-Stamp(Hour:Min:Sec),Subject-Name,Class-Type,Faculty-Name,Class-Time')
```

```
45
46    def LoadModelAndRun():
47        s__no = 1
48        recognized__students__list.clear()
49        sub__name, cls__type, faculty__name, cls__time = input('''\nEnter the following
          Details:\nFormat: <Sub_name>,<Class_type>,<Faculty_Name>,<Class_time>\n--->''').
          split(',')
50        try:
51            trained_face_recognizer=numpy.load(savedModelLocation)
52        except:
53            print('\n\nTrain the model first!')
54            return
```

```
if (prediction[1])<=100 and (prediction[1])>85:
    current__time = datetime.now().strftime('%H:%M:%S')
    df = pd.read_csv(f'Attendance/Attendance-{date_today_detailed}.csv')
    if(recognized__students__list.count(names[prediction[0]])==0):
        recognized__students__list.append(names[prediction[0]])
        with open(f'Attendance/Attendance-{date_today_detailed}.csv', 'a')
        as f:
            f.write(f'\n{s__no},{names[prediction[0]]},{current__time},
            {sub__name},{cls__type},{faculty__name},{cls__time}')
            s__no += 1
        print('%s - %s' % (names[prediction[0]],"marked PRESENT"))
        cv2.putText(im,'%s - %.0f%s' % (names[prediction[0]],prediction[1],
        "%"),(x-10, y-10), cv2.FONT_HERSHEY_PLAIN,1,(0, 255, 0))
        play__sound(str(names[prediction[0]]) + "marked PRESENT")
        #play__sound("next student, please come forward")
```