**A PROJECT REPORT**

**ON**

**ATTENDANCE SYSTEM BY FACE RECOGNITION**

A PROJECT

Submitted in partial fulfillment of the requirements for the award of the degree

OF

BACHELOR OF COMPUTER APPLICATION

Submitted By

**RAUNAQ VAID (02491102018)**

**UNDER THE SUPERVISION OF:**

Mrs. Prabhjot Kaur



**SRI GURU TEGH BHAHADUR INSTITUTE OF MANAGEMENT AND INFORMATION TECHNOLOGY**

*GURUDWARA NANAK PIAO CAMPUS G.T. KARNAL ROAD,DELHI-110033*

(Approved By AICTE Ministry of HRD, Govt. of India)

*AFFILIATED TO GURU GOBIND SINGH INDRAPRASTHA UNIVERSITY,DELHI*

2018-2021

CANDIDATE’S DECLARATION

I hereby certify that the work which is being presented in the project is the project report entitled ***ATTENDANCE SYSTEM BY FACE RECOGNISTION*** in partial fulfilment of the requirement for the awards of the degree of ***“BACHELOR OF COMPUTER APPLICATION“*,** submitted in the ***Sri Guru Tegh Bahadur of Management and Information Technology, Guru Gobind Singh Indraprastha University*** is an authentic record of my work carried out under the supervision of Mrs.

Prabhjot Kaur .

The matter embodied in this project report has not been submitted by us for the award of my other degree or diploma.

Date: 02/01/2021

Raunaq Vaid

CERTIFICATE

This is to certify that the project entitled: ***“Attendance system by face recognition”*** done by ***Mr. Raunaq Vaid U***niversity Enrolment Number 02491102018 is an authentic work carried out by them at ***Sri Guru Tegh Bahadur******Institute Of Management & Information Technology***. The matter embodied in this project work has not been submitted earlier for the award of any degree or diplomas to the best of my knowledge and belief. The suggestion as approved by the faculty was duly incorporated.

DATE: 02/01/2021

HOD: Amandeep Singh

Guide: Mrs. Prabhjot Kaur

ACKNOWLEDGEMENT

I sincerely thank Mr. Amandeep Singh (HOD) **Sri Guru Tegh Bahadur Institute of Management & Information Technology** for giving me the opportunities to work under their able guidance.

As a write through these pages my mind recollects their invaluable suggestions, their encouragement at every step which was my guiding light.

Last but not the least my sincere thanks to my family and my friends for their valuable suggestions at every time.

Place: New Delhi

Raunaq Vaid

BCA – 6TH SEM.

HOD: Amandeep Singh

Guide: Mrs. Prabhjot kaur

ABSTRACT

Uniqueness or individuality of an individual face is the representation of one’s identity. In this project face of an individual is used for the purpose of attendance making automatically. Attendance of the student is very important for every college, universities and school. Conventional methodology for taking attendance is by calling the name or roll number of the student and the attendance is recorded. Time consumption for this purpose is an important point of concern. Assume that the duration for one subject is around 60 minutes or 1 hour & to record attendance takes 5 to 10 minutes. For every tutor this is consumption of time. To stay away from these losses, an automatic process is used in this project which is based on image processing. In this project face detection and face recognition is used. Face detection is used to locate the position of face region and face recognition is used for marking the understudy’s attendance. The database of all the students in the class is stored and when the face of the individual student matches with one of the faces stored in the database then the attendance is recorded.

**TABLE OF CONTENT**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Chapter No.** | **Topic** | | **Page No.** | |
| 1. | **INTRODUCTION** | |  | |
| 1.1 | Introduction to the Project | | 10 | |
| 1.2 | Aims and Objectives | | 10 | |
| 1.3 | Advantages of the Project | | 10 | |
| 1.4 | Disadvantages of the Project | | 10 | |
| **2.** | **REQUIREMENT & ANALYSIS** | |  | |
| 2.1 | Software Requirement Specification | | 11 | |
| 2.2 | Project Category | | 13 | |
| 2.3 | Software Tools | | 13 | |
| 2.4 | Technologies Used | | 14 | |
| 2.5 | Use Case Diagram | | 14 | |
| **3** | | | **SOFTWARE DESIGN** | |  |
| 3.1 | | | Introduction | | 18 |
| 3.2 | | | High Level/Detailed Design (HLD/DD) | | 18 |
| 3.3 | | | Context Diagram (Level 0) | | 19 |
| 3.4 | | | Data Flow Diagram (Level 1) | | 20 |
| **4** | | | **DATABASE DESIGN** | |  |
| 4.1 | | | Introduction | | 22 |
| 4.2 | | | Database Field Specifications | | 22 |
| 4.3 | | | Entity-Relationship Diagram | | 23 |
| **5** | | | **TESTING** | |  |
| 5.1 | | | Introduction | | 26 |
| 5.2 | | | Methods employed for testing | | 26 |
| 5.3 | | | Test cases | | 27 |
| **6** | | | **ROLES AND RESPONSIBILITIES** | |  |
| 6.1 | | | Project roles and responsibilities | | 29 |
| **7** | **CONCLUSION & ACHIEVMENTS** | | 31 | |
|  | **APPENDICES** | | 32 | |
|  | **SNAPSHOTS** | | 35-39 | |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **S.NO.** | **Figure** | **Page**  **Number** |
| 1 | Figure 1: The Waterfall Model | 13 |
| 2 | Figure 2:Use Case | 16 |
| 3 | Figure 3: Context Flow Diagram | 19 |
| 4 | Figure 4: Data Flow Diagram Level-1 | 20 |
| 5 | Figure 5: Database tables | 22-23 |
| 6 | Figure 8: ER diagram | 24 |

**LIST OF CSV FILE**

|  |  |  |
| --- | --- | --- |
| **S.NO.** | **Table** | **Page Number** |
| 1 | Student\_csv | 22 |
| 2 | Attendance\_csv | 22 |

**LIST OF SCREENSHOTS**

|  |  |  |
| --- | --- | --- |
| **S.NO.** | **Screenshots** | **Page Number** |
| 1 | Screenshot 1 | 35 |
| 2 | Screenshot 2 | 35 |
| 3 | Screenshot 3 | 36 |
| 4 | Screenshot 4 | 37 |
| 5 | Screenshot 5 | 38 |
| 6 | Screenshot 6 | 39 |

**CHAPTER-1**

**INTRODUCTION**

# 1.1 Introduction to the Project

Attendance is essential to both the teacher and the student of an educational organization. It is therefore very important to maintain it. The problem is when you look at the traditional process of taking attendance in class. Calling name or roll number of the student for attendance is not only a problem of time consumption but also it needs energy. So, an automatic attendance system can solve all of those issues. There are a couple of automatic attendance systems that are currently being used by a lot of institutions. One of such system is biometric technique and RFID system. Although it is automatic and a step ahead of traditional method it fails to meet the time constraint. The student has to wait in queue for giving attendance, which is time taking. This project introduces an involuntary attendance marking system, devoid of any kind of interference with the normal teaching procedure. The system can be also implemented during exam sessions or in other teaching activities where attendance is highly essential. This system eliminates classical student identification such as calling name of the student, or checking respective identification cards of the student, which can not only interfere with the ongoing teaching process, but also can be stressful for students during examination sessions. In addition, the students have to register in the database to be recognized. The enrolment can be done on the spot through the user-friendly interface.

# 1.2. Aims and Objectives

The aim of this project is to develop face recognition attendance system. Expected objectives in order to fulfill the aims are:

* Detect-To detect the face segment from the video frame.
* Extract-To extract the useful features from the face detected.
* Classify-To extract the useful features from the face detected.
* Record-To record the attendance of the identified student.

# 1.3 Advantages of the Project

### Computationally simple and fast.

### Shorter training time.

### Robust to local illumination changes.

### Robust to occlusion

# 1.4 Disadvantages of the Project

### Less accurate.

### High false positive rate.

**CHAPTER-2**

**REQUIREMENT ANALYSIS**

2.1 SOFTWARE REQUIREMENT SPECIFICATION

**2.1.1 CATEGORY**

Console Application.

**2.1.2 PURPOSE**

The purpose of developing this web application titled ‘Attendance system by face recognition” is to help the lecturers improve and organize the process of track and manage student attendance and absenteeism.

**2.1.3 SCOPE**

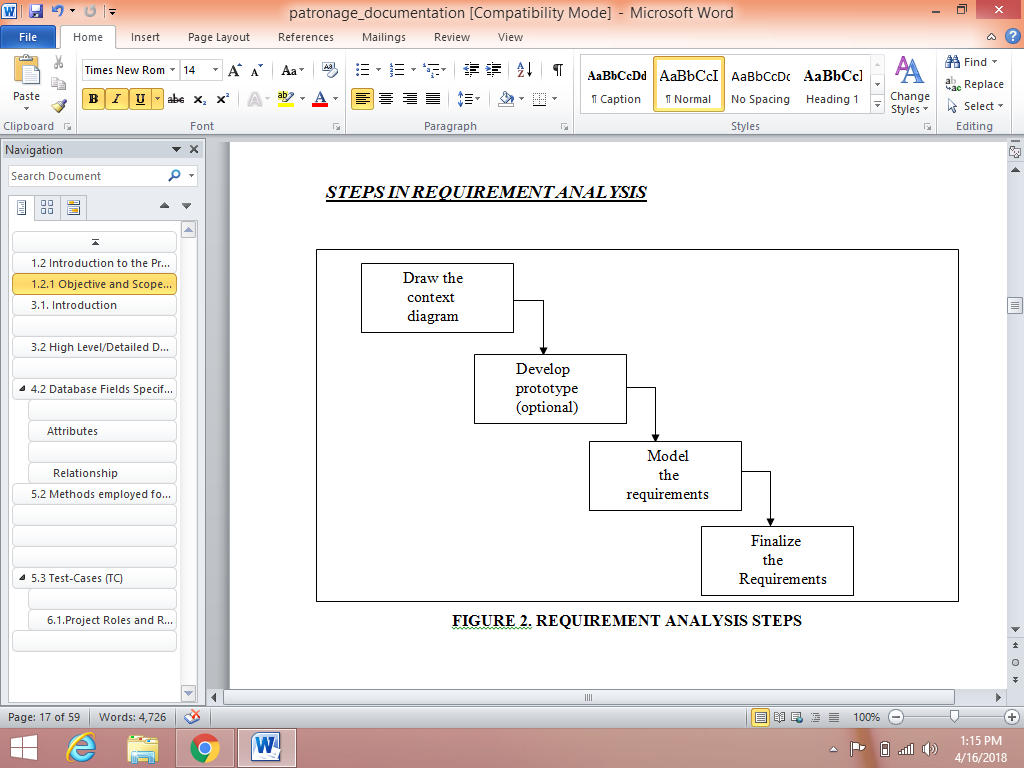
As face provides a unique identity of a person, it can be used to identify a person and verify his/her identity. Face recognition provides non-intrusive way to recognize a person. By using this system, the chances of fake attendance and proxies can be reduced.

2.1.4 PROJECT PLANNING

In order to develop this system, we gave enormous importance to scheduling because we believed if we want to provide the best of quality in a given period of time then we must give due importance to scheduling which also helped us to achieve a better result. Planning of this project includes the following things:

* Gather Data.
* Data Training.
* Choose algorithm and validation method.
* Examine the data.
* Use model with new data.

***STEPS IN REQUIREMENT ANALYSIS***



**FIGURE 1: REQUIREMENT ANALYSIS STEP (waterfall model)**

**2.2 PROJECT CATEGORY**

**Functional Requirements**

Functional requirements are features that allow the system to function as it was intended. Put another way, if the functional requirements are not met, the system will not work. Functional requirements are product **features** and focus on user **requirements**.

* Detect human face in input image.
* Perform image processing.
* Perform mathematical computation.
* Have an accuracy rate between 81%

**Non-Functional Requirements**

Simply said, a non-functional requirement is a specification that describes the system’s operation capabilities and constraints that enhance its functionality. These may be speed, security, reliability, etc.

* Correctness
* Reliability

**2.3 SOFTWARE TOOLS**

**DATABASE:** Microsoft excel

**CLIENT:** Microsoft Internet Explorer or any other web browser.

**DEVELOPMENT TOOLS:** Spyder 4.1.3

**PROGRAMMING LANGUAGE:** Python (version Python 3.7.6)

**2.4 TECHNOLOGIES USED**

Following is a tabular representation of the technology to be used for the development of the proposed solution:

|  |  |
| --- | --- |
| Operating System | Windows 10 |
| Technologies | Python with Machine Learning |
| Tools | Spyder 4.1.3 |
| Data Base | .csv file (Excel Sheet) |

**2.4.1 FRONT END** - **PYTHON**

Python is a high-level, general-purpose and a very popular programming language. Python programming language (latest Python 3) is being used in web development, Machine Learning applications, along with all cutting-edge technology in Software Industry. Python Programming Language is very well suited for Beginners, also for experienced programmers with other programming languages like C++ and Java.

This specially designed Python tutorial will help you learn Python Programming Language in most efficient way, with the topics from basics to advanced (like Web-scraping, Django, Deep-Learning, etc.) with examples.

In order to develop analysis three vital components, need to be installed on your computer system.

* Python − Python is a high-level, general-purpose and a very popular programming language. Python programming language (latest Python 3) is being used in web development, Machine Learning applications, along with all cutting-edge technology in Software Industry.
* Microsoft Excel − Microsoft Excel is a software program produced by Microsoft that allows users to organize, format and calculate data with formulas using a spreadsheet system. This software is part of the Microsoft Office suite and is compatible with other applications in the Office suite.
* Spyder − Spyder is a powerful scientific environment written in Python, for Python, and designed by and for scientists, engineers and data analysts. It offers a unique combination of the advanced editing, analysis, debugging, and profiling functionality of a comprehensive development tool with the data exploration, interactive execution, deep inspection, and beautiful visualization capabilities of a scientific package. Beyond its many built-in features, its abilities can be extended even further via its plugin system and API. Furthermore, Spyder can also be used as a PyQt5 extension library, allowing you to build upon its functionality and embed its components, such as the interactive console, in your own software.

**2.4.2 BACK END** - **.CSV File**

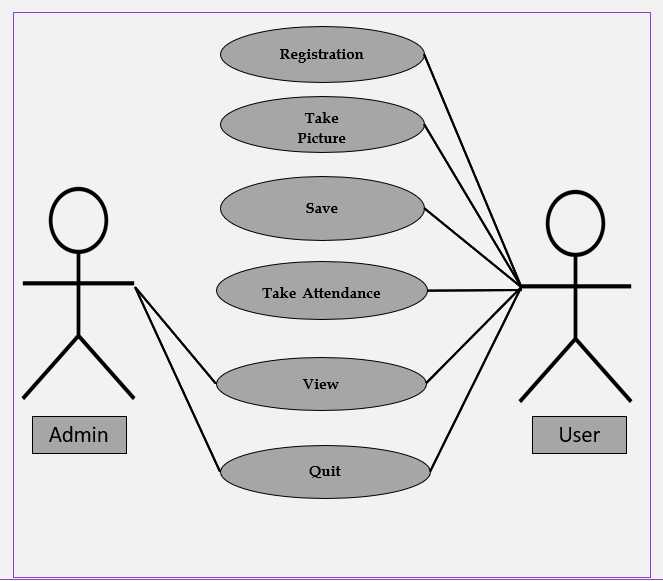
A Comma Separated Values (CSV) file is a plain text file that contains a list of data. These files are often used for exchanging data between different applications. For example, databases and contact managers often support CSV files. These files may sometimes be called Character Separated Values or Comma Delimited files.

CSV files is becoming so popular because of these following reasons:

* CSV files can be opened or edited by text editors like notepad.
* In data-warehouse, CSV follows a fairly flat, simple schema.
* Any programming language to parse CSV data is trivial, generating it is extremely easy.
* CSV is safe and can clearly differentiate between the numeric values and text. CSV does not manipulate data and stores it as-is.
* In CSV, you write column headers only once where as in Excel, you have to have a start tag and end tag for each column in each row.
* Importing CSV files can be much faster, and it also consumes less memory.
* It's easy to programmatically manipulate CSV since, after all, they are simple text files.
* CSV can be opened with any text editor in Windows like notepad, MS Excel, Microsoft Works 9, etc.

**2.5 USE CASE DIAGRAMS**

A **use case diagram** at its simplest is a representation of a user's interaction with the system and depicting the specifications of a use case. A use case diagram can portray the different types of users of a system and the various ways that they interact with the system. This type of diagram is typically used in conjunction with the textual use case and will often be accompanied by other types of diagrams.



**FIGURE 6: USE CASE DIAGRAM**

**CHAPTER – 3**

**SOFTWARE DESIGN**

# 3.1. Introduction

This chapter will focus on the design of the system using diagrams to illustrate graphically certain sections of the software system.

# 3.2 High Level/Detailed Design (HLD/DD)

**3.2.1 Data flow Diagrams (DFDs)**

As information moves through software, it is modified by a series of transformations. A *data flow diagram* is a graphical representation that depicts information flow and the transforms that are applied as data move from input to output.

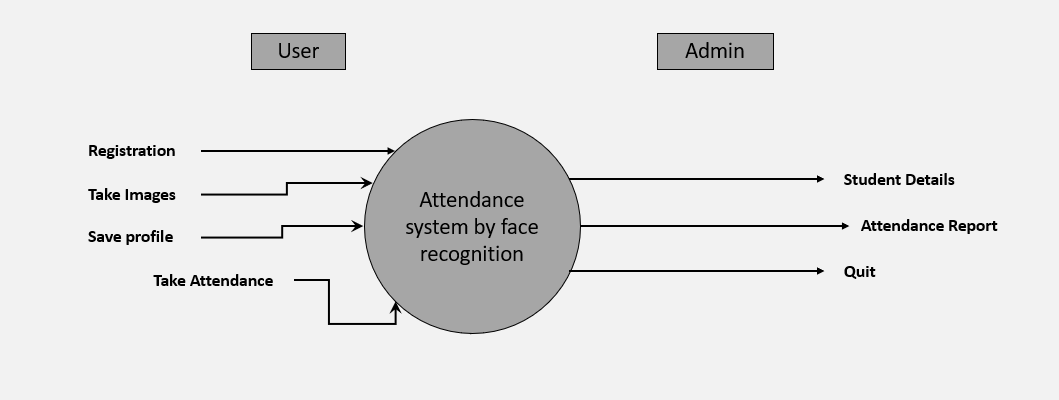
The data flow diagram may be used to represent a system or software at any level of abstraction. In fact, DFDs may be partitioned into levels that represent increasing information flow and functional detail. Therefore, the DFD provides a mechanism for functional modeling as well as information flow modeling.

**LIST OF SYMBOLS**

|  |  |  |
| --- | --- | --- |
| **SYMBOL** | **NAME** | **FUNCTION** |
|  | Data Flow | Connect process |
|  | Process | Perform some transformation of its input data to yield output data. |
|  | Source or sink | A source of system inputs or  sink of system outputs |
|  | Data Store | A repository of data, the  arrowhead indicate net  input and net outputs  to store |

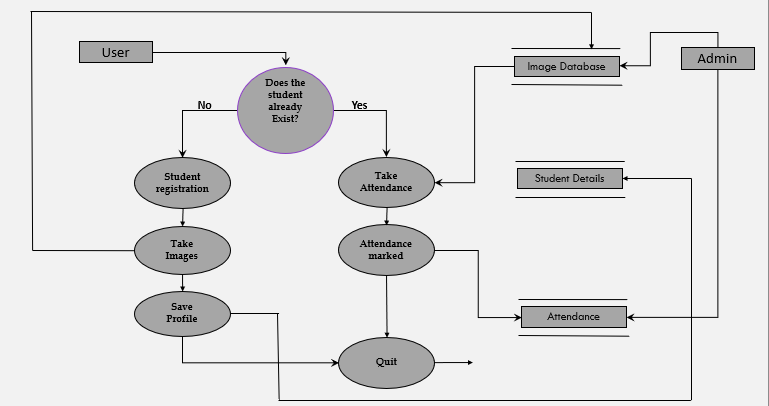
**3.3 Context Diagram**

A level 0 DFD is called fundamental system model or context model represents entire software element as a single bubble with input and output data indicating by incoming and outgoing arrows.



**FIGURE 7: CONTEXT DIAGRAM (DFD LEVEL 0)**

**3.4** **First Level DFD**



**FIGURE 8 : DFD LEVEL**

**CHAPTER – 4**

**DATABASE DESIGN**

**4.1 Introduction**

This article/tutorial will teach the basis of relational database design and explains how to make a good database design. It is a rather long text, but we advise to read all of it. Designing a database is in fact fairly easy, but there are a few rules to stick to. It is important to know what these rules are, but more importantly is to know why these rules exist, otherwise you will tend to make mistakes!

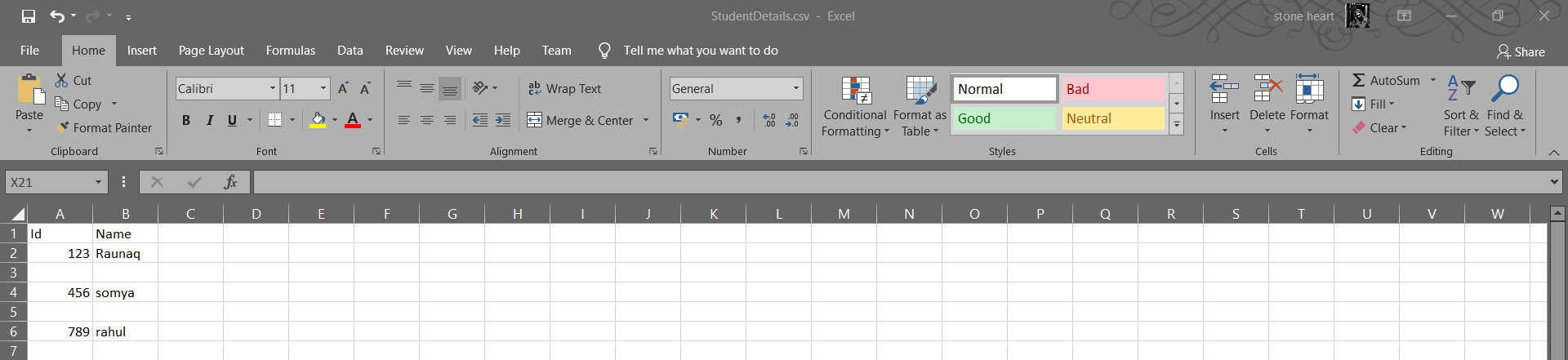
# 4.2 Database Fields Specification

A small write-up on the database, giving the fields, explaining each field etc should be written. This write-up can be evaluated on the following points.

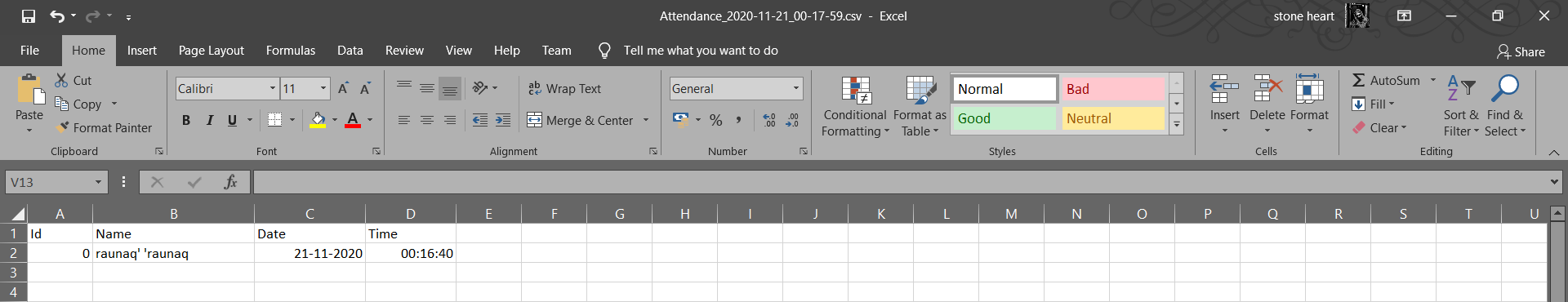
1. Clarity and conciseness of the database design. Like, whether key is defined, whether any redundant fields are there etc.
2. Whether data-storage calculations are made, and if so, whether they are done properly etc.
3. Whether any data-backup/recovery mechanism is discussed or being thought of.

Employee Number/Registration Number is the Key of the database. The range of valid values entered below as examples need not be taken as such. They can be modified by the team.

|  |  |
| --- | --- |
| **FILE NAME** | **DESCRIPTION** |
| *Studentdetails.csv* | Contains 2 columns (id and name) which will store information of every registered student.  w.r.t fig |
| *Attendance\_date\_time.csv* | Contains 4 columns (id, name, date and time) which will contain the attendance.  w.r.t fig |



**2.**

****

**4.3 Entity-Relationship Diagram**

It is a detailed logical representation of data for an organization and uses three main constructs.

**Entities**

It is a fundamental thing about which data may be maintained. Each entity has its own identity.

Entity Type is the description of all entities to which a common definition and common relationships and attributes apply.

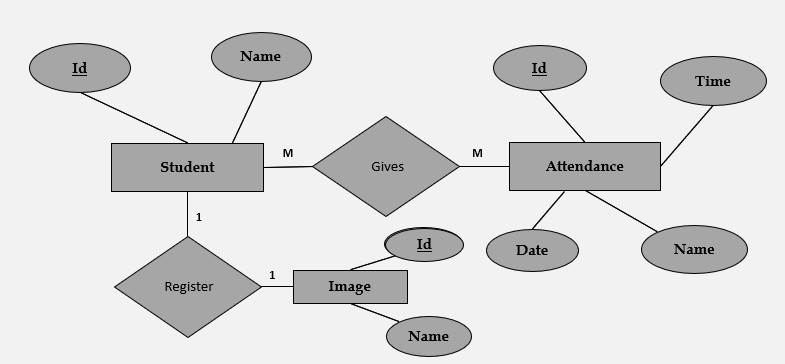
## 

## Attributes

Attributes are properties of entities. Attributes are represented by means of eclipses. Every eclipse represents one attribute and is directly connected to its entity (rectangle).

## Relationship

Relationships are represented by diamond shaped box. Name of the relationship is written in the diamond-box. All entities (rectangles), participating in relationship, are connected to it by a line.



**CHAPTER-5**

**TESTING**

**5.1 Introduction**

Executing a program with the intent of finding errors is called testing. Testing is vital to the success of any system. Testing is done at different stages within the development phase. System testing makes a logical assumption that if all parts of the system are correct, the goals will be achieved successfully. Inadequate testing or no testing at all leads to errors that may come up after a long time when correction would be extremely implementation. The testing of the system was done on both artificial and live data. In order to test data test cases are developed. Following are the various methods that are employed for testing:

# 5.2 Methods employed for Testing

**5.2.1 Unit Testing**

In unit testing the module is tested independently. It is done to test that the module does satisfy the functional specification. This is done to check syntax and logical errors in programs. At the time of preparation of technical specifications, unit test data was also prepared. The coding for that program was considered after verifying its output against this test data.

Following are the unit testing methods:

1. In Conditional Testing, the logical conditions that are given in the module were checked to see whether they satisfy the functionality of the module. This is done by using the test data was prepared.
2. In Loop Testing, different loops in the module like nested loops were tested using the data. Attempts to execute the loops to their maximum range are done.

# 

# 5.2.2 Integration Testing

In Integration testing whole system was checked when all the individual modules were integrated together in order to test whether the system is performing as according to the requirements specified. Interface errors if any were corrected. Test data was prepared was fed into the system to check whether the system fails to detects an error.

**5.2.3 Functional Testing**

This is done for each module/sub module of the system. Functional testing serve as a means of validating wheatear the functionality of the system confers the original user requirement i.e. does the module do what it was supposed to do? Separate schedules were made for functional testing. It involves preparation of test data, writing test cases, testing for conformance to test cases and preparation of bug’s listing for non-conformities.

**5.2.4 System Testing**

System testing is done when the entire system has been fully integrated. The purpose of the system testing is to test how the different modules interact with each other and whether the entire system provides the functionality that was expected.

System testing consists of the following steps:

* **Program Testing**
* **System Testing**
* **System Documentation**
* **User Acceptance Testing**

# 5.3 Test-Cases (TC)

The test-cases is basically a list of test cases that need to be run on the system. Some of the test cases can be run independently for some components (report generation from the database, for example, can be tested independently) and some of the test cases require the whole system to be ready for their execution. It is better to test each component as and when it is ready before integrating the components.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| NO. | ACTION | INPUT | EXPECTED OUTPUT | ACTUAL OUTPUT | RESULT |
| 1 | REGISTRATION | Person’s Id and name | Id and Name will be stored in the database | Id and Name will be stored in the database | Passed |
| 2 | TAKE IMAGE AND SAVE | Take images and stores the images of a face | Images are captured and stored | Images are captured and stored | Passed |
| 3 | TAKE ATTENDANCE | A live stream of a person’s face | Name of detected person is displayed on the screen | Name of detected person is displayed on the screen | Passed |
| 4 | VIEW AND QUIT | Attendance will be stored in the database | Attendance will be stored in the database | Attendance will be stored in the database | Passed |

**CHAPTER-6**

**ROLES AND RESPONSIBILITIES**

## 6.1. Project Roles and Responsibilities

**6.1.1 ROLE**

* Worked as a developer, designer, tester of the application.

**6.1.2 RESPONSIBILITIES**

* Work on definition of development requirements and priorities.
* Data migration.
* Interfaces with other systems.
* Reporting configuration and deployment.
* Set up and maintence of security rights and access permission.
* Contributing to technical strategy, policy and procedure.
* Development and operation of technical testing programmers.
* Production of technical documentation to agreed quality standards.
* Reporting on progress/issues to management and users.

**CHAPTER-7**

**CONCLUSION**

**AND**

**ACHIEVEMENTS**

**7.1 CONCLUSION**

This system eliminates classical student identification such as calling name of the student, or checking respective identification cards of the student, which can not only interfere with the ongoing teaching process, but also can be stressful for students during examination sessions. In addition, the students have to register in the database to be recognized.

In this approach, a face recognition based automated student attendance system is thoroughly described. The proposed approach provides a method to identify the individuals by comparing their input image obtained from recording video frame with respect to train image. This proposed approach able to detect and localize face from an input facial image, which is obtained from the recording video frame. Besides, it provides a method in pre-processing stage to enhance the image contrast and reduce the illumination effect. Extraction of features from the facial image is performed by applying LBH algorithm. The accuracy of this proposed approach is somewhere around 81%

**7.2 Future Scope**

As face provides a unique identity of a person, it can be used to identify a person and verify his/her identity. Face recognition provides non-intrusive way to recognize a person. By using this system, the chances of fake attendance and proxies can be reduced.

Automated Attendance System can be implemented in larger areas like in a seminar hall where it helps in sensing the presence of many people.

Sometimes the poor lighting condition of the classroom may affect image quality which indirectly degrades system performance, this can be overcome in the latter stage by improving the quality of the video or by using some algorithms

**Appendices**

#############

import tkinter as tk

from tkinter import ttk

from tkinter import messagebox as mess

import tkinter.simpledialog as tsd

import cv2,os

import csv

import numpy as np

from PIL import Image

import pandas as pd

import datetime

import time

############################################# FUNCTIONS ################################################

def assure\_path\_exists(path):

    dir = os.path.dirname(path)

    if not os.path.exists(dir):

        os.makedirs(dir)

##################################################################################

def tick():

    time\_string = time.strftime('%H:%M:%S')

    clock.config(text=time\_string)

    clock.after(200,tick)

###################################################################################

def contact():

    mess.\_show(title='Contact us', message="Please contact us on : '[nidhivermaa91@gmail.com](mailto:nidhivermaa91@gmail.com)' ")

###################################################################################

def check\_haarcascadefile():

    exists = os.path.isfile("haarcascade\_frontalface\_default.xml")

    if exists:

        pass

    else:

        mess.\_show(title='Some file missing', message='Please contact us for help')

        window.destroy()

##########################################################

def TrainImages():

    check\_haarcascadefile()

    assure\_path\_exists("TrainingImageLabel/")

    recognizer = cv2.face\_LBPHFaceRecognizer.create()

    harcascadePath = "haarcascade\_frontalface\_default.xml"

    detector = cv2.CascadeClassifier(harcascadePath)

    faces, ID = getImagesAndLabels("TrainingImage")

    try:

        recognizer.train(faces, np.array(ID))

    except:

        mess.\_show(title='No Registrations', message='Please Register someone first!!!')

        return

    recognizer.save("TrainingImageLabel\Trainner.yml")

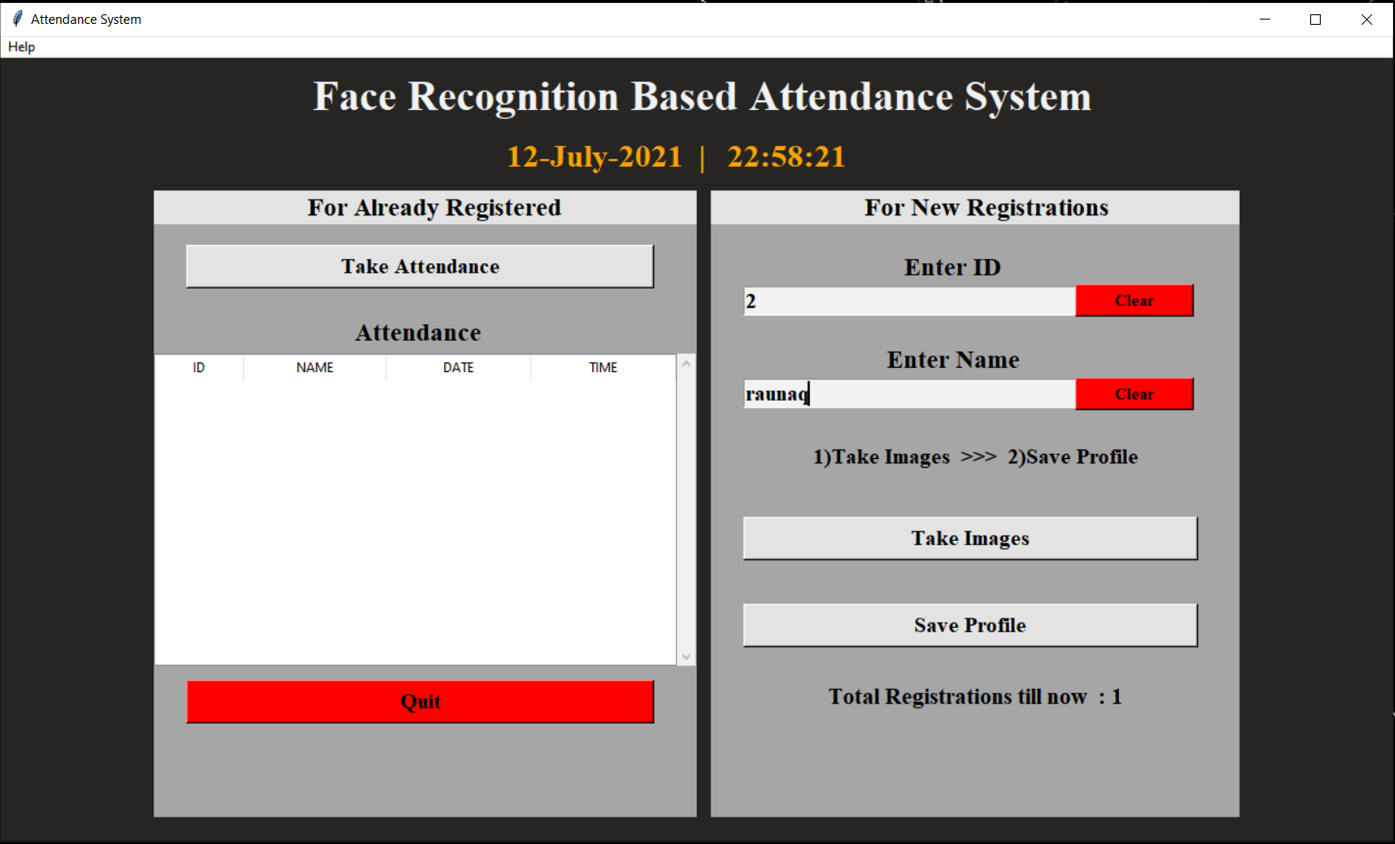
    res = "Profile Saved Successfully"

    message1.configure(text=res)

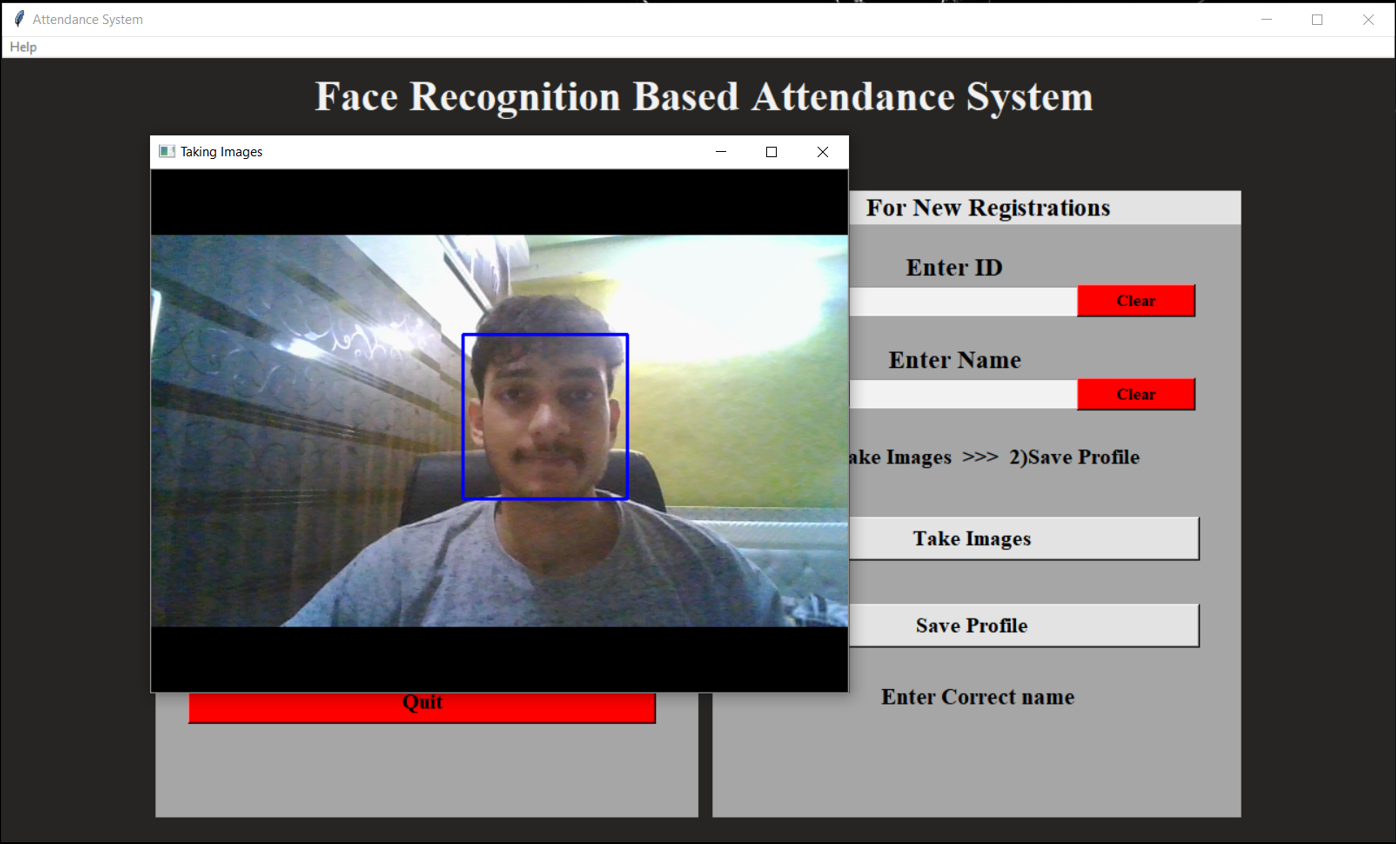
    message.configure(text='Total Registrations till now  : ' + str(ID[0]))

**SCREENSHOTS**

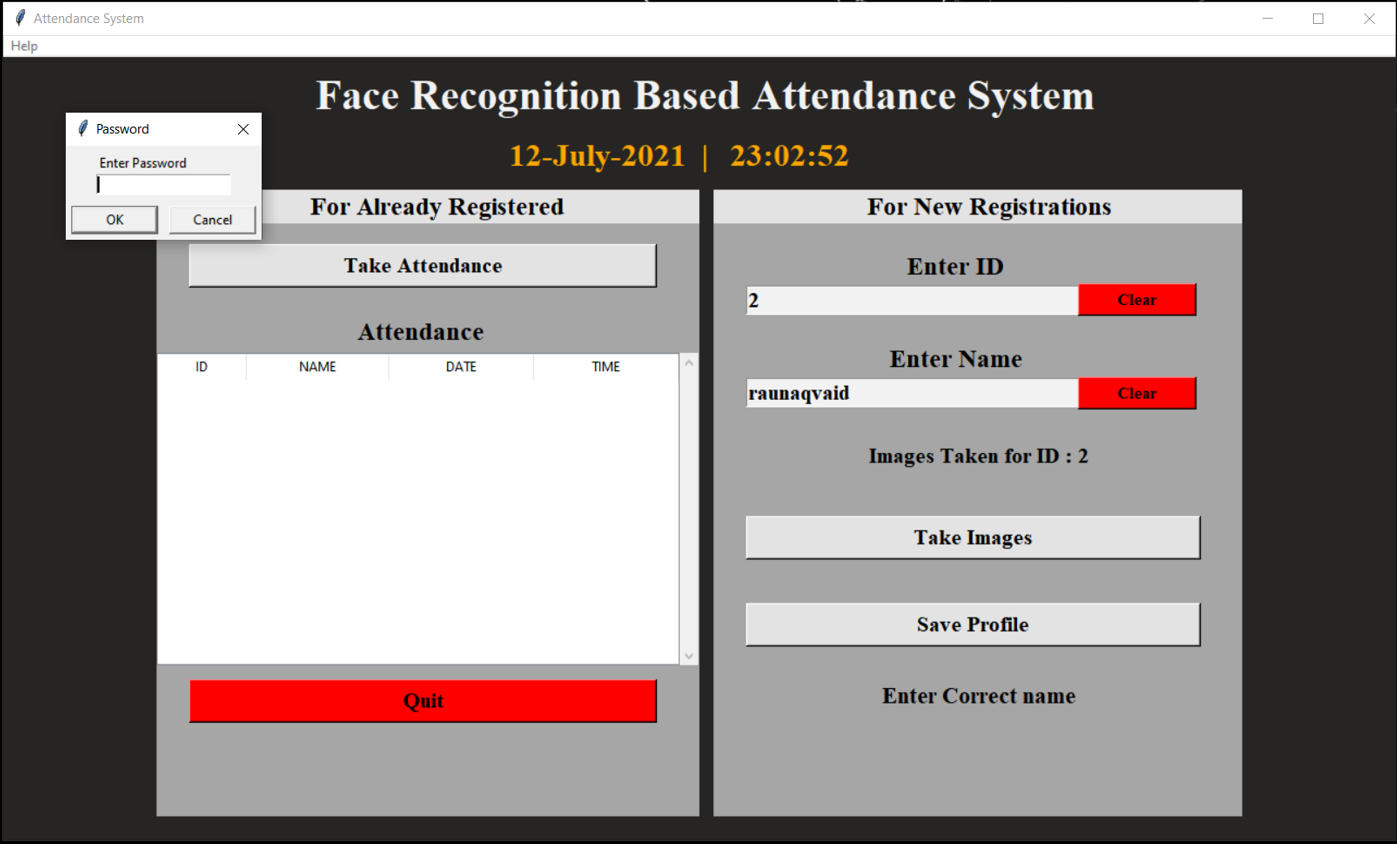
**1.**

****

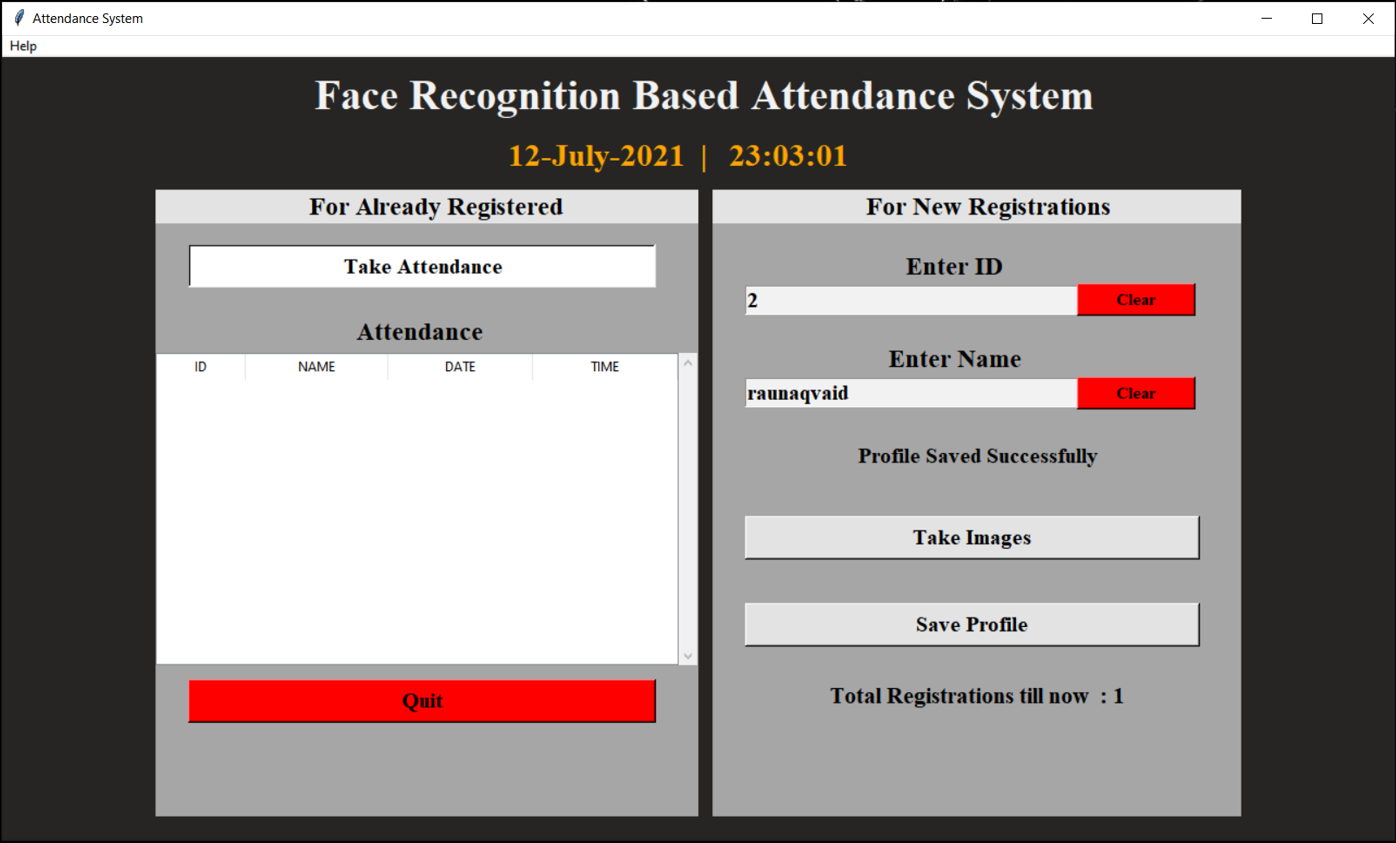
**2.**

****

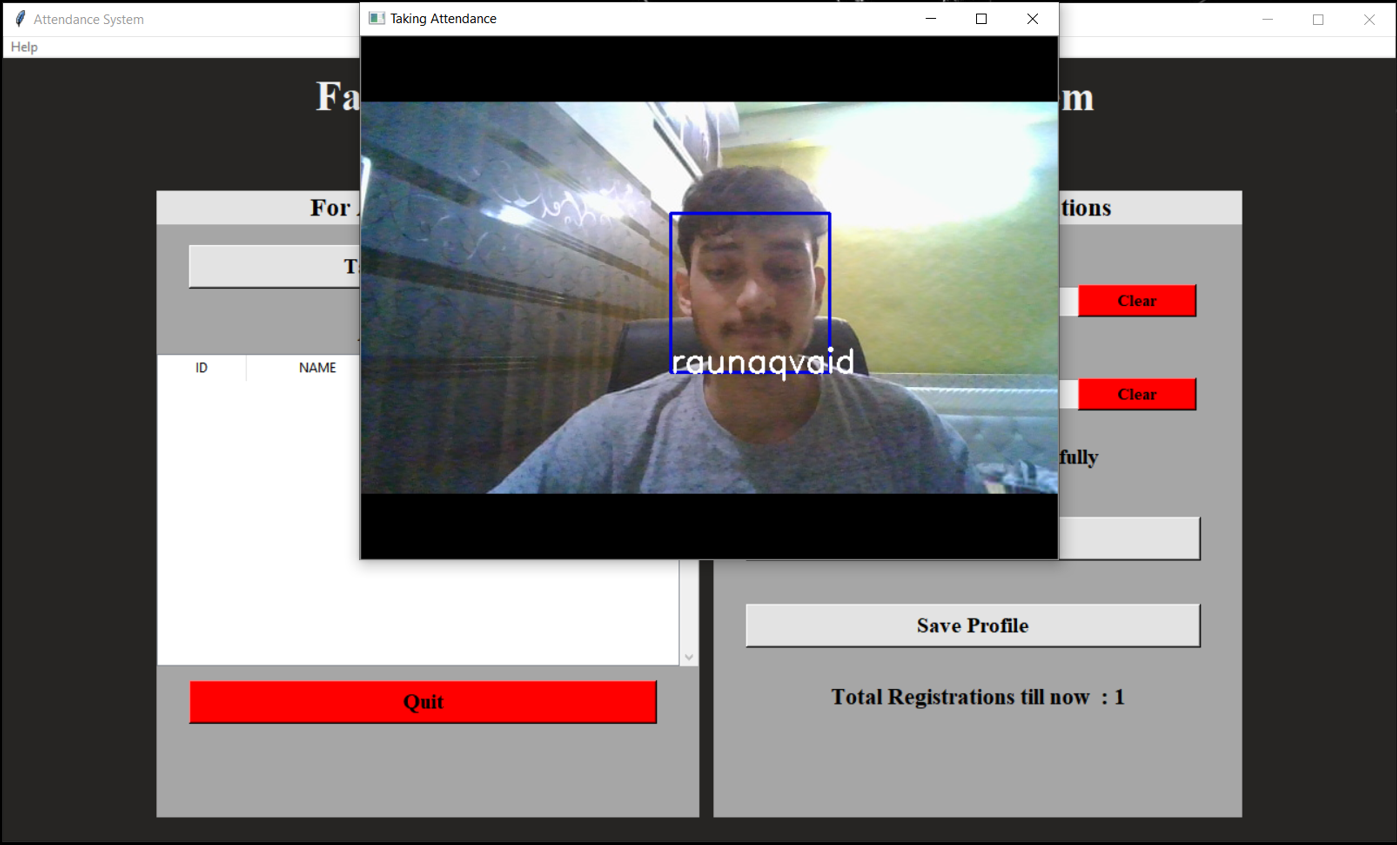
**3.**

****

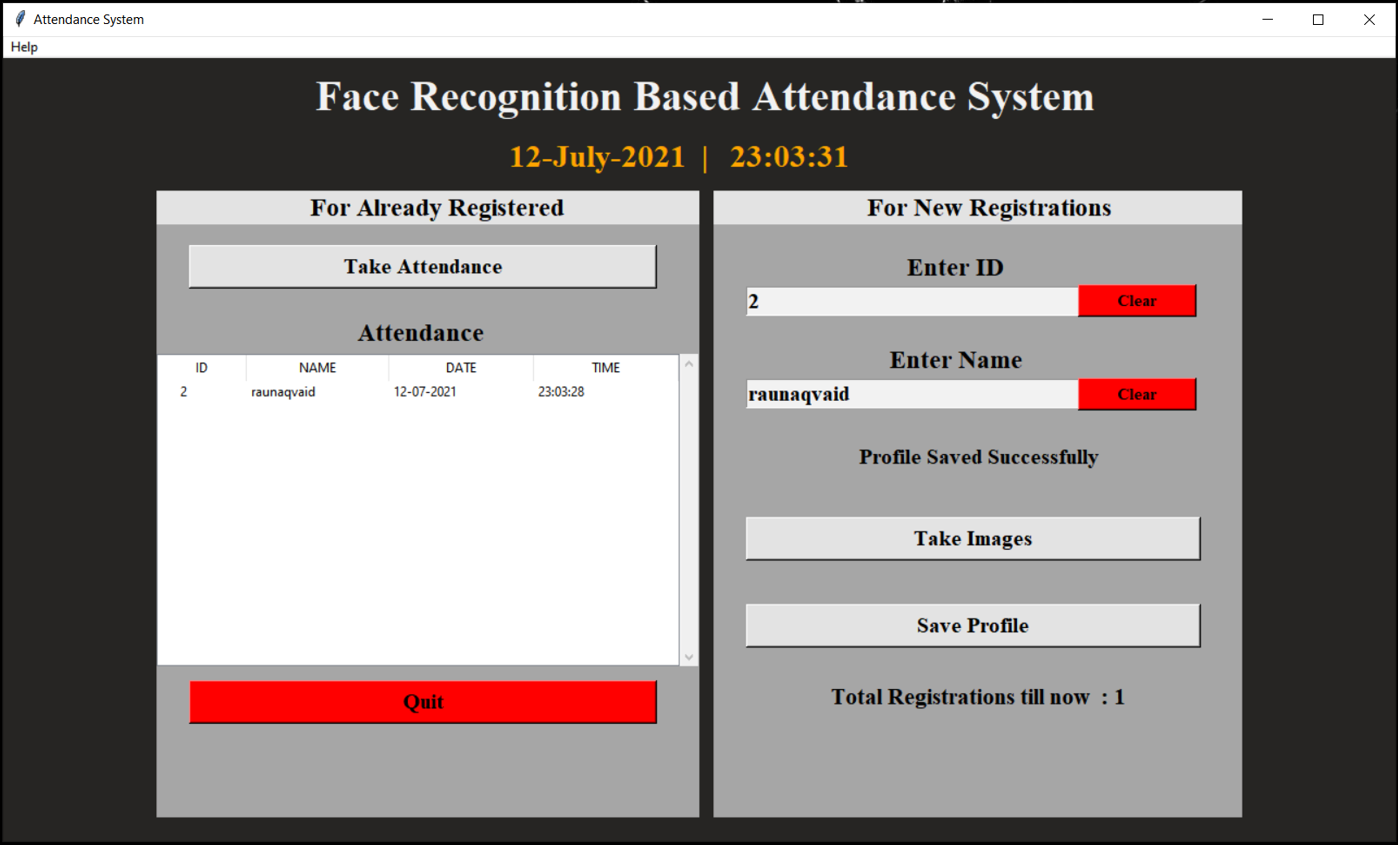
**4.**

****

**5.**

****

**6.**

****

**References**

[1]-Software Engineering by K.K Aggarwal

[2]-Software Engineering: A Practitioner’s approach’’ by Roger S. Pressman

[3]-https://www.tutorialspoint.com/sdlc/sdlc\_waterfall\_model.htm

[4]- <https://www.pyimagesearch.com/2018/06/18/face-recognition-with-opencv-python-and-deep-learning/>

[5]- <https://pypi.org/project/face_recognition/>

[6]- <https://towardsdatascience.com/a-guide-to-face-detection-in-python-3eab0f6b9fc1>

[7]- <https://google-images-download.readthedocs.io/en/latest/arguments.html>

[8]- Machine Learning for Absolute Beginners: A Plain English Introduction

Book by Oliver Theobald