A Project Report

on

AUTOMATED ATTENDANCE SYSTEM USING IMAGE PROCESSING

by

GORASIYA KASHYAP (IT037) (18ITUEF014)

DHAMELIYA ARPIT V. (EC017) (19ECUOS022)

FENIL RANGANI (IT130) (19ITUOS086)

B. Tech. SEM. VII

Faculty Supervisor
Prof. Sunil Vithlani

Bachelor of Technology

Degree of Information Technology

Submitted To

Department of Information Technology

Faculty of Technology,

Dharmsinh Desai University, Nadiad-387001.

ABSTRACT

The uniqueness or individuality of an individual is his face. 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 a 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 are used. Face detection is used to locate the position of the 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. Face detection and recognition is performed using Haar-Cascade classifier and Local Binary Pattern Histogram algorithm respectively. Faces are detected and recognized from live image of the classroom. This proposed system will be implemented with 4 phases such as Image Capturing, Segmentation of group image and Face Detection, Face comparison and Recognition, Updating of Attendance in database.

ACKNOWLEDGMENT

It has been a great privilege to work, right from the conceptualization of the topic of the report work of this project, under the guidance of my respected guide **Prof. Sunil Vithlani** Faculty in Information Technology Engineering Department.

We fill very much elevated in preparing the project report on **AUTOMATIC ATTENDANCE SYSTEM**. Despite our best efforts it is possible that some errors and mistakes may have gone unnoticed. We shall be great to the faculty who will kindly bring these mistakes to our notice. Suggestions from the faculties are most welcomed.

We would also like to thank the Head of Department **Dr. Vipul Dabhi & DDU SSIP CELL**, for their support during the work schedule. Finally, & most importantly, we would like to thank our friends and families for their continuous support throughout all our endeavours.

TABLE OF CONTENTS

CHAPTER:1

SYSNOPSIS

In this digital era, face recognition system plays a vital role in almost every sector. Face recognition is one of the mostly used biometrics. It can used for security, authentication, identification, and has got many more advantages. Despite of having low accuracy when compared to iris recognition and fingerprint recognition, it is being widely used due to its contactless and non-invasive process. Furthermore, face recognition system can also be used for attendance marking in schools, colleges, offices, etc. This system aims to build a class attendance system which uses the concept of face recognition as existing manual attendance system is time consuming and cumbersome to maintain. And there may be chances of proxy attendance. Thus, the need for this system increases. This system consists of four phases- dataset creation, face detection, face recognition, attendance updation. Dataset is created by the images of the students in class. Face detection and recognition is performed using Machine Learning and FaceNet algorithm respectively. Faces are detected and recognized from live streaming video of the classroom. Attendance will be saved in MS-Excel file.

Traditional method of attendance marking is a tedious task in many schools and colleges. It is also an extra burden to the faculties who should mark attendance by manually calling the names of students or passing the attendance sheet which might take about 5 minutes or more of entire session. This is time consuming. There are some chances of proxy attendance. Therefore, many institutes started deploying many other techniques for recording attendance like use of Radio Frequency Identification (RFID), iris recognition, fingerprint recognition, and so on. However, these systems are queue based which might consume more time and are intrusive in nature.

Face recognition has set an important biometric feature, which can be easily acquirable and is non-intrusive. Face recognition-based systems are relatively oblivious to various facial expression. Face recognition system consists of two categories: verification and face identification. Face verification is a 1:1 matching process, it compares face image against the template face images and whereas is a 1: n problems that compares a query face images. The purpose of this system is to build an attendance system which is based on face recognition techniques. Here face of an individual will be considered for marking attendance. Nowadays, face recognition is gaining more popularity and has been widely used.

CHAPTER 2

PROBLEM DEFINITION

2.1 Existing System and its Origin:

In the beginning, the model focuses on how face recognition incorporated with radio frequency identification (RFID) detect the authorized students and counts as they get in and get out form the classroom. The system keeps the authentic record of every registered student. The system also keeps the data of every student registered for a particular course in the attendance log and provides necessary information according to the need. After that they have designed and implemented an attendance system which uses iris biometrics. Initially, the attendees were asked to register their details along with their unique iris template. At the time of attendance, the system automatically took class attendance by capturing the eye image of each attendee, recognizing their iris, and searching for a match in the created database. The prototype was web based.

After that, authors proposed an attendance system based on facial recognition. The algorithms like Viola-Jones and histogram of oriented gradients (hog) features along with support vector machine (SVM) classifier were used to implement the system. Various real time scenarios such as scaling, illumination, occlusions and pose was considered by the authors. Quantitative analysis was done on the basis of peak signal to noise ratio (PSNR) values and was implemented in MATLAB GUI.

Further authors researches to get best facial recognition algorithm (Eigenface and Fisherface) provided by the open cv 2.4.8 by comparing the receiver operating characteristics (roc) curve and then implemented it in the attendance system. Based on the experiments carried out in this paper, the roc curve proved that, eigenface achieves better result than FisherFace. System implemented using eigenface algorithm achieved an accuracy rate of 70% to 90%.

Thereafter, authors proposed a method for student attendance system in classroom using face recognition technique by combining discrete wavelet transforms (DWT) and discrete cosine transform (DCT). These algorithms were used to extract the features of students' face followed by applying radial basis function (RBF) for classifying the facial objects. This system achieved an accuracy rate of 82%.

2.1.1 Disadvantages of current systems:

The following are the disadvantages of current system:

1. It is difficult to maintain important information in books or in registers.
2. More manual hours need to generate required reports.
3. It is tedious to manage a greater number of faces for the attendance.
4. Current systems can take only up to 8 faces at the same time.

2.2 Proposed System:

All the students of the class must register themselves by entering the required details and then their images will be captured and stored in the dataset. During each session, faces will be detected from live streaming video of classroom. The faces detected will be compared with images present in the dataset. If match found, attendance will be marked for the respective student. At the end of each session, list of present and absent students will be generated. This system takes a single photo of the class using IP-camera. Then faces will be cropped and processed for the attendance. This system uses FaceNet_keras.h5 model which have been trained using Machine Learning. The system is integrated with website and can work in local area network.

2.2.1 Advantages: The following are the advantages of **proposed system** :

1. Easy to take attendance just by a single click.
2. Can generate attendance reports automatically.
3. Easy to maintain.
4. Centralized database helps in avoiding conflicts.
5. Easy to use website that does not requires specific training.

CHAPTER 3

REQUIREMENTS

3.1 Functional Requirements:

1. The System holds all the details of the all the students for which attendance has to be taken.

2. It holds the datasets of all the students.

3. The system allows the admin to login and manage the attendance.

4. It also allows them to view the list of all students and teacher databases.

5. The system allows the teacher to directly mark presence or absence of any individual candidate even if he/she is not present in the class.

6. It requires to train the dataset after which it will generate the dataset.npz and embeddings.npz model files.

7. Training model requires quite a processing unit and time.

8. Website runs on HTML,CSS,JS.

9. Whenever attendance is taken then accordingly the attendance report will be automatically updated.

3.2 Non-Functional Requirements:

• Nonfunctional Requirements are characteristics or attributes of the system that can judge its operation. The following points clarify them:

• **Accuracy and Precision:** the system should perform its process in accuracy and Precision to avoid problems. But As Distance Between Camera and faces is more so some time it may not detect face accurately.

• **Modifiability:** the system should be easy to modify, any wrong should be correct.

• **Security:** the system should be secure and saving student's privacy.

• **Usability:** the system should be easy to deal with and simple to understand As We have provided user interface for ease.

• **Speed**

CHAPTER 4
TECHNICAL RESOURCES

4.1 Software Requirements

- **Software editor:** Visual Studio Code
- **Programming Language:** Python3.7
- **Database:** SQL Alchemy
- **Framework:** Flask and Jinja2
- **Model:** Facenet_keras.h5
- **Main Libraries:** TensorFlow and MTCNN
- **Web Development:** HTML, CSS, JS
- **Core OS:** Multiplatform OS support

CHAPTER 5
SYSTEM DESIGN

5.1 Use Case Diagram:

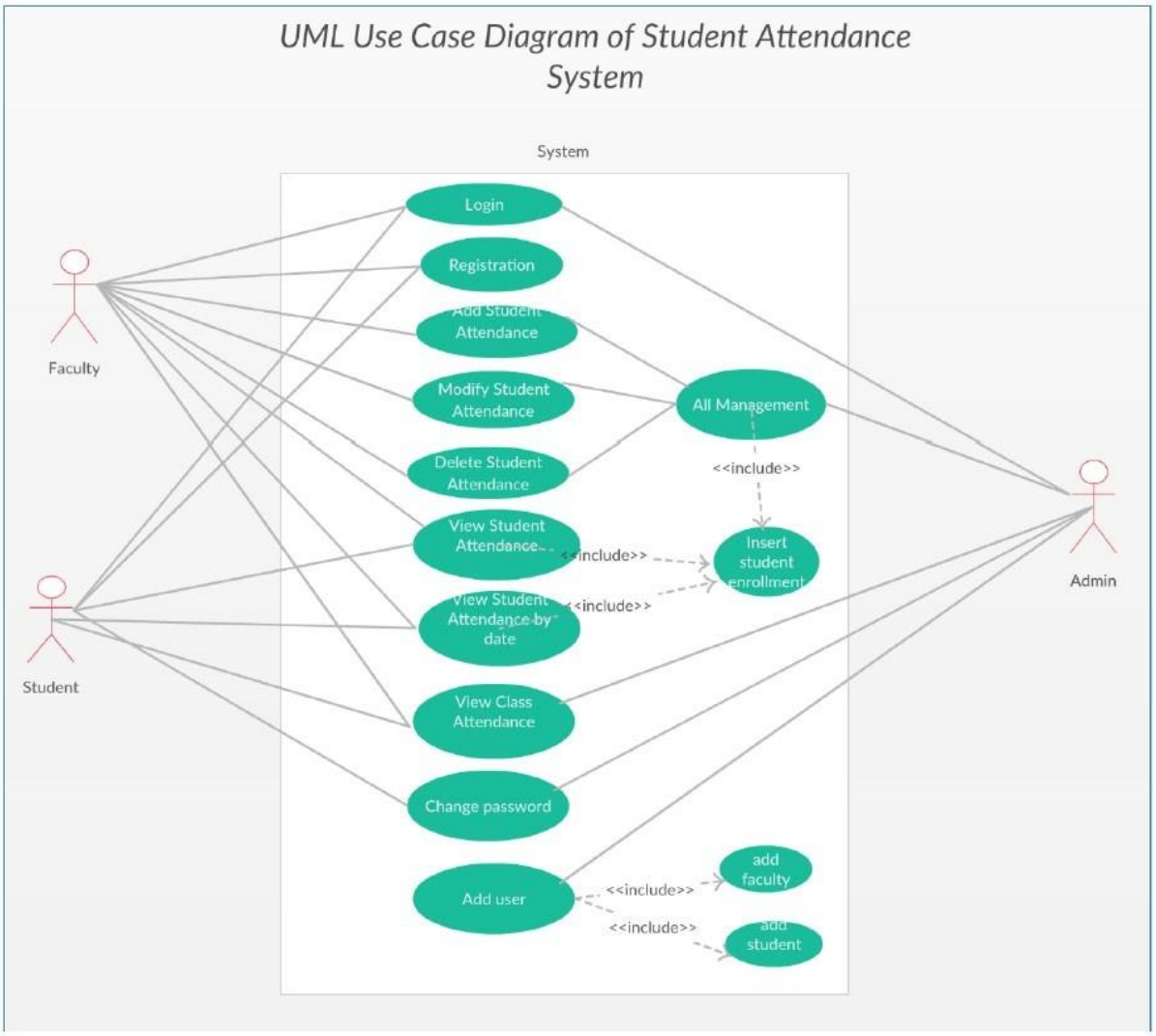


Fig.5.1 Use case Diagram

5.2 CLASS DIAGRAM:

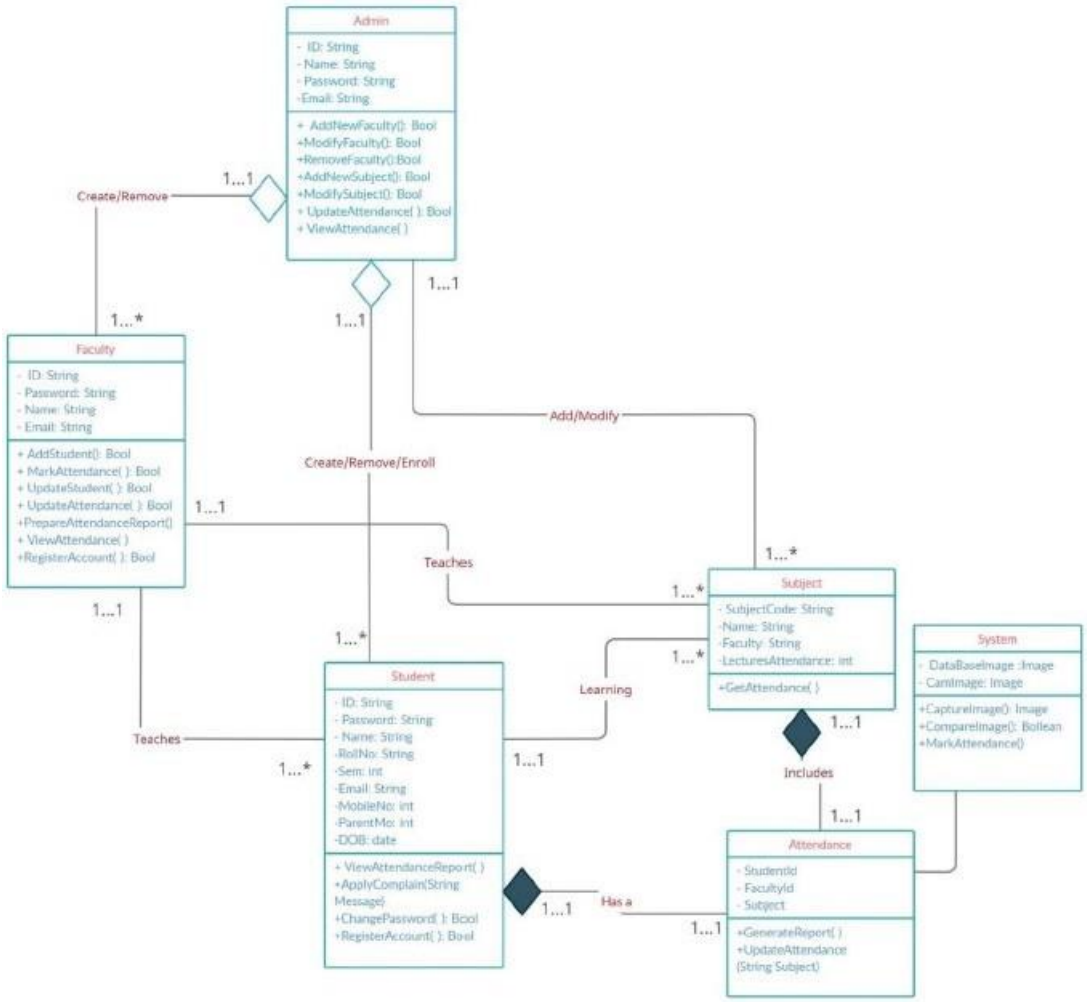


Fig. 5.2 Class Diagram

5.3 Flow Chart:

Flow Chart represents the flow of logic in any program or code.

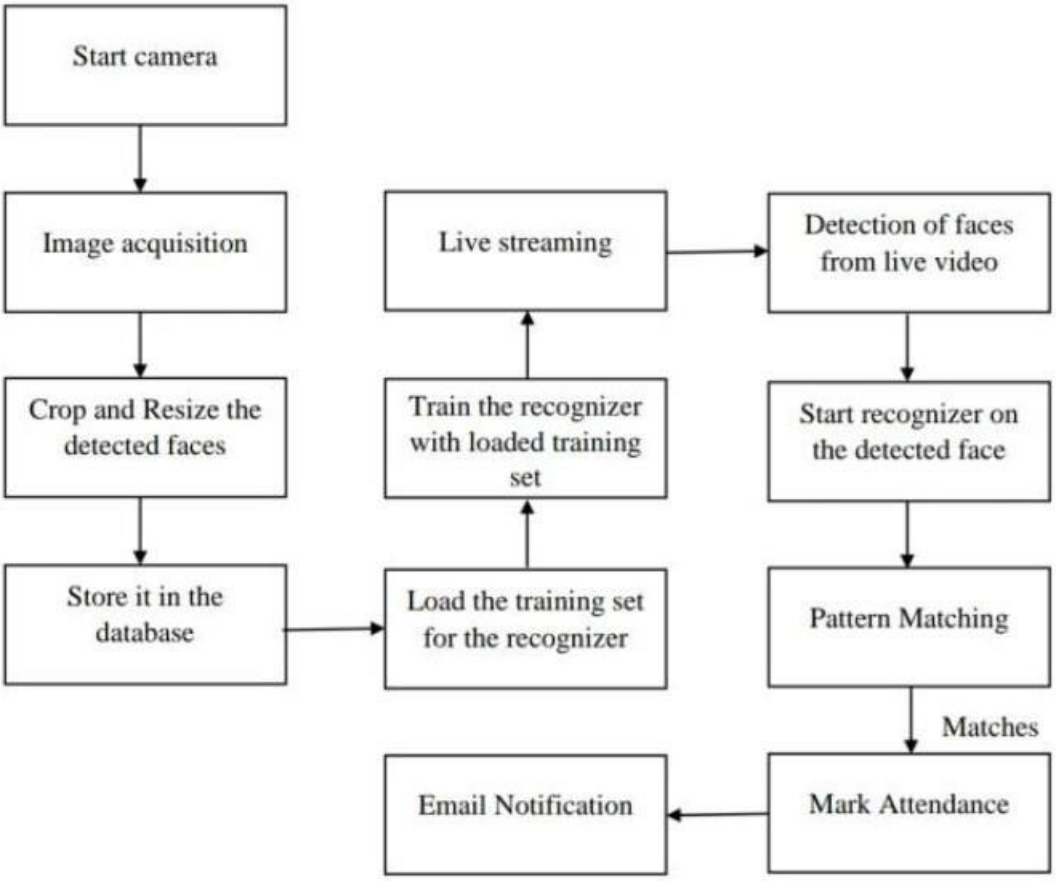


Fig.5.3 Flow Chart of Project

CHAPTER 6
IMPLEMENTATION PLANNING

6.1 IMPLEMENTATION ENVIRONMENT:

During the complete implementation we have worked on Visual Studio Code (VSCode) and Google Collab. Visual Studio Code is a free source-code GUI (Graphical User Interface) editor made by Microsoft for Windows, Linux and macOS. Google Collab allows anybody to write and execute arbitrary python code through the browser, and is especially well suited to machine learning, data analysis and education. Collab supports many popular machine learning libraries which can be easily loaded in your notebook. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality. It also provides extension to work in a team with a multi user experience. Even compiling and deploying of the application was done easily with the help of VScode.

For the database implementation, we have used SQL_Alchemy. Sql_alchemy is a powerful, open-source object-relational database system Specially designed for the Flask Framework Software Projects.

6.2 PROGRAM / MODULES SPECIFICATION:

Our entire system is divided into 6 modules:

1. **Detacting Face Module:**

This module will take image of whole class as input and it will detect faces from input image and detect coordinates of faces like eye, nose, lips, cheek. Here user will select the combined images of whole classroom and this module crop faces from those images. Give single faces as output and store it in some dir. (this module uses MTCNN model).

2. Creating Dataset Module

This module will be used to create data set. It will capture 100 images of each person automatically and store it at specified location which than furthered be used for training. This module uses HaarCaascade model for detect face and capture only faces avoiding background. We can use dataset for embedding and training.

3. Training Module

This module is for removing face coordinates from input training dataset. We will load dataset of each student for training and save all coordinates in compressed file NewDataset.npz.

4. Load model / extract embedding module

In this module we will load our pretrained model named Facenet and then by loading newdataset.npz obtained from train module we will extract embeddings (it is a list of 128 features of face of student) of each training face and store it in compressed file named Newembeddings.npz.

5. Add new student module

What if we want to add new student/ person in our classroom that's why we create this module. Using this module, we can create a new embedding for new student using his/her dataset. This will take very less time and embed new student in compressed file. FinalDataset.npz and FinalEmbedding.npz.

6. Take Attendance Module

This module will load face cropped Images in Detacting Face Module one by one and using FinalDataset.npz and FinalEmbedding.npz embedding files and classifier classified all person's classes comparing Features using SVM classifier give output as Attendance.csv file with names and timestamp of present student. This module is final module use for marking attendance. In this module we load coordinates file (DataSet.npz) and Embeddings file (Embeddings.npz). We also load cropped images obtained from module-1 one by one and compare its embeddings with already saved embeddings using SVM classifier and give output as label of matched embedding. This label will be the name of student.

6.3 CODING SNIPPETS

While writing our code we took the utmost care to follow the basic coding standards while writing a Python code like:

- Following proper naming conventions of local variables, global variables, constants and functions.
- Doing proper indentation.
- Proper error handling.
- Adding comments for better understanding.
- Proper Syntax

Here are some of the code snippets written during implementation:

6.3.1 Import:

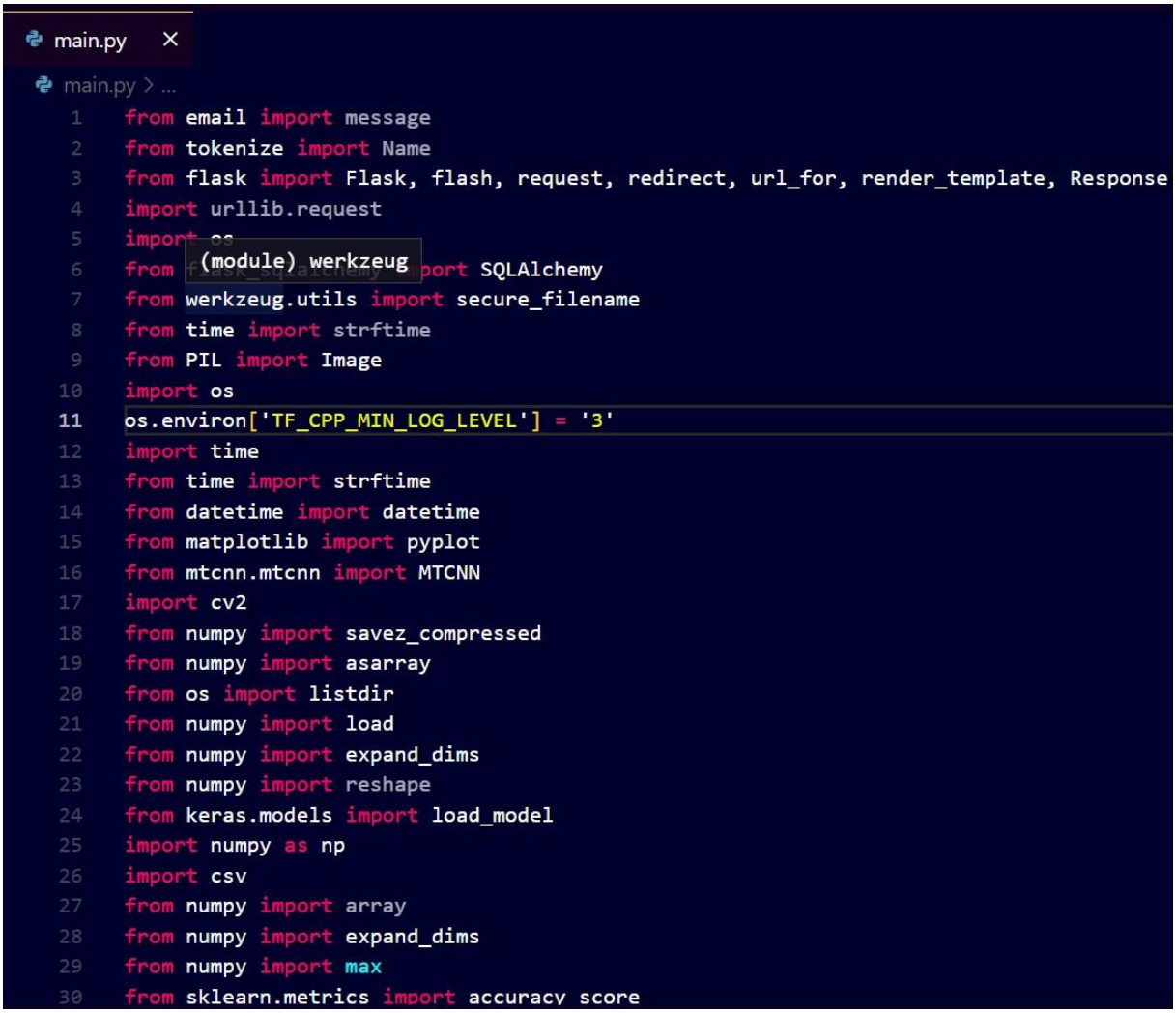


Fig. 6.1 Import Libraries

6.3.2 Main.py file:

main.py

```
266         std_list = []
267         #extracting embeddings
268         def extract_embeddings(model,face_pixels):
269             face_pixels = face_pixels.astype('float32')
270             mean = face_pixels.mean()
271             std  = face_pixels.std()
272             face_pixels = (face_pixels - mean)/std
273             samples = expand_dims(face_pixels,axis=0)
274             yhat = model.predict(samples)
275             return yhat[0]
276         directory='static/cropped/'
277         print(os.listdir('static/cropped/'))
278         dirs = os.listdir( directory )
279         model = load_model('facenet_keras.h5',compile=False)
280         data1 = load('DataSet.npz')
281         train_x,train_y = data1['arr_0'],data1['arr_1']
282
283         data = load('Embeddings.npz')
284         trainx,trainy= data['arr_0'],data['arr_1']
285
286         # i=0
287         name=[[]]
288         # data rows of csv file
289
290         global unfacecount
291         unfacecount = 0
292         for filename in dirs:
293             Img = directory + filename
294
```

Fig. 6.2 Take Attendance code

6.3.3 HTML code snippet:

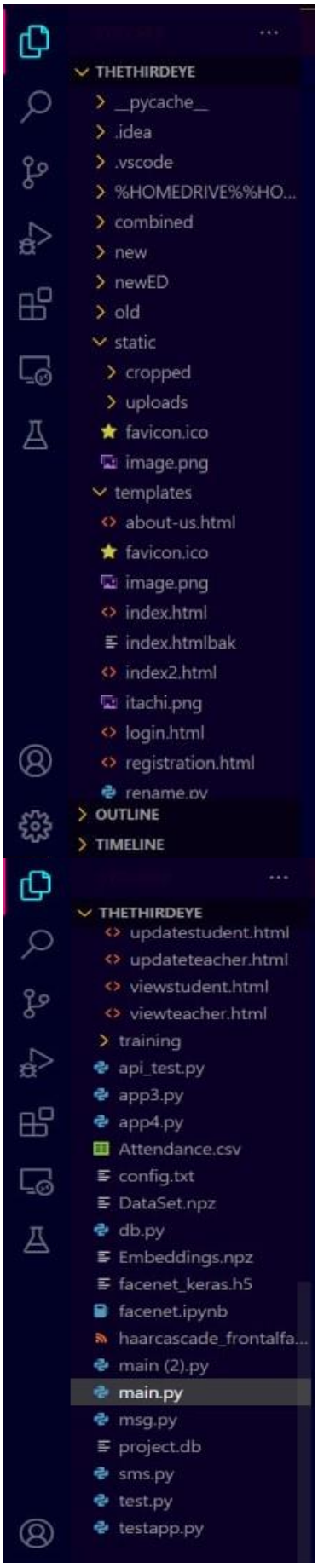
index.html

templates > index.html > html > body > div.bg-image > nav.navbar.navbar-expand-lg.navbar-dark.bg-dark > div.container-fluid > div#navbarSupportedC

```
<!doctype html>
<html lang="en">

<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>FaceNet-Attendance system</title>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0-beta1/dist/css/bootstrap.min.css" rel="stylesheet"
    integrity="sha384-0evHe/X+R7YkIZDRvuzKMRqM+OrBnVFBL6DOitfPri4tjfHxaWutUpFmBp4vmVor" crossorigin="anonymous">
</head>

<body>
  <div class="bg-image" style="background-image: linear-gradient(90deg, #00DBDE 0%, #FC00FF 100%);
        height: 250vh">

    <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
      <div class="container-fluid">
        <a class="navbar-brand" href="#">FaceNet</a>
        <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent"
          aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">
          <span class="navbar-toggler-icon"></span>
        </button>
        <div class="collapse navbar-collapse" id="navbarSupportedContent">
          <ul class="navbar-nav me-auto mb-2 mb-lg-0">
            <li class="nav-item">
              <a class="nav-link active" aria-current="page" href="#">Home</a>
            </li>
            <li class="nav-item">
              <a class="nav-link" href="/gen">Dataset</a>
            </li>
            <li class="nav-item dropdown">
              <a class="nav-link dropdown-toggle" href="#" id="navbarDropdown" role="button" data-bs-toggle="dropdown"
                aria-expanded="false">
                Select
```

Fig. 6.3 HTML for UI

6.3.4 Project Structure:



CHAPTER 7
TEST CASES

Various Test Case Scenarios:

ENTITY	TEST CASE	EXPECTED OUTPUT	ACTUAL OUTPUT	STATUS
Take Photo	Test Image	Taken photo using mobile camera High quality image.	Taken photo using mobile camera High quality image.	pass
Crop Faces From input images	Test Image	Input Image devide in Faces Images and saved in Folder	Input Image devide in Faces Images and saved in Folder	pass
Create a folder To Store Dataset Images	Creating dataset	Foder Create in Train Folder	Foder Create in Train Folder	pass
Generate Dataset	Creating dataset	Taken 100 images successfully To Embedding	Taken successfully 100 images successfully To Embedding	pass
Start embedding With facenet Model	Embedding Of Dataset	Give Compressed file Of 128 features of each student	Give Compressed file Of 128 features of each student	pass
Select the folder Of new Data set	Add new student	Select folder "new" in which we have new student's datasets	Select folder "new" in which we have new student's datasets	pass
Appending embedding	Add new student	It combines New students Embedding with Old students embeddings	It combines New students Embedding with Old students embeddings	pass
Remove old dataset	Add new student	New students dataset Remove from stored folder	New students dataset Remove from stored folder	pass

Generate FinalEmbedding	Final Embedding	Combine Old and new embeddings	Combine Old and new embeddings	pass
Generate FinalCompressed Dataset	Final Compressed Dataset	Combine old and new Compressed dataset	Combine old and new Compressed dataset	pass
Load cropped Images on by on	Take Attendance	Load Cropped faces Images Folder	Load Cropped faces Images Folder	pass
Load Model	Take Attendance	Load Model Load Compressed File Load Embedding	Load Model Load Compressed File Load Embedding	pass
Classified Whole Datasets	Take Attendance	Classified Faces using Embedding And Compressed Dataset File	Classified Faces using Embedding And Compressed Dataset File	pass
Matches Images	Take Attendance	Match Loaded Images with Classier Features	Match Loaded Images with Classier Features	pass
Mark attendance Create Attendance.csv	Take Attendance	Recognize Face Store name and timestamp In Csv file	Recognize Face Store name and timestamp In Csv file	
Exit	Exit	Exit from software	Exit from software	pass

CHAPTER 8
SOFTWARE GUI DESIGN

8.1 FRONT END WEBPAGE

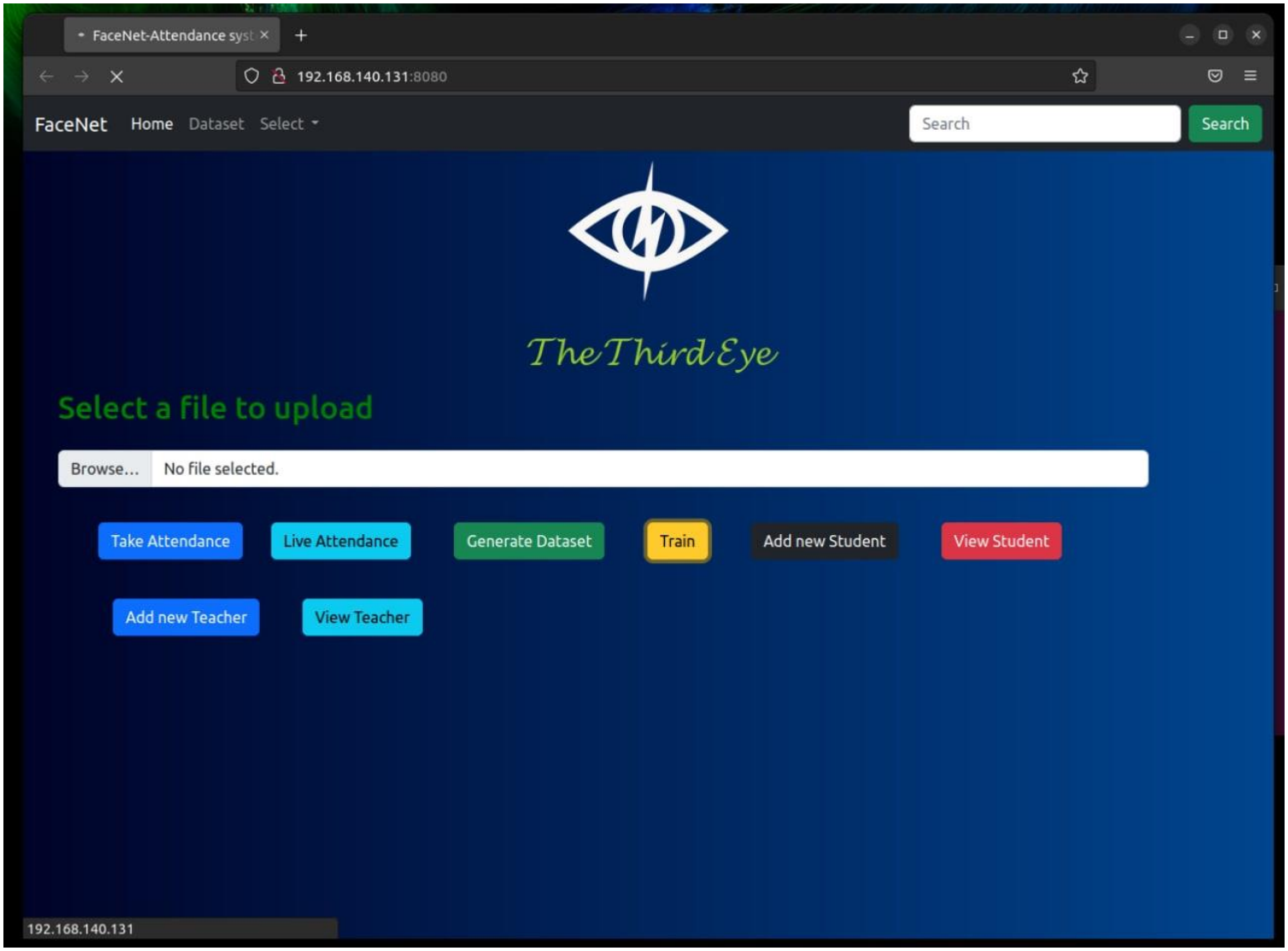


Fig.8.1 HTML homepage

8.2 View Student Database:

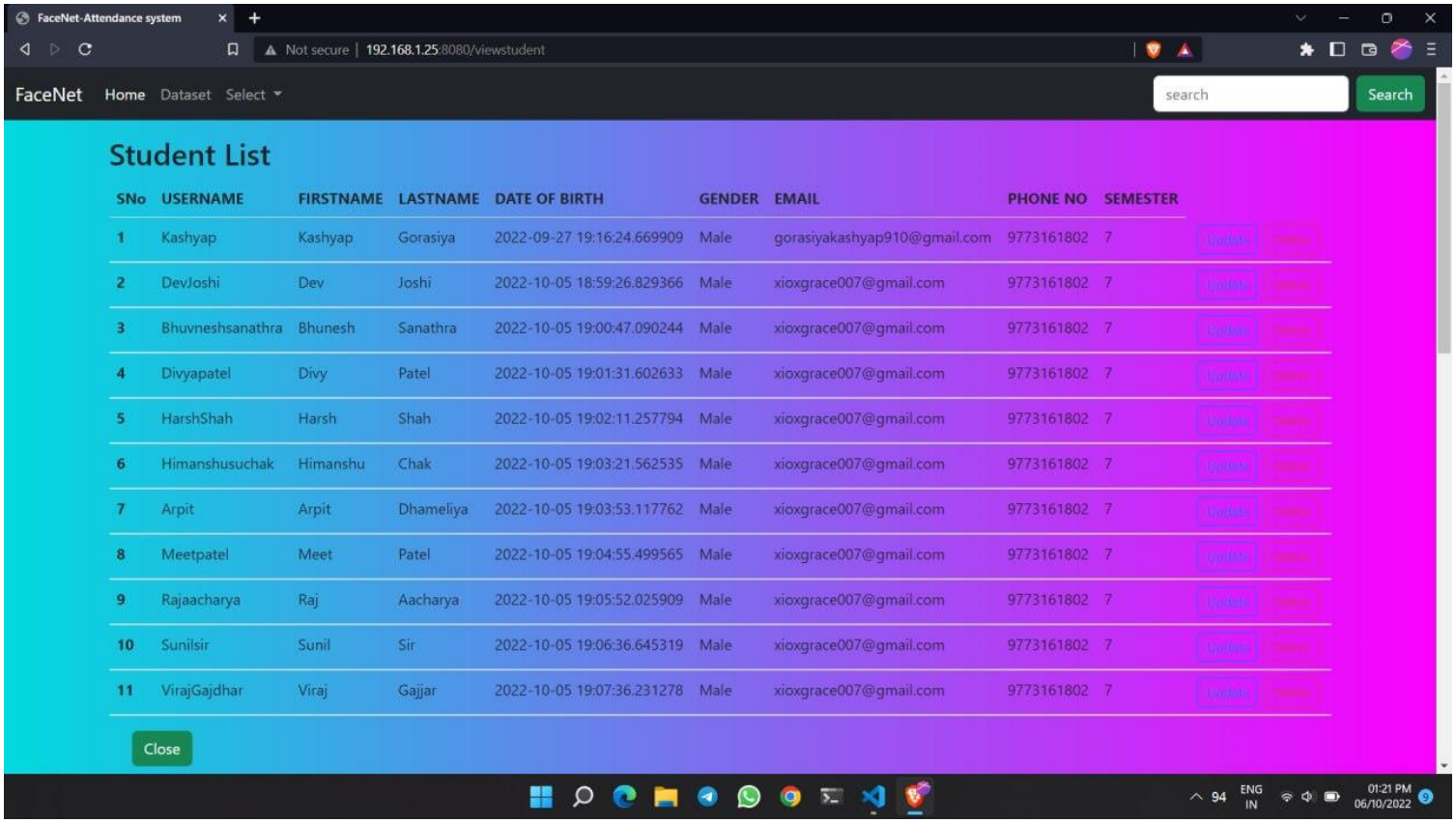
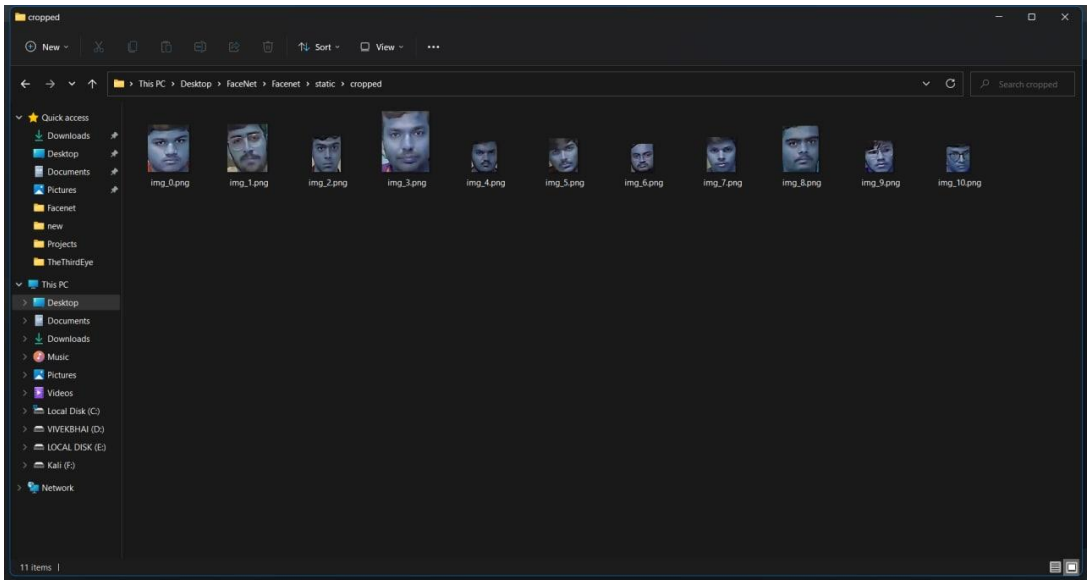


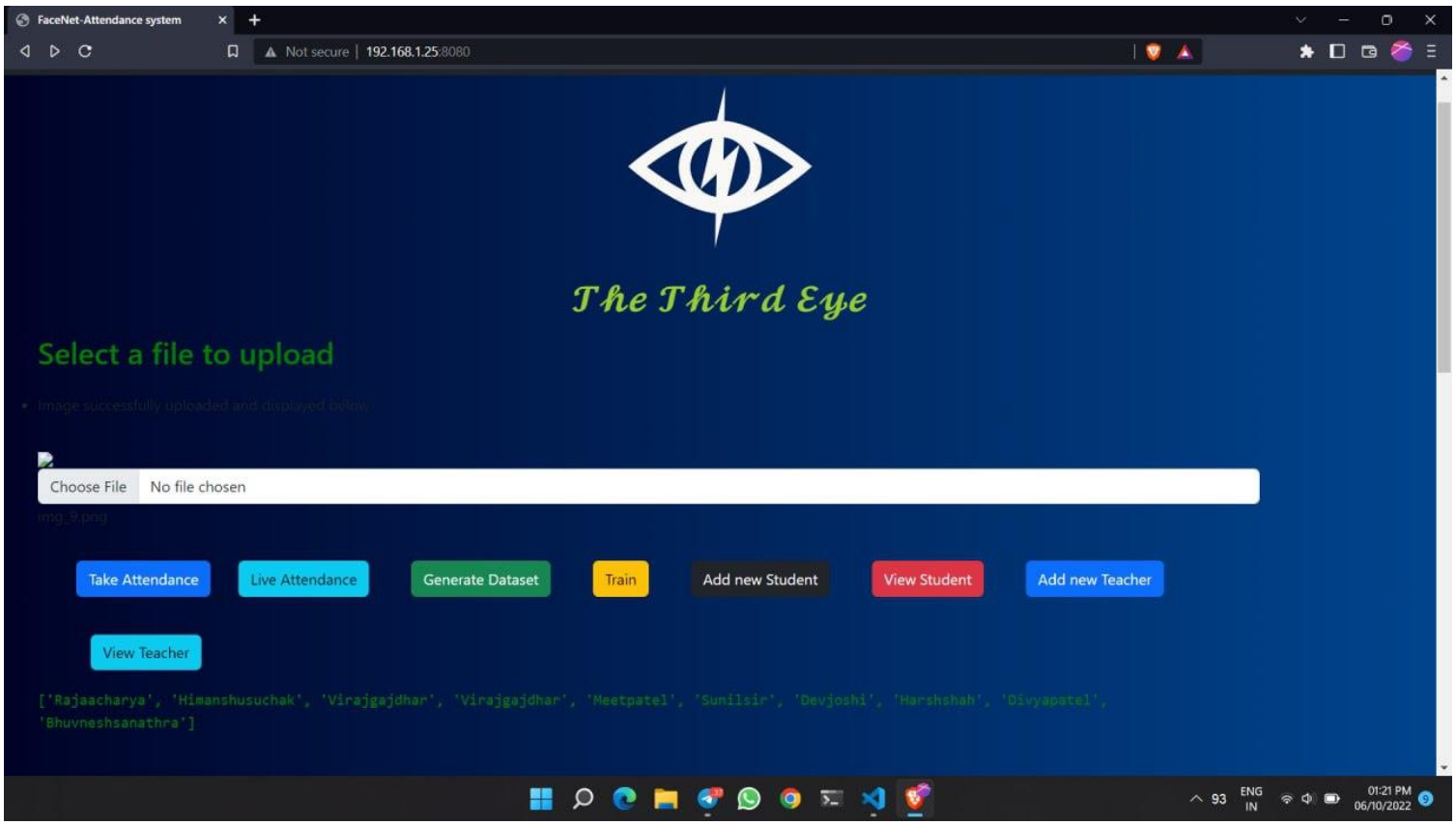
Fig. 8.2 Student database

8.3 Input Image For Attendance:

8.3 Cropped faces



8.4 ON TAKING ATTENDANCE:



8.5 GENERATED DATASET

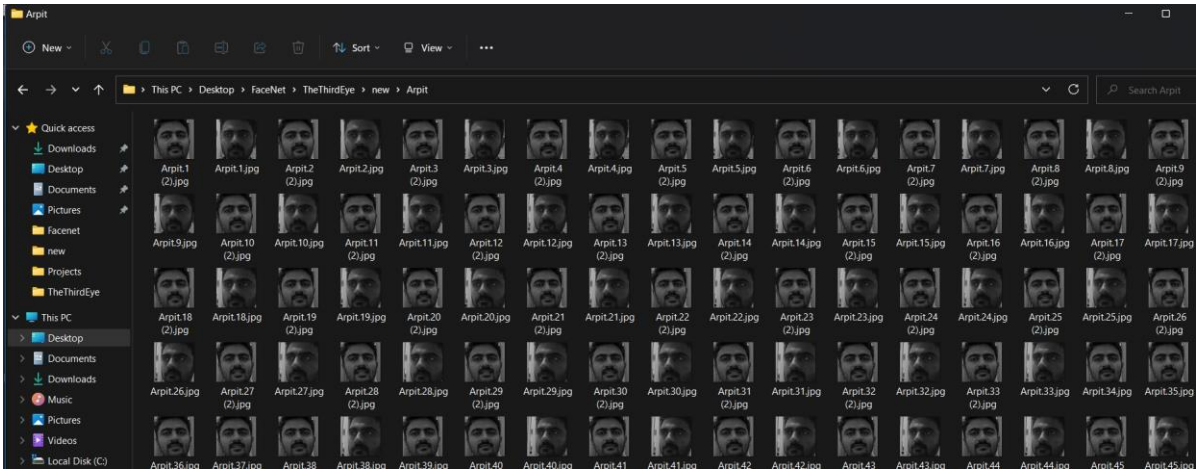


Fig. 8.5 dataset

8.6 Backend:

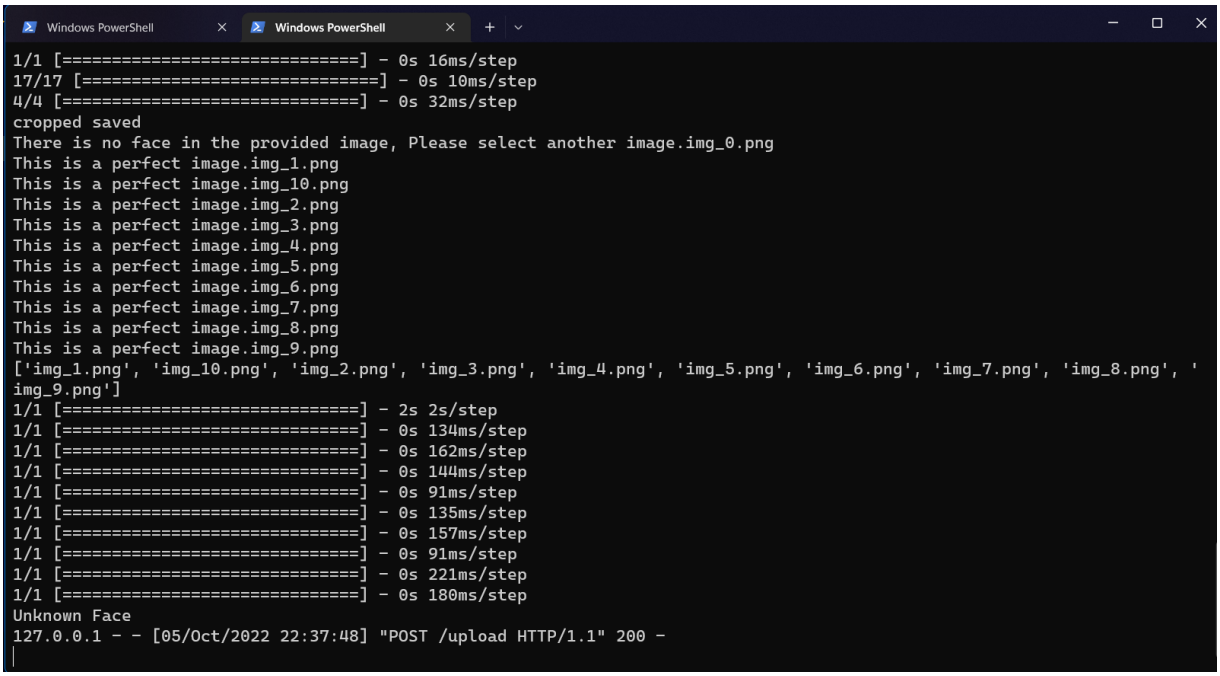


Fig. 8.6 Backend Server

8.7 Generated Attendance report:

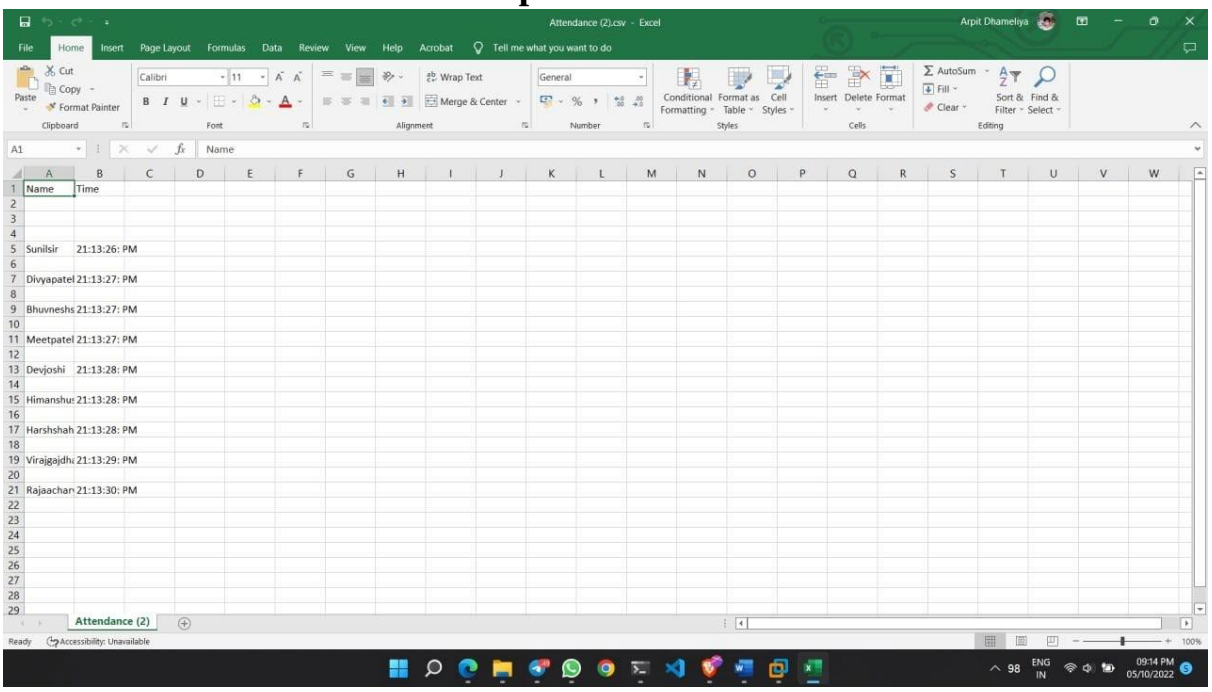
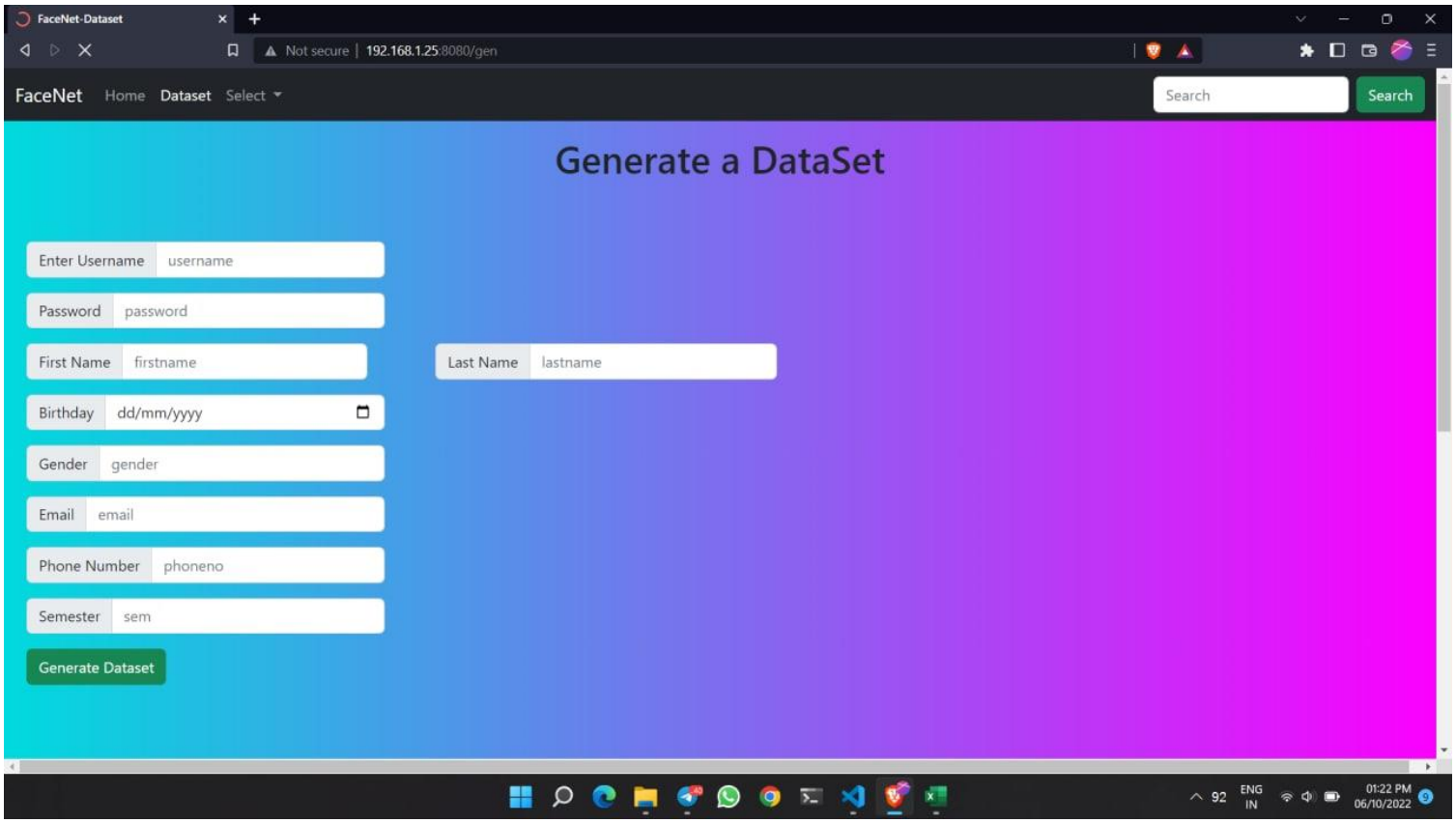


Fig. 8.7 Attendance Excel report

8.8 GUI For Generate Dataset:



8.9 Hardware System Photo:

Figure:1

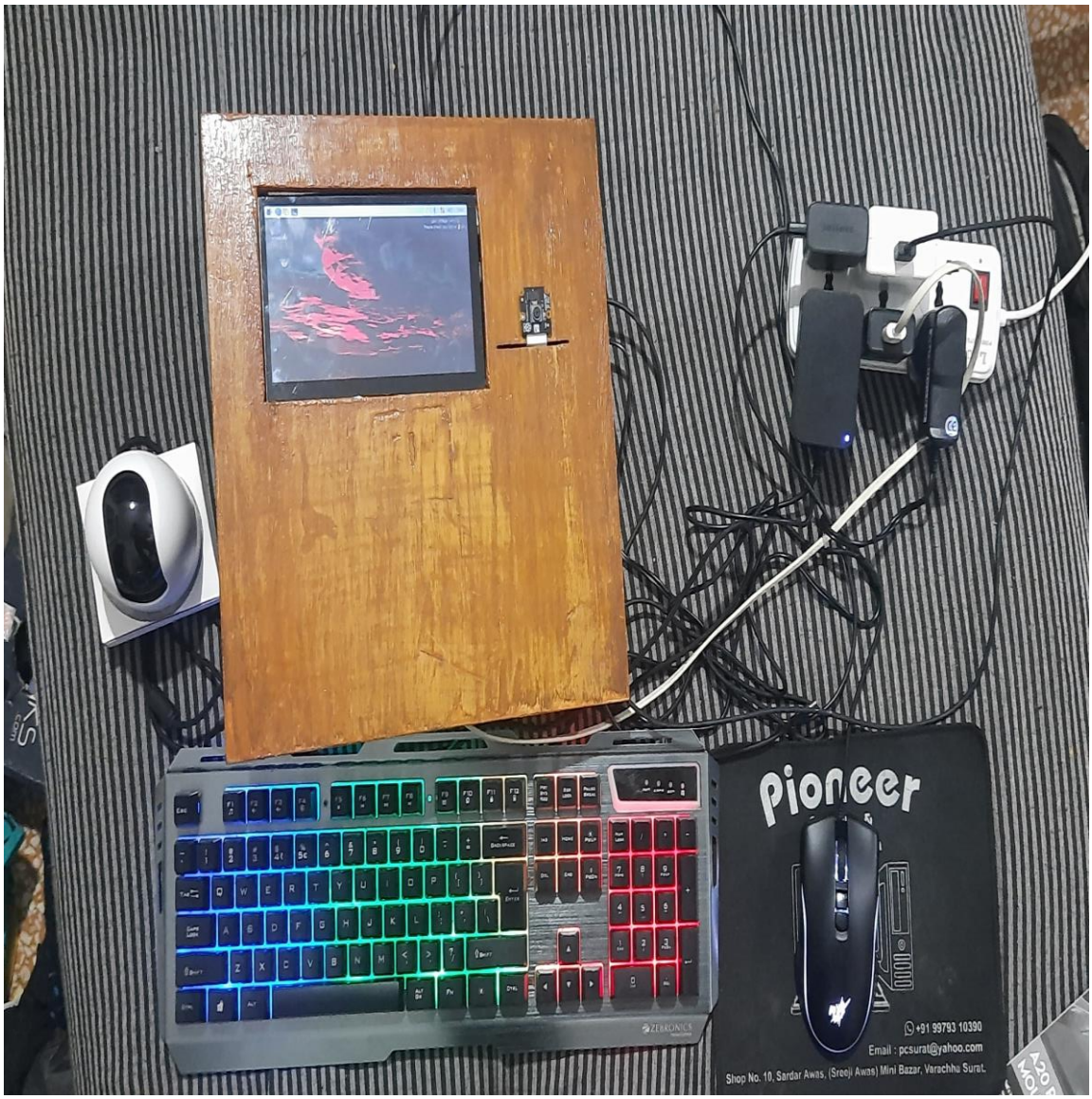

Pioneer
+91 99793 10390
Email : pcsurat@yahoo.com
Shop No. 10, Sardar Awas, (Sreeji Awas) Mini Bazar, Varachhu Surat.

Figure:2

CHAPTER 9
LIMITATIONS AND FUTURE ENHANCEMENTS

LIMITATIONS:

8.7.1 Currently our GUI Interface is not Good.
8.7.2 Our System is not Compatible to Run and Loading this FaceNet MTCNN model that's why we use Google Colab.
8.7.3 If faces are similar more than 50% then may probably Face mismatch occurs.
8.7.4 If face with mask and dark goggles then face not recognize.
8.7.5 Attendance report student can view and apply complain.

FUTURE ENHANCEMENTS:

8.7.6 Improve in GUI Interface.
8.7.7 Implement API using Google cloud / AWS cloud connect with our GUI interface.
8.7.8 We can Set some hardware and make a product like thing.
8.7.9 We can set raspberry pi as transmitter for sending cropping images Receive same answer/ attendance file on the same raspberry pi.
8.7.10 Also, we can make research paper on this FACENET model and Project.
8.7.11 Increase speed of embedding and compressing.
8.7.12 We can implement message sending system using GSM module to the patents.

CHAPTER 10
CONCLUSION AND DISCUSSION

10.1 CONCLUSION

- Software is easy to use but it requires user is pre-knowledge of system and refers the user manual.
- Software is efficient to recognize 30 to 40 faces show name and detail and generate .csv file.
- In Software there is no mismatch occurs all faces are detected and recognize very well.
- If there are not some students not embedded then for their faces Show Unknown.
- Software can easily embed new coming students.

10.2 DISCUSSION

10.2.1 Self-Analysis of Project Viabilities:

In our opinion, this project has served the goal that we set when we started. It provides a facility for automated attendance using face recognition where users can easily generate attendance report. Attendance report is easy to manage attendance for faculty. Adding new student is easy to faculty for new commers.

10.2.2 Problems encountered:

There were so many problems encountered during this project: -

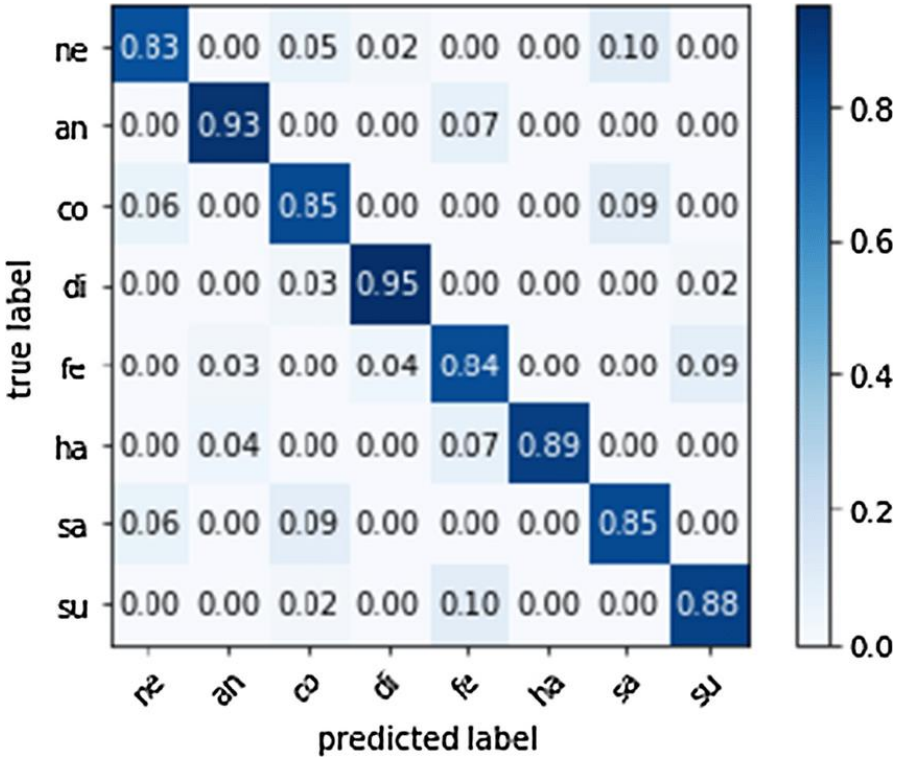
1. Problem to Detect faces with all coordinates and features.
2. Need to change some functionality fully like we have change whole 4 Model and done whole work again.
3. Haarcascade and LBPH algorithm – In this model many miss matches occurs.
4. MobileNet V2 – Work perfectly for 8-10 Students but there is no option of embedding.
5. VGG16 – this model takes more and more time in Training Dataset
6. Finally we got FACENET model.
7. Many instances where the errors weren't understandable. (Like Code Marsal error:)

10.3 Summary of Project Work:

We have completed our project work using software engineering and system analysis and design approach. We have done our work with planned scheduling pertaining to time constraints and result oriented progress in project development. We have been given an industrial exposure we can also use our project as working product after changing in this which is very beneficial for our field.

10.4 Accuracy (Confusion Matrix):

Confusion Matrix:



CHAPTER 11
BIBLIOGRAPGHY

11.1 FACENET MODEL

https://machinelearningmastery.com/how-to-develop-a-face-recognition-system-using-facenet-in-keras-and-an-svm-classifier/

11.2 FOR PRETRAINED FACENET MODEL

https://projectsflix.com/machine-learning/deepface-and-facenet-for-face-recognition/

11.3 STACK OVERFLOW

https://stackoverflow.com/

11.4 MEDIUM

https://medium.com/