

# REPORT FOR ADVANCED AI HOME SECURITY

As a project work for Course

## PYTHON PROGRAMMING (INT 213)

---

Name : Gibran Khan Tareen  
Registration Number : 12100173  
Program : B.Tech CSE (Lateral Entry)  
Semester : Third  
School : School of Computer Science  
and Engineering (CSE)

Name of the University : Lovely Professional University

Date of submission : 20th NOVEMBER 2021

---

Lovely Professional University Jalandhar, Punjab, India.



**L**OVELY  
**P**ROFESSIONAL  
**U**NIVERSITY

---

*Transforming Education Transforming India*

# **ADVANCED AI HOME SECURITY**

**20<sup>th</sup> NOVEMBER 2021**

## **Abstract**

Today, with the continuous development in world of science and technology, it is hard to estimate how far we will go and what else we will be able to accomplish. According to experts, the future of the world seems to be dominated by Artificial Intelligence in some way or the other.

With the advancements in technology, we also see that there is a constant surge in the heists, thefts and other crimes. Even after installing Face Recognition locks at home or places, still people become the victims of thefts and burglaries. Therefore it is important to address this problem. So I decided to make an Advanced AI Security System (which is mainly to be used on the Front Door Camera), which uses the concepts of AI with added security layers which will ultimately determine the person at the door and in case the person fails to verify himself after failing all his chances, the Local police and the owner of the house will be alerted immediately of the intruder who is trying to enter the house. (Please refer to Workflow Section for better understanding of this project).

## **Acknowledgement**

I had a great experience working on this project and got to learn a plenty of new skills through this project. First I would like to thank Almighty Allah who made it possible for me to do anything. After that I would like to dedicate this project to my Late Grandfather- Mohd Nayeem Khan, my Gradma, my Parents, my litte sister, and my maternal Grandparents because without their kind support and help I would not have been able to complete my project. I would also like to express my sole gratitude to my teacher and mentor (Prof. Sagar Pande Sir) who gave me this golden opportunity to do this project on the topic (Advanced AI Home Security). The project helped me to learn how to do proper Research and I learned about many new things while doing the project.

# Table of Contents

<b>1. ABSTRACT</b>	<b>2</b>
<b>2. INTRODUCTION</b> 2.1 CONTEXT 2.2 MOTIVATION 2.3 IDEA	<b>4</b>
<b>3. TEAM MEMBERS WITH ROLES</b> 3.1 TEAM LEADER 3.2 CONTRIBUTIONS	<b>5</b>
<b>4. LIBRARIES</b> 4.1 NAME OF THE LIBRARIES USED 4.2 WHY THEY ARE USED	<b>6</b>
<b>5. Proposed Modules</b>	<b>7</b>
<b>6. Screenshots</b>	<b>8</b>
<b>7. Concept of Face Recognition</b>	<b>13</b>
<b>8. Workflow and Test Cases</b>	<b>20</b>
<b>9. References</b>	<b>23</b>

# Introduction

## 1.1 Context

This project has been done as part of my course for the CSE at Lovely Professional University . Supervised by Sagar Pande, I have three months to fulfill the requirements in order to succeed the module.

## 1.2 Motivations

Since my very start, I have been extremely interested in everything which is related to Artificial Intelligence. This project was a great occasion to give me the time to learn and confirm my interest for this field. The fact that I can use AI in the field of Security as well and using by additional AI concepts helped me more to enhance my interest in the field of Artificial Intelligence. That's why I decided to have my project around Artificial Intelligence

## 1.3 Idea:-

Well if I remember correctly, it was exactly 3 years ago in 2018 when a friend of mine in Canada became a victim of theft at his home. Thieves entered through his front door of home even though he had lock system at his house. At last it was also found that they had destroyed the CCTV footage for it and even damaged the CCTV. This was the moment at which the idea struck my mind that I should develop such a realtime system that completely manages "Known Faces" and "Unknown Faces" with help of using the concepts of Artificial Intelligence. Since this system will be always present at the front door, so even if someone tries to break-in, the Advanced AI system, immediately issue an alert and double lock all the room doors also along with this, a photo of intruder will be sent to Local police response team and Owner. So thus I chose to take this Advanced AI Home Security System as my approach. My goal ultimately was to develop a system that has additional security layers even after Face Recognition and also an advanced AI system that will handle the "Un-known faces" and understand their purpose of visit whether it is authorized by the owner of the house or not, also advanced alerting system to alert the dedicated response team immediately in the case of the confirmed suspicion.

# **Team Members**

**THIS PROJECT IS MADE BY ONLY ONE STUDENT:-**

**Gibran Khan Tareen (Registration no. 12100173):**

## **Contributions:-**

1. Coding
2. Thought Process
3. GUI Programming
4. Artificial Intelligence  
(Concept Framing)
5. Testing the whole Project
6. Making the Report

# Libraries

Now, for my Home Security Project to work, it is necessary to install some specific Python Modules and Libraries. Also apart from these modules, some specific program files are required to run a complete functionality. My Project is fully tested and working on Python 3.9.1

- 1) **pyttsx3**:- a text-to-speech conversion library in Python
- 2) **SpeechRecognition**: Library for performing speech recognition in Python
- 3) **Datetime**: a Python module which supplies classes for manipulating dates and times.
- 4) **OpenCV (cv2)**: OpenCV-Python is the Python API of OpenCV which is used for Image processing and Face-recognition functions.
- 5) **Smtplib**: a module defines an SMTP client session object that can be used to send mail to any Internet machine with an SMTP
- 6) **Sys**: a module in Python which provides various functions and variables that are used to manipulate different parts of the Python runtime environment
- 7) **Pyautogui**: is a Python automation library used to click, drag, scroll, move, etc. It can be used to click at an exact position. It provides cross-platform support for managing mouse and keyboard operations through code to enable automation of task
- 8) **Numpy**: a highly optimized library for numerical operations
- 9) **Playsound**: this module is a cross platform module that can play audio files.
- 10) **Plyer**: it is a platform-independent api to use features commonly found on various platforms, notably mobile ones, in Python. It can be used to access the features of your hardware / platforms.
- 11) **SpeechRecognition**: Library for performing speech recognition in Python
- 12) **FaceRecognition**: Library to Recognize and manipulate faces from Python or from the command line
- 13) **Time**: a predefined Python module which allows us to handle various operations regarding time

# Proposed Modules:

## 1) Encoding Face images as Dataset:-

1. Insert Face images manually in the 'Dataset Ki Images' Folder
2. Those Images will be converted into Dataset then (by the Face\_Recognition module itself)

## 2) Face Recognition + Security Verification

1. Match the face in the webcam with the encoded dataset images
2. If face matches, then to verify the person fully an additional password required (which will already be set by the owner himself)

## 3) Handling Intruders:

1. Ask Unknown person (his purpose of visit)
2. Also listen for "Recovery Phrase" which is set by owner to allow guests into the house + His time of entry will also be noted

## 4) Alert System:

1. Three chances will be given to the "Known Face" Person to verify the entry password.
2. Incase that person (who's face matches datasets) fails to verify the "Entry Password" even after 3 attempts, all the house locks will be doubled and an immediate alert will be sent to the dedicated Response team (police etc)
3. Incase of Unknown person (who's face is not matched with the dataset faces), he will be asked for his name and purpose of visit. If owner has sent the person, he must give him a "Recovery Phrase", then my AI System will also be listening for that Phrase and if he confirms it then Security system will allow guest into the house



**Diagram 1:**  
Basic Layout of the Project Modules

# Screenshots:

1.Start of Program (Encoding to Datasets completed by Python module, of the given Face Images):-

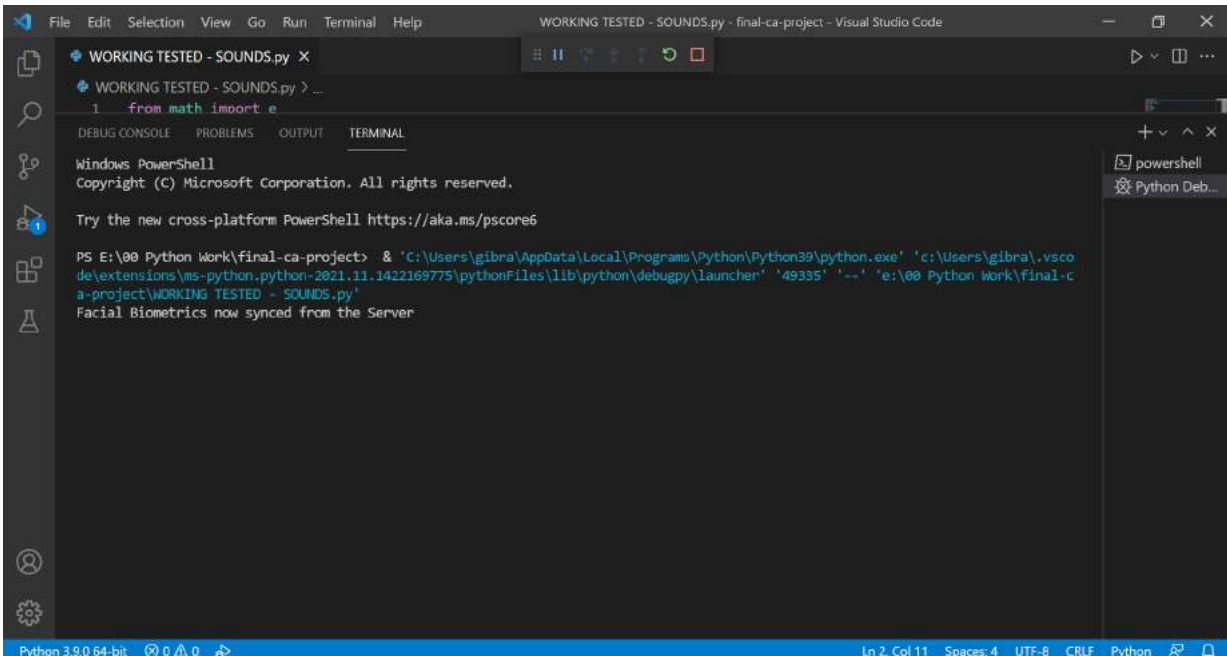


Image 1: Encoding to Datasets completed by Python module

2.Webcam opens for Verification"ej gemi'hqt'æeg'lp'y gdeco with dataset images):-

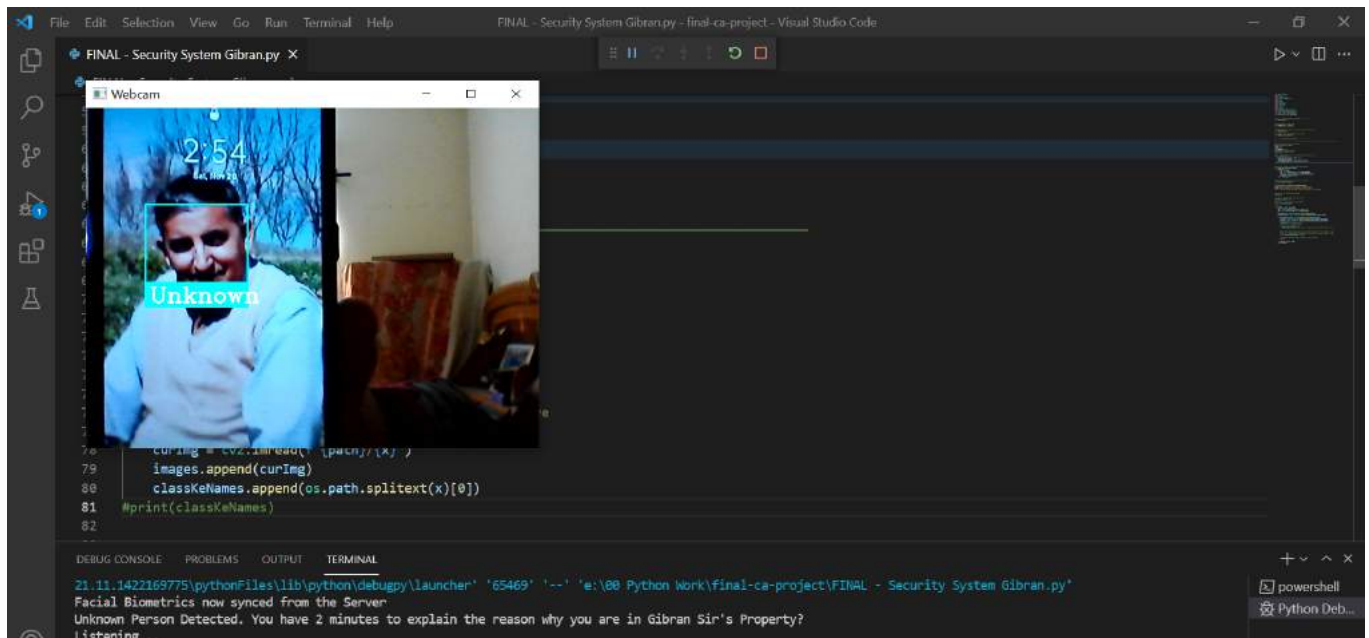
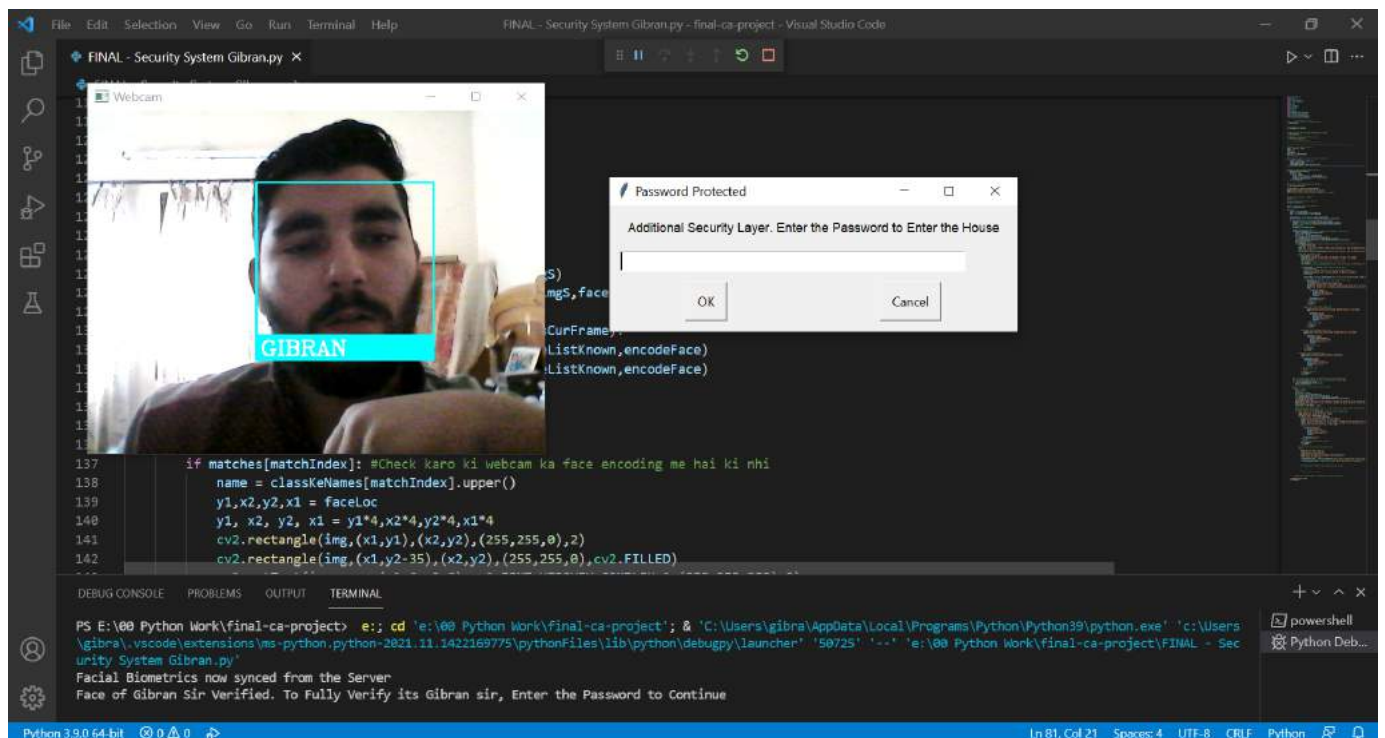


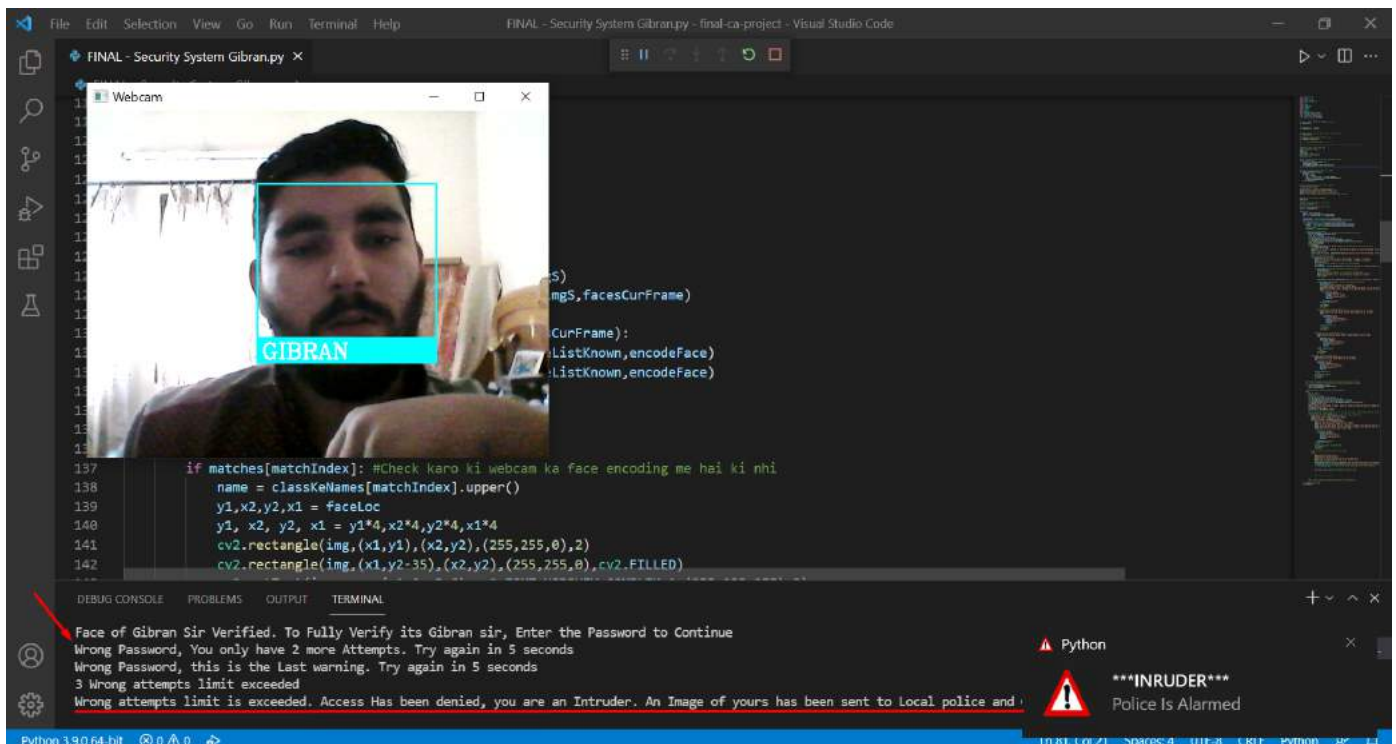
Image 2: Webcam opens for Face Verification





**Image 3:** Incase of Matched Face, additional password is asked

### 3. Alarm Raised Incase of Confirmed Intruder (See Workflow to understand better):



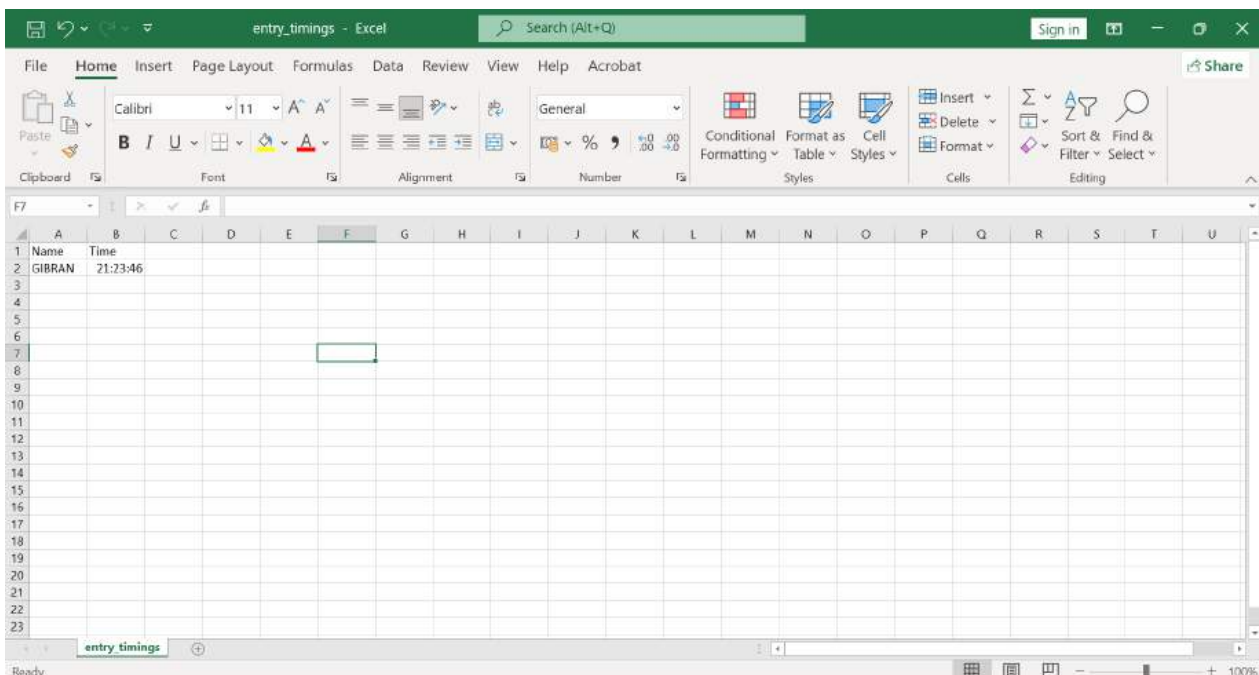
**Image 4:** Alarm Raised Incase of Confirmed Intruder

4.Images at test case Completion are Saved automatically in the Project Folder (having filename as respective test case name):-



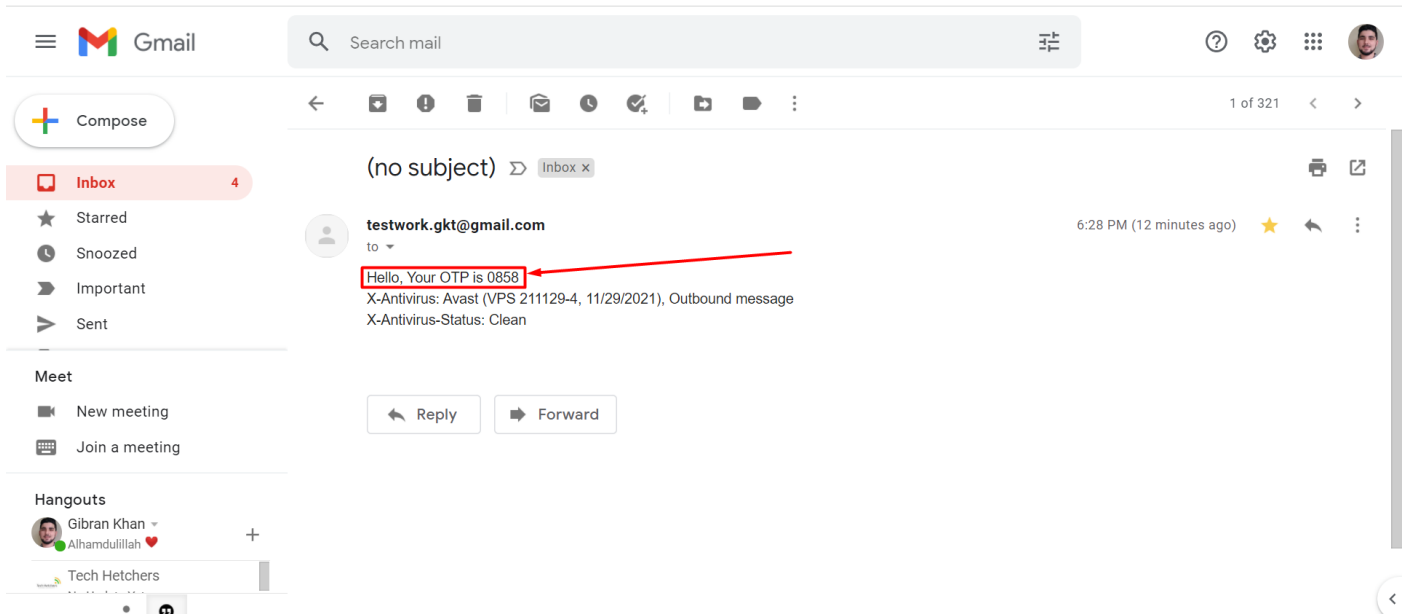
**Image 5:** Image at each test case is saved in the system

5.A separate record of successful entries is maintained by the AI System in the Project Folder, having Name of Detected face and time when Door was opened by the Security system:-

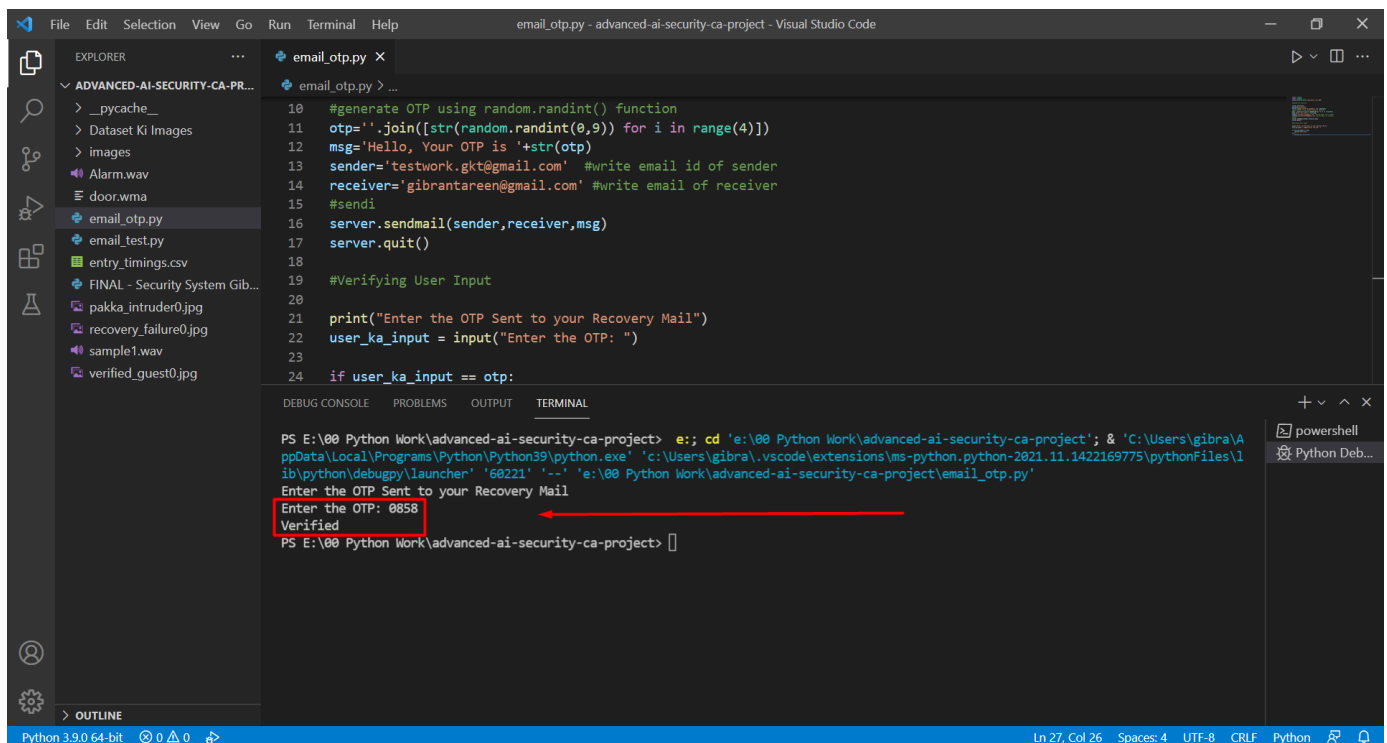


**Image 6:** A separate record of successful entries maintained by AI System

6. In case of Recovery Mode attempt, the user will be sent an OTP via Email (which will be registered in Owner's records) and the OTP will be verified by the system to check whether the user is authorized or not:



**Image 7:** A Login OTP sent to the user's registered recovery email



**Image 8:** System ask to enter OTP and verifies it

## Artificial Intelligence:-

The field of computer science and engineering that attempts to simulate the features of human intelligence or capabilities through the help of machines is called Artificial Intelligence. My project is solely based on the concepts of Artificial Intelligence.

## Flow Diagram of the Approach:



**Diagram 2:** Approach used to frame cases and solutions to problems

# Concept of Face Recognition in Python:-

With help of FaceRecognition module in python we can manipulate and work upon faces. The world's simplest face recognition library built using dlib's state-of-the-art face recognition, is actually built with deep learning. The model has an accuracy of 99.38% on the Labeled Faces in the Wild benchmark.

This also provides a simple face\_recognition command line tool that lets you do face recognition on a folder of images from the command line!

Following are the basic functions we can do in Python:

## 1) Find faces in pictures

Find all the faces that appear in a picture:



Input



Output

```
import face_recognition
image = face_recognition.load_image_file("your_file.jpg")
face_locations = face_recognition.face_locations(image)
```

**Image 9: Python Code to find faces in pictures**

## 2) Find and manipulate facial features in pictures:

**Image 9: Python Code to Find and manipulate facial features**

```
import face_recognition
image = face_recognition.load_image_file("your_file.jpg")
face_landmarks_list = face_recognition.face_landmarks(image)
```

Find and manipulate facial features in pictures

Get the locations and outlines of each person's eyes, nose, mouth and chin.



Input



Output



### 3) Find and identify faces in pictures:

#### Identify faces in pictures

Recognize who appears in each photo.



Input



Picture contains  
"Joe Biden"

Output

#### Image 10:

Python Code to  
Find and identify  
faces in pictures

```
import face_recognition
known_image = face_recognition.load_image_file("biden.jpg")
unknown_image = face_recognition.load_image_file("unknown.jpg")

biden_encoding = face_recognition.face_encodings(known_image)[0]
unknown_encoding = face_recognition.face_encodings(unknown_image)[0]

results = face_recognition.compare_faces([biden_encoding], unknown_encoding)
```

### 4) Find and identify faces in realtime camera feed:

We can use this module with other powerful libraries and modules like OPEN-CV, to detect faces in realtime (The thing which we have done in our Project)



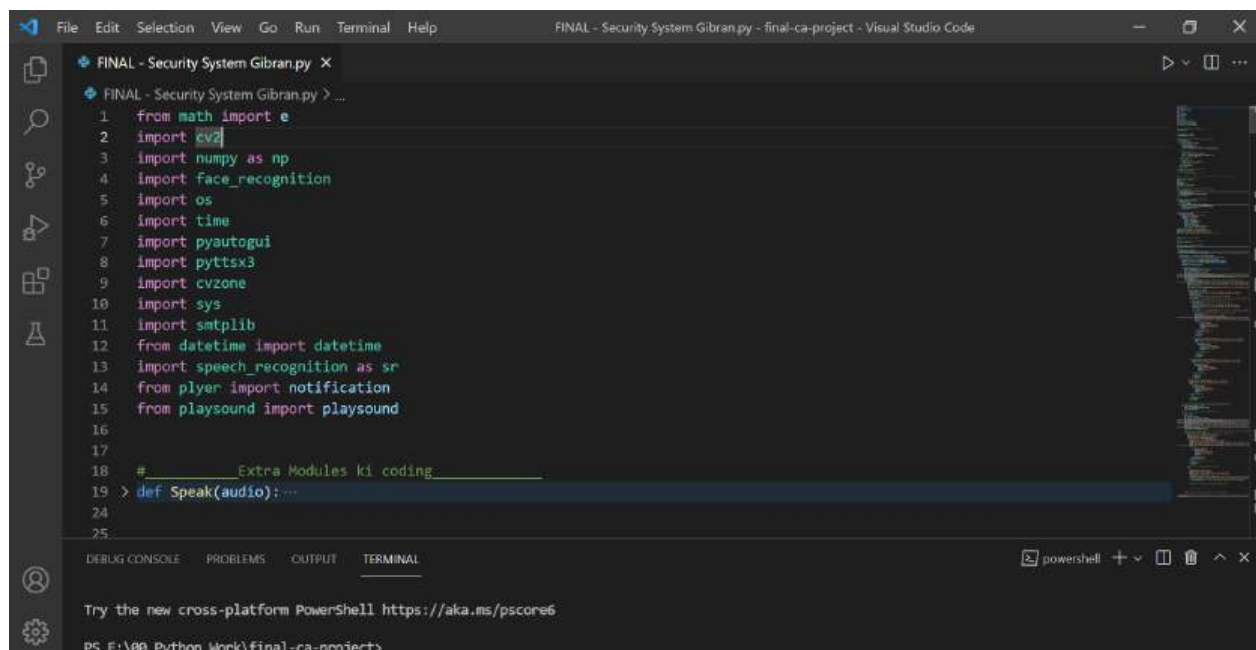
#### Image 11:

Python Code to  
Find and identify  
faces in realtime  
scenario (web cam)

Images Source: [PyPi.org/project/face\\_recognition](https://pypi.org/project/face_recognition)

# Code Design Implementation:-

## Modules Used:

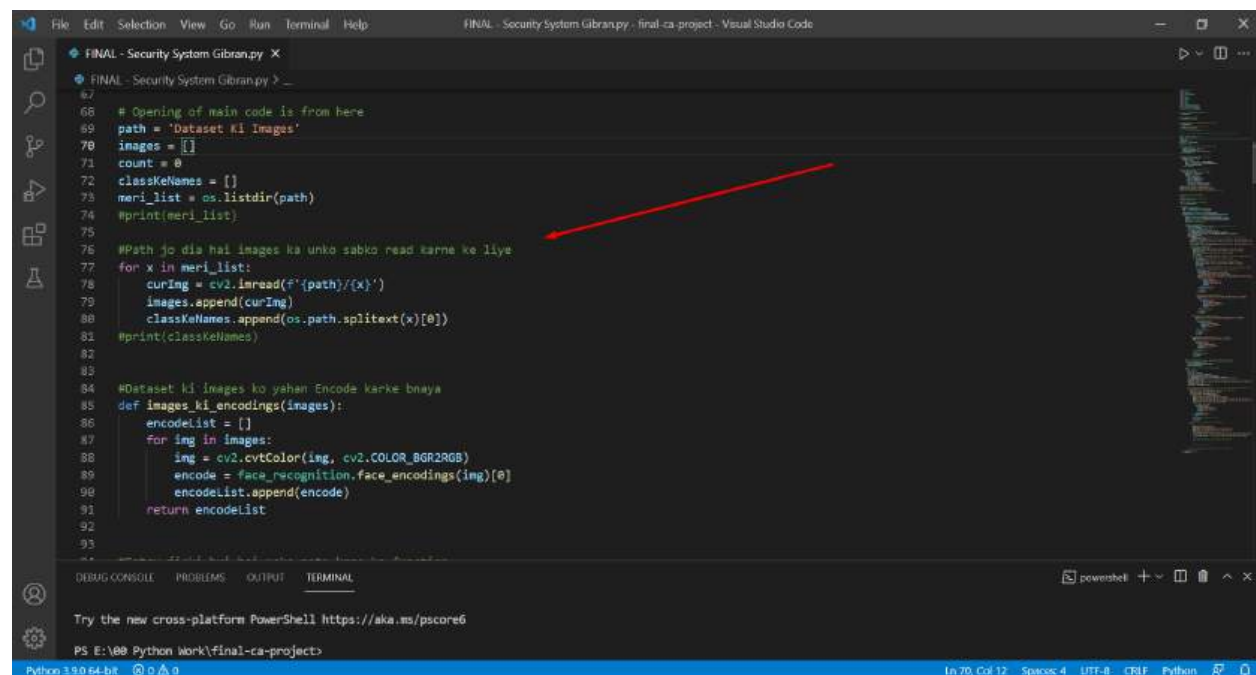


The screenshot shows a Visual Studio Code editor window with a Python file named 'FINAL - Security System Gibran.py'. The code lists various Python modules used in the project, including 'math', 'cv2', 'numpy', 'face\_recognition', 'os', 'time', 'pyautogui', 'pyttsx3', 'cvzone', 'sys', 'smtplib', 'datetime', 'speech\_recognition', 'plyer', and 'playsound'. A comment indicates that these are 'Extra Modules ki coding'. The terminal at the bottom shows the PowerShell prompt 'PS E:\00 Python Work\final-ca-project>'.

```
1 from math import e
2 import cv2
3 import numpy as np
4 import face_recognition
5 import os
6 import time
7 import pyautogui
8 import pyttsx3
9 import cvzone
10 import sys
11 import smtplib
12 from datetime import datetime
13 import speech_recognition as sr
14 from plyer import notification
15 from playsound import playsound
16
17
18 #_____Extra Modules ki coding_____
19 > def Speak(audio):...
```

Image 12: Modules of Python used in my project

## Changing given Images to Encoding Datasets:

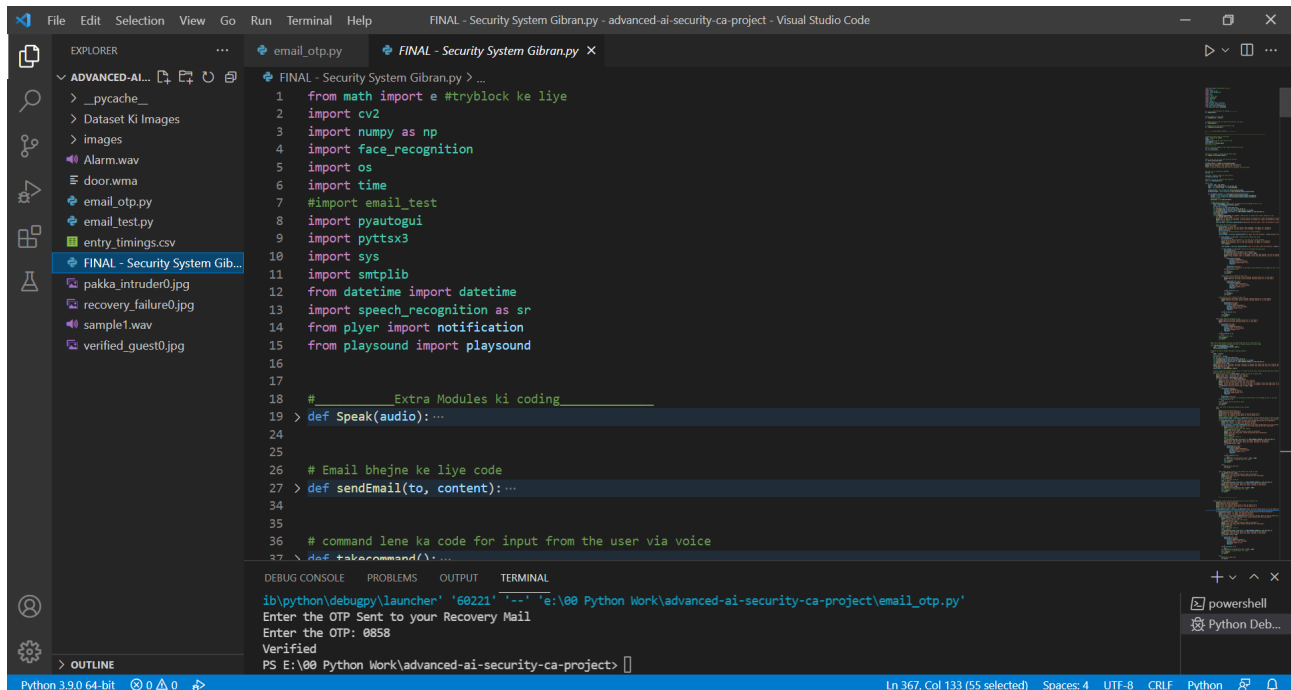


The screenshot shows a Visual Studio Code editor window with a Python file named 'FINAL - Security System Gibran.py'. The code defines a function 'def images\_to\_encoding\_datasets(images):' which iterates through a list of image paths, reads each image, and encodes it using 'face\_recognition.face\_encodings'. A red arrow points to the line 'cur\_img = cv2.imread(f'{path}/{x}'). The terminal at the bottom shows the PowerShell prompt 'PS E:\00 Python Work\final-ca-project>'.

```
67
68 # Opening of main code is from here
69 path = 'Dataset Ki Images'
70 images = []
71 count = 0
72 class_names = []
73 meri_list = os.listdir(path)
74 #print(meri_list)
75
76 #Path jo dia hai images ka unko sabko read karne ke liye
77 for x in meri_list:
78     cur_img = cv2.imread(f'{path}/{x}')
79     images.append(cur_img)
80     class_names.append(os.path.splitext(x)[0])
81 #print(class_names)
82
83
84 #Dataset ki images ko yahan Encode karke bnaya
85 def images_to_encoding_datasets(images):
86     encodelist = []
87     for img in images:
88         img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
89         encode = face_recognition.face_encodings(img)[0]
90         encodelist.append(encode)
91     return encodelist
92
93
```

Image 13: Code for changing image to datasets

## Rest Code:

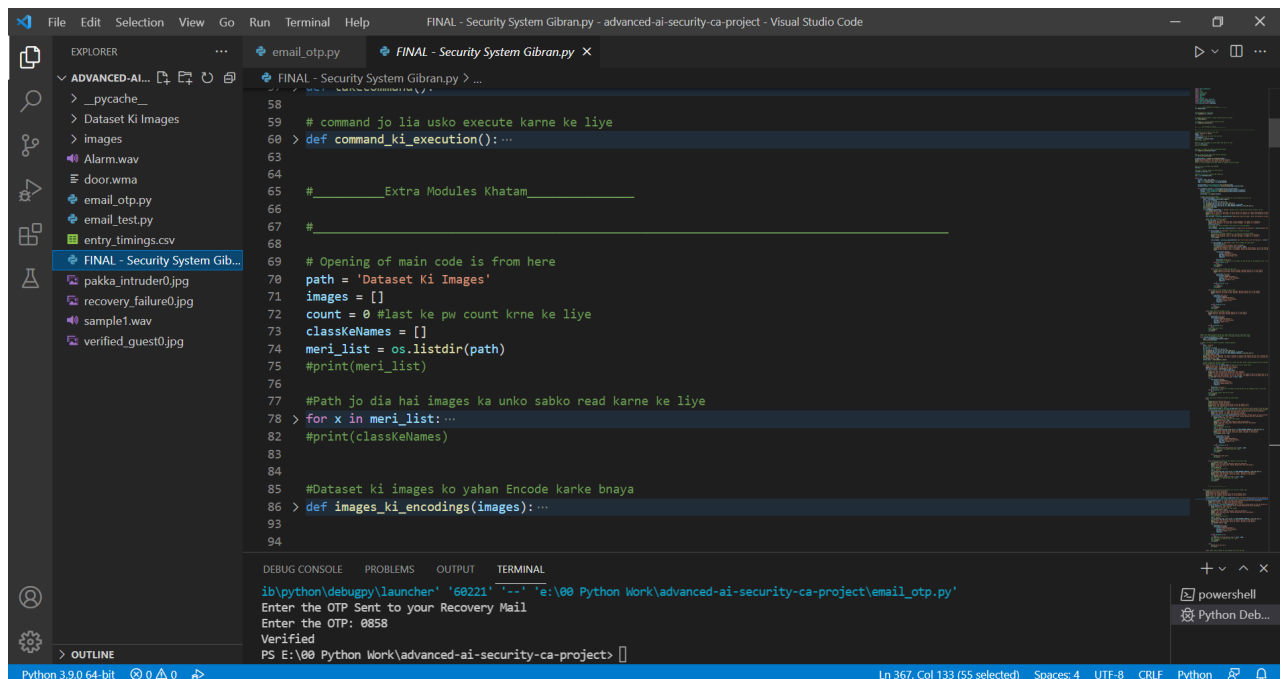


```
1 from math import e #tryblock ke liye
2 import cv2
3 import numpy as np
4 import face_recognition
5 import os
6 import time
7 #import email_test
8 import pyautogui
9 import pyttsx3
10 import sys
11 import smtplib
12 from datetime import datetime
13 import speech_recognition as sr
14 from player import notification
15 from playsound import playsound
16
17
18 # _____Extra Modules ki coding_____
19 > def Speak(audio): ...
20
21
22
23
24
25
26 # Email bhejne ke liye code
27 > def sendEmail(to, content): ...
28
29
30
31
32
33
34
35
36 # command lene ka code for input from the user via voice
37 > def takeCommand(): ...
```

DEBUG CONSOLE PROBLEMS OUTPUT TERMINAL

```
ib\python\debugpy\launcher '60221' '--' 'e:\00 Python Work\advanced-ai-security-ca-project\email_otp.py'
Enter the OTP Sent to your Recovery Mail
Enter the OTP: 0858
Verified
PS E:\00 Python Work\advanced-ai-security-ca-project>
```

Image 14: A Screenshot from my project code



```
58
59 # command jo lia usko execute karne ke liye
60 > def command_ki_execution(): ...
61
62
63
64
65 # _____Extra Modules Khatam_____
66
67
68
69
70 # Opening of main code is from here
71 path = 'Dataset Ki Images'
72 images = []
73 count = 0 #last ke pw count krne ke liye
74 classKeNames = []
75 meri_list = os.listdir(path)
76 #print(meri_list)
77
78 #Path jo dia hai images ka unko sabko read karne ke liye
79 > for x in meri_list: ...
80
81 #print(classKeNames)
82
83
84
85 #Dataset ki images ko yahan Encode karke bnaya
86 > def images_ki_encodings(images): ...
87
88
89
90
91
92
93
94
```

DEBUG CONSOLE PROBLEMS OUTPUT TERMINAL

```
ib\python\debugpy\launcher '60221' '--' 'e:\00 Python Work\advanced-ai-security-ca-project\email_otp.py'
Enter the OTP Sent to your Recovery Mail
Enter the OTP: 0858
Verified
PS E:\00 Python Work\advanced-ai-security-ca-project>
```

Image 15: A Screenshot from my project code



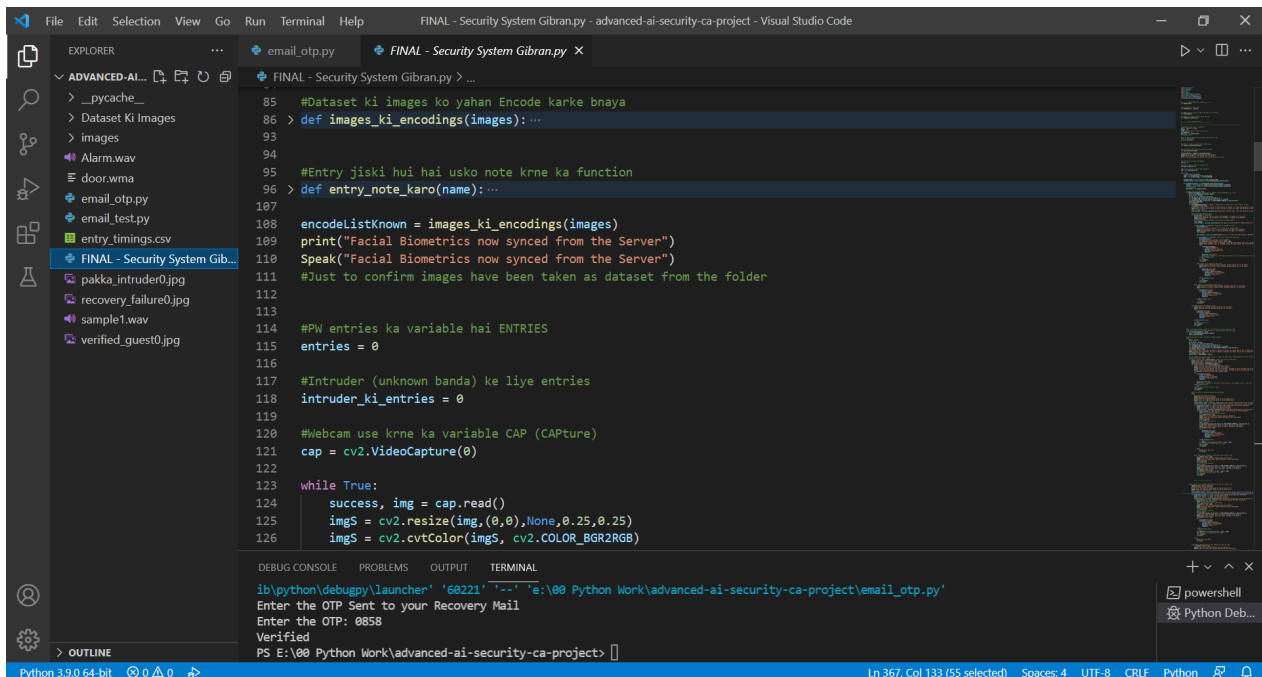


Image 16: A Screenshot from my project code

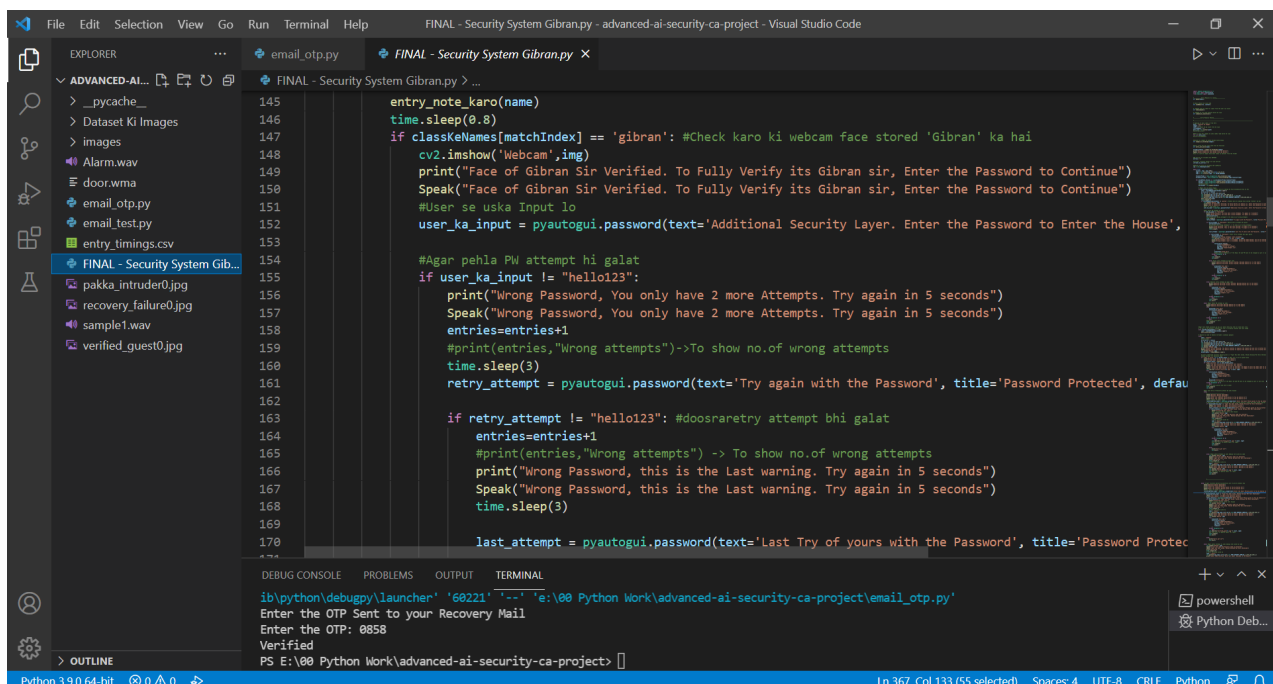


Image 17: A Screenshot from my project code

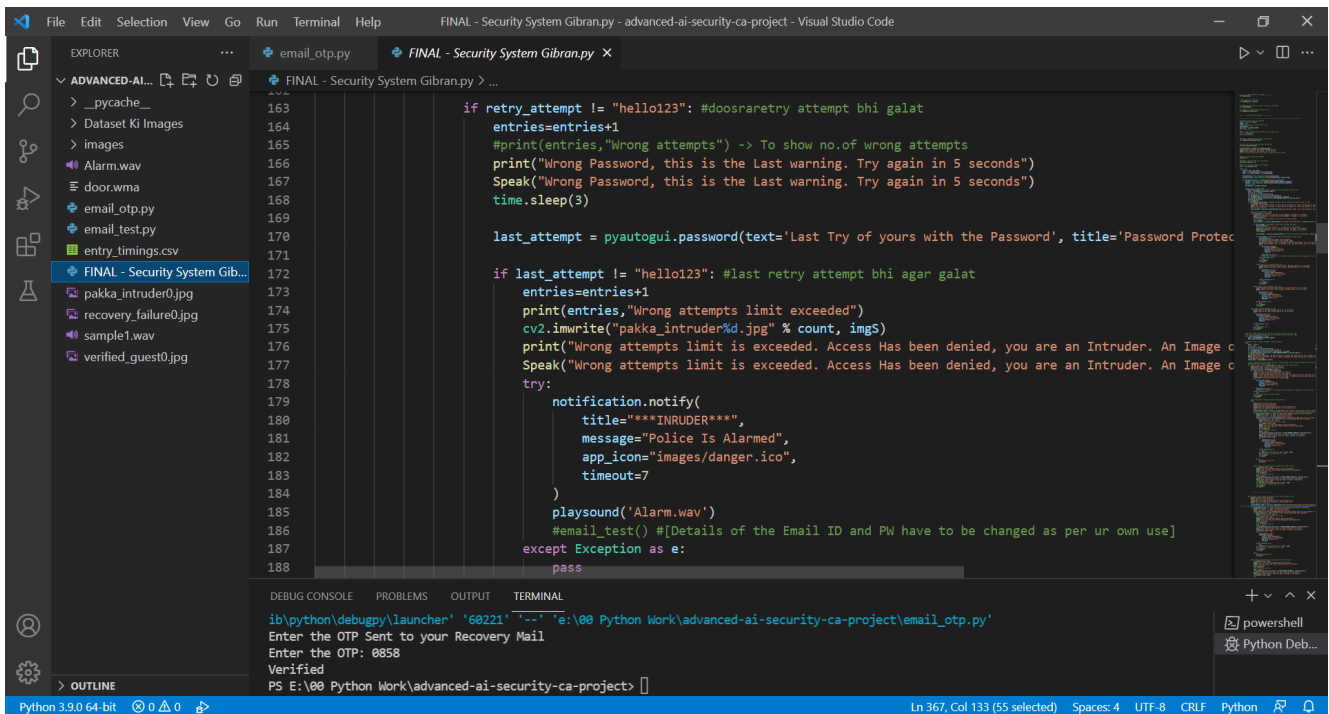


Image 18: A Screenshot from my project code

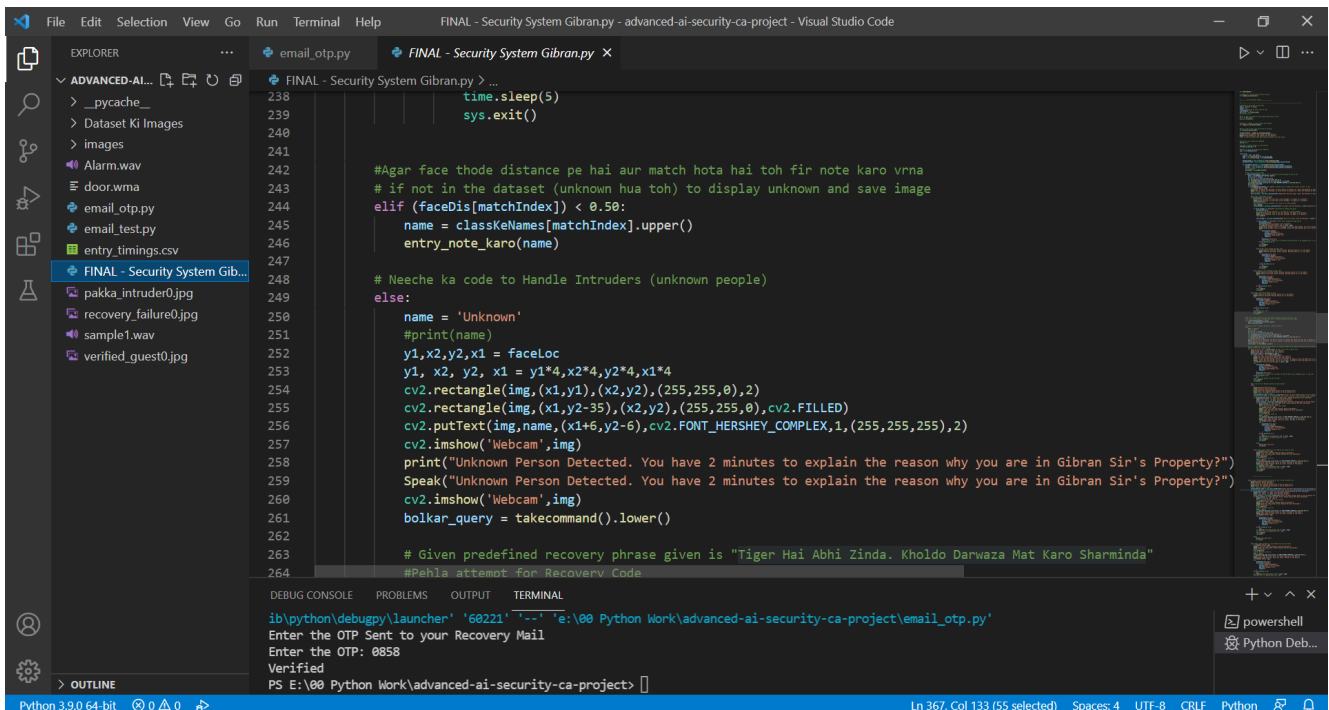
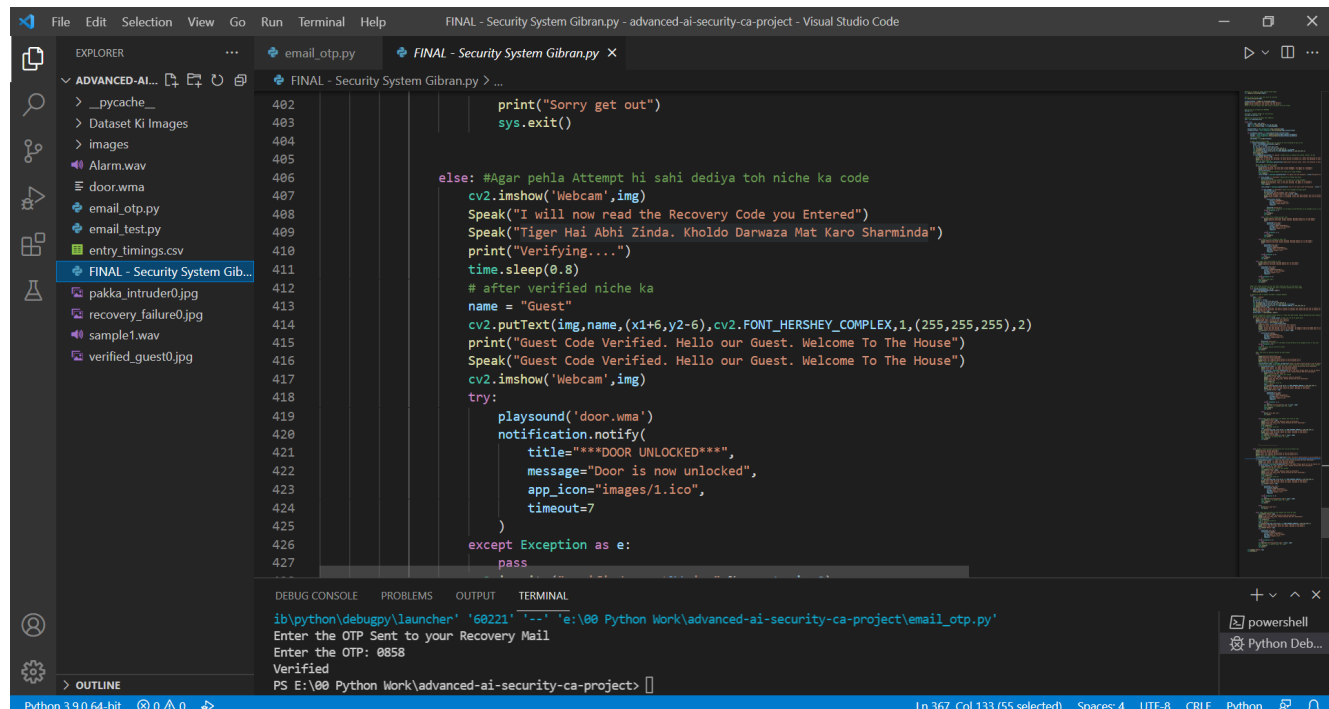
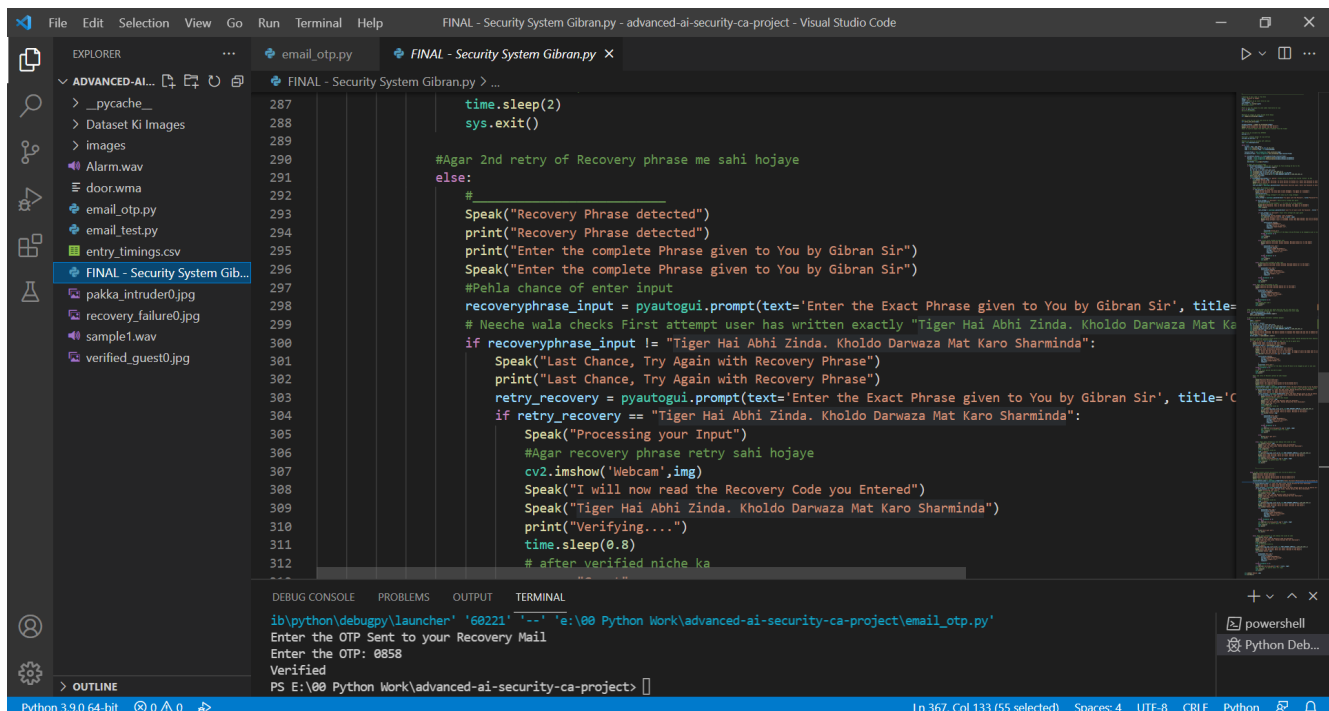


Image 19: A Screenshot from my project code



# Workflow of my Security System:

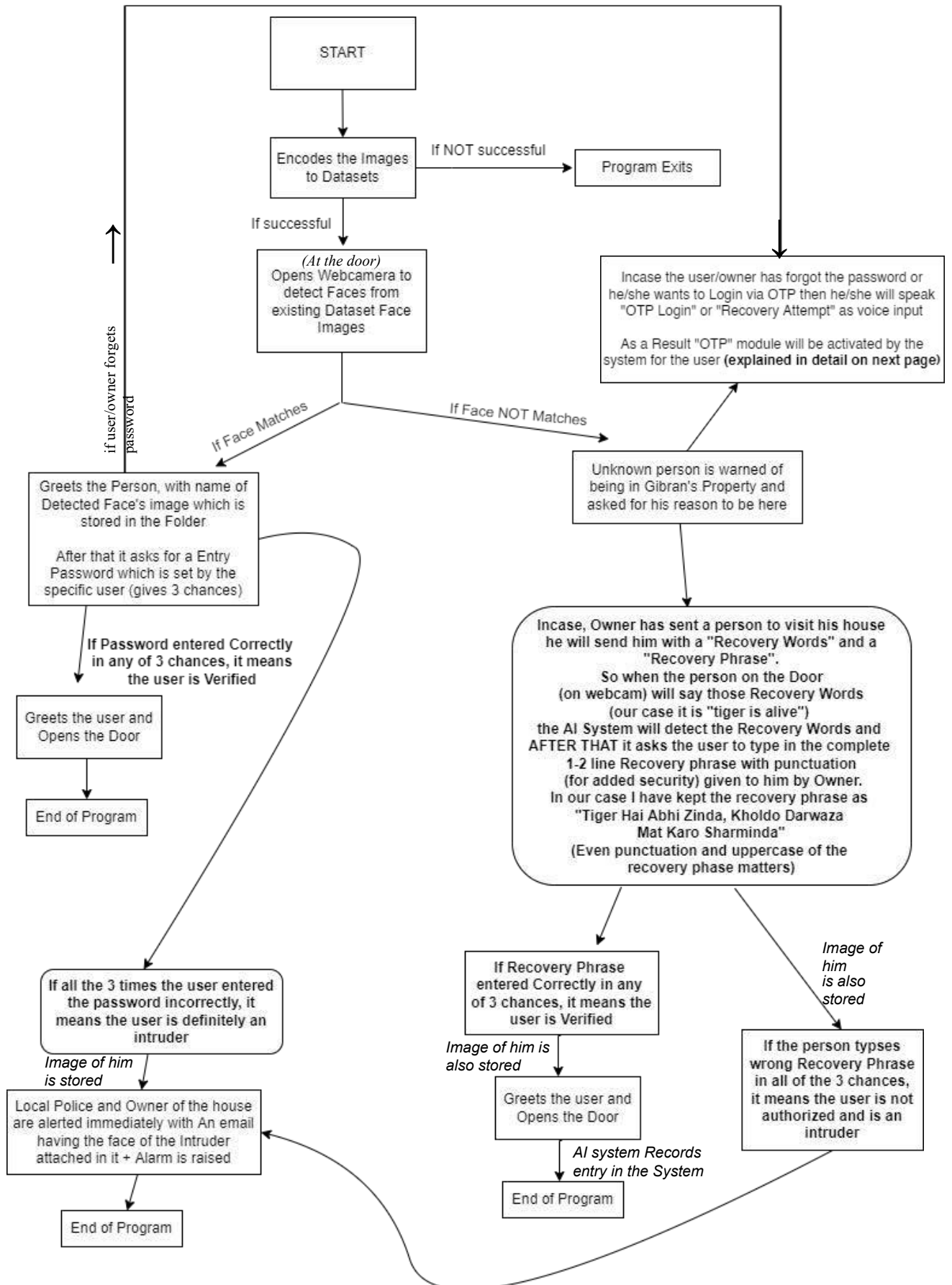
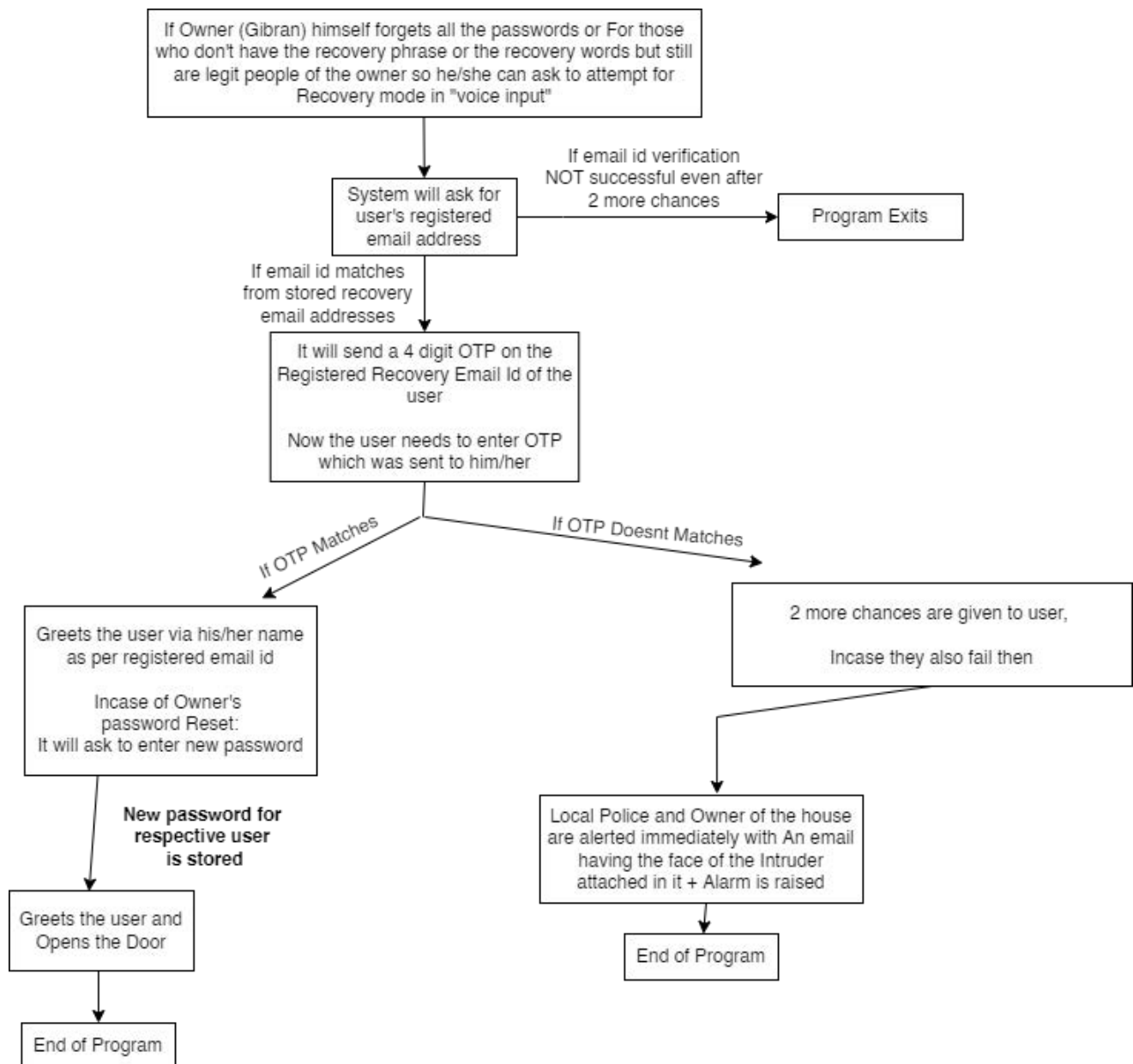


Diagram 3: Workflow of Complete Project

# Workflow of OTP Entry + Recovery Module:



**Diagram 4:** Workflow of OTP Login + Recovery

## Test Cases:-

A test case is a set of conditions or variables under which a tester will determine whether a system under test satisfies requirements or works correctly. The process of developing test cases can also help find problems in the requirements or design of an application.

### All the Test cases:

s.no	Testing name	Description	Expected result	Actual result	remarks	output
1.	Images Encoding to Dataset	Preuploading by the user	Displays Confirmation	Displays Confirmation	Images have been encoded correctly	passed
2.	Program Initiating	Running complete program	Program run	Successfully running	Install all required modules and libraries	passed
3.	Known person handling	initiated when detects known person in webcam	Gives 3 chances to enter correct entry password	Running	NO error or System issue	failed
4.	Intruder handling	initiated when detects intruder in webcam	Gives 3 chances to intruder to prove himself	passed	NO error or System issue	Passed
5.	Email Alert	Sending of alert via Email to the owner	Alert must be sent successfully	Alert sent successfully	Email ID and password for the account must be changed by user to work it	passed

## Conclusion:-

*It is my hope that this document will be of huge help to every AI Enthusiast. I have used a different AI approach while coding my project which has proved to be very beneficial for me and it has been designed in such a way that it is easy to understand if you want to grasp knowledge of Artificial Intelligence. After the formulation and continuous evaluation of all of my test cases I have reached the maximum accuracy of 95% Home Security Analysis but I will work even more to increase this accuracy to 100% Inshallah.*

## References:

To conduct this project I referred to the following books:

- 1) *How to think like a computer scientist: Learn with Python*, Allen B. Downey
- 2) *Project-Based Introduction to Programming*, by Eric Matthes
- 3) *Mastering OpenCV 4 with Python*: Alberto Fernández Villán

### 1.1 Pypi.org:-

I have used this site for getting the knowledge of every python module and also the command line codes to install the modules:

<https://pypi.org/project/face-recognition/>

### 1.2 Stackoverflow:-

I have used this site for solving different problems which occurred during this project. <https://stackoverflow.com/questions/68826091/the-specified-device-is-not-open-or-is-not-recognized-by-mci>

### 1.3 Youtube:-

I have used this site for solving installation problems which occurred due to Modules or Libraries' maintenance errors (old modules not working in new version of Python)  
[https://www.youtube.com/watch?v=pHrgi8QLcKk&ab\\_channel=ImportPyeidetic](https://www.youtube.com/watch?v=pHrgi8QLcKk&ab_channel=ImportPyeidetic)

# Appendix:

## Diagrams:

- 1) Basic layout of Proposed Modules (Page 7)
- 2) Approach used to frame cases and solutions to problems (Page 12)
- 3) Workflow of Complete Project (Page 20)
- 4) Workflow of OTP Entry + Recovery Module (Page 21)

## Images:

- 1) Encoding to Datasets completed by Python module (Page 8)
- 2) Webcam opens for Face Verification (Page 8)
- 3) Incase of Matched Face, additional password is asked (Page 9)
- 4) Alarm Raised Incase of Confirmed Intruder (Page 9)
- 5) Image at each test case is saved in the system (Page 10)
- 6) A separate record of successful entries maintained by AI System (Page 10)
- 7) A Login OTP sent to the user's registered recovery email (Page 11)
- 8) System ask to enter OTP and verifies it (Page 11)
- 9) Python Face Recognition Concept (Page 13,14)
- 10) Modules of Python used in my project (Page 15)