Smart Security For Home Using IoT

# By

### Batch number---> 23

### 1)G.Raghava Ranjith

### 2)G.Shiva Sai

### 3)K.Vijay Krishna

### 4)K.Pujith Chowdary

Contents

#### 1. Introduction

1.1 overview

1.2 Purpose

#### 2.Literature Survey

2.1 Existing problem

2.2 Proposed solution

#### 3.Theoritical Analysis

3.1 Block diagram

3.2 Software designing/Hardware

#### 4.Experimental Investigations

#### 5.Flow chart

#### 6.Results

#### 7.Advantages&Disadvantages

#### 8.Applications

#### 9.Conclusion

#### 10.Future scope

#### 11.Biblography

#### Appendix

#### A.source code

# 1 introduction

Today, Internet application development demand is very high. So **IoT** is a major technology by which we can produce various useful internet applications. Basically, **IoT** is a network in which all physical objects are connected to the internet through network devices or routers and exchange data

##### 1.1 overview

The Internet of Things (**IoT**) is helping create safer cities, homes, **and** businesses by enabling both private **and** public organizations to securely **and** remotely monitor facilities **and** public spaces **in** real-time **with smart security and** surveillance solutions.

##### 1.2 purpose

An important factor to consider when we talk about device is Security. security is a very important feature of a device and maybe the most crucial one. Home security made a drastic changes in the past few decades and continue to advance much more in the coming years. Previously home security systems meant having an alarm that would go off when somebody would break in but a smart secure home can do much more than that. Therefore the main objective of our work is to design a system which can alert the owner and others of an intruder break-in by sending a notification to their smart phones

## 2 literarure survey

##### 2.1 Existing problem

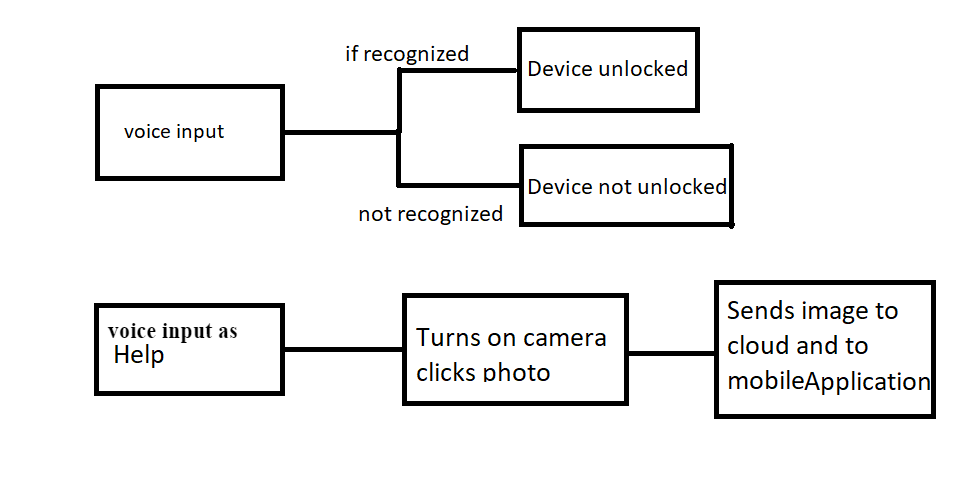
Due to existing project we can unlock our devices using password.if any other person knows our password they can access to our device.

##### 2.2 proposed solution

With this voice recognization we can unlock our device through particular voice commands so that another person cannot unlock our device. if any person tries to unlock our device it captures the image of that person and sends it to cloud

### 3.Theoritical analysis

##### 3.1 Block Diagram



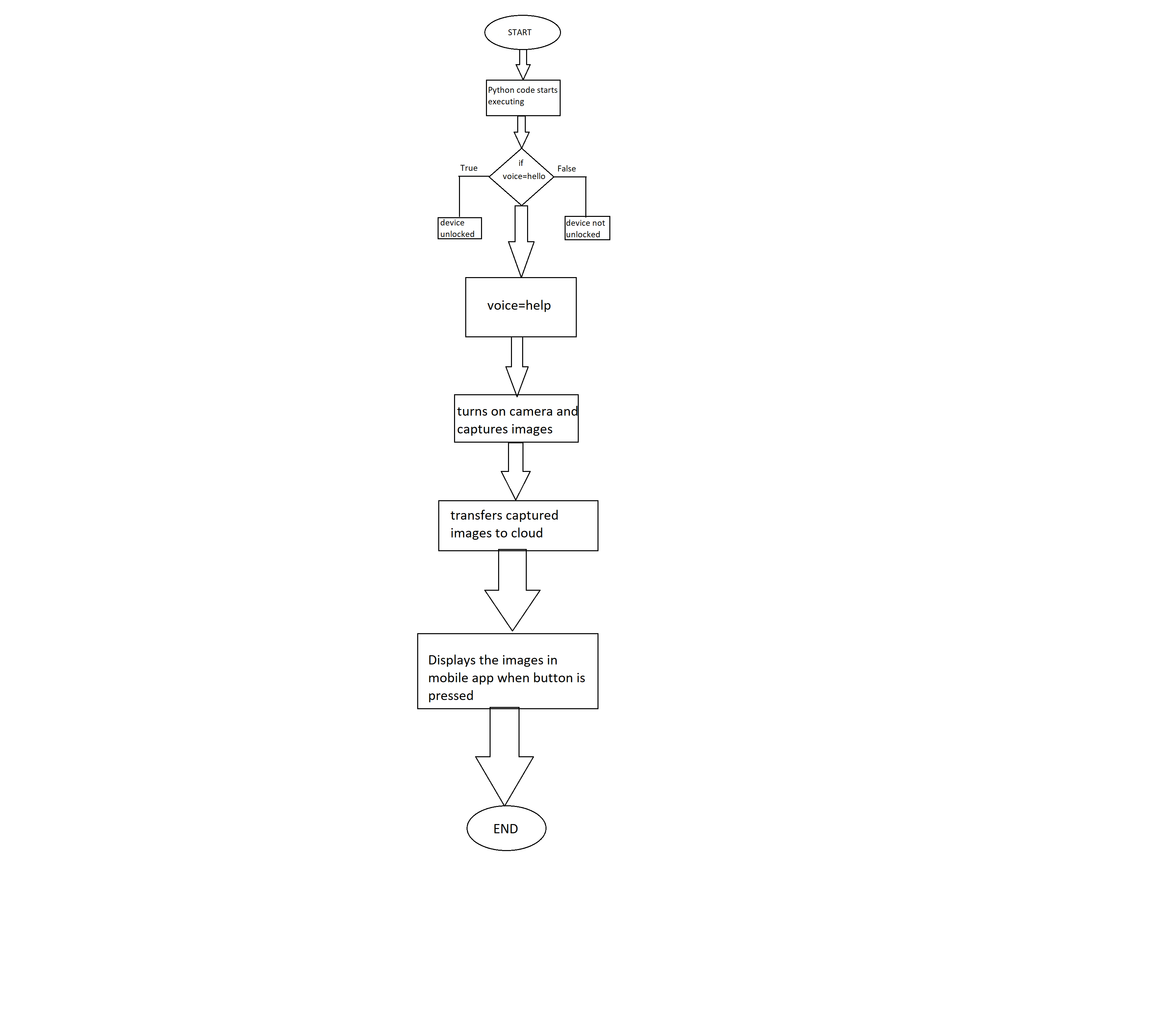
##### 3.2 Hardware/Software design

Python IDLE , Node\_Red , Cloudant DB, speech-To-Text , MIT APP inventor,Fast2SmS

### 4.Experimental Investigation

we observed when the voice command hello is given the device gets unlocked and when we give voice command other than hello device won't unlocked. if we give voice command as help then the camera wakes up and clicks the photo and upload it to the cloud and sends the image to mobile application

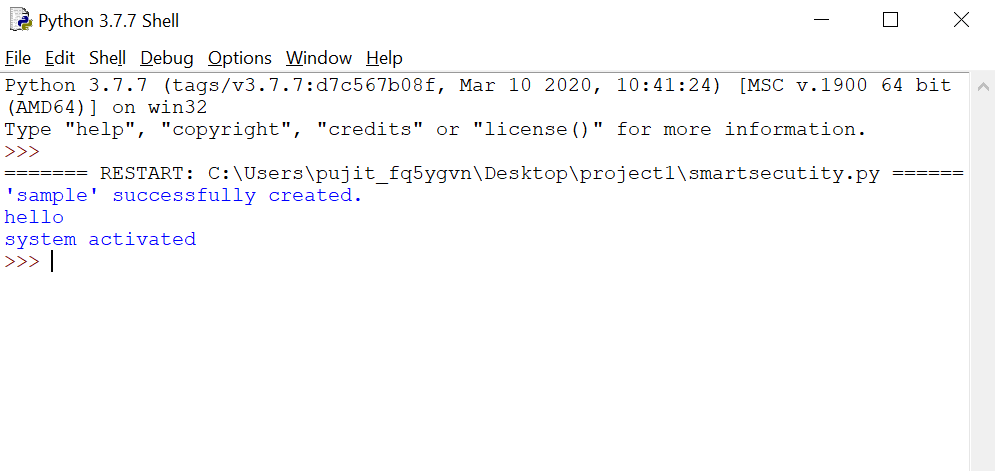
### 5.Flow Chart



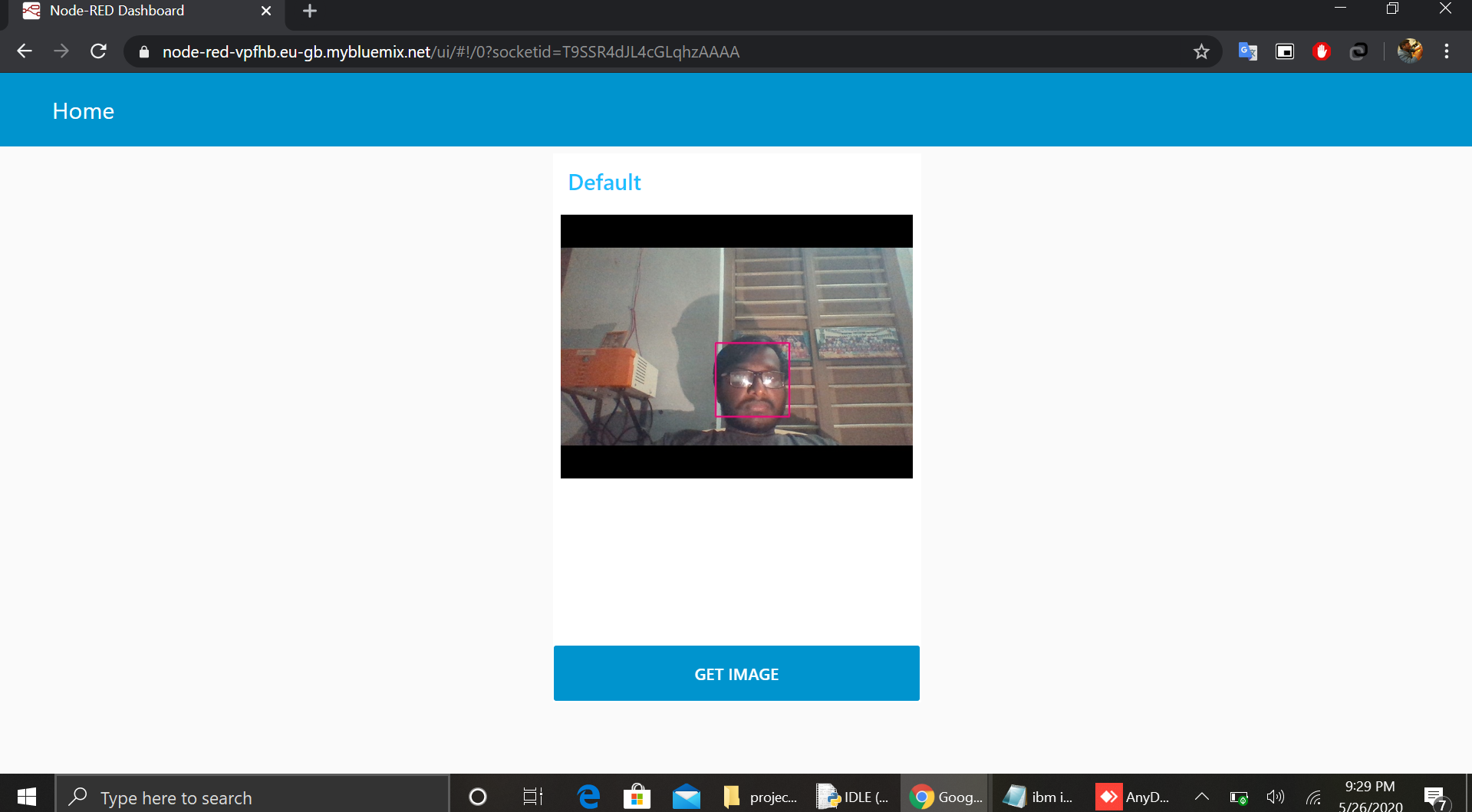
### 6.Results

output of python code

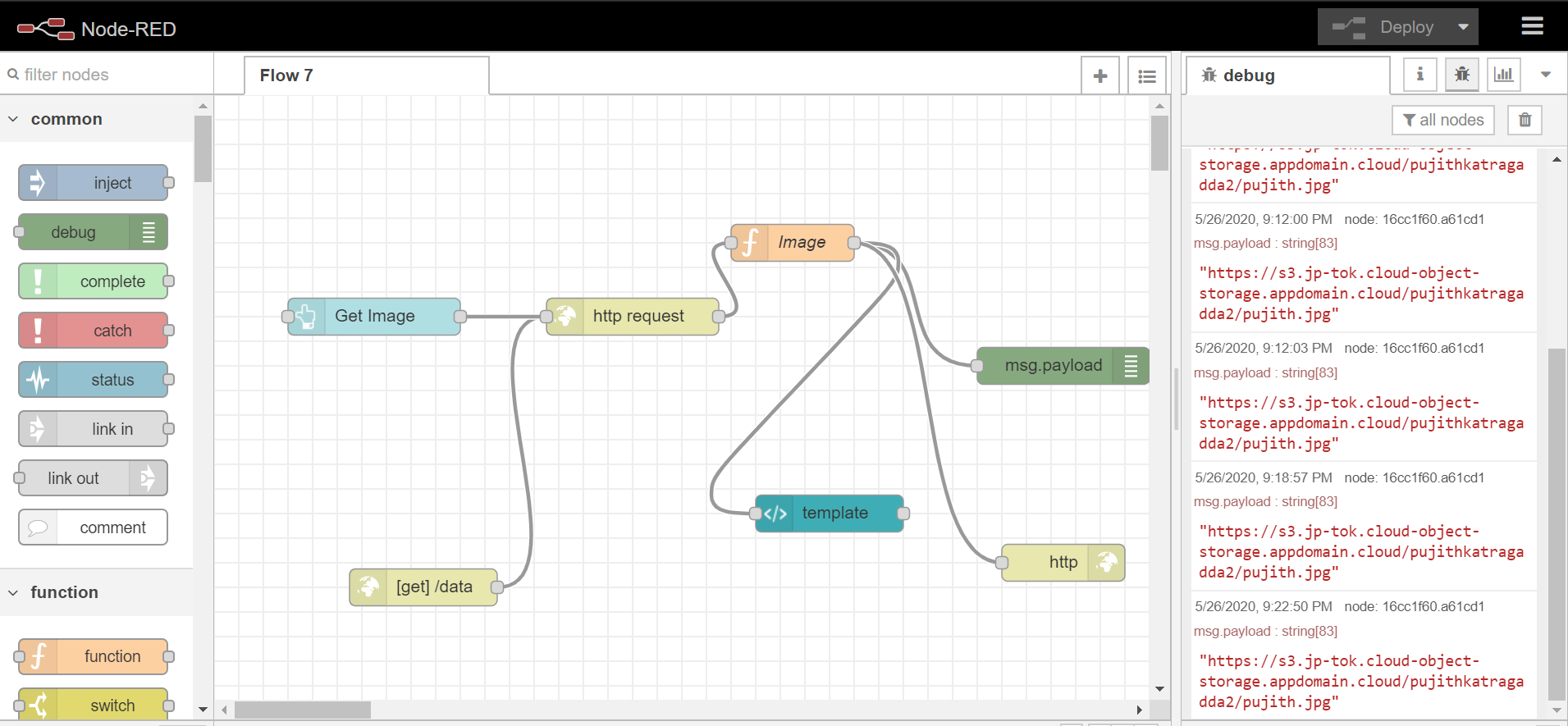




#### Output of NODE RED ui

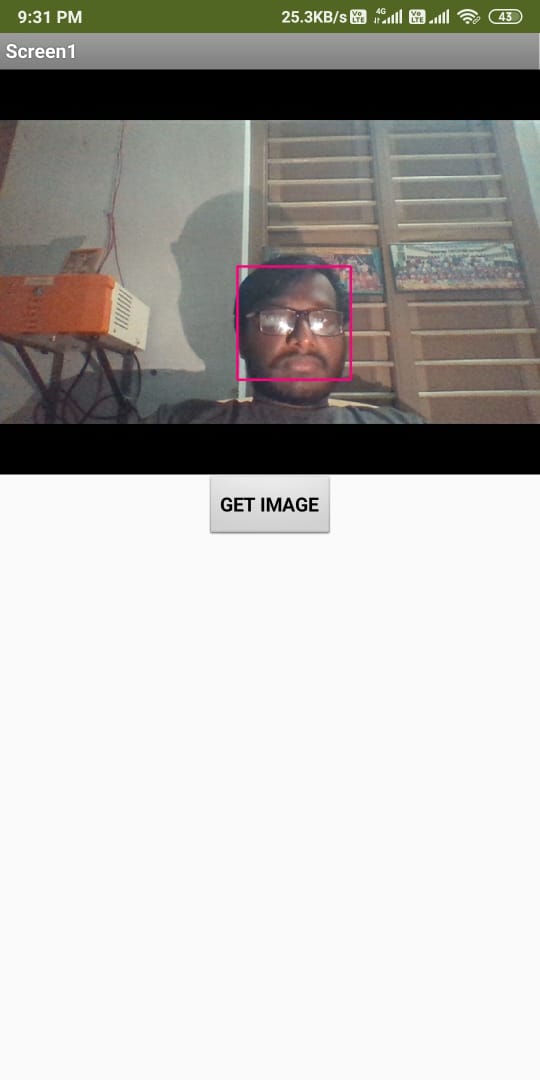


### Output of node RED flow



### 

### Output of MIT APP inventor



### 7.Advantages & Disadvantages

##### Advantages

1. Access Data
2. Cost Effective
3. Privacy and Security
4. Dependability

##### Disadvantages

1.compatibility

2.complexity

3.Technlogy takes control of life

4.lesser employment of menial staff

### 8.Applications

1.smart cities

2.Home automation

3.smart security

4.Health Monitoring

### 9.Conclusion

with this project if any unauthorized person tries to unlock our device the device captures the images and send them to cloud and also to the mobile application

### 10.Future Scope

This developed system can also be installed in industrial and commercial applications such as personal lockers in banks where only authorized persons have access to unlock the device

### 11.Bibilography

1.kumar mandula"mobile based home automation using internet of things(iot)"2015(iccicct),ieee

##### Appendix

import json

from os.path import join, dirname

from ibm\_watson import SpeechToTextV1

from ibm\_cloud\_sdk\_core.authenticators import IAMAuthenticator

import cv2

import numpy as np

import datetime

import ibm\_boto3

from ibm\_botocore.client import Config, ClientError

from playsound import playsound

from cloudant.client import Cloudant

from cloudant.error import CloudantException

from cloudant.result import Result, ResultByKey

#It will read the first frame/image of the video

video=cv2.VideoCapture(0)

face\_classifier=cv2.CascadeClassifier("haarcascade\_frontalface\_default.xml")

authenticator = IAMAuthenticator('F81s0KCKqAmAvZMErULa2x5pFKRVxKaQ0whfHiEsqOKZ')

speech\_to\_text = SpeechToTextV1(

authenticator=authenticator

)

speech\_to\_text.set\_service\_url('https://api.au-syd.speech-to-text.watson.cloud.ibm.com/instances/26b487b3-9f78-466e-aefc-f493c3583c7d')

COS\_ENDPOINT = "https://s3.jp-tok.cloud-object-storage.appdomain.cloud" # Current list avaiable at https://control.cloud-object-storage.cloud.ibm.com/v2/endpoints

COS\_API\_KEY\_ID = "0ic2LN\_Lle86XvjBAft-BrTyCaFWwzxQGfnKH6Ryds7o" # eg "W00YiRnLW4a3fTjMB-odB-2ySfTrFBIQQWanc--P3byk"

COS\_AUTH\_ENDPOINT = "https://iam.cloud.ibm.com/identity/token"

COS\_RESOURCE\_CRN = "crn:v1:bluemix:public:cloud-object-storage:global:a/fbf65509580645fba6b9c6048de4e823:2c5780c1-1e7f-469b-8760-7b33c30c52cf::" # eg "crn:v1:bluemix:public:cloud-object-storage:global:a/3bf0d9003abfb5d29761c3e97696b71c:d6f04d83-6c4f-4a62-a165-696756d63903::"

# Create resource

cos = ibm\_boto3.resource("s3",

ibm\_api\_key\_id=COS\_API\_KEY\_ID,

ibm\_service\_instance\_id=COS\_RESOURCE\_CRN,

ibm\_auth\_endpoint=COS\_AUTH\_ENDPOINT,

config=Config(signature\_version="oauth"),

endpoint\_url=COS\_ENDPOINT

)

client = Cloudant("e110f3bc-b2e8-483b-9c09-7ebd72c72c1b-bluemix", "962ef67bd0897965b5ce8d4964fee0cf992c757adb5a35a6c185163dda592dbf", url="https://e110f3bc-b2e8-483b-9c09-7ebd72c72c1b-bluemix:962ef67bd0897965b5ce8d4964fee0cf992c757adb5a35a6c185163dda592dbf@e110f3bc-b2e8-483b-9c09-7ebd72c72c1b-bluemix.cloudantnosqldb.appdomain.cloud")

client.connect()

#Provide your database name

database\_name = "sample"

my\_database = client.create\_database(database\_name)

if my\_database.exists():

print(f"'{database\_name}' successfully created.")

def audiofile(a):

with open(join(dirname(\_\_file\_\_), './.', a),

'rb') as audio\_file:

results = speech\_to\_text.recognize(

audio=audio\_file,

content\_type='audio/mp3',

).get\_result()

b=results['results'][0]['alternatives'][0]['transcript']

print(b)

if b[0:4]=="help":

cam()

elif b[0:5]=="hello":

print("system activated")

else:

print("give the correct input")

def multi\_part\_upload(bucket\_name, item\_name, file\_path):

try:

print("Starting file transfer for {0} to bucket: {1}\n".format(item\_name, bucket\_name))

# set 5 MB chunks

part\_size = 1024 \* 1024 \* 5

# set threadhold to 15 MB

file\_threshold = 1024 \* 1024 \* 15

# set the transfer threshold and chunk size

transfer\_config = ibm\_boto3.s3.transfer.TransferConfig(

multipart\_threshold=file\_threshold,

multipart\_chunksize=part\_size

)

# the upload\_fileobj method will automatically execute a multi-part upload

# in 5 MB chunks for all files over 15 MB

with open(file\_path, "rb") as file\_data:

cos.Object(bucket\_name, item\_name).upload\_fileobj(

Fileobj=file\_data,

Config=transfer\_config

)

print("Transfer for {0} Complete!\n".format(item\_name))

except ClientError as be:

print("CLIENT ERROR: {0}\n".format(be))

except Exception as e:

print("Unable to complete multi-part upload: {0}".format(e))

def cam():

while True:

#capture the first frame

check,frame=video.read()

gray=cv2.cvtColor(frame, cv2.COLOR\_BGR2GRAY)

#detect the faces from the video using detectMultiScale function

faces=face\_classifier.detectMultiScale(gray,1.3,5)

print(faces)

#drawing rectangle boundries for the detected face

for(x,y,w,h) in faces:

cv2.rectangle(frame, (x,y), (x+w,y+h), (127,0,255), 2)

cv2.imshow('Face detection', frame)

cv2.imwrite("face.jpg",frame)

multi\_part\_upload("pujithkatragadda2", "pujith.jpg", r"face.jpg")

json\_document={"link":COS\_ENDPOINT+"/"+"pujithkatragadda2"+"/pujith.jpg"}

new\_document = my\_database.create\_document(json\_document)

# Check that the document exists in the database.

if new\_document.exists():

print(f"Document successfully created.")

#waitKey(1)- for every 1 millisecond new frame will be captured

Key=cv2.waitKey(1)

if Key==ord('q'):

#release the camera

video.release()

#destroy all windows

cv2.destroyAllWindows()

a="help.mp3"

playsound(a)

audiofile(a)