SMART DOORBELL

A project report submitted in partial fulfilment of the requirements for the award ofthe Degree of **Bachelor of Technology** in

COMPUTER SCIENCE AND ENGINEERING

By

Rahul Mohan(19011M2102)

Saket Gande(19011M2108)

A.Sai Akash(19011M2217)

Under the guidance of

Dr. D. Vasumathi

Professor & Head of the Department of cse

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING JNTUH UNIVERSITY COLLEGE OF ENGINEERING KUKATPALLY, HYDERABAD-500 085

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING JNTUH UNIVERSITY COLLEGE OF ENGINEERING KUKATPALLY, HYDREABAD-500 085

DECLARATION

We, **Rahul Mohan(19011M2102), Saket Gande(19011M2108), A.Sai Akash(19011M2217)** hereby declare that the project report entitled "**SMART DOORBELL**", carried out by us under the guidance of **Dr. D. Vasumathi**, is submitted in partial fulfilment of the requirements for the award of the Degree of **Bachelor of Technology**. This is a record of bonafide work carried out by us and the results embodied in this project have not been reproduced
/copied from any source.

-Rahul Mohan(19011M2102)
-Saket Gande(19011M2108)
-A.Sai Akash(19011M2217)

Date:

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING JNTUH UNIVERSITY COLLEGE OF ENGINEERING KUKATPALLY, HYDREABAD-500 085

CERTIFICATE BY THE SUPERVISOR

This is to certify that the project report entitled "**SMART DOORBELL**", being submitted by **Rahul Mohan(19011M2102), Saket Gande(19011M2108), A.Sai Akash(19011M2217),**in partial fulfilment of the requirements for the award of the Degree of **Bachelor of Technology**, is a record of bonafide work carried out by them. The results are verified and found satisfactory.

Dr. D Vasumati,

Professor & Head of the Department,

Department of Computer Science and Engineering,

JNTUH College of Engineering,

Hyderabad.

Date:

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING JNTUH UNIVERSITY COLLEGE OF ENGINEERING KUKATPALLY, HYDREABAD-500 085

CERTIFICATE BY THE HEAD OF THE DEPARTMENT

This is to certify that the project report entitled "**SMART DOORBELL**", being submitted by **Rahul Mohan(19011M2102), Saket Gande(19011M2108), A.Sai Akash(19011M2217),** in partial fulfilment of the requirements for the award of the Degree of **Bachelor of Technology**, is a record of bonafide work carried out by them. The results are verified and found satisfactory.

Dr. D Vasumati,

Professor & Head of the Department,

Department of Computer Science and Engineering,

JNTUH College of Engineering,

Hyderabad.

Date:

ACKNOWLEDGEMENT

We would like to express sincere thanks to our Supervisor **Dr. D Vasumati**, Professor and Head of the department of Computer Science &Engineering, JNTUH-CEH for her admirable guidance and inspiration both theoretically and practically and most importantly for the drive to complete the project successful. Working under such an eminent guide was our privilege.

We owe a debt of gratitude to **Dr. D Vasumati** , Professor and Head of the department ofComputer Science &Engineering for her kind consideration and encouragement in carryingout this project successfully.

We are grateful to the Project Review Committee members and Department of ComputerScience & Engineering who have helped in successfully completing this project by givingtheir valuable suggestion and support.

We express thanks to our parents for their love, care and moral support without which wewould have not been able to complete this project. It has been a constant source of inspiration for all our academic endeavor. Last but not the least, we thank the Almighty formaking us a part of the world.

-Rahul Mohan(19011M2102)

-Saket Gande(19011M2108)

-A.Sai Akash(19011M2217)

Date:

ABSTRACT

In today's world, we might have visitors at unexpected times (i.e the times when we are outside the home or quarantined at our home due to viral disease). During such situations, it is hard to attend to our visitors and maybe we can miss meeting some essential visitors. We want to create a solution for this problem by creating a smart door Smart Doorbell / Video Intercom System model. Which would help us to interact with our visitors when we are unable to meet them.

It specifically focuses on the development of an IOT based home automation system that is able to control various components via internet or be automatically programmed to operate from ambient conditions. In this project, we design the development of a firmware for smart control which can successfully be automated minimizing human interaction to preserve the integrity within whole electrical devices in the home.

Our Smart Doorbell / Video Intercom System opens a virtual meeting door for our visitors when they press a button. It also sends us a notification through the internet with a link to interact with the guest.

In this project, we want to use the concept of IoT and Raspberry-pi-powered hardware to interact with our visitors through internet calls.

TABLE OF CONTENTS

1. INTRODUCTION

1.1 Objective

Internet of Things (IOT) is a concept where each device is assign to an IP address and through that IP address anyone makes that device identifiable on internet. The mechanical and digital machines are provided with unique identifiers (UIDs) and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction. Basically, it started as the "Internet of Computers." Research studies have forecast an explosive growth in the number of "things" or devices that will be connected to the Internet. The resulting network is called the "Internet of Things" (IoT). The recent developments in technology which permit the use of wireless controlling environments like, Bluetooth and Wi-Fi that have enabled different devices to have capabilities of connecting with each other. Smart Doorbell is a intercom device which opens a virtual meeting door for our visitors when they press a button. It also sends us a notification through the internet with a link to interact with the guest.

Elements of Smart Doorbell:

i. Photo capturing

ii. Gmail notification

Characteristics of Smart Library:

i. Notifies the owner if any person is at our doorstep and rings the bell.

ii. Provides video com interface through Gmail notification.

Objectives of proposed system:

- User friendly
- Time feasibility
- Accurate and efficient

1.2 Scope

This provides a contactless interface between owner and visitors.

This system provides the feature that can be used in every household doorbell.

1.3 Overview of the Report:

Smart Doorbell have been playing an important role in protecting the security of modern homes since they were invented . A doorbell allows visitors to announce their presence and request entry into a building as well as enables the occupant to verity the identity of the guests to help prevent home robbery or invasion at a moment's notice . There are two types of doorbells depending on the requirement of wall wiring: the wired doorbells and the wireless doorbells . The former requires a wire to connect both

the front door button and the back door button to a transformer, while the latter transfer the signal wirelessly using telephone technology . Modern buildings are typically equipped with wireless doorbell systems that employ radio technology to signal doorbells and answer the doors remotely. Although these doorbells are much more convenient than wired ones, they do not always satisfy the demands of modern homes for the following three reasons. First, the answering machines are normally located at a fixed place (often near to the door), if a occupant wants to answer the doorbell, he/she has to go to the answering machines. Second, if the occupant would like to see the visitors outside, he/she has to go to door. Third, the occupant has no way to answer or admit guests when he/she is not at home, nor to keep a record of guests. As smart home technology matures, smart doorbells can solve this problem greatly by connecting the doorbells to the Internet(or a local network) and allowing users to answer the bell through a smart device such as a smartphone or tablet . This enables a home owner to answer and admit a visitor anywhere when a smart device connecting to the Internet is available .

A smart doorbell is an integral part of a smart home, which helps protect the security of the home by avoiding unwanted access such as robbery and invasion . The controller of the smart home can potentially answer the bell and decide whether to admit a visitor outside the door or not through adaptive learning and other technologies . Because of the important role that smart doorbells play on building a smart home, many techniques and methodologies have been invented during past few years . The existing smart

doorbells provide an integrated solution, which means that the working mechanisms or the implementation details are hidden and unknown to the users. If there is a failure, users have to seek help from professionals for repairs or maintenance. It is also very likely that users need to replace the whole smart doorbells due to a failure of a component in the system.

1.4 BENEFITS

Reduced wiring issues

Considering the increase in price of copper, thus increases the possibility of the wire to be stolen.The use of a wireless remote system to control home appliances means no wire for thieves to steal.

Extended range

We can connect to doorbell from where ever we are since its is connected to WiFi.

Security

As the connection of the control of the HAS is established over a secure network the systemensures security to the maximum extent.

Integrable and extensive nature

The prototype designed can be integrated to a larger scale. Also it has an extensive nature beingable to add or remove the appliances under control according to application.

2.ANALYSIS AND DESIGN

2.1.1 Use-Case Diagram

Use-case diagrams describe the high-level functions and scope of a system. These diagrams also identify the interactions between the system and its actors. The use cases and actors in use-case diagramsdescribe what the system does and how the actors use it, but not how the system operates internally.

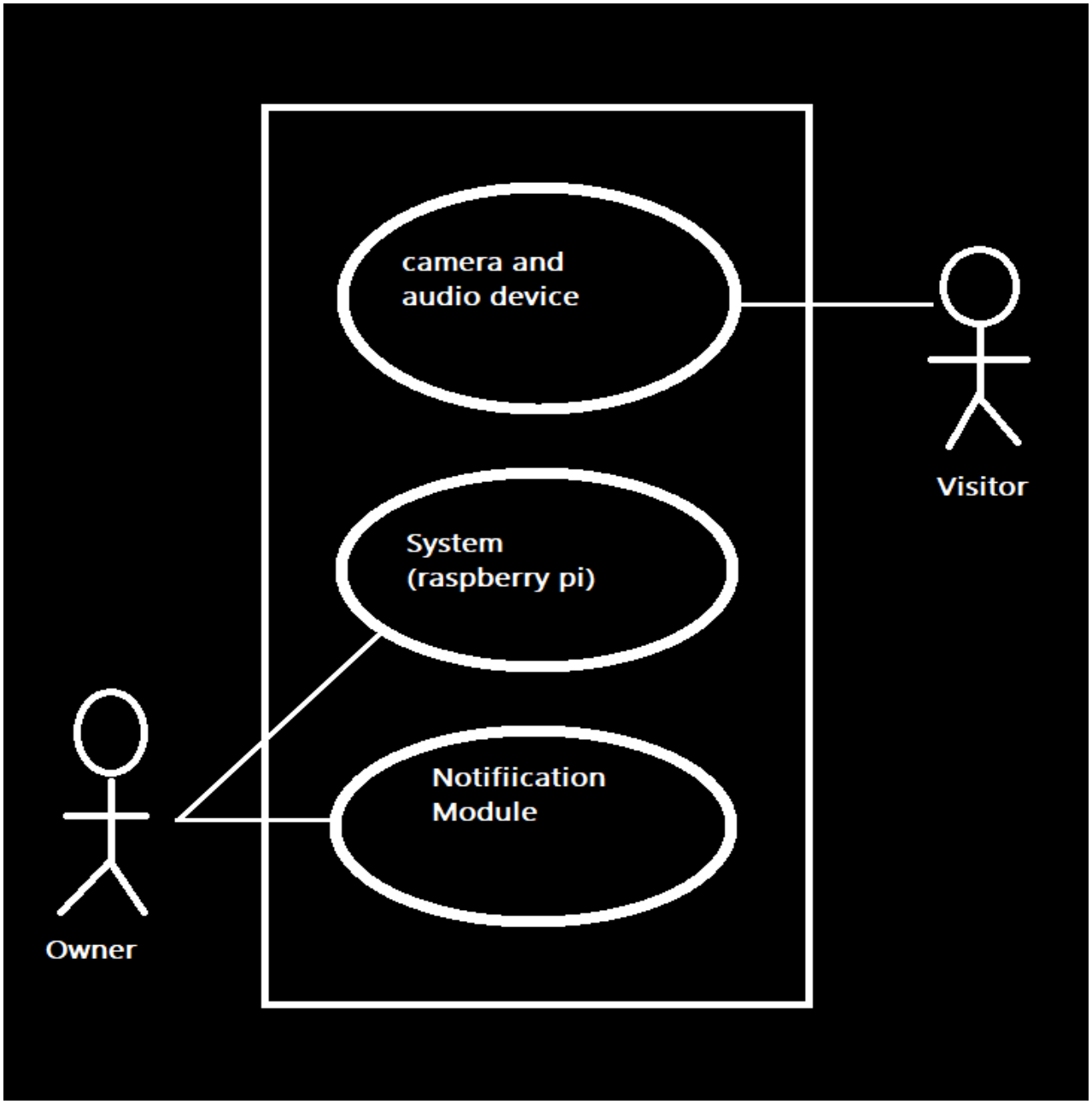


Figure 2.2 Use-Case Diagram

2.1.2 Sequence Diagram

A sequence diagram is a type of interaction diagram because it describes how—and in what order-a group of objects works together. These diagrams are used by software developers and

business professionals to understand requirements for a new system or to document an existing process.

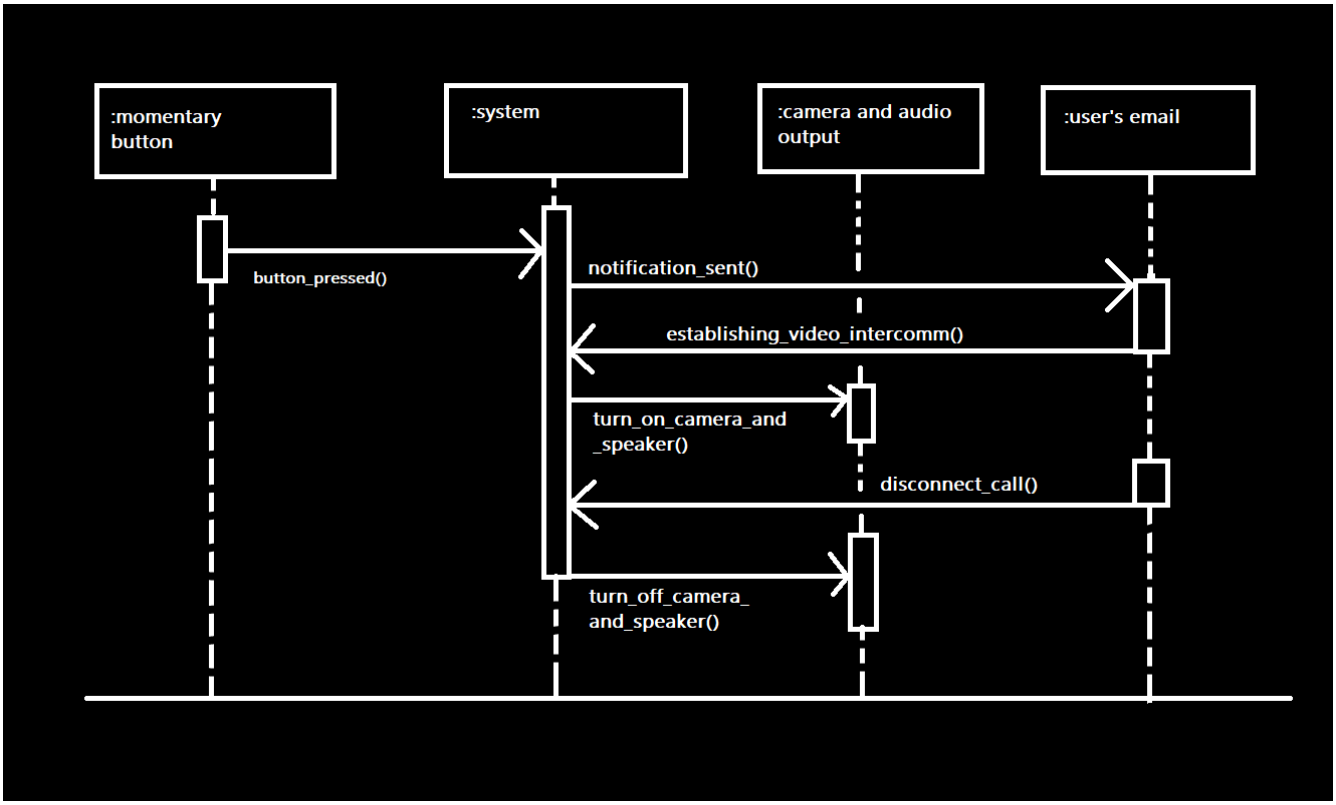
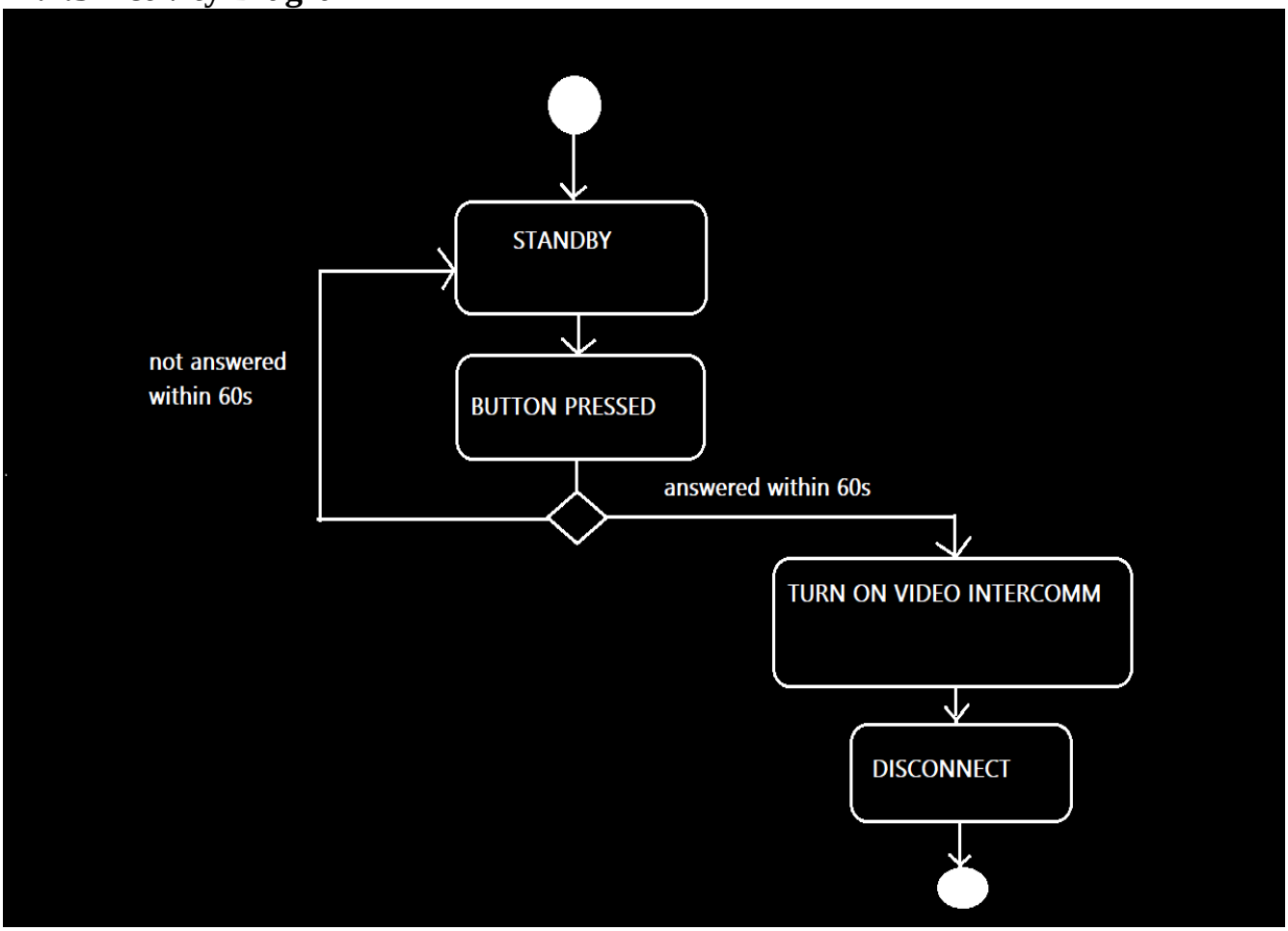


Figure 2.3 Sequence Diagram

2.1.3 Activity Diagram



3.THEORY

3.1IOT (INTERNET OF THINGS)

IOT as a term has evolved long way as a result of convergence of multiple technologies, machine learning, embedded systems and commodity sensors. IOT is a system of interconnected devices assigned a UIDS, enabling data transfer and control of devices over a network. It reduced the necessity of actual interaction in order to control a device. IOT is an advanced automation and analytics system which exploits networking, sensing, big data, and artificial intelligence technology to deliver complete systems for a product or service. These systems allow greater transparency, control, and performance when applied to any industry or system.

3.1.1 Features of IOT

3.1.1.1 Intelligence

IOT comes with the combination of algorithms and computation, software & hardware that makesit smart. Ambient intelligence in IOT enhances its capabilities which facilitate the things to respondin an intelligent way to a particular situation and supports them in carrying out specific tasks. In spite of all the popularity of smart technologies, intelligence in IOT is only concerned as a means ofinteraction between devices, while user and device interaction are achieved by standard input methods and graphical user interface

3.1.1.2 Connectivity

Connectivity empowers the Internet of Things by bringing together everyday objects. Connectivity of these objects is pivotal because simple object level interactions contribute towards collective intelligence in the IOT network. It enables network accessibility and compatibility in the things.
With this connectivity, new market opportunities for the Internet of things can be created by the networking of smart things and applications

3.1.1.3 Dynamic Nature

The primary activity of Internet of Things is to collect data from its environment, this is achieved with the dynamic changes that take place around the devices. The state of these devices change dynamically, example sleeping and waking up, connected and/or disconnected as well as the context of devices including temperature, location and speed. In addition to the state of the device,the number of devices also changes dynamically with a person, place and time

3.1.1.4 Enormous Scale

The number of devices that need to be managed and that communicate with each other will be much larger than the devices connected to the current Internet. The management of data

generated from these devices and their interpretation for application purposes becomes more critical. Gartner (2015) confirms the enormous scale of IOT in the estimated report where it statedthat 5.5 million new things will get connected every day and 6.4 billion connected things will be in use worldwide in 2016, which is up by 30 percent from 2015. The report also forecasts that the number of connected devices will reach 20.8 billion by 2020

3.1.1.5 Sensing

IOT wouldn't be possible without sensors that will detect or measure any changes in the environment to generate data that can report on their status or even interact with the environment. Sensing technologies provide the means to create capabilities that reflect a true awareness of the physical world and the people in it. The sensing information is simply the analog input from the physical world, but it can provide a rich understanding of our complex world

3.1.1.6 Heterogeneity

Heterogeneity in Internet of Things as one of the key characteristics. Devices in IOT are based on different hardware platforms and networks and can interact with other devices or service platforms through different networks. IOT architecture should support direct network connectivity between heterogeneous networks. The key design requirements for heterogeneous things and their environments in IOT are scalabilities, modularity, extensibility and interoperability.

3.1.1.7 Security

IOT devices are naturally vulnerable to security threats. As we gain efficiencies, novel experiences, and other benefits from the IOT, it would be a mistake to forget about security concerns associated with it. There is a high level of transparency and privacy issues with IOT. It is important to secure the endpoints, the networks, and the data that is transferred across all of it means creating a security paradigm.

3.1.2 Advantages of IOT

3.1.2.1 Communication

IOT encourages the communication between devices, also famously known as Machine-to-Machine (M2M) communication. Because of this, the physical devices are able to stay connected and hence the total transparency is available with lesser inefficiencies and greater quality.

3.1.2.2 Automation and Control

Due to physical objects getting connected and controlled digitally and centrally with wireless infrastructure, there is a large amount of automation and control in the workings. Without human intervention, the machines are able to communicate with each other leading to faster and timely output.

3.1.2.3 Information

It is obvious that having more information helps making better decisions. Whether it is mundane decisions as needing to know what to buy at the grocery store or if your company has enough widgets and supplies, knowledge is power and more knowledge is better.

3.1.2.4 Monitor

The second most obvious advantage of IOT is monitoring. Knowing the exact quantity of supplies or the air quality in your home, can further provide more information that could not have previously been collected easily. For instance, knowing that you are low on milk or printer ink could save you another trip to the store in the near future. Furthermore, monitoring the expiration of products can and will improve safety.

3.1.2.5 Time

As hinted in the previous examples, the amount of time saved because of IOT could be quite large. And in today's modern life, we all could use more time.

3.1.2.6 Money

The biggest advantage of IOT is saving money. If the price of the tagging and monitoring equipment is less than the amount of money saved, then the Internet of Things will be very widely adopted.

IOT fundamentally proves to be very helpful to people in their daily routines by making the appliances communicate to each other in an effective manner thereby saving and conserving energy and cost. Allowing the data to be communicated and shared between devices and then translating it into our required way, it makes our systems efficient.

3.1.2.7 Automation of daily tasks leads to better monitoring of devices

The IOT allows you to automate and control the tasks that are done on a daily basis, avoiding human intervention. Machine-to-machine communication helps to maintain transparency in the processes. It also leads to uniformity in the tasks. It can also maintain the quality of service. We can also take necessary action in case of emergencies.

3.1.2.8 Efficient and Saves Time

The machine-to-machine interaction provides better efficiency, hence; accurate results can be obtained fast. This results in saving valuable time. Instead of repeating the same tasks every day, it enables people to do other creative jobs.

3.1.2.9 Saves Money

Optimum utilization of energy and resources can be achieved by adopting this technology and keeping the devices under surveillance. We can be alerted in case of possible bottlenecks, breakdowns, and damages to the system. Hence, we can save money by using this technology.

3.1.2.10 Better Quality of Life

All the applications of this technology culminate in increased comfort, convenience, and better management, thereby improving the quality of life.

3.1.2.11 Compatibility

Currently, there is no international standard of compatibility for the tagging and monitoring equipment. I believe this disadvantage is the most easy to overcome. The manufacturing companies of these equipment just need to agree to a standard, such as Bluetooth, USB, etc. This is nothing new or innovative needed.

3.1.2.12 Complexity

As with all complex systems, there are more opportunities of failure. With the Internet of Things, failures could sky rocket. For instance, let's say that both you and your spouse each get a message saying that your milk has expired, and both of you stop at a store on your way home, and you both purchase milk. As a result, you and your spouse have purchased twice the amount that you both need. Or maybe a bug in the software ends up automatically ordering a new ink cartridge for your printer each and every hour for a few days, or at least after each power failure, when you only need a single replacement.

3.1.2.13 Privacy/Security

With all of this IOT data being transmitted, the risk of losing privacy increases. For instance, how well encrypted will the data be kept and transmitted with? Do you want your neighbours or employers to know what medications that you are taking or your financial situation?

3.1.2.14 Safety

Imagine if a notorious hacker changes your prescription. Or if a store automatically ships you an equivalent product that you are allergic to, or a flavour that you do not like, or a product that is already expired. As a result, safety is ultimately in the hands of the consumer to verify any and all automation.

As all the household appliances, industrial machinery, public sector services like water supply and transport, and many other devices all are connected to the Internet, a lot of information is available on it. This information is prone to attack by hackers. It would be

very disastrous if private and confidential information is accessed by unauthorized intruders.

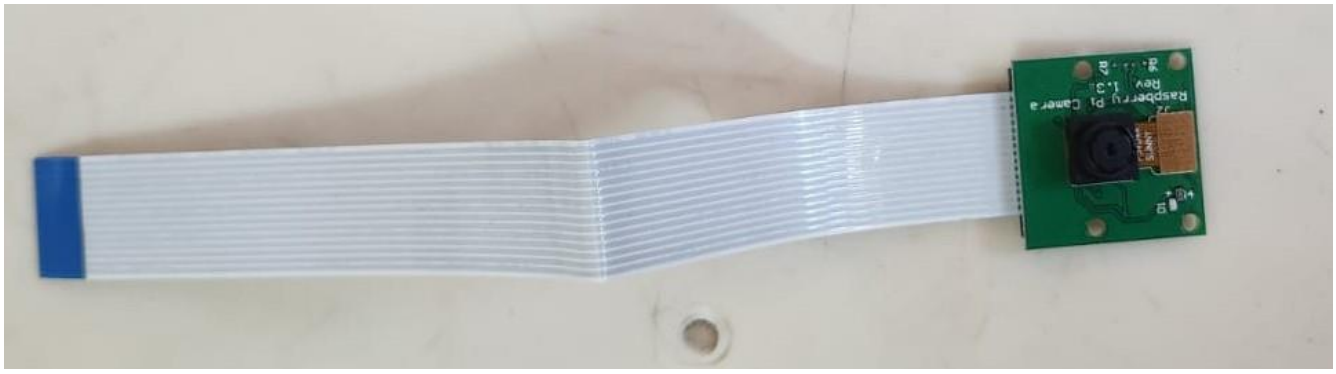
3.1.2.15 Lesser Employment of Menial Staff

The unskilled workers and helpers may end up losing their jobs in the effect of automation of daily activities. This can lead to unemployment issues in the society. This is a problem with the advent of any technology and can be overcome with education. With daily activities getting automated, naturally, there will be fewer requirements of human resources, primarily, workers and less educated staff. This may create Unemployment issue in the society.

4.Components and diagram

1. The Raspberry Pi Camera Module is a custom designed add-on for Raspberry Pi. It attaches to Raspberry Pi by way of one of the small sockets on the board upper surface. This interface uses the

dedicated CSi interface, designed especially for interfacing to cameras.

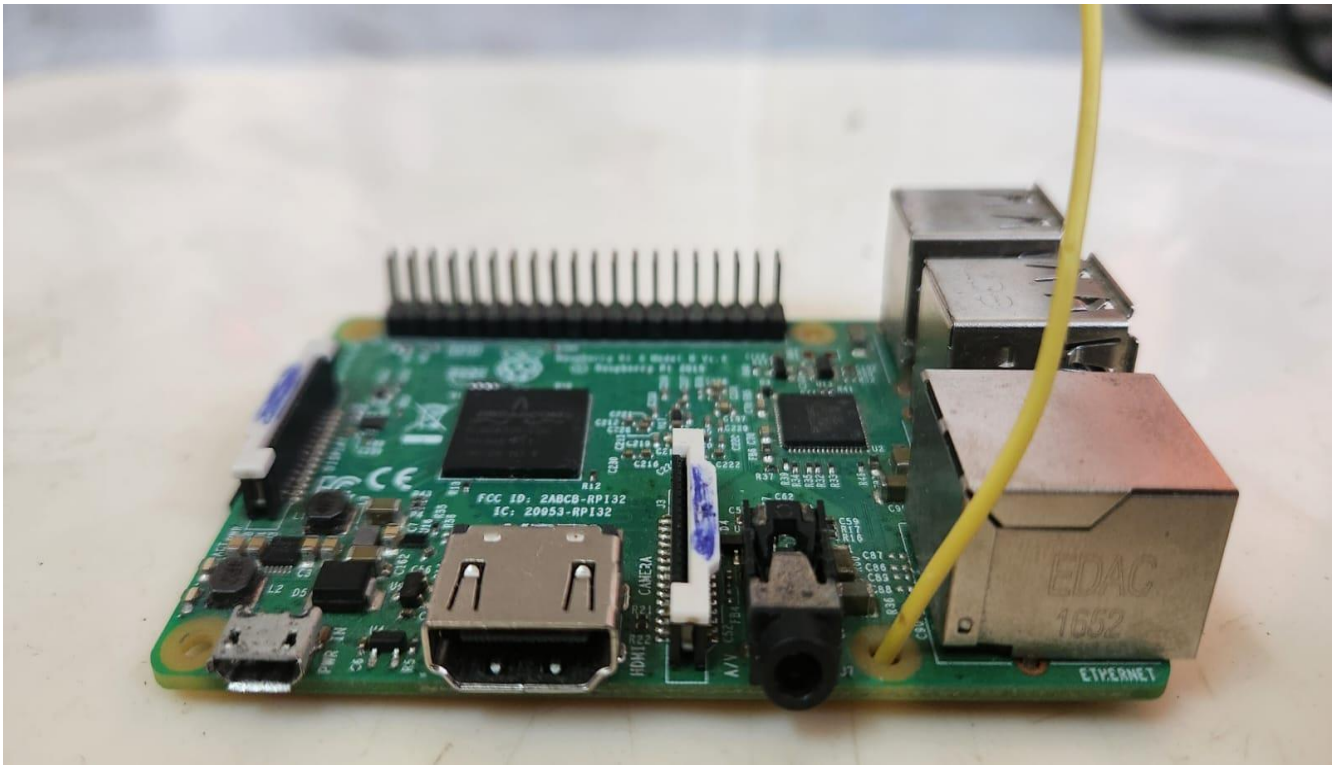


2.Rasberry Pi Speaker

3.USB Microphone

4.Push Button

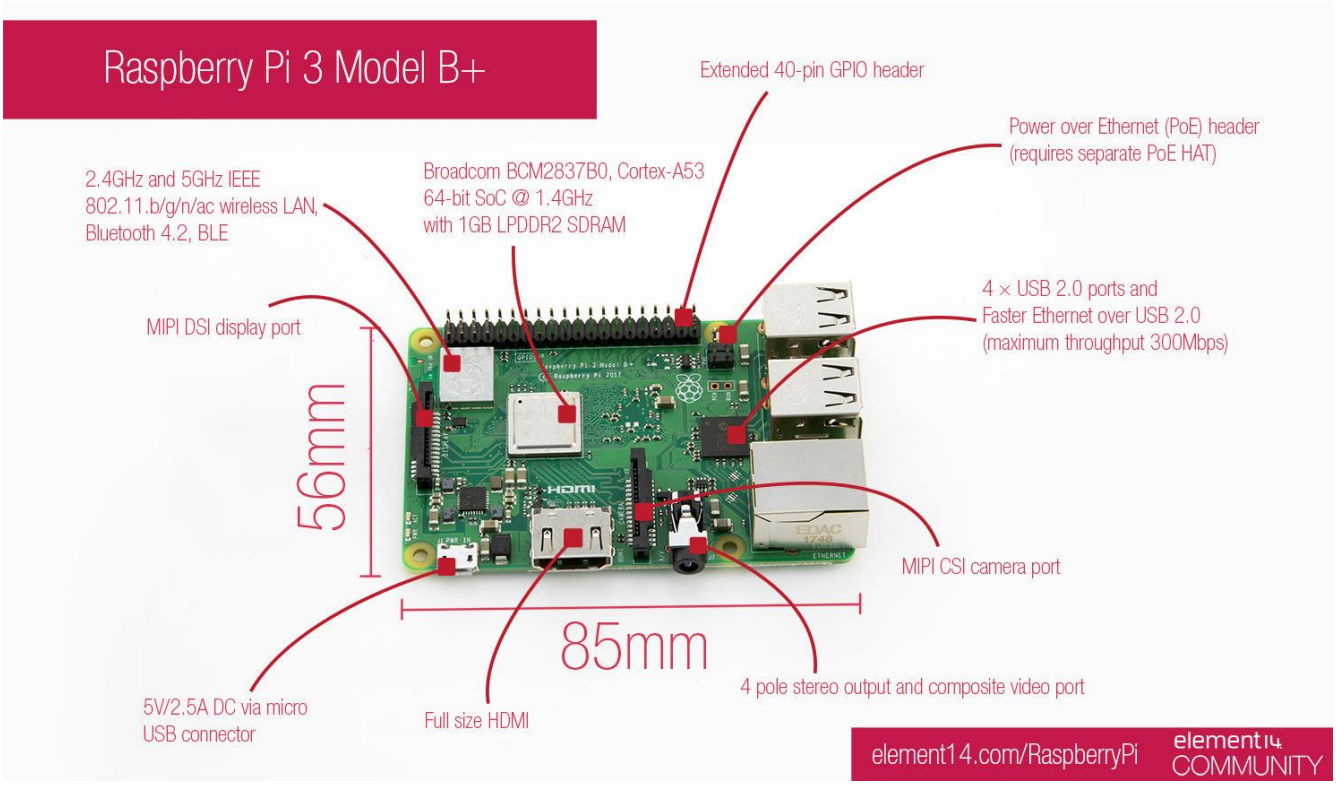
5. Rasberry Pi 3B+



The Raspberry Pi 3 Model B+ is the latest product in the Raspberry Pi 3 range, boasting a 64-bit quad core processor running at 1.4GHz, dual-band 2.4GHz and 5GHz wireless LAN, Bluetooth 4.2/BLE, faster Ethernet, and PoE capability via a separate PoE HAT

The dual-band wireless LAN comes with modular compliance certification, allowing the board to be designed into end products with significantly reduced wireless LAN compliance testing, improving both cost and time to market.

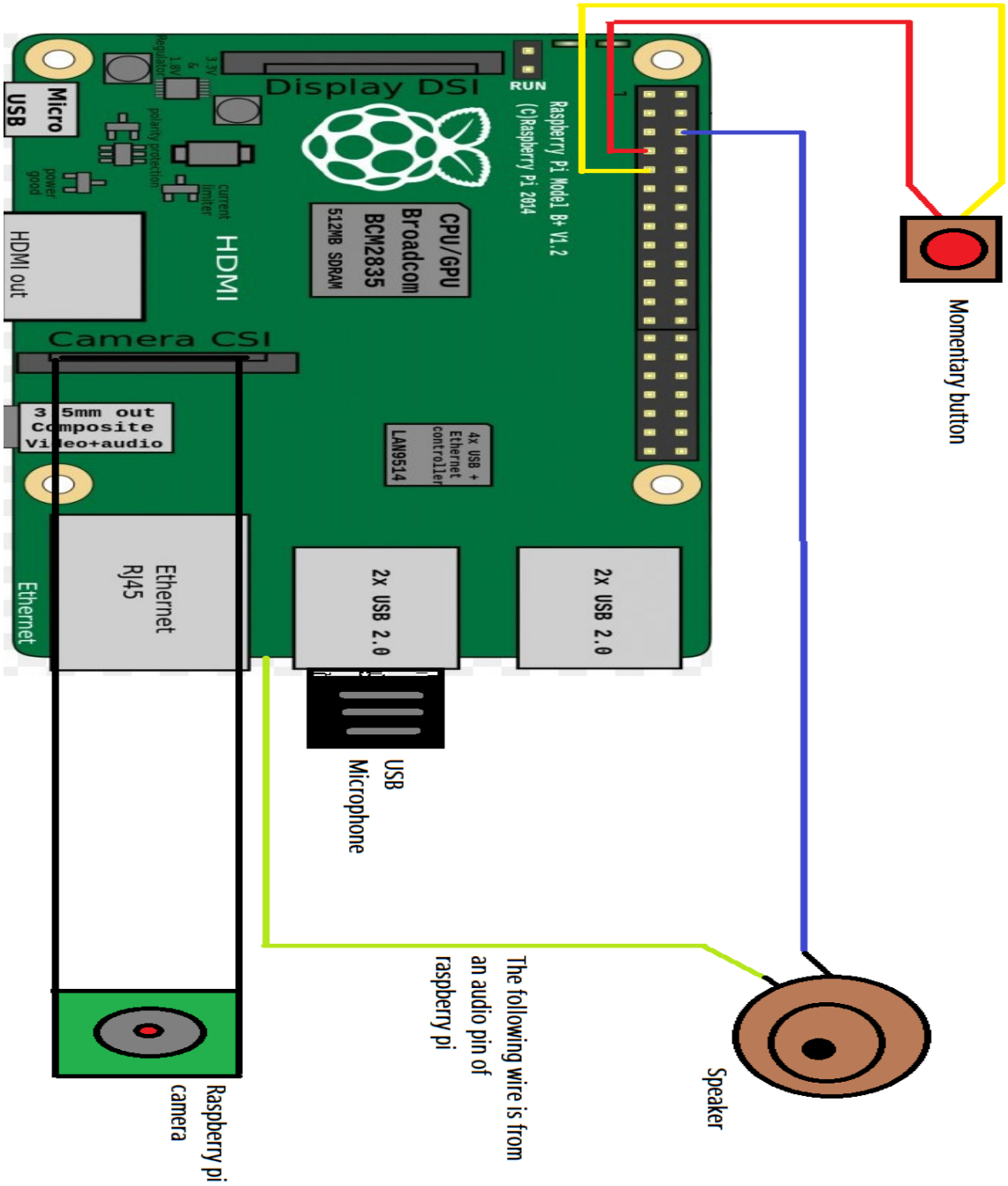
The Raspberry Pi 3 Model B+ maintains the same mechanical footprint as both the Raspberry Pi 2 Model B and the Raspberry Pi 3 Model B.



Specifications

Type of Hardware	Description
System on Chip (SoC)	BCM2837
CPU	Quad Cortex A53 @ 1.2 GHz
Instruction set	ARMv8-A
RAM	1GB SDRAM
Storage slot	Yes, for a microSD
Ethernet	10/100
Wireless	802.11n / Bluetooth 4.0 [12]
Supported Interface	Video Output: HDMI / Composite Audio Output: HDMI / Headphone GPU: 400MHz VideoCore IV
USB Port	4
GPIO	40

RASBERRY PI-DIAGRAM



5.PROPOSED SOLUTION AND IMPLEMENTATION

5.1PROPOSED SOLUTION:

- Our smart doorbell/video intercom system opens a virtual meeting door for our visitors when they press a button
- It sends us a notification through the internet with a link to interact with the guests
- The solution features include:
    - Voice and video assisted interface
    - Instant online alerts on mobile through email
    - Photo Capture
    - Real time location

5.2-IMPLEMENTATION:

Implementation is the stage where the theoretical design is turned into a working system. The most crucial stage in achieving a new successful system and in giving confidence on the new system for the users that it will work efficiently and effectively. It involves careful planning, investigation of the current system and its constraints on implementation, design of method to achieve the change-over and an evaluation of change over methods apart from planning.

The implementation phase comprises of several activities. The required hardware and software acquisition is carried out.

1. **Attaching the button to the wires:**
   For the button, we used a momentary push switch. To connect the button to the Raspberry Pi, we took two jumper wires and stripped the ends off one side of the wires (keeping the female connectors).

   One end of the wire is being attached to the GPIO 16 pin whereas the other end of the wire is grounded (attached to pin number 34)

2. **Attach components to the housing:**

It includes plugging of USB mic to an USB slot of raspberry pi, which is used to capture the audio from the visitor's end. Connecting the speaker which transmits the voice from the owner's end to the doorbell. An end of the speaker is connected to the audio pin of the raspberry pi and the other end of the raspberry pi is connected to the ground pin (here the ground pin being PIN 39). And at last Arducam OV5647 5MP Camera The camera module will be connected directly to the Raspberry Pi. It will provide 720p 60fps video capturing and streaming for the doorbell, which is a requirement for most customers. The camera provides the basis for most of the most of the doorbell's security features. The video from the camera will be streamed to a web server, which can then be accessed by the homeowner.

3. Configuring the raspberry pi:

In this section, we will be working on the raspberry pi's operating system. In the following section, we will be using a 16GB memory card to store the OS contents of raspberry pi. For this project, we have selected 09-04-2017 updated version of raspberry pi's software to work on our smart doorbell.

Steps for installing raspberry pi's operating system:

- Downloading the imager file of the raspberry pi's OS. Source: http://downloads.raspberrypi.org/raspbian/images/raspbian-2019-04-09/2019-04-09-raspbian-strech.zip
- Downloading the raspberry pi imager software, which helps us to install the raspberry pi OS into our selected memory card
- Connect our SD card to the system from where we have downloaded the imager file of raspberry pi. Format the contents of SD card
- Open raspberry pi imager software, select the SD card as the storage and 2019-04-09-raspbian-strech.img file as the OS, which will be installed in the SD card.
- On selecting the respective OS and storage, click on write. Wait till the raspberry OS is completely installed the SD card.
- After the raspberry pi OS is completely installed in the SD card, eject the SD card and place it in the raspberry pi's SD card/memory card slot.

- Connect the raspberry pi to a monitor, also connect a keyboard and mouse which are used in configuring the raspberry pi's settings and writing the code.
- Power up the raspberry pi and the monitor connected to it. Select the language and the time zone. Set the user credentials for the raspberry pi. Now the Raspberry Pi is ready to use

4. Configure the Raspberry Pi settings for peripherals:

We'll have to do run some commands on the Raspberry Pi to make the doorbell work.

Enable the Camera

Enable the camera with raspi-config. In the terminal, type:

```
sudo raspi-config
```

Navigate to Interfacing Options, then camera.

Select yes and reboot.

Testing that the camera is working by running the following command in the terminal to save an image to your local directory:

```
sudo raspistill -o test.jpg
```

Enable the Microphone

Open the sound input settings

```
alsamixer -c 1
```

Press F4 to open the capture settings and bump the level up to 100. Press Esc to exit.

Test the audio capture with the following command:

```
arecord --device=hw:1,0 --format S16_LE --rate 44100 -c1 test.wav -V mono
```

press control-c to stop. It will generate a audio file in your local directory called test.wav

You can play the file with:

```
aplay test.wav
```

Updating the system:

To use the chromium browser of the raspberry pi efficiently, we have to update our raspberry pi. For that, we run the following command on the terminal:

```
sudo apt-get update
sudo apt-get upgrade
```

Installing the essential python modules:

Smtplib, pyautogui, RPi.GPIO and mailutis packages are to be installed using the terminal

```
sudo apt-get install ssmtp
sudo apt-get install mailutils
sudo apt-get install RPi.GPIO
sudo apt-get install pyautogui
```

5. **Enabling the video calls:**
   We are using Jitsi Meet to create video calls because it's free and easy to use. Visit Jitsi Meet and configure the site to use your camera/microphone.

   Settings:

```
Camera:                         mmal                          service                          16
Microphone: USB PnP Sound Device, USB Audio-Default Audio Device2019-04-09-raspbian-strech.
zip
```

6. **Developing the python program for video intercommunication system:**

Our smart doorbell does the following functionalities:

- Detection of momentary button being pressed
- Sending an email to the recipient stating that a visitor is waiting at the door
- Establishing the video intercommunication system
- Photo capture of the moment

Detection of momentary button being pressed:

In this section, we use RPi.GPIO module to detect the momentary button being pushed.

The sample code which facilitates the detection of momentary button is as followed:

```
import RPi.GPIO as GPIO
try:
+    while True:
        if GPIO.input(BUTTON_PIN)==GPIO.LOW:
            emailer()
except KeyboardInterrupt:
    GPIO.cleanup()
```

This sample code is the driver code of the program. It will be executed first as the program starts. cleanup() to clean up all the ports you've used, whereas input() detects the static moment of the button i.e. HIGH or LOW (Unpressed or Pressed). It takes PIN number as the input.

Sending an email to the recipient stating that a visitor is waiting at the door:

In this method, an email is sent to the user stating that a visitor is waiting at the door. For the following sub-program or the method, we used smtplib module to send an email and MIMEText module to draft the email in python.

The Sample code which facilitates the following process is given as:

```
import smtplib
from email.mime.text import MIMEText
```

```
def emailer():
    recipent='receiver@gmail.com'
    sender='sender@outlook.com'
    password="********"
    msg='''Someone is waiting at door, open the door
      Meet link:
      https://meet.jit.si/someRandomCall
      '''
    sub="doorbell"
    server=smtplib.SMTP('smtp.outlook.com',587)
    server.starttls()
    server.login(sender,password)
    msg=MIMEText(msg)
```

```
    msg['Subject']=sub
    msg['From']=sender
    msg['To']=recipent
    msg.set_param('importance','high value')
    server.sendmail(sender,recipent,msg.as_string())
    print("done sending the email")
```

Multipurpose Internet Mail Extensions (MIME) is an Internet standard that extends the format of email messages to support text in character sets other than ASCII, as well as attachments of audio, video, images, and application programs. For the following program, we also attach a captured image

The smtplib module defines an SMTP client session object that can be used to send mail to any internet machine with an SMTP or ESMTP listener daemon. For details of SMTP and ESMTP operation, consult RFC 821 (Simple Mail Transfer Protocol) and RFC 1869 (SMTP Service Extensions).

Capturing the image:

While momentary button is pushed, it captures the image of the visitor and attaches it to the email

Sample program to capture the image:

```
def capture_img():
if not os.path.exists(dir):
        os.makedirs(dir)
```

Establishing the video communication system:

We use Jitsi Meet to establish the video intercom between the visitor and the house owner. For that we use the following sample code in order to establish the intercom system:

```
import webbrowser
import pyautogui
import time
```

```
import os
```

```
url='https://meet.jit.si/someRandomCall'
webbrowser.open(url)
time.sleep(8)
pyautogui.press('enter')
time.sleep(60)
# Close the browser window
os.system("taskkill /f /im chrome.exe")
```

The webrowser package is used to open a jitsi meet session and the pyautogui package is used to perform some keyboard operations which helps to enter the jitsi meet.

7. **Testing the developed program:**

After wiring the complete code save the file with .py extension and execute this script using below command in the terminal

```
python filename.py
```

8. **Setting up the following program at the boot:**

   To avoid running the script every time you boot the pi, make the script executable and make it run automatically on every boot. To do this we execute the following command:

```
sudo nano /etc/rc.local
```

And paste python /home/pi/Desktop/doorbell.py & before ‘exit 0’ as shown in the below snapshot and save the file using ctrl+x and hit enter. Make sure you entered the correct destination of your python script.

Now, restart the Raspberry Pi and press the push button. If everything works fine you should receive an e-mail with photo as an attachment.

6-Libraries Used

SMTP Server

This module offers several classes to implement SMTP (email) servers.

The SMTPChannel has the following instance variables:

smtp_server

Holds the SMTPServer that spawned this channel.

conn

Holds the socket object connecting to the client.

addr

Holds the address of the client, the second value returned by socket.accept

received_lines

Holds a list of the line strings (decoded using UTF-8) received from the client. The lines have their "\r\n" line ending translated to "\n".

smtp_state

Holds the current state of the channel. This will be either COMMAND initially and then DATA after the client sends a "DATA" line.

seen_greeting

Holds a string containing the greeting sent by the client in its "HELO".

mailfrom

Holds a string containing the address identified in the "MAIL FROM:" line from the client.

rcpttos

Holds a list of strings containing the addresses identified in the "RCPT TO:" lines from the client.

received_data

Holds a string containing all of the data sent by the client during the DATA state, up to but not including the terminating "\r\n.\r\n".

fqdn

Holds the fully qualified domain name of the server as returned by socket.getfqdn().

peer

Holds the name of the client peer as returned by conn.getpeername() where conn is conn.

The SMTPChannel operates by invoking methods named smtp_<command> upon reception of a command line from the client. Built into the base SMTPChannel class are methods for handling the following commands (and responding to them appropriately):

Command	Action taken
HELO	Accepts the greeting from the client and stores it in seen_greeting. Sets server to base command mode.
EHLO	Accepts the greeting from the client and stores it in seen_greeting. Sets server to extended command mode.
NOOP	Takes no action.
QUIT	Closes the connection cleanly.
MAIL	Accepts the "MAIL FROM:" syntax and stores the supplied address as mailfrom. In extended command mode, accepts the **RFC 1870** SIZE attribute and responds appropriately based on the value of *data_size_limit*.
RCPT	Accepts the "RCPT TO:" syntax and stores the supplied addresses in the rcpttos list.
RSET	Resets the mailfrom, rcpttos, and received_data, but not the greeting.
DATA	Sets the internal state to DATA and stores remaining lines from the client in received_data until the terminator "\r\n.\r\n" is received.
HELP	Returns minimal information on command syntax

Command	Action taken
VRFY	Returns code 252 (the server doesn’t know if the address is valid)
EXPN	Reports that the command is not implemented.

7-Code Snippet

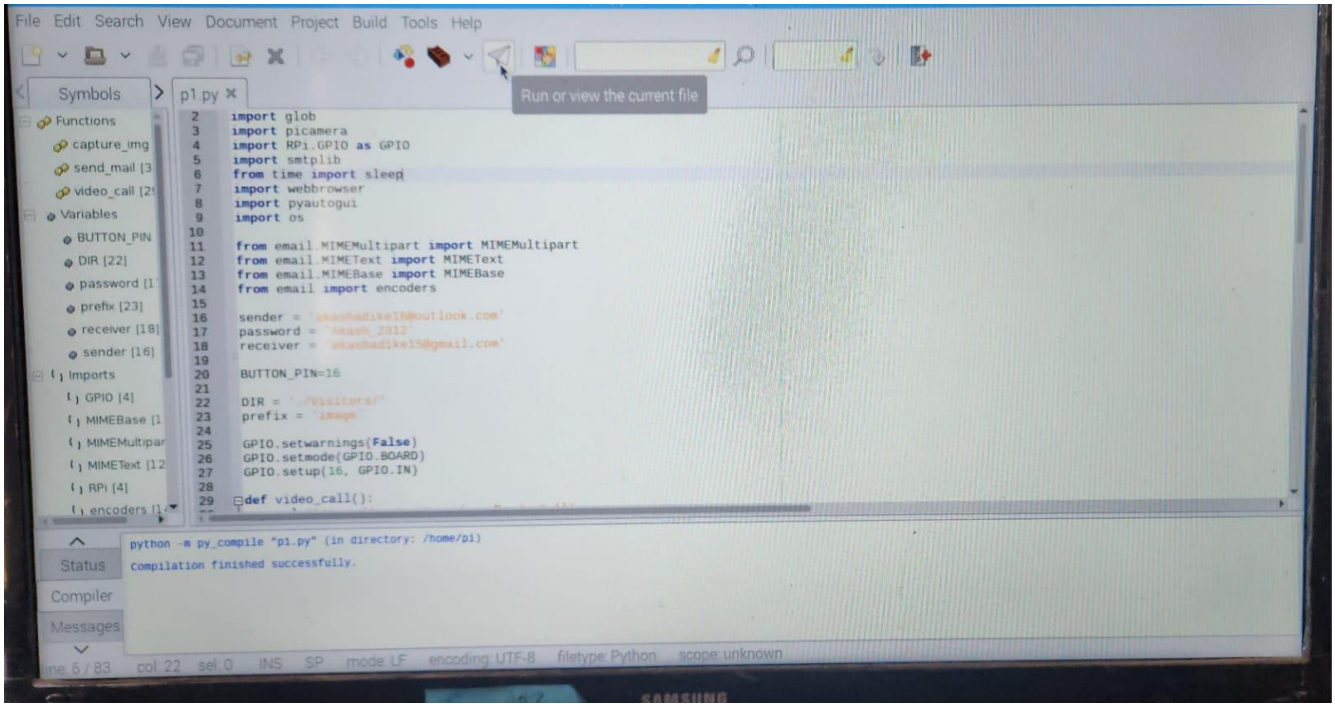
```
import os
import glob
import picamera
import RPi.GPIO as GPIO
import smtplib
from time import sleep
import webbrowser
import pyautogui
import os

from email.MIMEMultipart import MIMEMultipart
from email.MIMEText import MIMEText
from email.MIMEBase import MIMEBase
from email import encoders

sender = 'sender@outlook.com'
password = '**********'
receiver = 'receiver@gmail.com'

BUTTON_PIN=16

DIR = './Visitors/'
prefix = 'image'

GPIO.setwarnings(False)
GPIO.setmode(GPIO.BOARD)
GPIO.setup(16, GPIO.IN)

def video_call():
    url='https://meet.jit.si/someRandomCall'
    webbrowser.open(url)
    sleep(5)
    pyautogui.press('enter')
    sleep(60)
    os.system("taskkill /f /im chrome.exe")

def send_mail(filename):
    msg = MIMEMultipart()
    msg['From'] = sender
```

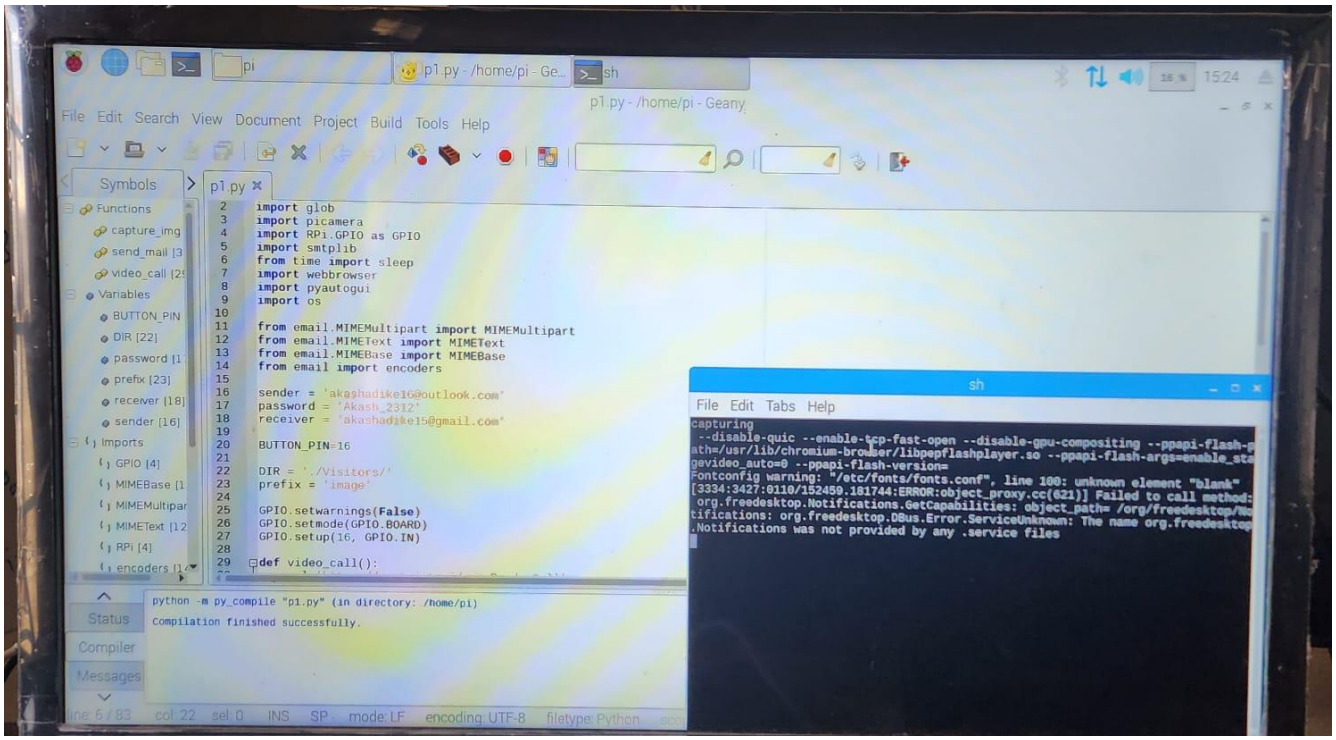
```
    msg['To'] = receiver
    msg['Subject'] = 'Visitor'

    body = 'You have a visitor in the house. Check them out here:
     https://meet.jit.si/someRandomCall'
    msg.attach(MIMEText(body, 'plain'))
    attachment = open(filename, 'rb')
    part = MIMEBase('application', 'octet-stream')
    part.set_payload((attachment).read())
    encoders.encode_base64(part)
    part.add_header('Content-Disposition', 'attachment; filename= %s' % filename)
    msg.attach(part)
    server = smtplib.SMTP('smtp.outlook.com', 587)
    server.starttls()
    server.login(sender, password)
    text = msg.as_string()
    server.sendmail(sender, receiver, text)
    server.quit()



def capture_img():
    print('capturing')
    if not os.path.exists(DIR):
        os.makedirs(DIR)
    files = sorted(glob.glob(os.path.join(DIR, prefix + '[0-9][0-9][0-9].jpg')))
    count = 0

    if len(files) > 0:
        count = int(files[-1][-7:-4])+1
    filename = os.path.join(DIR, prefix + '%03d.jpg' % count)
    with picamera.PiCamera() as camera:
        pic = camera.capture(filename)
    send_mail(filename)
    video_call()
try:
     while True:
            if GPIO.input(BUTTON_PIN)==GPIO.LOW:
                    capture_img()

except KeyboardInterrupt:
     GPIO.cleanup()
```

8.RESULT

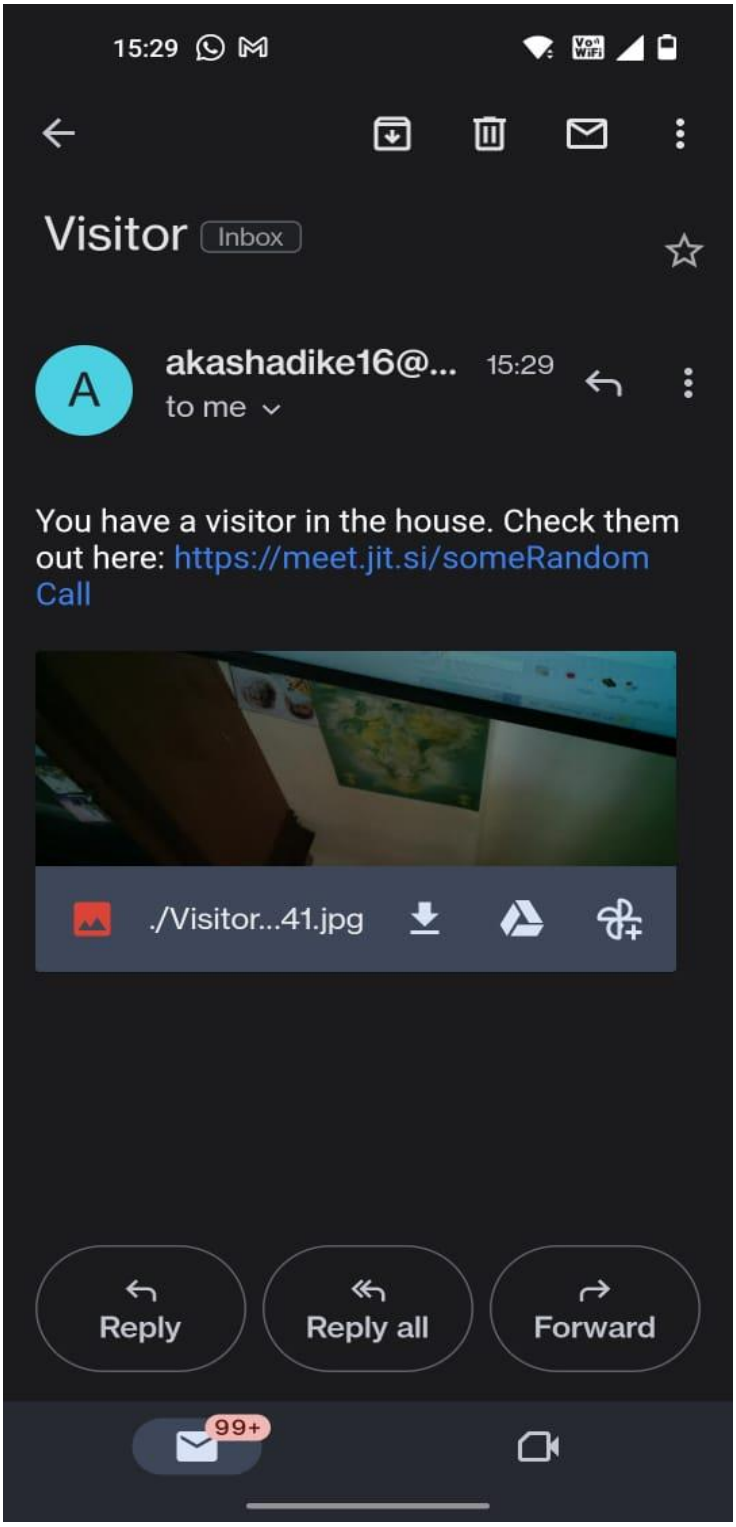
Step1: Ring the bell

Step 2: Code runs in the background

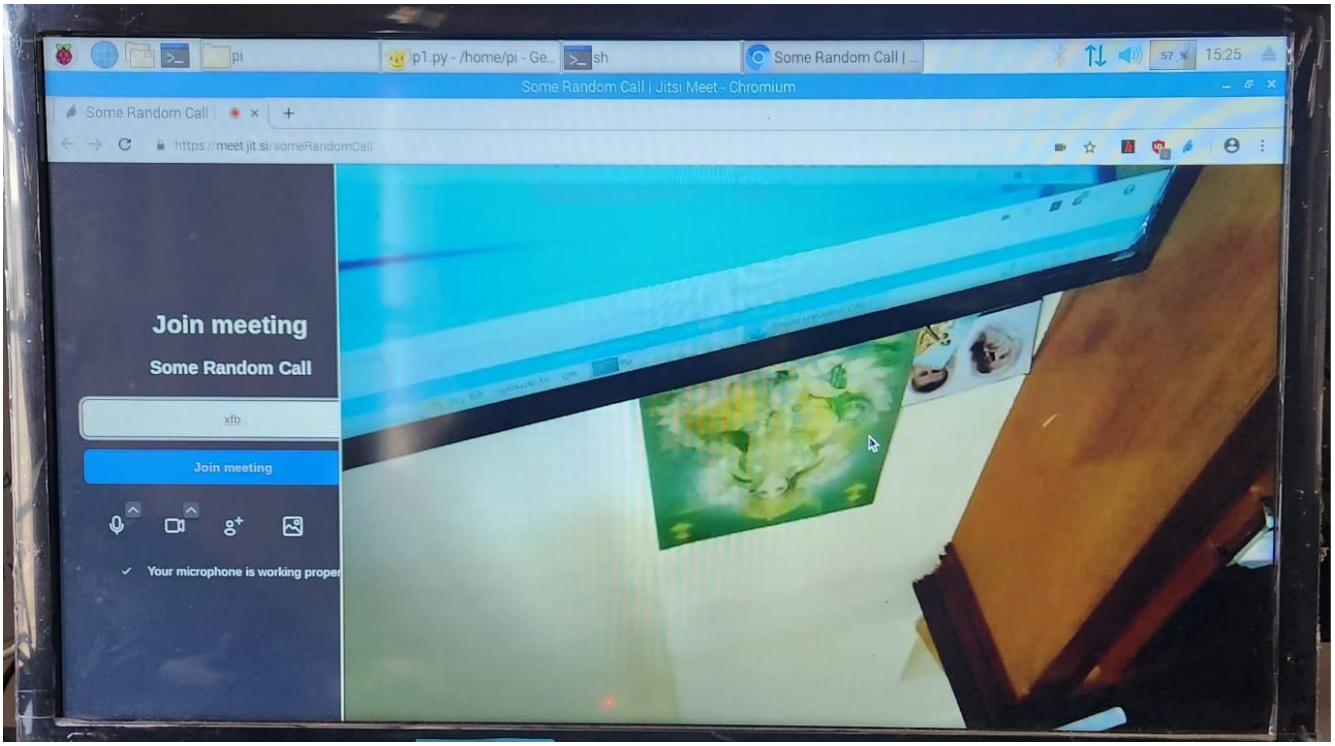



p1.py - /home/pi - Geany
File Edit Search View Document Project Build Tools Help
Symbols
p1.py
Functions
capture_img
send_mail [3
video_call [2
Variables
BUTTON_PIN
DIR [22]
password [1
prefix [23]
receiver [18]
sender [16]
Imports
GPIO [4]
MIMEBase [1
MIMEMultipar
MIMEText [12
RPi [4]
encoders [1
import glob
import picamera
import RPi.GPIO as GPIO
import smtplib
from time import sleep
import webbrowser
import pyautogui
import os
from email.MIMEMultipart import MIMEMultipart
from email.MIMEText import MIMEText
from email.MIMEBase import MIMEBase
from email import encoders
sender = 'akashadike16@outlook.com'
password = 'Akash_2312'
receiver = 'akashadike15@gmail.com'
BUTTON_PIN=16
DIR = './Visitors/'
prefix = 'image'
GPIO.setwarnings(False)
GPIO.setmode(GPIO.BOARD)
GPIO.setup(16, GPIO.IN)
def video_call():
Status
Compiler
Messages
python -m py_compile "p1.py" (in directory: /home/pi)
Compilation finished successfully.
sh
File Edit Tabs Help
capturing
--disable-quic --enable-tcp-fast-open --disable-gpu-compositing --ppapi-flash-p
ath=/usr/lib/chromium-browser/libpepflashplayer.so --ppapi-flash-args=enable_sta
gevideo_auto=0 --ppapi-flash-version=
Fontconfig warning: "/etc/fonts/fonts.conf", line 100: unknown element "blank"
[3334:3427:0110/152450.181744:ERROR:object_proxy.cc(621)] Failed to call method:
org.freedesktop.Notifications.GetCapabilities: object_path= /org/freedesktop/No
tifications: org.freedesktop.DBus.Error.ServiceUnknown: The name org.freedesktop
.Notifications was not provided by any .service files

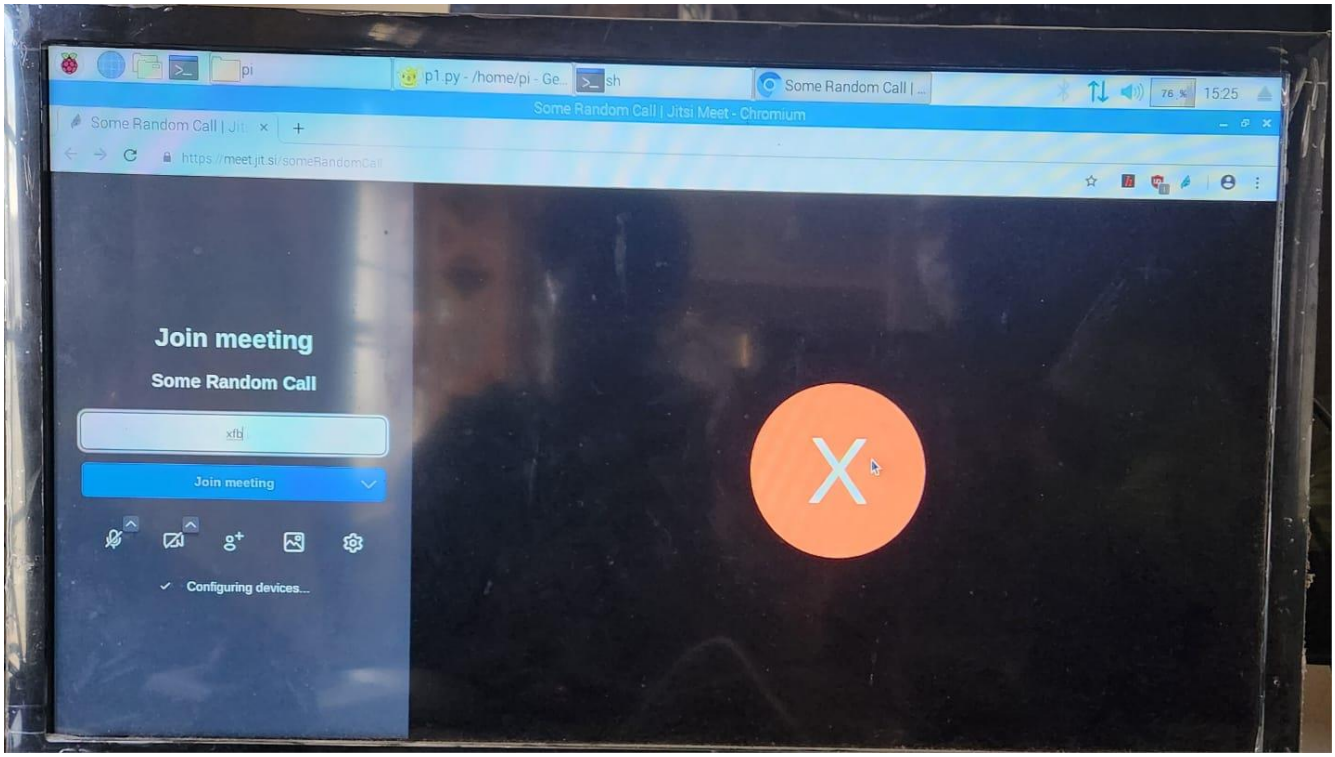
Step 3: Notification is sent to owner’s Mail



Step 4: Owner can answer the call through JITSI MEET



Step 5: The Meet closes



9.CONCLUSION:

- Today, in uncertain times we might have unexpected guests and we won't be able to interact with them properly due to viral diseases (such as COVID 19 pandemic) or we might outside of our house.
- We believe that this to be an efficient approach to interact with guests at uncertain times from any corner of the world, and to maintain social distancing
- Our project has currently able to send notifications through email to the owner regarding the visitor's visit and was able to capture a photo of the visitor.
- We are currently trying to get a proper video intercom interface which can quality of video intercom communication.
- All the peripherals of the smart doorbell are properly working, and the code was able to execute at the boottime of the raspberry pi

In the event that any of these modules prove to be difficult, the team has a plan of attack to get the project back on schedule. If the application proves to be a problem, developing a web application may be more feasible given the time restraints. If the web server does not work, then implementing a server more compatible with the OpenTok SDK may be a better option.

If the audio is an issue, using the 3.5" audio jack for audio output may be an alternative worth looking into. Lastly, if all features work, but not together, it may be more feasible to run a demo showing all modules working instead of spending hours automating the modules together. Ideally, none of these potential problems will arise, and the team will be able to make their vision a reality. Regardless, this team has planned accordingly the best possible way to implement every module of this project within the scope of each member's individual abilities and will deliver a functional, polished product that performs all the functions outlined in this report. Along with that, we would also like to add new features to the smart doorbell such as facial detection, camera streaming etc.

10.Future Scope:

- Our Smart Doorbell can be an essential part of house automation.
- Further features such as a particular guest detection (such as detecting a courier boy or a family member and giving them a default message or instruction) and automated human detection (no usage of push button), can be added to our product

  • The Raspberry pi based door bell system is helpful for every types of user. Institutions according to their requirements. It is a paradigm for wireless door bell system based on new technologies.

  Further, in the future, it will become more secure compare to our previous model. Here we are specifying more secure features with good design.
- The proposed system is a suitable for low cost with

  more secure features MTQQ. Our next aim is to

  discuss about delay in between transmission of packet

  from sender to receiver part

11.REFERENCES

1. https//:www.researchgate.net
2. https//:www.wikipedia.org
3. https//:www.rapidonline.com
4. https//:www.schematics.com
5. https//:www.batteryuniversity.com
6. https//:www.thingspeak.com
7. https//:www.youtube.com