The purpose of this project was to replace a broken NuTone intercom system (which uses ethernet wiring as its connection) with a modern raspberry pi based system. Using the existing ethernet wiring has the advantage of not clogging up the wifi network with the audio signals

The materials used for this project:
(only one needed)
Raspberry Pi 3B+ or newer (as main controller / network access point)
POE Network switch
Raspberry Pi Touch Screen
(for each speaker)
Raspberry Pi Zero or Zero w
Waveshare audio hat (with 5W speakers you may want an additional one to make the main unit a speaker as well)
POE to microUSB and ethernet splitter
Ethernet to microUSB adapter
(for wiring)
If your system is like mine the ethernet cables are there but they don't have ethernet plugs on the ends so you will need proper tools for putting the cable ends on namely
RJ45 ends and crimper

SETUP
Here are the resources I used
Setting up the Raspberry Pi 3B+ as network access point:

https://linuxhint.com/raspberry_pi_wired_router/

Change default user (for convenience and security)
https://thepihut.com/blogs/raspberry-pi-tutorials/how-to-change-the-default-account-username-and-password

Install drivers for the audio hat
https://www.waveshare.com/w/upload/5/54/WM8960_Audio_HAT_User_Manual_EN.pdf

Reading GPIO in python
https://gpiozero.readthedocs.io/en/stable/

Python aslaaudio installation and documentation
https://larsimmisch.github.io/pyalsaaudio/

MY CODE
 On each divide you will need at least two python programs running at once. One to receive audio broadcasts from other devices and one to broadcast

Broadcast Code

```python
#!/usr/bin/env python3
import alsaaudio
import socket
import struct
import sys
from gpiozero import Button

button = Button(17)

device = "default"

audioIn = alsaaudio.PCM(alsaaudio.PCM_CAPTURE, alsaaudio.PCM_NORMAL,
device=device, periodsize=1,format=alsaaudio.PCM_FORMAT_S8)

audioOut = alsaaudio.PCM(device=device, periodsize=1,format=alsaaudio.PCM_FORMAT_S8)

broadcastIP = ('192.168.100.255',10000)
recieve_group = ('',10000)

cast = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
cast.setsockopt(socket.SOL_SOCKET, socket.SO_BROADCAST, 1)
i = 0


while (True):
  if (button.is_pressed):
    length, data = audioIn.read()
    #audioOut.write(data)
    # Send data to the multicast group
    print(len(data), sys.stderr)
    cast.sendto(data,broadcastIP)
    #cast.sendto(data,('192.168.100.1',10000))
```

Receive Code

```python
#!/usr/bin/env python3
import alsaaudio
import socket
import struct
import sys
from gpiozero import Button

device = "default"
```

```python
audioOut = alsaaudio.PCM(device=device,periodsize=1,format=alsaaudio.PCM_FORMAT_S8)

receive_group = ('',10000)
receive = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)

receive.bind(receive_group)

s =socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
s.setsockopt(socket.SOL_SOCKET, socket.SO_BROADCAST,1)
s.connect(('<broadcast>',0))
localAddress = s.getsockname()[0]
s.close()

while(True):
  data, address = receive.recvfrom(65536)
  print (localAddress, address)
  if(address[0] != localAddress):
    audioOut.write(data)
    print(len(data),sys.stderr)
```