

# ODT PROJECT

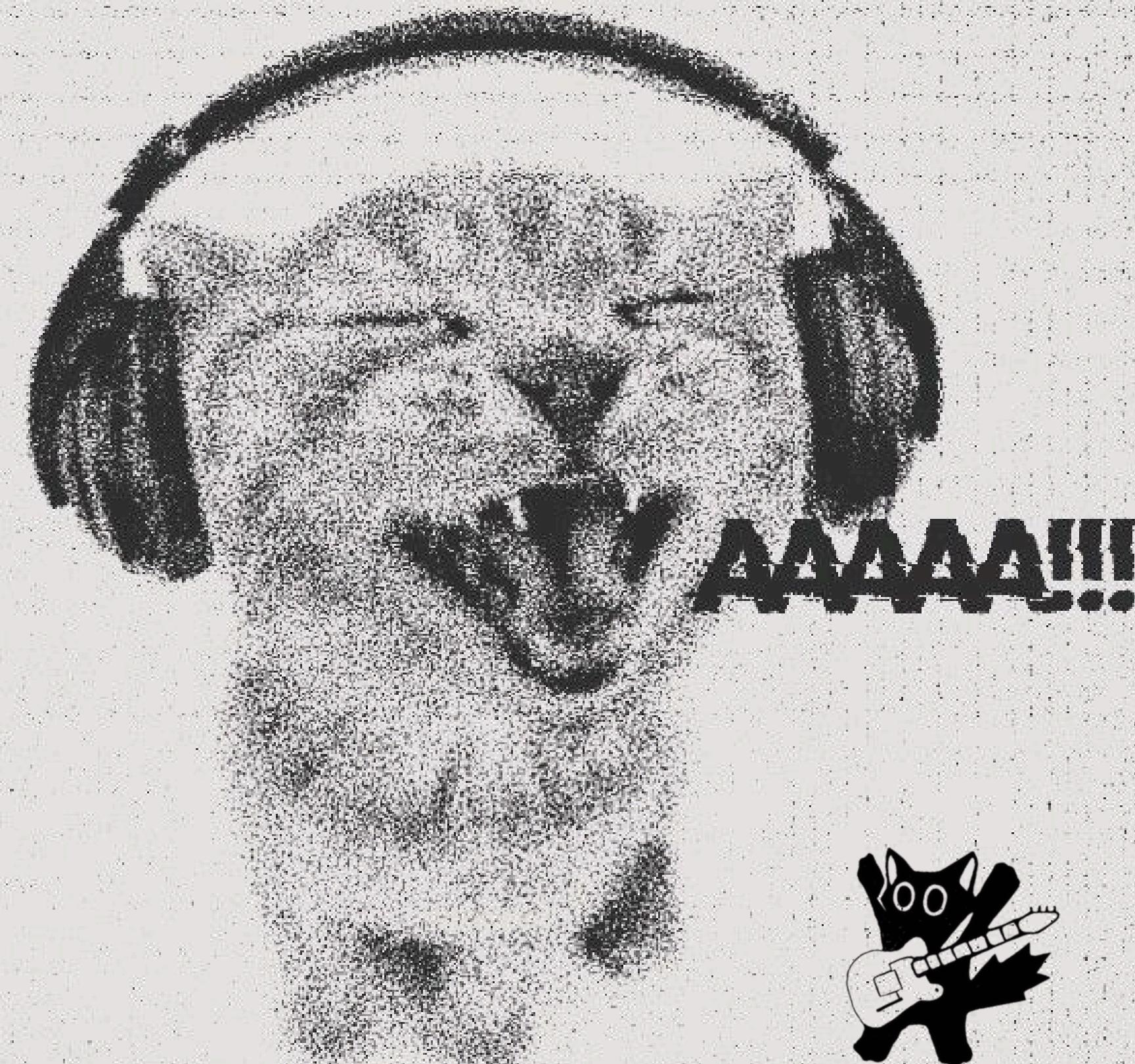
by punya  
& zalak

Start

>>> %Run -c \$EDITOR\_CONTENT

odt project: punya & zalak

>>>



# CONCEPT NOTE:

## DIGITAL GUITAR

- Made a fretboard: used push buttons to play the chords.
- Connected these push buttons to the ESP32 using jumper wires.
- Using the ultrasound sensor, we can play tunes on the “guitar” by shifting our hand in front of it and controlling it using distance.



# OUR CODE:

```
from machine import Pin, PWM, time_pulse_us
import time

audio = PWM(Pin(25))
audio.duty(0)

s1= Pin(15,Pin.IN,Pin.PULL_UP)
s2= Pin(4,Pin.IN,Pin.PULL_UP)
s3= Pin(14,Pin.IN,Pin.PULL_UP)
s4= Pin(27,Pin.IN,Pin.PULL_UP)
s5= Pin(26,Pin.IN,Pin.PULL_UP)
s6= Pin(25,Pin.IN,Pin.PULL_UP)

trig= Pin(5,Pin.OUT)
echo= Pin(19,Pin.IN)

D_chord = [294, 370, 440]
A_chord = [220, 277, 330]
G_chord = [196, 247, 294]
```

```
def get_distance():
    trig.off()
    time.sleep_us(2)
    trig.on()
    time.sleep_us(10)
    trig.off()

    duration = time_pulse_us(echo, 1, 30000)

    if duration < 0:
        return None

    distance = duration / 58
    return distance

def play_down(chord):
    for note in chord:
        audio.freq(note)
        audio.duty(512)
```

**“THIS IS  
NOT  
AI”**

## OUR CODE:

```
time.sleep(0.07)
speaker.duty(0)

def play_up(chord):
    for note in reversed(chord):
        audio.freq(note)
        audio.duty(512)
        time.sleep(0.07)
    audio.duty(0)

last_distance = get_distance()
while True:
    c1= s1.value()
    c2= s2.value()
    c3= s3.value()
    c4= s4.value()
    c5= s5.value()
    c6= s6.value()

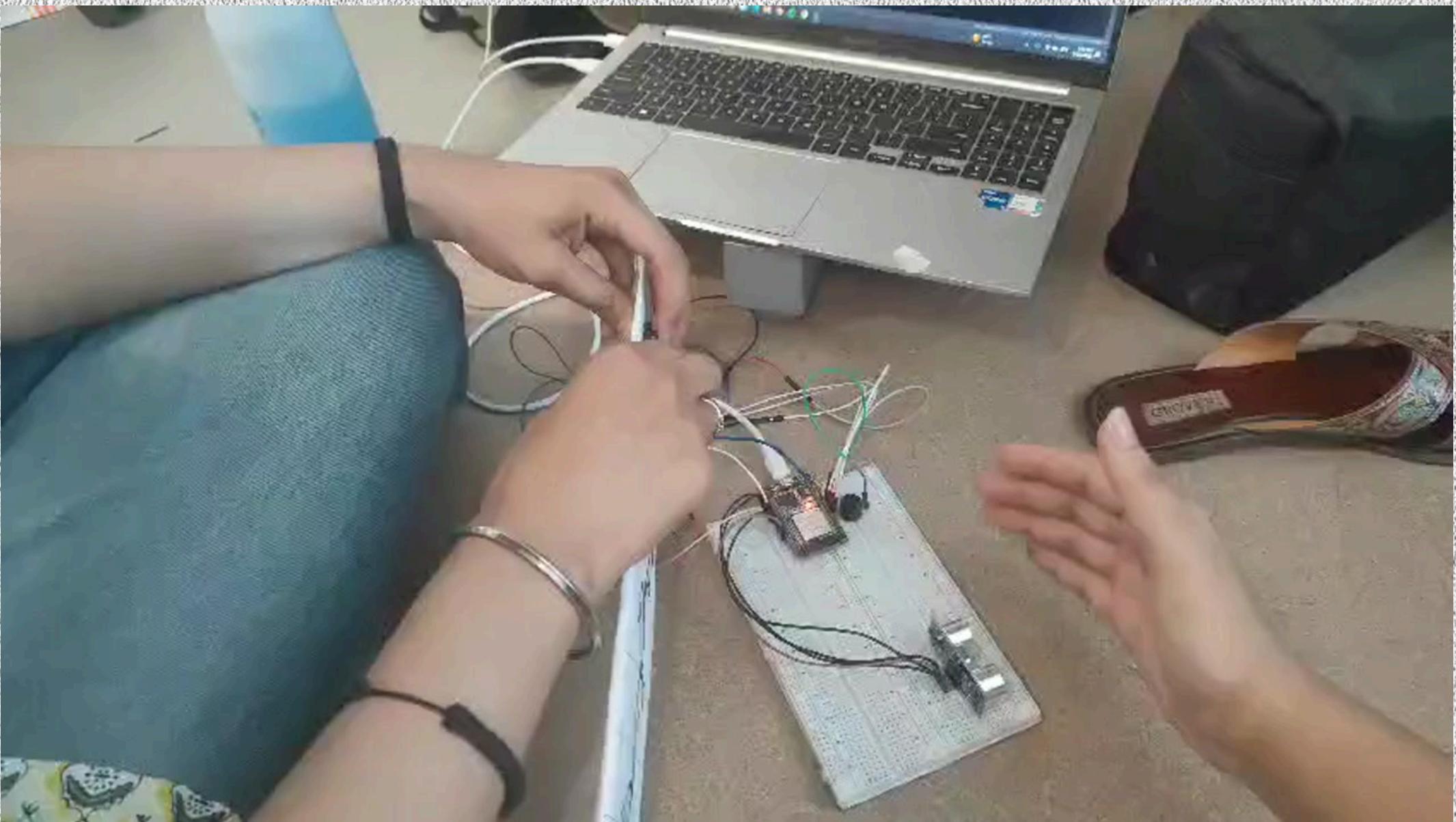
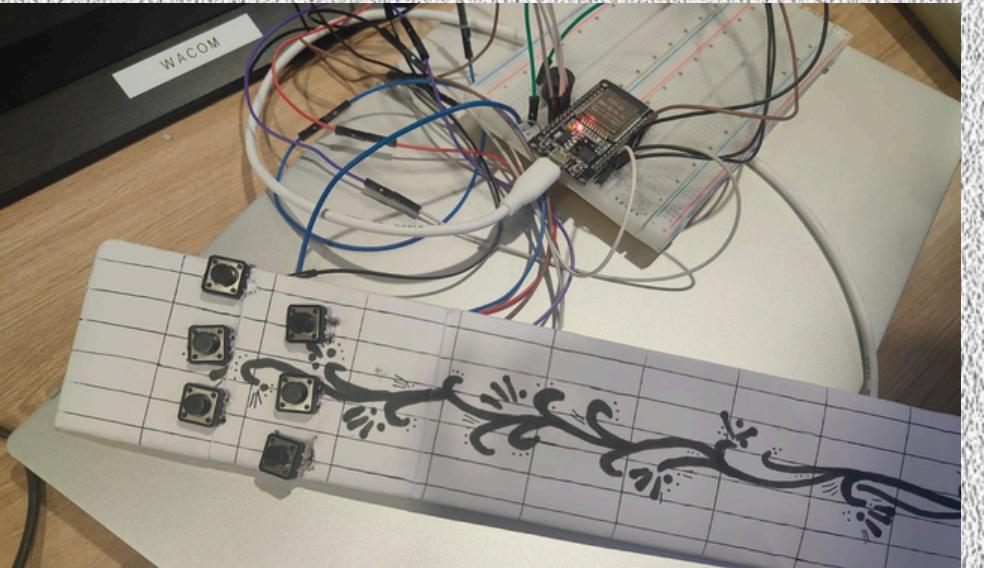
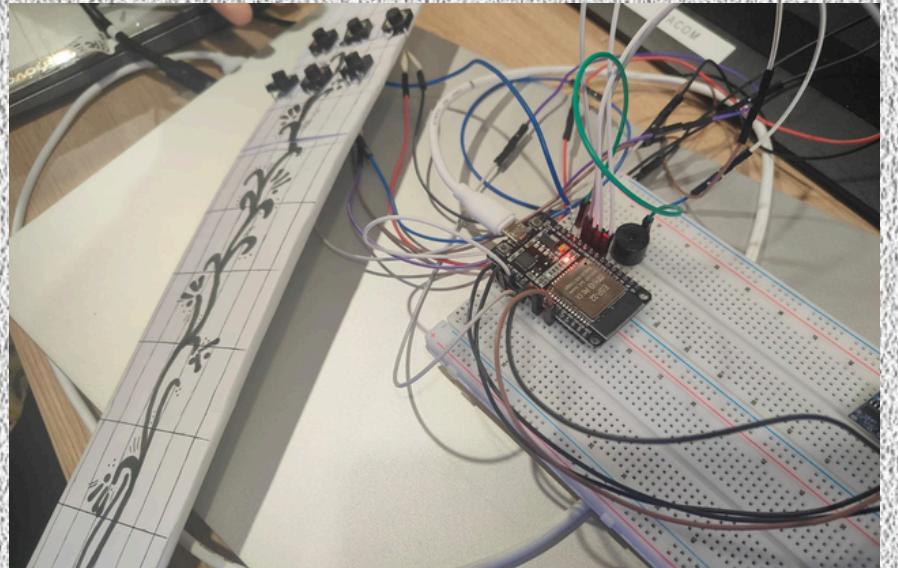
        if c1==0 and c2==0 and c3==0:
            current_chord= D_chord
        elif c1==0 and c5==0 and c6==0:
            current_chord= G_chord
        elif c5==0 and c4==0 and c3==0:
            current_chord= A_chord
        else:
            current_chord= None

    distance = get_distance()
    change = distance - last_distance
    if current_chord != None:
        if change < -6:
            play_down(current_chord)
        if change > 6:
            play_up(current_chord)
    last_distance = distance
    time.sleep(0.05)
```



**VIDEO:**

A video of the working model:





thank u  
bye.

