

Nume student:

Duduman Bianca-Elena

Grupa:

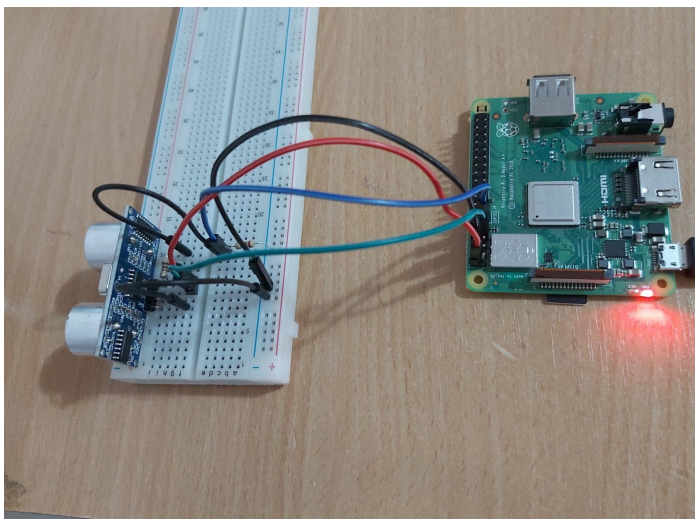
1305B

Adresa de email:

duduman.bianca.elena@gmail.com

Poza:**Numele proiectului:** Provisions checker (Verificator de provizii)

Un sistem ce foloseste un senzor de distanta pentru a verifica daca mai exista produse intr-un dulap. Daca distanta trece de distanta maxim(nu mai este nimic in dulap), este trimis un email de atentionare.

**Story:**

Proiectul a fost inspirat de proviziile create de oamenii ce au fost nevoiti sa intre in izolare din cauza pandemiei de Covid-19.

Cum functioneaza?

- Folosind senzorul de distanta, la fiecare ~4 sec se masoara distanta pana la primul obiect si se compara cu distanta retinuta.
- Daca se iau toate produsele din fata senzorului, sistemul va trimite un email cu textul “!!! Nu mai sunt provizii !!! E timpul sa mergem la cumparaturi!”

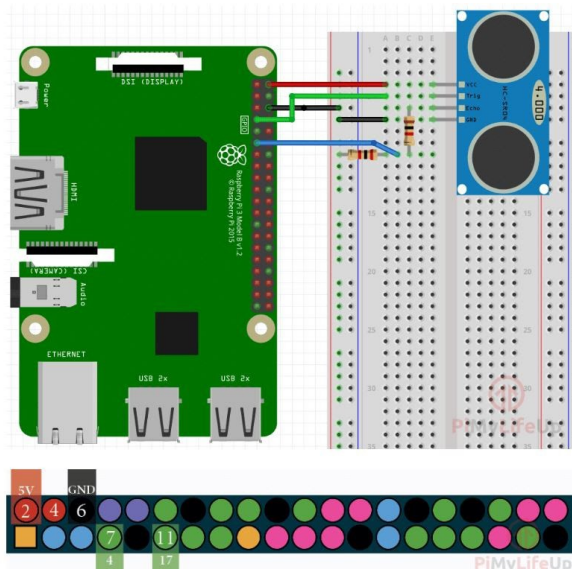
Componente hardware:

- Raspberry Pi 3 B+
- Senzor de distanta HC-SR04
- Breadboard
- Fire Mama-Tata
- Fire Tata-Tata
- Rezistenta 1 k Ω
- Rezistenta 2 k Ω

Aplicatii:

- Raspberry Pi Raspbian

Schema:



Note: Pinul Echo al senzorului are un output de 5V, iar pinilor GPIO de pe Raspberry Pi cer un input de 3.3V . De aceea este necesara folosirea unor rezistente pentru a crea un divizor de tensiune. Se pot folosi orice rezistente atat timp cat respecta formula:

$$V_{out} = V_{in} * \frac{Resistor_1}{Resistor_1 + Resistor_2}$$

Cod:

send_email.py

```
import yagmail

import keyring.backend
from keyrings.alt.file import PlaintextKeyring

keyring.set_keyring(PlaintextKeyring())

receiver = "biancadux3+person2@gmail.com"
body = "!!! Nu mai sunt provizii !!! E timpul sa mergem la cumparaturi!"

yag = yagmail.SMTP("biancadux3+person1@gmail.com", "Animale1999")
yag.send(
    to=receiver,
    subject="Alerta RaspberryPI",
    contents=body,
)
```

senzor_distanta.py

```
#!/usr/bin/python
import RPi.GPIO as GPIO
import time

dist_init = 33;

try:
    while 1:
        GPIO.setmode(GPIO.BOARD)

        PIN_TRIGGER = 7
        PIN_ECHO = 11
        pulse_start_time = 0
        pulse_end_time = 0

        GPIO.setup(PIN_TRIGGER, GPIO.OUT)
        GPIO.setup(PIN_ECHO, GPIO.IN)
```

```

GPIO.output(PIN_TRIGGER, GPIO.LOW)
#waiting for sensor to settle
time.sleep(2)

GPIO.output(PIN_TRIGGER, GPIO.HIGH)

time.sleep(0.00001)

GPIO.output(PIN_TRIGGER, GPIO.LOW)

while GPIO.input(PIN_ECHO)==0:
    pulse_start_time = time.time()
while GPIO.input(PIN_ECHO)==1:
    pulse_end_time = time.time()

pulse_duration = pulse_end_time - pulse_start_time
distance = round(pulse_duration * 17150, 2)

print
"Distance:",distance,"cm"

if distance >= dist_init:
    execfile("/home/pi/proiect/send_email.py")
    print "Provisions low. Email has been sent."
    break;
time.sleep(2);
finally:
    GPIO.cleanup()

```

Video: <https://www.youtube.com/watch?v=OsTDb5ueUpw>