

## ДОДАТОК А

### ЛІСТИНГ ПРОГРАМИ

```
import time
import speech_recognition as sr
import os
import socket
import random
import threading
import Adafruit_GPIO.SPI as SPI
import Adafruit_SSD1306
from PIL import Image
from PIL import ImageDraw
from PIL import ImageFont
from pushbullet import Pushbullet
import RPi.GPIO as GPIO
RST = 24
DC = 23
SPI_PORT = 0
SPI_DEVICE = 0
disp = Adafruit_SSD1306.SSD1306_128_32(rst=RST)
disp.begin()
disp.clear()
disp.display()
width = disp.width
height = disp.height
image = Image.new('1', (width, height))
GPIO.setmode(GPIO.BCM)
GPIO.setup(25, GPIO.OUT)
GPIO.setup(8, GPIO.OUT)
GPIO.setup(7, GPIO.OUT)
GPIO.setup(1, GPIO.OUT)
GPIO.output(25, GPIO.LOW)
GPIO.output(8, GPIO.LOW)
GPIO.output(7, GPIO.LOW)
GPIO.output(1, GPIO.LOW)
GPIO.setup(17, GPIO.IN, pull_up_down=GPIO.PUD_UP)
GPIO.setwarnings(False)
GPIO.setup(5, GPIO.OUT)
GPIO.setup(6, GPIO.IN)
GPIO.setup(13, GPIO.OUT)
```

```

GPIO.setup(19, GPIO.IN)
GPIO.output(5, False)
GPIO.output(13, False)
GPIO.setup(26, GPIO.IN)
x = 1
y = 1
r = sr.Recognizer()
def check_internet_connection(host="8.8.8.8", port=53, timeout=3):
    try:
        socket.setdefaulttimeout(timeout)
        socket.socket(socket.AF_INET, socket.SOCK_STREAM).connect((host,
port))
        return True
    except Exception as ex:
        return False
if check_internet_connection():
    print("Интернет соединение установлено.")
    draw = ImageDraw.Draw(image)
    draw.rectangle((0,0,width,height), outline=0, fill=0)
    draw.text((20, 8), 'WI-FI CONNECTED', fill=255)
    disp.image(image)
    disp.display()
else:
    print("Отсутствует интернет соединение.")
    draw = ImageDraw.Draw(image)
    draw.rectangle((0,0,width,height), outline=0, fill=0)
    draw.text((40, 8), 'NO WI-FI', fill=255)
    disp.image(image)
    disp.display()
def pressed_button():
    if GPIO.input(17) == True:
        print ('Button pressed')
        return wait_command()
    else:
        print ('Button don`t pressed')
def listen_for_hello_voxy():
    mic = sr.Microphone()
    with mic as source:
        r.adjust_for_ambient_noise(source)
        draw = ImageDraw.Draw(image)
        print('REC...')
        draw.rectangle((0,0,width,height), outline=0, fill=0)
        draw.ellipse((15, 10 , 55, 30), outline=1, fill=0)
        draw.ellipse((72, 10 , 112, 30), outline=1, fill=0)
        draw.ellipse((25, 15 , 45, 25), outline=1, fill=1)

```

```

draw.ellipse((82, 15 , 102, 25), outline=1, fill=1)
draw.rectangle((15,10,55,18), outline=0, fill=0)
draw.line((15, 18, 55, 18), fill=255)
draw.rectangle((72,10,112,18), outline=0, fill=0)
draw.line((72, 18, 112, 18), fill=255)
draw.rectangle((15,1,55,7), outline=0, fill=1)
draw.rectangle((72,1,112,7), outline=0, fill=1)
disp.image(image)
disp.display()
try:
    audio = r.listen(source, timeout=3) # Adjust the timeout as needed
    text = r.recognize_google(audio)
    print(text)
    if 'hello voxy' in text or 'hello Vox' in text or 'hello boxing' in text or 'hello
voxin' in text or 'hello foxy' in text or 'hello boxer' in text or 'hello voxing' in text or
'hello fox' in text:
        return wait_command()
    else:
        draw = ImageDraw.Draw(image)
        draw.rectangle((15,10,112,30), outline=0, fill=0)
        draw.line((15, 18, 55, 18), fill=255)
        draw.line((72, 18, 112, 18), fill=255)
        disp.image(image)
        disp.display()
        return False
except sr.WaitTimeoutError:
    draw = ImageDraw.Draw(image)
    draw.rectangle((15,10,112,30), outline=0, fill=0)
    draw.line((15, 18, 55, 18), fill=255)
    draw.line((72, 18, 112, 18), fill=255)
    disp.image(image)
    disp.display()
    print("You didn't say anything.")
    return False
except sr.UnknownValueError:
    draw = ImageDraw.Draw(image)
    draw.rectangle((15,10,112,30), outline=0, fill=0)
    draw.line((15, 18, 55, 18), fill=255)
    draw.line((72, 18, 112, 18), fill=255)
    disp.image(image)
    disp.display()
    print("Sorry, I could not understand what you said.")
    return False
except sr.RequestError:
    draw = ImageDraw.Draw(image)

```

```

    draw.rectangle((15,10,112,30), outline=0, fill=0)
    draw.line((15, 18, 55, 18), fill=255)
    draw.line((72, 18, 112, 18), fill=255)
    disp.image(image)
    disp.display()
    print("Internet Connection Lost")
    return check_internet_connection()
def move_robot():
    while x == 1 and y == 1:
        GPIO.output(5, True)
        time.sleep(0.00001)
        GPIO.output(5, False)
        while GPIO.input(6) == 0:
            pulse_start = time.time()
            while GPIO.input(6) == 1:
                pulse_end = time.time()
            pulse_duration = pulse_end - pulse_start
            distance1 = pulse_duration * 17150
            distance1 = round(distance1, 2)
            GPIO.output(13, True)
            time.sleep(0.00001)
            GPIO.output(13, False)
            while GPIO.input(19) == 0:
                pulse_start = time.time()
                while GPIO.input(19) == 1:
                    pulse_end = time.time()
                pulse_duration = pulse_end - pulse_start
                distance2 = pulse_duration * 17150
                distance2 = round(distance2, 2)
            if GPIO.input(26) == 1:
                print('It is edge of surface')
                GPIO.output(25, GPIO.LOW)
                GPIO.output(8, GPIO.HIGH)
                GPIO.output(7, GPIO.LOW)
                GPIO.output(1, GPIO.HIGH)
                time.sleep(0.5)
                GPIO.output(25, GPIO.LOW)
                GPIO.output(8, GPIO.LOW)
                GPIO.output(7, GPIO.LOW)
                GPIO.output(1, GPIO.LOW)
                time.sleep(5)
            elif distance1 <= 10 and distance2 <= 10:
                print('Object center')
                random_number0 = random.randint(1, 2)
                if random_number0 == 1:

```

```
GPIO.output(25, GPIO.HIGH)
GPIO.output(8, GPIO.LOW)
GPIO.output(7, GPIO.LOW)
GPIO.output(1, GPIO.HIGH)
time.sleep(0.5)
GPIO.output(25, GPIO.LOW)
GPIO.output(8, GPIO.LOW)
GPIO.output(7, GPIO.LOW)
GPIO.output(1, GPIO.LOW)
time.sleep(5)
elif random_number0 == 2:
    GPIO.output(25, GPIO.LOW)
    GPIO.output(8, GPIO.HIGH)
    GPIO.output(7, GPIO.HIGH)
    GPIO.output(1, GPIO.LOW)
    time.sleep(0.5)
    GPIO.output(25, GPIO.LOW)
    GPIO.output(8, GPIO.LOW)
    GPIO.output(7, GPIO.LOW)
    GPIO.output(1, GPIO.LOW)
    time.sleep(5)
else:
    GPIO.output(25, GPIO.LOW)
    GPIO.output(8, GPIO.HIGH)
    GPIO.output(7, GPIO.HIGH)
    GPIO.output(1, GPIO.LOW)
    time.sleep(0.5)
    GPIO.output(25, GPIO.LOW)
    GPIO.output(8, GPIO.LOW)
    GPIO.output(7, GPIO.LOW)
    GPIO.output(1, GPIO.LOW)
    time.sleep(5)
elif distance1 < distance2 and distance1 <= 10:
    print('Object right')
    GPIO.output(25, GPIO.HIGH)
    GPIO.output(8, GPIO.LOW)
    GPIO.output(7, GPIO.LOW)
    GPIO.output(1, GPIO.HIGH)
    time.sleep(0.5)
    GPIO.output(25, GPIO.LOW)
    GPIO.output(8, GPIO.LOW)
    GPIO.output(7, GPIO.LOW)
    GPIO.output(1, GPIO.LOW)
    time.sleep(5)
elif distance1 > distance2 and distance2 <= 10:
```

```
print('Object left')
GPIO.output(25, GPIO.LOW)
GPIO.output(8, GPIO.HIGH)
GPIO.output(7, GPIO.HIGH)
GPIO.output(1, GPIO.LOW)
time.sleep(0.5)
GPIO.output(25, GPIO.LOW)
GPIO.output(8, GPIO.LOW)
GPIO.output(7, GPIO.LOW)
GPIO.output(1, GPIO.LOW)
time.sleep(5)
else:
    print('No object, random move')
    random_number = random.randint(1, 5)
    if random_number == 1:
        GPIO.output(25, GPIO.HIGH)
        GPIO.output(8, GPIO.LOW)
        GPIO.output(7, GPIO.HIGH)
        GPIO.output(1, GPIO.LOW)
        time.sleep(0.5)
        GPIO.output(25, GPIO.LOW)
        GPIO.output(8, GPIO.LOW)
        GPIO.output(7, GPIO.LOW)
        GPIO.output(1, GPIO.LOW)
        time.sleep(5)
    elif random_number == 2:
        GPIO.output(25, GPIO.HIGH)
        GPIO.output(8, GPIO.LOW)
        GPIO.output(7, GPIO.LOW)
        GPIO.output(1, GPIO.HIGH)
        time.sleep(0.5)
        GPIO.output(25, GPIO.LOW)
        GPIO.output(8, GPIO.LOW)
        GPIO.output(7, GPIO.LOW)
        GPIO.output(1, GPIO.LOW)
        time.sleep(5)
    elif random_number == 3:
        GPIO.output(25, GPIO.LOW)
        GPIO.output(8, GPIO.HIGH)
        GPIO.output(7, GPIO.HIGH)
        GPIO.output(1, GPIO.LOW)
        time.sleep(0.5)
        GPIO.output(25, GPIO.LOW)
        GPIO.output(8, GPIO.LOW)
        GPIO.output(7, GPIO.LOW)
```

```

        GPIO.output(1, GPIO.LOW)
        time.sleep(5)
    elif random_number == 4:
        GPIO.output(25, GPIO.LOW)
        GPIO.output(8, GPIO.HIGH)
        GPIO.output(7, GPIO.LOW)
        GPIO.output(1, GPIO.HIGH)
        time.sleep(0.5)
        GPIO.output(25, GPIO.LOW)
        GPIO.output(8, GPIO.LOW)
        GPIO.output(7, GPIO.LOW)
        GPIO.output(1, GPIO.LOW)
        time.sleep(5)
    else:
        time.sleep(5)
def wait_command():
    x = 0
    mic = sr.Microphone()
    with mic as source:
        r.adjust_for_ambient_noise(source)
        draw = ImageDraw.Draw(image)
        draw.rectangle((0,0,width,height), outline=0, fill=0)
        draw.ellipse((15, 10 , 55, 30), outline=1, fill=0)
        draw.ellipse((72, 10 , 112, 30), outline=1, fill=0)
        draw.ellipse((25, 15 , 45, 25), outline=1, fill=1)
        draw.ellipse((82, 15 , 102, 25), outline=1, fill=1)
        draw.rectangle((15,1,55,7), outline=0, fill=1)
        draw.rectangle((72,1,112,7), outline=0, fill=1)
        disp.image(image)
        disp.display()
        print('REC 2...')
    try:
        audio = r.listen(source, timeout=5) # Adjust the timeout as needed
        text = r.recognize_google(audio)
        print(text)
        if 'save event' in text or 'add event' in text or 'create event' in text:
            return save_event()
        elif 'check event' in text or 'show event' in text:
            return check_event()
        elif 'delete event' in text:
            return delete_event()
        elif 'stop' in text:
            y = 0
            return listen_for_hello_voxy()
        elif 'move' in text:

```

```

        y = 1
        return listen_for_hello_voxy()
    else:
        print('Error. unknow command')
        return listen_for_hello_voxy()
except sr.WaitTimeoutError:
    draw = ImageDraw.Draw(image)
    draw.rectangle((15,10,112,30), outline=0, fill=0)
    draw.line((15, 18, 55, 18), fill=255)
    draw.line((72, 18, 112, 18), fill=255)
    disp.image(image)
    disp.display()
    print("You didn't say anything.")
    return listen_for_hello_voxy()
except sr.UnknownValueError:
    draw = ImageDraw.Draw(image)
    draw.rectangle((15,10,112,30), outline=0, fill=0)
    draw.line((15, 18, 55, 18), fill=255)
    draw.line((72, 18, 112, 18), fill=255)
    disp.image(image)
    disp.display()
    print("Sorry, I could not understand what you said.")
    return listen_for_hello_voxy()
except sr.RequestError:
    draw = ImageDraw.Draw(image)
    draw.rectangle((15,10,112,30), outline=0, fill=0)
    draw.line((15, 18, 55, 18), fill=255)
    draw.line((72, 18, 112, 18), fill=255)
    disp.image(image)
    disp.display()
    print("Internet Connection Lost")
    return check_internet_connection()
def save_event():
    mic = sr.Microphone()
    with mic as source:
        r.adjust_for_ambient_noise(source)
        print('Save event...')
        draw = ImageDraw.Draw(image)
        draw.rectangle((15,10,112,30), outline=0, fill=0)
        draw.rectangle((15,1,55,7), outline=0, fill=1)
        draw.rectangle((72,1,112,7), outline=0, fill=1)
        draw.line((15, 15, 20, 10), fill=1)
        draw.line((20, 10, 30, 10), fill=1)
        draw.line((30, 10, 35, 15), fill=1)
        draw.line((27, 20, 35, 15), fill=1)

```

```

draw.line((27, 20, 27, 25), fill=1)
draw.line((92, 15, 97, 10), fill=1)
draw.line((97, 10, 107, 10), fill=1)
draw.line((107, 10, 112, 15), fill=1)
draw.line((104, 20, 112, 15), fill=1)
draw.line((104, 20, 104, 25), fill=1)
disp.image(image)
disp.display()
try:
    audio = r.listen(source, timeout=3) # Adjust the timeout as needed
    text = r.recognize_google(audio)
    event = text
    with open("events.txt", "a") as file:
        file.write(f"{event}\n")
        print("loading")
        sleep(5)
        print("Event saved")
        pb = Pushbullet("o.Rml8Oc48BcbdL6flVkvIXePM2qVnLgAK")
        print(pb.devices)
        dev = pb.get_device('Xiaomi Mi 9T')
        push = dev.push_note("Event saved")
        return listen_for_hello_voxy()
except sr.WaitTimeoutError:
    draw = ImageDraw.Draw(image)
    draw.rectangle((15,10,112,30), outline=0, fill=0)
    draw.line((15, 18, 55, 18), fill=255)
    draw.line((72, 18, 112, 18), fill=255)
    disp.image(image)
    disp.display()
    print("You didn't say anything.")
    return
except sr.UnknownValueError:
    draw = ImageDraw.Draw(image)
    draw.rectangle((15,10,112,30), outline=0, fill=0)
    draw.line((15, 18, 55, 18), fill=255)
    draw.line((72, 18, 112, 18), fill=255)
    disp.image(image)
    disp.display()
    print("Извините, не могу распознать речь. Пожалуйста, повторите.")
except sr.RequestError as e:
    draw = ImageDraw.Draw(image)
    draw.rectangle((15,10,112,30), outline=0, fill=0)
    draw.line((15, 18, 55, 18), fill=255)
    draw.line((72, 18, 112, 18), fill=255)
    disp.image(image)

```

```

disp.display()
print(f'Ошибка сервиса распознавания речи; {e}')
except sr.RequestError:
    draw = ImageDraw.Draw(image)
    draw.rectangle((15,10,112,30), outline=0, fill=0)
    draw.line((15, 18, 55, 18), fill=255)
    draw.line((72, 18, 112, 18), fill=255)
    disp.image(image)
    disp.display()
    print("Internet Connection Lost")
    return check_internet_connection()
def check_event():
    mic = sr.Microphone()
    events_found = [] # To store all events found
    with mic as source:
        r.adjust_for_ambient_noise(source)
        print('Check event...')
        try:
            audio = r.listen(source, timeout=3) # Adjust the timeout as needed
            text = r.recognize_google(audio)
            event = text
            draw = ImageDraw.Draw(image)
            draw.rectangle((15,10,112,30), outline=0, fill=0)
            draw.rectangle((15,1,55,7), outline=0, fill=1)
            draw.rectangle((72,1,112,7), outline=0, fill=1)
            draw.line((15, 15, 20, 10), fill=1)
            draw.line((20, 10, 30, 10), fill=1)
            draw.line((30, 10, 35, 15), fill=1)
            draw.line((27, 20, 35, 15), fill=1)
            draw.line((27, 20, 27, 25), fill=1)
            draw.line((92, 15, 97, 10), fill=1)
            draw.line((97, 10, 107, 10), fill=1)
            draw.line((107, 10, 112, 15), fill=1)
            draw.line((104, 20, 112, 15), fill=1)
            draw.line((104, 20, 104, 25), fill=1)
            disp.image(image)
            disp.display()

            with open("events.txt", "r") as file:
                for line in file:
                    if f"{event}" in line:
                        events_found.append(line.strip()) # Add the event to the list
        if events_found:
            draw = ImageDraw.Draw(image)
            draw.rectangle((0,0,width,height), outline=0, fill=0)

```

```

draw.text((40, 8), 'EVENT FOUND', fill=255)
disp.image(image)
disp.display()
pb = Pushbullet("o.Rml8Oc48BcbdL6flVkvIXePM2qVnLgAK")
print(pb.devices)
dev = pb.get_device('Xiaomi Mi 9T')

for event in events_found: # Push each found event
    push = dev.push_note("Event Notification", event)
    print(event)

return listen_for_hello_voxy() # Return after processing all events
else:
    draw = ImageDraw.Draw(image)
    draw.rectangle((0,0,width,height), outline=0, fill=0)
    draw.text((40, 8), 'NO EVENT', fill=255)
    disp.image(image)
    disp.display()
    print('No event')
    return listen_for_hello_voxy()
except sr.WaitTimeoutError:
    print("You didn't say anything.")
    draw = ImageDraw.Draw(image)
    draw.rectangle((15,10,112,30), outline=0, fill=0)
    draw.line((15, 18, 55, 18), fill=255)
    draw.line((72, 18, 112, 18), fill=255)
    disp.image(image)
    disp.display()
except sr.UnknownValueError:
    draw = ImageDraw.Draw(image)
    draw.rectangle((15,10,112,30), outline=0, fill=0)
    draw.line((15, 18, 55, 18), fill=255)
    draw.line((72, 18, 112, 18), fill=255)
    disp.image(image)
    disp.display()
    print("Извините, не могу распознать речь. Пожалуйста, повторите.")
except sr.RequestError as e:
    draw = ImageDraw.Draw(image)
    draw.rectangle((15,10,112,30), outline=0, fill=0)
    draw.line((15, 18, 55, 18), fill=255)
    draw.line((72, 18, 112, 18), fill=255)
    disp.image(image)
    disp.display()
    print(f"Ошибка сервиса распознавания речи; {e}")
except sr.RequestError:

```

```

draw = ImageDraw.Draw(image)
draw.rectangle((15,10,112,30), outline=0, fill=0)
draw.line((15, 18, 55, 18), fill=255)
draw.line((72, 18, 112, 18), fill=255)
disp.image(image)
disp.display()
print("Internet Connection Lost")
return check_internet_connection()
def delete_event():
    r = sr.Recognizer()
    mic = sr.Microphone()
    with mic as source:
        r.adjust_for_ambient_noise(source)
        print('Delete event...')
        try:
            audio = r.listen(source, timeout=3) # Adjust the timeout as needed
            text = r.recognize_google(audio)
            print(text)
            event = text.strip()
            draw = ImageDraw.Draw(image)
            draw.rectangle((15,10,112,30), outline=0, fill=0)
            draw.rectangle((15,1,55,7), outline=0, fill=1)
            draw.rectangle((72,1,112,7), outline=0, fill=1)
            draw.line((15, 15, 20, 10), fill=1)
            draw.line((20, 10, 30, 10), fill=1)
            draw.line((30, 10, 35, 15), fill=1)
            draw.line((27, 20, 35, 15), fill=1)
            draw.line((27, 20, 27, 25), fill=1)
            draw.line((92, 15, 97, 10), fill=1)
            draw.line((97, 10, 107, 10), fill=1)
            draw.line((107, 10, 112, 15), fill=1)
            draw.line((104, 20, 112, 15), fill=1)
            draw.line((104, 20, 104, 25), fill=1)
            disp.image(image)
            disp.display()
            with open("events.txt", "r") as file:
                lines = file.readlines()
            with open("events.txt", "w") as file:
                for line in lines:
                    if event in line:
                        draw = ImageDraw.Draw(image)
                        draw.rectangle((0,0,width,height), outline=0, fill=0)
                        draw.text((40, 8),'SURE?', fill=255)
                        disp.image(image)
                        disp.display()

```

```

print("Are you sure?")
confirm_audio = r.listen(source, timeout=5)
confirm_text = r.recognize_google(confirm_audio).lower()
if "yes" in confirm_text:
    draw = ImageDraw.Draw(image)
    draw.rectangle((0,0,width,height), outline=0, fill=0)
    draw.text((40, 8),'COMPLETE', fill=255)
    disp.image(image)
    disp.display()
    print("complete")
    pb = Pushbullet("o.Rml8Oc48BcbdL6flVkvIXePM2qVnLgAK")
    print(pb.devices)
    dev = pb.get_device('Xiaomi Mi 9T')
    push = dev.push_note("Event deleted")
    return listen_for_hello_voxy()
elif "no" in confirm_text or "cancel" in confirm_text:
    draw = ImageDraw.Draw(image)
    draw.rectangle((0,0,width,height), outline=0, fill=0)
    draw.text((40, 8),'CANCELED', fill=255)
    disp.image(image)
    disp.display()
    print("complete")
    print("process cancelled")
    file.write(line)
    return listen_for_hello_voxy()# Write the line back to the file
else:
    file.write(line)
    return listen_for_hello_voxy()
except sr.WaitTimeoutError:
    print("You didn't say anything.")
    draw = ImageDraw.Draw(image)
    draw.rectangle((15,10,112,30), outline=0, fill=0)
    draw.line((15, 18, 55, 18), fill=255)
    draw.line((72, 18, 112, 18), fill=255)
    disp.image(image)
    disp.display()
except sr.UnknownValueError:
    print("Извините, не могу распознать речь. Пожалуйста, повторите.")
    draw = ImageDraw.Draw(image)
    draw.rectangle((15,10,112,30), outline=0, fill=0)
    draw.line((15, 18, 55, 18), fill=255)
    draw.line((72, 18, 112, 18), fill=255)
    disp.image(image)
    disp.display()
except sr.RequestError as e:

```

```

draw = ImageDraw.Draw(image)
draw.rectangle((15,10,112,30), outline=0, fill=0)
draw.line((15, 18, 55, 18), fill=255)
draw.line((72, 18, 112, 18), fill=255)
disp.image(image)
disp.display()
print(f"Ошибка сервиса распознавания речи; {e}")
except sr.RequestError:
    draw = ImageDraw.Draw(image)
    draw.rectangle((15,10,112,30), outline=0, fill=0)
    draw.line((15, 18, 55, 18), fill=255)
    draw.line((72, 18, 112, 18), fill=255)
    disp.image(image)
    disp.display()
    print("Internet Connection Lost")
    return check_internet_connection()
listen_thread = threading.Thread(target=move_robot)
listen_thread.start()
while True:
    listen_thread.start()
    if check_internet_connection():
        while True:
            if pressed_button() or listen_for_hello_voxy():
                while True:
                    if wait_command():
                        while True:
                            if save_event() or check_event() or delete_event():
                                continue

```

**ДОДАТОК В**  
**ДЕМОНСТРАЦІЙНИЙ МАТЕРІАЛ У ВИГЛЯДІ**  
**ПРЕЗЕНТАЦІЙ**

