A5.2 Learning Activity

Sensor system and actuation of the color of an object, and visual interface

Development

1. Use the following list of materials to prepare the activity

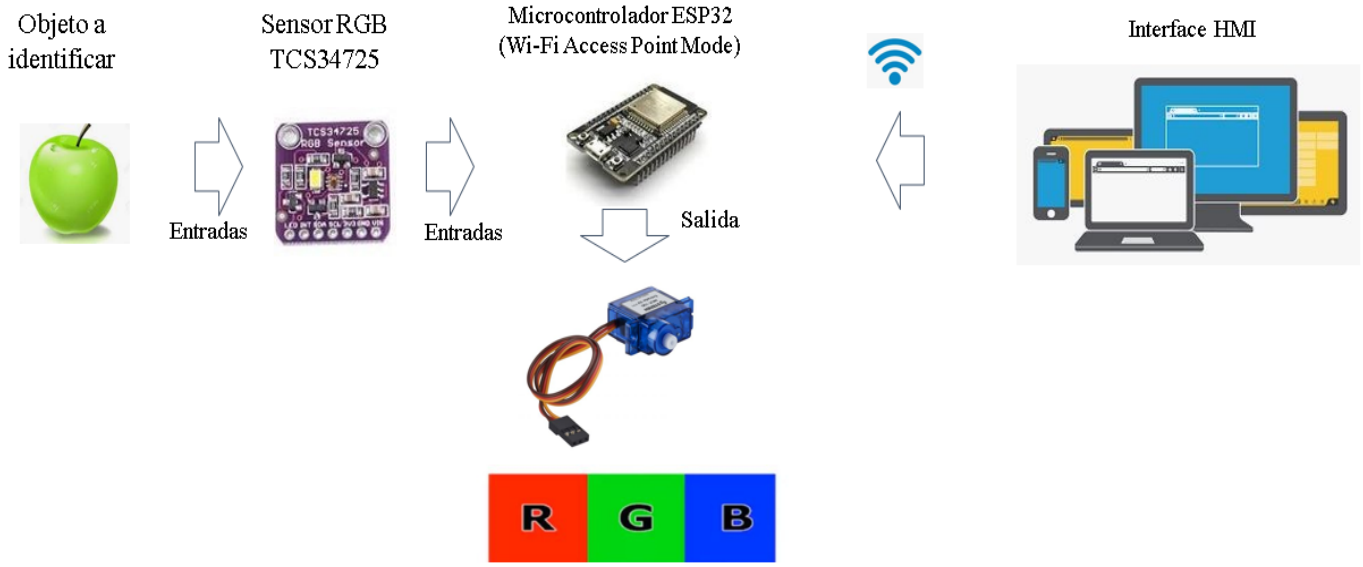
Cantidad	Descripción	Link
1	Sensor RGB TCS34725	Adafruit
1	Servomotor SG90	SanDoRobotics
1	Fuente de voltaje de 5V	UElectronic
1	NodeMCU ESP32	Naylamp Mechatronics
1	BreadBoard	Learn Sparkfun
1	Jumpers M/M	Lozurytech

2. Based on the images shown in **Figure 1**, make a system capable of meeting the following conditions

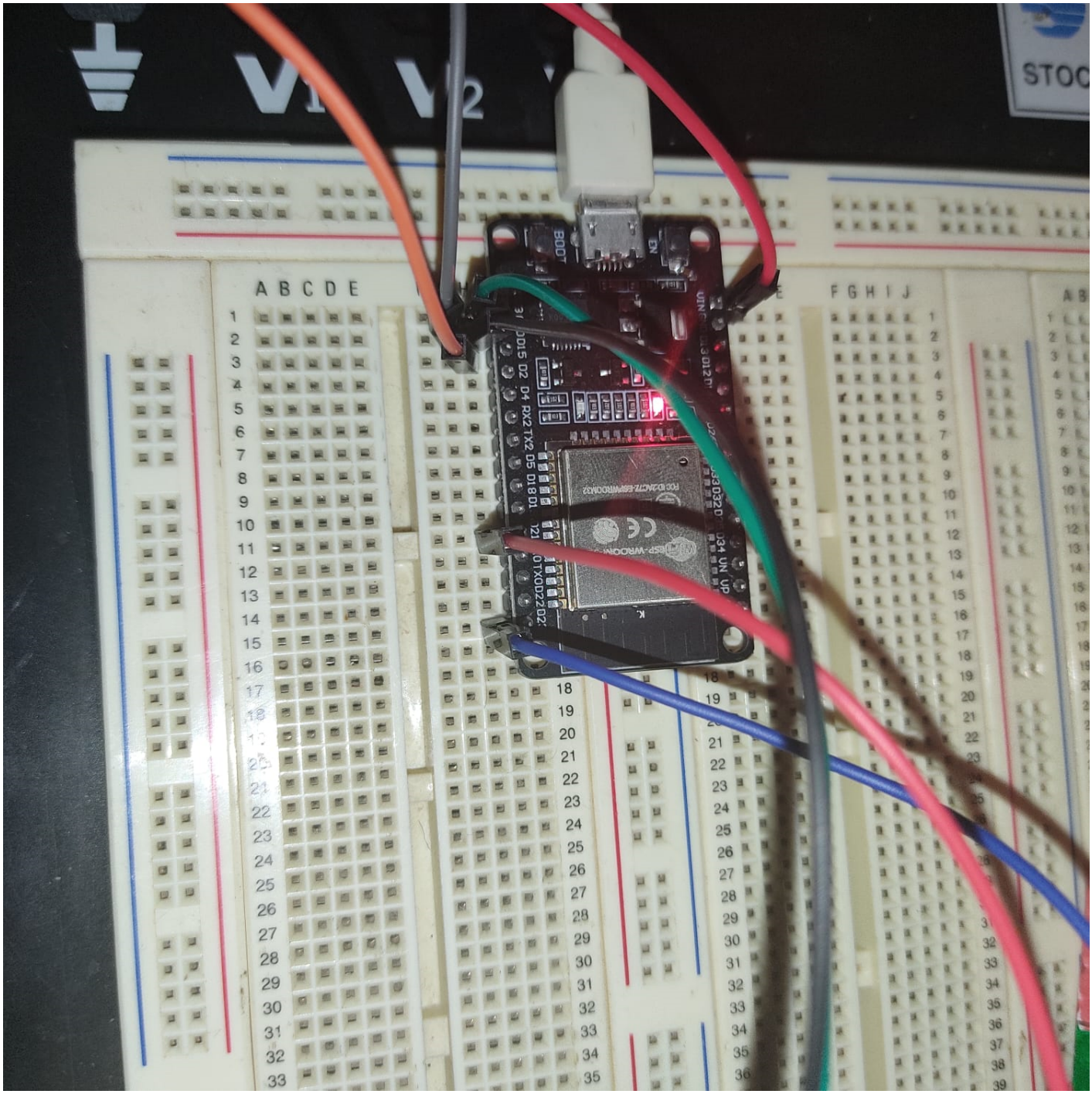
The first phase of the activity will consist of, when placing an object in front of the RGB sensor, it should identify what color it has (it is advisable to use Red, Green, and Blue objects for greater precision), which should be displayed on an interface visual what color was detected.

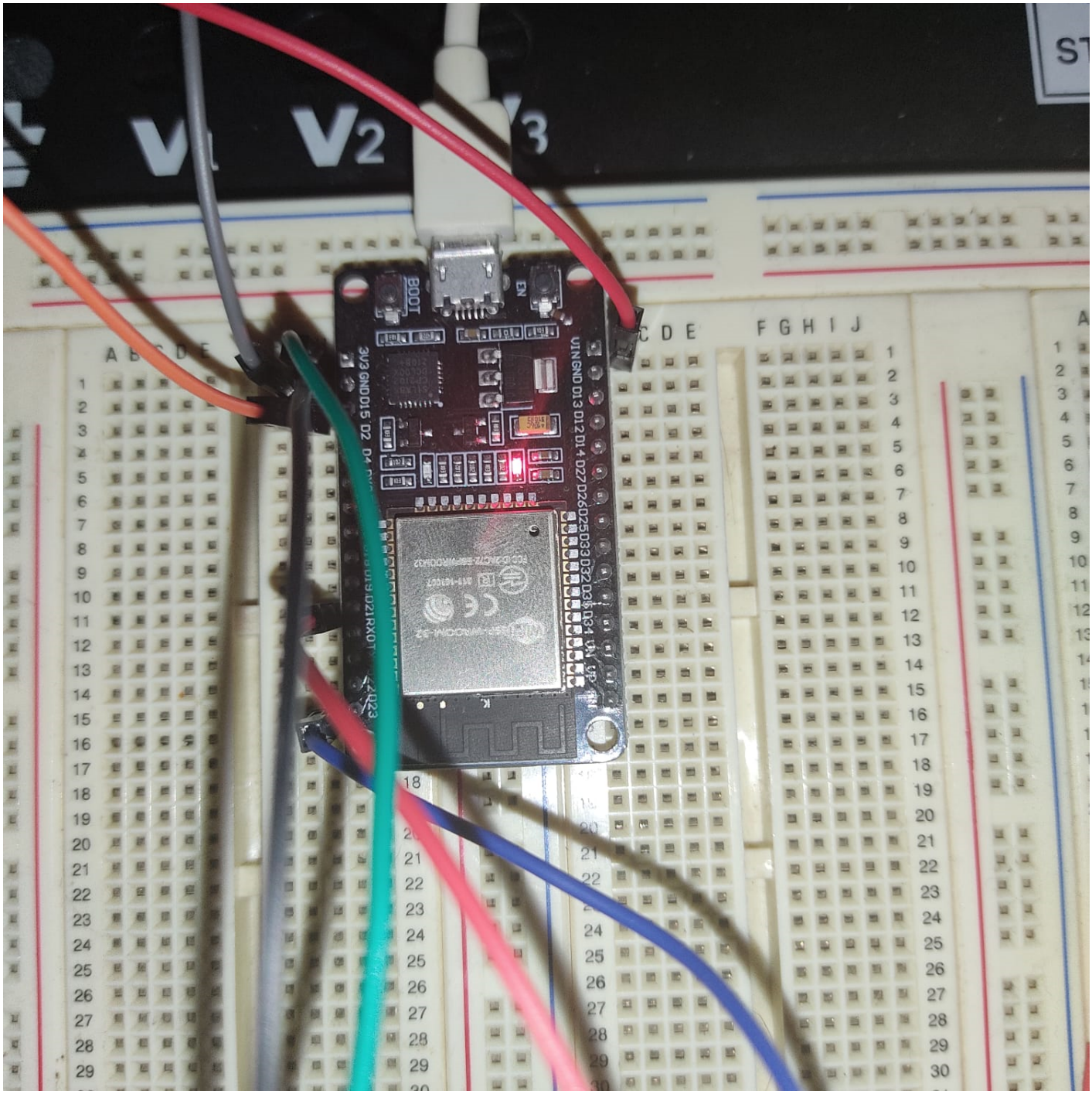
- The second phase will consist of adding an actuator and a color identifier supported by a color band as shown in image 1; When knowing the color of the object, the actuator should point to the color that is being detected.

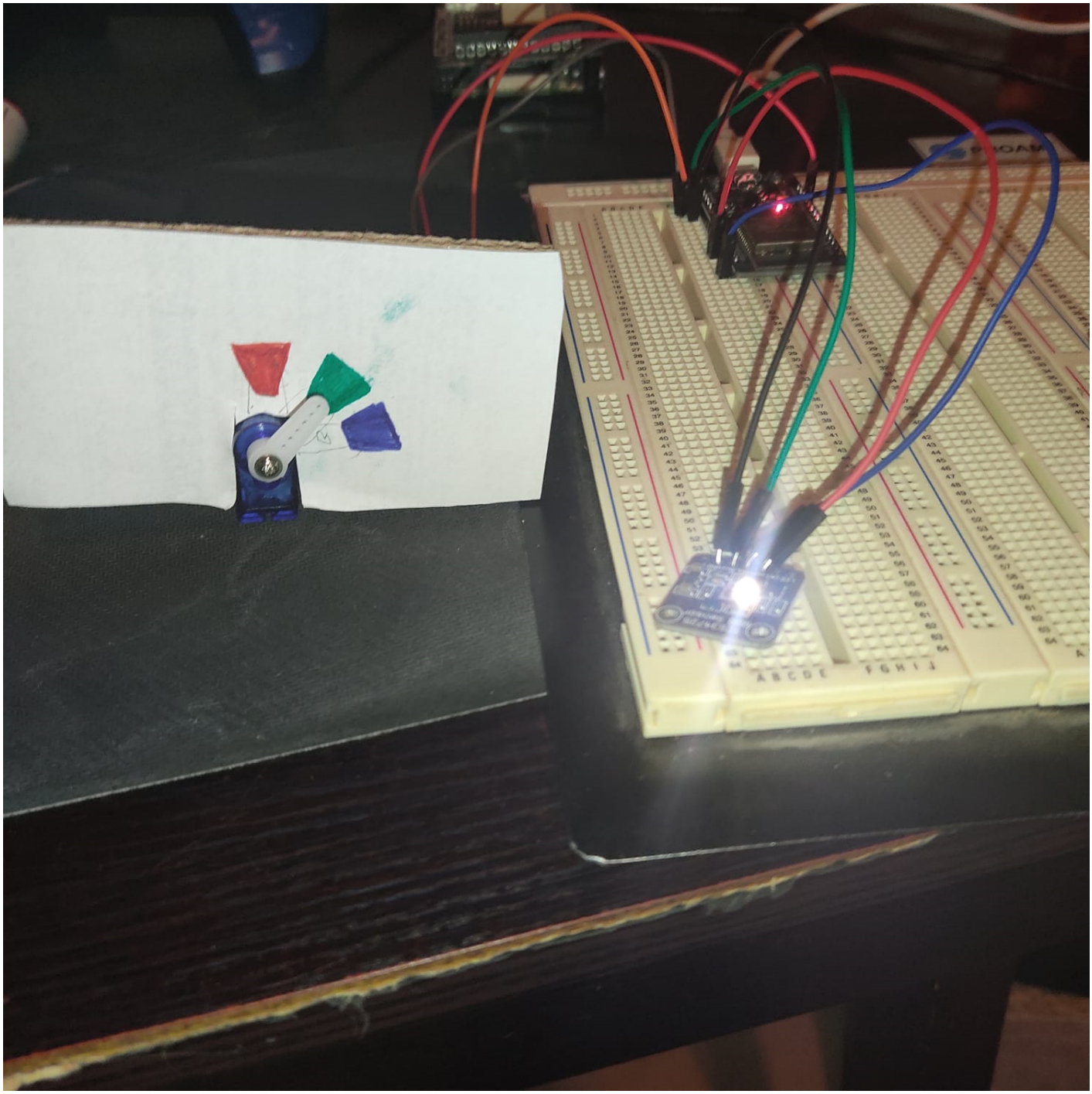
Figura 1 Circuito ESP32 Sensor de color y Servomotor



3. Place the image of the assembled circuit here


A B C D E
F G H I J
BOOT
EN
D15 D2 D4 RX2 TX2 D5 D18 D19
D34 VN VP
D1 TXD0 D2


V2
BOOT
EN
VIN GND D13 D12 D14 D27 D26 D25 D33 D32 D35 D34 VN VP
3V3 GND D15 D2 D4
D21 RX0 TX0



4. Place in this place the program created within the Arduino environment

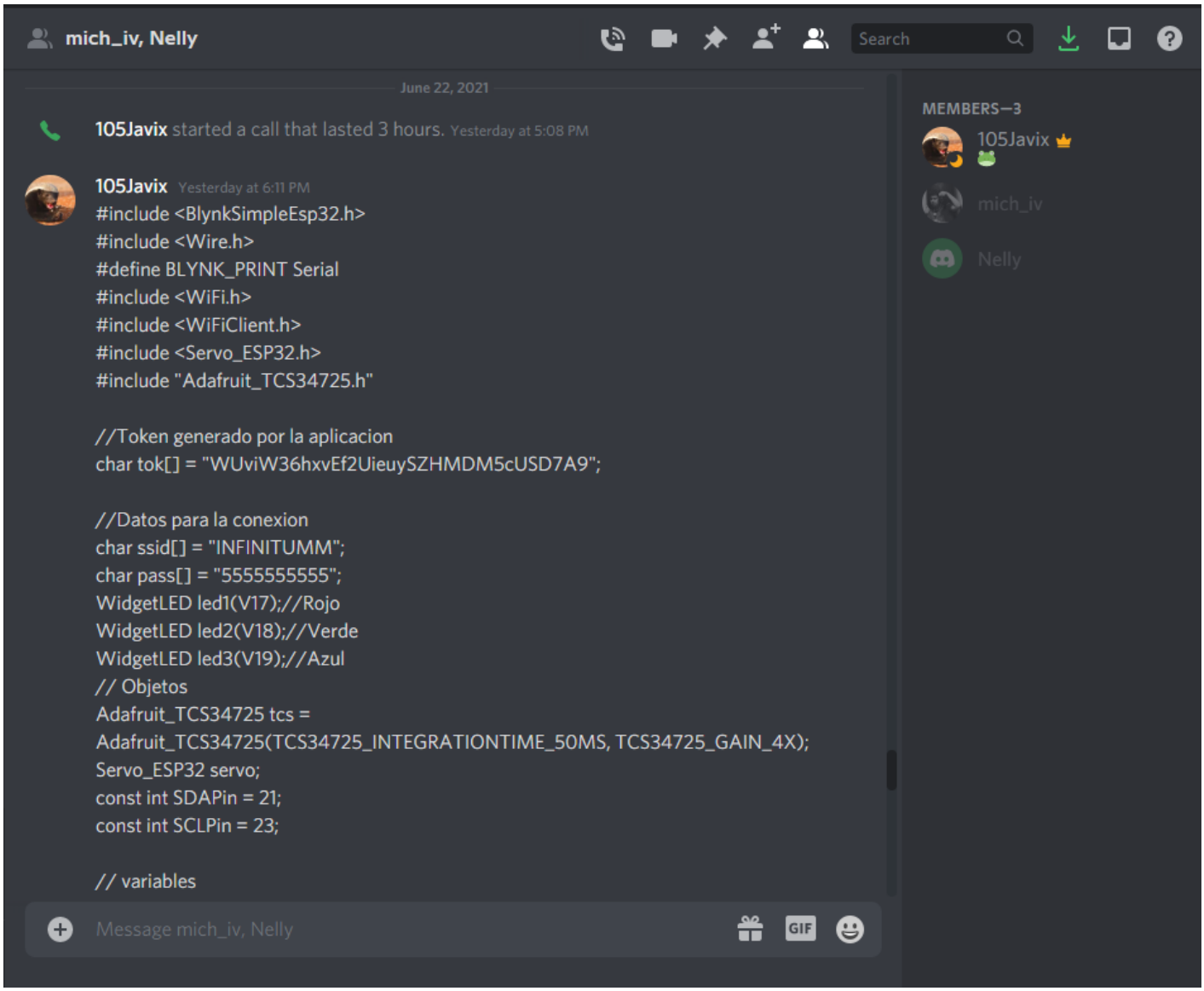
```
//Librerias de blynk
#include <BlynkSimpleEsp32.h>
#define BLYNK_PRINT Serial
//Para cablers SDA y SCL
#include <Wire.h>
//Librerias de wifi
#include <WiFi.h>
#include <WiFiClient.h>
//Libreria de servo y del TCS
```

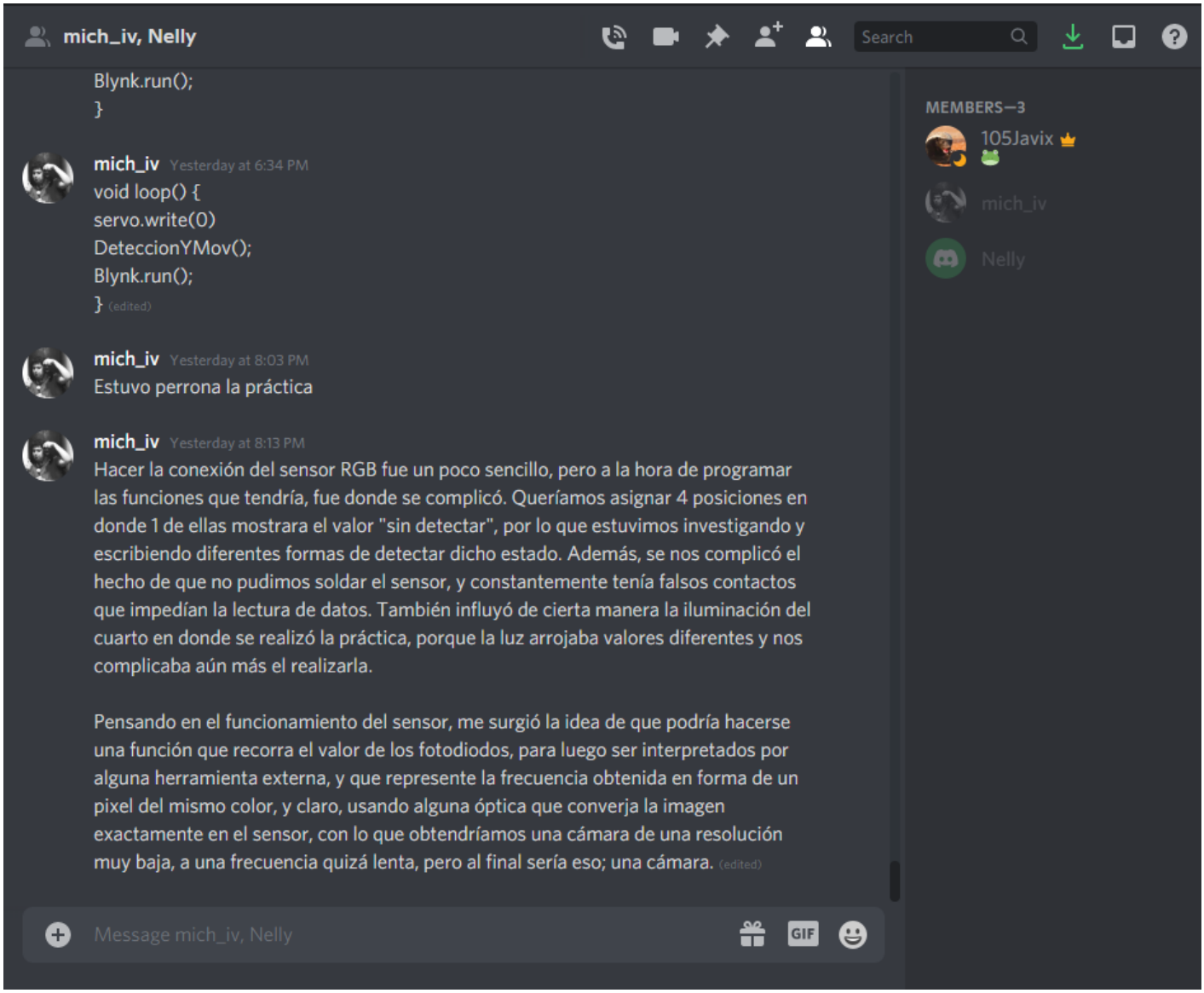
```
#include <Servo_ESP32.h>
#include "Adafruit_TCS34725.h"

//Token generado por blynk
char tok[] = "WUviW36hxvEf2UieuySZHMDM5cUSD7A9";

//Datos para la conexion
char ssid[] = "INFINITUMM";
char pass[] = "5555555555";
//Leds virtuales de Blynk
WidgetLED led1(V17);//Rojo
WidgetLED led2(V18);//Verde
WidgetLED led3(V19);//Azul
//Configuracion del sensor TCS
Adafruit_TCS34725 tcs = Adafruit_TCS34725(TCS34725_INTEGRATIONTIME_50MS,
TCS34725_GAIN_4X);
//Objeto del servo
Servo_ESP32 servo;
//Asignacion de los pins para el TCS
const int SDAPin = 21;
const int SCLPin = 23;

// Pin que controla servo
static const int servoPin = 15;

void setup() {
Serial.begin(115200);
Blynk.begin(tok, ssid, pass);
//Asignacion del pin al servo
servo.attach(servoPin);
//Lectura de datos del sensor
Wire.begin(SDAPin, SCLPin);
if (!tcs.begin())
  {
  Serial.println("Error al iniciar TCS34725");
  while (1) delay(1000);
  }
}

void Deteccion(){
  //interger de 16 bits
  uint16_t r, g, b, c;
  //Lectura de los valores del tcs
  tcs.getRawData(&r,&g,&b,&c);
  if(c > 0 && c < 150)
    {
      servo.write(180);
      Serial.println("Claro");
      Serial.println(c);
      Serial.println(r);
```

```
        Serial.println(g);
        Serial.println(b);
        //Cambiamos tag en blynk
        BLYNK_WRITE(V16);
        Blynk.virtualWrite(V16, "Sin detectar");
        //Led blynk
        led1.off();
        led2.off();
        led3.off();
       }
     else if(r > g && r > b)
     {
      servo.write(90);
      Serial.println("Rojo");
      Serial.println(c);
      Serial.println(r);
      Serial.println(g);
      Serial.println(b);
      //Cambiamos tag en blynk
      BLYNK_WRITE(V16);
      Blynk.virtualWrite(V16, "Rojo");
      //Led blynk
      led3.off();
      led2.off();
      led1.on();
     }
     else if (g > b) {
      servo.write(45);
      Serial.println("verde");
      Serial.println(c);
      Serial.println(r);
      Serial.println(g);
      Serial.println(b);
      //Cambiamos tag en blynk
      BLYNK_WRITE(V16);
      Blynk.virtualWrite(V16, "Verde");
      //Led blynk
      led3.off();
      led2.on();
      led1.off();
      } else{
      servo.write(0);
      Serial.println("Azul");
      Serial.println(b);
      //Cambiamos tag en blynk
      BLYNK_WRITE(V16);
      Blynk.virtualWrite(V16, "Azul");
      //Led blynk
      led3.on();
      led2.off();
```

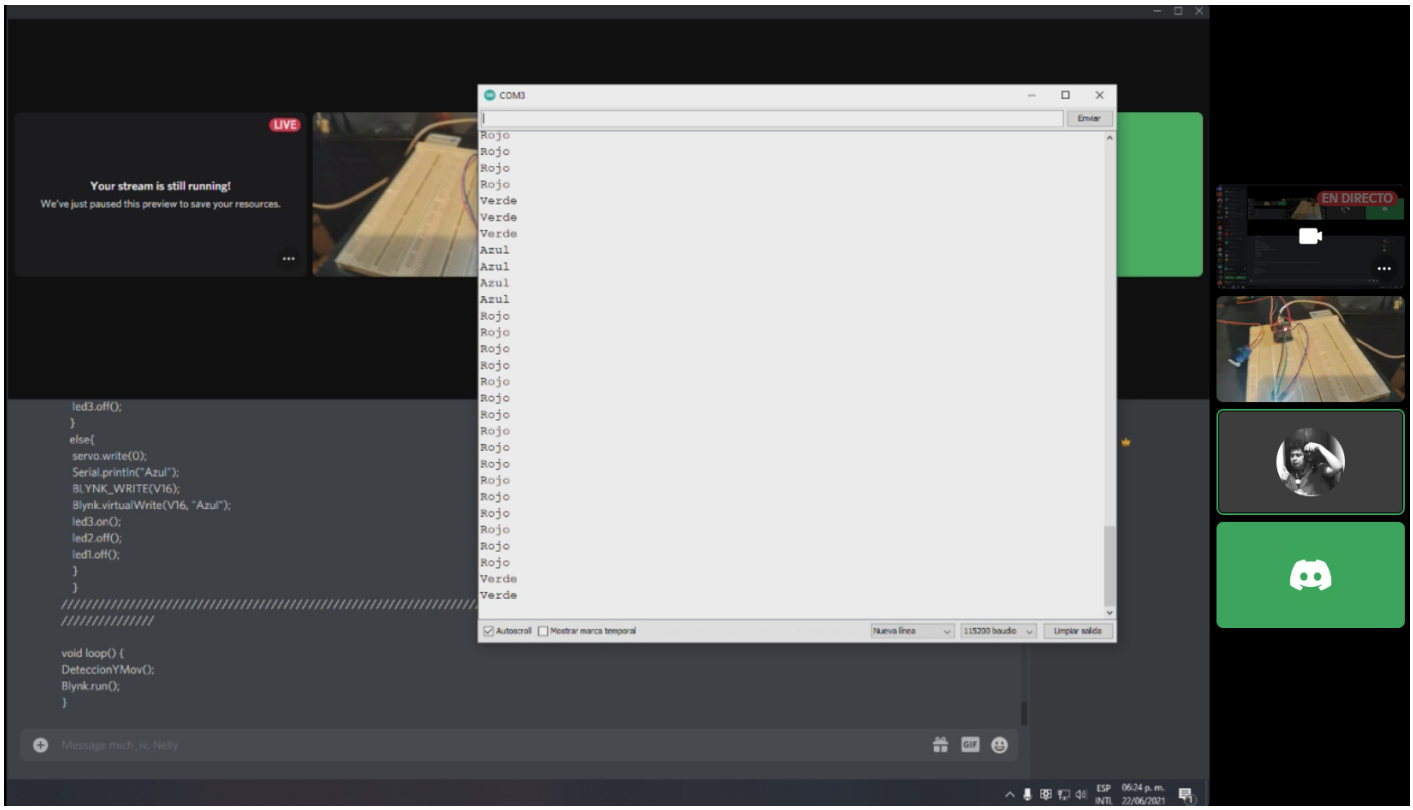
```
    led1.off();
    }
 }

void loop() {
Deteccion();
Blynk.run();
}
```

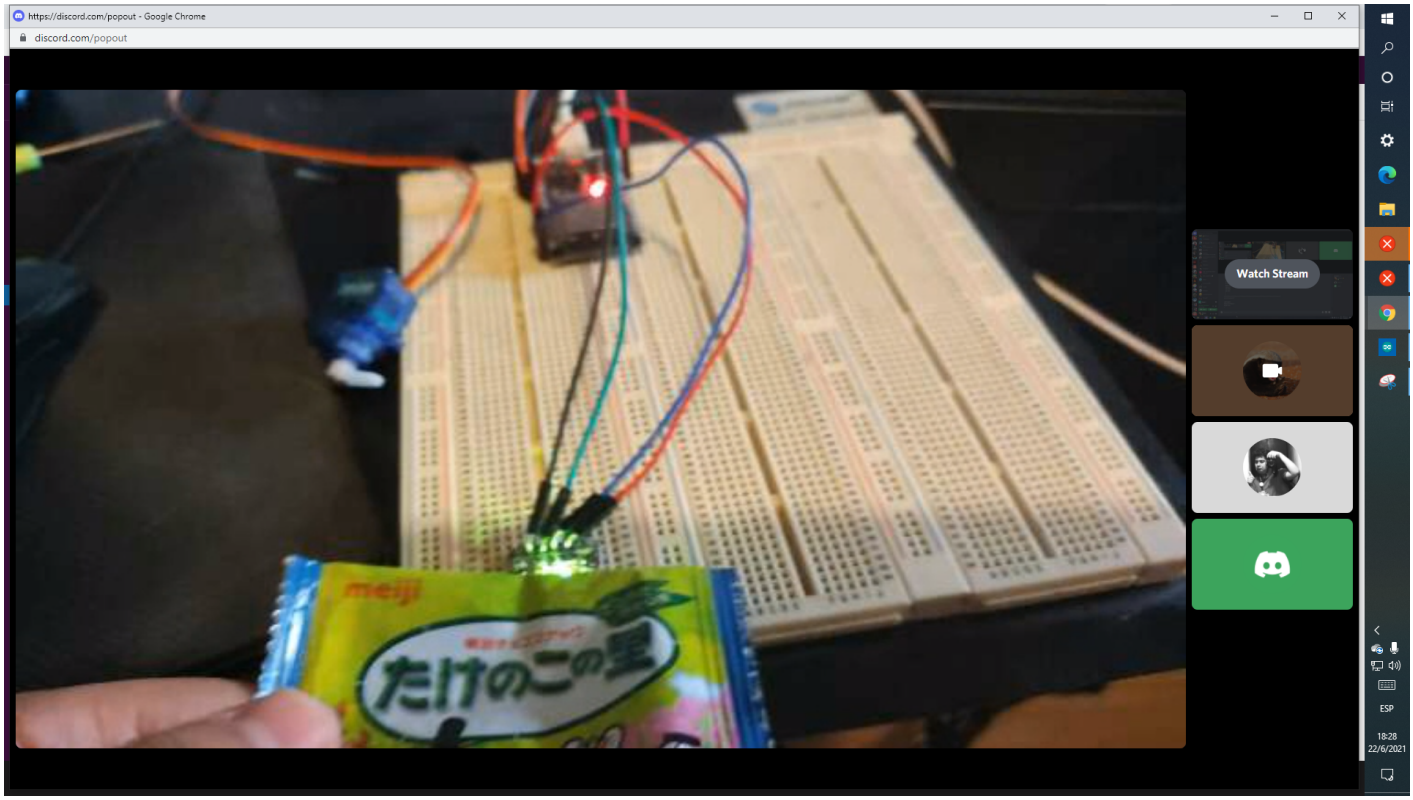
5. Place here evidence that you consider important during the development of the activity.



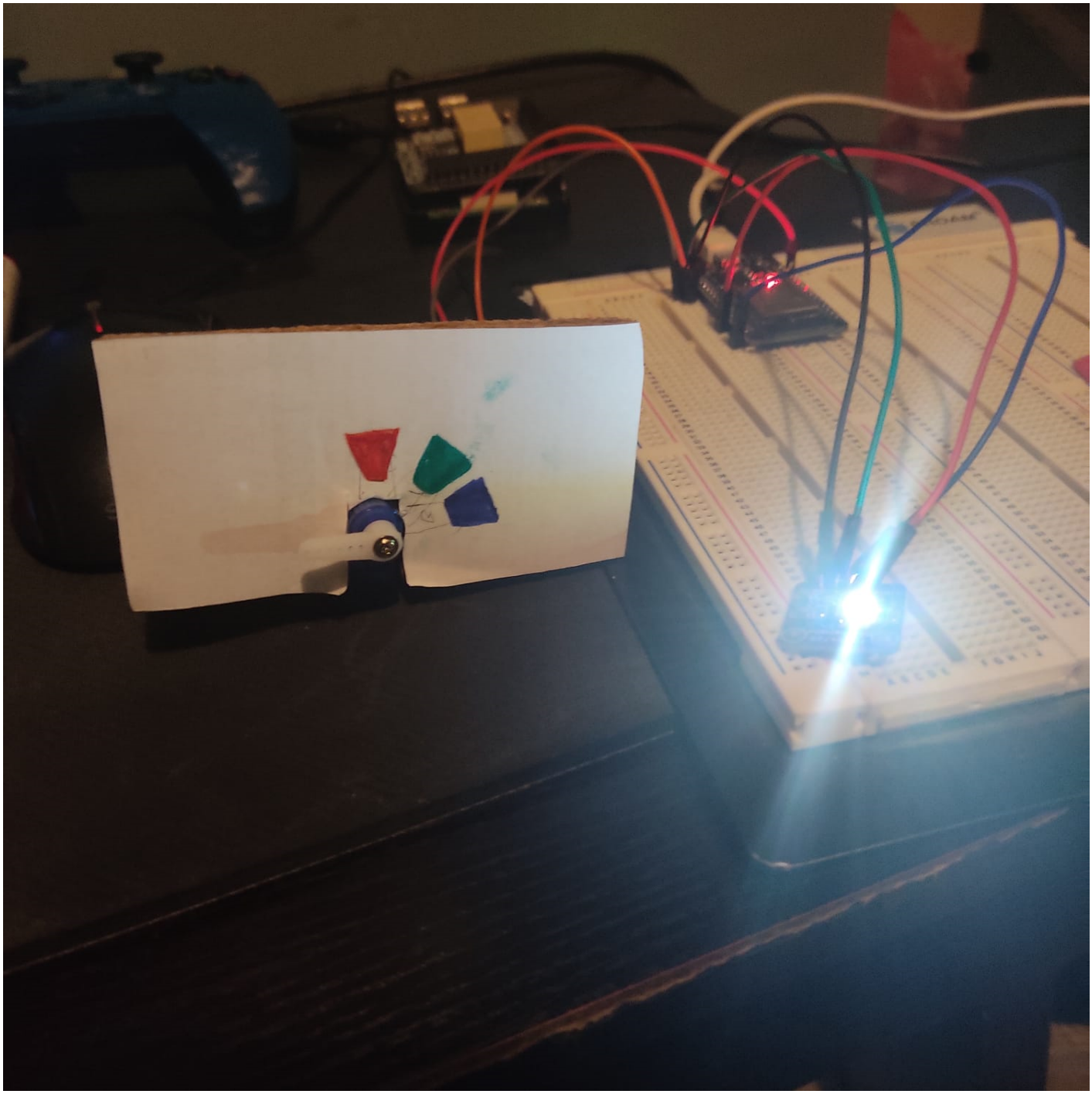


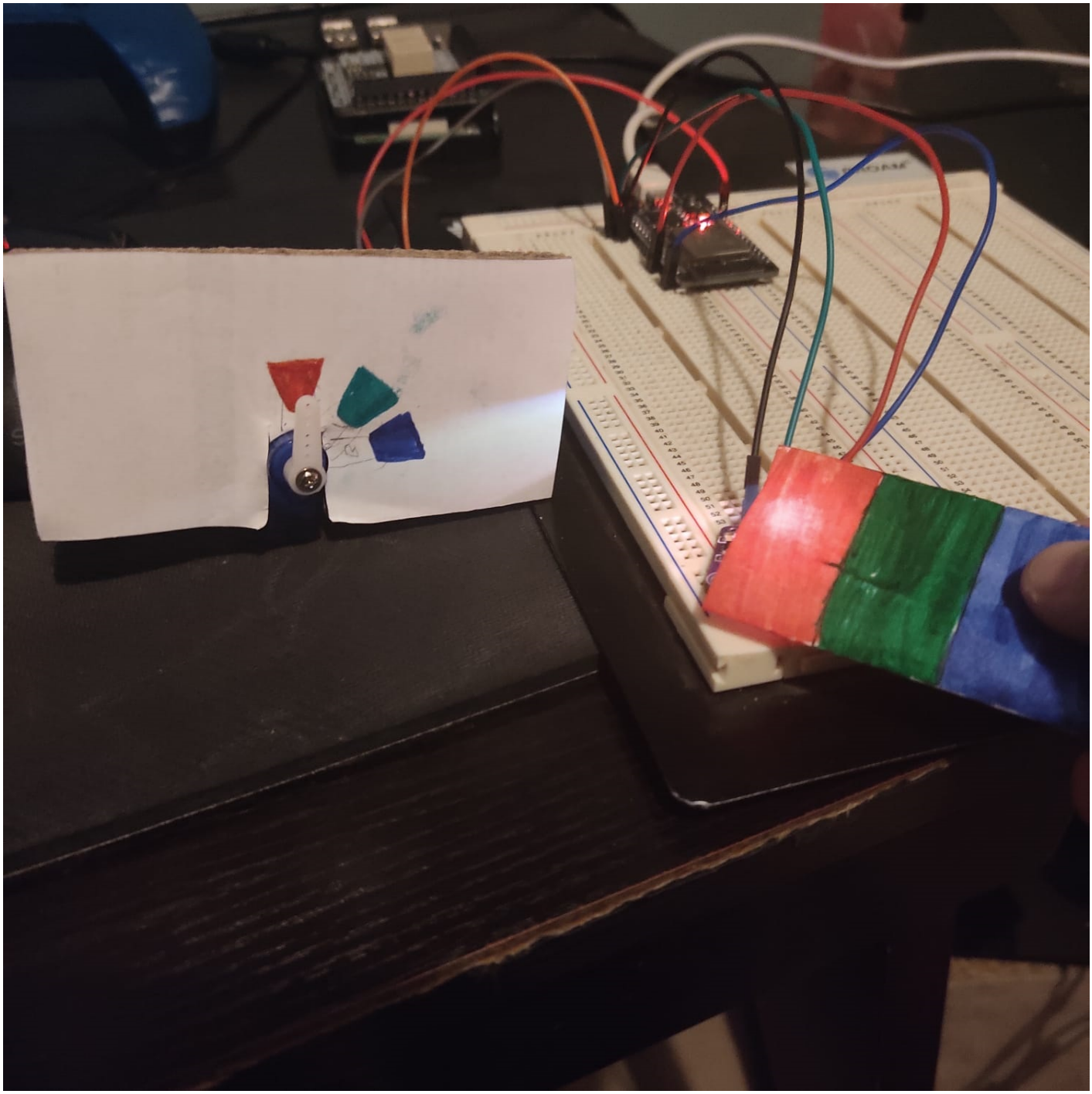
6. Insert images of **evidence** such as meetings of the team members held for the development of the activity

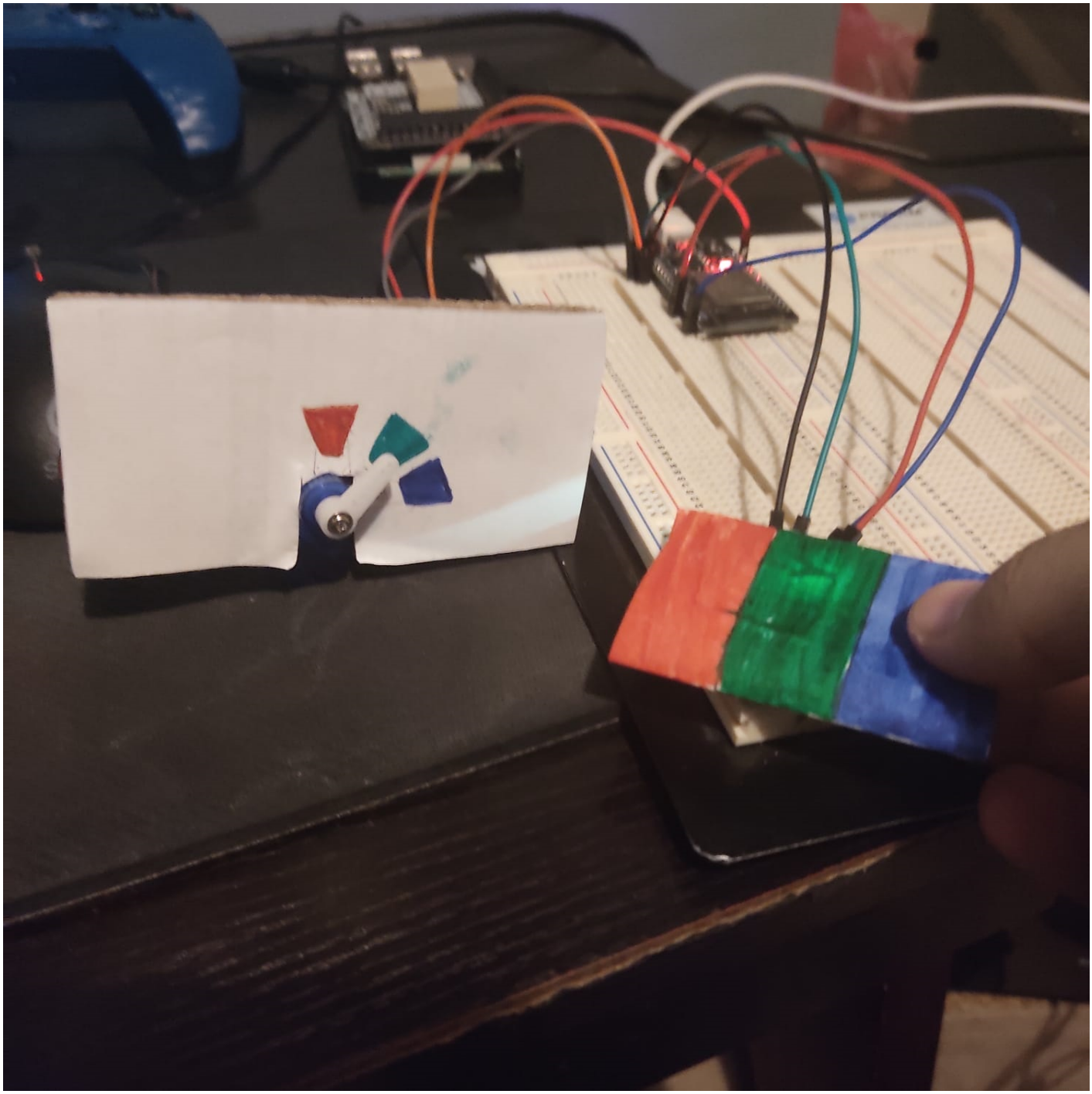


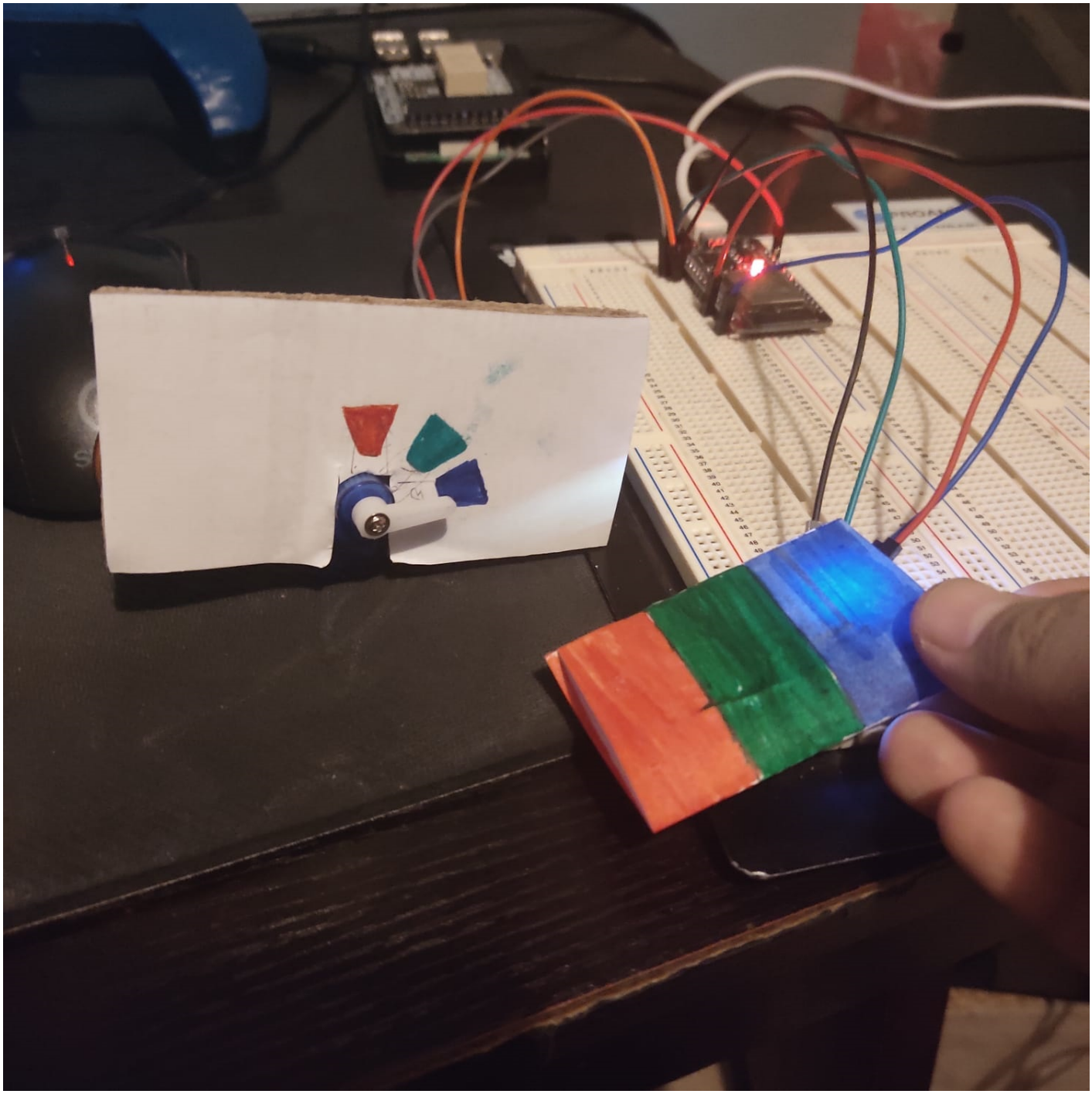


7. For the activity demonstration, more than one object must be used to cover at least three colors.









Nelly Quino

In the first part for this activity we used the ESP32 with the RGB TCS34725 sensor for detecting what color the object is so we needed to detect a object in three situations when was red, blue or green and then showed the colors changes in a interface in our case Blynk we used Wi-Fi to conect us with it how the last activity, In the second part we had to add a Servomotor SG90 which moved on a band of colors (green,red or blue) according what color the object is detected by the sensor for it we had to implemented some librarys like Servo_ESP32.h and Adafruit_TCS34725.h in the code so we could create objects with which we could work with the components and manipulating it until reading and received data. During the development of the activity we looked infomation

that helped us to development this activity how was requirement. We had some problems but we solve it changed values until it will even work. Try and failure.

📷 Michelle Gasca

Making the connection of the RGB sensor was a bit simple, but when it came to programming the functions it would have, it was where it got complicated. We wanted to assign 4 positions where 1 of them would show the value "not detected", so we were investigating and writing different ways to detect this state. In addition, we were complicated by the fact that we could not solder the sensor, and it constantly had false contacts that prevented the reading of data. The lighting of the room where the practice was carried out also influenced in a certain way, because the light threw different values and made it even more difficult for us to do it. Thinking about the operation of the sensor, I got the idea that a function could be made that traverses the value of the photodiodes, to later be interpreted by some external tool, and that represents the frequency obtained in the form of a pixel of the same color, and Of course, using some optics that converges the image exactly on the sensor, with which we would obtain a camera with a very low resolution, at a perhaps slow frequency, but in the end it would be that; a camera.

☕ Francisco Villarreal

During the development of the practice, getting the sensor and servomotor to work was relatively simple, since there were several tutorials that we could rely on and luckily we found libraries that gave us direct control of the servomotor and the LDR to read the colors with the sensor relatively straightforward. We could not make it stand by while there was no color, so we investigated and found a way to make a more raw reading of the colors giving us the option of clear in case no color was detected with that we managed to make the stand by work. In the case of the LDR, there are still options that we could observe to make different readings of the colors and also for what we were able to observe this the option of creating our own library depending on the colors that we have and knowing the values of those colors that the LDR throws on ESP32.

💣 Rubric

Criteria	Description	Score
Instructions	Do you fulfill each of the points indicated in the instruction section?	10
Sevelopment	Did you answer each one of the points requested in the development of the activity?	60
Demonstration	Was the student present in the explanation of the functionality of the activity?	20
Conclusions	Se incluye una opinión personal de la actividad por cada uno de los integrantes del equipo?	10

🎓 Members repositories

🐞 Nelly Quino

 Michelle Gasca

 Francisco Villarreal