



A5.2 Actividad de aprendizaje

Sistema sensor y actuacion del color de un objeto, e interface visual

Instrucciones

- Realizar un sistema de identificacion de color utilizando un sensor de RGB TCS34725, un nodeMCU ESP32, un actuador Servomotor SG90, cualquier protocolo de comunicación y una interface visual que puede ser desarrollada por el equipo o apoyandose de otras como Node-red por ejemplo.
- Toda actividad o reto se deberá realizar utilizando el estilo **MarkDown con extension .md** y el entorno de desarrollo VSCode, debiendo ser elaborado como un documento **single page**, es decir si el documento cuanta con imágenes, enlaces o cualquier documento externo debe ser accedido desde etiquetas y enlaces, y debe ser nombrado con la nomenclatura **A5.2_NombreApellido_Equipo.pdf**.
- Es requisito que el .md contenga una etiqueta del enlace al repositorio de su documento en GITHUB, por ejemplo **Enlace a mi GitHub** y al concluir el reto se deberá subir a github.
- Desde el archivo **.md** exporte un archivo **.pdf** que deberá subirse a classroom dentro de su apartado correspondiente, sirviendo como evidencia de su entrega, ya que siendo la plataforma **oficial** aquí se recibirá la calificación de su actividad.
- Considerando que el archivo **.PDF**, el cual fue obtenido desde archivo **.MD**, ambos deben ser idénticos.
- Su repositorio ademas de que debe contar con un archivo **readme.md** dentro de su directorio raíz, con la información como datos del estudiante, equipo de trabajo, materia, carrera, datos del asesor, e incluso logotipo o imágenes, debe tener un apartado de contenidos o indice, los cuales realmente son ligas o **enlaces a sus documentos .md**, *evite utilizar texto* para indicar enlaces internos o externo.
- Se propone una estructura tal como esta indicada abajo, sin embargo puede utilizarse cualquier otra que le apoye para organizar su repositorio.

```
- readme.md
- blog
  - C5.1_TituloActividad.md
  - C5.2_TituloActividad.md
- IMG
- docs
  - A5.1_TituloActividad.md
  - A5.2_TituloActividad.md
```

Desarrollo

1. Utilizar el siguiente listado de materiales para la elaboración de la actividad

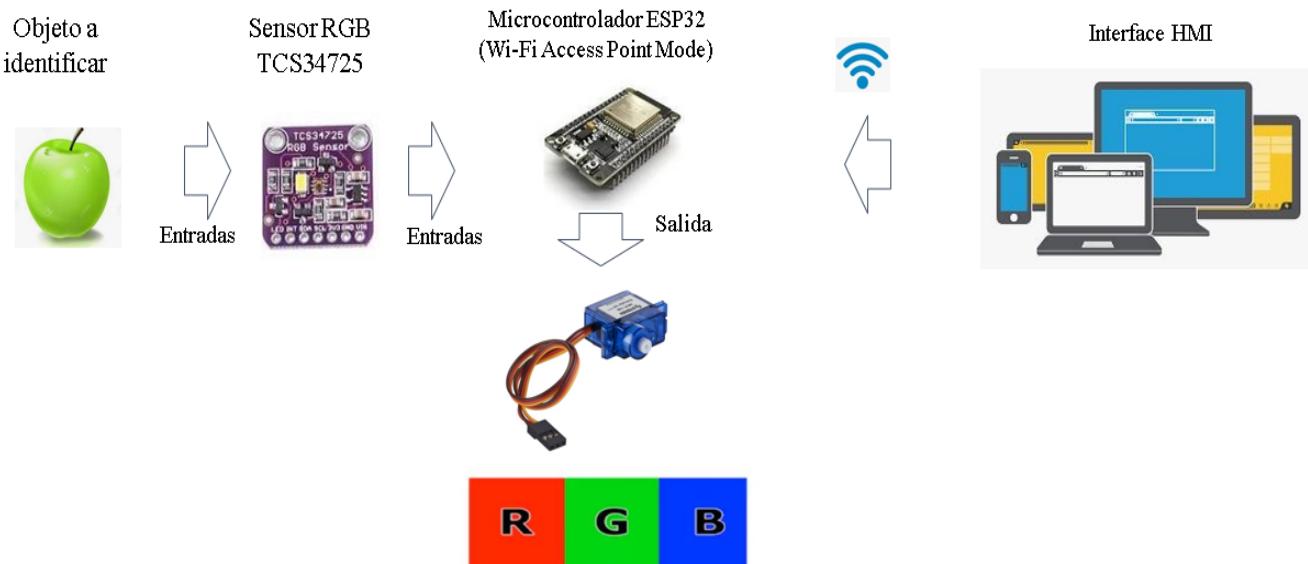
Cantidad	Descripción
1	Sensor RGB TCS34725
1	Sensor de Color TCS230

Cantidad	Descripción
1	Servomotor SG90
1	Fuente de voltaje de 5V
1	NodeMCU ESP32
1	BreadBoard
1	Jumpers M/M

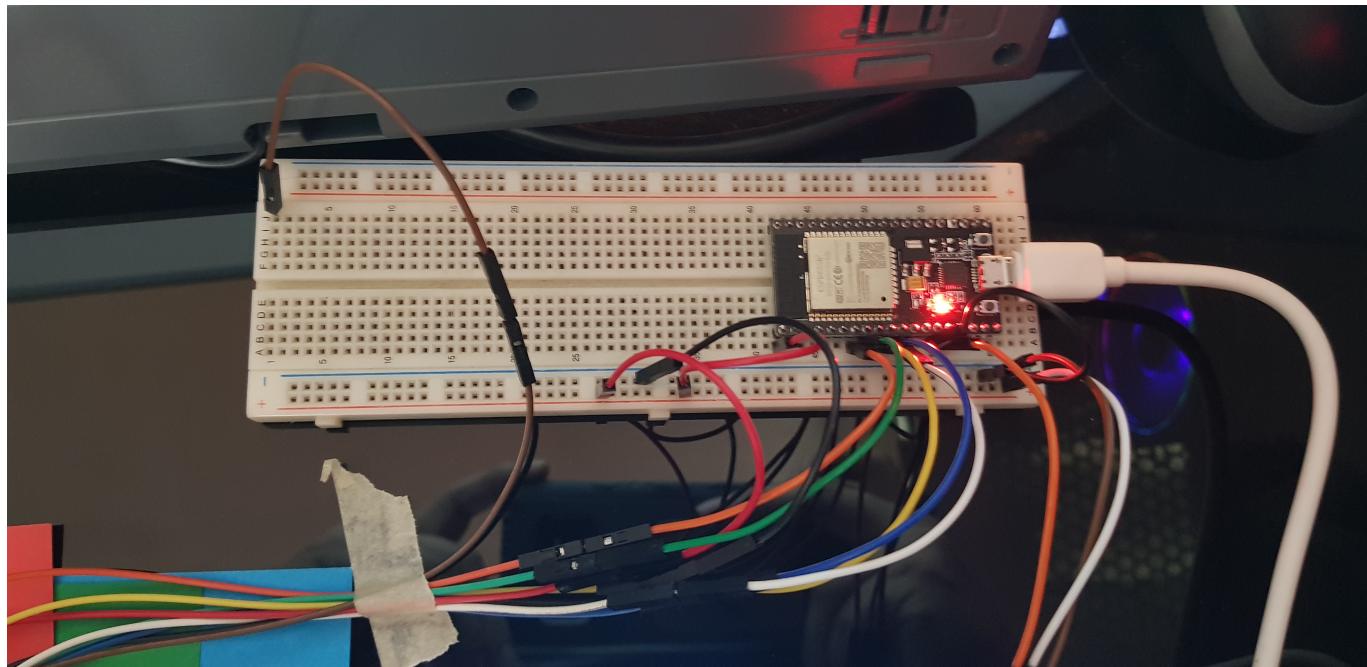
2. Basado en las imágenes que se muestran en las **Figura 1**, realizar un sistema capaz de cumplir con las siguientes condiciones:

- La primer fase de la actividad consistira en, al colocar un objeto frente al sensor RGB, este deberá identificar que color tiene (es recomendable utilizar objetos de color Rojo, Verde, y Azul para mayor precision), el cual deberá mostrar en un interface visual que color fue detectado.
- La segunda fase, consistira en agregar un actuador que y un identificador de color apoyandose de una banda de colores como se muestra en la imagen 1; al conocer el color del objeto el actuador deberá apuntar al color que se esta detectando.

Figura 1 Circuito ESP32 Sensor de color y Servomotor



3. Coloque aquí la imagen del circuito ensamblado



4. Coloque en este lugar el programa creado dentro del entorno de Arduino

```
#include <ESP32Servo.h>

// Sensor de Color
//
//    ESP32    TCS230 SENSOR COLOR
//    32        OUT
//    33        S3
//    25        S2
//    26        S1
//    27        S0
//    5V        VCC
//    GND       GND

//Objeto servo
Servo servo1;

//Declarar salidas
#define S0 27
#define S1 26
#define S2 25
#define S3 33
#define sensorSalida 32
#define servoPin 13

//Variables RGB
int Rojo_Frec = 0;
int Verde_Frec = 0;
int Azul_Frec = 0;

void setup() {
    //Pines RGB
    pinMode(S0, OUTPUT);
```

```
pinMode(S1, OUTPUT);
pinMode(S2, OUTPUT);
pinMode(S3, OUTPUT);
pinMode(sensorSalida, INPUT);

digitalWrite(S0,HIGH);
digitalWrite(S1,LOW);

servo1.attach(servoPin);

Serial.begin(9600);
}

void loop() {
// Configura el filtro ROJO para tomar lectura
digitalWrite(S2,LOW);
digitalWrite(S3,LOW);
delay(100);
Rojo_Frec= pulseIn(sensorSalida, LOW);
// Serial.print(" R= "); Serial.print(Rojo_Frec);
delay(100);

// Configura el filtro VERDE para tomar lectura
digitalWrite(S2,HIGH);
digitalWrite(S3,HIGH);
delay(100);
Verde_Frec = pulseIn(sensorSalida, LOW);
// Serial.print(" V= "); Serial.print(Verde_Frec);
delay(100);

// Configura el filtro AZUL para tomar lectura
digitalWrite(S2,LOW);
digitalWrite(S3,HIGH);
delay(100);
Azul_Frec = pulseIn(sensorSalida, LOW);
// Serial.print(" A= "); Serial.print(Azul_Frec);
delay(100);

//Rango de colores
if (Rojo_Frec < 100 && Verde_Frec > 250 && Azul_Frec > 200) //Rojo
{
  Serial.println(2);
  servo1.write(0);
}

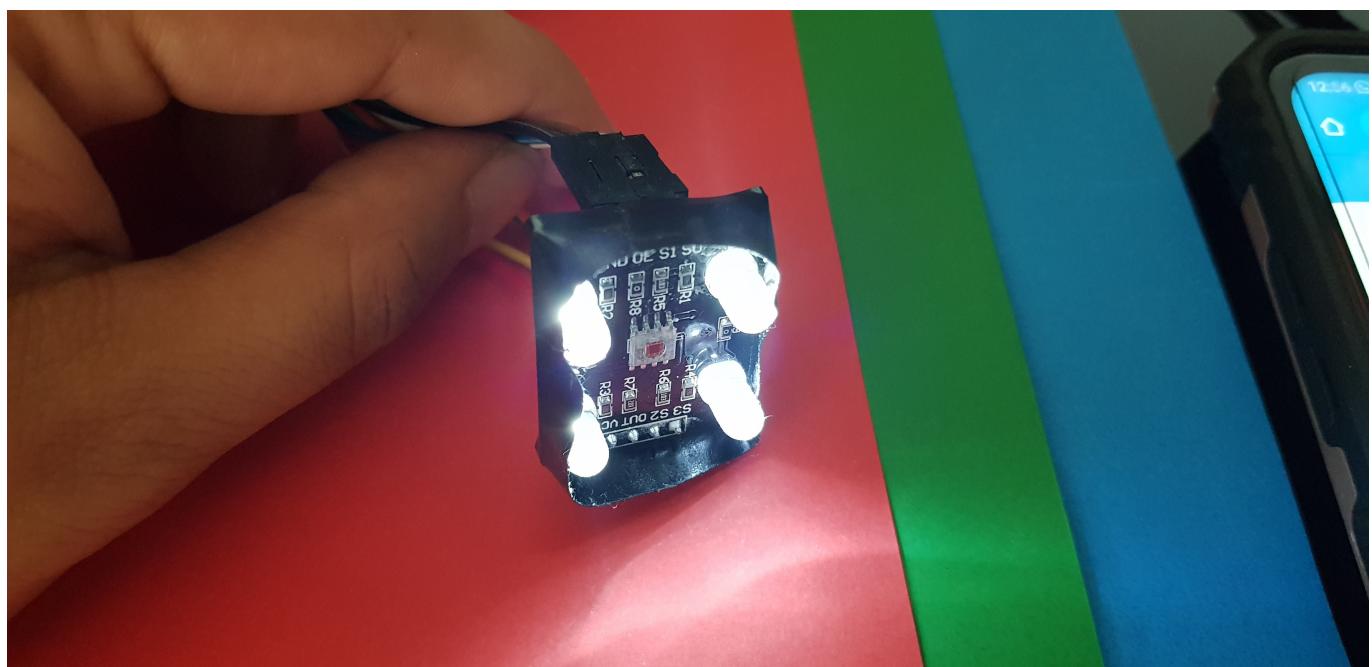
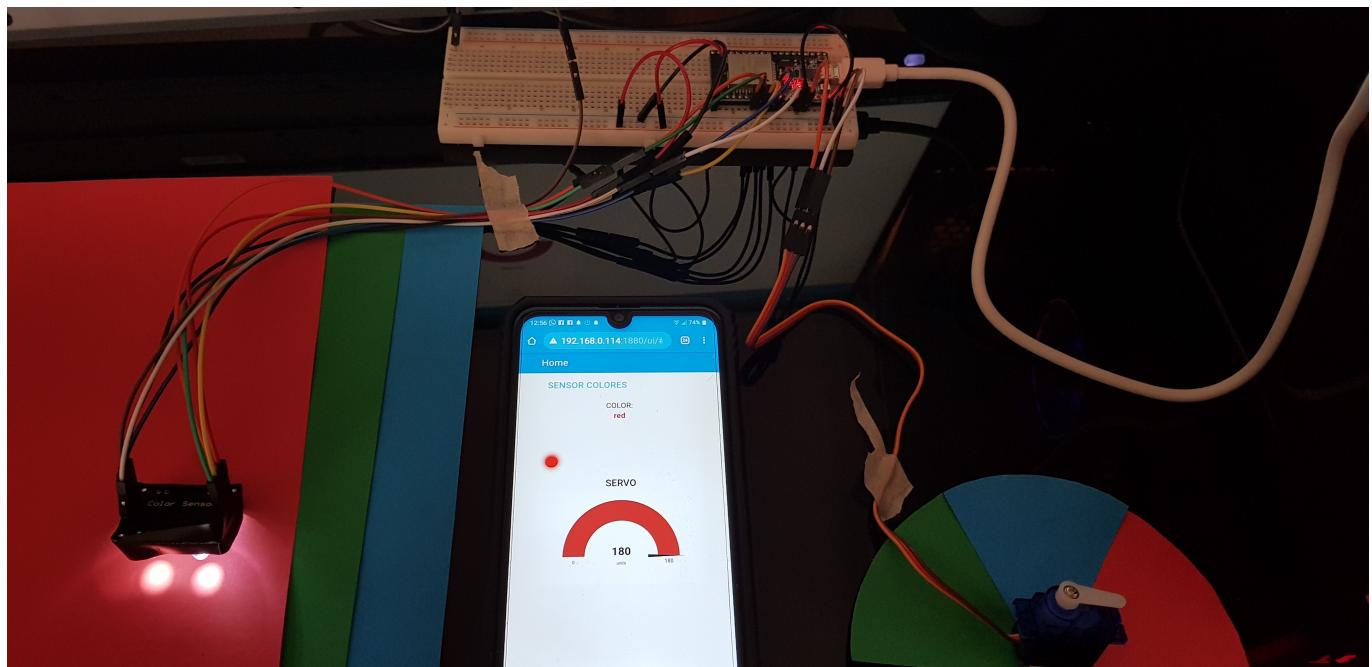
else if (Rojo_Frec > 270 && Verde_Frec < 200 && Azul_Frec < 100) //Azul
{
  Serial.println(1);
  servo1.write(90);
}

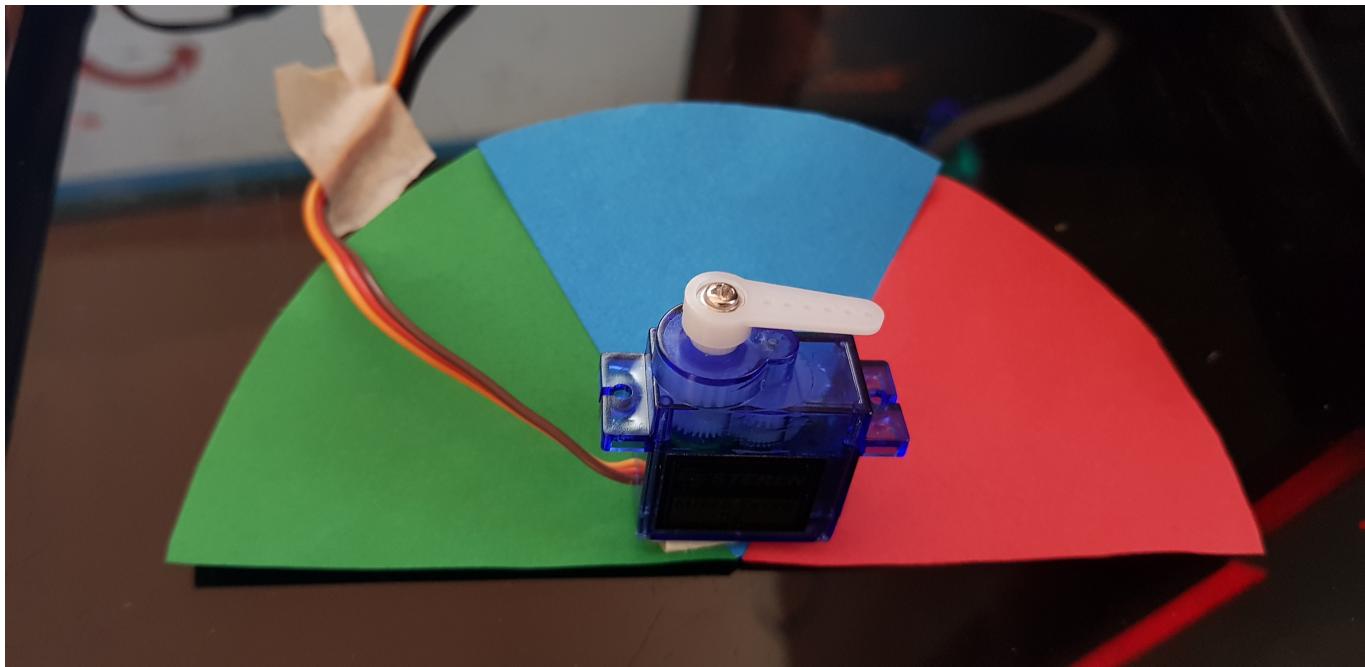
else if (Rojo_Frec > 280 && Verde_Frec < 250 && Azul_Frec < 300) //Verde
{
  Serial.println(0);
  servo1.write(180);
```

```
 }  
  
delay(200);  
}
```

5. Coloque aquí evidencias que considere importantes durante el desarrollo de la actividad.

- Evidencia física realizada por: Jose Alfredo Venegas Medina
- [Presentacion del circuito \(VIDEO\)](#)





Flow 1

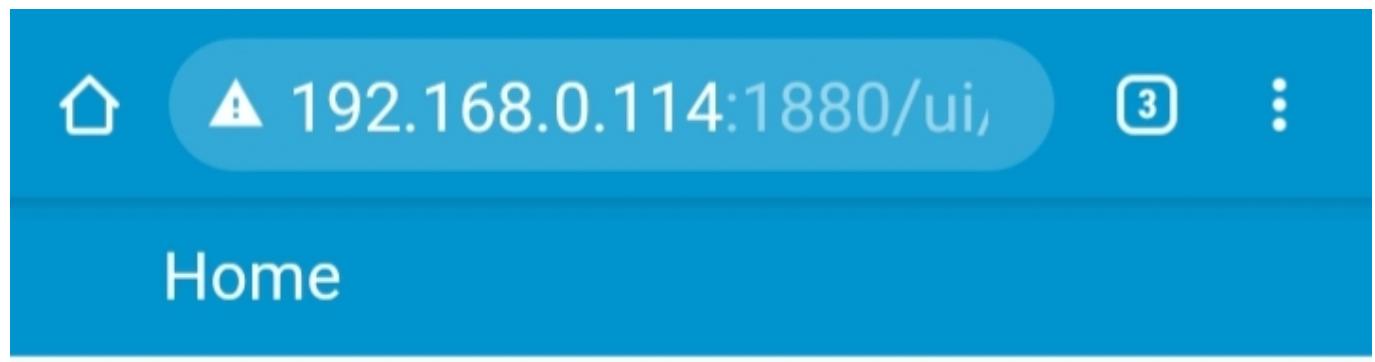
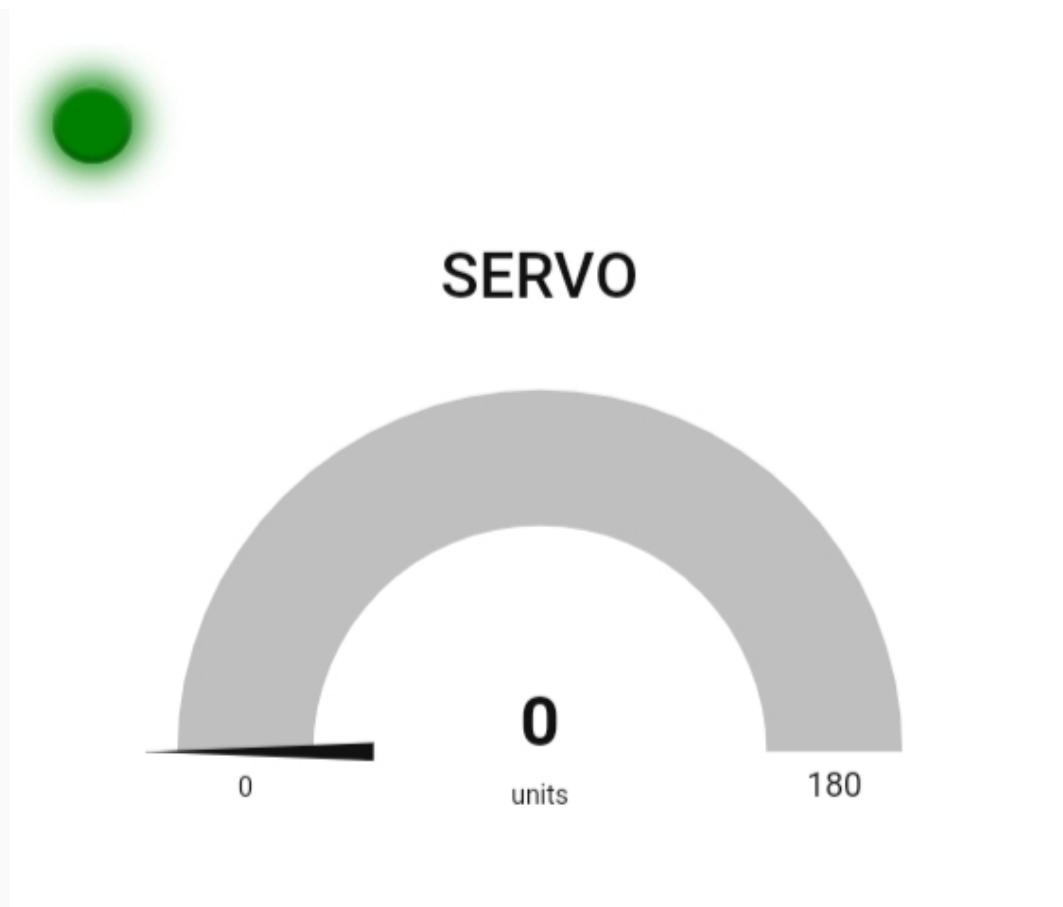
```
graph LR; COM4[COM4] --> f1[f]; f1 --> led[led]; COM4 --> f2[f]; f2 --> COLOR(COLOR: abc); COM4 --> f3[f]; f3 --> SERVO[SERVO]
```

192.168.0.114:1880/ui/

Home

SENSOR COLORES

COLOR:
green



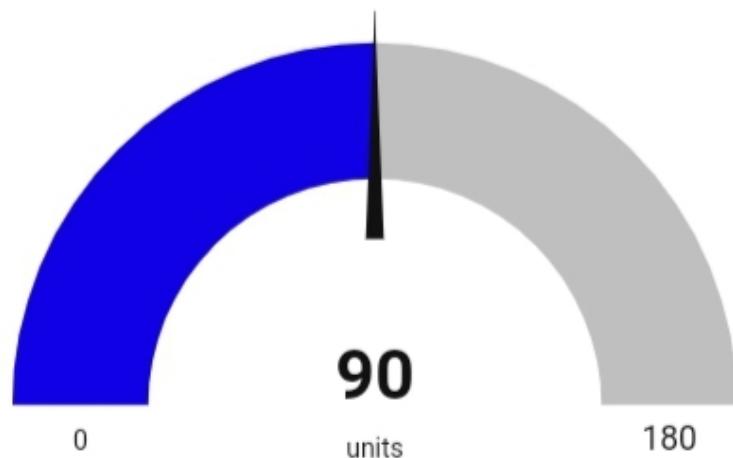
SENSOR COLORES

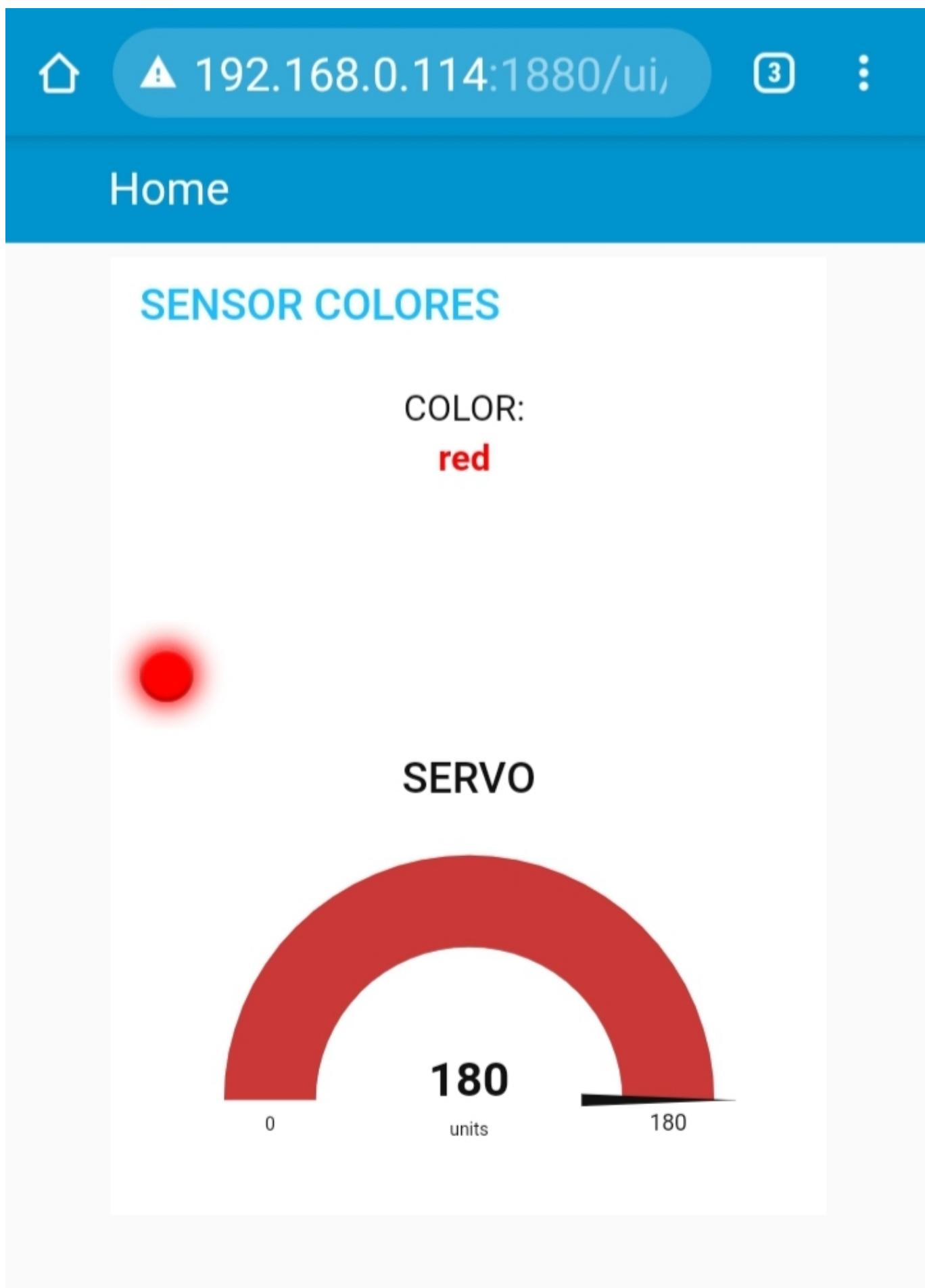
COLOR:

blue



SERVO





6. Para la demostracion de la actividad deberan utilizarse mas de un objeto para poder cubrir minimo tres colores.
7. Evidencia de reuniones

Screenshot of a Google Meet session showing a presentation of a color sensor project.

The presentation slide displays the following code in a text editor:

```

// A5.1_frecuencia_AZUL
// Archivo: Programa_Herramientas_Ayudante
// Autor: Jose Venegas
// Versión: 1.0
// Fecha: 13/01/2021
// Descripción: Programa que lee los colores de un sensor de colores y los imprime en la consola serial.

// Configura el filtro AZUL para tomar lectura
digitalWrite(S2, LOW);
digitalWrite(S3, HIGH);
delay(100);
Azul_Frec = pulseIn(sensorSalida, LOW);
// Serial.print(" A= "); Serial.print(Azul_Frec);
delay(100);

//Rango de colores
if (Rojo_Frec < 100 && Verde_Frec > 250 && Azul_Frec > 200) //Rojo
{
    Serial.println(2);
    servol.write(0);
}

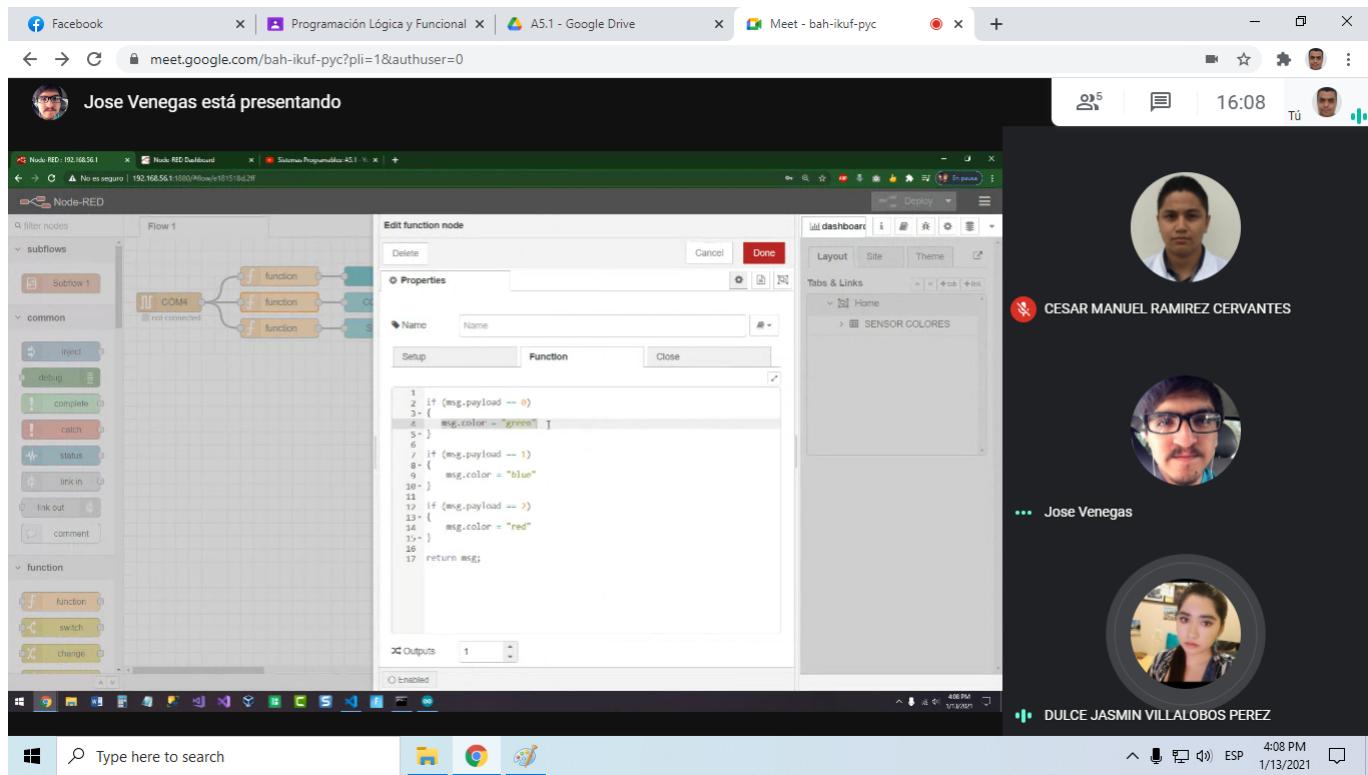
else if (Rojo_Frec > 270 && Verde_Frec < 200 && Azul_Frec < 100) //Azul
{
    Serial.println(1);
    servol.write(90);
}

```

The slide also includes a screenshot of the Visual Studio Code interface showing the project structure and code for "A5.1_frecuencia_AZUL". The interface shows multiple tabs for different files like "A5.1.h", "A5.1.ino", and "A5.1_md". The code in "A5.1.ino" is identical to the one shown in the text editor.

Participants in the Google Meet session:

- CESAR MANUEL RAMIREZ CERVANTES
- Jose Venegas
- DULCE JASMIN VILLALOBOS PEREZ



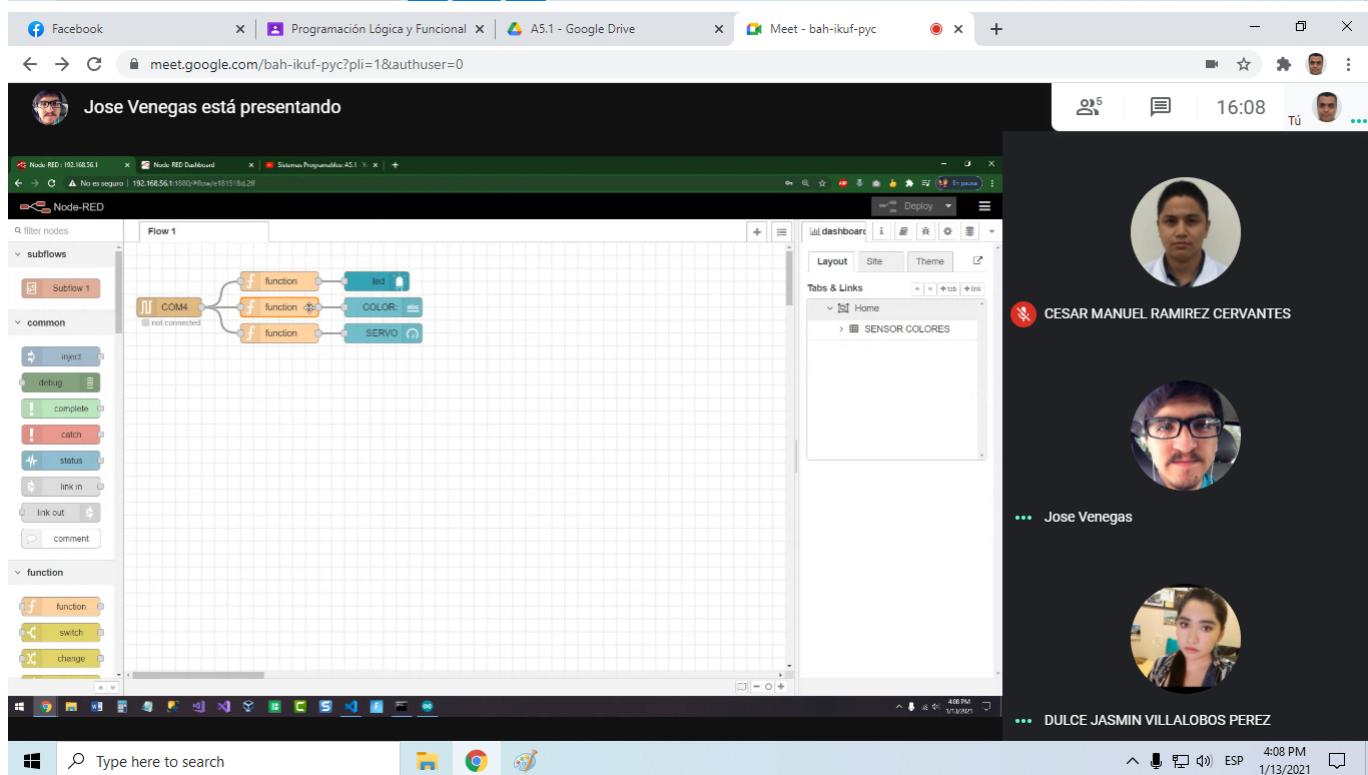
The screenshot shows a Node-RED interface running on a Windows desktop. On the left, the Node-RED palette lists various nodes like inject, debug, complete, catch, status, link in, link out, and comment. In the center, a flow is being edited. A 'COM4' node is connected to three 'function' nodes. These three function nodes are then connected to an 'led' node. The 'function' nodes contain the following JavaScript code:

```

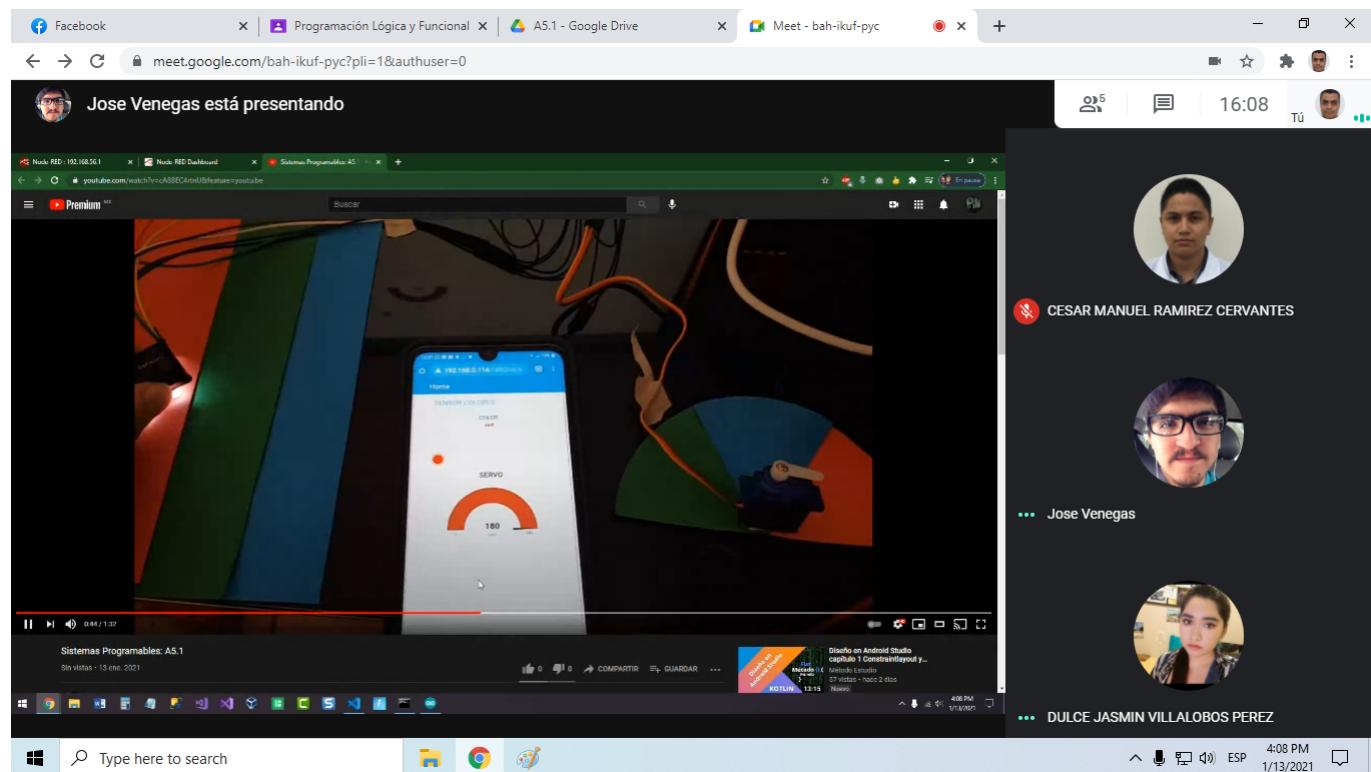
1 if (msg.payload == 0)
2 {
3   msg.color = "green"
4 }
5 if (msg.payload == 1)
6 {
7   msg.color = "blue"
8 }
9 if (msg.payload == 2)
10 {
11   msg.color = "red"
12 }
13 return msg;
14
15
16
17

```

On the right, a Google Meet video call is displayed. Five participants are shown in circular avatars. The participant at the top is CESAR MANUEL RAMIREZ CERVANTES. Below him is Jose Venegas. At the bottom is DULCE JASMIN VILLALOBOS PEREZ. The video call interface includes a 'Layout' tab, a 'Site' tab, and a 'Theme' tab. The time is 16:08.



This screenshot is nearly identical to the one above, showing the same Node-RED flow and Google Meet video call. The only difference is the timestamp, which has changed to 16:08.



8. Conclusiones

- **Acevedo Ensiso Pedro Gabriel:**

Dentro de la realización de esta práctica comprendí mejor el funcionamiento de un sistema de detección de colores, el cual pude ver con mejor claridad qué es lo que se necesita para ser implementado correctamente. Un sistema detector de colores puede ser una buena forma de automatizar muchos procesos industriales dentro de una fábrica tales como la organización de materiales hasta la automatización de montaje y ensamblado de elementos que posiblemente sean muy complicados para ser manejados por seres humanos. Podemos implementar por ejemplo un sistema detector de colores en un campo de cultivo con lo cual podemos separar correctamente las distintas frutas recolectadas basándose en su color. Con esto finalmente comprendo mejor las complicaciones que pueden existir al no tener bien calibrado un sistema como este ya que de no hacerlo correctamente podemos tener muchos errores los cuales pueden costarle mucho dinero a la empresa donde laboramos.

- **Ramirez Cervantes Cesar Manuel:**

En esta práctica con ayuda de un sensor RGB TCS34725 se realizó un circuito capaz de identificar el color de un objeto, utilizando objetos en este caso hojas de los colores principales del RGB para su mayor precisión. Una vez que el sensor detecta el color de dicho objeto envía esa información al ESP32 para que esté a su vez por medio de un actuador Servomotor SG90 indique visualmente que color es el que reconoció el sensor RGB lo interesante de esta práctica es que esta información de que color es el que está identificando el sensor RGB se va a mostrar en tiempo real en una interfaz web mostrando el color inclusive en nuestro caso muestra el lado hacia donde apunta el servomotor. Por esté motivo sistemas detectores de color son muy utilizados en procesos industriales ya que con ellos podemos automatizar mejor los procesos de producción porque nos permiten reconocer patrones de una manera automática.

- **Venegas Medina Jose Alfredo:**

En esta práctica se realizó un circuito capaz de detectar 3 colores, el rojo, azul y verde utilizando un

sensor capaz de medir frecuencias de colores, y un motor que se movería indicando en color detectado. Además de una página creada en Node-red la cual evalúa los distintos cambios de color que el sensor detecte. Si bien, esta fue una práctica laboriosa ya que se debían realizar diferentes puntos con los conocimientos previos de las anteriores prácticas se pudieron resolver con facilidad. Por otro lado, los sensores de color son realmente utilizados en las industrias de fabricación y transporte de producto clasificándolos por colores. Podría ser desde el color de la tapa, etiqueta, por el color del producto e incluso verificando el sello de calidad. Ejemplo de esto serían las piezas de juguetes como las de lego, las cuales, aunque muchas sean iguales entre sí, es necesario que sean de distintos colores para los distintos sets. Otra aplicación sería a la hora de seleccionar algunos productos alimenticios o de dulce como los panditas, M&M'S y los Skittles los cuales, dependiendo de su producto pueden variar los colores de estos. Así, en estos casos de la industria, el servomotor podría utilizarse para separar los productos por color.

- Villalobos Perez Dulce Jasmin:**

En esta práctica se utilizó un sensor RGB, con el objetivo principal de identificar el color que se pone de bajo del sensor, por ejemplo al poner una hoja o cartulina de color el sensor determinaría qué color tiene, esto es posible ya que el sensor puede detectar una serie de combinaciones de colores a partir de la información a través de la luz, utilizando tres sensores acoplados que realizan la separación de la luz que concuerdan con los colores rojo, verde y azul. Principalmente para lograr esto mi compañero tuvo que identificar cuál era su frecuencia de color, cuando de vínculo la conexión con el nodeRed, para trabajar con la interfaz gráfica y se pueda ver el resultado en pantalla, se envió un mensaje con un valor, este valor se convirtió en un entero para que sea más fácil de identificar, por ejemplo si se recibe un 2 es rojo, un 1 es azul y si es cero entonces es verde. Cuando se utilizó el servomotor se puso la condición de que si es rojo o sea el valor 0 la aguja del servomotor giraría 180 grados, si es verde regresaría o permanecería en 0 grados, y cuando es azul esta se posiciona en los 90 grados. De esta manera se mostró tanto en la interfaz gráfica como en la representación con el servomotor. Como se vio este sensor RGB se puede aplicar en una línea de producción por ejemplo al detectar el color de las bolsas de empaquetado, si todas son del mismo o se filtra una de un color diferente al de la línea principal.

 **Rubrica**

Criterios	Descripción	Puntaje
Instrucciones	Se cumple con cada uno de los puntos indicados dentro del apartado Instrucciones?	10
Desarrollo	Se respondió a cada uno de los puntos solicitados dentro del desarrollo de la actividad?	60
Demostración	El alumno se presenta durante la explicación de la funcionalidad de la actividad?	20
Conclusiones	Se incluye una opinión personal de la actividad por cada uno de los integrantes del equipo?	10

EQUIPO VERDE
Acevedo Ensiso Pedro Gabriel:

 Ir a mi Github

Ramirez Cervantes Cesar Manuel:

 Enlace a mi repositorio 

Venegas Medina Jose Alfredo:

 Mi Github

Villalobos Perez Dulce Jasmin:

  ENLACE - MI GITHUB