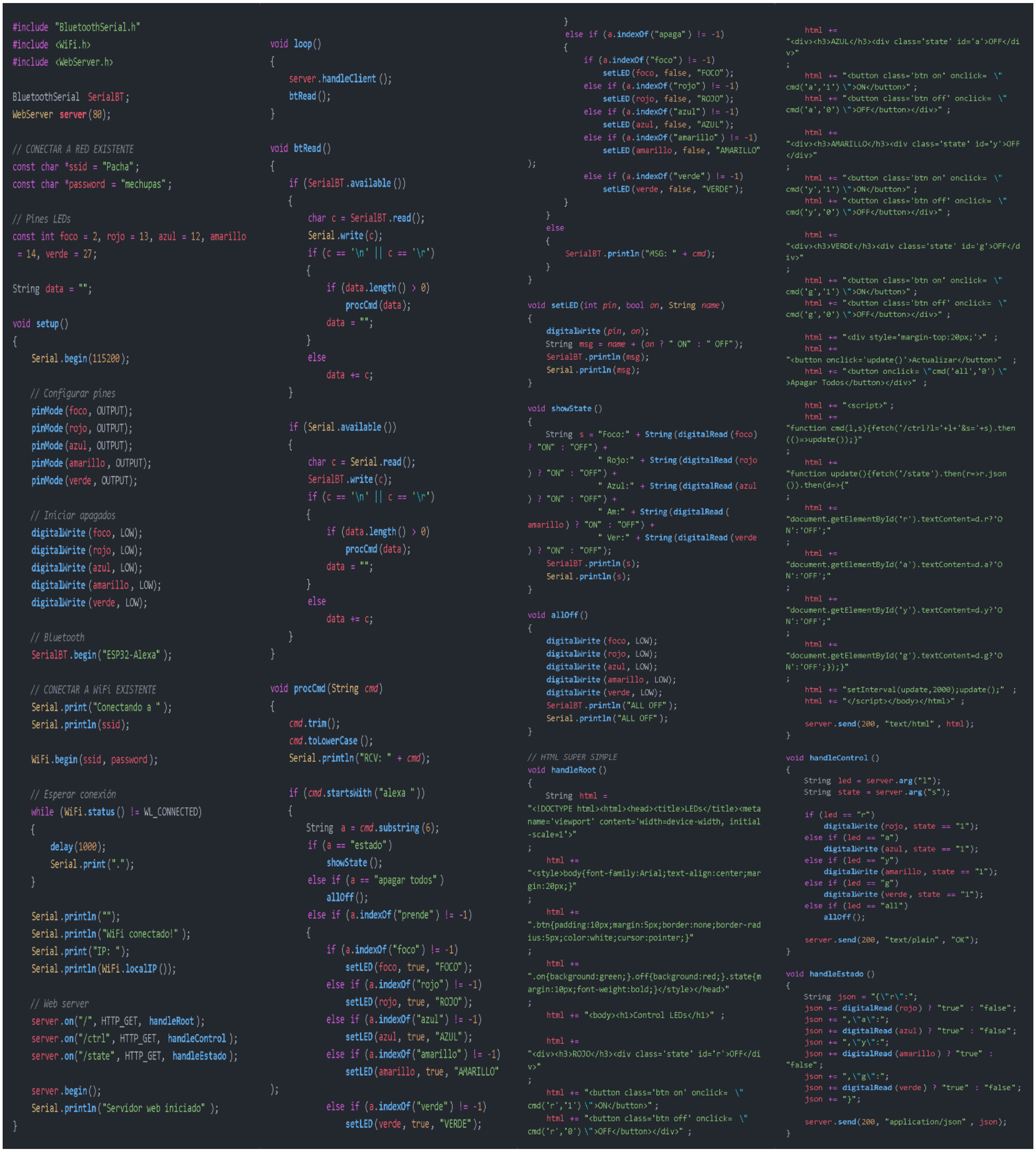
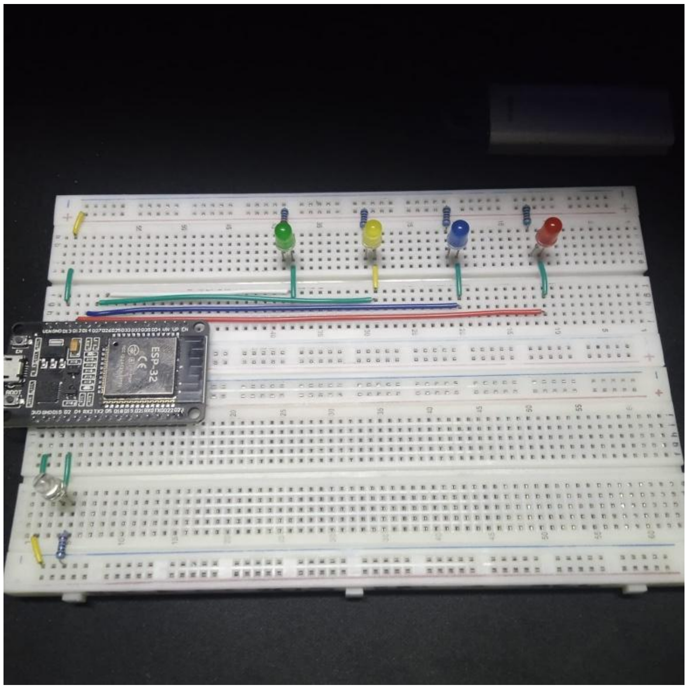
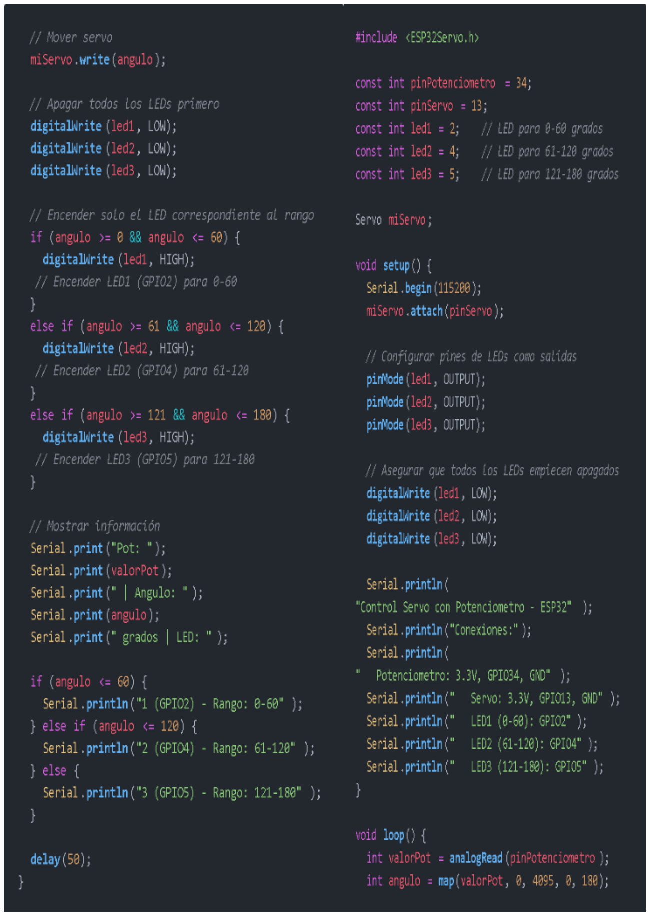
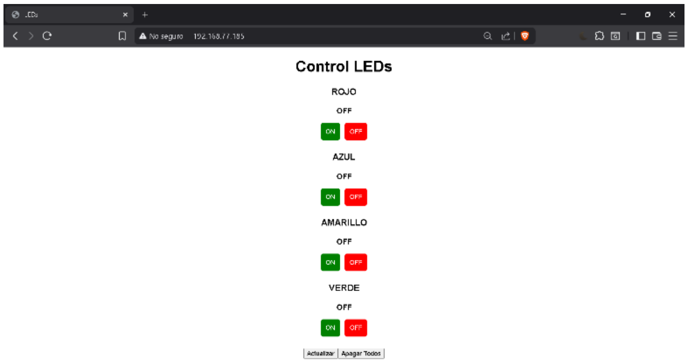


```
#include "BluetoothSerial.h"
#include <WiFi.h>
#include <WebServer.h>

BluetoothSerial  SerialBT;
WebServer server(80);

// CONECTAR A RED EXISTENTE
const char *ssid = "Pacha";
const char *password = "mechupas";

// Pines LEDs
const int foco = 2, rojo = 13, azul = 12, amarillo
 = 14, verde = 27;

String data = "";

void setup()
{
    Serial.begin(115200);

    // Configurar pines
    pinMode(foco, OUTPUT);
    pinMode(rojo, OUTPUT);
    pinMode(azul, OUTPUT);
    pinMode(amarillo, OUTPUT);
    pinMode(verde, OUTPUT);

    // Iniciar apagados
    digitalWrite(foco, LOW);
    digitalWrite(rojo, LOW);
    digitalWrite(azul, LOW);
    digitalWrite(amarillo, LOW);
    digitalWrite(verde, LOW);

    // Bluetooth
    SerialBT.begin("ESP32-Alexa" );

    // CONECTAR A WiFi EXISTENTE
    Serial.print("Conectando a " );
    Serial.println(ssid);

    WiFi.begin(ssid, password);

    // Esperar conexión
    while (WiFi.status() != WL_CONNECTED)
    {
        delay(1000);
        Serial.print(".");
    }

    Serial.println("");
    Serial.println("WiFi conectado!" );
    Serial.print("IP: ");
    Serial.println(WiFi.localIP());

    // Web server
    server.on("/", HTTP_GET,  handleRoot);
    server.on("/ctrl", HTTP_GET,  handleControl);
    server.on("/state", HTTP_GET,  handleEstado);

    server.begin();
    Serial.println("Servidor web iniciado" );
}

void loop()
{
    server.handleClient();
    btRead();
}

void btRead()
{
    if (SerialBT.available())
    {
        char c = SerialBT.read();
        Serial.write(c);
        if (c == '\n' || c == '\r')
        {
            if (data.length() > 0)
                procCmd(data);
            data = "";
        }
        else
            data += c;
    }

    if (Serial.available())
    {
        char c = Serial.read();
        SerialBT.write(c);
        if (c == '\n' || c == '\r')
        {
            if (data.length() > 0)
                procCmd(data);
            data = "";
        }
        else
            data += c;
    }
}

void procCmd(String cmd)
{
    cmd.trim();
    cmd.toLowerCase();
    Serial.println("RCV: " + cmd);

    if (cmd.startsWith("alexa "))
    {
        String  a = cmd.substring(6);
        if (a == "estado")
            showState();
        else if (a == "apagar todos" )
            allOff();
        else if (a.indexOf("prende") != -1)
        {
            if (a.indexOf("foco") != -1)
                setLED(foco, true, "FOCO");
            else if (a.indexOf("rojo") != -1)
                setLED(rojo, true, "ROJO");
            else if (a.indexOf("azul") != -1)
                setLED(azul, true, "AZUL");
            else if (a.indexOf("amarillo" ) != -1)
                setLED(amarillo, true, "AMARILLO"
);
            else if (a.indexOf("verde") != -1)
                setLED(verde, true, "VERDE");
        }
        else if (a.indexOf("apaga") != -1)
        {
            if (a.indexOf("foco") != -1)
                setLED(foco, false, "FOCO");
            else if (a.indexOf("rojo") != -1)
                setLED(rojo, false, "ROJO");
            else if (a.indexOf("azul") != -1)
                setLED(azul, false, "AZUL");
            else if (a.indexOf("amarillo" ) != -1)
                setLED(amarillo, false, "AMARILLO"
);
            else if (a.indexOf("verde") != -1)
                setLED(verde, false, "VERDE");
        }
    }
    else
    {
        SerialBT.println("MSG: " + cmd);
    }
}

void setLED(int pin, bool on, String name)
{
    digitalWrite(pin, on);
    String  msg = name + (on ? " ON" : " OFF");
    SerialBT.println(msg);
    Serial.println(msg);
}

void showState()
{
    String  s = "Foco:" + String(digitalRead(foco)
? "ON" : "OFF") +
                " Rojo:" + String(digitalRead(rojo
) ? "ON" : "OFF") +
                " Azul:" + String(digitalRead(azul
) ? "ON" : "OFF") +
                " Am:" + String(digitalRead(
amarillo) ? "ON" : "OFF") +
                " Ver:" + String(digitalRead(verde
) ? "ON" : "OFF");
    SerialBT.println(s);
    Serial.println(s);
}

void allOff()
{
    digitalWrite(foco, LOW);
    digitalWrite(rojo, LOW);
    digitalWrite(azul, LOW);
    digitalWrite(amarillo, LOW);
    digitalWrite(verde, LOW);
    SerialBT.println("ALL OFF" );
    Serial.println("ALL OFF" );
}

// HTML SUPER SIMPLE
void handleRoot()
{
    String  html =
"<!DOCTYPE html><html><head><title>LEDs</title><meta
name='viewport' content='width=device-width, initial
-scale=1'>"
;
    html +=
"<style>body{font-family:Arial;text-align:center;mar
gin:20px;}"
;
    html +=
".btn{padding:10px;margin:5px;border:none;border-rad
ius:5px;color:white;cursor:pointer;}"
;
    html +=
".on{background:green;}.off{background:red;}.state{m
argin:10px;font-weight:bold;}</style></head>"
;
    html += "<body><h1>Control LEDs</h1>" ;

    html +=
"<div><h3>ROJO</h3><div class='state' id='r'>OFF</di
v>"
;
    html += "<button class='btn on' onclick= \"
cmd('r','1') \">ON</button>" ;
    html += "<button class='btn off' onclick= \"
cmd('r','0') \">OFF</button></div>" ;

    html +=
"<div><h3>AZUL</h3><div class='state' id='a'>OFF</di
v>"
;
    html += "<button class='btn on' onclick= \"
cmd('a','1') \">ON</button>" ;
    html += "<button class='btn off' onclick= \"
cmd('a','0') \">OFF</button></div>" ;

    html +=
"<div><h3>AMARILLO</h3><div class='state' id='y'>OFF
</div>"
;
    html += "<button class='btn on' onclick= \"
cmd('y','1') \">ON</button>" ;
    html += "<button class='btn off' onclick= \"
cmd('y','0') \">OFF</button></div>" ;

    html +=
"<div><h3>VERDE</h3><div class='state' id='g'>OFF</d
iv>"
;
    html += "<button class='btn on' onclick= \"
cmd('g','1') \">ON</button>" ;
    html += "<button class='btn off' onclick= \"
cmd('g','0') \">OFF</button></div>" ;

    html += "<div style='margin-top:20px;'>" ;
    html +=
"<button onclick='update()'>Actualizar</button>"  ;
    html += "<button onclick= \"cmd('all','0') \"
>Apagar Todos</button></div>" ;

    html += "<script>" ;
    html +=
"function cmd(l,s){fetch('/ctrl?l='+l+'&s='+s).then
(()=>update());}"
;
    html +=
"function update(){fetch('/state').then(r=>r.json
()).then(d=>{"
;
    html +=
"document.getElementById('r').textContent=d.r?'O
N':'OFF';"
;
    html +=
"document.getElementById('a').textContent=d.a?'O
N':'OFF';"
;
    html +=
"document.getElementById('y').textContent=d.y?'O
N':'OFF';"
;
    html +=
"document.getElementById('g').textContent=d.g?'O
N':'OFF';});}"
;
    html += "setInterval(update,2000);update();"  ;
    html += "</script></body></html>" ;

    server.send(200, "text/html" , html);
}

void handleControl()
{
    String  led = server.arg("l");
    String  state = server.arg("s");

    if (led == "r")
        digitalWrite(rojo, state == "1");
    else if (led == "a")
        digitalWrite(azul, state == "1");
    else if (led == "y")
        digitalWrite(amarillo, state == "1");
    else if (led == "g")
        digitalWrite(verde, state == "1");
    else if (led == "all")
        allOff();

    server.send(200, "text/plain" , "OK");
}

void handleEstado()
{
    String  json = "{\"r\":";
    json += digitalRead(rojo) ? "true" : "false";
    json += ",\"a\":";
    json += digitalRead(azul) ? "true" : "false";
    json += ",\"y\":";
    json += digitalRead(amarillo) ? "true" :
"false";
    json += ",\"g\":";
    json += digitalRead(verde) ? "true" : "false";
    json += "}";

    server.send(200, "application/json" , json);
}
```





```
#include <ESP32Servo.h>

const int pinPotenciometro = 34;
const int pinServo = 13;
const int led1 = 2;    // LED para 0-60 grados
const int led2 = 4;    // LED para 61-120 grados
const int led3 = 5;    // LED para 121-180 grados

Servo miServo;

void setup() {
  Serial.begin(115200);
  miServo.attach(pinServo);

  // Configurar pines de LEDs como salidas
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
  pinMode(led3, OUTPUT);

  // Asegurar que todos los LEDs empiecen apagados
  digitalWrite(led1, LOW);
  digitalWrite(led2, LOW);
  digitalWrite(led3, LOW);

  Serial.println(
"Control Servo con Potenciometro - ESP32" );
  Serial.println("Conexiones:");
  Serial.println(
"  Potenciometro: 3.3V, GPIO34, GND" );
  Serial.println("  Servo: 3.3V, GPIO13, GND" );
  Serial.println("  LED1 (0-60): GPIO2" );
  Serial.println("  LED2 (61-120): GPIO4" );
  Serial.println("  LED3 (121-180): GPIO5" );
}

void loop() {
  int valorPot = analogRead(pinPotenciometro);
  int angulo = map(valorPot, 0, 4095, 0, 180);

  // Mover servo
  miServo.write(angulo);

  // Apagar todos los LEDs primero
  digitalWrite(led1, LOW);
  digitalWrite(led2, LOW);
  digitalWrite(led3, LOW);

  // Encender solo el LED correspondiente al rango
  if (angulo >= 0 && angulo <= 60) {
    digitalWrite(led1, HIGH);
   // Encender LED1 (GPIO2) para 0-60
  }
  else if (angulo >= 61 && angulo <= 120) {
    digitalWrite(led2, HIGH);
   // Encender LED2 (GPIO4) para 61-120
  }
  else if (angulo >= 121 && angulo <= 180) {
    digitalWrite(led3, HIGH);
   // Encender LED3 (GPIO5) para 121-180
  }

  // Mostrar información
  Serial.print("Pot: ");
  Serial.print(valorPot);
  Serial.print(" | Angulo: ");
  Serial.print(angulo);
  Serial.print(" grados | LED: ");

  if (angulo <= 60) {
    Serial.println("1 (GPIO2) - Rango: 0-60" );
  } else if (angulo <= 120) {
    Serial.println("2 (GPIO4) - Rango: 61-120" );
  } else {
    Serial.println("3 (GPIO5) - Rango: 121-180" );
  }

  delay(50);
}
```



Control LEDs

ROJO

OFF

AZUL

OFF

AMARILLO

OFF

VERDE

OFF