



**TECNOLÓGICO
NACIONAL DE MÉXICO**
INSTITUTO TECNOLÓGICO DE TIJUANA



INSTITUTO TECNOLÓGICO DE TIJUANA

SUBDIRECCIÓN ACADÉMICA

DEPARTAMENTO DE SISTEMAS Y COMPUTACIÓN

SEMESTRE ENERO JUNIO - 2024

INGENIERÍA EN SISTEMAS COMPUTACIONALES

LENGUAJES DE INTERFAZ

PRÁCTICA 3 - DOCUMENTACIÓN

JULIO ALEJANDRO HERNÁNDEZ LEÓN - 21211963
SANTY FRANCISCO MARTINEZ CASTELLANOS - 21211989
LUIS ROBERTO LEAL LUA - 21211970

RENÉ SOLIS REYES

Índice

Código Utilizado y Descripción.....	3
Fotografías de Página Funcional.....	8

Código Utilizado y Descripción

```
//Elaborado por Team Debian
//Martinez Castellanos
//Hernandez Leon
//Leal Lua

#include <WiFi.h>
//Datos sobre el internet, Nombre y Contraseña.
#ifndef STASSID
#define STASSID "INFINITUM5A66_2.4"
#define STAPSK "duke1975"
#endif

const char* ssid = STASSID;
const char* password = STAPSK;

int port = 8080;

WiFiServer server(port);

int ledSpeed = 1000; // Velocidad predeterminada del LED en milisegundos
bool ledState = false; // Estado actual del LED

void setup() {
  Serial.begin(115200);
  pinMode(LED_BUILTIN, OUTPUT);
  WiFi.mode(WIFI_STA);
  WiFi.setHostname("PicoW2");
  Serial.printf("Connecting to '%s' with '%s'\n", ssid, password);
  WiFi.begin(ssid, password);
```

```

while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(100);
}

    Serial.printf("\nConnected to WiFi\n\nConnect to server at %s:%d\n",
WiFi.localIP().toString().c_str(), port);

    server.begin();
}

const char* htmlContent = R"(
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>PicoW WebServer</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            text-align: center;
            background-color: #fff; /* Fondo blanco */
            color: #2c3e50; /* Texto oscuro */
            margin: 0; /* Eliminar margen predeterminado */
            padding: 0; /* Eliminar relleno predeterminado */
        }
        h1 {
            font-size: 36px; /* Tamaño grande de fuente */
            margin-top: 50px; /* Espacio superior */
            margin-bottom: 20px; /* Espacio inferior */
            font-weight: bold;

```

```

    }
    button {
        font-size: 16px;
        padding: 10px;
        margin: 10px;
        background-color: #85a8ff;
        color: #fff;
        border: none;
        border-radius: 5px;
        cursor: pointer;
    }
    .button-container {
        display: flex;
        justify-content: center;
        flex-wrap: wrap;
    }
</style>
</head>
<body>
    <h1>Team Debian</h1>
    <div class="button-container">
        <button onclick='turnOnLED()'>Encender</button>
        <button onclick='turnOffLED()'>Apagar</button>
        <button onclick='blinkLED()'>Parpadear</button>
    </div>
    <script>
        function turnOnLED() {
            // Encender
            var xhr = new XMLHttpRequest();
            xhr.open("GET", "/turnon", true);
            xhr.send();

```

```

    }
    function turnOffLED() {
        // Apagar
        var xhr = new XMLHttpRequest();
        xhr.open("GET", "/turnoff", true);
        xhr.send();
    }
    function blinkLED() {
        // Parpadear
        var xhr = new XMLHttpRequest();
        xhr.open("GET", "/blink", true);
        xhr.send();
    }
</script>
</body>
</html>
)";

void loop() {
    WiFiClient client = server.accept();
    if (client) {
        Serial.println("Nuevo cliente conectado.");
        String req = client.readStringUntil('\r');
        Serial.println(req);
        client.flush();

        if (req.indexOf("/turnon") != -1) {

            ledState = true;
            digitalWrite(LED_BUILTIN, HIGH);
        } else if (req.indexOf("/turnoff") != -1) {

```

```

        ledState = false;
        digitalWrite(LED_BUILTIN, LOW);
    } else if (req.indexOf("/blink") != -1) {

        for (int i = 0; i < 5; i++) {
            digitalWrite(LED_BUILTIN, HIGH);
            delay(500);
            digitalWrite(LED_BUILTIN, LOW);
            delay(500);
        }
    }

    client.println("HTTP/1.1 200 OK");
    client.println("Content-Type: text/html");
    client.println("Connection: close");
    client.println();

    client.print(htmlContent);

    client.stop();
    Serial.println("Cliente desconectado.");
}
}

```

Fotografías de Página Funcional

