

Materiales:

NodeMCU y cable USB.

Fotorresistencia LDR y resistencia de 1Kohmio

Podemos crear un servidor WEB trabajando el NodeMCU como AP o como estación. Los comando más importantes que debemos escribir son:

WiFiServer server(80); que crea un servidor que escucha las conexiones entrantes en el puerto 80 en el caso de un servidor WEB.

server.begin(); Indica al servidor que empiece a escuchar las conexiones entrantes, todo esto funcionando a través del protocolo http.

Para el caso de un AP como servidor web el código sería el siguiente:

```
#include <ESP8266WiFi.h>
WiFiServer server(80);
void setup() {
  Serial.begin(9600);
  WiFi.softAP("PRUEBA", "ricardo901");
  server.begin();
}

void loop(){

  WiFiClient client = server.available();
  if (!client) { return; }           //Chequea que el cliente esté conectado.
  while(!client.available()){ delay(1); } //Espera que el cliente envíe un dato.
  Serial.println((client.readStringUntil("\r")));
  client.println("HTTP/1.1 200 OK");
  client.println("Content-Type: text/html");
  client.println("");
  client.println("<!DOCTYPE HTML>");
  client.println("<html>");
  client.println("BIENVENIDOS");
  client.println("</html>");
  client.stop();
  delay(1);
}
```

En este caso utilizamos html para agregar contenido cuadros o botones en la página WEB. También se puede utilizar CSS para agregar estilo y colores a la página.

Vamos a incluir la lectura de un puerto como por ejemplo, la lectura tomada en el puerto análogo A0.

```
#include <ESP8266WiFi.h>
WiFiServer server(80);
void setup() {
  Serial.begin(9600);
  delay(100);
  WiFi.disconnect();
}
```

```

WiFi.softAP("PRUEBA1", "annie901");
server.begin();
}

void loop() {

  WiFiClient client = server.available();
  if (!client) {
    return;          //Chequea que el cliente esté conectado.
  }
  while (!client.available()) {
    delay(1);        //Espera que el cliente envíe un dato.
  }

  int valor = analogRead(A0);
  Serial.println(client.readStringUntil("\r"));
  client.println("HTTP/1.1 200 OK");
  client.println("Content-Type: text/html");

  client.println("Refresh: 5"); // Refresca la página WEB cada 5 segundos
  client.println();

  client.println("");
  client.println("<!DOCTYPE HTML>");

  client.println("<html>");
  client.println("BIENVENIDOS");
  client.println("<br>La temperatura es:<br>");
  client.println(valor);
  client.println("</html>");
  client.stop();
  delay(10);
}

```

Para refrescar automáticamente la página html y se actualice cada 5 segundo el valor medido en el puerto A0 se puede hacer incluyendo el comando: **client.println("Refresh: 5");**

Ahora como trabajo de clase, programe el dispositivo para que sea una estación WiFi y servidor WEB dentro de una red LAN inalámbrica, que transmita el valor de intensidad de luz medida por un LDR a través del puerto A0. Recuerde que para la configuración como estación WiFi deberá conocer el SSID y password de la red inalámbrica. Presente a través del monitor serial el valor del puerto A0 y la dirección IP utilizada por el NodeMCU.

Desde cualquier dispositivo conectado a la red debería verse a través de la página WEB el valor de medida de A0 presentado en una tabla de dos celdas como se muestra a continuación:

MEDIDA DE SENSOR DE LUZ LDR

Nivel de LUZ	50
--------------	----