

Estación de clima con uso de NodeMCU V1.0 y sensor DHT11

Universidad Distrital Francisco José de Caldas (Agosto de 2020)

Alvaro Alejandro Zarabanda Gutierrez
aazarabandag@correo.udistrital.edu.co
 20161020507

Resumen – Las comunicaciones, cada día que pasa son mas indispensables para la humanidad, debido a esto, a medida que pasa el tiempo estas van evolucionando y acomodándose a las necesidades, haciendo uso de nuevas tecnologías y haciendo optimización de recursos y ancho de banda disponible, esto gracias a la codificación. En esta ocasión, se hizo una práctica con NodeMCU y el sensor DHT11 para implementar un servidor web sencillo que muestre los datos de temperatura y humedad de un lugar en tiempo real, explicando los pasos y protocolos que se utilizan para realizar el proyecto.

Índice de Términos – Cliente, Comunicaciones, Direccion IP, Gateway, IoT, Puerto, Router, Sensor, Servidor, Trama, Transmisión, Wi-Fi.

I. INTRODUCCION

Este documento, abordara la teoría acerca del funcionamiento del modulo NodeMCU ESP8266, junto con el sensor DHT11 para la obtención de la temperatura y humedad. Con esto. Se hará un enfoque hacia las comunicaciones, incluyendo así temas de como se hace el transporte de los datos del sensor hacia el modulo y como es el funcionamiento del protocolo sobre el que funciona el servidor del ESP 8266.

Por otro lado, se hará uso de redirección de puertos en el router para que el servidor web sea accesible desde cualquier parte del mundo con el uso de la IP publica asignada por el ISP.

Ya teniendo en cuenta lo anterior, se mencionarán los pasos de cómo se implementó el servidor, y así de esta forma dejar el proyecto abierto a futuras modificaciones hechas por terceros.

II. MARCO TEÓRICO

A. ¿Que es ESP8266?

Esp8266 es un chip Wi-fi de bajo costo que funciona mediante protocolo TCP/IP además incluye un microcontrolador que es el encargado de manejar dicho protocolo y un software necesario para la conexión 802.11, posee entradas y salidas digitales de propósito general (GPIO) así como una entrada analógica (ADC de 10 bit). [1]



Fig. 1. ESP8266

B. ¿Qué es NodeMCU?

NodeMCU es un proyecto open-source para el desarrollo de un modelo sencillo de integrar IoT(Internet de las Things), para este fin, se desarrollan modelos de hardware y software que facilita el desarrollo de programas y aplicaciones basados en Wi-Fi, integrando un módulo ESP8266. En este módulo su programación se hace totalmente transparente al no requerir ningún cambio en sus pines para la programación al disponer de conexión USB al igual que Arduino, la placa dispone de un regulador de alimentación, así como un chip USB serial para la comunicación del ESP8266 directamente desde el computador,

además de un acceso fácil a los pines GPIO y una memoria flash de 1Mb para el almacenamiento de varios programas. [1]

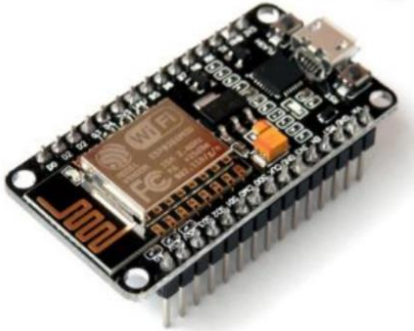


Fig. 2. NodeMCU con ESP8266

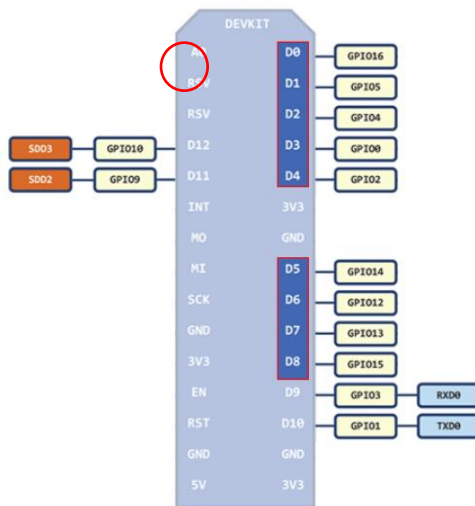


Fig 3. Pines utilizables NodeMCU

C. Sensor DHT11

El sensor DHT11 es un sensor digital de temperatura y humedad relativa del aire. Este ya posee un conversor de datos Analógico a Digital, de modo que solo es necesario conectarlo a un pin digital de nuestro NodeMCU. [2]

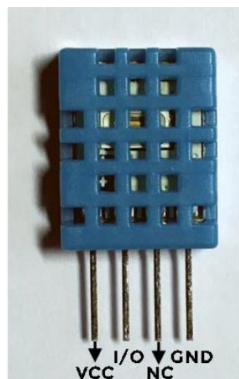


Fig. 4. Sensor DHT11

A continuación, se mostrarán sus especificaciones técnicas:

TABLA I
DATOS TÉCNICOS DHT11

Ítem	Descripción
Modelo	DHT11
Fuente de alimentación	3.3-5v DC
Rango de Funcionamiento	-20 a 60 °C
Precisión	<ul style="list-style-type: none"> • Humedad $\pm 2\%$ HR (Máx. $\pm 5\%$ HR) • Temperatura: $\pm 2^\circ\text{C}$

D. Redirección de puertos

La redirección de puertos, a veces llamado tunelado de puertos, es la acción de redirigir un puerto de red de un nodo de red a otro. Esta técnica puede permitir que un usuario externo tenga acceso a un puerto en una dirección IP privada (dentro de una LAN) desde el exterior vía un router con NAT activado. [3]

III. SERVIDOR WEB ESP8266

Para hacer la implementación del servidor web, se hace uso de la clase `ESP8266WebServer.h`, la cual tiene todos los métodos necesarios para manejar solicitudes HTTP. [4]

Para ello, se hará una instancia de la clase ya mencionada para así tener nuestro objeto servidor, al cual se le pasará como parámetro el puerto por el que se va a ejecutar, tal como se muestra en el siguiente ejemplo:

```
ESP8266WebServer server(80);
```

Otros datos a tener en cuenta, son los métodos mas comunes que tiene esta clase, los cuales son:

```
void begin();
```

Inicia el servidor

```
void handleClient();
```

Maneja las solicitudes entrantes de los clientes

```
void close();
void stop();
```

Deshabilitan el servidor

```
void on();
```

Genera los ficheros en el servidor

```
void send();
```

Envío de respuestas al cliente

```
void addHandler();
void onNotFound();
```

Controladores de solicitudes de cliente

Como podemos observar es una librería bastante útil para hacer montaje de servidores muy sencillos, sin embargo, es una alternativa bastante limitada en comparación con otros grandes servidores como Apache o Nginx, pero cabe recalcar que ese proyecto cuenta con licencia GNU Lesser General Public License v2.1, con la cual se pueden hacer futuras mejoras y modificaciones.

IV. MONTAJE DEL SISTEMA

A. Materiales

- NodeMCU
- DHT11
- Jumpers
- Resistencia de 330Ω
- Protoboard

B. Conexiones en protoboard

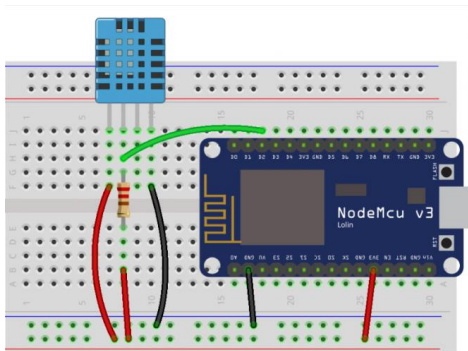


Fig. 5. Conexiones [5]

C. Código

En cuanto al código, se hizo uso de Arduino IDE para desarrollarlo. En el documento presente no se hará mucho enfoque en la explicación de este, sin embargo, en el siguiente link puede visualizarlo:

- https://github.com/aalejo25/Proyecto_Final_Netes/blob/master/Servidor_Clima/Servidor_Clima.ino

V. TRANSMISIÓN DE DATOS POR PARTE DEL SENSOR DHT11

Como ya se había mencionado al principio, el sensor ya incluye un convertidor de señal analógica a digital, esto quiere decir que el sensor ya hará el envío de datos al microcontrolador en formato digital. Para efectuar eso, se hace uso de una trama de

40 bits, los cuales se dividen en la siguiente información:

0011 0101 0000 0000 0001 1000 0000 0000 0100 1001
8 bits humedad 8 bits humedad 8 bits temperatura 8 bits temperatura 8 bits de paridad

Fig. 6. Trama de datos DHT11

El primer grupo de 8-bit es la parte entera de la humedad y el segundo grupo la parte decimal. Lo mismo ocurre con el tercer y cuarto grupo, la parte entera de la temperatura y la parte decimal. Por último, los bits de paridad para confirmar que no hay datos corruptos. [6]

Así mismo, tenemos unos bits de paridad, los cuales ayudan a asegurarnos de que la información sea correcta, para esto, la suma de los primeros 4 grupos de bits deben dar como resultado los bits de paridad.

$$0011\ 0101 + 0000\ 0000 + 0001\ 1000 + 0000\ 0000 = 0100\ 1101$$

Fig. 7. Comprobación bits de paridad

VI. CONFIGURACIÓN DE REDIRECCIÓN DE PUERTOS

Hacer el uso de redirección de puertos, da muchas ventajas en cuanto al manejo remoto o visualización de dispositivos, teniendo esta ventaja, podemos aplicar esto para ver de forma remota el servidor que se montó. Para hacer esto hay una serie de pasos sencillos:

A. Hallar dirección de Gateway

Para esto se debe tener en cuenta el sistema operativo que se está usando, en caso de Windows, se tiene que abrir el CMD(Command Prompt), y hacer uso del comando ipconfig.

Por otro lado, en caso de Linux, se debe abrir la terminal y hacer uso del comando route.

```
Adaptador de LAN inalámbrica Wi-Fi:

Sufijo DNS específico para la conexión. . . :
Vínculo: dirección IPv6 local. . . : fe80::d81e:f011:b8bc:f0f5%8
Dirección IPv4. . . . . : 192.168.0.221
Máscara de subred. . . . . : 255.255.255.0
Puerta de enlace predeterminada. . . . . : 192.168.0.1
```

Fig. 6. Comando ipconfig

B. Acceder a la configuración GUI del router

Ya teniendo la dirección de Gateway, desde cualquier navegador se puede acceder a la configuración, esto poniendo la dirección en la barra de búsqueda.

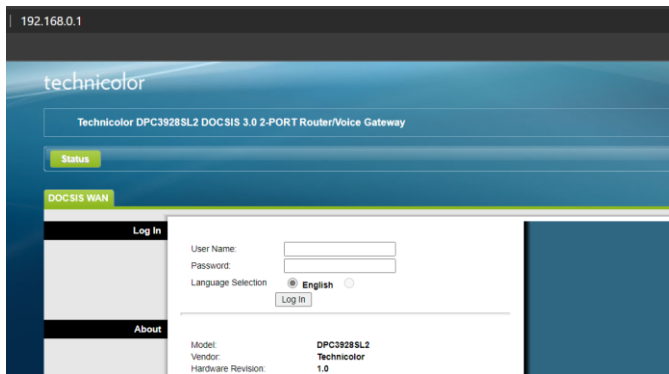


Fig. 7. GUI Router

En la mayoría de casos, el usuario y la contraseña son las que están por defecto dadas por el fabricante.

C. Configuración de redirección de puertos

En la interfaz de usuario, generalmente se debe buscar un apartado que dice Port Forwarding, donde se debe configurar la IP local del servidor y los puertos correspondientes.

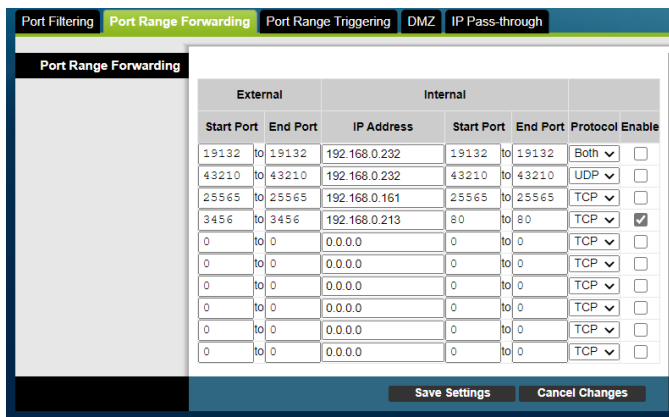


Fig. 8. Port Forwarding

Ya con esto, desde cualquier parte se puede acceder con IP pública.

VII. CONCLUSION

Con este proyecto, nos podemos dar cuenta de los beneficios que se puede obtener con este montaje, y podemos darnos cuenta que el IoT promete abrir las puertas a un mundo revolucionario donde todos los objetos estén conectados entre sí y permitir una interacción de estos con el entorno para generar una mejor experiencia al usuario.

VIII. REFERENCIAS

- [1] L. d. V. Hernández, «programarfacil.com,» 19 Agosto 2020. [En línea]. Available: <https://programarfacil.com/podcast/nodemcu-tutorial-paso-a-paso/>.
- [2] «Tecno Recursos,» Agosto 2020. [En línea]. Available: <https://www.picuino.com/es/arduprog/sensor-dht11.html#:~:text=El%20sensor%20DHT11%20es%20un,anal%C3%B3gico%20para%20realizar%20las%20lecturas..> [Último acceso: 19 Agosto 2020].
- [3] «Wikipedia,» [En línea]. Available: https://es.wikipedia.org/wiki/Redirecci%C3%B3n_de_puertos. [Último acceso: 19 Agosto 2020].
- [4] «github,» [En línea]. Available: <https://github.com/esp8266/Arduino/tree/master/libraries/ESP8266WebServer>. [Último acceso: 19 Agosto 2020].
- [5] «Prometec,» [En línea]. Available: <https://www.prometec.net/dht11-web-server-esp8266/>. [Último acceso: 19 Agosto 2020].
- [6] L. d. V. Hernández, «programarfacil.com,» [En línea]. Available: <https://programarfacil.com/blog/arduino-blog/sensor-dht11-temperatura-humedad-arduino/>. [Último acceso: 19 Agosto 2020].