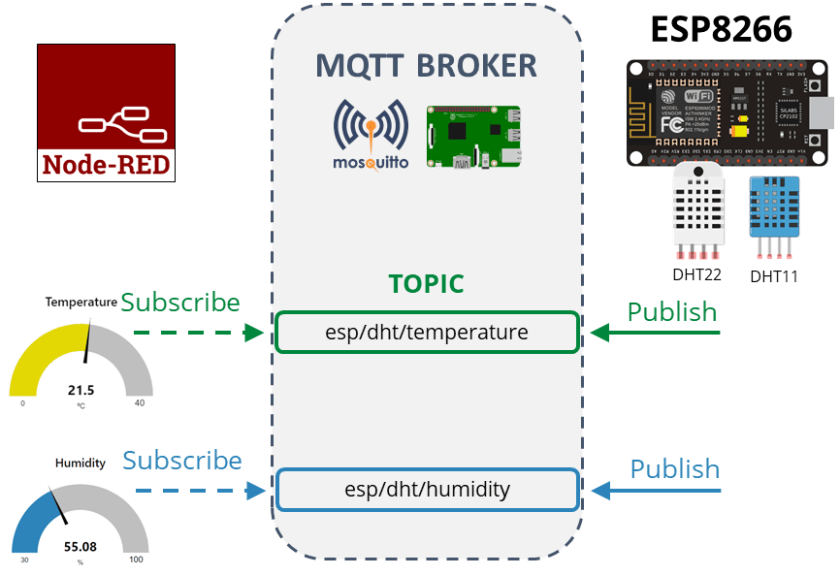
**Aprenda como publicar leituras de temperatura e umidade de um sensor DHT11 ou DHT22 via MQTT com o ESP8266 NodeMCU em qualquer plataforma que suporte MQTT ou qualquer cliente MQTT. Como exemplo, publicaremos as leituras dos sensores no painel Node-RED e o ESP8266 será programado usando o Arduino IDE.**

**Visão Geral do Projeto**

O diagrama a seguir mostra uma visão geral de alto nível do projeto que criaremos.



• O ESP8266 solicita leituras de temperatura e umidade do sensor DHT11 ou DHT22;

• As leituras de temperatura são publicadas no tópico esp / dht / temperature;

• As leituras de umidade são publicadas no tópico esp / dht / umidade;

• O Node-RED está inscrito nesses tópicos;

• O Node-RED recebe as leituras do sensor e as exibe nos medidores;

• Você pode receber as leituras em qualquer outra plataforma que suporte MQTT e manipular as leituras conforme desejar.

**Pré-requisitos**

Antes de continuar com este tutorial, verifique os seguintes pré-requisitos.

• Arduino IDE: Como programamos o ESP8266 usando o Arduino IDE, verifique se o complemento do ESP8266 está instalado.

• MQTT Broker: Para usar o MQTT, você precisa de um broker. Usaremos o broker Mosquitto instalado em um Raspberry Pi. É possível usar qualquer outro broker MQTT, incluindo um broker MQTT em nuvem.

• Bibliotecas MQTT: Para usar o MQTT com o ESP8266, usaremos a Biblioteca do cliente Async MQTT. Instalação de Async MQTT Client Library:

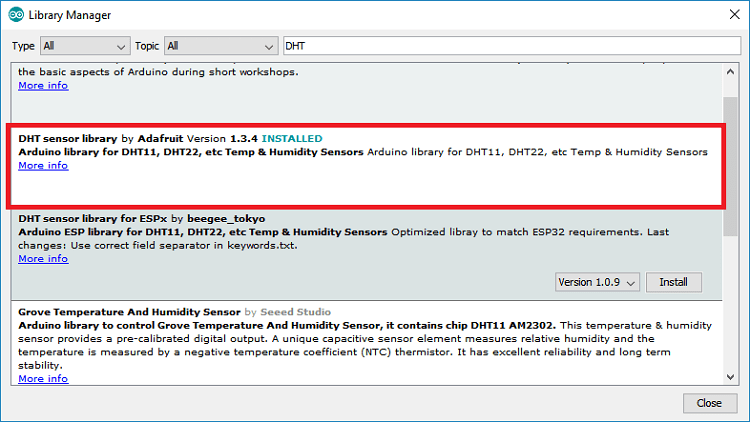
1. Clique no link: <https://github.com/marvinroger/async-mqtt-client/archive/master.zip> . Você deve ter uma pasta .zip na pasta Downloads.
2. Descompacte a pasta .zip e você deve obter a pasta async-mqtt-client-master.
3. Renomeie sua pasta de async-mqtt-client-master para async\_mqtt\_client.
4. Mova a pasta async\_mqtt\_client para a pasta de bibliotecas de instalação do Arduino IDE.
5. Por fim, abra novamente seu IDE do Arduino.

Como alternativa, você pode ir para Esboço> Incluir biblioteca> Adicionar. ZIP e selecione a biblioteca que você acabou de baixar.

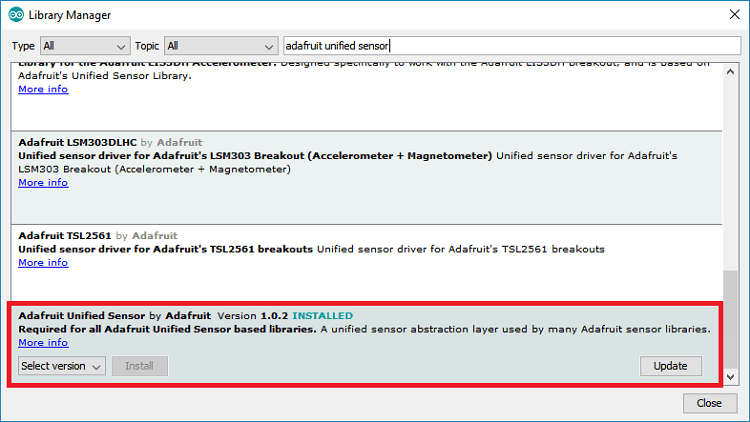
• DHT Sensor Libraries: Para ler a partir do sensor DHT, usaremos a biblioteca DHT da Adafruit. Para usar esta biblioteca, você também precisa instalar a biblioteca Adafruit Unified Sensor. Siga as próximas etapas para instalar essas bibliotecas.

1. Abra o Arduino IDE e vá para Sketch> Include Library> Manage Libraries. O Gerenciador de bibliotecas deve abrir.

2. Procure “DHT” na caixa Pesquisar e instale a biblioteca DHT a partir do Adafruit.

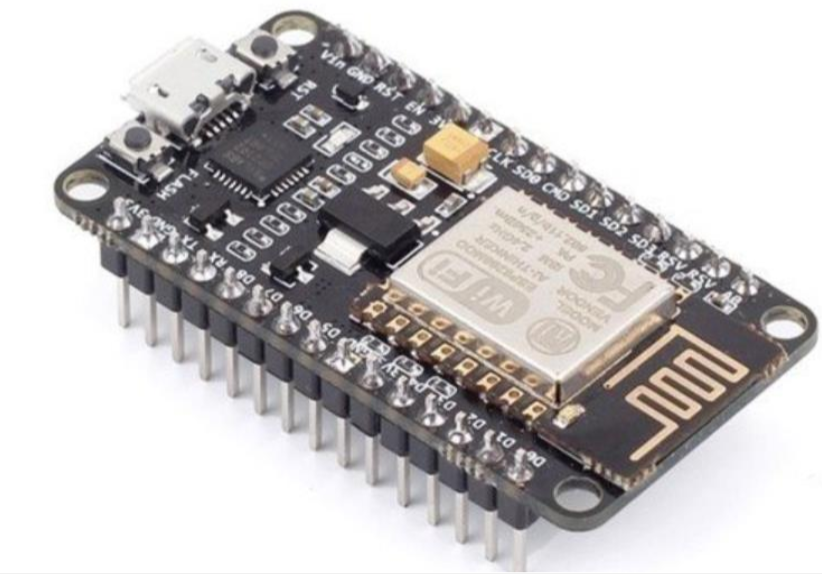


3. Depois de instalar a biblioteca DHT da Adafruit, digite "Adafruit Unified Sensor" na caixa de pesquisa. Role todo o caminho para encontrar a biblioteca e instalá-la.



• Para este tutorial, você precisa das seguintes partes:

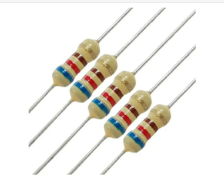
- ESP8266:

****

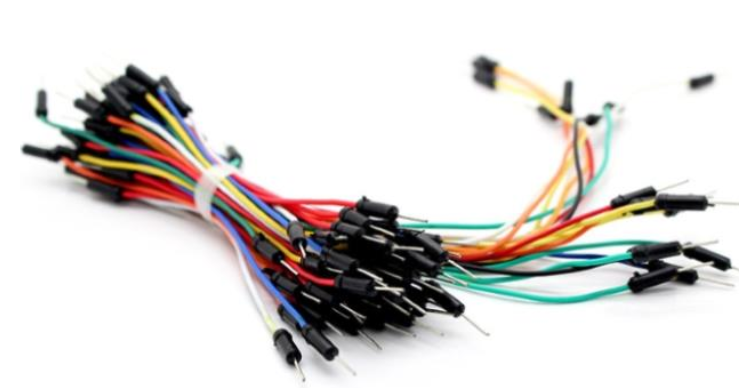
- DHT11:

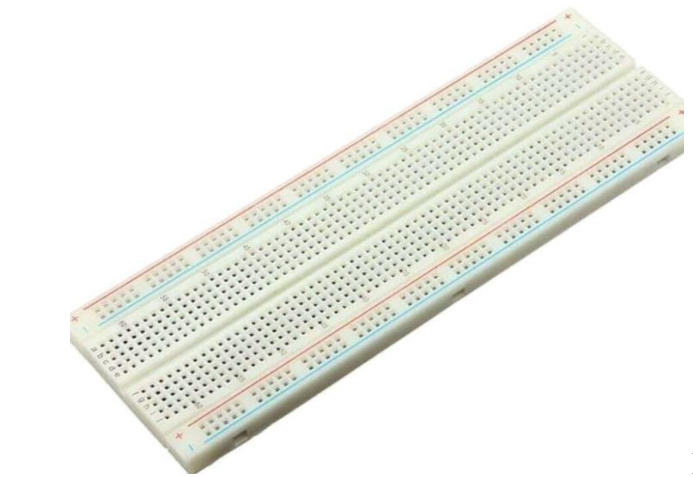


- Resistor 4.7k Ohm:



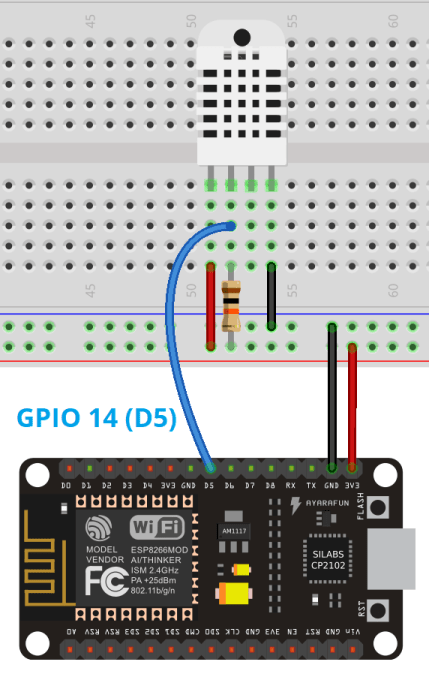
- Cabos e jumpers:

****

- Protoboard:****

**Diagrama esquemático**

Conecte o DHT11 ou DHT22 ao ESP8266, conforme mostrado no diagrama esquemático a seguir, com o pino de dados conectado ao GPIO 14.



**Nota:** se você possui um sensor DHT em uma placa de interrupção, ele vem com apenas três pinos e com um resistor de pull-up interno no pino 2, para que você não precise conectar o resistor. Você só precisa conectar VCC, dados e GND.

**Preparando Node-RED Dashboard**

O ESP8266 está publicando leituras de temperatura a cada 10 segundos nos tópicos esp / dht / temperature e esp / dht / umidade. Agora, você pode usar qualquer painel que suporte MQTT ou qualquer outro dispositivo que suporte MQTT para assinar esses tópicos e receber as leituras.

Como exemplo, criaremos um fluxo simples usando o Node-RED para assinar esses tópicos e exibir as leituras nos medidores.

Se você não possui o Node-RED instalado, siga os próximos tutoriais:

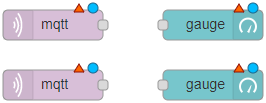
• <https://randomnerdtutorials.com/getting-started-with-node-red-on-raspberry-pi/>

• <https://randomnerdtutorials.com/getting-started-with-node-red-dashboard/>

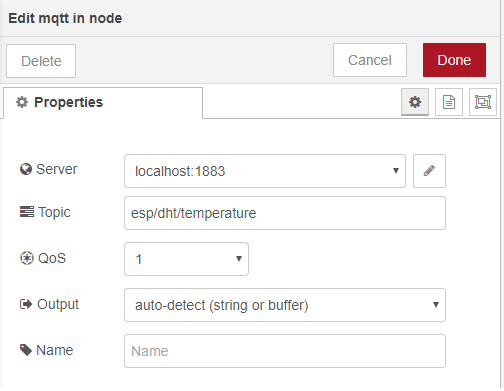
Tendo o Node-RED em execução no seu Raspberry Pi, vá para o seu endereço IP do Raspberry Pi seguido por: 1880.

<http://seu_endereco_ip:1880>

A interface Node-RED deve abrir. Arraste dois MQTT nos nós e dois nós do medidor para o fluxo.



Clique no node MQTT e edite suas propriedades.

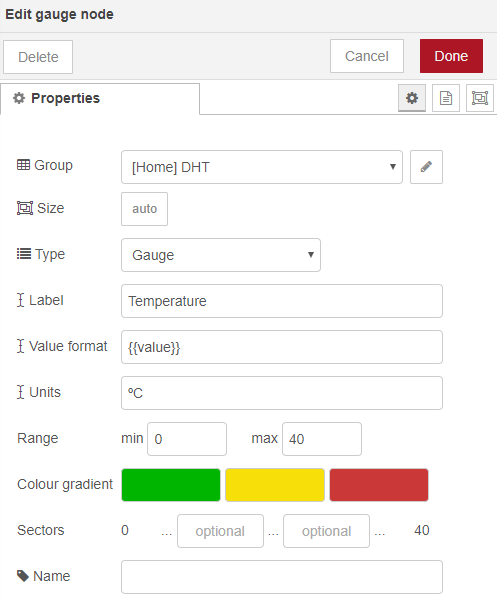


O campo Servidor refere-se ao intermediário MQTT. No nosso caso, o broker do MQTT é o Raspberry Pi, portanto está definido como localhost: 1883. Se você estiver usando um broker do Cloud MQTT, altere esse campo.

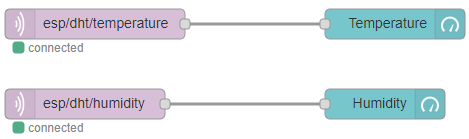
Insira o tópico no qual você deseja se inscrever e a QoS. Este nó MQTT anterior é inscrito no tópico esp / dht / temperature.

Clique no outro MQTT no nó e edite suas propriedades com o mesmo servidor, mas para o outro tópico: esp / dht / humidity.

Clique nos nós do medidor e edite suas propriedades para cada leitura. O seguinte nó é definido para as leituras de temperatura. Edite o outro nó do gráfico para as leituras de umidade.



Conecte seus nós como mostrado abaixo:



Por fim, implante seu fluxo (pressione o botão no canto superior direito).

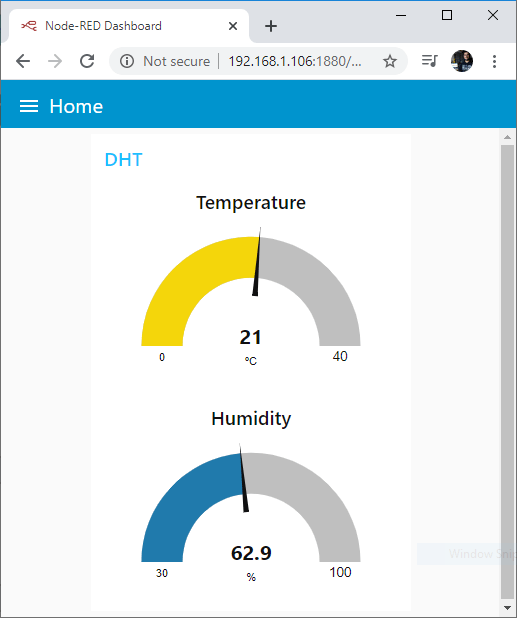
Deploy Node-RED button

**Demonstração**

Vá para o seu endereço IP do Raspberry Pi seguido por: 1880 / ui.

<http://seu_endereço_ip:1880/ui>

Você deve ter acesso às leituras atuais de temperatura e umidade do DHT no painel. Você pode usar outros nós do tipo painel para exibir as leituras de maneiras diferentes.



Pronto! Você tem sua placa ESP8266 publicando leituras de temperatura e umidade DHT no Node-RED via MQTT.