

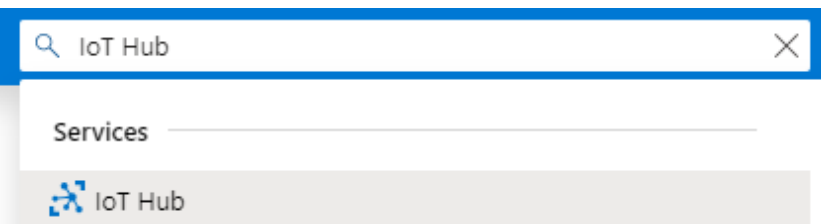
02 | Activación Servicio de Azure IoT Hub

<https://portal.azure.com/>

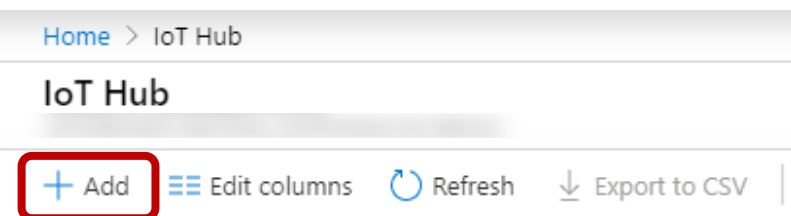


Creación del servicio y primer dispositivo

1. Buscar “IoT Hub” en el portal de Azure.



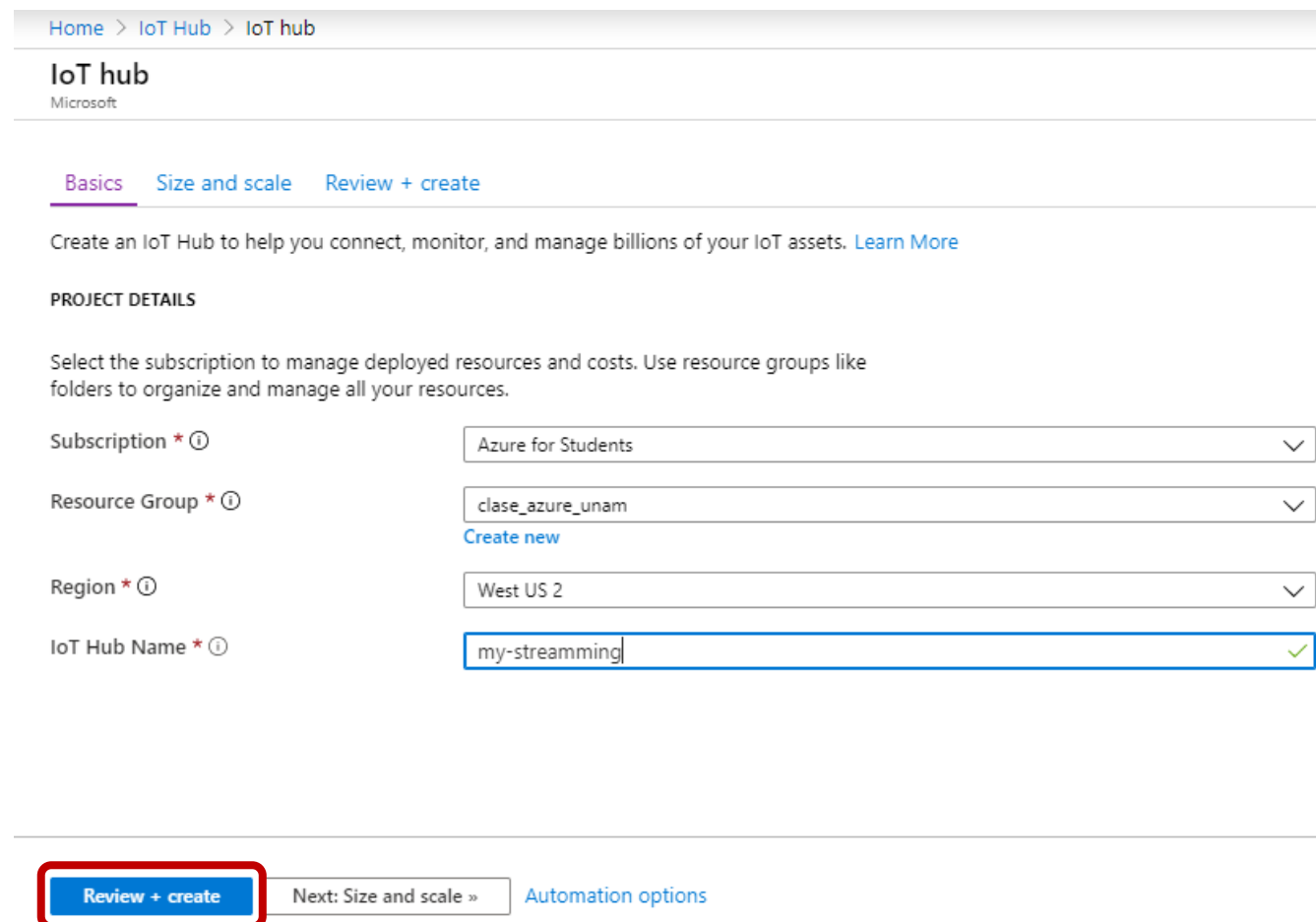
2. Seleccionar “add”



3. Llenar los campos con la información solicitada:

- Subscription
- Resource group
- Region
- IoT Hub Name

y seleccionar “Review + reate”.



4. Revisar la información del IoT Hub y seleccionar “créate”.

Home > IoT Hub > IoT hub

IoT hub

Microsoft

Basics

Size and scale

Review + create

BASICS

Subscription ⓘ

Azure for Students

Resource Group ⓘ

clase_azure_unam

Region ⓘ

West US 2

IoT Hub Name ⓘ

my-streamming

SIZE AND SCALE

Pricing and scale tier ⓘ

S1

Number of S1 IoT Hub units ⓘ

1

Messages per day ⓘ

400.000

Cost per month

482.50 MXN

Create

« Previous: Size and scale

Automation options

5. Esperar a que finalice el despliegue del servicio y seleccionar “go to resource”.

Home > Microsoft.IoTHub-1021122643 - Overview

Microsoft.IoTHub-1021122643 - Overview

Deployment

Search (Ctrl+J)

Delete Cancel Redeploy Refresh

Overview

Inputs

Outputs

Template

✓ Your deployment is complete

Deployment name: Microsoft.IoTHub-1021122643

Subscription: Azure for Students

Resource group: clase_azure_unam

Start time: 10/21/2019, 12:26:18 PM

Correlation ID:

Deployment details (Download)

Next steps

Go to resource

6. Buscar “IoT devices” y seleccionarlo.

Home > Microsoft.IoTHub-1021122643 - Overview > my-streamming

my-streamming

IoT Hub

IoT devices

Move Delete Refresh

Events

Explorers

IoT devices

Resource group (change)

clase_azure_unam

Status

Active

Current location

West US 2

Subscription (change)

Azure for Students

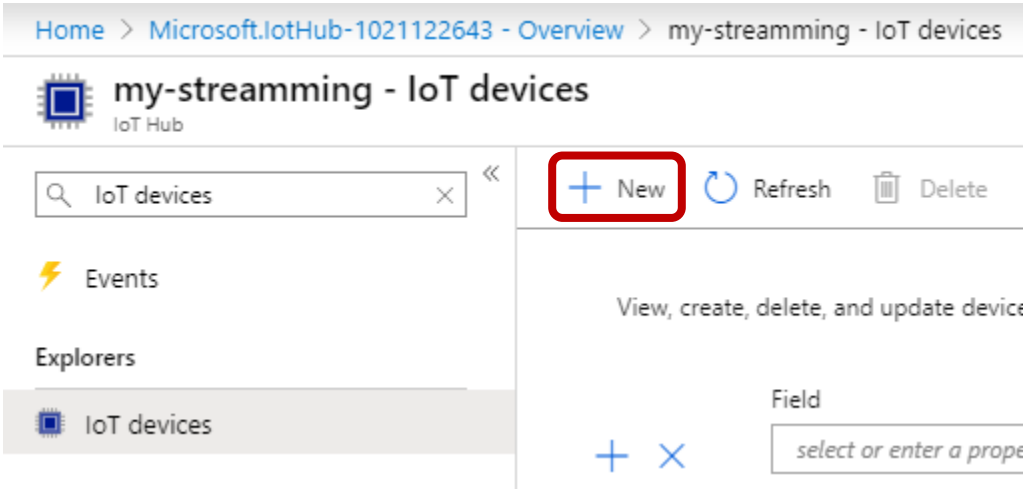
Subscription ID

8f7feb6a-a1d4-4ab0-b1ff-59afed5faf9a

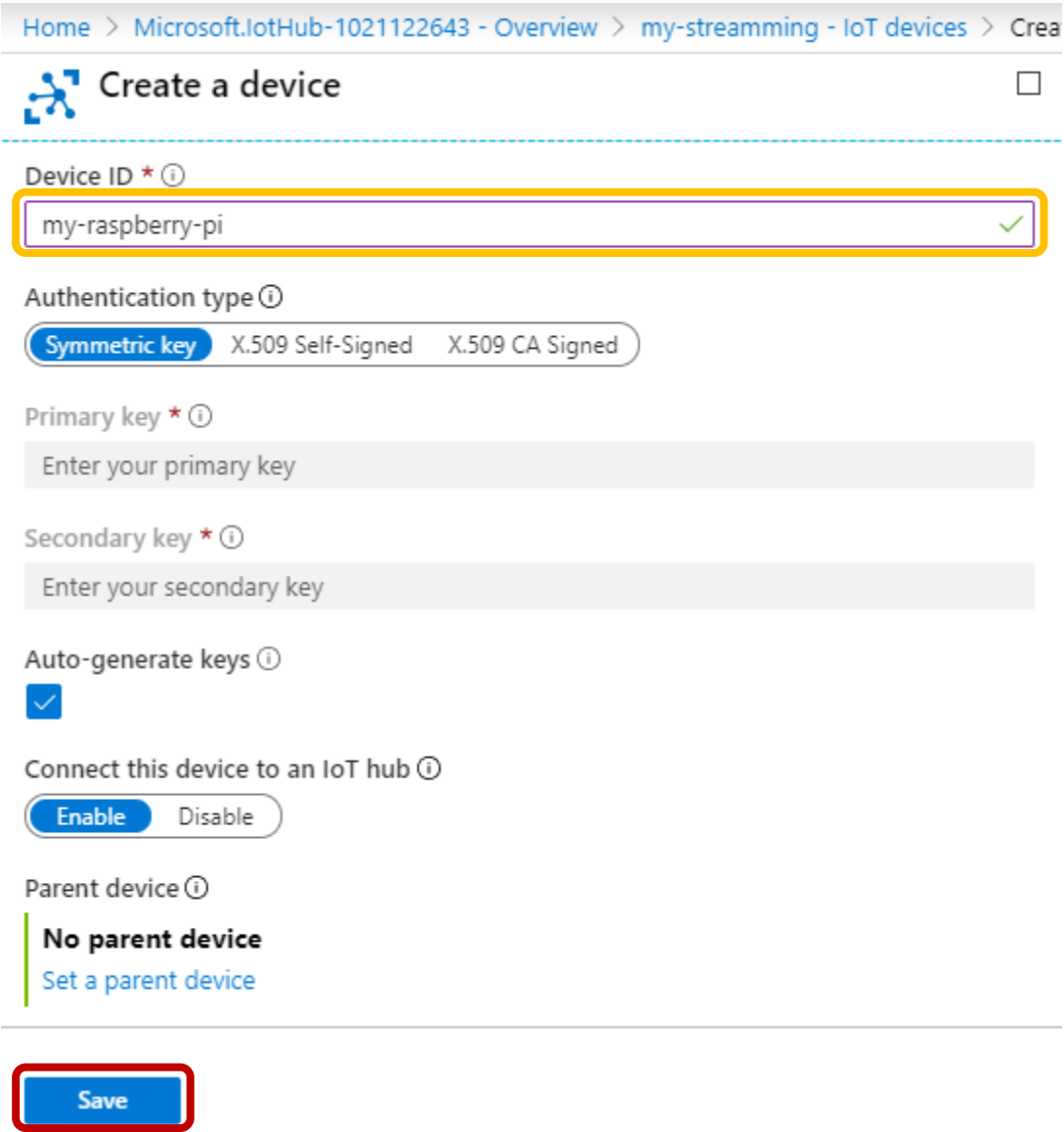
Tags (change)

Click here to add tags

7. Seleccionar “New”.




8. Llenamos únicamente el campo “Device ID” y le damos en “save”.



9. Confirmamos que se generó el nuevo dispositivo y lo seleccionamos.

Home > Microsoft.IoTHub-1021122643 - Overview > my-streamming - IoT devices

 my-streamming - IoT devices

IoT Hub

IoT devices

Events

Explorers

IoT devices

NewRefreshDelete

View, create, delete, and update devices in your IoT Hub.

+X

Field

select or enter a property name

Operator

=

Value

specify constraint value

+ Add a new clause

Query devices

Switch to query editor

DEVICE ID	STATUS	LAST ACTIVITY TIME (UTC)	LAST STATUS UPDA...	AUTHE...	CLOUD ...
my-raspberry-pi	Enabled	--	--	Sas	0

10. Mantenemos esta página abierta y que utilizaremos “Primary Connection String” en un paso futuro.

Home > Microsoft.IoTHub-1021122643 - Overview > my-streamming - IoT devices > my-raspberry-pi

my-raspberry-pi

my-streamming

Save

Message to Device

Direct Method

Add Module Identity

Device Twin

Manage keys

Refresh

Device ID ⓘ

Primary Key ⓘ

Secondary Key ⓘ

Primary Connection String ⓘ

Secondary Connection String ⓘ

Enable connection to IoT Hub ⓘ

Parent device ⓘ

Distributed Tracing (preview) ⓘ

[Learn more](#)

my-raspberry-pi

.....

.....

.....

.....

☒ Enable

☐ Disable

No parent device

Not configured

Module Identities

Configurations



Conexión del dispositivo con IoT hub

Para este tutorial utilizaremos el simulador de un Raspberry Pi, al cual podemos acceder en la siguiente dirección:

<https://azure-samples.github.io/raspberry-pi-web-simulator/>

Sin embargo, Azure tiene diversos tutoriales para conectar un dispositivo “físico” al IoT Hub. Algunos de estos son:

Raspberry Pi con Node.js

<https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-raspberry-pi-kit-node-get-started>

Raspberry Pi con C

<https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-raspberry-pi-kit-c-get-started>

DevKit AZ3166 con Arduino (MXChip)

<https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-arduino-iot-devkit-az3166-get-started>

11. Al acceder a la URL mencionada previamente aparecer un minitutorial, el cual vamos a cerrar.

Raspberry Pi Azure IoT Online Simulator

Ayuda Español ▼

1 Paso 1 Paso 2 Paso 3 X

Introducción al simulador web de Raspberry Pi

- Área de ensamblaje.** You can see your device status.
- Área de codificación.** An online code editor for you to make an app on Raspberry Pi with Node.js
- Ventana de consola integrada.** You can see the output of your app.

[El siguiente paso](#)

[Ver tutorial completo](#) [Ver código fuente](#)

```

1 /*
2 * IoT Hub Raspberry Pi Model B - Microsoft Sample Code - Copyright (c) 2017 - Licensed MIT
...
).Client;
').Message;
e-mqtt').Mqtt;

ADDRESS() // defaults to 0x77

device connection string'];

Mantenemos sus cambios en el editor aunque actualic
    
```

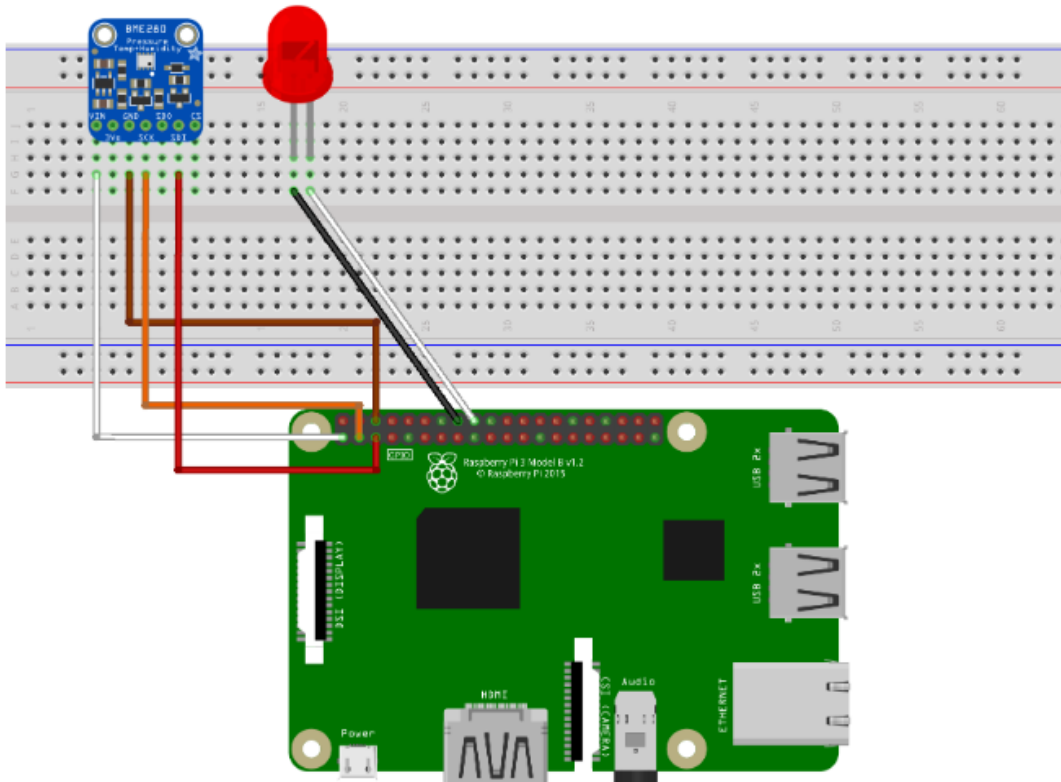
12. Ubicaremos la línea 15 donde está la variable “connectionString”



Raspberry Pi Azure IoT Online Simulator

Ayuda

Español ▼



```
1  /*
2  * IoT Hub Raspberry Pi NodeJS - Microsoft Sample Code - Copyright (c) 2017 - Licensed MIT
3  */
4  const wpi = require('wiring-pi');
5  const Client = require('azure-iot-device').Client;
6  const Message = require('azure-iot-device').Message;
7  const Protocol = require('azure-iot-device-mqtt').Mqtt;
8  const BME280 = require('bme280-sensor');
9
10 const BME280_OPTION = {
11   i2cBusNo: 1, // defaults to 1
12   i2cAddress: BME280.BME280_DEFAULT_I2C_ADDRESS() // defaults to 0x77
13 };
14
15 const connectionString = '[Your IoT hub device connection string]';
16 const LEDPin = 4;
17
18 var sendingMessage = false;
19 var messageId = 0;
20 var client, sensor;
21 var blinkLEDTIMEOUT = null;
22
23 function getMessage(cb) {
24
```

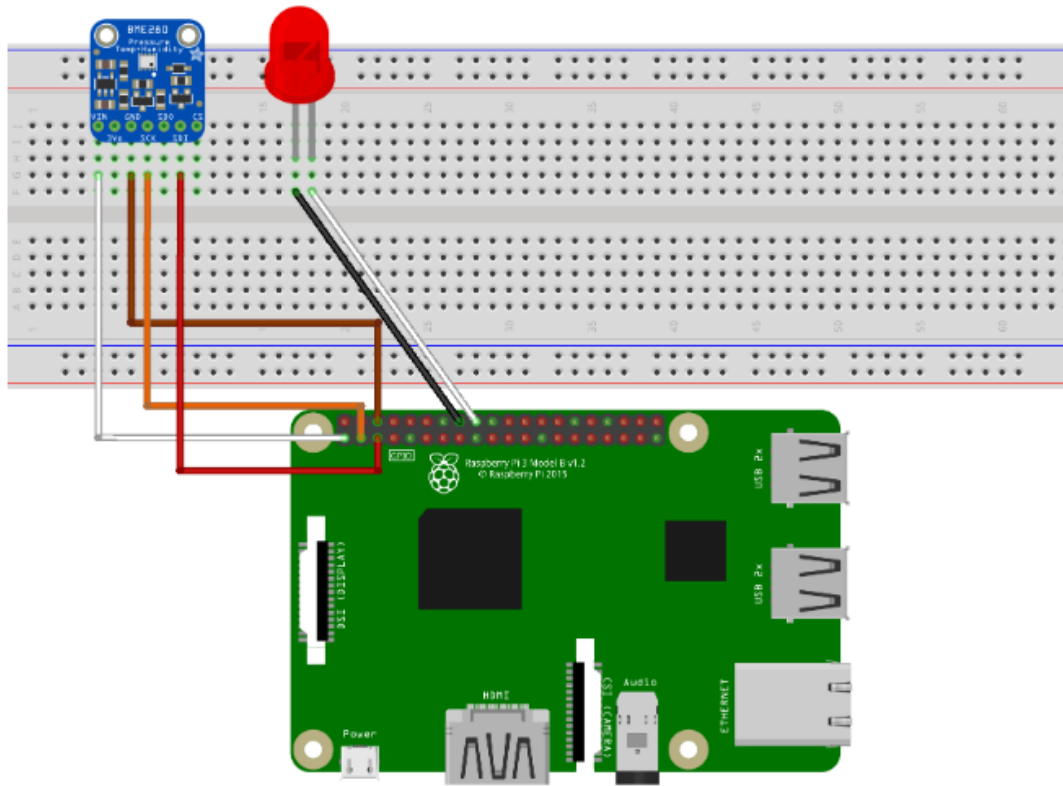
Run Reset

Haga clic en el botón 'Run' para ejecutar el código de ejemplo (cuando el ejemplo está en ejecución, Haga clic en el botón 'Stop' para detener el código de ejemplo en ejecución. Haga clic en 'Reset' para restablecer el código. Mantenemos sus cambios en el editor aunque actualic
> []

13. En este espacio vamos a introducir el valor de “Primary Connection String” obtenido en el paso 10.

Raspberry Pi Azure IoT Online Simulator

Ayuda Español ▼



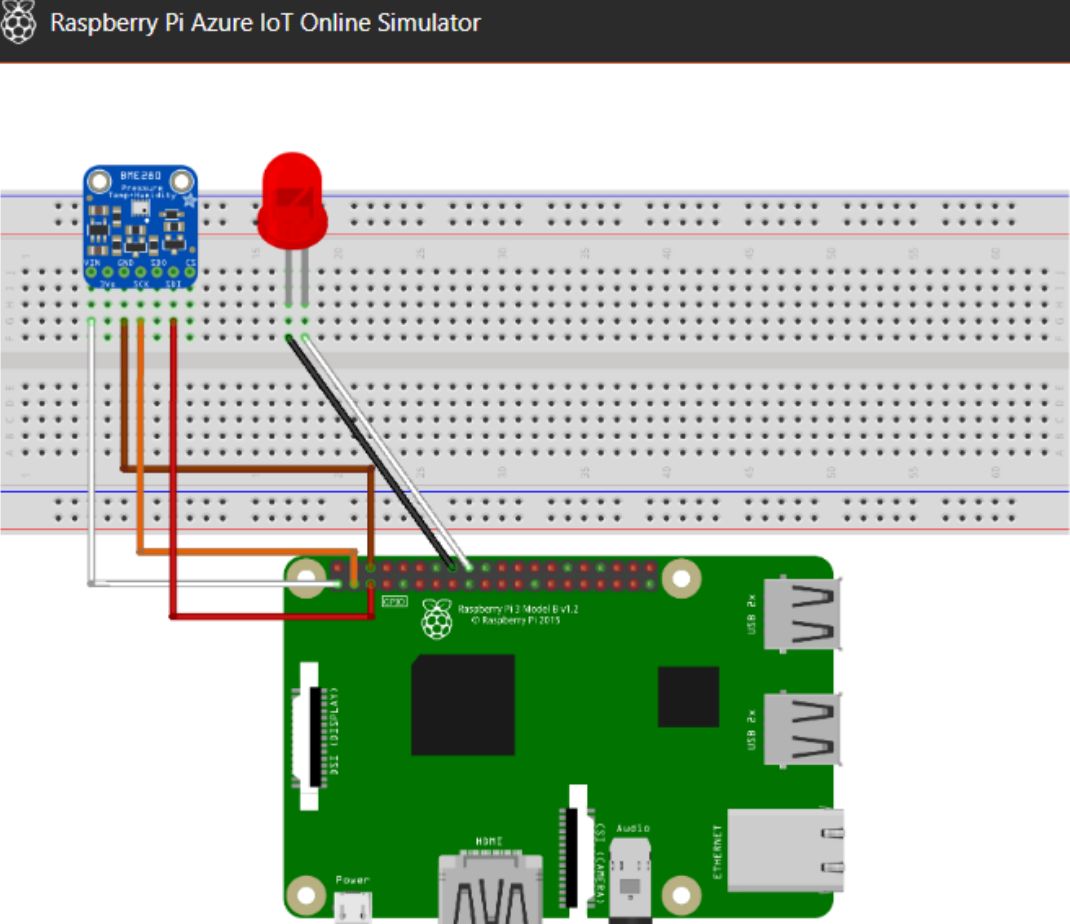
```
1  /*
2  * IoT Hub Raspberry Pi NodeJS - Microsoft Sample Code - Copyright (c) 2017 - Licensed MIT
3  */
4  const wpi = require('wiring-pi');
5  const Client = require('azure-iot-device').Client;
6  const Message = require('azure-iot-device').Message;
7  const Protocol = require('azure-iot-device-mqtt').Mqtt;
8  const BME280 = require('bme280-sensor');
9
10 const BME280_OPTION = {
11   i2cBusNo: 1, // defaults to 1
12   i2cAddress: BME280.BME280_DEFAULT_I2C_ADDRESS() // defaults to 0x77
13 };
14
15 const connectionString = 'HostName=my-streamming.azure-devices.net;DeviceId=my-raspberry-pi';
16 const LEDPin = 4;
17
18 var sendingMessage = false;
19 var messageId = 0;
20 var client, sensor;
21 var blinkLEDTIMEOUT = null;
22
23 function getMessage(cb) {
24
```

Run Reset

Haga clic en el botón 'Run' para ejecutar el código de ejemplo (cuando el ejemplo está en ejecución, Haga clic en el botón 'Stop' para detener el código de ejemplo en ejecución. Haga clic en 'Reset' para restablecer el código. Mantenemos sus cambios en el editor aunque actualic

> □

14. Seleccionamos "Run" y observamos unos minutos los mensajes que comienza a generar.



Raspberry Pi Azure IoT Online Simulator

Ayuda Español ▼

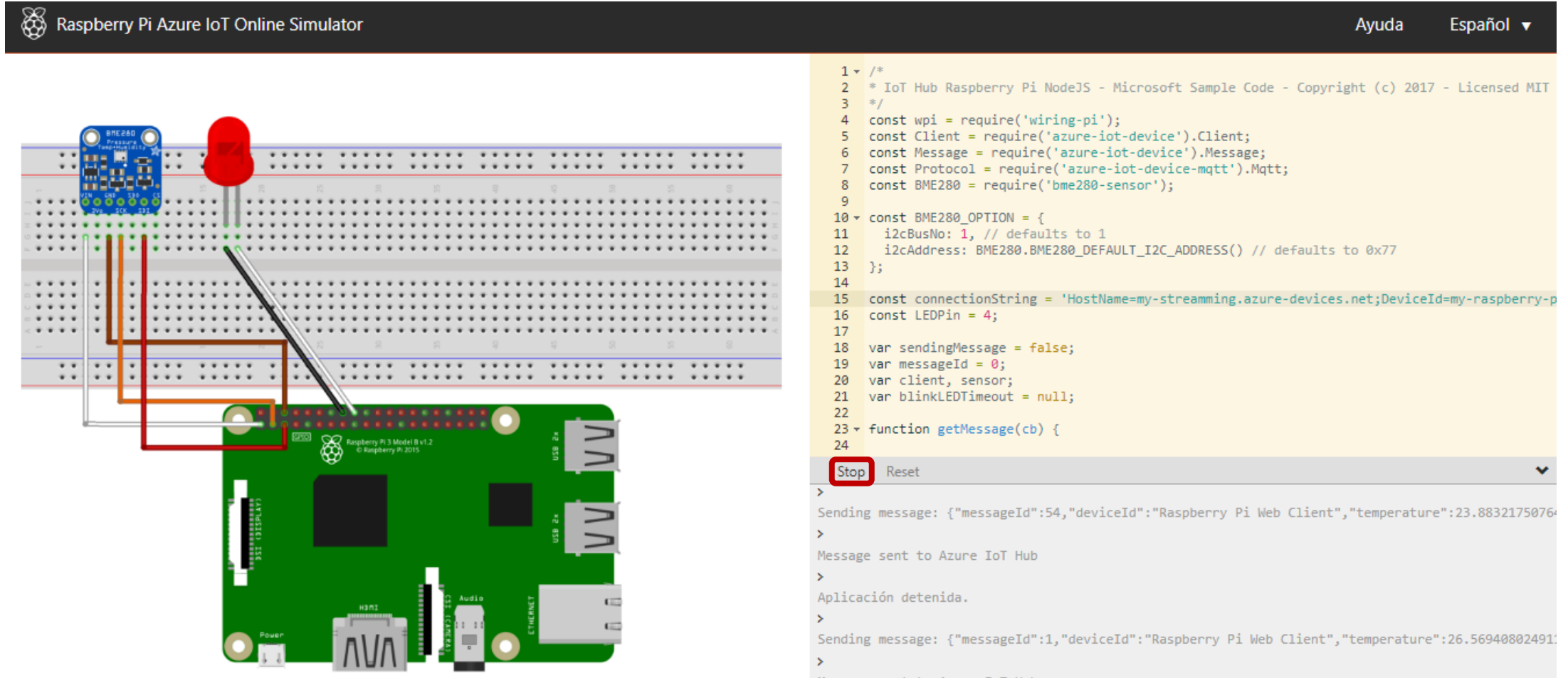
```
1 /*
2  * IoT Hub Raspberry Pi NodeJS - Microsoft Sample Code - Copyright (c) 2017 - Licensed MIT
3  */
4  const wpi = require('wiring-pi');
5  const Client = require('azure-iot-device').Client;
6  const Message = require('azure-iot-device').Message;
7  const Protocol = require('azure-iot-device-mqtt').Mqtt;
8  const BME280 = require('bme280-sensor');
9
10  const BME280_OPTION = {
11    i2cBusNo: 1, // defaults to 1
12    i2cAddress: BME280.BME280_DEFAULT_I2C_ADDRESS() // defaults to 0x77
13  };
14
15  const connectionString = 'HostName=my-streamming.azure-devices.net;DeviceId=my-raspberry-pi';
16  const LEDPin = 4;
17
18  var sendingMessage = false;
19  var messageId = 0;
20  var client, sensor;
21  var blinkLEDTIMEOUT = null;
22
23  function getMessage(cb) {
24
```

Run Reset

Haga clic en el botón 'Run' para ejecutar el código de ejemplo (cuando el ejemplo está en ejecución, Haga clic en el botón 'Stop' para detener el código de ejemplo en ejecución. Haga clic en 'Reset' para restablecer el código. Mantenemos sus cambios en el editor aunque actualic

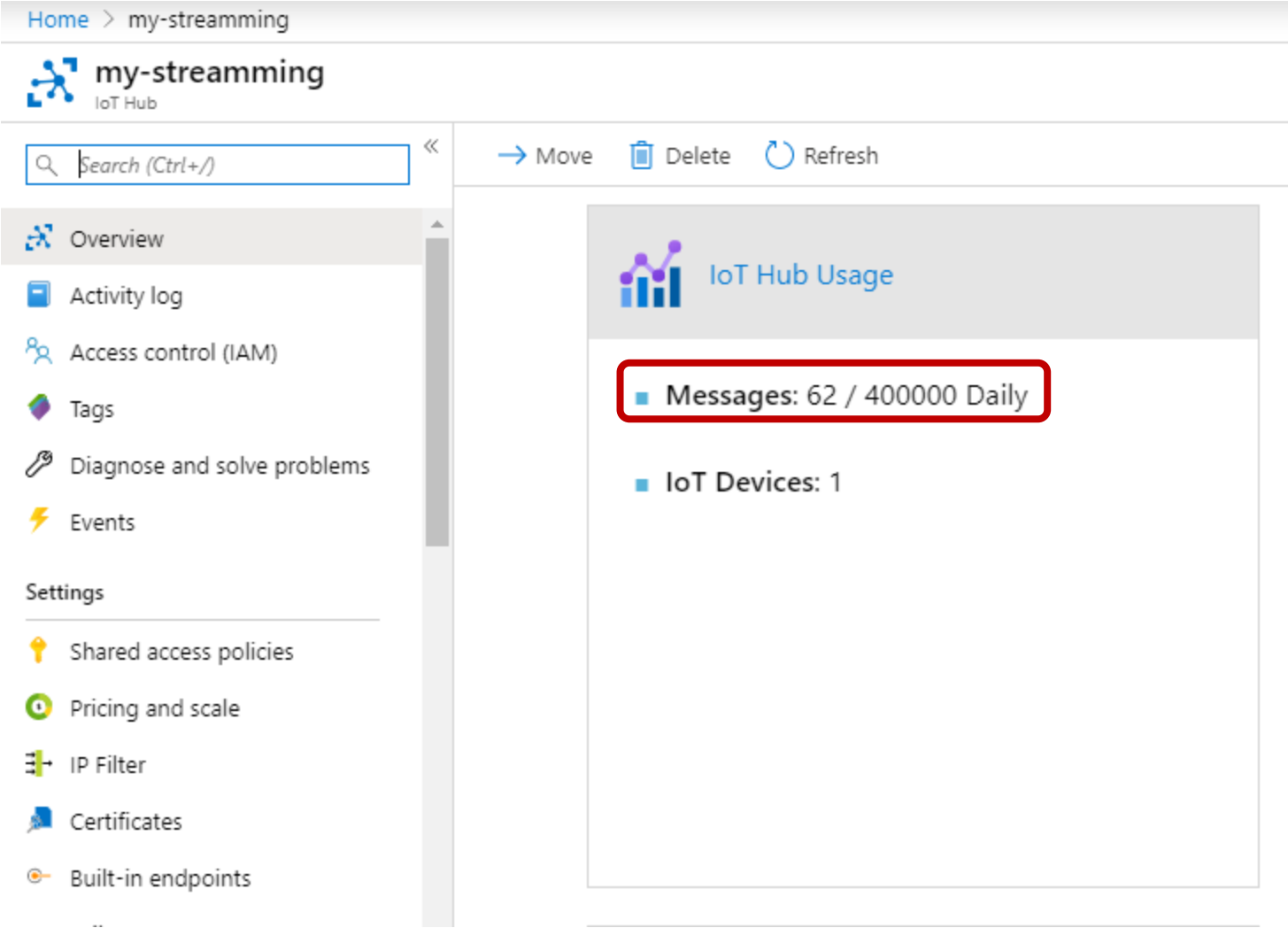
> □

15. Seleccionamos "Stop" y nos dirigimos a la página de azure portal.



```
1  /*
2  * IoT Hub Raspberry Pi NodeJS - Microsoft Sample Code - Copyright (c) 2017 - Licensed MIT
3  */
4  const wpi = require('wiring-pi');
5  const Client = require('azure-iot-device').Client;
6  const Message = require('azure-iot-device').Message;
7  const Protocol = require('azure-iot-device-mqtt').Mqtt;
8  const BME280 = require('bme280-sensor');
9
10 const BME280_OPTION = {
11   i2cBusNo: 1, // defaults to 1
12   i2cAddress: BME280.BME280_DEFAULT_I2C_ADDRESS() // defaults to 0x77
13 };
14
15 const connectionString = 'HostName=my-streamming.azure-devices.net;DeviceId=my-raspberry-pi';
16 const LEDPin = 4;
17
18 var sendingMessage = false;
19 var messageId = 0;
20 var client, sensor;
21 var blinkLETimeout = null;
22
23 function getMessage(cb) {
24
25   // Stop button
26
27   // Sending message: {"messageId":54,"deviceId":"Raspberry Pi Web Client","temperature":23.8832175076...
28   // Message sent to Azure IoT Hub
29   // Aplicación detenida.
30   // Sending message: {"messageId":1,"deviceId":"Raspberry Pi Web Client","temperature":26.56940802491...
31   // Message sent to Azure IoT Hub
```

16. En la sección de “Overview”, de nuestro servicio IoT Hub que creamos, buscamos la parte de “IoT Hub Usage”. Debemos observar que tenemos varios mensajes recibidos por el servicio (dependerá del tiempo que tuvimos funcionando el simulador).





Microsoft

©2014 Microsoft Corporation. All rights reserved. Microsoft, Windows, Office, Azure, System Center, Dynamics and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries. The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation. MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.