



UNIVERSIDADE
Estácio de Sá

| | |
|-----------------------------|---|
| Universidade | Estácio de Sá |
| Campus | Polo de Cobilândia / Vila – Velha/ES |
| Nome do Curso | Desenvolvimento Full Stack |
| Nome da Disciplina | RPG0027- Vamos interligar as coisas com a nuvem |
| Turma | 9001 |
| Semestre | Segundo Semestre de 2024 |
| Integrantes do Grupo | Tiago de Jesus Pereira Furtado |
| Matrícula | 202306189045 |

**VILA VELHA
2025**

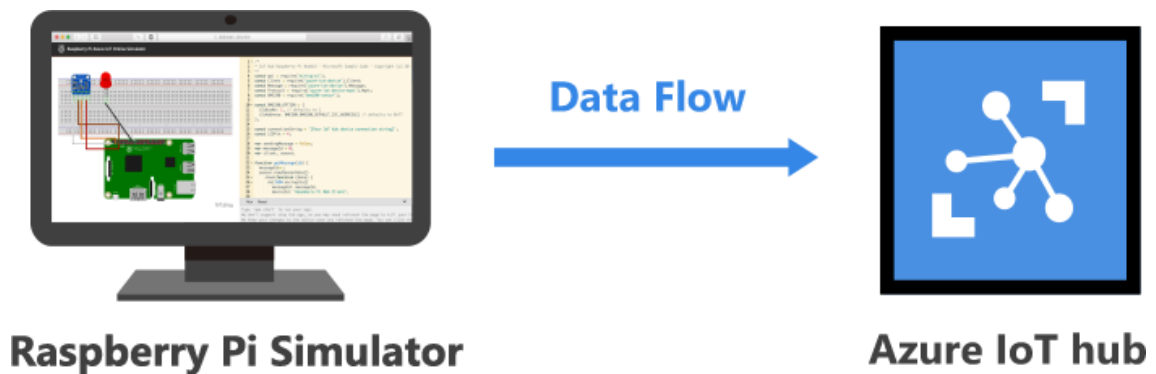
Microatividade 3: Conectar o simulador online Raspberry Pi ao Hub IoT do Azure

- Material necessário para a prática

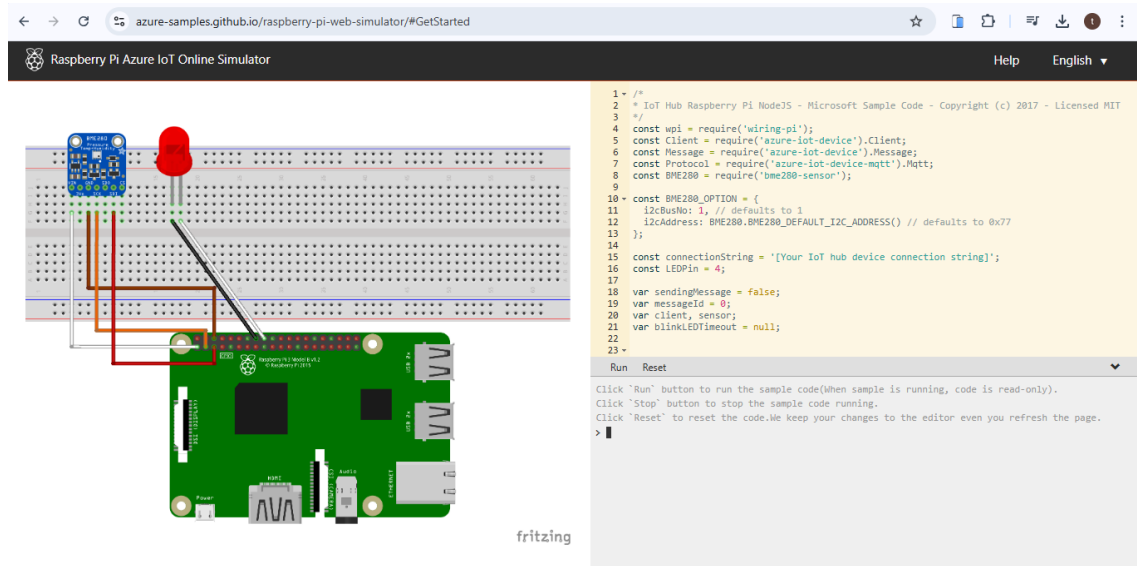
- Conta no Microsoft Azure.
- Navegador Web (Google Chrome, Firefox, MS Edge, Safari ou Opera).
- Raspberry Pi Azure IoT Online Simulator

- Procedimentos

Esta atividade proporcionará uma introdução às práticas fundamentais de interação com o simulador online do Raspberry Pi. Você iniciará aprendendo os conceitos básicos necessários para operar este simulador. Em seguida, o tutorial guiará você na conexão do simulador Pi diretamente à nuvem utilizando o Hub IoT do Azure.



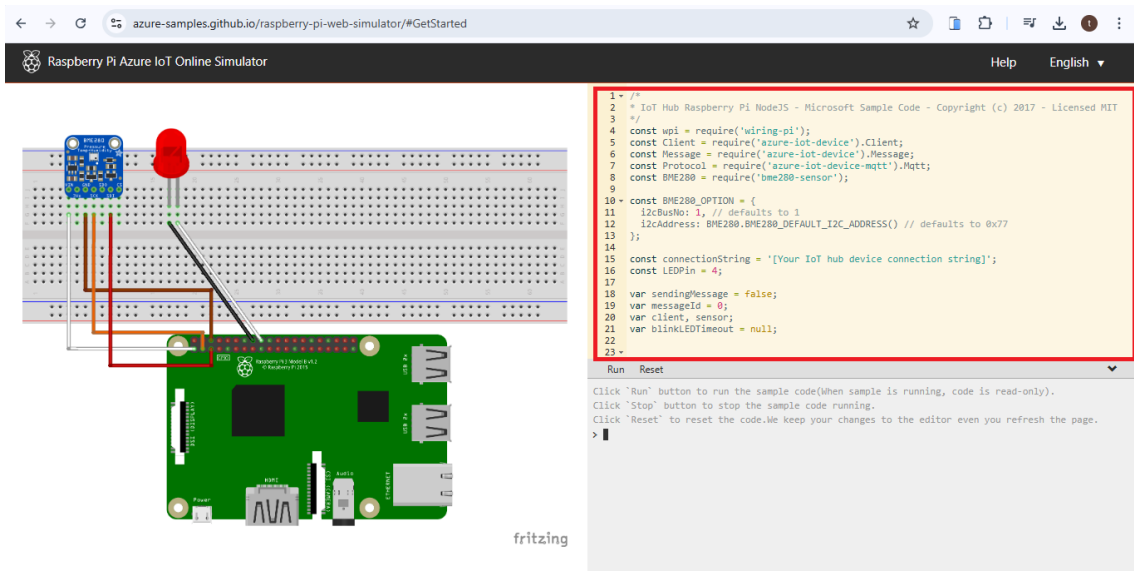
1. Acesse o Raspberry Pi Azure IoT Online Simulator visitando o endereço <https://azure-samples.github.io/raspberry-pi-web-simulator/#GetStarted>



Ao entrar no simulador, você se deparará com um ambiente dividido em três seções distintas. À esquerda, encontra-se a “Área de Montagem”. A “Área de Montagem” representa graficamente o simulador do Raspberry Pi, incluindo dispositivos e conexões simuladas. Nesta versão, a personalização da montagem não está disponível, mas você pode visualizar a configuração padrão:

- Sensor de Umidade BME280 (I2C.1): Simula a conexão de um sensor de umidade ao barramento I2C.1.
- LED (GPIO 4): Simula a conexão de um LED ao pino GPIO 4.

À direita, na parte superior, está reservada a área de codificação, proporcionando um espaço para desenvolver e testar seus códigos. Utilize o editor de código online para interagir com o Raspberry Pi. O aplicativo padrão exemplifica a coleta de dados do sensor simulado BME280, enviando-os para o Hub IoT do Azure. Vale destacar que o código desenvolvido é plenamente compatível com dispositivos Raspberry Pi físicos.



The screenshot displays the 'Raspberry Pi Azure IoT Online Simulator' web application. The interface is split into two main sections. On the left, there is a 3D visualization of a Raspberry Pi board with a breadboard and a BME280 sensor module connected to it. On the right, there is a code editor with a red border containing the following JavaScript code:

```
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-mqtt').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 const connectionString = '[Your IoT hub device connection string]';  
15 const LEDPin = 4;  
16  
17 var sendMessage = false;  
18 var messageId = 0;  
19 var client, sensor;  
20 var blinkLEDTIMEOUT = null;  
21  
22  
23
```

Below the code editor, there are 'Run' and 'Reset' buttons. Instructions below the buttons state: 'Click "Run" button to run the sample code (when sample is running, code is read-only). Click "Stop" button to stop the sample code running. Click "Reset" to reset the code. We keep your changes to the editor even you refresh the page.'

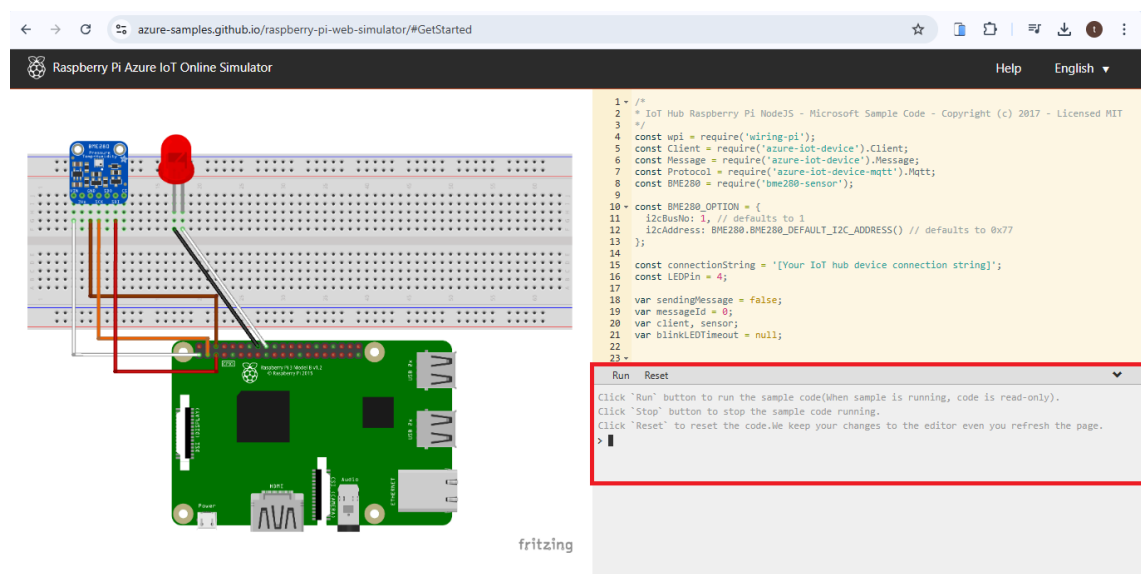
Logo abaixo, na parte inferior, você terá acesso ao console do simulador, onde informações e mensagens relevantes serão exibidas durante a execução. Na parte superior da janela, há três botões.

- Executar - Executar o aplicativo na área de codificação.
- Redefinir - Redefinir a área de codificação para o aplicativo de exemplo padrão.
- Recolher/Expandir: no lado direito, há um botão para recolher ou expandir a janela do console.

Esta organização facilita a interação e a compreensão do ambiente do simulador,

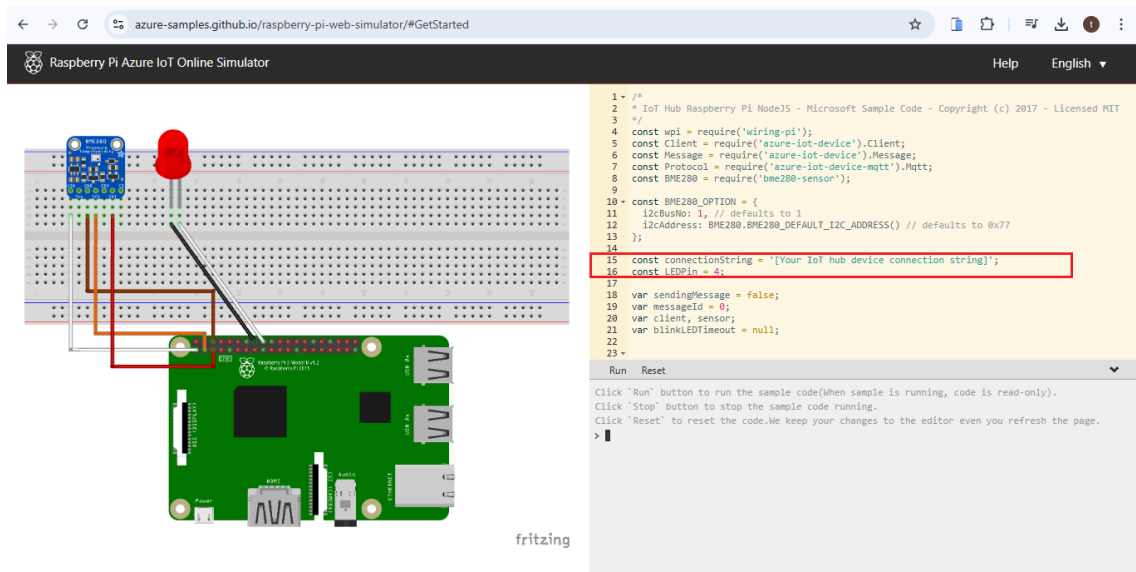
permitindo uma experiência mais intuitiva e eficiente no desenvolvimento para o

Raspberry Pi no contexto do Azure IoT.



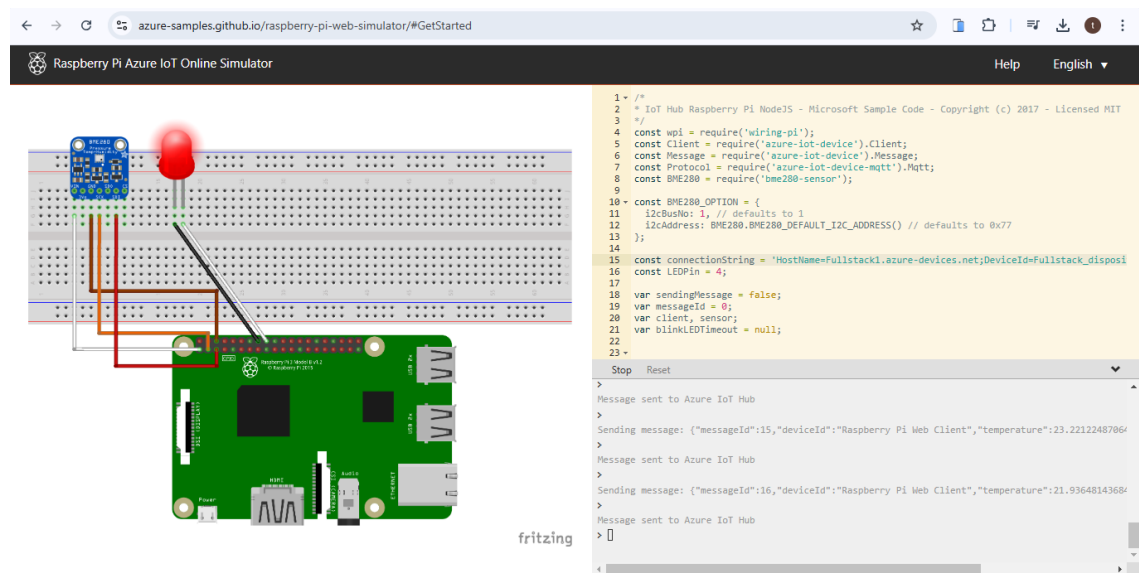
2. Na área de codificação, comece trabalhando com o aplicativo de exemplo padrão fornecido. Para estabelecer a conexão com o Hub IoT do Azure, siga os passos abaixo:

- a) Localize a linha 15 no código do aplicativo.
- b) Substitua o espaço reservado na linha 15 pela cadeia de conexão do dispositivo associado ao Hub IoT do Azure.



Essa ação garantirá que o aplicativo utilize a cadeia de conexão correta, permitindo uma comunicação segura e eficiente com o Hub IoT no ambiente Azure. Certifique-se de inserir a cadeia de conexão correta para garantir a integridade e a segurança da conexão. Este passo é crucial para que o seu dispositivo Raspberry Pi possa interagir de forma adequada com o Hub IoT no ambiente Azure.

3. Após realizar as configurações necessárias, selecione "Executar" ou digite npm start na janela de console integrada para iniciar a execução do aplicativo. Ao realizar essa ação, você será apresentado com a seguinte saída, exibindo os dados do sensor e as mensagens enviadas com sucesso para o seu Hub IoT. O led na área de montagem irá piscar.



Ao terminar a execução, clique em "Stop". 🖱️

- Resultados esperados 🌟

Ao concluir esta microatividade, espera-se que você tenha executado com sucesso o Simulador Online Raspberry Pi, realizando o envio efetivo de dados do sensor para o Hub IoT previamente configurado. Certifique-se de verificar a correta operação do aplicativo e a recepção adequada dos dados no Hub IoT do Azure. Essa etapa é crucial para validar a integração bem-sucedida entre o simulador do Raspberry Pi e o Hub IoT, garantindo que o fluxo de dados esteja funcionando conforme o esperado.

Segue Abaixo um o link do vídeo onde está sendo feito essa atividade: <https://youtu.be/Uuw5atkteLk>