

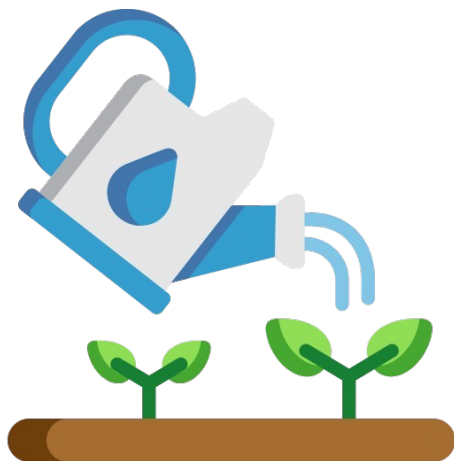
# Irrigador de Plantas Automático

Gustavo Giglio



# Introdução

Este sistema tem como objetivo auxiliar no monitoramento e facilitar o cuidado das plantas de forma automatizada. Somente irrigando o solo quando necessário. Desta forma ajudando a tirar a preocupação de quem não tem conhecimento ou tempo adequado para dar os devidos cuidados a suas plantas.



# Componentes

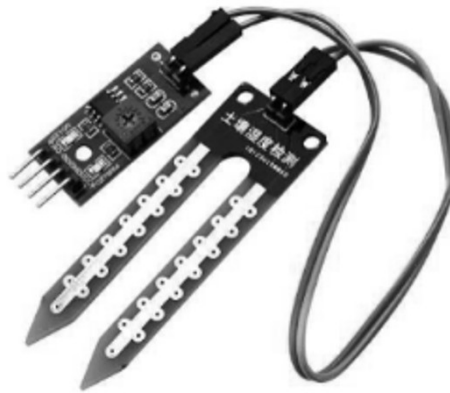


**INSTITUTO FEDERAL**  
Rio Grande do Sul



# Sensor de umidade do solo higrômetro

Este Sensor de Umidade do Solo Higrômetro foi feito para detectar as variações de umidade no solo, sendo que quando o solo está seco a saída do sensor fica em estado alto (HIGH), e quando úmido em estado baixo (LOW). Sua tensão de operação é de 3,3 a 5V.



O limite entre seco e úmido pode ser ajustado através do potenciômetro presente no sensor

# Sensor de umidade do solo higrômetro

Modulo Relé 5V de 1 Canal, que funciona como um interruptor, essencial para controlar diretamente o acionamento do irrigador da planta quando o sensor de umidade detecta a má condição do solo, atuando como um pino de controle do atuador, que é a bomba de água.



# Mini bomba da água submersa

Esta mini bomba é perfeita para quem precisa mantê-la debaixo da água. Ela possui uma abertura onde a água entra e então é bombeada para a saída.



Vazão máxima: 120L/hora

Sua tensão de operação é de 3v a 6V.

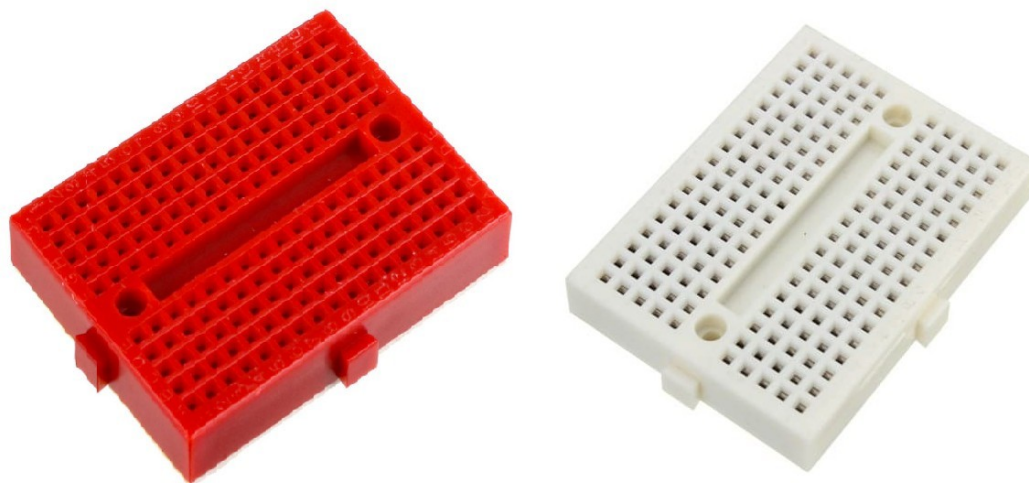


**INSTITUTO FEDERAL**  
Rio Grande do Sul



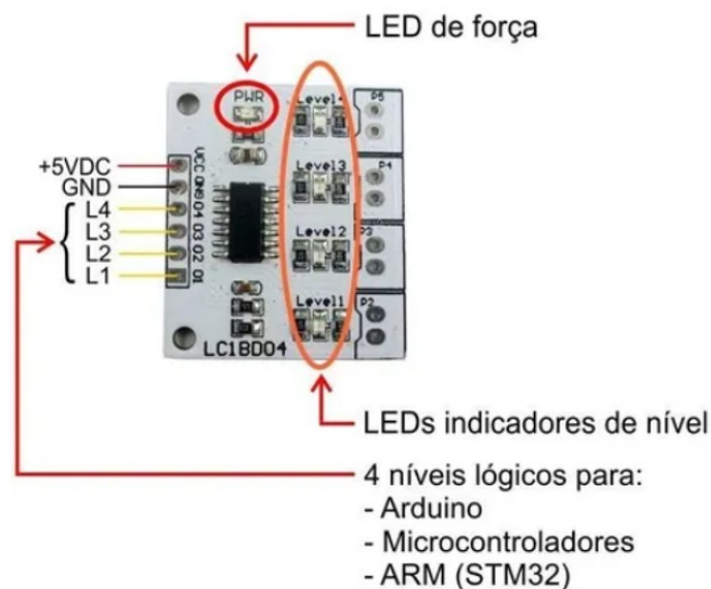
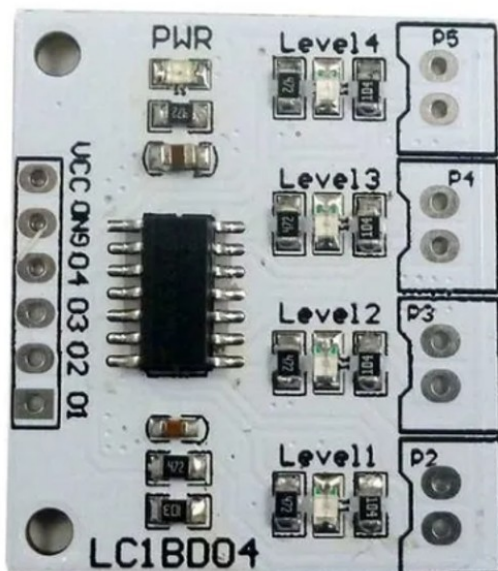
# Mini protoboard

2x mini protoboard 170 pontos, para fazer a ligação dos componentes dos circuitos eletrônicos com a definição de entradas de cada ferramenta para o seu devido e adequado funcionamento.



# Sensor de nível de água

Este sensor é bom para ocasiões que exigem abastecimento automático de água e troca de água. Possui 4 leds indicadores de nível de água.



Sua tensão de operação é de 5v.



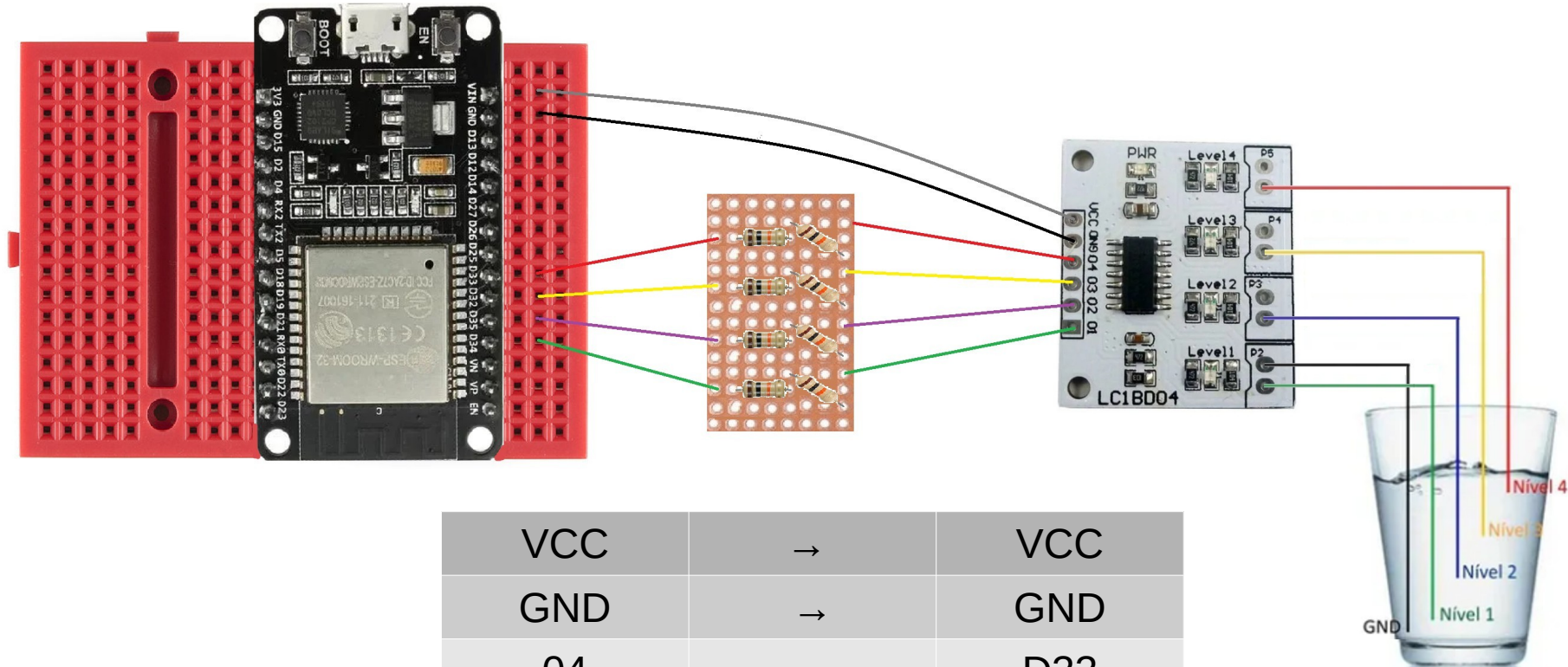
# Modelagem



**INSTITUTO FEDERAL**  
Rio Grande do Sul

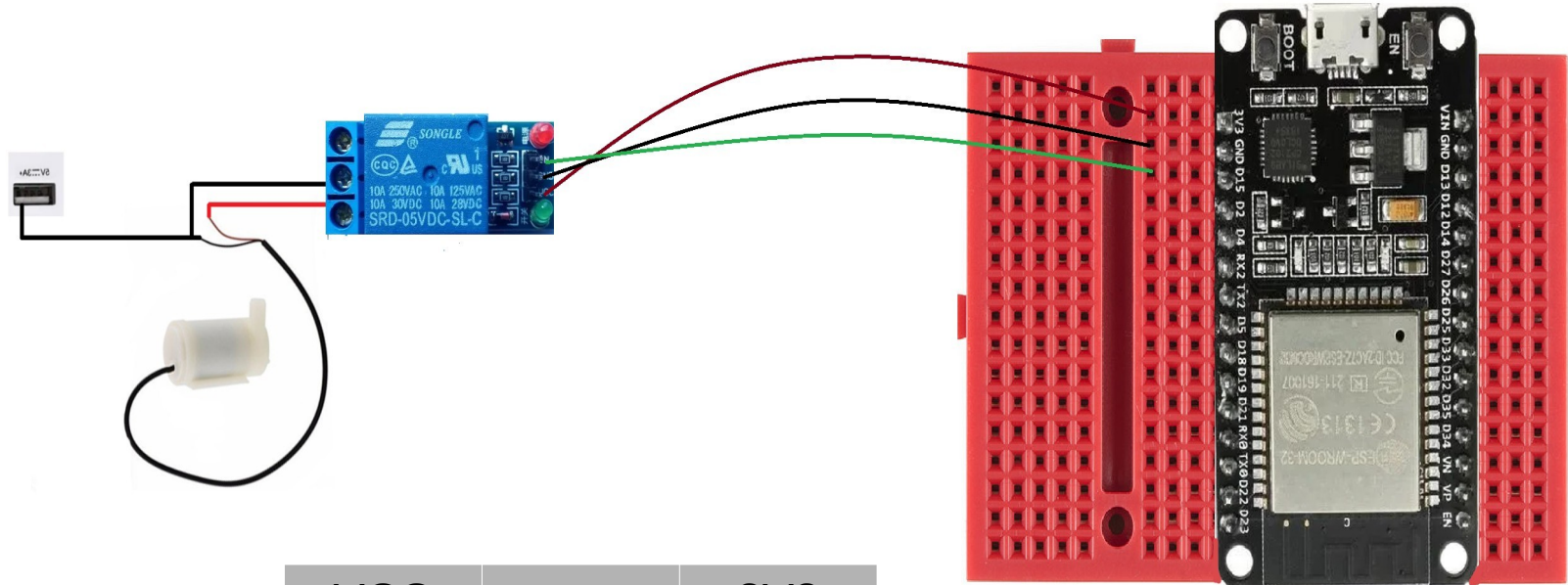


# Sensor de nível de água



VCC	→	VCC
GND	→	GND
04	→	D33
03	→	D32
02	→	D35
01	→	D34

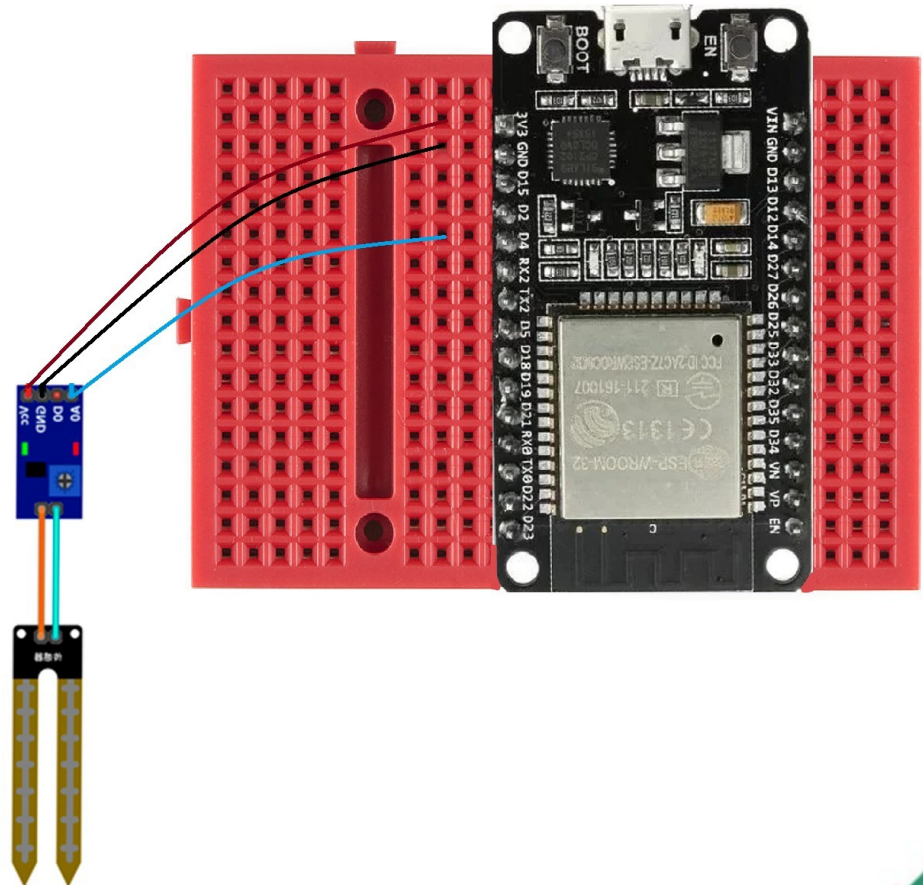
# Bomba da água



VCC	→	3V3
GND	→	GND
IN	→	D15

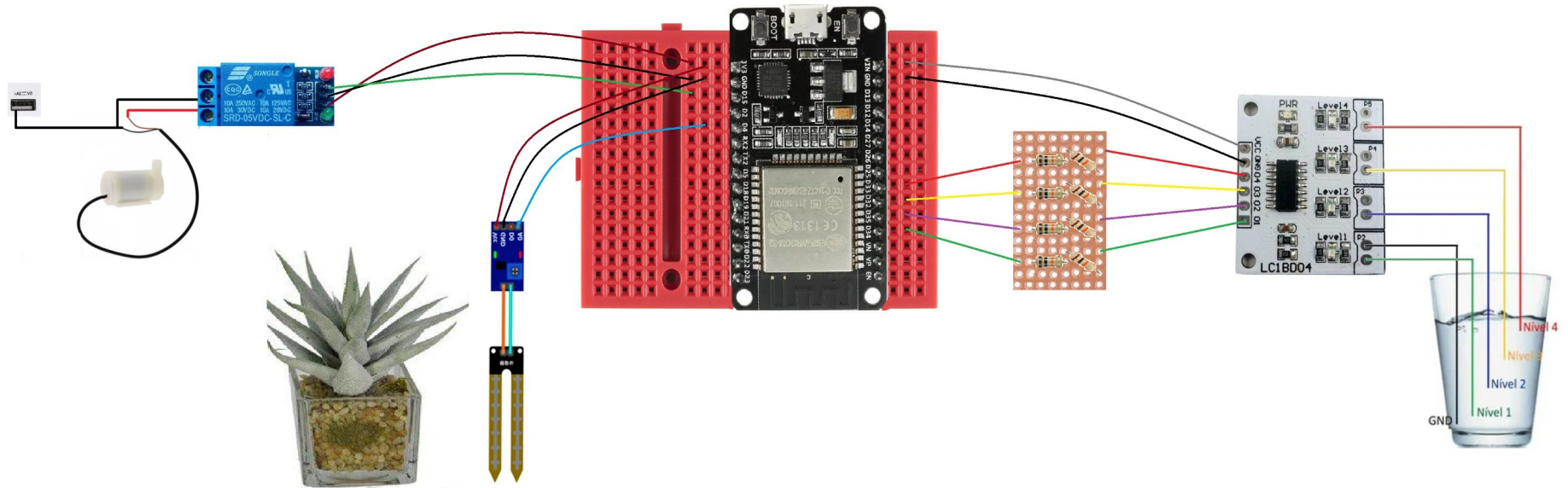
# Bomba da água

VCC	→	3V3
GND	→	GND
A0	→	D4





# Projeto Completo



# Código



**INSTITUTO FEDERAL**  
Rio Grande do Sul



# Declaração dos pinos

```
1
2  const int pino_sensor = 2; //pino conencatdo ao sensor de umidade
3  int leitura_sensor = 0;
4  const int pino_bomba = 15; //pino conectado ao rele da bomba
5  int n1 = 33; //pino conencatdo ao sensor de nivel agua mais baixo
6  int n2 = 32; //pino conencatdo ao sensor de nivel agua baixo
7  int n3 = 35; //pino conencatdo ao sensor de nivel agua meio
8  int n4 = 34; //pino conencatdo ao sensor de nivel agua alto
9
```



# Setup

```
10 void setup(){
11     Serial.begin(9600); //Inicializa a serial com taxa de 9600 bauds
12     pinMode(pino_sensor, INPUT); //Define o sensor de umidade como entrada do sistema
13     pinMode(pino_bomba, OUTPUT); //Define a bomba como saída do sistema
14     pinMode(n1, INPUT); //Define o sensor de nível água como entrada do sistema
15     pinMode(n2, INPUT); //Define o sensor de nível água como entrada do sistema
16     pinMode(n3, INPUT); //Define o sensor de nível água como entrada do sistema
17     pinMode(n4, INPUT); //Define o sensor de nível água como entrada do sistema
18 }
```





# Loop – parte(1/2)

```
20 void loop(){
21     leitura_sensor = analogRead(pino_sensor); //Sensor de umidade faz leitura do solo
22
23     Serial.print("Status: ");
24     if(leitura_sensor >= 1500){ //Verifica se o solo está seco ou umido
25         Serial.print("Solo Seco ");
26         digitalWrite (pino_bomba, LOW); //Se estiver seco, desliga o relé e aciona a bomba
27     } else {
28         Serial.print("Solo umido");
29         digitalWrite (pino_bomba, HIGH); //Se estiver umido, mantém o relé da bomba desligada
30     }
31     Serial.print(" | Leitura: ");
32     Serial.print(leitura_sensor);
33
34     //Realiza as leituras dos sensores e as armazena em duas variaveis
35     int leitura_sensor_1 = digitalRead(n1);
36     int leitura_sensor_2 = digitalRead(n2);
37     int leitura_sensor_3 = digitalRead(n3);
38     int leitura_sensor_4 = digitalRead(n4);
39 }
```



# Loop – parte(2/2)

```
40 Serial.print(" | Quantidade Agua: ");
41
42 //Verifica se o primeiro sensor esta na ausencia de liquido
43 if(leitura_sensor_4 == HIGH) {
44     Serial.print("100% Cheio"); //Printa a mensagem mostrando que o balde está 100% Cheio
45     Serial.println("");
46 }
47
48 //Verifica se o segundo sensor esta na ausencia de liquido
49 else if(leitura_sensor_3 == HIGH) {
50     Serial.print("75% Cheio"); //Printa a mensagem mostrando que o balde está 75% Cheio
51     Serial.println("");
52 }
53
54 //Verifica se o terceiro sensor esta na ausencia de liquido
55 else if(leitura_sensor_2 == HIGH) {
56     Serial.print("50% Cheio"); //Printa a mensagem mostrando que o balde está 50% Cheio
57     Serial.println("");
58 }
59
60 //Verifica se o quarto sensor esta na ausencia de liquido
61 else if(leitura_sensor_1 == HIGH) {
62     Serial.print("25% Cheio"); //Printa a mensagem mostrando que o balde está 25% Cheio
63     Serial.println("");
64 }
65
66 //Avisa que acabou o liquido e precisa encher
67 else if(leitura_sensor_1 == LOW) {
68     Serial.print("Encher balde"); //Printa a mensagem mostrando que o balde está com a água quase acabando
69     Serial.println("");
70 }
71
72 delay(1000); //Aguarda 10 segundos para outra leitura
73 }
```





Desenvolvido para ser utilizado em projetos IoT, com ele conseguimos comunicar através do celular com nossa placa ESP-32 e controlá-la via Wi-fi ou Bluetooth.

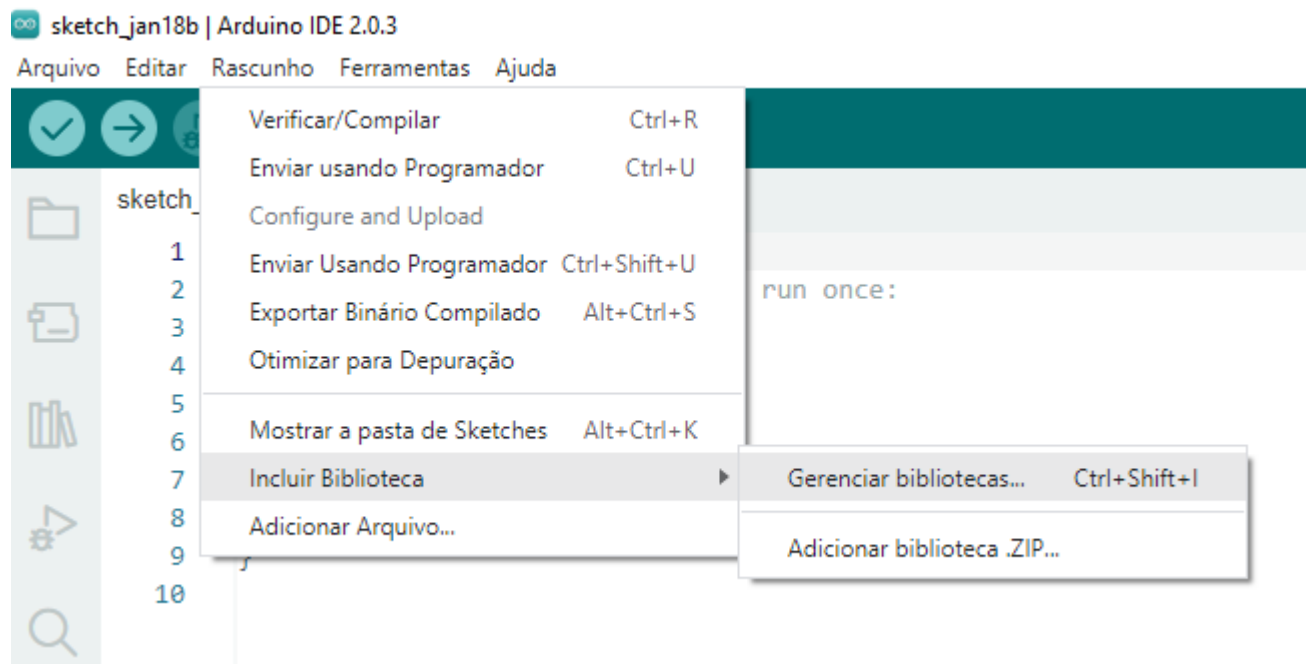


**INSTITUTO FEDERAL**  
Rio Grande do Sul



# Configurando o blynk

Para as programações funcionarem corretamente é necessário instalar a biblioteca do Blynk na IDE do Arduino



# Configurando o blynk

Digite blynk e selecione a biblioteca criado por Volodymyr Shymanskyy.

GERENCIADOR DE BIBLIOTECAS

blynk

Tipo: 

All

Topic: 

All

Blynk por Volodymyr Shymanskyy

Versão 1.1.0

INSTALLED

It supports WiFi, BLE, Bluetooth, Ethernet, GSM, USB, Serial. Works with many boards like ESP8266, ESP32, Arduino UNO, Nano, Due, Mega, Zero, MKR100, Yun, Raspberry Pi, Particle, Energia, ARM mbed, Intel Edison/Galileo/Joule, BBC micro:bit, DFRobot, RedBearLab, Microduino, LinkIt ONE ...  
Build a smartphone app for your project in minutes!

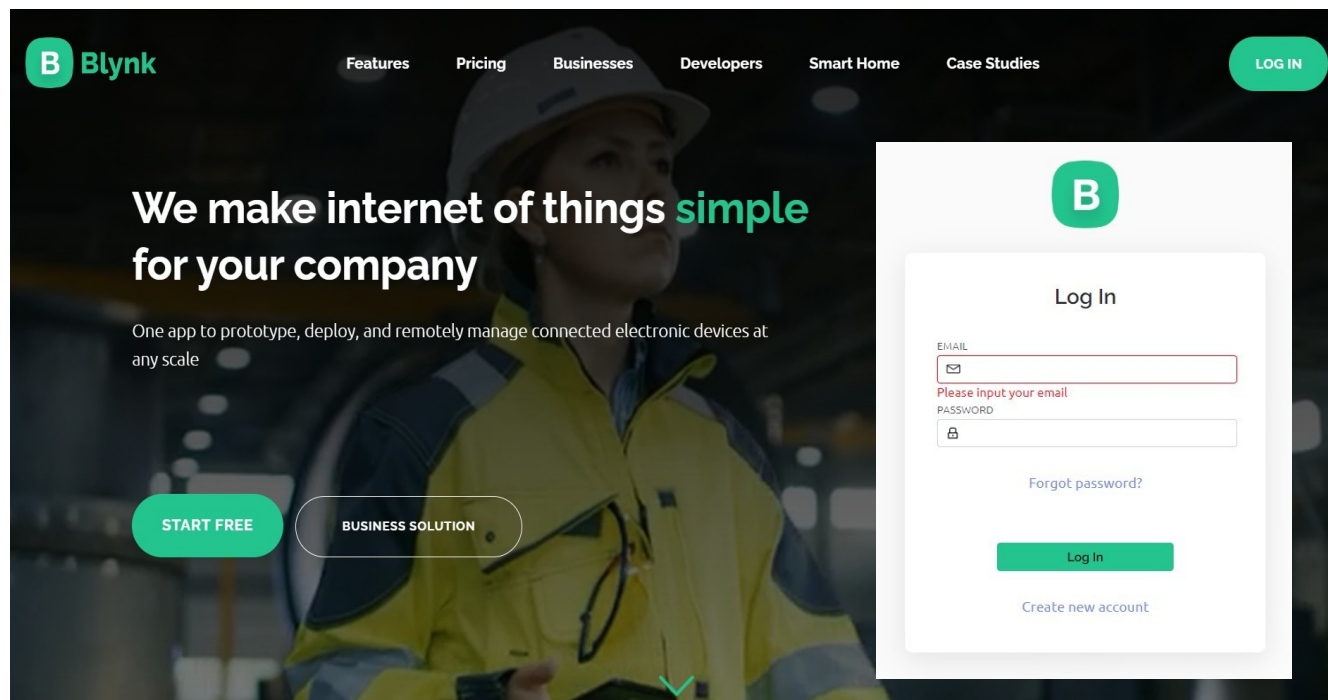
[Mais informações](#)

1.0.1

INSTALAR

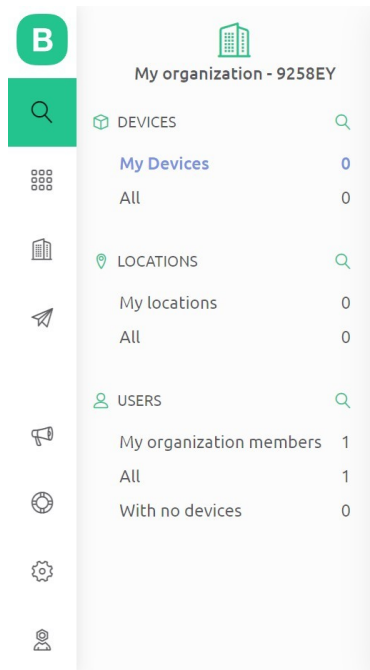
# Configurando o blynk

- Acesse o site da blynk: <https://blynk.io/>
- Faça o seu login.



# Configurando o blynk

- Crie um novo device.



Region: ny3 [Privacy Policy](#)

# Configurando o blynk

- Adicione um nome para sua template.

## Create New Template

NAME

Sistema de irrigador

HARDWARE

ESP32

CONNECTION TYPE

WiFi

DESCRIPTION

Sistema que notifica o usuário que ele está sem água e precisa reabastecer o tanque com água para continuar o processo d irrigar|

128 / 128

Cancel

Done



**INSTITUTO FEDERAL**  
Rio Grande do Sul



# Configurando o blynk

## Sistema de irrigador

Info

Metadata

Datastreams

Events

Automations

Web Dashboard

Mobile Dashboard

TEMPLATE NAME

Sistema de irrigador

HARDWARE

ESP32

CONNECTION TYPE

WiFi

DESCRIPTION

Sistema que notifica o usuário que ele está sem água e precisa reabastecer o tanque com água para continuar o processo d irrigar

TEMPLATE ID

TMPL6kTaSqLV

MANUFACTURER

My organization 9258EY

128 / 128

OFFLINE IGNORE PERIOD

00 hrs 00 mins 00 secs

HOTSPOT PREFIX

Hotspot Prefix

TEMPLATE IMAGE (OPTIONAL)



Add image

Upload from computer or drag-n-drop  
.png or .jpg, minimum width 500px

FIRMWARE CONFIGURATION

```
#define BLYNK_TEMPLATE_ID "TMPL6kTaSqLV"  
#define BLYNK_DEVICE_NAME "Sistema de irrigador"
```

Template ID and Device Name should be included at the top of your main firmware



**INSTITUTO FEDERAL**  
Rio Grande do Sul

# Configurando o blynk

- Vá para Datastreams e crie uma nova Datastreams.
- Selecione a opção virtual pin.

## Sistema de irrigador

[Info](#) [Metadata](#) [Datastreams](#) [Events](#) [Automations](#) [Web Dashboard](#) [Mobile Dashboard](#)

### Datastreams

Datastreams is a way to structure data that regularly flows in and out from device. Use it for sensor data, any telemetry, or actuators.

+ New Datastream

Digital

Analog

Virtual Pin



Enumerable

Location **UPGRADE**

# Configurando o blynk

- Configure pino virtual.


## Virtual Pin Datastream

PIN:  DATA TYPE:

UNITS:



MIN:  MAX:  DEFAULT VALUE:

 ADVANCED SETTINGS

Cancel

Create


## Virtual Pin Datastream

PIN:  DATA TYPE:

UNITS:

MIN:  MAX:  DEFAULT VALUE:

 ADVANCED SETTINGS

Cancel

Create

# Configurando o blynk

- Após criar e configurar os pinos, clique em Web Dashboard.

## Sistema de irrigador

... Cancel Save

Info Metadata Datastreams Events Automations Web Dashboard Mobile Dashboard



Search datastream

+ New Datastream

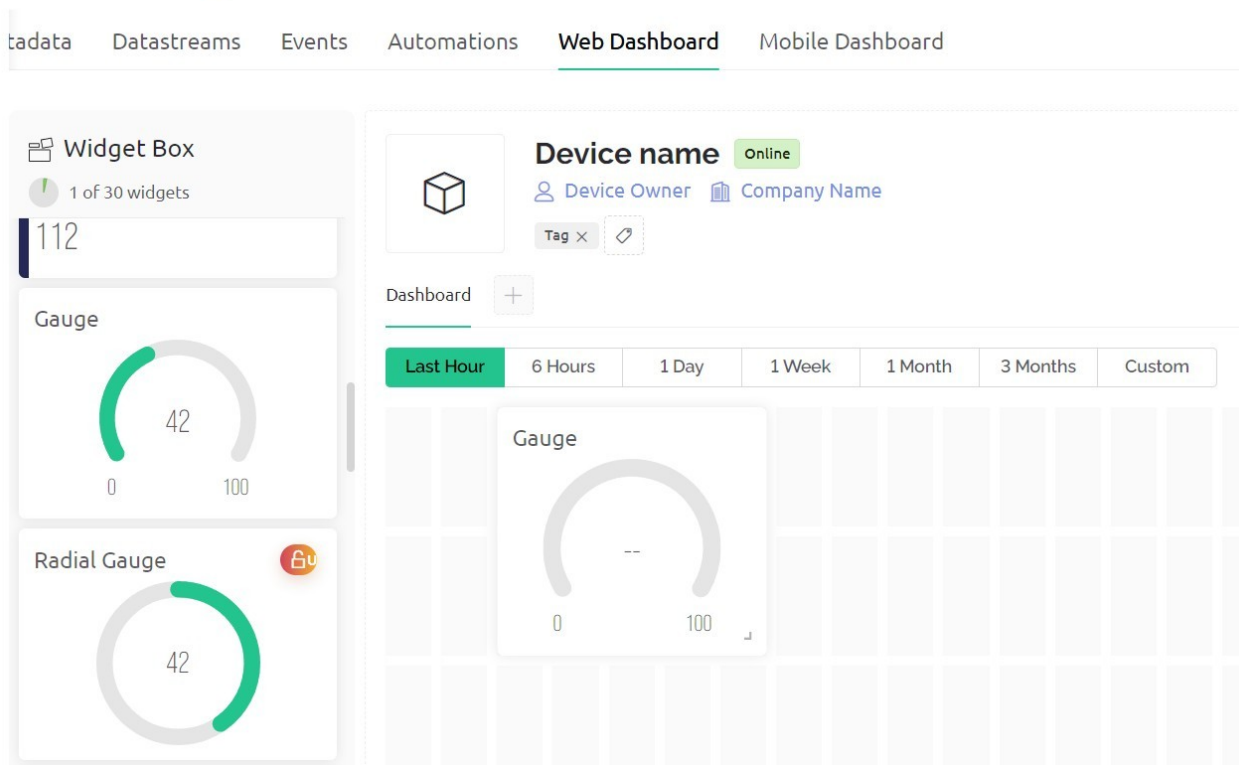
2 Datastreams

<input type="checkbox"/>	Id	Name	Alias	Color	Pin	Data Type	Units	Is Raw	Min	Max	Actions
	1	Umidade	Umidade		V0	Integer		false	0	100	
	2	bomba agua	bomba agua		V1	Integer		false	0	1	

# Configurando o blynk

- Precione e arraste o gauge para a dashboard.
- Clique em cima do gauge para configura-lo.

Sistema de irrigador



# Configurando o blynk

- Selecione o sensor de umidade(v1) e salve.

## Gauge Settings

TITLE (OPTIONAL)

Umidade

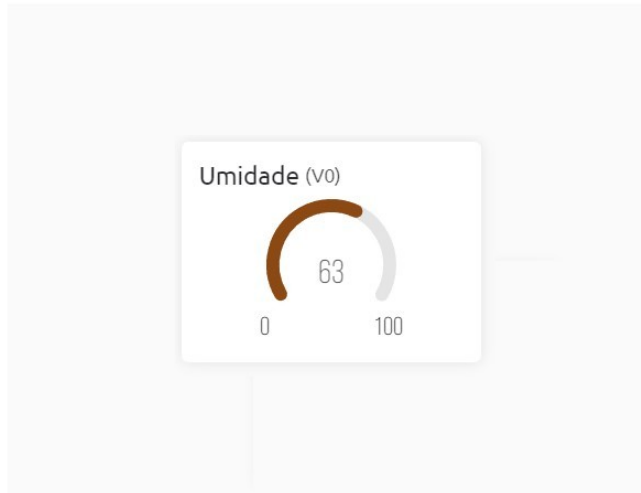
Datastream

Umidade (V0)

bomba agua (V1)

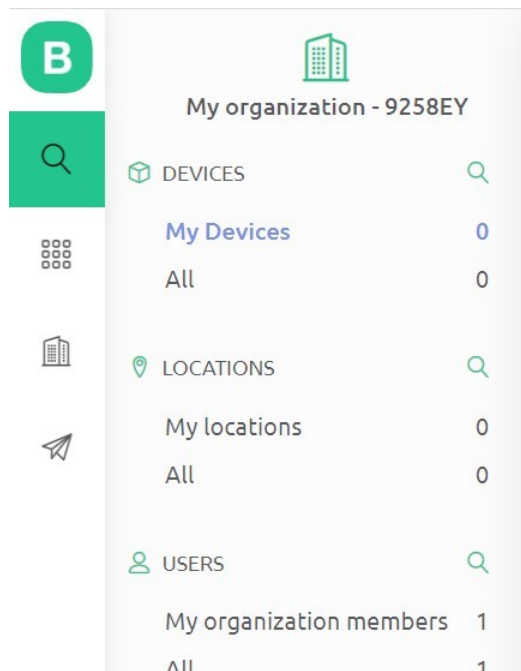
Umidade (V0)

☒ Change color based on value



# Configurando o blynk

- Clique na lupa e crie um novo device



All of your devices will be here.

You can activate new devices by using  
your app for IOS or Android



Download for IOS



Download for Android

+ New Device



**INSTITUTO FEDERAL**  
Rio Grande do Sul


# Configurando o blynk

- Selecione a opção from template


New Device

Choose a way to create new device


From template




Scan QR code



Manual entry



 Create a device by filling in a simple form

Cancel




# Configurando o blynk

- Selecione o seu sistema e clique em criar.

## New Device

Create new device by filling in the form below

TEMPLATE

Sistema de irrigador

Sistema de irrigador

New Device

Cancel

Create



**INSTITUTO FEDERAL**  
Rio Grande do Sul

# Configurando o blynk

- Vá para janela device info e copie o token gerado no FIRMWARE CONFIGURATION e insira no código.

The screenshot displays the Blynk mobile application interface. On the left is a sidebar with a green header containing a 'B' logo, a search icon, and a list of device icons. The main content area shows the 'Sistema de irrigador' device page. At the top, it says 'My organization - 9258EY' and 'Offline'. Below this is a 'Back' button and a search bar. A list of devices shows 'Sistema de irrigador' selected. The main panel has tabs for 'Dashboard', 'Timeline', 'Device Info' (selected), 'Metadata', and 'Actions Log'. Under 'Device Info', there are sections for 'STATUS' (Offline), 'LAST UPDATED' (7:56 PM Today), 'DEVICE ACTIVATED' (7:56 PM Today by 10070389@restinga.ifrs.edu.br), 'AUTHOKEN' (bdDN - .... - .... - ....), 'MANUFACTURER' (My organization 9258EY), 'ORGANIZATION' (My organization - 9258EY), and 'TEMPLATE NAME' (Sistema de irrigador). A 'FIRMWARE CONFIGURATION' section shows a code block with the following text:

```
#define BLYNK_TEMPLATE_ID "TMPL6kTaSqLV"
#define BLYNK_DEVICE_NAME "Sistema de irrigador"
#define BLYNK_AUTH_TOKEN "bdDNwLh4N0DTiEQVwCBvJfOXHVKtIqU_"
```

Below the code block, it states: 'Template ID, Device Name, and AuthToken should be declared at the very top of the firmware code.'