

**Controle do Documento**

**Histórico de revisões**

| **Data** | **Autor** | **Versão** | **Resumo da atividade** |
| --- | --- | --- | --- |
| 18/11/2022 | Grupo Koalyptus | 1 | Revisão e Seções: 1, 2, 3 |
| 02/12/2022 | Grupo Koalyptus | 2 | Revisão e Seções: 3, 4, 5 |
| 16/12/2022 | Grupo Koalyptus | 3 | Revisão e Seções: 6, 7, 8 |

**Índice**

[**1. Introdução**](#_heading=h.2et92p0) **[3](#_heading=h.2et92p0)**

[1.1. Solução](#_heading=h.tyjcwt) [3](#_heading=h.tyjcwt)

[1.2. Arquitetura da Solução](#_heading=h.3dy6vkm) [3](#_heading=h.3dy6vkm)

[**2. Componentes e Recursos**](#_heading=h.1t3h5sf) **[4](#_heading=h.1t3h5sf)**

[2.1. Componentes de hardware](#_heading=h.2s8eyo1) [4](#_heading=h.2s8eyo1)

[2.2. Componentes externos](#_heading=h.17dp8vu) [4](#_heading=h.17dp8vu)

[2.3. Requisitos de conectividade](#_heading=h.3rdcrjn) [4](#_heading=h.3rdcrjn)

[**3. Guia de Montagem**](#_heading=h.26in1rg) **[5](#_heading=h.26in1rg)**

[**4. Guia de Instalação**](#_heading=h.35nkun2) **[6](#_heading=h.35nkun2)**

[**5. Guia de Configuração**](#_heading=h.44sinio) **[7](#_heading=h.44sinio)**

[**6. Guia de Operação**](#_heading=h.z337ya) **[8](#_heading=h.z337ya)**

[**7. Troubleshooting**](#_heading=h.1y810tw) **[9](#_heading=h.1y810tw)**

[**8. Créditos**](#_heading=h.2xcytpi) **10**

# 

# 1. Introdução

## 1.1. Solução (sprint 3)

A proposta da solução é automatizar o monitoramento de temperatura e umidade dentro das casas de vegetação da Gerdau Florestal por meio de um sistema IoT robusto e resistente a condições de alta temperatura e umidade causada pela irrigação por nebulização.

Desse modo, o projeto faz integração em uma protoboard de um ESP 32 e um sensor de umidade e temperatura (AHT10), além de LEDs indicativos de status e um display LCD, tudo conectado com uma rede de internet, enviando os dados para um sistema em nuvem.

Assim, o dispositivo envia os dados em intervalos personalizados de tempo, de 1 em 1 minuto. Dessa maneira, o analista poderá utilizar essas informações para trabalhar os dados, gerar dashboards, identificar padrões e passar ao supervisor, que poderá designar alguém para ajustar as condições da estufa a fim de chegar na temperatura ideal de crescimento e desenvolvimento das mudas.

#### Benefícios da solução

1. Coleta de dados com maior confiabilidade, granularidade, padronização e automação;
2. Obtenção de informações consistentes de forma rápida, auxiliando na execução das tarefas com mais agilidade;
3. Melhoria na relação ergonômica e comunicação entre analista, coordenador e supervisor e demais trabalhadores na execução de atividades, resultando em menores erros;
4. Controle do estado do viveiro a partir de aferições frequentes sobre o status, tornando o que era manual em um processo mais fácil e eficiente;
5. Melhor alocamento do tempo de funcionários através do processo de automação, aumentando assim, a produtividade visando utilizar o tempo dedicado a demais atividades;
6. Diminuição de gastos devido a baixa mortalidade das mudas, tendo um viveiro mais rentável e com melhor aproveitamento de toda a estrutura existente;
7. Produção em grande quantidade e com qualidade superior, garantindo mudas mais uniformes e saudáveis;
8. Geração de maior valor para a operação, tendo em vista um modelo de produção de alto nível de conhecimento e inovação;
9. Exemplo de sustentabilidade, contribuindo para a diminuição do descarte de mudas;
10. Maior flexibilidade para situações adversas e atípicas referente ao clima e a sazonalidade.

## 1.2. Arquitetura da Solução (sprint 3)

A fim de migrar o abstracionismo para conceitos mais visuais, foi desenvolvido um diagrama de arquitetura do projeto em que é mostrado cada etapa, componente e função dos elementos presentes, tais como suas ligações referente ao dispositivo que mede a temperatura e umidade. A partir disso, é possível visualizar de forma mais sintética toda a estrutura do projeto de forma compreensível e organizada.

Com os requisitos levantados e os objetivos do produto, é possível elaborar o diagrama da solução a ser desenvolvida, permitindo a visualização dos microcontroladores, os sensores, a comunicação entre interface/controle e com o servidor, além de ilustrar os tipos de ligações entre esses elementos.

Link para o arquivo PDF do diagrama da arquitetura para visualização com mais qualidade: [Diagrama Koalyptus](https://drive.google.com/file/d/1ANz0T1VDLbKwKkDTqCYSDxxv2MQWGMC-/view?usp=sharing)

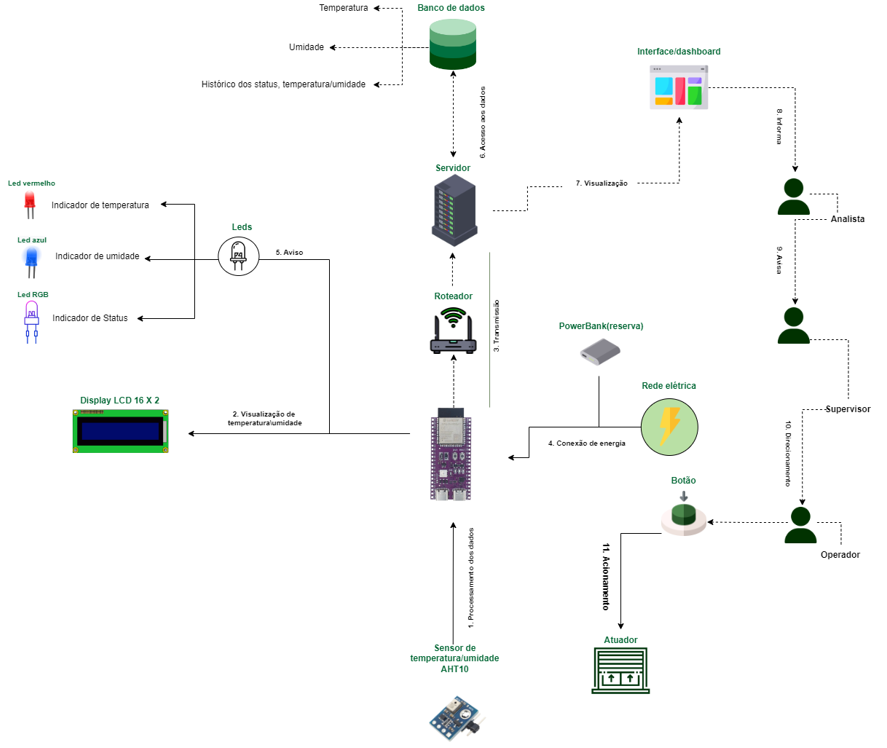
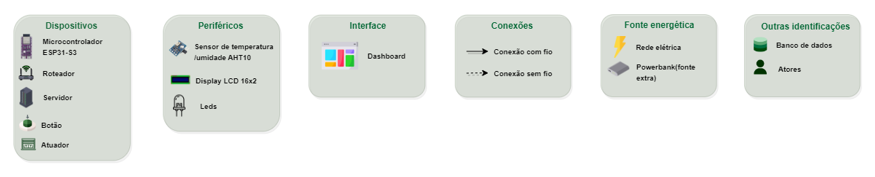


Diagrama da arquitetura



Legendas do diagrama

# 2. Componentes e Recursos (sprint 3)

## 2.1. Componentes de hardware

A seguir estão listados os componentes de hardware necessários para a montagem do dispositivo IoT.

| **Representação** | **Componente** | **Descrição** | **Tipo: entrada / saída / atuador** |
| --- | --- | --- | --- |
|  | ESP32 - S3 - WROOM - 1 | O ESP32 - S3 é um microcontrolador que ao ter conexão com os sensores, capta os dados e o armazena com maior velocidade de transmissão para o servidor e banco de dados. O microcontrolador possui a maioria dos principais componentes de um computador, tendo a função de controlar todo o sistema e o seu uso garante a implementação de um ambiente seguro e rápido processamento das informações. Ou seja, a partir do microcontrolador é possível acompanhar remotamente a umidade e a temperatura processadas e armazenadas através da transmissão. | Entrada/Saída |
|  | Display (LCD)/PCF8574T | A função do display é informar as medições dos sensores de temperatura e umidade, atualizando esta informação de minuto a minuto. Ademais, o display mostra o status da conexão com internet e algum erro com determinado sensor/ componente. A princípio, sua principal função é mostrar a temperatura/umidade para visualizar e consultar em tempo real diretamente no dispositivo. | Saída. |
|  | Conversor I2C | O módulo I2C pode ser considerado um conversor com função de manipular as conexões de um display. | Saída |
|  | AHT10 (sensor de temperatura/umidade) | É um sensor de tamanho 15x10x8mm que mede e coleta os valores da temperatura e da umidade do ambiente. A sua comunicação é do tipo I2C e é alimentado por uma tensão de 1,6V a 6V. As leituras de temperatura são na faixa de -40°C a 85°C, enquanto a umidade relativa é de 0% a 100%. O tempo de resposta para a temperatura é de 5s a 30 s e o de umidade é de 8s. Além disso, possui uma precisão de +/-2% para a umidade e de +/-0,3°C. | Entrada |
|  | Led Azul de 5mm | Enquanto as medidas de umidade estiverem dentro do padrão o led fica apagado, caso as condições fujam do ideal ele acende e pisca de forma intermitente. | Saída |
|  | Led Vermelho de 5mm | Enquanto as medidas de temperatura estiverem dentro do padrão o led fica apagado, caso as condições fujam do ideal ele acende e pisca de forma intermitente. | Saída |
|  | Led RGB QAPASS | Indica o status do dispositivo, quando o dispositivo estiver funcionando normalmente, o led RGB estará aceso emitindo a cor verde. No entanto, se ocorrer alguma falha no dispositivo, fazendo com que não funcione corretamente, o led RGB irá emitir a cor amarela e irá piscar constantemente. | Saída |
|  | Jumpers (macho-macho) | Responsáveis por realizar conexões entre os componentes do hardware. | n/a |
|  | Resistor - 1k ohm e 10k ohm, respectivamente | Componente que limita o fluxo da corrente elétrica em um circuito. Assim, por meio do chamado efeito joule, ele é capaz de transformar a energia elétrica em energia térmica, ou seja, o dispositivo faz oposição à passagem da corrente elétrica, oferecendo resistência. | n/a |
|  | Protoboard/ Breadboard - Minipa de 1680 | Considerada uma matriz de contato, ou placa de ensaio, é uma placa com furos de conexões condutoras para montagem de circuitos elétricos experimentais experimentais. | n/a |
|  | Power Bank | Fonte de energia portátil via rede/bateria conectado ao microcontrolador com o cabo USB. | Entrada |
|  | Cabo do tipo USB-C e USB. | Cabo com uma entrada sendo USB-C, para se conectar com o ESP 32, e a outra entrada deve ser similar ao do Power Bank ou computador, depende de quem estiver utilizando, no nosso caso utilizamos USB. | n/a |

## 

## 2.2. Componentes externos

A seguir estão listados os componentes externos que fazem parte da solução, além de eventuais serviços em nuvem, softwares, entre outras aplicações utilizadas.

| **Representação** | **Componente** | **Descrição** | **Tipo: entrada / saída / atuador** |
| --- | --- | --- | --- |
|  | Interface WEB | Utilizando a interface WEB, o parceiro terá acesso a uma visualização da situação atual de cada dispositivo. Ela receberá informações atualizadas do banco de dados. | Saída |
|  | Roteador wifi | É o que fará a conexão do dispositivo com a internet, dessa forma, permitindo que ele envio dados para o banco de dados. | Atuador |
|  | Computador | O computador é responsável por configurar os dispositivos e fazê-los funcionar. | Atuador |
|  | Arduíno IDE | A função do Arduino IDE é simples, no computador, ele se encarrega de compilar o código em C++ e enviá-lo para o dispositivo, além de poder ser usado como debugger para acompanhar o que acontece no sensor. | Atuador |
|  | HTML | HTML é a linguagem de marcação de hipertexto utilizado na construção da interface WEB. | Atuador |
|  | CSS | O CSS é a linguagem de folha de estilo em cascata usada na estilização do HTML da interface WEB. Ele serve principalmente para criar uma visualização aconchegante e fácil de entender e utilizar. | Atuador |
|  | Google Excel | Plataforma do google usada como banco de dados provisório, onde são guardadas as informações coletadas pelo dispositivo e distribuídas, utilizando seu API embutido, à interface WEB. | Atuador |
|  | JavaScript | Linguagem de programação utilizada para funções simples na interface WEB como troca de telas e para pegar informações do banco de dados. | Atuador |
|  | C++ | Linguagem de programação utilizada no microcontrolador Esp32-S3 para funções de Leds, display, sensores do dispositivo e para o envio de informações ao banco de dados. | Atuador |

## 2.3. Requisitos de conectividade

### Versão 1.1 (Sprint 3 )

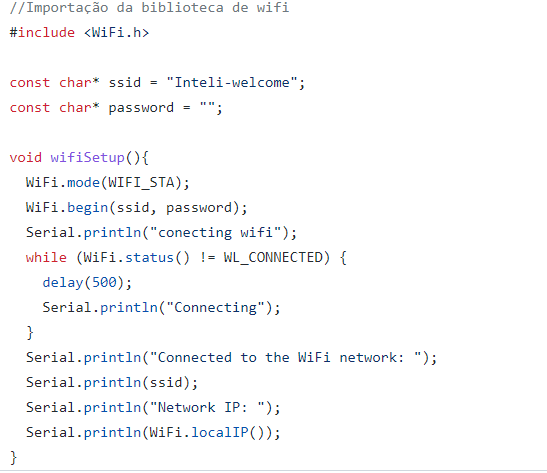
Os requisitos de conectividade envolvem as redes, protocolos de rede e especificações do back-end para o funcionamento dos dispositivos. Na primeira conexão é possível citar a energia, que precisa estar conectada ao ESP 32 para que ele possa distribuir a corrente para a protoboard e assim, conseguir ligar todos os componentes do microcontrolador, incluindo o AHT 10, os LEDs, do Display LCD e outros.

Por outro lado, existe a conexão da rede Wifi, a qual o ESP 32 se conecta para conseguir mandar as informações coletadas pelo sensor AHT 10 para o banco de dados e para que toda a interface web possa funcionar com êxito. Desse modo, para que as informações sejam enviadas para este banco de dados, utilizamos o protocolo HTTP, que é um protocolo de comunicação utilizado para para sistemas de informação de hipermídia distribuídos, permitindo a comunicação de transmissão de páginas web de um servidor para o navegador, no caso, que estabelece como deve ser feito o GET e POST feitas na Backend efetivamente, para que todo o sistema consiga se interligar, permitindo exibir informações dos sensores para o front-end e o envio de informações para o banco de dados.

Já sabendo como funciona o protocolo de conectividade para realmente ter acesso aos dados é necessário realizar os seguintes passos:

##### 1° Passo

* Para ter acesso a rede de wifi é preciso acessar o código fonte do sistema e entrar na biblioteca wifi.h .
* Alterar as variáveis ssid e password com o nome e senha da rede wifi que você deseja utilizar o sistema.
* Após isso, basta compilar o código e enviar para o ESP-32 que já estará funcionando.



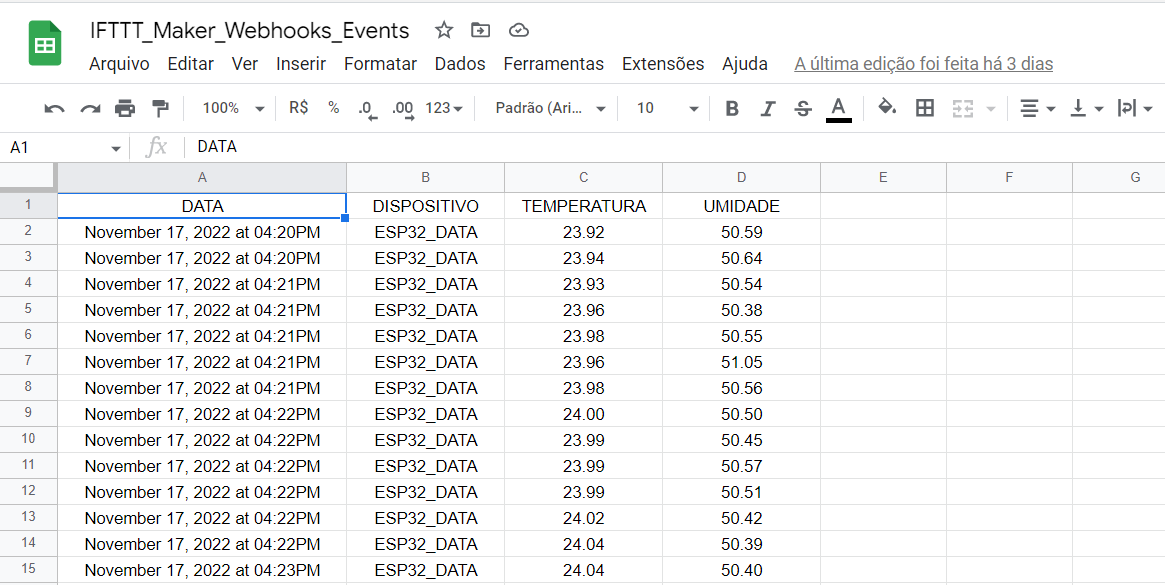
Assim, o sensor já conectado a rede wifi do local e caso esteja ligado a alguma fonte de energia já está coletando e enviando os dados.

Com isso, é possível acessar uma planilha com todas as medições já realizadas.

##### 

##### 2° Passo

* Acesse o seguinte link e tenha acesso a todas as medições realizadas pelo ESP32
* <https://docs.google.com/spreadsheets/d/1jP9yXgPAHUr0RjDQ1sTmqObHoKnP-r0I6hto-3vMw80/edit?usp=sharing>



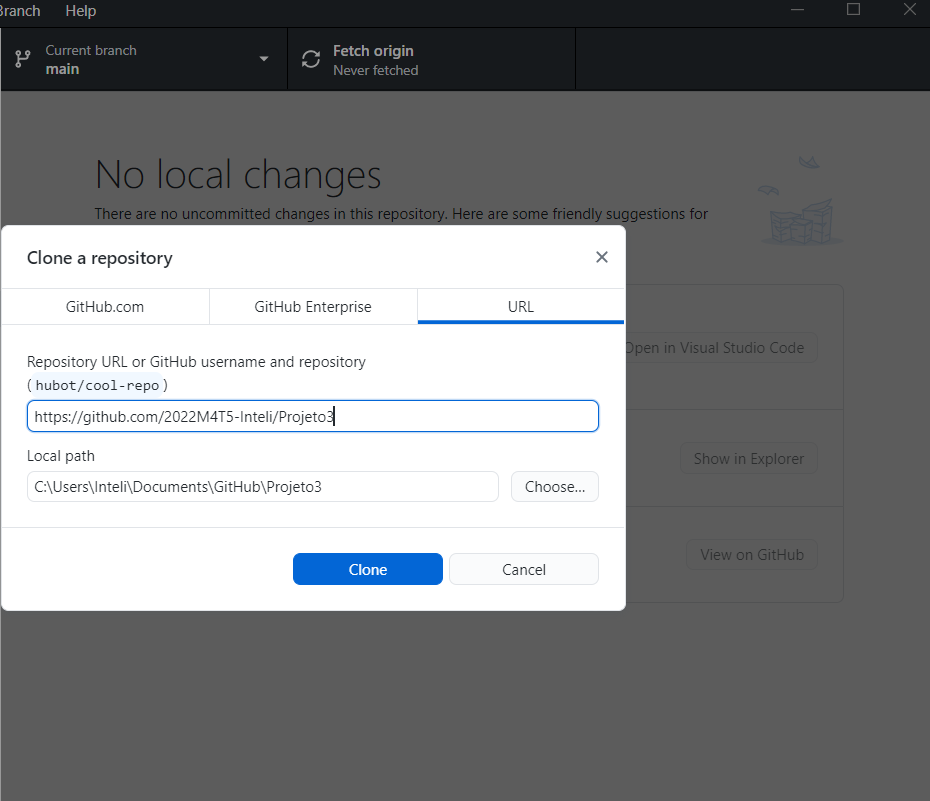
Também é possível ter acesso ao frontend do site. E consultar as medidas em tempo real .

###### 

##### 3° Passo

Acesse o github desktop e clone o seguinte repositório:

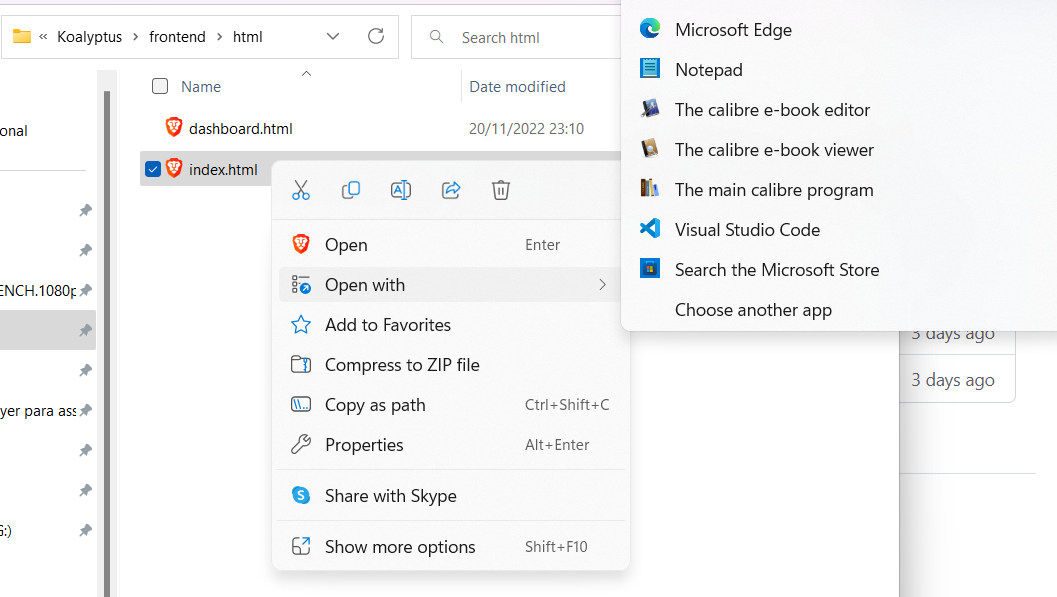
<https://github.com/2022M4T5-Inteli/Projeto3>



###### 

##### 4° Passo

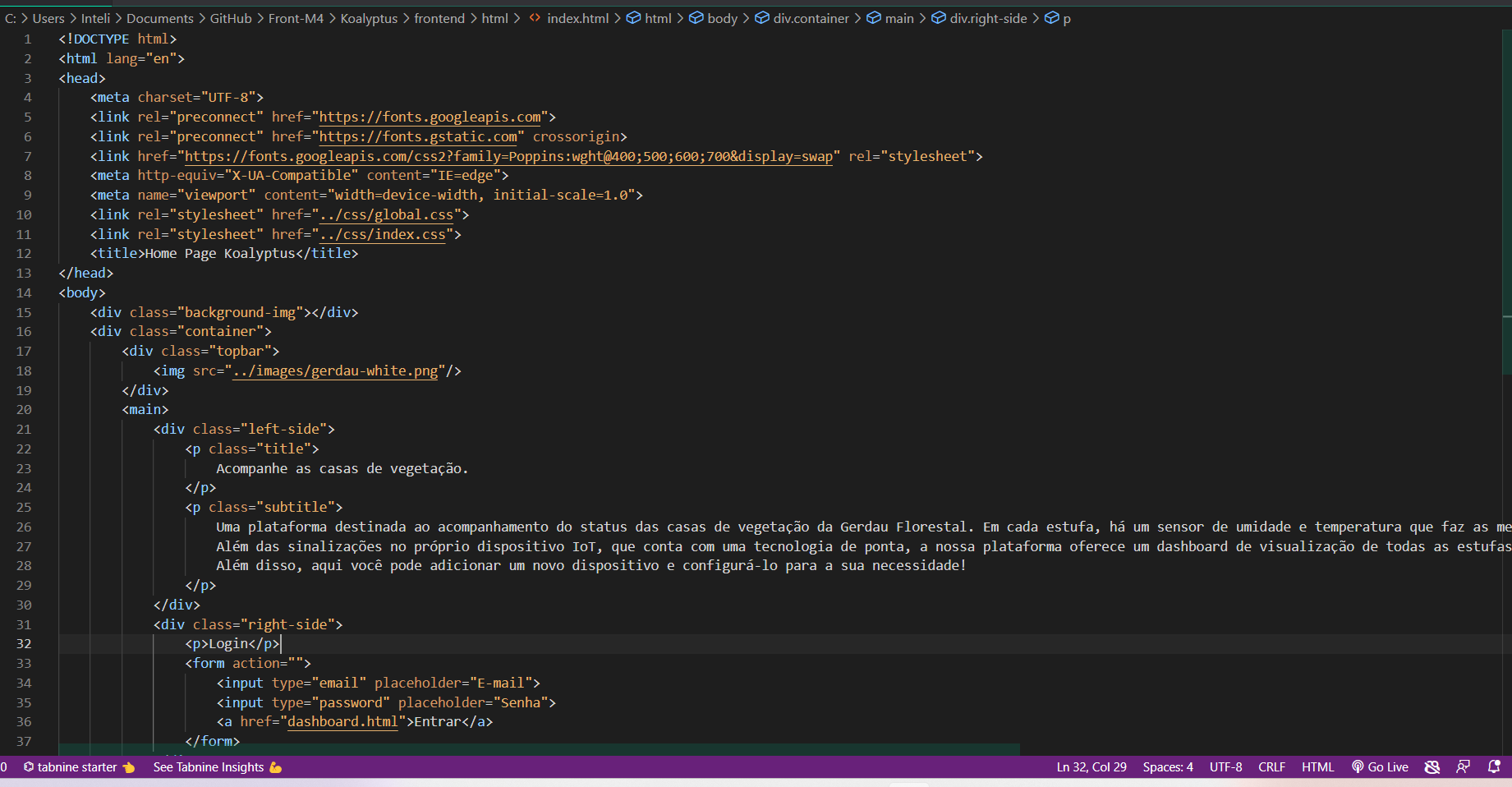
* Encontre a pasta onde foi clonado o projeto do github
* E siga a seguinte ordem de pastas
* Front Koalyptus >Koalyptus > frontend > html >
* Execute o seguinte arquivo com o Visual Studio Code
* index.html



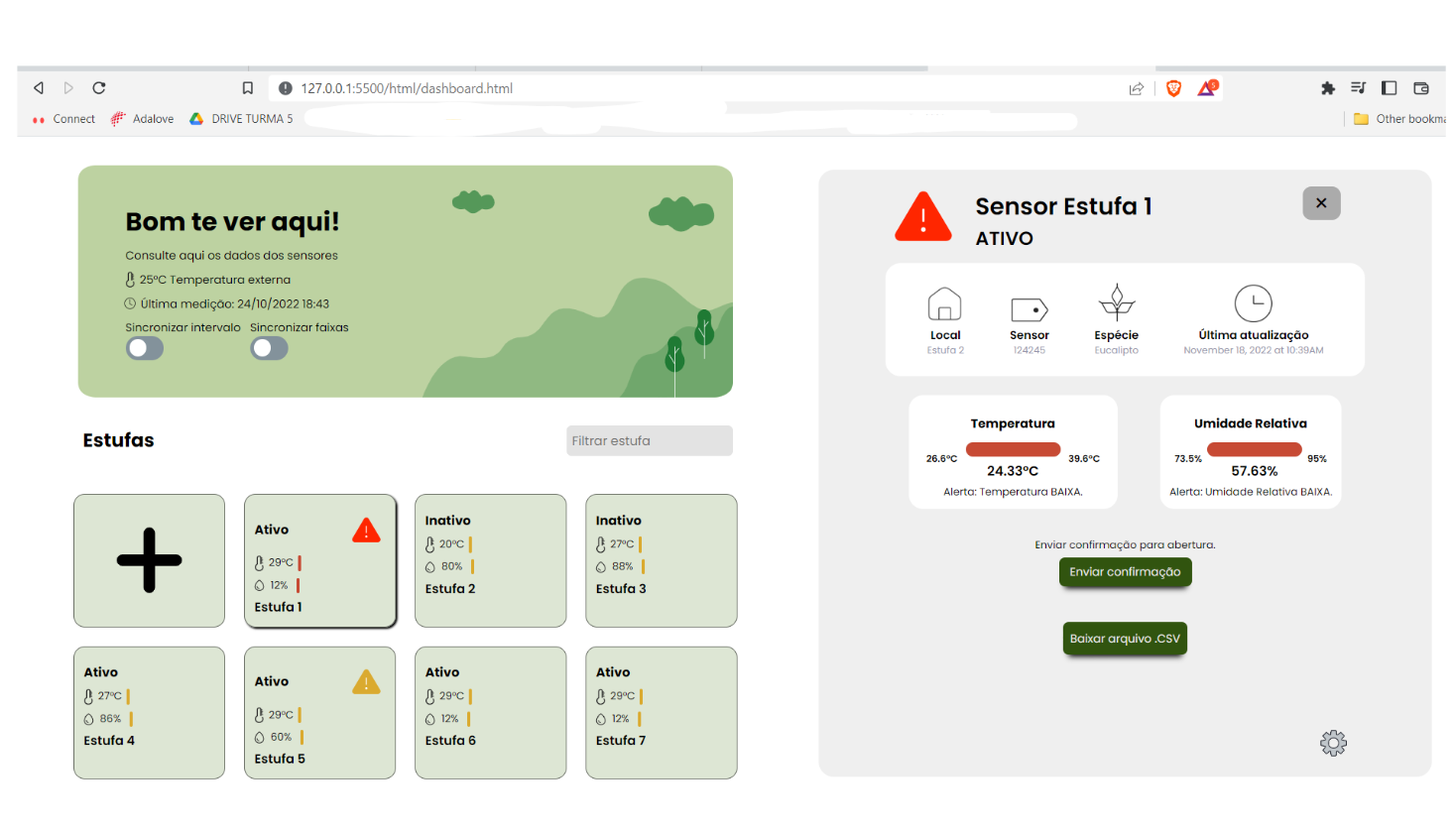
###### 

##### 5° Passo

* Execute a página como servidor live a partir do VSCode
* Basta clicar no botão roxo “Go live” que está localizado no canto inferior direito da tela .



Pronto o site já está rodando e é possível acompanhar todas as informações.



|  |
| --- |

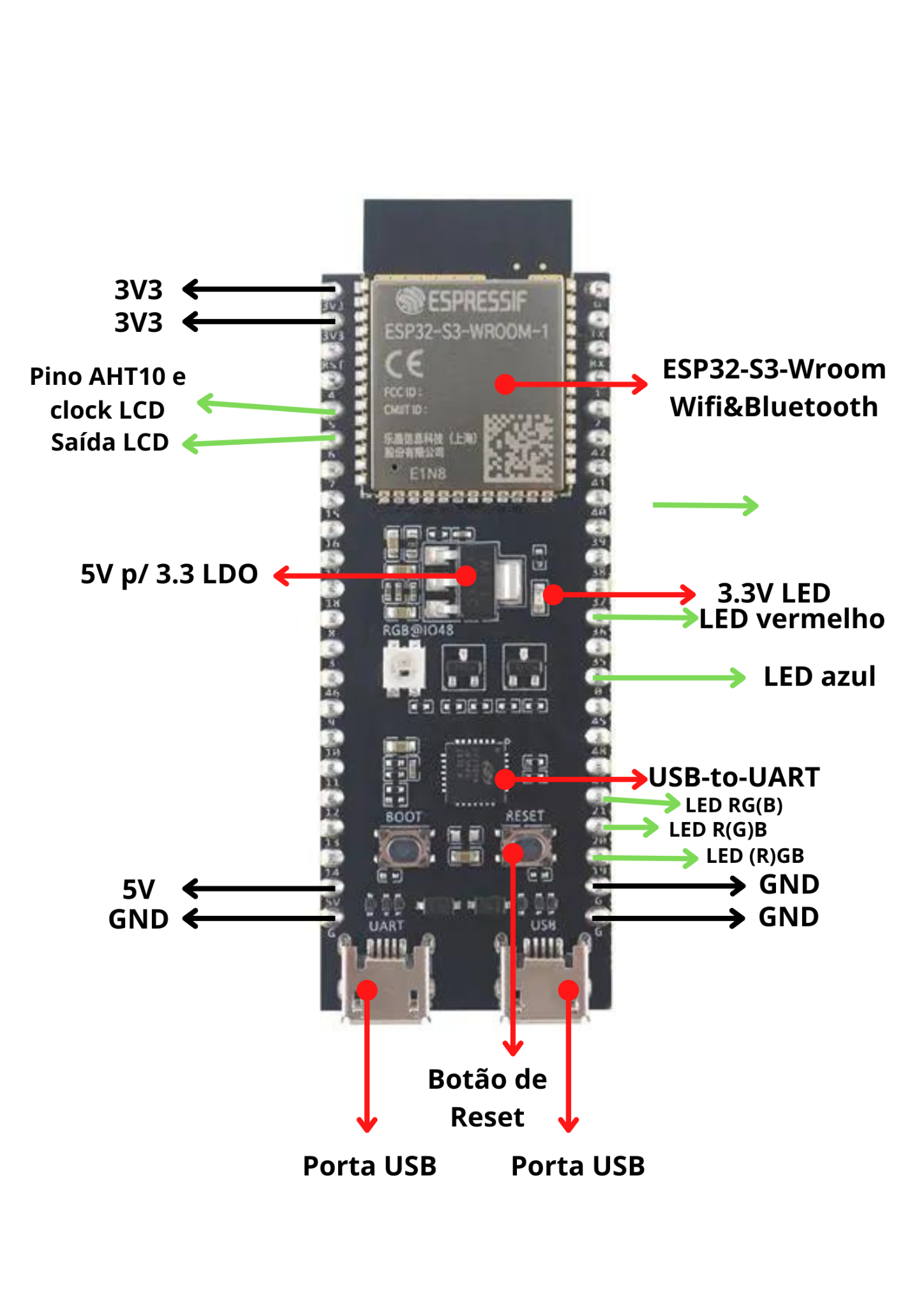
**3. Guia de Montagem**

Neste guia, será apresentado o passo a passo para realizar a montagem do sistema IoT. Antes de tudo, será apresentado o ESP 32-S3 com todas as conexões necessárias (**Figura 1)** e, posteriormente, as etapas de forma detalhada para o processo de construção.

## 3.1 ESP 32-S3

Na figura ao lado (**Figura 1**) é apresentado todas as conexões necessárias para o sistema IoT funcionar. A organização é da seguinte maneira:

* Pino 4: AHT10 e Clock LCD;
* Pino 5: Saída LCD;
* Pino 37: LED vermelho;
* Pino 36: LED azul;
* Pinos 19, 20, 21: RGB;
* GND: ligação dos componentes;
* 5V e 3.3V: alimentação.



**Figura 1 - Detalhamento do ESP32-S3 e suas portas**

### 3.1.1 Protoboard

Para iniciar a montagem, primeiramente é preciso de uma protoboard, como é demonstrado na Figura. Desse modo, alinhamos as etapas conforme as figuras a seguir:

|  |  |  |
| --- | --- | --- |

**Fonte: Ded Componentes**

A área em vermelho é a área onde o microcontrolador deve ser posicionado. O microcontrolador, ESP 32-S3, deve ser posicionado no centro da protoboard de forma que os pinos da direita e da esquerda não se conectem e que sobre espaço para utilizá-los em ambos os lados. Lembrando que o ESP 32 precisa se encaixar na protoboard e o encaixe não deve ser feito nas regiões positiva/negativa.

## 

## 3.2 Implementação dos componentes de Hardware

Nesta seção é apresentado a implementação dos componentes já apresentados na seção 2.1.

### 

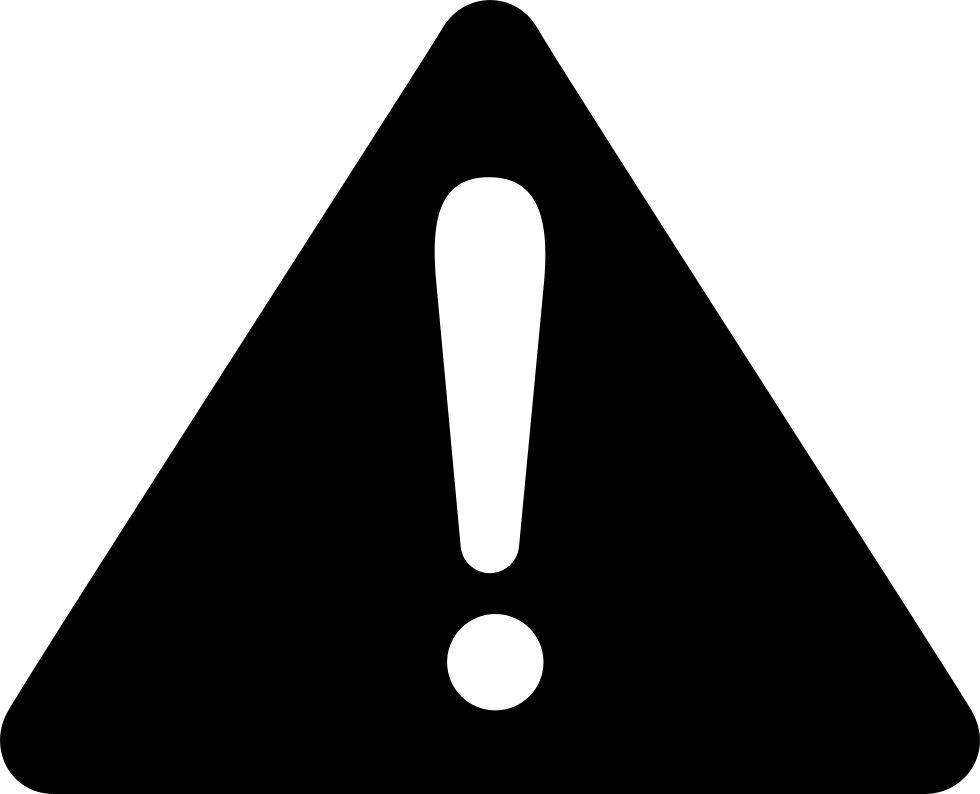
### 3.2.1 LEDs

Temos 3 LEDs diferentes, vermelho, azul e RGB, que têm a capacidade de ascender múltiplas cores diferentes. Os LEDs vermelho e azul tem dois pinos, um positivo e outro negativo. O pino de menor comprimento deve ser ligado a um resistor de 220 ohms, que por sua vez, deve estar conectado ao GND do microcontrolador, utilizando um jumper macho-macho, já o pino de maior comprimento deve ser ligado aos pinos de número 36 e 37, nos Leds azul e vermelho, respectivamente, como demonstrado na Figura 3.

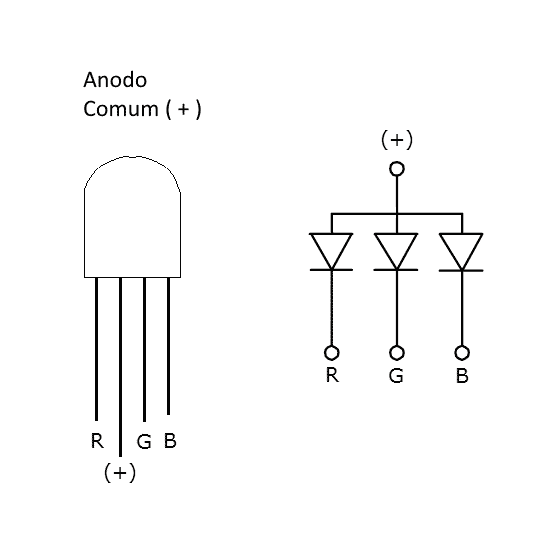
Para este dispositivo, utilizamos o LED RGB de ânodo comum, que possui 4 pinos diferentes. O maior pino deve ser conectado à fonte de energia 3v3 do microcontrolador. Os outros três pinos devem ser conectados a resistores diferentes de 220 ohms, e estes, devem estar conectados nas portas 19, 20 e 36 do microcontrolador, onde a porta de número 19 servirá para alimentar a luz vermelha, o de número 20 para a luz verde e o de número 21 para a luz azul.

**Divisão de cores por fios**

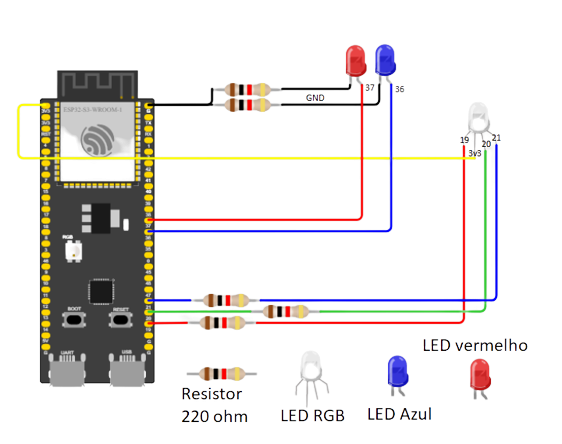
| fio GND(terra): preto |
| --- |
| Fio Led azul: azul(36, GND) |
| Fio Led vermelho: vermelho(37, GND) |
| fios Led RGB: **R** vermelho(19), **G** verde(20), amarelo(3V3), **B** azul(21) |

Atenção, as cores dos fios foram utilizadas para fins organizacionais e de compreensão, portanto, não é obrigatório seguir as cores abordadas.

Aqui há uma demonstração das hastes do LED RGB e como foi conectado adequadamente:

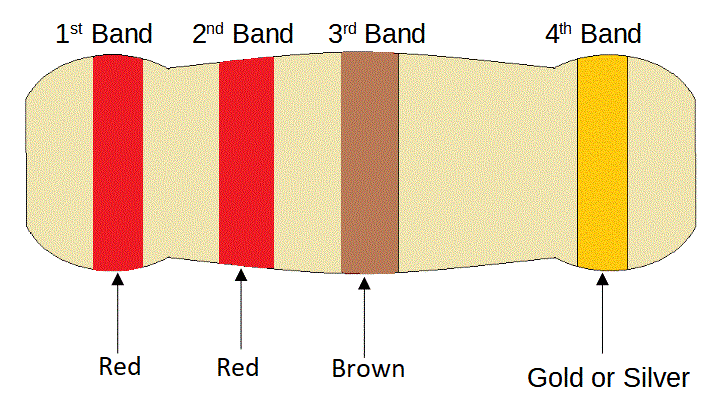


Abaixo é possível visualizar a conexão dos leds em seus respectivos pinos e com as cores disponíveis anteriormente na tabela:



**Figura 3 - Implementação dos Leds**

É Importante se certificar de que cada led precisa ter um resistor em seus catodos(perna negativa do led). O resistor usado e que já foi citado é o de 220 Ohm que, fisicamente, possui as seguintes cores:

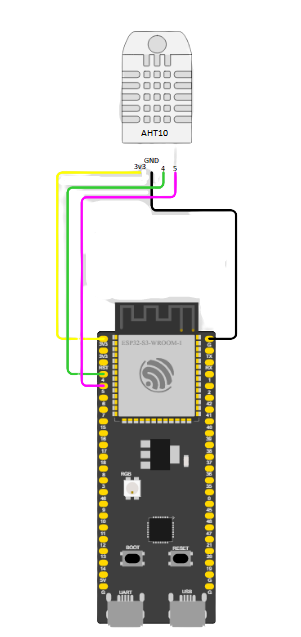


Fonte: Eletronics reference

### 

### 3.2.2 Sensor AHT 10

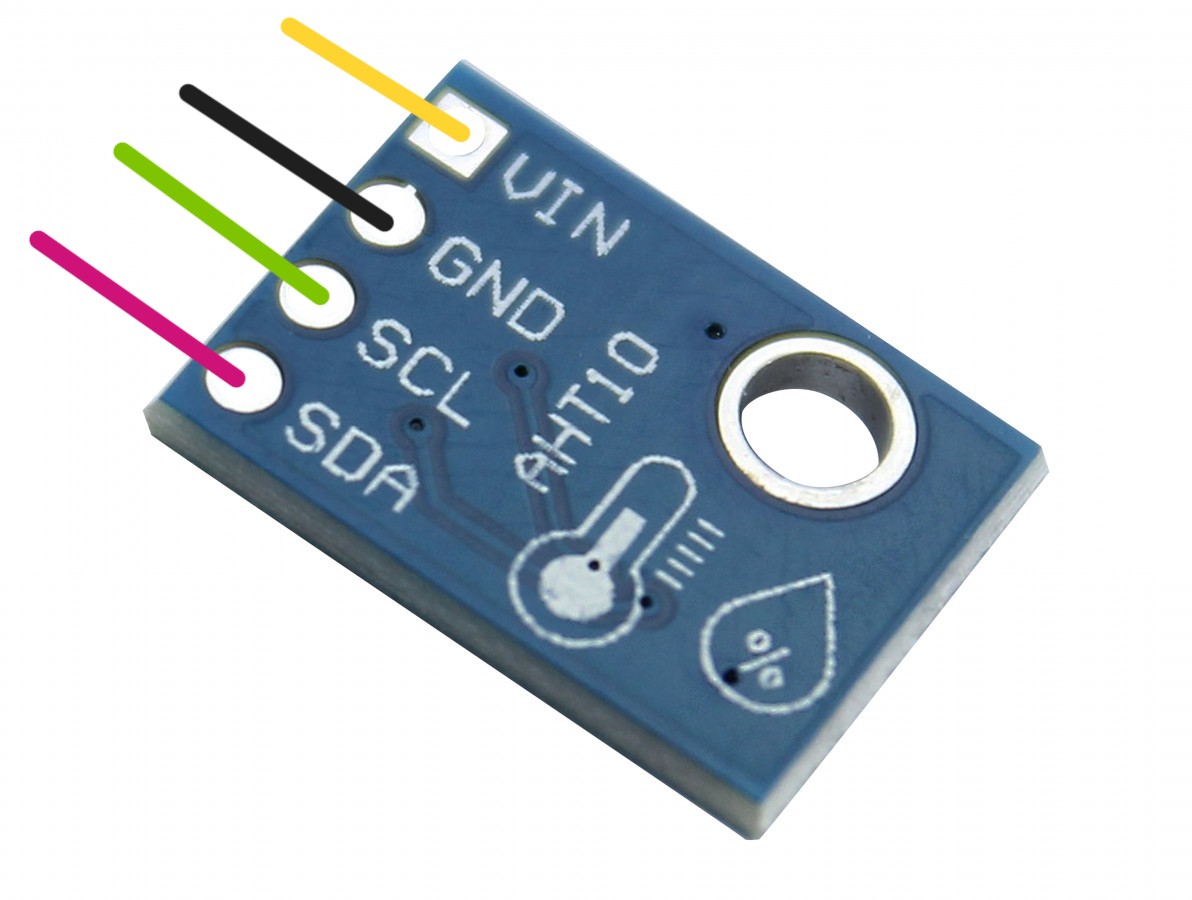
O sensor AHT 10 possui 4 pinos, o VIN, GND, SCL e SDA, da esquerda para a direita, respectivamente. Utilizando jumpers macho-macho, conectamos o pino VIN a uma saída 3V3, o GND do sensor a um GND do microcontrolador, o SCL ao pino de número 4 e o SDA ao pino de número 5 no ESP 32-S3 como na Figura 4.



**Figura 4 - Sensor de temperatura/umidade AHT10**

Lembrando que no simulador foi usado o DHT22, enquanto no projeto físico foi feito uso do AHT 10, mas isso não compromete o processo de montagem.

Abaixo o sensor tem as seguintes conexões e suas respectivas portas com base na organização dos fios por cores:



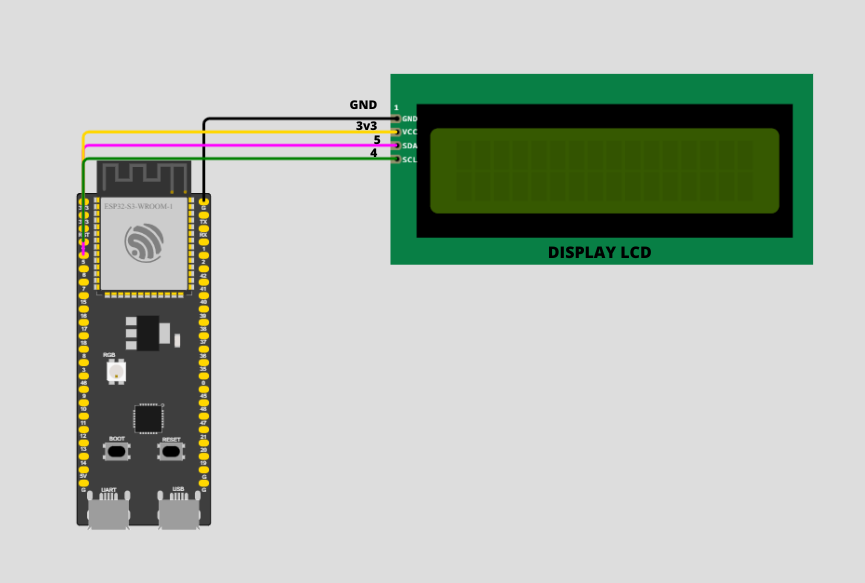
Fonte: RS Robótica

| VIN - 3V3 | SCL - porta 4 |
| --- | --- |
| GND | SDA - porta 5 |

### 

### 3.2.3 Display LCD

O display LCD consta com 1 pino de energia (VCI), 1 pino terra (GND), 1 pino SCL e 1 pino SDA. Utilizando jumpers, o pino de energia (VCI) deve ser conectado a uma saída 3v3 do ESP32-S3. O pino terra (GND) se conecta ao GND do mesmo. O pino SCL e SDA, como no sensor AHT10, devem ser conectados, respectivamente, aos pinos de número 4 e 5, como na Figura 5.



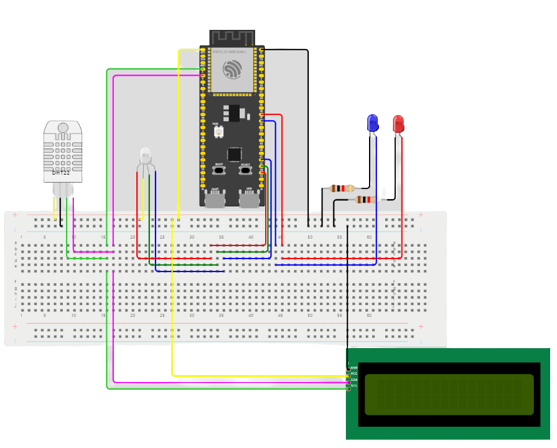
**Figura 5 - Display LCD/PCF8574T**

**Lembrete: O LCD não precisa de encaixe na protoboard, apenas ter as devidas conexões.**

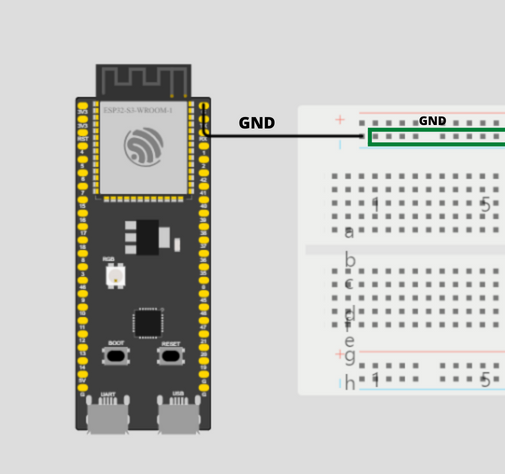
Tabela de organização por cores em relação a figura 5 (conexões):

| GND | SDA - porta 5 |
| --- | --- |
| 3V3 - alimentação | SCL - porta 4 |

No fim, a construção do dispositivo deve estar parecida com a figura abaixo.



Vale destacar que, na protoboard física, é interessante utilizar fios jumpers ao se tratar do GND, este fio é conectado ao GND e ao encaixe negativo da placa, fazendo com que toda a fileira negativa da protoboard é transformada em alimentação GND, permitindo que diversas conexões sejam feitas com maior organização. Exemplo:

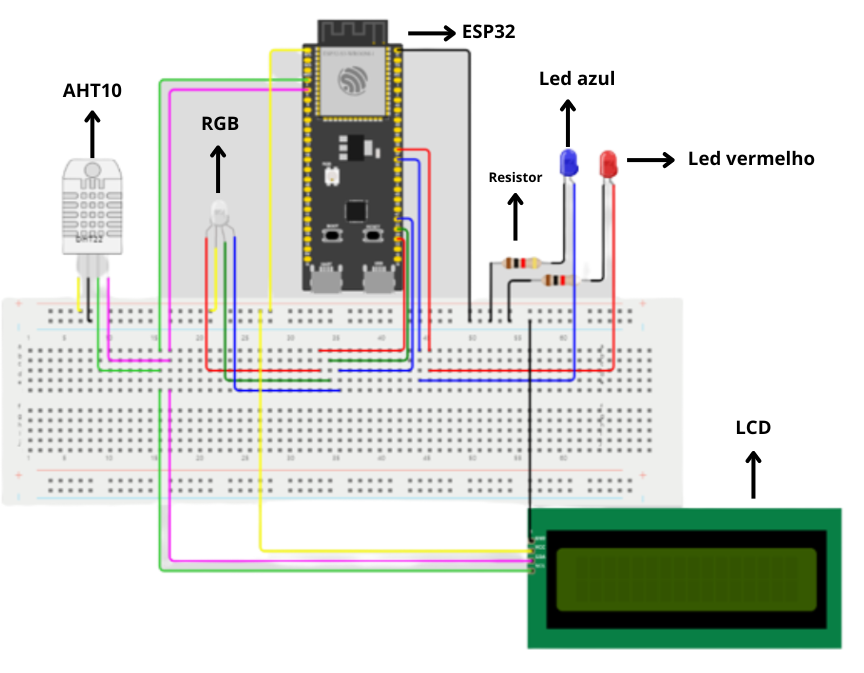


A área em verde significa que, após ser feita a conexão do GND com o encaixe negativo, a área contornada(fileira negativa) se transforma em alimentação GND.

### 

### 3.2.4 Resumo

Resumo do dispositivo com a montagem completa e organização visual tabelada:



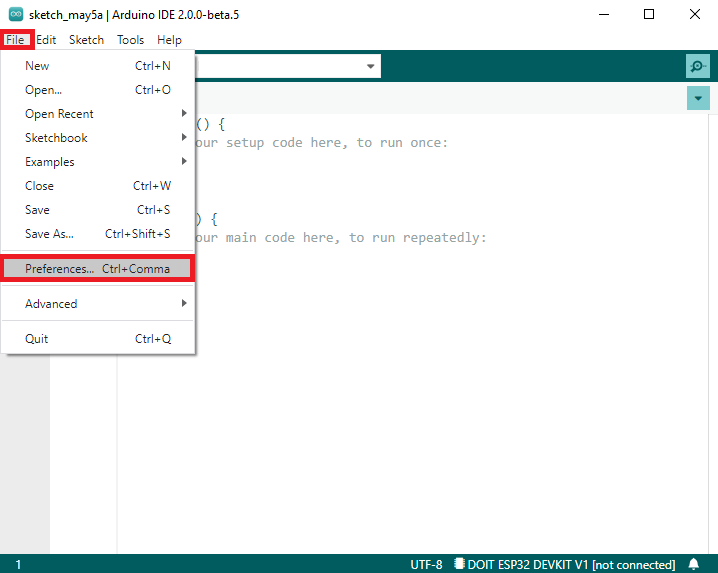
|  | **Componente** | **Portas(ESP32)** | **Fios (cores)** |
| --- | --- | --- | --- |
|  | LED azul | Azul: haste maior 36  Preto(GND): haste menor |  |
|  | LED vermelho | Vermelho:haste maior 37  Preto(GND):haste menor |  |
|  | LED RGB | Vermelho: **R**  Verde: **G** 20  Azul: **B** 21  Amarelo: 3V3 |  |
|  | AHT10 | Preto: GND  Amarelo: VIN 3V3  Verde: SCL 4  Rosa: SDA 5 |  |
|  | LCD | Preto: GND  Vermelho: 3V3  Verde: SDA 5  Azul: SCL 4 |  |

### 

### 3.3 Arduíno IDE

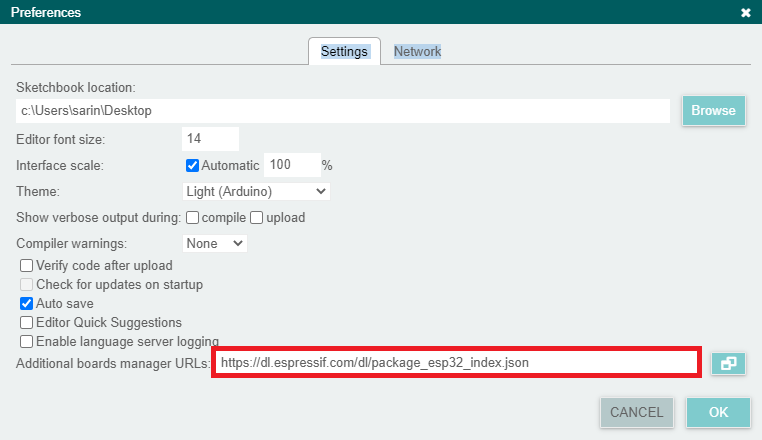
Em seu computador, instale do site “<https://www.arduino.cc/en/software>” o Arduíno IDE. Este aplicativo é responsável por passar informações de códigos ao microcontrolador. Após a instalação, abra o programa, vá em “File” e selecione a opção de preferências, como na Figura 4. Lá você se depará com um campo com um link. Apague o link já existente e substitua-o com “<https://dl.espressif.com/dl/package_esp32_index.json>” como a Figura 5, e clique em “ok”.

Figura 4



fonte: <https://randomnerdtutorials.com/installing-esp32-arduino-ide-2-0/>

Figura 5



fonte: <https://randomnerdtutorials.com/installing-esp32-arduino-ide-2-0/>

Voltando à tela inicial, clique no ícone do microcontrolador, como na Figura 6. Lá você deve digitar Esp32 e instalar a versão disponibilizada pelo “Espressif Systems”.

Figura 6

# 

fonte: <https://randomnerdtutorials.com/installing-esp32-arduino-ide-2-0/>

### 3.3.1 Bibliotecas e configurações

Com o Arduíno IDE instalado, na tela inicial clique no ícone de bibliotecas, indicado na Figura 7. Lá, você deve instalar uma série de bibliotecas. Para instalá-las, digite no campo de pesquisa “Adafruit AHT10” e “liquidcrystal frank” por Frank de Brabander e clique no botão “install”, como visto na Figura 8.

Figura 7



Figura 8

****

### 

### 3.3.2 Código e inicialização

Com o ambiente configurado, o microcontrolador conectado ao computador e as bibliotecas instaladas, clique em “Board Manager” e selecione o dispositivo Esp32-S3 Dev Module, selecionar a porta conectada no Arduíno IDE e carregar o código fornecido clicando no botão de upload, como mostrado na Figura 9 e Figura 10.

Figura 9

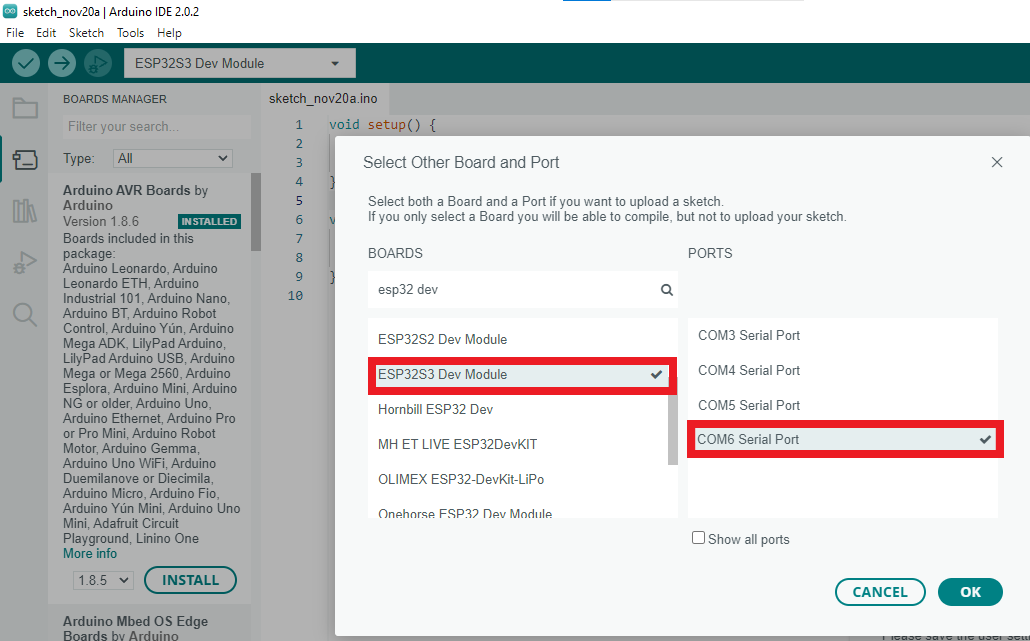
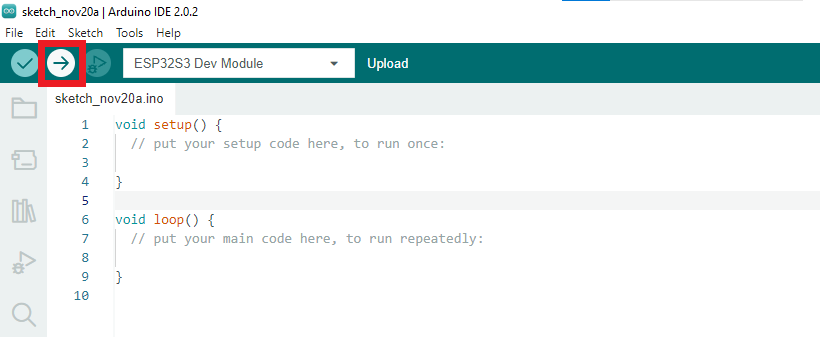
****

Figura 10

****

# 

### 3.4 Interface IoT

Para cada caso de leitura de temperatura e umidade, bem como casos de erro, a interface do dispositivo apresentará uma resposta específica nos LEDs e no Display LCD, que foi montado seguindo os parâmetros de faixa de temperatura e umidade informados pelo parceiro e que podem ser vistos no diagrama presente na imagem 1. Os casos podem ser observados nas figuras que seguem.

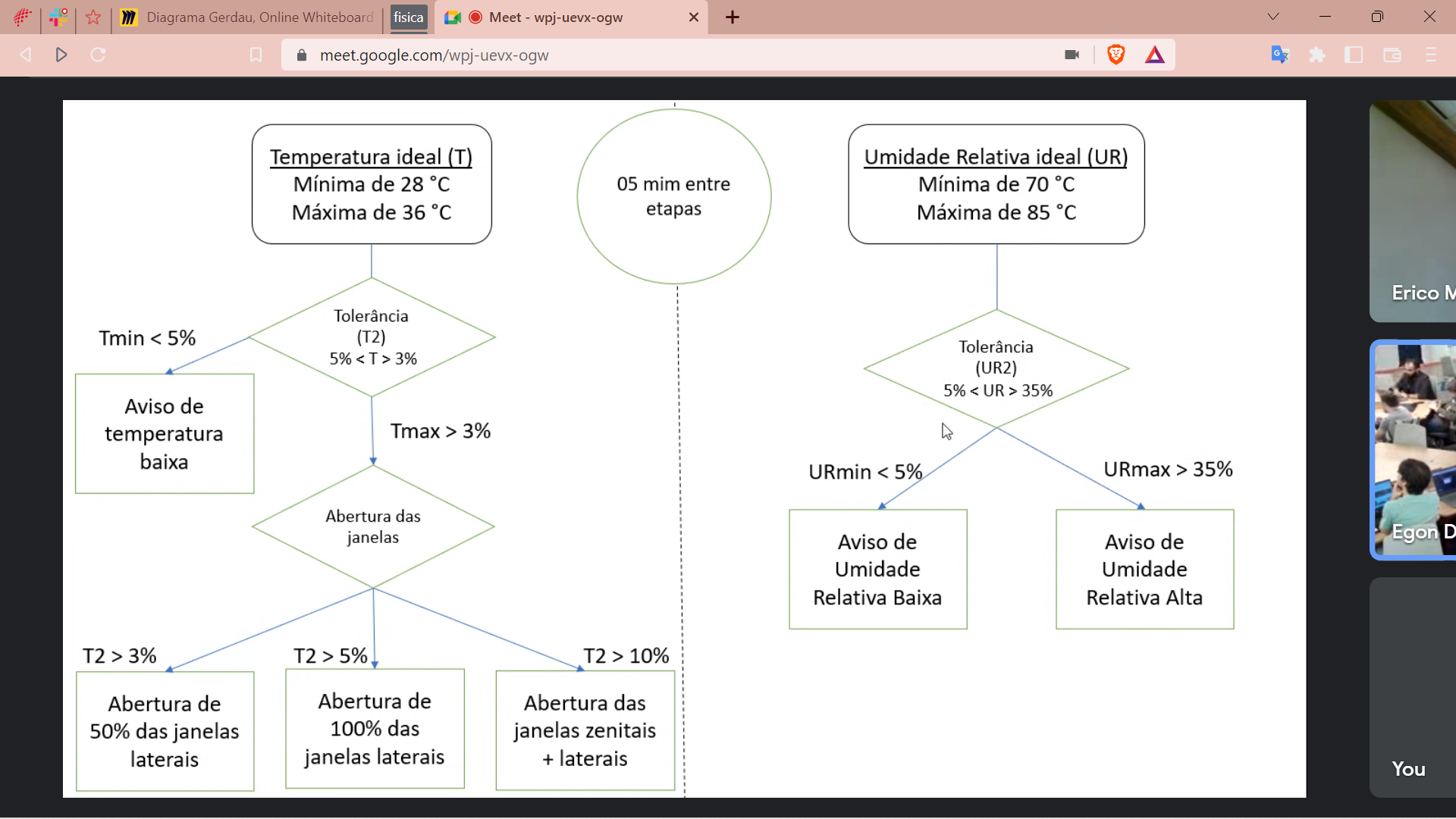
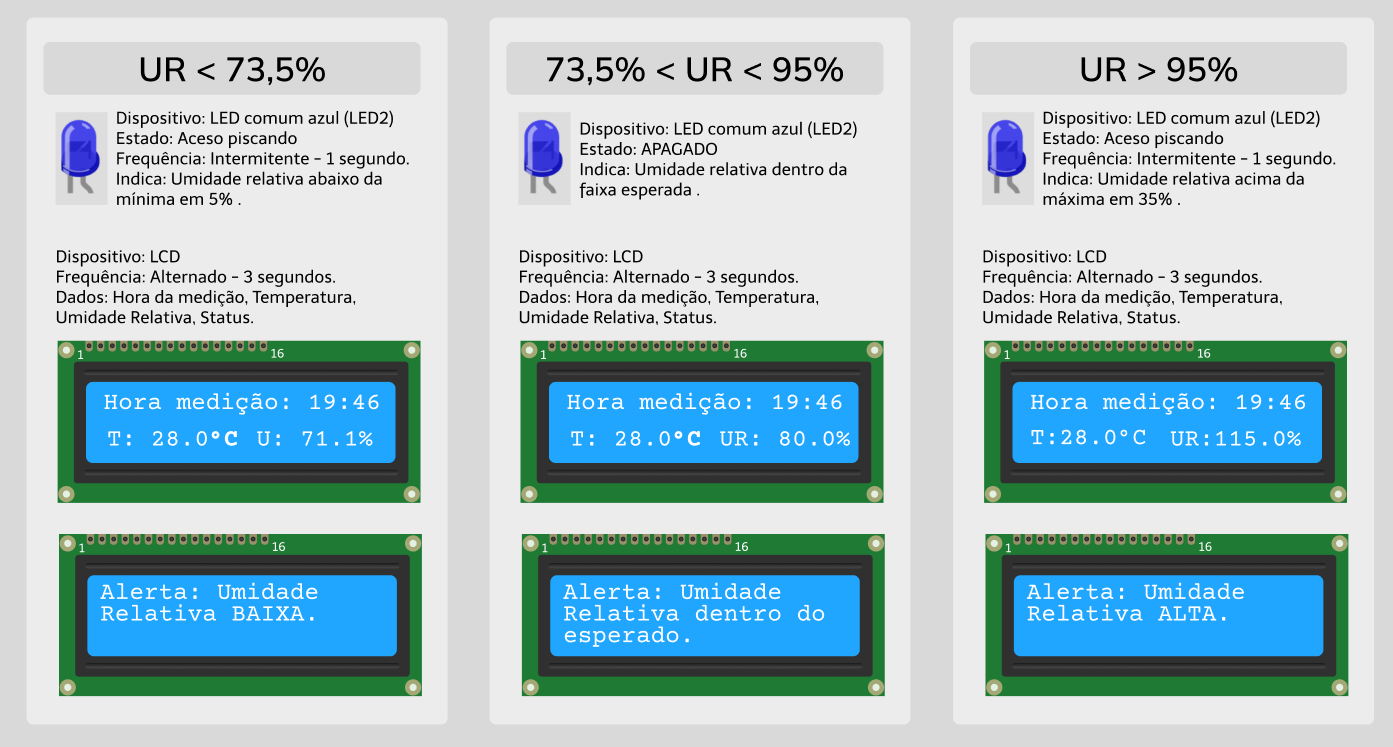


Imagem 1: parâmetros informados pelo parceiro

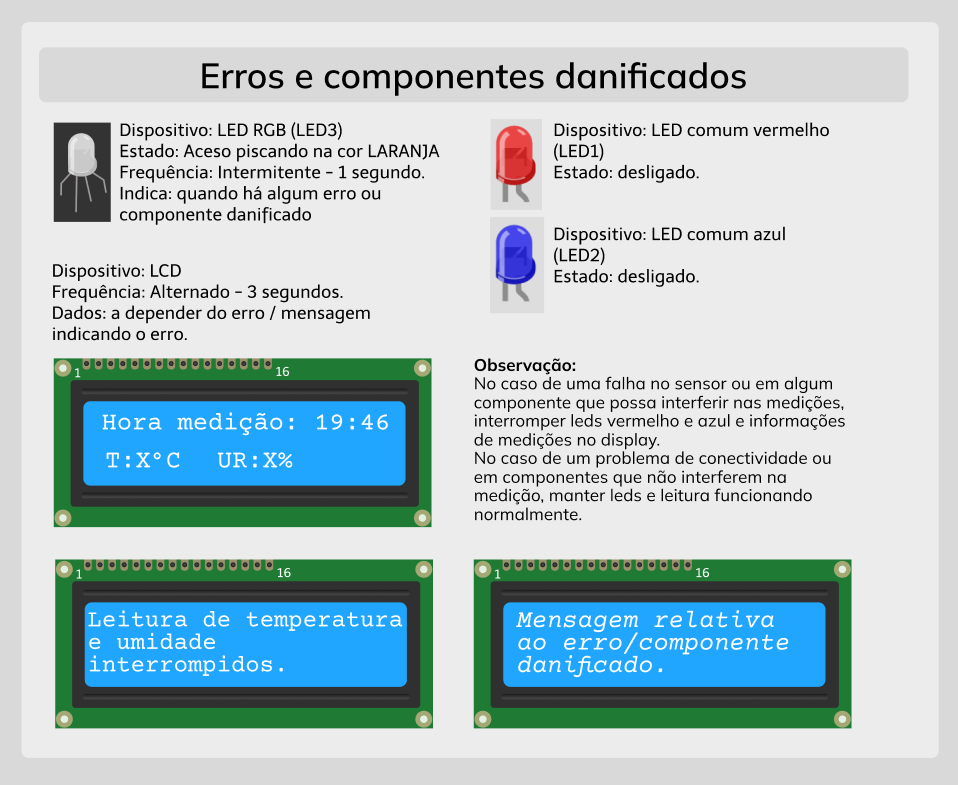
### 

### Legenda: esquema de LED e Display para as diferentes medições de temperatura, seguindo as faixas indicadas.



### Legenda: esquema de LED e Display para as diferentes medições de umidade, seguindo as faixas indicadas

# 

Legenda: esquema para o caso de haver algum tipo de erro/dano, em conjunto com as medições de temperatura e umidade.

### 

### 3.4.1 Códigos de erro:

***Erros relacionados ao sensor:***

**ERR\_01 :** Quando o sensor não consegue ler os dados de temperatura e/ou umidade e/ou o microcontrolador não recebe os dados do sensor.

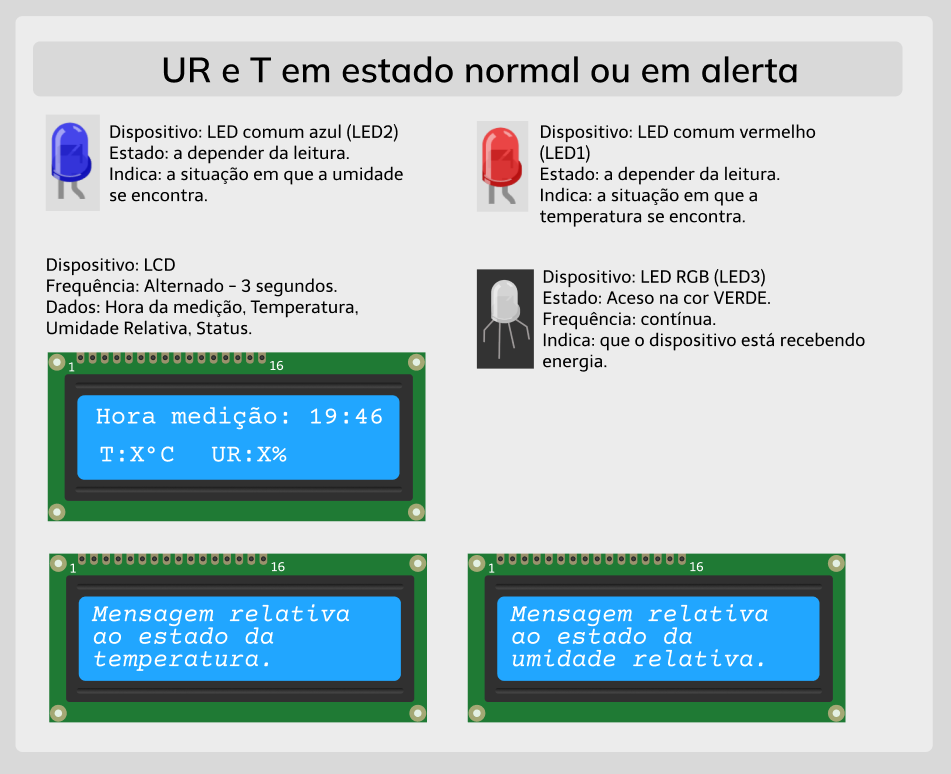
. **ERR\_02:** Quando o sensor começa a ter medidas instáveis ou muito extremas que saiam muito do padrão.

***Erros relacionados ao microcontrolador:***

**ERR\_03:** Quando o sensor não for capaz de conectar-se à rede wifi ou quando ela estiver sem internet.

**ERR\_04:** Quando houver qualquer tipo de problema na conexão entre o servidor web e o dispositivo, tais que incluem: queda de internet, falta de energia, demora no recebimento de dados, etc. Este código de erro será informado na interface web.

**ERR\_05:** Quando o microcontrolador estiver tendo dificuldade para enviar os dados coletados e não ser capaz de descobrir o motivo.



Legenda: esquema de LEDs e Display para o caso do dispositivo estar recebendo energia, em conjunto com umidade e temperatura

# 

# 4. Guia de Instalação

**1º Passo**

Conecte o dispositivo a uma fonte de energia, utilizando a porta “COM” do ESP-32. Quando o dispositivo iniciar, em seu computador, conecte-se à rede wifi “Koalyptus” e utilize a senha “koalaslyptus” para entrar.

**2º Passo**

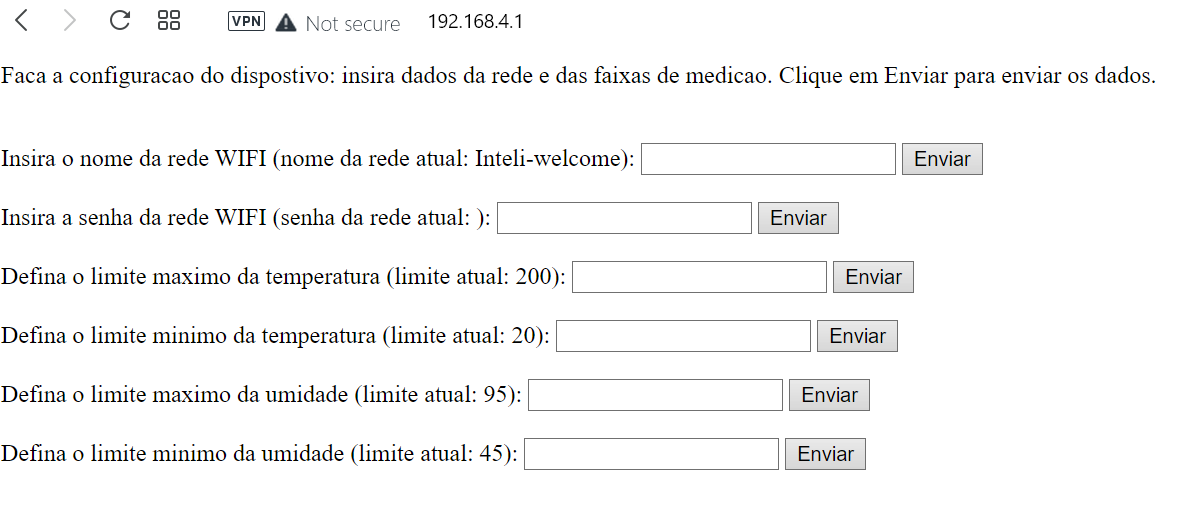
Quando estiver conectado, abra seu navegador e vá para o URL “192.168.4.1”. Lá você encontrará dois campos de texto, onde deve então inserir o nome da rede wifi externa e a senha. Após preencher as barras de texto, clique no botão “Enviar” para que sejam salvos no microcontrolador.

**3º Passo**

Depois disso, é só reiniciar o dispositivo, desconectando-o e conectando novamente à fonte de energia, e o dispositivo se conectará automaticamente à rede wifi. Caso o dispositivo não se conecte, é possível que o nome ou senha estejam digitados de forma incorreta. Neste caso, é só repetir o processo desde o 1º Passo e colocar as informações certas.

**Limite temperatura/umidade**

A tela de configuração pode também ser utilizada para definir os limites de temperatura e umidade relativa para os pontos de atenção que o dispositivo deve reconhecer. Para isso, basta modificar os números nos campos de suas respectivas funções e enviá-los.



# 5. Guia de Configuração

Esse guia tem o propósito de descrever em passos como configurar o dispositivo com base na aplicação web, ele inclui quatro passos essenciais que levam à configuração dos parâmetros de temperatura e umidade desejada, ao intervalo de medição e a rede Wifi que busca conectar.

##### 1° Passo

Na primeira página do frontend, preencha seu login e senha de acordo com seus dados para iniciar e clique no botão entrar para se direcionar ao Dashboard. Em seguida, no dashboard, clique em algum dos quadrados que contém os dados das estufas para abrir o painel que mostra mais detalhes.



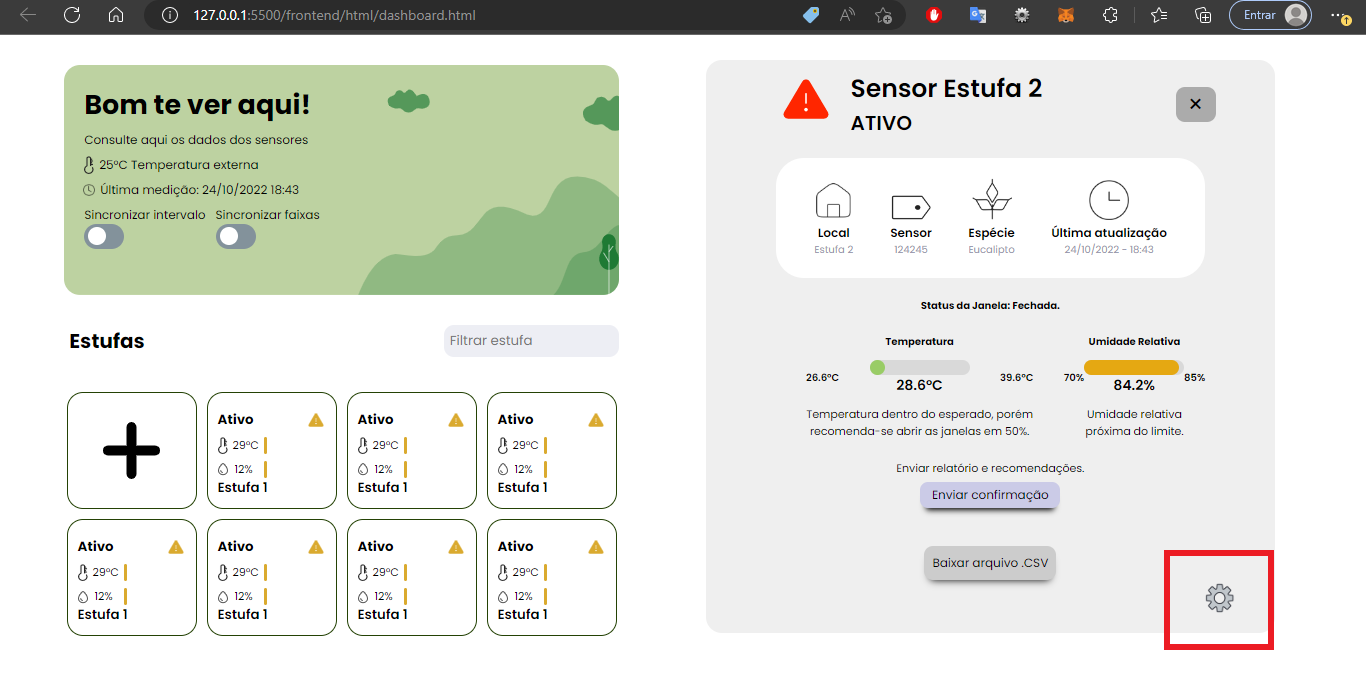
##### 

##### 

##### 

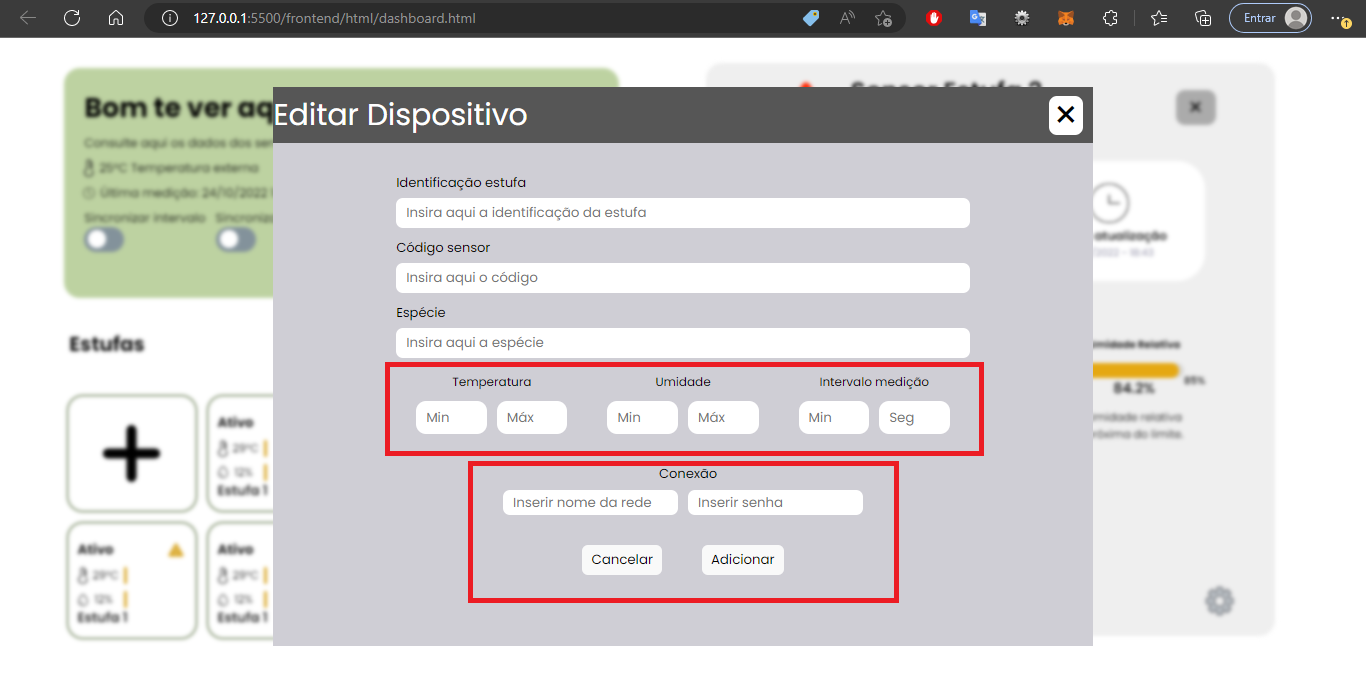
##### 2° Passo

Assim que abrir a tela que mostra os detalhes da estufa, clique na engrenagem posicionada no canto inferior direito da tela para abrir as configurações daquela estufa, onde poderá editar o dispositivo.



##### 3° Passo

Com o painel de configurações aberto, preencha os campos de intervalo de temperatura, umidade, medição de tempo e os dados que permitem conectar à rede Wifi.



##### 4° Passo

Depois de preencher os campos instruídos na etapa anterior, clique no botão “Adicionar” para salvar aquelas configurações de rede e intervalos de medição, umidade e temperatura

##### 

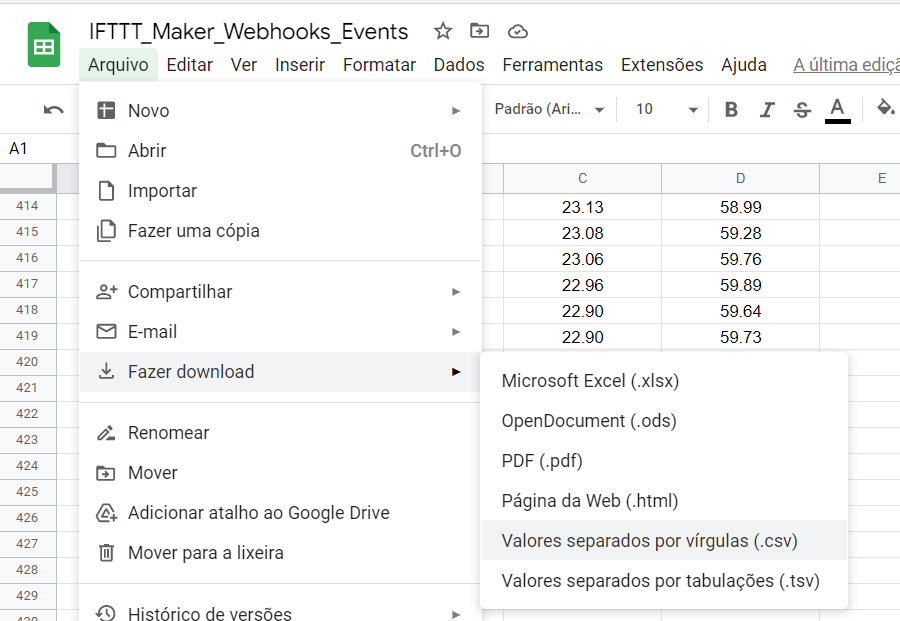
# 5.1 Limitações do sistema e recomendações

Devido a limitações do sistema o IFTTT consegue anotar apenas 2000 linhas na planilha , então recomendamos salvar diariamente o csv e apagar as outras linhas. Assim, sempre há espaço para anotar os dados.

Basta acessar o seguinte link <https://docs.google.com/spreadsheets/d/1jP9yXgPAHUr0RjDQ1sTmqObHoKnP-r0I6hto-3vMw80/edit#gid=0>

E clicar no seguinte botão.

Arquivo>Fazer download > .csv



# 6. Guia de Operação

# 7. Troubleshooting

| **#** | **Problema** | **Possível solução** |
| --- | --- | --- |
| 1 | Queda de energia | Ativar geradores ou conectar o dispositivo em uma fonte energética como bateria com uso de cabo USB |
| 2 | Fonte de energia desconectado | Conectar cabo a fonte de energia após receber aviso do ambiente desconectado |
| 3 | Queda de rede(internet) |  |
| 4 | Falha de envio de dados do sensor para o microcontrolador |  |
| 5 | Sensor não lê a temperatura/umidade | Troca do sensor ou verificação de mal contato |
| 6 | Demora de recebimento dos dados | Consultar se há principalmente oscilações na internet, queda e má conexão com a fonte de energia ou mal contato com a fonte. Também logar novamente e atualizar a página. |
| 7 | LED não pisca |  |
| 8 |  |  |
| 9 |  |  |
| 10 |  |  |

# 8. Créditos