

**Controle do Documento**

**Histórico de revisões**

| **Data** | **Autor** | **Versão** | **Resumo da atividade** |
| --- | --- | --- | --- |
| 15/11/22 | Arthur Fraige | <Versão 1.0> | * Tópico 1.1 - Solução |
| 17/11/22 | Raphael Lisboa | <Versão 1.0.> | * Tópico 3 - Guia de montagem |
| 20/11/22 | Patricia Honorato | <Versão 1.0> | * Tópico 3 - Guia de montagem - Diagramas * 2.3 - Requisitos de Conectividade - Descrição |
| 20/11/22 | Arthur Fraige | <Versão 1.0> | * 2.3. Requisitos de conectividade * 1.1. Solução |
| 20./11/22 | Sophia Tosar | <Versão 1.0> | * 2.1 - Componentes do Hardware * 2.2- Componentes Externos |
| 20/11/22 | Priscila Falcão | <Versão 1.0> | * 1.2 - Arquitetura da Solução |

**Índice**

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

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

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

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

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

[2.2. Componentes externos](#_heading=h.17dp8vu) 6

[2.3. Requisitos de conectividade](#_heading=h.3rdcrjn) 7

[**3. Guia de Montagem**](#_heading=h.26in1rg) **8**

[**4. Guia de Instalação**](#_heading=h.9iubjgqpvih5) **16**

[**(sprint 4)**](#_heading=h.1ksv4uv) **16**

[**5. Guia de Configuração**](#_heading=h.44sinio) **17**

[**(sprint 4)**](#_heading=h.2jxsxqh) **17**

[**6. Guia de Operação**](#_heading=h.z337ya) **18**

[**(sprint 5)**](#_heading=h.3j2qqm3) **18**

[**7. Troubleshooting**](#_heading=h.1y810tw) **19**

[**(sprint 5)**](#_heading=h.4i7ojhp) **19**

[**8. Créditos**](#_heading=h.2xcytpi) **20**

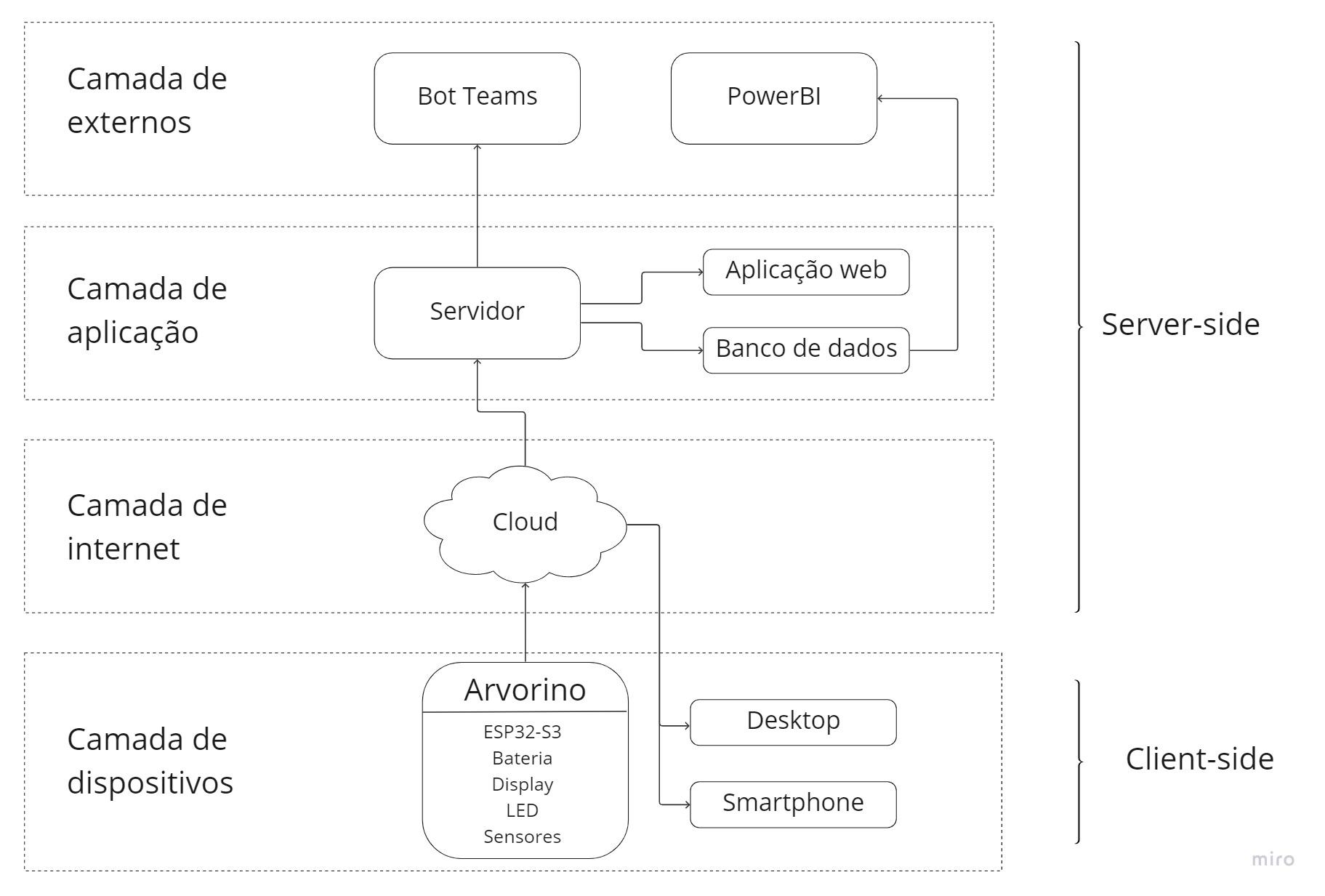
[**(sprint 5)**](#_heading=h.1ci93xb) **20**

# 1. Introdução

## 1.1. Solução

O Dispositivo capta dados como temperatura e umidade relativa do ar, através de sensores, e envia esses dados para um banco, onde lá serão utilizados para interpretação e assim servirem de inputs para os motores responsáveis pela automação de controle das janelas. O protótipo deverá ser introduzido na estufa da Gerdau, de forma que ele irá automatizar a coleta de dados de temperatura e umidade relativa do ar por minuto. A partir dessa coleta o protótipo envia um comando para as janelas da estufa se abrirem ou fecharem, de forma que esse processo tenha mais granularidade e confiabilidade, diminuindo o risco de mortalidade das mudas e aumentando o índice de produção consequentemente.

## 1.2. Arquitetura da Solução



**Figura 1 - Diagrama da arquitetura da solução**

# 2. Componentes e Recursos

## 2.1. Componentes de hardware

| Componente | Descrição |
| --- | --- |
|  | **nome:** ESP32-S3-WROOM-1  **funcionalidade:** Coletar e processar as informações provenientes dos sensores a cada 1 minuto. |
|  | **nome:** sensor ATH10  **funcionalidade:** Enviar informações de temperatura e umidade . |
|  | **nome:** LED RGB  **funcionalidade:**  Componente que pisca para alertar o usuário em determinadas situações, tal como o sucesso do envio para o banco de dados. |
|  | **nome:** Display LCD  **funcionalidade:** Exibe as informações de temperatura e umidade do ar na estufa. |
|  | **nome:** jumpers macho-macho  **funcionalidade:**  faz a conexão entre componentes eletrônicos |
|  | **nome:** resistor 1k ohms  **funcionalidade:** são componentes eletrônicos cuja principal função é limitar o fluxo de cargas elétricas. |
|  | **nome:** resistor 10k ohms  **funcionalidade:** são componentes eletrônicos cuja principal função é limitar o fluxo de cargas elétricas. |
|  | **nome:** Protoboard da marca Minipa  **funcionalidade:** ela é uma placa com diversos furos e conexões condutoras verticais e horizontais para a montagem de circuitos elétricos |
|  | **nome:** Power Bank  **funcionalidade:** bateria |
|  | **nome**: conversor I2C  **funcionalidade:** O conversor I2C tem a função de manipular as conexões de um display . |

**Tabela 1 - Componentes do Hardware**

## 2.2. Componentes externos

Liste aqui componentes como computadores, tablets e/ou celulares que deverão fazer parte da sua solução, bem como eventuais serviços em nuvem, softwares de edição de código ou outras aplicações utilizadas.

| Componente | Descrição |
| --- | --- |
|  | **Nome:**  Dispositivo com acesso à internet (computador/tablet/ celular)  **Funcionalidade:** O dispositivo com acesso à internet, que pode ser um computador, um tablet ou um celular, será responsável por carregar a plataforma WEB do produto. |
|  | **Nome:** Arduino Ide  **Funcionalidade:**Aplicação usada para compilar o código e enviar as instruções para o microcontrolador. |
|  | **Nome:** VSCode  **Funcionalidade:** O Visual Studio Code é um editor de código. |
|  | **Nome:** Power BI  **Funcionalidade:** Dashboard para análise de dados recebidos do banco |
|  | **Nome:** Servidor da aplicação  **Funcionalidade:** Aplicação utilizada para armazenar um sistema externo e suas funcionalidades |
|  | **Nome:** Microsoft Teams  **Funcionalidade:** Bot aderido à plataforma Teams para alerta de parâmetros fora dos níveis ideais |
|  | **Nome:** HTML  **Funcionalidade:**Linguagem utilizada para construção da página web. |
|  | **Nome:** CSS  **Funcionalidade:** Linguagem utilizada para construção do estilo (fontes, cores, espaçamento e etc) da página web em html. |
|  | **Nome:** JavaScript  **Funcionalidade:** Linguagem utilizada para dinamizar a página web html, com a implementação de informações incrementáveis do banco de dados |
|  | **Nome:** NodeJS  **Funcionalidade:** Ferramenta utilizada para possibilitar a comunicação entre o local e o servidor |
|  | **Nome:** C++  **Funcionalidade:** Linguagem de baixo nível utilizada para definir instruções à placa ESP32 |
|  | **Nome:** DB Browser  **Funcionalidade:** O database browser é um aplicativo que permite visualizar, editar e executar scripts SQL em banco de dados Oracle, ODBC e SQL Server. |
|  | **Nome:** SQLite  **Funcionalidade:** Linguagem específica projetada para gerenciar e manipular dados em banco |

**Tabela 2 - componentes externos.**

## 

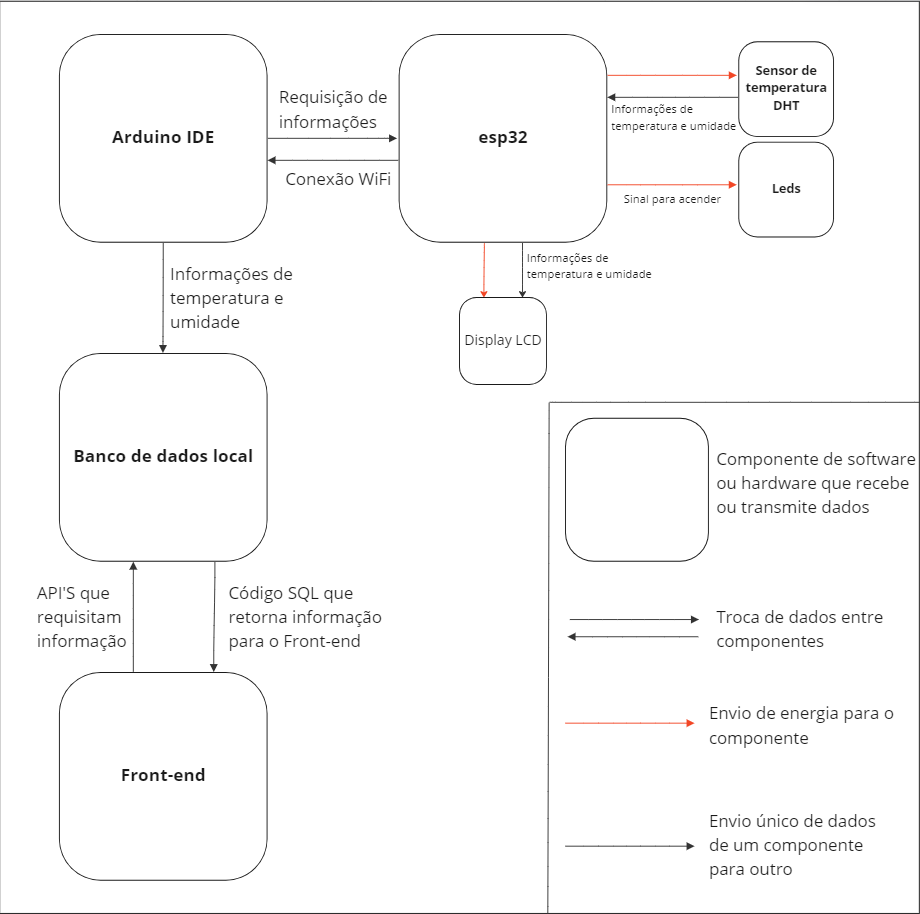
## 

## 

## 

## 2.3. Requisitos de conectividade

O diagrama de conectividade abaixo fornece uma visualização das ligações realizadas por esse projeto IOT. O sensor de umidade e temperatura AHT10 é ligado ao ESP32-S3 por meio de pinos analógicos. O led RGB é ligado à placa por meio de pinos convencionais. A placa se conecta ao wifi e servidor web por meio de um script executado no Arduino IDE. Para interação com o banco de dados, utiliza um método de interação indireto HTTP via requisição para o servidor web, que executa o script javascript e atende a requisição de dados enviada pela placa. Em seguida, esse script extrai e processa os dados e os envia para o banco de dados no SQLite. Por último, após o script processar os dados, esses dados são enviados novamente para o ESP32-S3 e são também exibidos no servidor web.



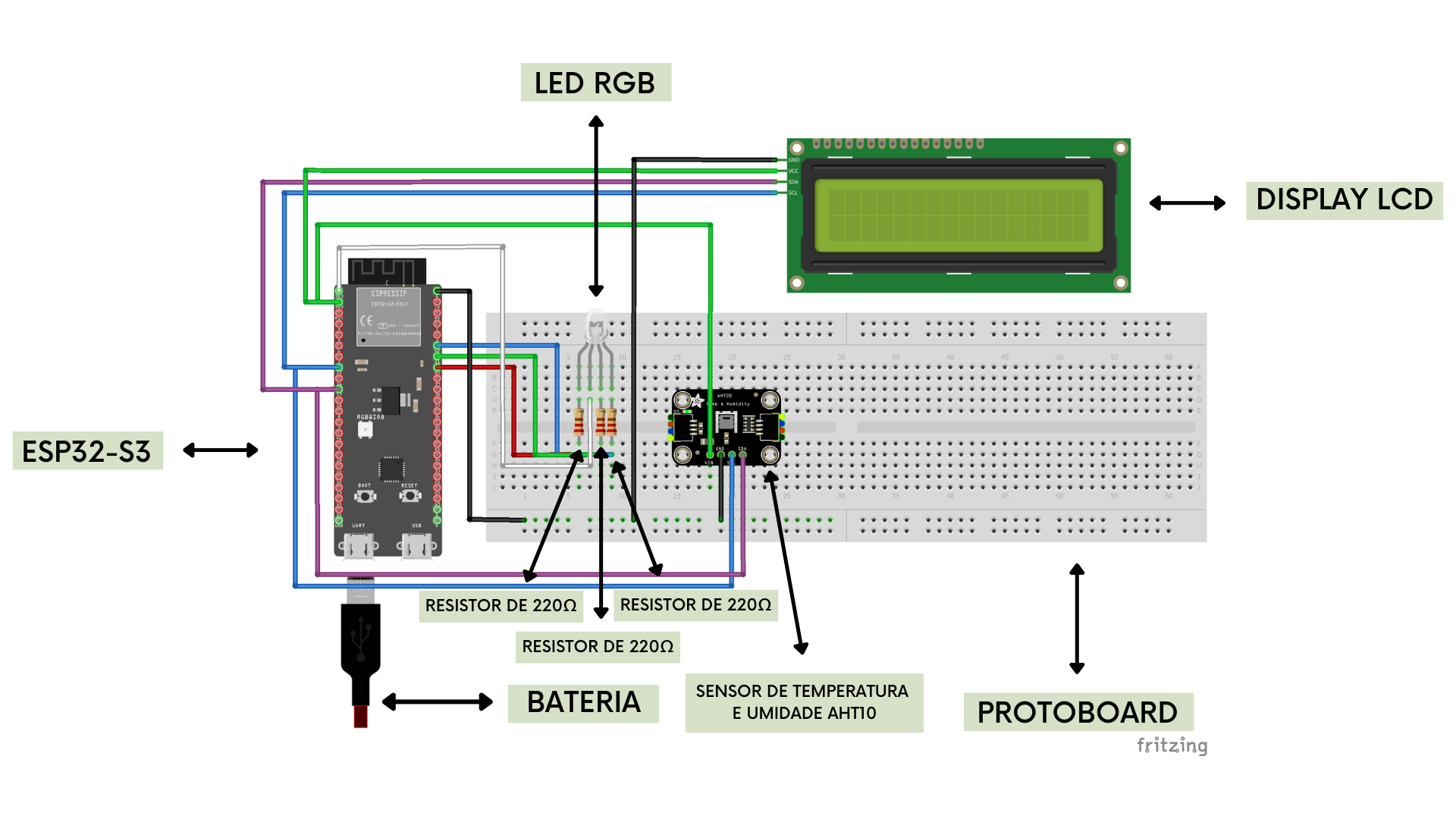
**Diagrama 2 - Diagrama de conectividade do projeto IOT.**

# 3. Guia de Montagem

Neste tópico, será apresentado o passo a passo para realização da montagem do mecanismo IOT. Primeiramente, serão descritos os componentes e as entradas que estes deverão ser conectados na placa do ESP32. Os componentes utilizados para montagem do mecanismo IOT estão descritos na tabela 1 deste manual de instrução. Por último, será apresentado um diagrama do circuito elétrico dos respectivos componentes utilizados.

**Componentes do Hardware**

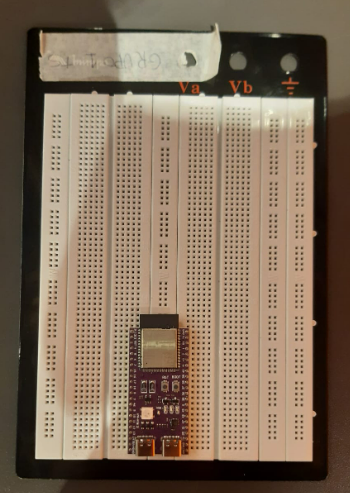
O diagrama a seguir apresenta todos oscomponentes utilizados na montagem do mecanismo e suas disposições na protoboard. A placa ESP32-S3 deve ser conectada ao protoboard, na figura abaixo ela se encontra na lateral apenas para visualização do cabeamento dos pinos com os componentes no circuito.

****

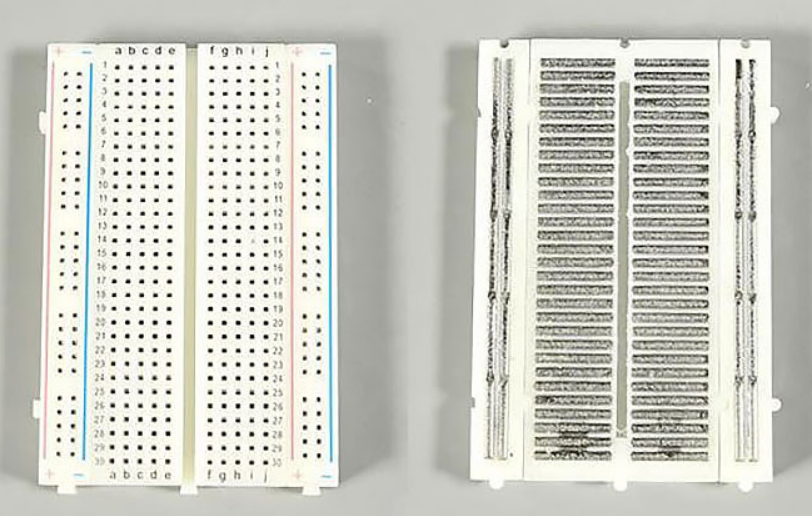
**Diagrama 3 - Diagrama de montagem de componentes.**

**ESP-32**

O microcontrolador precisa ser posicionado no centro das protoboards, assim como pode ser visto na figura 1. É importante que o ESP32-S3 seja posicionado no centro, pois somente assim irá sobrar linhas para a conexão de demais componentes/fios nas portas indicadas pelo microcontrolador, isso se faz necessário tendo em vista de que no interior das protoboard os pontos que se encontram nas mesmas linhas estão submetidos a um mesmo potencial elétrico (Figura 1), assim caso um componente esteja conectado nessa mesma linha seria a mesma coisa que estivesse conectado diretamente à porta. O mesmo ocorre com as colunas laterais, que são divididas em positivo (+) e negativo (-), no caso do negativo, é muito utilizada para a conexão de diversos componentes ao pino do terra (GND) que tem a função de fechar o circuito



**Figura 1 - Indicação de onde deve ser posicionado o microcontrolador ESP32-S3.**



**Figura 2 - Na esquerda uma protoboard vista por cima e na direita uma protoboard vista por dentro para a ilustração das linhas e colunas de mesmo potencial elétrico**

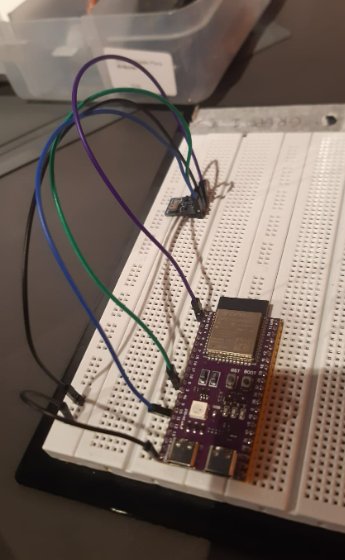
**Sensor AHT-10**

Para o encaixe do sensor AHT-10, responsável pela leitura e coleta de temperatura e umidade relativa do ar, é necessário tomar conhecimento sobre o protocolo I2C, o protocolo em questão é uma forma de comunicação serial entre componentes. De acordo com o protocolo, cada componente possui uma de duas funções existentes: mestre (master) ou escravo (slave), na maioria das vezes microcontroladores como Arduino, ESP-32, entre outros , é dada a função de mestre, a qual ele apresenta a funcionalidade de dar ordens aos demais componentes, esses que recebem as instruções são dados como escravos. Para que a comunicação seja feita de forma sincronizada sem que haja perda de dados, se faz necessária a utilização de 2 portas: SDA (Serial Data) que realiza a transferência de dados e SCL (Serial Clock) que realiza a sincronização na transferência. Sabendo do protocolo, vamos dar uma olhada atrás do sensor para sabermos onde devemos conectar cada coisa (Figura 2).

Agora sim podemos ir para a montagem, conecte o sensor em algum lugar da protoboard, desde que seja cada pino em uma linha diferente, isso se dá visto que todas as linhas possuem o mesmo potencial, e é necessária a diferença de potencial (DDP) para que seja possível a transferência de elétrons, portanto, uma corrente elétrica. Tendo feito isso: conecte um jumper ligando da porta 3V3 do ESP-32 até o pino VIN, indicado pelo sensor, em seguida vamos definir o pino 10 para representar o SDA, conectando um jumper da porta 10 até o pino indicado por SDA no sensor e o pino 8 para representar o SCL, também conectando um jumper que vá da porta 8 até o pino indicado por SCL, é importante manter essas 2 portas (SDA e SCL) nesses exatos pinos, pois são portas especiais para essa funcionalidade que elas exercem. Por fim fechamos o circuito conectando o pino indicado por GND na porta GND do ESP-32, mas como iremos fazer conexões nessa mesma porta para quase todos os componentes, conecte um jumper que vá do sensor até uma das colunas verticais encontradas na lateral da protoboard, no fim dessa mesma coluna conecte um jumper até a porta GND do ESP-32, isso irá facilitar muito a montagem a medida que formos inserindo cada vez mais componentes na protoboard, além de deixar todo o sistema mais organizado. Sua montagem no final deve ser semelhante com a montagem indicada na figura 4.



**Figura 3- Indica o verso do sensor AHT-10.**



**Figura 4 - Indicando como deve ficar a montagem do circuito do**

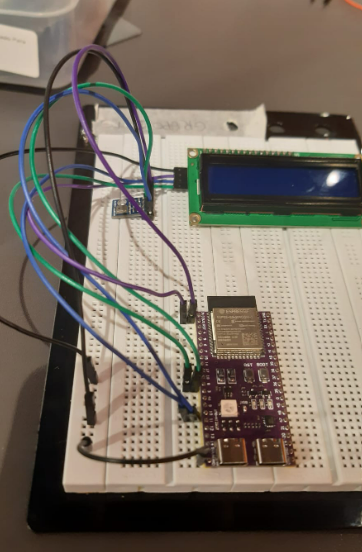
**sensor AHT-10 através das instruções.**

**Display LCD**

O display LCD, assim como o sensor de umidade e temperatura AHT-10, também segue o protocolo I2C, portanto é de extrema importância ligar os pinos SDA e SCL do display nas mesmas portas que foram ligadas o sensor (10 e 8), pois somente assim o microcontrolador (master) será capaz de sincronizar os dados entre os componentes (slaves).

Para a montagem do display, vamos utilizar um outro tipo de jumper, visto que não vamos prendê-lo diretamente à protoboard, estamos acostumados a usar o jumper do modelo “macho-macho”, um jumper que contém um pino de encaixe nas 2 extremidades, agora para a montagem do display, vamos utilizar um do modelo “macho-fêmea” que em uma extremidade possui um pino de encaixe e na outra possui um buraco para que seja encaixado um outro pino ali.

Assim como já foi dito, conecte ao lado de cada um dos jumpers das portas 10 e 8, um outro jumper ligando os pinos SDA e SCL indicados no display, respectivamente. Conecte também o pino VCC do display na porta 3V3 do ESP-32 e o pino GND na mesma coluna lateral a qual foi conectado o fio terra do sensor. Seguindo todos os passos sua montagem será igual à indicada na figura 4.



**Figura 5 - Indicação de como deve ficar a montagem do sensor ao final da montagem do display LCD.**

*Obs.: O display da imagem se encontra apenas apoiado na protoboard, ele não está diretamente conectado como os demais componentes*

**Led RGB**

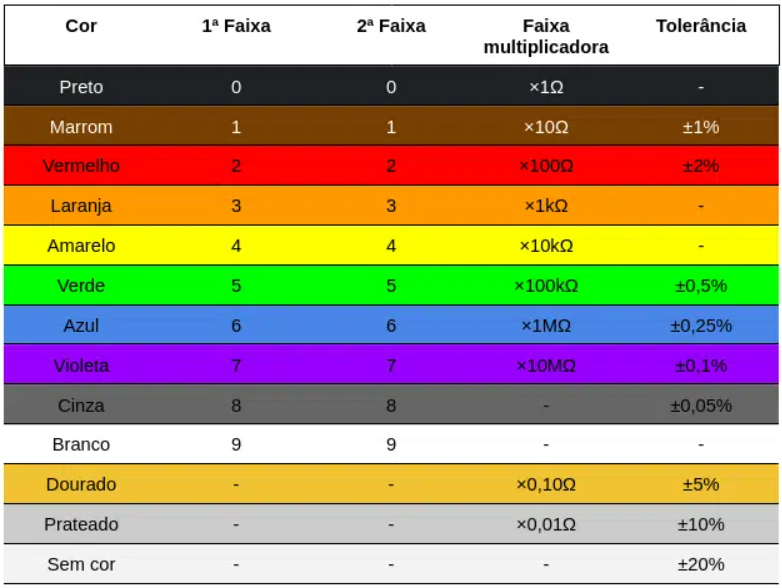
O led RGB tem esse nome devido a capacidade que ele possui de brilhar nas 3 principais cores: vermelho (R), verde (G) e azul (B) e com isso combinar diferentes intensidades e cores para que seja formada qualquer cor do espectro de luz visível. Para isso, o led possui 4 pernas ao invés de 2 como qualquer outro led convencional com a capacidade de acender em apenas 1 cor, cada perna do led é responsável por uma das 3 cores principais e a maior é responsável pelo aterramento (GND).

Antes de montar o circuito é necessário descobrir qual o tipo de led que você tem, existem atualmente no mercado 2 modelos: os leds de ânodo comum e os de cátodo comum. Os leds de ânodo comum recebem a energia elétrica através da perna maior e seu aterramento é feita através das portas numéricas, com isso ocorre uma inversão na lógica de funcionamento, fazendo com que na hora de programar LOW significa “acender” e HIGH significa “apagar”, essa inversão ocorre por conta de como o circuito elétrico está funcionando: para que o led acenda é necessária não só a chegada de energia, mas também a saída, como a maior perna se encontra recebendo energia a todo momento, na hora em que eu escrevo no código “LOW”, eu estou fechando a porta e consequentemente o circuito também, sendo feito assim um aterramento e com isso o funcionamento do circuito se completa, o mesmo ocorre quando é definido o código de intensidade de cada cor, devida a inversão, 0 remete a intensidade mais forte e 255 a intensidade mais fraca; já os leds de cátodo comum funcionam como a maioria, sem nenhuma inversão de lógica. Para saber o led que você possui, é necessário utilizar um multímetro com a função de teste de continuidade. Por meio deste, é fornecida uma pequena tensão para verificar-se a existência de ligação nos pontos de provas, e ao ser ligado, este recebe uma pequena tensão em determinadas configurações, verificando-se em qual possível cenário será acendido.

É muito importante também lembrar que leds são componentes de resistência muito baixa, tanto que é exatamente por esse motivo que eles são capazes de realizar a emissão de luz, por isso é necessária a implementação de um resistor para evitar que o componente se queime. No caso, iremos utilizar um resistor de 220Ω (para saber o código de cores do resistor, verifique a figura 6) para cada uma das pernas que representa o RGB, seguindo as instruções dadas pela figura 7.

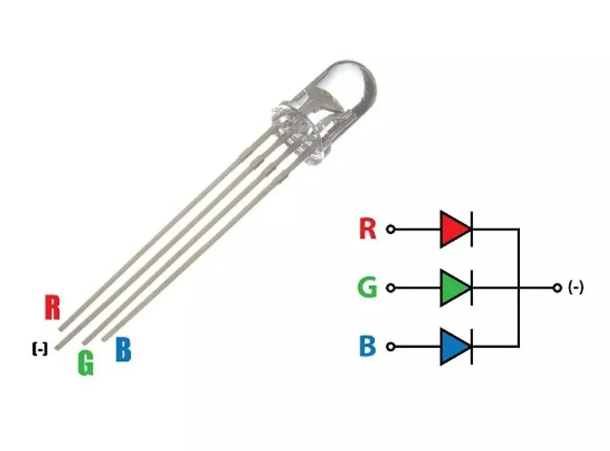
Sabendo de todas as especificações do led, podemos ir à montagem: conecte nas portas 37 um fio para indicar a cor vermelha, na porta 38 um fio para indicar a cor verde e na porta 39 um fio para indicar a cor azul,, conecte até algum lugar ao resistor e a outra ponta do resistor nas pernas que indicam as cores do led (lembre-se sempre que o resistor deve começar em uma linha da protoboard e terminar em outra para que exista diferença de potencial, o mesmo vale para as 4 pernas do led) e por fim conecte a maior perna do led na porta 3V3, caso você tenha um led do modelo ânodo comum ou a maior perna do led na coluna da lateral da protoboard em que todos os componentes estão indo para o GND, caso você tenha um led no modelo cátodo comum.

Se todas as instruções foram seguidas da forma correta, o circuito deve ter ficado de acordo com a figura 8.



**Figura 6 - Tabela de cores para a identificação da resistência**

**de cada resistor.**



**Figura 7 - Led RGB com indicação de qual cor**

**cada perna do led representa.**



**Figura 8 - Montagem do circuito do Led RGB.**

Ao fim da montagem de todos os componentes, na protoboard deve ficar assim como a figura 9. É importante manter uma organização de onde cada componente será disposto para evitar o entrelaçamento de fios, visando que seja possível realizar um reparo rápido e sem complicações, caso venha a ser necessário.

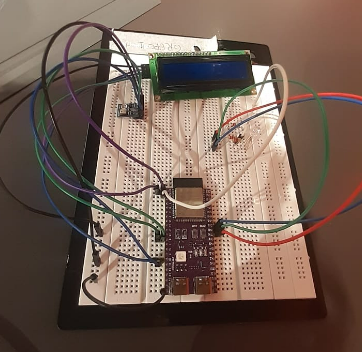
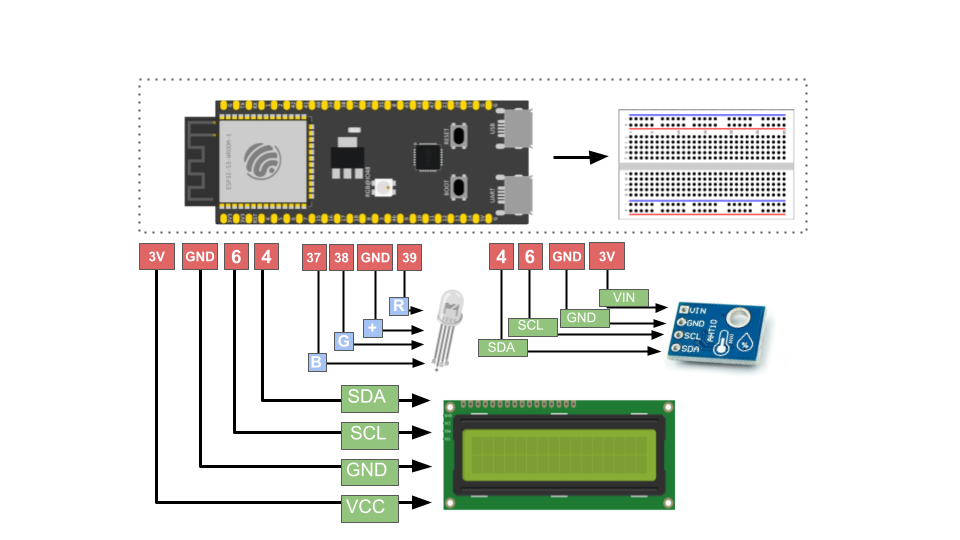


Figura 9 - Montagem final do mecanismo.

**Diagrama da montagem do circuito elétrico do mecanismo**

Para auxiliar na montagem do circuito elétrico, abaixo estão descritas todas as conexões dos componentes com a placa. Os pinos utilizados estão descritos de acordo com as saídas presentes para cada um dos componentes. Para melhor compreensão dos pinos utilizados, consulte a descrição dos componentes realizadas neste tópico.

****

**Figura 11 -Diagrama da montagem do circuito elétrico do mecanismo.**

# 4. Guia de Instalação

### (sprint 4)

Descreva passo-a-passo como instalar os dispositivos IoT no espaço físico adequado, conectando-os à rede, de acordo com o que foi levantado com seu parceiro de negócios.

Não deixe de especificar propriedades, limites e alcances dos dispositivos em relação ao espaço destinado.

Especifique também como instalar softwares nos dispositivos.

Utilize fotografias, prints de tela e/ou desenhos técnicos para ilustrar o processo de instalação.

# 5. Guia de Configuração

### (sprint 4)

Descreva passo-a-passo como configurar os dispositivos IoT utilizando os equipamentos devidos (ex. smartphone/computador acessando o servidor embarcado ou a página na nuvem).

Utilize fotografias, prints de tela e/ou desenhos técnicos para ilustrar o processo de configuração.

# 6. Guia de Operação

### (sprint 5)

Descreva os fluxos de operação entre interface e dispositivos IoT. Indique o funcionamento das telas, como fazer leituras dos dados dos sensores, como disparar ações através dos atuadores, como reconhecer estados do sistema.

Indique também informações relacionadas à imprecisão das eventuais localizações, e como o usuário deve contornar tais situações.

Utilize fotografias, prints de tela e/ou desenhos técnicos para ilustrar os processos de operação.

# 7. Troubleshooting

### (sprint 5)

Liste as situações de falha mais comuns da sua solução (tais como falta de conectividade, falta de bateria, componente inoperante etc.) e indique ações para solução desses problemas.

| **#** | **Problema** | **Possível solução** |
| --- | --- | --- |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |
| 5 |  |  |

# 8. Créditos

### (sprint 5)

Seção livre para você atribuir créditos à sua equipe e respectivas responsabilidades