

**Controle do Documento**

**Histórico de revisões**

| **Data** | **Autor** | **Versão** | **Resumo da atividade** |
| --- | --- | --- | --- |
| 18/11/2022 | Sarah Ribeiro | 1.0 | Criação do documento |
| 18/11/2022 | Leandro Custódio | 1.1 | [3. Guia de Montagem](#_v51amp5m28ia) |
| 03/12/2022 | Felipe Sampaio | 2.0 | [4. Guia de Instalação](#_ns4i2ee2va9l) |
| 03/12/2022 | Felipe Sampaio | 2.1 | [5. Guia de Configuração](#_xbfutowuk20a) |

**Índice**

[**1. Introdução**](#_3p4k6d3g6219) **3**

[1.1. Solução](#_rlngioqecbyk) 3

[1.2. Arquitetura da Solução](#_61uhcal2j77f) 3

[**2. Componentes e Recursos**](#_uvfjwzlomuzy) **4**

[2.1. Componentes de hardware](#_jafy6yk85z5g) 4

[2.2. Componentes externos](#_dq0hfd7wcjor) 4

[2.3. Requisitos de conectividade](#_yxhdlhc9u11x) 4

[**3. Guia de Montagem**](#_v51amp5m28ia) **5**

[**4. Guia de Instalação**](#_ns4i2ee2va9l) **6**

[**5. Guia de Configuração**](#_mjz06zt366c7) **7**

[**6. Guia de Operação**](#_vcwsg1gripyk) **8**

[**7. Troubleshooting**](#_omvzmwr1fxwv) **9**

[**8. Créditos**](#_t6okuol326z9) **10**

# 1. Introdução

## 1.1. Solução (sprint 3)

A solução proposta é um protótipo de Internet das Coisas(IOT) que tem como objetivo identificar ativos em um ambiente indoor. A solução é composta por dois itens principais: tags e beacons. Os beacons serão posicionados em posições chaves dentro do ambiente e as tags serão utilizadas nos objetos que precisam ser localizados. Com base na posição da tag em relação a cada beacon será possível identificar a localização do objeto no ambiente, que poderá ser visualizada através de um aplicativo web.

## 1.2. Arquitetura da Solução (sprint 4)



**Dispositivos utilizados:**

**Beacon (ESP32-s3):** utilizado como ponto de referência para que a tag consiga ser localizada. Irão enviar para a tag a distância em que eles se encontram dela.

**Beacon com Tag (ESP32-s3):** Coleta os dados relativos a distância dele em relação aos beacon e envia para API, esses valores são utilizados para o cálculo da triangulação.

**Tag (ESP32-s3):** colocado no item a ser localizado.Também é um microcontrolador que reunirá todas as informações de distância recebidas dos beacons e as enviará via roteador para o nosso servidor. Nas tags também serão acoplados buzzers e LED’s para que seja mais simples a localização do item pelo usuário. Além disso, um acelerômetro e um sensor de temperatura também estarão unidos à tag. O primeiro para que possamos detectar quando o ativo estiver em movimento e o segundo para que possamos impedir um superaquecimento da placa.

**Banco de Dados:** armazenará as informações que virão da Tag e poderão ser acessadas através da aplicação web.

**Interface para o usuário:** Será uma aplicação web, desenvolvida com o propósito de ser a interface de controle e uso para localizar as tags. Por meio da API e do protocolo http iremos acessar as informações do banco de dados e externalizá-las para o usuário.

# 2. Componentes e Recursos

### (sprint 3)

## 2.1. Componentes de hardware

| **Componente** | **Fornecedor** | **Detalhes Técnicos** | **Link** |
| --- | --- | --- | --- |
| Esp32 -> s3 | Loja online - [Savarati](https://www.saravati.com.br/) | wifi / bluetooth , Dual USB - C | [link](https://www.saravati.com.br/produto/placa-esp32-s3-wroom-1-n16r8-wifi-bluetooth-dual-usb-c.html) |
| Resistores | Loja online - [Savarati](https://www.saravati.com.br/) | 220 Ohms | [link](https://www.saravati.com.br/resistor) |
| Jumpers | Loja online - [Savarati](https://www.saravati.com.br/) | Fêmea/Fêmea e Macho/Fêmea | [link](https://www.saravati.com.br/buscar?q=jumpers) |

## 2.2. Componentes externos

| **Componente Externo** | **Função** |
| --- | --- |
| **Dispositivo com acesso web:**  **Computador/Tablet/Disp. Mobile** | Acessar a aplicação Web que disponibilizará a localização da Tag. |
| **mongoDB** | Banco de Dados utilizado |
| **Arduino IDE** | Aplicação responsável para enviar o código para o ESP32 |
| **Codesandbox** | Servidor que hospeda a API |

## 2.3. Requisitos de conectividade

**Ambiente de programação:** Arduino IDE

**Rede:** Wifi

**Protocolo de rede:** http

**Banco de dados:** MongoDB

**Linguagem:** C++

Exemplo de uso de imagem em coluna única:

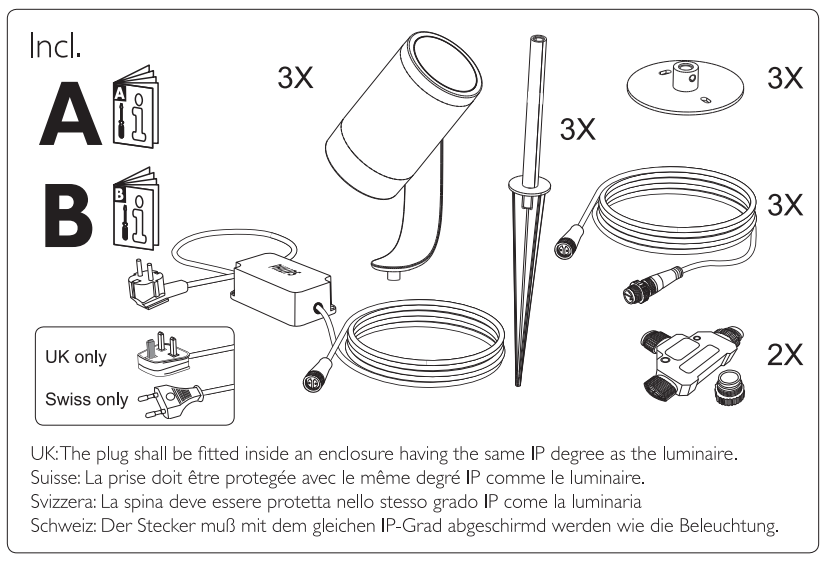


Figura 1: use sempre uma legenda e mencione o número

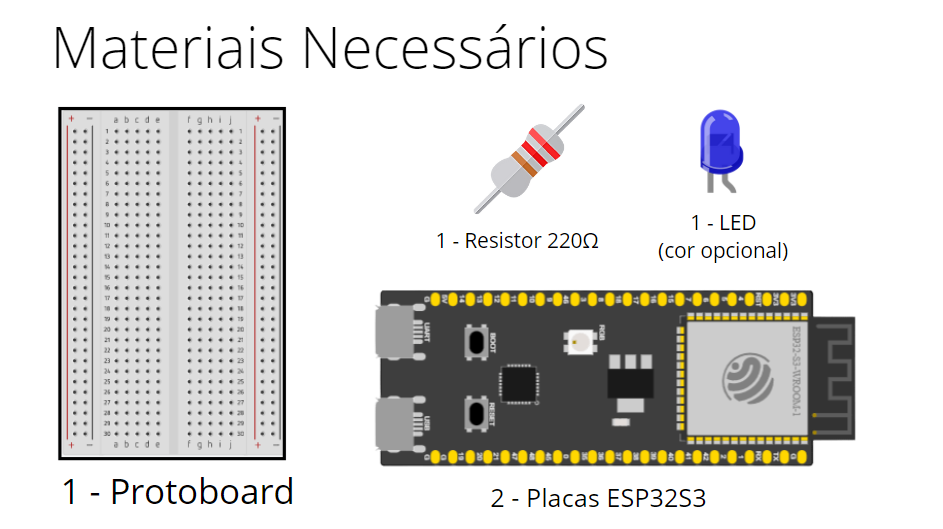
da figura no corpo do texto. Cuidado para que detalhes

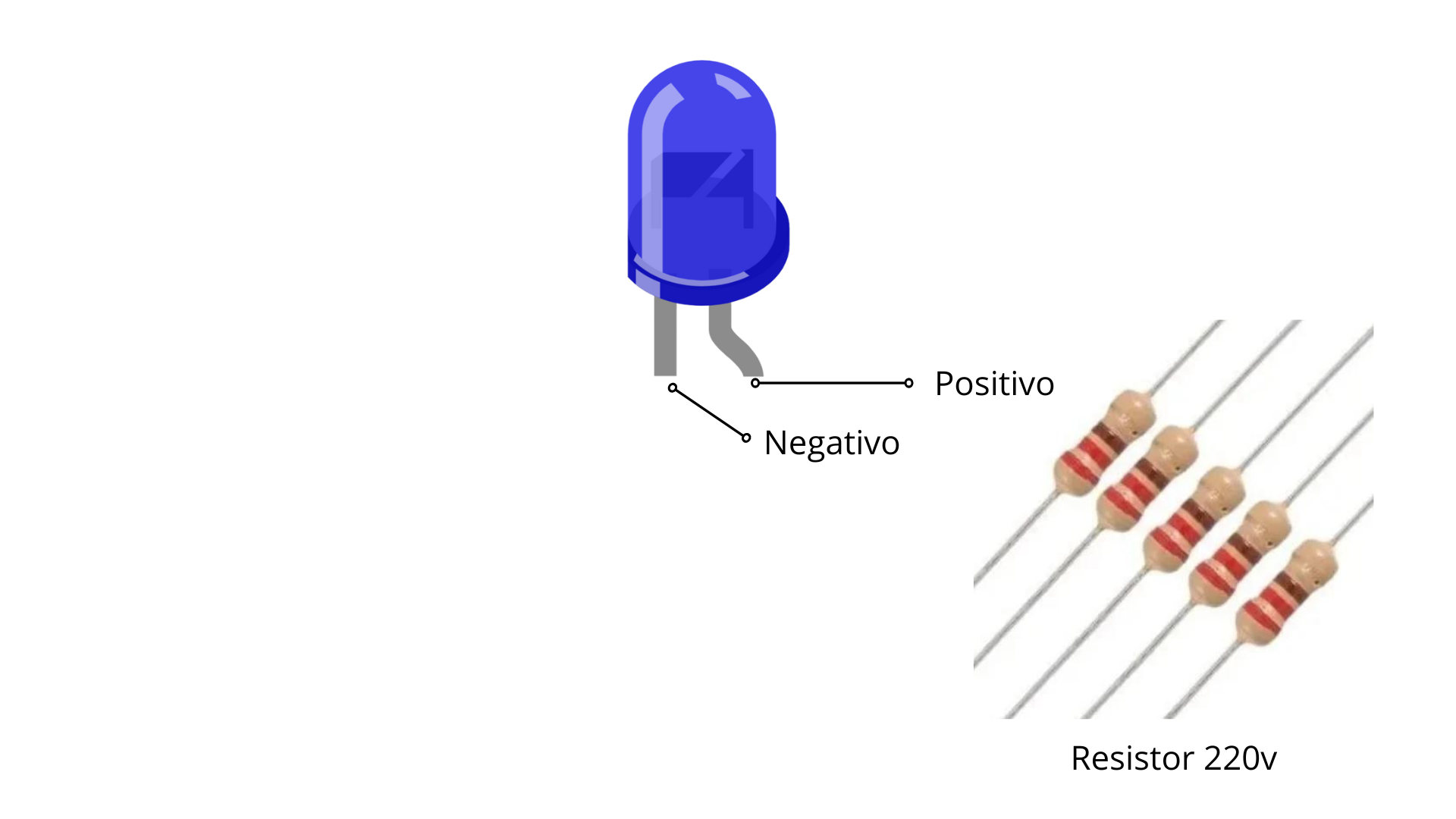
da imagem não fiquem ilegíveis, como na imagem.

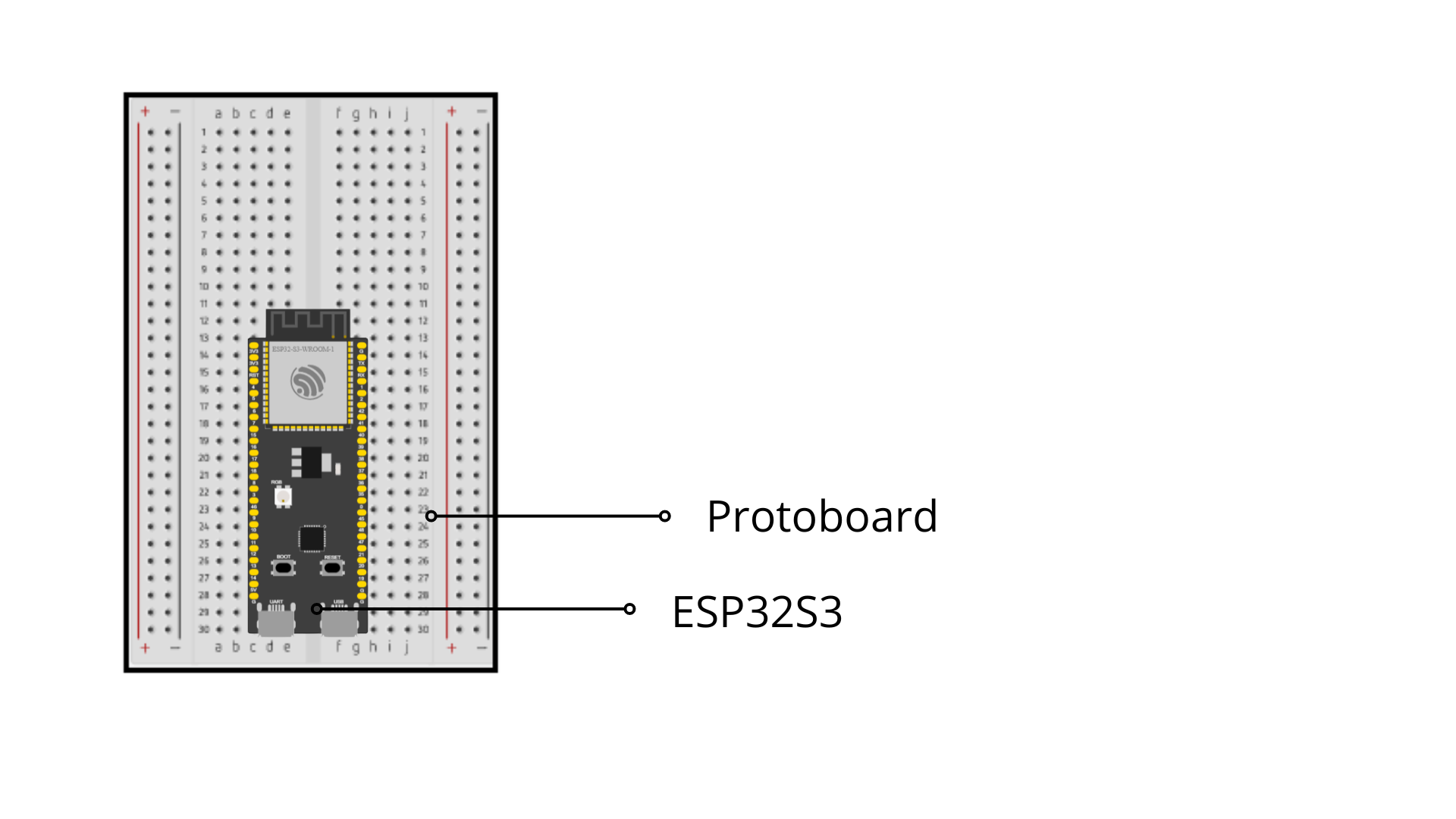
# 3. Guia de Montagem

### (sprint 3)

Para montagem de nossa solução é necessária muita atenção nos itens a serem utilizados e na forma como são conectados.

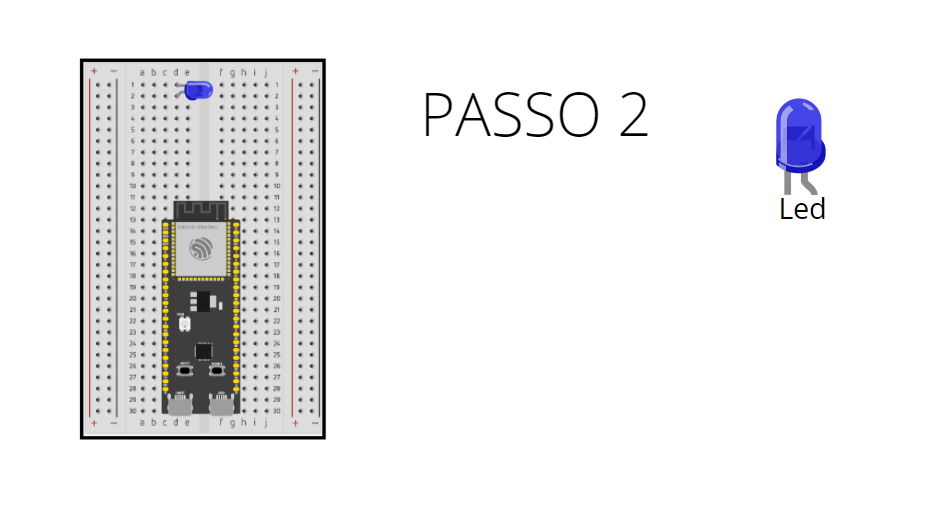
**Materiais necessários:**





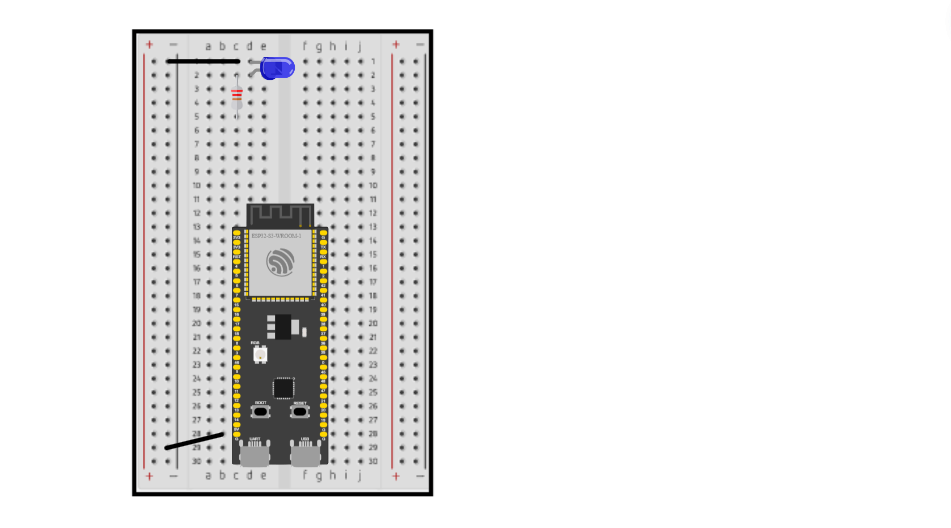
**Passo 1:** Conecte sua placa ESP32S3 na protoboard.

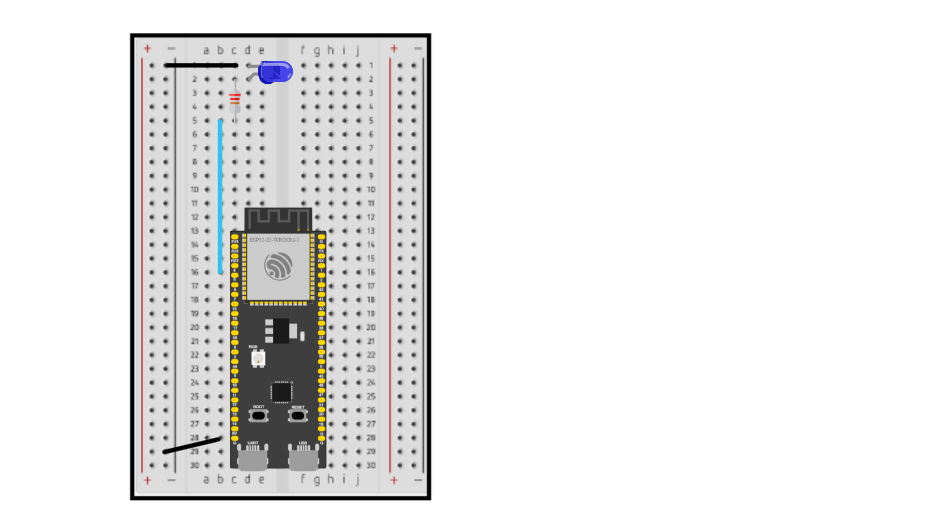
**Passo 2:** Conecte um Led na protoboard.

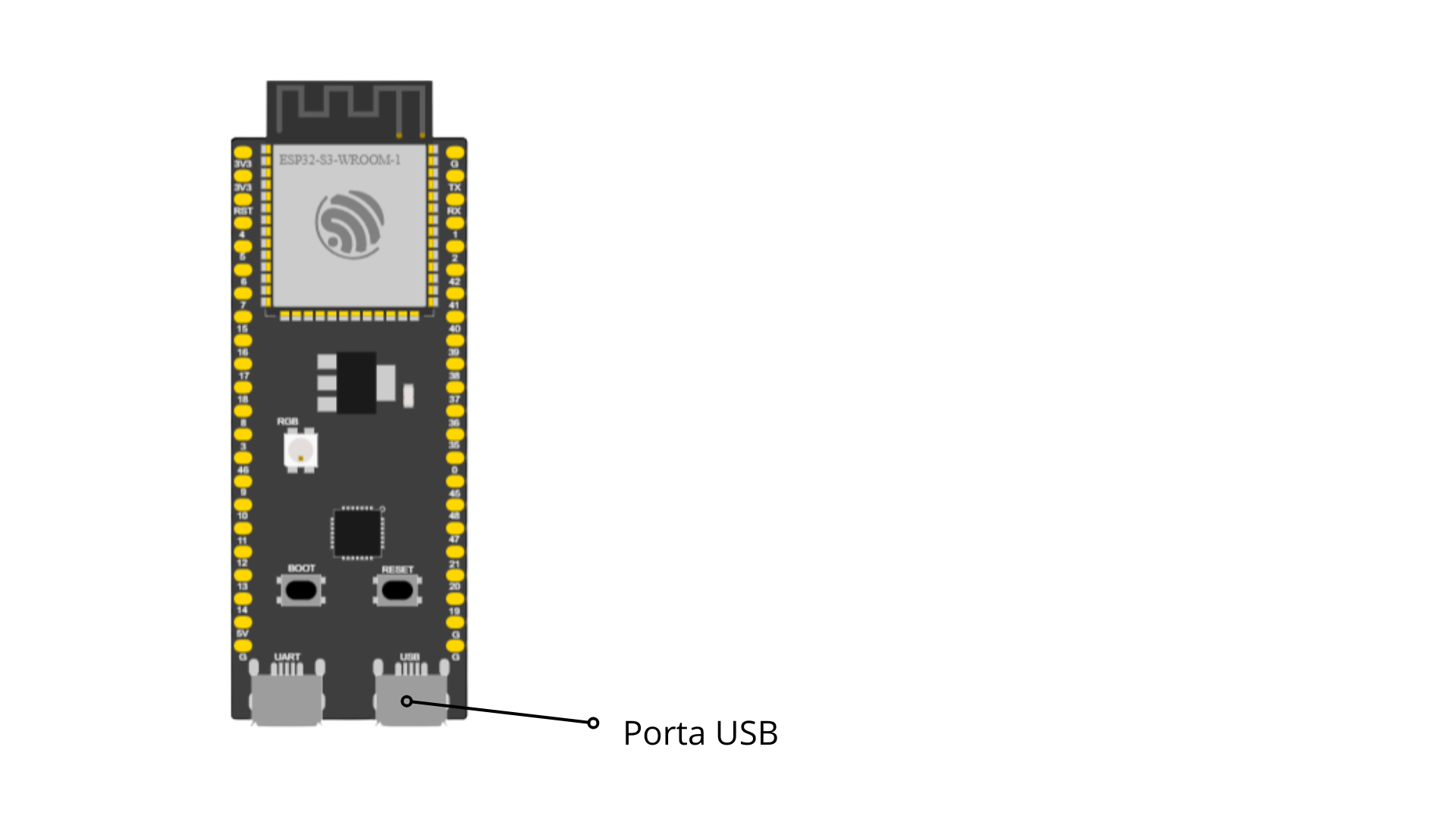


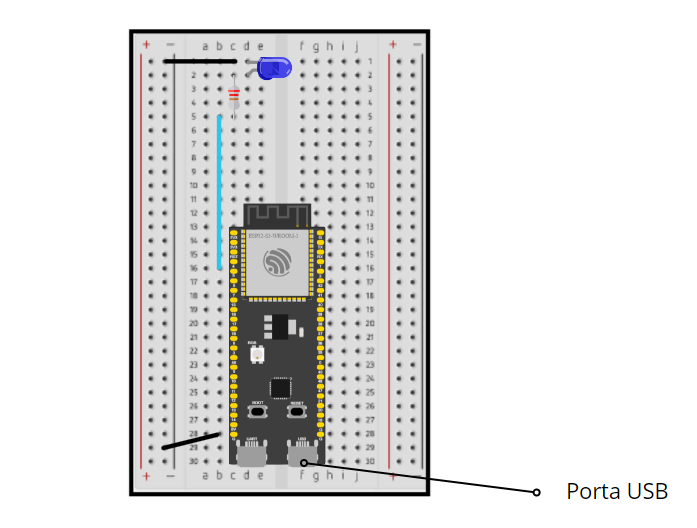
**Passo 3:** Conecte qualquer lado do resistor de 220Ω na perna positiva do led.

**Passo 4:** Conecte os fios terra (GND) assim como demonstrado na imagem ao lado.

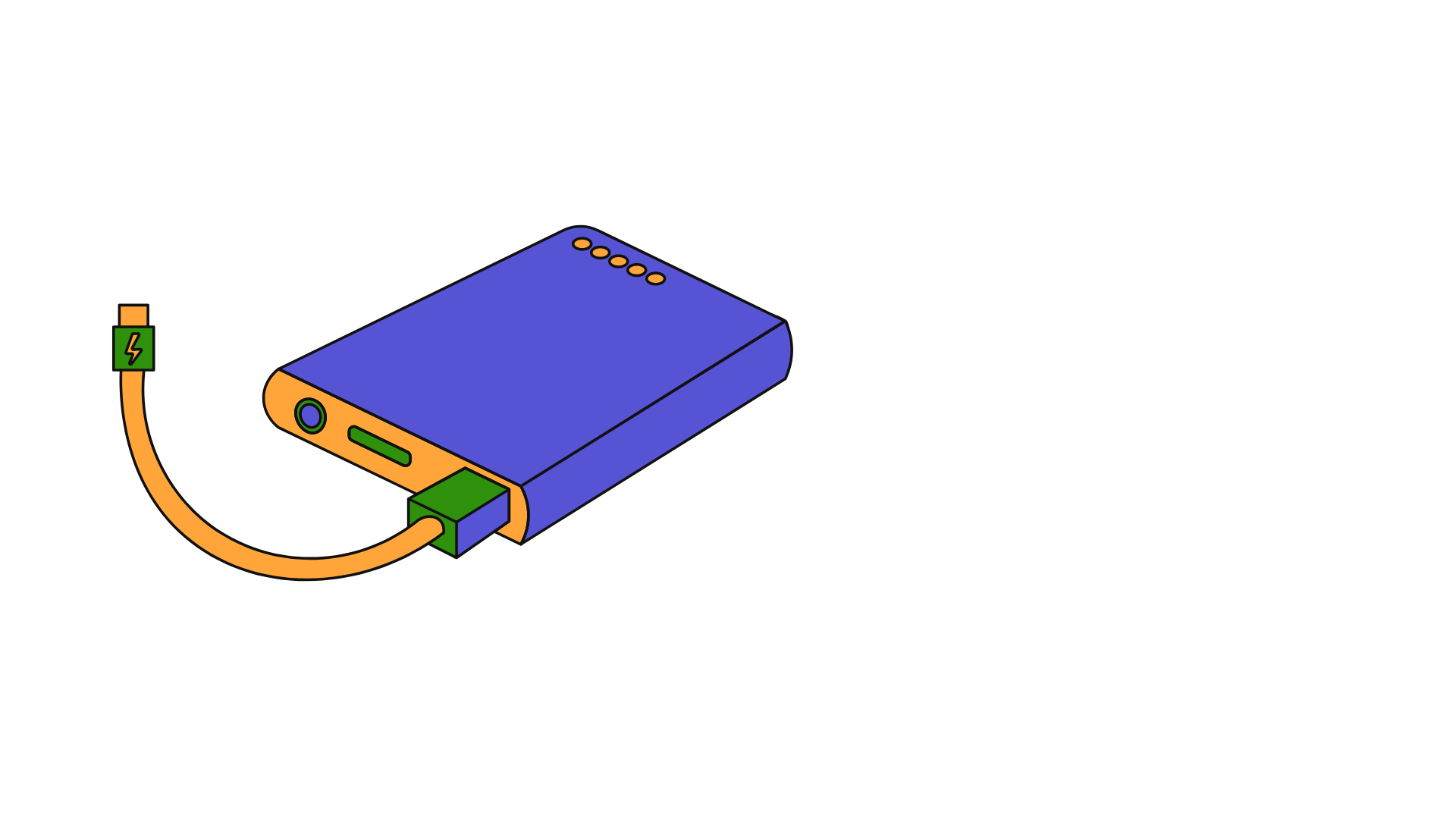


**Passo 5:** Conecte um fio na porta n° 4 do microcontrolador e o conecte no resistor do Led: ligue como na imagem abaixo.

**Passo 6:** Por meio da porta USB indicada, insira o código recebido do Iniciator e do Responder para cada microcontrolador, respectivamente, através da opção upload na Arduino IDE. 



**Passo 7:** Conecte os microcontroladores em power banks.

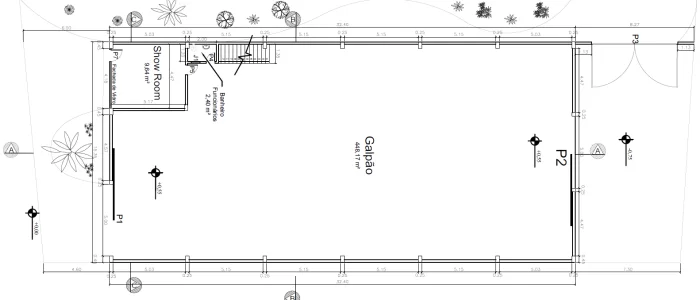


# 4. Guia de Instalação

### (sprint 4)

A solução final consiste em um conjunto de microcontroladores, com funções de beacon e tags para localização indoor. Nesse sentido, a seguir está o **guia completo de instalação desses equipamentos nos ambientes fechados**.

*\*A imagem da planta baixa, usada nessa seção, é meramente ilustrativa e descreve apenas como deverá ser realizada a distribuição e instalação dos beacons no galpão.\**



**Componentes necessários para instalação de Componentes físicos:**

* Mínimo de 4 microcontroladores ESP 32-S3;
* 1 Cabo USB-C;
* 1 Protoboard convencional;
* 1 LED
* Jumpers (macho macho);
* Resistores (220 *Ω);*
* Roteador com Wi-Fi;

**Materiais/aplicações necessários para configuração ou modificação de código:**

* Computador ou notebook;
* [Arduino IDE](https://docs.arduino.cc/software/ide-v1/tutorials/Windows);
* [Visual Studio Code;](https://visualstudio.microsoft.com/pt-br/downloads/)
* [Code Sandbox](https://codesandbox.io/signin?continue=/dashboard/recent) ou outra aplicação para deploy;

Para iniciar a instalação, é preciso adicionar os códigos respectivos a cada microcontrolador, todos são iguais, a diferença será o tipo de código implementado a cada um. Abaixo estão listados os códigos hospedados no GitHub com seus respectivos links:

**1°**: É preciso fazer download dos códigos para alterar os nomes e senhas das redes Wi-Fi para aquela da rede de acesso para os ESP’s *Initiator*, pois é por ela que é realizada a conexão com a API e banco de dados. Faça isso, após instalar a [IDE arduino](https://docs.arduino.cc/software/ide-v1/tutorials/Windowshttps://docs.arduino.cc/software/ide-v1/tutorials/Windows):

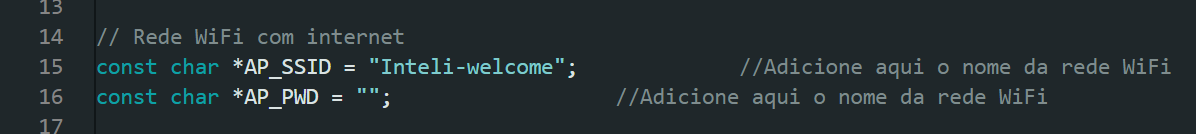
**2°:**  Configure os microcontroladores com os códigos abaixo, para fazer isso é necessário ter um computador/notebook, e nele a IDE do arduino para upload e alteração de código. Abaixo estão os links com os códigos nos [GitHub’s do FindU](https://github.com/2022M4T3-Inteli/Projeto2):

***Responder* (2 microcontroladores):** link;

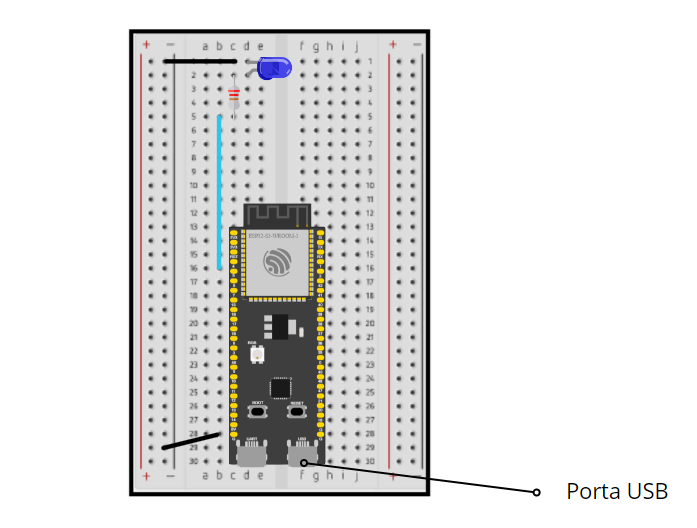
***Initiator* (1 microcontrolador): link**

**Initiator-Responder (1 microcontrolador): link**

Substitua o nome da rede dentro das aspas, para a rede local do galpão (a mesma do rotador), o “AP\_SSID” significa o nome da rede de conexão e o “AP\_PWD” significa a senha da rede em questão, **não esqueça essa substituição dentro dos Initiators.**

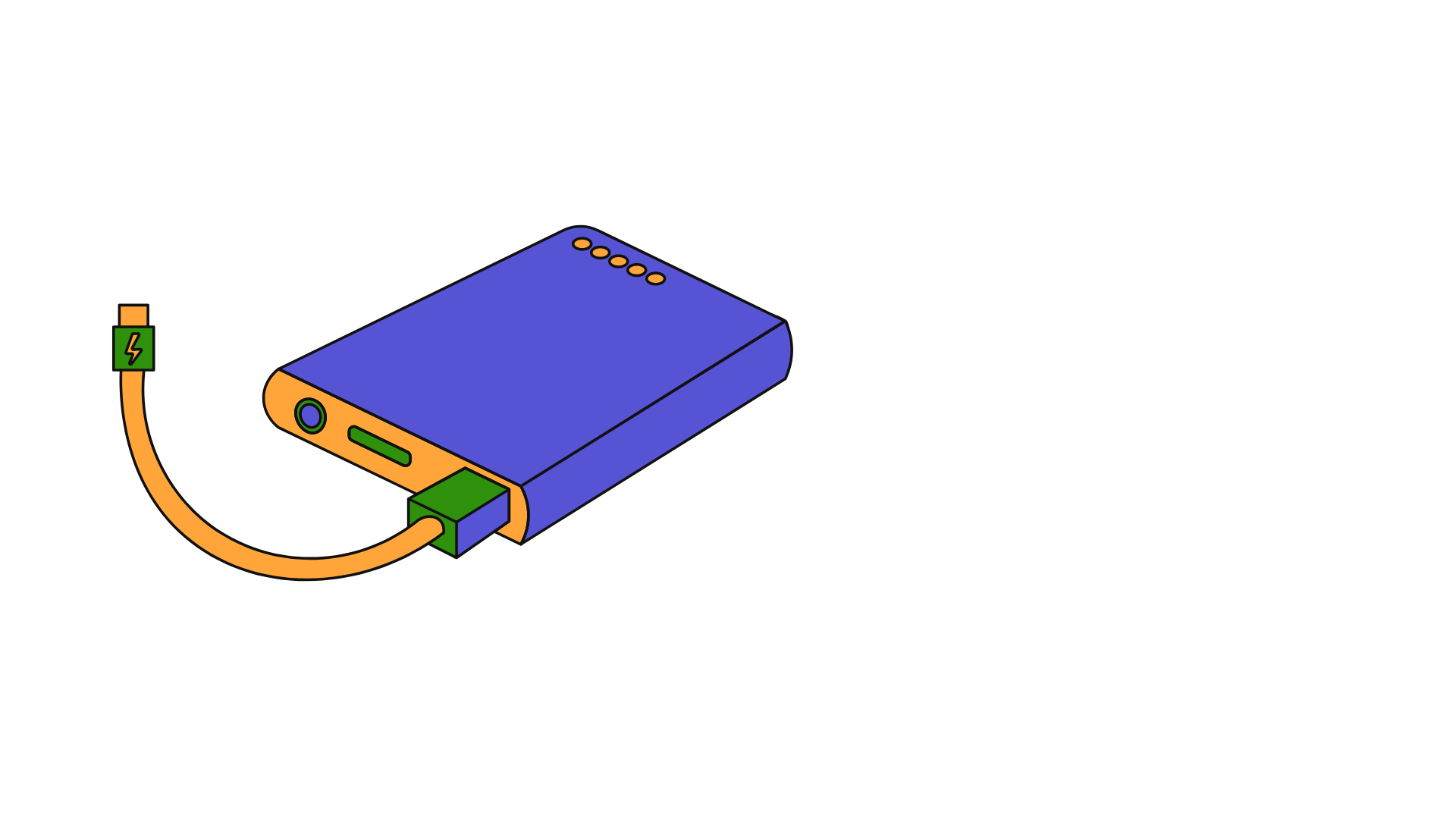
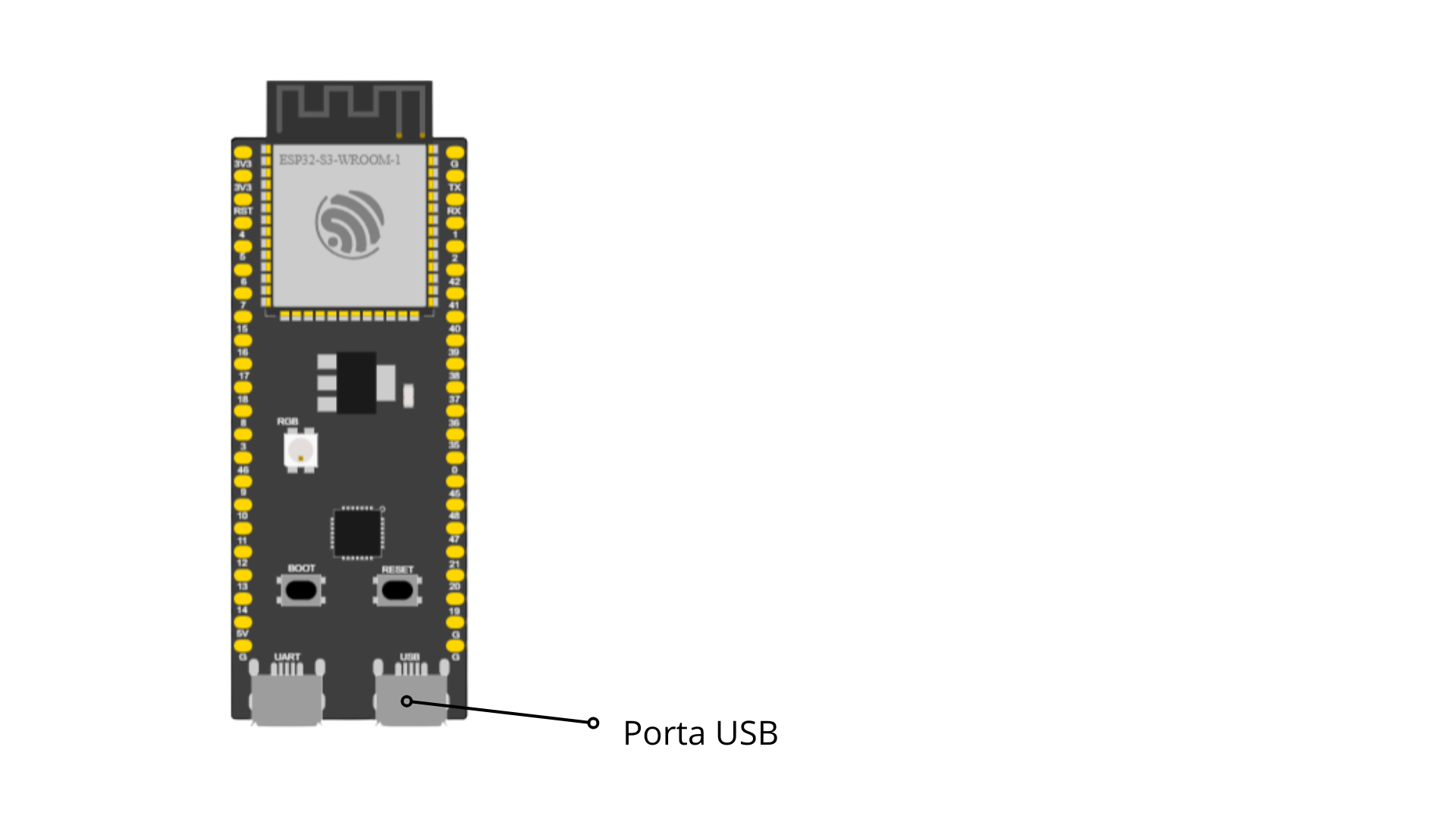


**3°:** Após alterar e fazer o upload dos códigos para as placas, é o momento de realizar a montagem presente no item [3. Guia de Montagem](#_v51amp5m28ia) para a Tag (microcontrolador com o código Initiator), realize essa montagem para fixação e teste de funcionamento através do Led.

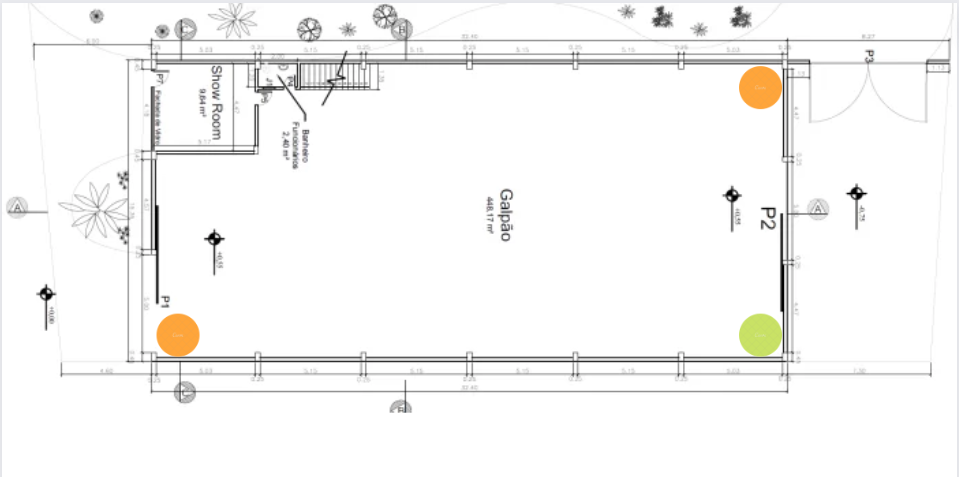


**Obs:** Os beacons funcionam apenas com o microcontrolador, porém é possível colocá-los em uma protoboard ou realizar alguma soldagem para diferenciação ou evitar exposição à poeira ou superfícies irregulares.

**4°:** Conecte todos os ESP’s à uma fonte de bateria, de preferência powerbanks, para que haja alimentação para o funcionamento dos microcontroladores. Conecte na porta USB, como mostrada abaixo:



**5°:** Estamos quase no final, agora com todos os ESP’s montados e com o código interno, resta realizar a instalação dos beacons dentro do galpão. Para isso, é importante ressaltar que é altamente recomendável que eles fiquem posicionados nas extremidades para um maior alcance e precisão, assim como apresentado na ilustração com os pontos amarelos representando os beacons.

****

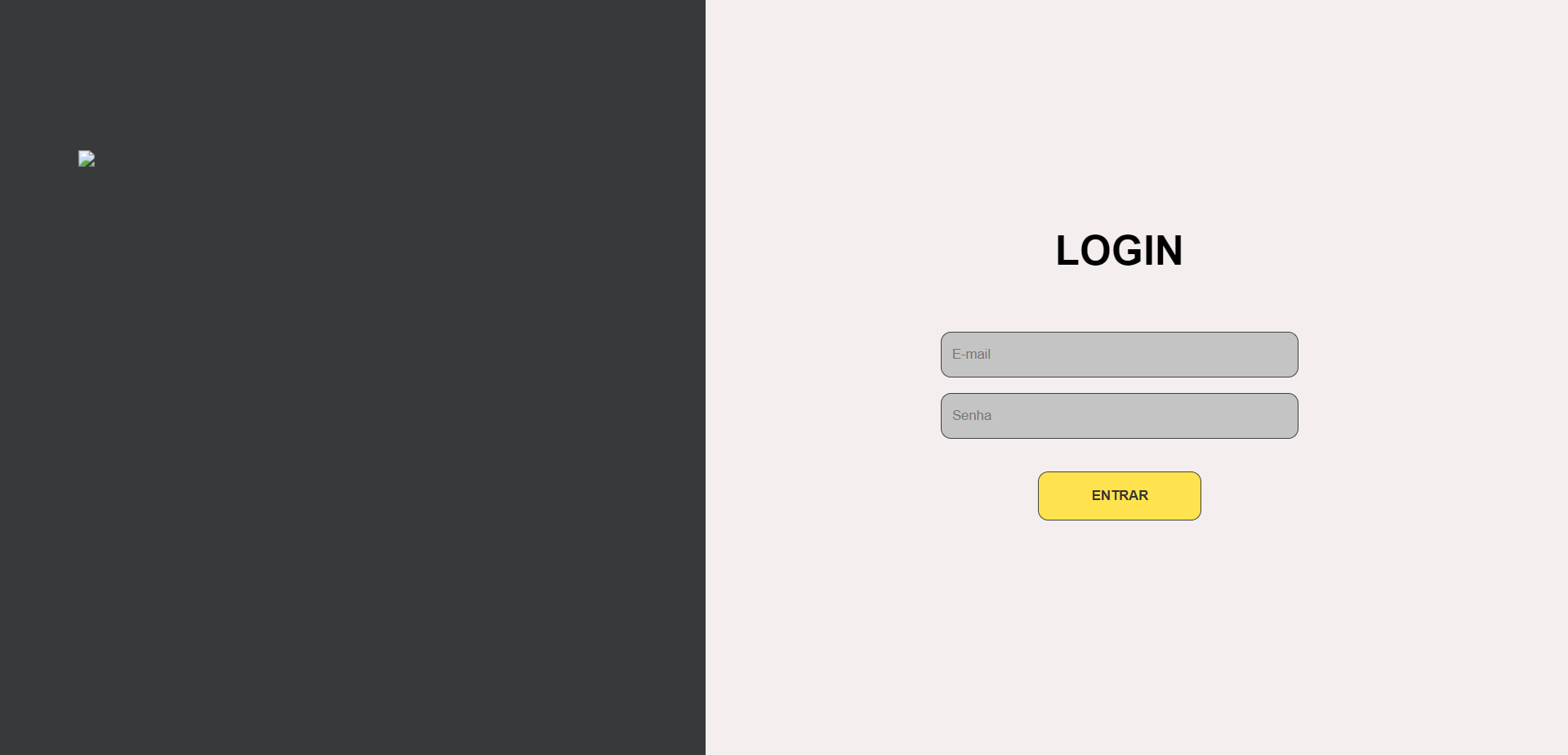
# Os pontos em amarelo representam os beacons com o código *Responder*, já o beacon em verde representa aquele com código *Initiator/Responder*, é importante que ele se situe entre os dois de maneira perpendicular (formando um ângulo de 90° graus), pois ele será responsável por coletar a distância entre os outros dois e enviar para o Front-end, onde será calculado o posicionamento através da triangulação. É importante ressaltar que as dimensões do galpão não ultrapassem os 12x12 metros, pois a partir dessas dimensões, o sinal do ESP se torna instável, o que aumenta a possibilidade de imprecisão na visualização e nos valores de distância apresentados.

**6°**: Após a instalação dos beacons é possível adicionar a tag no galpão, pois já haverá a conexão e a visualização na tela inicial da aplicação WEB.

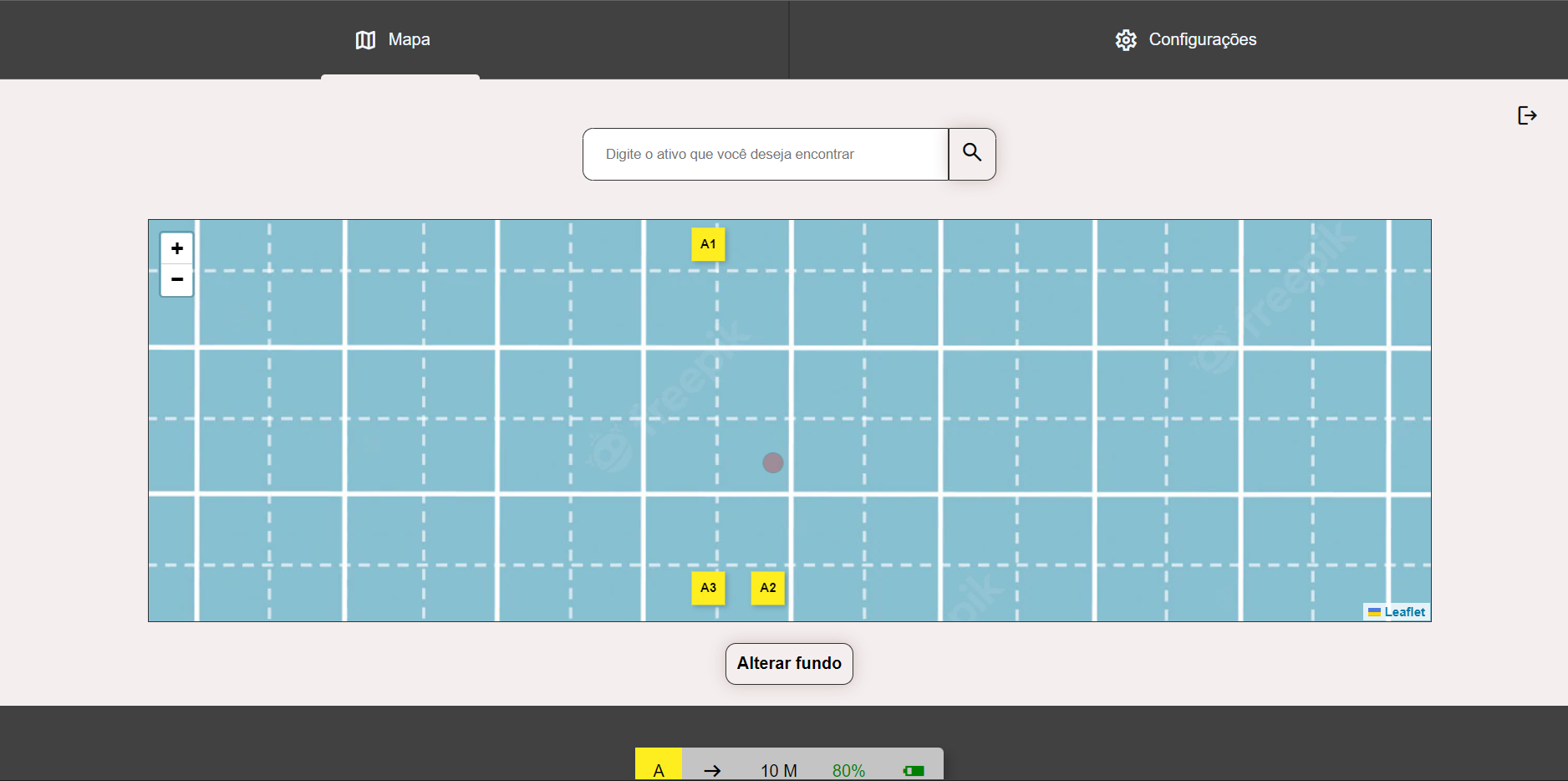


# 5. Guia de Configuração

A tela inicial do projeto é a tela de login, nela o usuário fará um rápido cadastramento, para logo acessar o sistema.

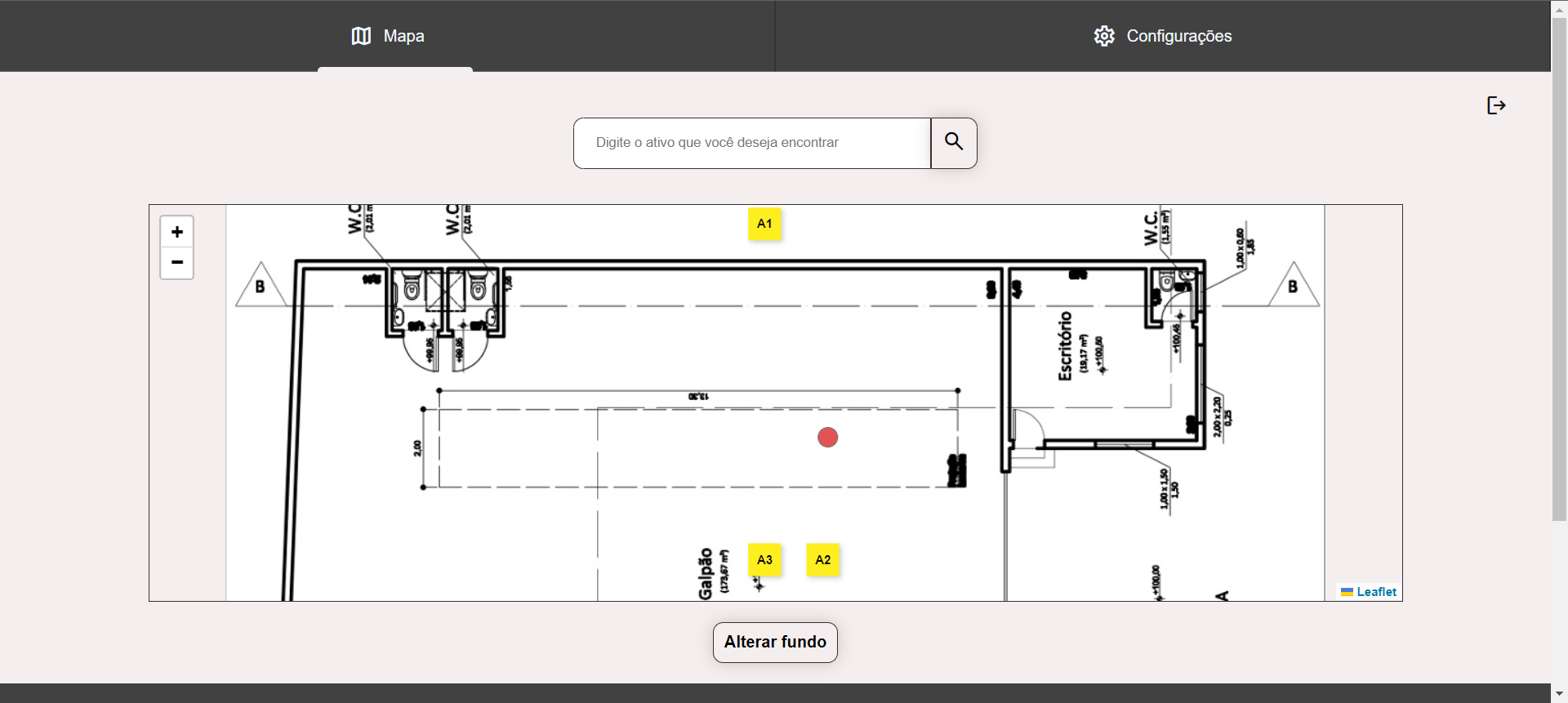


Ao clicar no botão entrar, o usuário é direcionado para uma tela de visualização de posicionamento e informações.

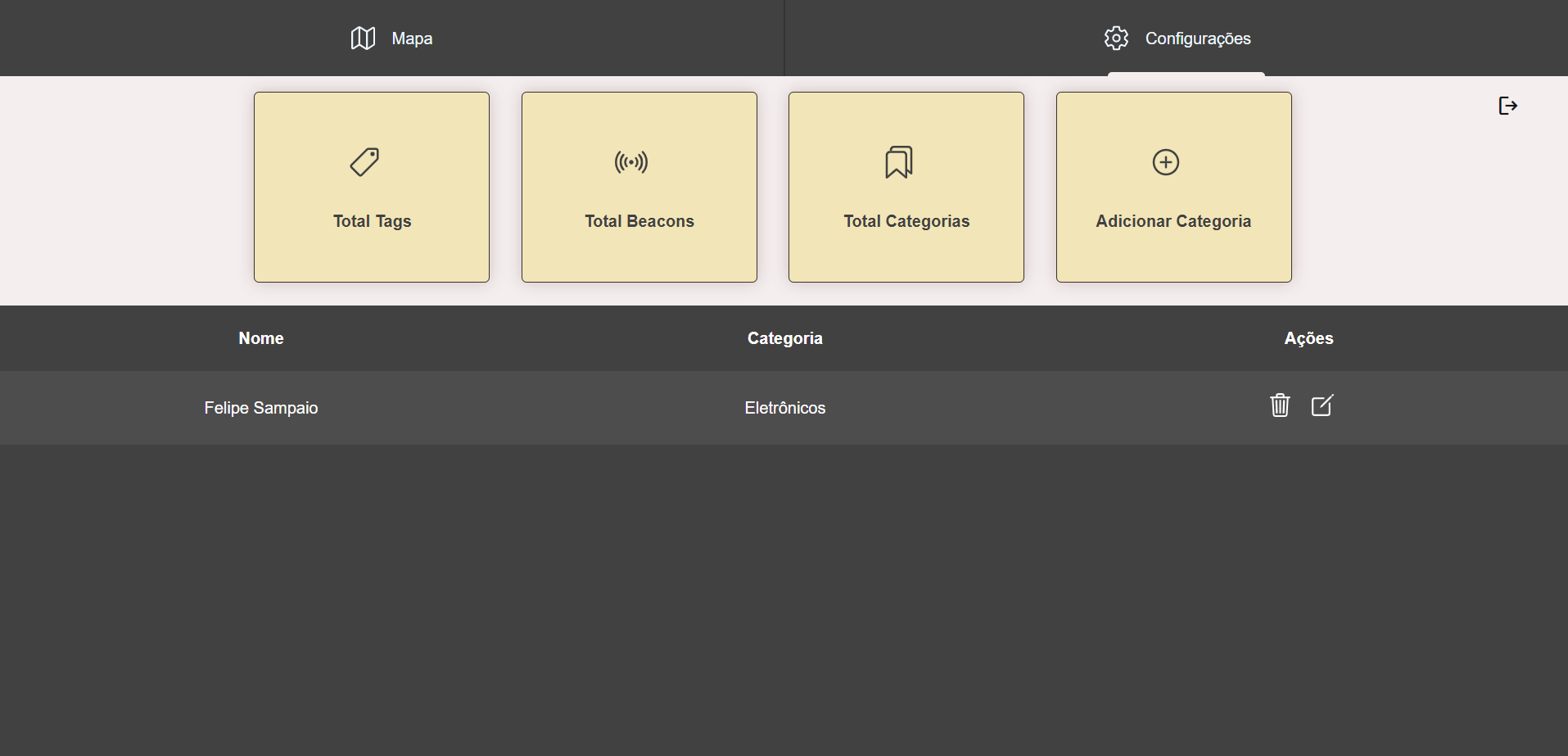


Nessa tela, é possível adicionar uma planta baixa do ambiente em que a tag estiver instalada, oferecendo uma noção mais precisa do posicionamento em relação ao ambiente. O usuário pode arrastar a tag e os beacons como quiser pelo mapa, porém as proporções de distâncias serão mantidas, ou seja ele arrasta todos como um bloco único.

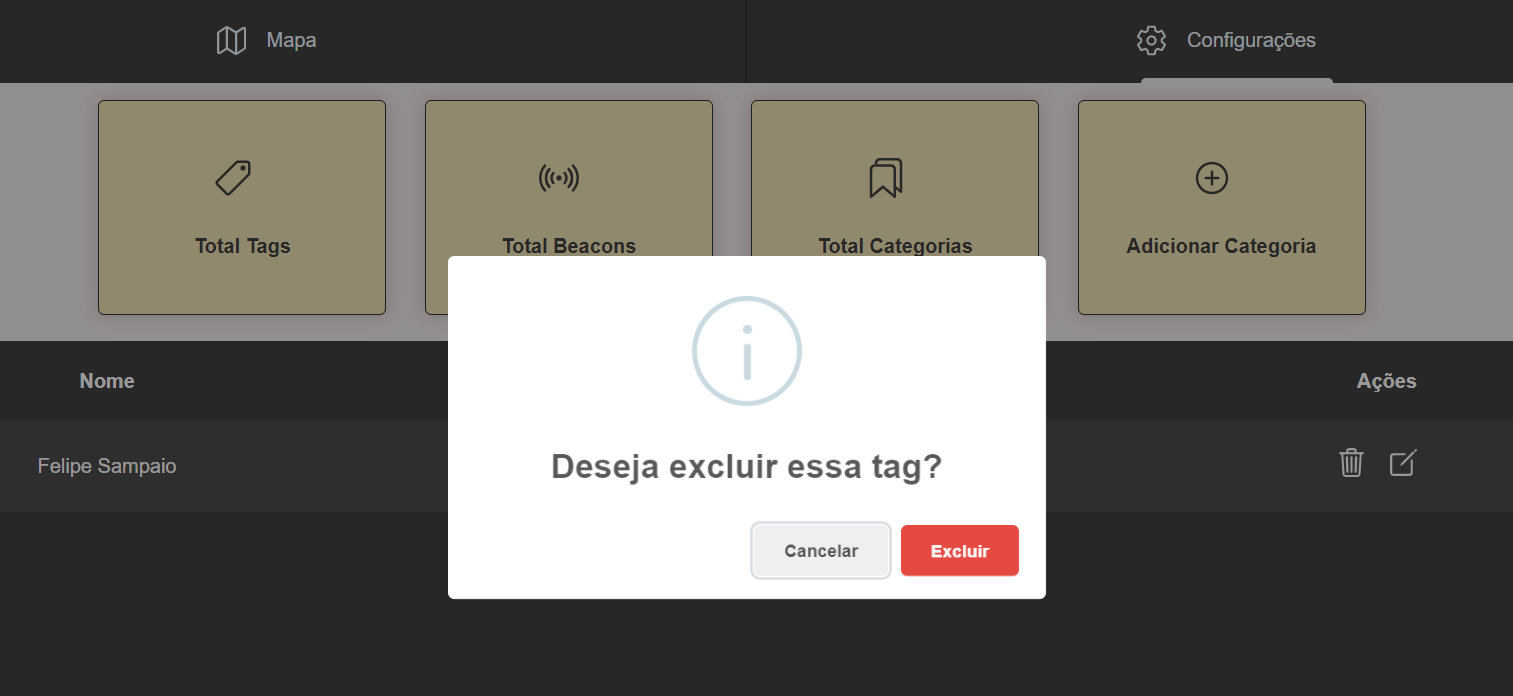
Para localização de erro, é preciso sempre verificar se há 3 beacons e pontos vermelhos como tag, pois são eles a principal parte da solução final, já em caso de erro na distância apresentada, é preciso estar atento se a tag realmente está no local apresentado no mapa, então com certa periodicidade verificar em que região do galpão está o objeto a ser identificado e se está coerente com a visualização :



Ao clicar na tela de configurações na navbar, o usuário é levado a uma tela de cadastramento e edição de tag já apresentada no mapa, ou seja é possível alterar o nome e categoria de uma tag, assim como cadastrar e incluir uma nova categoria:



Além disso, é possível ter uma rápida visualização de informações imediatas, tais como: “Total de tags”, “Total de beacons”, “Total de categorias”, junto a um cadastramento de uma nova categoria. Outra funcionalidade interessante são os feedbacks em forma de modais para executar uma ação, então o usuário precisa confirmar antes de executar alguma decisão:



# 6. Guia de Operação

### (sprint 5)

Após a instalação e configuração dos beacons, é o momento de acessar a aplicação WEB para uma visualização da posição da tag. O primeiro contato do usuário com a interface é a tela de login, na qual é possível cadastrar um email e senha de acesso:

# 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