

# Manual de Instruções

## CONNECT 4

### Rede Nacional para Educação e Pesquisa

## Controle do Documento

### Histórico de revisões

07/12/2023	Antonio Artimonte, Matheus Ribeiro dos Santos	4.0	Atualizações gerais no manual conforme reunião com a orientadora
------------	---	-----	--

Data	Autor	Versão	Resumo da atividade
29/11/2023	Antonio Artimonte, Fernando Machado, Matheus Ribeiro	1.0	Criação do documento. Atualização de todos os setores. Finalização de versão.
04/12/2023	Antonio Artimonte, Fernando Machado, Matheus Ribeiro	2.0	Reedição das atualizações anteriores.
05/12/2023	Antonio Artimonte, Fernando Machado, Matheus Ribeiro	3.0	Reedição das atualizações anteriores.

# Índice

<b>1. Introdução.....</b>	<b>4</b>
1.1. Solução.....	4
1.2. Arquitetura da Solução.....	4
<b>2. Componentes e Recursos.....</b>	<b>5</b>
2.1. Componentes de hardware.....	5
2.2. Componentes externos.....	6
2.3. Requisitos de conectividade.....	6
<b>3. Guia de Montagem.....</b>	<b>6</b>
<b>4. Guia de Instalação.....</b>	<b>9</b>
4.1 Estrutura de Hardware:.....	9
a. Passo 1.....	9
b. Passo 2.....	10
4.2 Estrutura do Software.....	10
c. Passo 3:.....	10
<b>5. Guia de Configuração.....</b>	<b>10</b>
5.1 Instalações prévias.....	11
5.2 Principais bibliotecas.....	12
5.3 Encapsulamento e configurações.....	14
<b>6. Guia de Operação.....</b>	<b>16</b>
6.1 Tela de Login.....	16
6.2 Tela de perfil.....	17

6.3 Main Page.....	17
6.4 Dashboard do ativo selecionado.....	18
6.5 Página relatórios.....	19
6.6 Página favoritos.....	20
6.7 Landing-Page.....	20
6.8 Página de cadastro.....	21
6.9 Hardware.....	22
<b>7. Troubleshooting.....</b>	<b>23</b>
<b>8. Créditos.....</b>	<b>28</b>

## Índice de Imagens

Imagem 1 - Arquitetura da solução - página 4
Imagem 2 - Materiais para o hardware - página 5
Imagem 3 - Etapa 1 da montagem - página 6
Imagem 4 - Etapa 2 da montagem - página 7
Imagem 5 - Etapa 3 da montagem - página 7
Imagem 6 - Etapa 4 da montagem - página 8
Imagem 7 - Etapa 5 da montagem - página 8
Imagem 8 - Tela do site do Arduino. - página 11
Imagem 9 - Extração - página 11

Imagem 10 - Pasta extraída - página 12

Imagem 11 - Barra lateral do Arduino IDE - página 13

Imagem 12 - Barra de pesquisa de bibliotecas do Arduino IDE - página 13

Imagem 13 - Barra de pesquisa de bibliotecas do Arduino IDE - página 14

Imagem 14 - Senha e Nome do Wi-Fi - página 15

Imagem 15 - Botão de upload no Arduino IDE - página 15

Imagem 16 - Tela de Login - página 16

Imagem 17 - Meu perfil - página 17

Imagem 18 - Main Page - página 18

Imagem 19 - Dashboard - página 18

Imagem 20 - Página do relatório- página 19

Imagem 21 - Página dos favoritos - página 20

Imagem 22 - Landing Page - página 21

Imagem 23 - Página do cadastro- página 21

Imagem 24 - Hardware- página 22

## Índice Tabelas

Tabela 1 - Visão de funcionamento dos componentes - página 9

Tabela 2 - Troubleshooting - página 17

- Diretrizes de instalação
- Procedimentos de configuração
- Orientações de operação
- Resolução de problemas (Troubleshooting)

# 1. Introdução

Nos subtópicos abaixo, segue uma breve descrição a respeito da solução GeoLoc e a arquitetura da solução da mesma.

## 1.1. Solução

Nosso projeto, GeoLoc, tem como intuito auxiliar a Rede Nacional de Educação e Pesquisa (RNP) a localizar os seus ativos geograficamente, utilizando um sistema de *checkpoints*<sup>1</sup> e de GPS.

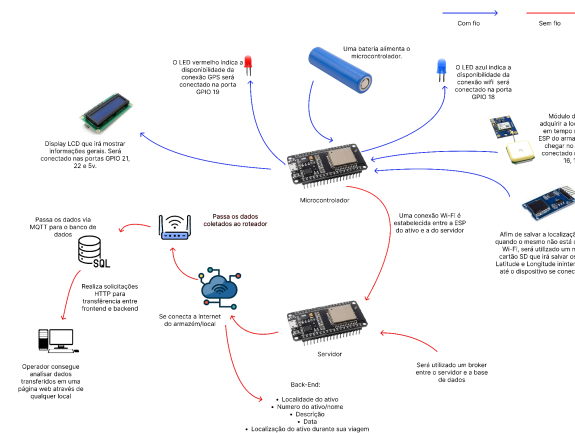
O propósito do manual de orientações é guiar os colaboradores da RNP, possibilitando uma compreensão do que foi realizado e facilitando a usabilidade de maneira independente e direta, possibilitando rápida e fácil instalação. O manual está organizado em seções que incluem:

- Componentes e recursos
- Instruções de montagem

<sup>1</sup> Checkpoints: locais em que o ativo será capaz de mandar dados

## 1.2. Arquitetura da Solução

Imagem 1 - Arquitetura da solução



Fonte: o próprio autor.

A imagem acima (**Imagem 1**) representa a arquitetura da solução proposta, na qual, a parte superior, contém um microcontrolador (*ESP32*), como base central que encaminha informações tanto para o LCD, quanto para ambos os LED presentes, além de receber informações do módulo GPS (*NEO-6M*). Para fins de informação, o LED vermelho representa a disponibilidade do GPS, enquanto o azul representa a conexão com a *Wi-Fi* do *ESP32* com o outro *ESP*. Já o LCD funciona para mostrar se os dados já foram ou não enviados. No servidor, por sua vez, o *ESP32* funciona como um *Access Point* de *Wi-Fi* que irá por sua vez se conectar ao *ESP* do ativo, que irá passar os dados de GPS para o servidor, e logo passar para a nuvem.

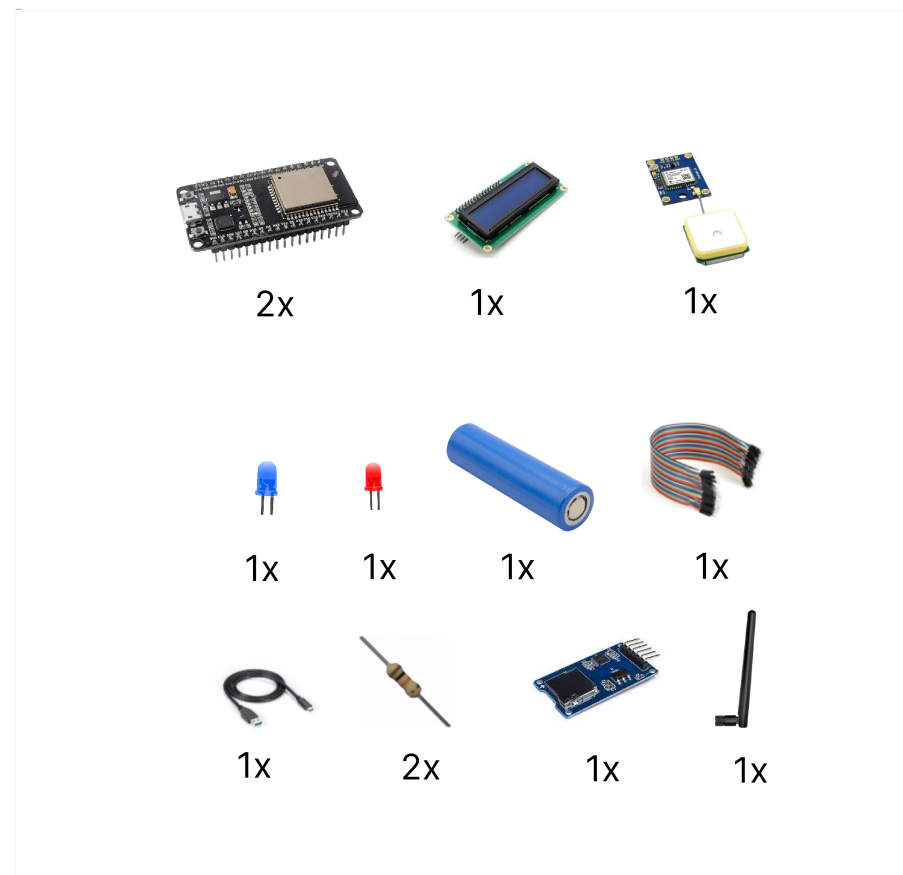
## 2. Componentes e Recursos

Abaixo seguem tópicos a respeito dos componentes de Hardware e os recursos necessários para construção

### 2.1. Componentes de hardware

Na imagem abaixo (**Imagem 2**) será possível visualizar todos os componentes necessários para se instalar o *hardware*. Além da quantidade dos mesmos componentes.

Imagem 2 - Materiais para o *hardware*



Fonte: o próprio autor.

Dos objetos listados acima, os mesmos são: 2x *ESP32*, 1x LCD 16x2, 1x GPS NEO-6M, 1x LED Azul, 1x LED Vermelho, 1x bateria lítio/íon, 1x Conjunto cabos Jumpers macho-fêmea, 1x Cabo USB-MicroUSB, 2x Resistores 220Ohm, 1x Módulo micro SD, 1x Antena WI-FI 2.4 GHz

## 2.2. Componentes externos

Na implementação proposta, é indispensável contar com um computador com acesso à internet. Além disso, serão empregados o *HiveMQ* como *broker* e o *SQLite 3* como sistema de gerenciamento de banco de dados.

## 2.3. Requisitos de conectividade

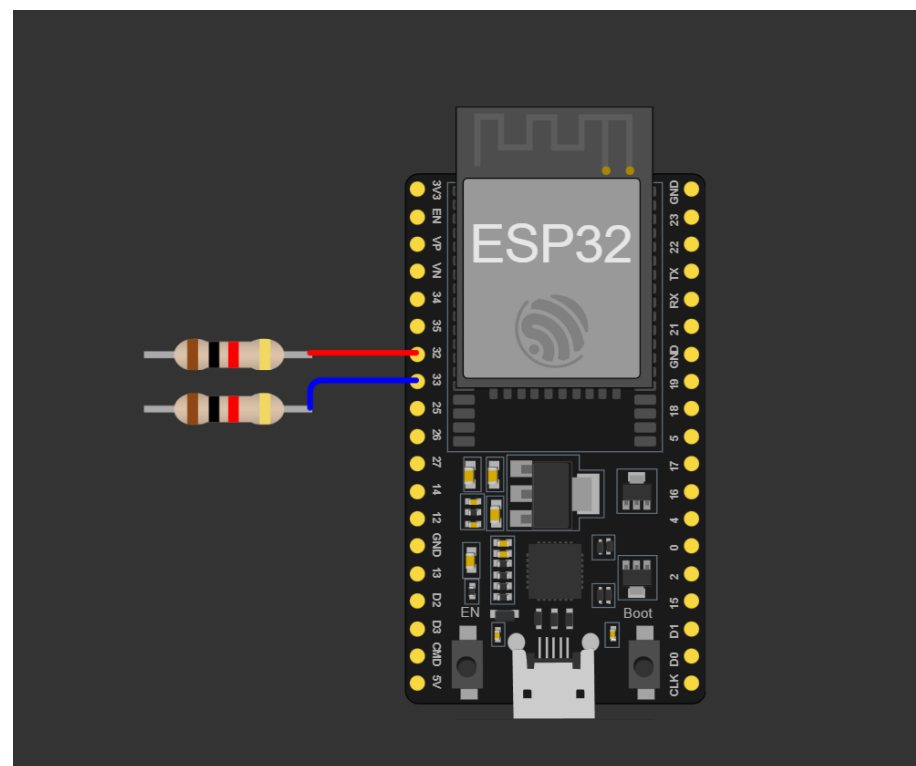
- Protocolo *MQTT*: protocolo de comunicação do *broker* com o *ESP32*;
- Protocolo *HTTPS*: protocolo de internet;
- *Back end*: está composto de *endpoints* para todas as páginas do front-end e um *endpoint* de comunicação com o banco de dados em *Node.js*.

## 3. Guia de Montagem

### Primeiro passo (Imagem 1) - Montagem do *ESP32*:

Passo 1: Conectar uma entrada do *ESP* ao cabo Jumper macho-macho ao resistor como mostrado na Figura 2;

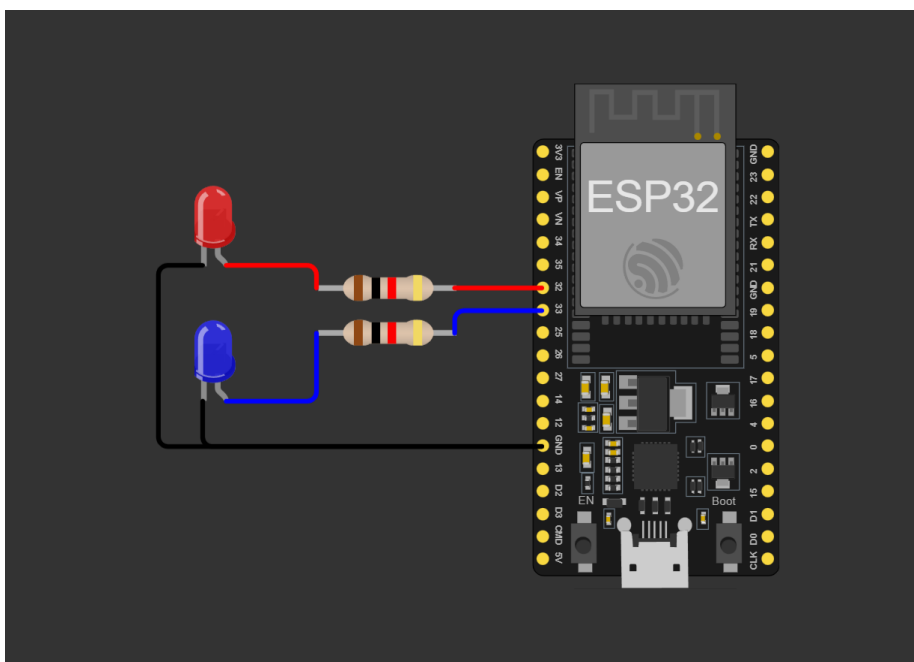
Imagem 3 - Parte 1 da montagem



Fonte: o próprio autor.

Passo 2: Conectar os leds nos resistores e nas devidas conexões com o *ESP32*. O LED vermelho será referente a porta GPIO 32 do *ESP* e o LED azul a porta GPIO 33 do *ESP* como pode ser visto na Figura 3;

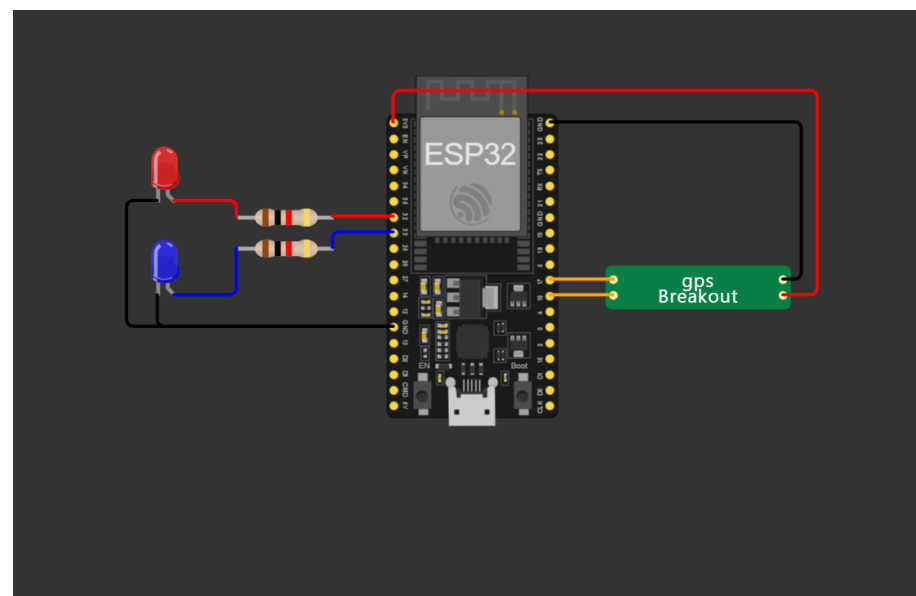
Imagem 4 - Parte 2 da montagem



Fonte: o próprio autor.

Passo 3: Conectar o GPS no *ESP*, a entrada VCC do GPS será conectada a 3v3 do *ESP*, GND do GPS no GND do *ESP*, RX na porta 17 e TX 16 seguindo o mesmo princípio da Figura 4;

Imagem 5 - Parte 3 da montagem

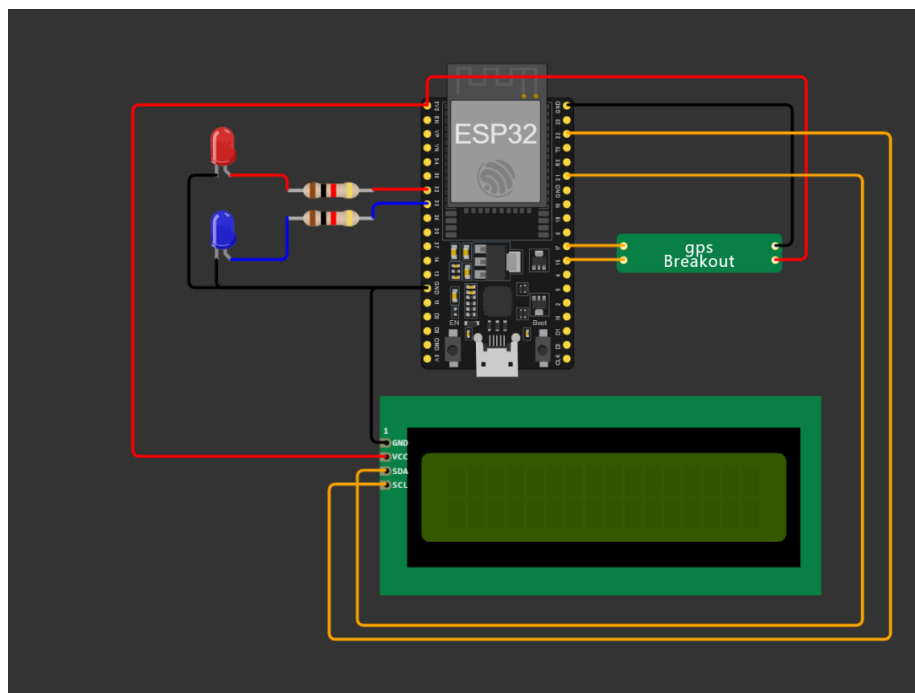


Fonte: o próprio autor.



Passo 4: Conectar o LCD no *ESP*, GND do LCD no GND do *ESP*, a entrada do LCD no pino de 5 volts do *ESP*, SDA no pino 21 do *ESP*, SCL no pino 22 do *ESP* do mesmo modo que na Figura 5;

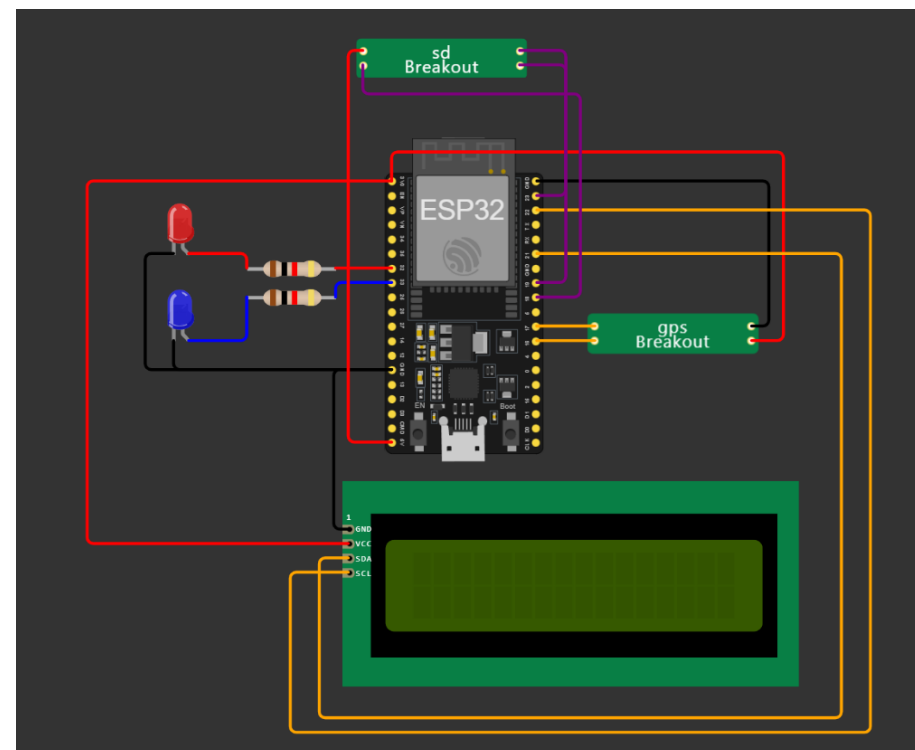
Imagem 6 - Parte 4 da montagem



Fonte: o próprio autor.

Passo 5: Conectar o MOSI do módulo SD no pino GPIO 23, MISO do SD no GPIO 19, CS do módulo SD no GPIO 5, VCC do SD no 5v e GND do SD no GND do *ESP32*

Imagem 7 - Parte 5 da montagem.



Fonte: o próprio autor.

## 4. Guia de Instalação

A fim de assegurar o correto funcionamento de ambos os dispositivos, é crucial realizar, em um primeiro momento, a instalação dos cabos de maneira adequada. Este procedimento inicial é fundamental para estabelecer uma conexão eficiente entre os aparelhos, contribuindo para um desempenho otimizado.

### 4.1 Estrutura de Hardware:

#### a. Passo 1

É necessário verificar o funcionamento tanto dos LEDs, que devem ser conectados aos respectivos conectores do *ESP32* através de um resistor de 220 Ohms, o LCD conectado a entrada 5V do ESP e o GPS conectado a porta 3V3, qualquer uma destas portas alteradas poderá resultar em um curto-circuito ou mau funcionamento do GPS

Tabela 1 - Visão de Funcionamento dos Componentes

Componente	Cor/Informação	Status
Display	Informação conectividade ao protocolo MQTT	MQTT conectado ou não
Display	Informação conectividade a Wi-Fi	Status da conexão Wi-Fi
Display	GPS conectando	GPS tentando a conexão com satélite
Display	GPS não encontrado	Erro com os fios do GPS ou GPS
Display	GPS encontrado	GPS encontrado e funcionando
Display	Dados localização adquiridos	Dados da localização foram adquiridos
Led vermelho	Piscando	Conectando
Led vermelho	Vermelho	GPS não conectado
Led azul	Piscando	Conectando
Led azul	Azul	Wi-Fi conectada
Led azul	Apagado	Problemas no cabeamento/energia
Led vermelho	Apagado	Problemas no

		cabeamento/energia
--	--	--------------------

## b. Passo 2

Após minuciosa verificação dos status individuais de cada componente mencionado anteriormente, é imperativo proceder para a próxima etapa do processo. Este passo, essencial para a continuidade do procedimento, assegura que todas as condições estejam devidamente analisadas antes de avançar.

## 4.2 Estrutura do Software

A estrutura do Software consiste na estrutura do código do ESP32

### c. Passo 3:

O software é composto por duas etapas, a parte do ESP32 e a parte relacionado a aplicação, sendo dividida abaixo em 2 subtópicos diferentes, 3.1 e 3.2

### 3.1 Software no Servidor Embarcado:

A fim de permitir a instalação do servidor no *ESP32* disponível no armazém, torna-se imprescindível estabelecer a sua conexão com a rede *Wi-Fi* interna do armazém. Essa interligação é de fundamental importância, pois possibilita que os dados adquiridos pelo dispositivo sejam devidamente enviados para a nuvem, contribuindo assim para a eficácia do sistema.

- *ssidWIFI*: ( Nome da rede Wi-Fi do local )
- *senhaWIFI*: ( Senha da respectiva rede Wi-Fi do local )

### 3.2 Software com Hospedagem Online

O software se destaca pela inclusão de uma aplicação web interativa, enriquecendo a experiência do usuário ao proporcionar uma visualização mais aprimorada dos dados de localização de cada ativo. Essa funcionalidade visa não apenas oferecer uma perspectiva mais detalhada, mas também assegurar uma compreensão mais eficaz das informações disponíveis.

Além disso, o software incorpora um sólido sistema de login, cujo propósito é reforçar a segurança do sistema, prevenindo qualquer possível vazamento de dados. A autenticação é crucial para garantir o acesso autorizado e a proteção integral das informações sensíveis, consolidando a confiabilidade do software em ambientes diversos.

## 5. Guia de Configuração

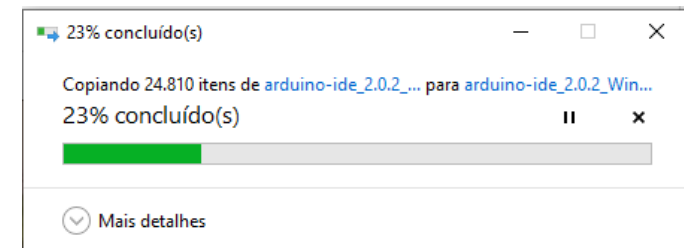
Abaixo segue um guia de instalação e configuração da parte "virtual" onde será requisitado a instalação de alguns aplicativos, junto da configuração dos mesmos.

## 5.1 Instalações prévias

Para realizar a configuração adequada do *ESP32*, e, assim, do protótipo, é necessário instalar o *Arduino IDE*, um software de edição de código para microprocessadores. O link de download pode ser encontrado no link: <https://www.arduino.cc/en/software>.

Acessando o link indicado, faça o download baseando-se no seu sistema operacional em utilização. Para o caso deste manual, o utilizado foi o “*Windows ZIP File*” que já acompanha o sistema operacional do *Windows*. Caso use *MACOS*, apenas abra diretamente.

Imagem 9 - Extração.



Fonte: o próprio autor.

Imagem 8 - Tela do site do Arduino.

Downloads



Fonte: [Site do Arduino](https://www.arduino.cc/en/software)

A extração costuma ser rápida, mas depende da capacidade de processamento do dispositivo utilizado para a configuração.

A pasta abaixo apresenta a pasta resultante após a extração, demonstrando, entre os arquivos, um atalho para o aplicativo.

## 5.2 Principais bibliotecas

Dando continuidade ao processo de instalação, deve-se adicionar as bibliotecas necessárias para o funcionamento do dispositivo.

É ideal que se siga o procedimento abaixo com todas as bibliotecas indicadas, sendo estas:

<iostream> - versão C++11

<cmath> - versão C++11

"NMEA.h" - versão 1.2.0

<WiFiMulti.h> - versão 1.8.0

<HTTPClient.h> - versão 0.5.0

<TinyGPSPlus.h> - versão 1.0.3

"UbidotsEsp32Mqtt.h" - versão 2.5.8

"secrets.h" - versão 1.1.0

<WiFiClientSecure.h> - versão 1.0.0

"WiFi.h" - versão 0.0.3

"ESPAsyncWebServer.h" - versão 1.0.1

<PubSubClient.h> - versão 2.8

<ArduinoJson.h> - versão 6.21.4

Imagem 10 - Pasta extraída.

Nome	Data de modificação	Tipo	Tamanho
locales	04/12/2022 12:19	Pasta de arquivos	
resources	04/12/2022 12:19	Pasta de arquivos	
swiftshader	04/12/2022 12:38	Pasta de arquivos	
Arduino IDE	04/12/2022 12:19	Aplicativo	137.278 KB
chrome_100_percent.pak	04/12/2022 12:19	Arquivo PAK	139 KB
chrome_200_percent.pak	04/12/2022 12:19	Arquivo PAK	203 KB
d3dcompiler_47.dll	04/12/2022 12:19	Extensão de aplica...	4.419 KB
ffmpeg.dll	04/12/2022 12:19	Extensão de aplica...	2.640 KB
icudtl	04/12/2022 12:19	Arquivo DAT	9.977 KB
libEGL.dll	04/12/2022 12:19	Extensão de aplica...	432 KB
libGLSv2.dll	04/12/2022 12:19	Extensão de aplica...	7.648 KB
LICENSE.electron	04/12/2022 12:19	Documento de Te...	2 KB
LICENSES.chromium	04/12/2022 12:19	Chrome HTML Do...	5.306 KB
resources.pak	04/12/2022 12:38	Arquivo PAK	5.733 KB
snapshot_blob.bin	04/12/2022 12:38	Arquivo BIN	49 KB
v8_context_snapshot.bin	04/12/2022 12:38	Arquivo BIN	161 KB
vk_swiftshader.dll	04/12/2022 12:38	Extensão de aplica...	4.363 KB
vk_swiftshader_icd	04/12/2022 12:38	Arquivo JSON	1 KB
vulkan-1.dll	04/12/2022 12:38	Extensão de aplica...	716 KB

Fonte: o próprio autor.

<Wire.h> - versão 0.3.2

<LiquidCrystal\_I2C.h> - versão 1.1.2

"FS.h" - versão 1.1.0

"SD.h" - versão 1.0.0

"SPI.h" - versão 1.1.3

Acesse o ícone abaixo, presente na aba lateral do Arduino IDE (cuja instalação está descrita acima).

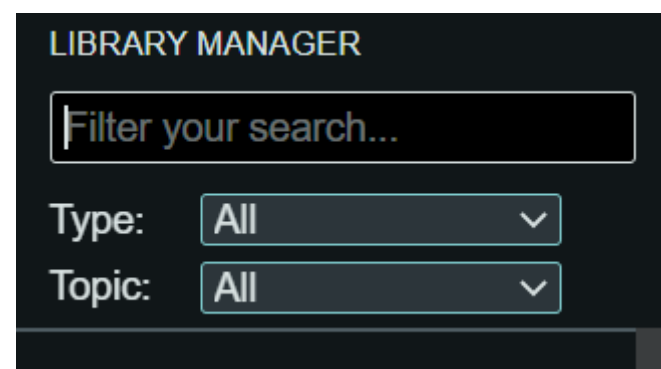
Imagem 11 - Barra lateral do Arduino IDE.



Fonte: o próprio autor.

Insira o nome da biblioteca que está indicada na lista acima e busque pelo resultado.

Imagem 12 - Barra de pesquisa de bibliotecas do Arduino IDE.



Fonte: o próprio autor.

Aperte no botão "INSTALL". Após isso, repita o processo com as outras bibliotecas indicadas.

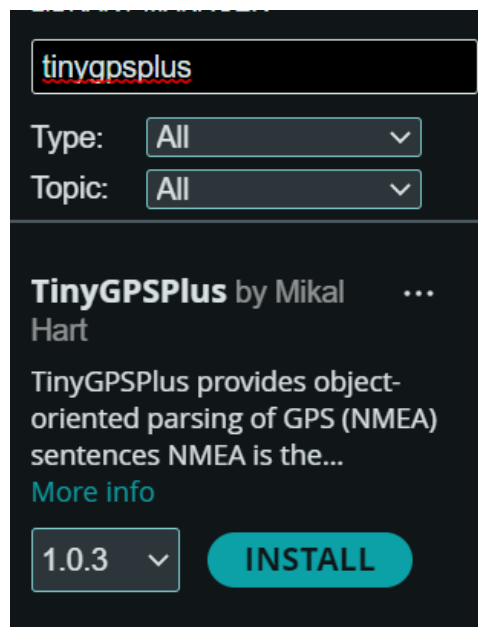
## 5.3 Encapsulamento e configurações

Para finalizar a instalação dos microprocessadores, é preciso adicionar os códigos para o *ESP32* que ficará junto do ativo e ao *ESP32* que ficará dentro do armazém. Ambos os códigos a serem utilizados estão presentes dentro do repositório do GitHub do grupo.

Cada um dos processadores acima utiliza um código distinto: Para o *ESP32* do ativo, utiliza-se o arquivo chamado de [AtivoCode.ino](#). Já para o *ESP32* presente dentro do armazém, utiliza-se o arquivo chamado [ArmazemCode.ino](#). Para ambos, deve-se copiar o código dentro do *Arduino IDE* e realizar as alterações necessárias.

O primeiro detalhe essencial a ser modificado é o SSID e a senha. No código [ArmazemCode.ino](#), é necessário que se alterem as informações de Wi-Fi, adicionando a rede local, e, assim, permitindo que o *ESP32* envie as informações via rede para um servidor central, que as armazenaria propriamente. Insira estes dados nas linhas indicadas abaixo:

Imagem 13 - Barra de pesquisa de bibliotecas do Arduino IDE



Fonte: o próprio autor.

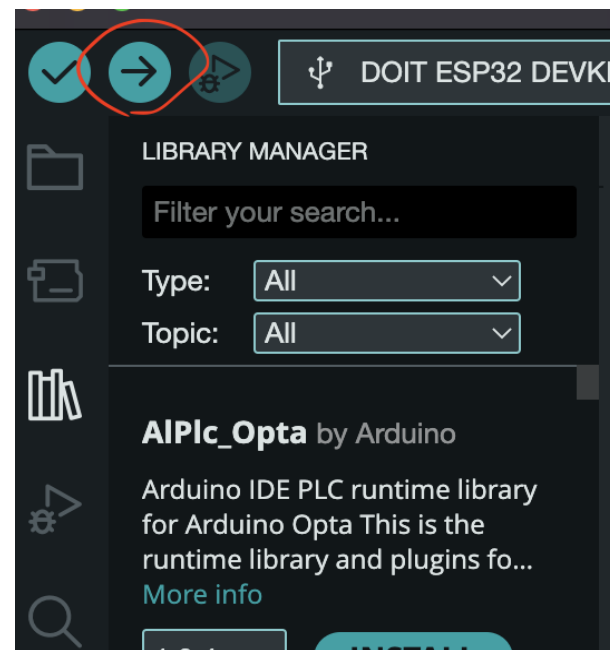
Imagem 14 - Senha e Nome do Wi-Fi

```
39
40  const char* ssidWIFI      = "Iphone12";    // Nome do WIFI
41  const char* passwordWIFI  = "rodrigo123";  // Senha do WIFI
```

Fonte: o próprio autor.

Finalmente, deve-se clicar no botão indicado abaixo, fazendo *upload* do código inserido no microprocessador. Após isso, os dispositivos estarão prontos para serem utilizados.

Imagem 15 - Botão de upload no Arduino IDE



Fonte: autoria própria.



## 6. Guia de Operação

Um Guia de Operação, que é uma parte essencial do nosso manual de instruções, é um documento elaborado para orientar os usuários sobre como realizar diversas operações ou tarefas relacionadas ao produto, ou equipamento. Ele fornece passos detalhados e explicações claras, garantindo que os usuários possam seguir as instruções de maneira eficaz. Este guia é projetado com a intenção de facilitar a compreensão, independentemente do nível de conhecimento prévio do usuário sobre o assunto.

### 6.1 Tela de Login

Quando o usuário acessar o *link* da página da solução desenvolvida, ele terá que inserir seus dados de *login*, ou seja, usuário e senha. Esse processo acontece na página mostrada na figura abaixo (**Imagem 16**).

Imagem 16 - Tela de Login



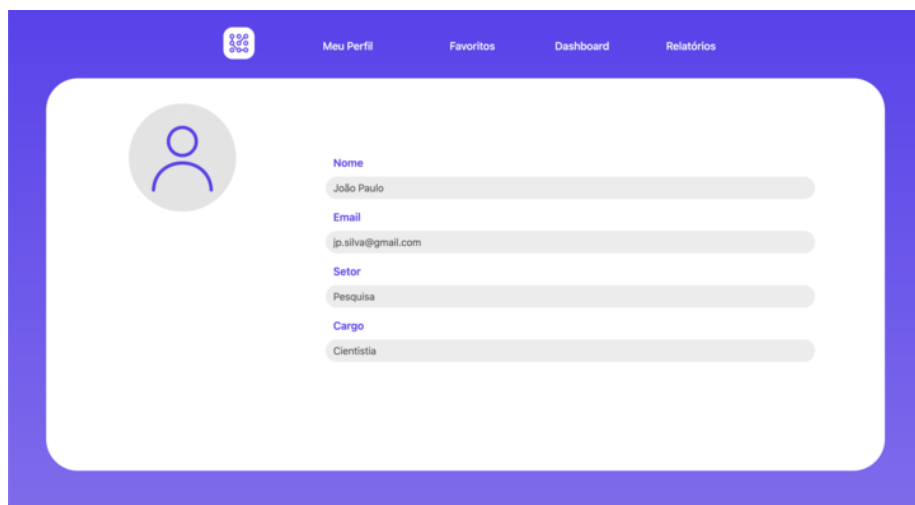
Fonte: autoria própria.

Na página do Login, há tanto a opção de login para entrar na aplicação, quanto a opção de cadastro para cadastrar um novo usuário

## 6.2 Tela de perfil

A tela de perfil simplifica a interação do usuário com a plataforma trazendo as informações sobre o próprio perfil de usuário, isto é, nome, email, setor e cargo como na imagem abaixo (**Imagem 17**).

Imagem 17 - Meu perfil



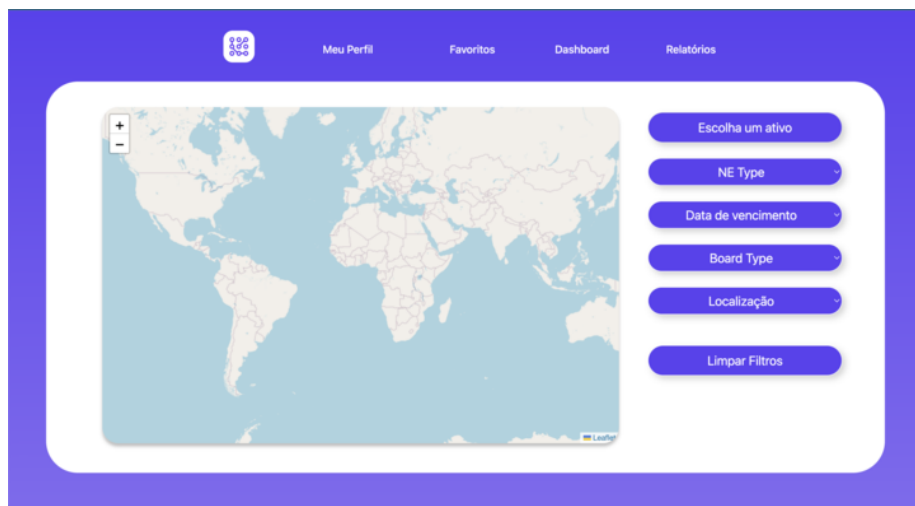
Fonte: autoria própria

Na página do perfil do usuário, será possível notar todas as informações do mesmo, desde Nome, Email, Setor e Cargo relacionados.

## 6.3 Main Page

A tela foi concebida com o propósito de apresentar a localização dos ativos por meio de um gráfico de mapa. O foco principal é permitir que os usuários realizem a seleção de ativos de forma fácil e direta. A interface inclui botões dedicados que facilitam a escolha dos ativos desejados, simplificando a experiência do usuário ao fornecer uma visualização clara e detalhada das localizações associadas a cada ativo no mapa como representado abaixo (**Imagem 18**).

Imagem 18 - Main Page



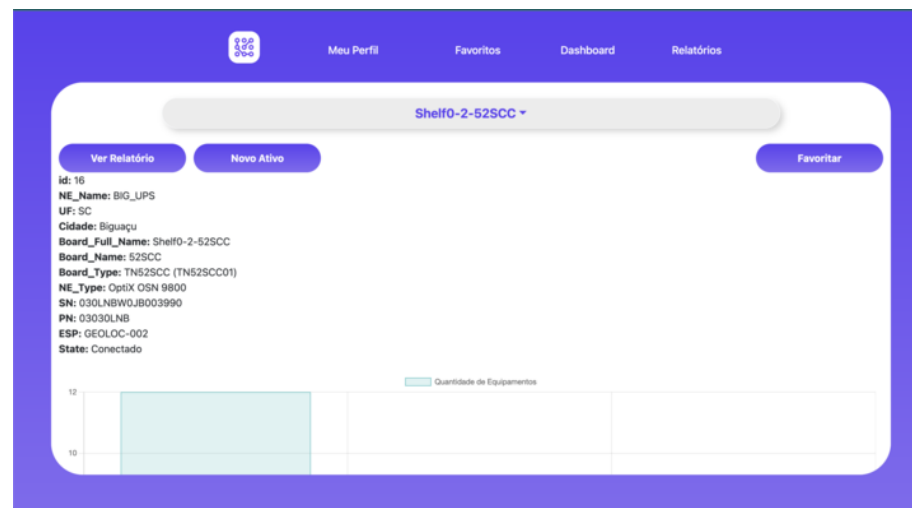
Fonte: autoria própria.

Acima é possível ver a Main Page, na mesma o usuário tem a capacidade de escolher um ativo para mostrar os pontos do mesmo no mapa, selecionar os pontos por tipo NE, data de vencimento, tipo da placa e por fim localização, além de ser capaz de limpar todos os filtros.

## 6.4 Dashboard do ativo selecionado

Partindo da seleção do ativo na página anterior, o usuário conseguirá visualizar as informações na pop-up, como mostra a **imagem 19**.

Imagem 19 - Dashboard



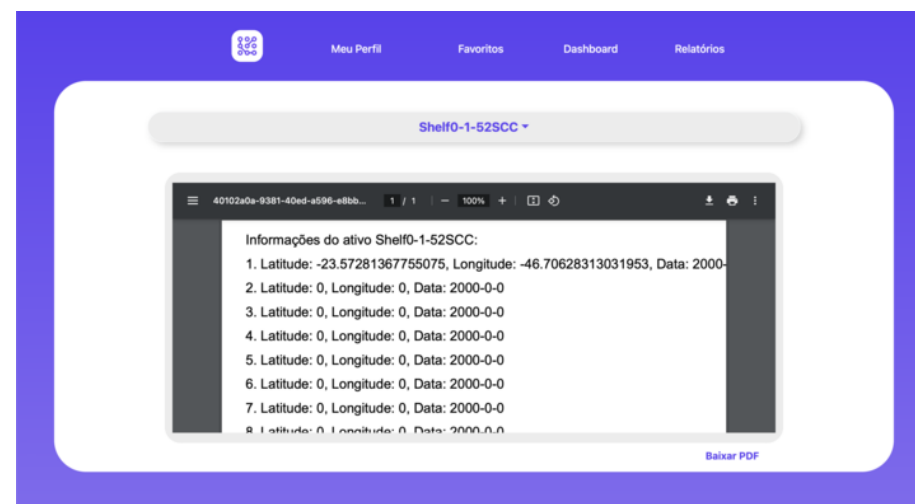
Fonte: autoria própria.

Na tela direcionada a Dashboard, há 4 opções, sendo elas a de escolha do ativo, o mecanismo para ver o relatório de tal ativo, adicionar um novo ativo e favoritar o ativo escolhido. Além disso, na mesma página há 2 gráficos, um que mostra as cidades onde os ativos estão presentes e outro que mostra quantos ativos estão em funcionamento/conectados. Por fim, também é possível ver as informações principais do ativo escolhido.

## 6.5 Página relatórios

A função principal da tela é elaborar um relatório abrangente que contenha todas as informações relevantes, incluindo latitude, longitude e data associadas a um ativo específico. Em seguida, converte esses dados em um documento em formato PDF, proporcionando a praticidade de baixar o arquivo para facilitar o compartilhamento ou arquivamento como mostra a **Imagem 20**.

Imagem 20 - Página do relatório



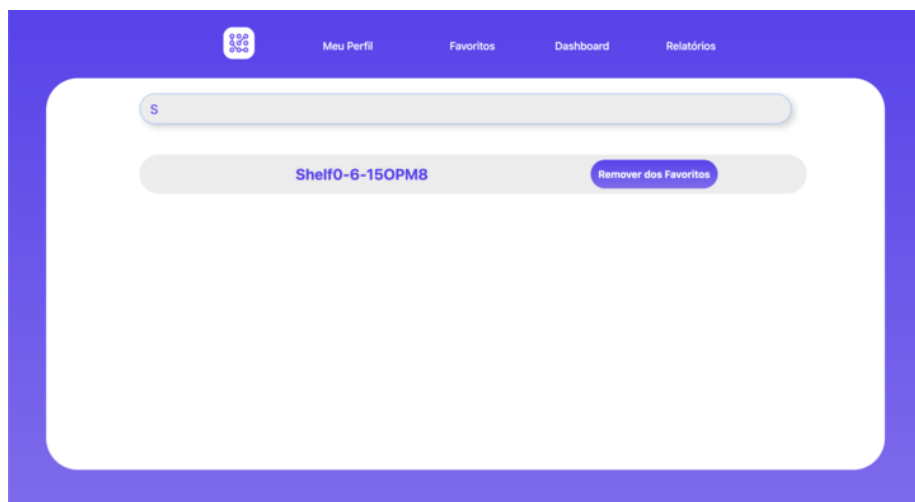
Fonte: autoria própria.

Na página relacionada aos relatórios, há tanto a barra estilo "drop down" para se escolher o ativo, quanto a caixa de PDF com um PDF gerado para o respectivo ativo mostrando as informações de localização do mesmo.

## 6.6 Página favoritos

A principal função da página de favoritos é permitir que um usuário escolha algum ativo e favorite o mesmo afim de que possa voltar em tal ativo através de um modo mais veloz

Imagem 21 - Página dos favoritos



Fonte: autoria própria.

Na imagem acima, é perceptível tanto a presença da barra de pesquisa destinada à investigação dos ativos já marcados como

favoritos quanto a identificação do botão que permite remover um ativo da lista de favoritos assim que ele é localizado.

## 6.7 Landing-Page

A landing page será a primeira página que aparecerá após o login, terá fins apenas de estilismo como na **Imagem 22**.

Imagem 22 - Landing Page



Fonte: autoria própria.

Na landing page será possível ver um pequeno resumo da aplicação web, além de iniciar uma transição sutil entre o login e a main page. Para ir até a main page basta clicar no botão "Localizar ativos" como na **Imagem 22**.

## 6.8 Página de cadastro

Ná página do cadastro será possível cadastrar um novo usuário na plataforma e adicionar os seguintes campos mostrados na figura abaixo (**Imagem 23**)

Imagem 23 - Página do cadastro



The registration form is set against a dark blue background. It features a white rounded rectangle containing a grey user icon on the left. To the right of the icon are five input fields labeled "Email", "Senha", "Nome", "Setor", and "Cargo". A blue "Cadastrar" button is located at the bottom right of the white area.

Fonte: autoria própria.

Na página de cadastro, será possível o usuário inserir todas as informações do mesmo afim de poder estar realizando a autenticação.

## 6.9 Hardware

Na **Imagem 24** abaixo, pode-se observar o hardware desenvolvido pelo grupo Connect4. Este, pretende informar a localização de um ativo dentre os diferentes armazéns, além de seu histórico de transportes. Essas informações são medidas através do módulo GPS presente abaixo, que indica a localização geográfica do protótipo. Além disso, o mesmo apresenta um visor LCD que demonstra se os sinais estão sendo congruientemente absorvidos (sendo estes de Wi-Fi e de GPS). Ademais, existe uma indicação visual desta mesma representação por meio dos LEDs azul e vermelho. Se o LED azul estiver aceso, a conectividade com a internet está indisponível. Se o LED vermelho estiver aceso, a conectividade com o GPS não está sendo possível.

Imagem 24 - Hardware



Fonte: autoria própria.

Acima segue uma imagem que reflete como será o protótipo físico após ser montado completamente em sua caixa de origem.

## 7. Troubleshooting

*Troubleshooting*, em português “solução de problemas”, tem como objetivo listar as possíveis falhas de comportamento da solução proposta pela Connect4 e suas respectivas soluções como representado na tabela (**Tabela 2**).



#	Problema	Possível solução
1	O dispositivo IoT perde a conexão com a rede.	Verificar as configurações de rede do dispositivo, garantir a força do sinal e considerar a implementação de um mecanismo de reconexão no firmware do dispositivo.
2	O dispositivo IoT fica sem bateria.	Implementar um sistema de monitoramento de bateria para fornecer alertas de bateria fraca. Substituir ou recarregar regularmente as baterias com base nos padrões de uso.
3	Componentes físicos do dispositivo IoT apresentam mal funcionamento.	Realizar verificações de hardware de rotina e implementar mecanismos de tratamento de erros. Em caso de falha, substituir ou reparar os componentes com defeito.
4	Os dados enviados do dispositivo para o banco de dados ou aplicativo contém erros	Implementar mecanismos de verificação de erros no protocolo de transmissão de dados. Utilizar checksums ou

		outros métodos para garantir a integridade dos dados.
5	Problemas de conexão com o servidor de banco de dados.	Verificar as configurações de rede, credenciais e disponibilidade do servidor. Implementar mecanismos de tentativa de reconexão para conexões de banco de dados.
6	O aplicativo web não está respondendo ou falha ao carregar.	Verificar o status do servidor web, resolver quaisquer problemas com o ambiente de hospedagem e otimizar o aplicativo para desempenho.
7	Acesso não autorizado ao dispositivo IoT ou banco de dados.	Implementar mecanismos de autenticação e autorização seguros. Atualizar regularmente os protocolos de segurança e aplicar patches para corrigir vulnerabilidades.
8	Os dados de localização recebidos são imprecisos ou desatualizados	Implementar um algoritmo robusto de rastreamento de localização. Utilizar várias fontes para dados de localização e aplicar métodos de correção de erros

9	O sistema tem dificuldades em lidar com um número crescente de dispositivos	Projetar o sistema com escalabilidade em mente. Utilizar soluções baseadas em nuvem que possam escalar dinamicamente com base na demanda
10	Fatores externos, como interferência ou condições climáticas extremas, afetam o desempenho do dispositivo	Usar hardware robusto quando aplicável. Implementar medidas para mitigar interferências ambientais

Tabela 2- Troubleshooting

## 8. Créditos

O grupo Connect4 que desenvolveu a solução GeoLoc, descrita neste manual é composto por:

- [Antonio Guimarães](#)
- [Fernando Machado](#)
- [João Paulo da Silva](#)
- [Matheus Ribeiro](#)
- [Matheus Mendes](#)
- [Olin Costa](#)