

**SERVIÇO NACIONAL DE APRENDIZAGEM INDUSTRIAL**  
**TÉCNICO EM ELETRÔNICA**

GIVALDO DIAS  
GUILHERME HENRIQUE  
ISABELA SCEPANIUK  
KAUÃ BRACCIO  
LUCAS BAREA  
NAILSON SOUSA  
RAFAEL SOUSA  
VICTOR MATOS

**MINIESTAÇÃO METEOROLÓGICA**

JAGUARIÚNA-SP

2022

GIVALDO DIAS  
GUILHERME HENRIQUE  
ISABELA SCEPANIUK  
KAUÃ BRACCIO  
LUCAS BAREA  
NAILSON SOUSA  
RAFAEL SOUSA  
VICTOR MATOS

## **MINIESTAÇÃO METEOROLÓGICA**

Trabalho de Conclusão de Curso  
apresentado ao Curso de Técnico de  
Eletrônica da escola SENAI Jaguariúna.

Orientadores: Carlos Henrique S. Coelho

Rodolfo Favaro Massaro

JAGUARIÚNA-SP

2022

Trabalho de Conclusão de Curso apresentado ao curso de Técnico de Eletrônico da Escola SENAI como requisito parcial para obtenção do grau tecnólogo em eletrônica.

Jaguariúna, \_\_\_\_ de Dezembro de 2022

**BANCA EXAMINADORA:**

---

Ricardo Andreatta Paraguassu

---

Carlos Henrique Santos Coelho

---

Rodolfo Favaro Massaro

JAGUARIÚNA-SP

2022

A Deus, aos nossos familiares,  
amigos e a todos os nossos professores  
que foram fundamentais no  
desenvolvimento deste trabalho,  
especialmente aos professores  
orientadores Carlos Henrique Santos  
Coelho e Rodolfo Favaro Massaro.

## **AGRADECIMENTOS**

Primeiramente a Deus, por permitir nosso sucesso e aos nossos familiares e amigos por todo o apoio.

Aos professores orientadores, Carlos Henrique Santos Coelho e Rodolfo Favaro Massaro, por não desistirem de nossa ideia e por incentivar e instruir em todos os momentos durante a execução do trabalho.

Aos nossos colegas de classe, pelo incentivo e pela troca de experiência.

“A felicidade de sua vida  
depende da qualidade de seus  
pensamentos”.

Marco Aurélio

## **RESUMO**

Esse projeto visa apresentar a construção de uma miniestação meteorológica, a qual será capaz de fornecer o monitoramento das variáveis climáticas, captadas através de sensores, tais como: Sensor de umidade relativa do ar, temperatura, altitude, pressão atmosférica e índice pluviométrico.

As informações coletadas pela miniestação serão transmitidas para um website, onde será possível visualizar os dados, gerando também um banco de dados, a fim de armazenar todas as informações coletadas.

Palavras-chaves: variáveis climáticas, sensores, dados.

## **ABSTRACT**

This project aims to present the construction of a mini meteorological station, which will be able to provide the monitoring of climate variables, captured through sensors, such as: Sensor of relative humidity, temperature, altitude, atmospheric pressure and pluviometric index.

The information collected by the mini station will be transmitted to a website, where it will be possible to view the data, also generating a database, in order to store all the information collected.

Keywords: climate variables, sensors, data.



## LISTA DE ILUSTRAÇÕES

Figura 1 – Placa ESP32 WiFi + Bluetooth.....	24
Figura 2 – Pinagem do ESP32.....	25
Figura 3 – Sensor e Módulo do YL-83.....	26
Figura 4 – Pinagem do Módulo de Controle do YL-83.....	27
Figura 5 – Sensor Temperatura/Umidade DHT11.....	28
Figura 6 – Pinagem do DHT11.....	29
Figura 7 – Sensor BMP280.....	30
Figura 8 – Módulo de bateria Shield 18650 + Cabo USB.....	31
Figura 9 – Partes do Shield V3 Carregador de Bateria 18650.....	31
Figura 10 – Biblioteca thingspeak-arduino.....	33
Figura 11 – Bibliotecas Instaladas.....	34
Figura 12 – Criação do Canal – “Channels”.....	34
Figura 13 – Fields – Criação dos Gráficos.....	35
Figura 14 – Chave de Escrita – Write API Key.....	36
Figura 15 – Chave de Escrita no código.....	36
Figura 16 – Nome e Senha da Rede de Wi-Fi no código.....	36
Figura 17 – Fase de Testes dos componentes.....	37
Figura 18 – Home do Website – Desktop.....	37
Figura 19 – Home do Website – Desktop.....	38
Figura 20 – Home do Website – Desktop.....	38
Figura 21 – QR Code.....	38
Figura 22 – Botões de Direcionamento.....	39
Figura 23 – Caixa de Acrílico Pronta.....	40
Figura 24 – Desenho Técnico da Caixa de Acrílico.....	41
Figura 25 – Desenho Técnico da Caixa de Acrílico (3D).....	41
Figura 26 – Desenho Técnico da Caixa de Acrílico (3D).....	42
Figura 27 – Teste feito com o Sensor de Temperatura.....	42
Figura 28 – Teste feito com o Sensor de Umidade.....	43

Figura 29 – Teste feito com o Sensor de Pressão.....	43
Figura 30 – Teste feito com o Sensor de Altitude.....	44
Figura 31 – Teste feito com o Sensor de Índice Pluviométrico.....	44

## LISTA DE TABELAS

Tabela 1 – Especificações Técnica ESP32.....	25
Tabela 2 – Especificações Técnica YL-83.....	27
Tabela 3 – Especificações Técnica DHT11.....	29
Tabela 4 – Especificações Técnica BMP280.....	30
Tabela 5 – Especificações Técnica Shield 18650.....	32
Tabela 6 – Parte I Cronograma.....	47
Tabela 7 – Parte II Cronograma.....	48
Tabela 8 – Tabela de custos.....	49

## **LISTA DE ABREVIATURAS E SIGLAS**

VCC	Voltagem Corrente Contínua
GND	Ground
IoT	Internet of Things (Internet das Coisas)
VS Code	Visual Studio Code
HTML	Hypertext Markup Language
CSS	Cascading Style Sheet
INMET	Instituto Nacional de Meteorologia

## SÚMARIO

<b>1</b>	<b>INTRODUÇÃO .....</b>	<b>14</b>
<b>2</b>	<b>JUSTIFICATIVA .....</b>	<b>15</b>
<b>3</b>	<b>OBJETIVOS .....</b>	<b>16</b>
<b>3.1</b>	<b>Objetivo Geral .....</b>	<b>16</b>
<b>3.2</b>	<b>Objetivos Específicos .....</b>	<b>16</b>
<b>4</b>	<b>PROCEDIMENTOS METODOLÓGICOS .....</b>	<b>17</b>
<b>4.1</b>	<b>Meteorologia .....</b>	<b>17</b>
<b>4.2</b>	<b>Tempo Atmosférico .....</b>	<b>17</b>
<b>4.3</b>	<b>Temperatura .....</b>	<b>18</b>
<b>4.4</b>	<b>Umidade do Ar .....</b>	<b>18</b>
<b>4.5</b>	<b>Chuva e Índice Pluviométrico .....</b>	<b>18</b>
<b>4.6</b>	<b>Clima Atmosférico .....</b>	<b>19</b>
<b>4.7</b>	<b>Pressão Atmosférica e Altitude .....</b>	<b>19</b>
<b>4.8</b>	<b>Software .....</b>	<b>20</b>
<b>4.8.1</b>	<b>Visual Studio Code .....</b>	<b>20</b>
<b>4.8.2</b>	<b>Arduino IDE .....</b>	<b>20</b>
<b>4.8.3</b>	<b>ThingSpeak .....</b>	<b>21</b>
<b>4.8.4</b>	<b>SolidWorks .....</b>	<b>21</b>
<b>4.9</b>	<b>Linguagens de Programação .....</b>	<b>21</b>
<b>4.9.1</b>	<b>HTML .....</b>	<b>22</b>
<b>4.9.2</b>	<b>CSS .....</b>	<b>22</b>
<b>4.9.3</b>	<b>JavaScript .....</b>	<b>22</b>
<b>5</b>	<b>REVISÃO DE LITERATURA .....</b>	<b>24</b>
<b>5.1</b>	<b>Placa ESP32 - WiFi/Bluetooth .....</b>	<b>24</b>
<b>5.2</b>	<b>Sensor de Chuva - YL-83 .....</b>	<b>26</b>
<b>5.3</b>	<b>Sensor de Temperatura e Umidade – DHT11 .....</b>	<b>28</b>
<b>5.4</b>	<b>Sensor de Pressão Atmosférica e Altitude - BMP280 .....</b>	<b>29</b>
<b>5.5</b>	<b>Módulo de Bateria Shield 18650 .....</b>	<b>31</b>
<b>6</b>	<b>DESENVOLVIMENTO .....</b>	<b>33</b>
<b>6.1</b>	<b>Programação do IDE no ESP32 .....</b>	<b>33</b>
<b>6.2</b>	<b>Envio da Leitura dos Sensores para o ThingSpeak .....</b>	<b>33</b>
<b>6.3</b>	<b>Montagem do Hardware .....</b>	<b>36</b>

<b>6.4</b>	<b>Website .....</b>	<b>37</b>
<b>6.5</b>	<b>Construção do Website .....</b>	<b>39</b>
<b>6.6</b>	<b>Caixa de Acrílico .....</b>	<b>39</b>
<b>6.7</b>	<b>Análise da Coleta de Dados .....</b>	<b>42</b>
<b>7</b>	<b>POSSÍVEIS MELHORIAS .....</b>	<b>45</b>
<b>8</b>	<b>CONCLUSÃO .....</b>	<b>46</b>
<b>9</b>	<b>CRONOGRAMA .....</b>	<b>47</b>
<b>10</b>	<b>ORÇAMENTO .....</b>	<b>49</b>
	<b>REFERÊNCIAS.....</b>	<b>50</b>

## 1 INTRODUÇÃO

As empresas cada vez mais buscam aumentar o seu lucro e reduzir os seus custos, e para que isso se torne possível deve-se planejar estratégias que permitam a evolução constante para a mesma não se torne ultrapassada. Tendo em vista esta colocação, a pesquisa tem como foco importância da implementação de uma miniestação meteorológica de baixo custo dentro da instituição Senai.

A captação de dados meteorológicos é realizada através de estações meteorológicas automáticas ou convencionais, sistemas compostos por diversos sensores e dispositivos capazes de aferir informações relacionadas ao clima e o tempo de uma localidade, porém, essas estações meteorológicas não são encontradas em todas as regiões, deste modo, em muitos casos faz-se necessário utilizar dados de localidades vizinhas quando se deseja aplicar informações agro meteorológicas onde não há disponibilidade dessas informações, fator que diminui a precisão de processos de mensuração de variáveis e fenômenos ligados ao gerenciamento agrícola dependentes de informações meteorológicas.

Sendo assim, este trabalho teve por objetivo desenvolver um sistema de captação e leitura de dados meteorológicos capaz de abranger fatores climáticos ligados especificamente a localidade onde se encontra instalado de forma eletrônica e automática, e com tais dados estimar esses valores de forma precisa.

## **2 JUSTIFICATIVA**

O intuito do projeto é disponibilizar por meio da estação meteorológica o monitoramento das variáveis climáticas, dentro do ambiente escolar.

A partir desses dados será possível acompanhar, por exemplo, a umidade relativa do ar (UR). Tal informação permitirá descobrir se o ambiente precisa que um controle de umidade seja instalado, tendo em vista que circuitos e baterias podem sofrer danificações, ao passarem longos períodos expostos ao vapor de água presente no ar (isso quando a umidade relativa do ar estiver alta). Além disso, aparelhos eletrônicos também podem ter seu funcionamento comprometido. Podendo assim gerar, prejuízos financeiros para a Instituição.

Sendo assim, a implementação da miniestação meteorológica ajudaria a minimizar esses riscos, garantindo a produtividade nas aulas práticas nos laboratórios.



### **3 OBJETIVOS**

#### **3.1 Objetivo Geral**

Implementar uma miniestação meteorológica de baixo custo dentro do ambiente escolar, com o objetivo geral de disponibilizar o monitoramento das variáveis climáticas.

#### **3.2 Objetivos Específicos**

- a) Prototipação da miniestação;
- b) Coleta de dados, a partir da tecnologia Wi-Fi;
- c) Desenvolvimento de um website para a visualização dos dados;

## **4 PROCEDIMENTOS METODOLÓGICOS**

Neste capítulo, será apresentada uma visão geral sobre os conceitos abordados neste trabalho para um melhor entendimento da problemática tratada.

### **4.1 Meteorologia**

O significado de meteorologia define o estudo dos fenômenos atmosféricos, ou seja, é a ciência que estuda a atmosfera terrestre.

Embora seja conhecida por seus aspectos mais tradicionais – a previsão do tempo e a climatologia –, a meteorologia é muito mais complexa e vasta, afinal a própria atmosfera é extensa, variável e detentora de um incrível número de fenômenos.

Os processos que ocorrem na atmosfera terrestre e que são estudados pela meteorologia são os que acontecem na camada mais próxima da superfície que recebe a maior parte das atividades do ser humano e, aliás, é o espaço onde se sentem todos os efeitos que as condições atmosféricas podem exercer na vida das pessoas.

No estudo da atmosfera da Terra e seus fenômenos, entende-se os processos físicos e químicos que determinam as condições atmosféricas que variam em diferentes escalas espaciais e temporais, desde turbulências locais até a circulação oceânica (RAFAELA CORTES, 2022).

### **4.2 Tempo Atmosférico**

O Tempo é o estado momentâneo das condições atmosféricas ou meteorológicas de um dado lugar, em um determinado momento e está sujeito a variações. Quando alguém pergunta: “Como está o tempo hoje?”, pretende saber se está frio ou quente, seco ou úmido, chuvoso ou ensolarado. O tempo é, portanto, a condição atual da atmosfera, que pode mudar de um instante ao outro (BRASILESCOLA, POR RAFAELA SOUSA).

Temos como principais elementos do tempo: A Temperatura, a Umidade, a Chuva e o Vento.

### **4.3 Temperatura**

A temperatura corresponde ao estado térmico do ar atmosférico, ou seja, de frio ou calor. Essa variável meteorológica é medida por estações meteorológicas por meio de sensores que estão expostos diretamente, ou não, aos raios solares.

A temperatura do ar está relacionada aos movimentos de translação e rotação da Terra e também a outros fatores que influenciem a variação de temperatura, como o vento, a umidade relativa do ar, a latitude, a altitude, a proximidade com corpos d'água e as circulações oceânicas e atmosféricas (YNOUE et al., 2017).

### **4.4 Umidade do Ar**

De forma simplificada, a umidade do ar indica a presença de vapor de água na atmosfera, ou seja, H<sub>2</sub>O em estado gasoso. É a partir dela que se define se um clima é seco ou úmido. Também chamada de umidade atmosférica, ela irá influenciar na sensação térmica de cada local, influenciando no calor que sentimos e que nosso corpo consegue aguentar.

Diversos fatores influenciam a umidade do ar: entre os principais, estão as massas de ar. As mais secas, como as continentais, diminuem a umidade, enquanto as mais úmidas, como as oceânicas, a aumentam, o que ajuda na formação de chuvas. Zonas com mais árvores e corpos de água, como rios e lagos, têm maior umidade do ar. Ações humanas como o desmatamento diminuem esse índice.

Já a umidade relativa do ar, como o nome já explicita, mede a quantidade de vapor d'água atmosférica em relação à saturação do vapor disponível no ambiente, ou seja, quanta água em estado gasoso há no ar em relação ao máximo que poderia haver na temperatura atual. Quanto mais perto de 100%, mais água presente na atmosfera (AUGUSTO DALA COSTA, 2022).

### **4.5 Chuva e Índice Pluviométrico**

Chuvas são a forma como a água retorna para a superfície terrestre na sua forma líquida, sendo, portanto, uma importante etapa do ciclo hidrológico. Trata-se de um tipo de precipitação. Elas são ocasionadas pela ascensão do ar quente na

atmosfera, processo que dá origem a nuvens e precipitações. De acordo com a maneira como se formam, as chuvas são classificadas em convectiva, frontal ou orográfica. A medição do seu volume é realizada com o auxílio de um pluviômetro (BRASILESCOLA, POR PALOMA GUITARRARA).

Para poder acompanhar a quantidade de chuvas numa determinada região, os pesquisadores climáticos criaram o índice pluviométrico (medido em milímetros). Este é calculado da seguinte forma: as estações meteorológicas marcam um espaço no terreno de uma determinada região. Medem e acompanham a quantidade de chuva que cai ali durante o ano. Este índice é uma boa referência para se conhecer o clima de uma região (SUAPESQUISA, 2022).

#### **4.6 Clima Atmosférico**

O Clima é uma condição duradoura do ambiente atmosférico e equivale ao conjunto dos tipos de tempos mais comuns em um determinado lugar ao longo de um período de aproximadamente 30 anos. Representa, portanto, um padrão geral das condições meteorológicas (variações anuais de temperatura, umidade, pressão do atmosférica e ventos), que se alteram de acordo com as estações do ano (BRASILESCOLA, POR RAFAELA SOUSA).

Temos como principais elementos do tempo: Radiação, Pressão Atmosférica e entre outros. E como principais fatores do clima, temos: Latitude, Altitude, Correntes Marítimas e entre outros.

#### **4.7 Pressão Atmosférica e Altitude**

Pressão atmosférica é a força exercida pela massa de gases da atmosfera sobre uma superfície qualquer. Seu valor não é constante, podendo variar de acordo com a altitude, além de influenciar na temperatura de um determinado local.

Ao compararmos a atmosfera a um fluido, compreendemos que a pressão exercida sobre os corpos presentes na atmosfera depende da profundidade em que eles se encontram. Nesse sentido, a maior pressão atmosférica é encontrada em áreas com alturas iguais ou menores do que o nível do mar.

Em outras palavras, quanto maior a altitude, menor a pressão atmosférica. Isso acontece porque a força da gravidade influencia mantendo a maior parte do ar

próxima à superfície. Nesse sentido, pode-se dizer que regiões mais altas possuem o ar rarefeito, com menos densidade.

O aparelho usado para medição é o barômetro e foi inventado pelo físico e matemático Evangelista Torricelli, em 1643. O instrumento é semelhante a um termômetro, entretanto possui finalidade diferente. A unidade usada para medir a pressão neste instrumento é conhecida por bar.

## **4.8 Software**

Nessa seção serão apresentados os softwares que foram utilizados no desenvolvimento do projeto.

### **4.8.1 Visual Studio Code**

O Visual Studio Code é software editor de código-fonte criado pela Microsoft com o objetivo de auxiliar programadores na criação de códigos de softwares. Logo, ele é muito utilizado pelos desenvolvedores para escrever, editar e gerenciar os códigos que estão desenvolvendo, especialmente nas fases de codificação e de testes.

O VS Code está disponível para Windows, Mac e Linux, os quais são os principais sistemas operacionais utilizados (REDAÇÃO XP EDUCAÇÃO, 2022).

### **4.8.2 Arduino IDE**

Arduino IDE é o software Arduino gratuito que facilita o desenvolvimento e a gravação de códigos diretamente no microcontrolador. Através deste, é possível realizar o Upload dos códigos para a placa tanto em sistemas operacionais Windows quanto Linux, demonstrando sua funcionalidade e versatilidade.

Além de uma compatibilidade com quase todos os sistemas operacionais, o Arduino IDE torna possível a programação de todos os modelos de placas Arduino e sempre que o projeto apresentar algum problema em sua configuração ou até mesmo em seu código, o mesmo irá gerar uma notificação e elencar onde pode estar o problema.

Atualmente existem dois métodos disponíveis para o Upload de códigos Arduino, seja via Arduino IDE, software instalado junto ao computador, quanto via navegador através de uma plataforma online recentemente desenvolvida pela Empresa Arduino (MATHEUS GEBERT, 2019)

#### **4.8.3 ThingSpeak**

O ThingSpeak é uma plataforma IoT que permite, sem custo algum, o upload de dados numéricos, os quais serão plotados ao longo do tempo na forma de gráficos. Ou seja, se você deseja monitorar qualquer grandeza numérica (umidade, temperatura, pressão, luminosidade etc.) à distância via Internet, o ThingSpeak é uma das plataformas IoT mais indicadas.

O ThingSpeak possui tanto planos grátis quanto pagos (comerciais). Além de menores limitações, os planos comerciais oferecem suporte a uso do MATLAB, algo que pode ser interessante dependendo da complexidade do processamento de dados numéricos que você precisa na plataforma IoT (FILIPEFLOP, 2020).

#### **4.8.4 SolidWorks**

O SolidWorks é um software de desenho de CAD 3D para modelagem de peças e montagens em desenhos 3D e 2D. Oferece uma ampla gama de soluções para cobrir os aspectos envolvidos no processo de desenvolvimento do produto, permitindo criar, projetar, simular, fabricar, publicar e gerenciar os dados do processo de design.

Sua função é muito específica. Permite acelerar o processo de desenvolvimento de produtos, economizando tempo e custos e dando lugar à inovação de produtos (DATECH, [20--]).

### **4.9 Linguagens de Programação**

Nessa seção serão apresentadas as linguagens de programação que foram utilizadas no desenvolvimento da criação do website do projeto.

### **4.9.1 HTML**

Sigla para HyperText Markup Language (Linguagem de Marcação de Hipertexto), o HTML é o componente base da web. Isso quer dizer que ele permite a construção de websites e a inserção de novos conteúdos, como imagens e vídeos, por meio dos hipertextos.

Os hipertextos são a junção de vários elementos — palavras, vídeos e conteúdos —, que, quando conectados, estabelecem uma rede de dados que permite a comunicação, o armazenamento e o compartilhamento de informações.

Por exemplo, ao visitar um website, você encontra diversas informações com formatações diferentes, como parágrafos, bullets e fontes distintas. Pois então, essa estruturação é realizada por meio do HTML (EQUIPE TOTVS, 2020).

### **4.9.2 CSS**

CSS é chamado de linguagem Cascading Style Sheet (Folha de Estilo em Cascatas) e é usado para estilizar elementos escritos em uma linguagem de marcação como HTML. O CSS separa o conteúdo da representação visual do site. Pense na decoração da sua página. Utilizando o CSS é possível alterar a cor do texto e do fundo, fonte e espaçamento entre parágrafos. Também pode criar tabelas, usar variações de layouts, ajustar imagens para suas respectivas telas e assim por diante (ARIANE G., 2022).

### **4.9.3 JavaScript**

JavaScript é uma linguagem de programação de alto nível criada, a princípio, para ser executada em navegadores e manipular comportamentos de páginas web.

Segundo a Mozilla Foundation, atual nome da antiga Netscape Communications Corporations, empresa responsável pela criação do JS, "JavaScript é uma linguagem de programação, leve, interpretada, orientada a objetos, baseada em protótipos e em first-class functions (funções de primeira classe), mais conhecida como a linguagem de script da Internet."

Com seus scripts é possível incluir, em uma página estática, elementos dinâmicos como mapas, formulários, operações numéricas, animações, infográficos interativos e muito mais (UGO ROVENA, 2021).



## 5 REVISÃO DE LITERATURA

Neste capítulo, será apresentada uma visão geral sobre os principais componentes e dispositivos eletrônicos que foram utilizados neste trabalho para um melhor entendimento do desenvolvimento do projeto.

### 5.1 Placa ESP32 - WiFi/Bluetooth

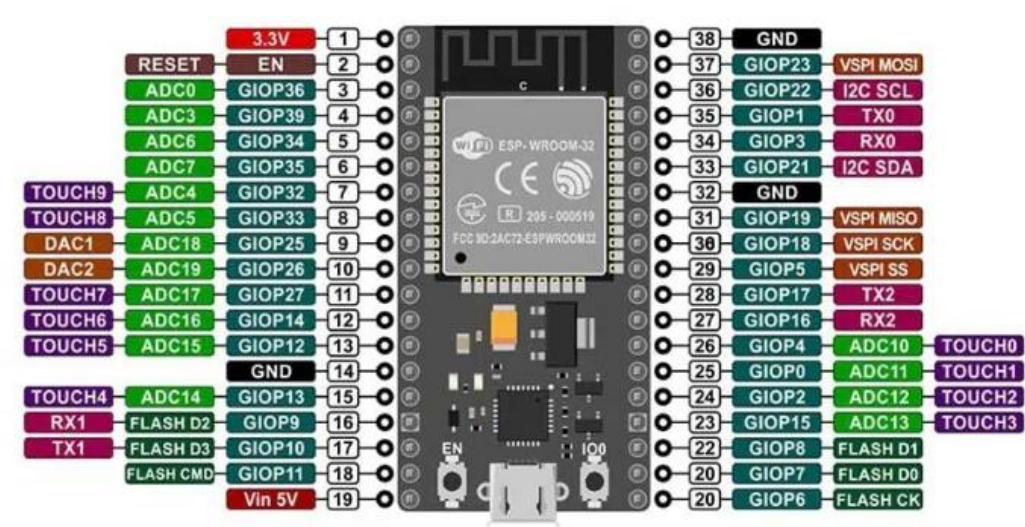
O módulo ESP32 é um dispositivo IoT (Internet das Coisas). Ele é constituído por um microprocessador dual core Tensilica Xtensa de 32 bits com suporte à rede Wi-Fi 802.11 b/g/n e bluetooth versão 4.2, e com a memória flash integrada (MINATEL,2017). O ESP32, possui 32 pinos GPIO (general-purpose input/output), ou seja, portas programáveis de entrada e saída de dados que são usadas para dispor uma interface entre os periféricos e os microprocessadores. Dentro os pinos GPIO, o ESP32 possui alguns pinos tipos diferentes, dentre estes os pinos somente digitais, os habilitados para toque capacitivo e entre outros. Os pinos GPIOs habilitados para toque capacitivo e GPIOs analógicos podem ser configurados como GPIOs digitais (ESPRESSIF, 2019).

Figura 1: Placa ESP32 WiFi + Bluetooth



Fonte: ELETROGATE

Figura 2: Pinagem do ESP32



Fonte: ELETROGATE

Tabela 1: Especificações Técnicas do ESP32

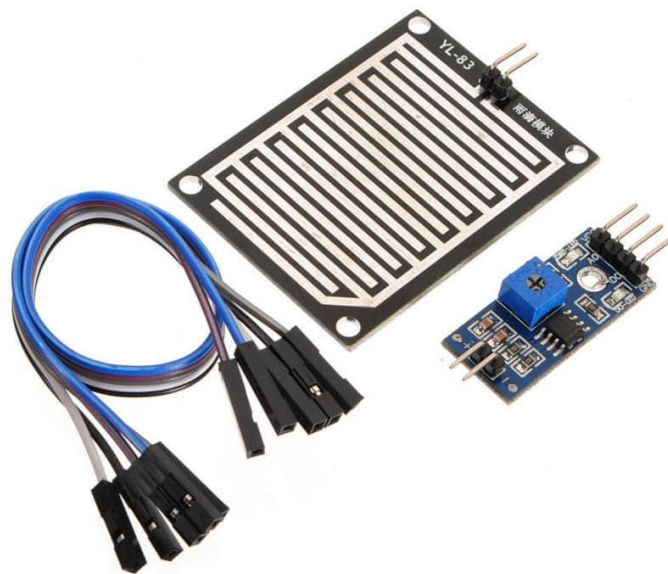
ESPECIFICAÇÕES	
CPU	Xtensa® Dual-Core 32-bit LX6
ROM	448 KBytes
RAM	520 KBytes
FLASH	4 MB
Clock máximo	240MHz
Wireless padrão	802.11 b/g/n
Conexão Wifi	2.4Ghz (máximo de 150 Mbps)
Portas GPIO (Entradas e Sidas)	32
Tensão de operação	4,5 ~ 9V
Taxa de transferência	110-460800bps
Dimensões	51 mm x 27,5 mm x 5 mm

Fonte: ELETROGATE

## 5.2 Sensor de Chuva - YL-83

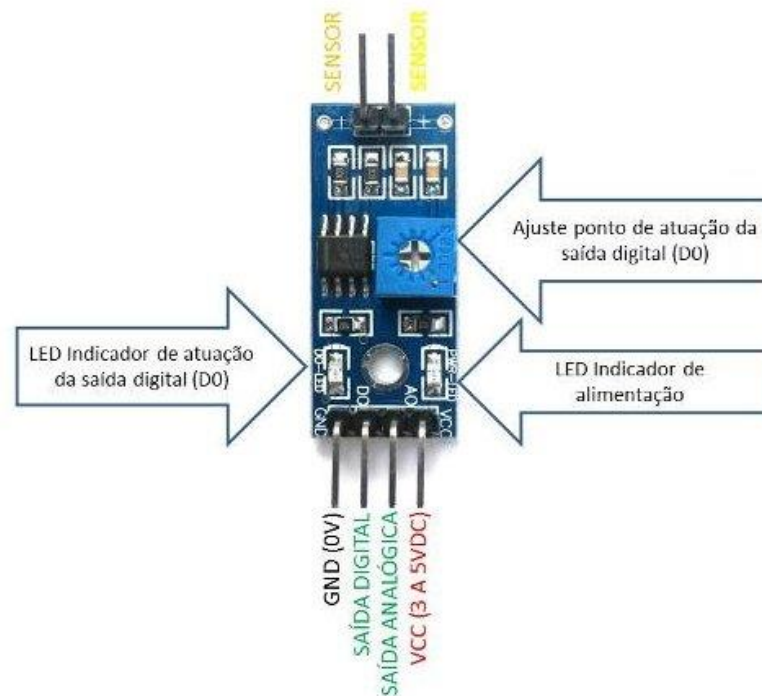
Este sensor pode ser usado para monitorar uma variedade de condições climáticas, como gotas de chuva ou de neve. O sensor verifica se há ou não presença de chuva. O limite entre tempo seco e chuva pode ser ajustado através do potenciômetro presente no sensor, que regulará a saída digital D0. O módulo de controle, ilustrado na Figura 4, tem 2 pinos que vão se comunicar com a placa do sensor, e na outra extremidade, 4 pinos de sinal e alimentação: A0 (sinal analógico), D0 (sinal digital), GND (terra) e Vcc (alimentação). A alimentação vai de 3,3 à 5 Volts (ADILSON THOMSEN, 2014).

Figura 3: Sensor e Módulo do YL-83



Fonte: FILIPEFLOP (2014)

Figura 4: Pinagem do Módulo de Controle do YL-83



Fonte: MARINOSTORE

Tabela 2: Especificações Técnicas do YL-83

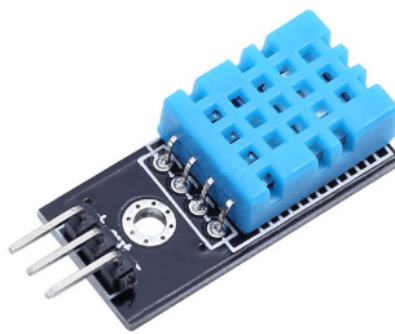
ESPECIFICAÇÕES	
Tensão de funcionamento	3.3V – 5V
Capacidade de condução do sensor	100mA
Extensão dos jumpers	20,8cm
Dimensões do sensor (CxLxE)	55x40x1,5mm
Dimensões do módulo (CxLxE)	32x14x6mm
Saída	Analógica e Digital

Fonte: ARDUINO (2018)

### 5.3 Sensor de Temperatura e Umidade – DHT11

O DHT11 é um sensor de baixo custo capaz de medir a temperatura e a umidade do ambiente. Este componente é constituído de duas partes principais: um sensor de umidade capacitivo e um sensor de temperatura termistor NTC, isto é um resistor sensível à variação de temperatura.

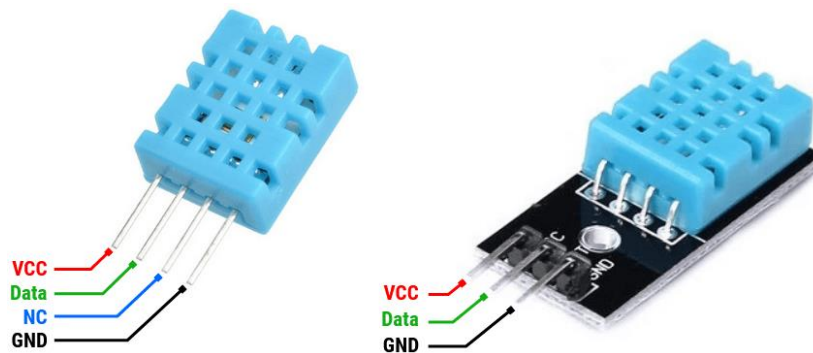
Figura 5: Módulo DHT11



Fonte: MASTERWALKER

A Figura 6 demonstra a estrutura do sensor DHT11, que possui 4 terminais. O primeiro (VCC) é o terminal de alimentação, que conforme datasheet do fabricante pode ser entre 3V e 5,5V. O segundo (DATA) é o pino de comunicação de dados, será através dele que o valor da temperatura e umidade serão comunicados a placa microcontroladora. O terceiro (NC) é um pino vazio. Por fim, o último (GND) é o terminal de terra do sensor.

Figura 6: Pinagem do DHT11



Fonte: CIRCUITGEEKS (2021)

Tabela 3: Especificações Técnicas do DHT11

ESPECIFICAÇÕES	
Faixa de medição de umidade	20 a 90% UR
Faixa de medição de temperatura	0° a 50°C
Interfaces de comunicação	3-5VDC (5,5VDC máximo)
Corrente	200uA a 500mA, em stand by de 100uA a 150 uA
Precisão de umidade de medição	± 5,0% UR
Precisão de umidade de temperatura	± 2.0 °C
Dimensões (incluindo os terminais)	23 x 12 x 5mm

Fonte: VIDADESILÍCIO

#### 5.4 Sensor de Pressão Atmosférica e Altitude - BMP280

O sensor BMP280 é um sensor desenvolvido pela Bosch, sendo este um dos modelos de uma família (GZECAM, 2017).

O BMP280 é um sucessor do BMP180, com ganhos em termos de precisão e consumo de energia além do tamanho 63% menor, tornando comum o seu uso em dispositivos móveis e portáteis.

O módulo BMP280 funciona com interfaces I2C ou SPI e tensão de 3V, sendo que o baixo consumo de energia permite o funcionamento por longos períodos com alimentação por bateria, e é indicado para projetos como drones, estações meteorológicas, dispositivos com GPS, relógios, etc. (FILIPEFLOP, 2022).

Figura 7: Sensor BMP280



Fonte: MULTILÓGICA-SHOP

Tabela 4: Especificações Técnica do BMP280

ESPECIFICAÇÕES	
Tensão de operação	3V
Corrente consumida	2.7µA
Interfaces de comunicação	I2C e SPI
Faixa de medição para pressão	300 - 1100hPa (equiv. +9000 à -500m acima/abaixo do nível do mar) com precisão de ±0.12hPa
Dimensões (sem os pinos)	15 x 12 x 2,3 mm

Fonte: FILIPEFLOP



## 5.5 Módulo de Bateria Shield 18650

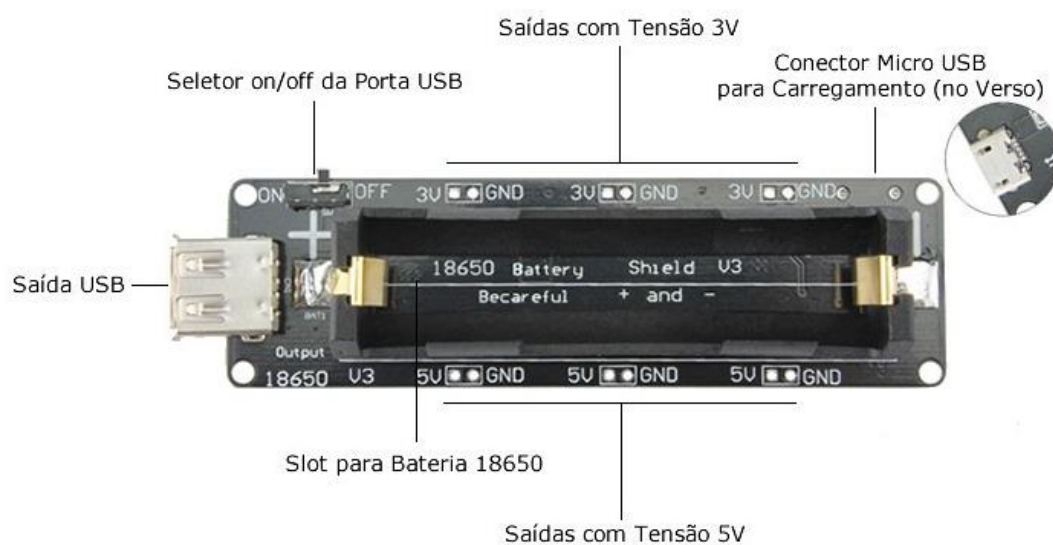
O Shield V3 Carregador de Bateria 18650 é um dispositivo que tem como finalidade comportar uma bateria 18650 para alimentação de projetos eletrônicos. Podendo, ser utilizado para alimentar o ESP8266, ESP32, Arduino, Raspberry Pi, sensores e módulos, com capacidade de corrente de até 2A, o Shield possui 3 interfaces de saída 3V/1A, e 3 interfaces de saída 5V/2<sup>a</sup>, além disso, possui um slot que permite um rápido e firme encaixe (USINAINFO, 2022).

Figura 8: Módulo de bateria Shield 18650 + Cabo USB



Fonte: SMARTKITS

Figura 9: Partes do Shield V3 Carregador de Bateria 18650



Fonte: USINAINFO



Tabela 5: Especificações Técnica do Módulo de Bateria – Shield 18650

<b>ESPECIFICAÇÕES</b>	
<b>Proteção</b>	Contra sobrecarga e subcarga
<b>Conector</b>	Micro USB
<b>Corrente de carregamento</b>	0.5A
<b>Chave de controle para saída USB</b>	1
<b>Tensão de entrada</b>	5 - 8V
<b>Três interfaces de saída</b>	3V/1A
<b>Três interfaces de saída</b>	5V/2A
<b>Led indicador de carga</b>	Verde para carga total/vermelho para descarregado
<b>Composição</b>	Placa de Fenolite e componentes eletrônicos
<b>Dimensões</b>	98mm x 29mm x 10mm
<b>Peso</b>	30g

Fonte: ELETROGATE

## 6 DESENVOLVIMENTO

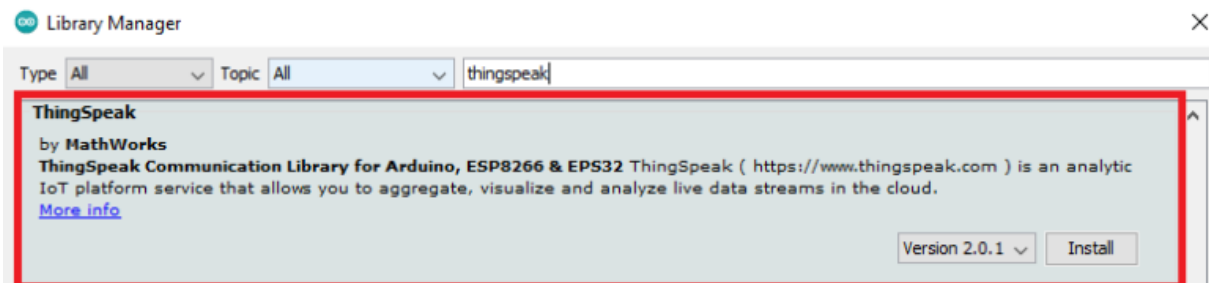
### 6.1 Programação do IDE no ESP32

Com a programação pronta no Intergrated Development Environment (IDE) do Arduino conforme o anexo, a próxima etapa se passa pela aplicação do comando verify (verificar), que realiza uma checagem de toda a programação, não contendo nenhum erro na programação, segue-se para o botão upload para ser encaminhada ao microcontrolador ESP32. Para mais informações da programação, é possível visualizá-la na integra no anexo (MCROBERTS, 2011).

### 6.2 Envio da Leitura dos Sensores para o ThingSpeak

O primeiro passo foi instalar a biblioteca do ThingSpeak no Arduino IDE, para que assim, fosse possível enviar as leituras dos sensores para o ThingSpeak. Usou-se a biblioteca thingspeak-arduino, conforme ilustrado na figura 10.

Figura 10: Biblioteca thingspeak-arduino



Fonte: RANDOMNERD

Em seguida, utilizando o gerenciador de bibliotecas do Arduino IDE, foram instaladas as bibliotecas para se comunicar com os sensores utilizados no projeto, conforme mostra a figura 11. Na sequência foi desenvolvida o restante da programação, essas informações podem ser visualizadas no anexo.

Figura 11: Bibliotecas instaladas

```

1 //Bibliotecas do Projeto.
2
3 #include <WiFi.h> //Inclui Biblioteca para utilizar o WiFi.
4 #include <DHT.h> //Inclui Biblioteca para utilizar o sensor DHT11.
5 #include <Wire.h> //Inclui Biblioteca para comunicar com dispositivos I2C.
6 #include <SPI.h> //Inclui Biblioteca usada para o microcontrolador para se comuni
7 #include <Adafruit_BMP280.h> //Inclui Biblioteca para utilizar o sensor BMP280.
8 #include <Adafruit_Sensor.h> //Inclui Biblioteca de sensores Adafruit.

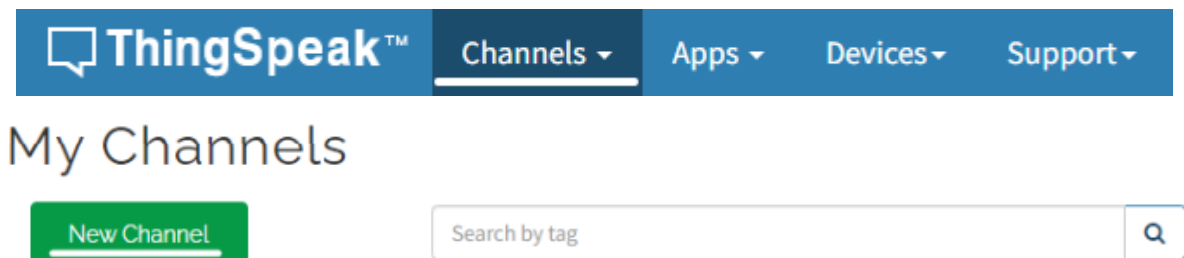
```

Fonte: Próprio Autor

Após realizar o cadastro na plataforma do ThingSpeak, crio-se um canal (Channels), o qual é uma espécie de espaço para o projeto dentro do ThingSpeak, onde contém os gráficos e dados gerados.

Dentro do canal, foram adicionadas as informações referentes ao projeto. Como, por exemplo, os “Fields”. Cada Field equivale a um gráfico diferente, no caso desse projeto foram adicionados cinco gráficos. Conforme apresentado na figura 13.

Figura 12: Criação do Canal – “Channels”



Fonte: Próprio Autor

Figura 13: Fields – Criação dos Gráficos

## New Channel

<b>Name</b>	<input type="text" value="Miniestação Meteorológica"/>	
<b>Description</b>	<input type="text" value="Projeto Conclusão de Curso SENAI JAGUARIUNA Miniestação Meteorológica com o objetivo de"/>	
<b>Field 1</b>	<input type="text" value="Temperatura"/>	<input checked="" type="checkbox"/>
<b>Field 2</b>	<input type="text" value="Umidade Relativa do A"/>	<input checked="" type="checkbox"/>
<b>Field 3</b>	<input type="text" value="Pressão Atmosférica"/>	<input checked="" type="checkbox"/>
<b>Field 4</b>	<input type="text" value="Altitude"/>	<input checked="" type="checkbox"/>
<b>Field 5</b>	<input type="text" value="Índice Pluviométrico"/>	<input checked="" type="checkbox"/>
<b>Field 6</b>	<input type="text" value=""/>	<input type="checkbox"/>

Fonte: Próprio Autor

Para conseguir enviar os dados coletados para o canal criado, foi preciso obter a chave de escrita (Write API Key). Sem ela, não é possível enviar os dados.

Após esse processo, a chave de escrita foi adicionada na programação do código no Arduino IDE, na linha "String chave\_escrita\_thingspeak = " " ; ". Conforme mostra a figura 15.

Também foi preciso adicionar o nome e a senha da rede de wi-fi que foram utilizadas, nas seguintes linhas "#define SSID\_REDE " " e #define SENHA\_REDE " ". Conforme ilustrado na figura 16.

Figura 14: Chave de Escrita

Write API Key

Key

[Generate New Write API Key](#)

Fonte: Próprio Autor

Figura 15: Chave de Escrita no Código

```
/* constantes e variáveis globais */
char endereco_api_thingspeak[] = "api.thingspeak.com";
String chave_escrita_thingspeak = "G5WQT38KXR1ZJLCU";
unsigned long last_connection_time;
```

Fonte: Próprio Autor

Figura 16: Nome e senha da rede Wi-Fi no Código

```
/* defines - wi-fi */
#define SSID_REDE "Galaxy A2087B2" /* nome da rede que se deseja conectar */
#define SENHA_REDE "78787878" /* senha da rede que se deseja conectar */
#define INTERVALO_ENVIO_THINGSPEAK 30000 /* intervalo entre envios de dados ao ThingSpeak (em ms) */
```

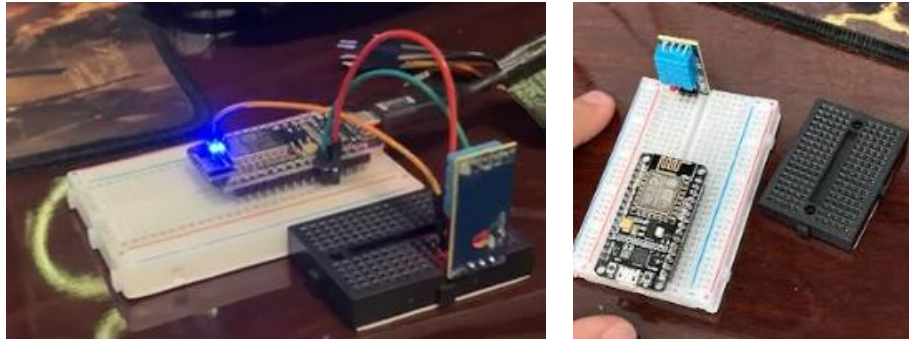
Fonte: Próprio Autor

### 6.3 Montagem do Hardware

Primeiramente foi realizada uma etapa de testes nos sensores, junto a programação, a fim de verificar se os componentes estavam funcionando adequadamente, conforme o esperado.

Logo, após isso, foi desenvolvida a comunicação de todos os componentes.

Figura 17: Testes dos componentes



Fonte: Próprio Autor

## 6.4 Website

Com o intuito de transformar o acesso aos dados coletados prática e dinâmica, foi desenvolvido um website, o qual é compatível tanto para a versão desktop quanto para a versão mobile, conforme apresentado nas figuras 18, 19 e 20. Na aba “Home” do website há um botão “Clique Aqui”, o qual direciona o usuário para o ThingSpeak, software onde estão armazenadas todas as informações coletadas pela miniestação meteorológica no formato de gráfico em tempo real.

Para ser direcionado ao website, o usuário deve apontar a câmera traseira do celular para o QR Code localizado na parte superior do protótipo. Conforme ilustrado na figura 21.

Figura 18: Home do Website – Desktop



Fonte: Próprio Autor

Figura 19: Home do Website – Desktop



Fonte: Próprio Autor

Figura 20: Home do Website – Desktop



Fonte: Próprio Autor

Figura 21: QR Code



Fonte: Próprio Autor

## 6.5 Construção do Website

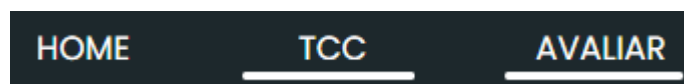
O software VS Code foi utilizado para realizar a construção de toda a programação, a qual foi desenvolvida a partir das linguagens de programação, HTML, CSS e JavaScript.

Como o projeto foi desenvolvido para atuar como um recurso educacional, o usuário terá a opção de visualizar a parte escrita do projeto, nesse caso o TCC em PDF. Para que futuros alunos possam se inspirar no projeto.

Além disso, também no website, os usuários poderão deixar suas avaliações sobre o projeto elaborado.

Essas informações estão disponíveis na Landing Page do website, conforme mostra a figura 22.

Figura 22: Botões de direcionamento



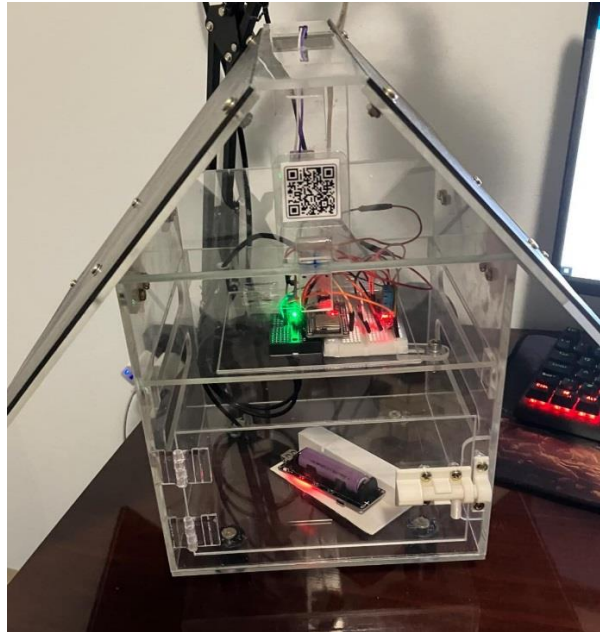
Fonte: Próprio Autor

## 6.6 Caixa de Acrílico

Com o propósito de gerar um compartimento resistente para armazenar os componentes de hardware, optou-se por construir uma caixa de acrílico. Conforme ilustrado na figura 23.



Figura 23: Caixa de Acrílico Pronta – com os componentes



Fonte: Próprio Autor

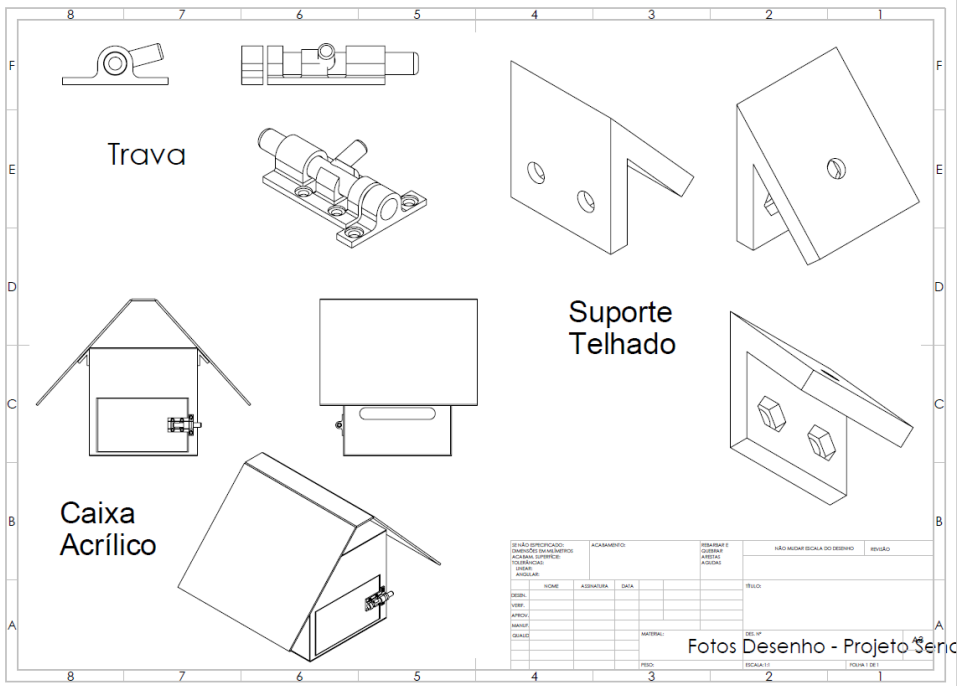
#### 6.6.1 Construção da Caixa de Acrílico

Para a construção da caixa de acrílico utilizou-se o software SolidWorks, onde foi possível gerar, projetar e simular o projeto da caixa em 3D, conforme mostra a figura 24.

A caixa de acrílico possui alguns vazamentos laterais estratégicos, os quais asseguram o bom funcionamento dos sensores, garantindo que os dados coletados sejam precisos e confiáveis.

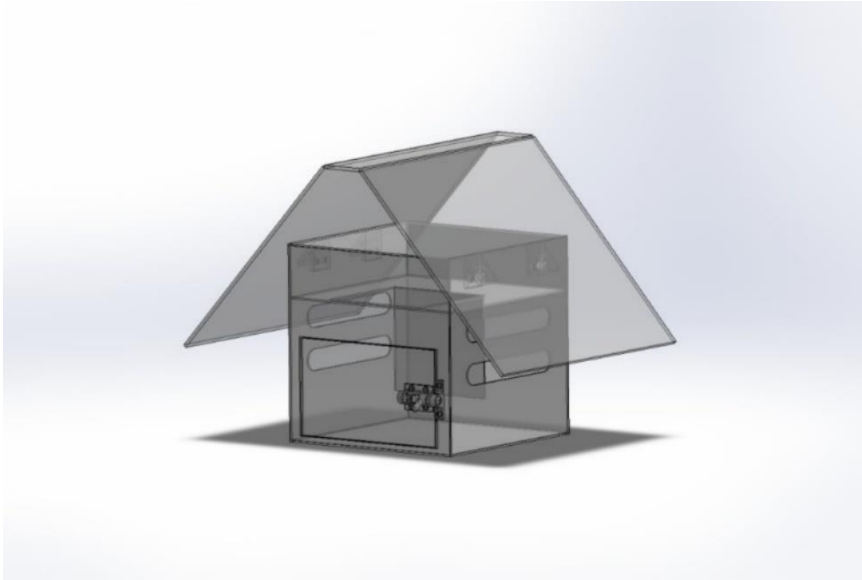
Isso porque, caso não houvesse esses vazamentos, poderia haver um aquecimento no interior da caixa, o que levaria aos sensores a capturarem dados climáticos irregulares.

Figura 24: Desenho Técnico da Caixa de Acrílico



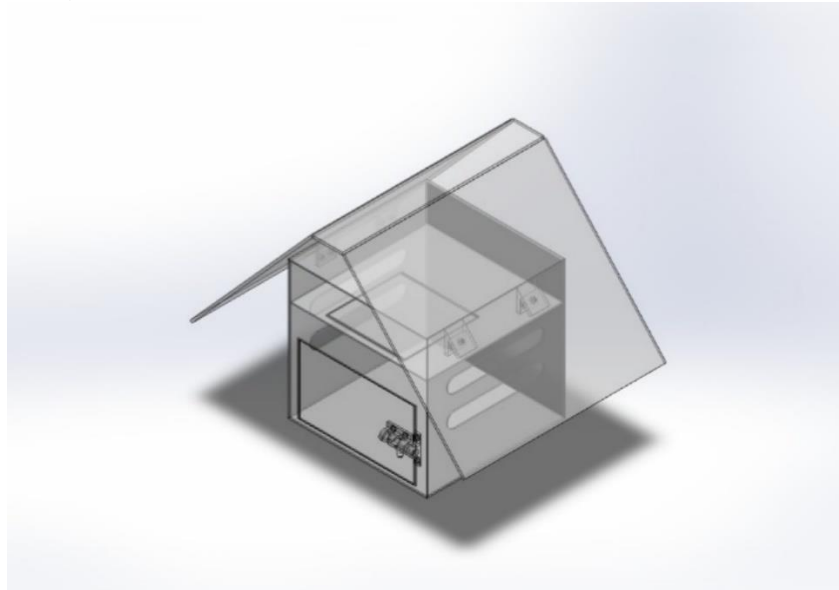
Fonte: Próprio Autor

Figura 25: Desenho Técnico da Caixa de Acrílico em 3D



Fonte: Próprio Autor

Figura 26: Desenho Técnico da Caixa de Acrílico em 3D

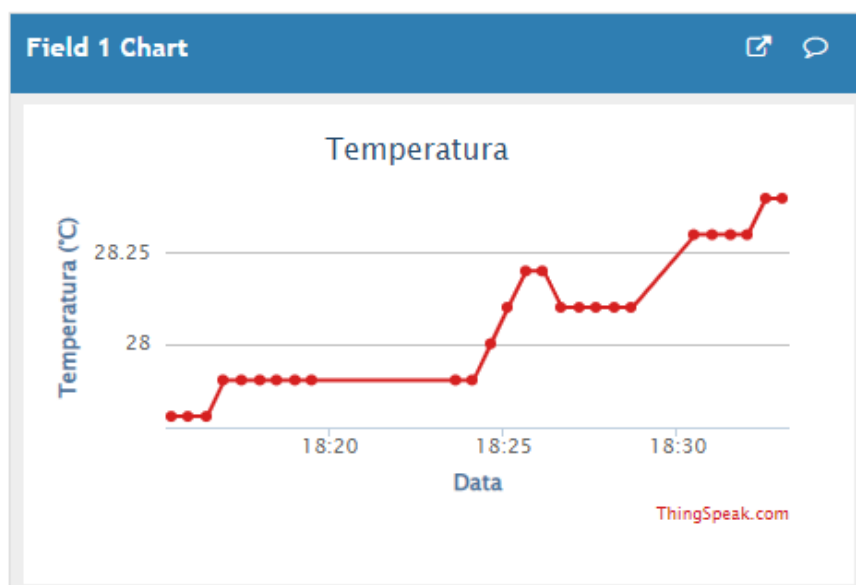


Fonte: Próprio Autor

## 6.7 Análise da Coleta de Dados

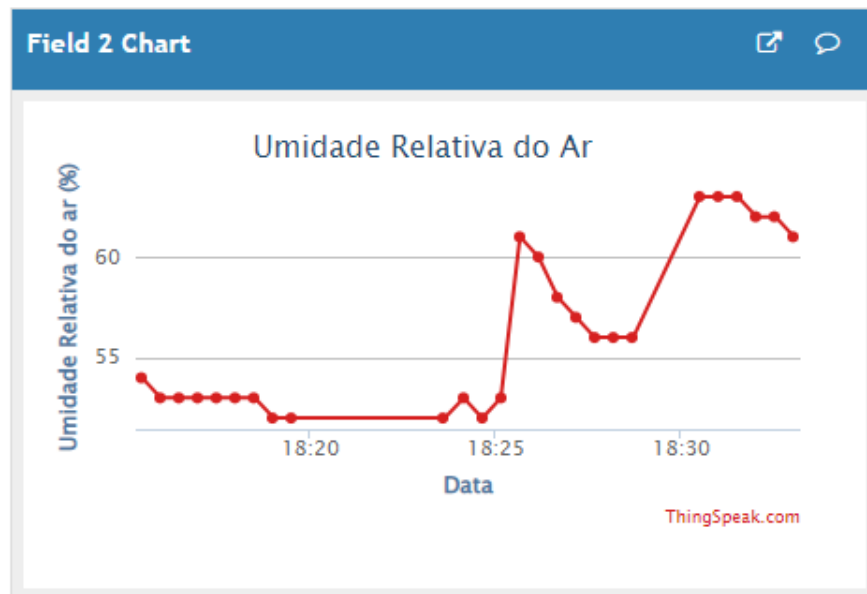
As informações coletadas pela miniestação meteorológica foram comparadas com os dados disponíveis em tempo real pelo INMET (Instituto Nacional de Meteorologia). Onde foi verificado se havia ou não precisão nos dados obtidos pelos sensores da miniestação meteorológica.

Figura 27: Teste feito com o Sensor de Temperatura



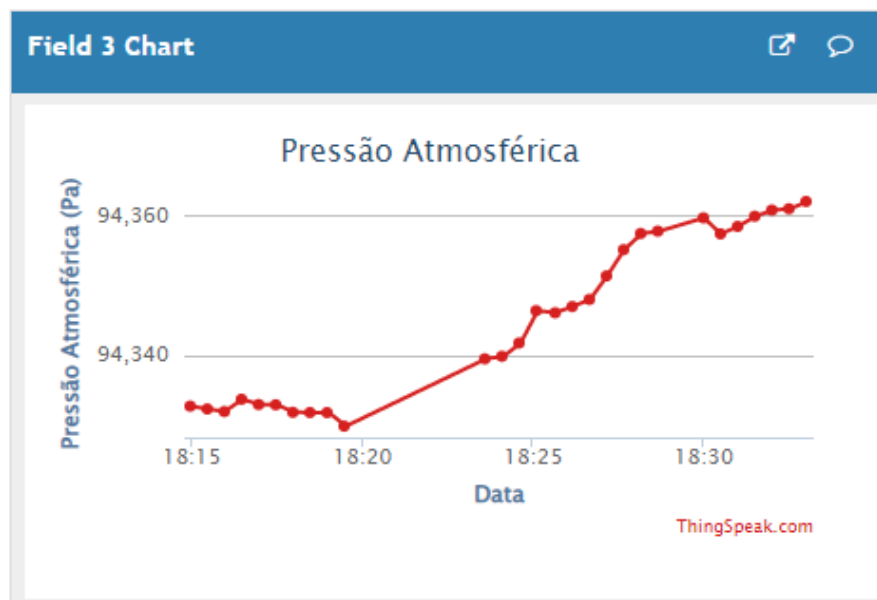
Fonte: Próprio Autor

Figura 28: Teste feito com o Sensor de Umidade



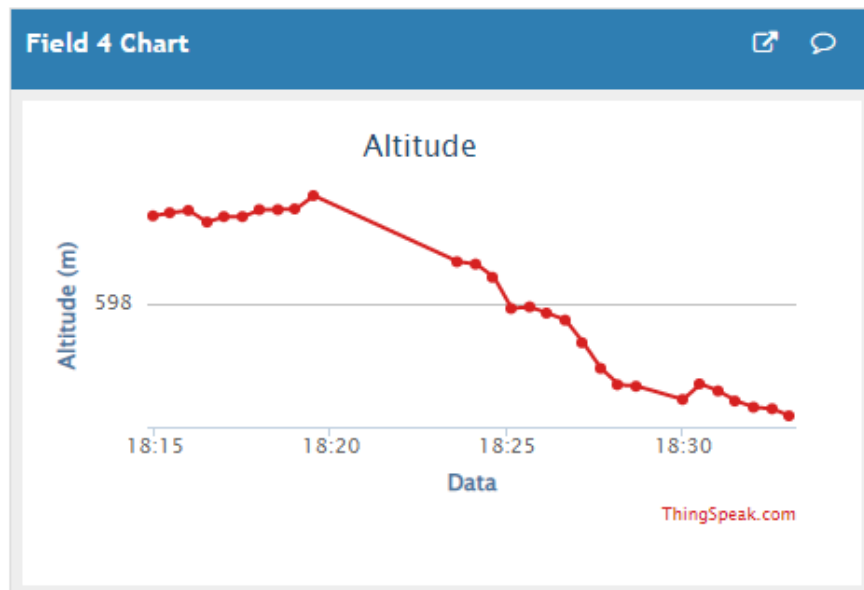
Fonte: Próprio Autor

Figura 29: Teste feito com o Sensor de Pressão



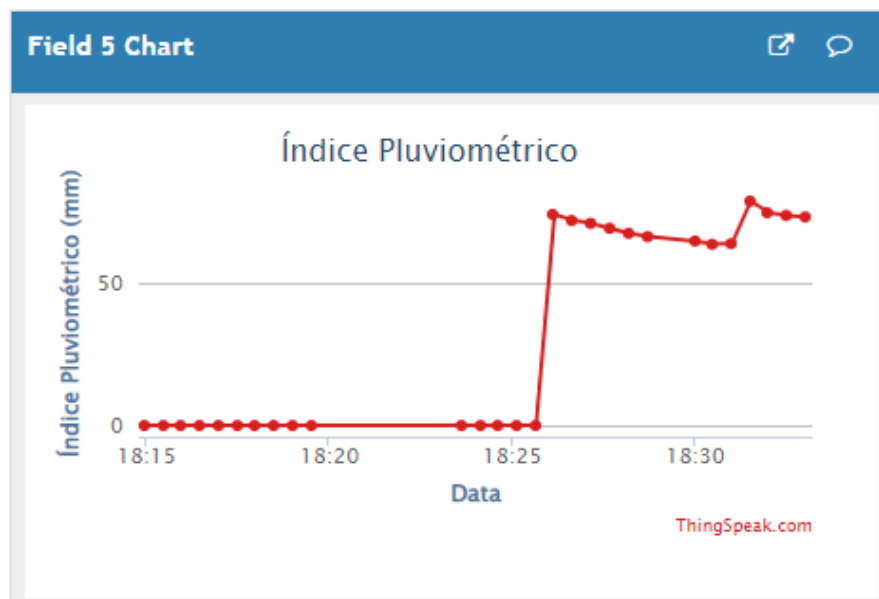
Fonte: Próprio Autor

Figura 30: Teste feito com o Sensor de Altitude



Fonte: Próprio Autor

Figura 31: Teste feito com o Sensor de Índice Pluviométrico



Fonte: Próprio Autor

Os testes com a miniestação meteorológica foram realizados no horário das 08h00 às 11h00, iniciado no dia 01/08/2022 até o dia 31/08/2022, totalizando 31 dias de teste.

## **7 POSSÍVEIS MELHORIAS**

Embora os objetivos desse projeto tenham sido alcançados, no andamento do mesmo, observaram-se alguns pontos que podem passar por melhorias futuramente. Algumas melhorias sugeridas são:

- a) Utilização de microcontrolador com maior capacidade, para que seja possível incluir a leitura de mais variáveis climáticas;
- b) Incluir um sistema de alimentação através de placa solar, para que fosse possível economizar a carga da bateria durante o dia, prolongando assim a sua durabilidade.

## 8 CONCLUSÃO

Neste trabalho foi apresentada uma Miniestação Meteorológica automática de baixo custo, a qual é de suma importância para diversas áreas, principalmente para auxiliar os agricultores nas suas plantações.

Observando a diferença de preços entre o protótipo da miniestação desenvolvida e os demais modelos de estações meteorológicas automáticas disponíveis no mercado, notou-se que o modelo de miniestação criada com o ESP32, possui um preço bastante acessível para qualquer usuário, apesar de não conter todos os sensores que caracterizam uma estação meteorológica automática.

Além disso, a miniestação meteorológica desenvolvida pode ser viável para implementação em residências, escolas, salas comerciais e vários outros ambientes, já que ela atende diversas finalidades. Além, de ser portátil, facilitando assim a sua locomoção.

## 9 CRONOGRAMA

Tabela 6: Parte I do Cronograma

SENAI		Legenda		Em processo		Realizado		Atrasado		Gerente de Projeto: Lucas Barea																
Título do Projeto: Mini Estação Meteorológica																										
Item	Atividades	Área	Responsável	Duração		Julho					Agosto					Setembro										
				Início	Término	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5						
1	Definição dos grupos	Inic	Grupo	27/07/2022	29/07/2022	x	x	x					x	x	x	x	x	x	x	x	x	x	x	x	x	x
2	Definição do tema	Inic	Grupo	01/08/2022	04/08/2022	x	x	x						x	x	x	x	x	x	x	x	x	x	x	x	x
3	Solicitação de Aprovação	Inic	Isabela/Lucas	01/08/2022	05/08/2022	x	x	x						x	x	x	x	x	x	x	x	x	x	x	x	x
4	(Aprovado SIN?)	Inic	Diretoria	05/08/2022	05/08/2022	x	x	x	x	x				x	x	x	x	x	x	x	x	x	x	x	x	x
5	Lista de materiais (BOM)	Planj	Isabela/Lucas	15/08/2022	16/08/2022	x	x	x	x	x	x			x	x	x	x	x	x	x	x	x	x	x	x	x
6	Plan. Detalhamento técnico dos Componentes	Planj	Isabela/Lucas	31/08/2022	31/08/2022	x	x	x	x	x	x	x		x	x	x	x	x	x	x	x	x	x	x	x	x
7	Planilha de custo	Planj	Isabela/Lucas	15/08/2022	16/08/2022	x	x	x	x	x	x	x		x	x	x	x	x	x	x	x	x	x	x	x	x
8	Cotação e Compra	Planj	Isabela/Lucas	15/08/2022	28/09/2022	x	x	x	x	x	x	x				x	x	x	x	x	x	x	x	x	x	x
9	Chegada do material	Planj	Lucas	15/08/2022	28/09/2022	x	x	x	x	x	x	x		x												
10	Criação do Organograma	Planj	Isabela	11/08/2022	18/08/2022	x	x	x	x	x	x	x														
11	Elaboração do Escopo	Planj	Isabela	11/08/2022	20/10/2022	x	x	x	x	x	x	x														
12	Criação do Plano de Ação	Planj	Isabela	11/08/2022	31/08/2022	x	x	x	x	x	x	x														
13	Teste do Sensor (Temp - do Senai)	Exe	Equipe de teste	11/08/2022	31/08/2022	x	x	x	x	x	x	x														
14	Teste do Programa	Exe	Equipe de teste	11/08/2022	31/08/2022	x	x	x	x	x	x	x														
15	Teste do Sensor (Unid - do Senai)	Exe	Equipe de teste	11/08/2022	31/08/2022	x	x	x	x	x	x	x														
16	Teste do Programa	Exe	Equipe de teste	11/08/2022	31/08/2022	x	x	x	x	x	x	x														
17	Criação do Website	Exe	Equipe de Prog.	01/09/2022	23/11/2022	x	x	x	x	x	x	x														
18	Montagem da Caixa de Acrílico	Exe	Equipe de montagem	01/11/2022	22/11/2022	x	x	x	x	x	x	x														
19	Sub montagem (Testes)	Exe	Equipe de montagem	01/10/2022	01/11/2022	x	x	x	x	x	x	x														
20	Criação dos Slides	Conclusão	Isabela	18/11/2022	23/11/2022	x	x	x	x	x	x	x														
21	Try-out	Conclusão	Grupo	24/11/2022	24/11/2022	x	x	x	x	x	x	x														
22	Apresentação Final	Conclusão	Grupo	14/12/2022	15/12/2022	x	x	x	x	x	x	x														

Fonte: Próprio Autor





## 10 ORÇAMENTO

A tabela 8 abaixo demonstra referência ao orçamento de custo dos itens usados no projeto.

**Tabela 8 – Preços dos componentes**

Qtd.	Especificação	Valor Unitário	Total
1	ESP32 - WiFi + Bluetooth	R\$ 75,00	R\$ 75,00
1	Sensor de Chuva - YL-83	R\$ 9,90	R\$ 9,90
1	Sensor de Pressão e Temperatura - BMP280	R\$ 10,90	R\$ 10,90
1	Módulo de Bateria - Shield 18650	R\$ 55,00	R\$ 55,00
1	Protoboard 400 pontos	R\$ 22,00	R\$ 22,00
1	Protoboard 170 pontos	R\$ 10,00	R\$ 10,00
1	Caixa de Acrílico + Mão de Obra	R\$ 360,00	R\$ 360,00
1	Suporte "Pé" da Caixa de Acrílico	R\$ 70,00	R\$ 70,00
			Frete: R\$ 13,61
			Total: 626,41

Os valores mostrados acima podem sofrer variações de acordo com o local onde os produtos são adquiridos, ainda assim, os componentes necessários para a montagem da miniestação meteorológica apresentam custos significativamente baixos.

## REFERÊNCIAS

ABREU MURTA, José Gustavo. Sensores DHT11 e DHT22: Guia Básico dos Sensores de Umidade e Temperatura. **Eletrogate**. 12 set. 2022. Disponível em: <https://blog.eletrogate.com/sensores-dht11-dht22/>. Acesso em: 25 nov. 2022.

BERTOLETI, Pedro. Envie dados de temperatura e umidade relativa do ar para o ThingSpeak com o ESP8266. **Filipeflop**. 13 mar. 2020. Disponível em: <https://www.filipeflop.com/blog/esp8266-com-thingspeak/>. Acesso em: 09 Dez. 2022.  
CHUVA. **Sua Pesquisa**. [2019?]. Disponível em: <https://www.suapesquisa.com/geografia/chuva.htm>. Acesso em: 10 Dez. 2022.

COMO usar Visual Studio Code e por que aprender essa ferramenta?. **XP Educação**. 13 nov. 2022. Disponível em: <https://blog.xpeducacao.com.br/como-usar-visual-studio-code/#:~:text=O%20Visual%20Studio%20Code%20%C3%A9,s%C3%A3o%20Linux%2C%20Mac%20e%20Windows>. Acesso em: 09 Dez. 2022.

CORTES, Rafaela. Meteorologia – O que é? Significado, Sinônimos, Antônimos, Rimas e Exemplos. **Gestão Educacional**. 10 Dez. 2022. Disponível em: <https://www.gestaoeducacional.com.br/meteorologia-o-que-e/>. Acesso em: 10 Dez. 2022.

DALA COSTA, Augusto. O que é umidade relativa do ar? Níveis ideais e limites. **Canaltech**. 15 set. 2022. Disponível em: <https://canaltech.com.br/ciencia/o-que-e-umidade-relativa-do-ar-niveis-ideais-e-limites-225411/>. Acesso em: 10 Dez. 2022.

G, Ariane. O que é CSS? Guia Básico para Iniciantes. **Hostinger Tutoriais**. [2022]. Disponível em: <https://www.hostinger.com.br/tutoriais/o-que-e-css-guia-basico-de-css>. Acesso em: 08 Dez. 2022.

GUITARRARA, Paloma. Chuvas. **Brasil Escola**. [20--]. Disponível em: <https://brasilecola.uol.com.br/geografia/chuvas-precipitacoes.htm>. Acesso em: 10 Dez. 2022.

O que é HTML? Saiba como esse recurso funciona. **TOTVS**. 04 nov. 2020. Disponível em: <https://www.totvs.com/blog/developers/o-que-e-html/>. Acesso em: 09 Dez. 2022.

OLIVEIRA, Jailson. ESP32 – Especificação Técnica. **XProjetos**. 22 jul. 2018. Disponível em: <https://xprojetos.net/esp32-especificacao-tecnica/>. Acesso em: 25 nov. 2022.

POZA, Stephen. Diferencias entre SolidWorks e AutoCAD:Qual ferramenta CAD devo escolher?. **Datech**. [20--]. Disponível em: <https://www.datech.es/software/diferencias-entre-solidworks-y-autocad/>. Acesso em: 08 Dez. 2022.

ROVEDA, Ugo. JAVASCRIPT: O QUE É, PARA QUE SERVE E COMO FUNCIONA O JS?. **Kenzie**. 28 out. 2021. Disponível em: <https://kenzie.com.br/blog/javascript/>. Acesso em: 08 Dez. 2022.

SENSOR De Pressão e Temperatura Bmp280. **Filipeflop**. [20--]. Disponível em: <https://www.filipeflop.com/produto/sensor-de-pressao-e-temperatura-bmp280/>. Acesso em: 24 nov. 2022.

SHIELD V3 Carregador de Bateria 18650 com USB, Saída 5V / 3V e Proteção de Sobrecarga ou Descarga Excessiva. **Usinainfo**. [20--]. Disponível em: <https://www.usinainfo.com.br/carregador-de-bateria/shield-v3-carregador-de-bateria-18650-com-usb-saida-5v-3v-e-protecao-de-sobrecarga-ou-descarga-excessiva-5319.html>. Acesso em: 24 nov. 2022.

SOUSA, Rafaela. Diferença entre tempo e clima. **Brasil Escola**. [20--]. Disponível em: <https://brasilecola.uol.com.br/geografia/diferenca-entre-tempo-clima.htm>. Acesso em: 10 Dez. 2022.

STRAUB, Matheus Gebert. ARDUINO IDE – O SOFTWARE PARA GRAVAÇÃO DE CÓDIGOS NO ARDUINO. **Usinainfo**. 24 out. 2019. Disponível em: <https://www.usinainfo.com.br/blog/arduino-ide-o-software-para-gravacao-de-codigos-no-arduino/>. Acesso em: 09 Dez. 2022.

THOMSEN, Adilson. Sensor de chuva YL-83. **Filipeflop**. 18 fev. 2014. Disponível em: <https://www.filipeflop.com/blog/sensor-de-chuva-yl-83/>. Acesso em: 24 nov. 2022.

YNOUE, Rita Yuri. Meteorologia: noções básicas. **Google Livros**. [2017]. Disponível em: <https://books.google.com.br/books?hl=pt-BR&lr=&id=fdCdgAAQBAJ&oi=fnd&pg=PT9&dq=YNOUE+et+al.,+2017&ots=1F60ZPICao&sig=jO3CYvxXuroF2b03XuntTUDAuV4#v=onepage&q=YNOUE%20et%20al.%2C%202017&f=false>. Acesso em: 10 Dez. 2022.

## APÊNDICES

### Programação do Website

#### Estrutura HTML

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Mini-Estação Meteorológica</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <header>
    <h1 class="logo">TEAM ONE</h1>
    <nav id="nav">
      <button aria-label="Abrir Menu" id="btn-mobile" aria-haspopup="true" aria-
controls="menu" aria-expanded="false">Menu
      <span id="hamburger"></span>
    </button>
    <ul id="menu" role="menu">
      <li><a href="#" id="refresh">HOME</a></li>
      <li><a href="#" target="_blank">ESCOPO</a></li>
      <li><a href="https://forms.gle/USPK7qdKeSdBMoZy9"
target="_blank">AVALIAR</a></li>
    </ul>
  </nav>
</header>
<main>
  <div class="pagina-inicial">
    <div class="box-page">
      <h1>A melhor forma de descobrir<br>dados Meteorológicos</h1>
      <a href="https://thingspeak.com/channels/1892968"
target="_blank"><button class="btn">Clique aqui</button></a>
    </div>
    
  </div>
  <div class="usuarios">
    <h1 class="titulo">Usamos</h1>
    <div class="imgs">
      
      
      
      
      
    </div>
  </div>
</div>
```

```

<div class="dados">
  
  <div class="box-dados">
    <h1>As principais ferramentas utilizadas<br>para a Mini-Estação
Meteorológica</h1>
    <p>Utilizamos várias linguagens com cada uma tendo uma
finalidade,<br>como aqui em baixo
    mostrando a porcentagem de uso delas.
  </p>
  <div class="gráfico">
    <div class="dados-gráfico">
      <p>Html</p>
      <p>60%</p>
    </div>
    <span></span>
    <div class="dados-gráfico">
      <p>CSS</p>
      <p>75%</p>
    </div>
    <span></span>
    <div class="dados-gráfico">
      <p>JavaScript</p>
      <p>40%</p>
    </div>
    <span></span>
    <div class="dados-gráfico">
      <p>PHP</p>
      <p>85%</p>
    </div>
    <span></span>
    <div class="dados-gráfico">
      <p>ThingSpeak</p>
      <p>20%</p>
    </div>
    <span></span>
  </div>
</div>
</div>
<div class="box-carrousel">
  <h1>Portfólio</h1>
  <p>Tenha acesso as fotos de nosso trabalho</p>
  <div class="carrousel">
    <div class="btn-slide">
      <button class="btn-voltar control"></button>
      <button class="btn-avança control"></button>
    </div>
    <div class="galeria">

```

```

    <div div class="img-utcc">
      
      
      
      
      
      
    </div>
  </div>
</div>
</div>
</main>

<script>
  const controls = document.querySelectorAll(".control");
  let currentItem = 0;
  const items = document.querySelectorAll(".item");
  const maxItems = items.length;

  controls.forEach((control) => {
    control.addEventListener("click", (e) => {
      isLeft = e.target.classList.contains("btn-voltar");

      if (isLeft) {
        currentItem -= 1;
      } else {
        currentItem += 1;
      }

      if (currentItem >= maxItems) {
        currentItem = 0;
      }

      if (currentItem < 0) {
        currentItem = maxItems - 1;
      }

      items.forEach((item) => item.classList.remove("current-item"));

      items[currentItem].scrollIntoView({
        behavior: "smooth",
        inline: "center"
      });

      items[currentItem].classList.add("current-item");
    });
  });
</script>

```

```

<script>
  const btn = document.querySelector('#refresh')

  btn.addEventListener('click', () => {
    location.reload()
  })
</script>
<script>
  const btnMobile = document.getElementById('btn-mobile');

  function toggleMenu(event) {
    if (event.type === 'touchstart') event.preventDefault();
    const nav = document.getElementById('nav');
    nav.classList.toggle('active');
    const active = nav.classList.contains('active');
    event.currentTarget.setAttribute('aria-expanded', active);
    if (active) {
      event.currentTarget.setAttribute('aria-label', 'Fechar Menu');
    } else {
      event.currentTarget.setAttribute('aria-label', 'Abrir Menu');
    }
  }

  btnMobile.addEventListener('click', toggleMenu);
  btnMobile.addEventListener('touchstart', toggleMenu);
</script>
</body>
</html>

```

## Estrutura CSS

```

@import
url('https://fonts.googleapis.com/css2?family=Bebas+Neue&family=Poppins:wght@200;400&display=swap');

body{
  margin: 0;
  color: white;
  font-family: Poppins, sans-serif;
  font-size: 2.5vh;
  background-color: #1D282C;
}

header{
  background-color: #1D282C;
  margin: 0 5%;
  box-sizing: border-box;
  height: 70px;

```



```
padding: 1rem;
display: flex;
align-items: center;
justify-content: space-between;
}

.logo{
margin-right: auto;
}

ul{
display: flex;
}

li{
list-style: none;
display: inline-block;
margin: 10px;
padding: 0px 20px;
}

a{
text-decoration: none;
color: white;
transition: all 0.3s ease 0s;
}

a:hover{
color: #73817B;
}

main{
background-color: #1D282C;
}

.pagina-inicial{
display: flex;
justify-content: flex-start;
padding: 10px;
}

.box-page{
margin: 12% 7%;
}

.img-inicial{
height: 450px;
margin: 3% 10%;
margin-left: auto;
```

```
    animation: flutuacao 6s ease-in-out infinite;
    position: relative;
    z-index: 0;
}

.btn{
    outline: none;
    width: 100%;
    font-size: 2.5vh;
    font-family: Poppins, sans-serif;
    background: #73817B;
    color: white;
    border-radius: 20px;
    border: none;
    padding: 10px;
    width: 200px;
    cursor: pointer;
    box-shadow: 0px 10px 40px -12px #73817B;
    transition: all 0.3s ease 0s;
}

.btn:hover{
    background: #354243;
    box-shadow: 0px 10px 40px -12px #354243;
}

.usuarios{
    background: #F3F5FA;
}

.titulo{
    display: flex;
    justify-content: center;
    color: #73817B;
    padding: 10px 0px;
}

.imgs{
    display: flex;
    justify-content: center;
    padding: 5px 0px 80px;
}

.img{
    height: 80px;
    filter: grayscale(100%);
    padding: 0px 50px;
    transition: all 0.3s ease 0s;
}
```

```
.img-js{
  filter: grayscale(100%);
  transition: all 0.3s ease 0s;
  padding: 0px 40px;
}

.img-js:hover{
  filter: none;
  animation-name: anima;
  animation-duration: 0.5s;
  animation-timing-function: ease-in-out;
  animation-iteration-count: infinite;
  animation-direction: alternate;
}

.img:hover{
  filter: none;
  animation-name: anima;
  animation-duration: 0.5s;
  animation-timing-function: ease-in-out;
  animation-iteration-count: infinite;
  animation-direction: alternate;
}

.dados{
  background-color: white;
}

.img-dados{
  height: 330px;
  margin: 6% 0px;
}

.dados{
  display: flex;
  justify-content: center;
  padding: 20px 0px 100px;
}

.box-dados{
  color: #354243;
  margin: 1% 0px;
}

.dados-gráfico{
  display: flex;
  justify-content: space-between;
}
```

```

.dados-gráfico > span{
  font-weight: 500;
  font-size: 18px;
}

.box-carrousel > h1,p{
  display: flex;
  justify-content: center;
  padding: 20px 0px 0px 0px 200px;
}

.img-utcc{
  -ms-overflow-style: none; /* IE and Edge */
  scrollbar-width: none; /* Firefox */
}

.carrousel{
  position: relative;
  padding: 15px;
  max-width: 640px;
  margin: 0 auto;
}

.galeria{
  overflow-x: auto;
}

.galeria::-webkit-scrollbar {
  display: none;
}

.item {
  width: 300px;
  height: 300px;
  flex-shrink: 0;
  transition: all 600ms ease-in-out;
  opacity: 0.5;
}

.current-item {
  opacity: 1;
}

.img-utcc{
  display: flex;
  flex-flow: row nowrap;
  gap: 10px;
}

```

```
.btn-avança,.btn-voltar{
  position: absolute;
  top: 0;
  left: -30px;
  right: auto;
  bottom: 0;
  font-size: 40px;
  line-height: 300px;
  width: 60px;
  color: #fff;
  transition: all 600ms ease-in-out;
  background: none;
  cursor: pointer;
  border: none;
  outline: none;
}
```

```
.btn-avança{
  right: -12px;
  left: auto;
  text-align: right;
  background: none;
}
```

```
#menu {
  display: flex;
  list-style: none;
  gap: 0.5rem;
}
```

```
#menu a {
  display: block;
  padding: 0.5rem;
}
```

```
#btn-mobile {
  display: none;
  color: white;
  font-family: Poppins,sans-serif;
}
```

```
@media (max-width: 1250px) {

  header{
    display: flex;
    align-items: center;
    justify-content: space-between;
```

```

    margin: 0;
    position: fixed;
    width: 100%;
    top: 0;
    z-index: 1;
}

.pagina-inicial{
    display: flex;
    justify-content: center;
    align-items: center;
    flex-direction: column-reverse;
    padding: 10px;
}

.box-page{
    font-size: 13px;
}

.img-inicial{
    margin-top: 100px;
    height: 250px;
    animation: flutuacao 6s ease-in-out infinite;
}

.pagina-inicial{
    display: flex;
    align-items: center;
    justify-content: center;
    flex-direction: column-reverse;
    text-align: center;
    font-size: 20px;
    margin: 12% 7%;
}

.imgs{
    display: flex;
    flex-wrap: wrap;
    background: #F3F5FA;
}

.img{
    height: 80px;
    filter: grayscale(100%);
    margin: 10px;
    transition: all 0.3s ease 0s;
    position: relative;
    z-index: 0;
}

```

```
.dados{
  display: flex;
  justify-content: center;
  flex-direction: column;
  padding: 50px;
  font-size: 15px;
}

.img-dados{
  height: 230px;
  margin: 6% 0px;
}

.carrousel{
  position: relative;
  padding: 15px;
  width: 300px;
}

.box-carrousel{
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
}

.item {
  height: 280px;
  width: 260px;
  flex-shrink: 0;
  transition: all 600ms ease-in-out;
  position: relative;
  z-index: 0;
}

.box-carrousel > h1,p{
  display: flex;
  justify-content: center;
  margin: 14px;
}

.logo{
  font-size: 25px;
}

#menu {
  display: flex;
  position: fixed;
  top: 40px;
```

```

    right: 10px;
    width: 100%;
    background: #1D282C ;
    transition: 0.6s;
    visibility: hidden;
    overflow-y: hidden;
    flex-direction: column;
  }
  #nav.active #menu {
    height: calc(100vh - 70px);
    visibility: visible;
    overflow-y: auto;
  }

  #btn-mobile {
    display: flex;
    width: 100px;
    padding: 0.5rem 1rem;
    font-size: 1rem;
    border: none;
    background: none;
    cursor: pointer;
    gap: 0.5rem;
  }
  #hamburger {
    border-top: 2px;
    width: 15px;
  }
  #hamburger::after,
  #hamburger::before {
    content: "";
    display: block;
    width: 20px;
    height: 2px;
    background: currentColor;
    margin-top: 5px;
    position: relative;
    z-index: 1;
  }
  #nav.active #hamburger::before {
    transform: rotate(135deg);
  }
  #nav.active #hamburger::after {
    transform: rotate(-135deg);
    top: -7px;
  }
}

@keyframes flutuacao {

```



```
0%,
100% {
  transform: translateY(0px);
}

50% {
  transform: translateY(-20px);
}
}

@keyframes anima {
  0% {
    transform: scale(1);
  }

  100% {
    transform: scale(1.05);
  }
}
```

Estrutura JS (JavaScript)

```
const btn = document.querySelector('#refresh')

btn.addEventListener('click', () => {
  location.reload()
})
```

## ANEXOS

//Bibliotecas do Projeto.

```
#include <WiFi.h> //Inclui Biblioteca para utilizar o WiFi.
#include <DHT.h> //Inclui Biblioteca para utilizar o sensor DHT11.
#include <Wire.h> //Inclui Biblioteca para comunicar com dispositivos I2C.
#include <SPI.h> //Inclui Biblioteca usada para o microcontrolador para se
comunicar com um ou mais dispositivos em curtas distâncias.
#include <Adafruit_BMP280.h> //Inclui Biblioteca para utilizar o sensor BMP280.
#include <Adafruit_Sensor.h> //Inclui Biblioteca de sensores Adafruit.
```

/\*

\* Defines do Projeto.

\*/

//GPIO do NodeMCU que o pino de comunicação do sensor está ligado.

```
#define DHTPIN 5 //Define pino 5.
```

```
#define BMP_SCK (13) //Define pino 13.
```

```
#define BMP_MISO (12) //Define pino 12.
```

```
#define BMP_MOSI (11) //Define pino 11.
```

```
#define BMP_CS (10) //Define pino 10.
```

/\* defines - wi-fi \*/

```
#define SSID_REDE "Galaxy A2087B2" /* nome da rede que se deseja conectar */
```

```
#define SENHA_REDE "78787878" /* senha da rede que se deseja conectar */
```

```
#define INTERVALO_ENVIO_THINGSPEAK 30000 /* intervalo entre envios de
dados ao ThingSpeak (em ms) */
```

```
#define DHTTYPE DHT11 // DHT 11
```

/\* constantes e variáveis globais \*/

```
char endereco_api_thingspeak[] = "api.thingspeak.com";
```

```
String chave_escrita_thingspeak = "G5WQT38KXR1ZJLCU"; /* sua chave de escrita
do seu canal no Thingspeak */
```

```
unsigned long last_connection_time;
```

```
WiFiClient client;
```

```
DHT dht(DHTPIN, DHTTYPE);
```

```
Adafruit_BMP280 sensor_bmp;
```

```
const int RAIN_ADC = 34; //Define pino 34.
```

/\* prototypes \*/

```
void envia_informacoes_thingspeak(String string_dados);
```

```
void init_wifi(void);
```

```
void conecta_wifi(void);
```

```
void verifica_conexao_wifi(void);
```

```
float read_rain(void);
```

```

/*
 * Implementações
 */

/* Função: envia informações ao ThingSpeak
 * Parâmetros: String com a informação a ser enviada
 * Retorno: nenhum
 */
void envia_informacoes_thingspeak(String string_dados)
{
    if (client.connect(endereco_api_thingspeak, 80))
    {
        /* faz a requisição HTTP ao ThingSpeak */
        client.print("POST /update HTTP/1.1\n");
        client.print("Host: api.thingspeak.com\n");
        client.print("Connection: close\n");
        client.print("X-THINGSPEAKAPIKEY: "+chave_escrita_thingspeak+"\n");
        client.print("Content-Type: application/x-www-form-urlencoded\n");
        client.print("Content-Length: ");
        client.print(string_dados.length());
        client.print("\n\n");
        client.print(string_dados);

        last_connection_time = millis();
        Serial.println("- Informacoes enviadas ao ThingSpeak!");
    }
}

/* Função: inicializa wi-fi
 * Parametros: nenhum
 * Retorno: nenhum
 */
void init_wifi(void)
{
    Serial.println("-----WI-FI -----");
    Serial.println("Conectando-se a rede: ");
    Serial.println(SSID_REDE);
    Serial.println("\nAguarde...");

    conecta_wifi();
}

/* Função: conecta-se a rede wi-fi
 * Parametros: nenhum
 * Retorno: nenhum
 */
void conecta_wifi(void)

```

```

{
  /* Se ja estiver conectado, nada é feito. */
  if (WiFi.status() == WL_CONNECTED)
  {
    return;
  }

  /* refaz a conexão */
  WiFi.begin(SSID_REDE, SENHA_REDE);

  while (WiFi.status() != WL_CONNECTED)
  {
    delay(100); //Delay 0,1seg
  }

  Serial.println("Conectado com sucesso a rede wi-fi \n");
  Serial.println(SSID_REDE);
}

/* Função: verifica se a conexao wi-fi está ativa
 * (e, em caso negativo, refaz a conexao)
 * Parametros: nenhum
 * Retorno: nenhum
 */
void verifica_conexao_wifi(void)
{
  conecta_wifi();
}

float read_rain(void)
{
  unsigned int rain_raw;
  float rain_deal = 0.0;

  rain_raw = 4095 - analogRead(RAIN_ADC);
  rain_deal = rain_raw/40.95;

  return rain_deal;
}

void setup()
{
  Serial.begin(9600);
  Serial.println("Teste medidor");
  delay(3000); //Delay 3seg
  last_connection_time = 0;

  /* Inicializa sensor de temperatura e umidade relativa do ar */

```

```

dht.begin();
//Verifica a conexão do sensor BMP280
if (!sensor bmp.begin(0x76))
{
  Serial.println("Sensor não encontrado. Verifique as conexoes!");
  while (1);
}
sensor_bmp.setSampling(Adafruit_BMP280::MODE_NORMAL, /* Modo de
operação. */
  Adafruit_BMP280::SAMPLING_X2, /* Sobreamostragem de
temperatura */
  Adafruit_BMP280::SAMPLING_X16, /* Sobreamostragem de pressão.
*/
  Adafruit_BMP280::FILTER_X16, /* Filtrando. */
  Adafruit_BMP280::STANDBY_MS_500); /* Modo de espera. */

/* Inicializa e conecta-se ao wi-fi */
init_wifi();
}

//loop principal
void loop()
{
  char fields_a_serem_enviados[100] = {0};
  float temperatura_lida = 0.0; //Define como do tipo float ou flutuante.
  float umidade_lida = 0.0; //Define como do tipo float ou flutuante.
  float pressao_lida = 0.0; //Define como do tipo float ou flutuante.
  float altitude_lida = 0.0; //Define como do tipo float ou flutuante.
  float chuva_lida = 0.0; //Define como do tipo float ou flutuante.

  /* Força desconexão ao ThingSpeak (se ainda estiver conectado) */
  if (client.connected())
  {
    client.stop();
    Serial.println("- Desconectado do ThingSpeak");
    Serial.println();
  }

  /* Garante que a conexão wi-fi esteja ativa */
  verifica_conexao_wifi(); //Verifica conexão WiFi

  /* Verifica se é o momento de enviar dados para o ThingSpeak */
  if(millis() - last_connection_time > INTERVALO_ENVIO_THINGSPEAK )
  //Intervalo de envio.
  {
    temperatura_lida = dht.readTemperature(); //Realiza a leitura do dado de
temperatura.

```

```
    umidade_lida = dht.readHumidity(); //Realiza a leitura do dado de umidade.
    pressao_lida = sensor_bmp.readPressure(); //Realiza a leitura do dado de
    pressão.
    altitude_lida = sensor_bmp.readAltitude(1013.25); //Realiza a leitura do dado de
    altitude.
    chuva_lida = read_rain(); //Realiza a leitura do dado de chuva.

    sprintf(fields_a_serem_enviados,"field1=%.2f&field2=%.2f&field3=%.2f&field4=%.2f&
    field5=%.2f", temperatura_lida, umidade_lida, pressao_lida, altitude_lida,
    chuva_lida); //Respectivo campo de cada dado.
    envia_informacoes_thingspeak(fields_a_serem_enviados); //Envia os dados ao
    Thingspeak.
  }
  delay(1000); //Delay 1seg
}
```