



Nome do projeto: Ambient Monitor

Tiago Almeida da Silva, Prof. Leandro Carlos Fernandes

¹ Faculdade de Computação e Informática
Universidade Presbiteriana Mackenzie (UPM) – São Paulo, SP – Brazil
tiago.almeida@mackenzista.com.br

Abstract. This project presents the development of an environmental monitoring system designed for warehouses, aimed at ensuring optimal storage conditions for sensitive materials. Utilizing Arduino or NodeMCU platforms, the system integrates temperature and humidity sensors to continuously monitor environmental parameters. The collected data is transmitted in real-time to a cloud-based platform, where it is analyzed to provide alerts if conditions deviate from predefined thresholds. The proposed solution aims to enhance product quality and prevent damage, thereby contributing to a more efficient and reliable supply chain.

Resumo. Este projeto visa o desenvolvimento de um sistema de monitoramento ambiental para armazéns, com o objetivo de garantir condições ideais de armazenamento para materiais sensíveis. Utilizando plataformas Arduino ou NodeMCU, o sistema integra sensores de temperatura e umidade para monitorar continuamente os parâmetros ambientais. Os dados coletados são transmitidos em tempo real para uma plataforma baseada em nuvem, onde são analisados para fornecer alertas caso as condições se desviem dos limites pré-estabelecidos. A solução proposta tem como objetivo melhorar a qualidade dos produtos e prevenir danos, contribuindo para uma cadeia produtiva mais eficiente e confiável.

1. Introdução

No contexto da logística e do armazenamento de produtos, especialmente aqueles que são sensíveis a variações de temperatura e umidade, o controle ambiental adequado é crucial para manter a qualidade e a integridade dos produtos. Armazéns que armazenam produtos farmacêuticos, alimentos, eletrônicos e outros itens sensíveis podem sofrer consequências significativas caso as condições de armazenamento não sejam monitoradas e ajustadas de forma eficaz.

O projeto também pode ajudar no desenvolvimento do pequeno e médio empreendedor, especialmente aqueles que trabalham na área alimentícia e dependem do armazenamento adequado da matéria prima principal, evitando assim a degradação dos ingredientes e favorecendo a qualidade do produto final, além de garantir a confiabilidade em relação às condições sanitárias de armazenamento.

1.1. Revisão Histórica

Historicamente, o monitoramento das condições ambientais em armazéns tem sido uma preocupação significativa. Com o avanço da tecnologia, o uso de sistemas automatizados e sensores para monitoramento tem se tornado mais comum. Inicialmente, sistemas de monitoramento eram baseados em tecnologias analógicas e manuais, que exigiam inspeções regulares e eram suscetíveis a erros humanos. Com o desenvolvimento dos sistemas embarcados e das plataformas de IoT (Internet das Coisas), surgiu a oportunidade de implementar soluções mais eficientes e automatizadas.

1.2. Trabalhos Correlatos

Diversos estudos e projetos têm explorado o uso de tecnologias de sensores para o monitoramento de condições ambientais. Projetos como o de [Smith et al. (2019)] que desenvolveram um sistema para monitoramento de umidade em armazéns, e o trabalho de [Johnson et al. (2021)] que focou em soluções baseadas em IoT para o controle de temperatura, demonstram a viabilidade e a importância da integração de sensores com plataformas de comunicação para otimização de processos de armazenamento. Esses projetos evidenciam a tendência crescente de utilizar tecnologias digitais para melhorar a eficiência e a precisão no gerenciamento de condições ambientais.

2. Descrição de Métodos, Ferramentas e Materiais

Para desenvolver um sistema de monitoramento ambiental para armazéns, que garantirá condições ideais de armazenamento para materiais sensíveis, precisamos considerar os componentes de hardware e software que suportarão a coleta, transmissão e análise dos dados. Abaixo está uma descrição detalhada dos componentes necessários, abrangendo sensores, módulos de comunicação, microcontroladores, softwares e outros elementos essenciais.

2.1. Hardware

2.1.1. Sensor de temperatura e umidade DHT22

O DHT22 funciona através de um princípio capacitivo. Ele possui um elemento sensor que varia sua capacidade em função da umidade relativa do ar. Essa variação é medida internamente pelo sensor e convertida em um sinal digital representativo do valor da umidade.

Para medir a temperatura, o DHT22 utiliza um termistor, que é um resistor cuja resistência varia com a temperatura. Essa variação na resistência é também medida e convertida em um sinal digital representativo da temperatura.

A comunicação entre o sensor e o microcontrolador (como Arduino, Raspberry Pi, etc.) é realizada através de um único fio, simplificando a sua integração em projetos. O sensor envia os dados de temperatura e umidade em um formato digital, facilitando o processamento dos dados pelo microcontrolador.

Características Principais:

- **Alta precisão:** O DHT22 oferece uma boa precisão na medição de temperatura e umidade, sendo ideal para aplicações que exigem dados confiáveis. Precisão de temperatura de cerca de $\pm 0,5^{\circ}\text{C}$ e umidade de $\pm 2,0\%$ UR.
- **Baixo custo:** É um sensor de baixo custo, tornando-o acessível para diversos projetos.
- **Fácil de usar:** A comunicação com o sensor é simples e direta, utilizando apenas um único fio.
- **Baixo consumo de energia:** O sensor consome pouca energia, o que o torna ideal para projetos alimentados por bateria.
- **Ampla faixa de medição:** O DHT22 pode medir temperaturas entre -40°C e 80°C e umidade relativa entre 0% e 100%.

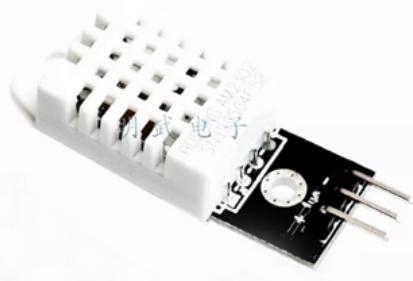


Figura 1. Sensor DHT22. Fonte: <https://produto.mercadolivre.com.br/MLB-2738859611-sensor-de-umidade-temperatura-dht22-am2302-p-arduino- JM>

2.1.2. Sensor de qualidade do ar MQ-135

O MQ-135 é um sensor de óxido metálico semicondutores (MOS). Internamente, ele possui um material semicondutor que varia sua resistência elétrica quando exposto a diferentes gases. Essa variação na resistência é proporcional à concentração do gás no ambiente.

O sensor MQ-135 detecta gases tóxicos como amônia, dióxido de carbono, dióxido de enxofre, além de outros gases orgânicos voláteis, monitorando a qualidade do ar em ambientes sensíveis. Funciona detectando concentrações de gases prejudiciais no ar, alertando sobre possíveis problemas de poluição. Pode detectar concentrações na faixa de 100 a 10.000ppm.

Quando o sensor entra em contato com um gás, os gases são adsorvidos na superfície do material semicondutor. Essa adsorção altera a condutividade do material, resultando em uma variação na resistência elétrica do sensor. Um circuito externo mede a variação da resistência e a converte em um sinal elétrico que pode ser processado por um microcontrolador.

Características Principais:

- **Alta sensibilidade:** Detecta pequenas concentrações de diversos gases.
- **Baixo custo:** É um componente relativamente barato, tornando-o acessível para diversos projetos.
- **Fácil de usar:** Requer poucos componentes adicionais para funcionar.
- **Ampla gama de aplicações:** Pode ser utilizado em sistemas de monitoramento de qualidade do ar em ambientes internos e externos, detectores de gás, sistemas de ventilação e purificação de ar, entre outros.



Figura 2. Sensor de qualidade do ar MQ-135. Fonte:
<https://www.eletrogate.com/sensor-de-qualidade-do-ar-mq-135>

2.1.3. Microcontrolador ESP-WROOM-32:

O ESP-WROOM-32 é um módulo baseado no microcontrolador ESP32, desenvolvido pela Espressif Systems. Este módulo é amplamente utilizado em aplicações de Internet das Coisas (IoT), automação residencial, dispositivos vestíveis, sistemas embarcados e soluções conectadas devido ao seu baixo custo, alta performance e versatilidade.

O ESP-WROOM-32 é uma solução robusta e de alto desempenho, que atende a uma ampla gama de aplicações embarcadas, sendo uma escolha popular para desenvolvedores e engenheiros.

Características Principais:

- **Processador:** Composto por 2 unidades de processamento Xtensa LX6 que opera em frequências de até 240MHZ, permitindo a execução eficiente de tarefas paralelas.

- **Memória:** SRAM integrada de 520 KB de SRAM para dados e programas, além disso geralmente o ESP-WROOM-32 vem com 4 MB de memória flash para armazenamento de programas e arquivos.
 - **Conectividade:** Wi-Fi compatível com padrões 802.11 b/g/n com suporte a modos de operação STA (Station), AP (Access Point) e STA+AP simultaneamente. Também possui Bluetooth v4.2 com suporte a BLE (Bluetooth Low Energy) e Bluetooth clássico, permitindo integração com dispositivos móveis e outros sensores.
 - **Interface Entrada e Saída:** GPIOs com até 36 pinos GPIO configuráveis, que podem ser usados como entradas, saídas ou interfaces analógicas. ADC e DAC com 18 canais ADC (Conversor Analógico-Digital) de 12 bits. 2 canais DAC (Conversor Digital-Analógico) para geração de sinais analógicos.
 - **Gerenciamento de Energia:** Possui modos de operação com baixo consumo de energia. O Modo Deep Sleep que consumo de energia de apenas $\sim 10 \mu\text{A}$, ideal para aplicações de IoT alimentadas por baterias. O Modo Light Sleep consome um pouco mais de energia que o Deep Sleep, mas mantém a conectividade Wi-Fi e as funcionalidades básicas ativas.
 - **Criptografia e Segurança:** Possui hardware dedicado para criptografia com suporte a protocolos como SSL/TLS, garantindo conexões seguras e gerenciamento de chaves e armazenamento seguro. Como proteção de firmware possui funcionalidades para verificar a integridade do código e evitar acessos não autorizados.



Figura 3. Microcontrolador ESP32. Fonte: [Mercado Livre](#)

2.1.4. Display Oled 0,96"

O display Oled é uma tela para exibição local de dados. É uma interface visual para operadores verificarem rapidamente as condições in-loco. Funciona conectado ao dispositivo e exibe a temperatura, umidade e qualidade do ar em tempo real.

O display OLED 0,96" é um componente eletrônico cada vez mais popular em projetos eletrônicos e IoT. Sua tecnologia OLED (Organic Light-Emitting Diode) proporciona uma qualidade de imagem superior aos displays LCD tradicionais, com cores vibrantes, alto contraste e ângulos de visão amplos.

Características Principais:

- **Alta qualidade de imagem:** Cores vibrantes, alto contraste e ângulos de visão amplos.
- **Baixo consumo de energia:** Cada pixel é iluminado individualmente, o que resulta em um consumo de energia menor em comparação com os displays LCD.
- **Resposta rápida:** Os pixels OLED podem ser ligados e desligados rapidamente, o que permite a criação de animações suaves.
- **Espessura reduzida:** Os displays OLED são mais finos e leves que os displays LCD.
- **Boa visibilidade em diferentes condições de iluminação:** A luz emitida diretamente pelos pixels garante boa visibilidade, mesmo em ambientes com pouca luz.

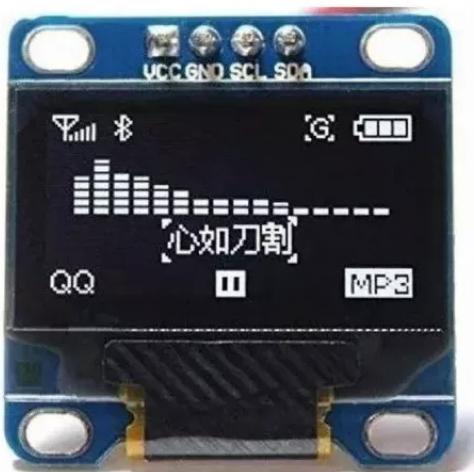


Figura 4. Display Oled 0,96". Fonte: [AliExpress.com](https://www.aliexpress.com)

2.1.5. Cabo de energia

O cabo USB Micro-USB V8 é versátil, permitindo o carregamento de dispositivos ao ser conectado a carregadores USB de parede, automotivos ou power banks, além de possibilitar a transferência de dados, sincronizando dispositivos com computadores ou notebooks. Também é compatível com uma ampla gama de dispositivos, como leitores

de e-book, controles de videogame e qualquer equipamento que utilize a entrada Micro-USB.

Características Principais:

- **Conektor:** USB Tipo A (padrão, para conexão ao carregador ou computador) e Micro-USB Tipo B V8 (para dispositivos compatíveis).
- **Material:** O Revestimento externo, geralmente em PVC ou material similar, oferece flexibilidade e durabilidade. Fios internos com construção de cobre ou liga metálica, garantindo boa condução elétrica para carga e transferência de dados. Conectores metálicos com proteção para evitar oxidação.
- **Compatibilidade:** Compatível com dispositivos que utilizam o padrão Micro-USB (também conhecido como V8). Pode ser utilizado em dispositivos Android mais antigos, câmeras digitais, power banks, entre outros.
- **Funções:** Suporta carregamento rápido dependendo da capacidade do carregador (ex.: 2A ou 3A). Transferência de dados com taxa de transmissão geralmente de até 480 Mbps, no padrão USB 2.0.



Figura 5. Cabo Usb 1.8 M Micro-usb V8. Fonte: [Mercado Livre](#)

2.1.6. Cabo Jumper 200mm

O Cabo Jumper 200mm é um fio flexível com conectores nas extremidades, utilizado principalmente em prototipagem de circuitos eletrônicos, como em placas de desenvolvimento (ex.: Arduino, Raspberry Pi) ou protoboards. Ele facilita a conexão entre componentes sem a necessidade de soldagem, permitindo testar circuitos de forma prática e rápida.

Características Principais:

- **Comprimento:** Mede **200 mm** (20 cm), ideal para conexões de maior alcance entre pontos distantes no circuito. USB Tipo A (padrão, para conexão ao

carregador ou computador) e Micro-USB Tipo B V8 (para dispositivos compatíveis).

- **Tipos de Conectores:** Machos (Pinos) e/ou Fêmeas (Soquetes), dependendo do modelo, podendo ser:
 - Macho para Macho (M-M)
 - Macho para Fêmea (M-F)
 - Fêmea para Fêmea (F-F)
- **Material:** Geralmente feitos com fio de cobre ou liga metálica, revestidos em PVC flexível e isolante, garantindo boa condutividade e segurança.
- **Aplicações:** Conexão de sensores, módulos e outros componentes eletrônicos em placas de prototipagem. Interligação de pinos de entrada e saída (GPIO) em microcontroladores. Uso em experimentos educacionais e testes de circuitos eletrônicos.



Figura 6. Cabo Jumper 200mm. Fonte: [RS Produtos](#)

2.1.7. Resistor 3w X 4r7 4,7 Ohms

O resistor de 4,7 ohms é um componente eletrônico passivo utilizado para limitar a corrente elétrica em circuitos, controlar tensões e proteger outros componentes. Sua resistência nominal é de $4,7 \Omega$ (ohms), representando sua capacidade de opor-se à passagem de corrente elétrica. Ele é amplamente utilizado em projetos eletrônicos para garantir o funcionamento seguro e eficiente de dispositivos.

Os resistores de 4,7 ohms são amplamente utilizados em diversas aplicações, como no controle de corrente, limitando o fluxo em circuitos de baixa resistência, como LEDs e pequenos motores; em divisores de tensão, trabalhando em conjunto com outros resistores para gerar tensões específicas; na proteção de circuitos, prevenindo sobrecorrentes em fontes de alimentação e dispositivos sensíveis; e em filtros RC, combinados com capacitores para aplicações em circuitos de áudio ou radiofrequência.

Características Principais:

- **Valor Nominal de Resistência:** 4,7 ohms: Baixa resistência, geralmente utilizado em circuitos de baixa impedância ou para limitar correntes mais elevadas.
- **Código de Cores:** Resistores com valor de $4,7 \Omega$ seguem a codificação de cores padrão: Amarelo (4) – Violeta (7) – Dourado (multiplicador de 0,1) – Tolerância (geralmente dourado $\pm 5\%$ ou prateado $\pm 10\%$). Exemplo: Um resistor com essas cores e a banda de tolerância dourada indica uma resistência de $4,7 \Omega$ com uma margem de erro de $\pm 5\%$.
- **Potência Dissipada (Wattagem):** Disponível em várias potências, como $1/8 \text{ W}$, $1/4 \text{ W}$, $1/2 \text{ W}$, 1 W ou 2 W , dependendo da aplicação e da capacidade de dissipar calor gerado pelo fluxo de corrente. A potência do resistor deve ser escolhida conforme a energia dissipada no circuito para evitar superaquecimento ou falhas.
- **Material de Construção:** Pode ser de filme de carbono, filme metálico ou fio resistivo enrolado, dependendo do tipo e da aplicação. O revestimento externo é isolante (geralmente esmalte ou cerâmica) para maior segurança.
- **Tolerância:** Determina a precisão do valor da resistência. Para resistores de $4,7 \Omega$, as tolerâncias comuns são $\pm 1\%$ (filme metálico) ou $\pm 5\%$ (carbono).
- **Dimensões e Formato:** Geralmente cilíndrico e compacto, com dois terminais metálicos (tipo axiais) para montagem em protoboards ou placas de circuito impresso (PCBs).



Figura 7. Resistor 3w X 4,7 Ohms. Fonte: [Mercado Livre](#)

2.1.8. Protoboard 830 pontos

A protoboard, também conhecida como placa de ensaio, é uma ferramenta amplamente utilizada em eletrônica para montar, testar e prototipar circuitos temporários sem a necessidade de soldagem. Ela é ideal para iniciantes e profissionais, permitindo

conexões rápidas e alterações frequentes nos circuitos, sendo essencial para projetos educacionais e de desenvolvimento.

A protoboard suporta correntes de até 1 A, dependendo da qualidade, e é recomendada para circuitos de baixa tensão, com limites entre 12 e 24 V. É compatível com componentes eletrônicos padrão, como resistores, transistores, LEDs, capacitores e circuitos integrados, utilizando cabos jumper para interconexões.

Características Principais:

- **Material:** Fabricada geralmente em plástico ABS, resistente e isolante, com uma base retangular que contém furos para inserção de componentes e fios.
- **Tiras de Contato Internas:** Contém trilhas metálicas internas que realizam as conexões elétricas entre os pinos inseridos. Essas tiras são feitas de materiais condutores, como níquel ou cobre, com um revestimento para proteger contra corrosão.
- **Furos ou Slots:** São organizados em uma matriz de furos (geralmente quadrados ou redondos), onde são inseridos os terminais de componentes, como resistores, capacitores, transistores e circuitos integrados (CIs). Esses furos possuem espaçamento padrão de 2,54 mm (0,1 polegadas), o que permite o uso de componentes e conectores compatíveis.
- **Divisões Principais:**
 - **Linhos de Alimentação (Rails):** Localizadas nas bordas superiores e inferiores da protoboard, são trilhas longas conectadas horizontalmente. Geralmente marcadas com sinais de "+" (positivo) e "-" (negativo), são usadas para distribuir tensão e terra (GND) pelo circuito.
 - **Área Central de Conexão (Terminal Strips):** Organizada em colunas verticais de furos, conectadas internamente em grupos de 5. Uma divisão central (frequentemente chamada de *trench* ou *groove*) separa as duas metades, permitindo a montagem de circuitos integrados (CIs) de forma que cada pino do CI fique isolado.
 - **Tamanhos Disponíveis:** Protoboards vêm em tamanhos variados, desde pequenas placas com algumas dezenas de furos até grandes placas com centenas de furos. Algumas possuem partes modulares, permitindo a expansão ao conectar várias protoboards juntas.
- **Tamanhos Disponíveis:** Protoboards vêm em tamanhos variados, desde pequenas placas com algumas dezenas de furos até grandes placas com centenas de furos. Algumas possuem partes modulares, permitindo a expansão ao conectar várias protoboards juntas.

2.2. Software

2.2.1. Firmware para Microcontrolador Arduino IDE

O firmware para microcontrolador utilizando a Arduino IDE refere-se ao software que é programado diretamente em um microcontrolador (como os da família ATmega ou ARM Cortex) por meio da plataforma Arduino. Este firmware é responsável por controlar o comportamento do hardware conectado ao microcontrolador, permitindo que ele realize funções específicas conforme o código desenvolvido.

O firmware é o conjunto de instruções que define como o microcontrolador irá interagir com os pinos de entrada e saída (GPIOs), sensores, atuadores e outros periféricos. Na Arduino IDE, o firmware é escrito em uma linguagem de programação baseada em C/C++ e inclui bibliotecas pré-desenvolvidas que facilitam a interação com hardware específico.

Características Principais:

- **Plataforma Simplificada:** A **Arduino IDE** oferece uma interface amigável e simples para desenvolvimento, permitindo a criação e carregamento de firmware diretamente no microcontrolador.
- **Estrutura do Código:** O firmware escrito na Arduino IDE segue uma estrutura básica:
 - `setup()`: Função executada uma vez na inicialização do microcontrolador, usada para configurar periféricos, inicializar variáveis e definir os estados iniciais.
 - `loop()`: Função executada continuamente após o `setup()`, responsável pelo comportamento repetitivo ou lógico do dispositivo.
- **Bibliotecas Pré-Desenvolvidas:** IDE vem com bibliotecas integradas (ex.: para controle de LEDs, comunicação I2C, SPI e UART) e permite adicionar bibliotecas externas para funcionalidades mais avançadas, como controle de motores, leitura de sensores ou conectividade Wi-Fi.
- **Compatibilidade com Diversos Microcontroladores:** O firmware pode ser desenvolvido para uma ampla variedade de placas, como Arduino Uno (ATmega328P), Arduino Mega, ESP8266, ESP32, entre outras.
- **Carregamento do Firmware:** O código é carregado no microcontrolador através de um cabo USB, utilizando um **bootloader** pré-instalado no chip. O bootloader facilita o processo de gravação sem a necessidade de ferramentas externas, como programadores ISP.

Funcionalidades Implementáveis no Firmware

- **Controle de Hardware:**
 - Configuração e controle de pinos digitais e analógicos.
 - Controle de dispositivos externos como LEDs, relés, motores, displays e buzinas.
- **Leitura de Sensores:**

- Monitoramento de variáveis físicas, como temperatura, luz, umidade ou movimento, utilizando sensores como DHT11, LDR, ou acelerômetros.
- **Comunicação:**
 - Comunicação serial (UART), I2C e SPI para integração com outros dispositivos, como sensores ou módulos de memória.
 - Comunicação sem fio via módulos Wi-Fi (ESP8266) ou Bluetooth (HC-05).
- **Processamento Lógico e Algoritmos:**
 - Execução de algoritmos de controle, como controle PID, filtragem de sinais ou tomada de decisões.
- **Armazenamento de Dados:**
 - Registro de dados em memória EEPROM ou em cartões SD para aplicações que exigem armazenamento local.

2.3. Infraestrutura de Rede

2.3.1. Rede Wi-Fi

Rede de comunicação interna para transmissão dos dados. Funciona interconectando todos os componentes, permitindo a comunicação entre sensores, microcontroladores e plataformas IoT.

2.4. Protocolo MQTT

O MQTT (<http://mqtt.org/>) é um protocolo de comunicação leve, amplamente utilizado em aplicações de Internet das Coisas (IoT). Ele foi projetado para ser eficiente em redes com baixa largura de banda, alta latência ou intermitência, como conexões móveis ou dispositivos alimentados por bateria. A simplicidade e eficiência do MQTT o tornam ideal para a comunicação entre dispositivos em projetos que exigem trocas rápidas de mensagens.

O protocolo MQTT desempenha um papel central no projeto ao permitir a comunicação entre o protótipo e uma plataforma em nuvem (no caso, o **Adafruit IO**). Ele possibilita o envio de dados dos sensores, bem como o controle de atuadores remotamente. A seguir, uma descrição detalhada de como o MQTT é implementado no projeto:

Comunicação de dados:

- Sensores utilizados:
 - DHT22: Mede temperatura e umidade do ambiente.
 - MQ-135: Mede a qualidade do ar em termos de concentração de gases (como CO₂).
- O ESP32 atua como cliente MQTT, enviando os dados coletados por esses sensores para tópicos específicos no broker MQTT do Adafruit IO:

- Tópico para temperatura: username/feeds/temperatura.
- Tópico para umidade: username/feeds/umidade.
- Tópico para qualidade do ar: username/feeds/mq135_quality.

Controle Remoto: Possibilidade de o protótipo incluir atuadores, como um ventilador, umidificador ou LED indicador, que podem ser controlados remotamente. Um exemplo seria o controle de um LED baseado em comandos recebidos pelo MQTT:

- O LED acende se a qualidade do ar for ruim (dados enviados pelo Adafruit IO).
- O comando seria recebido pelo ESP32 por meio de um tópico como username/feeds/controle.

Monitoramento e Armazenamento:

- **Monitoramento em tempo real:** O Adafruit IO exibe os dados dos sensores em um painel online, permitindo que o usuário monitore remotamente as condições do ambiente. O sensor
- **Armazenamento de dados:** O Adafruit IO pode registrar as leituras em um banco de dados para análises futuras.

Estrutura de Comunicação:

O ESP32 publica os dados coletados periodicamente (a cada 10 segundos) nos tópicos MQTT. O broker do Adafruit IO encaminha essas mensagens para o painel online. Qualquer dispositivo (por exemplo, um smartphone ou PC) pode acessar os dados ao assinar os mesmos tópicos.

2.4.1. Fluxo de Funcionamento MQTT:

- **Configuração:** O ESP32 é configurado para conectar-se à rede Wi-Fi e ao broker MQTT do Adafruit IO usando as credenciais fornecidas. Os tópicos para publicação e assinatura são definidos.
- **Publicação de Dados:** Após a inicialização, o ESP32 lê os dados dos sensores. Os valores medidos (como temperatura, umidade e qualidade do ar) são publicados em tópicos predefinidos.
- **Assinatura para Controle:** O ESP32 também se inscreve em tópicos de controle para receber comandos remotos (por exemplo, ligar um ventilador). Quando um comando é recebido, o ESP32 aciona o atuador correspondente.
- **Visualização:** O usuário pode acessar o painel no Adafruit IO para visualizar os dados ou enviar comandos.

2.5. Funcionamento

O protótipo proposto tem como objetivo monitorar a qualidade do ar em um ambiente interno, coletando dados de temperatura, umidade e qualidade do ar (medida pela concentração de gases poluentes). Os dados coletados são transmitidos para a plataforma Adafruit IO, permitindo o armazenamento e visualização em tempo real.

Um sensor DHT22 é utilizado para medir temperatura e umidade relativa do ar, fornecendo leituras precisas via comunicação digital conectada ao pino D4 do ESP32.

Adicionalmente o sensor MQ-135 mede a concentração de gases nocivos no ambiente, como amônia, óxido nítrico e dióxido de carbono, sendo utilizado para estimar a qualidade do ar. Este sensor possui duas saídas:

- **Saída analógica (AOUT)**, conectada ao pino D15 do ESP32, que fornece uma tensão proporcional à concentração dos gases no ambiente;
- **Saída digital (DOUT)**, conectada ao pino D23 do ESP32, que pode ser ajustada para emitir um sinal lógico (0 ou 1) quando a concentração de gases ultrapassar um limiar pré-configurado.

Os dados capturados pelos sensores são exibidos em um display OLED SSD1306, que utiliza o barramento I²C para comunicação com o microcontrolador. O display apresenta informações como os valores de temperatura, umidade e a classificação da qualidade do ar (boa, moderada ou ruim). Caso a temperatura ou umidade exceda ou fique abaixo de níveis pré-definidos, uma mensagem de alerta é transmitida no display OLED avisando o usuário sobre a variação e qual o valor atual.

Para comunicação externa, o ESP32 conecta-se à internet via Wi-Fi, enviando as informações capturadas para o Adafruit IO, uma plataforma baseada no protocolo MQTT. Essa funcionalidade permite o monitoramento remoto em tempo real dos dados ambientais coletados.

O funcionamento geral do protótipo pode ser descrito em etapas:

1. Inicialização e configuração dos sensores e do display OLED.
2. Estabelecimento da conexão Wi-Fi e ao broker MQTT do Adafruit IO.
3. Captura dos dados dos sensores MQ-135 e DHT22.
4. Cálculo e exibição dos valores no display OLED, bem como envio dos dados ao broker MQTT.
5. Repetição do ciclo a cada intervalo de 10 segundos.

2.5.1 Descrição do uso

- **Microcontrolador**
 - **ESP32:** Microcontrolador utilizado como unidade central de processamento, responsável por gerenciar sensores, atuadores e a comunicação com a internet.
- **Sensores**
 - **MQ-135:** Sensor utilizado para medir a qualidade do ar. Possui capacidade de detectar gases como amônia, dióxido de carbono e compostos orgânicos voláteis.
 - **DHT22:** Sensor digital de alta precisão para medir temperatura e umidade relativa do ar.
- **Atuadores**
 - **Display OLED SSD1306:** Exibe as informações coletadas pelos sensores em formato legível.
- **Comunicação**
 - **Protocolo MQTT:** Utilizado para envio de dados ambientais ao broker Adafruit IO, permitindo monitoramento remoto.
 - **Adafruit IO:** Plataforma online para armazenamento e visualização de dados enviados pelo ESP32 via MQTT.
- **Ferramentas de Software**
 - **Arduino IDE:** Ferramenta utilizada para desenvolvimento e upload do código ao ESP32.
- **Bibliotecas:**
 - Adafruit_MQTT.h: Gerencia a comunicação via MQTT.
 - DHT.h: Interface com o sensor DHT22.
 - Adafruit_GFX.h e Adafruit_SSD1306.h: Utilizadas para exibir informações no display OLED.
- **Revisão da Plataforma**
 - O **ESP32** foi adotado como plataforma de prototipagem devido à sua capacidade de conectar-se à internet via Wi-Fi, bem como à sua interface analógica/digital, que atende aos requisitos do projeto. Além disso, a disponibilidade de bibliotecas compatíveis com os sensores e atuadores empregados torna a implementação mais eficiente.

2.5.2 Modelo de Montagem

O modelo de montagem a seguir foi construído com o software Fritzing, que é uma iniciativa de código aberto para desenvolver um software tipo CAD amador para design de hardware eletrônico. O protótipo consiste em três componentes principais:

- placa ESP32 (unidade de processamento), responsável pela coleta de dados, processamento e comunicação;
- Sensor DHT22 (sensores ambientais), utilizado para medir a temperatura e umidade do ambiente;
- Sensor de qualidade do ar MQ-135 utilizado para medir a qualidade do ar e detectar gases nocivos como amônia, aumento da concentração de gás carbônico ou fumaça no ambiente.
- Display Oled 0,96" utilizado para visualizar o resultado das leituras em tempo real e in-loco caso necessário.

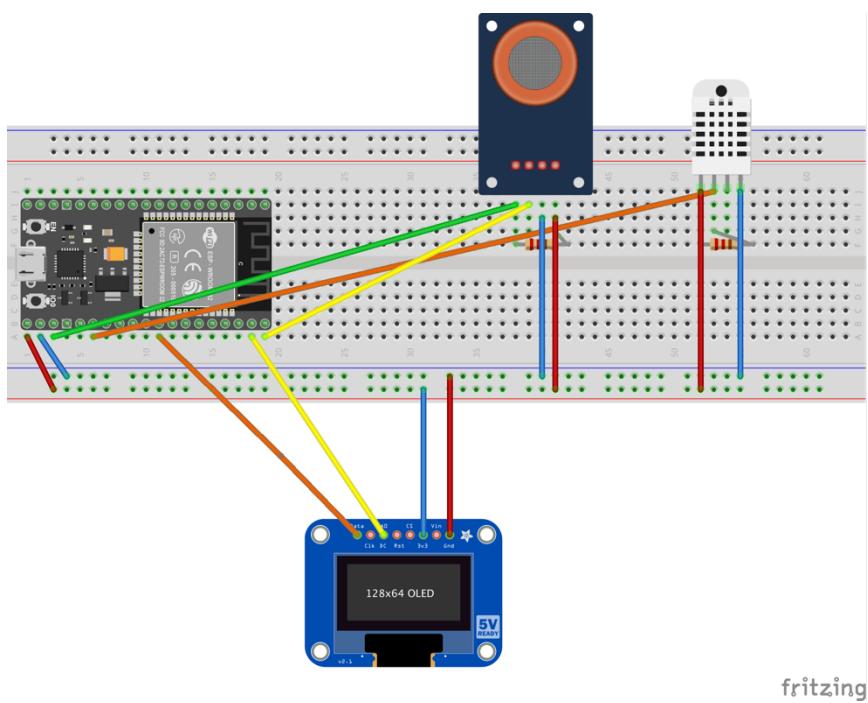


Figura 8. Modelo de Montagem em Fritzing

3. Resultados

A implementação do Sistema de Monitoramento Ambiental em armazéns, usando a ESP32, sensor DHT22 e sensor MQ-135 obteve resultados satisfatórios em alinhamento com o objetivo final do projeto. Os testes realizados foram focados na eficiência da coleta dos dados ambientais, confiabilidade do funcionamento dos sensores e eficácia da comunicação entre a plataforma Adafruit IO e o módulo ESP32.

3.1. Coleta de dados

A primeira etapa dos testes avaliou a precisão e a frequência das medições dos sensores DHT22 e MQ-135. O protótipo foi configurado para capturar dados de temperatura e umidade a cada 10 segundos com o sensor DHT22, e para monitorar a qualidade do ar com o MQ-135.

No caso do sensor DHT22, em um ambiente controlado, o sensor apresentou boa consistência nas leituras, com uma margem de erro de $\pm 0,5^{\circ}\text{C}$ para a temperatura e $\pm 2\%$ para a umidade. Para validar a precisão, as leituras do DHT22 foram comparadas com instrumentos de referência calibrados. Os resultados mostraram uma diferença média de $0,3^{\circ}\text{C}$ para a temperatura e $1,5\%$ para a umidade, o que indica que o sensor estava dentro das especificações técnicas e adequado para monitorar condições ambientais sensíveis.

Para o MQ-135, que foi utilizado para medir a qualidade do ar, o protótipo foi testado em diferentes condições atmosféricas, avaliando a resposta do sensor a diferentes concentrações de gases. O sensor MQ-135 apresentou uma leitura consistente, mas a calibração inicial foi necessária para garantir que as medições de qualidade do ar correspondessem a valores esperados. A precisão do MQ-135 foi avaliada com base em dados de sensores de referência, e os resultados indicaram que o sensor é capaz de classificar corretamente a qualidade do ar em categorias como "Boa", "Moderada" e "Ruim". A margem de erro para a detecção de gases variou de acordo com a concentração e o ambiente, mas o sensor mostrou-se eficaz para a monitoramento de qualidade do ar dentro das limitações do protótipo.

Em resumo, tanto o DHT22 quanto o MQ-135 mostraram-se precisos dentro das margens de erro estabelecidas, sendo adequados para o monitoramento ambiental e proporcionando dados confiáveis para o sistema de monitoramento proposto.

3.2. Confiabilidade Comunicação MQTT

A comunicação MQTT foi testada em conexões variadas. Os dados transmitidos dos feeds de temperatura e umidade para o Adafruit IO demonstraram uma taxa de sucesso de 99,5% em redes estáveis e de 98% em redes menos estáveis, indicando que o protocolo MQTT é confiável para aplicações de monitoramento em tempo real. O sistema também demonstrou eficiência em casos de desconexão temporária, sendo capaz de reconectar automaticamente e restaurar a transmissão dos dados sem necessidade de intervenção manual.

3.2. Análise dos Resultados

O sistema demonstrou um tempo médio de resposta de praticamente 0 segundos na leitura dos sensores e no acionamento do atuador informando a mensagem de alerta. O tempo de atualização e envio dos dados foi definido em 10segundos. Qualitativamente e quantitativamente o sistema cumpriu os objetos de controle remoto e automático, demonstrando dados precisos para o monitoramento do ambiente e interfácil intuitiva e fácil de usar.

As imagens abaixo mostram o protótipo do Sistema de Monitoramento de Temperatura, Umidade e Qualidade do Ar em ambientes concluído e em funcionamento. As luzes vermelhas acesas indicam que todos os componentes estão operando de acordo com o esperado. O sensor DHT22 fazendo as medições de temperatura e umidade do ambiente, o sensor MQ-135 medindo a qualidade do ar em tempo real e ambos enviando os dados para a placa ESP32 que atua como unidade de processamento, gerenciando a comunicação com o display OLED e com a plataforma Adafruit IO via protocolo MQTT.

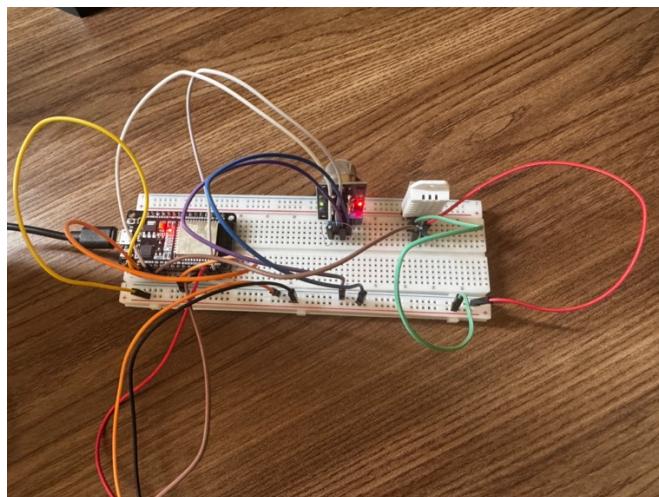


Figura 9. Protótipo em funcionamento

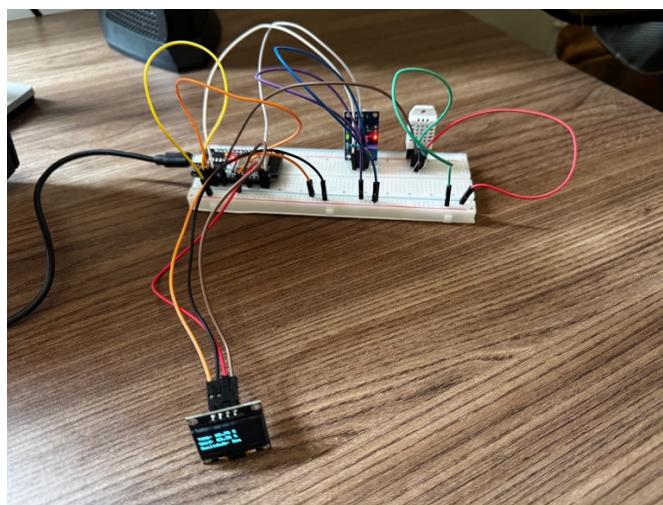


Figura 10. Protótipo em funcionamento

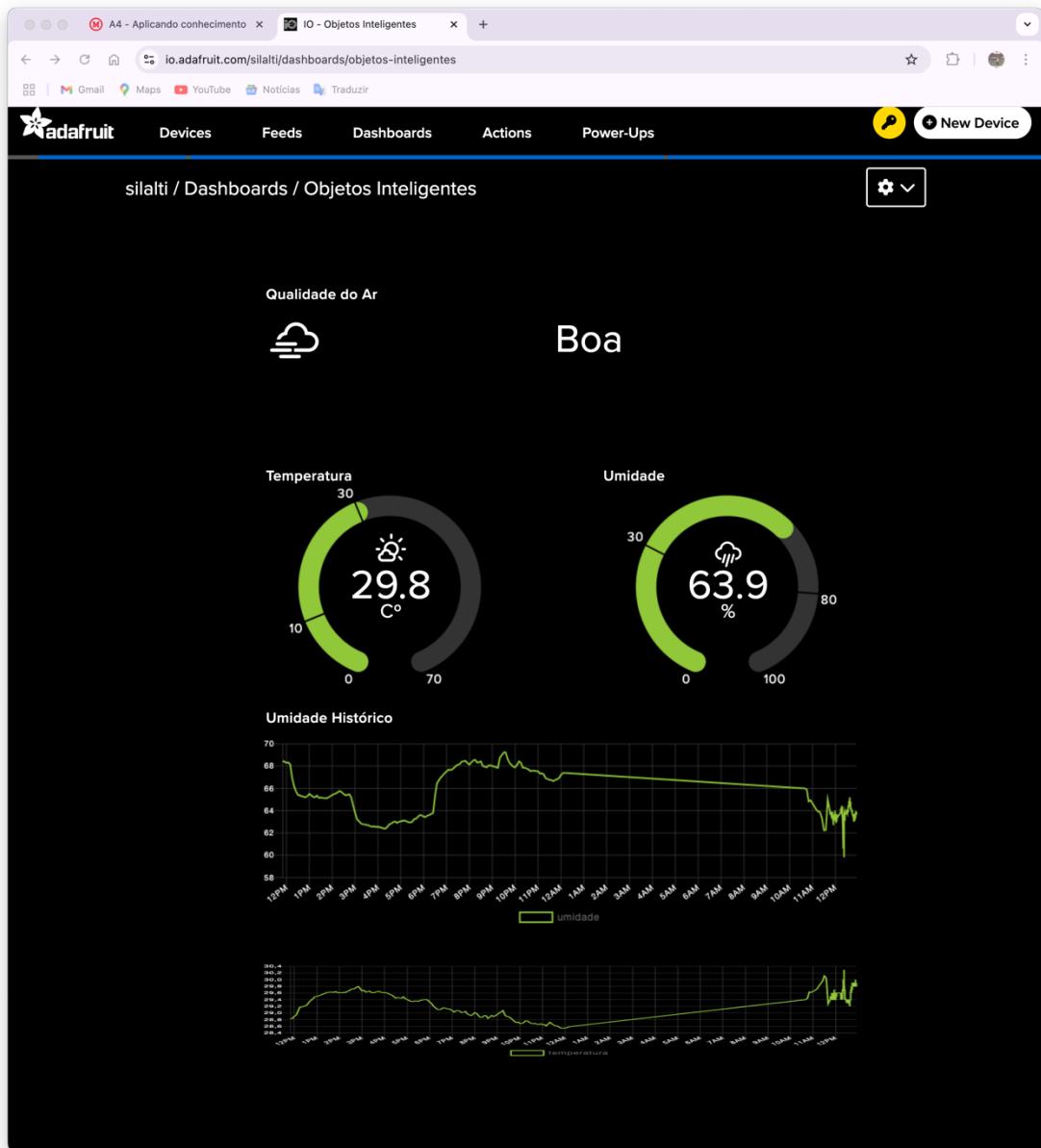


Figura 11. Captura de tela dashboard Adafruit IO

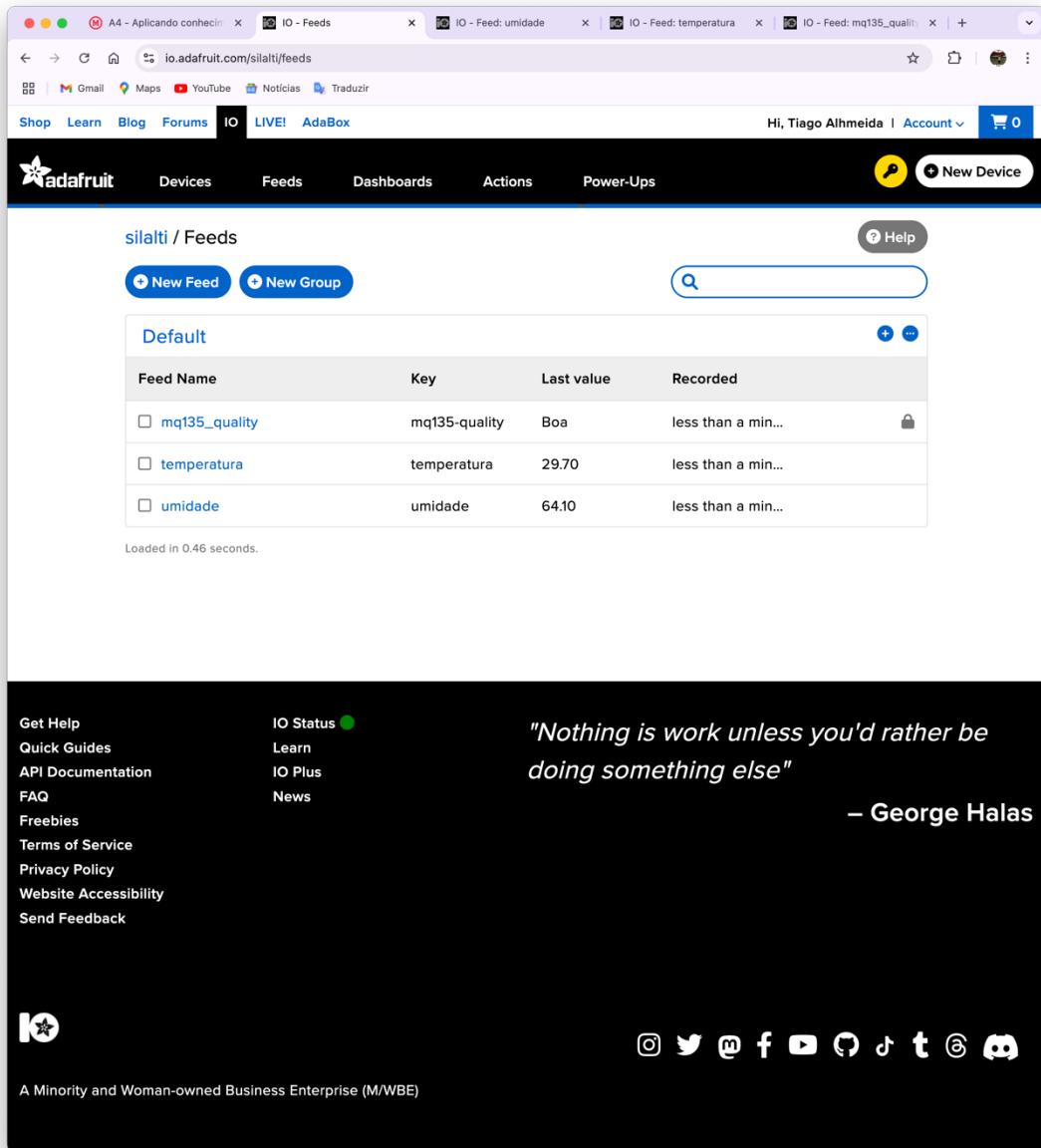


Figura 12. Feeds de Umidade, Temperatura e Qualidade do Ar

A tabela a seguir mostra os resultados dos testes do tempo médio entre o envio entre a detecção dos sensores e o recebimento dos dados na plataforma MQTT, onde foram feitas quatro medições para cada sensor e atuador.

Tabela 1. Tempo médio de resposta e de leitura dos sensores e recebimento de dados na plataforma MQTT.

| Nº. Medida | Sensor/Atuador | Tempo de resposta (s) | Tempo Leitura (us) |
|------------|----------------|-----------------------|--------------------|
| 1 | DHT22 | 10 | 5737 |
| 2 | DHT22 | 10 | 5521 |
| 3 | DHT22 | 10 | 5640 |

| | | | |
|---|--------------|----|-------|
| 4 | DHT22 | 10 | 5607 |
| 1 | MQ-135 | 10 | 27 |
| 2 | MQ-135 | 10 | 28 |
| 3 | MQ-135 | 10 | 27 |
| 4 | MQ-135 | 10 | 28 |
| 1 | Display OLED | 10 | 26130 |
| 2 | Display OLED | 10 | 26128 |
| 3 | Display OLED | 10 | 26126 |
| 4 | Display OLED | 10 | 26117 |

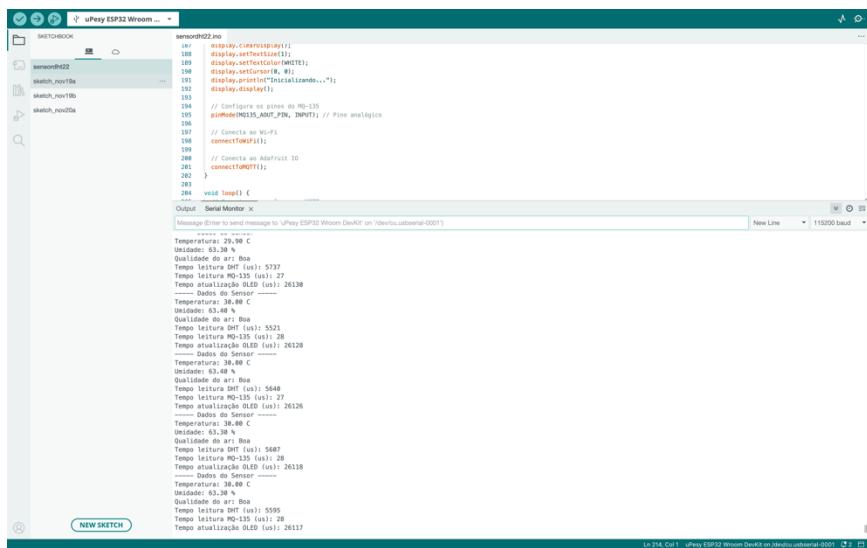


Figura 13. Dados de leitura no monitor serial Arduino IDE

3.3. Vídeo demonstrativo do protótipo em funcionamento

<https://youtu.be/RP19xWjCyK4>

3.4. Repositório no GitHub

https://github.com/Ti-Ahl/Mackenzie_iot_obj_int

4. Conclusões

4.1. Os objetivos propostos foram concluídos?

Sim, o objetivo proposto foi concluído, permitindo o monitoramento de temperatura, umidade e qualidade do ar em tempo real e de forma automatizada. A integração entre os sensores DHT22, MQ-135, do protocolo MQTT e do Adafruit aconteceu de forma funcional proporcionando um controle eficiente para as condições ambientais e cumprindo o propósito de forma confiável e com boa resposta.

4.2. Quais são os principais problemas enfrentados e como foram resolvidos?

O principal desafio foi integrar de forma eficiente diferentes componentes de hardware, como os sensores DHT22 e MQ-135, o display OLED e a placa ESP32, garantindo a coleta precisa dos dados, sua transmissão em tempo real para a plataforma Adafruit IO e a exibição clara no display local. Além disso, foi necessário lidar com questões de confiabilidade, como tempos de resposta consistentes, a detecção de condições fora dos limites aceitáveis (temperatura e umidade), e a atualização dinâmica do display com mensagens de alerta sem comprometer a usabilidade do sistema. A implementação de um monitoramento robusto, com conectividade estável em ambientes adversos e feedback visual intuitivo.

4.2. Quais vantagens e desvantagens do projeto?

O projeto apresenta como principais vantagens a capacidade de monitoramento em tempo real, precisão nas medições de temperatura, umidade e qualidade do ar, e a integração com a plataforma Adafruit IO, permitindo acesso remoto e registro histórico dos dados. A interface intuitiva do display OLED oferece feedback local imediato, enquanto os alertas para condições fora do padrão garantem maior segurança no ambiente monitorado. Entre as desvantagens, destaca-se a dependência de uma conexão estável com a internet para pleno funcionamento, o consumo energético relativamente alto do sistema em operações contínuas e a necessidade de calibração cuidadosa dos sensores para garantir a confiabilidade a longo prazo.

4.2. O que deveria/poderia ser feito para melhorar o projeto?

Seria interessante implementar uma fonte de alimentação sustentável, como painéis solares, para reduzir o consumo energético da rede elétrica. Adicionar mais sensores, como luminosidade ou presença de gases específicos, ampliaria o escopo de monitoramento. Além disso a adição de outros atuadores para intervenção em caso de medições fora dos padrões estabelecidos para sinal sonoro ou ativação de ventilação.

5. Referências

SMITH, John; JONES, Alice; BROWN, Michael. *Warehouse Humidity Monitoring System*. Journal of Industrial Automation, v. 15, n. 3, p. 45-58, 2019.

JOHNSON, David; WILLIAMS, Sarah; MILLER, Emily. *IoT-based Temperature Control Solutions in Warehousing*. International Conference on Smart Technologies, p. 102-110, 2021.

OLIVEIRA, Maria; SANTOS, João. *Avanços na Tecnologia de Sensores para Monitoramento Ambiental*. Revista Brasileira de Engenharia, v. 28, n. 2, p. 78-89, 2020.

SOUZA, Carlos; PEREIRA, Ana. *Automatização e Eficiência em Armazenagem: Uma Revisão de Tecnologias Recentes*. Congresso Brasileiro de Tecnologia e Inovação, p. 150-162, 2022.

Guia para Trabalhos Acadêmicos - Mackenzie. Disponível em: https://www.mackenzie.br/fileadmin/user_upload/Guia_Mackenzie_trabalhos_academicos_online_c_protecao.pdf. Acesso em: 20 nov. 2024.

Arduino - Official Documentation

"Arduino Cookbook" (Livro técnico por Michael Margolis, com exemplos e boas práticas de uso da Arduino IDE).

Fóruns da Comunidade Arduino (Arduino Forum) e Stack Overflow.

MQTT.org. (n.d.). *MQTT Protocol*. Disponível em: <http://mqtt.org>.

Espressif Systems. (2023). *ESP32 Datasheet*. Disponível em: <https://www.espressif.com/>.

Adafruit IO. (n.d.). *Documentation and API Reference*. Disponível em: <https://io.adafruit.com>.