**Manual de Instalação e Configuração do Sistema de Monitoramento de Gás**

**Versão: 1.1 | Data: dezembro de 2024**

**1. Introdução**

Este manual detalha o processo completo de instalação e configuração do sensor de gás MQ-2 integrado a um sistema de monitoramento que inclui coleta de dados via API, armazenamento em banco de dados e exibição em um dashboard web.

**2. Requisitos**

**2.1. Equipamentos Necessários**

**1. Hardware:**

1. Sensor de Gás MQ-2
2. Placa Arduino Uno R3
3. Fonte de alimentação para Arduino (USB conectado ao Computador)
4. Protoboard
5. Fios jumper
6. Notebook

**2. Software:**

1. Arduino IDE (última versão)
2. Node.js (para a API)
3. MySQL Server + (Mysql Workbench)
4. Virtual Machine (Opicional)
5. Visual Studio Code (HTML, CSS, Javascript, API)
6. GitHub (Versionamento do Projeto)

**Ferramentas Necessárias (ferramentas utilizadas para implantar o sensor)**

**3. Requisitos do Ambiente:**

1. Rede Wi-Fi estável para integração com API e dashboard
2. Computador ou servidor próximo ao local de instalação para processamento de dados
3. Entrada de gás livre para implantação do Arduino R3 (MQ2, Protoboard)

**4. Procedimento de Instalação**

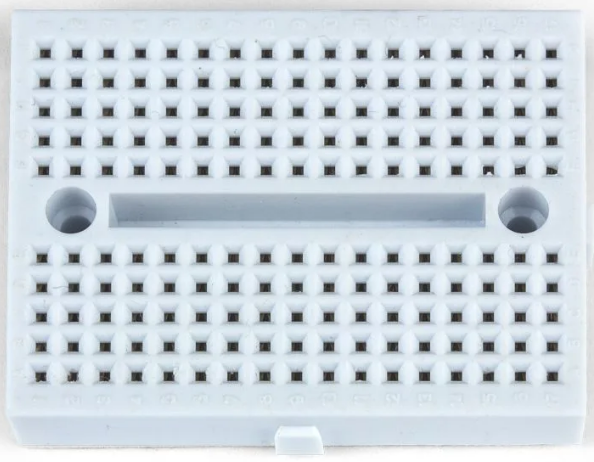
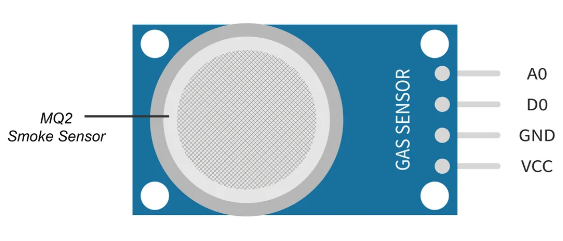
**5.1. Preparação do Local**

Escolha um local elevado e limpo para instalar o sensor.

(Imagem de um exemplo de cozinha para a instalação)

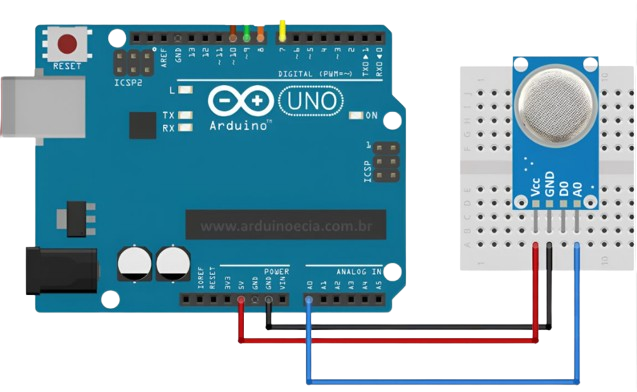
(Garanta fácil acesso ao sensor para manutenção.)

**5. Conexões com o Sensor MQ-2**

1. Conecte o sensor à protoboard.

**6. Faça as conexões dos fios (Jumpers):**

1. VCC do sensor ao 5V do Arduino. (Jumper Red)
2. GND do sensor ao GND do Arduino. (Jumper Black)
3. (A0 OUT) para a A0 do Arduino. (Jumper Blue**)**

**(**Verifique que todas as conexões estejam corretas e firmes.)

**7. Configuração do Código no Arduino**

1. Instale a IDE (Arduino IDE) <https://www.arduino.cc/en/software>
2. Inicie a IDE

**Interface gráfica do usuário, Texto

Descrição gerada automaticamente**

1. Escreva o Código e compile para verificar

**Texto

Descrição gerada automaticamente**

**CODIGO   
“**

**const int PINO\_SENSOR\_MQ2 = A0;**

**const int VALOR\_MINIMO = 100;**

**const int VALOR\_MAXIMO = 1000;**

**// Configuração Inicial:**

**void setup(){**

**Serial.begin(9600);**

**}**

**//Loop Principal:**

**void loop(){**

**int valorSensor = analogRead(PINO\_SENSOR\_MQ2);**

**float porcentagem = ((float)(valorSensor - VALOR\_MINIMO) / (VALOR\_MAXIMO - VALOR\_MINIMO)) \* 100;**

**if (porcentagem < 0){**

**porcentagem = 0;**

**} else if(porcentagem > 100){**

**porcentagem = 100;**

**}**

**Serial.println(porcentagem);**

**delay(1000);**

**}**

**”**

**8. Conexão do Arduino via USB a o computador para a transferência do código**

1. Conecte o arduino montado ao computador via USB
2. Transfira o código para o arduino conectado (Certifique que o código tenha sido executado antes de transferir, para evitar problemas no código).
3. Verificar se o led do sensor está ligado (led red), após a transferência do código.

**9. Crie a API para enviar dados do sensor ao banco**

1. Faça o download do Git Bash pelo link <https://git-scm.com/book/pt-br/v2/Come%C3%A7ando-Instalando-o-Git>
2. Execute e instale no notebook conectado com o Arduino. Após a instalação.
3. Crie um repositório no notebook e dentro do repositório clique com o botão esquerdo e selecione o git bash.

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

1. No terminal do git bash, faça o git clone do: [BandTec/dat-acqu-ino: Data Acquisition Arduino API](https://github.com/BandTec/dat-acqu-ino) (API)

Texto

Descrição gerada automaticamente

1. Acesse o diretório pelo terminal usando o ‘cd dat-acqu-ino’

Texto

Descrição gerada automaticamente

1. Faça a instalação do Modulos, usando o comando ‘npm install’

Texto

Descrição gerada automaticamente

**10. Configurar a API junto ao banco de dados:**

1. Configure o Banco de Dados MySQL
2. Instale o MySQL Server em seu computador.
3. Crie um banco de dados para armazenar os dados do sensor com o comando:

**CREATE DATABASE gasbusters;**

1. No banco de dados, crie a tabela para armazenar os dados do sensor:

**CREATE TABLE medidas (**

**Id int primary key auto\_increment,**

**sensor float**

**);**

* 1. **Configuração do Ambiente:**

1. instale o Visual Studio Code e configure o ambiente de desenvolvimento com HTML, CSS e JavaScript.
2. Atualize as Configurações da API, usando o VS Code, no diretório da API (dat-acqu-ino), localize o arquivo de configuração do banco de dados (main.js).
3. Edite o arquivo para incluir as credenciais do banco de dados, como no exemplo abaixo:

Texto

Descrição gerada automaticamente

**EXEMPLO:  
“**

**host: 'localhost',**

**user: 'gasbusters',**

**password: 'Busters#gas',**

**database: 'gasbusters',**

**port: 3306**

**”**

1. Teste a Conexão com o Banco de Dados
2. Execute o servidor da API, no terminal com o comando:

**“**

**npm start**

**“**

1. Verifique no terminal se a conexão com o banco foi estabelecida com sucesso.
2. Verifique o Envio de Dados
3. Abra o navegador e acesse o endpoint da API configurado (exemplo: <http://localhost:3300/sensores>).
4. Confirme se os dados do sensor estão sendo registrados no banco de dados.

**11. Finalização e Testes**

1. **Teste Completo do Sistema**
   1. Verifique se os dados do sensor estão sendo lidos corretamente.
   2. Confirme que os dados estão sendo enviados à API e armazenados no banco de dados.
   3. Assegure-se de que o dashboard exibe as informações em tempo real.
   4. Ofereça um meio de suporte técnico para dúvidas e problemas adicionais.

**11. Manutenção e Atualização**

1. Limpeza regular do sensor para evitar acumulação de poeira.
2. Atualizações do código e integração com o banco de dados conforme novas versões forem lançadas.

**12. Contato para Suporte**

**E-mail: suporte@gasbusters.com**

**Telefone: (11) 99999-9999**