

Relatório Técnico

Nº Grupo: 01

Nome dos integrantes: Gustavo Pietro, Kauan Kadooca, Nattalia Siqueira, Pedro Souza, Victor Gastardeli, Vitor Anzai

Turma: 1CCOK

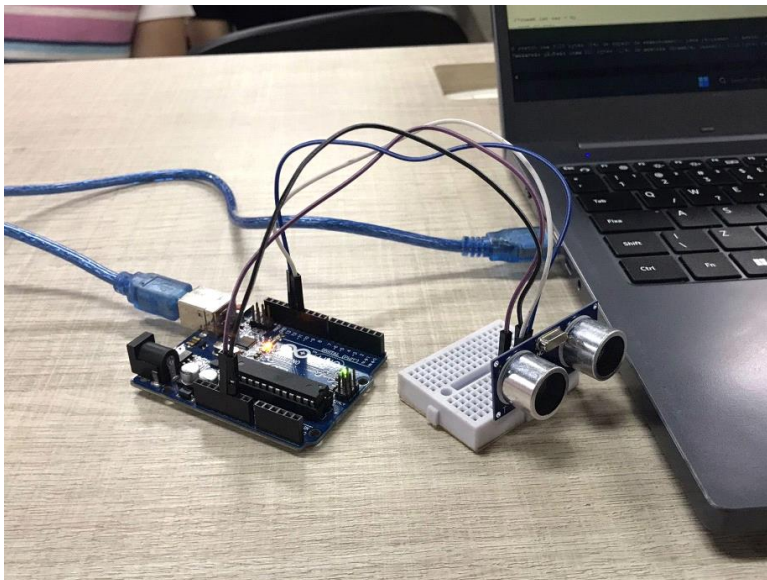
Tema do projeto: Gestão de estoque de atacarejo

Sensor: HC-SR04

Introdução:

O sistema de monitoramento tem como objetivo reduzir falhas e perdas de inventário em atacarejos utilizando um sensor ultrassônico, que será responsável por registrar a distância que os produtos armazenados se encontram, para que haja uma melhor verificação e controle de estoque.

Arquitetura de Montagem do Sensor:

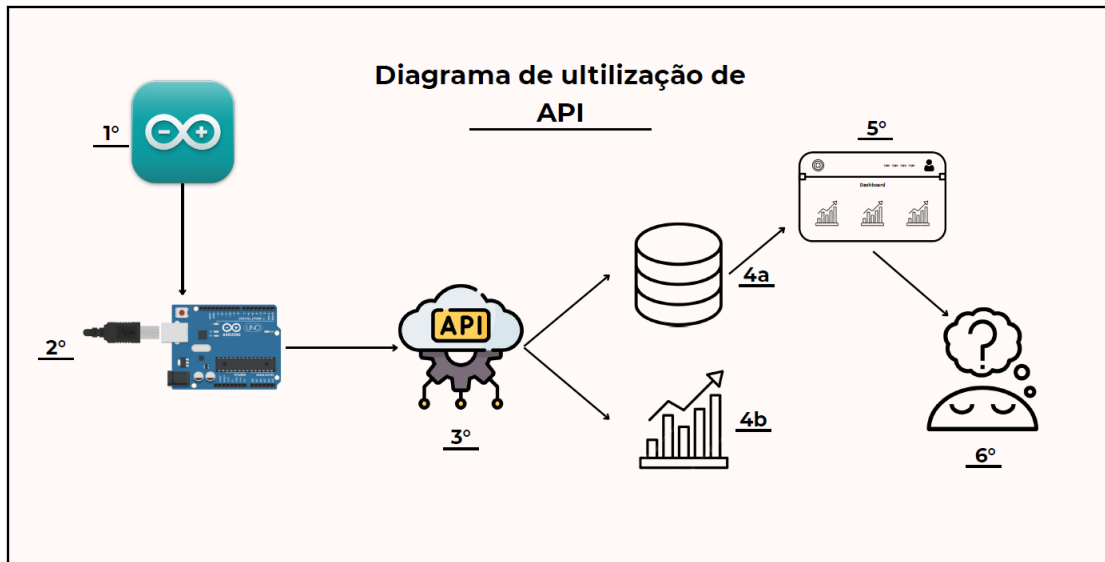


Nossa montagem acompanha:

- Quatro cabos jumper;
- Um cabo USB;
- Um sensor ultrassônico;
- Duas portas analógicas;
- Uma porta de aterramento;
- Uma porta de tensão;

- Uma mini protoboard;
- Um sensor HC-SR04.

Arquitetura do Sistema



Estrutura de funcionamento:

1º - Compilar código no Arduino IDE:

Utilização da API e compilação do código no Arduino, para realizar a captura dos dados.

2º - Coleta de dados:

O sensor HC-SR04 irá coletar os dados do ambiente.

3º - Arduino até a API:

Compartilhamento dos dados com a API, podendo seguir dois caminhos:

4a – Visualização desses dados na dashboard presente no próprio código da API.

4b - Envio para o banco de dados do projeto.

5º - Utilizar o BD no site:

Utilização dos dados para a exibição na dashboard HTML do projeto.

6º - Visualização:

Visualizar em tempo real a exibição dos dados coletados pelo sensor.

Código do Projeto

```
// habilita ou desabilita a inserção de dados no banco de dados
const HABILITAR_OPERACAO_INSERTIR = true;
```

```
// função para comunicação serial
const serial = async (
  valoresSensorAnalogico,
  valoresSensorDigital,
) => {
  // conexão com o banco de dados MySQL
  let poolBancoDados = mysql.createPool(
    {
      host: 'localhost',
      user: 'aluno',
      password: 'Sptech#2024',
      database: 'grupo1',
      port: 3307
    }
  ).promise();
```

As alterações mais significativas no código foram realizadas nos blocos de inserção no Banco de Dados, onde foi necessário habilitar a operação de inserção via código e adicionar o usuário “Aluno” do MySQL da Máquina Virtual “Lubuntu”, para que a API pudesse se comunicar com o MySQL.

```
// este insert irá inserir os dados na tabela "medida"
await poolBancoDados.execute(
  'INSERT INTO registro (sensorDigital) VALUES (?)',
  [sensorDigital]
);
console.log("valores inseridos no banco: " + sensorDigital + "cm");
}
```

Alteramos a tabela de acordo com o Banco de Dados do nosso projeto e removemos a variável “sensorAnalogico”, afim de adequar o código ao nosso projeto.

Por fim, comentamos toda a estrutura de leitura analógica do HTML, removendo partes menos essenciais, como a Label “Centímetros”.

```
var sensorDigital = new Chart(document.getElementById('sensorDigital').getContext('2d'), {
  type: 'line',
  data: {
    datasets: [{
      label: 'Centímetros',
      borderColor: '#63B1BC',
      backgroundColor: '#0762C8'
    }]
  }
});
```

Resultados Iniciais

Conseguimos coletar e armazenar os dados coletados pelo Arduino diretamente no Banco de Dados com sucesso através da API configurada, podendo assim visualizar esses dados através de um gráfico configurado previamente. Realocamos a importância do Arduino IDE, que agora tem a única função de compilar o código no Arduino para a API realizar suas operações.