

## Relatório Técnico

**Nº Grupo:** 02

**Nome dos integrantes:** Robson Guilherme, Daniel Foschini, João Coimbra, Lavinia Todarelli, Roger Elias, Samuel Lorenço.

**Turma:** 1ADSB

**Tema do projeto:** Fluxo de pessoas em shopping centers e centros comerciais.

**Sensor:** HC-SR04 (ultrassônico)

### Introdução

O projeto tem como objetivo desenvolver um sistema de monitoramento de fluxo de pessoas em shopping centers, utilizando sensores ultrassônicos integrados a uma plataforma Arduino. A proposta visa registrar e analisar o movimento de visitantes em tempo real, fornecendo dados que possam auxiliar na definição de estratégias comerciais, como o ajuste de valores de aluguel conforme o fluxo de público em cada entrada do shopping.

### Arquitetura de Montagem do Sensor

- Para nossa montagem utilizamos 4 jumpers
- Uma protoboard
- Um Microcontrolador
- Um sensor HC-SR04 (ultrassônico)

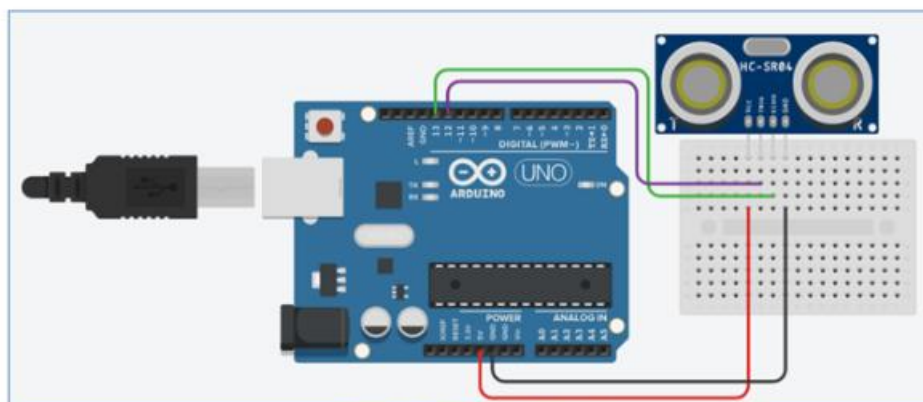
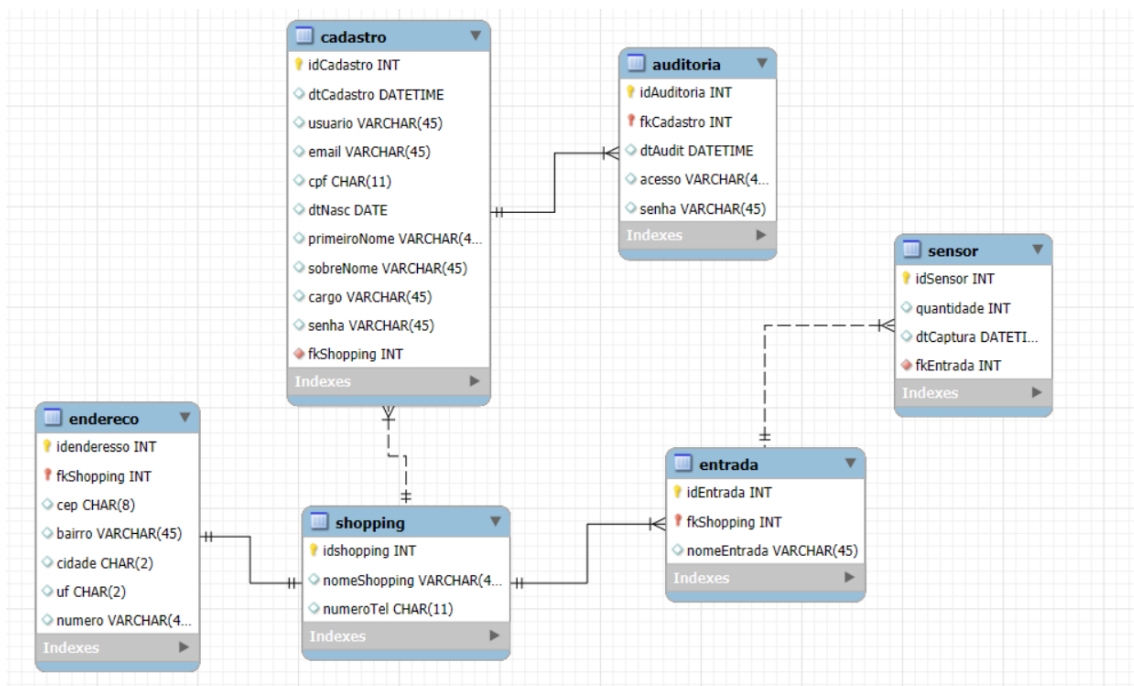
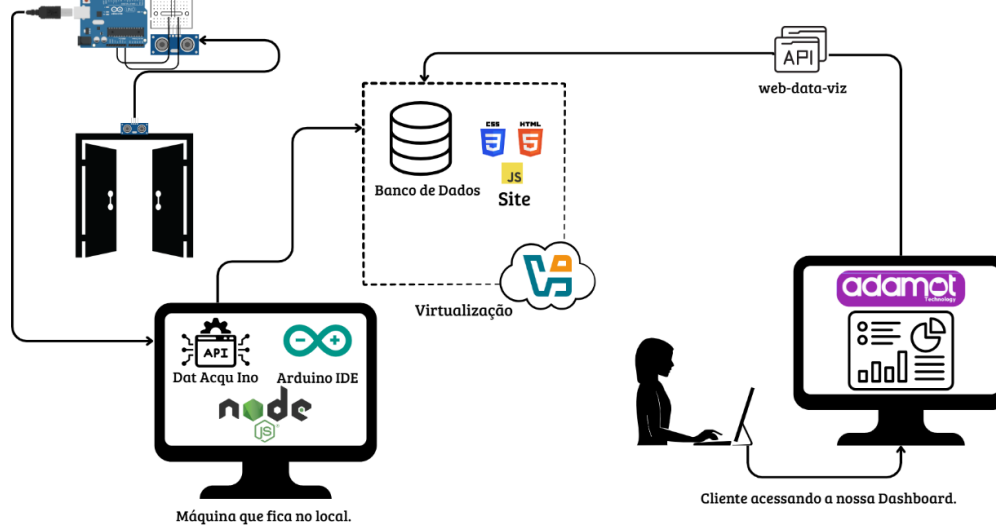


Figura 2 - Arquitetura HC-SR04



## Arquitetura do Sistema

### Diagrama de Solução Técnica



## Código do Projeto

### Código arduino IDE:

```
1  #include <Ultrasonic.h>
2
3  int pinoTrigger = 7;
4  int pinoEcho = 8;
5  HC_SR04 sensor1(7,8); // (trigger, echo)
6
7  void setup() {
8      Serial.begin(9600);
9  }
10
11  void loop() {
12
13      float distancia = sensor1.distance();
14
15      if(distancia > 0 && distancia < 10) {
16          Serial.println("1");
17      } else {
18          Serial.println("0");
19      }
20
21      delay(1000);
22  }
```

### Node JS- dat-acqu-ino

```
// conexão com o banco de dados MySQL
let poolBancoDados = mysql.createPool({
  {
    host: '127.0.0.1',
    user: 'api',
    password: 'senhaapi',
    database: 'adsb2',
    port: 3306
  }
}).promise();
```

```

61 // insere os dados no banco de dados (se habilitado)
62 if (HABILITAR_OPERACAO_INSERIR) {
63
64     // este insert irá inserir os dados na tabela "medida"
65     await poolBancoDados.execute(
66         'INSERT INTO registro (valor, dtHora) VALUES (?, ?)',
67         [sensorDigital, dataHora]
68     );
69     console.log("valores inseridos no banco:" + sensorDigital + ", " + dataHora);
70 }

```

```

✓ function pegarData() {

    var dataAtual = new Date;
    dataAtual = dataAtual.toString().split(' ');
    var ano = dataAtual[3];
    var dia = dataAtual[2];
    var mes = dataAtual[1] === 'Jan' ? '01' :
    dataAtual[1] === 'Feb' ? '02' :
    dataAtual[1] === 'Mar' ? '03' :
    dataAtual[1] === 'Apr' ? '04' :
    dataAtual[1] === 'May' ? '05' :
    dataAtual[1] === 'Jun' ? '06' :
    dataAtual[1] === 'Jul' ? '07' :
    dataAtual[1] === 'Aug' ? '08' :
    dataAtual[1] === 'Sep' ? '09' :
    dataAtual[1] === 'Oct' ? '10' :
    dataAtual[1] === 'Nov' ? '11' : '12' ;
    let hora = dataAtual[4];
    var dataCompleta = `${ano}-${mes}-${dia}-${hora}`

    return dataCompleta;

}

```

**Obs: adicionamos a função pegarData;**

## Resultados Iniciais

```
1 • create database adamot;
2
3 • use adamot;
4
5 • create table registro(
6     id int primary key auto_increment,
7     valor int,
8     dtHora datetime
9 );
10
11 • select valor ValorRegistrado, dtHora HoraRegistro from registro;
```

ValorRegistrado	HoraRegistro
1	2025-10-09 09:41:48
1	2025-10-09 09:41:49
0	2025-10-09 09:41:50
0	2025-10-09 09:41:51
0	2025-10-09 09:41:52

```
61 // insere os dados no banco de dados (se habilitado)
62 if (HABILITAR_OPERACAO_INSERIR) {
63
64     // este insert irá inserir os dados na tabela "medida"
65     await poolBancoDados.execute(
66         'INSERT INTO registro (valor, dtHora) VALUES (?, ?)',
67         [sensorDigital, dataHora]
68     );
69     console.log("valores inseridos no banco:" + sensorDigital + ", " + dataHora);
70 }
71
72 });
73
74 // evento para lidar com erros na comunicação serial
75 arduino.on('error', (mensagem) => {
76     console.error("Erro no arduino (Mensagem: " + mensagem + ")");
77 });
78 }
79
80
81 // função para criar e configurar o servidor web
```

PROBLEMAS SAÍDA CONSOLE DE DEPURACÃO TERMINAL PORTAS HISTÓRICO DE SQL MONITOR DE TAREFAS

```
windows@DESKTOP-FNE13L2 MINGW64 ~/Downloads/ArqComp/API - ArqComp/dat-acqu-ino
$ npm start
0
valores inseridos no banco:0, 2025-10-09-09:42:24
0
valores inseridos no banco:0, 2025-10-09-09:42:25
0
valores inseridos no banco:0, 2025-10-09-09:42:26

windows@DESKTOP-FNE13L2 MINGW64 ~/Downloads/ArqComp/API - ArqComp/dat-acqu-ino
$
```