



Universidade de Araraquara – Uniara
Curso de Sistemas de Informação

FELIPE NARDACIONI RA:04723-301

LUAN QUALHARELI SEGURA RA:04723-303

JOÃO MIGUEL BASILE MACHADO RA:04723-014

VICTOR RODRIGUES VILLALON RA:04723-023

VINICIUS MARIANO DE OLIVEIRA RA: 05223-056

Alerta Cidadão IoT: Guia de Replicação de um Sistema de Monitoramento de Nível de Risco Ambiental

Araraquara–SP

2025

1. Introdução e Contexto do Projeto

O presente projeto de extensão visa demonstrar, por meio de um protótipo simulado, a aplicação prática e acessível da Internet das Coisas (IoT) na mitigação de riscos sociais. O sistema desenvolvido, denominado "Alerta Cidadão IoT", simula o monitoramento contínuo do nível de água (em um rio ou reservatório), acionando um alerta visual quando a condição de risco é atingida.

O projeto cumpre o papel educativo ao fornecer um Guia de Replicação completo, permitindo que iniciantes em eletrônica e programação compreendam o ciclo de funcionamento de um dispositivo IoT: sensorização, processamento e atuação.

1.1. Objetivos do Projeto

- **Objetivo Geral:** Desenvolver e documentar um protótipo simulado de monitoramento IoT, fornecendo um guia didático para facilitar a replicação e o entendimento de tecnologias aplicadas à segurança comunitária.
- **Objetivo Específico:** Fornecer um tutorial passo a passo utilizando a plataforma Wokwi (simulador online) para a montagem do circuito, programação e teste da lógica de alerta.

2. Descrição Técnica e Conceitual

2.1. O Conceito IoT no Projeto

O "Alerta Cidadão IoT" opera como um sistema de borda (Edge Computing), onde a decisão de alerta é tomada localmente no microcontrolador, garantindo uma resposta imediata (baixo *latency*).

- Coleta de Dados (Sensorização): O Sensor Ultrassônico HC-SR04 é o elemento de entrada, que coleta dados ambientais (distância, que simula o nível).
- Processamento Local: O Arduino Uno executa o código C++, convertendo a leitura do sensor em centímetros e comparando o valor com o limite de risco
- Atuação: Se a condição de risco for verdadeira, o LED é acionado e o Display LCD exibe a mensagem de alerta.

2.2. Componentes (Virtuais)

Componente	Função no Circuito	Pinos Utilizados
Arduino Uno R3	Processamento e Controle Lógico	-
Sensor Ultrassônico HC-SR04	Mede a Distância (Nível)	Digital D8 e D9
Display LCD 16x2 I2C	Exibe o Status do Nível	Analógico A4 e A5
LED Vermelho e Resistor	Atuador de Alerta Visual	Digital D10

3. Guia de Replicação Passo a Passo

O projeto pode ser replicado na íntegra através da plataforma de simulação gratuita [Wokwi](#).

3.1. Montagem do Circuito (Fiação)

Para garantir o funcionamento, todas as conexões de alimentação, terra e dados devem ser feitas rigorosamente, conforme a tabela. Utilizamos a comunicação I2C para otimizar a fiação do display.

 Tabela Completa de Conexões do Circuito

Passo	Componente	Pino Componente no	Conecta ao Pino (Arduino Uno)	Tipo de Fio / Finalidade
1	Energia	VCC (LCD e Sensor)	5V	Alimentação do Circuito (5 Volts)
2	Terra	GND (todos)	GND	Terra
3	Sensor HC-SR04	Trig	D9 (Digital)	Sinal de Disparo
4	Sensor HC-SR04	Echo	D8 (Digital)	Sinal de Retorno
5	Display LCD I2C	SDA (Dados)	A4 (Analógico 4)	Comunicação I2C
6	Display LCD I2C	SCL (Clock)	A5 (Analógico 5)	Comunicação I2C
7	LED de Alerta	Ânodo (+) (Via Resistor)	D10 (Digital)	Saída de Alerta Controlada

3.2. Código C++ (Lógica de Risco)

O código é responsável por: a) incluir as bibliotecas I2C, b) calcular a distância em centímetros e c) acionar a lógica de alerta se a `distanciaCM` for menor ou igual a 10.0 cm.

C++

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

const int TRIG_PIN = 9;
const int ECHO_PIN = 8;
const int LED_ALERTA = 10;

LiquidCrystal_I2C lcd(0x27, 16, 2);

const float DISTANCIA_RISCO_CM = 10.0;

long duracao;
float distanciaCM;

void setup() {
    Serial.begin(9600);

    pinMode(TRIG_PIN, OUTPUT);
    pinMode(ECHO_PIN, INPUT);
    pinMode(LED_ALERTA, OUTPUT);

    lcd.init();
    lcd.backlight();

    lcd.print("Alerta Cidadao");
    lcd.setCursor(0, 1);
    lcd.print("Monitorando...");
    delay(2000);
    lcd.clear();
}

void loop() {
    digitalWrite(TRIG_PIN, LOW); delayMicroseconds(2);
    digitalWrite(TRIG_PIN, HIGH); delayMicroseconds(10);
    digitalWrite(TRIG_PIN, LOW);

    duracao = pulseIn(ECHO_PIN, HIGH);
    distanciaCM = duracao * 0.0343 / 2;

    Serial.print("Distancia: ");
}
```

```

Serial.print(distanciaCM);
Serial.println(" cm");

lcd.clear();
lcd.setCursor(0, 0);

lcd.print("Nível: ");
lcd.print(distanciaCM, 1);
lcd.print(" cm");

if (distanciaCM <= DISTANCIA_RISCO_CM && distanciaCM > 0) {
    lcd.setCursor(0, 1);
    lcd.print("!!! ALERTA RISCO !!!");
    digitalWrite(LED_ALERTA, HIGH);
} else {
    lcd.setCursor(0, 1);
    lcd.print("Nível OK.");
    digitalWrite(LED_ALERTA, LOW);
}

delay(1000);
}

```

4. Demonstração e Evidências Finais

4.1. Teste de Funcionamento

O teste de funcionamento deve ser realizado no Wokwi, utilizando o controle deslizante (slider) do Sensor Ultrassônico para simular o nível da água.

1. Status Normal: Distância ajustada para valores altos (ex: \$400 \text{ cm}\$). O LCD deve exibir "Nível OK." e o LED Vermelho deve estar apagado.
2. Status de Risco: Distância ajustada para valores abaixo de \$10,0 \text{ cm}\$\$. O LCD deve exibir "!!! ALERTA RISCO !!!" e o LED Vermelho deve acender.

4.2. Evidência de Replicação (Link)

O projeto simulado pode ser acessado, testado e replicado imediatamente através do link direto do Wokwi, o que facilita o estudo e a validação do conceito IoT.

Link do Protótipo Funcional no Wokwi:

<https://wokwi.com/projects/448868914343538689>