

# Sistema de Monitoramento de lixo na Amazônia: IoT com ESP8266



# Integrantes da Equipe



**Eduardo Elias  
Nunes**



**Douglas  
Rosas dos  
Santos**



**Emanuel Tiago  
De Souza Da  
Silva**

# Introdução

**Problema:** Nas últimas décadas a geração de resíduos sólidos vem aumentando significativamente devido ao desenvolvimento das cidades, onde grande parte desses resíduos são descartados ou incinerados de forma incorreta gerando assim prejuízos ao meio ambiente.

**Justificativa:** A elaboração de um sistema de monitoramento de lixo baseado em IoT, torna-se fundamental para redução dos impactos causados por resíduos sólidos descartados incorretamente, bem como o projeto promove o incentivo da utilização de tecnologias para conscientização e solução de problemas ambientais na Amazônia.

# Objetivos

## Geral:

- Desenvolver um produto capaz de identificar o nível de resíduo em uma lixeira, usando IoT e um microcontrolador ESP8266.

## Específicos:

- Estudos sobre microcontroladores e microprocessadores e implementações;
- Realização de pesquisas sobre Internet das coisas e aplicações em problemas;
- Elaboração de um produto voltado para sustentabilidade na Amazônia

# Metodologia

- Foram realizadas pesquisas bibliográficas em artigos, livros e sites;
- Validou-se cada componente do sistema para verificar se o mesmo estava funcionando adequadamente;
- Montagem dos circuitos iniciais;
- Elaboração do Firmware; (GitHub)
- Simulações no Wokwi;
- Implementação do sistema. (funcionamento na prática)



# Esboço da Solução

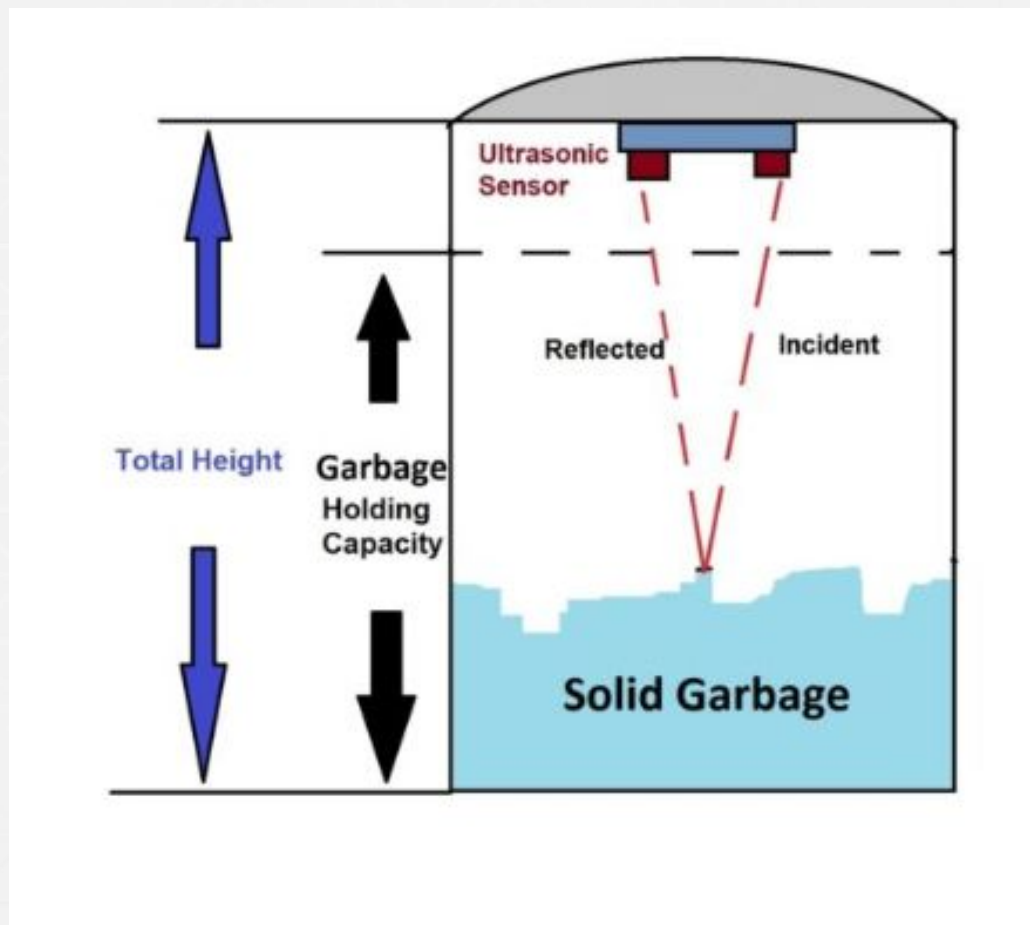
Sistema de Monitoramento de Lixo Baseado em IoT.

É uma tecnologia emergente que é utilizada para monitorar o nível de preenchimento de resíduos de lixeiras públicas e industriais. O objetivo fundamental de um sistema de monitoramento do nível de lixo é ajudar os serviços municipais a recolher o lixo no momento certo antes que uma lixeira transborde e cause desconforto ao público em geral.

Nosso projeto visa pequenos pontos e ainda um protótipo inicial.

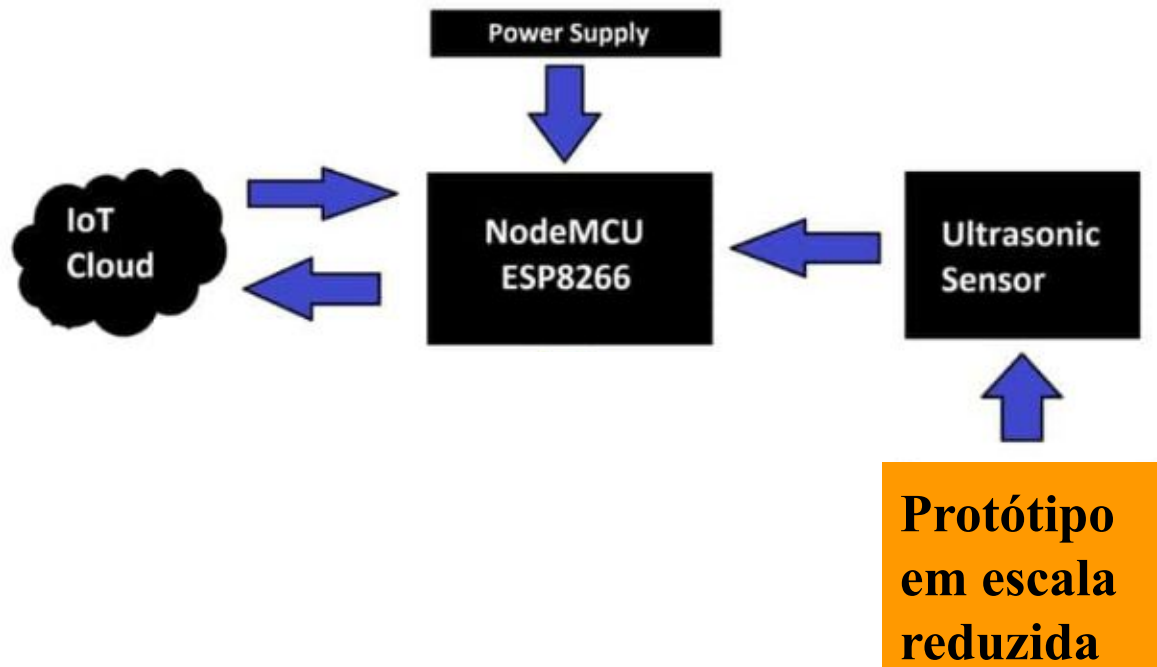
Vamos utilizar Buzzer e um Led que serão utilizados para alertar sobre o nível e conexão. O EXP8266 para fazer a análise dos dados e, posteriormente, lançados para um servidor (Celular) o qual irá apresentar todos os dados coletados para que o operador possa visualizar e acompanhar o nível da lixeira.

# Detecção do nível do lixo



FONTE: Adaptado de (ELETROINICS PROJECT HUB - 2022)

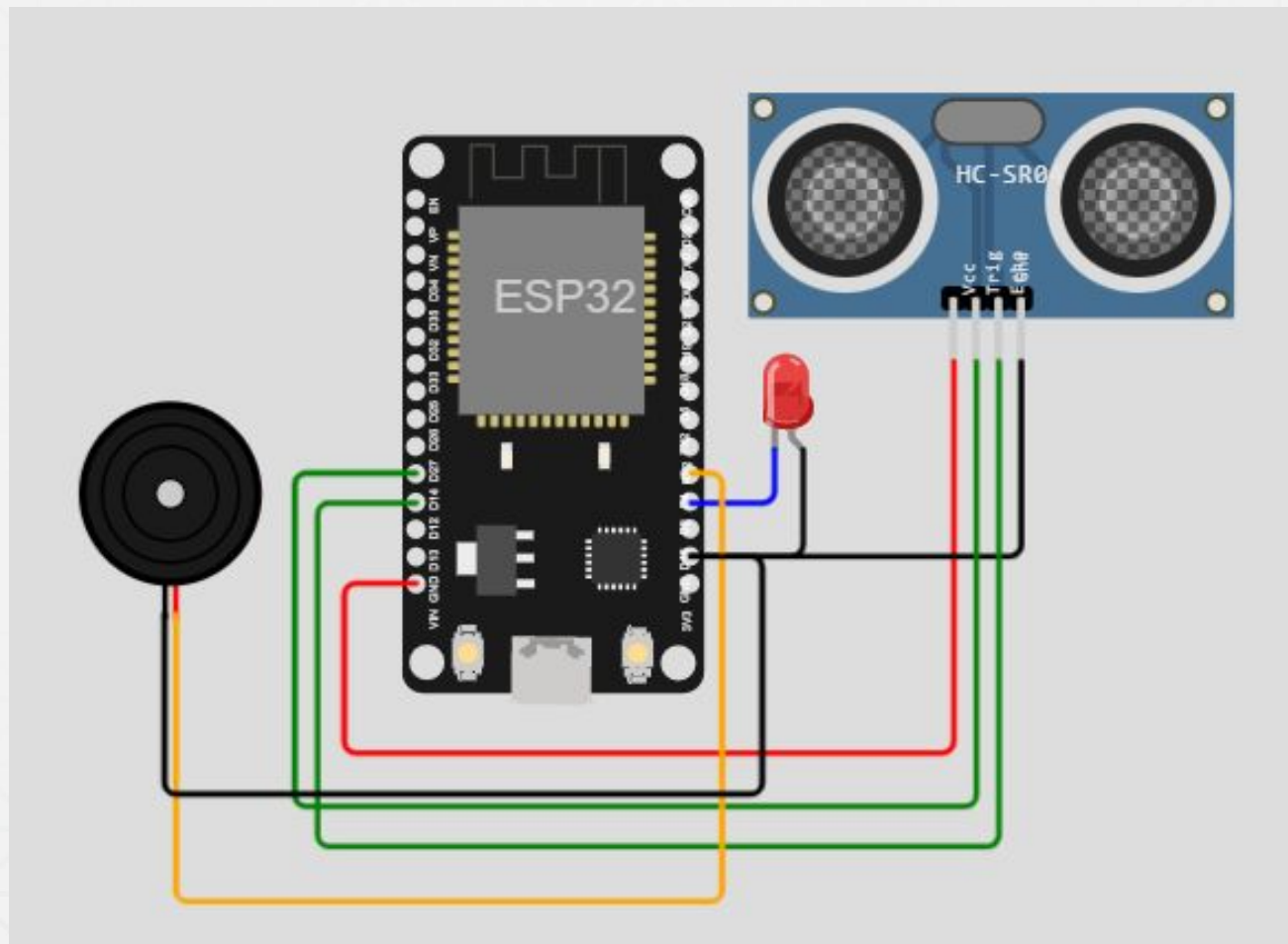
# Diagrama do Projeto



(C) Electronics-project-hub.com



# Esquema do Circuito



FONTE: AUTORES

# Simulação

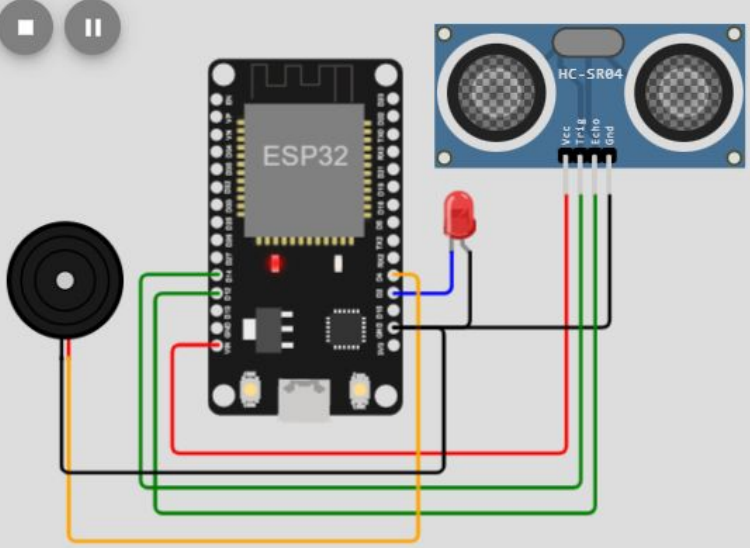
WOKWI

SAVE SHARE

sketch.ino • diagram.json • Library Manager

```
24 void setup()
25 {
26   Serial.begin(115200);
27   pinMode(LED_BUILTIN, OUTPUT);
28   pinMode(trigger, OUTPUT);
29   pinMode(echo, INPUT);
30   snsr_to_max = total_height - hold_height;
31   delay(2500);
32 }
33 void loop()
34 {
35   measure();
36
37   Serial.print("NIVEL DA LIXEIRA:");
38   Serial.print(bin_lvl);
39   Serial.println("%");
40
41   for (i = 0; i < Minute; i++)
42   {
43     Serial.println("-----");
44     Serial.println("System Standby...");
45     Serial.print(i);
46     Serial.println(" Minutes elapsed.");
47     delay(20000);
48     delay(20000);
49     delay(20000);
50   }
51 }
```

Simulation



NIVEL DA LIXEIRA:32%

-----

System Standby...

0 Minutes elapsed.

FONTE: AUTORES

# Simulação

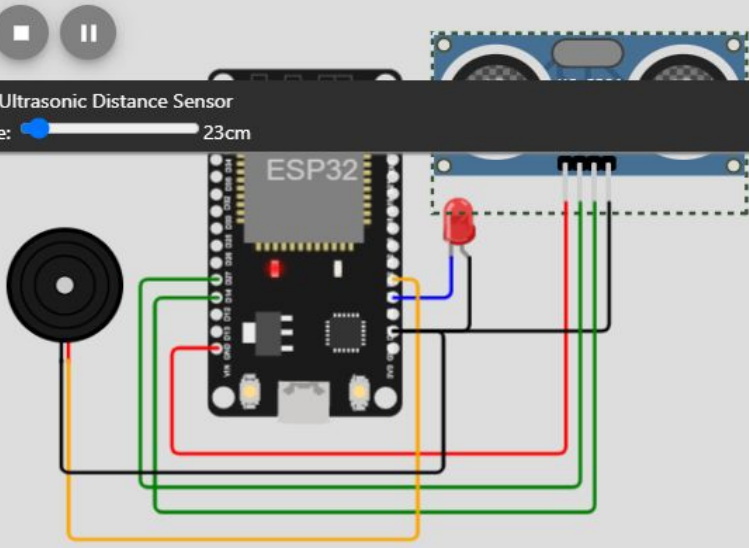
WOKWI SAVE SHARE

sketch.ino • diagram.json • Library Manager

```
24 void setup()
25 {
26   Serial.begin(115200);
27   pinMode(LED_BUILTIN, OUTPUT);
28   pinMode(trigger, OUTPUT);
29   pinMode(echo, INPUT);
30   snsr_to_max = total_height - hold_height;
31   delay(2500);
32 }
33 void loop()
34 {
35   measure();
36
37   Serial.print("NIVEL DA LIXEIRA:");
38   Serial.print(bin_lvl);
39   Serial.println("%");
40
41   for (i = 0; i < Minute; i++)
42   {
43     Serial.println("-----");
44     Serial.println("System Standby...");
45     Serial.print(i);
46     Serial.println(" Minutes elapsed.");
47     delay(20000);
48     delay(20000);
49     delay(20000);
50   }
51 }
```

Simulation

Editing Ultrasonic Distance Sensor  
Distance:  23cm



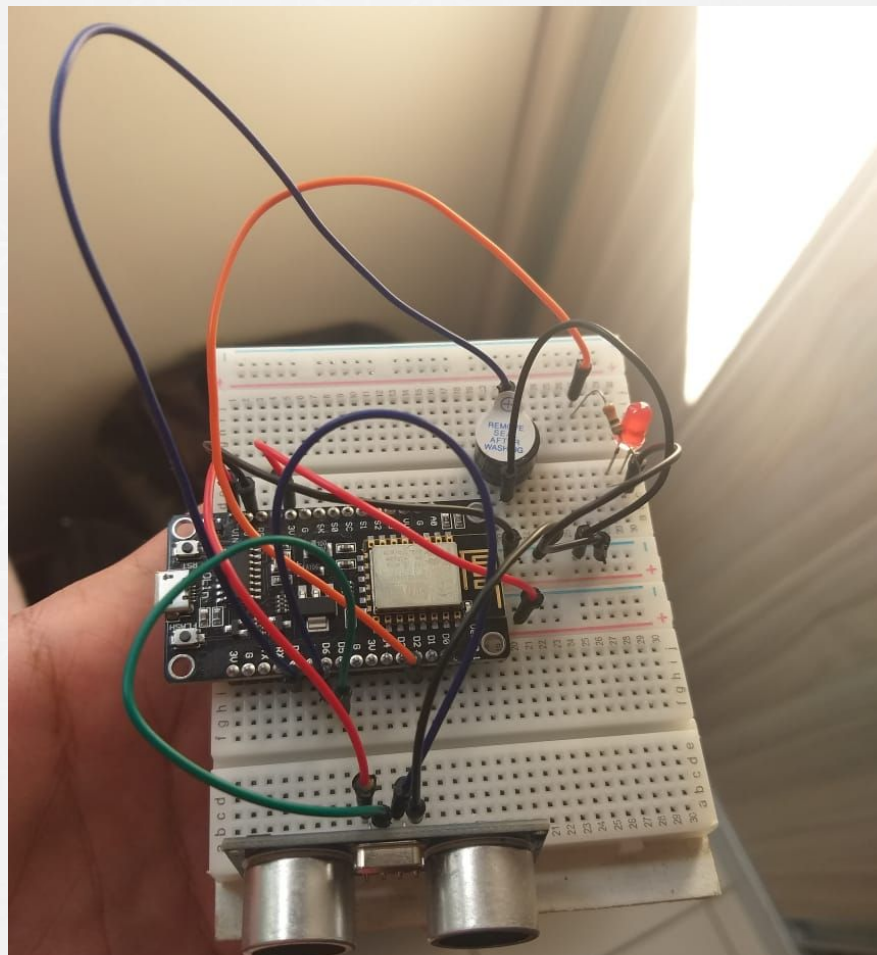
NIVEL DA LIXEIRA:32%

-----

System Standby...

0 Minutes elapsed.

# Montagem do Circuito



FONTE: AUTORES



# Firmware do projeto

Sistema\_de\_Monitoramento\_de\_lixo | Arduino 1.8.16

Arquivo Editar Sketch Ferramentas Ajuda



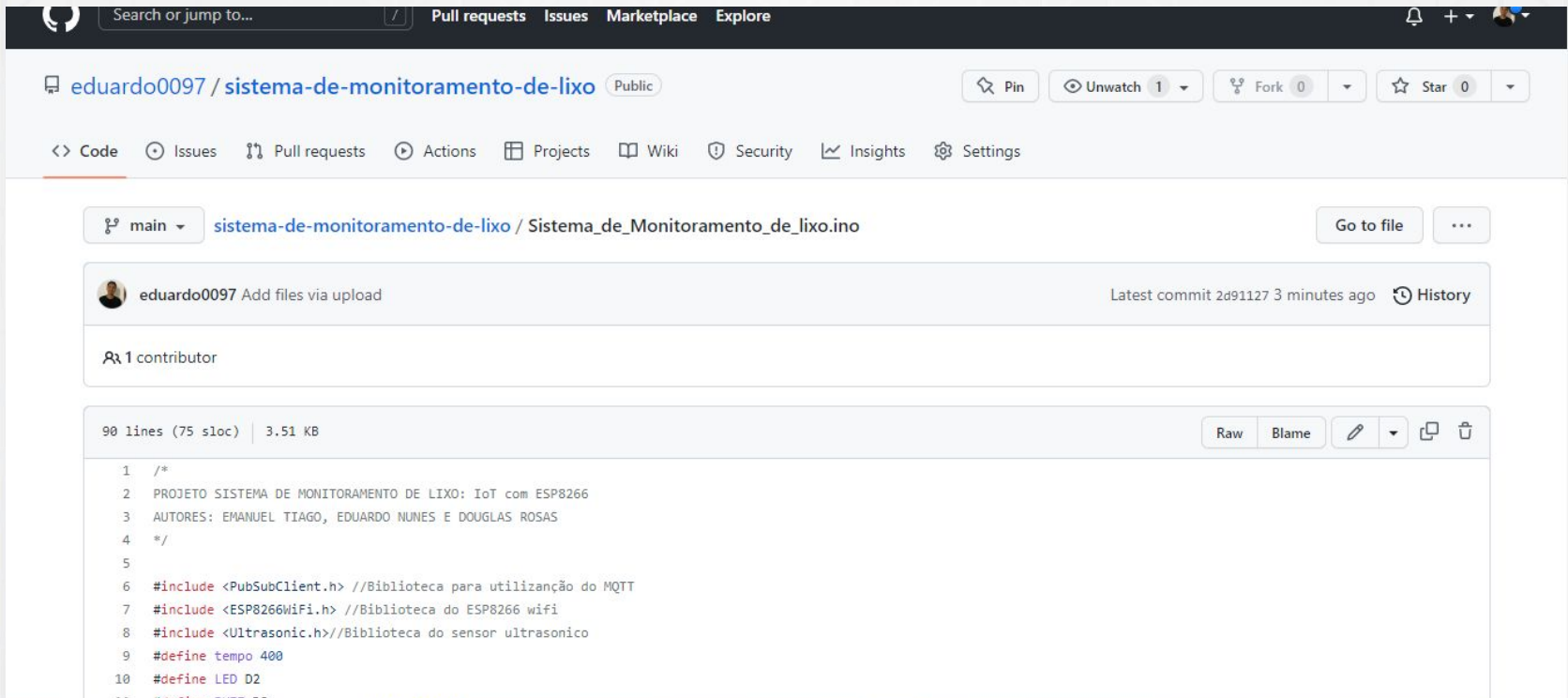
Sistema\_de\_Monitoramento\_de\_lixo

```
/*  
PROJETO SISTEMA DE MONITORAMENTO DE LIXO: IoT com ESP8266  
AUTORES: EMANUEL TIAGO, EDUARDO NUNES E DOUGLAS ROSAS  
*/  
  
#include <PubSubClient.h> //Biblioteca para utilização do MQTT  
#include <ESP8266WiFi.h> //Biblioteca do ESP8266 wifi  
#include <Ultrasonic.h> //Biblioteca do sensor ultrasonico  
#define tempo 400  
#define LED D2  
#define BUZZ D8  
#define trigger D5  
#define echo D6  
  
Ultrasonic ultrasonic(trigger, echo); // define as porta trigger e echo, respectivamente  
int distancia;  
char variavel[16];  
const char* ssid = "Polux";  
const char* password = "odisseia";  
//const char* ssid = "REDE EMANUEL 2.4"; //Nome da Rede WIFI que está usando  
//const char* password = "hellboy0"; // Senha da Rede Wifi que está usando  
//const char* ssid = "Fablab";  
//const char* password = "@fablab2020!";  
  
const char* topic_subl = "ultra-34"; //Topico onde o esp é um assinant  
const char* topic_pub = "ultra-455";  
const char* mqtt_id = "esp8266-Emanuel"; //ID do usuario, deve ser unico para cada esp8266
```

FONTE: AUTORES



# Código no GitHub

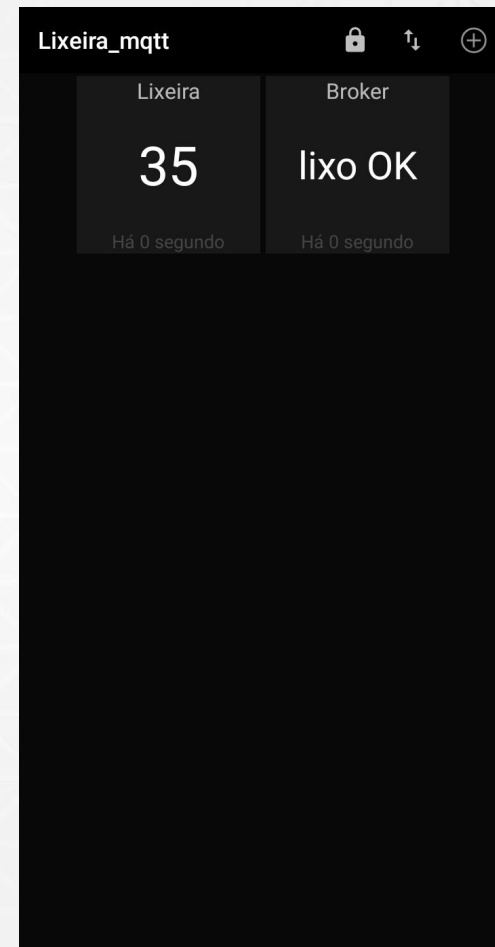
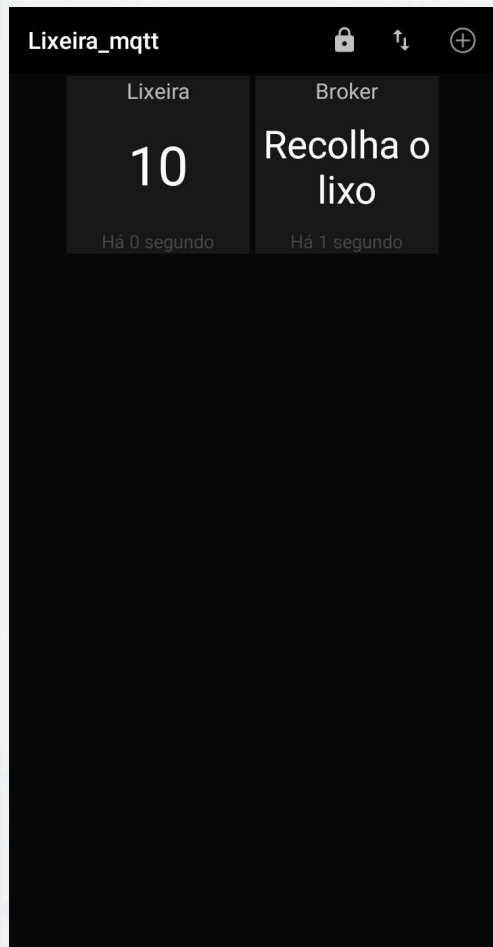


The screenshot shows the GitHub interface for the repository 'sistema-de-monitoramento-de-lixo' by user 'eduardo0097'. The repository is public and has 1 contributor. The file 'Sistema\_de\_Monitoramento\_de\_lixo.ino' is selected, showing 90 lines of code (75 sloc) and 3.51 KB. The code is in C++ and includes comments in Portuguese. The latest commit is by 'eduardo0097' 3 minutes ago.

```
1 /*
2 PROJETO SISTEMA DE MONITORAMENTO DE LIXO: IoT com ESP8266
3 AUTORES: EMANUEL TIAGO, EDUARDO NUNES E DOUGLAS ROSAS
4 */
5
6 #include <PubSubClient.h> //Biblioteca para utilização do MQTT
7 #include <ESP8266WiFi.h> //Biblioteca do ESP8266 wifi
8 #include <Ultrasonic.h> //Biblioteca do sensor ultrasonico
9 #define tempo 400
10 #define LED D2
11 #define Buzzer D3
```

FONTE: AUTORES

# Resultados



fonte: Autor

# Considerações Finais

Este trabalho apresenta de forma resumida o projeto de um sistema de monitoramento de lixo baseado em IoT com ESP8266, cuja principal finalidade do protótipo desenvolvido é informar ao usuário em tempo real a capacidade da lixeira alertando-o com um som e ligando o LED quando a mesma atingir sua capacidade máxima.

O presente projeto visa incentivar as pessoas a respeito da elaboração de tecnologias que buscam solucionar ou amenizar os problemas socioambientais da Amazônia.

# Referências bibliográficas

- [1] Eletronics Projects Hub. IoT Based Garbage Monitoring System | ESP8266 | Arduino. Acesso em 26 ago de 2022. Disponível em: <<https://electronics-project-hub.com/iot-based-garbage-monitoring-system/>>
- [2] SANTOS, Rui. Learn ESP 32 with Arduino. Disponível em: <<https://randomnerdtutorials.com/courses>>. Acesso em: 25 de set. 2022;
- [4] ESP32 Boards, TTGO. Disponível em: <[http://api.riot-os.org/group\\_\\_boards\\_\\_esp32\\_\\_ttgo-t-beam.html#details](http://api.riot-os.org/group__boards__esp32__ttgo-t-beam.html#details)>. Acesso em: 28 set. 2022;
- [5] FSR® 400 Series Data Sheet. Disponível em: <[https://files.seeedstudio.com/wiki/Grove-Round\\_Force\\_Sensor\\_FSR402/res/FSR402.pdf](https://files.seeedstudio.com/wiki/Grove-Round_Force_Sensor_FSR402/res/FSR402.pdf)>. acesso em: 22 set. 2022.

# Links importantes

## Repositório do GitHub

- [https://github.com/eduardo0097/sistema-de-monitoramento-de-lixo/blob/main/Sistema de Monitoramento de lixo.ino](https://github.com/eduardo0097/sistema-de-monitoramento-de-lixo/blob/main/Sistema%20de%20Monitoramento%20de%20lixo.ino)

## Projeto Original

- <https://electronics-project-hub.com/iot-based-garbage-monitoring-system/>

## Site de simulação

- <https://wokwi.com/>



**OBRIGADO!**

