

Save Money

Documentação (IOT)

Junho/2023

Florianópolis/SC

**17/12/2015**

Florianópolis/SC

**SUMÁRIO**

[1. INTRODUÇÃO 3](#_Toc364938596)

[2. OBJETIVO 4](#_Toc364938597)

[3. DESENVOLVIMENTO 5](#_Toc364938599)

[4. CONCLUSÃO 6](#_Toc364938602)

[5. REFERENCIAS 7](#_Toc364938602)

1. INTRODUÇÃO

Segundo pesquisas, cerca de 56% dos Brasileiros tem dificuldades em administrar seus fundos. Tendo isso em vista, a nossa equipe desenvolveu a Save Money, um cofre particular onde você tem total controle do seu dinheiro.

É comprovado que no Brasil, ainda não temos educação financeira nas escolas, dados mostram que 56% da população não tem controle sobre seu dinheiro, e consequentemente não consegue cumprir suas metas.

Diante desse problema a ideia é controlar melhor o seu dinheiro, utilizando um sistema onde você decide a sua meta. O cofre contém um sistema que funciona por meio de sensores, conseguindo identificar o valor de cada nota e registrando o valor no sistema do tago.io.

Save Money pode atingir diversos públicos tendo em vista que o controle de dinheiro é um problema que atinge boa parte da população brasileira.

A solução que tivemos foi criar a Save Money, que vai auxiliar a população a ter maior controle sobre seus gastos, criando metas, você pode arrecadar a quantia necessária para cumprir suas metas e sonhos. Caso você queira desistir de alguma meta ou altera-la você também pode.

2. OBJETIVOS

O objetivo da Save Money e ajudas as pessoas que tem dificuldade em guardar seu dinheiro... Segundo pesquisas, cerca de 56% dos Brasileiros tem dificuldades em administrar seus fundos. Tendo isso em vista, Nós desenvolvemos um sistema para que as pessoas possam guardar seu dinheiro e realizar suas metas com mais facilidade.

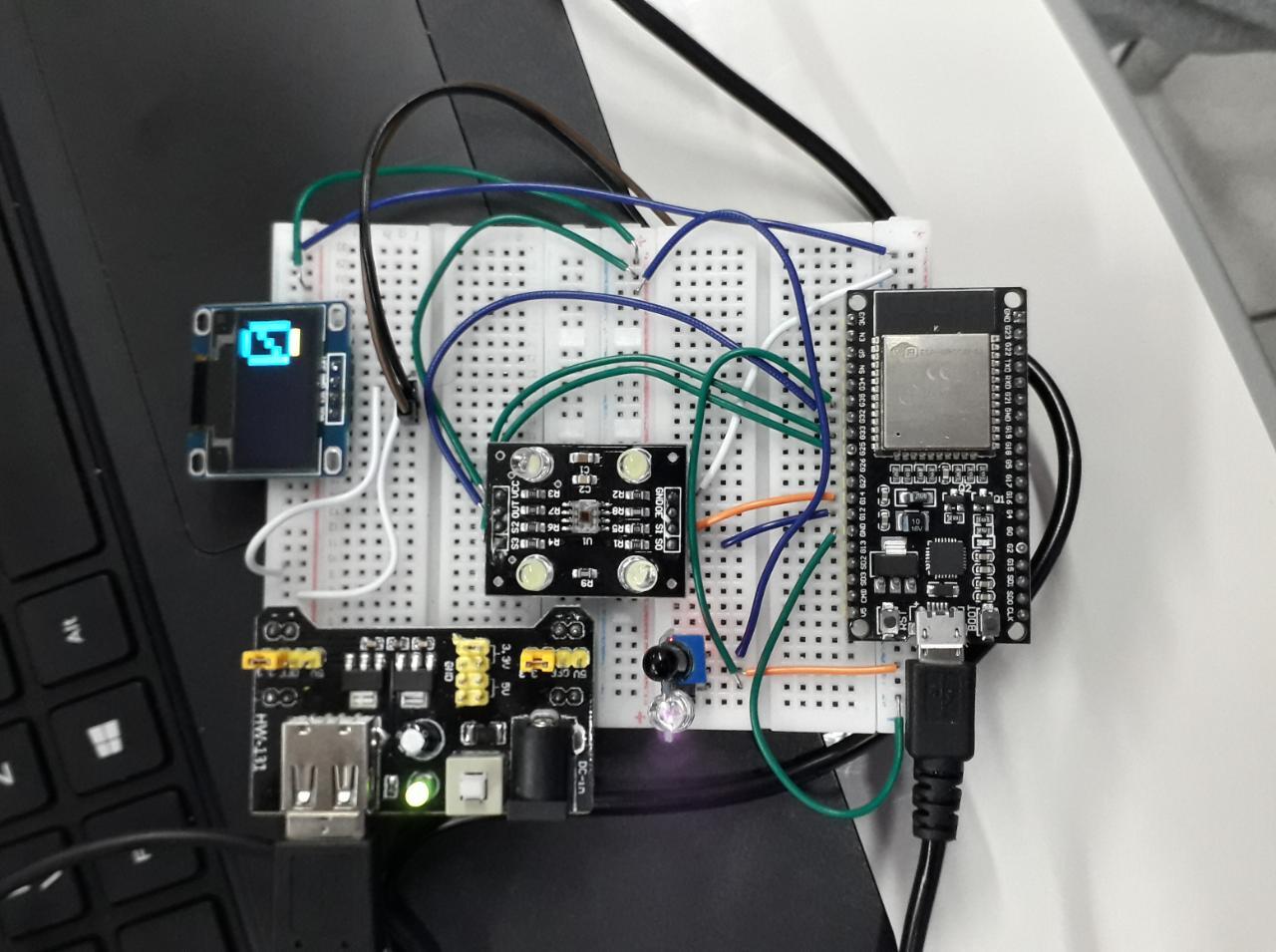
É comprovado que no Brasil, ainda não temos educação financeira nas escolas, dados mostram que 56% da população não tem controle sobre seu dinheiro, e consequentemente não consegue cumprir suas metas de algum bem material.

3. DESENVOLVIMENTO

As tecnologias utilizadas foram, foi a IDE do arduino, o micro-controlador ESP32, TSC230 que é um sensor de cor que usamos para conseguir identificar a nota colocada pelo usuário, usamos também o FC51 que é nada mais que sensor de presença para que quando ele for ativado, consiga mandar a informação que tem alguma coisa na frente dele e assim fazer com que o TSC230 ative e consiga identificar a nota, e por final usamos o display LCD OLED para que consiga mostrar a quantidade de dinheiro que o usuário inseri-o.

Foi usado no nosso projeto a plataforma da Tago.IO para armazenar as informações do cofre do usuário... Pelo Tago.IO o usuário consegue criar,visualizar, editar e excluir a suas metas de forma fácil.

Aqui esta o modelo físico de como ficou o projeto:



E aqui esta o código utilizado:

#include <Wire.h>

#include <Adafruit\_GFX.h>

#include <Adafruit\_SSD1306.h>

#include "EspMQTTClient.h"

#include <ArduinoJson.h>

#define SCREEN\_WIDTH 128 // OLED display width, in pixels

#define SCREEN\_HEIGHT 64 // OLED display height, in pixels

// Declaration for an SSD1306 display connected to I2C (SDA, SCL pins)

Adafruit\_SSD1306 display(SCREEN\_WIDTH, SCREEN\_HEIGHT, &Wire, -1);

char json\_valor[100];

char json\_presenca[100];

char json\_nota[100];

const int s0 = 12;

const int s1 = 14;

const int s2 = 33;

const int s3 = 25;

const int out = 32;

int red = 0;

int green = 0;

int blue = 0;

int color=0;

int redcolor = 0;

int greencolor = 0;

int orangecolor = 0;

int yellowcolor = 0;

const int pinoSensor = 35;

int presenca = 0;

int valor = 0;

int valorSensor = 0;

int valorSensorAnterior = 0;

int nota = 0;

EspMQTTClient client

(

"FIESC\_IOT", //nome da sua rede Wi-Fi

"C6qnM4ag81", //senha da sua rede Wi-Fi

"[mqtt.tago.io](http://mqtt.tago.io/)", // MQTT Broker server ip padrão da tago

"Token", // username

"9080c766-7ea3-493f-bc8f-efdcb1e2feac", // Código do Token

"SA FINAL", // Client name that uniquely identify your device

1883 // The MQTT port, default to 1883. this line can be omitted

);

void leitura\_sinais()

{

valorSensor = !digitalRead(pinoSensor);

}

void converte\_json()

{

StaticJsonDocument<300> sjson\_presenca;

sjson\_presenca["variable"] = "presenca";

sjson\_presenca["value"] = valorSensor;

serializeJson(sjson\_presenca, json\_presenca);

}

void converte\_jsonValor()

{

StaticJsonDocument<300> sjson\_valor;

sjson\_valor["variable"] = "valor";

sjson\_valor["value"] = valor;

serializeJson(sjson\_valor, json\_valor);

}

void converte\_jsonNota()

{

StaticJsonDocument<300> sjson\_nota;

sjson\_nota["variable"] = "nota";

sjson\_nota["value"] = nota;

serializeJson(sjson\_nota, json\_nota);

}

void envia\_msg()

{

client.publish("presenca", json\_presenca); // You can activate the retain flag by setting the third parameter to true

}

void envia\_msgValor()

{

client.publish("presenca", json\_valor); // You can activate the retain flag by setting the third parameter to true

}

void envia\_msgNota()

{

client.publish("presenca", json\_nota); // You can activate the retain flag by setting the third parameter to true

}

void mostrarDisplay(){

display.setTextSize(8);

display.setTextColor(WHITE);

display.setCursor(0, 10);

display.clearDisplay();

display.println(valor);

display.display();

}

void setup()

{

Serial.begin(115200);

if(!display.begin(SSD1306\_SWITCHCAPVCC, 0x3C)) { // Address 0x3D for 128x64

Serial.println(F("SSD1306 allocation failed"));

for(;;);

}

delay(2000);

display.clearDisplay();

display.setTextSize(8);

display.setTextColor(WHITE);

display.setCursor(0, 10);

// Display static text

display.println(valor);

display.display();

Serial.begin(9600);

pinMode(s0, OUTPUT);

pinMode(s1, OUTPUT);

pinMode(s2, OUTPUT);

pinMode(s3, OUTPUT);

pinMode(out, INPUT);

digitalWrite(s0, HIGH);

digitalWrite(s1, HIGH);

pinMode(pinoSensor, INPUT);

}

void loop()

{

//if(digitalRead(pinoSensor) == LOW){

// Serial.println("true");

leitura\_sinais();

if(valorSensor != valorSensorAnterior)

{

valorSensorAnterior = valorSensor;

converte\_json();

envia\_msg();

digitalWrite(s2, LOW);

digitalWrite(s3, LOW);

//count OUT, pRed, RED

red = pulseIn(out, digitalRead(out) == HIGH ? LOW : HIGH);

digitalWrite(s3, HIGH);

//count OUT, pBLUE, BLUE

blue = pulseIn(out, digitalRead(out) == HIGH ? LOW : HIGH);

digitalWrite(s2, HIGH);

//count OUT, pGreen, GREEN

green = pulseIn(out, digitalRead(out) == HIGH ? LOW : HIGH);

Serial.print("R Intensity:");

Serial.print(red, DEC);

Serial.print(" G Intensity: ");

Serial.print(green, DEC);

Serial.print(" B Intensity : ");

Serial.println(blue, DEC);

delay(500);

// if(red<30 & red>20 & green<35 & green>25 & blue<32 & blue>22){

// delay(1000);

// redcolor++;

// Serial.print("Red");

// Serial.begin(115200);

// delay(2000);

// display.clearDisplay();

// }

if(red<33 & red>24 & green<31 & green>20 & blue<25 & blue>15){

delay(1000);

yellowcolor++;

valor = valor + 2;

converte\_jsonValor();

envia\_msgValor();

nota = 2;

converte\_jsonNota();

envia\_msgNota();

Serial.println("\n2 REAIS ");

delay(200);

mostrarDisplay();

}

}

client.loop();

delay(1000);

}

void processa\_msg(const String payload)

{

StaticJsonDocument<300> msg;

DeserializationError err = deserializeJson(msg, payload);

if (err) {

Serial.print(F("deserializeJson() failed with code "));

Serial.println(err.f\_str());

}

Serial.print("var:");

String var = msg["variable"];

Serial.println(var);

if(var == "zerar")

{

Serial.print("value:");

String val = msg["value"];

Serial.println(val);

valor =0;

converte\_jsonValor();

envia\_msgValor();

mostrarDisplay();

}

}

void onConnectionEstablished()

{

client.subscribe("recebe", [] (const String &payload) {

Serial.println(payload);

processa\_msg(payload);

});

}

4. CONCLUSÃO

Nós da equipe Save Money acreditamos que conseguimos alcançar nosso objetivo do projeto, pois conseguimos criar oque tínhamos planejado da forma com que qualquer usuario consiga usar de forma rápida e pratica.

5. REFERENCIAS

Esp32io.com

randomnerdtutorials.com

blog.eletrogate.com

unprogramador.com