



**INSTITUTO
FEDERAL**

Santa Catarina

Câmpus
São José

Spec Eletri

Projeto Integrador 2

Alunos: Beatriz Paz Faria, Igor Budag de Oliveira e Júlia Espíndola Steinbach
Professor: Adilson Jair Cardoso

21 de fevereiro de 2025

Sumário

1. Logomarca	3
2. Canvas	4
3. UML	4
4. Componentes importantes	5
4.1. Node-RED	5
4.2. Broker Mosquitto	5
4.3. Banco de dados	5
4.4. ESP32	5
4.5. AWS EC2	5
5. Node-RED	6
5.1. Sensores	6
5.2. Status e Acionamento	7
5.3. Bancos de Dados	7
5.4. Dashboard	8
6. Código ESP	8
7. Custos e Lucros esperados	12

1. Logomarca

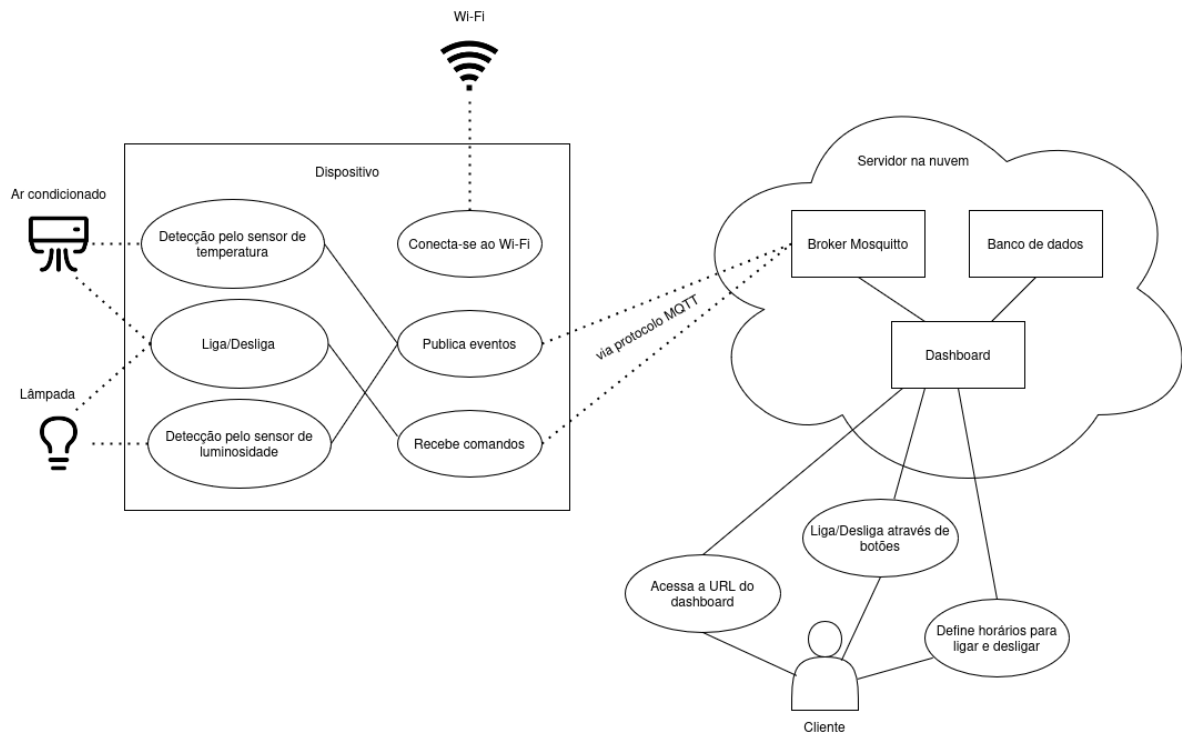


2. Canvas

NOME DO PROJETO		SPEC ELETRI	PROPÓSITO (Objetivo Estratégico)		Monitoramento Elétrico				
OBJETIVO (Verbo infinitivo + Produto) - Monitorar os gastos energéticos em tempo real e remotamente. - Ligar ou desligar equipamentos conforme a necessidade.	CARACTERÍSTICAS DO PRODUTO/SERVIÇO (Descrever com o máximo de características) - Acesso remoto da temperatura e luz do ambiente - Agendamento do horário para ligar e desligar lâmpada e/ ou ar condicionado - Aviso ao usuário sobre o status dos dispositivos - Controle dos equipamentos a distância	STAKEHOLDERS (Patrocinador, Cliente, Equipe e Resistentes) - Empresas com diversos aparelhos eletrônicos. - Fornecedores de insumo.	PREMISSAS & RESTRIÇÕES (O que TEM QUE ser considerado?) - É necessário uma conexão com a rede em ambas as pontas para a comunicação cliente e aparelho. - O custo energético não necessariamente vai diminuir com esse produto, mas se for monitorado corretamente pode sim diminuir.	RISCOS (O QUE pode impactar?) - Sem rede o usuário perde o contato com o controlador. - A precisão do sensor pode afetar a detecção de luz ou temperatura. Sem atualização no banco de dados, o equipamento pode ficar ligado em um ambiente vazio.	JUSTIFICATIVA (Listar problemas) - Verificar o estado de aparelhos eletrônicos exige tempo e presença no local onde estão. - O gasto energético pode acabar saindo muito caro no final do mês, se não monitorado corretamente.	ESCOPO (O que será feito?) - O usuário consegue comprovar se os dispositivos estão ligados através dos sensores. - Pode desligar e ligar automaticamente com base no horário previsto em um banco de dados. - É possível alterar o status manualmente pela interface a distância.	ENTREGAS (Escrever no participio passado. Ex.: Rel. elaborado ou Rel. aprovado) - Protótipo funcional.	DATAS (Para cada entrega) - 1 Semestre.	INVESTIMENTO (Para cada entrega) - R\$ 277.130,00.
BENEFÍCIOS (Quais serão os ganhos?) - Redução do custo de energia. - Prevenir danos dos equipamentos por uso excessivo. - Contribuir para práticas mais sustentáveis.	NÃO ESCOPO (O que NÃO será feito?) - Não alterará a temperatura do ar condicionado. - Não controlará outros dispositivos que não envolvem luzes ou temperaturas. - Não alertará sobre manutenção dos equipamentos.								

Obra disponibilizada com Licença Creative Commons Atribuição 4.0 Internacional

3. UML



4. Componentes importantes

4.1. Node-RED

Interface utilizada para verificação do estado dos fatores afetando o ESP32, tais como temperatura, luminosidade, umidade, além disso é capaz de pedir para o ESP ligar ou desligar o relé remotamente.

4.2. Broker Mosquitto

Middleware que permite a comunicação entre o Node-RED e o ESP, utilizando o modelo de comunicação MQTT publish/subscribe, comumente presente em dispositivos IoT.

4.3. Banco de dados

Utilizando do mongodb é possível automatizar o gerenciamento energético que o produto oferece, possibilitando definir um horário onde o relé liga, e um que ele desliga.

4.4. ESP32

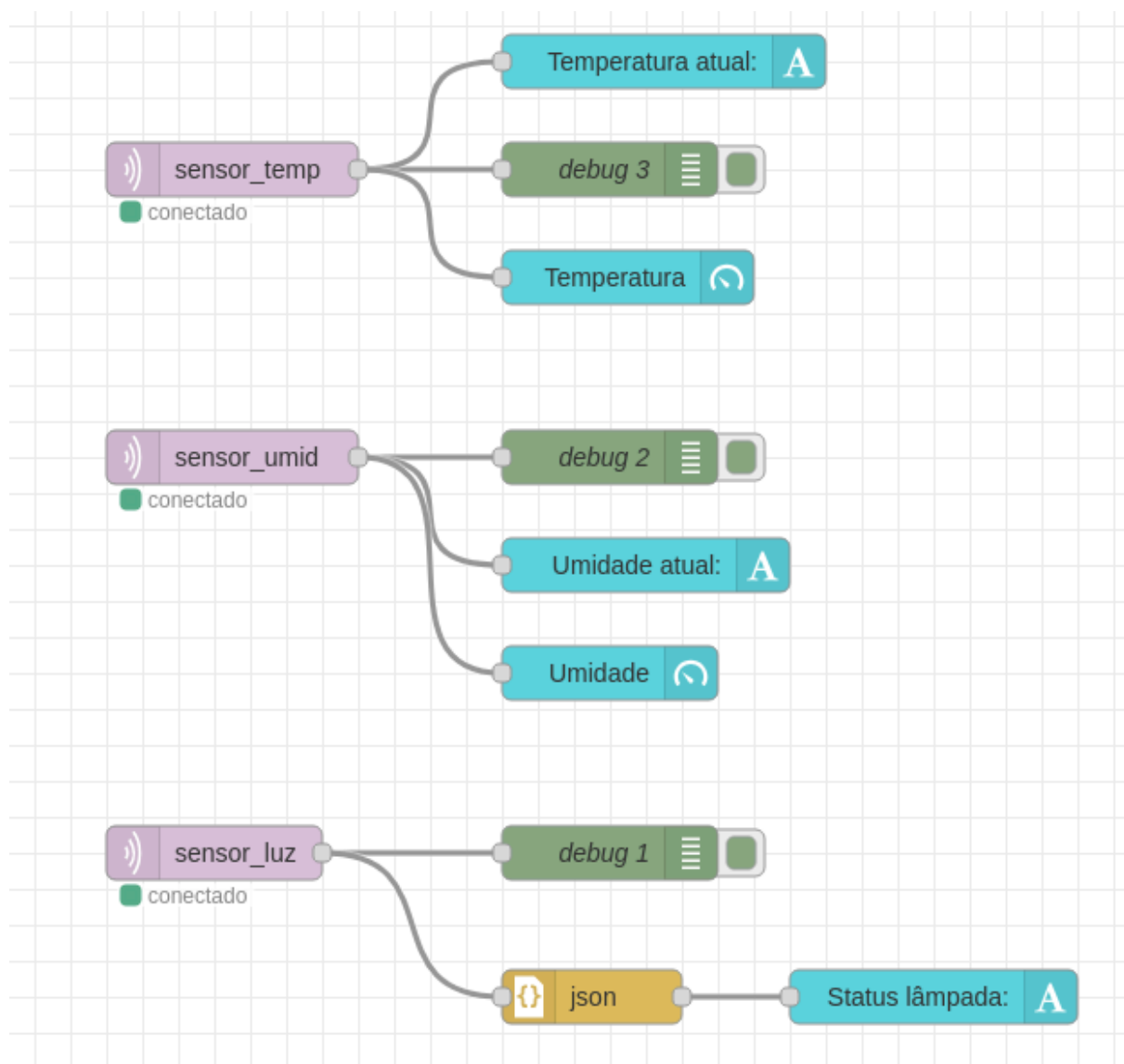
É o dispositivo principal do produto, responsável pela coleta de informações, as quais são entregues ao Node-RED para permitir o monitoramento. Além disso, como mencionado anteriormente, é o que energiza o relé.

4.5. AWS EC2

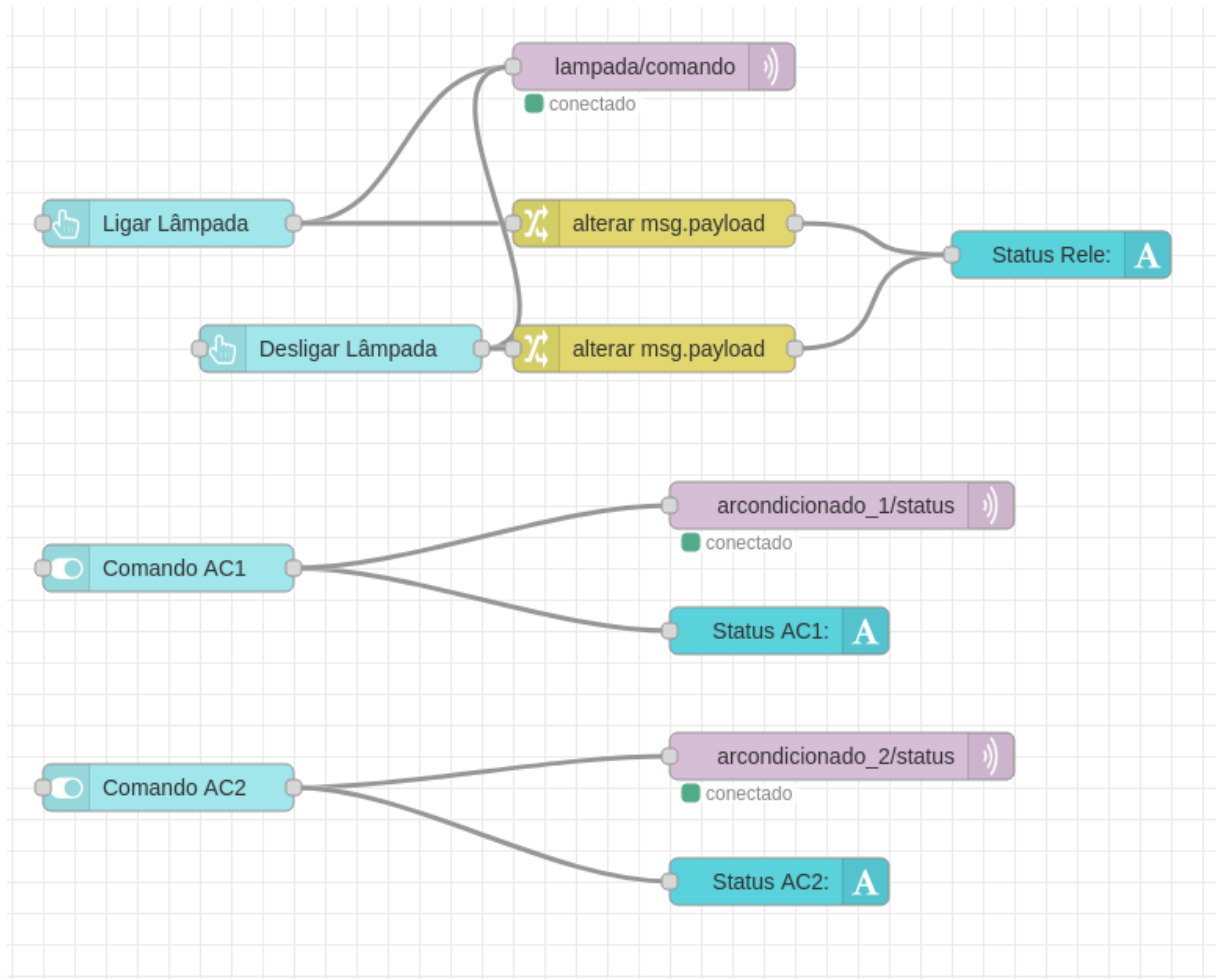
Essa plataforma foi usada para hospedar tanto o Node-RED, quanto o Mosquitto e o banco de dados, permitindo assim a comunicação à distância com o ESP, desde que ambas as partes possuam acesso à internet.

5. Node-RED

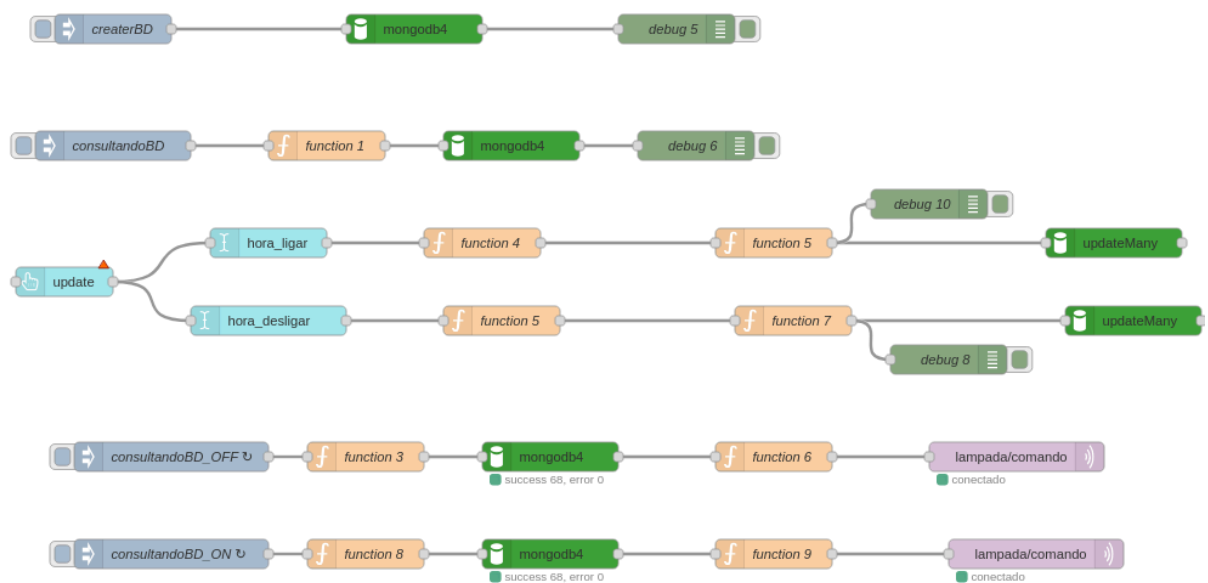
5.1. Sensores



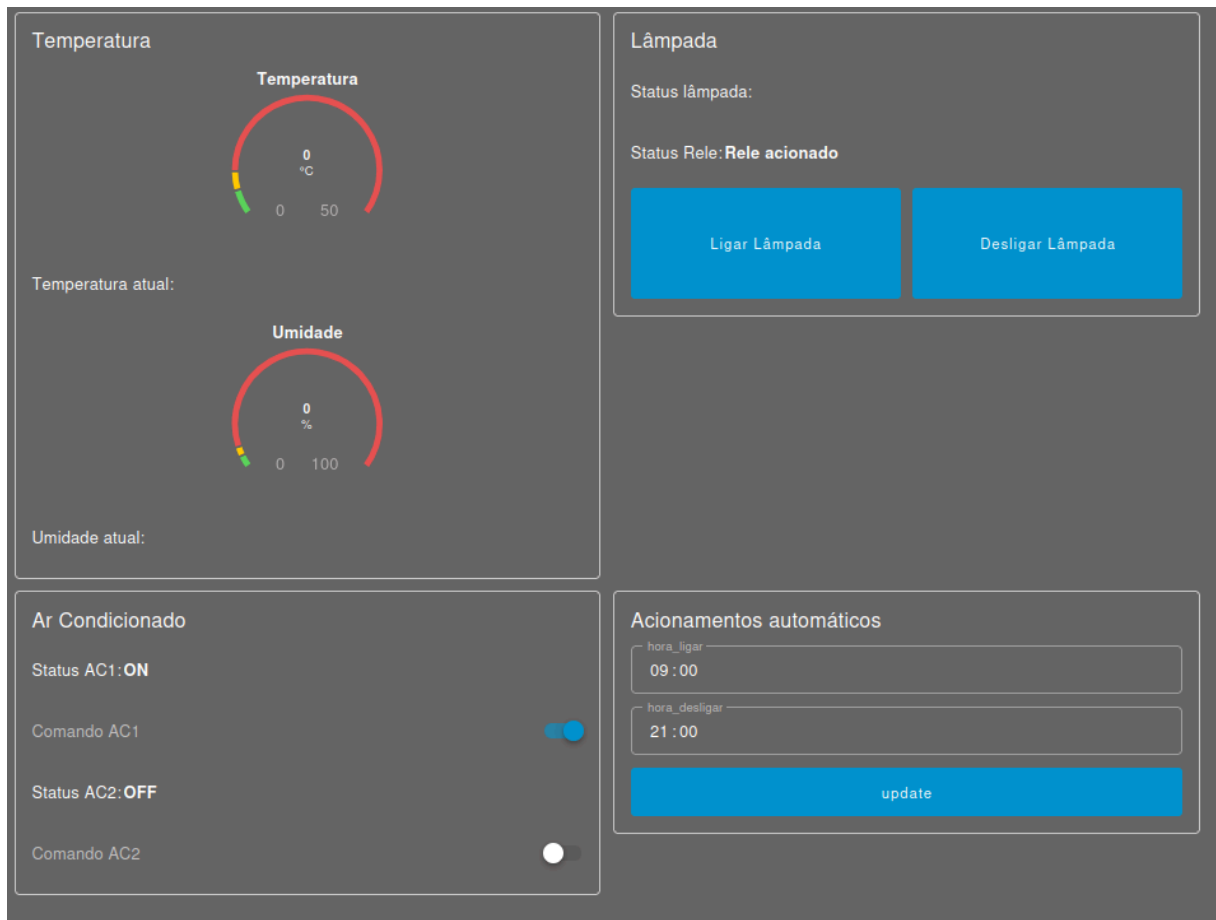
5.2. Status e Acionamento



5.3. Bancos de Dados



5.4. Dashboard



6. Código ESP

```
#include <WiFi.h>
#include <PubSubClient.h>
#include <DHT.h>
#include <Wire.h>
#include <Adafruit_ADS1X15.h>

// Configurações do DHT22
#define DHTPIN 4 // GPIO do sensor
#define DHTTYPE DHT22 // Tipo de sensor
DHT dht(DHTPIN, DHTTYPE);

// Criar um objeto para o ADS1115
Adafruit_ADS1115 ads;

// Pinos para controle
#define RELAY_PIN 13 // Pino do relé (lâmpada)
#define LED_AC_1 12 // LED indicando AC1
#define LED_AC_2 14 // LED indicando AC2

// Configurações de WiFi
```



```

const char* ssid = "Fabio";
const char* password = "F0023339";

// Configurações do MQTT (Mosquitto)
const char* mqttServer = "18.231.157.84";
const int mqttPort = 1883;
const char* mqttUser = "";
const char* mqttPassword = "";
const char* mqttTempTopic = "sensor_temp";
const char* mqttHumTopic = "sensor_umid";
const char* mqttLdrTopic = "sensor_luz";
const char* mqttRelayTopic = "lampada/comando";
const char* mqttAc1StatusTopic = "arcondicionado_1/status";
const char* mqttAc2StatusTopic = "arcondicionado_2/status";

// Variáveis globais
WiFiClient espClient;
PubSubClient mqttClient(espClient);
float temperature = 0.0;
float humidity = 0.0;
int16_t ldrValue = 0;

// Controle de tempo
unsigned long lastPublishTime = 0;
const unsigned long publishInterval = 2000;

void connectToWiFi() {
    Serial.print("Conectando ao WiFi: ");
    Serial.println(ssid);
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
        Serial.print(".");
        delay(500);
    }
    Serial.println("\nConectado ao WiFi!");
}

void connectToMQTT() {
    mqttClient.setServer(mqttServer, mqttPort);
    while (!mqttClient.connected()) {
        Serial.print("Conectando ao broker MQTT...");
        if (mqttClient.connect("ESP32Client", mqttUser, mqttPassword)) {
            Serial.println("Conectado!");
            mqttClient.subscribe(mqttRelayTopic);
            mqttClient.subscribe(mqttAc1StatusTopic);
            mqttClient.subscribe(mqttAc2StatusTopic);
        } else {
            Serial.print("Falha na conexão, rc=");
            Serial.print(mqttClient.state());
            Serial.println(" Tentando novamente em 5 segundos...");
            delay(5000);
        }
    }
}

```

```

    }
}

void mqttCallback(char* topic, byte* payload, unsigned int length) {
    String message = "";
    for (unsigned int i = 0; i < length; i++) {
        message += (char)payload[i];
    }
    Serial.print("Mensagem recebida no tópico ");
    Serial.print(topic);
    Serial.print(": ");
    Serial.println(message);

    // Controle da lâmpada
    if (String(topic) == mqttRelayTopic) {
        if (message == "ON") {
            digitalWrite(RELAY_PIN, HIGH);
            Serial.println("Lâmpada ligada");
        } else if (message == "OFF") {
            digitalWrite(RELAY_PIN, LOW);
            Serial.println("Lâmpada desligada");
        }
    }

    // Controle do status do ar-condicionado 1
    else if (String(topic) == mqttAc1StatusTopic) {
        if (message == "ON") {
            digitalWrite(LED_AC_1, HIGH);
            Serial.println("Ar-condicionado 1 LIGADO");
        } else if (message == "OFF") {
            digitalWrite(LED_AC_1, LOW);
            Serial.println("Ar-condicionado 1 DESLIGADO");
        }
    }

    // Controle do status do ar-condicionado 2
    else if (String(topic) == mqttAc2StatusTopic) {
        if (message == "ON") {
            digitalWrite(LED_AC_2, HIGH);
            Serial.println("Ar-condicionado 2 LIGADO");
        } else if (message == "OFF") {
            digitalWrite(LED_AC_2, LOW);
            Serial.println("Ar-condicionado 2 DESLIGADO");
        }
    }
}

void publishToMQTT() {
    char message[50];

    snprintf(message, sizeof(message), "%.2f", temperature);
    mqttClient.publish(mqttTempTopic, message);
}

```

```

    snprintf(message, sizeof(message), "%.2f", humidity);
    mqttClient.publish(mqttHumTopic, message);

    snprintf(message, sizeof(message), "{\"ldrStatus\":\"%s\"}", (ldrValue >
5000) ? "Lâmpada APAGADA" : "Lâmpada ACESA");
    mqttClient.publish(mqttLdrTopic, message);

    snprintf(message, sizeof(message), "{\"arcondicionado1\":\"%s\"}",
digitalRead(LED_AC_1) == HIGH ? "Ligado" : "Desligado");
    mqttClient.publish(mqttAc1StatusTopic, message);

    snprintf(message, sizeof(message), "{\"arcondicionado2\":\"%s\"}",
digitalRead(LED_AC_2) == HIGH ? "Ligado" : "Desligado");
    mqttClient.publish(mqttAc2StatusTopic, message);
}

void setup() {
    Serial.begin(115200);
    dht.begin();
    Wire.begin(21, 22);
    if (!ads.begin(0x49)) {
        Serial.println("Falha ao inicializar o ADS1115!");
        while (1);
    }

    pinMode(RELAY_PIN, OUTPUT);
    pinMode(LED_AC_1, OUTPUT);
    pinMode(LED_AC_2, OUTPUT);

    connectToWiFi();
    mqttClient.setCallback(mqttCallback);
    connectToMQTT();
}

void loop() {
    if (!mqttClient.connected()) {
        connectToMQTT();
    }
    mqttClient.loop();

    unsigned long currentTime = millis();
    if (currentTime - lastPublishTime >= publishInterval) {
        temperature = dht.readTemperature();
        humidity = dht.readHumidity();
        ldrValue = ads.readADC_SingleEnded(3);

        if (!isnan(temperature) && !isnan(humidity)) {
            publishToMQTT();
        }

        lastPublishTime = currentTime;
    }
}

```

```

}
delay(100);
}

```

7. Custos e Lucros esperados

Custos previstos			
Nome	Descrição	Valor em R\$	/Peça
Caixa		40000	40
Montagem PCB		0	0
Confeção PCB		20000	20
Montagem		20000	20
Integradores	Construção das peças	100000	100
ESP32		40000	40
Fonte		20000	20
Outros		20000	20
Marketing		4000	4
Patente		8000	8
Aluguel		100	0,1
Água		100	0,1
Energia		100	0,1
Internet		100	0,1
Telefone		100	0,1
Salário		1000	
Imposto sócio	Prolabori (11%)	110	
Salário total	3 sócios	3330	3,33
Advogado		800	0,8
Contador		500	0,5

Quantidade de peças:	1000
----------------------	------

Análise dos Dados		
Nome	Descrição	Valor em R\$
Custo/produto		277,13
Custo de todas as peças		277130
Faturamento esperado	100%	554260
Poupança economica	20% do lucro	110852
Valor mínimo ganho/produto	considerando o faturamento	554,26
Valor recebido pós imposto	cobrando o valor mínimo	468,6
Variação recebido/mínimo		84,55%

Valor sugerido do Produto	Valor que satisfaça o lucro esperado	655,54
---------------------------	--------------------------------------	--------