

ESCOLA SENAI DE ITU  
CENTRO EDUCACIONAL 401  
ENSINO MÉDIO INTEGRADO AO TÉCNICO

BIANCA BLANCO  
CAMILLA DOS SANTOS GARCEZ GONÇALVES  
GEOVANA CLEMENTE CRUZ  
GUILHERME DE MATOS MONTE  
LIVIA MAYUMI HAYASHIDA

**PROJETO IOT: SISTEMA INTELIGENTE DE OTIMIZAÇÃO DE ENERGIA**

SALTO  
2025

BIANCA BLANCO  
CAMILLA DOS SANTOS GARCEZ GONÇALVES  
GEOVANA CLEMENTE CRUZ  
GUILHERME DE MATOS MONTE  
LIVIA MAYUMI HAYASHIDA

**PROJETO IOT: SISTEMA INTELIGENTE DE OTIMIZAÇÃO DE ENERGIA**

Trabalho apresentado à disciplina IoT do Ensino médio e técnico da Escola SESI/SENAI de Salto, como avaliação parcial do 3º semestre/termo do 2º ano.  
Professor(a): Marlon Fanger Rodrigues, Celso Rodrigo Giusti e Daniel Manoel Filho.

SALTO  
2025

## SUMÁRIO

1 INTRODUÇÃO .....	4
2 DESCRIÇÃO.....	5
3 HARDWARE .....	6
4 PINAGEM.....	9
5 INSTRUÇÕES .....	10
6 CONCLUSÃO .....	11

## **1 INTRODUÇÃO**

Diante dos desafios atuais para economia de energia, o grupo SunHeat IoTechs desenvolveu um sistema inteligente para otimizar o uso de ar-condicionado em uma sala de aula do curso de Desenvolvimento de Sistemas. O objetivo principal é reduzir gastos energéticos e tornar o ambiente mais confortável, utilizando tecnologia para automatizar decisões que normalmente dependem do usuário.

O projeto surgiu da necessidade de equilibrar conforto térmico e sustentabilidade, já que o uso incorreto ou excessivo de equipamentos de controle de temperatura pode gerar desperdício e diminuir a vida útil das máquinas. Com o sistema proposto, torna-se possível monitorar e controlar a temperatura do ambiente de forma automática, visando o bem-estar dos alunos e professores, além de contribuir para práticas mais eficientes dentro da instituição.

## 2 DESCRIÇÃO

A equipe coletou informações sobre uma sala de aula do SENAI, considerando o número de computadores, estudantes e profissionais presentes no ambiente. No total, foram contabilizadas 33 pessoas e 34 máquinas. A partir desses dados, foi possível calcular a temperatura ideal para garantir conforto térmico e bom desempenho dos equipamentos.

Com base na análise, concluímos que a sala deve ser mantida entre 21 °C e 24 °C, com umidade relativa do ar entre 40% e 60%. Temperaturas abaixo de 21 °C não apresentam benefícios adicionais e podem causar desconforto aos ocupantes. Já valores acima de 24–25 °C, somados ao calor gerado pelo uso constante dos notebooks, podem reduzir a vida útil das máquinas e atrapalhar a concentração dos usuários. Assim, a faixa que melhor equilibra conforto e desempenho fica entre 22 °C e 23 °C, mantendo a umidade recomendada.

Para auxiliar no monitoramento, será instalado um sensor de movimento na porta, responsável por registrar quantas pessoas entram e saem da sala, dado essencial para o cálculo da temperatura adequada. Além disso, o sistema contará com um indicador visual:

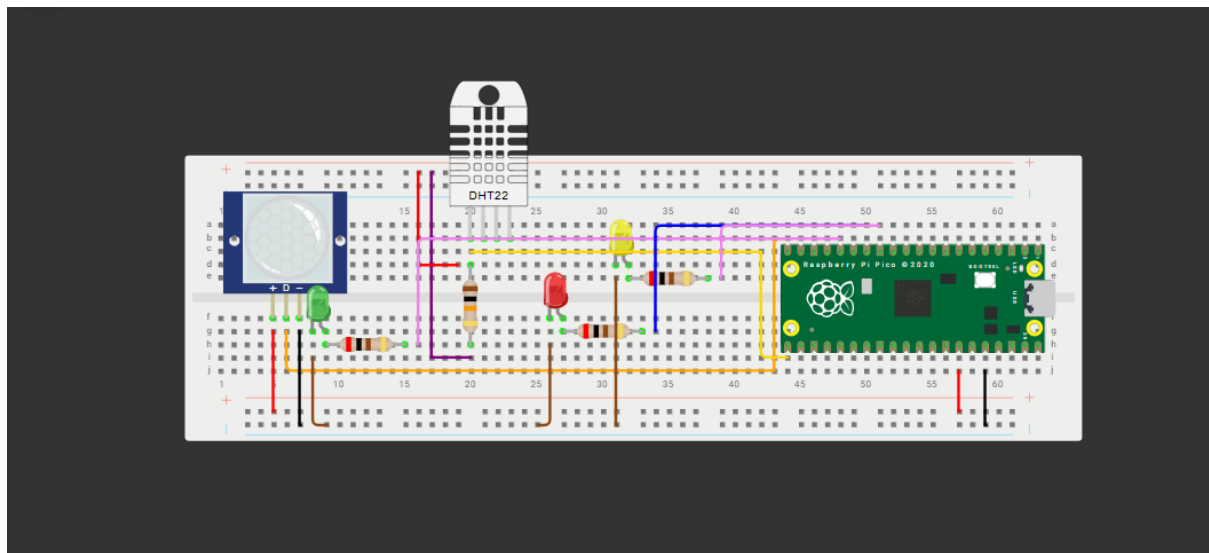
- **Luz vermelha:** parâmetros muito acima ou abaixo do ideal, indicando ambiente quente ou frio demais;
- **Luz amarela:** valores próximos ao limite recomendado;
- **Luz verde:** condições dentro da faixa ideal.

O sistema só funcionará após o sensor de movimento detectar uma pessoa dentro da sala.

Com essa solução, será possível otimizar o consumo de energia e promover melhores condições de saúde e bem-estar aos estudantes e profissionais, ajustando a temperatura e a umidade do ar-condicionado de forma automatizada e proporcional ao número de indivíduos e equipamentos presentes no local, mantendo o ambiente sempre dentro da média térmica recomendada.

### 3 HARDWARE

- DHT22 (sensor de temperatura e umidade): sensor que mede a temperatura e umidade. Ele será responsável por calcular a sensação térmica da sala e ficará do lado oposto ao ar-condicionado.
- HC-SR501 PIR (sensor de movimento): sensor que irá identificar a presença de pessoas na sala a partir do movimento de entrada e saída. Para isso ele ficará conectado na porta.
- LED (vermelho, amarelo, verde): dependendo do valor que irá ser passado pelo sensor de temperatura e umidade, a luz de LED será ativada na sala.
- Raspberry Pi: é o equipamento usado para ler os sensores e processar os dados. A partir do funcionamento da placa serão extraídos os dados como a temperatura e a umidade atual.



```

1  from machine import Pin      # Importa a classe Pin para controlar GPIOs(pinos em microcontroladores)
2  from dht import DHT22       # Importa o driver específico para o sensor DHT22
3  from utime import sleep     # Importa a função sleep para introduzir pausas
4
5  # Inicializa o sensor DHT22 na porta digital GP15
6  sensor = DHT22(Pin(16))
7  # Inicializa o sensor PIR (movimento) na porta GP14 como entrada
8  pir = Pin(14, Pin.IN) # Configura o pino 14 como entrada
9
10 # Inicializa os pinos digitais GP2, GP3 e GP4 como saídas para controlar os LEDs
11 led_verde = Pin(12, Pin.OUT)
12 led_amarelo = Pin(10, Pin.OUT)
13 led_vermelho = Pin(3, Pin.OUT)
14
15 # Define uma função para garantir que todos os LEDs estejam desligados
16 def leds_off():
17     led_verde.off()
18     led_amarelo.off()
19     led_vermelho.off()
20
21 while True: # Loop principal
22     # 1- Leitura dos Sensores
23     sensor.measure()      # Solicita a medição ao sensor DHT22 (temperatura e umidade)
24     temp = sensor.temperature() # Armazena o valor da temperatura em °C
25     movimento = pir.value() # Lê o estado do sensor PIR (0=sem movimento, 1=movimento)
26
27     # 2- Exibição no Console
28     print(f"Temperatura: {temp}°C | Movimento: {movimento}")
29
30     # 3- Lógica de Sinalização
31     leds_off() # Garante que todos os LEDs estejam desligados antes de ligar o estado atual
32

```

```
if movimento == 0:
    # CONDIÇÃO 1: Sem movimento (ambiente inativo)
    # Acende o LED Verde (sinalizando que está tudo "normal" ou inativo)
    led_verde.on()

else:
    # CONDIÇÃO 2: Movimento detectado (pessoas ativas)

    # Subcondição A: Temperatura ideal
    if temp < 26:
        # Temperatura abaixo de 26°C → OK
        led_verde.on()

    # Subcondição B: Temperatura de atenção
    elif 26 <= temp <= 28:
        # Temperatura entre 26°C e 28°C → Atenção
        led_amarelo.on()

    # Subcondição C: Temperatura de alerta/perigo
    elif temp > 28:
        # Temperatura acima de 28°C → Perigo/Alerta

        # Pisca o LED Vermelho rapidamente
        led_vermelho.on()
        sleep(0.3)
        led_vermelho.off()
        sleep(0.3)
        # Pula o 'sleep(1)' final e reinicia o loop 'while True',
        # garantindo que o pisca-pisca continue rapidamente.
        continue

# Pausa de 1 segundo (usada apenas se não estiver piscando o LED Vermelho)
sleep(1)
```



## **4 PINAGEM**

A pinagem é a identificação de cada pino de um componente eletrônico ou conector, indicando sua função e onde ele deve ser ligado. Ela funciona como um guia de conexão, auxiliando a montagem dos circuitos de forma correta e garantindo que todos os componentes funcionem como esperado. Com base nisso as conexões do nosso projeto irão ficar desta maneira:

### **DHT22**

- VCC - 3.3V
- GND - GND
- DATA -

### **Sensor PIR**

- VCC - 5V ou 3.3V
- GND - GND
- OUT - GP14

### **LEDs**

- LED 1 - GPIO 2 (com resistor)
- LED 2 - GPIO 3 (com resistor)
- LED 3 - GPIO 4 (com resistor)

### **Biblioteca**

- Arquivo dht.py - deve estar salvo na placa (Raspberry Pi Pico)

## 5 INSTRUÇÕES

Primeiramente, a equipe SunHeat IoTechs fez a prototipação do projeto no Wokwi para testar o código e confirmar que tudo funcionaria antes de montar a versão física. O código foi escrito em Python e depois as conexões foram simuladas no Raspberry Pi virtual do Wokwi, evitando possíveis problemas ou danos aos componentes reais.

Depois disso, transferimos o código para o Thonny e montamos as conexões na Raspberry Pi física exatamente como no simulador. Em seguida, conectamos a placa ao computador pelo cabo USB para rodar o programa.

Para que o projeto funcione corretamente, é importante que o DHT22 esteja ligado da forma certa, o sensor PIR conectado ao pino GP14, os LEDs instalados nos GPIO 2, 3 e 4 com seus resistores, e que a biblioteca dht.py esteja presente na placa.

## 6 CONCLUSÃO

O projeto SunHeat IoTechs foi desenvolvido com o propósito de criar uma solução tecnológica capaz de otimizar o uso do ar-condicionado em uma sala de aula, conciliando conforto térmico, economia de energia e preservação dos equipamentos. Ao longo do processo, buscamos compreender de forma precisa as necessidades do ambiente, analisando fatores como quantidade de pessoas, número de máquinas em funcionamento e condições ideais de temperatura e umidade. A partir dessas informações, estruturou-se um sistema inteligente voltado para tornar o espaço mais eficiente e agradável para alunos e professores e benéfico para os equipamentos.

Durante o desenvolvimento, os objetivos propostos foram alcançados. Foi possível definir parâmetros térmicos adequados, projetar a instalação de sensores para monitorar movimento e ocupação e propor um sistema de alertas visuais que auxilia na identificação de situações fora dos padrões recomendados. Além disso, a equipe realizou estudos que permitiram entender como o calor gerado pelo uso intenso dos equipamentos influencia tanto o bem-estar dos usuários quanto a vida útil das máquinas, reforçando a relevância da automação no controle ambiental.

O projeto também trouxe aprendizados significativos para a equipe, tanto no âmbito técnico quanto no âmbito pessoal, estimulando colaboração, responsabilidade e visão sustentável. A experiência contribuiu ainda para ampliar o entendimento sobre o impacto de práticas de consumo energético consciente dentro de instituições de ensino.

Neste sentido, conclui-se que o projeto SunHeat IoTechs cumpriu com êxito seus objetivos iniciais, apresentando uma proposta funcional, sustentável e alinhada às demandas contemporâneas por eficiência energética. A iniciativa evidencia o potencial da tecnologia como ferramenta para promover ambientes mais confortáveis, inteligentes e responsáveis.