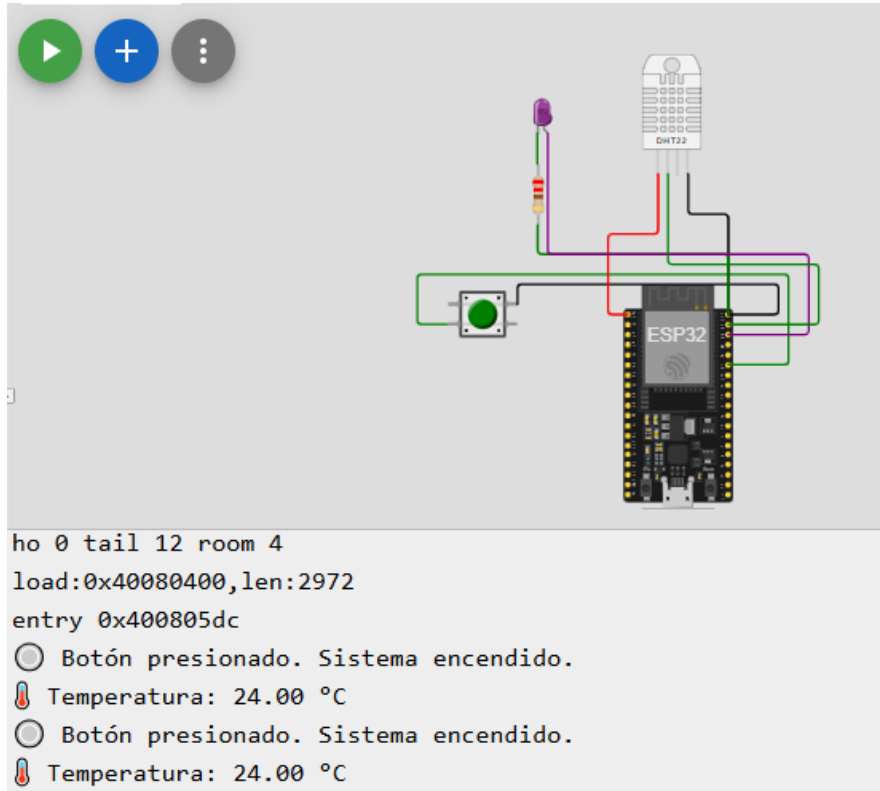


4. Simular un sensor de temperatura, una luz con dimer, un botón de encendido y apagado mediante Wokwi, Proteus, LabView; etc con ESP32.



Link: <https://wokwi.com/projects/428875384804434945>

Componentes Utilizados

- **ESP32 DevKit C v4**
 - Microcontrolador utilizado para manejar las entradas/salidas y la comunicación serial.
- **Sensor DHT22**
 - Sensor utilizado para medir la temperatura y humedad ambiente. En este caso, solo se utiliza para medir la temperatura.
 - **Pin de conexión:** GPIO 23
- **Botón**
 - Se utiliza para encender o apagar el sistema de monitoreo de temperatura.
 - **Pin de conexión:** GPIO 21
- **LED**
 - Se utiliza para indicar el estado del sistema mediante un parpadeo cuyo ritmo está determinado por la temperatura medida por el sensor DHT22.
 - **Pin de conexión:** GPIO 22
- **Resistencia (220Ω)**
 - Se usa en serie con el LED para limitar la corriente que pasa a través de él y evitar daños al componente.

Conexiones del Sistema

Componente	Pin en ESP32	Función
Sensor DHT22 (VCC)	3.3V	Alimentación del sensor
Sensor DHT22 (GND)	GND	Tierra del sensor
Sensor DHT22 (NC)	GPIO 23	Lectura de temperatura
Botón (1º terminal)	GND	Tierra del botón
Botón (2º terminal)	GPIO 21	Entrada del botón (activación)
LED (Ánodo)	GPIO 22	Salida de la señal del LED
LED (Cátodo)	Resistencia (220Ω)	Limitador de corriente del LED
Resistencia (220Ω)	GND	Tierra para el LED

Funcionamiento del Sistema

- Encendido del Sistema:**
 - El sistema está inicialmente apagado.
 - Al presionar el **botón**, el sistema se enciende y comienza a medir la temperatura ambiente utilizando el **sensor DHT22**.
- Lectura de la Temperatura:**
 - El **sensor DHT22** mide la temperatura ambiente y la envía al microcontrolador ESP32 a través del pin **GPIO 23**.
 - Si la lectura es válida, la temperatura se imprime en el monitor serial para su visualización. Si hay un error en la lectura, se muestra un mensaje de error.
- Control del LED:**
 - El **LED** comienza a parpadear cuando el sistema está encendido.
 - La velocidad del parpadeo del LED depende de la temperatura medida. A medida que la temperatura aumenta, el tiempo de espera entre los ciclos de encendido y apagado del LED disminuye, haciendo que parpadee más rápido.
 - La relación es tal que temperaturas más altas (30-40°C) harán que el LED parpadee más rápido, mientras que temperaturas más bajas (20-25°C) resultarán en un parpadeo más lento.
- Apagado del Sistema:**
 - Si el **botón** no está presionado, el sistema permanece apagado, y el LED se apaga.

Diagrama de Conexiones

```

ESP32      | Componente
-----|-----
GPIO 23 ---->| DHT22 (NC)
GPIO 21 ---->| Botón (2º terminal)
GPIO 22 ---->| LED (Ánodo)
GND ---->| DHT22 (GND), Botón (1º terminal), LED (Cátodo)
  
```

3.3V ----->| DHT22 (VCC)
220Ω Resistor ----->| LED (Cátodo)

Diagrama de Funcionamiento:

1. **Encendido por Botón:**
 - El botón en **GPIO 21** cuando se presiona envía una señal de **HIGH** que activa el sistema.
 2. **Lectura de Temperatura:**
 - El sensor DHT22 se comunica con el microcontrolador a través de **GPIO 23** para proporcionar la temperatura medida.
 3. **Control de LED:**
 - El LED en **GPIO 22** parpadea a diferentes velocidades en función de la temperatura, controlado por los valores de **delayTime** en el código, mapeados a partir de la temperatura.
-

Resumen de Comportamiento

- **Botón presionado:** Enciende el sistema, permitiendo la lectura de la temperatura.
- **Temperatura medida:** Se visualiza en el monitor serial y se ajusta la velocidad de parpadeo del LED.
- **Botón no presionado:** El sistema permanece apagado y el LED se apaga.