

# Diseño e Implementación de Sistemas Embebidos, Laboratorio 2.

Nombre: Joan Esteban Velasco Larrea.

## Reto 1 – ESP32 Standalone Web Server

### Análisis de Requerimientos.

#### Objetivo:

Diseñar e implementar un servidor web HTTP autónomo ejecutándose sobre un ESP32, capaz de monitorear variables ambientales (temperatura y humedad) y permitir la interacción del usuario a través de una interfaz web accesible desde cualquier dispositivo dentro de la misma red local. El sistema debe ser no bloqueante, confiable y fácilmente extensible.

#### Requerimientos Funcionales.

1. Monitoreo Ambiental
  - El sistema debe leer temperatura y humedad relativa usando un sensor DHT22.
  - Las variables deben actualizarse de manera periódica.
  - Los valores deben:
    - Mostrarse en una interfaz web
    - Enviarse por el monitor serial
    - Visualizarse localmente en una pantalla OLED
2. Interfaz de Usuario (Web)
  - El sistema debe proporcionar una página web http alojada directamente en el ESP32.
  - La interfaz debe permitir al usuario:
    - Ingresar el setpoint de velocidad del motor (RPM).
    - Visualizar el valor del RPM.
  - El hardware del motor no es obligatorio, por lo que el enfoque está en la interfaz y el flujo de datos.
  - El setpoint ingresado debe mostrarse en tiempo real en el monitor serial.
3. Requisitos de la Interfaz Web
  - La página debe ser responsive, accesible desde celulares, tablets y computadores.
  - La interfaz debe mostrar claramente:
    - Temperatura
    - Humedad
    - Campo de entrada para RPM

- Timestamp de la última actualización (hora obtenida vía NTP).

## Requerimientos no Funcionales.

1. Conectividad:
  - El sistema debe conectarse a una red WiFi local.
  - El servidor HTTP debe operar en el puerto 80.
2. Confiabilidad:
  - El sistema debe mantener la operación continua del servidor web.
  - La lectura de sensores no debe bloquear la atención de clientes HTTP.
  - El timestamp debe mantenerse sincronizado usando NTP.
3. Simplicidad y Escalabilidad:
  - La arquitectura debe separar claramente:
    - Adquisición de datos
    - Servidor Web
    - Interfaz de usuario

## Análisis y Diseño del sistema (Arquitectura y Componentes).

### Arquitectura General.

El sistema está compuesto por los siguientes bloques funcionales:

- Capa de Sensado
  - Sensor DHT22 (Temperatura y Humedad)
- Capa de Procesamiento
  - ESP32 (Control lógico, servidor web y sincronización NTP)
- Capa de Interfaz Local
  - Pantalla OLED SSD1306 (I2C).
- Capa de Comunicación
  - WiFi (HTTP + NTP)
- Capa de Usuario
  - Navegador Web (PC o Móvil).

### Componentes Utilizados

- Microcontrolador ESP32
- Sensor DHT22
- Pantalla OLED SSD1306 (128x64, I2C).
- Red WiFi local
- Navegador Web
- Entorno de Desarrollo Arduino IDE.

## Diseño e Integración de Hardware.

El diseño de este reto se separó en dos partes, una para la medida de temperatura y humedad (figura 1a) y otra para el control del motor (figura 1b). En la figura 1a el potenciómetro representa un sensor lm35.

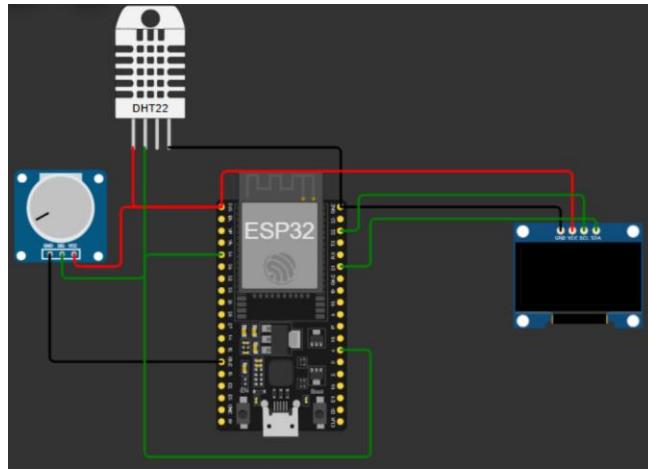


Figura 1a. Arquitectura de hardware utilizada, temperatura y humedad.

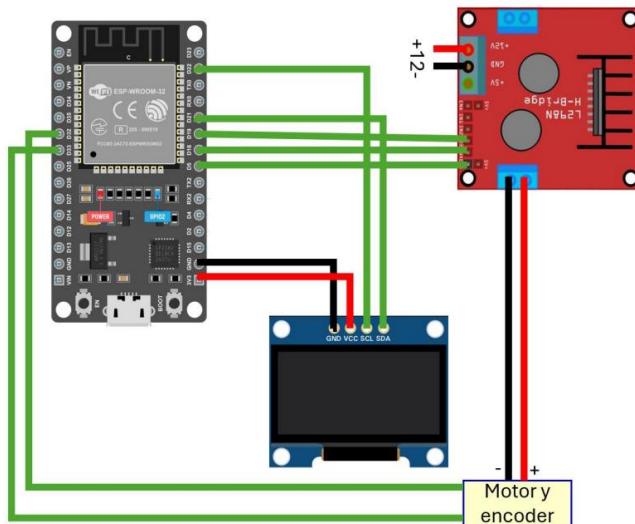


Figura 1b. Arquitectura de hardware utilizada, control motor.

## Desarrollo del firmware.

### Organización del Código.

El reto se implementa en dos variantes funcionales:

1. ex\_1\_1.ino – Monitor Ambiental Web
  - Lectura del sensor DHT22.
  - Servidor Web HTTP con endpoint /data.

- Envío de datos en formato JSON.
  - Interfaz web que actualiza datos mediante fetch().
  - Sincronización de hora NTP para generar timestamps.
  - Visualización local en pantalla OLED.
2. ex\_1\_2.ino – Interfaz Web para Setpoint de Motor
- Página web con campo de entrada para RPM
  - Envío del setpoint en el monitor serial
  - Arquitectura preparada para integración con control de motor (PID).
  - Uso de NTP para timestamp de actualización.
  - Máquina de estados para control no bloqueante.

Los códigos correspondientes se encuentran en la carpeta ex1 del repositorio en el apartado de referencias.

## Interfaces y Protocolos

- Interfaces Hardware
  - I2C
    - OLED SSD1306
    - SDA GPIO 21
    - SCL GPIO 22
  - GPIO
    - DHT22 GPIO15
  - ADC
    - Entrada analógica para temperatura adicional.
- Protocolos de comunicación
  - HTTP
    - / (página principal)
    - /data (datos ambientales)
    - /set\_rpm (Recepción de setpoint)
    - /status (estado del sistema)
  - NTP
    - Sincronización horaria para timestamps
  - WiFi
    - Conexión a red local

## Validación y Resultados.

- El servidor web es accesible desde cualquier dispositivo conectado a la red.
- Los valores de temperatura y humedad se actualizan periódicamente.
- El timestamp refleja correctamente la hora local sincronizada por NTP.

- El setpoint ingresado desde la web se recibe correctamente y se muestra por Serial.
- El sistema opera de forma estable sin bloqueos.

Se logró implementar exitosamente un **servidor web autónomo en ESP32**, integrando sensado ambiental, visualización local, sincronización temporal y una interfaz web interactiva.

El diseño modular y basado en estados permite escalar el sistema hacia aplicaciones más complejas como control remoto de actuadores reales o integración con plataformas IoT.

## Referencias.

Repositorio

[https://github.com/joanvel/Lab02\\_Diseno\\_e\\_Implementacion\\_de\\_Sistemas\\_Embebidos/tree/main](https://github.com/joanvel/Lab02_Diseno_e_Implementacion_de_Sistemas_Embebidos/tree/main)

Github: