***Smart Temperature Control System***

**By Barasa Mercy**

**24/U/4881/GIM**

**Introduction**

In Northern Uganda where temperature levels directly affect product quality such as fashion boutiques, shoe storage, or material warehouses maintaining optimal conditions is essential. This project presents a smart, Internet of Things based solution using the ESP32 microcontroller to monitor and control ambient temperature and humidity. It integrates sensor data, automatic fan control, and a user-friendly web interface, offering both automation and remote access.

**Problem Statement**

Traditional temperature control systems often lack flexibility, remote monitoring, or intelligent automation. For small businesses or personal storage spaces, installing expensive HVAC systems is impractical. This project solves several key challenges:

* **Manual Monitoring:** Eliminates the need for constant human supervision
* **Overheating Risks:** Automatically activates cooling when temperature exceeds safe limits especially in northern regions.
* **Remote Access:** Allows users to monitor and control the system via Wi-Fi
* **User Feedback:** Provides real-time data and control through a web interface

**Technical Description**

**Core Components**

* **ESP32-WROOM-32:** A powerful microcontroller with built-in Wi-Fi, ideal for IoT applications
* **DHT11 Sensor:** Measures temperature with digital output
* **Fan (Actuator):** Controlled via GPIO pin to regulate temperature
* **WebServer Library:** Hosts a local web interface accessible via IP
* **HTML/CSS Interface:** Displays sensor data and control buttons for user interaction

**Circuit Design**

The ESP32 is powered via USB and connected to the DHT11 sensor on GPIO pin 23. A fan is connected to GPIO pin 19 and controlled using digital signals. The system uses pull-up resistors and capacitors for stability, and a CH340G USB-to-serial converter for programming.

**Software Logic**

* **Sensor Reading:** The DHT11 provides temperature data every few seconds
* **Fan Control:** If temperature exceeds 25°C, the fan turns on automatically. It turns off when temperature drops below 18°C
* **Web Interface:** Users can view live data and manually toggle the fan using buttons
* **Refresh Function:** Updates sensor readings without reloading the entire page
* **Time Display:** Shows last update time using millis() for tracking system activity