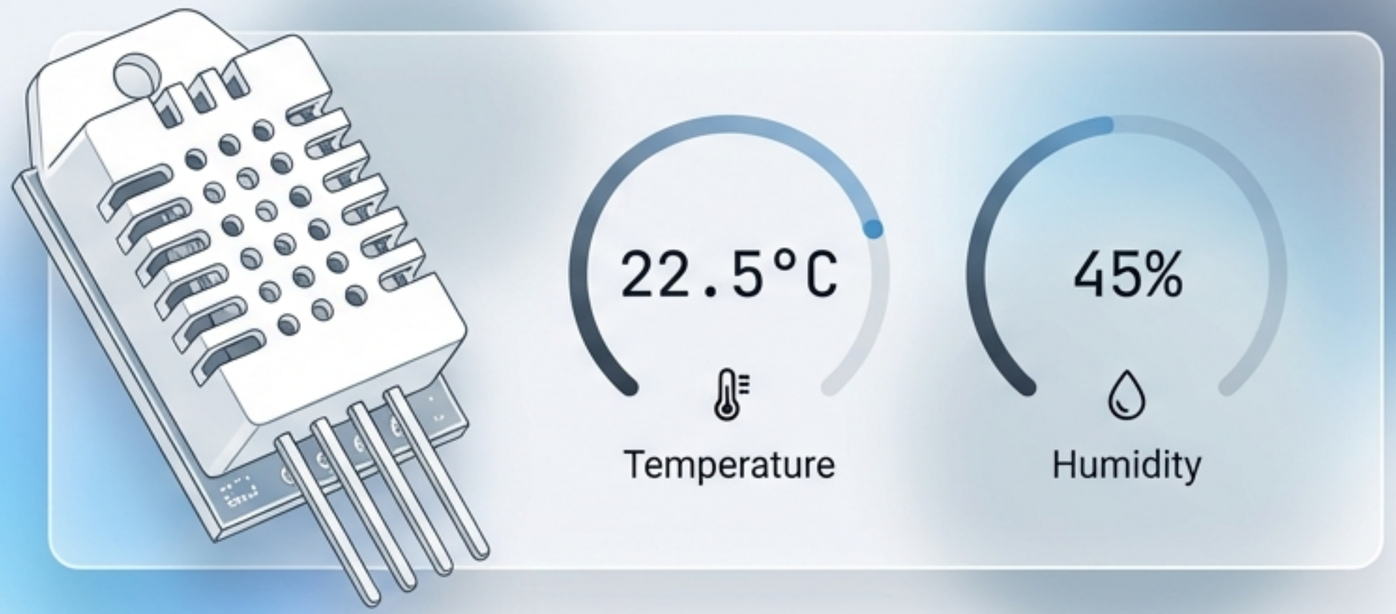
IoT Environmental Intelligence

A Real-Time Monitoring, Analysis, and Alert System



Project by Ravi Kumar Thekare
ProIT - IoT Project (WS2526)
HTW Berlin

Beyond Basic Sensing

This comprehensive IoT system measures real-time conditions using a DHT22 sensor and ESP32, transforming raw inputs into complex environmental insights via Node-RED logic.

Precision

Real-time acquisition of **Temperature** and **Humidity** using the **DHT22** sensor and **ESP32** microcontroller.

Intelligence

Calculation of advanced metrics including **Dew Point** (**Magnus-Tetens**), **Absolute Humidity**, and **Saturation Depression**.

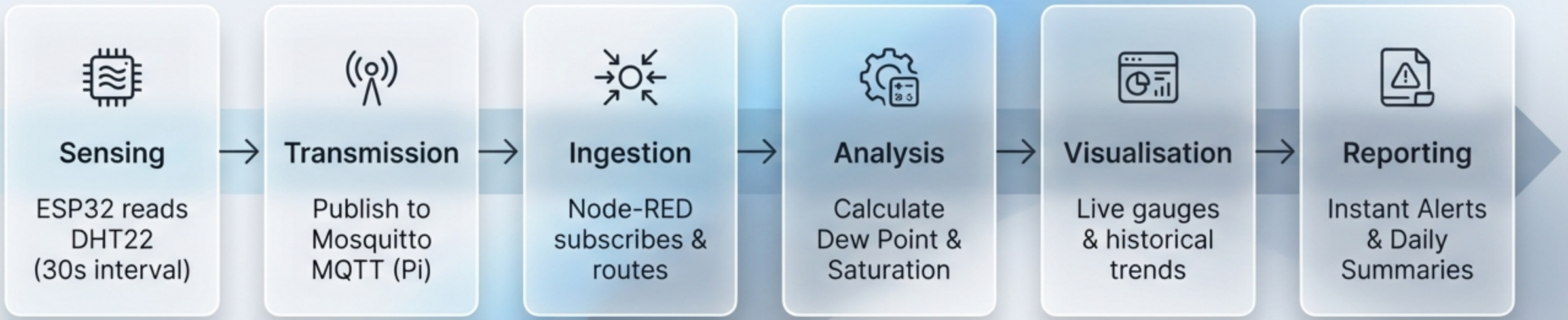
Connectivity

Automated Telegram integration for immediate threshold alerts and a comprehensive **Bot Command Suite**.

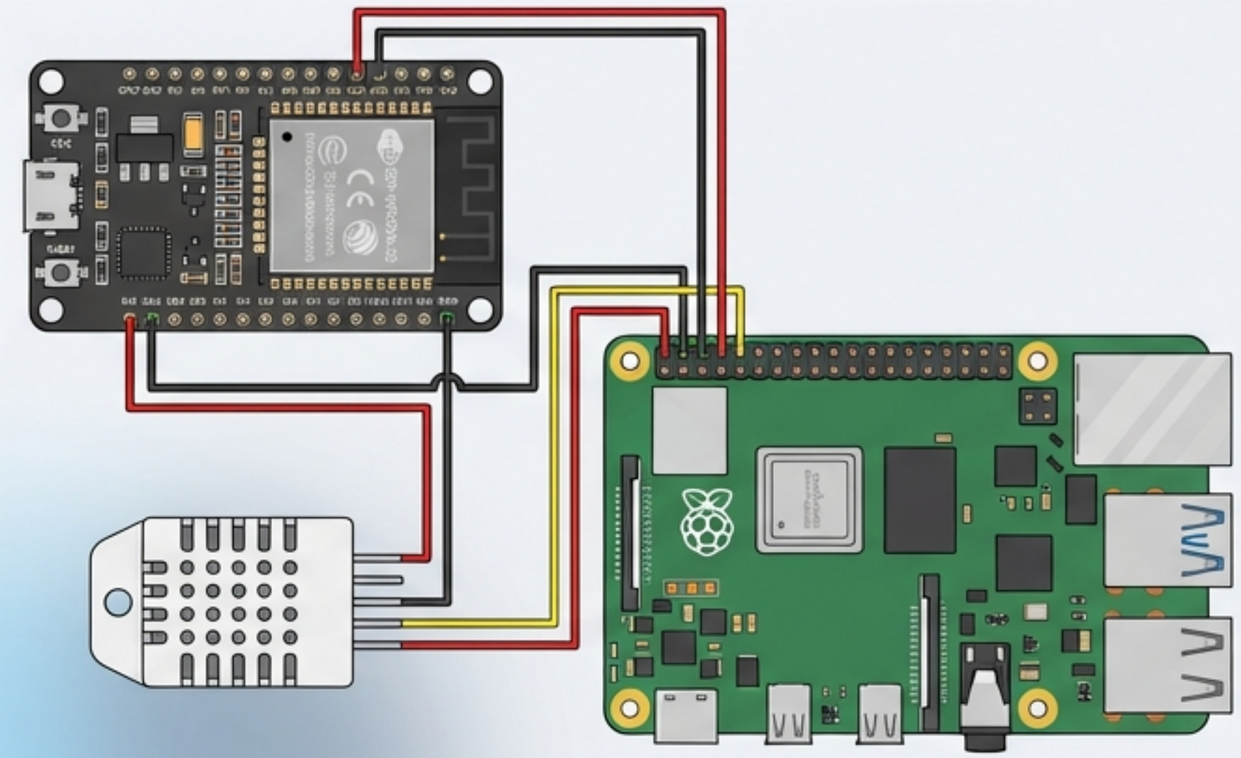
Visualisation

A mobile-responsive dashboard featuring a modern custom **Glassmorphism UI**.

The Architecture of Data Flow



The Hardware Stack



Microcontroller

ESP32 Development Board
(The Edge Node)

Sensor

DHT22 (Temperature & Humidity)

Processing Unit

Raspberry Pi (Hosting Broker & Node-RED)

Power

Standard USB 5V supply

The ESP32 acts as the field agent, capturing and forwarding environmental data to the central server (Pi) with minimal power consumption.

Firmware Logic and The Loop

Developed in Thonny IDE with MicroPython

1. Initialise

Import MQTT libraries, define pins

2. Connect

WiFi handshake, then MQTT Broker connection

3. The Loop

Repeats every 30s

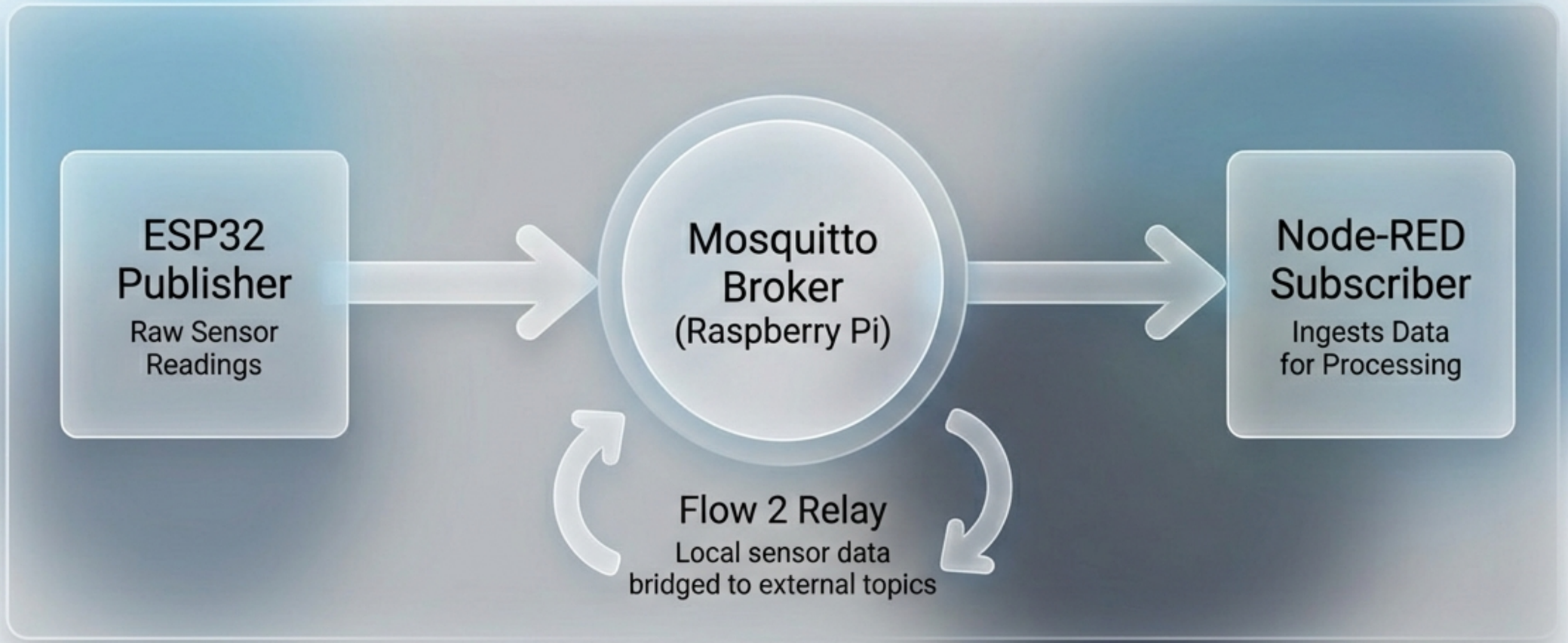
```
while True:
    try:
        sensor.measure()
        temp = sensor.temperature()
        hum = sensor.humidity()

        client.publish(topic_temp, temp)
        client.publish(topic_hum, hum)

        sleep(30)
    except OSError:
        reset()
```

The firmware ensures a 'fire and forget' mechanism, keeping the edge device lightweight.

The Nervous System: MQTT & Mosquitto



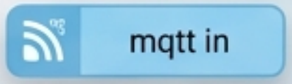
The Mosquitto Broker acts as the central post office. The ESP32 publishes readings, and Node-RED subscribes to them, decoupling the hardware from the software logic.

The Node-RED Ecosystem

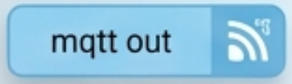
Node-RED on the Raspberry Pi handles all logic, routing, and UI generation.

Palette

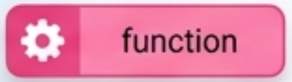
Logic & Routing
Inter



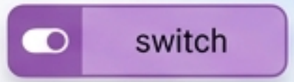
JetBrains Mono



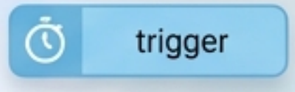
mqtt out



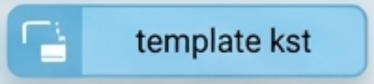
function



switch

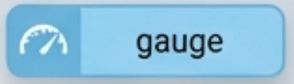


trigger

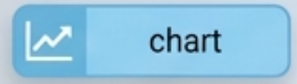


JetBrains Mono

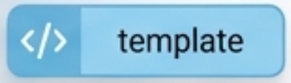
Dashboard UI
Inter



gauge

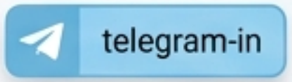


chart

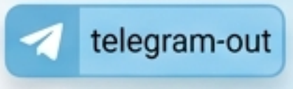


(Glassmorphism CSS)
JetBrains Mono

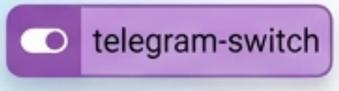
Telegram Integration
Inter



telegram receiver

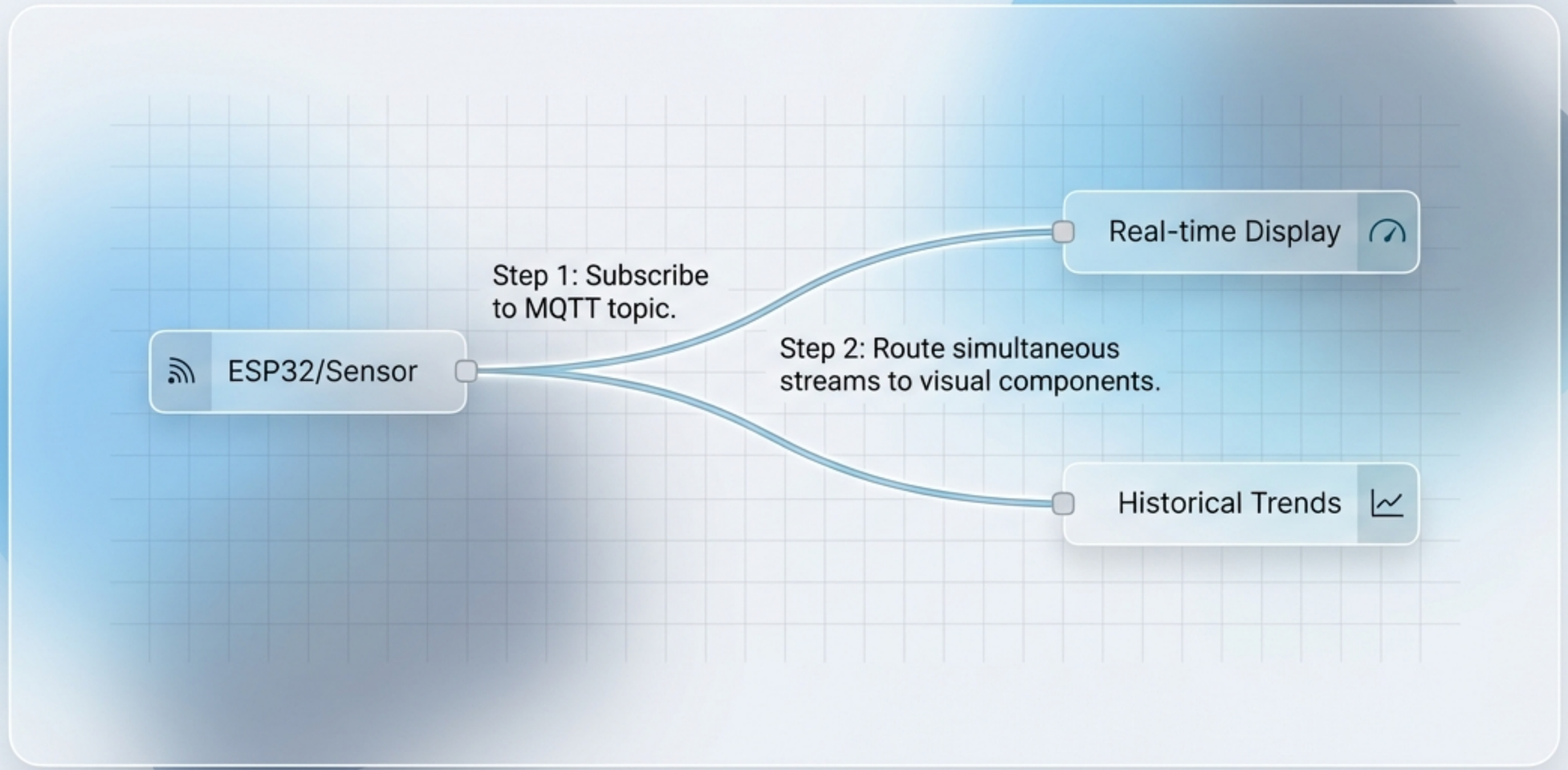


telegram sender



telegram switch

Ingestion: Climate Streams (Flow 1)



Advanced Environmental Analysis

$$Tdp = \frac{b * \alpha(T,RH)}{a - \alpha(T,RH)}$$

Dew Point

Calculated via the Magnus-Tetens formula. Indicates the temperature at which air becomes saturated.

Absolute Humidity

The total mass of water vapour present in a given volume of air.

Saturation Depression

The difference between current temperature and dew point.

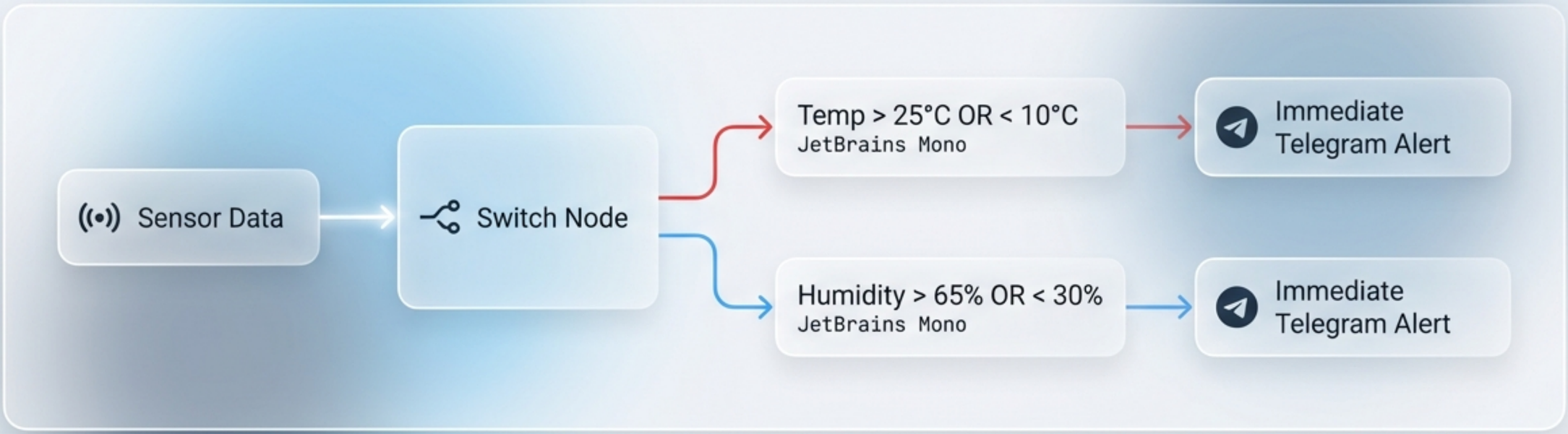
Visualisation: The Glassmorphism Dashboard

Custom CSS injection via `ui_template` for frosted glass effect.

Live Gauges for instant status.

Mobile-responsive design accessed via local network.

Logic for Immediate Action



The system uses `switch` nodes to monitor safety boundaries. Critical breaches trigger separate and immediate alerts via the `Telegram Bot API`.

Two-Way Communication: The Telegram Bot

Roboto

`JetBrains Mono`

Utilises `telegram-in` and `telegram-out` nodes to create a conversation loop.

The user can query system health remotely without accessing the web dashboard.

Intelligent Reporting Strategy

Balancing Urgency with Usability

Immediate Alerts

Triggered ONLY by threshold breaches.

Purpose: Urgent Action.

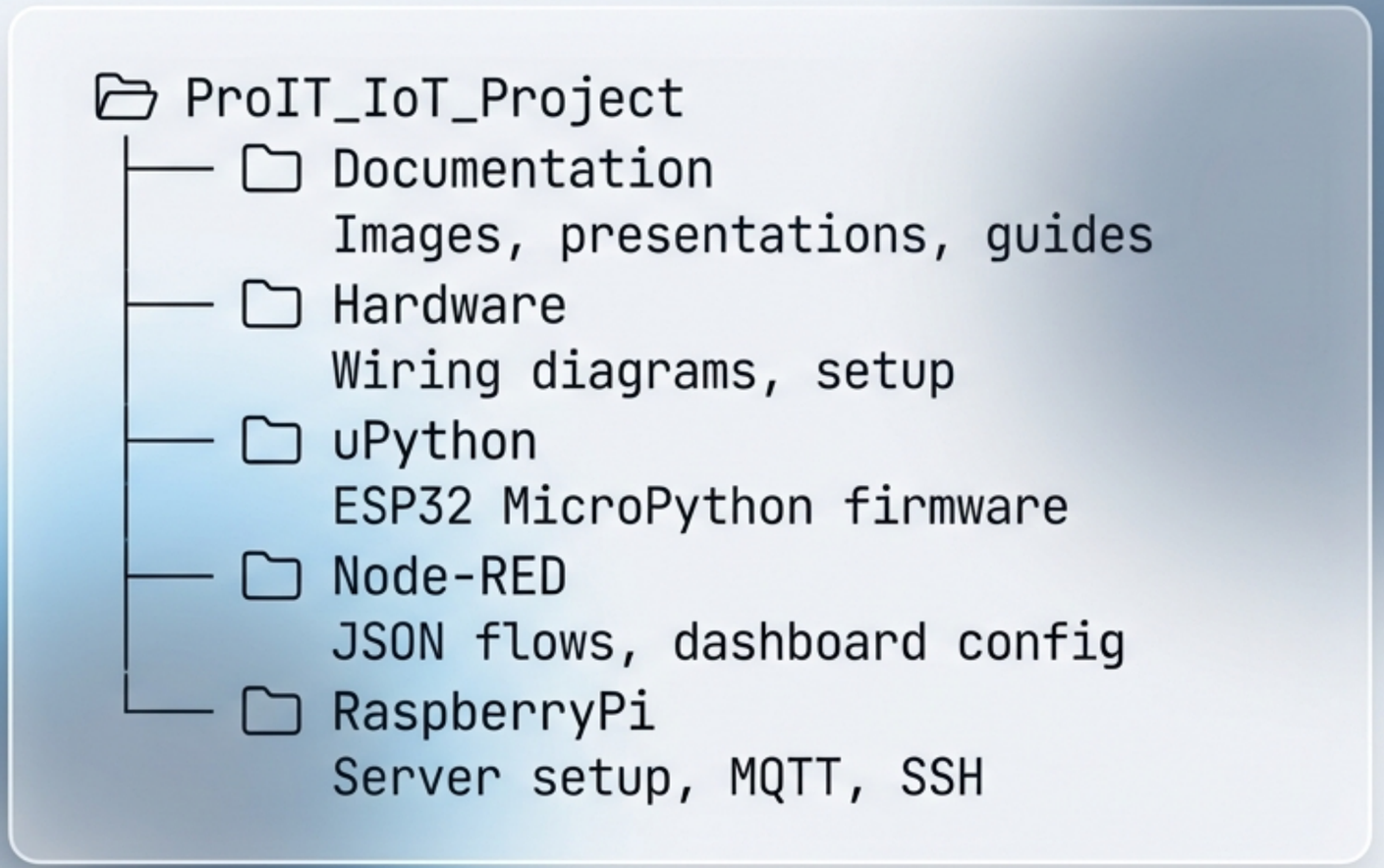
Daily Report

Consolidated 'Environmental Analysis' summary.

Purpose: Comprehensive overview.

Sent once per 24 hours via scheduler nodes to prevent notification fatigue.

Project Structure & Implementation



A well-documented, open-source engineering effort designed for reproducibility.

Summary & Acknowledgments

This project successfully integrates low-cost hardware with sophisticated software logic, delivering a system that offers real-time monitoring, complex environmental derivation, and intelligent user alerting.

Author: Ravi Kumar Thekare
Context: ProIT - IoT Project (Winter Semester 2025/26)
Institution: HTW Berlin

Empowering users with environmental awareness through intelligent automation.