ALLOWEAR LITE - IOT-BASED MATERNAL HEALTH VITAL MONITOR

INTRODUCTION:

This project is a cloud based simulation of an IoT based maternal health monitoring system inspired by the SaveMom Allowear product. The objective is to design and demonstrate wearable style prototype capable of continuously sensing and transmitting vital parameters such as body temperature, heart rate, heart rate variability, motion/steps, stress level and baby cry detection. The system is fully implemented online using ESP32 in simulated in Wokwi programmed in PlatformIO (VS code) and integrated with the public MQTT broker (HiveMQ), the live readings are visualised from Node-red dashboard that represents a remote maternal health monitoring setup.

DATA FLOW:

Architecture explanation

The data flow follows a cloud connected IoT structure:

1. ESP32 controller reads multiple sensors (temperature, heart rate, motion, sound)
2. Data is packaged as a JSON string and published on the MQTT topic (celsia/health/data) hosted on HiveMQ public broker
3. Node red is subscribed to the same topic, process this data
4. Dashboard widgets (gauges, charts and alerts) display live readings

Data Flow Diagram

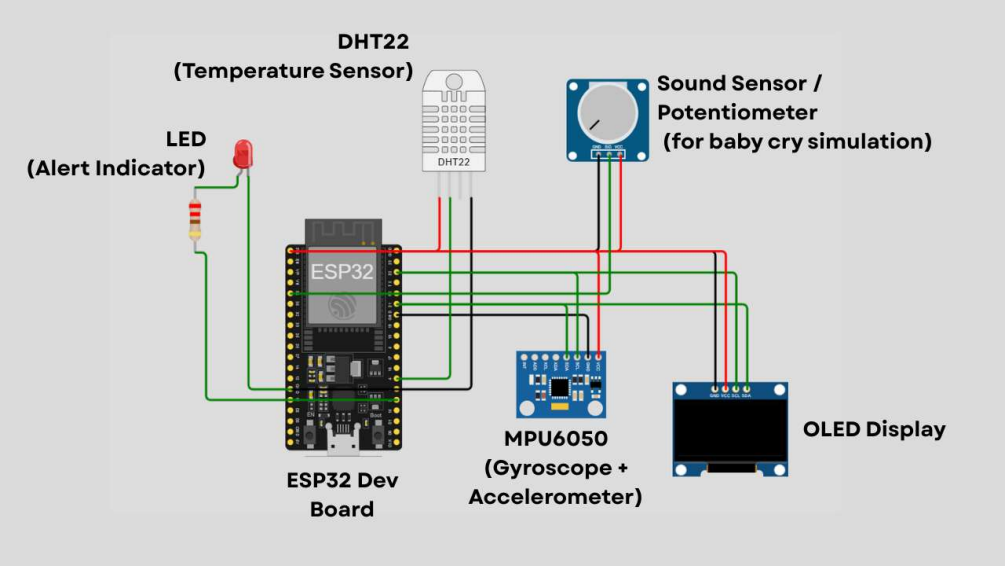
Sensors → ESP32 → MQTT Broker (HiveMQ) → Node-RED Dashboard → Alerts/Visualization

HARDWARE SETUP AND CIRCUIT CONFIGURATION:

This prototype is implemented online and hardware mapping is given below,

Component	Pin(ESP32)	Connection Purpose
DHT22 (Temp Sensor)	GPIO 4	Temperature data
MPU6050(Gyroscope)	SDA – GPIO 21, SCL – GPIO 22	Motion and Step Tracking
OLED Display	SDA – GPIO 21, SCL – GPIO 22	Optional Local display
Potentiometer	GPIO 34 (Analog Input)	Simulated Baby Cry input
LED (Alert)	GPIO 2	Visual Alert Indicator

All components share a common ground(GND) & powered from ESP32's 3.3V pin.



Circuit Diagram of Online ESP32 Setup (Wokwi Simulation)

SOFTWARE IMPLEMENTATION :

Platforms Used

- Wokwi - circuit simulation and pin testing
- PlatformIO (vs code) – Code compilation and firmware generation
- HiveMQ Broker - MQTT Cloud Communication.
- Node-Red - Dashboard and visualisation interface

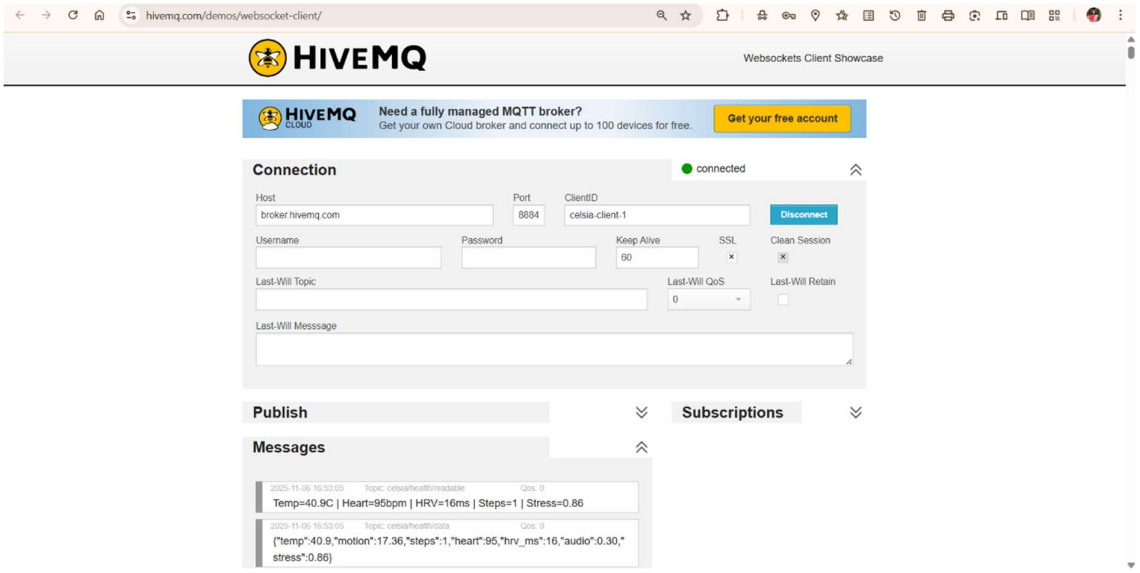
MQTT details

Field	Value
Host	broker.hivemq.com
Port	1883
Client Id used	Celsia-client-1
Topic	celsia/health/data
Protocol	MQTT v3.1
Authentication	None (public broker)

MQTT Web client:

Used for testing and viewing live data:

https://www.hivemq.com/demos/websocket-client/



MQTT Broker Connected and Live data messages visible in HiveMQ Client

Source code outline:

The ESP32 program reads simulated or sensor values, formats them as JSON, and publishes via MQTT. The main.cpp is the source file that contains the program. firmware.elf and firmware.build is generated from the PlatformIO project. wokwi.toml is the file generated from the Wokwi online simulator.

Example Jason payload:

{"temp": 36.8, "heart": 74, "steps": 2035, "stress": 0.45, "hrv": 47.0, "audio": 0.2}

```
PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS

Connecting to WiFi..
                WiFi connected. IP: 10.13.37.2
Connecting to MQTT ... connected
Published JSON: {"temp":38.4,"motion":17.36,"steps":1,"heart":75,"hrv_ms":3,"audio":0.30,"stress":0.69}
Published Text: Temp=38.4C | Heart=75bpm | HRV=3ms | Steps=1 | Stress=0.69
Published JSON: {"temp":38.4,"motion":17.36,"steps":1,"heart":75,"hrv_ms":4,"audio":0.30,"stress":0.69}
Published Text: Temp=38.4C | Heart=75bpm | HRV=4ms | Steps=1 | Stress=0.69
```

Code features

- Auto Wi-Fi connection to “Wokwi-Guest” Network
- MQTT Reconnect Logic
- Randomized Heart Rate, HRV, stress and baby cry signals for simulation
- Publishing interval

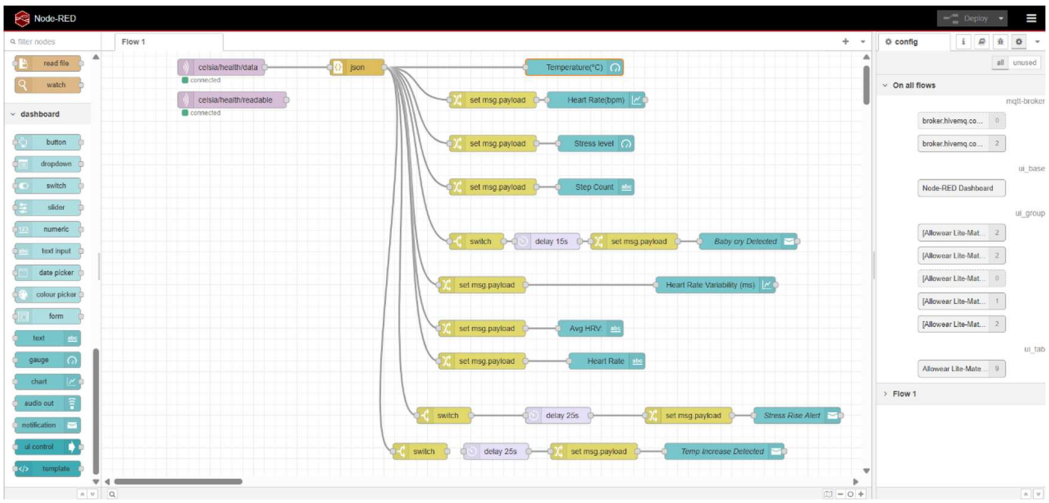
DASHBOARD AND VISUALISATION

Platform

Node-RED (hosted locally on http://localhost:1800/ui)

Flow configuration

Nodes used: mqtt in, json, change, gauge, chart, text, notification

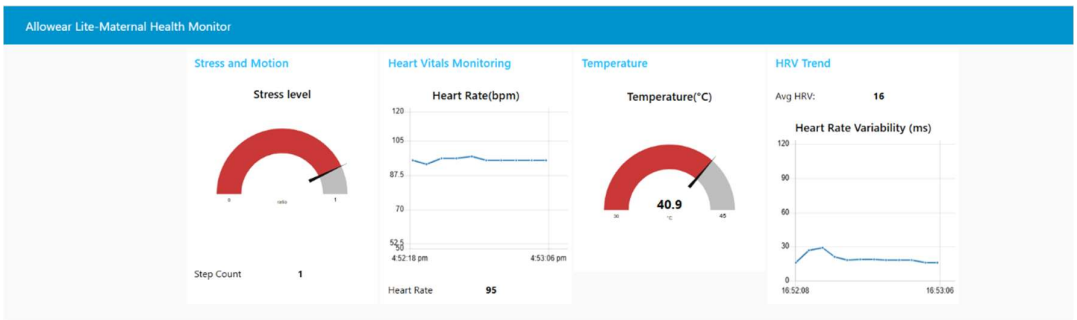


Node-Red Flow Design

DASHBOARD VISUALISATION

Each sensor output is represented by visual elements:

- Temperature(°C): Gauge with range 30-45 °C
- Heart rate(bpm): Chart with varying values per minute
- HRV(ms): Chart with varying values per minute
- Stress level: Gauge
- Steps count: Text display
- Baby cry alert: Popup notification



Node-Red Dashboard displaying live sensor data

POWER SUPPLY

Even though the system is simulated online, the prototype is built around low power hardware ESP32. Its operating voltage 3.3 V and Max current drawn is less than 500 mA.

INNOVATIVE FEATURES IMPLEMENTED

Feature	Description
HRV calculation	Simulated heart rate variability tracking (advanced cardiac indicator)
Stress Estimation	Derived from HRV and heart rate correlation
Baby cry detection	Potentiometer-based analog simulation for Sound triggered alert

$$HRV = \sqrt{\sum \frac{(RRi - \quad)^2}{N-1}}$$

Where: RRi – Time interval btw the i-th and (i+1)-th heartbeat

RR – Mean RR Intreval; N – Number of Samples

Stress = HR/HRV

CONCLUSION

This Allowear Lite Project has demonstrated the IoT based Maternal Health Monitoring, entirely simulated online. It replicates the key idea behind SaveMom Allowear, continues maternal health tracking while integrating additional analytics like HRV and stress detection.