**Relatório CardioIA Fase3 – Parte 2 — Fluxo de Comunicação MQTT e Configuração do Dashboard**

**1. Introdução**

O presente relatório descreve o fluxo de comunicação MQTT e a configuração do painel de monitoramento (dashboard) desenvolvidos na **Parte 2 do projeto CardioIA**, que tem como objetivo simular a coleta e transmissão de dados fisiológicos — batimentos cardíacos (BPM), temperatura corporal e umidade — utilizando um **ESP32** em conjunto com o **protocolo MQTT** e o **Node-RED Dashboard**.

Essa integração permite visualizar em tempo real os dados enviados pelo dispositivo, identificar variações críticas e emitir alertas automáticos, simulando o funcionamento de um sistema de monitoramento remoto de saúde baseado em Internet das Coisas (IoT).

**2. Fluxo de Comunicação MQTT**

O protocolo **MQTT (Message Queuing Telemetry Transport)** foi escolhido por ser leve, eficiente e amplamente utilizado em aplicações IoT. Ele segue o modelo **publicador/assinante (publish/subscribe)**, onde os dispositivos publicam mensagens em tópicos específicos e outros componentes (como o Node-RED) se inscrevem nesses tópicos para receber os dados.

No contexto do projeto, o fluxo funciona da seguinte forma:

1. **Publicador (ESP32 ou simulação):**  
   O microcontrolador ESP32 coleta (ou simula) os valores de temperatura, umidade e batimentos cardíacos. Esses dados são formatados em JSON, por exemplo:
2. {"temperature": 36.8, "humidity": 47.4, "bpm": 91}

Em seguida, o ESP32 publica essas informações no **tópico MQTT**:

cardioIA/vitor/telemetry

1. **Broker MQTT (HiveMQ Cloud):**  
   O broker atua como intermediário entre o publicador e os assinantes.  
   Neste projeto, foi utilizado o **HiveMQ Cloud**, que oferece um servidor gratuito e seguro com conexão TLS (porta 8883).  
   Ele recebe as mensagens publicadas e as repassa para todos os sistemas inscritos no mesmo tópico.
2. **Assinante (Node-RED):**  
   O **Node-RED** foi configurado como cliente MQTT assinante do tópico cardioIA/vitor/telemetry.  
   Cada vez que o ESP32 publica uma nova mensagem, o Node-RED recebe os dados instantaneamente, permitindo sua exibição e análise no painel.

Esse ciclo de **publicação, intermediação e assinatura** caracteriza o funcionamento essencial do protocolo MQTT, garantindo comunicação contínua e em tempo real entre o dispositivo e a interface visual do usuário.

**3. Configuração do Node-RED e Dashboard**

O **Node-RED** foi utilizado tanto para o processamento das mensagens MQTT quanto para a criação da interface de monitoramento. O fluxo desenvolvido é composto pelos seguintes nós principais:

* **Inject + Function:** geram e formatam os dados simulados (quando não há ESP32 real).
* **MQTT Out:** publica as informações no broker HiveMQ.
* **MQTT In:** recebe as mensagens publicadas e as encaminha para o dashboard.
* **Converte JSON:** converte o texto recebido em formato JSON para leitura dos valores individuais.
* **Gauge (BPM):** exibe graficamente o valor dos batimentos cardíacos.
* **Chart (Temperatura/Umidade):** mostra a variação das medições em tempo real.
* **Texto (Alerta):** apresenta mensagens automáticas de alerta quando o BPM ultrapassa 100 batimentos por minuto ou quando a temperatura é superior a 38 °C.

Os nós são interligados de forma sequencial, de modo que cada dado recebido é imediatamente processado e refletido na interface visual do usuário.  
O **painel do Dashboard**, acessado pelo endereço http://127.0.0.1:1880/ui, foi organizado com o título **“CardioIA Monitor”**, contendo indicadores de BPM, temperatura e umidade atualizados continuamente.

**4. Resultados e Conclusão**

A configuração implementada possibilitou estabelecer uma comunicação estável entre o **dispositivo publicador (ESP32 ou simulado)** e o **servidor Node-RED** via **MQTT**, validando o fluxo completo de dados IoT.

O painel desenvolvido permite **monitorar em tempo real** as condições do “paciente” e emitir **alertas automáticos** em caso de anomalias, demonstrando um caso prático de integração entre hardware, nuvem e interface gráfica.

Além de cumprir os requisitos propostos na Parte 2 do projeto, a arquitetura criada oferece base sólida para futuras expansões, como armazenamento histórico, integração com bancos de dados e notificações via aplicativos de mensagens.

Assim, o trabalho evidencia o potencial do uso combinado de **ESP32, MQTT e Node-RED** em soluções acessíveis e escaláveis para **monitoramento remoto de parâmetros vitais** no contexto de saúde digital.