

# **Relatório PARTE 01: Fluxo de Funcionamento e Lógica de Resiliência**

# Projeto: Sistema Vestível de Monitoramento Cardíaco (Edge Computing)



## Resumo Executivo

O sistema é um protótipo de Edge Computing baseado em ESP32 (ESP32 DevKit) para monitoramento de sinais vitais. Ele prioriza a resiliência por meio de armazenamento de dados em sistema de arquivos local (LittleFS) e memória RAM, garantindo que a coleta de dados não seja interrompida mesmo na ausência de conectividade Wi-Fi/MQTT.

---

## 1. Arquitetura e Componentes de Hardware

O sistema é construído sobre a plataforma ESP32 e utiliza os seguintes componentes:

- **Microcontrolador:** ESP32 DevKit.
- **Sensores:**
  -  **DHT22:** Leitura de temperatura e umidade.
  -  **Botão (HEART\_RATE\_BUTTON):** Utilizado para simular a frequência cardíaca (BPM).
- **Indicadores de Status (LEDs):**
  - LED Azul (Wi-Fi)
  - LED Verde (MQTT)
  - LED Vermelho (Alertas).
- **Armazenamento Local:** LittleFS (substituindo o SPIFFS depreciado).






## 2. Fluxo de Coleta e Persistência de Dados (Edge Computing)

O fluxo de dados no Edge é executado periodicamente a cada 5 segundos (**SENSOR\_INTERVAL = 5000 ms**) e foca em nunca perder uma leitura:

1. **Leitura e Simulação:** O sistema lê DHT22 e obtém o valor da frequência cardíaca (**heartRate**), que é simulado com variação orgânica entre 68 e 115 BPM.
2. **Criação do Pacote:** As leituras são empacotadas na estrutura **SensorData** (**{temperature, humidity, heartRate, timestamp, sent}**).
3. **Persistência (Dupla Camada):**
  - **Memória RAM (Buffer):** O pacote é imediatamente gravado no **offlineBuffer**. A capacidade é de **1000 amostras (MAX\_STORED\_READINGS)**.
  - **LittleFS (Persistente):** O pacote é anexado ao arquivo **/sensor\_data.json** para garantir **persistência offline**.
4. **Publicação Imediata:** Se **ON-LINE** (**wifiConnected && mqttConnected**), o sistema tenta publicar o pacote imediatamente via MQTT.

### 3. Lógica de Resiliência Offline e Sincronização

A resiliência é gerenciada pela alternância simulada do status Wi-Fi e pela função de sincronização inteligente (`syncOfflineData()`).

Cenário	Status	Comportamento do Sistema
Wi-Fi/MQTT Desconectado	 OFFLINE	Coleta continua a cada 5s e armazena em RAM e LittleFS. LEDs de comunicação ficam desligados.
Wi-Fi Reconectado	 ONLINE	Restabelece a conexão MQTT. O LED Verde (MQTT) acende se for bem-sucedido.
Sincronização	 SYNC ATIVO	<code>syncOfflineData()</code> publica registros pendentes do LittleFS um a um.
Limpeza	 LIMPEZA SEGURA	Após todos os dados serem confirmados como enviados (registros no arquivo persistente), <code>clearOfflineData()</code> remove o arquivo <code>/sensor_data.json</code> .
Reinício com Falha	 RECOVERY	<code>loadOfflineData()</code> verifica <code>/sensor_data.json</code> no boot e recarrega <b>apenas</b> os dados pendentes ( <code>sent == false</code> ) no buffer RAM.