



**UNIVERSIDADE FEDERAL DE SANTA CATARINA**  
**CENTRO DE CIÊNCIAS, TECNOLOGIAS E SAÚDE DO CAMPUS ARARANGUÁ**  
**CURSO DE GRADUAÇÃO EM ENGENHARIA DE COMPUTAÇÃO**  
**Redes sem Fios - (2025.2)**

**Relatório Final:**

**Sistema de Monitoramento de Consumo de Água em Tempo Real**

**Autores:**

**André Gaspar & Ducher M Seidi**

**Araranguá-SC**

**2025**

## **1. Descrição e Objetivos do Projeto**

O projeto visa implementar um sistema integrado de monitoramento de consumo hídrico em tempo real, utilizando o avanço tecnológico para otimizar a aquisição de dados.

### **1.1. Estrutura e Componentes Gerais**

O sistema é composto por duas partes principais:

- **Unidade Medidora (Hardware):** Instalada em residências ou indústrias, utiliza um microcontrolador com conexão Wi-Fi (ESP 32) em conjunto com um sensor de vazão (YF-S201) para coletar dados.
- **Plataforma Web:** Administra e visualiza as informações coletadas.

### **1.2. Objetivo e Justificativa**

O intuito principal é criar um protótipo para acompanhamento remoto do consumo de água, disponibilizando ao usuário uma visão geral e elaborada do seu volume utilizado (medido em litros).

### **1.3. Funcionalidades para o Usuário**

A plataforma web permite aos usuários:

- Visualizar consumo em tempo real.
- Ver histórico de consumo por período.
- Receber alertas sobre vazamentos.

O fluxo geral do projeto é: Sensor de medição >> Banco de dados >> Aplicação web.

## **2. Levantamento Bibliográfico (Trabalhos Relacionados)**

A fase inicial consistiu em pesquisar trabalhos que propunham sistemas de monitoramento de consumo residencial em tempo real.

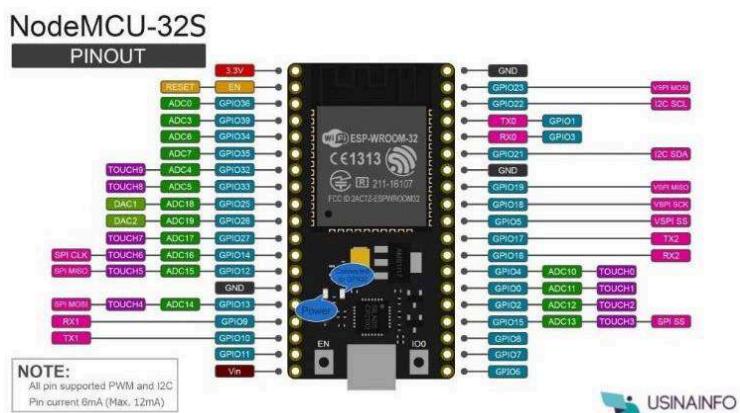
<b>Autor(es)</b>	<b>Ano</b>	<b>Tecnologias Chave</b>	<b>Foco Principal</b>
Tiago Duarte	2023	Sensor de vazão, Microcontrolador Wi-Fi, Firebase, Django (Python)	Visualização dos dados para administradores de concessionárias .

Souza, Nunes, Bianchini	2016	Raspberry Pi, Sensor de vazão, Python	Monitoramento em tempo real, plotagem de gráficos de consumo, e funcionalidade de "twittar" informações .
Dutra	2019	ESP826612-E, Sensor YF-S201, Firebase, Swift 4.0	Aplicativo mobile para iOS para acompanhamento remoto do consumo .
Vendemiatti	2020	Sensores de vazão e pressão, Microcontrolador Wi-Fi, API em PHP	Detecção de vazamentos e alterações em padrões de consumo residenciais/industriais .

### 3. Desenvolvimento e Componentes Utilizados

#### 3.1. Componentes de Hardware

- **Microcontrolador ESP32 (Modelo DevKit V1 NodeMCU-32S):** MCU de baixo custo e alto desempenho, escolhido por sua capacidade de processamento e transmissão de dados via Wi-Fi, sendo ideal para acompanhamento em tempo real .



Shutterstock

- **Sensor de Fluxo de Água YF-S201 (1–30 L/min):** Componente de medição que fica em série com o fluxo de água . Gera pulsos de tensão (saída de pulso de efeito Hall) proporcionais ao volume, através de uma roda de pás com um ímã .



### 3.2. Plataformas e Linguagens de Software

Categoria	Tecnologia	Função no Projeto
Firmware	Linguagem <b>C/C++</b> (via Arduino IDE)	Programação do ESP 32, leitura dos pulsos do YF-S201, conversão em volume (litros) e transmissão instantânea via Wi-Fi .
Backend	<b>Node.js v14+</b> (Express, Mongoose)	Gerenciamento de dados cadastrais, cálculo de consumo mensal e fornecimento da API de dados .
Frontend	<b>HTML, CSS, JavaScript, Biblioteca React 19</b> (Chart.js 4, jsPDF)	Construção da interface de usuário, exibição do consumo em tempo real, histórico e alertas .

<b>Banco de Dados</b>	<b>MongoDB 5.0+</b> (ou Firebase no planejamento inicial)	Armazenamento dos dados de consumo e cadastros. O MongoDB foi escolhido para dados de séries temporais .
-----------------------	---	--

## 4. Arquitetura do Sistema e Implementação Técnica

O sistema é uma solução IoT completa , seguindo um fluxo linear de dados em três camadas :

1. **Camada de Percepção (Hardware):** Coleta de dados físicos.
2. **Camada de Rede e Processamento (Backend):** Transporte, validação e armazenamento .
3. **Camada de Aplicação (Frontend):** Visualização e interação com o usuário.

### 4.1. Desenvolvimento de Hardware e Firmware

- **Implementação do Sensor:** A leitura no ESP 32 utiliza **interrupções de hardware** para precisão. O fator de conversão é de **450 pulsos = 1 litro** .
- **Sincronização Temporal:** O firmware usa **NTP (Network Time Protocol)** com a biblioteca **time.h** para obter o timestamp exato (GMT-3), essencial para a rastreabilidade dos dados.
- **Comunicação:** Os dados são transmitidos via protocolo **HTTP** (biblioteca **HTTPClient.h**) em formato **JSON**. O envio ocorre periodicamente (a cada 5 segundos).

### 4.2. Backend e Regras de Negócio

O servidor Node.js/Express gerencia endpoints RESTful e armazena os dados no MongoDB.

- **Cálculo de Consumo:** A lógica reside no servidor. O consumo é calculado dinamicamente pela diferença entre leituras acumuladas: **Consumo = Total Atual - Total Início Período**.
- **Segurança:** Implementada com **JSON Web Tokens (JWT)** para autenticação e hash criptográfico (**bcrypt js**) para proteção de senhas.

### 4.3. Frontend e Interface do Usuário

A interface React oferece um **Dashboard Interativo** (gráficos Chart.js), **Relatórios** (geração automática de PDF via jsPDF) e filtros temporais dinâmicos.

## 5. Resultados Obtidos e Desempenho (KPIs)

Os testes práticos confirmaram a viabilidade técnica, demonstrando os seguintes indicadores de desempenho:

1. **Latência:** Tempo médio entre leitura e atualização no gráfico **inferior a 5 segundos**.
2. **Precisão:** Margem de erro combinada de aproximadamente **(+/-)10%** (hardware) + **(+/-)1%** (processamento), considerada aceitável para monitoramento residencial.
3. **Disponibilidade:** **99.9%** em condições normais de rede, garantida pelas rotinas de reconexão no firmware.

### Resultados Esperados (PoC)

O projeto entregou o Protótipo de Medição Precisa e a plataforma de gerenciamento, cumprindo a Prova de Conceito (PoC) ao verificar o correto registro de consumo de água pela interface web .

## 6. Análise e Gerenciamento de Riscos

Os pontos críticos identificados e os planos de mitigação são cruciais para a gestão do projeto.

Risco Identificado	Probabilidad e	Impact o	Ação Corretiva/Mitigação
Falha na conexão Wi-Fi	Média	Alta	Implementar reconexão automática e armazenar temporariamente dados no buffer.
Ruído ou erro de leitura no sensor de vazão	Média	Média	Calibrar o sensor e filtrar leituras com médias móveis para suavizar ruídos.
Falha de energia elétrica	Baixa	Média	Utilizar fonte estável de 5V e considerar backup com bateria.

Problemas de configuração do Firebase (ou MongoDB)	Média	Média	Testar com dados locais antes da integração e validar credenciais da API.
Atraso no desenvolvimento da interface web	Média	Baixo	Dividir tarefas e priorizar backend e comunicação antes do design da interface.

## 7. Cronograma

O desenvolvimento foi dividido em fases sequenciais ao longo de 4 semanas.

Etapa	Semana 1	Semana 2	Semana 3	Semana 4
Pesquisa e aquisição de componentes	X			
Desenvolvimento do hardware	X	X		
Programação de firmware		X		
Criação da plataforma web			X	
Testes e validação				X

## **8. Conclusões do Projeto**

O projeto do **Sistema de Monitoramento de Consumo de Água em Tempo Real** demonstrou ser uma solução técnica viável e eficaz para o acompanhamento hídrico em ambientes residenciais e industriais.