



SISTEMA PARA MONITORAMENTO DE PARÂMETROS FÍSICOS E QUÍMICOS DE CORPOS D'ÁGUA

Cristopher Vidal Machado
UDC
Foz do Iguaçu, Brasil
cristophervidal3@gmail.com

Felipe da Maia Bueno
UDC
Foz do Iguaçu, Brasil
felipe_mbsti@hotmail.com

Natan Rodrigues Rauber
UDC
Foz do Iguaçu, Brasil
rauber@protonmail.com

Luciano Santos Cardoso
UDC
Foz do Iguaçu, Brasil
luciano.cardoso@udc.edu.br

Abstract — The growing use of IoT concepts in technology has been impacting the execution of several activities that, previously, could only be carried out by specific professionals in certain areas. This work demonstrates the process of studying, designing and developing a hardware and software system capable of collecting and sending data related to physical and chemical parameters of water bodies via the internet. The system aims to facilitate the collection of this type of data, to simplify and streamline the process in relation to a laboratory analysis. Studies in the area of water quality monitoring were systematically analyzed and are the basis for the study and choice of parameters to be monitored. Several relevant works in the areas of automation and electronic monitoring were analyzed, being added to the technical basis for choosing the equipment and sensors that were used. The developed system has a dynamic architecture that allows its adaptation, and for this work it was built in a way that allows to collect, through sensors, data of the parameters of temperature, turbidity, pH and solids dissolved in water bodies. The data collected by the sensors are sent to a system that stores and makes these data available on a public website, so anyone interested can consult them later at any time.

Keywords — *Internet of Things; IoT; Monitoring; System;*

Resumo — A crescente utilização dos conceitos de IoT na tecnologia vem impactando na execução de diversas atividades que, anteriormente, poderiam ser realizadas apenas por profissionais específicos de determinadas áreas. Este trabalho demonstra o processo de estudo, concepção e desenvolvimento de um sistema de hardware e software capaz de realizar a coleta e envio de dados referentes a parâmetros físicos e químicos de corpos d'água via internet. O sistema tem por objetivo facilitar a coleta deste tipo de dado, para simplificar e agilizar o processo em relação a uma análise laboratorial. Estudos na área de monitoramento da qualidade da água foram analisados de maneira sistemática e são base do estudo e escolha dos parâmetros a serem monitorados. Vários trabalhos relevantes nas áreas de automação e monitoramento eletrônico foram analisados, sendo agregados ao embasamento técnico para a escolha dos equipamentos e sensores que foram utilizados. O sistema desenvolvido possui arquitetura dinâmica que permite sua adaptação, e para este trabalho foi construído de forma que permite coletar, por meio de sensores, dados dos parâmetros de temperatura, turbidez, pH e sólidos dissolvidos em corpos d'água. Os dados coletados pelos sensores são

enviados a um sistema que armazena e disponibiliza estes dados em um site público, para que qualquer pessoa interessada possa consultá-los posteriormente em qualquer momento.

Palavras-chave — *Internet das Coisas; IoT; Monitoramento; Sistema.*

I. INTRODUÇÃO

O conceito de *Internet of Things* (Internet das Coisas – IoT) foi criado em 1999 por Kevin Ashton e, segundo o mesmo autor em [1], refere-se a uma revolução tecnológica que tem como objetivo conectar coisas ou equipamentos à rede mundial de computadores. Desde então, diversos outros autores criaram definições complementares para o termo, para Tanwar, Tyagi e Kumar em [2] pode ser explicada, de maneira resumida, como um grande número de sensores conectados a uma série de dispositivos, desde pessoais até industriais, que estão conectados à internet. A utilização de IoT em sistemas de monitoramento pode reduzir muito o tempo necessário para a coleta e análise de dados, além da possibilidade de automação de todo o processo. Segundo Silva Júnior em [3], para estimativa da qualidade da água, geralmente, são utilizadas análises laboratoriais que demandam uma quantidade considerável de tempo, além do tempo necessário para o deslocamento até o local de coleta das amostras analisadas. Este estudo possibilitou o desenvolvimento de um sistema para coleta e transmissão de dados referentes a parâmetros físicos e químicos de corpos d'água de forma mais simples e rápida, o mesmo é composto por um protótipo de coleta de dados que faz a leitura de sensores e envia os dados coletados pela internet, uma aplicação recebe e disponibiliza os dados coletados para consulta por meio da internet. O sistema pode ser adaptado para coletar dados referentes a outros parâmetros, dependendo da necessidade.

II. REFERENCIAL TEÓRICO

O monitoramento de parâmetros físicos e químicos de corpos d'água através de sensoriamento é um assunto multidisciplinar, pois envolve um embasamento nas áreas de sensoriamento, automação e redes de computadores, além da necessidade de estudar os parâmetros físicos e químicos que serão monitorados em corpos d'água. As áreas de estudo abordadas neste trabalho estão detalhadas a seguir.



A. Indicadores de Qualidade em Corpos D'água

A aferição da qualidade da água é realizada a partir de estudos de determinadas variáveis físicas, químicas e biológicas. O Ministério da Saúde Brasileiro dispõe algumas portarias sobre os procedimentos de controle e de vigilância da qualidade da água para consumo humano e seu padrão de potabilidade [4]. Segundo Araújo e Oliveira em [5], os parâmetros físicos temperatura, turbidez e condutividade em conjunto com o indicador químico potencial hidrogeniônico (pH), são suficientes para identificar problemas na qualidade da água. A resolução do CONAMA (Conselho Nacional do Meio Ambiente) 430/2011 descreve as características desejáveis destes parâmetros de qualidade para corpos d'água, que variam dependendo da atividade para a qual o corpo d'água será utilizado [6]. O monitoramento de parâmetros físicos e químicos de corpos d'água pode ser feito através de sensores, desenvolvidos individualmente para fazer a leitura de parâmetros específicos.

1) *Temperatura*: A temperatura é uma medida de intensidade de calor, que influencia diretamente na qualidade da água, pois um aumento da temperatura pode alterar a taxa de reações químicas e biológicas. Aumentos de temperatura também podem indicar diminuição de solubilidade de gases ou aumento da taxa de gases liberados [7].

2) *Turbidez*: Segundo Sperling em [7], a turbidez na água pode ser de origem natural e não necessariamente apresenta inconvenientes sanitários, porém uma grande quantidade de sólidos suspensos pode servir como abrigo para microrganismos patogênicos. Quando a origem dos sólidos não for natural, pode representar a presença de elementos tóxicos. A turbidez representa a quantidade de sólidos suspensos em uma solução líquida, e sua unidade de medida é chamada Unidade Nefelométrica de Turbidez (UNT).

3) *Potencial Hidrogeniônico*: Segundo Lima et al. em [8] o pH é uma escala logarítmica que indica a acidez ou alcalinidade de uma solução aquosa. Sperling indica em [7] que medições de pH baixo ou pH alto podem causar irritação na pele e olhos de seres humanos e, em termos de corpos de água, condições de pH baixos podem indicar presença de efluentes industriais, enquanto um pH alto pode representar presença de algas na vida aquática. Para soluções aquosas com temperatura de 25°C, as soluções com $\text{pH} < 7$ dizem-se ácidas e soluções com $\text{pH} > 7$ dizem-se alcalinas ou básicas, e as soluções neutras, a 25°C, têm $\text{pH} = 7,0$ [9].

4) *Condutividade Elétrica*: A condutividade elétrica indica a capacidade natural da água de transmitir a corrente elétrica em função da presença de substâncias dissolvidas, que se dissociam em ânions e cátions. Em termos práticos as soluções nas quais se faz presente a maioria dos compostos inorgânicos apresentam condutividade elevada e, em contrapartida, compostos orgânicos que não se dissociam em soluções aquosas, quando presentes reduzem a transmissão da corrente elétrica. Águas naturais apresentam usualmente condutividade elétrica inferior a $100\mu\text{S}/\text{cm}$ (microsiemens por centímetro), podendo atingir $1000\mu\text{S}/\text{cm}$ em corpos d'água receptores de elevadas cargas de efluentes domésticos e industriais [10].

B. IoT em Atividades de Monitoramento e Sensoriamento

Segundo Ashton et al. em [1], IoT refere-se a uma revolução tecnológica que tem como objetivo conectar coisas ou equipamentos à rede mundial de computadores. Kevin Ashton foi quem propôs o termo IoT, em 1999. Em anos posteriores diversos autores propuseram suas próprias definições para o termo. Para Tanwar, Tyagi e Kumar em [2], IoT pode ser explicada, de maneira resumida, como um grande número de sensores conectados a uma série de dispositivos, desde pessoais até industriais, que estão conectados à internet. Muitos pesquisadores iniciaram seus estudos e publicaram uma série de resultados de seus trabalhos em monitoramento de parâmetros utilizando IoT, em áreas como domótica, saúde, agricultura, cidades inteligentes e muitas outras. Através destes trabalhos é possível identificar os principais conceitos que devem ser utilizados para desenvolver soluções em IoT. O trabalho de Shete e Agrawal [11] descreve que soluções de IoT em monitoramento geralmente tendem a se tornar complexas, de acordo com a quantidade de tecnologias integradas envolvidas, porém podem promover um impacto positivo tecnologicamente e socialmente nos locais onde implantadas.

C. Conceitos de Transmissão de Dados

Segundo Silva Júnior em [3], um dos pontos mais importantes de uma arquitetura de hardware de monitoramento é a comunicação entre os dispositivos e, cada componente, precisa ser compatível com o protocolo dos dados que transitam nos barramentos, para que estes dados possam ser distribuídos de forma rápida e otimizada.

Alguns protocolos de comunicação são amplamente utilizados nos sensores comerciais como o *Inter-Integrated Circuit* (Circuito Inter-integrado – I2C), o *Serial Peripheral Interface* (Interface Periférica Serial – SPI) e o *Universal Asynchronous Receiver/Transmitter* (Receptor/Transmissor Universal Assíncrono – UART). Cada protocolo tem características peculiares que apresentam vantagens e desvantagens em sua utilização. O protocolo UART foi criado por Gordon Bell, pela empresa DEC entre 1960 e 1965. Bell desenvolveu um módulo de recebimento e transmissão de dados que basicamente convertia os dados a serem transmitidos de um barramento paralelo para a forma serial, e fazia o inverso com os dados seriais recebidos, convertendo-os para a forma paralela. Foi amplamente adotada em projetos eletrônicos desde sua criação [12]. O protocolo I2C, desenvolvido pela empresa Philips em 1982, tem como principal objetivo simplificar a comunicação entre uma unidade de controle e diversos dispositivos de baixa complexidade pertencentes ao mesmo circuito. O I2C tem como ponto forte a baixa complexidade de hardware, aliado a um elegante conjunto de funcionalidades embutidas no próprio protocolo como: arbitragem, endereçamento do dispositivo escravo, confirmação de recebimento e detecção de colisões [13]. O SPI é um protocolo que permite a comunicação do microcontrolador com diversos outros componentes eletrônicos, formando uma rede. É uma especificação de interface de comunicação em série, síncrona, utilizada para comunicação de curta distância, principalmente em sistemas embarcados. A interface foi desenvolvida pela Motorola e se tornou um padrão amplamente aceito no mercado. Os dispositivos SPI se comunicam utilizando modo "full duplex", usando uma arquitetura "mestre-escravo" com um único mestre. O

dispositivo mestre origina a trama para a leitura e a escrita. Múltiplos dispositivos escravos são suportados e são a vantagem do uso deste protocolo [14].

O *Transmission Control Protocol* (Protocolo de Controle de Transmissão – TCP) foi projetado para oferecer um fluxo de bytes ponto a ponto confiável em uma inter-rede não confiável. Uma inter-rede é uma rede composta por diversas partes, que podem ter topologias, larguras de banda, retardos, tamanhos de pacote e outros parâmetros completamente diferentes. O TCP foi projetado para se adaptar dinamicamente às propriedades da inter-rede e ser robusto diante dos vários tipos de falhas que podem ocorrer [15]. Segundo o mesmo autor, cada máquina compatível com o protocolo TCP tem uma entidade de transporte TCP que gerencia fluxos e interfaces TCP para a camada do *Internet Protocol* (Protocolo da Internet – IP). Uma entidade TCP aceita fluxos de dados do usuário provenientes de processos locais, divide-os em partes e envia cada parte em um datagrama (entidade de dados completa e independente que contém informações suficientes para ser roteada desde origem até o destino) IP distinto. Quando os datagramas IP que contêm dados TCP chegam a uma máquina, eles são enviados à entidade TCP, que restaura os fluxos de bytes originais [15].

O *Hypertext Transfer Protocol* (HTTP) é um protocolo a nível de aplicação para sistemas distribuídos, colaborativos e envio de dados multimídia, que funciona sobre o protocolo TCP. O HTTP é utilizado pela "*World-Wide Web Global Information Initiative*" desde 1990. Sua primeira versão, definida como HTTP/0.9, era um protocolo muito simples para transferência de dados brutos via internet. Já em sua versão HTTP/1.0, o protocolo melhorou consideravelmente, permitindo que mensagens fossem enviadas em formato MIME-like, contendo metainformação sobre os dados transferidos, modificadores na semântica da requisição/resposta, cada metainformação é separada por uma sequência de escape *Carriage Return, Line Feed* (CR LF), estas mensagens estão divididas em Cabeçalhos (*Headers*) e Corpo (*Body*). Cada cabeçalho de requisição precisa informar o método a ser executado no servidor [16].

III. MATERIAIS E MÉTODOS

A. Metodologia

Esta pesquisa foi realizada e aplicada na cidade de Foz do Iguaçu, situada no extremo oeste do estado do Paraná. Foi feito um estudo de caso sobre monitoramento de parâmetros de corpos d'água variados. Do ponto de vista do problema abordado, este é um trabalho de coleta de dados, e o monitoramento foi feito, em um primeiro momento, num ambiente totalmente controlado. Os dados de monitoramento coletados referentes a parâmetros físicos e químicos de corpos d'água foram utilizados para a calibração dos sensores, testes do sistema e povoamento da base de dados.

Segundo Souza e Diesel em [17] uma pesquisa experimental implica a formulação do problema de pesquisa, a revisão bibliográfica, a formulação de hipóteses, a definição do método, a realização do experimento e coleta de dados, a análise e a interpretação dos resultados e a elaboração do trabalho final. Neste trabalho, uma série de procedimentos técnicos, baseados na bibliografia do tema, foram utilizados

para coletar os dados. Para isso, foram utilizados sensores digitais previamente calibrados.

B. Recursos de Hardware

1) *Plataforma de Prototipagem NodeMCU v3*: O NodeMCU v3 é uma plataforma de hardware e software voltada para prototipagem. O NodeMCU v3 Lolin Wi-Fi ESP8266 ESP-12E é composto de um firmware que é executado em um ESP8266 modelo ESP-12E, que fornece suporte ao padrão IEEE 802.11 b/g/n e a toda pilha do protocolo TCP/IP em conjunto com uma solução Wi-Fi de rede completa. O NodeMCU pode ser programado em C++ e LUA, utilizando uma entrada micro USB para programação e alimentação, contém uma memória flash interna para armazenamento dos programas e pode ser integrado a sensores externos e outros dispositivos específicos por meio de portas GPIO mostradas em [18, Fig. 1].

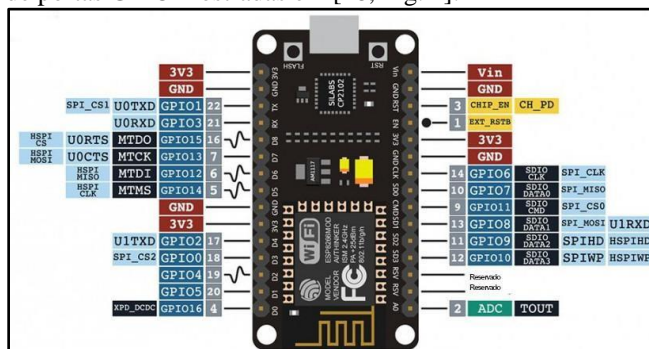


Fig. 1. Plataforma de prototipagem NodeMCU [18]

2) *Sensor de Turbidez*: O sensor de turbidez ou turbidímetro é o equipamento utilizado para medir a turbidez de um líquido. A aferição compara o espalhamento (ou desvio na trajetória) de um feixe de luz ao passar pela amostra, com o de um feixe de igual intensidade, ao passar por uma suspensão padrão [19]. Para este trabalho não foi possível a obtenção um sensor de turbidez, por este motivo os dados apresentados posteriormente referentes a este parâmetro são simulados, ou seja, gerados pelo sistema de forma aleatória, em uma faixa de valores predefinida.

3) *Sensor de Condutividade Elétrica*: O sensor de condutividade elétrica mede a capacidade de uma substância de conduzir corrente elétrica. A maioria dos equipamentos de medida de condutividade elétrica se baseiam na medida de tensão, aplicando uma corrente alternada em dois eletrodos de uma célula de condutividade elétrica imersa em uma solução [20]. O modelo do sensor de condutividade elétrica utilizado neste trabalho é o KS0429 da Keystudio, um sensor *plug and play* compatível com a plataforma ESP utilizada, que através de seu eletrodo permite medir materiais condutores, como sólidos suspensos, metais pesados e íons condutores na amostra de água. A medição feita por este sensor apresenta valores em ppm (partes por milhão) que equivalem, cada uma, à $1.56\mu\text{S}/\text{cm}$. A leitura é feita na faixa de 0 a 1000ppm (0 a $1560\mu\text{S}/\text{cm}$).

4) *Sensor de pH*: Segundo Ahmed et al. em [21], o sensor de pH tem a capacidade de inferir medidas na quantidade de prótons, que possibilita indicar a neutralidade, acidez ou mesmo a alcalinidade de uma solução líquida. O

modelo do sensor de pH utilizado neste trabalho é o PH4502C, composto por dois eletrodos, um de referência (prata/cloreto de prata) e um eletrodo de vidro sensível a íons de hidrogênio. O módulo PH4502C tem a finalidade de condicionar o sinal que vem do sensor para uso com Arduino, ESP e demais controladores. A leitura é feita na faixa de 0 a 14pH e na faixa de temperatura de 0 a 80°C.

5) *Sensor de Temperatura*: Para inferir medidas de temperatura, será utilizado o sensor DS18B20. Esse sensor fornece a temperatura em graus Celsius com resolução de 9 a 12 bits, operam numa faixa de -55°C a +125°C, com uma precisão de 0,5°C na faixa -10°C a 85°C [22].

C. Recursos de Software

1) *Ambiente de Desenvolvimento Integrado Arduino*: O Ambiente de Desenvolvimento Integrado Arduino (IDE Arduino) é um editor de código fonte capaz de compilar e carregar programas, que são desenvolvidos na linguagem de programação C++. Permite acesso rápido ao compilador e envio dos binários ao microcontrolador e um acesso ao terminal de entrada e saída dos dispositivos a serem programados. Este software foi utilizado no trabalho para escrever código fonte, compilar e enviar os binários ao dispositivo de controle NodeMCU.

2) *Linguagem de Programação Ruby*: Ruby é uma linguagem de programação criada em 1995 pelo japonês Yuri Matsumoto. É uma linguagem limpa, direta, orientada a objetos e relativamente simples de se aprender e trabalhar, com muitas semelhanças ao Perl, SmallTalk e Python. Possui suporte a diversos tipos de sistemas operacionais (multiplataforma) como Linux, Windows, Solaris e outros. Possui muitas funcionalidades interessantes como o Ruby Gems (biblioteca Gratuita disponível na internet), Code Blocks (blocos de código pronto), Mixins (reposta à herança múltipla), tipagem dinâmica e outras características [17]. Foi utilizada neste trabalho em conjunto com o Ruby on Rails, um framework que torna mais fácil desenvolver, disponibilizar e manter aplicações web.

3) *Heroku*: É uma plataforma que permite ao usuário implantar, executar e gerenciar aplicações online com suporte à diversas linguagens de programação. A instalação da aplicação, bem como de qualquer alteração da aplicação na plataforma Heroku pode ser feita via integração com GitHub. Um repositório de código-fonte da aplicação no GitHub integrado com o Heroku permite que sejam feitos *commits* (upload de alterações no código-fonte da aplicação) no repositório, para que alterações feitas no código-fonte surtam efeito na aplicação. O Heroku permite a utilização de *add-ons*, que são serviços adicionais disponibilizados por terceiros ou pelo próprio Heroku. Por meio destes *add-ons* é possível utilizar serviços de bancos de dados, incluindo Heroku Postgres e o JawsDB MySQL, que são os serviços de bancos de dados PostgreSQL e MySQL respectivamente [23].

4) *Git*: O sistema de controle de versão moderno mais usado no mundo hoje é o Git, um projeto de código aberto maduro e com manutenção ativa desenvolvido em 2005 por Linus Torvalds, o famoso criador do kernel do sistema operacional Linux. Um número impressionante de projetos

de software depende do Git para controle de versão, incluindo projetos comerciais e de código-fonte aberto. Os desenvolvedores que trabalharam com o Git estão bem representados no pool de talentos de desenvolvimento de software disponíveis e funcionam bem em uma ampla variedade de sistemas operacionais e Ambientes de Desenvolvimento Integrado (IDE) [24].

IV. LEVANTAMENTO E ANÁLISE DE REQUISITOS

Para definir os elementos que compõem o sistema e sua estrutura é necessário definir quais serão suas funcionalidades, nesta seção são detalhados os requisitos para o software desenvolvido. De forma simplificada, a etapa de montagem do protótipo se baseia na conexão dos sensores à plataforma NodeMCU e, quando necessário, à uma fonte de energia dedicada ao sensor.

A. Requisitos Funcionais

Os requisitos funcionais descrevem o que o sistema deve fazer. Esses requisitos dependem do tipo de software que está sendo desenvolvido, dos usuários a quem o software se destina e da abordagem geral considerada pela organização ao redigir os requisitos. Eles descrevem as funções do sistema detalhadamente [25]. Abaixo estão presentes os requisitos funcionais levantados para o sistema proposto.

- RF001: O sistema deve ter a capacidade de coletar dados de sensores;
- RF002: O sistema deve possuir a capacidade de enviar, via internet, dados de monitoramento, provenientes de sensores;
- RF003: O sistema deve permitir cadastrar os dados de monitoramento coletados por sensores;
- RF004: O sistema deve permitir manter (criar, alterar, consultar, excluir) os locais de monitoramento, compostas por nome do local, cidade, estado, latitude e longitude.

B. Requisitos Não Funcionais

Os requisitos não funcionais são aqueles não diretamente relacionados às funções específicas fornecidas pelo sistema. Eles podem estar relacionados às propriedades emergentes do sistema, como confiabilidade, tempo de resposta e espaço de armazenamento [25]. Abaixo estão presentes os requisitos não funcionais levantados para o sistema proposto:

- RN001: Os dispositivos que utilizam o sistema devem possuir conexão com a internet;
- RN002: O sistema deve coletar dados dos sensores instalados;
- RN003: O sistema deve armazenar dados de sensores em um banco de dados;
- RN004: O sistema deve trafegar os dados ao servidor utilizando o protocolo HTTP com o conteúdo em formato de dados JSON;
- RN005: O sistema deve possuir uma interface web para consulta dos dados de monitoramento.

C. Diagrama de Casos de Uso

O diagrama de casos de uso é o diagrama mais geral e informal da *Unified Modeling Language* (Linguagem de Modelagem Unificada – UML), utilizado normalmente nas fases de levantamento e análise de requisitos do sistema,

embora venha a ser consultado durante todo o processo de modelagem e possa servir de base para outros diagramas. Apresenta uma linguagem simples e de fácil compreensão para que os usuários possam ter uma ideia geral de como o sistema irá se comportar. Procura identificar os atores (usuários, outros sistemas ou até mesmo algum hardware especial) que utilizarão de alguma forma o software, bem como os serviços, ou seja, as funcionalidades que o sistema disponibilizará aos atores, conhecidas nesse diagrama como casos de uso [26]. Um caso de uso tem como objetivo descrever como será o uso de uma funcionalidade de um sistema, permitindo que tanto o usuário quanto os programadores e outras pessoas envolvidas no desenvolvimento do sistema entendam a funcionalidade de uma maneira única, seguindo um padrão específico. Em [27, Fig. 2] é apresentado o diagrama de casos de uso modelado para o sistema proposto.

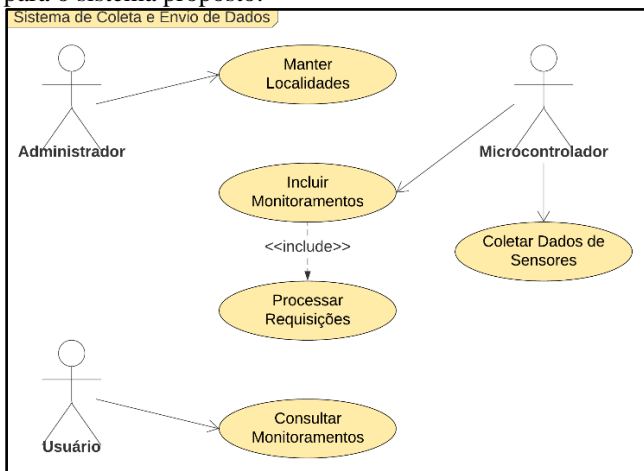


Fig. 2. Diagrama de casos de uso do sistema [27]

V. IMPLEMENTAÇÃO

A. Sistema de Coleta e Envio de Dados

Este sistema fica localizado fisicamente no microcontrolador e foi implementado em linguagem C++ utilizando o ambiente de desenvolvimento Arduino IDE e tem papel de agente de coleta de dados dos sensores, construção da mensagem, detecção de erros e envio dos dados coletados para o servidor, que recebe os dados em formato JSON, via API. A programação do microcontrolador ESP8266 foi executada utilizando a IDE Arduino. Após a instalação das bibliotecas de compatibilidade do microcontrolador ESP8266 e escrita do código, o sistema já estava pronto para enviar os dados. Todos os algoritmos implementados seguiram a mesma lógica: (1) efetuam uma leitura de todos os sensores; (2) capturam o tempo atual do microcontrolador; (3) enviam os dados para o servidor utilizando o protocolo HTTP; (4) capturam o tempo atual do microcontrolador novamente; (5) calculam a diferença entre o tempo inicial e o tempo final da execução; (6) enviam o valor de tempo gasto para o servidor. Os sensores foram conectados a portas digitais e a um conversor analógico digital, para as quais foram enviados sinais, posteriormente convertidos em dados. O conversor analógico digital utilizado é o ADS1115, com resolução de 16 bits e que utiliza

o protocolo de comunicação I2C, que permite maior precisão da leitura analógica, e foi necessário neste projeto pois o microcontrolador ESP82266 possui apenas uma porta analógica. A calibragem dos sensores foi executada utilizando uma solução padrão de calibragem para o sensor de pH, que acompanha o produto enviado pelo fabricante. A temperatura do líquido no momento da calibragem era de 25°C (±0.5°C) (valor sugerido pelos fabricantes dos sensores), e o mesmo fornecia valores de condutividade elétrica de 50µS (±20) e pH 7 (±0.5).

B. Sistema de Monitoramento

O sistema foi desenvolvido utilizando a linguagem de programação Ruby, e possui uma camada de interface web que será acessada pelos usuários, e uma camada de aplicação que recebe os dados de monitoramentos enviados pelo microcontrolador e os inclui no banco de dados. Para o desenvolvimento foi utilizado um serviço da plataforma Heroku para aplicações em Ruby, que armazena e disponibiliza o sistema criado. As alterações no projeto foram feitas utilizando o Visual Studio Code e atualizadas no Heroku utilizando o Git, que modifica o estado do servidor ao efetuar *commit*. Para comunicação com o software presente no Heroku com o agente microcontrolador, foi utilizado o protocolo HTTP em conjunto com a arquitetura Rest. O interpretador Ruby versão 2.5.0 em conjunto com as bibliotecas padrão da linguagem são a principal tecnologia utilizada na implementação da aplicação. As seguintes bibliotecas foram utilizadas na implementação do projeto:

- rails – Biblioteca de desenvolvimento web e base do framework de mesmo nome;
- mysql2 – Biblioteca de conexão com bancos de dados;
- puma – Biblioteca que disponibiliza um conjunto de funções de servidor de aplicação;
- sass-rails – Biblioteca para utilizar recursos avançados visuais da *Hyper Text Markup Language* (Linguagem de Marcação de Hipertexto – HTML);
- jbuilder - Biblioteca que facilita a utilização de JSON nos retornos das API's;
- rack-cors - Biblioteca que fornece elementos básicos de segurança.

O código fonte do servidor está presente em um repositório Git público¹, sendo necessário informar no arquivo de configuração os dados de banco de dados e inicializar o serviço Ruby através do comando *rails server* para inicializar o serviço. O banco de dados utilizado para manter os dados de monitoramento foi configurado utilizando a interface textual fornecida pelo MariaDB (sistema de gerenciamento de banco de dados), onde foram criadas todas as estruturas de acesso aos dados de monitoramento. A interface do usuário é feita via navegador web, utilizando o protocolo HTTP para carregar um conteúdo HTML, que permite acesso às funcionalidades do sistema, utilizando suas classes de controle. O código foi compilado utilizando a IDE Arduino e pode ser utilizado em microcontroladores da família ESP8266.

O esquema apresentado em [28, Fig. 3] representa a ligação de hardware dos componentes utilizados no circuito

¹ <https://github.com/CristopherVidalMachado/monitoramento-udc>

do agente coletor dos dados de sensores. Esta modelagem foi feita utilizando a plataforma Fritzing, uma iniciativa de código aberto para design de hardware eletrônico desenvolvida na University of Applied Sciences Potsdam.

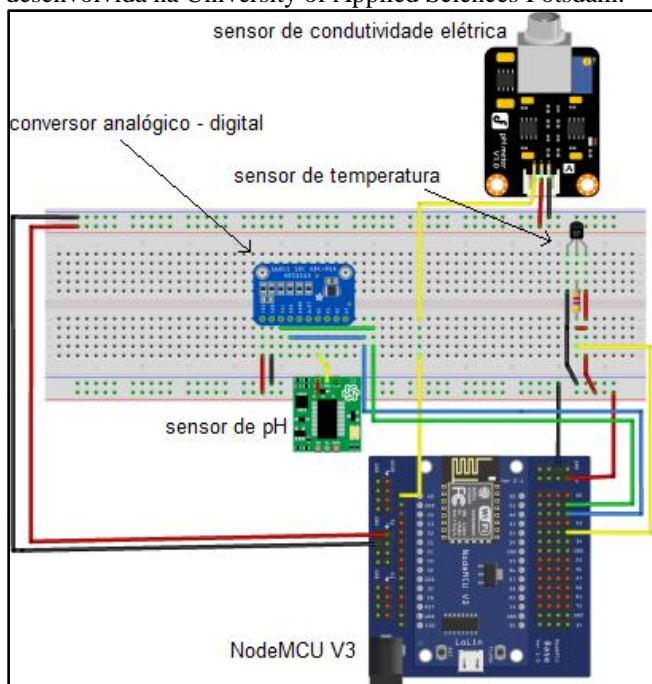


Fig. 3. Diagrama de hardware do sistema de coleta e envio de dados [28]

Em [29, Fig. 4] é mostrada a implementação física do agente coletor de dados, que utiliza sondas a prova de água onde estão presentes os sensores descritos anteriormente na implementação.

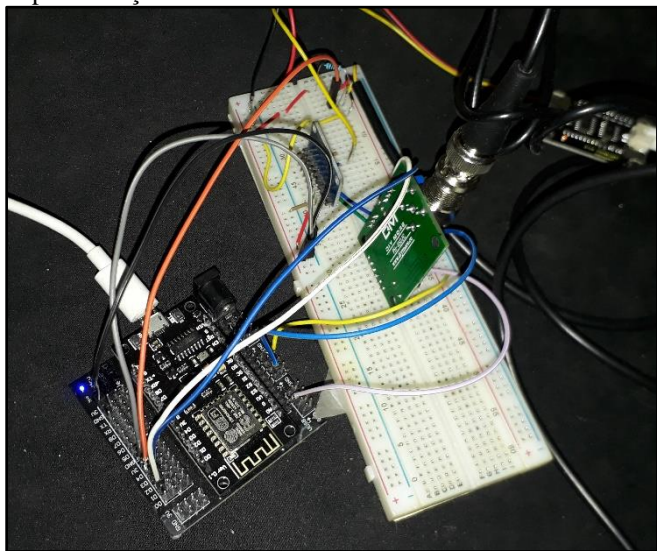


Fig. 4. Implementação do agente de coleta e envio de dados [29]

VI. ANÁLISE DE RESULTADOS

Após a calibragem do sistema de monitoramento proposto ser finalizada, foram feitos testes consecutivos no sistema utilizando provetas de vidro com corpos de água e líquidos de calibragem. Os parâmetros de temperatura, pH e condutividade elétrica foram amostrados utilizando os sensores do sistema e comparados com a leitura de outros equipamentos de aferição dos respectivos parâmetros durante

o período de monitoramento para verificar a confiabilidade dos dados presentes no sistema. Foram coletados diversos dados, provenientes dos sensores, em 4 situações diferentes:

- Água do fornecimento público;
- Solução de calibragem de condutividade;
- Solução de calibragem de pH igual 4;
- Solução de calibragem de pH igual 10.

A princípio foram efetuadas amostras utilizando a temperatura de 25 °C, indicada pelo fabricante de soluções de calibragem como referência. A amostra inicialmente utilizada foi coletada a partir do serviço público de fornecimento, armazenada utilizando uma proveta, e sua temperatura elevada até 25 °C. Através do sistema de monitoramento foram aferidas leituras de pH próximos aos valores de 6.8 - 7.0, que condizem com a leitura do sensor no instante da amostra. Através de um teste de fita de pH, que consiste na imersão de uma fita com componentes químicos que reagem ao líquido no qual é introduzida, fornecendo cor resultante que representa uma leitura de pH como apresentado em [30, Fig. 5], foi possível observar valores próximos de pH 7 na proveta em questão.



Fig. 5. Escala de medição da fita de teste de pH [30]

Outras amostras de água foram utilizadas para aferir os parâmetros adquiridos através dos sensores, que fornecem ambos a temperatura de 25 °C, valor de pH 4 e 10 respectivamente. Assim sendo possível identificar no sistema a alteração de um pH próximo a neutro para um pH baixo entre 3,9 - 4,1. O mesmo procedimento foi repetido com a solução de calibragem de pH 10, que forneceu leituras no sistema de aproximadamente 9,9-10,0. As fitas mostradas em [31, Fig. 6], representam os testes para pH baixo, alto e neutro coletados em conjunto com os sensores do sistema de coleta.

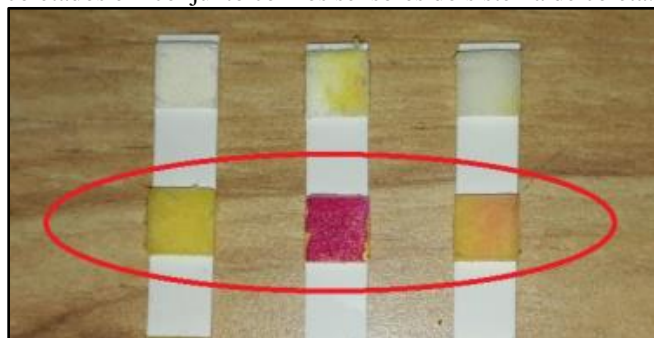


Fig. 6. Fitas de teste de pH [31]

Para a leitura e checagem dos dados de temperatura, foi necessário a utilização de um termômetro analógico, para conferência do valor lido na plataforma. O termômetro possui uma leitura simplificada de temperatura, com erro de até 1 °C. Foram feitos monitoramentos utilizando o termômetro inserido diretamente na proveta em conjunto com os outros sensores, como detalhado em [32, Fig. 7].



Fig. 7. Leitura do sensor de temperatura [32]

Os dados do sensor de condutividade elétrica foram analisados utilizando uma solução para calibragem que possui valores pré definidos. As análises foram feitas com o líquido a 25 °C, fornecendo leituras com taxa de erro de $\pm 10 \mu\text{S}/\text{cm}$.

Foram realizadas 40 coletas através do sistema, das quais 38 foram recebidas pelo sistema de consulta, uma taxa de 95% de coletas bem sucedidas. Os dados foram disponibilizados para consulta através da interface web após serem coletados. Através da análise dos logs, foi possível detectar comportamento adequado do sistema em relação a erros ou falhas. A coleta feita durante os testes do sistema é mostrada em [33, Fig. 8].

Monitoramentos					
°C	pH	NTU	$\mu\text{S}/\text{cm}$	Localidade	Data
25.3	7.5	50.0	12.0	Localidade teste	2020-11-24 16:40:08 -0300
25.3	7.5	50.0	12.0	Localidade teste	2020-11-24 16:41:08 -0300
25.0	6.7	51.0	56.0	Localidade teste	2020-11-24 16:42:09 -0300
25.2	6.8	52.0	57.9	Localidade teste	2020-11-24 16:43:09 -0300
25.4	6.9	52.0	58.0	Localidade teste	2020-11-24 16:44:10 -0300
25.3	6.7	52.0	53.0	Localidade teste	2020-11-24 16:45:10 -0300
25.3	6.9	50.0	56.0	Localidade teste	2020-11-24 16:46:11 -0300
25.7	6.8	52.0	58.0	Localidade teste	2020-11-24 16:47:11 -0300
25.7	6.7	50.0	54.0	Localidade teste	2020-11-24 16:48:12 -0300
25.8	6.9	50.0	56.0	Localidade teste	2020-11-24 16:49:12 -0300
26.0	7.0	51.0	58.0	Localidade teste	2020-11-24 16:50:13 -0300
25.8	4.0	51.0	39.0	Localidade teste	2020-11-24 16:51:13 -0300
25.6	4.0	50.0	41.0	Localidade teste	2020-11-24 16:52:14 -0300
25.1	4.0	51.0	38.0	Localidade teste	2020-11-24 16:53:15 -0300
26.6	4.0	50.0	34.0	Localidade teste	2020-11-24 16:54:15 -0300
25.5	4.0	50.0	39.0	Localidade teste	2020-11-24 16:55:16 -0300
25.4	4.0	52.0	40.0	Localidade teste	2020-11-24 16:56:16 -0300
25.6	4.0	50.0	35.0	Localidade teste	2020-11-24 16:57:17 -0300
25.4	4.0	50.0	34.0	Localidade teste	2020-11-24 16:58:17 -0300
25.7	4.1	50.0	36.0	Localidade teste	2020-11-24 16:59:18 -0300
25.3	4.0	51.0	37.0	Localidade teste	2020-11-24 17:00:18 -0300
25.5	4.0	52.0	36.0	Localidade teste	2020-11-24 17:01:19 -0300
25.2	10.0	50.0	56.0	Localidade teste	2020-11-24 17:02:19 -0300
25.1	9.9	50.0	54.0	Localidade teste	2020-11-24 17:03:20 -0300
25.0	10.0	52.0	36.0	Localidade teste	2020-11-24 17:04:20 -0300
24.9	9.9	52.0	37.0	Localidade teste	2020-11-24 17:05:21 -0300
25.0	10.0	50.0	34.0	Localidade teste	2020-11-24 17:06:21 -0300
25.3	10.0	50.0	35.0	Localidade teste	2020-11-24 17:07:22 -0300
25.2	10.0	52.0	38.0	Localidade teste	2020-11-24 17:08:23 -0300
25.3	9.9	51.0	41.0	Localidade teste	2020-11-24 17:09:23 -0300
25.0	9.9	51.0	40.0	Localidade teste	2020-11-24 17:10:24 -0300
26.0	10.0	51.0	57.0	Localidade teste	2020-11-24 17:11:24 -0300

Fig. 8. Monitoramentos feitos durante testes do sistema [33]

VII. CONCLUSÃO

O objetivo final proposto para este trabalho foi o desenvolvimento de um protótipo que coleta e, por meio da internet, envia dados referentes a parâmetros de qualidade da água. Os parâmetros utilizados para demonstrar esse conceito foram escolhidos com base no estudo feito sobre monitoramento de qualidade da água, sendo estes temperatura, turbidez, condutividade elétrica e pH. Os dados coletados pelos sensores são enviados a um sistema que armazena e disponibiliza estes dados em um site público, para que qualquer pessoa interessada possa consultá-los.

Os resultados obtidos durante os testes do sistema comprovaram que ao menos 95% das coletas e envio de dados foram bem sucedidas, tendo 38 leituras cadastradas corretamente no sistema, em um período de 40 minutos de monitoramento, com 1 coleta por minuto. A precisão das aferições feitas pelos sensores foi apresentada no capítulo de análise de resultados, comparando a leitura efetuada pelos sensores com outros métodos de aferição dos parâmetros monitorados, para validação dos dados apresentados no sistema.

O estudo e o protótipo desenvolvidos neste trabalho visaram facilitar futuros trabalhos de análise da qualidade da água, tornando o processo mais ágil se comparado a uma análise tradicional. Neste sentido, o sistema criado apresentou grande vantagem, visto que métodos tradicionais de aferição da qualidade da água exigem muito tempo para que sejam feitas leituras, referentes a um momento específico. O sistema criado permitiu que fossem feitas várias

leituras dos parâmetros monitorados ao longo do tempo, sendo possível analisar os dados de forma mais clara e precisa, além de permitir que as leituras fossem feitas remotamente.

As vantagens obtidas com o sistema, em relação a métodos tradicionais de aferição da qualidade da água foram:

- Coleta de dados automática, facilitando e reduzindo os custos da operação;
- Acesso remoto, permitindo maior comodidade na aquisição e análise dos dados;
- Fácil armazenamento de um grande número de leituras dos parâmetros monitorados, facilitando análises futuras;
- Fácil adaptação do sistema para monitoramento de outros parâmetros.

Trabalhos na área de monitoramento da qualidade da água foram analisados de maneira sistemática e serviram como base para o estudo e escolha dos parâmetros monitorados. Vários trabalhos relevantes nas áreas de IoT, automação e monitoramento eletrônico foram analisados, sendo agregados ao embasamento técnico para a escolha dos equipamentos e sensores que foram utilizados.

Uma das dificuldades encontradas no desenvolvimento do trabalho foi o levantamento inicial do material de estudo e identificação dos parâmetros a serem utilizados, pois a bibliografia deste tema é extensa e existem diversas fontes de consulta e linhas de pesquisa nesta área. Outras dificuldades foram encontradas na implementação do hardware do projeto, onde foi necessária a compra de sensores extras, devido a falhas e defeitos causados pelo transporte inadequado, erros na montagem e documentação e instruções de funcionamento incorretas.

O detalhamento feito ao longo do trabalho permite a replicação do sistema criado em trabalhos derivados e serve como base para adaptação do mesmo para monitoramentos variados. Os conceitos de IoT e monitoramento apresentados podem ser aprofundados, permitindo a construção de uma arquitetura de hardware e software mais rápida, segura e confiável. Os trabalhos futuros terão foco na melhoria da arquitetura de software da solução e melhoria da qualidade da interface de usuário, permitindo que a solução seja de fato de interesse para projetos de monitoramento de parâmetros referentes à qualidade da água ou outros, de acordo com as necessidades dos interessados.

REFERÊNCIAS

- [1] ASHTON, K. et al. That ‘internet of things’ thing. *RFID journal*, v. 22, n. 7, p. 97–114, 2009.
- [2] TANWAR, S.; TYAGI, S.; KUMAR, N. *Multimedia Big Data Computing for IoT Applications: Concepts, Paradigms and Solutions*. Springer, 2019. v. 163.
- [3] SILVA JÚNIOR, A. G. d. Sistema mecatrônico de tempo real para monitoramento da qualidade da água embarcado em veleiro robótico autônomo. Dissertação (Mestrado) — Universidade Federal do Rio Grande do Norte, 2015.
- [4] BRASIL, M. d. S. *Portaria nº2.914*, de 12 de dezembro de 2011. dispõe sobre os procedimentos de controle e de vigilância da qualidade da água para consumo humano e seu padrão de potabilidade. *Diário Oficial da União*, n. 12, 2011.
- [5] ARAÚJO, M. C. d.; OLIVEIRA, M. B. M. d. Monitoramento da qualidade das águas de um riacho da universidade federal de Pernambuco, Brasil. *Revista Ambiente & Água, SciELO Brasil*, v. 8, n. 3, p.247–257, 2013.
- [6] CONAMA, R. Dispõe sobre condições e padrões de lançamento de efluentes. *CONAMA 430/2011*, v. 14, p. 06–14, 2011.
- [7] SPERLING, M. V. *Introdução à Qualidade das Águas e ao Tratamento de Esgotos*. Minas Gerais: Editora UFMG, 1996. v. 1.
- [8] LIMA, E. L. et al. Módulo de sensores para monitoramento da qualidade da água com transmissão sem fio utilizando plataforma de prototipagem. Universidade Estadual de Feira de Santana, 2018.
- [9] CORRÊA, C. pH. *Revista de Ciência Elementar*, Casa das Ciências, v. 2, n. 2, 2014.
- [10] LIBÂNIO, M. *Fundamentos de Qualidade e Tratamento de Água*. Mato Grosso: Átomo, 2008.
- [11] SHETE, R.; AGRAWAL, S. IoT based urban climate monitoring using raspberry pi. In: *IEEE. 2016 International Conference on Communication and Signal Processing (ICCSP)*. 2016. p. 2008–2012.
- [12] BELL, C. G.; MUDGE, J. C.; MCNAMARA, J. E. *Computer Engineering: A DEC View of Hard-ware Systems Design*. Digital Press, 2014.
- [13] SEMICONDUCTORS, P. *The I2C-BUS Specification*. Philips Semiconductors, v. 9397, n. 750, p. 00954,2000.
- [14] LEENS, F. An introduction to I2C and SPI protocols. *IEEE Instrumentation & Measurement Magazine*, IEEE, v. 12, n. 1, p. 8–13, 2009.
- [15] TANENBAUM, A. S. *Redes de computadores*. Amsterdam: Pearson educación, 2003.
- [16] FIELDING, R. et al. Hypertext Transfer Protocol – http/1.1. RFC 2616, 1999.
- [17] SOUZA, R. S. d.; DIESEL, V. *Metodologia da pesquisa*. Brasil, 2008.
- [19] EPA. *Guidance Manual for Compliance with the Interim Enhanced Surface Water Treatment Rule: Turbidity Provisions*. 1999.

- [20] NETO, A. J. S. et al. Avaliação de um circuito eletrônico para medição da condutividade elétrica de soluções nutritivas. *Engenharia Agrícola*, SciELO Brasil, v. 25, n. 2, p. 427–435, 2005.
- [21] AHMED, U. et al. Efficient water quality prediction using supervised machine learning. *Water*, Multi-disciplinary Digital Publishing Institute, v. 11, n. 11, p. 2210, 2019.
- [22] MAXIM INTEGRATED. DS18B20 Datasheet. 2019. Disponível em: <<https://datasheets.maximintegrated.com/en/ds/DS18B20.pdf>>. Acesso em: 12 mar. 2020.
- [23] ROCCO, B. C. d. Integrando educação e planejamento previdenciários: uma ferramenta de simulação e aprendizagem na web. 2017.
- [24] ATlassian. O que é o Git: seja um profissional do Git com este guia. 2020. Disponível em: <<https://www.atlassian.com/br/git/tutorials/what-is-git>>.
- [25] SOMMERVILLE, I. *Engenharia de Software*. São Paulo, SP: Pearson Prentice Hall, 2011.
- [26] GUEDES, G. *UML 2 - Uma Abordagem Prática*. Novatec Editora, 2018. ISBN 9788575226445. Disponível em: <<https://books.google.com.br/books?id=mJxMDwAAQBAJ>>.