



INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
DE GOIÁS - IFG
CAMPUS FORMOSA
TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

DERECK BRIAN SOUSA DE ARAUJO

**DESENVOLVIMENTO DE UM SISTEMA AUTOMATIZADO PARA
LIMPEZA E MANUTENÇÃO DE PISCINAS**

DERECK BRIAN SOUSA DE ARAUJO

DESENVOLVIMENTO DE UM SISTEMA AUTOMATIZADO PARA LIMPEZA E
MANUTENÇÃO DE PISCINAS

Orientador: Prof. Me. Afrânio Furtado de
Oliveira Neto

*Este trabalho é dedicado ao menininho que,
quando pequeno, queria ser eu.*

AGRADECIMENTOS

Gostaria de expressar meus mais sinceros agradecimentos à minha mãe, que esteve ao meu lado ao longo de todo este projeto, oferecendo apoio incondicional e sendo uma verdadeira fonte de luz em minha vida. Agradeço também ao meu pai, que sempre acreditou em mim, oferecendo carinho, apoio e segurança. Essa formatura sempre foi nosso sonho, e agora, ao concluir este trabalho, percebo o quanto esse sonho se concretizou, trazendo-me uma enorme gratidão por sua confiança.

Não poderia deixar de agradecer ao meu irmão, que é uma das pessoas mais importantes e que mais amo em minha vida. Sua ajuda, carinho e apoio são o reflexo deste projeto. Sem ele, nada disso seria possível! Obrigado por me fazer querer ser melhor a cada dia.

Agradeço imensamente à minha namorada, Ana Clara, que tornou e continua tornando meus dias mais leves, possibilitando o desenvolvimento deste projeto e, mais do que isso, mudando completamente a minha vida. Seu carinho e amor me ajudam de maneiras inexplicáveis, tornando meus dias muito mais felizes e tranquilos.

Agradecer também ao meu orientador Afrânio que me guiou no desenrolar do projeto, me fornecendo todo o apoio necessário, ideias e conforto em momentos de ansiedade.

Por fim, estendo minha gratidão aos meus amigos e familiares, que sempre estiveram presentes ao longo dessa jornada. Faço um agradecimento especial à minha tia Regiane, que contribuiu imensamente para este projeto com sua revisão atenta e dicas valiosas de escrita. Aos meus colegas, que compartilharam comigo a ansiedade e os desafios desta etapa, deixo o meu muito obrigado. Foi uma linda jornada, e torço para que todos alcancem seus objetivos e realizem seus sonhos.

Muito Obrigado.

“Daqui a vinte anos, você vai se decepcionar mais com as coisas que não fez do que com as que fez. Então, solte as amarras, navegue para longe do porto seguro. Aproveite os ventos favoráveis em suas velas. Explore. Sonhe. Descubra.

(H. Jackson Brown Jr)

RESUMO

A crescente incorporação de tecnologias de automação em ambientes residenciais tem impulsionado o desenvolvimento de sistemas voltados à otimização de tarefas cotidianas. Nesse contexto, este trabalho apresenta o desenvolvimento de um sistema automatizado para limpeza e manutenção de piscinas residenciais, fundamentado nos princípios da automação residencial e da Internet das Coisas (IoT). O sistema integra sensores, atuadores e controladores microprocessados com a finalidade de monitorar e tratar a água de forma automatizada, reduzindo significativamente a necessidade de intervenção manual. A proposta busca ampliar a eficiência no uso de recursos, aumentar a segurança na manipulação de produtos químicos e promover práticas sustentáveis no consumo de água e energia. A metodologia adotada envolveu a revisão de normas técnicas, o estudo dos componentes mecânicos e eletrônicos empregados, bem como o desenvolvimento e validação de um protótipo funcional. Os resultados indicam que a automação dos processos de limpeza e tratamento de piscinas é tecnicamente viável, contribui para a redução de falhas humanas, otimiza o tempo de manutenção e assegura padrões adequados de qualidade da água. Conclui-se que o sistema desenvolvido configura-se como uma solução prática, acessível e alinhada às tendências contemporâneas de domótica e automação inteligente.

Palavras-Chave: Automação Residencial. IoT. Piscinas Residenciais. Manutenção Automatizada.

ABSTRACT

The increasing integration of automation technologies in residential settings has driven the development of systems designed to streamline daily tasks. In this context, this paper presents the development of an automated cleaning and maintenance system for residential swimming pools, grounded in the principles of home automation and the Internet of Things (IoT). The system integrates sensors, actuators, and microprocessor-based controllers to automatically monitor and treat water, significantly reducing the need for manual intervention. This proposal aims to enhance resource efficiency, improve safety during chemical handling, and promote sustainable water and energy consumption practices. The methodology involved reviewing technical standards, analyzing the mechanical and electronic components, and developing and validating a functional prototype. Results indicate that automating pool cleaning and treatment processes is technically feasible; it mitigates human error, optimizes maintenance time, and ensures appropriate water quality standards. It is concluded that the developed system represents a practical, accessible solution aligned with contemporary trends in smart home automation.

Keywords: Home Automation. Internet of Things (IoT). Residential Swimming Pools. Automated Maintenance.

LISTA DE ILUSTRAÇÕES

| | |
|--|----|
| Figura 1 – Primeiro registro histórico de piscina – “Grandes Banhos de Mohenjodaro”. | 15 |
| Figura 2 – Componentes estruturais básicos de uma piscina. | 16 |
| Figura 3 – Principais acessórios utilizados na limpeza física de piscinas. | 17 |
| Figura 4 – Faixa de pH | 18 |
| Figura 5 – Estojo para Análise de Parâmetros Químicos da Água | 22 |
| Figura 6 – Arquitetura centralizada de automação residencial | 25 |
| Figura 7 – Arquitetura descentralizada de automação residencial | 26 |
| Figura 8 – Placa Arduino Uno com pinos digitais e analógicos | 29 |
| Figura 9 – Raspberry Pi Modelo B+ | 30 |
| Figura 10 – Bomba de água utilizada majoritariamente em aquários | 31 |
| Figura 11 – Fases do Rational Unified Process RUP. | 34 |
| Figura 12 – Exemplo de Diagrama de Casos de Uso | 36 |
| Figura 13 – Diagrama de Caso de Uso do Sistema | 44 |
| Figura 14 – Testes em bancada | 47 |
| Figura 15 – Fluxograma de validação de dados e tomada de decisão do sistema | 48 |
| Figura 16 – Modal de recomendação de dosagem | 49 |
| Figura 17 – Protótipo final do sistema de automação montado. | 53 |
| Figura 18 – Relé com optoacoplador. | 54 |
| Figura 19 – Validação do sensor de pH. | 55 |
| Figura 20 – Validação do sensor de temperatura. | 56 |
| Figura 21 – Interface Web: Dashboard de monitoramento em tempo real. | 57 |
| Figura 22 – Sensor de Temperatura MF58 (NTC 10K) | 68 |
| Figura 23 – Sensor de Nível LC26M-40 | 69 |
| Figura 24 – Sensor de pH | 70 |
| Figura 25 – Sensor de Turbidez ST100 | 70 |
| Figura 26 – Bomba Submersa JT100 | 71 |
| Figura 27 – Placa Arduino Uno R3 (Controlador Físico) | 71 |
| Figura 28 – Raspberry Pi 3 Model B (Gateway) | 72 |

LISTA DE TABELAS

| | |
|---|----|
| Tabela 1 – Tabela de Dosagem de Produtos Químicos | 21 |
| Tabela 2 – Evolução das tecnologias de automação residencial ao longo dos anos. | 24 |
| Tabela 3 – Comparativo de funcionalidades entre trabalhos correlatos e o sistema proposto | 38 |
| Tabela 4 – Requisitos Funcionais do Sistema de Automação de Piscinas | 42 |
| Tabela 5 – Requisitos Não Funcionais do Sistema de Automação de Piscinas | 43 |
| Tabela 6 – Resumo das Especificações Técnicas dos Componentes de Hardware | 46 |
| Tabela 7 – Resumo dos Testes de Validação Sistêmica | 51 |

SUMÁRIO

| | | |
|----------|--|-----------|
| 1 | INTRODUÇÃO | 11 |
| 1.1 | Objetivo Geral | 12 |
| 1.2 | Objetivos Específicos | 12 |
| 2 | FUNDAMENTAÇÃO TEÓRICA | 14 |
| 2.1 | PISCINAS E SUA MANUTENÇÃO | 14 |
| 2.1.1 | HISTÓRICO E POPULARIZAÇÃO DAS PISCINAS | 14 |
| 2.1.2 | COMPONENTES BÁSICOS DE UMA PISCINA | 15 |
| 2.1.3 | NORMAS E PROCEDIMENTOS TÉCNICOS DE LIMPEZA MANUAL | 17 |
| 2.1.4 | PRODUTOS QUÍMICOS E ACESSÓRIOS USADOS NA LIMPEZA DE PISCINAS | 20 |
| 2.2 | FUNDAMENTOS DA AUTOMAÇÃO RESIDENCIAL | 22 |
| 2.2.1 | HISTÓRICO E EVOLUÇÃO DA AUTOMAÇÃO RESIDENCIAL . . . | 24 |
| 2.2.2 | CONCEITOS TÉCNICOS DE AUTOMAÇÃO RESIDENCIAL | 25 |
| 2.2.3 | COMPONENTES BÁSICOS | 26 |
| 2.3 | FERRAMENTAS UTILIZADAS PARA AUTOMAÇÃO | 27 |
| 2.3.1 | COMPONENTES FÍSICOS E DE CONTROLE (<i>HARDWARE</i>) . . . | 28 |
| 2.3.2 | FERRAMENTAS DE SOFTWARE E METODOLOGIAS DE DESENVOLVIMENTO | 31 |
| 3 | TRABALHOS CORRELATOS | 37 |
| 4 | DESENVOLVIMENTO | 40 |
| 4.1 | Tipo de Pesquisa e Etapas de Construção | 40 |
| 4.2 | Processo de Desenvolvimento de Software | 40 |
| 4.3 | Fase de Concepção (<i>Inception</i>) | 41 |
| 4.4 | Fase de Elaboração (<i>Elaboration</i>) | 43 |
| 4.5 | Fase de Construção (<i>Construction</i>) | 46 |
| 4.6 | Fase de Transição (<i>Transition</i>) | 50 |
| 5 | RESULTADOS E DISCUSSÃO | 52 |
| 5.1 | Integração do Protótipo Físico | 52 |
| 5.2 | Interface de Controle e Monitoramento | 56 |
| 5.3 | Análise de Desempenho e Discussão | 57 |
| 5.4 | Trabalhos futuros | 58 |
| 6 | CONSIDERAÇÕES FINAIS | 60 |

| | |
|--|-----------|
| REFERÊNCIAS | 62 |
| APÊNDICES | 67 |
| APÊNDICE A – DETALHAMENTO DOS COMPONENTES DE HARDWARE | 68 |
| A.1 Sensores | 68 |
| A.2 Atuadores e Controladores | 71 |
| APÊNDICE B – CÓDIGOS FONTE DO SISTEMA | 73 |
| B.1 Firmware do Microcontrolador (Arduino) | 73 |
| B.2 Script do Gateway (Python) | 75 |
| B.3 Controller do Back-end (Java/Spring Boot) | 77 |
| B.4 Front-end (React) | 78 |
| B.5 Algoritmo de Recomendação Química e Alertas (React) | 79 |

1 INTRODUÇÃO

O avanço tecnológico consolidou a automação como elemento essencial na otimização de processos produtivos. Desde a Primeira Revolução Industrial, no século XVIII, esse conceito evoluiu até o paradigma contemporâneo da Indústria 4.0, caracterizado pela integração entre sistemas ciberfísicos, sensoriamento inteligente e soluções digitais (GENYO, 2024). A relevância desse movimento é evidenciada por dados da Confederação Nacional da Indústria, segundo os quais 72% das empresas que implementaram tecnologias digitais registraram aumento de produtividade, enquanto 60% reduziram custos operacionais (Confederação Nacional da Indústria, 2022). A expansão desse cenário repercutiu no ambiente doméstico, onde dispositivos conectados, assistentes virtuais e sistemas inteligentes vêm ampliando o conceito de automação residencial. O setor apresenta crescimento expressivo: no Brasil, a automação residencial avançou 21,8% entre 2023 e 2024 (Money Report, 2024), e projeções globais estimam crescimento médio anual de 27,9% até 2032 (Revista Mundo Elétrico, 2023). Dentro dessa perspectiva de expansão tecnológica, torna-se pertinente examinar áreas específicas do ambiente doméstico que permanecem fortemente dependentes de processos manuais, como a limpeza e a manutenção de piscinas.

Embora a automação residencial se consolide em diversos domínios, a manutenção de piscinas residenciais ainda é majoritariamente manual, exigindo do usuário conhecimento sobre parâmetros físico-químicos da água, dosagens de produtos, cálculo de volume e identificação visual de anomalias (Pool News, 2020). Essas atividades, quando executadas sem precisão, podem gerar desperdício de água e energia, uso inadequado de substâncias químicas e custos elevados de manutenção. Além disso, falhas na dosagem ou na análise dos parâmetros podem comprometer a saúde dos usuários e o desempenho dos equipamentos (LAVOR, 2019). Nesse contexto, formula-se o seguinte problema de pesquisa: Como projetar e implementar um sistema automatizado capaz de realizar a limpeza e a manutenção de piscinas residenciais, reduzindo a intervenção manual e promovendo eficiência operacional, segurança sanitária e sustentabilidade no uso de recursos?

A partir desse questionamento, estabelece-se a seguinte hipótese central: a integração entre sensores, atuadores e controladores microprocessados, associados a tecnologias de automação e Internet das Coisas (IoT), possibilita monitorar parâmetros essenciais da água, como pH, temperatura, turbidez, nível e acionar automaticamente equipamentos de tratamento, garantindo maior precisão na manutenção e minimizando erros humanos. Presume-se, portanto, que um sistema automatizado possa otimizar recursos, aprimorar a qualidade da água e assegurar um processo contínuo e confiável. Se confirmada, essa hipótese reforça a necessidade de investigar sua aplicabilidade técnica, econômica e operacional.

A justificativa para o desenvolvimento deste estudo fundamenta-se em três dimensões complementares. A dimensão social envolve a necessidade de preservar a saúde do usuário, considerando que procedimentos inadequados de tratamento da água podem gerar riscos sanitários. A dimensão ambiental relaciona-se ao uso racional de água e energia, uma vez que a automação tende a reduzir ciclos excessivos de filtragem e dosagens inadequadas de produtos químicos. Em relação à dimensão tecnológica, foi realizada uma análise com base nas pesquisas bibliográficas encontradas, que revelou uma lacuna no desenvolvimento de sistemas que integrem de maneira completa o monitoramento de parâmetros de qualidade da água (como pH e turbidez) com a automação dos atuadores. Embora existam soluções isoladas para aquecimento ou controle químico, observa-se uma carência de estudos recentes que proponham uma arquitetura unificada de IoT, capaz de implementar o ciclo de controle fechado e a gestão remota de dados para piscinas residenciais.

1.1 OBJETIVO GERAL

Desenvolver um sistema automatizado para limpeza e manutenção de piscinas residenciais, com capacidade de reduzir a intervenção manual, otimizar o uso de produtos químicos e promover economia de água e energia, integrando sensores, atuadores e dispositivos IoT ao processo.

1.2 OBJETIVOS ESPECÍFICOS

- Minimizar erros operacionais durante o processo de limpeza por meio de automação e controle preciso de dosagem química.
- Proporcionar maior praticidade e segurança nas etapas de limpeza e monitoramento da piscina.
- Implementar o monitoramento automatizado dos níveis de pH, turbidez, temperatura e nível da água, centralizando as informações em uma interface web.
- Reduzir a dependência de intervenções manuais constantes por meio de alertas do sistema e acionamentos automáticos da bomba de água com base nos parâmetros de qualidade.

Com base nessa contextualização, observa-se que a automação aplicada à manutenção de piscinas representa um campo em expansão, mas ainda pouco explorado academicamente. Para sustentar a proposta apresentada, o próximo capítulo reúne os fundamentos teóricos sobre piscinas e seus métodos tradicionais de tratamento, os princípios da automação residencial, as tecnologias empregadas em sistemas automatizados e os componentes necessários para integrar sensores e controladores inteligentes. Essa fundamentação constitui o embasamento conceitual indispensável para compreender as etapas metodológicas e o desenvolvimento do sistema proposto.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta os fundamentos conceituais e técnicos que sustentam o desenvolvimento do sistema automatizado proposto. Inicialmente, descrevem-se os aspectos estruturais e os procedimentos de manutenção de piscinas, com ênfase nas práticas e parâmetros que orientam a qualidade da água. Em seguida, são expostos os princípios da automação residencial, os componentes e protocolos relevantes à integração de sensores e controladores, e, por fim, as ferramentas de hardware e software empregadas no projeto. A seleção e a organização dos tópicos procuram estabelecer a base teórica necessária para justificar escolhas de projeto e para interpretar os resultados apresentados nos capítulos subsequentes.

2.1 PISCINAS E SUA MANUTENÇÃO

A compreensão da automação aplicada à manutenção de piscinas exige o entendimento prévio de sua estrutura física, de seu funcionamento hidráulico e dos métodos tradicionais utilizados para preservar a qualidade da água. Esses elementos formam a base sobre a qual se fundamentam as etapas de monitoramento e controle automatizado, permitindo identificar quais parâmetros são passíveis de coleta e quais processos podem ser otimizados com o uso de sensores e atuadores. Para isso, esta seção apresenta um panorama histórico das piscinas, descreve seus componentes estruturais e discute os procedimentos manuais de limpeza (físicos e químicos) que orientam a manutenção convencional. A partir dessa exposição, evidenciam-se as limitações dos métodos tradicionais e a necessidade de soluções automatizadas.

2.1.1 HISTÓRICO E POPULARIZAÇÃO DAS PISCINAS

O termo piscina deriva do latim *piscis*, utilizado para designar reservatórios destinados à recreação, natação, rituais ou uso coletivo ([Engevil Piscinas, 2025](#)). Existem registros de tanques de banho que remontam a aproximadamente 2600 a.C., como os “Grandes Banhos de Mohenjodaro”, cuja função possivelmente se relacionava a práticas ceremoniais. Conforme ilustrado na [Figura 1](#), trata-se de uma das primeiras estruturas documentadas que apresentam características semelhantes às piscinas contemporâneas. Ao longo dos séculos, diferentes civilizações desenvolveram estruturas semelhantes, adaptadas a necessidades sociais, culturais e recreativas.

Com o avanço tecnológico do século XX, o uso de materiais como gunite (mistura de cimento, areia e água) e a incorporação de sistemas de filtração e cloração possibilitaram a manutenção contínua da água, sem necessidade de esvaziamento recorrente ([FIBRATEC](#),

2021). Tais inovações contribuíram para a popularização das piscinas residenciais e demandaram o desenvolvimento de práticas de manutenção física e química, que servem de referência para os sistemas automatizados contemporâneos.

Figura 1 – Primeiro registro histórico de piscina – “Grandes Banhos de Mohenjodaro”.



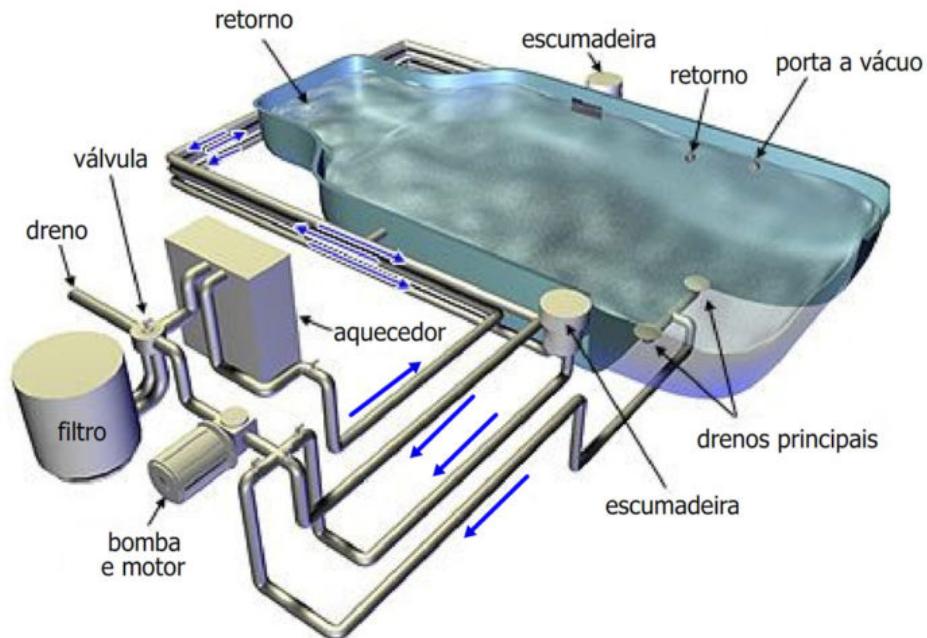
Fonte: [Fibratec \(2021\)](#).

A seguir, apresentam-se os principais componentes estruturais que permitem o funcionamento de uma piscina moderna e que, futuramente, serão integrados à automação.

2.1.2 COMPONENTES BÁSICOS DE UMA PISCINA

Segundo [Harris \(2025\)](#), as piscinas residenciais e públicas, apesar das diferenças de tipologia e porte, compartilham um conjunto de elementos funcionais que garantem a circulação, filtragem e tratamento da água. Esses componentes formam a estrutura hidráulica básica que permite a manutenção contínua da qualidade da água e o correto funcionamento de dispositivos auxiliares. A [Figura 2](#) apresenta uma visão geral desses elementos estruturais, destacando a bomba, o filtro, os drenos, as tubulações e os pontos de retorno, que compõem o ciclo de movimentação e purificação do volume total da piscina.

Figura 2 – Componentes estruturais básicos de uma piscina.



Fonte: Harris (2025).

A bomba motorizada cumpre a função central de impulsionar a água pelo sistema, enviando-a para as etapas de filtração e tratamento químico. Já o filtro, geralmente preenchido com areia ou elementos sintéticos, remove partículas sólidas, como poeira, folhas e microrganismos, garantindo maior transparência à água. Os drenos e skimmers são responsáveis pela coleta inicial da água, enquanto as tubulações de PVC¹ interligam todos os elementos hidráulicos, assegurando o fluxo contínuo. Em algumas configurações, incluem-se ainda aquecedores que regulam a temperatura da água, agregando conforto ao uso recreativo.

Além desses elementos fixos, a manutenção cotidiana envolve equipamentos manuais destinados à limpeza física da piscina. A Figura 3 reúne os principais acessórios utilizados nesse processo, como o aspirador de escova, a peneira, a escova de parede e o cabo telescópico, que permite alcançar regiões de difícil acesso. Segundo Benedito et al. (2024), esses instrumentos são indispensáveis para remover resíduos decantados, partículas flutuantes e biofilmes aderidos às paredes da piscina, constituindo o primeiro nível de intervenção antes do tratamento químico.

¹ Sigla para Poli(cloreto de vinila), um polímero termoplástico versátil, conhecido por sua durabilidade, resistência química e ampla utilização em tubos, conexões e revestimentos.

Figura 3 – Principais acessórios utilizados na limpeza física de piscinas.



Fonte: Adaptado de [Benedito et al. \(2024\)](#).

A compreensão desses componentes, tanto estruturais quanto acessórios, é fundamental para o desenvolvimento de sistemas automatizados, uma vez que muitos dos processos realizados manualmente, como circulação, remoção de impurezas e monitoramento da qualidade da água, podem ser otimizados por meio de sensores, atuadores e controladores eletrônicos, tema aprofundado nos tópicos subsequentes.

2.1.3 NORMAS E PROCEDIMENTOS TÉCNICOS DE LIMPEZA MANUAL

A manutenção adequada de uma piscina não depende apenas dos equipamentos estruturais apresentados na seção anterior, mas também da aplicação criteriosa de normas e procedimentos técnicos que asseguram a qualidade sanitária da água. Conforme destaca [Portal Ideia \(2025\)](#), a ausência de práticas corretas de limpeza pode resultar em diversos problemas de saúde, como dermatites, micoses e outras infecções, o que reforça a importância de um tratamento contínuo e devidamente monitorado. Para prevenir tais riscos, a Associação Brasileira de Normas Técnicas (ABNT) estabelece parâmetros essenciais que orientam o controle físico e químico da água.

Diversos poluentes influenciam diretamente esses parâmetros, incluindo suor, urina, cabelos, óleos naturais da pele, insetos, folhas e formação de algas, conforme observa [Atcllor \(2021\)](#). Esses elementos alteram, sobretudo, o pH da água e a concentração de cloro, variáveis fundamentais para garantir segurança aos usuários. Segundo [Leite \(2020\)](#), o pH é um indicador que expressa a acidez ou alcalinidade do fluido, variando entre valores menores que 7 (ácidos) e maiores que 7 (alcalinos). A [Figura 4](#) apresenta a

faixa recomendada para piscinas, entre 7,2 e 7,8, parâmetro indispensável para prevenir irritações e preservar a integridade dos equipamentos.

Figura 4 – Faixa de pH



Fonte: Atclllor (2021)

Embora a limpeza física, obtida por escovas, redes ou aspiradores, remova apenas impurezas visíveis, ela é insuficiente para eliminar substâncias dissolvidas, como suor, urina e óleos, que permanecem em suspensão ou solução e ultrapassam a capacidade de retenção dos filtros (ATCLLLOR, 2021). Por esse motivo, a manutenção adequada deve integrar procedimentos químicos eficazes, de modo a combater microrganismos e restabelecer o equilíbrio da água.

Além disso, uma piscina considerada limpa precisa atender a critérios objetivos, como a ausência de bactérias do grupo coliforme ou *Staphylococcus aureus*², boa visibilidade do fundo e superfície livre de sujeiras, conforme determinado por Atclllor (2021) e pelas normas sanitárias vigentes. Esses requisitos reforçam que a manutenção envolve mais do que a aparência da água, exigindo uma análise sistemática dos parâmetros fisico-químicos.

A determinação da quantidade de produtos necessários para o tratamento também dependem do conhecimento da área e do volume da piscina, que variam de acordo com seu formato geométrico. Como apresentado nas equações desta seção, diferentes fórmulas são aplicadas em piscinas retangulares, circulares ou ovais, incluindo casos em que o fundo é inclinado, nos quais se considera a profundidade média. Esses cálculos garantem precisão na dosagem e evitam desperdícios, além de evitar desequilíbrios químicos que comprometeriam o processo de desinfecção. O dimensionamento da piscina é fundamental para determinar a quantidade adequada de produtos químicos e garantir uma higienização eficiente, sem excessos ou falhas de tratamento. O cálculo da área e do volume varia conforme o formato da piscina, sendo possível aplicar diferentes fórmulas geométricas para cada tipo de estrutura.

² Bactéria coco Gram-positiva, frequentemente encontrada na pele e nas fossas nasais humanas, responsável por infecções de gravidade variável.

Piscina Retangular

$$A = \text{comprimento} \times \text{largura}$$

$$V = \text{comprimento} \times \text{largura} \times \text{profundidade}$$

Piscina Circular

$$A = \pi r^2$$

$$V = \pi r^2 h$$

Piscina Oval (Elíptica)

$$A = \pi \cdot \frac{a}{2} \cdot \frac{b}{2}$$

$$V = A \cdot h$$

Nos casos em que a piscina possui fundo inclinado, a profundidade considerada deve ser a média entre a parte mais rasa e a mais funda:

$$h_m = \frac{h_{\text{maior}} + h_{\text{menor}}}{2}$$

Com o volume da piscina devidamente determinado, é possível avançar para as etapas que compõem o tratamento químico da água. Segundo Silva (2021) , o tratamento completo da água segue uma sequência de cinco etapas: oxidação, coagulação e floculação, decantação, filtração e correção do pH, semelhante aos processos utilizados em estações industriais de tratamento de água. Na oxidação, adiciona-se cloro para eliminar matéria orgânica e facilitar a remoção de metais como ferro e manganês. Em seguida, agentes coagulantes, como sulfato de alumínio ou cloreto férreo, desestabilizam partículas suspensas, que posteriormente se agrupam durante a fase de floculação. A decantação deposita essas partículas no fundo da piscina; após isso, o filtro retém as impurezas acumuladas. Por fim, realiza-se a correção do pH, garantindo estabilidade química e prevenindo tanto a corrosão das tubulações quanto danos aos usuários.

Entretanto, a execução manual dessas etapas está sujeita a falhas operacionais que comprometem a eficácia teórica do processo. Segundo Genco Química Industrial (2004), a aplicação de produtos químicos sem a devida precisão frequentemente resulta em sobredosagem ou uso incorreto, gerando efeitos adversos. O excesso de algicidas³, por exemplo, pode provocar espuma na superfície, enquanto o uso inadequado de decantadores

³ São produtos químicos usados para matar ou prevenir o crescimento de algas em ambientes aquáticos

à base de sulfato de alumínio pode deixar resíduos em suspensão, turvando a água. Um risco ainda mais crítico é a mistura accidental ou proposital de compostos incompatíveis, que pode provocar reações químicas extremas durante o manuseio.

Essas falhas no equilíbrio químico impactam diretamente o conforto dos usuários e a integridade da infraestrutura. A falta de controle rigoroso do pH gera desconforto imediato: valores abaixo de 7,2 causam irritação e aceleram a corrosão de equipamentos. Por outro lado, o pH acima de 7,8 reduz drasticamente a eficiência da desinfecção e favorece incrustações nas tubulações. Adicionalmente, a dosagem desbalanceada de oxidantes pode gerar subprodutos irritantes, responsáveis por odores fortes, frequentemente confundidos com excesso de produto.

No âmbito sanitário, a intermitência característica do tratamento manual cria janelas de contaminação. A ineficiência nos ciclos de filtragem e circulação, somada ao desequilíbrio químico, permite a proliferação de microrganismos e o aumento do risco de transmissão de doenças. Por fim, a imprecisão acarreta problemas estéticos, como a turbidez gerada pelo rápido desenvolvimento de algas ou alterações na cor da água devido à reação incorreta de produtos com metais presentes no tanque.

Diante da complexidade e dos riscos apresentados pela manipulação incorreta, torna-se indispensável o conhecimento aprofundado dos insumos utilizados. A seção a seguir detalha os produtos químicos e acessórios empregados na limpeza, abordados na [subseção 2.1.4](#), cujas características fundamentam as decisões de automação do sistema proposto.

2.1.4 PRODUTOS QUÍMICOS E ACESSÓRIOS USADOS NA LIMPEZA DE PISCINAS

O tratamento adequado e corretamente executado durante a limpeza de uma piscina, tanto físico quanto químico, é essencial para assegurar a qualidade da água e prevenir infecções ou doenças de origem hídrica. Dessa forma, é fundamental compreender quais produtos utilizar, como aplicá-los corretamente e qual o método mais eficiente para a realização do tratamento físico.

A manutenção química da água dependem da correta escolha, dosagem e aplicação de produtos destinados ao controle dos parâmetros físico-químicos. Segundo [Atclllor \(2021\)](#), esses produtos atuam no ajuste do pH, na estabilização da alcalinidade, na desinfecção da água e na remoção de partículas e metais dissolvidos. A [Tabela 1](#) apresenta uma tabela de dosagem que orienta a aplicação dos principais agentes químicos utilizados na rotina de manutenção.

Tabela 1 – Tabela de Dosagem de Produtos Químicos

| TABELA DE DOSAGEM | PRODUTO | APLICAÇÃO | DOSAGEM 1.000 Litros | VOLUME PISCINA (LITROS) | | | | | | | |
|----------------------------|--------------------------------------|-----------------------|-------------------------|-------------------------|------|------|-------|-------|-------|-------|-------|
| | | | | 10 | 20 | 30 | 40 | 50 | 60 | 70 | 80 |
| AJUSTE | Elevador Alcalinidade (Bicarbonato) | Sempre que Necessário | 20 grs | 200 | 400 | 600 | 800 | 1.000 | 1.200 | 1.400 | 1.600 |
| | pH + (Líquido) pH 6.8 – 7.0 | | 15 ml | 150 | 300 | 450 | 600 | 750 | 900 | 1.050 | 1.200 |
| | pH + (Líquido) pH abaixo 6.8 | | 20 ml | 200 | 400 | 600 | 800 | 1.000 | 1.200 | 1.400 | 1.600 |
| | Elevador pH (Barrilha) pH 6.8 – 7.0 | | 10 grs | 100 | 200 | 300 | 400 | 500 | 600 | 700 | 800 |
| | Elevador pH (Barrilha) pH abaixo 6.8 | | 20 grs | 200 | 400 | 600 | 800 | 1.000 | 1.200 | 1.400 | 1.600 |
| | pH- (Líquido) | | 10 ml | 100 | 200 | 300 | 400 | 500 | 600 | 700 | 800 |
| SANITIZAÇÃO | Cloro Tradicional 65% | 2x a 3x Semana | 4 grs | 40 | 80 | 120 | 160 | 200 | 240 | 280 | 320 |
| | Dicloro Estabilizado 50% - 55% | | 4 grs | 40 | 80 | 120 | 160 | 200 | 240 | 280 | 320 |
| | Cloro 3x1/ Multiação 40%- 42% | | 5 grs | 50 | 100 | 150 | 200 | 250 | 300 | 350 | 400 |
| | Cloro Líquido 12% | | 100 ml | 1 L. | 2 L. | 3 L. | 4 L. | 5 L. | 6 L. | 7 L. | 8 L. |
| CLARIFICAÇÃO DECANTAÇÃO | Clarificante Líquido | 1x Semana | 4 ml | 40 | 80 | 120 | 160 | 200 | 240 | 280 | 320 |
| | Sulfato de Alumínio | | 30 grs | 300 | 600 | 900 | 1.200 | 1.500 | 1.800 | 2.100 | 2.400 |
| ÁGUA VERDE OU METAIS | Sulfato de Cobre | 1x Semana | 2 grs | 20 | 40 | 60 | 80 | 100 | 120 | 140 | 160 |
| | Algicida Manutenção | | 5 ml | 50 | 100 | 150 | 200 | 250 | 300 | 350 | 400 |
| | Algicida Choque | Sempre que Necessário | 7 ml | 70 | 140 | 210 | 280 | 350 | 420 | 490 | 560 |
| | Genquest / Sol. Água Poço | | 20 ml | 200 | 400 | 600 | 800 | 1.000 | 1.200 | 1.400 | 1.600 |

Fonte: Atclllor (2021).

Os produtos empregados no tratamento químico incluem:

- **Elevador de alcalinidade:** eleva a alcalinidade total, garantindo estabilidade ao pH.
- **Barrilha e pH+:** utilizados para elevar o pH quando este se encontra abaixo do ideal.
- **Redutor de pH (pH-):** empregados para diminuir o pH quando a água está excessivamente alcalina.
- **Hipoclorito de sódio, cloro, dicloro, multiação:** agentes sanitizantes responsáveis pela desinfecção.
- **Sulfato de alumínio e clarificantes:** agentes de coagulação e decantação.
- **Sulfato de cobre e algicida:** utilizados para combater a proliferação de algas.
- **Removedores de metais (Genquest, solução para água de poço):** eliminam manchamentos causados por íons metálicos.

A aplicação correta desses produtos depende da medição sistemática dos parâmetros da água. Para isso, utilizam-se estojos de análise específicos, capazes de identificar valores

de pH, alcalinidade e teor de cloro. A Figura 5 apresenta um exemplo de estojo de análise utilizado na avaliação da qualidade da água.

Figura 5 – Estojo para Análise de Parâmetros Químicos da Água



Fonte: [Genco \(2025\)](#).

Os valores obtidos nas análises orientam a escolha dos produtos a serem aplicados. Caso o pH esteja abaixo de 7,0, recomenda-se o uso de barrilha ou elevador de pH. Se a alcalinidade estiver reduzida, utiliza-se o elevador de alcalinidade. Quando o teor de cloro estiver baixo, adiciona-se cloro líquido ou granulado, conforme a dosagem apresentada anteriormente.

A turbidez é um dos principais indicadores da necessidade de tratamento químico. Em situações de água opaca, recomenda-se o uso de clarificantes. Quando a água apresenta partículas em suspensão, aplicam-se procedimentos de floculação ou decantação, geralmente realizados com sulfato de alumínio.

Após a aplicação dos produtos, é necessário aguardar entre 6 e 12 horas antes de iniciar a aspiração do fundo da piscina, garantindo assim a eficácia das reações químicas.

Com a compreensão desses produtos e procedimentos, evidencia-se a complexidade do tratamento manual, o que reforça a necessidade de soluções tecnológicas baseadas em automação residencial, assunto abordado na próxima seção.

2.2 FUNDAMENTOS DA AUTOMAÇÃO RESIDENCIAL

Após compreender os aspectos estruturais e os métodos tradicionais de manutenção de piscinas, torna-se necessário examinar os princípios da automação residencial, uma vez que o sistema proposto se insere nesse contexto tecnológico. Este tópico aborda a

evolução histórica da automação aplicada ao ambiente doméstico, seus conceitos técnicos, componentes essenciais e protocolos de comunicação, que viabilizam o controle remoto e inteligente de diferentes dispositivos. Essa fundamentação permite compreender como as tecnologias emergentes podem ser aplicadas para aprimorar processos cotidianos, incluindo a manutenção automatizada de piscinas.

A automação residencial consiste na integração de sistemas tecnológicos destinados ao controle e à otimização de funções domésticas, como segurança, iluminação, climatização e comunicação. Essa integração, tem como propósito promover a qualidade de vida dos moradores e a racionalização no uso de recursos da habitação (MURATORI; BÓ, 2011).

O principal objetivo da automação residencial é proporcionar comodidade e segurança aos usuários, por meio do acionamento remoto e da integração de dispositivos inteligentes (DAGOSTIM; JORGE, 2022). A automação residencial é composta por um conjunto de benefícios fundamentais que estruturam o conceito de casa inteligente. Entre seus principais pilares, destacam-se:

- **Conforto:** tem como objetivo facilitar tarefas cotidianas, permitindo ao usuário controlar dispositivos como lâmpadas, ar-condicionados e sistemas de irrigação de forma remota (DAGOSTIM; JORGE, 2022).
- **Segurança:** a integração de câmeras, fechaduras eletrônicas e sensores de presença possibilita o monitoramento remoto da residência, reforçando a proteção e a praticidade (DAGOSTIM; JORGE, 2022).
- **Economia:** a automação contribui para o uso racional de energia, com sistemas capazes de desligar lâmpadas automaticamente e ajustar a climatização conforme a necessidade, evitando desperdícios e promovendo maior eficiência energética (DAGOSTIM; JORGE, 2022).

Para que a automação funcione de forma adequada, é necessário que os dispositivos possuam conectividade, acesso à internet e capacidade de comunicação com um sistema central de controle, responsável pela coleta e troca de informações entre os equipamentos.

Praticamente todos os aparelhos eletrônicos que possuem algum tipo de acionamento podem ser automatizados, como sistemas de iluminação, portões, climatização e segurança. Esses dispositivos são conectados a uma central de controle, que pode ser acessada por meio de um *display touch*⁴, localizado na própria central, aplicativos para smartphones ou comandos de voz (DAGOSTIM; JORGE, 2022).

⁴ *Display touch*: superfície sensível ao toque que permite interação direta com o sistema.

2.2.1 HISTÓRICO E EVOLUÇÃO DA AUTOMAÇÃO RESIDENCIAL

A automação residencial, embora recente quando comparada a outras áreas tecnológicas, apresenta avanços significativos ao longo das últimas décadas. Na década de 1970, surgiram nos Estados Unidos os primeiros módulos inteligentes com base na transmissão de dados pela rede elétrica doméstica, utilizando a tecnologia PLC (*Power Line Communication*) (MURATORI; BÓ, 2011). Essa inovação marcou o início da integração entre dispositivos elétricos e sistemas de comunicação, permitindo que comandos simples fossem enviados por meio da fiação já existente.

Com o avanço da informática, o aprimoramento dos sistemas embarcados e a popularização da internet, a automação residencial passou a incorporar dispositivos capazes de monitorar e controlar equipamentos à distância, consolidando o conceito contemporâneo de residência conectada. Esse movimento possibilitou o desenvolvimento de soluções cada vez mais acessíveis e eficientes, ampliando o uso de sistemas inteligentes em diferentes ambientes domésticos.

A Tabela 2 apresenta a evolução de algumas das principais tecnologias utilizadas na automação residencial ao longo dos anos, evidenciando o crescimento expressivo de funcionalidades como monitoramento de segurança, controle de iluminação, sistemas de áudio distribuído e gerenciamento energético.

Tabela 2 – Evolução das tecnologias de automação residencial ao longo dos anos.

| Tecnologia | 2003 | 2004 | 2005 | 2006 | 2015(*) |
|----------------------------|------|------|------|------|---------|
| Cabeamento estruturado | 42% | 61% | 49% | 53% | 80% |
| Monitoramento de segurança | 18% | 28% | 29% | 32% | 81% |
| Multiroom audio | 9% | 12% | 15% | 16% | 86% |
| Home Theater | 9% | 8% | 11% | 12% | 86% |
| Controle de iluminação | 1% | 2% | 6% | 8% | 75% |
| Automação integrada | 0% | 2% | 6% | 6% | 70% |
| Gerenciamento de energia | 1% | 5% | 11% | 11% | 62% |

Fonte: Muratori e Bó (2011).

Ao compreender a evolução histórica e o crescimento das tecnologias aplicadas ao ambiente doméstico, torna-se possível identificar os fundamentos que sustentam os sistemas automatizados contemporâneos. O avanço dessas soluções evidencia uma trajetória marcada pela ampliação da conectividade, pela integração entre diferentes dispositivos e pela busca crescente por eficiência, conforto e segurança.

No próximo tópico, são apresentados os principais conceitos e componentes que constituem a base dos sistemas de automação, incluindo controladores, sensores, atuadores

e protocolos de comunicação. Esses elementos são essenciais para compreender o funcionamento e a integração entre os dispositivos que possibilitam a automação em ambientes residenciais.

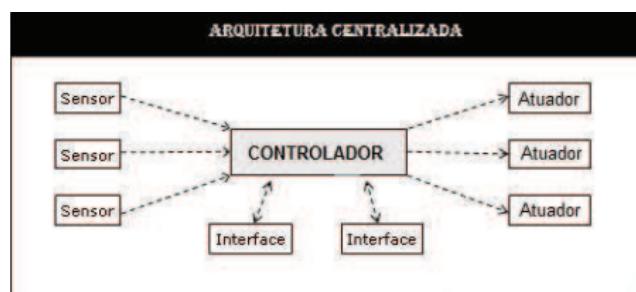
2.2.2 CONCEITOS TÉCNICOS DE AUTOMAÇÃO RESIDENCIAL

Tecnicamente denominada domótica⁵, a automação residencial tem como principal objetivo acionar, monitorar, integrar e controlar diferentes variáveis de uma residência, como iluminação, climatização, áudio e vídeo, a fim de promover eficiência, comodidade e segurança ao usuário (OLIVEIRA; ALVES, 2019). No Brasil, o termo mais utilizado é automação residencial, uma tradução derivada da expressão americana *home automation*. Apesar disso, essa tradução não abrange plenamente a concepção de domótica, que envolve não apenas o controle remoto, mas a integração inteligente de sistemas.

O uso de tecnologias residenciais tem crescido de maneira expressiva no país; entretanto, o setor da construção civil ainda não acompanha plenamente o ritmo de evolução observado em áreas como a indústria automotiva, que já emprega amplamente tecnologias embarcadas⁶ (HIPÓLITO; SILVA, 2018). Essa diferença evidencia a necessidade de maior disseminação dos conceitos técnicos que sustentam a automação no ambiente doméstico.

De acordo com Accardi e Dodonov (2012), a forma como os componentes de um sistema residencial se comunicam depende diretamente da arquitetura adotada, que pode ser centralizada ou descentralizada. Em uma arquitetura centralizada, todos os dispositivos se conectam a um único controlador principal, responsável por processar e executar todas as ações do sistema; esse arranjo é ilustrado na Figura 6. Essa configuração exige que o controlador possua alta capacidade de processamento e confiabilidade operacional.

Figura 6 – Arquitetura centralizada de automação residencial



Fonte: Hipólito e Silva (2018).

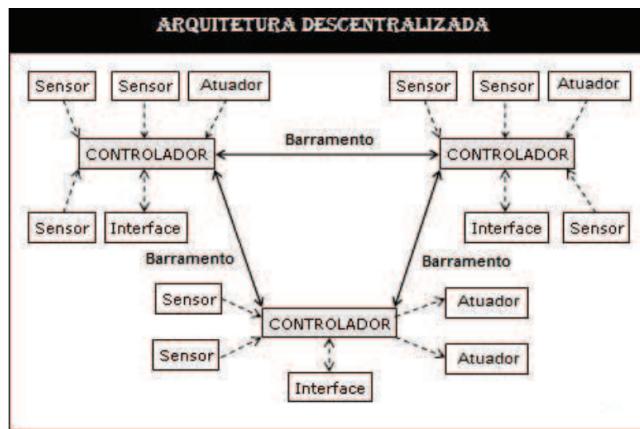
Na arquitetura descentralizada, diferentes controladores coexistem e se comunicam

⁵ Domótica: termo que designa a integração de tecnologias destinadas ao controle inteligente de ambientes residenciais.

⁶ Tecnologia embarcada: computador especializado, composto por hardware e software dedicados, integrados a um sistema maior para execução de funções específicas.

entre si por meio de um barramento de dados⁷, compartilhando o gerenciamento dos dispositivos interconectados; esse modelo encontra-se esquematizado na Figura 7. O modelo descentralizado distribui a responsabilidade entre múltiplos módulos, reduzindo a dependência de um único ponto de falha e aumentando a flexibilidade da instalação.

Figura 7 – Arquitetura descentralizada de automação residencial



Fonte: Hipólito e Silva (2018).

A compreensão dessas arquiteturas é essencial para o desenvolvimento e a implementação de sistemas automatizados, pois define a maneira como os dispositivos se organizam, trocam informações e executam comandos. Nas seções seguintes, serão apresentados os principais conceitos e componentes técnicos que constituem a base dos sistemas de automação residencial, incluindo controladores, sensores, atuadores e protocolos de comunicação.

2.2.3 COMPONENTES BÁSICOS

A automação residencial é composta por diversos elementos que, em conjunto, permitem o controle eficiente dos dispositivos instalados no ambiente doméstico. Esses componentes vão desde sensores simples até centrais complexas de automação. A seguir, são apresentados os principais elementos que estruturam essa tecnologia.

- **Camadas de dispositivos:**

1. **Sensores:** Segundo Leite (2020), o sensor é um dispositivo sensível ao ambiente no qual está inserido, capaz de detectar alterações em variáveis como temperatura, luminosidade ou movimento. Sua função consiste em captar essas mudanças e convertê-las em sinais elétricos que possam ser interpretados por um controlador e posteriormente utilizados pela rede de automação.

⁷ *Barramento de dados:* sistema de comunicação que permite a troca de informações entre dispositivos sem depender de um ponto central de controle.

2. **Atuadores:** São dispositivos eletromecânicos acionados pelo sistema para executar funções específicas, como ativar lâmpadas, fechaduras magnéticas, motores, válvulas ou sirenes [Hipólito e Silva \(2018\)](#). O atuador materializa a ação física determinada pela lógica programada.
 3. **Controladores:** Responsáveis por monitorar os dados coletados pelos sensores e acionar os respectivos atuadores conforme a lógica definida. O controlador pode operar como módulo independente ou integrar-se a centrais mais complexas ([HIPÓLITO; SILVA, 2018](#)).
 4. **Interfaces:** Dispositivos que permitem ao usuário interagir com o sistema automatizado, como painéis digitais, páginas web, aplicativos móveis ou assistentes de voz ([ACCARDI; DODONOV, 2012](#)). Essas interfaces constituem a camada de acesso humano ao sistema.
- **Camada de comunicação/rede:** Segundo [Accardi e Dodonov \(2012\)](#), a comunicação entre os dispositivos ocorre por meio de protocolos, que funcionam como acordos que definem regras e padrões para a troca de informações. Assim, o protocolo estabelece como os equipamentos interagem entre si dentro do sistema.
Entre os protocolos mais utilizados em automação residencial estão *Ethernet*, *X-10*, *HomePNA* e *Wi-Fi*. Alguns foram desenvolvidos especificamente para ambientes residenciais, enquanto outros derivam de aplicações industriais ou comerciais.
 - **Camada de controle/automação lógica:** Também denominada central de automação, essa camada representa o núcleo lógico do sistema, responsável por gerenciar os dispositivos conectados. Ela processa os dados recebidos e executa ações conforme as instruções programadas.
A configuração da central é realizada por meio de software dedicado, acessado a partir de computadores ou dispositivos móveis. Essa estrutura é escalável⁸, permitindo a adição contínua de novos dispositivos conforme a necessidade de expansão.

Nesse contexto, o próximo capítulo apresenta o desenvolvimento de um sistema automatizado para limpeza de piscinas, aplicando os conceitos estudados e demonstrando como a integração entre hardware e software pode oferecer uma solução segura, eficiente e inovadora para a manutenção residencial.

2.3 FERRAMENTAS UTILIZADAS PARA AUTOMAÇÃO

Esta seção apresenta as principais ferramentas, dispositivos e tecnologias empregadas no desenvolvimento do sistema automatizado proposto, abrangendo tanto os

⁸ *Escalável*: característica de sistemas capazes de aumentar sua capacidade ou complexidade sem perda de desempenho.

componentes físicos de hardware quanto as plataformas de controle utilizadas na integração entre sensores, atuadores e algoritmos de decisão.

2.3.1 COMPONENTES FÍSICOS E DE CONTROLE (*HARDWARE*)

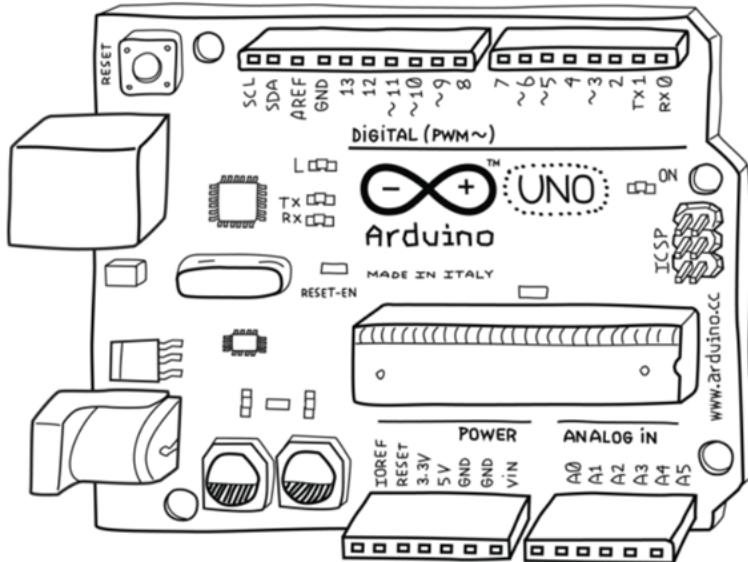
O sistema de automação requer componentes capazes de processar informações, interagir com o ambiente físico, executar ações mecânicas e realizar medições em tempo real. Entre esses elementos estão plataformas de prototipagem, microcomputadores, sensores especializados e atuadores eletromecânicos.

Arduino

Segundo Silva e Fernandes (2025), o Arduino é uma plataforma de prototipagem eletrônica de código aberto que permite o desenvolvimento de projetos simples ou complexos de forma acessível. Seu caráter open-source possibilitou a formação de uma comunidade global de desenvolvedores, tornando-o um dos elementos centrais da cultura maker. Essa cultura, por sua vez, deriva do movimento *Do It Yourself* (DIY), que incentiva a criação de projetos próprios com o uso de ferramentas acessíveis, como impressoras 3D e microcontroladores (BROCKVELD; TEIXEIRA; SILVA, 2017).

A placa Arduino é composta por diversos componentes, sendo o microcontrolador o núcleo responsável pelo processamento das instruções. Conforme destaca Massimo Banzi, um dos criadores da plataforma, o Arduino é consideravelmente menos potente que um computador convencional, porém extremamente útil para construção de dispositivos interativos dada sua simplicidade, baixo custo e modularidade (SILVA; FERNANDES, 2025).

Figura 8 – Placa Arduino Uno com pinos digitais e analógicos



Fonte: Silva e Fernandes (2025).

A Figura 27 apresenta o Arduino Uno, que possui 14 pinos digitais e 6 pinos analógicos, permitindo a conexão com uma variedade de sensores e atuadores, possibilitando a interação direta entre o sistema e o ambiente físico. São alguns dos elementos que compõem o Arduino: a placa protoboard, que consiste em uma base utilizada para construir dispositivos eletrônicos temporários, sem a necessidade de soldagem, utilizando *jumpers*⁹ para a conexão (Grupo de Robótica, 2012).

Raspberry Pi

De acordo com Jucá e Pereira (2018), o Raspberry Pi é considerado um dos menores computadores do mundo, apresentando dimensões semelhantes às de um cartão de crédito. O dispositivo inclui porta *HDMI*¹⁰, portas *USB*¹¹, pinos de conexão *GPIO*¹², entrada *Ethernet*, conector P2 para áudio, módulo *Wi-Fi* e *Bluetooth* integrados.

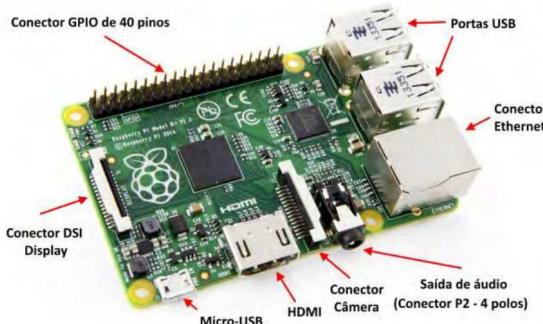
⁹ Condutor curto usado para fechar, abrir ou desviar parte de um circuito eletrônico

¹⁰ *High-Definition Multimedia Interface*: interface digital para transmissão de áudio e vídeo em alta definição.

¹¹ (*Universal Serial Bus*) é um padrão universal para conectar, comunicar e fornecer energia entre computadores e periféricos (mouses, teclados, pen drives, etc.)

¹² *GPIO (General-Purpose Input/Output)* são portas configuráveis em microcontroladores (como Raspberry Pi e Arduino) que atuam como entradas (para ler sensores ou botões) ou saídas (para controlar LEDs e motores)

Figura 9 – Raspberry Pi Modelo B+



Fonte: [Jucá e Pereira \(2018\)](#).

Como observado na Figura 28, o hardware do Raspberry Pi funciona de maneira integrada em uma única placa, o que reduz custos e possibilita que o dispositivo execute diversas funções de um computador convencional, como acesso à internet, edição de textos e reprodução de vídeos. Além disso, seus pinos *GPIO* permitem interação com sensores, possibilitando aplicações em automação e sistemas embarcados.

Em outros contextos, o poder de processamento e a conectividade nativa do Raspberry Pi são explorados para desempenhar a função de *Gateway IoT*. Ele atua como o elo entre a camada física (Arduino) e a camada de aplicação (Servidor Web), viabilizando a transmissão de dados e o controle remoto do sistema.

Sensores Específicos

Os sensores desempenham papel fundamental na automação, pois realizam a leitura de variáveis físicas ou químicas e as convertem em sinais elétricos interpretáveis pelo sistema. Entre os sensores utilizados no projeto, destacam-se:

- **Sensor de Temperatura:** segundo [Leite \(2020\)](#), identifica variações térmicas em equipamentos ou processos, permitindo ajustes automáticos conforme a condição medida.
- **Sensor de pH:** mede o nível de acidez ou alcalinidade da água, informação essencial para o controle químico em piscinas, lagos ou reservatórios ([LEITE, 2020](#)).
- **Sensor de Turbidez:** avalia o grau de turbidez de um líquido pela comparação entre um feixe de luz incidente e outro transmitido através de uma amostra ([CARDOSO, 2011](#)).
- **Sensor de Nível:** de acordo com [Souza \(2018\)](#), detecta a altura ou volume de líquidos ou materiais granulares em um reservatório, emitindo sinais conforme o nível varia.

Atuadores (Motores)

Os atuadores são responsáveis por executar ações físicas decorrentes das decisões tomadas pelo sistema. Em um ambiente automatizado, eles podem acionar bombas, válvulas, motores, travas ou iluminação, de acordo com os dados enviados pelos sensores ou com comandos diretos do usuário ([FLORÊNCIO, 2015](#)).

No contexto da automação de piscinas, um exemplo de atuador é a bomba de água como a da Figura 10. Quando o sensor de nível identifica redução significativa no volume do reservatório, o controlador aciona a bomba para restabelecer o nível adequado. Da mesma forma, o usuário pode ativar manualmente o sistema por meio de uma interface digital.

As bombas de água utilizadas em sistemas residenciais funcionam, em sua maioria, de forma submersa, instaladas dentro do reservatório do filtro. Entre suas vantagens estão a facilidade de instalação e a operação silenciosa, uma vez que o corpo da bomba permanece submerso ([LUCIFABIO et al., 2023](#)).

Figura 10 – Bomba de água utilizada majoritariamente em aquários



Fonte: [Lucifabio et al. \(2023\)](#).

A compreensão dos componentes físicos utilizados no sistema permite visualizar como sensores, controladores e atuadores interagem para formar uma estrutura automatizada funcional. No capítulo seguinte, apresenta-se o desenvolvimento do sistema proposto, evidenciando a integração entre hardware e software necessária para aprimorar o processo de limpeza de piscinas residenciais.

2.3.2 FERRAMENTAS DE SOFTWARE E METODOLOGIAS DE DESENVOLVIMENTO

Linguagens de programação, segundo [Jesus \(2025\)](#), constituem um conjunto de regras semânticas e sintáticas que permitem a comunicação de instruções a um computador.

Por meio delas, o desenvolvedor define os dados utilizados, a forma de armazenamento e as ações que o sistema deve executar mediante condições específicas. Essas linguagens possibilitam a construção de programas, sites, aplicações móveis e diversos outros tipos de software.

No desenvolvimento de sistemas modernos, observa-se a divisão entre front-end e back-end. O front-end corresponde à camada visual e interativa com a qual o usuário mantém contato direto. Essa camada pode ser construída com HTML, CSS e JavaScript, bem como com frameworks como React, que ampliam a produtividade e a modularidade do processo de desenvolvimento (JÚNIOR et al., 2025). Já o back-end, conforme [Calça \(2022\)](#), é responsável por toda a infraestrutura lógica que sustenta as ações realizadas no front-end, incluindo autenticação, comunicação com o banco de dados e processamento de operações internas.

Java

A linguagem Java é amplamente utilizada no desenvolvimento de aplicações web e corporativas. Classificada como orientada a objetos e multiplataforma, apresenta desempenho, segurança e confiabilidade adequados para aplicações robustas, incluindo sistemas empresariais, tecnologias de servidor, aplicações móveis e soluções voltadas para big data ([CALÇA, 2022](#)).

Framework e Spring Boot

Segundo [Calça \(2022\)](#), um framework consiste em um conjunto estruturado de componentes reutilizáveis que oferece ao desenvolvedor uma base pré-configurada, reduzindo a necessidade de criação de código do zero. Entre os frameworks Java, destaca-se o Spring Boot, desenvolvido com o intuito de impulsionar o desenvolvimento de aplicações back-end ao simplificar configurações iniciais e disponibilizar módulos integrados. O Spring Boot também possui compatibilidade com o Spring Security, o que facilita a implementação de mecanismos de autenticação e controle de acesso.

React

De acordo com [Sousa \(2025\)](#), React é uma biblioteca JavaScript criada pelo Facebook, voltada para o desenvolvimento de interfaces web interativas. Sua arquitetura baseada em componentes reutilizáveis permite maior organização e reaproveitamento de código, reduzindo o tempo de desenvolvimento e promovendo maior consistência visual e funcional no front-end.

Ambiente de Desenvolvimento Integrado do Arduino

O Ambiente de Desenvolvimento Integrado (IDE) do Arduino é responsável por compilar e enviar instruções para a placa microcontrolada. A IDE traduz o código escrito em linguagem de alto nível para instruções compreendidas pelo microcontrolador, permitindo desde operações simples, como acionar um LED, até o controle de bombas e motores a partir de dados coletados por sensores ([SILVA; FERNANDES, 2025](#)).

A interface da IDE é intuitiva, possibilitando que usuários com diferentes níveis de experiência desenvolvam seus projetos. Além disso, oferece ferramentas de compilação, verificação e comunicação com a placa, tornando-se essencial para a integração entre o ambiente digital e o físico.

Banco de Dados

Segundo [Date \(2004\)](#), um banco de dados é um sistema destinado ao armazenamento, organização e recuperação estruturada de informações. Ele permite inserir, consultar, atualizar e excluir registros conforme a necessidade da aplicação. Em um sistema de automação, o banco de dados pode armazenar parâmetros coletados por sensores, registros de operações, dados históricos e configurações do sistema, possibilitando um acompanhamento preciso do funcionamento do ambiente automatizado.

Engenharia de Software

A engenharia de software busca desenvolver sistemas de alta qualidade de maneira eficiente e econômica, aplicando princípios de engenharia ao processo de criação, manutenção e documentação do software ([VASCONCELOS et al., 2006](#)). Esse processo envolve diferentes etapas do ciclo de vida do software, desde a concepção até a implantação, que incluem análise, projeto, implementação, testes e manutenção.

Os engenheiros de software devem empregar abordagens sistemáticas e metodologias adequadas ao problema, às restrições e aos recursos disponíveis, de modo a aumentar a probabilidade de sucesso do projeto e garantir que o software atenda às necessidades dos usuários dentro do prazo e orçamento estabelecidos.

RUP (Rational Unified Process)

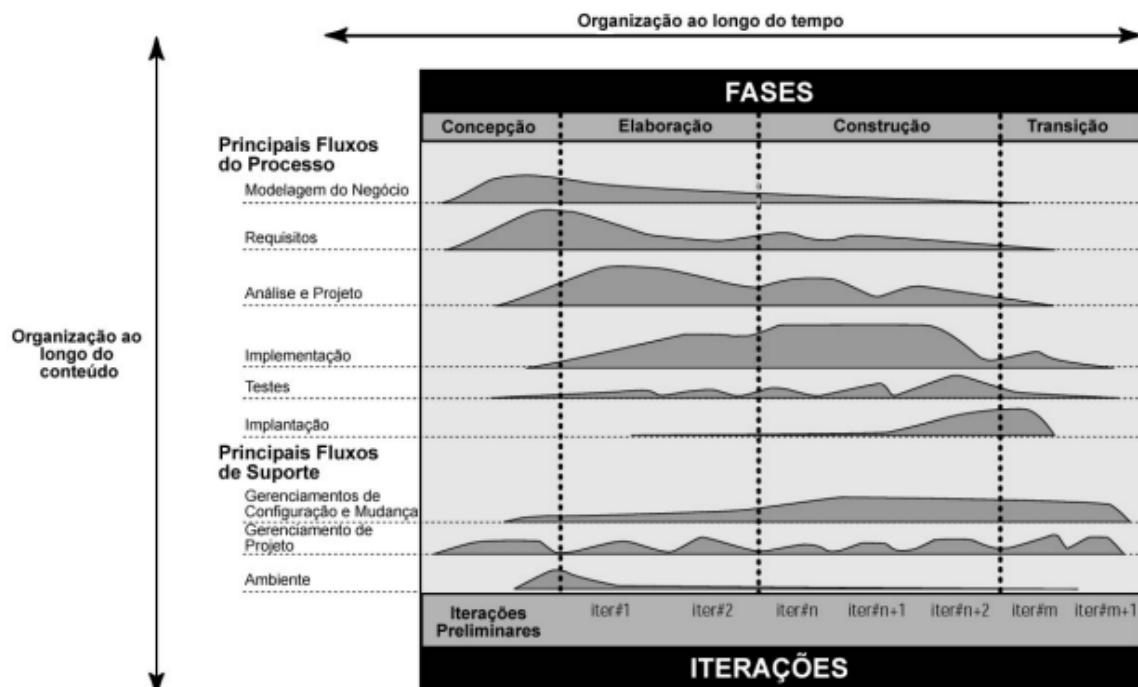
O Rational Unified Process (RUP), segundo [Piske \(2003\)](#), é uma metodologia estruturada de desenvolvimento de software com base em práticas recomendadas (*best practices*) e voltada à redução de riscos durante o projeto. O RUP caracteriza-se por ser iterativo e incremental: o sistema é desenvolvido em ciclos sucessivos, nos quais novas funcionalidades são incorporadas ao produto ([MOREIRA, 2025](#)).

De acordo com [Vasconcelos \(2018\)](#), o RUP é composto por quatro fases principais:

- **Concepção:** definição do problema, escopo e objetivos fundamentais;
- **Elaboração:** modelagem dos casos de uso e arquitetura inicial;
- **Construção:** desenvolvimento e integração do software;
- **Transição:** testes práticos, avaliação de desempenho e ajustes finais.

A figura 11 apresenta uma representação visual mais intuitiva das fases do RUP, constituída por um eixo horizontal que representa o tempo, sendo dividido nas diferentes fases do processo. O eixo vertical, por sua vez, representa os artefatos, que são documentos essenciais que auxiliam na estruturação e no acompanhamento do progresso do projeto ([ALLEBRANDT, 2004](#)).

Figura 11 – Fases do Rational Unified Process RUP.



Fonte: [Allebrandt \(2004\)](#).

A adoção do RUP justifica-se pela necessidade de gerenciar a complexidade inerente à integração entre componentes de hardware (IoT) e software web. A abordagem iterativa e a ênfase na mitigação de riscos mostram-se essenciais para assegurar que desafios técnicos, como a comunicação serial e a calibração de sensores, sejam validados ainda nas fases iniciais (Elaboração), evitando retrabalho crítico. Dessa forma, a estrutura de fases apresentada fundamenta a organização cronológica do desenvolvimento do protótipo.

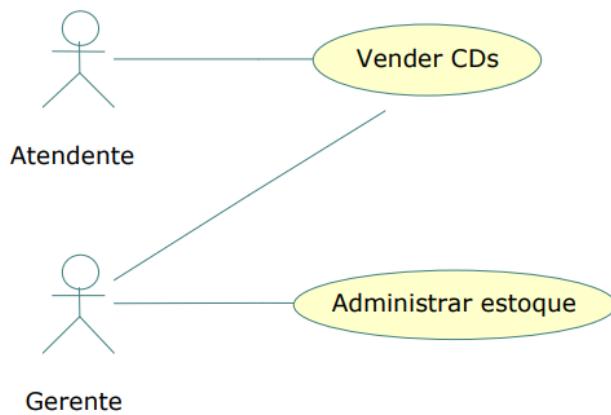
Linguagem UML

A Linguagem de Modelagem Unificada (UML) constitui um conjunto padronizado de notações gráficas destinado à representação visual de sistemas de software, sendo amplamente empregada para apoiar a especificação, a documentação e a compreensão tanto da estrutura quanto do comportamento dos sistemas (CASTRO; CRUZ; ODDONE, 2013). Sua utilização favorece a construção de uma linguagem comum entre analistas, desenvolvedores e demais partes interessadas, reduzindo ambiguidades na interpretação dos requisitos e das soluções propostas (BERNHARDT et al., 2023).

No contexto da Engenharia de Software, a UML desempenha papel fundamental nas fases iniciais do desenvolvimento, especialmente na análise e no projeto do sistema. Por meio de seus diferentes tipos de diagramas, torna-se possível representar desde aspectos funcionais, como as interações entre usuários e o sistema, até aspectos comportamentais e estruturais, contribuindo para a rastreabilidade entre requisitos, decisões arquiteturais e implementação.

Dentre os diagramas definidos pela UML, o Diagrama de Casos de Uso destaca-se por permitir a modelagem das funcionalidades do sistema sob a perspectiva do usuário, evidenciando os objetivos que este pretende alcançar ao interagir com a aplicação. Esse tipo de diagrama abstrai detalhes de implementação e concentra-se na relação entre atores e serviços oferecidos pelo sistema, sendo particularmente adequado para a validação de requisitos funcionais e para o alinhamento entre as expectativas dos usuários e as soluções técnicas propostas. A Figura 12 ilustra um exemplo de diagrama de casos de uso simplificado, conforme apresentado por (Universidade Estadual Paulista, 2025).

Figura 12 – Exemplo de Diagrama de Casos de Uso



Fonte: [Universidade Estadual Paulista \(2025\)](#).

De forma complementar, a UML também contempla diagramas voltados à representação do fluxo de atividades e da dinâmica de execução do sistema, os quais permitem compreender a lógica operacional associada às funcionalidades modeladas. Esses recursos tornam a UML particularmente adequada para projetos que envolvem sistemas interativos e automatizados, nos quais a clareza na definição de processos e decisões é essencial para garantir consistência entre análise, projeto e implementação.

Assim, a adoção da UML neste trabalho justifica-se por sua capacidade de fornecer uma visão estruturada e padronizada do sistema proposto, independentemente da tecnologia ou da metodologia de desenvolvimento empregada. O uso desses modelos contribui para a organização do processo de desenvolvimento e serve como base para a comparação com soluções existentes na literatura, aspecto aprofundado no capítulo dedicado aos trabalhos correlatos.

3 TRABALHOS CORRELATOS

A revisão bibliográfica permitiu identificar diferentes abordagens tecnológicas aplicadas à automação de piscinas e ambientes residenciais correlatos, variando desde soluções baseadas em acionamentos temporizados até sistemas fundamentados em controladores microprocessados e controladores lógicos programáveis. Esses trabalhos apresentam contribuições relevantes, porém adotam estratégias distintas quanto ao nível de automação, aos parâmetros monitorados e aos mecanismos de interação com o usuário. A seguir, são analisadas as principais propostas identificadas, com ênfase em seus escopos, limitações e diferenças em relação à solução desenvolvida neste trabalho.

Uma das abordagens iniciais encontradas é a de Campos (2014), que propôs um sistema amplo de automação residencial utilizando a plataforma Arduino. Nesse projeto, a piscina é tratada como um subsistema, cujo controle de filtragem é realizado exclusivamente por meio de agendamento temporal (*timer*), sem a leitura de parâmetros físico-químicos da água. Embora o trabalho apresente uma integração eficiente entre diferentes ambientes residenciais, observa-se a ausência de sensores voltados à análise da qualidade da água, o que limita a capacidade do sistema de atuar de forma inteligente e adaptativa no tratamento químico da piscina.

Com foco específico no controle químico da água, Brandão e Rocha (2018) desenvolveram um sistema de malha fechada utilizando o Arduino Uno para o controle de pH e desinfecção. O diferencial dessa proposta reside na utilização de um sensor de potencial de oxirredução (ORP) para estimar a concentração de cloro livre, associado ao sensor de pH, com atuação realizada por bombas dosadoras peristálticas. Apesar da robustez do controle químico, a interação com o usuário restringe-se a um display LCD local, inexistindo conectividade com a Internet ou recursos de monitoramento remoto, o que reduz a flexibilidade e a escalabilidade da solução.

Na mesma linha de automação voltada ao equilíbrio químico, Boeira (2019) apresentaram o protótipo denominado *Autopool*, desenvolvido com base no Arduino Mega. O sistema integra sensores de pH, nível e chuva, além de bombas adaptadas para a dosagem de produtos químicos. Um aspecto relevante desse trabalho é a inclusão de um mecanismo de segurança por meio de sensor de movimento, com o objetivo de prevenir acidentes. Entretanto, o sistema opera de forma isolada (*offline*), sendo configurado exclusivamente por meio de interface física local, e não contempla o monitoramento da turbidez da água.

Em uma proposta mais recente, Oliveira et al. (2023) desenvolveram um sistema de automação de área de lazer utilizando o microcontrolador ESP32, destacando-se pela conectividade Wi-Fi integrada e pelo dimensionamento da infraestrutura elétrica. O

sistema possibilita o acionamento remoto de bombas e iluminação, caracterizando-se como uma solução de controle digital à distância. Todavia, a ausência de sensores para monitoramento da qualidade da água faz com que as decisões sobre o momento adequado de filtragem dependam diretamente da intervenção do usuário, não configurando, portanto, um ciclo de automação autônomo.

Em contraste com os trabalhos analisados, a solução proposta neste TCC diferencia-se por implementar um ciclo de automação fechado (*closed-loop*) integrado aos princípios da Internet das Coisas (IoT). Enquanto propostas como as de Campos (2014) e Oliveira et al. (2023) concentram-se em agendamento temporal ou acionamento remoto, e os trabalhos de Brandão e Rocha (2018) e Boeira (2019) priorizam o controle químico local, o sistema aqui desenvolvido combina sensores de pH e turbidez para subsidiar decisões autônomas sobre o tratamento químico da água. Ademais, a disponibilização de dados históricos e a gestão por meio de uma interface web ampliam a capacidade de monitoramento e análise do sistema ao longo do tempo.

A Tabela 3 apresenta uma síntese comparativa das funcionalidades contempladas nos trabalhos correlatos e na proposta desenvolvida neste estudo, evidenciando os diferenciais da solução apresentada.

Tabela 3 – Comparativo de funcionalidades entre trabalhos correlatos e o sistema proposto

| Referência | Automação Química | Leitura de Turbidez | Interface Web/IoT | Controle Autônomo | Baixo Custo | Segurança (Alarme) |
|-------------------------|-------------------|---------------------|-------------------|-------------------|-------------|--------------------|
| Campos (2014) | | | ✓ | | ✓ | ✓ |
| Brandão e Rocha (2018) | ✓ | | | ✓ | ✓ | |
| Boeira (2019) | ✓ | | | ✓ | ✓ | ✓ |
| Sulimann e Haupt (2014) | ✓ | | | ✓ | | |
| Oliveira et al. (2023) | | | ✓ | | ✓ | |
| Este Trabalho | ✓ | ✓ | ✓ | ✓ | ✓ | |

Fonte: Autoria própria (2025).

A análise comparativa evidencia a existência de uma lacuna no desenvolvimento de sistemas de baixo custo capazes de realizar, de forma simultânea, o monitoramento analítico de pH e turbidez, com atuação autônoma e conectividade via Internet. Essa lacuna, identificada de maneira recorrente nos trabalhos analisados, é diretamente abordada pela proposta apresentada neste estudo, a qual busca integrar monitoramento, tomada de decisão automática e acesso remoto em uma solução unificada.

Diante desse cenário, a automação aplicada à manutenção de piscinas apresenta-se como um campo em expansão, ainda carente de abordagens que conciliem baixo custo,

autonomia operacional e integração com tecnologias de Internet das Coisas. Assim, o capítulo seguinte descreve o desenvolvimento da solução proposta, detalhando a metodologia adotada, a modelagem do sistema, as decisões arquiteturais e as etapas de implementação que materializam a proposta delineada a partir das lacunas identificadas nos trabalhos correlatos.

4 DESENVOLVIMENTO

4.1 TIPO DE PESQUISA E ETAPAS DE CONSTRUÇÃO

A pesquisa caracteriza-se como aplicada, de abordagem mista e de natureza exploratória e experimental. Inicialmente, foi realizado um estudo teórico sobre automação residencial, sensores, atuadores e controladores associados à Internet das Coisas (IoT), fundamentado nos autores apresentados no [Capítulo 2](#). Essa etapa possibilitou a compreensão dos processos físico-químicos envolvidos no tratamento da água, das normas sanitárias aplicáveis e dos requisitos técnicos necessários para a integração entre hardware, software e dispositivos embarcados.

O levantamento bibliográfico incluiu artigos científicos, manuais técnicos e normas da Associação Brasileira de Normas Técnicas (ABNT), assegurando base conceitual e técnica suficiente para orientar as decisões posteriores de arquitetura, modelagem e implementação do sistema. Os critérios adotados para a seleção das fontes contemplaram a atualidade das informações, priorizando publicações dos últimos dez anos para temas relacionados à tecnologia, automação residencial e Internet das Coisas (IoT); a relevância técnica, com a utilização de manuais de fabricantes e guias especializados para fundamentar os processos físico-químicos de tratamento da água; e a robustez teórica, com a seleção de autores consolidados e trabalhos acadêmicos reconhecidos nas áreas de Engenharia de Software e arquitetura de sistemas.

Essa etapa inicial permitiu mapear os requisitos técnicos necessários à integração entre hardware, software e dispositivos embarcados, fornecendo subsídios consistentes para orientar as decisões de projeto adotadas ao longo do desenvolvimento do sistema.

4.2 PROCESSO DE DESENVOLVIMENTO DE SOFTWARE

Neste projeto, adotou-se o *Rational Unified Process* (RUP) como metodologia de desenvolvimento, em razão de sua estrutura iterativa e incremental, do foco na mitigação de riscos e da ênfase na modelagem e na documentação do software. Conforme apresentado na fundamentação teórica, o RUP organiza o processo de engenharia de software em quatro fases principais: Concepção, Elaboração, Construção e Transição. Cada fase possui objetivos específicos que orientam a evolução do sistema, assegurando a rastreabilidade entre requisitos, arquitetura e implementação.

Considerando que o sistema desenvolvido envolve automação e dispositivos IoT, o ciclo de vida do desenvolvimento de hardware foi adaptado às fases do RUP. Nesse contexto, a definição e a seleção dos componentes eletrônicos ocorreram durante a fase de

Concepção, enquanto a montagem física e a calibração dos sensores foram realizadas de forma integrada à fase de Construção, em paralelo à implementação do software.

A adoção dessa metodologia estabelece um fluxo contínuo de desenvolvimento, no qual cada artefato produzido em uma fase serve como base para a validação da fase subsequente. Assim, os requisitos levantados na fase de Concepção subsidiam a modelagem arquitetural realizada na fase de Elaboração, cujos diagramas orientam a implementação na fase de Construção. Por fim, a conformidade do sistema desenvolvido é verificada durante a fase de Transição, por meio de testes funcionais e ajustes finais.

As subseções seguintes descrevem detalhadamente cada uma dessas fases, destacando as atividades executadas e sua relação com os artefatos produzidos ao longo do processo.

4.3 FASE DE CONCEPÇÃO (*INCEPTION*)

Na fase de Concepção foram definidos o problema de pesquisa, o escopo inicial do projeto e os requisitos fundamentais do sistema. O problema identificado refere-se às dificuldades enfrentadas pelos usuários no processo manual de manutenção de piscinas, decorrentes da necessidade de medições frequentes, cálculos físico-químicos, além do risco de desperdício de produtos ocasionado pela aplicação imprecisa das dosagens recomendadas.

Com o objetivo de mitigar essas limitações, o sistema foi concebido para automatizar a leitura contínua dos principais parâmetros físico-químicos da água, reduzir a subjetividade na avaliação da qualidade, atuar de forma autônoma na reposição de água com base no sensor de nível e disponibilizar interfaces remotas para monitoramento e acompanhamento do processo de manutenção.

O levantamento de requisitos resultou na definição dos requisitos funcionais (RF) e não funcionais (RNF), apresentados nas Tabelas 4 e 5, respectivamente.

Tabela 4 – Requisitos Funcionais do Sistema de Automação de Piscinas

| Código | Descrição |
|-------------|--|
| RF01 | O sistema deve monitorar automaticamente os níveis de pH, turbidez, temperatura e nível da água. |
| RF02 | O sistema deve acionar automaticamente a bomba de água conforme os parâmetros coletado pelo sensor de nível. |
| RF03 | O sistema deve disponibilizar uma interface web para visualização dos parâmetros monitorados. |
| RF04 | O sistema deve permitir o cadastro e o armazenamento dos dados coletados no banco de dados. |
| RF05 | O sistema deve emitir alertas quando algum parâmetro ultrapassar os limites estabelecidos. |
| RF06 | O sistema deve gerar gráficos com base nos dados históricos dos parâmetros monitorados. |

Fonte: Autoria própria (2025).

Para a validação dos requisitos definidos, foram estabelecidos critérios quantitativos de aceitação, considerando o desempenho esperado para um protótipo IoT. A precisão dos sensores (RF01) foi definida com margem de erro inferior a $\pm 5\%$ em relação a medições de referência. Quanto ao desempenho temporal, estabeleceu-se que o tempo de latência total, compreendido entre a aquisição do dado físico pelo microcontrolador e sua visualização na interface web, não deve exceder oito segundos, caracterizando uma monitoração em tempo quase real. No que se refere à comunicação (RNF03), o sistema é considerado estável quando as requisições HTTP apresentam taxa de sucesso superior a 95% em operação contínua, com tempo de resposta do servidor inferior a seis segundos por requisição. O requisito de automação (RF02) é considerado atendido quando o acionamento do atuador ocorrer em até três segundos após a detecção da falta de água.

Tabela 5 – Requisitos Não Funcionais do Sistema de Automação de Piscinas

| Código | Descrição |
|--------------|--|
| RNF01 | O sistema deve utilizar o banco de dados PostgreSQL para armazenamento das informações. |
| RNF02 | A interface web deve ser responsiva e acessível em dispositivos móveis e desktops. |
| RNF03 | A comunicação entre o gateway (Raspberry Pi) e o servidor deve ocorrer utilizando o protocolo HTTP. |
| RNF04 | O sistema deve ser desenvolvido utilizando o framework Spring Boot no back-end e React no front-end. |

Fonte: Autoria própria (2025).

Para atender ao escopo definido, adotou-se uma arquitetura distribuída de hardware e software. Na camada física, selecionou-se o Arduino Uno para a leitura dos sensores e o acionamento dos atuadores, enquanto o Raspberry Pi foi designado para a comunicação entre o microcontrolador, o servidor e a aplicação web. Na camada de software, definiu-se o uso do framework Spring Boot no back-end, em virtude de sua robustez na implementação de APIs REST, React para o desenvolvimento da interface web responsiva e PostgreSQL para a persistência e o gerenciamento dos dados coletados.

4.4 FASE DE ELABORAÇÃO (*ELABORATION*)

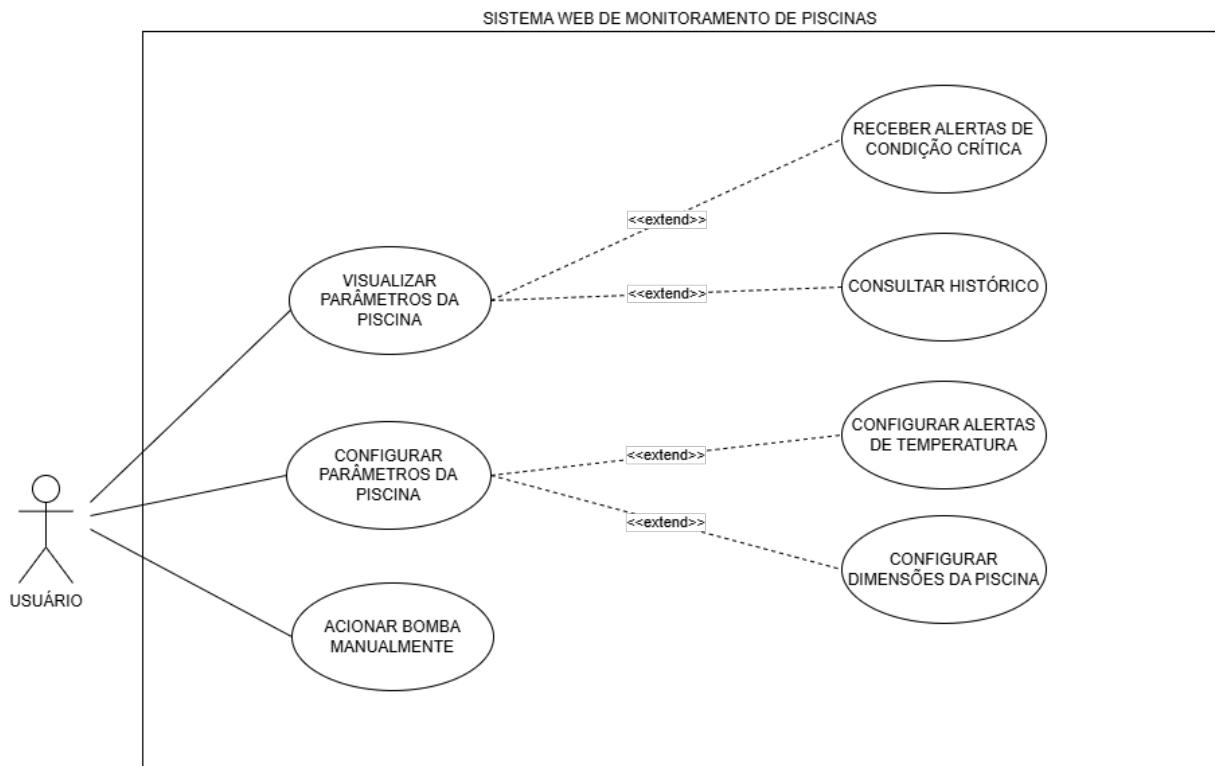
Com o objetivo de mitigar riscos técnicos e validar as principais decisões arquiteturais do sistema, foram desenvolvidos protótipos funcionais parciais, conforme recomendado pelo Rational Unified Process (RUP). Nessa etapa, realizaram-se testes preliminares de integração entre os sensores, o microcontrolador Arduino e o Raspberry Pi, com foco na verificação da estabilidade da comunicação serial, da latência no envio de dados e da viabilidade da arquitetura proposta. Esses experimentos permitiram confirmar que a infraestrutura de hardware selecionada atende aos requisitos de desempenho estabelecidos na fase de Concepção.

Simultaneamente, foram definidos os principais artefatos estruturais do sistema, incluindo a modelagem dos casos de uso, a arquitetura de comunicação entre os dispositivos físicos e a especificação dos componentes embarcados que compõem o protótipo. Dessa forma, a fase de Elaboração estabelece a ligação entre os requisitos identificados na Concepção e a implementação conduzida na fase de Construção, assegurando que o sistema seja desenvolvido com base em uma arquitetura previamente validada e devidamente documentada.

Modelagem de Caso de Uso

A Figura 13 apresenta o Diagrama de Casos de Uso do sistema, detalhando as interações permitidas ao usuário, como a visualização de parâmetros e a configuração de alertas, bem como a integração dos dispositivos físicos com o back-end.

Figura 13 – Diagrama de Caso de Uso do Sistema



Fonte: Autoria própria (2025).

Arquitetura Geral do Sistema

O sistema foi concebido a partir de uma arquitetura distribuída, composta por sensores, um microcontrolador Arduino, um microcomputador Raspberry Pi, um back-end desenvolvido com o framework Spring Boot e uma interface web implementada em React. O fluxo principal de funcionamento do sistema ocorre conforme as seguintes etapas:

1. Coleta dos dados pelos sensores conectados ao Arduino;
2. Transmissão das leituras do Arduino para o Raspberry Pi;
3. Envio periódico das informações ao servidor por meio de requisições HTTP;
4. Armazenamento dos dados no banco de dados PostgreSQL;
5. Apresentação dos parâmetros ao usuário por meio da interface web.

Essa arquitetura favorece a escalabilidade e o desacoplamento entre as camadas do sistema, em conformidade com os princípios discutidos no [Capítulo 2](#). Entretanto, a distribuição dos componentes físicos introduz riscos operacionais relevantes, como a dependência da integridade funcional de múltiplos dispositivos. A falha de um único nó, seja um sensor ou um controlador, pode comprometer temporariamente o fluxo de dados.

Além disso, a comunicação baseada em protocolos HTTP está sujeita a latências e instabilidades de rede, especialmente em infraestruturas domésticas, o que exige a implementação de mecanismos de tratamento de exceções, tentativas de reconexão automática e tolerância a falhas. Tais estratégias são fundamentais para garantir que o sistema seja capaz de recuperar sua operabilidade sem a necessidade de intervenção humana após falhas momentâneas de conectividade.

Componentes Utilizados no Sistema

A seguir, são apresentados os principais componentes físicos selecionados para o desenvolvimento do protótipo, incluindo sensores, atuadores e dispositivos de controle.

Sensores de Monitoramento

Para a medição da temperatura da água, optou-se pelo sensor NTC 10K (Modelo MF58), ilustrado na Figura [22](#) (Apêndice A). O componente foi integrado ao circuito por meio de um divisor de tensão e conectado a uma entrada analógica do Arduino para leitura dos dados térmicos.

O monitoramento do nível da água é realizado pelo sensor LC26M-40, apresentado na Figura [23](#) (Apêndice A). Fabricado em polipropileno, o dispositivo fornece um sinal digital (nível lógico alto ou baixo) diretamente ao microcontrolador, permitindo a identificação imediata de níveis críticos no reservatório.

A qualidade química da água é aferida por um Sensor de pH (Figura [24](#), Apêndice A) e um Sensor de Turbidez Modelo ST100 (Figura [25](#), Apêndice A). O sensor de pH monitora a acidez e a alcalinidade, enquanto o sensor de turbidez identifica partículas em suspensão. Ambos geram sinais analógicos processados pelo Arduino e enviados ao servidor para análise e tomada de decisão.

Atuadores

Para o controle do fluxo de água, foi utilizado uma bomba submersa modelo JT100 (3V a 5V), conforme detalhado na Figura [26](#) (Apêndice A). Com vazão entre 1000 e 1500 ml/min e elevação de até 1 metro, este equipamento é responsável pelo controle de nível de água da piscina, sendo acionado quando o sensor de nível indicar leituras abaixo do limite configurado.

Controladores e Gateway

O controle direto de hardware é exercido pela placa Arduino Uno R3 (Figura 27, Apêndice A). Sua escolha deve-se à robustez nas leituras analógicas e à vasta disponibilidade de bibliotecas para controle dos atuadores em tempo real.

Como elemento intermediário (*Gateway*), utiliza-se o Raspberry Pi 3 Model B, exibido na Figura 28 (Apêndice A). Equipado com processador, ele gerencia a comunicação Serial com o Arduino e executa os *scripts* responsáveis pelo envio dos dados à API, isolando a lógica de rede da lógica de controle físico.

Tabela 6 – Resumo das Especificações Técnicas dos Componentes de Hardware

| Componente | Função Principal | Tensão | Comunicação / Sinal |
|-----------------------------|---|--------|--------------------------------------|
| Raspberry Pi 3 B | Gateway de comunicação e envio de dados ao servidor | 5 V | Serial (USB) / Wi-Fi (HTTP) |
| Arduino Uno R3 | Leitura de sensores e controle de atuadores | 5 V | Serial / I/O digital e analógico |
| Sensor de Temperatura (NTC) | Monitoramento térmico da água | 5 V | Analógico (divisor de tensão) |
| Sensor de Nível | Detecção de nível crítico do reservatório | 5 V | Digital (On/Off) |
| Sensor de pH | Medição da acidez e alcalinidade da água | 5 V | Analógico |
| Sensor de Turbidez | Avaliação da transparência da água | 5 V | Analógico |
| Bomba Submersa | Circulação e filtragem da água | 3–6 V | Acionamento via relé (sinal digital) |

Fonte: Autoria própria (2025).

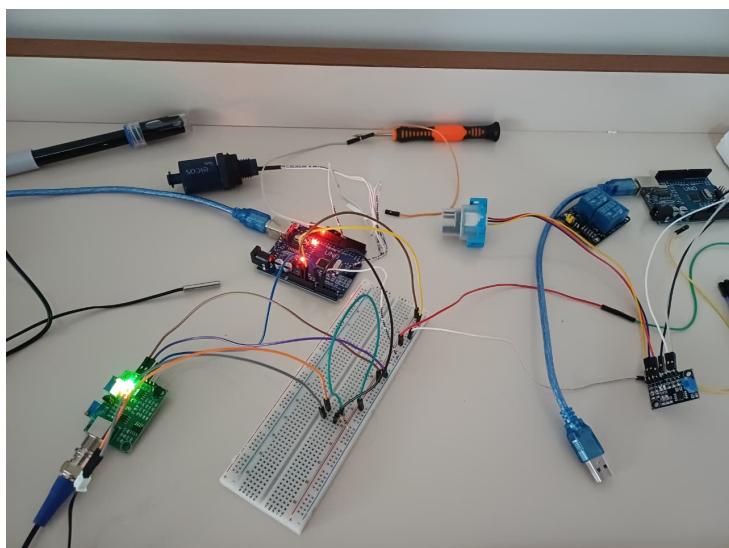
4.5 FASE DE CONSTRUÇÃO (*CONSTRUCTION*)

Em conformidade com a metodologia RUP, a fase de Construção foi conduzida por meio de ciclos iterativos e incrementais, nos quais funcionalidades do sistema foram implementadas e validadas progressivamente. O desenvolvimento iniciou-se pela camada física, com a implementação e a validação individual dos sensores e atuadores conectados ao microcontrolador. Após a estabilização da aquisição dos dados, procedeu-se à implementação da camada lógica, englobando o back-end, o banco de dados e o front-end.

A etapa final concentrou-se na integração completa entre hardware e software, validando o fluxo de transmissão das informações, desde a coleta no ambiente físico até a apresentação na interface web. O desenvolvimento do sistema foi estruturado em duas camadas principais: a camada embarcada, responsável pela leitura, pré-processamento e envio dos dados coletados; e a camada de aplicação, encarregada de receber, armazenar e disponibilizar essas informações aos usuários.

Na camada embarcada, o Arduino foi programado por meio de sua IDE nativa para realizar a leitura dos sensores analógicos e digitais, bem como o acionamento condicional dos atuadores. A montagem física e a organização dos componentes durante os testes de bancada podem ser visualizadas na Figura 14.

Figura 14 – Testes em bancada

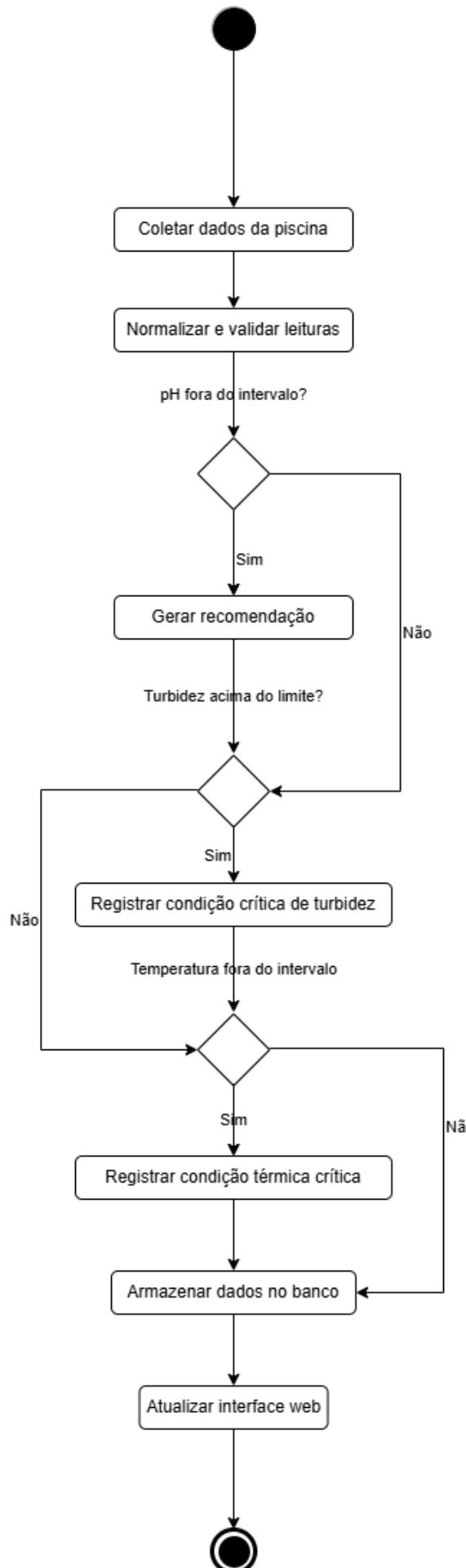


Fonte: Autoria própria (2025).

Conforme apresentado no Apêndice B, a Seção B.1 ilustra a rotina de coleta dos parâmetros da água e a lógica empregada para o controle dos dispositivos físicos.

Com o intuito de garantir a confiabilidade das informações coletadas e evitar o armazenamento de dados incorretos, foi desenvolvido um fluxo lógico de validação e normalização antes da persistência no banco de dados. A Figura 15 demonstra o algoritmo de decisão executado pelo sistema a cada ciclo de leitura.

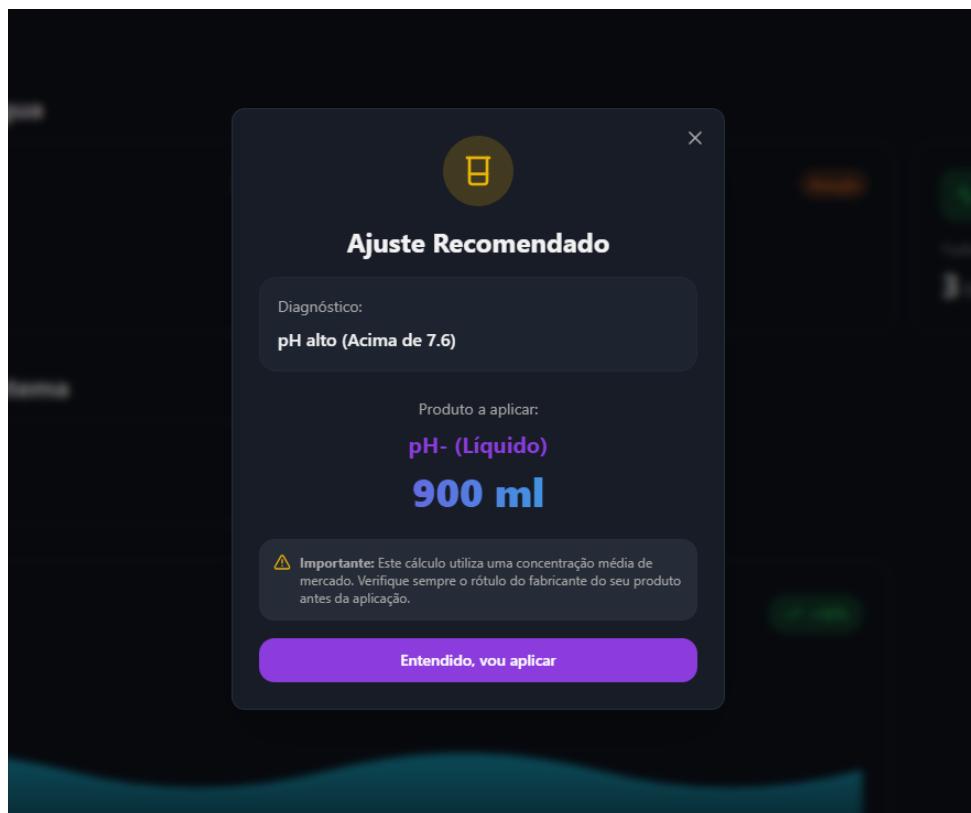
Figura 15 – Fluxograma de validação de dados e tomada de decisão do sistema



O Raspberry Pi desempenhou a função de *gateway* de comunicação do sistema. Um *script* desenvolvido em Python foi responsável por estabelecer a comunicação serial com o microcontrolador, interpretar os dados recebidos e encaminhá-los ao servidor por meio de requisições HTTP do tipo POST. A lógica de implementação desse *gateway*, incluindo os mecanismos de tratamento de falhas de comunicação e a serialização dos dados em formato JSON, encontra-se detalhada no Apêndice B, Seção B.2.

Na camada de aplicação, o *back-end*, desenvolvido utilizando o *framework* Spring Boot, centraliza a lógica de persistência e fornece os dados base para o sistema. Uma funcionalidade fundamental implementada foi a personalização do monitoramento: o sistema gerencia um perfil de usuário com as dimensões físicas da piscina e os limites de alerta. O *back-end* utiliza esses parâmetros para calcular dinamicamente o volume exato (em litros) a cada requisição. A partir desse dado, a interface *web* executa o algoritmo de recomendação de dosagem química, sugerindo correções precisas ao usuário, como evidenciado na Figura 16. O *controller* responsável pelo cálculo de volume e persistência está apresentado no Apêndice B, Seção B.3, enquanto a lógica de recomendação e alertas é detalhada na Seção B.5. Essa arquitetura assegura a integridade, a consistência e a disponibilidade do histórico de monitoramento.

Figura 16 – Modal de recomendação de dosagem



Fonte: Autoria própria (2025).

Por fim, a interface do usuário, desenvolvida em React, consome os dados fornecidos

pela API para apresentar o estado atual da piscina em tempo quase real. A implementação da lógica de consumo da API e o gerenciamento de estado para visualização dos parâmetros e acionamento manual dos atuadores estão descritos no Apêndice B, Seção B.4.

Além do monitoramento passivo, a interface web foi dotada de lógica reativa para auxílio à tomada de decisão. Ao detectar que parâmetros críticos (como pH ou turbidez) estão fora dos limites estabelecidos, o sistema habilita uma funcionalidade interativa de cálculo de correção. Por meio dela, o usuário pode visualizar instantaneamente a sugestão de dosagem exata dos produtos químicos necessários (ex: quantidade de clarificante ou redutor de pH), calculada com base no volume específico da piscina parametrizado anteriormente.

4.6 FASE DE TRANSIÇÃO (*TRANSITION*)

Na fase de Transição, o sistema desenvolvido foi submetido a testes práticos com o objetivo de validar seu desempenho, estabilidade e conformidade com os requisitos estabelecidos nas fases anteriores. Foram realizados testes unitários nos sensores, abrangendo a calibração do sensor NTC de temperatura e pH, a verificação da estabilidade do sensor de nível e simulações de condições críticas de funcionamento das bombas, de modo a prevenir situações de *dry-run*. Paralelamente, testes de integração avaliaram o fluxo completo dos dados, desde a aquisição no ambiente físico até a apresentação das informações na interface web, conforme detalhado na Tabela 7.

Tabela 7 – Resumo dos Testes de Validação Sistêmica

| Teste Realizado | Objetivo do Teste | Critério/Resultado |
|--------------------------|--|---|
| Calibração do Sensor NTC | Verificar a precisão térmica por comparação com termômetro de referência | Margem de erro aceitável ($< \pm 2^\circ\text{C}$) |
| Calibração do pH | Ajuste de ganho via <i>trimpot</i> do módulo comparado com valor de referência | Erro absoluto médio $< 0,3 \text{ pH}$ |
| Calibração de Turbidez | Definição dos limiares de tensão para água limpa e suja (sensor ST100) | Diferenciação clara (Limpa $> 4\text{V}$ / Suja $< 1\text{V}$) |
| Estabilidade do Nível | Confirmar o acionamento da boia sem oscilações espúrias | <i>Debounce</i> lógico funcional |
| Integração de Dados | Validar o fluxo completo Arduino → Raspberry Pi → Banco de Dados | Persistência verificada no PostgreSQL |
| Latência da Interface | Medir o tempo entre a leitura do sensor e a atualização na interface web | Atualização em tempo quase real ($< 5\text{s}$) |

Fonte: Autoria própria (2025).

Adicionalmente, foram realizados testes de responsividade da interface web e de confiabilidade da comunicação entre o servidor e os dispositivos embarcados. Os ajustes finais envolveram a correção de algoritmos de conversão dos sensores, a adequação de temporizações de leitura e transmissão, bem como melhorias na visualização gráfica dos dados apresentados ao usuário.

Embora os testes tenham validado a funcionalidade do sistema, observaram-se limitações decorrentes da escala reduzida do protótipo. A discussão detalhada sobre esses desafios e o desempenho quantitativo do sistema será apresentada a seguir.

5 RESULTADOS E DISCUSSÃO

Este capítulo apresenta e discute os principais resultados obtidos ao longo do desenvolvimento do sistema automatizado de limpeza e manutenção de piscinas. São analisados o desempenho operacional da solução proposta, os desafios técnicos enfrentados durante a implementação e as estratégias adotadas para mitigá-los. A discussão contempla a integração do protótipo físico, a usabilidade da interface de controle desenvolvida e a confiabilidade dos dados coletados, destacando os avanços alcançados em relação aos métodos manuais tradicionais, bem como as limitações observadas no contexto experimental.

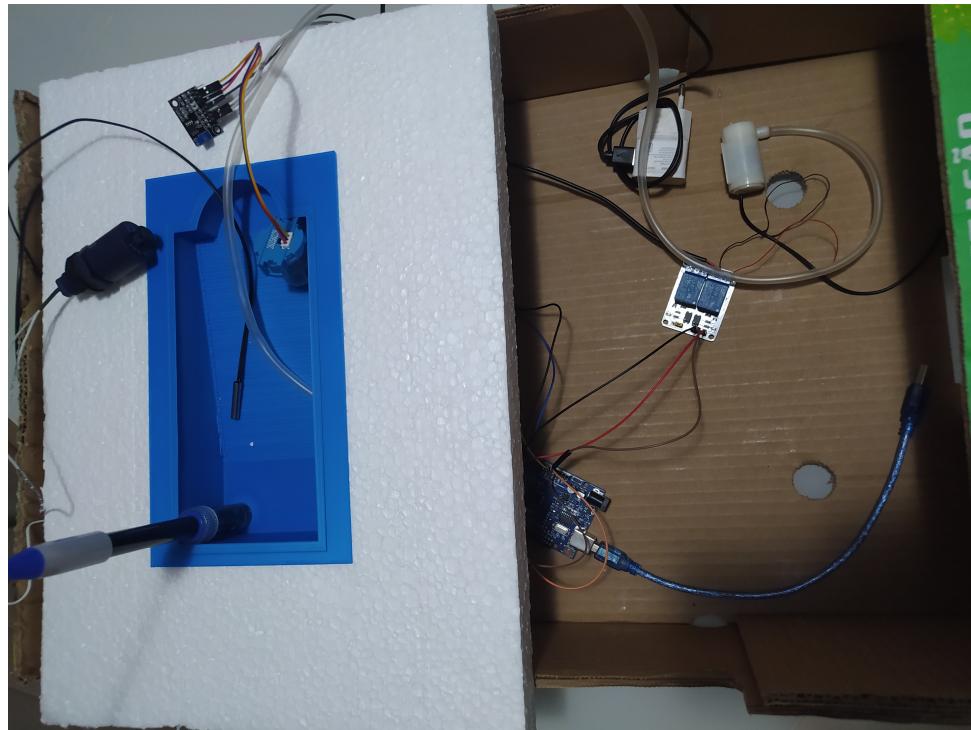
5.1 INTEGRAÇÃO DO PROTÓTIPO FÍSICO

O desenvolvimento do protótipo físico possibilitou a consolidação da arquitetura definida na fase de Elaboração, permitindo validar, de forma prática, a integração entre sensores, atuadores e a central de controle. O conjunto foi acondicionado em uma caixa de papelão coberta por uma placa de isopor com o objetivo de simular um ambiente real de operação e prover proteção básica aos componentes eletrônicos contra respingos e umidade, características inerentes ao contexto de uso em piscinas.

A disposição dos sensores de pH, turbidez, temperatura e nível no reservatório de testes mostrou-se adequada para a realização de leituras contínuas, contribuindo para a obtenção de dados consistentes ao longo dos experimentos. Essa configuração evidenciou boa confiabilidade na coleta das variáveis monitoradas, aspecto essencial para o funcionamento correto do ciclo de automação proposto.

A Figura 17 apresenta uma visão geral do protótipo montado, destacando a interligação entre a central de controle e os dispositivos responsáveis pela medição e atuação no sistema.

Figura 17 – Protótipo final do sistema de automação montado.

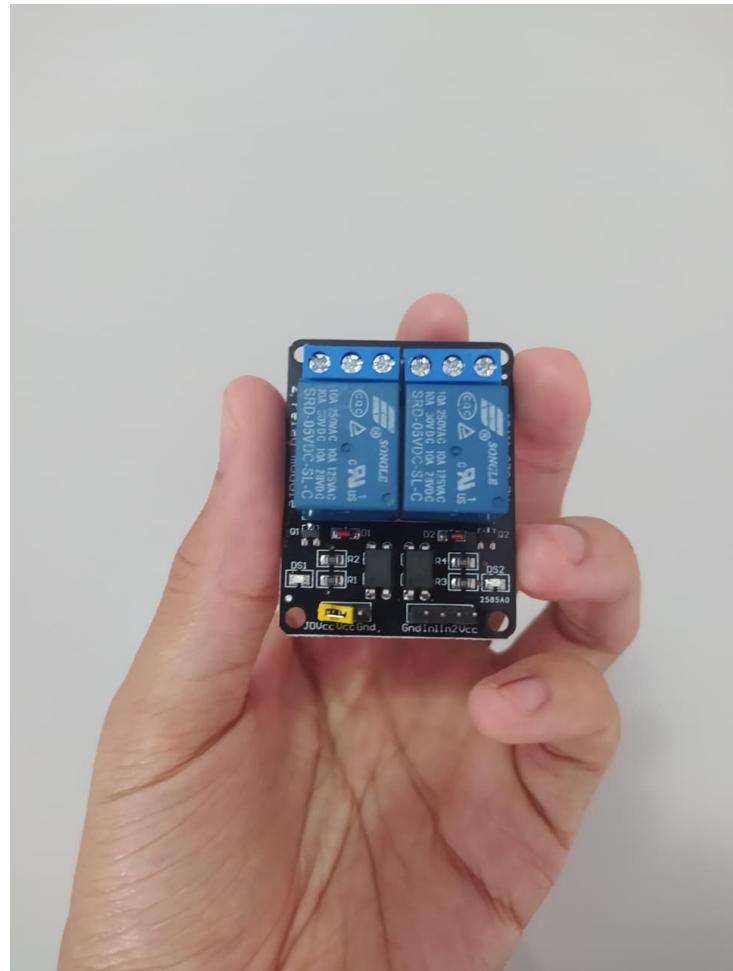


Fonte: Autoria própria (2025).

Durante os testes de integração, o conjunto de atuadores apresentou comportamento compatível com o esperado, com o acionamento da bomba d'água por meio do relé ocorrendo de forma quase imediata. Entretanto, nas etapas iniciais de validação, foi identificado um problema relacionado à estabilidade do sistema: o acionamento da bomba gerava ruídos eletromagnéticos e picos de tensão que interferiam no funcionamento do microcontrolador, comprometendo tanto a estabilidade das leituras dos sensores quanto a operação contínua do atuador.

Para mitigar esse problema, optou-se pela substituição do módulo de acionamento simples por um módulo relé com isolamento por optoacoplador. Essa solução proporcionou o isolamento galvânico entre o circuito de potência e o circuito lógico do Arduino, eliminando as interferências previamente observadas e aumentando a robustez do sistema. A Figura 18 apresenta o componente adotado após essa modificação.

Figura 18 – Relé com optoacoplador.

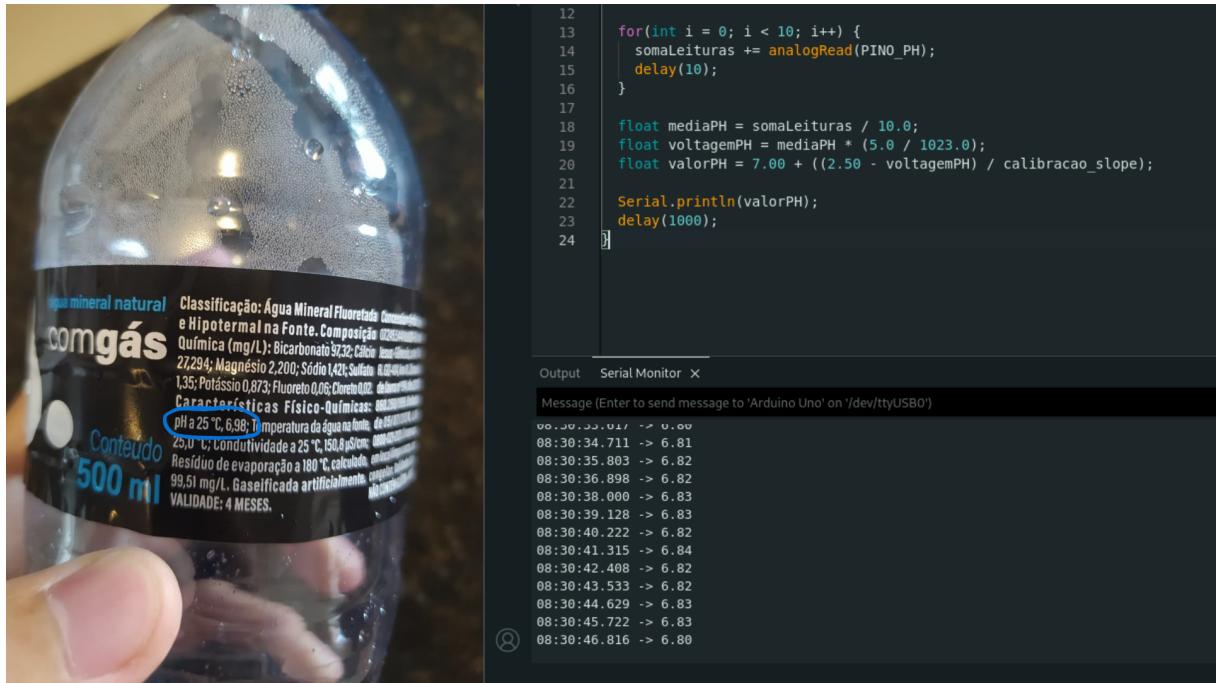


Fonte: Autoria própria (2025).

Além das questões relacionadas ao acionamento dos atuadores, foram identificadas limitações associadas à calibração inicial de alguns sensores. O sensor de pH, por exemplo, apresentou descalibração de fábrica, exigindo ajuste manual no módulo ao qual estava acoplado. A calibração foi realizada em nível de *hardware*, por meio do ajuste do potenciômetro (*trimpot*) de *offset* da placa condicionadora, alinhando a tensão de saída ao valor de referência esperado.

O procedimento de calibração foi validado utilizando-se uma garrafa de água mineral lacrada, cujo pH declarado é de 6,98. Após o ajuste, o sensor apresentou leituras variando entre 6,78 e 6,80, valor considerado satisfatório para o contexto experimental, conforme ilustrado na Figura 19.

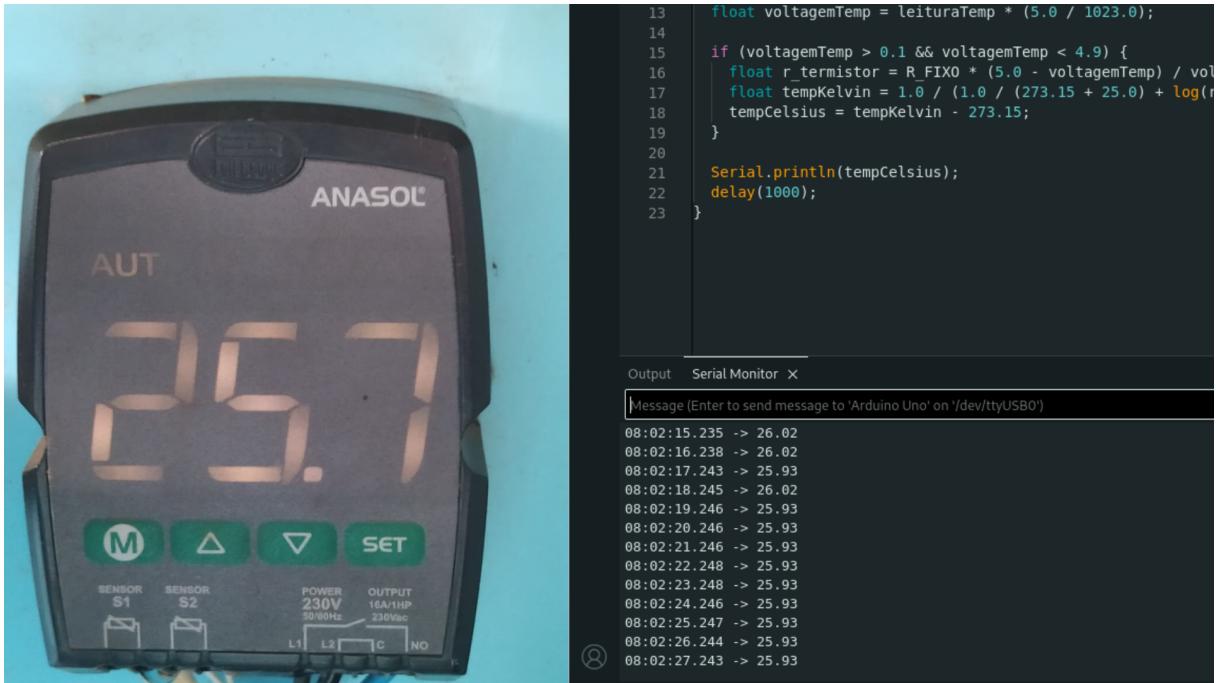
Figura 19 – Validação do sensor de pH.



Fonte: Autoria própria (2025).

De forma semelhante, o sensor de temperatura apresentou desvios iniciais em relação ao valor real da água. A correção foi realizada via *software*, por meio da aplicação de um fator de ajuste na equação do divisor de tensão, compensando variações na resistência nominal do termistor do tipo NTC. A validação do ajuste foi conduzida com base na comparação entre a leitura fornecida por um termômetro de piscina e os valores obtidos pelo sensor, registrando-se 25,7 °C no instrumento de referência e 25,93 °C no sistema desenvolvido. Essa proximidade entre os valores confirma a adequação do processo de calibração, conforme apresentado na Figura 20.

Figura 20 – Validação do sensor de temperatura.



Fonte: Autoria própria (2025).

5.2 INTERFACE DE CONTROLE E MONITORAMENTO

A interface de controle desenvolvida para o sistema, implementada no *front-end* com a biblioteca React, foi projetada com o objetivo de centralizar informações técnicas complexas em um *dashboard* visualmente intuitivo. Essa abordagem busca abstrair a complexidade inerente aos sensores e ao processamento dos dados, permitindo que o usuário compreenda rapidamente o estado da piscina sem a necessidade de conhecimentos técnicos avançados.

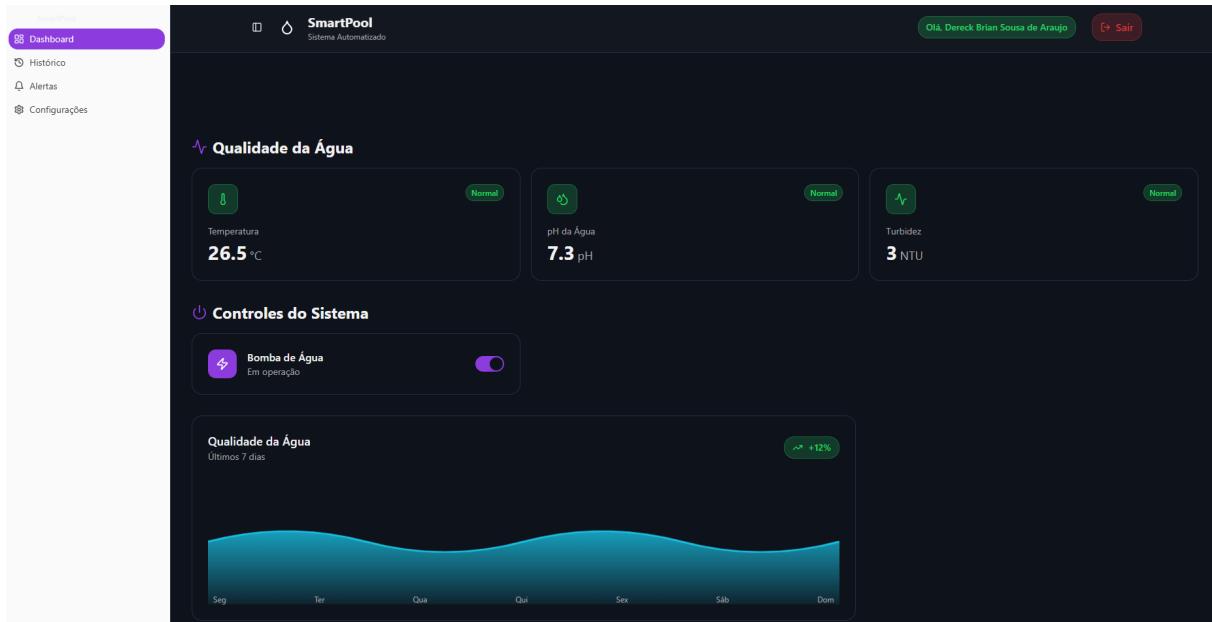
Diferentemente de soluções que exigem interpretação direta de valores brutos ou interação por meio de *displays LCD* acoplados ao hardware, a interface web apresenta os principais indicadores de qualidade da água de forma gráfica e organizada. Ao acessar o sistema, o usuário visualiza, em tempo real, os cartões de status referentes aos parâmetros monitorados, como pH, temperatura, turbidez e estado da bomba.

A comunicação entre o *front-end* e o *back-end*, desenvolvido em Spring Boot, garante a atualização periódica dos dados exibidos, permitindo o acompanhamento contínuo da evolução do tratamento da água. Além do monitoramento passivo, a interface oferece recursos de controle ativo, possibilitando ao usuário ligar ou desligar manualmente o sistema de filtragem. Essa funcionalidade permite a sobreposição temporária da lógica de automação em situações específicas, como manutenções corretivas ou testes operacionais.

A Figura 21 apresenta a tela principal do sistema, evidenciando a disposição dos

indicadores e o foco na clareza das informações exibidas.

Figura 21 – Interface Web: Dashboard de monitoramento em tempo real.



Fonte: Autoria própria (2025).

5.3 ANÁLISE DE DESEMPENHO E DISCUSSÃO

Os testes práticos realizados no cenário de validação demonstraram que o sistema é capaz de executar de forma consistente o ciclo completo de automação proposto, abrangendo as etapas de leitura dos sensores, processamento dos dados, transmissão das informações e atuação sobre os dispositivos físicos. Essa integração confirmou a viabilidade da arquitetura adotada para aplicações de automação residencial em piscinas.

No que se refere à precisão das medições, o sensor de temperatura NTC 10K apresentou elevada estabilidade, com variações consideradas desprezíveis quando comparadas às leituras obtidas por meio de um termômetro de referência. Esse comportamento evidencia a adequação do componente para monitoramento térmico contínuo em ambientes domésticos. Em contrapartida, os sensores analógicos voltados à avaliação da qualidade da água demandaram um tratamento mais cuidadoso, tanto em nível de *hardware* quanto de *software*.

Durante os testes iniciais, observou-se que, na ausência de técnicas de filtragem digital, as leituras de pH e turbidez apresentavam oscilações momentâneas decorrentes de ruídos elétricos e variações ambientais. Essas flutuações poderiam ocasionar acionamentos

indevidos da bomba, comprometendo a eficiência do sistema e aumentando o desgaste dos componentes. Para mitigar esse comportamento, foram implementados filtros de média móvel no *firmware* do Arduino, estratégia que reduziu significativamente o ruído nas leituras e proporcionou maior estabilidade ao processo decisório. Após essa intervenção, o sistema apresentou desempenho compatível com os requisitos de uso doméstico, mantendo respostas coerentes com as condições reais da água.

Outro aspecto relevante avaliado foi a latência de comunicação, definida como um dos requisitos não funcionais críticos do sistema. Os experimentos indicaram que o tempo médio entre a detecção de um evento físico (como a identificação de nível inadequado de água) e a atualização correspondente na interface web permaneceu, na maioria dos testes, abaixo de cinco segundos. Esse resultado valida a eficiência da arquitetura distribuída baseada em API REST e demonstra que a solução atende às demandas típicas de aplicações IoT residenciais, nas quais a resposta em tempo quase real é suficiente para garantir segurança e confiabilidade operacional.

Apesar dos resultados positivos, algumas limitações foram identificadas. Destacam-se a sensibilidade dos sensores analógicos de baixo custo, mais suscetíveis a ruídos e variações ambientais, e a dependência da estabilidade da rede Wi-Fi local, fator que pode impactar a atualização contínua da interface em cenários com conectividade limitada. Tais restrições não comprometem a funcionalidade do protótipo, mas indicam a necessidade de aprimoramentos para aplicações em maior escala ou ambientes com exigências mais rigorosas. Esses aspectos orientam as propostas de evolução do sistema, discutidas na seção de trabalhos futuros.

5.4 TRABALHOS FUTUROS

Como continuidade deste trabalho, diversas possibilidades de aprimoramento podem ser exploradas a fim de ampliar a robustez, a autonomia e a escalabilidade do sistema desenvolvido. Uma evolução natural consiste na integração de um sensor de cloro livre, baseado em potencial de oxirredução (ORP), permitindo completar o ciclo de automação química da piscina e oferecer um controle mais preciso da desinfecção da água.

Outra linha promissora refere-se à aplicação de algoritmos de aprendizado de máquina voltados à manutenção preditiva. A partir do histórico operacional armazenado no sistema, seria possível identificar padrões de funcionamento e antecipar falhas em bombas, sensores ou outros componentes, reduzindo custos de manutenção e aumentando a confiabilidade da solução ao longo do tempo.

Adicionalmente, recomenda-se a implementação do sistema em piscinas de alvenaria em escala real, possibilitando a avaliação do desempenho em condições mais próximas do uso cotidiano. Esse cenário permitiria analisar o comportamento do sistema sob diferentes

volumes de água, condições ambientais e perfis de utilização, contribuindo para a validação de sua aplicabilidade prática e para eventuais ajustes de projeto.

No que diz respeito à escalabilidade, futuras versões do sistema podem incorporar suporte a múltiplas piscinas e usuários simultâneos, ampliando seu escopo de aplicação para condomínios, clubes ou empreendimentos de lazer. Para sustentar essa expansão, propõe-se a adoção de uma arquitetura orientada a eventos no *back-end*, utilizando o padrão de projeto *Observer*. Essa abordagem permitiria que a interface web fosse notificada de forma proativa e praticamente instantânea sempre que novas leituras fossem processadas, eliminando a necessidade de requisições cíclicas (*polling*)¹. Como resultado, espera-se uma redução no consumo de recursos computacionais, maior eficiência na comunicação e uma experiência de uso mais fluida para o usuário final.

¹ Técnica na qual um sistema realiza verificações periódicas em intervalos regulares para identificar a ocorrência de novos dados ou eventos.

6 CONSIDERAÇÕES FINAIS

Este trabalho teve como objetivo desenvolver um sistema automatizado para o monitoramento e a manutenção de piscinas residenciais, integrando tecnologias de Internet das Coisas (IoT) a uma arquitetura de software distribuída. Para alcançar esse propósito, foram investigados os principais parâmetros físico-químicos da água, bem como conceitos e protocolos associados à automação residencial, possibilitando a adaptação de componentes de baixo custo para a leitura de sensores e o controle de atuadores. Adicionalmente, foi desenvolvida uma aplicação completa, composta por *back-end* em Spring Boot e *front-end* em React, viabilizando o gerenciamento remoto e a visualização dos dados em tempo quase real.

Os resultados obtidos demonstraram a viabilidade técnica da solução proposta, evidenciando que o protótipo desenvolvido é capaz de reduzir significativamente a necessidade de intervenção manual no processo de manutenção da piscina, ao mesmo tempo em que promove maior segurança sanitária aos usuários. A automação do monitoramento e do acionamento dos dispositivos permitiu maior precisão no controle dos parâmetros avaliados, minimizando falhas associadas à operação manual e contribuindo para a estabilidade do tratamento da água.

A adoção de hardware acessível e software livre configura um avanço relevante no contexto da domótica residencial, ao demonstrar que soluções automatizadas podem ser implementadas de forma economicamente viável. O estudo confirmou que a arquitetura baseada em *gateway*, na qual o Raspberry Pi atua como intermediário da comunicação e o Arduino executa o controle físico dos dispositivos, apresenta-se como uma alternativa robusta e escalável. Dessa forma, a hipótese central do trabalho (de que a integração entre sensores, atuadores e tecnologias IoT pode otimizar o uso de recursos hídricos e energéticos) foi validada, resultando em uma gestão mais eficiente, sustentável e econômica.

Do ponto de vista científico e tecnológico, a principal contribuição deste trabalho reside na aplicação prática da metodologia Rational Unified Process (RUP) em um projeto de IoT, evidenciando sua adequação para o desenvolvimento de sistemas que envolvem integração entre hardware e software. Destaca-se, ainda, a implementação de um ciclo de controle fechado, no qual os dados coletados pelos sensores influenciam diretamente o comportamento do sistema, sem dependência constante da intervenção humana. Metodologicamente, a pesquisa integra conceitos de eletrônica embarcada, redes de computadores e desenvolvimento web, oferecendo um modelo que pode ser replicado e adaptado em futuros projetos acadêmicos ou aplicações comerciais.

Apesar dos avanços alcançados, algumas limitações foram identificadas ao longo

do desenvolvimento. A dependência da infraestrutura de rede Wi-Fi local mostrou-se um ponto crítico para o monitoramento remoto contínuo, podendo impactar a atualização da interface em cenários de instabilidade de conexão. Ainda assim, as rotinas de controle local implementadas no microcontrolador contribuíram para mitigar riscos operacionais, assegurando o funcionamento básico do sistema mesmo diante de falhas de comunicação.

De forma geral, os resultados obtidos reforçam que a automação residencial constitui um campo acessível e em constante expansão. Este trabalho demonstra que é possível desenvolver soluções tecnológicas relativamente complexas com recursos limitados, promovendo não apenas maior conforto e praticidade no ambiente doméstico, mas também a conscientização sobre o uso racional da água, a eficiência energética e a modernização dos processos de manutenção residencial.

REFERÊNCIAS

ACCARDI, A.; DODONOV, E. Automação residencial: elementos básicos, arquiteturas, setores, aplicações e protocolos. *Revista TIS*, v. 1, n. 2, 2012. Citado 2 vezes nas páginas 25 e 27.

ALLEBRANDT, P. C. *Reengenharia de Sistemas com RUP: Estudo de Caso: APUFSC*. Monografia (Trabalho de Conclusão de Curso (Graduação em Ciências da Computação)) — Universidade Federal de Santa Catarina, Florianópolis, 2004. Disponível em: <<https://repositorio.ufsc.br/bitstream/handle/123456789/183849/TCC-PauloCesarAllebrandt.pdf?sequence=-1>>. Acesso em: 31 dez. 2025. Citado na página 34.

ATCLLOR. *Guia de Tratamento de Piscinas*. [S.l.], 2021. Disponível em: <<https://atcllor.com.br/download/Guia-do-Tratamento.pdf>>. Citado 4 vezes nas páginas 17, 18, 20 e 21.

BENEDITO, J. N. F. et al. *Projeto: expansão de uma empresa prestadora de serviços em tratamento e limpeza de piscinas*. Trabalho de Conclusão de Curso — Centro Paula Souza, São Paulo, 2024. Citado 2 vezes nas páginas 16 e 17.

BERNHARDT, G. H. et al. A utilização da linguagem de modelagem unificada (uml) na engenharia de software. In: *Anais do Encontro Científico Cultural Interinstitucional*. [s.n.], 2023. v. 21. Disponível em: <<https://www4.fag.edu.br/anais-2023/Anais-2023-48.pdf>>. Citado na página 35.

BOEIRA, A. M. *Autopool: sistema de automação de piscina*. Trabalho de Conclusão de Curso — Centro Universitário Unifacvest, Lages, SC, 2019. Citado 2 vezes nas páginas 37 e 38.

BRANDÃO, A. F.; ROCHA, M. F. S. *Sistema de controle automático para desinfecção e correção de pH aplicado em piscinas*. Trabalho de Conclusão de Curso — Universidade Tecnológica Federal do Paraná, Cornélio Procópio, PR, 2018. Citado 2 vezes nas páginas 37 e 38.

BROCKVELD, M. V. V.; TEIXEIRA, C. S.; SILVA, M. R. d. A cultura maker em prol da inovação: boas práticas voltadas a sistemas educacionais. In: *Anais da Conferência ANPROTEC*. [s.n.], 2017. Disponível em: <<https://via.ufsc.br/wp-content/uploads/2017/11/maker.pdf>>. Citado na página 28.

CALÇA, J. V. J. *Análise comparativa entre os frameworks Django e Spring Boot*. Trabalho de Conclusão de Curso — Centro Paula Souza, São Paulo, 2022. Citado na página 32.

CAMPOS, R. A. F. *Automação residencial utilizando arduino e aplicação web*. Trabalho de Conclusão de Curso — Centro Universitário de Brasília, Brasília, 2014. Citado 2 vezes nas páginas 37 e 38.

CARDOSO, T. G. *Sensor de turbidez para análise de amostras de água*. Trabalho de Conclusão de Curso — Universidade Estadual da Paraíba, Campina Grande, 2011. Citado na página 30.

CASTRO, F. R.; CRUZ, F. M. da; ODDONE, N. E. O paradigma da orientação a objetos, a linguagem unificada de modelagem (uml) e a organização e representação do conhecimento. *Informação & Informação*, v. 18, n. 1, p. 82–105, 2013. Disponível em: <<https://ojs.uel.br/revistas/uel/index.php/informacao/article/view/9547/pdf>>. Citado na página 35.

Confederação Nacional da Indústria. *Indústria 4.0: Cinco anos depois*. 2022. Disponível em: <https://static.portaldaindustria.com.br/media/filer_public/cd/a2/cda22223-5c33-4a5f-af4e-f5a5d64b3d85/sondespecial_industria40_cincoanosdepois_abril2022.pdf>. Acessado em: 27 mai. 2025. Citado na página 11.

DAGOSTIM, N. E.; JORGE, G. B. Automação residencial: Aliando tecnologia e praticidade. In: FAG. *Anais do Encontro Científico Cultural Interinstitucional*. 2022. Disponível em: <<https://www4.fag.edu.br/anais-2022/Anais-2022-53.pdf>>. Citado na página 23.

DATE, C. J. *Introdução a sistemas de bancos de dados*. Rio de Janeiro: Elsevier Brasil, 2004. Citado na página 33.

EICOS. *Sensor de nível montagem vertical LC26M-40*. 2025. Disponível em: <<https://www.eicos.com.br/sensor-de-nivel/montagem-vertical/LC26M-40/>>. Acessado em: 02 dez. 2025. Citado na página 69.

Engevil Piscinas. *A História da Piscina*. 2025. Disponível em: <<https://www.engevilm piscinas.com.br/historia-da-piscina/>>. Acessado em: 01 jul. 2025. Citado na página 14.

FIBRATEC. *Conheça a história das piscinas*. 2021. Disponível em: <<https://fibratecpiscinas.com.br/blog/historia-das-piscinas>>. Acessado em: 01 jul. 2025. Citado na página 15.

FLORÊNCIO, G. *Central microcontrolada para automação residencial: controle sem fio para sensores e atuadores*. Trabalho de Conclusão de Curso — Instituto Federal de Educação, Ciência e Tecnologia de São Paulo, São Paulo, 2015. Citado na página 31.

Genco. *Estojo de Testes*. 2025. Disponível em: <<https://www.genco.com.br/estojo-ot>>. Acessado em: 02 out. 2025. Citado na página 22.

Genco Química Industrial. *Guia completo para tratamento de águas de piscinas residenciais*. 3. ed. São Paulo, 2004. Disponível em: <http://www.itaicipiscinas.com.br/images/dicas/guia_genco.pdf>. Citado na página 19.

GENYO. *Automação: O que é e como colher os benefícios na sua empresa*. 2024. Disponível em: <<https://genyo.com.br/automacao/>>. Acessado em: 25 mai. 2025. Citado na página 11.

Grupo de Robótica. *Introdução ao Arduino*. 2012. Disponível em: <https://d1wqxts1xzle7.cloudfront.net/32413349/Arduino_Destacom-libre.pdf>. Acessado em: 12 dez. 2025. Citado na página 29.

HARRIS, T. *Como funcionam as Piscinas*. 2025. Disponível em: <<https://home.howstuffworks.com/swimming-pool.htm>>. Acessado em: 12 set. 2025. Citado 2 vezes nas páginas 15 e 16.

HIPÓLITO, J. G.; SILVA, M. d. J. d. Automação residencial com arduino. *Revista Eletrônica UNIFAFIBE*, 2018. Disponível em: <<https://unifafibe.com.br/revistasonline/arquivos/revistaelectrica/sumario/69/06022019135904.pdf>>. Citado 3 vezes nas páginas 25, 26 e 27.

Hu Infinito. *Sensor de temperatura MF58 NTC 10k prova d'água*. 2025. Disponível em: <<https://www.huinfinito.com.br/sensores/1595-sensor-de-temperatura-mf58-ntc-10k-prova-d-agua.html>>. Acessado em: 02 dez. 2025. Citado 2 vezes nas páginas 68 e 71.

JESUS, W. S. C. de. *Introdução à Lógica e a Linguagem de Programação*. 2025. Disponível em: <https://www1.univap.br/wagner/Logica_Prog-1.pdf>. Acessado em: 2025. Citado na página 31.

JUCÁ, S.; PEREIRA, R. *Aplicações práticas de sistemas embarcados Linux utilizando Raspberry Pi*. Rio de Janeiro: PoD Editoria, 2018. Citado 2 vezes nas páginas 29 e 30.

JÚNIOR, P. F. da S. et al. Desenvolvimento de front-end e back-end para nós sensores sem fio. In: *ResearchGate*. [s.n.], 2025. Disponível em: <<https://www.researchgate.net/publication/376763873>>. Citado na página 32.

LAVOR, C. F. C. d. *Verificação da conformidade da qualidade da água de piscinas de acordo com a NBR 10818/2016*. Trabalho de Conclusão de Curso — Universidade Federal do Ceará, Fortaleza, 2019. Citado na página 11.

LEITE, M. Y. F. *Plataforma de monitoramento da qualidade da água utilizando sistemas embarcados*. Trabalho de Conclusão de Curso — Universidade Federal Rural do Semi-Árido, Mossoró, 2020. Citado 3 vezes nas páginas 17, 26 e 30.

LUCIFABIO, E. M. et al. *Aquário automatizado*. Trabalho de Conclusão de Curso — Centro Paula Souza, São Paulo, 2023. Citado na página 31.

Mercado Livre. *Sensor de Turbidez para Arduino ST100*. 2025. Disponível em: <<https://www.mercadolivre.com.br/sensor-de-turbidez-para-arduino-st100>>. Acessado em: 04 dez. 2025. Citado na página 70.

Mercado Livre. *Sensor pH - Módulo de leitura pH4502c*. 2025. Disponível em: <<https://www.mercadolivre.com.br/sensor-ph--modulo-de-leitura-ph4502c>>. Acessado em: 04 dez. 2025. Citado na página 70.

Mercado Livre. *Uno R3 Atmega328 Compatível com Arduino*. 2025. Disponível em: <<https://www.mercadolivre.com.br/uno-r3-atmega328>>. Acessado em: 04 dez. 2025. Citado na página 71.

Money Report. *Automação residencial cresceu 21,8%*. 2024. Disponível em: <<https://www.moneyreport.com.br/negocios/automacao-residencial-cresceu-218/>>. Acessado em: 27 mai. 2025. Citado na página 11.

MOREIRA, W. A. *Processos Tradicionais de Desenvolvimento de Software*. 2025. Disponível em: <https://www.cin.ufpe.br/~processos/TAES3/Livro/00-LIVRO/01-PTDS-v7_CORRIGIDO_Versao_Final_OK.pdf>. Acessado em: 2025. Citado na página 33.

MURATORI, J. R.; BÓ, P. H. D. Capítulo i: Automação residencial: histórico, definições e conceitos. *O Setor Elétrico*, p. 70–77, 2011. Disponível em: <https://www.osetoreletrico.com.br/wp-content/uploads/2011/04/Ed62_fasc_automacao_capI.pdf>. Citado 2 vezes nas páginas 23 e 24.

OLIVEIRA, G. F.; ALVES, M. C. O. Domótica: substituição da fiação de retorno nas instalações elétricas por cabeamento de dados e sistemas microcontrolados. *SITEFA*, v. 2, n. 1, p. 391–403, 2019. Disponível em: <<https://publicacoes.fatecsertaozinho.edu.br/sitefa/article/view/61/66>>. Citado na página 25.

OLIVEIRA, L. M. S. d. et al. *Automação da área de lazer*. Trabalho de Conclusão de Curso — Centro Paula Souza, São Paulo, 2023. Citado 2 vezes nas páginas 37 e 38.

PISKE, O. R. Rup - rational unified process. *Angus Young*, v. 16, n. 11, 2003. Disponível em: <https://www.angusyoung.org/arquivos/artigos/trabalho_rup.pdf>. Citado na página 33.

Pool News. Pool automation opportunity for pool owners. *Pool Magazine*, mar 2020. Disponível em: <<https://www.poolmagazine.com/features/pools/pool-automation-opportunity-for-pool-owners/>>. Citado na página 11.

Portal Ideia. *Limpeza e Manutenção de Piscinas*. 2025. Disponível em: <<https://portalidea.com.br/cursos/e643d94d6e685cffa352bd7370bf84a7.pdf>>. Acessado em: 24 jul. 2025. Citado na página 17.

Raspberry Pi. *Raspberry Pi 3 Model B*. 2025. Disponível em: <<https://www.raspberrypi.com/products/raspberry-pi-3-model-b/>>. Acessado em: 04 dez. 2025. Citado na página 72.

Revista Mundo Elétrico. *Mercado de automação residencial segue em forte crescimento*. 2023. Disponível em: <<https://www.revistamundoeletrico.com.br/tecnologia/tecnologia-tecnologia/mercado-de-automacao-residencial-segue-em-forte-crescimento/>>. Acessado em: 27 mai. 2025. Citado na página 11.

SILVA, E. L. S.; FERNANDES, S. C. *Automação de telescópio dobsoniano para a democratização do ensino da astronomia*. Trabalho de Conclusão de Curso (Tecnólogo em Análise e Desenvolvimento de Sistemas) — Instituto Federal de Educação, Ciência e Tecnologia de Goiás, Formosa, 2025. Citado 3 vezes nas páginas 28, 29 e 33.

SILVA, S. M. *Estudo de caso sobre o processo de tratamento de água em uma piscina industrial de testes hidrostáticos*. Trabalho de Conclusão de Curso (Engenharia Mecânica) — Universidade Federal do Rio de Janeiro, Macaé, 2021. Citado na página 19.

SOUZA, M. V. A. d. *Desenvolvimento em React: a influência de Clean Code e Design Patterns na visão dos desenvolvedores*. Trabalho de Conclusão de Curso — Universidade Federal do Ceará, Fortaleza, 2025. Citado na página 32.

SOUZA, M. O. *Sensor de nível tipo deslocador com autocompensação da densidade do líquido*. Dissertação (Mestrado) — Universidade Federal de Sergipe, São Cristóvão, 2018. Citado na página 30.

SULIMANN, C. E. D.; HAUPT, L. *Automatização do processo de tratamento de piscinas: aplicação de reagentes e limpeza da água*. Projeto de Iniciação Científica — Instituto Federal Catarinense, Luzerna, SC, 2014. Citado na página 38.

- Universidade Estadual Paulista. *Diagrama de Casos de Uso*. 2025. Disponível em: <https://moodle.unesp.br/pluginfile.php/25934/mod_resource/content/1/diagrama_casos_uso.pdf>. Acessado em: 14 dez. 2025. Citado 2 vezes nas páginas 35 e 36.
- VASCONCELOS, A. L. M. *Concepção de um Sistema para a Metodologia Sala de Aula Aberta*. Trabalho de Conclusão de Curso — Universidade de Pernambuco, Recife, 2018. Citado na página 33.
- VASCONCELOS, A. M. L. D. et al. *Introdução à Engenharia de Software e à Qualidade de Software*. 2006. Disponível em: <http://nti.facape.br/jocelio/es/apostilas/Mod.01.MPS_Engenharia&QualidadeSoftware_V.28.09.06.pdf>. Acessado em: 2025. Citado na página 33.

Apêndices

APÊNDICE A – DETALHAMENTO DOS COMPONENTES DE HARDWARE

Neste apêndice são apresentados os componentes físicos utilizados na montagem do protótipo, conforme referenciado no Capítulo de Desenvolvimento.

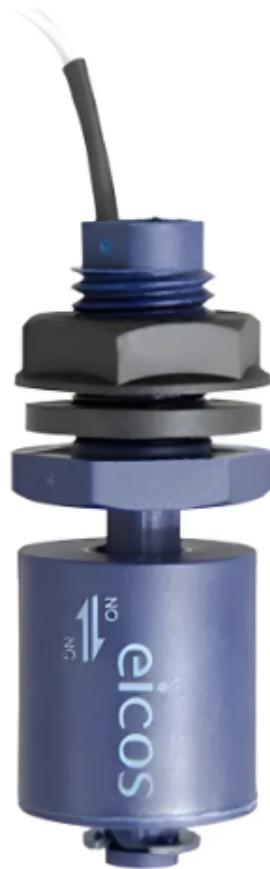
A.1 SENSORES

Figura 22 – Sensor de Temperatura MF58 (NTC 10K)



Fonte: ([Hu Infinito, 2025](#)).

Figura 23 – Sensor de Nível LC26M-40



Fonte: ([EICOS, 2025](#)).

Figura 24 – Sensor de pH



Fonte: ([Mercado Livre, 2025b](#)).

Figura 25 – Sensor de Turbidez ST100



Fonte: ([Mercado Livre, 2025a](#)).

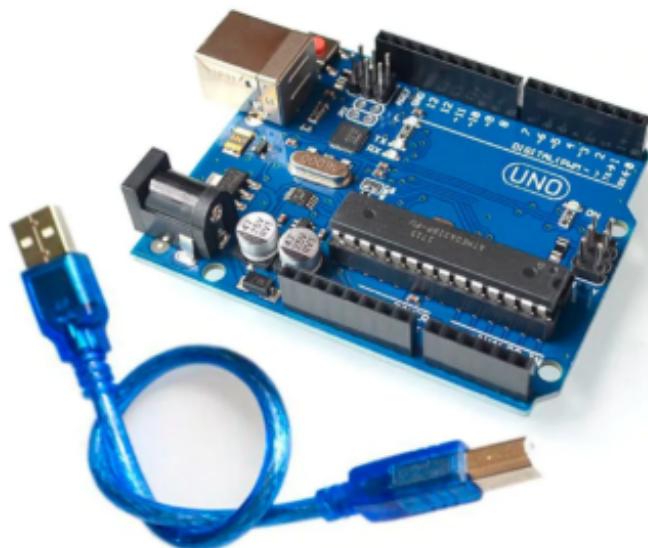
A.2 ATUADORES E CONTROLADORES

Figura 26 – Bomba Submersa JT100



Fonte: ([Hu Infinito, 2025](#)).

Figura 27 – Placa Arduino Uno R3 (Controlador Físico)



Fonte: ([Mercado Livre, 2025c](#)).

Figura 28 – Raspberry Pi 3 Model B (Gateway)



Fonte: ([Raspberry Pi, 2025](#)).

APÊNDICE B – CÓDIGOS FONTE DO SISTEMA

A seguir, apresentamos os trechos essenciais dos scripts desenvolvidos para o funcionamento do sistema de automação, demonstrando a integração entre as camadas de *Firmware*, *Gateway* e Aplicação.

B.1 FIRMWARE DO MICROCONTROLADOR (ARDUINO)

O código abaixo é responsável pela leitura dos sensores em tempo real e controle imediato do relé da bomba, garantindo segurança operacional mesmo sem conexão de rede.

```
#define PINO_PH A0
#define PINO_TEMP A1
#define PINO_RELE 6
#define PINO_BOIA 7

// Configuracao NTC 10k
#define R_FIXO 10000.0
#define RT0 10000.0
#define BETA 3950.0

float calibracao_slope = 0.18;
int estadoBomba = 0;
bool forcarLigado = false;

// Variavel para controlar o tempo do envio
unsigned long ultimoEnvio = 0;

void setup() {
    Serial.begin(9600);

    pinMode(PINO_PH, INPUT);
    pinMode(PINO_BOIA, INPUT_PULLUP);
    pinMode(PINO_RELE, OUTPUT);

    digitalWrite(PINO_RELE, LOW);
    Serial.println("INICIO: MODO TURBO ATIVADO");
}

void loop() {
```

```
// 1. LEITURA E CONTROLE IMEDIATO
if (Serial.available() > 0) {
    char comando = Serial.read();
    if (comando == 'L') forcarLigado = true;
    if (comando == 'D') forcarLigado = false;
}

int leituraBoia = digitalRead(PINO_BOIA);

// Logica de Seguranca Local
bool deveLigar = (leituraBoia == HIGH) || (forcarLigado == true);

if (deveLigar) {
    digitalWrite(PINO_RELE, HIGH);
    estadoBomba = 1;
}
else {
    digitalWrite(PINO_RELE, LOW);
    estadoBomba = 0;
}

// 2. ENVIAR DADOS (A CADA 2 SEGUNDOS)
if (millis() - ultimoEnvio > 2000) {
    ultimoEnvio = millis();

    // Calculo Temperatura
    float tempCelsius = 0;
    int leituraTemp = analogRead(PINO_TEMP);
    float voltagemTemp = leituraTemp * (5.0 / 1023.0);
    if (voltagemTemp > 0.1 && voltagemTemp < 4.9) {
        float r_termistor = R_FIXO * (5.0 - voltagemTemp) / voltagemTemp;
        float tempKelvin = 1.0 / (1.0 / (273.15 + 25.0) + log(r_termistor / RT0) / BETA);
        tempCelsius = tempKelvin - 273.15;
    }
}

// Calculo pH
long somaLeituras = 0;
for(int i = 0; i < 10; i++) {
    somaLeituras += analogRead(PINO_PH);
    delay(10);
}
float mediaPH = somaLeituras / 10.0;
```

```

        float voltagemPH = mediaPH * (5.0 / 1023.0);
        float valorPH = 7.00 + ((2.50 - voltagemPH) / calibracao_slope);

        // Serializacao JSON
        Serial.print("{\"t\":");
        Serial.print(tempCelsius);
        Serial.print(",\"n\":");
        Serial.print(leituraBoia == HIGH ? 0 : 1);
        Serial.print(",\"p\":");
        Serial.print(valorPH);
        Serial.print(",\"b\":");
        Serial.print(estadoBomba);
        Serial.println("}");

    }
}

```

B.2 SCRIPT DO GATEWAY (PYTHON)

Este *script* roda no Raspberry Pi, atuando como ponte entre a porta Serial (USB) e a API REST.

```

import serial
import json
import requests
import time
import sys

# --- CONFIGURACOES ---
PORTA_ADUINO = '/dev/ttyUSB0'
VELOCIDADE = 9600
IP_PC = "10.0.0.241"
URL_BACKEND = f"http://[{IP_PC}]:8080/api/sensor/receber"

def conectar_arduino():
    while True:
        try:
            print(f" Tentando conectar ao Arduino em {PORTA_ADUINO}...")
            ser = serial.Serial(PORTA_ADUINO, VELOCIDADE, timeout=1)
            time.sleep(2)
            print(" Conectado com sucesso!")
            return ser
        except serial.SerialException:

```

```
print(" Arduino nao encontrado. Tentando de novo em 3s...")
time.sleep(3)

# --- LOOP PRINCIPAL ---
arduino = conectar_arduino()

while True:
    try:
        if arduino.in_waiting > 0:
            try:
                linha = arduino.readline().decode('utf-8', errors='ignore').strip()

                if not linha: continue

                print(f" Enviando: {linha}")
                dados = json.loads(linha)

                # Envia e aguarda resposta do comando (Ligar/Desligar)
                resposta = requests.post(URL_BACKEND, json=dados, timeout=2)

                if resposta.status_code == 200:
                    try:
                        comando_java = resposta.json()
                        if comando_java.get("ligarBomba") == True:
                            arduino.write(b'L')
                        elif comando_java.get("ligarBomba") == False:
                            arduino.write(b'D')
                    except json.JSONDecodeError:
                        pass

                except json.JSONDecodeError:
                    pass
                except requests.exceptions.RequestException as e:
                    print(f" Erro de conexao com o Java: {e}")

            except (OSError, serial.SerialException):
                if 'arduino' in locals() and arduino.is_open:
                    arduino.close()
                arduino = conectar_arduino()

    except KeyboardInterrupt:
        if 'arduino' in locals() and arduino.is_open:
```

```
    arduino.close()
    break
```

B.3 CONTROLLER DO BACK-END (JAVA/SPRING BOOT)

Controlador REST responsável por receber os dados do *Gateway* e persisti-los no banco PostgreSQL.

```
package com.ifg.residIFG.controllers;

import com.ifg.residIFG.domain.piscinas.LeituraSensor;
import com.ifg.residIFG.domain.user.User;
import com.ifg.residIFG.dto.SensorDashboardDTO;
import com.ifg.residIFG.repository.LeituraSensorRepository;
import com.ifg.residIFG.repository.UserRepository;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.*;

import java.util.List;
import java.util.Map;

@RestController
@RequestMapping("/api/sensor")
@CrossOrigin(origins = "*")
public class SensorController {

    @Autowired
    private LeituraSensorRepository repository;
    @Autowired
    private UserRepository userRepository;

    @PostMapping("/receber")
    public void receberDados(@RequestBody Map<String, Object> dados) {
        Double temp = Double.valueOf(dados.get("t").toString());
        Integer nivelRaw = Integer.valueOf(dados.get("n").toString());
        Boolean nivel = (nivelRaw == 1);
        Double ph = Double.valueOf(dados.get("p").toString());
        Integer bombaRaw = Integer.valueOf(dados.get("b").toString());
        Boolean bombaAtiva = (bombaRaw == 1);

        // Leitura de Turbidez (tb) com tratamento de nulo
        Double turbidez = 0.0;
```

```

        if (dados.get("tb") != null) {
            turbidez = Double.valueOf(dados.get("tb").toString());
        }

        LeituraSensor leitura = new LeituraSensor(temp, nivel, ph, bombaAtiva, turbidez);
        repository.save(leitura);
    }

    @GetMapping("/atual")
    public SensorDashboardDTO getAtual() {
        // 1. Busca a ultima leitura do sensor
        LeituraSensor leitura = repository.findTopByOrderByDataHoraDesc();

        // 2. Busca dimensoes do usuario para calculo de volume
        List<User> users = userRepository.findAll();
        Double volumeCalculado = 0.0;

        if (!users.isEmpty()) {
            User unicoUsuario = users.get(0);
            // Formula: (Comp * Larg * Prof) * 1000 = Litros
            float v = unicoUsuario.getComprimentoPiscina() * unicoUsuario.getLarguraPiscina();
            volumeCalculado = (double) (v * 1000);
        }

        // Retorna DTO com dados do sensor + volume calculado
        return new SensorDashboardDTO(leitura, volumeCalculado);
    }
}

```

B.4 FRONT-END (REACT)

Trecho do componente principal responsável pelo *polling* de dados e envio de comandos manuais.

```

useEffect(() => {
    const buscarDados = async () => {
        try {
            const response = await api.get("/api/sensor/atual");

            if (response.data) {
                setTemperaturaAtual(response.data.temperatura || 0);
                setNivelAgua(response.data.nivelOk);
            }
        } catch (error) {
            console.error('Erro ao buscar dados:', error);
        }
    };
    buscarDados();
});

```

```

        if (response.data.ph) {
            setPhAtual(response.data.ph);
        }

        setPumpActive(response.data.bombaAtiva);
    }
} catch (error) {
    console.error("Erro ao buscar sensores:", error);
}
};

buscarDados();
const intervalo = setInterval(buscarDados, 5000);
return () => clearInterval(intervalo);
}, []);
}

const handlePumpToggle = async (checked: boolean) => {
    setPumpActive(checked);

    const acao = checked ? "LIGAR" : "DESLIGAR";
    try {
        await api.post("/api/sensor/comando-bomba", { ligar: checked });
        toast.success(`Comando: ${acao} enviado`);
    } catch (error) {
        toast.error("Erro ao enviar comando.");
        setPumpActive(!checked);
    }
};

```

B.5 ALGORITMO DE RECOMENDAÇÃO QUÍMICA E ALERTAS (RE-ACT)

O trecho abaixo apresenta a função `verificarQuimica` na camada de aplicação. Este algoritmo compara os dados monitorados com as normas sanitárias e com as configurações do usuário, gerando as sugestões de dosagem proporcionais ao volume calculado.

```

// Algoritmo de Recomendacao de Dosagem e Alertas
const verificarQuimica = (ph: number, turbidez: number,
                           temp: number, volLitros: number) => {

    // Verifica tempo de silencio (Cooldown de alertas para nao spammar)

```

```
if (Date.now() < proximoAlerta) return;

let produto = "";
let dosagemPor1000 = 0;
let motivo = "";

// 1. Logica de Correcao de pH (Tabela de Referencia)
if (ph < 6.8) {
    produto = "pH+ (Liquido)";
    motivo = "pH muito baixo (< 6.8)";
    dosagemPor1000 = 20;
} else if (ph >= 6.8 && ph < 7.2) {
    if (ph <= 7.0) {
        produto = "pH+ (Liquido)";
        motivo = "pH levemente baixo";
        dosagemPor1000 = 15;
    }
} else if (ph > 7.6) {
    produto = "pH- (Liquido)";
    motivo = "pH alto (> 7.6)";
    dosagemPor1000 = 10;
}

// 2. Logica de Turbidez (Clarificacao se > 3.0 NTU)
else if (turbidez > 3.0) {
    produto = "Clarificante";
    motivo = 'Agua turva (${turbidez} NTU)';
    dosagemPor1000 = 4;
}

// 3. Alerta de Temperatura (Baseado na Configuracao do Usuario)
else if (temp < minTempConfig) {
    toast({
        title: "Temperatura Baixa",
        description: 'Atual: ${temp}°C (Minimo: ${minTempConfig}°C)',
    });
    // Define cooldown curto para temperatura
    setProximoAlerta(Date.now() + (5 * 60 * 1000));
    return;
}

// Calculo Final e Disparo da Sugestao
```

```
if (produto !== "") {  
    const qtdTotal = (volLitros / 1000) * dosagePor1000;  
  
    const tempoSilencio = 20 * 60 * 1000; // 20 min sem repetir  
    setProximoAlerta(Date.now() + tempoSilencio);  
  
    toast({  
        variant: "destructive",  
        title: 'Atenção: ${motivo}',  
        description: 'Clique para ver a correção.',  
        action: (  
            <ToastAction altText="Ver Dosagem"  
                onClick={() => setIsModalOpen(true)}>  
                Ver Dosagem  
            </ToastAction>  
        ),  
    });  
}  
};
```