

**CENTRO UNIVERSITÁRIO DO SUL DE MINAS – UNIS/MG**  
**BACHARELADO EM ENGENHARIA ELÉTRICA**  
**FABRÍCIO MORALES DE OLIVEIRA**

**PROTÓTIPO DE GERENCIAMENTO DO NÍVEL E TRATAMENTO DE ÁGUA VIA**  
**RÁDIO FREQUÊNCIA**

**Varginha**  
**2017**

**FABRÍCIO MORALES DE OLIVEIRA**

**PROTÓTIPO DE GERENCIAMENTO DO NÍVEL E TRATAMENTO DE ÁGUA VIA  
RÁDIO FREQUÊNCIA**

Trabalho de Conclusão do Curso apresentada ao curso de Engenharia Elétrica do Centro Universitário do Sul de Minas – UNIS-MG como pré-requisito para obtenção do título de bacharel em Engenharia Elétrica sob orientação do Professor Me. Eduardo Henrique Ferroni.

**Varginha  
2017**

**FABRÍCIO MORALES DE OLIVEIRA**

**PROTÓTIPO DE GERENCIAMENTO DO NÍVEL E TRATAMENTO DE ÁGUA VIA  
RÁDIO FREQUÊNCIA**

Trabalho de Conclusão do Curso apresentada ao curso de Engenharia Elétrica do Centro Universitário do Sul de Minas – UNIS-MG como pré-requisito para obtenção do título de bacharel em Engenharia Elétrica pela banca examinadora composta pelos membros:

Aprovado em: \_\_\_/\_\_\_/\_\_\_

---

Prof. Me Eduardo Henrique Ferroni

---

Prof. Me Helciner Vítor Ferreira

---

Prof. Me Hugo Rodrigues Vieira

OBS.:

Dedico este trabalho primeiramente a Deus, ao meu pai, Fábio, minha mãe, Marilda, minha Irmã, Mônica, minha Esposa, Daiane, e a todos os meus amigos e colegas que de forma especial e carinhosa me deram força e coragem, me apoiando nos momentos de dificuldades.

## AGRADECIMENTOS

Primeiramente agradeço a Deus que permitiu que tudo isso acontecesse, ao longo de minha vida, e não somente nestes anos como universitário, mas que em todos os momentos de minha vida. Ao meu Orientador, Professor Me Eduardo Henrique Ferroni, pelo suporte no pouco tempo que lhe coube, pelas suas correções e incentivos. Á todos os Professores do UNIS em especial ao Professor Me Hugo Rodrigues Vieira pelo paciente trabalho de revisão deste trabalho. Ao meu compadre Professor Me Erasmo Evangelista de Oliveira pelo paciente trabalho de revisão e orientação deste trabalho. Aos amigos e colegas da Sisponio, pelo incentivo e pelo apoio constante. Em especial ao Anderson José da Silva ao Régis e Gisela. Ao meu pai, Fábio, minha mãe, Marilda, minha Irmã, Mônica e a minha esposa, Daiane. E a todos que direta ou indiretamente fizeram parte da minha formação, o meu muito obrigado.

“Uma pessoa que nunca cometeu um engano é  
porque nunca tentou fazer nada”

Albert Einstein

## RESUMO

Neste trabalho é apresentado um protótipo para controle e automação de uma estação de tratamento e distribuição de água utilizando a plataforma Rádium. Como objetivo geral, o trabalho visa apresentar uma solução eficiente para o controle do nível e do tratamento de água nos reservatórios de uma companhia de saneamento e abastecimento de água. Destacam-se como principais características do protótipo apresentado, um controle para atender municípios cuja topografia obrigue o emprego de acionamento de bombas de estações elevatórias de água, monitorando assim a vazão para macromedição não intrusiva, bem como o monitoramento de parâmetros de qualidade da água, por meio da leitura de medições dos sensores hidrostáticos de turbidez, dosagem de cloro, flúor, ferro. No protótipo proposto temos dois componentes principais: a plataforma de controle e automação (acionamento das bombas dos reservatórios, leitura das medições dos sensores que monitoram a qualidade e o nível da água) e um *software* de monitoramento. Como ferramenta de controle e automação foi utilizado a plataforma Rádium. Tal ferramenta foi utilizada, pois facilita o entendimento do sistema e permite configurá-lo de acordo com os requisitos e parâmetros de controle. Além disso, a plataforma Rádium possibilita uma comunicação com uma rede de sensores sem fio, permitindo a comunicação com os pontos automatizados via rádio frequência, utilizando a frequência de 915 MHz (Dentro da faixa liberada pela Anatel para aplicações industriais, científicas e médicas - Faixa de Frequência ISM). O *software* de monitoramento foi desenvolvido para centralizar e exibir os dados de medições por meio de uma interface única, de forma a permitir uma interação com o usuário, além de possibilitar que outros meios de interface com o sistema sejam desenvolvidos, como uso de páginas ou aplicativos móveis.

**Palavras-chave:** Controle e Automação. Rede de sensores sem fio. Monitoramento. Qualidade de água. Plataforma Rádium.

## **ABSTRACT**

*This work presents a prototype for control and automation of a water treatment and distribution station using the RADIUINO platform. As a general objective, the work aims to present an efficient solution for controlling the level and treatment of water in the reservoirs of a water supply and sanitation company. The main characteristics of the prototype presented are a control to meet municipalities whose topography requires the use of pumps for water lift stations, thus monitoring the flow to non-intrusive macro mediation, as well as the monitoring of water quality parameters, through the measurement of hydrostatic turbidity sensors, chlorine, fluorine and iron dosing. In the proposed prototype we have two main components: the control and automation platform (activation of the pumps of the reservoir, reading of the measurements of the sensors that monitor the quality and the level of the water) and a software of monitoring. The RADIUINO platform was used as a control and automation tool. This tool was used because it facilitates the understanding of the system and allows to configure it according to the requirements and control parameters. In addition, the RADIUINO platform enables communication with a wireless sensor network, allowing communication with the automated points via radio frequency, using the frequency of 915 MHz (Within the range released by Anatel for industrial, scientific and medical applications - Range of Frequency ISM). Monitoring software has been developed to centralize and display measurement data through a single interface to allow user interaction and to enable other system interface media to be developed such as page usage or applications.*

**Keywords:** *Control and automation. Wireless sensor network. Monitoring. Waterquality. RADIUINO Platform*

## LISTA DE ILUSTRAÇÕES

Figura 01 – Esquema de uma IOT.....	16
Figura 02 – Dispersão dos nós sensores.....	17
Figura 03 – Componentes dos nós sensores.....	18
Figura 04 – Esquema Plataforma Arduino.....	20
Figura 05 – Diagrama de blocos da lógica de funcionamento do sistema <i>Radiuind</i> via RF....	21
Figura 06 – Mapa do SAAE Boa Esperança MG.....	24
Figura 07 – Diagrama Esquemático dos Reservatórios ETA e Bombeamento SAAE Boa Esperança MG.....	25
Figura 08 – Analisador de Cloro Residual Amperométrico.....	26
Figura 09 – Medidor de Turbidez.....	27
Figura 10 – Medidor de Flúor.....	27
Figura 11 – Sensor de Nível Hidrostático.....	28
Figura 12 – Sensor de Vazão.....	28
Figura 13 – Contator (WEG 2017).....	29
Figura 14 – <i>Software</i> de desenvolvimento Arduino.....	30
Figura 15 – Módulo de RF do Arduino (Radiuino).....	32
Figura 16 – Diagrama da solução: Comunicação por de RF da rede de sensores sem fio com a base central.....	34
Figura 17 – Tela do <i>Software</i> em operação <i>online</i> .....	37
Figura 18 – Tela do <i>Software</i> em operação <i>off-line</i> .....	38

## LISTA DE ABREVIATURAS E SIGLAS

ANATEL – Agência Nacional de Telecomunicações

ETA – Estação de Tratamento de Água

ETE – Estação de tratamento de esgoto

GPS – *Global positioning system*

IDE – *Integrated Development Environment*

IEEE – Instituto de Engenheiros Eletricistas e Eletrônicos

IOT – *Internet of Things*

ISM – *Industrial, Scientific and Medical*

MYSQL – *Structured Query Language*

RF – Radiofrequência

RSSF – Rede de Sensores Sem Fio

SAAE – Serviço Autônomo de Água e Esgoto

SGDB – Sistema de gerenciamento de banco de dados

UNIS-MG – Centro Universitário do Sul de Minas

## SUMÁRIO

<b>1 INTRODUÇÃO .....</b>	<b>11</b>
<b>1.1 Contextualização.....</b>	<b>11</b>
<b>1.2 Definição do Problema .....</b>	<b>12</b>
<b>1.3 Objetivos Geral e Específicos .....</b>	<b>13</b>
<b>1.4 Solução proposta.....</b>	<b>13</b>
<b>1.5 Organização do Trabalho .....</b>	<b>14</b>
<b>2REFERENCIAL TEÓRICO .....</b>	<b>15</b>
<b>2.1 Internet das Coisas – IOT.....</b>	<b>15</b>
<b>2.2 Redes de Sensores sem Fio – RSSF .....</b>	<b>17</b>
<b>2.3 Plataforma Arduino e Rádiumo.....</b>	<b>19</b>
2.3.1 Arduino.....	20
2.3.2 Rádiumo.....	21
<b>3 METODOLOGIA.....</b>	<b>23</b>
<b>3.1 Estudo e caracterização sobre o local de implantação .....</b>	<b>23</b>
<b>3.2 Identificação dos Parâmetros a serem monitorados e dos sensores a serem utilizados .....</b>	<b>25</b>
<b>3.3 Identificação dos Atuadores .....</b>	<b>29</b>
<b>3.4 Ferramentas .....</b>	<b>30</b>
3.4.1 Ambiente de Desenvolvimento do Arduino .....	30
3.4.2 Módulo RF BE900.....	30
3.4.3 Ambiente de Desenvolvimento do Sistema de Gerenciamento <i>Desktop</i> .....	32
<b>4 PROTÓTIPO PROPOSTO .....</b>	<b>33</b>
<b>4.1 Plataforma de Automação e Controle(Ambiente Sensor e Ambiente Base) .....</b>	<b>34</b>
4.1.1 Ambiente base .....	34
4.1.2 AmbienteSensor.....	35
<b>4.2 Sistema de Gerenciamento de Gerenciamento <i>Desktop</i> .....</b>	<b>35</b>
4.2.1 Principais Funcionalidades do Sistema de Gerenciamento .....	36
4.2.2 Telas de Funcionamento e Modos de Operação do Sistema .....	37
<b>5 RESULTADOS E DISCUSSÕES.....</b>	<b>39</b>
<b>6 CONCLUSÃO E TRABALHOS FUTUROS.....</b>	<b>41</b>
<b>REFERÊNCIAS.....</b>	<b>44</b>

# 1 INTRODUÇÃO

Neste trabalho o foco principal foi o desenvolvimento de um protótipo para controle e automação de uma estação de tratamento e distribuição de água utilizando a plataforma Rádiuino<sup>1</sup>. Nesse sentido, o trabalho visa apresentar uma solução eficiente para o controle do nível e da qualidade da água nos reservatórios de uma companhia de saneamento e abastecimento de água.

Destacam-se como principais características do protótipo proposto: monitoramento do nível de água dos reservatórios, controle automático do acionamento de bombas de estações elevatórias de água, monitoramento da vazão por meio de macro medição não intrusiva<sup>2</sup>, bem como o monitoramento de parâmetros de qualidade da água, por meio da leitura de medições dos sensores hidrostáticos de turbidez, dosagem de cloro e flúor.

## 1.1 Contextualização

É comum, em algumas cidades, a falta de água e em outros momentos, o desperdício excessivo, onde caixas d'água transbordam por falta de controle do nível (MOURA, 2015). Atacar este problema é uma questão estratégica para os municípios, na medida em que pode equacionar questões relativas à sustentabilidade sócio-econômico-ambiental.

Com a automação do processo do tratamento de água é possível verificar periodicamente as amostras para atestar a qualidade da água entregue a população. Além disso, é possível um monitoramento da vazão para macro medição não intrusiva, evitando transbordos nos reservatórios e conseqüentemente desperdícios de água.

Utilizando-se de bombas ligadas diretamente na saída do tratamento de água da ETA, é possível a coleta de porções de água de forma automática, em períodos pré-definidos, os volumes necessários para realizar uma análise de PH, cor, cloro residual, coliformes e a turbidez da água, e permitir a dosagem de componentes que qualificam positivamente a água tratada, como flúor.

---

<sup>1</sup> A plataforma Rádiuino consiste na integração da plataforma Arduino com transceptores de Radiofrequência. A plataforma Arduino é voltada para projetos que necessitam de prototipagem rápida e funcional. Existe uma grande variedade de sensores que podem ser utilizados no Arduino tais como de temperatura, umidade, distância, luminosidade e som. Neste trabalho utilizaremos sensores de nível, vazão e qualidade da água acoplado a transceptores de rádio frequência. *Fonte: <http://radiuino.cc>*

<sup>2</sup> Macro medição não intrusiva: medição por meio de sensores de vazão ultrassônicos colocados de fora da tubulação que não provoca alteração no regime de escoamento da água. *Fonte: <https://nivetec.com.br>*

Em termos econômicos, temos a imediata ausência de desperdícios com transbordamentos, utilização das bombas hidráulicas de forma racional e um número bem inferior da força humana empregada em todo o processo, pois não existe a necessidade de acompanhamento no local do funcionamento da distribuição, limitando-se apenas ao monitoramento na central de abastecimento, em turnos específicos, normalmente dois ou três turnos de trabalho por dia.

Dentro deste contexto, é necessário um processo de gestão eficiente do controle dos níveis dos reservatórios, bem como de todo o processo de tratamento da água. A seguir, definimos o problema descrevendo os pontos a serem atacados neste trabalho.

## **1.2 Definição do Problema**

O processo de gestão e tratamento de água em grande escala realizado em companhias de saneamento básico, em função da utilização de técnicas operacionais não otimizadas e adequadas acaba por gerar um alto desperdício de água, comprometendo assim o abastecimento local e afetando a população. Automatizar um sistema significa deixá-lo de maneira independente do trabalho manual que o ser humano realiza, e assim as falhas são minimizadas e o resultado é mais preciso e objetivo.

Neste trabalho o foco central é a automação e controle dos níveis de água nos reservatórios e o controle da qualidade da água por meio de uma rede de sensores sem fio. Dessa forma, o problema consiste em: dados os parâmetros qualidade da água (cloro, flúor, turbidez), vazão e de nível do reservatório, como podemos propor um protótipo composto por uma rede de sensores sem fio que monitore estes parâmetros e os envie para uma base central.

Assim, com o desenvolvimento de um protótipo para controle e automação de uma estação de tratamento e distribuição de água utilizando a plataforma Rádiumo teremos uma solução eficiente para o controle do nível e do tratamento de água nos reservatórios de uma companhia de saneamento e abastecimento de água.

Uma vez caracterizado o problema, apresentamos a seguir os objetivos gerais e específicos deste trabalho, no que se refere ao protótipo proposto.

### 1.3 Objetivos Geral e Específicos

O objetivo geral deste trabalho é aprimorar os processos de tratamento e controle dos reservatórios de uma ETA de uma companhia de abastecimento de água e saneamento de um município de pequeno porte.

Como objetivos específicos do trabalho destacamos os seguintes itens:

- a) Aprimorar os conhecimentos sobre IOT e redes de sensores sem fio, investigando como essas tecnologias podem contribuir para automação e controle dos processos de uma ETA.
- b) Estudo das plataformas Arduino e Rádiuino, no intuito de combiná-las com nós sensores e identificar como estas ferramentas podem ser utilizadas no processo de monitoramento do nível e da qualidade da água em reservatórios de uma companhia de abastecimento de água.
- c) Verificar a viabilidade econômica da utilização de um protótipo de gerenciamento de nível e qualidade de água, bem como controle de forma automática, o acionamento de bombas de recalque e faça um monitoramento não intrusivo de vazão d'água.
- d) Minimizar erros existentes nos processos de medições manuais, melhorando a produtividade e um melhor aproveitamento da mão-de-obra.

### 1.4 Solução proposta

Conforme explica-se anteriormente, as principais funcionalidades do protótipo proposto são o monitoramento do nível da água dos reservatórios de uma ETA, controle automático do acionamento de bombas de estações elevatórias de água por meio de atuadores, monitoramento da vazão, bem como o monitoramento de parâmetros de qualidade da água, tais como turbidez, dosagem de cloro, flúor e ferro.

A arquitetura da solução proposta apresenta dois componentes principais:

- a) **Plataforma de Controle e Automação<sup>3</sup>**: composta por um **Ambiente Sensor** e um **Ambiente Base** desenvolvidos em plataforma Rádiuino. O Ambiente Sensor é responsável pela leitura das medições dos sensores de vazão, qualidade e do nível da água; bem como pelo acionamento das bombas dos reservatórios de água. Já o Ambiente Base coletará os dados enviados pelo Ambiente Sensor e os enviará para um *software* de monitoramento. O

---

<sup>3</sup> Composta de um ambiente sensor e um ambiente base, desenvolvido em plataforma Rádiuino.

*hardware*<sup>4</sup> utilizado nos ambientes sensor e base permite uma extensa possibilidade de desenvolvimento, sendo utilizado para automação e o controle em vários cenários, inclusive no tratamento de água em companhias de abastecimento de água. Nesta plataforma proposta, também há a possibilidade, à partir dos parâmetros medidos, fazer a liberação de compostos químicos, como por exemplo, aplicação de flúor, cloro e outras substâncias utilizadas para corrigir os parâmetros que estiverem fora dos padrões de qualidade da água. No entanto, tal funcionalidade não está no escopo deste projeto.

- b) **Software de Monitoramento Desktop**<sup>5</sup>: responsável por centralizar e exibir os dados de medições recebidos do Ambiente Base, que coletou tais medições por meio do Ambiente Sensor. O *software* de monitoramento permite centralizar as informações em uma interface única, de forma a melhorar a interação humano computador. Como trabalhos futuros poderão ser desenvolvidos outras versões do *software* para plataformas móveis (Android, IOS) e plataforma Web.

Portanto, a partir da interação dos Ambientes Sensor e Base com o *Software* de Monitoramento, temos uma solução de automação e controle dos processos de tratamento, bombeamento e controle dos níveis dos reservatórios da companhia de abastecimento de água, com um custo operacional relativamente baixo.

Isto se deve ao fato do uso de uma plataforma de baixo custo e flexível; nós sensores relativamente baratos e o uso de comunicação sem fio entre os diferentes pontos de captação, armazenamento e bombeamento com a central de monitoramento.

Para um melhor entendimento da solução apresentada, a seguir apresentamos a estrutura em que o trabalho foi organizado.

## 1.5 Organização do Trabalho

No Capítulo 2 apresenta-se algumas referências bibliográficas importantes que se relacionam com o trabalho, tais como RSSF, IOT e Plataforma Rádium. No Capítulo 3 apresentamos a metodologia de desenvolvimento do trabalho, que descreve o cenário em que a solução foi proposta, bem como todo o ferramental utilizado: *hardware* (sensores, microcontroladores, atuadores, etc) e *software*. No Capítulo 4 apresentamos detalhes da implementação e validação da solução proposta. Por último no Capítulo 5 são apresentados os resultados e discussões e no Capítulo 6 a conclusão e trabalhos futuros.

<sup>4</sup> Módulos de rádio BE900. Vide: <http://www.radioit.com.br/datasheets/BE900-Datasheet-v1.5.pdf>.

<sup>5</sup> *Software* desenvolvido em Object Pascal na IDE Borland Delphi.

## 2 REFERENCIAL TEÓRICO

As tecnologias de redes de sensores sem fio (RSSF) e *Internet* das Coisas (IOT) estão presentes na Plataforma Rádiumo que utilizamos para prover a solução de controle e automação do tratamento e abastecimento da água.

Sendo assim, nas seções a seguir, antes de descrevermos com mais detalhes a Plataforma Rádiumo, falaremos um pouco mais sobre os temas IOT e RSSF.

### 2.1 *Internet* das Coisas – IOT

O sensoriamento ubíquo<sup>6</sup> habilitado pelas tecnologias da rede de sensores sem fio (RSSF) pode impactar positivamente em várias áreas da vida moderna. Oferecendo a capacidade de medir, inferir e compreender indicadores ambientais e recursos naturais para ambientes urbanos. A proliferação desses dispositivos em uma rede de comunicação e atuadores interconectado cria a *Internet* das coisas (IoT<sup>7</sup>), em que os sensores e os atuadores se misturam perfeitamente com o ambiente que nos rodeia e a informação é compartilhada em plataformas para desenvolver uma imagem operacional comum (GUBBI, 2013).

Segundo Gubbi (2013), a definição de *Internet* de Coisas para ambientes inteligentes é a Interligação de dispositivos de detecção e atuação que oferecem a capacidade de compartilhar informações entre plataformas através de uma estrutura unificada, desenvolvendo uma imagem operacional comum para permitir aplicações inovadoras. Isso é conseguido através do sensoriamento ubíquo e contínuo, análise de dados e representação de informações na nuvem como a estrutura unificadora.

Segundo estudo da CISCO, desde 2015 já existem mais dispositivos conectados à rede do que pessoas. A previsão é que em 2030, 500 bilhões de dispositivos e 7,6 bilhões de pessoas estejam conectados à *internet*. Isto significa uma melhoria da qualidade de vida, maior eficiência produtiva (mais produtos de qualidade com menor custo) está cada vez mais dependente do sensoriamento, coleta, transmissão, armazenamento e distribuição das informações em massa.

---

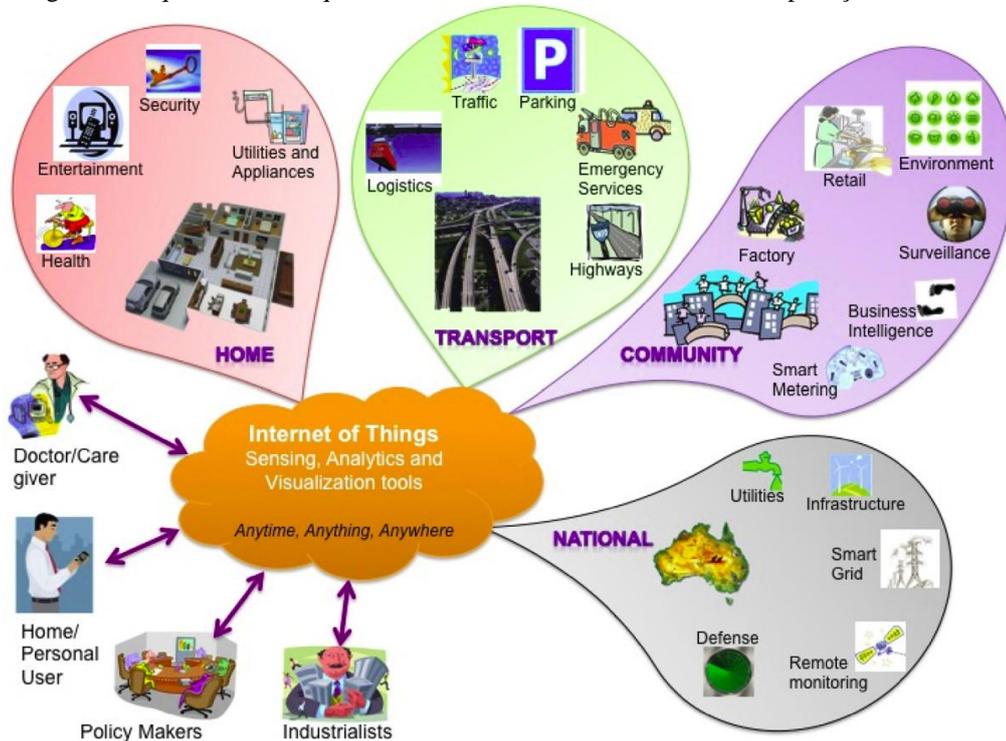
<sup>6</sup> Sensoriamento Ubíquo integra a computação ao ambiente incluindo dispositivos de sensoriamento capazes de monitorar e controlar (dando comandos a um computador) atuadores de uma forma que as pessoas nem percebam que eles estejam atuando.

<sup>7</sup> Do inglês *Internet of Things*.

Existem vários domínios de aplicação que serão impactados pela tecnologia emergente da *Internet das Coisas*. As aplicações podem ser classificadas com base no tipo de disponibilidade de rede, cobertura, escala, e heterogeneidade, repetibilidade, envolvimento e impacto do usuário (GUBBI, 2013).

As aplicações podem ser classificadas em quatro domínios: (1) Pessoal e Doméstico; (2) Empresarial; (3) Serviços de Utilidade Pública; e (4) Mobilidade. Tais aplicações podem ser vistas na Figura 1, que representa a IoT em uma escala de um indivíduo ou casa, em uma comunidade, em escala nacional ou regional e móvel IoT, que geralmente é distribuído em outros domínios, principalmente devido para a natureza da conectividade e da escala (GUBBI, 2013).

Figura 1. Esquema da IOT que mostra os usuários finais e as áreas de aplicação com base em dados.



Fonte: Gubbi (2013).

Existe um grande cruzamento nas aplicações e o uso de dados entre domínios. Por exemplo, O IoT pessoal e doméstico produz dados de uso de eletricidade na casa e disponibiliza para a empresa de eletricidade que, por sua vez, pode otimizar a oferta e a demanda do usuário. A *internet* permite o compartilhamento de dados entre diferentes serviços e fornecedores de forma transparente criando múltiplas oportunidades de negócios (GUBBI, 2013).

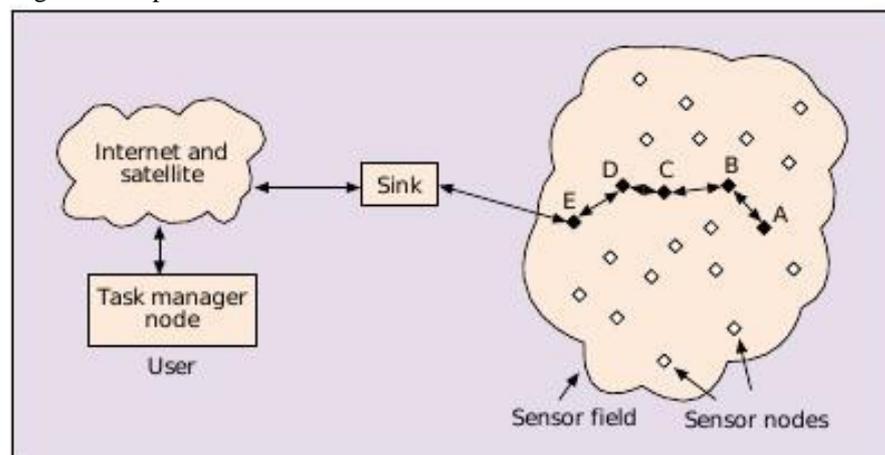
A *Internet* das coisas é constituída de computador, rede local, *internet*, protocolos, base de dados, comunicação e sensores. Uma rede de sensores sem fio (RSSF) é um componente importante para uma IoT. Na seção a seguir falaremos um pouco mais sobre a tecnologia RSSF.

## 2.2 Redes de Sensores sem Fio – RSSF

Segundo Akyildiz (2002), as redes de sensores sem fio (RSSF) são um tipo específico de redes *ad hoc* sem fio com características compostas por dispositivos, chamados nós sensores, que apresentam capacidade de sensoriamento, processamento e comunicação sem fio.

Os nós sensores são geralmente distribuídos em uma região de monitoramento, como mostrado na Figura 2. Cada um destes nós sensores espalhados tem a capacidade de coletar dados e encaminhar os dados de volta para o nó sorvedouro (sink). Os dados são roteados de volta para o nó sink por meio de uma arquitetura de múltiplos saltos sem infra-estrutura através do coletor, como mostrado na Figura 2 (AKYILDIZ, 2002).

Figura 2. Dispersão dos nós sensores.



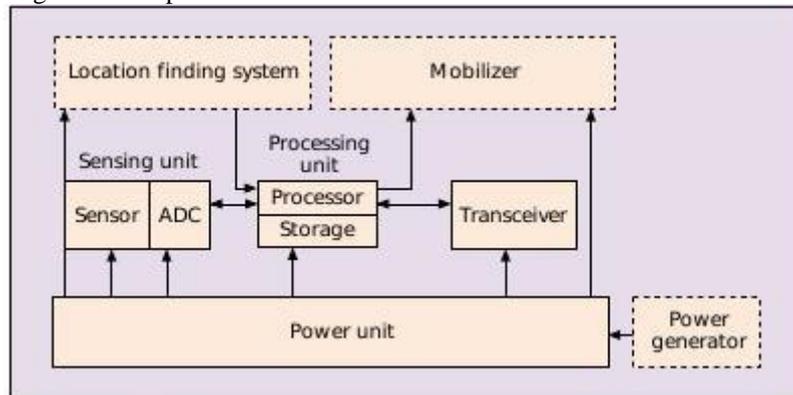
Fonte: Akyildiz (2002).

O projeto da rede de sensores, como descrito na Figura 2 é influenciado por muitos fatores, incluindo tolerância a falhas, escalabilidade, custos de produção, o ambiente operacional, topologia de rede de sensores, as limitações de *hardware*, mídia, transmissão e consumo de energia (AKYILDIZ, 2002).

Os nós sensores são constituídos por um sistema computacional, transceptor sem fio, fonte de energia e dispositivos sensores. O sistema computacional é composto por um

processador e um conjunto de memórias RAM e ROM, conforme Figura 3 (AKYILDIZ, 2002).

Figura 3. Componentes dos nós sensores.



Fonte: Akyildiz (2002).

O transceptor permite a comunicação sem fio, que pode ser via infravermelho, radiofrequência ou laser. A fonte de energia provê energia para o nó sensor operar. Geralmente é uma bateria, mas estudos estão sendo desenvolvidos para se utilizar recursos da natureza para a geração de energia, como o sol e o vento (ROUNDY, 2004). Os dispositivos sensores são os componentes que sensoriam o ambiente, podendo ser de diversos tipos como termistor, umidade, luminosidade, magnetômetro e acelerômetro (AKYILDIZ, 2002).

Os avanços na capacidade de processamento, capacidade de memória e tecnologia de rádio permitiu o desenvolvimento de redes distribuídas com nós de comunicação de baixo custo. Esses nós são capazes de sensoriar, comunicar, e pode ser implementado a um custo muito baixo comparado com um sistema de sensores com fio tradicionais. Tais sistemas são chamados de redes de sensores sem fio (RSSFs). As RSSFs estão enfrentando muitos desafios, incluindo a limitação da largura de banda (ISM). A limitação de largura de banda é destaque devido à crescente demanda de utilização do espectro por vários sistemas sem fio, como Wi-Fi e Bluetooth (AKYILDIZ, 2002).

Redes de Sensores Sem fio (RSSFs) têm sido um tópico de pesquisa há mais de uma década, e a gama de aplicações vão desde o domínio militar, aplicações comerciais, industriais; automação residencial, gerenciamento de energia, etc. Tecnologias sem fio emergentes, tais como Zigbee<sup>8</sup> e IEEE 802.15.4<sup>9</sup>, permitiram o desenvolvimento de produtos

<sup>8</sup> <http://www.zigbee.org/>

<sup>9</sup> <http://www.ieee802.org/15/pub/TG4.html>

interoperáveis comercialmente, que é importante para atender às necessidades de escalabilidade e baixo custo.

Geralmente uma rede de sensores sem fio é usada para atender aplicações específicas e para isto, vários conceitos diferentes devem ser tratados para cada aplicação. Trata-se de uma rede altamente customizada e, portanto, várias questões devem ser definidas para cada projeto: Propagação, *Cross Layere* Consumo. Na essência, a IOT é uma rede de dados e a utilização de uma pilha de protocolos é necessária (KARL; WILLIG, 2007).

O uso de programadores específicos para cada plataforma e a necessidade de se usar protocolos de comunicação dificulta a modelagem de uma rede de sensores. Atuar nas camadas das pilhas de protocolos é um desafio.

Em setembro de 1999, as Redes de Sensores sem Fio (RSSF) foram identificadas pelo *Business Week* como uma das tecnologias mais impactantes para o século XXI. Além disso, em janeiro de 2003 o MIT's *Technology Review* afirmou que as RSSF são uma das dez principais tecnologias emergentes. As RSSF tiveram um crescimento de US\$ 0.45 bilhões em 2011, com uma perspectiva de crescimento acima de US\$ 2 bilhões em 2021.

O padrão 802.15.4 usado pelas RSSF obriga o desenvolvimento de *firmware* para atender às necessidades das aplicações. Várias soluções no mercado utilizam programadores próprios. No entanto, a plataforma Radiumo tem como proposta facilitar a atuação em algumas camadas do protocolo, descomplicando a criação de uma rede de sensores (Souza 2011).

O Arduino quebra os paradigmas do uso de microcontroladores, devido à simplicidade, segundo a revista (IEEE Spectrum) de outubro de 2011. É uma plataforma *open source* com *hardware* barato e *software* baseado na linguagem *Wiring IDE Processing*. O sucesso do Arduino se deve ao fato de que o mesmo apresenta em seu *hardware* uma pinagem própria que abstrai a utilização da pinagem do microcontrolador.

### **2.3 Plataforma Arduino e Radiumo**

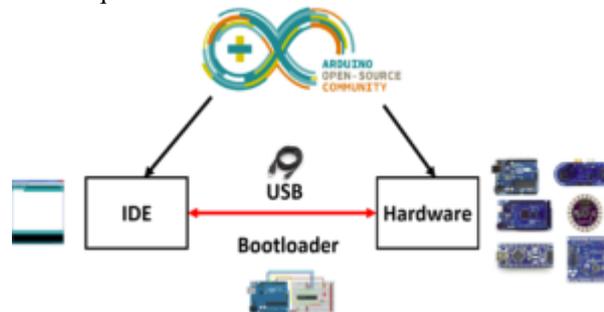
A Plataforma Arduino consiste em um sistema que permite a um usuário com conhecimentos técnicos interagir com o ambiente através de dispositivos. O Radiumo é uma plataforma livre para a criação de rede de sensores sem fio que utilizamos para prover a solução de controle e automação do tratamento e abastecimento de água. Sendo assim, nas seções a seguir, falaremos um pouco mais sobre as Plataformas Arduino e Radiumo.

### 2.3.1 Arduino

O Arduino é um *hardware* com *software* livre, ou seja, não há necessidade de arcar com despesas de royalties para a empresa dona da plataforma, nem a necessidade de aquisição de *softwares* específicos para realizar a programação, pois a plataforma Arduino possui uma IDE grátis para que o usuário possa desenvolver o *firmware* do seu projeto, o que faz com que os custos sejam reduzidos.

A plataforma Arduino consiste em um sistema que permite a um usuário com conhecimentos técnicos interagir com o ambiente através de dispositivos e equipamentos. Segundo Rodrigues (2012), o Arduino foi criado pelo professor Massimo Banzi na Itália, o qual queria ensinar programação de computadores de forma que esse conhecimento pudesse ser aplicado a projetos de arte, automação e robótica.

Figura 4. Esquema Plataforma Arduino

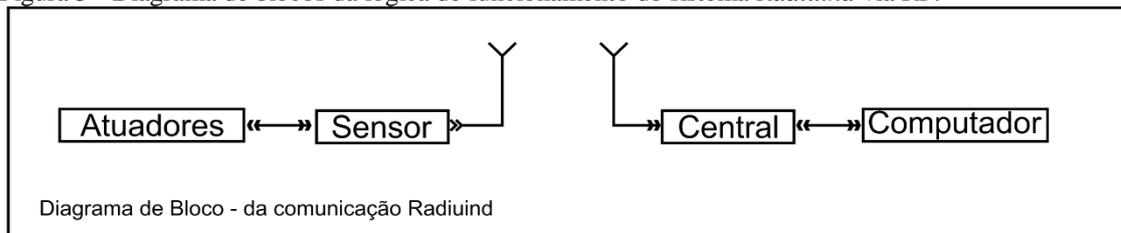


Fonte: Rodrigues (2012).

O Arduino também pode ter suas aplicações estendidas utilizando placas que contêm outros dispositivos, as quais são facilmente conectadas a ele. Estas placas são chamadas de módulos e podem funcionar como receptores GPS, módulos de rede *ethernet* ou *wireless*, dentre outros (MICROBERTS, 2010).

Com toda esta versatilidade temos o ambiente ideal para realizar o presente projeto, pode-se associar o Arduino com outros dispositivos, Segundo Ferroni (2015) embora a plataforma Arduino possui um número extremamente amplo de aplicações, para fins de facilitar ainda mais este processo, o Arduino conta com a possibilidade de se integrar a outras placas cuja programação é feita em micro controlador independente daquele localizado na placa do Arduino. A Figura 5 nos mostra a lógica do *Radiuind* quanto à comunicação RF.

Figura 5 - Diagrama de blocos da lógica de funcionamento do sistema *Radiuind* via RF.



Fonte: Adaptado de Ferroni (2015).

### 2.3.2 Radiuino

O Radiuino é uma plataforma livre para criação de rede de sensores sem fio. Os pesquisadores dessa área buscam o desenvolvimento de *hardware*, *firmware* e *software* que tenham como objetivo criar aplicações de fácil utilização para construção de protótipos rede de sensores sem fio de forma amigável. Permitindo que pessoas, mesmo sem grande experiência em programação, consigam montar seus protótipos descrevendo de uma maneira simples a interação com redes de sensores sem fio (FERRONI, 2015).

Ademais, esta plataforma tem sido amplamente empregada nos mais diversos tipos de aplicação, desde controle de irrigação em jardins (HUGH; JONATHAN, 2009 apud FERRONI, 2015, p.1), controle de dispositivos em residências (HUGH; JONATHAN, 2009 apud FERRONI, 2015, p.1), até treinamento de rede de sensores sem fio no controle de máquinas elétricas (STALLINGS, 2004 apud FERRONI, 2015, p.1).

Sendo assim, nesta seção procuramos explicar um pouco melhor no que consiste esta plataforma. Segundo Branquinho (2014), a plataforma consiste na integração da plataforma Arduino com transceptores<sup>10</sup> de Radiofrequência. Os dispositivos físicos podem ser programados no Ambiente de Desenvolvimento Integrado (IDE) do Arduino, porém incluindo neste ambiente de desenvolvimento algumas bibliotecas específicas<sup>11</sup>.

As bibliotecas Radiuino descrevem o protocolo de comunicação da plataforma e funcionalidades da camada de aplicação (RADIUINO, 2017). As bibliotecas Radiuino já estão inclusas no IDE do Arduino e tipicamente se mostram compatíveis com todas as versões do Arduino, porém é recomendável o uso da versão mais recente.

A plataforma Radiuino é dividida em três partes (BRANQUINHO, 2014):

a) *Hardware*;

<sup>10</sup> Transceptor é um rádio que combina as funcionalidades de transmissão/recepção num único dispositivo.

<sup>11</sup> As bibliotecas para a plataforma Radiuino podem ser baixadas do endereço <http://radiuino.cc/tutoriais/>. Existem duas bibliotecas principais: A primeira delas é chamada de “Radiuino” e a segunda é a “Radiuino4App”. Fonte: <http://radiuino.cc/tutoriais/>

- b) *Firmware* estruturado em pilha de protocolo;
- c) *Software*.

Desta forma, torna-se possível a criação rede de sensores sem fio para monitoração e controle de processos e sua integração com computadores ou *Internet*. Além disso, é uma excelente ferramenta para prova de conceito e o desenvolvimento de protótipos para demonstração. Além disso, segundo Branquinho (2014), destaca-se como objetivos principais da plataforma:

- a) Facilidade de uso para pequenas redes de sensores;
- b) Arquitetura estruturada prevendo as funções necessárias para a criação de rede de sensores fim-a-fim (desde o processo a ser monitorado/controlado até o PC ou a Web);
- c) *Hardware* de baixo custo e facilmente encontrado; possibilidade de desenvolvimento de *hardware* para aplicações específicas;
- d) *Firmware* estruturado com a identificação clara de onde inserir as instruções desejadas; *software* de monitoração e controle (para telemetria, telecomando e telesupervisão);
- e) Integração com a *Internet*;
- f) Adaptado para a *Internet* das Coisas.

Por todos estes motivos elencados acima lançamos mão da plataforma no desenvolvimento de nosso protótipo. No capítulo a seguir explicamos um pouco melhor os materiais e métodos empregados no protótipo proposto (sensores, atuadores, ambiente de desenvolvimento, *hardware*).

### 3 METODOLOGIA

A seguir apresentamos a metodologia de desenvolvimento do trabalho, que descreve o cenário em que a solução foi proposta, bem como todo o ferramental utilizado: *hardware* (sensores, módulos de comunicação, atuadores, etc) e *software*.

Na metodologia proposta identificamos as seguintes etapas:

- a) Estudo e caracterização sobre o local de implantação;
- b) Identificação dos parâmetros a serem monitorados e dos sensores a serem utilizados;
- c) Identificação dos atuadores;
- d) Definição das ferramentas:
  - Ambiente de desenvolvimento integrado a ser utilizado no desenvolvimento das aplicações embarcadas nos ambientes sensor e base (IDE Arduino);
  - Módulo de comunicação por radiofrequência a ser utilizado para comunicação entre os ambientes sensor e base;
  - Ambiente de desenvolvimento integrado a ser utilizado no sistema *desktop* de gerenciamento dos níveis de água e acionamento das bombas ( IDE Borland *Delphi*);
- d) Desenvolvimento da aplicação embarcada no Radiuino:
  - Ambiente Sensor;
  - Ambiente Base;
- e) Desenvolvimento do Sistema *desktop* de gerenciamento dos níveis de água e acionamento das bombas (IDE Borland *Delphi*);
- g) Análise dos resultados.

A seguir, descrevemos melhor cada uma destas etapas, começando por um estudo e caracterização do local onde foi implantada a solução.

#### 3.1 Estudo e caracterização sobre o local de implantação

Cada local possui características topográficas específicas e uma topologia única do sistema de distribuição de água tratada, estes parâmetros são determinantes na configuração da RSSF, como por exemplo, o tipo de sensor a utilizar, o *hardware* que deverá ser usado para acionar as bombas, antenas, potência dos rádios e os tipos de sensores que serão aplicados.

Outro fator a ser observado é a questão a visibilidade da transmissão das informações via rádio. Se o local estiver obstruído por prédios, montanhas ou algum outro tipo de obstáculo deve-se utilizar antenas com ganhos maiores ou utilizar antenas repetidoras de sinal.

Neste trabalho o ambiente a ser monitorado é uma companhia de abastecimento de água e esgoto (SAAE - Serviço Autônomo de Água e Esgoto) do Município de Boa Esperança. O SAAE é uma autarquia municipal que conta com uma Estação de Tratamento de Água no Bairro Alvorada e cinco reservatórios nos bairros Alvorada, Nova Era, Jardim Nova Esperança, Aeroporto e Centro. Além disso, conta com uma ETE (Estação de Tratamento de Esgoto) no Dique da Represa dos lagos dos encantos, conforme pode ser visto no mapa apresentado na Figura 6.

Para caracterizar de forma adequada os locais onde seriam instalados os módulos de comunicação dos

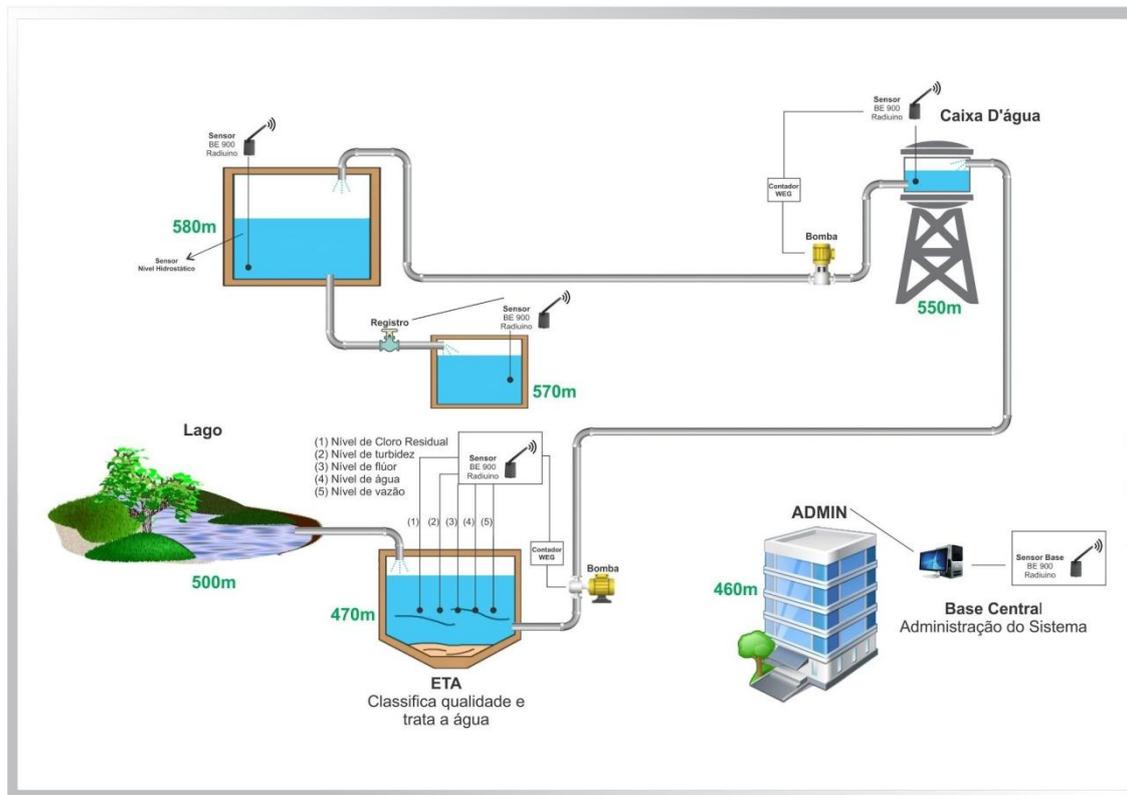
Figura 6 - Mapa do SAAE Boa Esperança MG



Fonte: Google Maps (2007).

Na Figura 7 a seguir apresentamos um esquema dos reservatórios:

Figura 7 - Diagrama Esquemático dos Reservatórios ETA e Bombeamento SAAE Boa Esperança MG



Fonte: o autor.

Também foi feita uma caracterização do ambiente aplicando testes de confiabilidade utilizando dois rádios do tipo BE 900 que serão utilizados como transceptores acoplados aos sensores a serem utilizados para se comunicarem com a base central que transmitirão as informações ao microcomputador.

Nos Testes de confiabilidade foram realizados utilizando dois rádios do tipo BE 900, uma base para conexão ao microcomputador, um nó sensor, uma antena direcional modelo ARC-PAC9113B0 de 12,5 dBi e uma antena *fiberglass omni* direcional de 8 dBi.

### 3.2 Identificação dos Parâmetros a serem monitorados e dos sensores a serem utilizados

Existe no mercado uma enorme variedade de sensores para os mais diversos fins. Neste trabalho fizemos uma investigação prévia de quais os sensores mais adequados para serem utilizado no monitoramento dos parâmetros de qualidade da água, bem como da vazão e do nível dos reservatórios.

Para o monitoramento da qualidade e nível da água, primeiramente identificamos quais são os parâmetros monitorados, e os sensores utilizados. Inicialmente, identificamos a necessidade de medição dos seguintes parâmetros:

- a) Nível de Cloro Residual;
- b) Nível de turbidez;
- c) Nível de flúor;
- d) Nível de água;
- e) Nível de vazão.

Uma vez identificados os parâmetros a serem medidos, o próximo passo foi escolher os sensores que seriam utilizados nas medições. Numa pesquisa, selecionamos os sensores são da marca Nivetec (NIVETEC, 2017) e os contatores da marca WEG, para o acionamento dos motores com corrente elétrica.

Figura 8– Analisador de Cloro Residual Amperométrico.



Fonte: Nivetec (2017).

As medições de cloro estáveis e confiáveis, com pH integrado e compensação de temperatura, permite uma melhor precisão mesmo em temperaturas mais altas (de 0 a 50 °C). Sua segurança é garantida através de um comutador de fluxo integrado que impede a cloração ou adição de produtos químicos quando não houver fluxo.

Além de medições *on-line* precisas, sua manutenção é reduzida (apenas uma vez por ano), economizando visitas técnicas periódicas e demais despesas.

Figura 9 – Medidor de Turbidez.



Fonte: Nivetec (2017).

O Turbidímetro é um equipamento capaz de medir o grau de turvação de vários líquidos. A turvação de um líquido corresponde à concentração de partículas sólidas em seu meio, sendo que o mesmo funciona emitindo um feixe de luz através de uma amostra líquida e a partir disso, as partículas sólidas que estão presentes no líquido refletem o feixe de luz, e assim o Turbidímetro faz a análise da quantidade de partículas no líquido e determina ou não a pureza de determinado produto.

Figura 10 – Medidor de Flúor



Fonte: Nivetec (2017).

O flúor é um dos elementos químicos mais encontrados na natureza e possui grande capacidade de reagir com outros elementos químicos. Desta forma, pode desenvolver compostos orgânicos e inorgânicos. A análise em água para medir o flúor é feita, geralmente, com o uso de um fotômetro ou também conhecido como fotocolorímetro.

Figura 11 – Sensor de Nível Hidrostático.



Fonte: Nivetec (2017).

O transmissor de nível tipo hidrostático é um medidor de nível que transforma a coluna do líquido em sinal eletrônico. Usado para monitorar o nível continuamente de água tratada, esgoto e efluentes industriais.

Figura 12 – Sensor de Vazão.



Fonte: Nivetec (2017).

O medidor de vazão magnético, também conhecido por tensão induzida ou como eletromagnético, é um medidor que utiliza o princípio de medição que se apoia na lei de Faraday, ele continua sendo aperfeiçoado e aprimorado para algumas aplicações. Este medidor de vazão volumétrico, não detém peças móveis e atende um vasto número de aplicações, tais como: medição de vazão de cervejas, ácidos em geral, águas residuais ou em qualquer líquido sujo que seja condutor ou à base de água.

### 3.3 Identificação dos Atuadores

Para o acionamento de motores com corrente elétrica alta, foi utilizado relés/contatores de alta potência, ligados na placa Arduino através das saídas digitais.

Figura 13 – Contator (WEG 2017).



Fonte: Weg (2017).

O contator é um dispositivo eletromecânico que permite, a partir de um circuito de comando, efetuar o controle de cargas num circuito de potência. Essas cargas podem ser de qualquer tipo, de tensão diferente do circuito de comando, e até conter múltiplas fases.

- a) Os contatores podem ser dos tipos principais, que geralmente possuem três contatos NA (normalmente abertos) de potência, dois NA's auxiliares, ou seja, de comando, e mais dois NF's (normalmente fechados) auxiliares, também para o comando;
- b) Também podem ser do tipo auxiliar, que possuem contatos apenas de comando, ou seja, seus contatos suportam uma menor corrente do que os principais;
- c) Vale lembrar que os contatores em geral possuem os chamados blocos aditivos, que são vendidos separadamente, varia de fabricante para fabricante, e têm a função de proporcionar contatos adicionais ao contator (alguns modelos são de acoplamento frontal, e outros de acoplamento lateral).

Vantagens do Contator:

- a) Comando à distância;
- b) Elevado número de manobras;
- c) Grande vida útil mecânica;
- d) Pequeno espaço para montagem;
- e) Garantia de contato imediato;

- f) Tensão de operação de 85 a 110% da tensão nominal prevista para contator;
- g) Maior número de contatos.

### 3.4 Ferramentas

Nesta seção apresentamos as principais ferramentas que foram utilizadas no protótipo proposto, tanto no desenvolvimento dos ambientes sensor e base, quanto no software de monitoramento. Nas subseções a seguir descrevemos o ambiente de desenvolvimento integrado do Arduino utilizado na programação dos ambientes base e sensor; o módulo de comunicação utilizado nos nós sensores (sensor e base) e por último uma descrição do ambiente integrado de desenvolvimento de software utilizado no software *desktop* de gerenciamento dos níveis dos reservatório e do acionamento das bombas.

#### 3.4.1 Ambiente de Desenvolvimento do Arduino

Neste trabalho, tanto Ambiente Sensor quanto o Ambiente Base foram programados utilizando o ambiente de desenvolvimento do Arduino. Os programas em Arduino podem ser divididos em três partes principais: estrutura, valores (variáveis e constantes) e funções. A linguagem Arduino é baseada em C/C++. Para programar, utilizamos o *software* fornecido pelo próprio Arduino:

Figura 14– *Software* de desenvolvimento Arduino



Fonte: Facchin (2015).

#### 3.4.2 Módulo RF BE900

Para transmitir as informações coletadas pelos sensores será utilizado o módulo BE900. O módulo tem dois componentes principais: um microcontrolador e um transceptor o

integrados, que possibilita aplicações de monitoração e controle, podendo ser totalmente programado (BE900, 2017).

No protótipo proposto, o módulo de comunicação BE900 terá um papel fundamental na comunicação de dados e no acionamento dos atuadores, transferindo os dados coletados pelos sensores e enviados para as entradas analógicas do microcontrolador. Em seguida, por radiofrequência, os dados são transmitidos do ambiente sensor para ambiente base e do ambiente base para o software de monitoramento. Outra importante funcionalidade, será a transmissão de dados do ambiente base para o ambiente sensor, enviando sinais para as saídas digitais do microcontrolador onde estarão os elementos atuadores que atuarão no acionamento automático das bombas.

O módulo BE900 é um módulo de comunicação extremamente flexível que utiliza o processador AVR Atmega328 e o transceptor TI CC1101 RF com filtro passa-faixa para maior sensibilidade e imunidade a interferência a ruído, ajustado para operar na banda não licenciada ISM de 915 MHz (BE900, 2017).

O módulo possui modos de operação com baixo consumo de potência, relógio de tempo real e atende às regulamentações da ANATEL e FCC. Como demais características destacam-se as seguintes (BE900, 2017):

- a) IDE Arduino: o BE900 pode ser programado utilizando a IDE Arduino e recorrer a maioria dos *softwares* e bibliotecas desenvolvidos para esta plataforma;
- b) E/S do Arduino: os 15 pinos de E/S disponíveis podem ser mapeados como pinos Arduino;
- c) Microcontrolador: AVR Atmega328, microcontrolador de 8 bits com alto desempenho e baixo consumo, com 32k de ROM, 2k de RAM, 1k de EEPROM e clock de 8MHz;
- d) Comunicação sem fio: o módulo possui um TI CC1101, transceptor de RF em um único chip para bandas não-licenciadas ISM com filtro passa-faixa para maior sensibilidade e menor interferência de ruídos, ajustado para operar na banda de 915 MHz (902- 907,5 MHz e 915-928 MHz);
- e) Flexibilidade de programação: Apesar de poder ser programado via Arduino, o módulo também pode ser programado utilizando a interface AVR, com a vantagem de já ter o transceptor embarcado.

A partir deste ponto utilizaremos o nome de Radiumo para referenciar a plataforma, que é justamente o Arduino destinado a RF (Figura 15).

Figura 15 - Módulo de RF do Arduino (Radiuino).



Fonte: Branquinho (2014).

### 3.4.3 Ambiente de Desenvolvimento do Sistema de Gerenciamento *Desktop*

Para o desenvolvimento do software desktop de gerenciamento do nível de água dos reservatórios e para controle de acionamento das bombas, foi utilizado como ferramenta de desenvolvimento de software o IDE Borland Delphi 7, que por sua vez, permite que a programação seja feita utilizando linguagem pascal orientada a objetos. O IDE Borland Delphi, que permite o uso da linguagem Pascal, gerando executável 100% compilado com desempenho equivalente a programas escritos em linguagem “C”.

Para armazenar as informações, é utilizado o Sistema de Gerenciamento de Banco de Dados (SGDB) Mysql, que permite acesso multiusuário, possui nativamente acesso as principais linguagens de desenvolvimento web (como PHP) e é disponibilizado sob licença *freeware*.

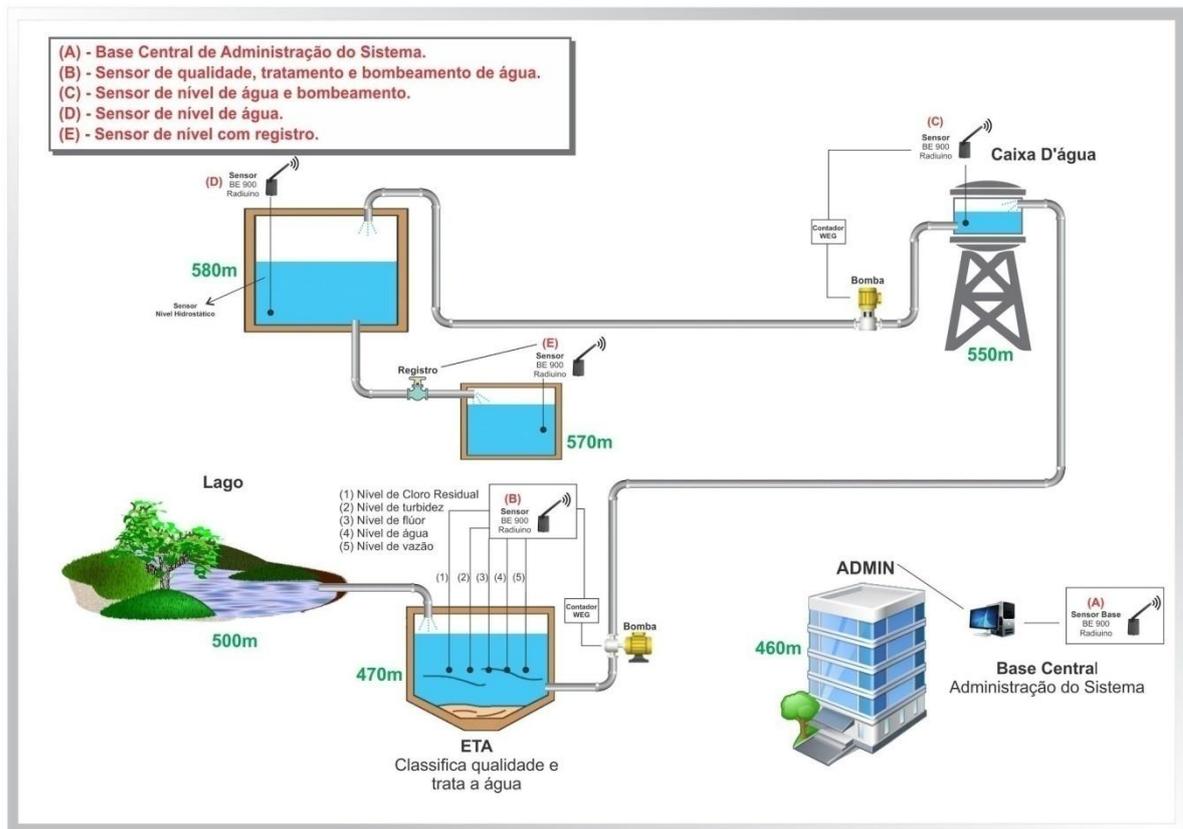
No Anexo A deste trabalho encontra-se o código fonte do software de gerenciamento do nível de água. Na seção 4.2 explicamos melhor o funcionamento do software de gerenciamento dos níveis dos reservatórios e acionamento das bombas de recalque.

## 4 PROTÓTIPO PROPOSTO

A implementação foi feita em cada reservatório de água utilizando sensores de nível hidrostático (Figura 11), sensores de vazão magnético (Figura 12), sensores de medição da qualidade da água: Analisador de Cloro Residual Amperométrico (Figura 8), Medidor de Turbidez (Figura 9) e Medidor de Flúor (Figura 10) e contadores para realizar o acionamento de todas as bombas. Todos estes sensores são da marca Nivetec e os contadores é da marca WEG estão melhores descritos na seção 3.2 deste trabalho.

Os sensores de nível, vazão e qualidade da água (turbidez, cloro e flúor) foram acoplados às entradas analógicas do módulo de comunicação BE900 (Figura 14). Os atuadores e contadores utilizados no acionamento das bombas também foram acoplados a um módulo de comunicação BE900 e recebem os comandos de liga/desliga por meio das entradas digitais do módulo. Tal ambiente, denominaremos a partir de agora de Ambiente Sensor(conforme apresentado na seção 3.5.1). O Ambiente Sensor envia os dados coletados para o Ambiente Base que também é constituído por um módulo de comunicação BE900 (conforme apresentado na seção 3.5.2). O módulo da base central coleta as informações dos sensores via radiofrequência e as disponibilizam para o *software* de gerenciamento que fica instalado em um servidor de dados na sede Administrativa da companhia de abastecimento de água e esgoto. A Figura 16 ilustra o diagrama da solução proposta:

Figura 16 - Diagrama da solução: Comunicação por de RF da rede de sensores sem fio com a base central a tela do software de gerenciamento dos níveis de água, tratamento e acionamento das bombas de recalque.



Fonte: o autor.

## 4.1 Plataforma de Automação e Controle (Ambiente Sensor e Ambiente Base)

Para desenvolver a aplicação embarcada no Radiuino, foram desenvolvidos dois tipos de ambientes: uma para a base e outra para o sensor. A base é a interface entre o computador e os sensores espalhados na rede de sensores sem fio, e o sensor é cada local onde ficará o monitoramento do nível, qualidade e vazão da água, bem como o controle automático de acionamento das bombas para abastecimento dos reservatórios.

### 4.1.1 Ambiente base

O desenvolvimento da aplicação para esse módulo reflete a interação entre o usuário, a automação e os equipamentos distribuídos na rede sem fio. Toda informação que chegar através da rede é enviado para a porta de entrada de dados via porta serial (Com Virtual), permitindo ao aplicativo desktop desempenhar sua função de monitoramento e automação. O código que permite tal operação é apresentado no (Anexo B) – Código para Radiuino Base.

#### 4.1.2 Ambiente Sensor

Os módulos Radiuino que são instalados para monitoramento dos reservatórios d'água e acionamento das bombas são chamados de sensores. Eles transmitem os dados para a base em pacotes de dados de 52 bytes cada.

A implementação desenvolvida para um sensor de aferição de nível de água é exibida no (Anexo C) Código para Radiuino Sensor

Existe a necessidade de criar dois tipos de *software* embarcado: base e sensor.

Base é o nome escolhido para o *hardware* ligado ao computador central, através de uma saída USB, que cria uma porta Serial Virtual, que permite ao *software Desktop* desenvolvido em *Delphi*, receber as informações dos demais pontos de monitoramento.

Sensor é o nome escolhido para o *hardware* que ficará ligado a caixas d'água e motores das bombas que transportam água para os demais reservatórios.

Toda informação que é recebida pela base, é repassada diretamente ao computador da central, que ficará responsável por determinar quais ações devem ser empregadas, tendo como referência todas as informações que acabou de receber. Ou seja, a base trabalha de forma transparente no envio/recebimento das informações.

A comunicação entre a base e o sensor é feita sempre por blocos de 52 *bytes*. Assim sendo, é possível definir que determinado *byte* seja o identificador do destinatário do bloco de informações que está sendo enviado pela base, ou que outro *byte* específico seja responsável por exibir o status atual de ligado/desligado do motor, ou mesmo informações de sensores analógicos. Isso se deve ao fato de que toda informação enviada é “escutada” por todos os sensores que estiverem no rádio de comunicação da base.

A ordem de envio dos *bytes* fica a critério do programador do sistema. Como padrão, é utilizado o espaço entre o *byte* 0 e o 7 como cabeçalho de quem deve receber o pacote. Do *byte* 8 ao 15 define-se a leitura das portas analógicas e digitais e do *byte* 16 em diante, estão disponíveis para futuras implementações.

#### 4.2 Sistema de Gerenciamento de Gerenciamento *Desktop*

O Sistema de Gerenciamento *Desktop* é responsável por monitorar o nível dos reservatórios e pelo acionamento das bombas. Para tanto, as informações recebidas dos pontos de monitoramento e distribuição devem ser centralizadas, preferencialmente na ETA e deve servir de base para iniciar todo o processo. O *software desktop* permite aos operadores locais

o acesso a toda informação sobre a funcionalidade da rede, oferecendo uma análise gráfica aos supervisores e diretores.

No Anexo A deste trabalho encontra-se o código fonte do Sistema de Gerenciamento dos Níveis de água dos reservatórios. A seguir descrevemos as principais funcionalidades disponibilizadas pelo sistema e algumas telas demonstrando a interface com o usuário.

#### 4.2.1 Principais Funcionalidades do Sistema de Gerenciamento

Como principais funcionalidades do sistema de gerenciamento dos níveis de água dos reservatórios e acionamento das bombas de recalque, destacam-se as seguintes:

- a) Cadastro de Reservatórios;
- b) Cadastro de Bombas de Recalque;
- c) Monitoramento gráfico níveis dos reservatórios;
- d) Acionamento automático e/ou manual das bombas de recalque;
- e) Permissões de acesso Supervisor/Operadores;
- f) Histórico de eventos;
- g) Relatórios para acompanhamento operacional e gerencial;
- h) Análise dos dados remotamente via *internet* (\*);
- i) Log de acessos (\*);
- j) Monitoramento on-line do nível dos reservatórios;
- k) Monitoramento níveis de cloro, manganês, turbidez, flúor, ferro (\*);
- l) Monitoramento da vazão de água na tubulação (no projeto piloto, essa opção não está disponível);
- m) Vários gráficos para facilitar o entendimento dos históricos (\*);
- n) Acionamento remoto das bombas de recalque *interface desktop* para os operadores locais;
- o) *Interface Web* para consulta de resultados por parte da direção, podendo acessar via *smartphone* (\*);
- p) Cadastro de usuários com diversos níveis de prioridade (\*).

\* no projeto piloto, essa opção está disponível

No intuito de mostrar a interação com o usuário na subseção a seguir apresentamos algumas de telas do sistema.

#### 4.2.2 Telas de Funcionamento e Modos de Operação do Sistema

O sistema opera em modo *online* e modo *offline*. A seguir apresentamos algumas telas exemplificando o funcionamento do sistema. Na Figura 17 podemos ver o sistema operando em modo online.

No modo de operação online é possível realizarmos todas as medições em tempo real. Por meio da tela de monitoramento é possível verificar a quantidade exata de litros de água contida no reservatório e a quantidade que falta para chegar no nível máximo. Também possível verificar o estado da bomba (ligada/ desligada).

Outra característica do sistema é a possibilidade de parametrização. Para tanto, o sistema conta com uma tela de configuração, na qual é possível alterar em tempo real os parâmetros referentes ao nível do reservatório, e também agendar o horário de ligar/desligar as bombas.

Na tela de monitoramento, também há a opção de acionamento manual das bombas (ícone amarelo da Figura 18). Com um clique é possível fazer o acionamento manual das bombas, ou seja, pelo software é possível enviar qualquer configuração que se deseje ao Ambiente Sensor. Obviamente, claro que isto só será possível se houver comunicação entre o Ambiente Base e o Ambiente Sensor e comunicação na porta serial do PC com o Ambiente Base.

Figura 17 - Tela do *Software* em operação em modo online



Fonte: o autor.

O sistema também opera no modo *offline*, a Figura 18 mostra o Software de gerenciamento em modo *offline*, ou seja, sem comunicação com o Ambiente Base e o Ambiente Sensor. Este modo de operação foi desenvolvido para permitir o funcionamento independente e autônomo dos ambientes sensor e base, mesmo em caso de algum evento de indisponibilidade, problemas de comunicação ou conexão com o servidor de banco de dados.

O modo *offline*, pode atuar também caso haja algum problema de conectividade, ocasionado, por exemplo, por alguma tempestade, que pode provocar o desalinhamento das antenas. Nestes casos, o software emite um sinal que está sem comunicação, o módulo sensor funciona normalmente gerenciando o ligamento e o desligamento sem a necessidade do servidor.

Na Figura 18 abaixo são mostrados os parâmetros de configuração do sistema, bem como o alerta de que o sistema está *offline*.

Figura 18-Tela do Software em operação em modo *offline*



Fonte: o autor.

## 5 RESULTADOS E DISCUSSÃO

Após a implantação do projeto, foi possível coletar os dados monitorados, bem como transmiti-los de forma *on-line* e permitir o acionamento das bombas hidráulicas no exato momento que foi configurado.

Além disso, todo o sistema passou a ser operado sem intervenção humana, ou seja, o sistema é autônomo na coleta e distribuição de dados, sendo necessárias apenas manutenções preventivas e corretivas. Lembrando que os sensores são dispositivos discretos que não interferem em nada na distribuição de água, nem no seu tratamento.

O projeto opera de forma autônoma e fornece informações de seu funcionamento de forma online. Todos os dados são transmitidos via rádio frequência para o Ambiente Base desenvolvido no Rádiuino, o qual faz o tratamento dos dados para gerar os relatórios exigidos pelo cliente.

O protótipo também veio contribuir para a melhoria no tratamento e distribuição nas estações de tratamento de água, na medida em que permitiu um aprimoramento na automação dos processos, sobretudo o controle de nível e de qualidade da água. Tal automação traz uma contribuição essencial para a sociedade, na medida em que evita os desperdícios e oferece uma qualidade melhor para água.

Outra contribuição do trabalho foi utilizar plataforma Rádiuino, que se mostrou uma plataforma de baixo custo, eficiente e que tornou o projeto economicamente viável. Além disto, o domínio do *software* empregado fica sob responsabilidade de quem estiver implantando o sistema, possibilitando o uso de equipamentos com custos mais acessíveis e com garantia funcional de toda a solução

No que se refere à economia de insumos, foram obtidos junto à diretoria da Companhia de Abastecimento de Água as seguintes informações

- a) **Redução no custo da fatura de energia elétrica:** houve uma redução no consumo de energia elétrica na ordem de aproximadamente 29% após a implantação da solução proposta;
- b) **Otimização dos recursos humanos:** no processo anterior de acompanhamento do bombeamento de água para as estações elevatórias estavam alocados 20 trabalhadores que trabalhavam em turnos e eram responsáveis pelo acionamento manual das bombas, tinham a incumbência de ligar e desligar as bombas, acompanhando o funcionamento 24 horas por dia, 7 dias por semana;

- c) **Economia de insumos no tratamento de água:** houve uma economia de aproximadamente de 43% nos produtos de tratamento como ozônio, peróxido de hidrogênio permanganato de potássio, raios ultravioleta, cloro e seus componentes.
- d) **Diminuição do volume de água que deixou de ser desperdiçado:** em termos volume de água desperdiçado, com a implantação do sistema há a possibilidade de se evitar o desperdício de um volume de 1.000 litros de água por minuto em virtude do transbordo dos reservatórios.

Tais resultados ratificam a viabilidade econômica do projeto e a melhoria nos processos de tratamento da água, na medida em que melhoraram a qualidade da água e reduzem os custos financeiros com a aquisição de produtos químicos.

Na Tabela 01 são apresentados o comparativo dos gastos com o tratamento da água, antes e depois da implantação do protótipo.

Tabela 01 - Comparativo dos gastos

Item	Sistema Antigo	Sistema Novo
Contrato profissional para operação	R\$ 40.000,00	0,00
Energia Elétrica	R\$ 15.000,00	R\$ 5.000,00
Produtos Tratamento de Água	R\$ 200.000,00	R\$ 120.000,00
Desgaste bombas (conserto manutenção)	R\$ 50.000,00	R\$ 12.000,00

\*Os valores são mensais e aproximados.

FONTE: o autor

## 6 CONCLUSÃO E TRABALHOS FUTUROS

Neste trabalho é proposto um protótipo de gerenciamento do nível e da qualidade da água, bem como o acionamento automático do bombeamento de água da companhia de água e esgoto do município de Boa Esperança, utilizando a plataforma Rádiuino.

O foco principal do projeto é à automação e controle dos níveis de água nos reservatórios e o controle da qualidade da água por meio de uma rede de sensores sem fio. Sendo assim, o grande desafio é automatizar o processo de coleta dos parâmetros qualidade da água (cloro, flúor, turbidez), vazão e de níveis dos reservatórios. Para tanto é desenvolvido um protótipo composto por uma rede de sensores sem fio para monitorar estes parâmetros e os enviá-los para uma base central.

Partimos da hipótese de que o desenvolvimento de um protótipo para controle e automação de uma estação de tratamento e distribuição de água utilizando a plataforma Rádiuino seria uma solução eficiente para o controle do nível e do tratamento de água nos reservatórios e na ETA do SAAE de Boa Esperança - MG.

Antes da solução proposta, não havia um processo muito eficiente no gerenciamento do nível dos reservatórios e no acionamento das bombas. Devido a isso, ocorriam desperdícios de água e o acionamento das bombas era feito de forma manual.

Nessas situações, além dos desperdícios de água, ocorria o desperdício de energia elétrica e produtos para tratamento da água. Os profissionais que ficavam responsáveis pelo ligamento e desligamento das bombas não o faziam adequadamente, como consequência, acontecia o transbordamento de água pela cidade.

Desta forma, em um efeito cascata, o maior prejudicado é a população devido à falta de água proveniente de uma técnica ultrapassada de gerenciamento. Os produtos de tratamento de água eram aplicados de forma inadequada, sem a dosagem correta, afetando assim o nível da qualidade da água da cidade.

Com testes feitos com o protótipo, verificou-se que os principais objetivos do trabalho foram atingidos. Uma vez que foi possível realizar um monitoramento mais eficiente do nível de água dos reservatórios, controle do acionamento de bombas de estações elevatórias de água, monitoramento da vazão por meio de macro medição não intrusiva, bem como o monitoramento de parâmetros de qualidade da água, por meio da leitura de medições dos sensores hidrostáticos de turbidez, dosagem de cloro e flúor.

Além disto, após a implantação do protótipo proposto, foram alcançados como resultados a redução dos valores de consumo de energia, economia no uso de produtos químicos utilizados no tratamento da água e na diminuição da mão de obra empregada, eliminando assim os desperdícios que antes ocorriam na estação de tratamento da cidade.

Em termos percentuais conseguimos uma redução de 29% no custo da fatura de energia elétrica, também houve uma diminuição de 20 trabalhadores que ficavam por conta de ligar e desligar as bombas 24 horas por dia, 7 dias por semana. Outra economia gerada foi na aquisição de produtos químicos, aproximadamente 43% de redução nos custos com produtos de tratamento como Ozônio, Peróxido de hidrogênio, Permanganato de potássio, Raios ultravioletas, Cloro e seus componentes. Comprovando assim uma viabilidade econômica capaz de aumentar a qualidade da água e de reduzir os custos financeiros.

Os principais obstáculos encontrados no projeto foram convencer a Instituição SAAE a acreditar na idéia proposta, liberação de recursos para o investimento, disponibilização de tempo dos profissionais envolvidos durante a implementação. Também foi o grande desafio o contato com temas tão inovadores como IoT e redes de sensores sem fio, o que impactou na realização de várias pesquisas bibliográficas necessárias ao desenvolvimento e implantação do protótipo. Também constituiu-se um processo trabalhoso as medições de tamanho e distância entre os locais de captação, armazenamento e bombeamento de água, bem como a investigação dos sensores e antenas mais adequados.

Por fim, pode-se dizer que o desenvolvimento do protótipo é uma oportunidade ímpar de colocar em prática conhecimentos de IoT, redes de sensores sem fio, controle e automação e desenvolvimento em plataforma Rádium. O protótipo encontra-se ainda num estágio inicial, mas já atende minimamente o aprimoramento dos processos de monitoramento dos reservatórios e do tratamento de água de uma companhia de abastecimento água.

Como trabalho futuro pretende-se desenvolver novas versões do *software* de gerenciamento em plataformas *Web* e plataformas móveis *Android* e *IOS*, visando ter uma gama maior de ferramentas de monitoramento e possibilitando a mobilidade dos responsáveis pelo monitoramento.

Outro potencial ponto de melhoria no protótipo proposto será a inclusão de uma funcionalidade que possibilite à partir dos parâmetros medidos, fazer a liberação de compostos químicos, como por exemplo, aplicação de flúor, cloro e outras substâncias utilizadas para corrigir os parâmetros que estiverem fora dos padrões de qualidade da água. Neste sentido, pretende-se desenvolver no ambiente sensor a liberação automática através de

válvulas solenóides dos produtos químicos (cloro, flúor, etc), com base nas medições e nos parâmetros mínimos de qualidade da água.

Portanto, pode-se concluir que o protótipo proposto, ainda que esteja num estágio inicial, pôde contribuir de sobremaneira para melhoria dos processos de monitoramento do nível, vazão e qualidade da água da companhia de abastecimento de água do município de Boa Esperança. Uma vez que, melhorou a eficiência dos processos de tratamento e bombeamento da água; gerou economia de gastos com energia, produtos químicos e mão de obra; evitando assim o desperdício de insumos.

Logo, pode-se dizer que os objetivos traçados inicialmente neste projeto foram alcançados e o protótipo proposto se mostrou uma ferramenta inovadora de promoção da sustentabilidade socioeconômica e ambiental no âmbito do cenário em foi aplicada.

## REFERÊNCIAS

- AKYILDIZ, Ian F. et al. Wireless sensor networks: a survey. **Computer networks**, v. 38, n. 4, p. 393-422, 2002.
- ANATEL. **Regulamentação Rádio Frequência**, 2007. Disponível em: <<http://www.anatel.gov.br/setorregulado/regulamentacao-radiofrequencia>>. Acesso em: 10 fev. 2017.
- A, R. Paixão, A, C. Uzêda, D, D. Dias, M, A. Duarte, S. Amorim, H, S. A Placa Arduino: Uma Opção De Baixo Custo Para Experiências De Física Assistidas Pelo Pc. **Revista Brasileira De Ensino De Física**, v. 33, n. 1, 1702, 2011.
- BE900. **Datasheet do Módulo de comunicação por RF BE900**, 2017. Disponível em: <<http://www.radioit.com.br/datasheets/BE900-Datasheet-v1.5.pdf>>. Acesso em: 10 fev. 2017.
- BLACK, P. O. "**Bombas**". Rio De Janeiro: Ao Livro Técnico Sa, 1979.
- BRANQUINHO, O. C.. **Plataforma Rádiuino**: Para Estudo de Rede de Sensores Sem Fio, 2014. Disponível em: <<http://www.foruns.unicamp.br/ArquivosBibliotecaVirtual/Palestras/21-10/PlataformaRadiuinov14.pdf>>. Acesso em: 26 mar. 2017.
- CISCO. **Internet of Things**, 2015. Disponível em: <[www.cisco.com/go/iot](http://www.cisco.com/go/iot)>. Acesso em: 25 mar. 2017.
- FACCHIN, Martino. **Arduino Ide 1.6.6 Released And Available For Download**. Blog Arduino, 2015. Disponível em: <<https://blog.arduino.cc/2015/11/03/arduino-ide-1-6-6-released-and-available-for-download/>>. Acesso em: 27 mar. 2017.
- FERRONI, Eduardo H. - Controle de Demanda de Energia Baseado em Sistemas Supervisório e Rede de Sensores sem Fio. In.: **ANAIS DO XX CONGRESSO BRASILEIRO DE AUTOMÁTICA**, Belo Horizonte, 2014.
- FERRONI, E. H. et al. A plataforma Arduino e suas aplicações. **Revista UIIPS**, v. 3, n. 2, p. 133-148, 2015.
- GARCEZ, L.N. - "**Elementos De Engenharia Hidráulica e Sanitária**". São Paulo: EdgardBlucherLtda, 1969.
- GUBBI, Jayavardhana et al. Internet of Things (IoT): A vision, architectural elements, and future directions. **Future generation computer systems**, v. 29, n. 7, p.1645-1660, 2013.
- KARL, Holger; WILLIG, Andreas. **Protocols and architectures for wireless sensor networks**. New York: John Wiley& Sons, 2007.
- LUQUETA, G. R., D. BRANQUINHO, O.C. **Plataforma Rádiuino**. Campinas: UNICAMP, 2014.
- MICROBERTS, M. **Beginning Arduino**. Nova Iorque: Copyright, 2010.

MOURA A. J. A crise hídrica no Brasil: A água como elemento raro e caro. **Revista Científica Eletrônica**, 2015. Disponível em: <<http://www.facimed.edu.br/site/revista/?onChange=Ler&ID=70>>. Acessado em: 01 de jun. 2017

NIVITEC. **Sensores de automação**, 2017. Disponível em:<<https://nivetec.com.br/>>. Acesso em: 22 mar. 2017.

RADIUINO. **Bibliotecas Radiumino**, 2017. Disponível em: <<http://radiuino.cc/tutoriais/>>. Acesso em: 25 mar. 2017.

RAPPAPORT, **Theodore S. Wireless Communications: Principles And Practice**. 2. ed. Dorling Kindersley, 2009.

RODRIGUES, L et al. **Introdução ao Arduino**. Campo Grande: Fundação Universidade Federal do Mato Grosso do Sul - UFMS, 2012.

## Anexo A ( Código fonte do monitor de nível)

```

Unit Monitor Nivel;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  StdCtrls, CPort, ExtCtrls, pngimage, JvExControls, JvGradient, ComCtrls,
  inifiles, Buttons;

type
  TSensorAD = record
    valor : byte;
    minimo : byte;
    maximo : byte;
  end;

type
  TForm1 = class(TForm)
    ComPort1: TComPort;
    GroupBox1: TGroupBox;
    Shape1: TShape;
    shapead0: TShape;
    GroupBox2: TGroupBox;
    Shape2: TShape;
    shapead1: TShape;
    GroupBox3: TGroupBox;
    Image1: TImage;
    imgBombaDesligadaAD1: TImage;
    imgBombaLigadaAD1: TImage;
    Label11: TLabel;
    Image4: TImage;
    imgBombaDesligadaAD0: TImage;
    imgBombaLigadaAD0: TImage;
    Label12: TLabel;
    Label13: TLabel;
    Label14: TLabel;
    Label15: TLabel;
    Label16: TLabel;
    Label17: TLabel;
    Label18: TLabel;
    Label19: TLabel;
    Label20: TLabel;
    Label21: TLabel;
    Label22: TLabel;
    Label23: TLabel;
    Panell1: TPanel;
    Label24: TLabel;
    Label25: TLabel;
    Label26: TLabel;
    Label27: TLabel;
    Image7: TImage;
    Panel2: TPanel;
    Label28: TLabel;
    Timer1: TTimer;
    Label10: TLabel;
    Label9: TLabel;
  end;

```

```

    JvGradient1: TJvGradient;
tmVerificaSeEstaLigado: TTimer;
    Image2: TImage;
    Panel3: TPanel;
tmPisca: TTimer;
    SpeedButton1: TSpeedButton;
ckRele: TCheckBox;
ckLedVermelho: TCheckBox;
ckLedAmarelo: TCheckBox;
procedure Button1Click(Sender: TObject);
procedurebtEnviaComandoClick(Sender: TObject);
procedure ComPort1RxChar(Sender: TObject; Count: Integer);
procedureFormCreate(Sender: TObject);
procedureFormShow(Sender: TObject);
procedure Timer1Timer(Sender: TObject);
proceduretmVerificaSeEstaLigadoTimer(Sender: TObject);
proceduretmPiscaTimer(Sender: TObject);
procedure SpeedButton1Click(Sender: TObject);
private
ultimopacote : TDatetime;
podeenviar : boolean;
sensorAD0 :TSensorAD;
sensorAD1 :TSensorAD;

EncherAd0 :boolean;
EncherAd1 :boolean;

EnchendoAd0 :boolean;
EnchendoAd1 :boolean;

vgSaida : array[0..51] of byte;
vgentrada : array of byte;
proceduretratavalores;
proceduregravalog(entrada : string);
{ Private declarations }
public
{ Public declarations }
end;

var
    Form1: TForm1;
vgArquivoini :Tinifile;
vgdebug :boolean;
vgpath : string;

implementation
uses
uconfig;
{$R *.DFM}

procedure TForm1.Button1Click(Sender: TObject);
begin
ifvgArquivoini.ReadString('CONFIG','PORTACOM','')<>' ' then
begin
comport1.Port := vgArquivoini.ReadString('CONFIG','PORTACOM','');
comport1.open;
end;
end;

procedure TForm1.gravalog(entrada: string);
var

```

```

f :textfile;
begin
if not vgdebug then exit;
assignfile(f,vgpath+'log.txt');
if not fileexists(vgpath+'log.txt') then rewrite(f) else append(f);
writeln(f,formatdatetime('hh:nn:ss,zzz',now)+' - '+entrada);
closefile(f);
end;

procedure TForm1.btEnviaComandoClick(Sender: TObject);
var
i :integer;
begin
for i := 0 to 51 do vgsaida[i] := 0;

vgsaida[8] := 1; // endereço q vai responder

ifckrele.checked then vgsaida[23] := 1;
ifckLedVermelho.checked then vgsaida[26] := 1;
ifckLedAmarelo.checked then vgsaida[29] := 1;
while not podeenviar do application.processmessages;
ComPort1.Write(vgSaida,52);

end;

procedure TForm1.tratavalores;
var
variacao : integer;
porcentagem : integer;
tamanhoshape : integer;
minimoamarelo : integer;
minimovermelho : integer;
litros : integer;

minimoliga, maximodesliga : integer;
begin

tamanhoshape := 300;
litros := 5;
minimoamarelo := vgArquivoini.ReadInteger('CONFIG','minimoamarelo',50);
minimovermelho := vgArquivoini.ReadInteger('CONFIG','minimovermelho',30);
minimoliga := vgArquivoini.ReadInteger('CONFIG','minimoliga',25);
maximodesliga := vgArquivoini.ReadInteger('CONFIG','maximodesliga',90);

variacao := sensorAD0.maximo - sensorAD0.minimo;
try
porcentagem := ((sensorad0.valor-sensorAD0.minimo) * 100) divvariacao;
except
porcentagem := 0;
end;
if porcentagem < 0 then porcentagem := 0; // pra tratar leituras falsas
porcentagem := 100-porcentagem; // inverter porque a leitura 0 é full

shapead0.Height := porcentagem * tamanhoshape div 100;
shapead0.brush.color := $00C08000;
ifporcentagem<minimoamarelo then shapead0.brush.color := clyellow;
ifporcentagem<minimovermelho then shapead0.brush.color := clred;
ifporcentagem<= minimoliga then EncherAd0 := true;
ifporcentagem>= maximodesliga then EncherAd0 := false;
if shapead0.height<0 then shapead0.height := 0;

```

```

shapead0.top := tamanhoshape + shape2.Top - shapead0.height;
Label15.caption := formatfloat('0.0',litros-(litros*porcentagem/100)) + ' x
litros';

variacao := sensorAD1.maximo - sensorAD1.minimo;
try
porcentagem := ((sensorad1.valor-sensorAD1.minimo) * 100) divvariacao;
except
porcentagem := 0;
end;
if porcentagem < 0 then porcentagem := 0; // pra tratar leituras falsas
Label21.caption := formatfloat('0.0',litros-(litros*porcentagem/100)) + ' x
litros';

porcentagem := 100-porcentagem;
shapead1.Height := porcentagem * tamanhoshape div 100;
shapead1.brush.color := $00C08000;
ifporcentagem<minimoamarelo then shapead1.brush.color := clyellow;
ifporcentagem<minimovermelho then shapead1.brush.color := clred;
ifporcentagem<= minimoliga then EncherAd1 := true;
ifporcentagem>= maximodesliga then EncherAd1 := false;
if shapead1.height<0 then shapead1.height := 0;
shapead1.top := tamanhoshape + shape2.Top - shapead1.height;

label14.caption := inttostr(litros) + ' litros';
label20.caption := inttostr(litros) + ' litros';

end;

procedure TForm1.ComPort1RxChar(Sender: TObject; Count: Integer);
var
c : char;
i :integer;
s : string;
shexa : string;
begin
ultimopacote := now+ strtotime('00:00:20');
podeenviar := false;
    ComPort1.ReadStr(s,count);
shexa := '';
for i := 1 to length(s) do
begin
shexa := shexa + inttohex(ord(s[i]),2)+' ';
end;

gravalog('<<<< '+shexa);
while s <> '' do
begin
setlength(vgentrada,length(vgentrada)+1);
vgentrada[length(vgentrada)-1] := ord(s[1]);
s := copy(s,2,length(s));
end;

timer1.enabled := false; // interface de sincronismo
timer1.enabled := true;

```

```

end;

procedure TForm1.FormCreate(Sender: TObject);
begin
  vgpath := extractfilepath(paramstr(0));
  ifvgpath[length(vgpath)] <> '\' then vgpath := vgpath + '\';
  vgarquivoini := Tinifile.create(vgpath+'config.cfg');

end;

procedure TForm1.FormShow(Sender: TObject);
begin
  vgdebug := vgarquivoini.readbool('config','debug',false);
  label9.visible := vgdebug;
  label10.visible := vgdebug;
  EncherAd0 := false;
  EncherAd1 := false;
  EnchendoAd0 := false;
  EnchendoAd1 := false;

  sensorAD0.maximo:=
  strtoint('$'+vgarquivoini.readstring('config','ADOMAXIMO','17'));
  sensorAD0.minimo:=
  strtoint('$'+vgarquivoini.readstring('config','ADOMINIMO','04'));
  sensorAD1.maximo:=
  strtoint('$'+vgarquivoini.readstring('config','AD1MAXIMO','17'));
  sensorAD1.minimo:=
  strtoint('$'+vgarquivoini.readstring('config','AD1MINIMO','04'));
  Image7.Left := width - Image7.width-20;
  Image2.LEFT := WIDTH - IMAGE2.WIDTH - 20;
  PANEL3.LEFT := IMAGE2.LEFT - PANEL3.WIDTH - 30;
  SpeedButton1.left := width - SpeedButton1.width - 20;
  Button1Click(self);
end;

procedure TForm1.Timer1Timer(Sender: TObject);
var
  c : char;
  i :integer;
  s : string;
  shexa : string;
begin
  timer1.enabled := false;
  if length(vgentrada) >= 52 then
  begin

    if (vgentrada[17] = $02) and
      (vgentrada[20] = $03) then
      begin
        sensorAD0.valor := vgentrada[16];

        imgBombaLigadaAD0.visible := vgentrada[18] = 1;
        EnchendoAd0 := vgentrada[18] = 1;

        imgBombaDesligadaAD0.visible := vgentrada[18] <> 1;
        ifvgentrada[18] = 1 then label17.caption := 'Sendo abastecido' else
        label17.caption := 'Distribuindo água';

        sensorAD1.valor := vgentrada[19];

```

```

imgBombaLigadaAD1.visible := vgentrada[21] = 1;
EnchendoAd1 := vgentrada[21] = 1;
imgBombaDesligadaAD1.visible := vgentrada[21] <> 1;
ifvgentrada[21] = 1 then label23.caption := 'Sendo abastecido' else
label23.caption := 'Distribuindo água';

tratavalores;

end;
shexa := '';
for i := 0 to 51 do
begin
shexa := shexa + inttohex(vgentrada[i],2)+' ';
end;

label9.caption := shexa;
setlength(vgentrada,0);
sleep(100);
podeenviar := true;
end;

end;

procedure TForm1.tmVerificaSeEstaLigadoTimer(Sender: TObject);

begin
tmVerificaSeEstaLigado.enabled := false;
if EncherAd0 <> EnchendoAd0 then
begin
ckRele.Checked := EncherAd0;
end;

if EncherAd1 <> EnchendoAd1 then
begin
ckLedVermelho.Checked := EncherAd1;
end;
try
if (EncherAd0 <> EnchendoAd0) or
(EncherAd1 <> EnchendoAd1) then btEnviaComandoClick(self);
except
end;

tmVerificaSeEstaLigado.enabled := true;
end;

procedure TForm1.tmPiscaTimer(Sender: TObject);
begin
ifultimopacote< now then
begin
Panel3.visible := not Panel3.visible;
Image2.visible := not Image2.visible;
end;
end;

procedure TForm1.SpeedButton1Click(Sender: TObject);
begin
application.createform(TfConfig, fconfig);
fconfig.showmodal;
iffconfig.modalresult = mrok then
begin
FormShow(self);

```

```
end;  
fconfig.free;  
end;
```

```
end.
```

## Anexo B (Código para RADIUINO Base)

```

#include <RADIUINO4APP.h>
#include <EEPROM.h>
#include <SPI.h>
#include "Headers.h"

byteint_rx = 0;
byteint_buff = 0;

void setup()
{
my_addr = 0;
power = 7;
channel = 21;
freq_offset = 0x33;
serial_baudrate = 9600;
Phy.initialize();
attachInterrupt(0, IntReceiveData,
RISING);
attachInterrupt(1, IntBufferOverflow,
RISING);
pinMode(GD00, INPUT);
Timer1.initialize(1000000);
Timer1.attachInterrupt(IntTimer1);
Serial.print("Radiuino! Base");
}
void loop()
{
if (Serial.available() > 0) {
Phy.receiveSerial();
}
}

voidIntReceiveData()
{
if (int_rx == 0)
{
int_rx = 1;
return;
}
if ( digitalRead(GD00) == HIGH ) {
if (Phy.receive_base(&g_pkt) == ERR)
return;
Phy.sendSerial(&g_pkt);
}
return;
}
voidIntBufferOverflow()
{
if (int_buff == 0) {
int_buff = 1;
return;
}
cc1101.Strobe(CC1101_SFRX);
cc1101.Strobe(CC1101_SRX);
return;
}

```

## Anexo C (Código para Radiuino Sensor)

```

#include <RADIUINO4APP.h>
#include <SoftwareSerial.h>
#include <EEPROM.h>
#include <SPI.h>
#include "Headers.h"
byteint_rx = 0;
byteint_buff = 0;
intleitorad=0;
byte valorad0=0;
byte valorad1=0;
byte lastad0=0;
byte lastad1=0;
byte rele1 = 0;
byte rele2 = 0;
byteenviar = 0;

SoftwareSerialRFID(7, 3);
void setup()
{
  my_addr = 10;           /* Endereço */
  power = 7;             /* Potência */
  channel = 21;          /* Canal */
  freq_offset = 0x30;    /* Offset de frequencia */
  serial_baudrate = 9600; /* Serial baudrate */
  Phy.initialize();
  Mac.initialize();
  Net.initialize();
  Transp.initialize();
  App.initialize();
  attachInterrupt(0, IntReceiveData, RISING);
  attachInterrupt(1, IntBufferOverflow, RISING);
  pinMode(GD00, INPUT);
  Timer1.initialize(1000);
  Timer1.attachInterrupt(IntTimer1);
  RFID.begin(9600);
  digitalWrite (IO0_PIN, LOW); // led vermelho
  digitalWrite (IO1_PIN, LOW); //
  digitalWrite (IO2_PIN, LOW); // LED VERDE
  digitalWrite (IO3_PIN, LOW); // RELE
  digitalWrite (IO4_PIN, LOW);
  digitalWrite (IO5_PIN, LOW); // LED AMARELO
}

void loop()
{
  char c;
  byteTagIndex = 0; //variavel para armazenar os caracteres lidos da serial
  verifySleepEntering();
  leitorad = analogRead(0);
  valorad0 = map(leitorad,0,1023,100,0);
  leitorad = analogRead(1);
  valorad1 = map(leitorad,0,1023,100,0);
  enviar = 0;
  if (lastad0 != valorad0){
    enviar = 1;
  }
  if (lastad1 != valorad1){

```

```

enviar = 1;
}
memset(&g_pkt, 0x00, sizeof(packet));
g_pkt.NetHdr[0] = 0x00;
g_pkt.NetHdr[1] = my_addr;
  g_pkt.AD0[0] = valorad0;
  g_pkt.AD0[1] = 0x02;
g_pkt.AD0[2] = rele1; // vou enviar o estado do rele1
g_pkt.AD1[0] = valorad1;
  g_pkt.AD1[1] = 0x03;
g_pkt.AD1[2] = rele2; // vou enviar o estado do rele1
lastad0 = valorad0;
  lastad1 = valorad1;
App.send(&g_pkt);
delay(1000);
}

```

```

voidIntReceiveData()
{
if (int_rx == 0)
  {
int_rx = 1;
return;
  }

if ( digitalRead(GD00) == HIGH )
  {
Phy.receive(&g_pkt);
if (g_pkt.AD2[1]==1)
  {
digitalWrite (IO3_PIN, HIGH); // RELE
  rele1=1;
  }
if (g_pkt.AD2[1]==0)
  {
digitalWrite (IO3_PIN, LOW); // RELE
  rele1=0;
  }
if (g_pkt.AD3[1]==1)
  {
digitalWrite (IO0_PIN, HIGH); // vermelho
  rele2=1;
  }
if (g_pkt.AD3[1]==0)
  {
digitalWrite (IO0_PIN, LOW); // vermelho
  rele2=0;
  }

if (g_pkt.AD4[1]==1)
  {
digitalWrite (IO5_PIN, HIGH); // amarelo
  }
if (g_pkt.AD4[1]==0)
  {
digitalWrite (IO5_PIN, LOW); // amarelo
  }
  }
return;
}

```

```

voidIntBufferOverflow()
{
if (int_buff == 0) {
int_buff = 1;
return;
}

cc1101.Strobe(CC1101_SFRX);
cc1101.Strobe(CC1101_SRX);
return;
}

voidverifySleepEntering ( void ) {
    ATOMIC_BLOCK(ATOMIC_FORCEON)
    {
if (Mac.time_to_sleep> 0) {
delayMicroseconds(1000);
while(Phy.txFifoFree() != CC1101_FIFO_SIZE);
Phy.lowPowerOn();
}
else if (Mac.time_to_sleep == 0) {
cc1101.Strobe(CC1101_SIDLE);
delay(1);
cc1101.Strobe(CC1101_SFTX);
cc1101.Strobe(CC1101_SFRX);
cc1101.Strobe(CC1101_SRX);
Mac.time_to_sleep = -1;
}
}
}

ISR(WDT_vect) {
    /* Verifica se precisamos decrementar o contador de Sleep */
if (Mac.time_to_sleep> 0) {
Mac.time_to_sleep--;
return;
}
}

```