

Sistema de monitoramento e controle do consumo de água residencial

Ricardo Barros Filtsoff, Daves Márcio Silva Martins

Curso de Bacharelado em Sistemas de Informação
Centro de Ensino Superior de Juiz de Fora (CES-JF)
Juiz de Fora – MG – Brasil

ricardofiltsoff@gmail.com, davesmartins@gmail.com

Abstract: *Recently the shortage of potable water in Brazil affected thousands of people and several municipalities adopted rationing measures. In the coming years water crises may repeat themselves. In this scenario, the conscious use of water becomes more and more necessary. The objective of this work is to present the development of a system connected to the Internet capable of monitoring and storing data of the residential water consumption independent of the utility meter. Through a mobile application the user can extract relevant information from historical data, receive alerts in case of abnormal consumption and remotely close the water record in the occurrence of claims.*

Resumo: *Recentemente a escassez de água potável no Brasil afetou milhares de pessoas e diversos municípios adotaram medidas de racionamento. Nos próximos anos as crises hídricas poderão se repetir. Diante deste cenário o uso consciente da água se torna cada vez mais necessário. O objetivo deste trabalho é apresentar o desenvolvimento de um sistema conectado à internet capaz de monitorar e armazenar dados do consumo de água residencial independente do medidor da concessionária. Através de um aplicativo de celular o usuário poderá extrair informações relevantes de dados históricos, receber alertas em caso de consumo anormal e fechar remotamente o registro de água na ocorrência de sinistros.*

1. Introdução

A água é um recurso natural essencial para a vida (FRANCISCO, 2018). Sua escassez traz consequências para toda a sociedade. O brasileiro consome em média 154 litros de água por dia conforme dados do Sistema Nacional de Informações sobre Saneamento (SNIS, 2016) o que representa em 44 litros a mais pelo recomendado pela Organização das Nações Unidas (ONU) (CNM, 2018).

Esse consumo excessivo piora a situação de escassez que o país enfrenta. De acordo com o relatório da Agência Nacional de Águas, as crises hídricas por secas ou estiagens afetaram 48 milhões de pessoas no Brasil entre 2013 e 2016 (ANA,2017).

Em uma residência, vazamentos e altos consumos contribuem para o desperdício de água. Na ocorrência de avarias em equipamentos hidráulicos ou até mesmo rompimento de tubulações, o tempo para descoberta dos defeitos é elevado favorecendo o desperdício. Diante destes fatos observa-se a necessidade de uma forma de monitoramento e controle automatizado do consumo de água.

Este trabalho está organizado na seguinte forma: iniciando-se pela introdução ao tema, relatando o problema e a justificativa. Em seguida descreve-se a solução proposta e os trabalhos relacionados. O funcionamento do sistema é mencionado seguido da arquitetura de hardware onde são informados os componentes utilizados. Na sequência, cita-se o protótipo, onde é demonstrado o esquema de ligação dos componentes e a programação desenvolvida. Em prosseguimento, o servidor web bem como o aplicativo e suas funcionalidades juntamente com o custo total dos componentes são mencionados. O artigo é finalizado com resultados obtidos, considerações finais e trabalhos futuros.

2. Problema

Nas instalações hidráulicas de uma residência diversos dispositivos podem apresentar vazamentos. Por exemplo, uma torneira que deixa escoar (por defeito) um filete de água de 1 milímetro de diâmetro, em apenas 24 horas, irá desperdiçar um volume de até 1.280 litros (CESAMA) e uma válvula de descarga com um pequeno filete representa o gasto de 144 litros por dia (SABESP).

Como o cliente calcula o consumo de água?

No atual modelo de medição de consumo de água residencial das concessionárias, o modo de acompanhamento do consumo é a medição manual, por se tratar de um equipamento puramente mecânico. Para realizar tal medição é necessário uma pessoa ir

ao hidrômetro, anotar valor dos números e realizar a subtração com a última medição registrada na fatura mensal atual (SABESP). Este método além de trabalhoso é sujeito a falhas humanas podendo haver avarias tanto ao anotar os números como no cálculo realizado, deixando a desejar a veracidade do resultado obtido.

3. Justificativa

Os hidrômetros mecânicos não dispõem meios de monitoramento automático e emissão de alertas, impossibilitando a antecipada tomada de decisão em casos de sinistros facilitando o desperdício de água, pois só será possível sua detecção, no caso de vazamentos visíveis, após o usuário encontrar o local do defeito.

Essas intercorrências só serão constatada uma possível suspeita após a chegada da fatura mensal e o usuário observar o aumento do consumo e nesse caso o desperdício não pode ser evitado.

Este cenário pode ser modificado com a utilização de um sistema automatizado de monitoramento do consumo de água residencial, permitindo ao usuário maior precisão e controle referentes ao uso da água. Esse sistema possibilita acompanhar consumo atual, realizar consultas referentes a consumos anteriores e gerar estimativas para a próxima fatura. O armazenamento dos dados permite realizar comparações com o histórico e gerar alertas para um aplicativo de celular na detecção de padrões anormais de consumo.

4. Solução proposta

Este trabalho propõe a busca por um meio de medição particular, de baixo custo, independente do hidrômetro da concessionária capaz de coletar e armazenar dados em nuvem, comparar informações com dados históricos, gerar estimativa de consumo atual de água e notificar usuário através de aplicativo de celular sobre consumo incomum de água ligado a possíveis vazamentos e sinistros ocorridos.

O objetivo geral do projeto é possibilitar o monitoramento do consumo de água residencial por parte do usuário final. Para alcançá-lo, alguns objetivos específicos deverão ser atendidos:

- Criar uma solução de hardware de baixo custo que realize a coleta do consumo de água;
- Armazenar dados obtidos em um servidor via internet;

- Realizar análises de consumo;
- Gerar notificações via aplicativo de celular para o usuário.
- Fechar remotamente o registro em caso de sinistros.

5. Trabalhos relacionados

Dentre os trabalhos relacionados com o objetivo de monitorar o consumo de água residencial, pode-se citar:

Pedro e Leandro (p2, 2016) cujo o trabalho apresenta o tema “Monitoramento do consumo de água utilizando ferramenta open source” onde utilizam uma aplicação web baseada em páginas *PHP*, *javascript* e *HTML* para monitorar o consumo através do sensor de fluxo de água e um *arduino*.

Já a solução de Renato (ALVES, 2015) do tema “Solução de monitoramento de consumo de água residencial” realiza o monitoramento individual dos dispositivos residenciais. O *arduino* trabalha como uma central que recebe os dados de consumo dos dispositivos via rádio frequência e envia dados a internet. Um aplicação web é utilizada para a interface de consulta dos dados.

Eduardo (ZANUTTO, 2017) apresentou uma proposta cujo o tema “Sistema de monitoramento do consumo de água controlado por um aplicativo Android” de controlar o consumo de água utilizando um *Raspberry*. Os dados coletados por um sensor de fluxo são enviados via *bluetooth* a um aplicativo *android* e armazenados em um banco de dados local.

6. Funcionamento

Um sensor de fluxo de água conectado à internet será instalado na tubulação de entrada de água da residência após hidrômetro da concessionária. Este sensor será responsável por coletar o fluxo de água e enviar dados a um servidor web. Através de um aplicativo de celular o usuário terá acesso aos dados coletados e será capaz de acompanhar o consumo, comparar dados históricos, receber notificações e fechar o registro de água do sistema remotamente. A figura 1 demonstra o funcionamento do sistema.

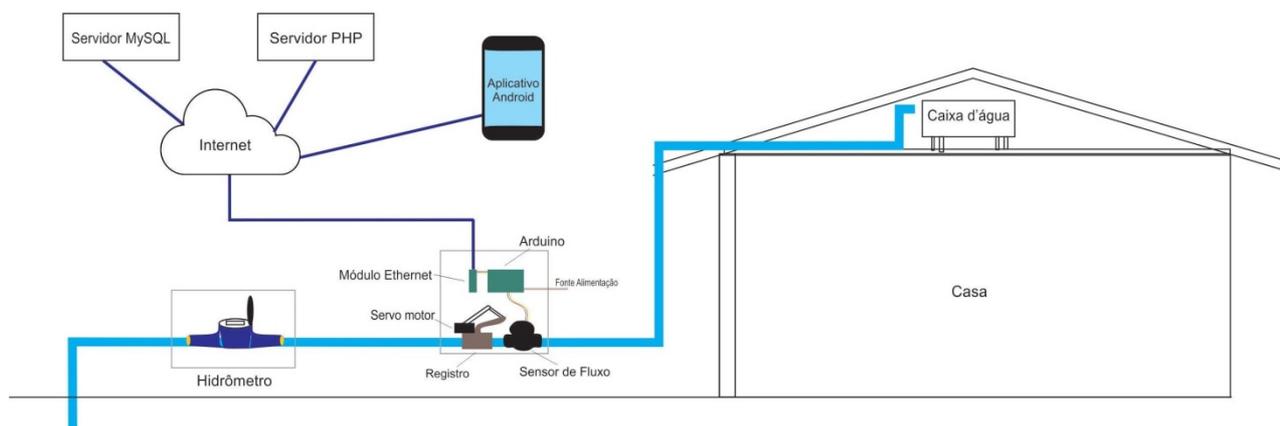


Figura 1 – Ilustração do funcionamento do sistema de monitoramento de água

Fonte: elaborado pelo autor 23/08/2018

7. Arquitetura de hardware

O sistema de monitoramento do consumo de água residencial será composto com os seguintes componentes:

7.1 Arduino

Arduino é uma plataforma eletrônica de código aberto baseada em hardware e software capaz de interpretar entradas e transformá-las em saídas, conforme conjunto de instruções contidas no microcontrolador. Possui IDE própria e linguagem de programação baseada em C/C++ responsáveis pelo desenvolvimento e envio das instruções ao hardware. Apresenta como características: baixo custo, multiplataforma e ambiente de programação simples (ARDUINO, 2017).

A placa possui pinos digitais que podem servir como entradas ou saídas de acordo com a definição no programa. Possui também entradas analógicas que permitem sinais variáveis. Os pinos digitais permitem apenas as opções: “ligado” ou “desligado”

Para este trabalho será usado o modelo *Arduino* UNO. Possui 14 pinos digitais, 6 entradas analógicas, um cristal de quartzo de 16 MHz, uma conexão USB, um conector de energia, um conector ICSP e um botão de reset (ARDUINO,2018).

O *Arduino* pode ser conectado a diversos módulos e sensores que emitem dados ou possam ser controlados (MCROBERTS 2011, p.23). Pode ser energizado via porta USB ou fonte externa (6v a 20V). A placa dispõe dos pinos GND, 5V, 3.3V para energizar módulos e componentes. A tensão de trabalho nos pinos digitais é 5V. Na figura 2 observa-se a placa *Arduino UNO* bem como seus pinos de conexão.



Figura 2 – Arduino UNO

Fonte: <https://store.arduino.cc/usa/arduino-uno-rev3>

Acesso em 04/06/2017

A placa *Arduino* é o componente responsável por receber os dados do sensor de fluxo de água e, acoplado a um módulo *ethernet*, enviar dados ao servidor web.

7.2 Sensor de fluxo de água

O sensor de fluxo de água (figura 3) é um dispositivo capaz de medir a vazão de líquidos. Em seu interior existe um rotor composto por paletas, semelhante às de um hidrômetro tradicional, acoplado a um ímã que trabalha em conjunto com um sensor responsável por enviar pulsos de acordo com a velocidade de rotação das paletas. Através desses pulsos é possível mensurar a vazão de água de acordo com a frequência de rotação (SEED, 2017).

Conectado às instalações hidráulicas da residência, o sensor de fluxo enviará ao *Arduino* a quantidade de pulsos coletados conforme vazão da água na tubulação. A figura 3 demonstra a visão interna de sensor de fluxo e suas paletas.



Figura 3 – Sensor de Fluxo de água DN32, visão interna

Fonte: <http://loja.multcomercial.com.br/sensor-de-fluxo-de-agua-dn32.html>

Acesso em 04/06/2017

7.3 Módulo Ethernet

O módulo *ethernet* (figura 4) estabelece comunicação com o *Arduino*, permitindo conexões em redes através da porta *ethernet* RJ45 contida neste dispositivo, o qual suporta vários protocolos internet. Este chip contém um controlador *ethernet* para conexão de rede do padrão 10BASE-T e uma interface de comunicação com o *Arduino* (ARDUINOBR,2018).



Figura 4 – Módulo ethernet ENC28J50

Fonte:<http://www.baudaeletronica.com.br/modulo-ethernet-enc28j60.html>

Acesso em 04/06/2017.

7.4 Servo motor

O servo motor (figura 5) é um motor na qual podemos controlar sua posição angular entre 0 e 180° sendo um atuador eletromecânico utilizado para posicionar e manter um objeto em uma determinada posição (VIDADESILICIO,2018).



Figura 5 – Servo motor

Fonte:<https://www.diyelectronics.co.za/store/servos/1668-towerpro-mg996r-servo-motor-360-degreeotation.html>

Acesso em 24/08/2018

Acoplado a um registro, o servo motor ao movimentar irá abrir ou fechar a passagem da água na tubulação.

8. Protótipo

8.1 Ligação dos componentes

Os módulos e componentes utilizados no projeto são energizados pela própria placa *Arduino*. O sensor de fluxo e o servo motor são ligados no pino 5V e GND para alimentação elétrica e pinos 2 e 6 para envio dos dados respectivamente. O módulo *ethernet* utiliza os pinos 13,12,11 e 10 para dados e é alimentado com 5V e GND. A ligação dos componentes se dá conforme a figura 6.

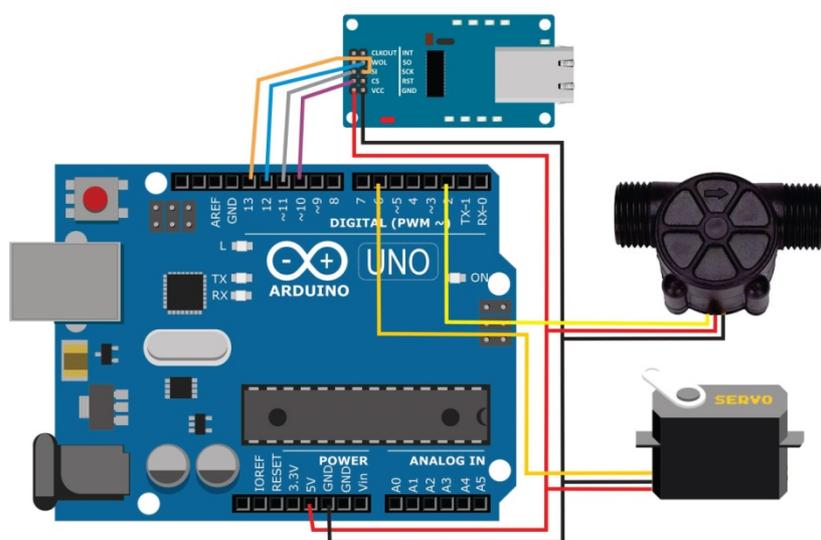


Figura 6 – Esquema de ligação elétrica dos componentes no arduino

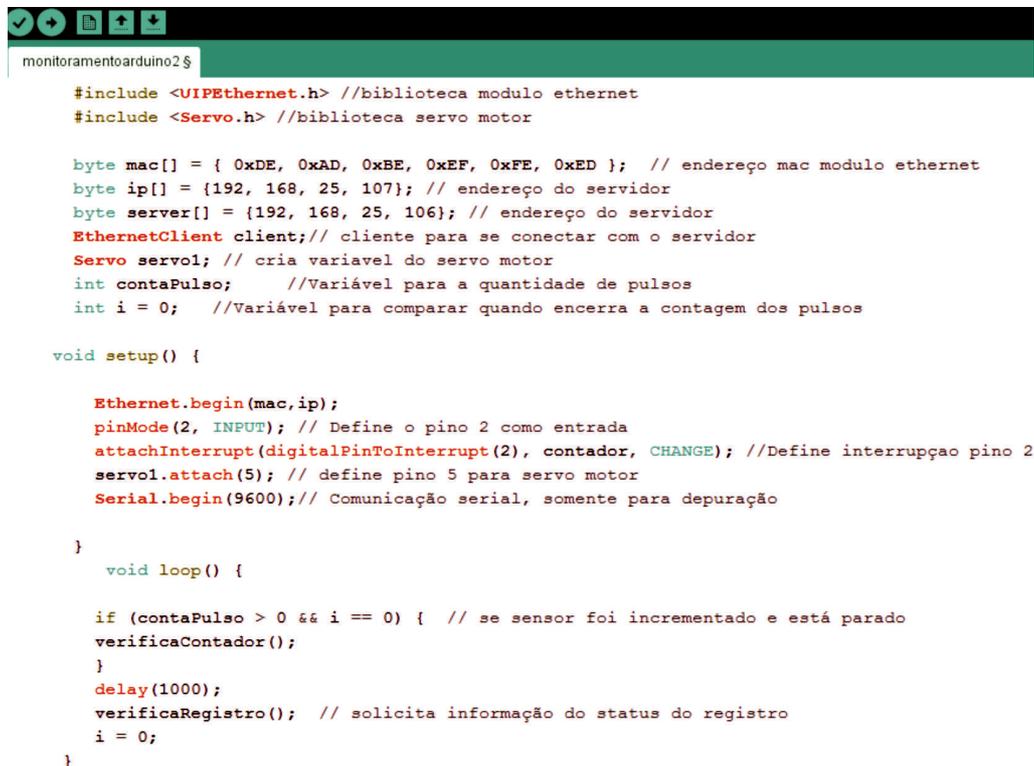
Fonte: elaborado pelo autor 23/08/2018

8.2 Programação Arduino

O software *Arduino IDE* (figura 7) permite a programação e o envio do código à placa *Arduino*. Em sua estrutura duas funções são essenciais: *setup()* - inicializa as variáveis e define o modo dos pinos e função *loop()* onde o algoritmo é escrito. A função *setup()* é executada apenas uma vez, enquanto a função *loop()* repete continuamente enquanto o *Arduino* estiver energizado. As variáveis e bibliotecas são declaradas antes do *setup()* (ARDUINO,2018).

No início do código as bibliotecas *UIPEthernet.h* (módulo ethernet) e *Servo.h* (servo motor) são incluídas. Variáveis inicializadas: do tipo *byte* para endereço de ip e *mac* do *Arduino*, ip servidor e do tipo *int* para comparação. Variáveis declaradas: tipo *int* para contar os pulsos, tipo *servo* para o servo motor e tipo *EthernetClient* para módulo *ethernet*. Na função *setup()* são atribuídos *ip* e *mac*, definido pino 2 como entrada e uma

função de interrupção onde a variável conta pulso é incrementada ao movimentar as paletas do sensor de fluxo. O pino 5 é definido para transmissão de dados do servo motor.



```
monitoramentoarduino2$
#include <UIPEthernet.h> //biblioteca modulo ethernet
#include <Servo.h> //biblioteca servo motor

byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED }; // endereço mac modulo ethernet
byte ip[] = {192, 168, 25, 107}; // endereço do servidor
byte server[] = {192, 168, 25, 106}; // endereço do servidor
EthernetClient client; // cliente para se conectar com o servidor
Servo servol; // cria variavel do servo motor
int contaPulso; //Variável para a quantidade de pulsos
int i = 0; //Variável para comparar quando encerra a contagem dos pulsos

void setup() {

  Ethernet.begin(mac,ip);
  pinMode(2, INPUT); // Define o pino 2 como entrada
  attachInterrupt(digitalPinToInterrupt(2), contador, CHANGE); //Define interrupção pino 2
  servol.attach(5); // define pino 5 para servo motor
  Serial.begin(9600); // Comunicação serial, somente para depuração
}

void loop() {

  if (contaPulso > 0 && i == 0) { // se sensor foi incrementado e está parado
    verificaContador();
  }
  delay(1000);
  verificaRegistro(); // solicita informação do status do registro
  i = 0;
}
```

Figura 7 – IDE arduino e código de programação

Fonte: elaborado pelo autor 27/08/2018

Na função *loop()*, o algoritmo faz a verificação da contagem dos pulsos. Sendo maior do que zero e ao cessar a contagem é feita a conversão em litros. A constante de conversão pulsos em litros foi obtida ao calibrar o sensor utilizando recipiente medidor acompanhando o número de pulsos pelo fluxo de um litro de água. O valor obtido foi 440 pulsos a cada litro de água. Após a conversão é invocada a função para enviar dados a uma página *PHP* passando como parâmetro o resultado em litros. O servidor processa os dados e o *Arduino* faz a leitura do retorno. Obtendo sucesso na persistência dos dados, a variável que armazena os pulsos é zerada. Caso haja falha, será feita uma nova tentativa. Ainda no loop, é solicitado ao servidor informação sobre status do registro. Se no retorno houver solicitação de abertura ou fechamento, a ação é executada e em seguida enviada requisição ao servidor informando que o status foi alterado. O diagrama (figura 8) representa a lógica contida na programação definida no *Arduino*.

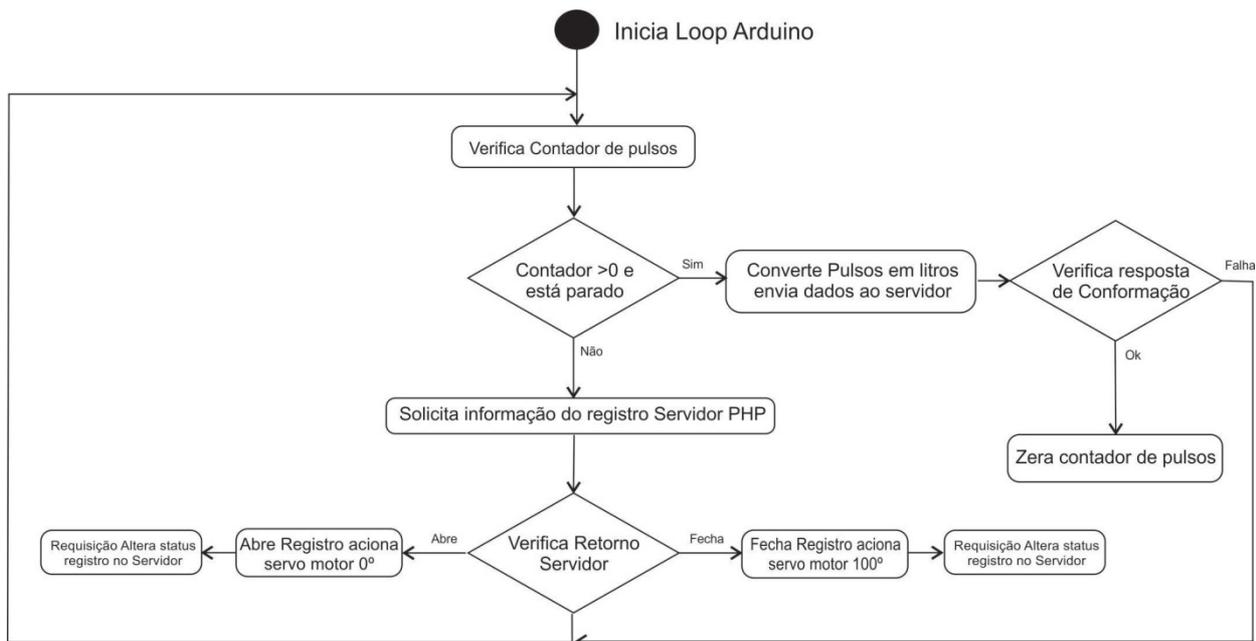


Figura 8 – Diagrama de funcionalidade do algoritmo do arduino.

Fonte: elaborado pelo autor 23/08/2018

9. Servidor Web

9.1 Páginas PHP

A comunicação entre o *Arduino*, o aplicativo e o servidor web se dá via arquitetura cliente/servidor. O *Arduino* e o aplicativo são os clientes que enviam requisições ao servidor. Os pedidos são processados gerando uma mensagem de resposta para o cliente. O servidor web desse projeto é composto de servidor apache e interpretadores para processamento de páginas *php* e servidor *MYSQL* para o banco de dados. As requisições são feitas via *GET* (*Arduino*) e *POST*(aplicativo) às páginas *php* informando a opção via parâmetro e valores quando necessário. Para o funcionamento do sistema foram desenvolvidas as seguintes páginas *php*:

- *conexao.php* - realiza conexão com o banco de dados *MYSQL*;
- *classesSM.php* - contém as classes, atributos e funções;
- *requisicaoArduino.php* - recebe requisições e parâmetros do *Arduino*;
- *requisicaoAPP.php* - recebe requisições e parâmetros do aplicativo.

As duas últimas classes citadas realiza a conexão com o banco de dados, processa os dados recebidos e realiza consultas ou inserções no banco de dados de acordo com a lógica do programa.

O retorno das páginas *php* para o *Arduino* são mensagens através de caracteres enquanto para o aplicativo são objetos *JSON*, conjuntos de pares e valores em uma formatação para troca de dados independente de linguagem de programação (*JSON*, 2018).

9.2 Banco de dados

O banco de dados “*sistemamonitoramento*” (figura 9) é composto por três tabelas para armazenamento dos dados. A tabela “*consumoagua*” recebe os dados coletados pelo sensor de fluxo já convertido em litros e a data é instanciada pelo banco de dados no momento da inserção. A data é utilizada para consultar dados históricos, gerar estimativas de consumo e comparações. A tabela registro armazena o estado atual do registro (aberto ou fechado) e as solicitações enviadas pelo aplicativo (pedido de alteração de status). Ao alterar o estado do registro através do aplicativo, o campo “*alterastatus*” do último registro é modificado. O *Arduino* requisita o status do último registro. Neste momento a página *requisicaoArduino.php* realiza consulta na tabela registro e faz comparação entre o estado atual e o pedido de alteração. Sendo os valores diferentes, o *Arduino* que recebe a resposta, executa a ação de acordo com o retorno e faz requisição a página *php* responsável pela inserção de um novo registro já passando como parâmetro o estado do registro alterado.

Tabelas banco de dados ‘sistemamonitoramento’

consumoagua		registro		notificacao	
# id	int(11)	# id	int(11)	# id	int(11)
datahora	timestamp	dataModificacao	timestamp	datahora	timestamp
litros	float	status	varchar(20)	mensagem	varchar(200)
		alterastatus	varchar(20)	visualizada	varchar(10)

Figura 9 – Tabelas do banco de dados

Fonte: elaborado pelo autor 27/08/2018

Para simular um servidor *apache*, *MYSQL* e interpretador *PHP* utilizou-se instalado em um computador o pacote *Xampp*, servidor independente de plataforma, de código aberto contendo *MYSQL*, *PHP* e *Perl* (*XAMPP*, 2018).

10. Aplicativo

O aplicativo desenvolvido na *IDE Android Studio* possibilita o monitoramento do consumo de água pelo usuário através de uma interface amigável e de fácil usabilidade. Na tela principal (figura 10) há seis funcionalidades acionadas por botões. Cada funcionalidade possui uma classe e uma respectiva tela (activity) para interação com o usuário.

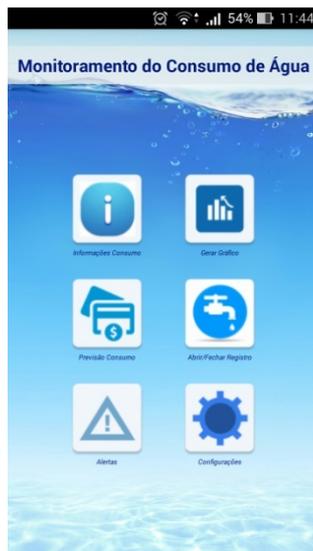


Figura 10 – Tela principal do aplicativo

Fonte: elaborado pelo autor 27/08/2018

As requisições do aplicativo ao servidor são feitas para a página *requisicaoAPP.php* passando os parâmetros via método *POST*. Os dados são transitados através de objetos *JSON* e o retorno das requisições é exibido na tela para o usuário. Registros não encontrados no banco de dados são atribuídos com o valor "0".

10.1 Funcionalidades

10.1.1 Informações de Consumo

A tela informações de consumo exibe dados sobre consumo do dia atual, anterior e as médias globais de consumo diário e mês atual. Ao clicar no botão "Informações de Consumo", uma função é executada e requisita à página *php* as informações. A página processa o pedido e retorna um objeto com os valores requisitados que é exibido na tela para o usuário.

10.1.2 Gráficos

Gráficos são essenciais para acompanhamento e comparação do consumo pelo usuário através dos dados históricos. A biblioteca “MPAndroidChart” adicionada ao projeto é responsável por gerar os gráficos do modelo barra.

Na tela dos gráficos, o usuário seleciona uma das opções:

- *Consumo diário por mês*: o mês e o ano são definidos para a consulta resultando o consumo total de cada dia do período;
- *Min / Med / Max Diário*: retorna os consumos máximo, mínimo e a média de consumo por dia com os dados registrados até o momento da consulta;
- *Consumo mensal por ano*: o ano é selecionado e retorna o consumo dos total dos meses daquele ano;
- *Min / Med / Max Mensal*: retorna os consumos máximo, mínimo e a média de consumo por mês com os dados históricos;

10.1.3 Previsão do Consumo

Para a estimativa de consumo, o usuário deve informar na própria tela, o dia do mês padrão que a concessionária realiza a medição do hidrômetro, que se encontra na fatura mensal. Esta data é inserida na primeira utilização e fica armazenada no aplicativo. Uma requisição é enviada ao servidor passando como parâmetro o dia da medição que processa os dados e retorna com o consumo do mês atual e a previsão em metros cúbicos de água. A média do consumo diário de água do mês atual é utilizada como base para estimar a previsão para os dias faltantes até a próxima medição. É exibido ao usuário o consumo do mês atual e a previsão até o dia da próxima medição.

10.1.4 Abrir / fechar registro

Ao acessar a tela para movimentar o registro, uma requisição é enviada ao servidor para consulta da situação atual do registro (aberto ou fechado). O retorno é exibido na tela do aplicativo. Através de um botão o usuário solicita a movimentação do registro à página *php* que insere no banco de dados o valor passado via parâmetro. A movimentação é efetuada no momento em que o *Arduino* requisita informações do registro e de acordo com o pedido executa a tarefa. Após a movimentação do registro, o usuário recebe uma notificação informando o novo estado do registro.

10.1.5 Alertas

As notificações ou alertas são essenciais para informar ao usuário sobre situações anormais e consumos excessivos sendo geradas nos seguintes casos:

- Abrir ou fechar o registro;
- Maior consumo diário do mês;
- Consumo diário acima da média;
- Maior consumo diário global;
- Consumo diário maior que o dobro da média geral.

As consultas e comparações com dados históricos no banco de dados são realizadas no momento em que o *Arduino* requisita a persistência de novos dados coletados pelo sensor de fluxo ao servidor *PHP*. Ao atingir alguma das situações de alerta, uma notificação é enviada ao aplicativo do usuário. Confirmando a visualização do alerta, o campo “visualizado” da tabela notificação é alterado para “sim”.

10.1.6 Configuração

Na tela de configuração o usuário informa a *URL* do servidor web responsável pelas requisições. Essa opção torna o aplicativo flexível às mudanças de hospedagem dos servidores. No campo id do sensor é incluído o código de referência definido pelo *Arduino* garantindo a comunicação com o sensor correto. O número do medidor a ser informado na tela de configuração pelo usuário é apenas de caráter informativo.

11. Custo

A tabela da figura 11 destaca o valor dos componentes utilizados nesse projeto.

Quantidade	Componente	Valor R\$
1	Arduino UNO	27,49
1	Sensor de fluxo água 1/2"	21,49
1	Módulo Ethernet	15,99
1	Servo motor	30,00
1	Registro esfera 1/2"	14,00
1	Niple de 1/2"	0,80
1	Luva rosqueada 1/2"	0,50
1	Joelho 1/2"	0,50
	TOTAL	R\$ 110,77

Figura 11 – Tabela de custo Fonte: elaborado pelo autor 13/09/2018

Os valores são referentes a data de setembro de 2018. Para hospedagem do servidor web pode ser utilizado serviços gratuitos. Uma solução testada foi o “000Webhost”, que oferece hospedagem grátis para páginas *PHP* e banco de dados *MYSQL*. A empresa oferece no plano gratuito 1GB de Espaço em Disco, 2 Bancos de dados *MySQL* e hospedagem de domínio gratuito. (WEBHOST, 2018).

12. Resultados obtidos

Para a análise do protótipo e coleta de dados, o sistema de monitoramento do consumo de água foi instalado nas instalações hidráulicas da pia da cozinha de uma residência (figura 12). A instalação do sistema não apresentou dificuldades. O sensor acoplado ao registro foi ligado em série com a tubulação de água. O *Arduino* foi energizado via fonte de alimentação externa e o módulo ethernet conectado a um cabo de rede. O servo motor foi fixado próximo ao registro. Os componentes foram acomodados e protegidos em um recipiente de plástico.



Figura 12 – Protótipo instalado
Fonte: elaborado pelo autor 27/08/2018

O sistema esteve em pleno funcionamento durante 12 dias obtendo 955 registros de consumo de água. Neste período, alertas foram enviados informando consumo anormal e acima da média. Após a instalação observou-se mudanças nos hábitos dos

moradores da residência. Ao estarem cientes que consumos altos alertas seriam enviados o desperdício foi evitado. De acordo com o gráfico gerado no aplicativo (figura 13), note-se o maior consumo no sábado (25/08), dia no qual houve maior concentração de pessoas na residência aumentando consideravelmente o consumo de água. No dia 27/08 segunda-feira foi registrado o menor consumo do período, onde neste dia os moradores estavam ausentes da residência durante a maior parte do dia.

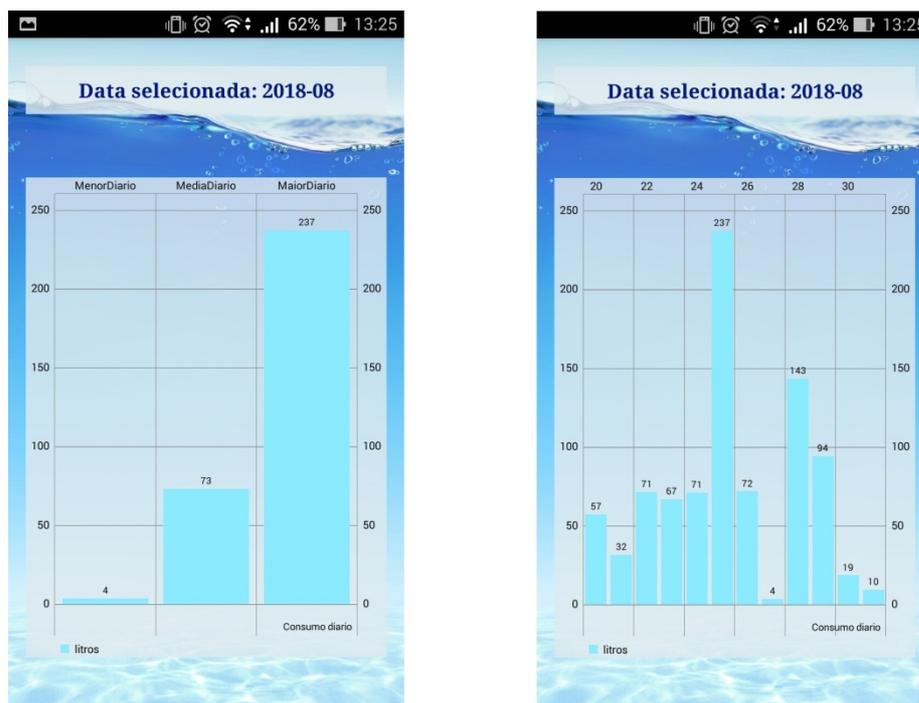


Figura 13 – Gráficos gerado pelo aplicativo android

Fonte: elaborado pelo autor 27/08/2018

Mesmo o sistema estando em funcionamento por poucos dias, os dados coletados foram compatíveis com o comportamento de consumo dos moradores durante período de funcionamento do sistema.

13. Considerações finais

Diante da perspectiva do uso consciente da água em ambiente doméstico, o sistema desenvolvido para o monitoramento do consumo de água obteve eficácia ao evitar o desperdício e atingiu os objetivos propostos. A autonomia do usuário monitorar o consumo, gerar estimativas e realizar comparações proporcionou maior precisão da utilização da água em ambiente residencial. O envio de alerta pelo sistema permite a tomada de decisão antecipada em situações anormais onde o usuário, mesmo distante de

sua residência pode fechar o registro, sendo útil em casos de viagens ou ocorrências de pane quando ausente da sua residência. A união de tecnologia, consciência ambiental e economia financeira favoreceram o uso consciente da água. O baixo custo, fácil instalação e usabilidade torna esse projeto acessível aos clientes que desejam monitorar o consumo de água em uma residência.

Este trabalho pode ser aprimorado em alguns requisitos. Em relação a segurança, pelo fato de estar conectado à internet, o sistema pode ser alvo de ataques e invasões. A implementação de algoritmos de criptografia e *hash* para acesso ao servidor web tornará o sistema mais seguro. A integração com a Internet das coisas (IoT) juntamente com a substituição do protocolo HTTP pelo protocolo MQTT tende melhorar alguns aspectos tais como: disponibilidade, desempenho, escalabilidade e integridade deste projeto.

Outro quesito é a autonomia de energia elétrica do *Arduino* e sensores. A utilização de uma bateria acoplada ao *Arduino* evitará perdas dos dados coletados em casos de queda de energia.

E, por fim, para atender um maior número de usuários é interessante o desenvolvimento de um aplicativo de celular híbrido, permitindo o funcionamento do sistema de monitoramento de consumo de água independente do sistema operacional.

Referências

FRANCISCO, Wagner de Cerqueira. "Água 2"; Brasil Escola. Disponível em <<https://brasilecola.uol.com.br/geografia/agua2.htm>>. Acesso em 03 de set.2018.

SNIS. Sistema Nacional de Informações sobre Saneamento: Diagnóstico dos Serviços de Água e Esgotos – 2016. Brasília: SNSA/MCIDADES, 2018. 218 p. : il.

CNM - Confederação Nacional de Municípios Brasileiro consome, em média, 154 litros de água por dia, aponta ONU <<https://www.cnm.org.br/comunicacao/noticias/brasileiro-consome-em-media-154-litros-de-agua-por-dia-aponta-onu>> Acesso em: 12 de set.2018.

ANA - Agência Nacional de Águas (Brasil). Conjuntura dos recursos hídricos no Brasil 2017 : relatório pleno / Agência Nacional de Águas. Brasília: ANA, 2017

CESAMA - Companhia de Saneamento Municipal – O consumo de água aumentou. Disponível em <<http://www.cesama.com.br/pdf/O-consumo-de-agua-aumentou.pdf>> acesso em 19 de ago.2018.

SABESP - Companhia de Saneamento Básico do Estado de São Paulo – Tabela de vazamento. Disponível em <http://site.sabesp.com.br/uploads/file/clientes_servicos/tabela_vazamento.pdf> acesso em 19 de ago.2018.

SABESP - Companhia de Saneamento Básico do Estado de São Paulo – Hidrômetros. Disponível em <<http://site.sabesp.com.br/site/interna/Default.aspx?secaold=324>> acesso em 19 de ago.2018.

GROSSKOPF, Pedro; CORREA PYKOSZ, Leandro. Monitoramento do consumo de água utilizando ferramenta open source. Trabalho apresentado no Congresso Nacional de Inovação e Tecnologia. Universidade do Estado de Santa Catarina (UDESC), São Bento do Sul, 2016.

TAVARES ZANUZZO, Lucas. Sistema de monitoramento do consumo de água controlado por um aplicativo Android. Trabalho de Conclusão de Curso (Graduação em Engenharia de Computação) Universidade de São Paulo - Escola de Engenharia de São Carlos, São Carlos, 2017.

ALVES, Renato Augusto de Paula. Solução de monitoramento de consumo de água residencial. Trabalho de Conclusão de Curso (Graduação em Sistemas de Informação) Centro Universitário Eurípides de Marília – UNIVEM Marília, 2015.

ARDUINO. Fundação - Introdução ao Arduino. Disponível em <<https://www.arduino.cc/en/Guide/Introduction>> Acesso em: 04 de jun.2017.

ARDUINO. Referência da Linguagem. Disponível em <<https://playground.arduino.cc/Portugues/Referencia>> Acesso em: 09 de jun.2017.

SEED. Sensor de fluxo de água - Introdução. Disponível em <http://wiki.seeed.cc/G1_and_2_inch_Water_Flow_Sensor> Acesso em: 09 de jun.2017.

VIDADESILICIO. Servo motor- Introdução. Disponível em <<https://portal.vidadesilicio.com.br/o-que-e-servomotor> > Acesso em: 22 de ago.2018.

ARDUINO. Referência da Linguagem. Disponível em <<https://playground.arduino.cc/Referencia/Loop>> Acesso em: 24 de ago.2018.

MCROBERTS, Michael. Arduino básico; [tradução Rafael Zanolli]. São Paulo :Novatec Editora, 2011.

ARDUINO. Visão geral arduino UNO. Disponível em <<https://store.arduino.cc/usa/arduino-uno-rev3>> Acesso em: 26 de ago.2018.

ARDUINOBR. Colocando o Arduino na rede com o ENC28J60. Disponível em <http://www.arduino.br/arduino/arduino_shield/colocando-o-arduino-na-rede-com-o-enc28j60-ethernet-shield/> Acesso em: 26 de ago.2018

XAMPP. O que é XAMMP? Disponível em <https://www.apachefriends.org/pt_br/index.html> Acesso em: 27 de ago.2018.

JSON. Introdução ao JSON. Disponível em <<https://www.json.org/json-pt.html>> Acesso em: 28 de ago.2018.

WEBHOST. Hospedagem de sites. Disponível em <<https://br.000webhost.com>> Acesso em: 10 de set.2018.