

## Proposta de Metodologia para Monitoramento de Grandezas Físicas Utilizando Ferramentas de Código Aberto

Gustavo F. de Lima<sup>1</sup>, Emerson Matheus M. do Nascimento<sup>2</sup>,  
Marcelo L. de Azevêdo Filho<sup>2</sup>, Jurandy M. Soares Junior<sup>3</sup>

<sup>1</sup>Professor do Curso Técnico em Mecatrônica

<sup>2</sup>Aluno do Curso Técnico em Informática

<sup>3</sup>Professor do Curso Técnico em Informática

Instituto Federal do Rio Grande do Norte (IFRN) – Campus Parnamirim  
Rua Antônia de Lima Paiva, 155, Nova Esperança, CEP: 59143-455, Parnamirim, RN

{gustavo.lima, jurandy.soares}@ifrn.edu.br

**Abstract.** *The graphic monitoring usually performed with spreadsheet and proprietary softwares for its elaboration. The objective of this work is presented an alternative methodology by using open source softwares to temperature monitoring of a didactic thermal system. A low cost didactic thermal system was created and one laptop was prepared with required programs. One experimental test was performed and temperature graph varied in range 38 to 42 °C, it displayed on browser. The result shows that is possible to temperature graphing monitoring of a thermal system by using open sources tools with success.*

**Resumo.** *O monitoramento gráfico costuma utilizar planilhas eletrônicas e/ou programas proprietários para sua elaboração. O objetivo deste trabalho é apresentar uma metodologia alternativa utilizando ferramentas de código aberto para o monitoramento de temperatura de um sistema térmico didático. Um sistema térmico de baixo custo foi montado e um computador preparado com os programas necessários. Um ensaio experimental foi realizado e o gráfico de temperatura variou de 38 a 42 °C, exibido na tela de um navegador de Internet. O resultado mostra que é possível monitorar graficamente a temperatura, com sucesso, de um sistema térmico utilizando ferramentas de código aberto.*

### Introdução

Monitorar é acompanhamento, avaliação e controle das condições ou de fenômenos, naturais ou artificiais, com o objetivo de obter dados quantitativos e qualitativos que possibilitem maior conhecimento sobre eles [Ormond 2006]. Algumas formas de monitoramento são os sinaleiros, a régua de LEDs, os visores alfanuméricos, os gráficos etc.

Uma revisão bibliográfica mostra alguns trabalhos sobre as formas de monitoramento e diferentes metodologias para apresentação dos dados. [de Lima 2013] apresentou um controle de temperatura utilizando o Arduino. Este salvou os valores de interesse em um cartão memória. Depois, com o auxílio de uma planilha eletrônica os gráficos foram desenhados. [da Silva et al. 2014] abordaram duas formas de monitoramento de nível de líquido: uma régua de LEDs e um gráfico de nível *versus* tempo. [Ribeiro et al. 2014] desenvolveram um sistema de monitoramento de motor/bombas de um sistema de

distribuição de água utilizando Arduino e *Shield* SIM900. O sistema mediu os valores de temperatura e corrente dos equipamentos e os enviou automaticamente por mensagem de texto para um celular cadastrado. [Ramos and Andrade 2015] abordaram o monitoramento de potência e vazão de água de uma instalação de pequeno porte. Os gráficos foram produzidos com ajuda de uma planilha eletrônica. [Dilly and Mendes 2015] apresentaram o monitoramento de umidade e temperatura utilizando Arduino e *Ethernet Shield*. As leituras foram enviadas para um servidor, que utilizando *JavaScript* apresentou os valores na tela do navegador de Internet. [Soares et al. 2017] trabalharam no monitoramento de variáveis elétricas e apresentaram os dados na tela de um supervisor.

Com base na revisão bibliográfica, existem diferentes formas para o monitoramento de grandezas físicas e diferentes formatos para apresentação dos dados.

O objetivo deste artigo é apresentar uma metodologia alternativa para apresentação de gráficos de monitoramento. Como exemplo é apresentado o gráfico de temperatura de um sistema térmico em tempo real na tela de um navegador de Internet (*browser*). O passo-a-passo desenvolvido para instalar e configurar as ferramentas de código aberto também é apresentado. Isso permite que outros trabalhos desenvolvam mais facilmente um ambiente de visualização interessante e de fácil implementação.

Para a continuação deste trabalho pretende-se, futuramente, apresentar o erro de medição, índice de ruído, comparar com outras ferramentas e melhorar a interface com o usuário para uma visualização dinâmica dos dados.

## **Prática Profissional**

Em nossa instituição de ensino, a prática profissional é compreendida como um componente curricular e se constitui em uma atividade articuladora entre o ensino, a pesquisa e a extensão, balizadora de uma formação integral de sujeitos para atuar no mundo em constantes mudanças e desafios [IFRN 2011].

A obtenção do Diploma de técnico de nível médio só é possível depois da conclusão, de carga horária mínima de 400 horas, realizada por meio de Estágio Curricular e desenvolvimento de projetos de pesquisa e/ou projetos de extensão.

Os projetos deverão contemplar o princípio da unidade entre teoria e prática, a aplicação dos conhecimentos adquiridos durante o curso, (...), do desenvolvimento de tecnologias e da construção de soluções para problemas [IFRN 2011].

Neste trabalho, os alunos desenvolveram a prática profissional no ano de 2016 e foram divididos em dois grupos, para um melhor desempenho, conforme as habilidades desenvolvidas em cada curso. O primeiro grupo foi responsável por construir o sistema térmico didático, montar o relé de estado sólido e programar a placa Arduino para ler a temperatura e controlar o aquecimento do sistema. E o segundo grupo foi responsável por instalar e configurar um servidor *web*, programar em *Python* e salvar em arquivo TXT as leituras de temperatura, instalar e configurar a biblioteca *Dygraphs*, programar em HTML e obter os gráficos de temperatura *versus* tempo.

## **Referencial Teórico**

Nesta seção são apresentados os principais componentes para construção de um sistema térmico didático baixo custo para a realização do monitoramento proposto.

## Sistema Térmico Didático

Para montagem do sistema térmico foram adquiridos os seguintes materiais: caixa de plástico (34 x 24 x 11 cm), base E27, lâmpada incandescente de 40 W/220 V e a base do sensor LM35, montada previamente. De posse dos materiais, a base E27 foi fixada na parte de baixo da caixa com o uso de fitas abraçadeiras e depois a base do LM35 foi fixada com cola quente na parte de cima da caixa, conforme a Fig. 1a.

Para testar o funcionamento dos componentes do sistema, a tampa da caixa foi colocada em seu lugar, a lâmpada ligada e um multímetro conectado ao sensor LM35 para medição da máxima tensão de saída. A máxima tensão de saída medida foi 0,45 V, o que corresponde a uma temperatura de 45 °C, conforme a Fig. 1b.



Figura 1. Sistema Térmico Didático.

Neste projeto, o sistema térmico didático forneceu dados de temperatura para a realização de um monitoramento gráfico utilizando ferramentas *open source*.

## Servidor Apache™

O Apache™ é o programa mais utilizados no desenvolvimento de servidores *web* e segundo a [Netcraft 2018] o servidor Apache™ está com 43,03% de fatia de mercado dos *sites* ativos, na frente da *nginx*™ (21,02%), Google™ (7,71%) e Microsoft™ (6,75%).

A instalação do Apache™ é realizada com o comando do Código 1.

```
1 $ sudo apt-get install apache2
```

### Código 1. Instalação do Apache

Para salvar arquivos no diretório (ou pasta) do Apache™ é necessário conceder permissão de escrita ao usuário normal, digitando o comando do Código 2.

```
1 $ sudo chmod 777 /var/www/html
```

### Código 2. Conceder Permissão

Neste projeto, o Apache™ hospedou a página HTML responsável por exibir o gráfico de temperatura *versus* tempo e o arquivo com os valores de tempo e temperatura.

## Linguagem Python

Em 1991 o holandês Guido von Rossum lançou a linguagem de programação *Python*. Ela é *open source* podendo ser usada gratuitamente e em praticamente qualquer sistema operacional. O *Python* é uma linguagem de programação poderosa, fácil de aprender, possui estrutura de dados de alto nível e permite ao programador concentrar-se na solução do problema, em vez de concentrar-se na linguagem propriamente dita [Alves 2013].

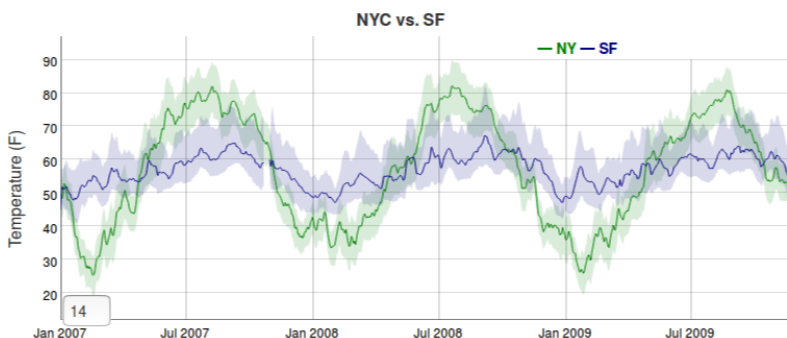
Neste trabalho, o *Python* foi utilizado para ler os valores de tempo e temperatura impressos na porta serial e salvá-los em um arquivo TXT dentro do diretório do Apache™, ou seja, /var/www/html/.

## Biblioteca Dygraphs

O *Dygraphs* é uma biblioteca desenvolvida utilizando a linguagem JavaScript™ e renderiza seus gráficos utilizando o canvas do HTML5. Ele foi desenvolvido para suportar uma quantidade grande de pontos e opera recebendo um arquivo de texto [Neto 2013].

O gráfico produzido é interativo: você pode passar o *mouse* para destacar os valores individuais. Você pode clicar e arrastar para ampliar. Clicar duas vezes para diminuir o *zoom* etc. [Dygraphs 2009].

A Figura 2 mostra um exemplo de gráfico interativo (na tela de um navegador) utilizando a biblioteca *dygraphs*.



**Figura 2. Gráfico Produzido com o Dygraphs**

Fonte: <http://dygraphs.com/>

A instalação do *Dygraphs* é realizada digitando o comando do Código 3.

```
1 $ sudo wget -P /var/www/html http://dygraphs.com/1.1.1/dygraph-combined-dev.js
```

**Código 3. Instalação do Dygraphs**

Concluída a etapa de instalação de programas foi iniciada a etapa das configurações para utilização da biblioteca gráfica.

## Metodologia

A etapa de configuração das ferramentas *open source* começa pelo Arduino, utilizando seu ambiente de desenvolvimento integrado (Integrated Development Environment, IDE). Assim, é sugerido o Código 4 para realizar a leitura de temperatura dentro do sistema térmico didático e imprimir na porta serial os valores de tempo e temperatura.

```

1 int pinoSensor=0; // configura pino zero para o sensor
2 float valorLido=0, temper=0; // inicializa variaveis
3
4 void setup(){
5   Serial.begin(9600); // inicializa porta serial
6 }
7 void loop(){
8   valorLido=analogRead(pinoSensor); // leitura do sensor
9   temper=(valorLido * 0.488); // converte tensao em temperatura
10  Serial.print(millis()/1000); // imprime tempo
11  Serial.print(","); // imprime virgula
12  Serial.println(temper); // imprime temperatura
13  delay(1000); // espera um segundo
14 }

```

#### Código 4. Configuração do Arduino

Em seguida, o Código 5 em *Python* com o nome “salvados.py” salvo no diretório “home” é responsável por ler os valores de tempo e temperatura existentes na porta serial e salvar em um arquivo TXT, chamado “/var/www/html/temperaturas.txt”.

```

1 import time
2 import serial
3
4 arduino = serial.Serial('/dev/ttyACM0', 9600) # abrir serial
5 arduino.setDTR(False) # reseta arduino
6 time.sleep(0.5) # espera meio segundo
7 arduino.flushInput() # apaga buffer
8 i=0
9 while i < 100: # total de leituras
10   serialValue=arduino.readline()
11   with open('/var/www/html/temperaturas.txt','a') as arquivo:
12     arquivo.write('%s' % serialValue) # salva valores
13
14   print '%s' % serialValue # tela
15   time.sleep(1) # espera um segundo
16   i+=1 # incrementa um
17
18 arduino.close() # fecha serial

```

#### Código 5. Script Python para Ler a Serial

A captura dos dados é realizada pelo comando do Código 6.

```

1 $ python salvados.py

```

#### Código 6. Execução do Programa

Depois, o Código 7 em HTML com o nome “testeDygraphs.html” salvo em “/var/www/html/” é configurado com os comandos básicos da biblioteca *Dygraphs* para desenhar o gráfico de temperatura *versus* tempo.

```

1 <html>
2 <head>
3 <script type="text/javascript"
4   src="dygraph-combined-dev.js">
5 </script>

```

```
6 </head>
7 <body>
8 Projeto Controle de Temperatura
9 <p>
10 Aplicacao da Biblioteca Dygraphs
11 <div id="div"
12 style="width:800px; height:400px;">
13 </div>
14 <script type="text/javascript">
15 g = new Dygraph(document.getElementById("div"),
16 "temperaturas.txt", {
17 title: "Temperatura x Tempo",
18 ylabel: "Temperatura (&#176C)",
19 xlabel: "Tempo (s)" }
20 );
21 </script>
22 Pratica Profissional 2016.1
23 </body>
24 </html>
```

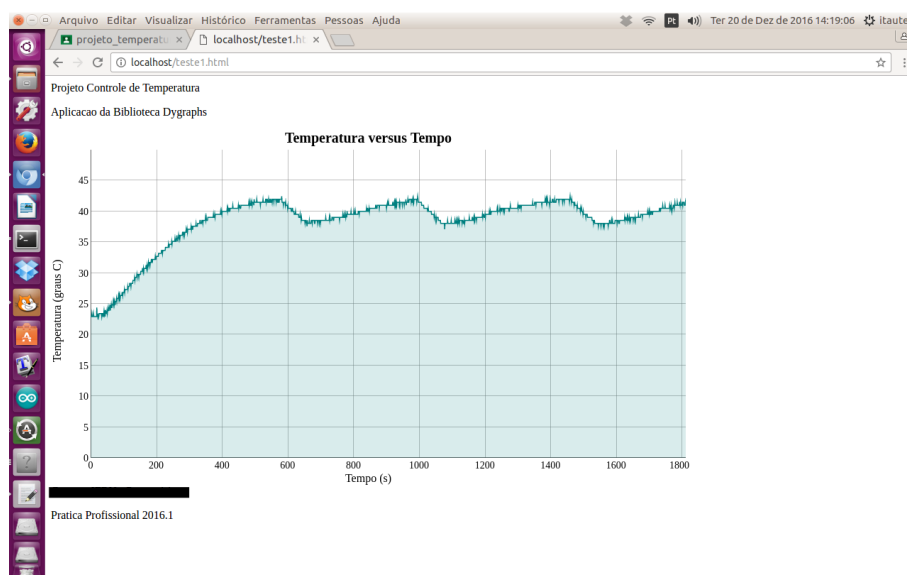
**Código 7. Página HTML**

E como último passo é preciso abrir um *browser* e digitar na barra de endereços “localhost/testeDygraphs.html” para visualizar o gráfico em questão.

## Resultado e Discussões

Para comprovação do correto funcionamento das configurações executadas nos programas foi realizado um primeiro ensaio experimental. Nele, o Arduino foi configurado com um controle liga/desliga. A temperatura máxima escolhida foi de 42 °C, a mínima foi de 38 °C. O tempo entre as leituras foi de 1 s.

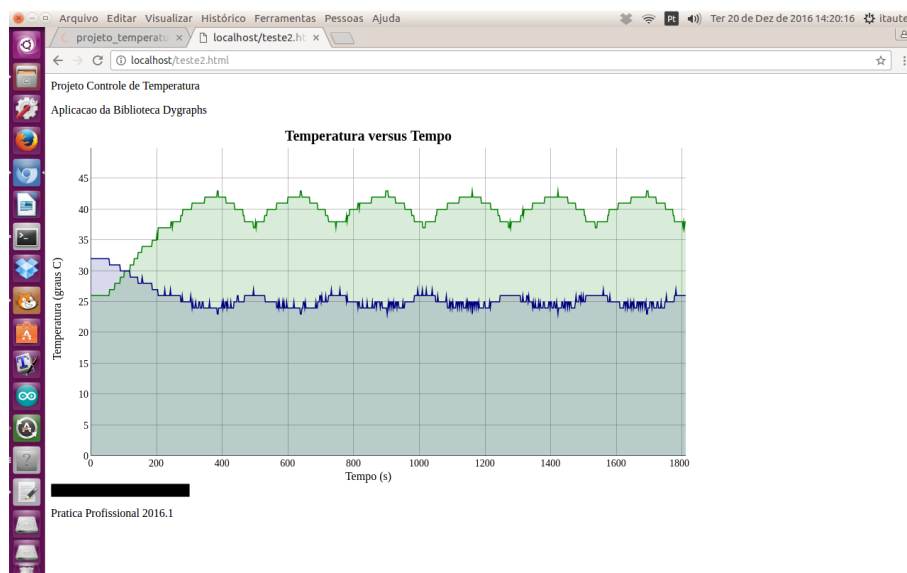
A Figura 3 apresenta o resultado deste primeiro ensaio.



**Figura 3. Primeiro Teste do Dygraphs**

No segundo ensaio, o sensor LM35 foi substituído por um sensor DHT11, que além da temperatura mede a umidade. Por causa dessa modificação o Arduino foi configurado para imprimir na porta serial os valores de tempo, temperatura e umidade. Os valores de temperatura máxima e mínima não foram alterados. A última modificação realizada foi no tempo entre as leituras que aumentou para 2 s.

A Figura 4 apresenta o resultado do segundo ensaio.



**Figura 4. Segundo Teste do *Dygraphs***

Uma vantagem da utilização do *Dygraphs* é a de permitir atualizações automáticas no gráfico, possibilitando um monitoramento em tempo real. Isto é obtido com a configuração do cabeçalho no arquivo HTML, com o Código 8, onde o valor “5” representa o tempo, em segundos, para a realização da atualização automática da página, consequentemente do gráfico.

```

1 <head>
2 <meta http-equiv="refresh" content="5";url=http://localhost/
  teste.html">
3 </head>

```

**Código 8. Complemento na Página HTML**

Por fim, o monitoramento proposto, neste artigo, pode ser realizado a distância utilizando a Internet. Para realizar esta etapa é preciso configurar o roteador. Este deverá permitir o acesso de usuários externos ao arquivo “testeDygraphs.html”.

## Conclusões

O objetivo deste trabalho foi apresentar uma metodologia para monitoramento de grandezas físicas utilizando um sistema térmico didático e ferramentas *open sources*, fáceis de configurar, de rápida aplicabilidade e disponíveis gratuitamente na Internet.

Um sistema térmico didático foi construído e um computador preparado com as ferramentas *open sources* para a realização dos ensaios experimentais. As ferramentas

são: o servidor Apache™, a linguagem de programação *Python* e a biblioteca *Dygraphs*. Depois de todas as configurações dos componentes ocorreu a leitura das temperaturas do sistema térmico e um gráfico foi desenhado na tela do *browser*.

Dois ensaios experimentais foram realizados e os gráficos apresentados com sucesso. No primeiro, somente a temperatura ao longo do tempo foi exibida. No segundo, as leituras de temperatura e umidade foram mostrados. Com base nesses gráficos é possível concluir que o monitoramento gráfico utilizando ferramentas *open source* foi bem sucedido. A principal contribuição, deste trabalho, é propor uma forma alternativa para produção de gráficos, com ferramentas de código aberto.

## Referências

- Alves, F. J. (2013). *Introdução à Linguagem de Programação Python*. Editora Ciência Moderna Ltda., Rio de Janeiro, 3 edition.
- da Silva, J. T., da Silva, J. T., and de Lima, G. F. (2014). Controle e monitoramento de nível utilizando o arduino uno. *IX Congresso Norte Nordeste de Pesquisa e Inovação (CONNEPI), São Luiz, MA*.
- de Lima, G. F. (2013). Utilização da placa arduino no controle de temperatura de um sistema de baixo custo. *VIII Congresso Norte Nordeste de Pesquisa e Inovação (CONNEPI), Salvador, BA*.
- Dilly, R. O. and Mendes, L. F. C. (2015). Aplicação em tempo real de monitoramento de umidade e temperatura utilizando arduino. *Caderno de Estudos em Sistemas de Informação*, 2(1).
- Dygraphs (2009). <http://dygraphs.com/>.
- IFRN (2011). Projeto pedagógico do curso técnico de nível médio em mecatrônica. *Instituto Federal do Rio Grande do Norte*.
- Netcraft (2018). March 2018 web server survey. <https://news.netcraft.com/>.
- Neto, O. P. M. (2013). Análise de bibliotecas para geração de gráficos na *WEB*. Projeto de Graduação, UFRGS, Porto Alegre, RS.
- Ormond, J. G. P. (2006). *Glossário de termos usados em atividades agropecuárias, florestais e ciências ambientais*. Banco Nacional de Desenvolvimento Econômico e Social, Rio de Janeiro, 3 edition.
- Ramos, M. C. and Andrade, V. S. (2015). Desenvolvimento, construção e calibração de uma central de monitoramento de consumo de energia elétrica e de água utilizando o microcontrolador arduino. *XII Encontro Nacional de Eng. e Desenvolvimento Social (ENEDS)*.
- Ribeiro, J. M., dos Santos, E. C. B. F., Oliveira, G. R., Miranda, R. A., and Resende, I. T. F. (2014). Sistema de monitoramento e controle de um processo de distribuição de água, usando arduino e protocolo gsm. *XX Congresso Brasileiro de Engenharia Química (COBEQ) 2014*.
- Soares, L. F., Brandi, J., Godinho, R. A., and Bertoncetto, R. (2017). Monitoramento de variáveis em uma tomada de uso geral - smart tug. *Congresso Técnico Científico da Engenharia e da Agronomia (CONTECC) 2017, Belém, PA*.