

Sistema IoT para Monitoramento de Temperatura e Umidade Ambientes e Acionamento Remoto de Cargas

Matheus de S. Lima¹, Francisco I. de S. Alves¹, Sandro C. S. Jucá¹

¹Eixo da Computação – Instituto Federal de Educação, Ciência e Tecnologia do Ceará (IFCE)
Av. Parque Central, 1315 – Distrito Industrial I – 61939-140 – Maracanaú – CE – Brazil

{matheusdslhc, ivanalves94}@gmail.com, sandrojuca@ifce.edu.br

Abstract. *This paper presents a general purpose and low cost Web System based on the Internet of Things principle for temperature/humidity monitoring and activation of loads in remote environments. The system hardware consists of an ESP8266 microcontroller that controls a DHT11 sensor to collect temperature and humidity data and an LED, which intends to emulate a load. State control and management of these devices occurs through the real-time database of the Firebase platform.*

Resumo. *Este artigo apresenta um sistema Web de propósito geral e baixo custo baseado no princípio da Internet das Coisas para monitoramento de temperatura e umidade e acionamento de carga em ambientes remotos. O hardware do sistema é composto por um microcontrolador ESP8266 que controla um sensor DHT11 para coletar dados de temperatura e umidade e um LED, que emula uma carga. O controle de estados e o gerenciamento destes dispositivos ocorre por meio do banco de dados em tempo real da plataforma Firebase.*

1. Introdução

Com a ascensão de novas tecnologias computacionais e de automação, unidas à consolidação de paradigmas como IoT (*Internet of Things*) e WSN (*Wireless Sensors Network*), é cada vez mais comum que tais mecanismos sejam combinados para realizar o monitoramento de ambientes e o controle de objetos de forma remota. Dentro deste escopo, vem tornando-se frequente o uso de dispositivos microcontroladores ligados à sensores de temperatura e umidade com a finalidade exclusiva de monitorar ambientes onde a temperatura é um parâmetro importante e exerce influência direta na segurança e no modo de operação de equipamentos, seja em ambientes industriais ou agrícolas, proporcionando assim uma diretiva efetiva e econômica [Zhu et al. 2010][Queiroz et al. 2016]. Tal ação é igualmente vista em ambientes de centros de processamento de dados no intuito de prover seu gerenciamento térmico, garantindo a segurança destes locais e o total funcionamento das máquinas presentes na área [Loup et al. 2011][Braga et al. 2017].

O monitoramento remoto de ambientes também pode ser aplicável no âmbito doméstico e com propósitos que vão desde o já citado monitoramento térmico do ambiente, mas que também incluem casos como o monitoramento de temperatura e umidade de solos em jardins e a utilização de sensores de presença para aumentar o quesito de segurança em domicílios. Em todos esses casos a análise dos dados gerados por sensores pode ajudar na obtenção de uma melhor percepção das condições gerais do ambiente e proporcionar ideias para sua melhoria [Bajrami e Murturi 2018].

Este artigo apresenta um sistema *Web* para monitoramento de temperatura e umidade ambientes com foco no gerenciamento de ambientes domésticos, pois proporciona a funcionalidade para acionar cargas de forma remota . A aplicação combina o uso de plataformas livres, como a placa NodeMCU e o ambiente de desenvolvimento integrado do Arduino (usada no desenvolvimento do *firmware* da aplicação), com recursos robustos e seguros da plataforma *Firebase*, como seu banco de dados NoSQL baseado em nuvem, diferindo-se de aplicações *open source* e que usam banco de dados relacionais para trabalhar diretamente com dados de sensores, sejam eles de temperatura ou não, como em [Alves e Jucá 2017]. O sistema usa o banco de dados do *Firebase* para guardar as informações de temperatura e umidade captadas por um sensor DHT11 e o estado de uma carga (ligada ou desligada), representada aqui por um simples LED (*Light Emitting Diode*), mas que pode ser substituído facilmente por um módulo com relé para controlar um dispositivo doméstico em um ambiente domiciliar real.

2. Componentes e Arquitetura do Sistema

Nesta seção são descritos os principais componentes do sistema e como estes encontram-se organizados para comunicarem-se entre si.

2.1. ESP8266

O ESP8266 é um *microchip WiFi* com pilha TCP (*Transfer Control Protocol*)/IP(*Internet Protocol*) completa e capacidade microcontroladora produzida pela empresa chinesa *Espressif Systems* [ESP8266Overview 2018]. O *chip* ESP8266 é combinado com um conector micro-USB (*Universal Serial Bus*) com interface USB-Serial para acoplamento a computadores, um regulador de tensão para 3.3 V e um conversor analógico-digital para compor a placa NodeMCU [FilipeFlop 2018].

2.2. DHT11

O DHT11 é um sensor de temperatura e umidade que permite medir temperaturas de 0 a 50 graus Celcius e umidade na faixa de 0% a 90%, com faixa de precisão de 2 graus para temperatura e 5% para umidade [Arduino&Cia 2013]. Tal sensor ainda conta com um controlador de 8 bits que converte os sinais de temperatura e umidade captados em um sinal serial que pode, por sua vez, ser enviado para um microcontrolador através de seu pino de dados [VidaDeCilício 2018].

2.3. Firebase

O *Firebase* é uma plataforma da *Google* que oferece serviços destinados ao desenvolvimento de aplicações, disponibilizando ferramentas de armazenamento de arquivos, autenticação, banco de dados, *hosting* e dentre outras [Firebase 2018]. A plataforma conta com bibliotecas de integração para a elaboração de interfaces entre seu banco de dados NoSQL em tempo real e dispositivos de *hardware* [FirebaseArduino 2017].

2.4. Arquitetura e Organização do Sistema

O sistema é composto por partes em *hardware*, em *software* e por serviços em nuvem. Na parte de *hardware*, o sensor DHT11 faz leituras de temperatura e umidade local, executa a conversão analógica-digital sobre estes dados e os envia para o NodeMCU, que os valida através de seu *software* embarcado antes de mandá-los para o *Firebase* através da *Internet*.

O NodeMCU também é responsável por conferir o estado da variável *booleana* referentes ao acionamento ou desligamento da cargas ligada à placa através de um processo de sincronização com o banco de dados. Do lado do *Firebase*, as atualizações no banco de dados da plataforma e o controle sobre o acionamento remoto da carga são então mostradas em uma página *Web* e podem ser verificadas pelos usuários do sistema através de um *Browser*. Um fluxograma do sistema é mostrado na Figura 1.

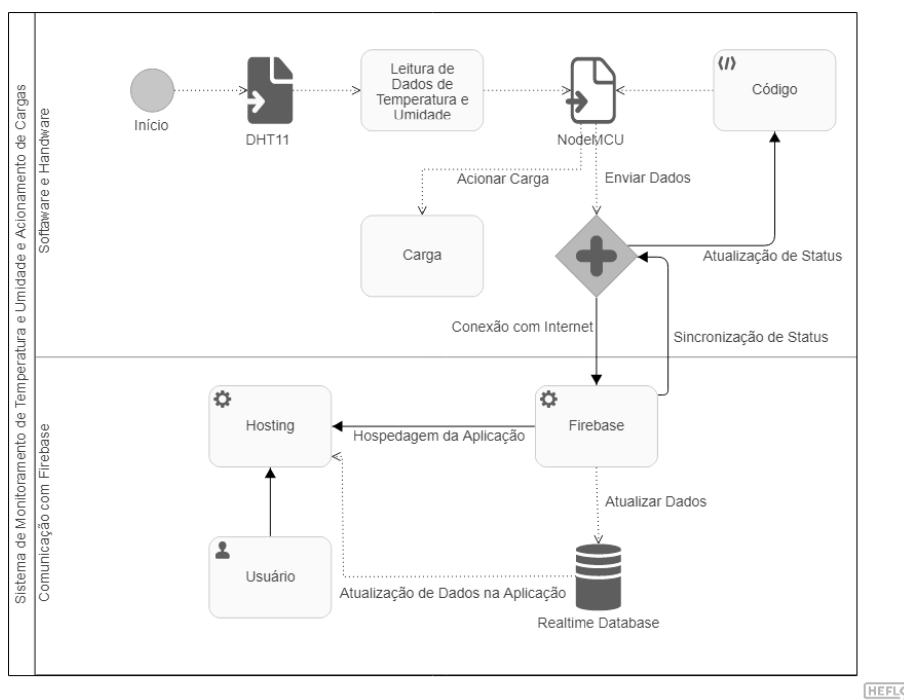


Figura 1. Arquitetura do Sistema.

3. Desenvolvimento

Inicialmente foi criado um novo projeto no *Console* do *Firebase* e adquiridos o segredo do banco de dados do projeto e sua URL (*Uniform Resource Locator*). Tais informações possibilitam que as bibliotecas do manipulem os dados obtidos pelo sistema através do *firmware* da placa NodeMCU [DHTSensorLibrary 2016][FirebaseArduino 2017].

Um esquemático do *hardware* do projeto montado sobre uma *proto board* pode ser visto na Figura 2, onde: (1) é a placa NodeMCU, (2) é o sensor DHT11 e (3) é o LED que emula uma carga conectada à placa.

O *firmware* do sistema foi montado para publicar os dados obtidos de temperatura e umidade no banco de dados do *Firebase* a cada dois minutos e consultar por mudanças na variável *booleana* referente ao estado da carga ligada ao NodeMCU. A Figura 3 mostra a estrutura do banco de dados dentro do *Firebase*.

A aplicação *Web* desenvolvida foi executada em um servidor local padrão disponibilizado pelo pacote de ferramentas do *Firebase* chamado *firebase-tools* que é baseado em *NodeJS* e permite gerenciamento do *hosting* e do banco de dados da aplicação através de um terminal de linha de comando [FirebaseCLI 2018].

O botão em questão encontra-se diretamente relacionado à variável "carga" no banco de dados de modo que o valor *booleano* da variável seja alterado a cada clique no botão apresentado pela aplicação. A sincronização do banco de dados com a placa NodeMCU é feita instantaneamente a cada clique no botão liga/desliga ou a cada atualização de dados de temperatura e umidade.

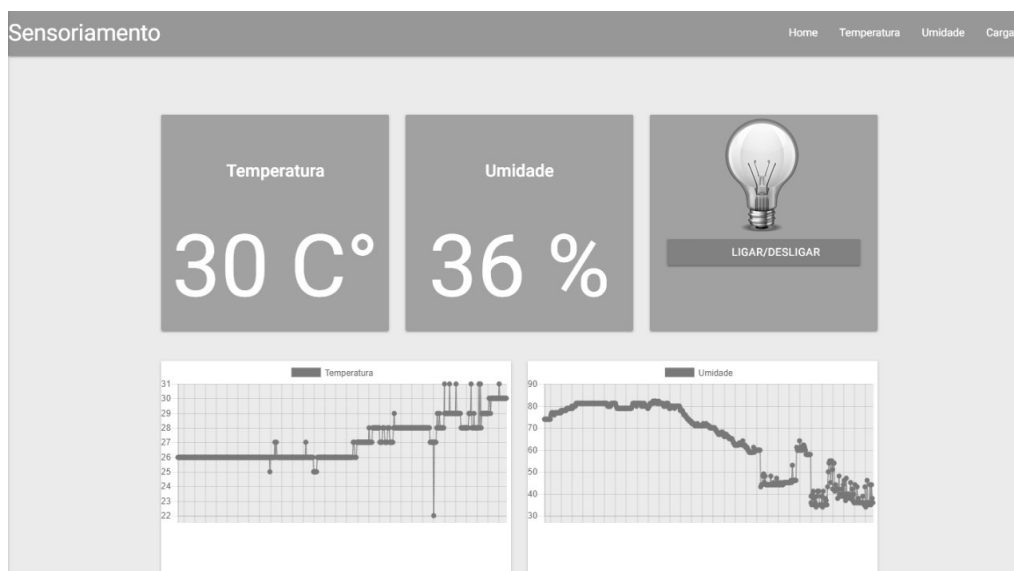


Figura 4. Front-end da aplicação de sensoriamento.

5. Considerações Finais e Trabalhos Futuros

Foi possível concluir através do desenvolvimento desse sistema de monitoramento, com princípios baseados em IoT, que aplicações microcontroladas de sensoriamento são viáveis com a utilização de poucos componentes, necessitando-se apenas dos dispositivos físicos em si e das ferramentas e bibliotecas certas para o desenvolvimento da aplicação que aliadas à plataformas poderosas como o *Firebase*, que lhe oferece inúmeros recursos robustos para aplicações, facilitam o desenvolvimento desse tipo de sistemas, possibilitando a evolução de forma rápida e eficaz de aplicações dessa natureza.

A intenção é continuar a desenvolver o projeto para integrar os recursos de autenticação disponibilizados pelo *Firebase*, provendo assim um nível ainda maior segurança aos usuários da aplicação. Para o monitoramento a nível local sem a necessidade de acesso à aplicação por meio de algum dispositivo, a intenção é exibir tantos os dados de temperatura e umidade quanto os gráficos dessas duas variáveis através de um *display OLED (Organic LED)*. Outras ideias como escolher através da plataforma *Web* as saídas dos pinos da placa NodeMCU que ficarão ativos, o intervalo de tempo das atualizações e o tipo de sensor de temperatura e umidade que será utilizado também estão entre as cotadas para o aprimoramento do sistema como um todo.

Referências

Alves, A. C. e Jucá, S. C. S. (2017). Wireless monitor-aplicativo web livre para receber e mostrar dados provenientes de equipamentos iot. Em *Livro Anais - Artigos e Minicursos*, volume 1, páginas 17–22. III Escola Regional de Informática do Piauí.

- Arduino&Cia (2013). Sensor de umidade e temperatura dht11. Disponível em: <https://www.arduinoecia.com.br/2013/05/sensor-de-umidade-e-temperatura-dht11.html>. Acesso em 15/06/2018.
- Bajrami, X. e Murturi, I. (2018). An efficient approach to monitoring environmental conditions using a wireless sensor network and nodemcu. *e & i Elektrotechnik und Informationstechnik*, 135(3):294–301.
- Braga, A. R., Maciel, F. A. O., Almeida, R. L., Aguilar, P. A., Gomes, D. G., e Andrade, R. M. (2017). Gerenciamento térmico e elétrico de um centro de dados utilizando sensoriamento iot. Em *SBCUP - 9º Simpósio Brasileiro de Computação*. XXXVII Congresso da Sociedade Brasileira de Computação.
- Chart.js (2016). Simples html5 charts using canvas tag. Disponível em: <https://github.com/chartjs/Chart.js>. Acesso em 15/06/2018.
- DHTSensorLibrary (2016). Adafruit dht humidity and temperature unified sensor library. Disponível em: <https://github.com/adafruit/DHT-sensor-library>. Acesso em 15/06/2018.
- ESP8266Overview (2018). Esp8266 overview. Disponível em: <https://www.espressif.com/en/products/hardware/esp8266ex/overview>. Acesso em 15/06/2018.
- FilipeFlop (2018). Módulo wifi esp8266 nodemcu esp-12. Disponível em: <https://www.filipeflop.com/produto/modulo-wifi-esp8266-nodemcu-esp-12/>. Acesso em 15/06/2018.
- Firebase (2018). Documentação firebase. Disponível em: <https://firebase.google.com/docs/?hl=pt-br>. Acesso em 15/06/2018.
- FirebaseArduino (2017). Repositório github para biblioteca firebasearduino. Disponível em: <https://github.com/firebase/firebase-arduino/>. Acesso em 15/06/2018.
- FirebaseCLI (2018). The firebase command line interface (cli). Disponível em: <https://github.com/firebase/firebase-tools>. Acesso em 15/06/2018.
- Loup, T., Torres, M., Milian, F., e Ambrosio, P. (2011). Bluetooth embedded system for room-safe temperature monitoring. *IEEE Latin America Transactions*, 9(6):911–915.
- Queiroz, T. A., Dias, D. L., Araújo, P. H., Figueiredo, R. P., e Jucá, S. C. (2016). Sistema embarcado linux para análise de sensores de temperatura dht11 e lm35.
- VidaDeCilício (2018). Dht11 - sensor de umidade e temperatura. Disponível em: <https://www.vidadesilicio.com.br/dht11-sensor-umidade-e-temperatura>. Acesso em 15/06/2018.
- Zhu, Q., Dongmei, Z., e Xunwen, S. (2010). Distributed remote temperature monitoring and acquisition system based on can bus. Em *Prognostics and Health Management Conference, 2010. PHM'10.*, páginas 1–4. IEEE.