

# Aplicação IoT Multiplataforma para Monitorar Temperatura e Umidade e Acionar Cargas em Ambientes Remotos

Matheus de S. Lima<sup>1</sup>, Francisco I. de S. Alves<sup>1</sup>, Sandro C. S. Jucá<sup>1</sup>

<sup>1</sup>Eixo 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 multiplatform system (with a web version and a mobile version) based on the Internet of Things principle to monitor temperature and humidity, as well as turn devices on and off, in remote environments. Temperature and humidity data collection is done through the sensor DHT11 controlled by an ESP8266 microcontroller that subsequently sends such information to a cloud database*

**Resumo.** *Este artigo apresenta um sistema multiplataforma (com uma versão Web e uma versão mobile) baseado no princípio da Internet das Coisas para monitorar temperatura e umidade, além de ligar e desligar dispositivos, em ambientes remotos. A coleta de dados de temperatura e umidade é feita através do sensor DHT11 controlado por um microcontrolador ESP8266 que posteriormente envia tais informações para um banco de dados na nuvem.*

## 1. Introdução

A combinação cada vez mais frequente entre conceitos como IoT (*Internet of Things*) e WSN (*Wireless Sensors Network*) propicia que o monitoramento de ambientes e o controle de objetos de forma remota seja feita de modo cada vez mais automatizado. A junção de dispositivos microcontroladores com sensores de temperatura e umidade possibilita que se monitore com melhor eficácia ambientes onde estas grandezas são parâmetros de grande importância devido à influência que exercem diretamente sobre o ambiente observado. Com isso, 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].

Visando todo este escopo de atuação, este artigo apresenta uma aplicação multiplataforma (com uma versão *Web* e uma versão em aplicativo para *smartphone*) de propósito geral para monitoramento de temperatura e umidade, podendo ser utilizado em qualquer ambiente que se tenha a necessidade de medir essas grandezas. A aplicação também possui suporte para o acionamento remoto de cargas, fazendo dela um projeto de Internet das Coisas.

O sistema como um todo é composto pela integração de uma placa NodeMCU, onde o *firmware* da aplicação é executado; um sensor DHT11, responsável por capturar os dados de temperatura e umidade do ambiente; o banco de dados não relacional baseado em nuvem da plataforma *Firestore* para guardar as informações obtidas pelo sensor e o estado (ligado ou desligado) dos dispositivos ligados à placa; e o protocolo MQTT.

## 2. Estado da Arte

Um exemplo claro de aplicação que envolve monitoramento de temperatura e umidade ocorre em ambientes de centros de processamento de dados. A principal finalidade é prover o gerenciamento térmico de tais ambientes, garantindo que esses locais permaneçam seguros, certificando-se que máquinas presentes na área atuem com total funcionalidade [Loup et al. 2011][Braga et al. 2017].

A aplicação de monitoramento de temperatura de umidade também é observada em setores agrícolas. Em [Makleyston e Correia 2013], o intuito de monitorar tais grandezas é interpretá-las e repassá-las aos usuários por meio de gráficos exibidos em um sistema *Web*, permitindo assim uma avaliação mais precisa dos níveis dessas grandezas, já que estas influenciam diretamente na colheita e no tempo de irrigação.

O trabalho de [Benaglia et al. 2017] mostra a atuação do monitoramento de temperatura e umidade aplicado à apicultura (criação racional de abelhas), pois os sistemas de criação de abelhas sofrem influências direta destas variáveis ambientais, que por sinal podem afetar o desenvolvimento da colônia e as características físico-químicas do mel. O objetivo do trabalho é avaliar quão eficiente é a termorregulação de caixas de dois materiais diferentes (madeira e isopor) através de sensores.

Ambientes laboratoriais também precisam de controle sobre temperatura e umidade do ar dependendo dos materiais que são manipulados internamente e das suas próprias condições de uso. A utilização deste tipo de abordagem quanto ao monitoramento das condições de um espaço laboratorial é observado em trabalhos como [Michel et al. 2008][Braga et al. 2009][de Oliveira Júnior et al. 2016].

## 3. Materiais e Métodos

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

### 3.1. ESP8266

O ESP8266 é um *chip WiFi*, com pilha TCP (*Transfer Control Protocol*)/IP (*Internet Protocol*) completa, com capacidade microcontroladora que combinado com um conector micro-USB (*Universal Serial Bus*) com interface USB-Serial para conexões com computadores, um regulador de tensão para 3.3 V e um conversor analógico-digital compõem a placa NodeMCU [FilipeFlop 2018].

### 3.2. 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% além de apresenta um controlador de 8 bits para a conversão dos dados captados em um sinal serial que pode, por sua vez, ser enviado para um microcontrolador através de seu pino de dados [Gay 2018].

### 3.3. 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*, dentre outras [Firebase 2018].

### 3.4. MQTT

MQTT (do inglês *Message Queue Telemetry Transport*) é um protocolo bastante utilizado em comunicações de IoT e desenvolvido com base na pilha TCP/IP [Yuan 2018]. Tal protocolo trabalha através do padrão de troca de mensagens Publicador/Subscriber, onde um elemento da rede que deseja receber determinadas informações de um publicador faz uma requisição destas para um outro elemento intermediário no processo de comunicação, chamado *broker*, que é capaz de gerir publicações e subscrições. Os elementos que desejam publicar informações também o fazem através do *broker*, enviando-lhe as informações que possuem [Barros 2015].

### 3.5. Ionic

Ionic é um *framework* usado para desenvolver aplicativos móveis multiplataforma. Tem como características o fato de ser *Open Source* e utilizar soluções empregadas na construção de *Front-ends* de soluções para *Web* como o HTML5 (*HyperText Markup Language*), o CSS3 (*Cascading Style Sheets*) e o *Javascript* [DevMedia 2015].

### 3.6. Organização do Sistema

O sistema em geral é composto por partes em *hardware*, em *software* e por serviços em nuvem como ilustrado na Figura 1, que o descreve com base na versão *Web* da aplicação. O sensor DHT11 é responsável por ler dados de temperatura e umidade local, executar a conversão analógica-digital sobre esses dados e ao final, enviá-los ao NodeMCU para validação através de seu *firmware* antes de finalmente mandá-los para o *Firebase*.

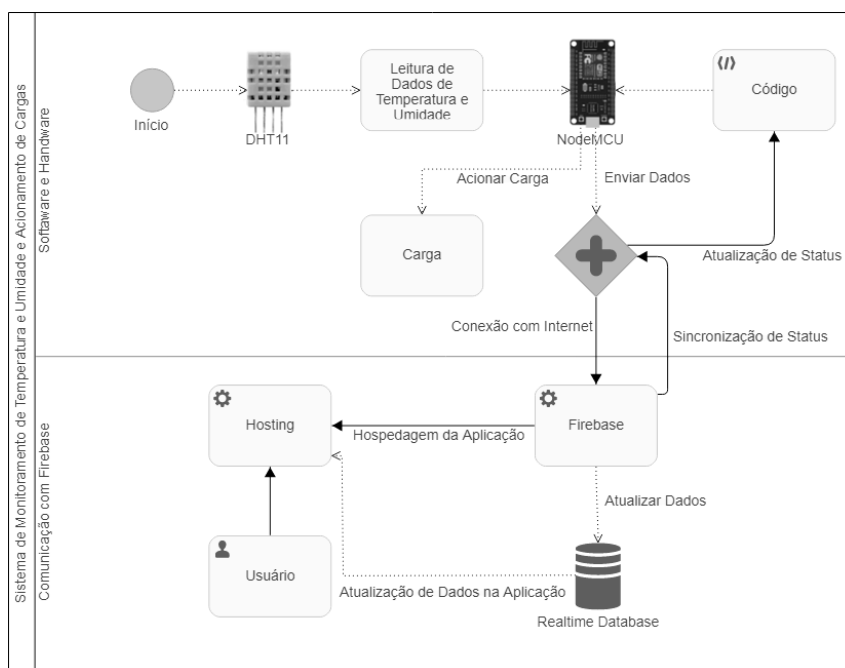


Figura 1. Arquitetura Geral do Sistema.

Outra ação do NodeMCU é conferir, por meio de um processo de sincronização com o banco de dados em nuvem, o estado das variáveis referentes ao estado das cargas ligadas à placa. Enquanto isso, as últimas atualizações de temperatura e umidade são exibidas diretamente tanto na aplicação móvel quanto na página *Web* para que assim possam ser verificadas pelos usuários do sistema.

## 4. Desenvolvimento

Para o serviço *Web* foi criado um projeto no *Console* do *Firebase* que proporcionasse o uso de bibliotecas do sensor DHT11 de modo que os dados obtidos pudessem ser diretamente tratados no *firmware* da placa NodeMCU para posteriormente armazená-los no banco de dados da nuvem [DHTSensorLibrary 2016][FirebaseArduino 2017].

O *firmware* do sistema foi montado no intuito tanto de publicar (um intervalo de tempo de dois minutos) os dados obtidos de temperatura e umidade diretamente no banco de dados quanto de realizar consultas em relação aos estados das possíveis cargas ligadas ao NodeMCU.

A aplicação *Web* foi desenvolvida para ser executada diretamente no *Firebase* através do seu serviço de *hosting* com seu *front-end* baseado em HTML5 e CSS3, devido a compatibilidade com os principais *browsers*, enquanto o *back-end* executa *Javascript* com auxílio da biblioteca *Chart.JS* para a exibição de gráficos referentes ao monitoramento de temperatura e umidade no decorrer do tempo de captura de dados [Chart.js 2016].

A versão mobile da aplicação também utiliza das mesmas tecnologias de desenvolvimento da versão *Web* mas com o diferencial de ter sido desenvolvida através do *framework* Ionic, utilizar arquivos JSON (*JavaScript Object Notation*) para estruturar o banco de dados, usar autenticação via *login* e a biblioteca *Highcharts* para montar os gráficos dentro da aplicação.

## 5. Resultados

A aplicação *Web* possibilitou o acesso via *Internet* aos valores de temperatura e umidade contidos no banco de dados. A apresentação desses dados também pode ser observada através de dois gráficos distintos que mostram as variações das duas grandezas ao decorrer do dia. A aplicação também apresenta funcionalidade para o acionamento de uma carga de forma remota, neste caso específico, limitando-se apenas a um elemento. A Figura 2 logo abaixo mostra o resultado final do *front-end* da aplicação *Web*.

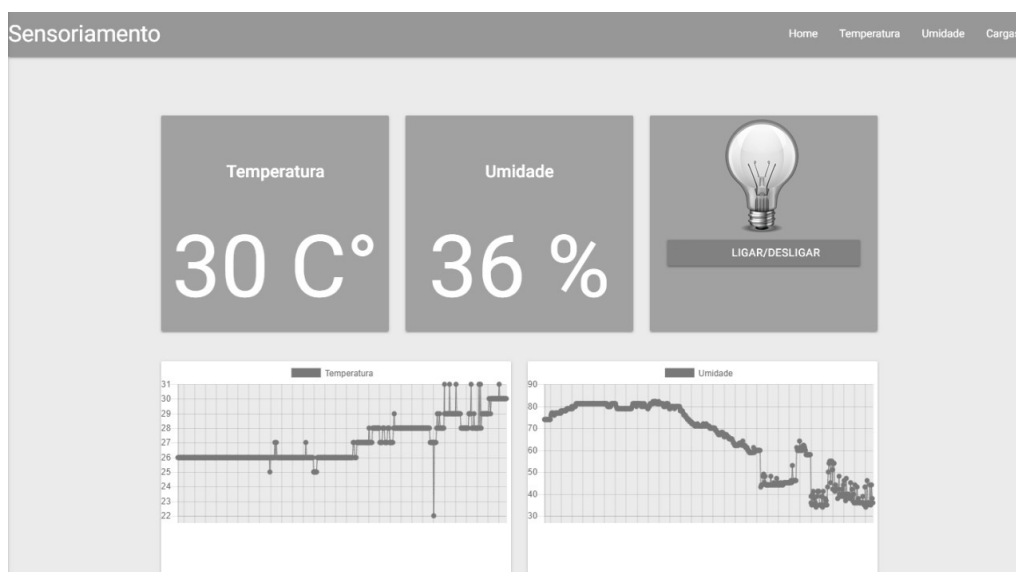
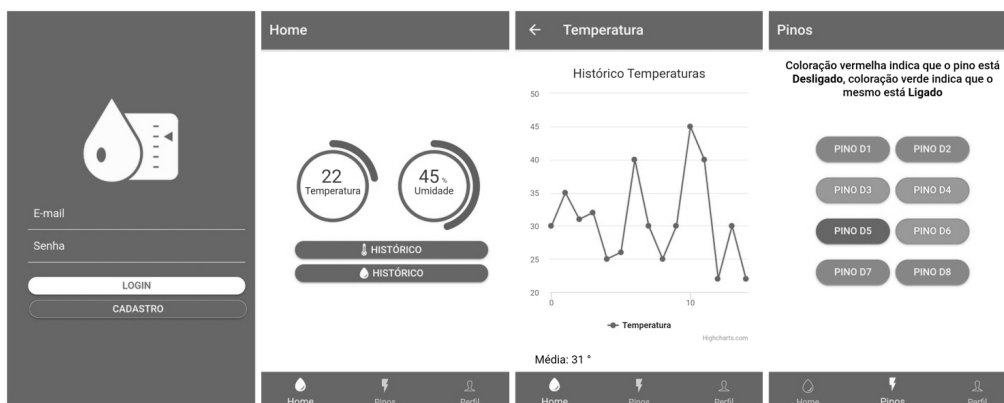


Figura 2. Front-end da aplicação *Web*

A aplicação móvel, ilustrada na Figura 3, possui dois diferenciais: um sistema de *login* baseado nos quesitos de autenticação do *Firestore* e a possibilidade de acionamento de mais de uma carga ligada à placa NodeMCU. As funcionalidades de exibição de gráficos se mantêm, porém apresentando dados referentes à média das grandezas no dia.



**Figura 3. Telas da aplicação *mobile***

Para os dois tipos de aplicação, os botões para acionar cargas de modo remoto encontram-se diretamente relacionado à variáveis com valores *booleanos* (*true* para ligado e *false* para desligado) que são alternados a cada clique nos botões apresentados em cada uma das aplicações. Sendo assim, a placa NodeMCU realiza a sincronização com o banco de dados de modo instantâneo a cada clique em qualquer um dos botões liga/desliga ou a cada atualização de dados de temperatura e umidade.

## 6. Considerações Finais e Trabalhos Futuros

Podemos concluir após o desenvolvimento dessa aplicação multiplataforma com princípios baseados em IoT que o monitoramento e controle de ambientes remotos são viáveis utilizando-se pouquíssimos recursos, combinando-se apenas dispositivos físicos microcontroladores com *frameworks* poderosos auxiliados por bibliotecas robustas. Outro fator importante de destaque é o uso de plataformas poderosas como o *Firestore*, principalmente por oferecer de modo gratuito inúmeros recursos úteis à aplicações do tipo, facilitando por completo a implementação destas, assim como a apresentada nesse artigo.

Para o futuro a intenção é desenvolver algumas funcionalidades com a intenção de aprimorar o projeto. Dentre estas estão a possibilidade de gerar relatórios que compõem dados importantes como média, valores máximos e mínimos, dentre outros, dentro de intervalos de tempo definidos pelo usuário. Outras ideias para a aplicação envolvem escolher através da aplicação *Web* as saídas dos pinos da placa NodeMCU que ficarão ativos, assim como é feito no aplicativo móvel, estabelecer intervalos de tempo para atualização de dados e selecionar que tipo de sensor de temperatura e umidade que será utilizado.

## Referências

- 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.

- Barros, M. (2015). Mqtt - protocolos para iot. Disponível em: <https://www.embarcados.com.br/mqtt-protocolos-para-iot/>. Acesso em 11/09/2019.
- Benaglia, B. G. E. et al. (2017). Sensores no monitoramento da temperatura e umidade interna de colônias de apis melifera africanizadas alojadas em caixa de madeira e ou isopor. Master's thesis, Universidade Estadual de Maringá.
- 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.
- Braga, A. R., Michel, H. C., Braga, L. C., e MP, C. (2009). Aplicação de redes de sensores sem fio para o monitoramento de variáveis ambientais.
- Chart.js (2016). Simple html5 charts using canvas tag. Disponível em: <https://github.com/chartjs/Chart.js>. Acesso em 11/09/2019.
- de Oliveira Júnior, A. J., de Souza, S. R. L., Vicentin, T. A., Glavina, A. S. G., Rodrigues, M. T., e Bailo, W. D. (2016). Uso de dispositivo movel em analise de limites de tolerância para exposição ao calor em um laboratório.
- DevMedia (2015). Introdução ao ionic. Disponível em: <https://www.devmedia.com.br/guia/ionic/38372>. Acesso em 11/09/2019.
- DHTSensorLibrary (2016). Adafruit dht humidity and temperature unified sensor library. Disponível em: <https://github.com/adafruit/DHT-sensor-library>. Acesso em 11/09/2019.
- FilipeFlop (2018). Módulo wifi esp8266 nodemcu esp-12. Disponível em: <https://www.filipeflop.com/produto/modulo-wifi-esp8266-nodemcu-esp-12/>. Acesso em 11/09/2019.
- Firebase (2018). Documentação firebase. Disponível em: <https://firebase.google.com/docs/?hl=pt-br>. Acesso em 11/09/2019.
- FirebaseArduino (2017). Repositório github para biblioteca firebasearduino. Disponível em: <https://github.com/firebase/firebase-arduino/>. Acesso em 11/09/2019.
- Gay, W. (2018). Dht11 sensor. Em *Advanced Raspberry Pi*, páginas 399–418. Springer.
- 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.
- Makleyston, D. e Correia, F. (2013). Sistema web para gerenciamento de uma rede de sensores sem fio aplicada à agricultura de precisão.
- Michel, H. C., Braga, A. R., Braga, C. M., e Braga, L. C. (2008). Caracterização do perfil de sistemas domóticos via redes de sensores sem-fio. Em *Congresso Brasileiro de Automática*.
- Yuan, M. (2018). Conhecendo o mqtt. volume 25. Disponível em: <https://www.ibm.com/developerworks/br/library/iot-mqtt-why-good-for-iot/index.html>. Acesso em 11/09/2019.