

Uso da IoT para monitoramento de temperatura, umidade e presença em datacenters

Guilherme da Silva Donizetti¹, Luiz Fernando da Costa Rodrigues¹, Márcio A. A. Santana², Eugênio Sper de Almeida^{1,3}

¹FATEC Cruzeiro – Centro Paula Souza (CPS)
Avenida Rotary, 383, Vila Paulista, Cruzeiro, SP – Brasil

²Metrologia Ambiental (COMIT) – Instituto Nacional de Pesquisas Espaciais (INPE)
Cachoeira Paulista, SP - Brasil

³Coordenação de Pesquisa Aplicada e Desenvolvimento Tecnológico (COPDT) –
Instituto Nacional de Pesquisas Espaciais (INPE)
Cachoeira Paulista, SP – Brasil

{guilherme.donizetti, luiz.fernando33,
eugenio.almeida}@fatec.sp.gov.br, {marcio.santana,
eugenio.almeida}@inpe.br

Abstract. *Critical computing environments, such as data centers, need to monitor environmental variables: temperature, humidity and the presence of people. This article presents an Internet of Things-based system for local monitoring of these variables in real time. Through low-cost materials and tools, the system facilitates the identification of points outside the curve and identifies the areas that are at risk in the monitored environment.*

Resumo. *Ambientes críticos de computação, como data centers, necessitam monitorar as variáveis ambientais: temperatura, umidade e presença de pessoas. Este artigo apresenta um sistema baseado em Internet das Coisas para monitoramento local dessas variáveis em tempo real. Através de materiais e ferramentas de baixo custo, o sistema facilita a identificação de pontos fora da curva e o identifica as áreas que estão em risco no ambiente monitorado.*

1. Introdução

Ambientes computacionais críticos e datacenters oferecem serviços contínuos de Tecnologia da Informação (TI) e contém servidores e demais equipamentos necessários para prover os serviços.

De acordo com a norma ANSI/TIA-942 (*Telecommunications Infrastructure Standard for Data Centers*), o controle de temperatura (entre 20°C e 25°C), umidade relativa do ar (40% e 45%) e presença de pessoas são diretrizes para garantir o funcionamento dos recursos e a segurança dos dados (ANSI/TIA, 2012).

Atualmente este monitoramento é realizado utilizando equipamentos proprietários que possuem um alto custo. Uma solução baseada em *Internet of Things (IoT)* permite a

diminuição do custo através da utilização de sensores e microcontroladores de baixo custo e ferramentas de código aberto. Ciuffoletti (2018) apresenta uma revisão de literatura sobre custos de projetos de *IoT*.

IoT é uma rede aberta que reúne objetos inteligentes e possui a capacidade de se auto-organizar e de compartilhar dados, informações e recursos, e com isso agir e reagir às mudanças do ambiente (Madakam et al., 2015). Desta forma, *IoT* se mostra como uma tecnologia apropriada para o monitoramento das variáveis ambientais.

O objetivo deste trabalho é desenvolver um sistema de monitoramento local de temperatura, umidade relativa e presença de pessoas em datacenters e outros ambientes computacionais críticos. Para isso, serão necessárias as seguintes etapas: programação de microcontroladores (MCUs) para aquisição de dados; programação dos recursos para transmissão dos dados; desenvolvimento do armazenamento; conectar ferramentas de visualização de dados históricos.

2. Fundamentação teórica

Em ambientes computacionais críticos, entre as medidas essenciais para garantia de segurança, estabilidade e eficiência dos serviços de TI, está o sistema de controle de acesso e o controle de temperatura e umidade relativa.

Além de controlar os limites de temperatura e umidade, é necessário o controle de sua variação. A temperatura pode variar entre mínimo e máximo ou vice-versa no espaço de 1 hora, permitindo dentro desse tempo variações de até 5 °C (ANÍBAL; RODRÍGUEZ, 2015).

Os dispositivos de computação em datacenters operam de forma ininterrupta e com isso liberaram alta taxa de calor, o suficiente para danificar os próprios dispositivos. Para melhorar a circulação do ar, os equipamentos são postos lado a lados criando fileiras, em que as filas são posicionadas uma de frente com a outra, isso resulta em corredor na parte frontal e outro na traseira dos equipamentos (ANSI/TIA, 2012).

A fim de dissipar o calor, no corredor da frontal dos dispositivos é incluído uma fonte de refrigeração caracterizando um corredor frio. O ar frio passa pelos racks e dispositivos dissipando o calor para o corredor da traseira, formando um novo fluxo de ar quente caracterizando um corredor quente (MENEZES, 2017), como na figura 1.

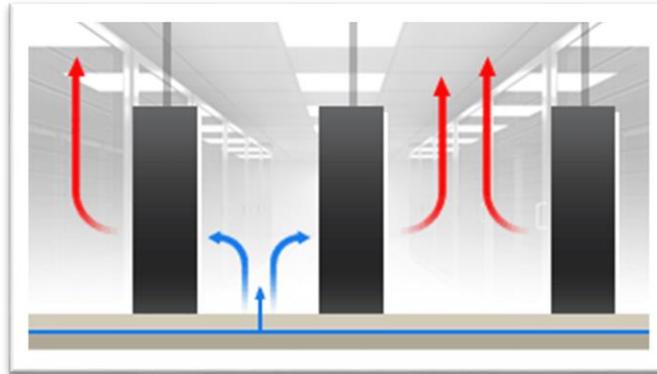


Figure 1. Fluxo de ar frio e quente formando corredores

O monitoramento das variáveis ambientais deve acontecer ao longo da linha central dos corredores frios para não haver superaquecimento pela baixa evasão de calor, nem super esfriamento pelo alto índice de refrigeração.

Para controle de acesso, o ambiente deve ser monitorado por sistema de micro-ondas ou infravermelho e sistema de Circuito Fechado de TV (CFTV) para detecção de acessos e geração de registro de acessos para análise futura (ANSI/TIA, 2012).

Caso as condições ambientais estejam fora das permissões de segurança, é necessário tomada de decisão assertiva e rápida. A *IoT* é uma tecnologia que lida com dados em campo para facilitar as tomadas de decisão, geralmente coleta um grande volume de dados para análises.

É uma rede de dispositivos conectados capazes de interagir com o ambiente por meio da captação de dados, processamento e resposta às condições externas, compartilhando informações entre si (MADAKAM et al., 2015).

As ferramentas escolhidas para criar esse fluxo de trabalho influenciam no desempenho. Para que a *IoT* alcance sua maior eficiência é necessário: softwares amigáveis de visualização dos dados; dispositivos e softwares de diferentes fabricantes devem operar em conformidade, ou estima-se perda de 40% da eficiência; os dados devem ser controlados de acordo com os direitos de uso e estarem constantemente disponíveis (BONILLA et al., 2018).

3. Ferramentas de IoT

As funcionalidades da IoT estão baseadas em seis itens: Reconhecimento (identificar cada dispositivo na rede); Percepção (captar dados para processamento); Comunicação (troca de mensagens entre dispositivos); Processamento (analisar e transformar dados); Serviços (atuar com base nos dados); Concordância (combinar os serviços com os recursos presentes na rede) (FREITAS, 2020).

Estes itens envolvem diferentes dispositivos e programas, por isso é preciso um protocolo para que possam atuarem juntos se comunicando corretamente. O *Message Queue Telemetry Transport* (MQTT) é um protocolo de comunicação assíncrona, o

emissor e receptor da mensagem são independentes, e por isso é apropriado para aplicações de IoT (Al-Fuqaha; Mohammadi, 2015).

O protocolo funciona por meio de um *broker* que é um servidor para receber as mensagens, cada mensagem é publicada em um tópico (endereço) específico e transmitida a todos os dispositivos subscritos ao tópico (Yuan, 2017).

4. Materiais e Métodos

Este projeto adotou ferramentas *open source* para o desenvolvimento da solução. O custo está relacionado apenas à aquisição de sensores e microcontroladores.

Para a captura dos dados de temperatura e umidade relativa foi usado o sensor M5StickC ENV HAT II.R¹. Suas características incluem faixa de operação entre 0 e 60 °C, precisão de 0.2 °C e umidade entre 0% e 100% com precisão de 2%. O sensor M5StickC PIR Hat² é um sensor de indução humana que capta o sinal de infravermelho em até 5 metros, com tempo de resposta de 2 segundos e opera em ambientes com temperatura entre -20 a 80 °C.

O M5StickC ESP32-PICO Mini IoT Development Kit³ é baseado no microcontrolador (MCU) ESP32-PICO⁴ e possui as seguintes funcionalidades: WiFi, Bluetooth e display de 1,14 polegadas acoplado. Os sensores são conectados diretamente ao MCU.

O microcontrolador foi programado utilizando o Arduíno IDE, através da biblioteca *Adafruit_Sensor*. Para que fosse reconhecido pela porta USB, foi usado o driver FTDI que realiza esse reconhecimento e o driver CP210 para o funcionamento do firmware.

Um workflow (Fig. 2) foi implementado utilizando o broker MQTT Mosquitto (1.6.12), Node-Red (1.2.2), InfluxDB (1.6.4), Grafana (7.1.2), agente SMTP chamado sSMTP e o framework Python Flask. Todas essas ferramentas foram programadas para garantir que os dados ambientais sejam capturados, transmitidos para armazenamento e visualizados.

Os tópicos do broker usaram o formato “prédio/sala/posição”. O prédio e a sala têm valores predefinidos, respectivamente CPT e DC. A posição é formada pela identificação do eixo X e Y de onde está o sensor. Um possível tópico é CPT/DC/AB0102, onde o tópico se refere aos dados do prédio CPT, da sala DC, coluna AE e linhas 01 e 02.

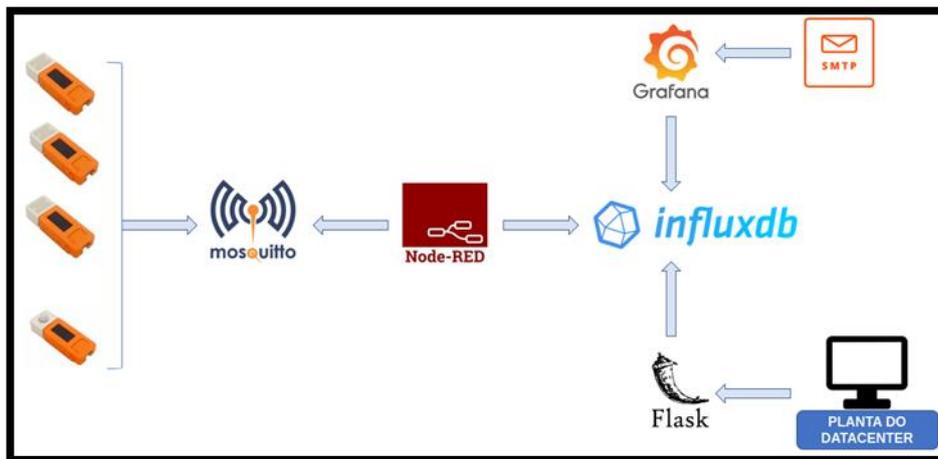


Figure 2. Workflow do projeto

Utilizando Flask, desenvolveu-se uma página Web que apresenta a planta baixa do datacenter, segundo o padrão da ANSI/TIA-942. Nela é apresentada a divisão dos pisos, destacando os corredores quentes e frios e os corredores de racks. Os pisos na horizontal formam uma linha, com identificação numérica, e na vertical formam colunas, identificados por letras.

Ao longo dos corredores frios foram posicionados os sensores com espaço de duas colunas entre si (120 centímetros). Quando os sensores enviam os dados coletados, enviam junto a identificação do piso onde estão localizados, formado pela junção da identificação da coluna e da linha. Então a aplicação consulta os dados de temperatura, umidade e a identificação do piso, com esses dados o programa localiza dentro da planta qual piso mudará de cor de acordo com os valores consultados.

5. Resultados e Discussões

Após coletar os dados, o MCU transmite para o *broker* Mosquitto através do protocolo MQTT. Os dados enviados ao Mosquitto são coletados pelo Node-Red.

O Node-Red possui recursos que executam funções específicas e são chamados “nós”. Os nós *MQTT In* lêem os dados publicados nos tópicos do *broker*. O nó *Functions* aceita qualquer programação funcional em JavaScript, sendo programado para adicionar metadados referentes à localização e modelo dos sensores e à margem de erro dos dados capturados. Por fim, os dados são transmitidos para o armazenamento por meio de um nó específico do banco de dados, e exibidos na área de *debug* no formato JSON.

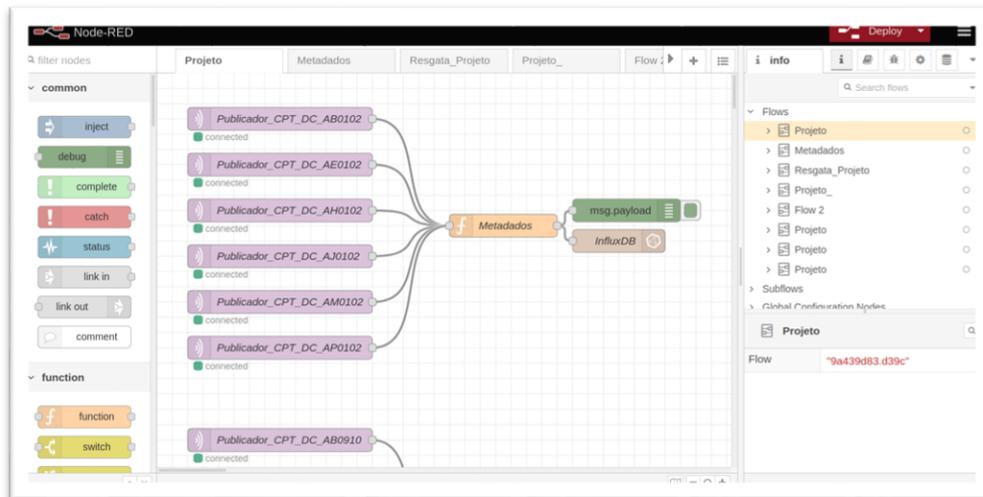


Figura 3. Leitura dos dados no broker pelo Node-Red

O projeto armazena os dados no InfluxDB, um sistema de gerenciamento de banco de dados não estruturado, específico para séries temporais e apresenta alto desempenho na gravação e leitura dos registros.

A visualização dos dados é feita através dos gráficos de linhas que apresentam a média dos valores de temperatura e umidade dos últimos 5 minutos de cada corredor frio, e atualizados a cada 5 segundos. O gráfico de Gauge apresenta o status das variáveis monitoradas dentro dos limites mínimos e máximos.

A cada atualização dos gráficos, as regras de alerta são analisadas, caso detecte algum ponto fora da curva o sistema notifica por e-mail enviando uma imagem do gráfico. Esses recursos foram aplicados utilizando o Grafana.

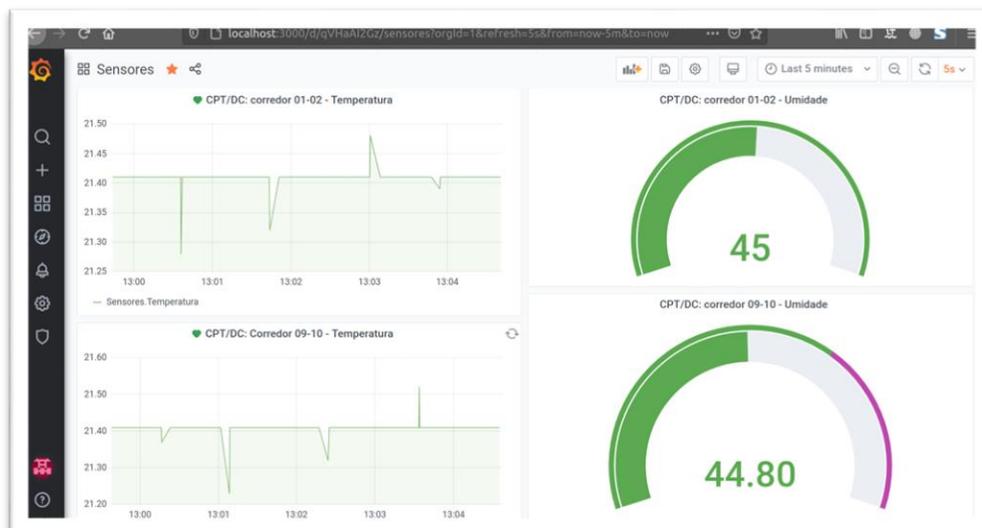


Figura 4. Visualização dos dados históricos dos sensores

Toda vez que o sensor de indução humana registra a presença de pessoas, o armazenamento recebe o valor 1, caso não haja presença é armazenado 0. Esses valores

também são representados em gráfico de linhas por oferecer uma interpretação muito clara.

O projeto inclui um agente SMTP local chamado sSMTP, que realiza o envio dos e-mails de alertas do Grafana. Primeiro as duas ferramentas estabelecem conexão e em seguida o agente utiliza uma conta do Gmail criada especificamente para essa tarefa e o e-mail pode ser transmitido.

Outra visualização dos dados é por meio da planta do datacenter, desenvolvida com ferramentas para Web. Ela consulta a base de dados a cada 5 segundos e seleciona os registros dos últimos 10 segundos para garantir leitura de todos os sensores.

Os dados retornados de cada consulta são analisados para atribuir Verde (se as variáveis ambientais estiverem dentro dos limiares), Amarelo (se as variáveis estiverem em cima dos limiares) e Vermelho (caso alguma variável esteja fora dos limiares). As cores são atribuídas no piso a que se referem os dados coletados e analisados.

PLANTA DO DATACENTER

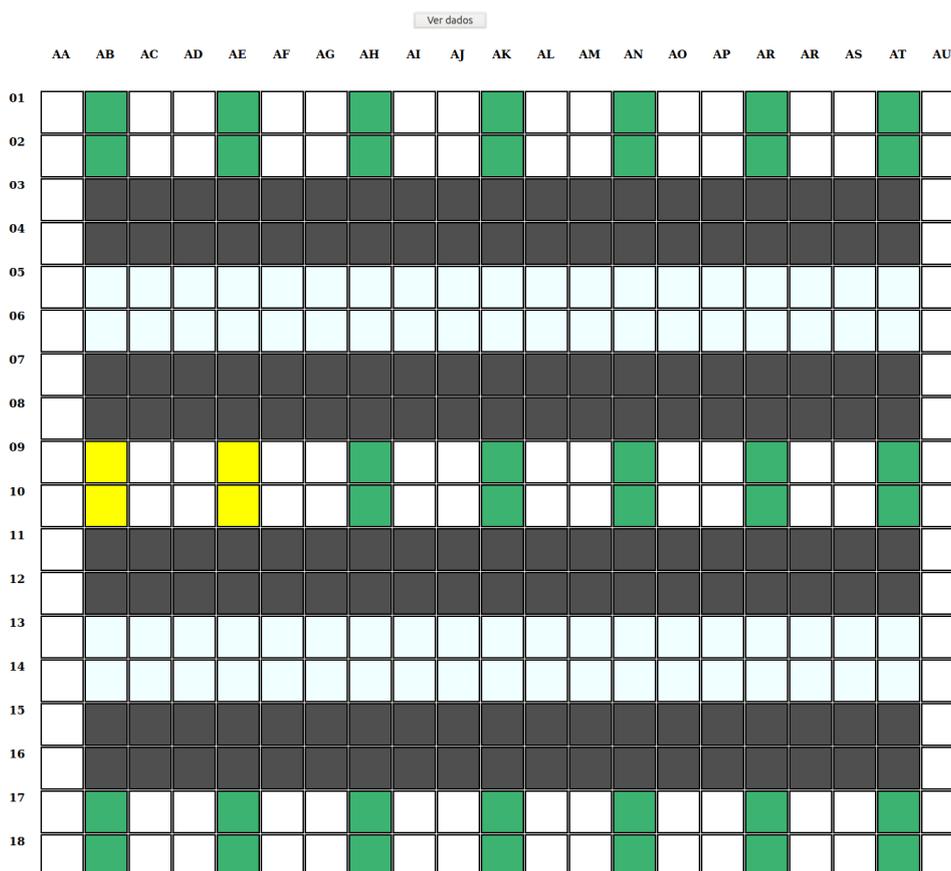


Figura 5. Visualização do estado das variáveis ambientais na planta do datacenter

Caso a aplicação não encontre dados referentes aos últimos 10 segundos, será exibido uma página informando que os dados não foram encontrados. A ferramenta apresenta dados atuais e individuais de cada sensor em campo.

6. Conclusões

Através do workflow e das ferramentas utilizadas nesse trabalho, foi possível o sistema de monitoramento de variáveis ambientais em datacenters utilizando Internet das Coisas com ferramentas de baixo custo.

O sistema opera em tempo real atendendo a criticidade de diferentes cenários de computação e apresenta os dados através de recursos fáceis de serem interpretados e que mostram as condições térmicas dos corredores do datacenter e sobre a presença de pessoas, para tomadas de decisão em menor tempo possível.

A visão da área do ambiente monitorado, através da planta baixa, permite o acompanhamento do estado da temperatura/umidade de cada ponto de coleta de dados, apresentando uma observação específica das condições ambientais.

7. Agradecimentos

Ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), pelo apoio financeiro por meio da bolsa do Programa Institucional de Bolsas de Iniciação em Desenvolvimento Tecnológico e Inovação (PIBITI), e à FATEC Cruzeiro/Centro Paula Souza pelo apoio para a realização desta pesquisa.

Referências

- Al-Fuqaha, A.; Mohammadi, M. (2015) “Internet of Things: A Survey on Enabling Technologies, Protocols and Applications”, https://www.researchgate.net/publication/279177017_Internet_of_Things_A_Survey_on_Enabling_Technologies_Protocols_and_Applications, Julho.
- ANSI/TIA-942, American National Standard Institute / Telecommunication Industry Association. (2012) “Normas de Infra-estrutura de Telecomunicações para Datacenters”.
- Aníbal, D., Rodríguez, N. (2015) “Estudio comparativo de sistemas de virtualización y de seguridad, caso de estudio Museo QCAZ de la PUCE”, <http://repositorio.puce.edu.ec/handle/22000/7685>, Junho.
- Bonilla, S. et al. (2018) “Indústria 4.0: conceitos e fundamentos”, 1.ed. Blucher, São Paulo.
- Ciuffoletti, A. (2018). “Low-cost IoT: A holist approach. Journal of Sensor and Actuator Networks”, 7(2), 19.
- Freitas, A. (2020). “RDEV- Uma solução baseada em IoT para coleta, monitoramento e visualização de temperatura e umidade de ambientes em tempo real”, <https://www.fateccruzeiro.edu.br/projetos/acervo/bd4c9ab730f5513206b999ec0d90d1fb.pdf>, Julho.
- Madakam, S., Ramachandrasastri, R. and Tripathi, S. (2015) “Internet of Things(IoT): A Literature Review”, https://www.scirp.org/pdf/JCC_2015052516013923.pdf.

Menezes, I. (2017) “A influência do DCIM na melhoria da climatização de um data center”, <https://www.riuni.unisul.br/handle/12345/3657>, Junho.

Sperry, G. (2014). “Why your data center should have aisle containment?”, <https://xmission.com/blog/2014/03/25/why-your-data-center-should-have-aisle-containment>, Julho.

Yuan, M. (2017) “Conhecendo o MQTT”, <https://developer.ibm.com/br/technologies/iot/articles/iot-mqtt-why-good-for-iot/>, Julho.