



MONITORAMENTO E CONTROLE DE SALAS DE REUNIÃO ATRAVÉS DE SISTEMA SUPERVISÓRIO E DASHBOARD

MONITORING AND CONTROL OF MEETING ROOMS THROUGH A SUPERVISORY SYSTEM AND DASHBOARD

MONITOREO Y CONTROL DE SALAS DE REUNIONES MEDIANTE SISTEMA DE SUPERVISIÓN Y SALPICADERO EN SALAS DE REUNIONES

Diego Batista Gonçalves ^{1*} & Sara Dereste dos Santos ²

^{1,2}Instituto Federal de Educação, Ciência e Tecnologia de São Paulo

^{1*} dbatistagoncalves@gmail.com ² sarad@ifsp.edu.br

ARTIGO INFO.

Recebido: 14.06.2023

Aprovado: 13.07.2023

Disponibilizado: 25.09.2023

PALAVRAS-CHAVE: Sistema de Controle; Monitoramento; Salas de Reunião; Iridium Mobile; IOT; Supervisório; Dashboard.

KEYWORDS: Control System; Monitoring; Meeting Room; Iridium Mobile; IOT; Supervisory; Dashboard.

PALABRAS CLAVE: Sistema de Control; Monitoreo; Salas de Reuniones; Iridium Mobile; IOT; Supervisión; Dashboard.

*Autor Correspondente: Diego, B. G.

RESUMO

O presente trabalho propõe a aplicação de um sistema de controle e monitoramento de salas de reunião, de forma emulada, para validar o emprego dessas tecnologias nos ambientes corporativos, auxiliando gestores, técnicos e empresas na manutenção desses ambientes que, com o advento do trabalho remoto e de espaços colaborativos, vêm se tornando cada vez mais importantes dentro das empresas. Neste trabalho, foi utilizado o software Iridium Mobile para criação do sistema supervisório que atua como controlador e gateway, enviando as informações para a plataforma TagoIO, por meio do protocolo HTTP, para a criação de uma interface gráfica (dashboard) para a equipe responsável pela manutenção da sala. A controladora RMC3, da empresa Crestron, foi empregada para atuar como Controlador Lógico Programável no sistema projetado para a sala de reunião, controlando as variáveis do tipo booleanas que indicam o estado dos equipamentos (monitor, projetor, matriz de vídeo, ar condicionado, cortina e iluminação). O trabalho permitiu mostrar, de forma clara e precisa, o uso dos equipamentos em uma sala de reunião, onde o usuário final consegue ter ideia do que está ocorrendo no ambiente e se mostrou eficaz quanto ao tempo de atualização dos dados na nuvem, inferior a 0,5s. dos dados na nuvem, inferior a 0,5s.

ABSTRACT

The present work proposes the application of a control and monitoring system for meeting rooms, in an emulated form, to validate the use of these technologies in corporate environments, helping managers, technicians and

companies in the maintenance of these environments that, with the advent of the home office and collaborative spaces, are becoming increasingly important within companies. In this work, the IRIDIUM MOBILE software was used to create the supervisory system that acts as a controller and gateway, sending information to the Tago.IO platform, through the HTTP protocol, to create a graphical interface (dashboard) for the team Responsible for room maintenance. The RMC3 controller, from Crestron, was used to act as a Programmable Logic Controller in the system designed for the meeting room, controlling the Boolean variables that indicate the status of the equipment (monitor, projector, video matrix, air conditioning, curtains and lighting). The work allowed showing, in a clear and precise way, the use of equipment in a meeting room, where the end user can have an idea of what is happening in the environment and proved to be effective in terms of updating time in the cloud, less than 0.5s.

RESUMEN

El presente trabajo propone la aplicación de un sistema de control y monitorización de salas de reuniones, en forma emulada, para validar el uso de estas tecnologías en entornos corporativos, ayudando a directivos, técnicos y empresas en el mantenimiento de estos entornos que, con la llegada de El trabajo remoto y los espacios colaborativos, cobran cada vez más importancia dentro de las empresas. En este trabajo se utilizó el software IRIDIUM MOBILE para crear el sistema de supervisión que actúa como controlador y puerta de enlace, enviando información a la plataforma Tago.IO, a través del protocolo HTTP, para crear una interfaz gráfica (dashboard) para el equipo Responsable de mantenimiento de habitaciones. Se utilizó el controlador RMC3, de la empresa Crestron, para actuar como Controlador Lógico Programable en el sistema diseñado para la sala de reuniones, controlando las variables booleanas que indican el estado de los equipos (monitor, proyector, matriz de vídeo, aire acondicionado, cortinas e iluminación). El trabajo permitió mostrar, de forma clara y precisa, el uso de los equipos en una sala de reuniones, donde el usuario final puede tener una idea de lo que sucede en el ambiente y demostró ser efectivo en cuanto al tiempo de actualización de datos en la nube, menor a 0.5s.



1. INTRODUÇÃO

Nos últimos anos os espaços de trabalho se tornaram muito dinâmicos, com salas de reuniões ganhando cada vez mais importância dentro das empresas. Muitas delas estão utilizando o conceito de escritório não territorial, que consiste em espaços de trabalhos sem demarcações, onde os funcionários se alocam onde acharem melhor (Andrade, 1996). Com isso, controlar e monitorar esses ambientes se tornou imprescindível, uma vez que os escritórios viraram grandes espaços abertos com salas de reuniões em pontos estratégicos.

Sistemas de monitoramento e controle antes empregados na indústria estão, agora, sendo empregados dentro dos escritórios com suas devidas modificações para garantir o melhor funcionamento dos ambientes corporativos e, assim, aumentar a produtividade e melhorar a qualidade de trabalho dos funcionários. Como defende Lautier (1999), o escritório é uma ferramenta de trabalho e, dessa forma, conhecer suas necessidades constitui valor para quem vai utilizá-lo.

Nesse contexto, o uso de tecnologias como sistemas supervisorios, internet das coisas (Internet of Things, IoT), telas (interfaces visuais também chamadas, em inglês, de *dashboards*) e processamento em nuvem para interpretação dos dados têm sido empregadas para melhor gerenciar os ambientes corporativos e auxiliar na tomada de decisões (Pleasant, 2023).

Com o IoT e o advento da tecnologia 5G, milhões de informações que antes passavam despercebidas no dia a dia, poderão ser coletadas e tratadas, em tempo real, de forma a gerar valor para as empresas e pessoas comuns, uma vez que a visualização dos dados facilita o entendimento sobre determinada coisa ou assunto (Okoh, Roy, & Mehnen, 2017). Metodologias como a Goals, Questions & Measures (GQM) que pode ser traduzida como Objetivos, Questões e Medidas, tornam-se importantes na hora da apresentação dessa quantia enorme de informações em forma de *dashboards*, pois visam um conjunto de questões e um conjunto de regras que separam as informações em três níveis, o conceitual, o operacional e o quantitativo, respectivamente. O objetivo define o que deve ser estudado e o porquê, as perguntas tratam das demandas que o projetista/desenvolvedor deve responder, e as medidas são os dados que devem ser coletados para responder às perguntas (Basili, Caldiera, & Rombach, 1994). Além disso, técnicas como *push-pull* usadas para mostrar-filtrar uma informação, determinam como o usuário final irá interagir com os dados apresentados no *dashboard*. A técnica mostrar é acionada assim que outras atividades terminam e como, às vezes, os dados não correspondem com o próximo passo, levam o usuário a usar a técnica filtrar, a fim de descobrir onde está o erro da informação (Janes & Succi, 2009).

Com base no exposto, o objetivo deste trabalho é propor um sistema de controle e monitoramento que seja de fácil operação para o usuário final, que deverá usar esses espaços e interagir com os equipamentos das salas de reunião e, também, para os responsáveis pela manutenção e funcionamento dos sistemas, que poderão fazer uso dos dados coletados a partir da rede para aprimorar os serviços e o controle dos espaços.

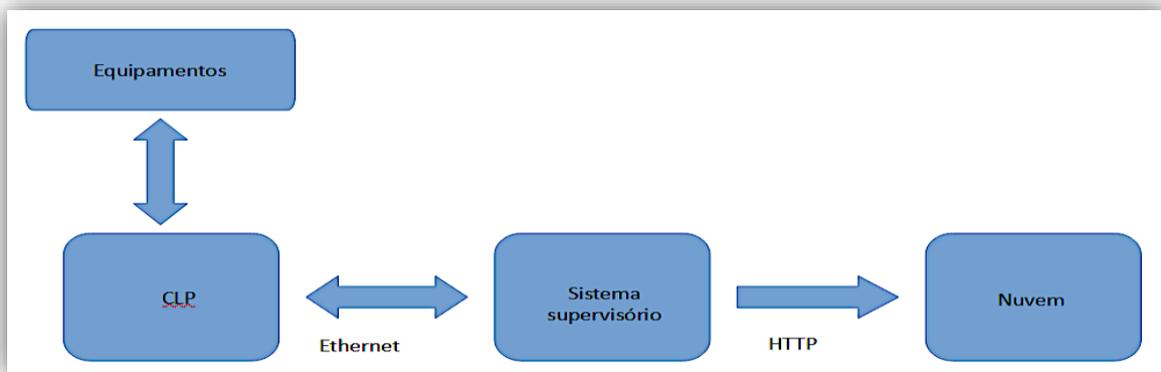


2. REFERENCIAL TEÓRICO

2.1 Sistemas Supervisórios

Segundo Paiola (2018), os sistemas supervisórios possuem telas que são animadas por dados provenientes de diversas fontes de campo que se comunicam através de diversos protocolos existentes no meio industrial. A constituição básica de um sistema supervisório possui um computador pessoal do tipo PC (*personal computer*) conectado a um sistema que tem como base um CLP (controlador lógico programável) e seus periféricos que, através de *interfaces* ou *gateways*, estabelecem a comunicação em todos os níveis de controle (Jurizato & Pereira, 2003). A Figura 1 mostra um exemplo de sistema supervisório básico com a adição de uma plataforma em nuvem para visualização dos dados.

Figura 1. Esquemático de um sistema supervisório básico.



Fonte: Autores

Com a evolução tecnológica, o sistema supervisório deixou de ser uma simples tela com dados do processo e passou a incorporar sistemas multifuncionais, integrando outros sistemas da linha de produção, tais como CRMs (Customer Relationship Management), que é um *software* que reúne informações dos clientes e geram *insights* instantâneos para a equipe de marketing, vendas e bancos de dados na nuvem. Com isso, passou a ser comum encontrar na lista de usuários do sistema diferentes perfis, que vão desde operadores até diretores da empresa (Paiola, 2018). Dessa forma, os sistemas supervisórios atuais devem ser multiplataforma, o que significa permitir acesso às informações a todos os usuários independentemente do dispositivo que utilizam.

2.2 IoT

O termo IoT foi usado pela primeira vez em 2009 por Kevin Ashton, professor e pesquisador do MIT (Massachusetts Institute of Technology), para introduzir a ideia de etiquetas RFID na cadeia de suprimentos da empresa P&G (Procter & Gamble). Segundo ele, se os computadores soubessem tudo o que há para saber sobre as coisas, usando os dados que eles coletam sem interferência humana, eles seriam capazes de rastrear, armazenar os dados e informar tudo sobre aquele produto. Seria possível saber quando as coisas precisam ser substituídas, reparadas e se são novas ou ultrapassadas (Ashton, 2009).

Hoje, vê-se que o conceito de IoT está revolucionando a sociedade. Há milhões de dispositivos no mundo em todo tipo de aplicação, gerando dados que serão usados para



rastrear, contar, substituir, recuperar, entre outras coisas, como imaginou o professor Kevin Ashton. Com isso, os dados que cada pessoa ou coisa gera se tornaram o novo petróleo da era atual e, uma vez organizados, podem fornecer informações valiosas para organizações e pessoas comuns, como visto no trabalho de Lopes e Moori (2020), em uma aplicação de IoT no setor de transporte, armazenamento e varejo que permitiu a tomada de decisão de forma proativa, conectada a recursos e com menos erros, garantindo maior flexibilidade e produtividade no setor.

2.3 Iridium Mobile

O programa Iridium Mobile inclui o controle de iluminação, áudio, vídeo, climatização, segurança, entre outros, tornando-se uma ferramenta poderosa para ser empregada em sistemas supervisórios. Ele possui quatro componentes em sua estrutura. O Iridium Studio é o primeiro deles, sendo o ambiente de desenvolvimento da *interface* gráfica e de projetos com servidores. O segundo componente, chamado de i3 pro, é um aplicativo instalado no dispositivo que o usuário utilizará para controlar e monitorar o seu sistema, podendo ser instalado nos sistemas operacionais IOS, Windows e Android. O terceiro componente é o Iridium Server que age com um *gateway* para o sistema de automação e IoT, sendo indicado para coletar, analisar, exportar e importar dados de um banco de dados externo e criar rotinas de programação. O último componente dessa ferramenta é o banco de *drivers* dos equipamentos que o *software* suporta e que, hoje, conta com mais de 20 fabricantes e 14 protocolos de comunicação (Iridium Mobile, 2022).

2.3.1 Política de Licenciamento Iridium

As licenças Iridium são compradas e armazenadas diretamente no site da Iridium Mobile. Esse serviço está disponível apenas para integradores registrados no site e as licenças são armazenadas no Iridium Cloud, onde são carregadas no aplicativo i3 pro para IOS, Android, Windows e MAC (Iridium Mobile, 2022).

Uma vez registrado no site Iridium Mobile pode-se usar o Iridium em modo de teste, para fins de treinamento e comissionamento antes de comprar a licença. A Iridium organiza seus integradores em três níveis: *bronze*, *silver* e *gold*. Quando um usuário tem o status *bronze*, o modo de teste funciona por 5 minutos, já com o status *silver* ou superior, por 20 minutos (Iridium Mobile, 2022).

Uma vez armazenado no Iridium Cloud, o integrador precisa compartilhar o projeto com o usuário final e o mesmo precisa estar registrado no site Iridium Mobile. O integrador também pode optar por deixar o projeto em modo de teste por até 60 dias ou licenciar o projeto e compartilhá-lo com o usuário final.

2.4 TagoIO

A TagoIO é uma plataforma de IoT que recebe dados de diferentes dispositivos e de diferentes setores da indústria, como sensores de temperatura de câmaras frigoríficas, rastreadores de automóveis, medidores de energia, eletrodomésticos, projetores, videoconferências, sensores de ocupação, entre outros, que se comunicam por diferentes protocolos. Essa plataforma conta com mais de 20 dispositivos certificados, além de



conectores AWS IoT, HTTPS e *endpoint* genérico. Em sua versão gratuita, ela permite 5 dispositivos conectados, criação de 5 *dashboards*, 10 usuários finais, 30 dias de armazenamento de dados, entre outros (TagoIO, 2022).

3. METODOLOGIA

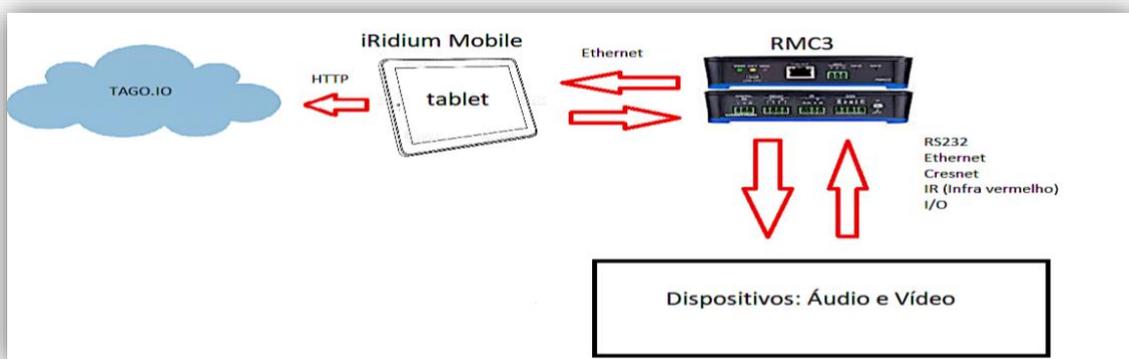
Para a construção de um *dashboard* para auxiliar no monitoramento e manutenção de salas de reuniões em ambientes corporativos, foi tomado como base um processador Crestron, que atua como CLP, se comunicando e controlando todos dispositivos contidos em um ecossistema de uma sala de reunião moderna.

A *interface* de controle foi elaborada com o *software* Iridium Studio que tem a responsabilidade de mostrar de forma animada o estado do sistema e transferir informações pertinentes dele para a plataforma TagoIO, a fim de ser apresentada em um *dashboard* para os gestores responsáveis pelo bom funcionamento do escritório e para a equipe de *facilities* (profissionais responsáveis pela manutenção do espaço).

Os ambientes corporativos utilizam diversos protocolos em seu meio. Por exemplo, um único processador Crestron pode conter vários barramentos de comunicação, tais como: Ethernet; RF; RS232; RS485; Cresnet; e Infravermelho. Cada barramento de comunicação pode controlar um ou mais equipamentos de acordo com o projeto.

Neste trabalho, os principais protocolos utilizados foram: Ethernet; RS232; SCIP (Secure Common Industrial Protocol); Cresnet; e HTTP (Hypertext Transfer Protocol) – como pode ser observado na Figura 2. Utilizou-se o protocolo Ethernet e RS232 para se estabelecer a comunicação com os dispositivos de áudio e vídeo, como microfones, monitores, videoconferência, entre outros, o Cresnet para se comunicar com os dispositivos de controle de iluminação e o HTTP para se comunicar com o servidor na nuvem, onde as informações são armazenadas e onde o *dashboard* foi construído. O protocolo HTTP, embora seja mais lento que o MQTT (Message Queuing Telemetry Transport) para enviar informação para a nuvem (Sakaki & Yokotani, 2019), foi escolhido por ser simples e por possuir uma vasta literatura sobre sua utilização em conjunto com JSON (JavaScript Object Notation), que é um formato de troca de dados muito utilizado para transferências entre um servidor *web* e uma aplicação móvel (Moço, 2015).

Figura 2. Estrutura de comunicação entre equipamentos e plataforma.



Fonte: Autores

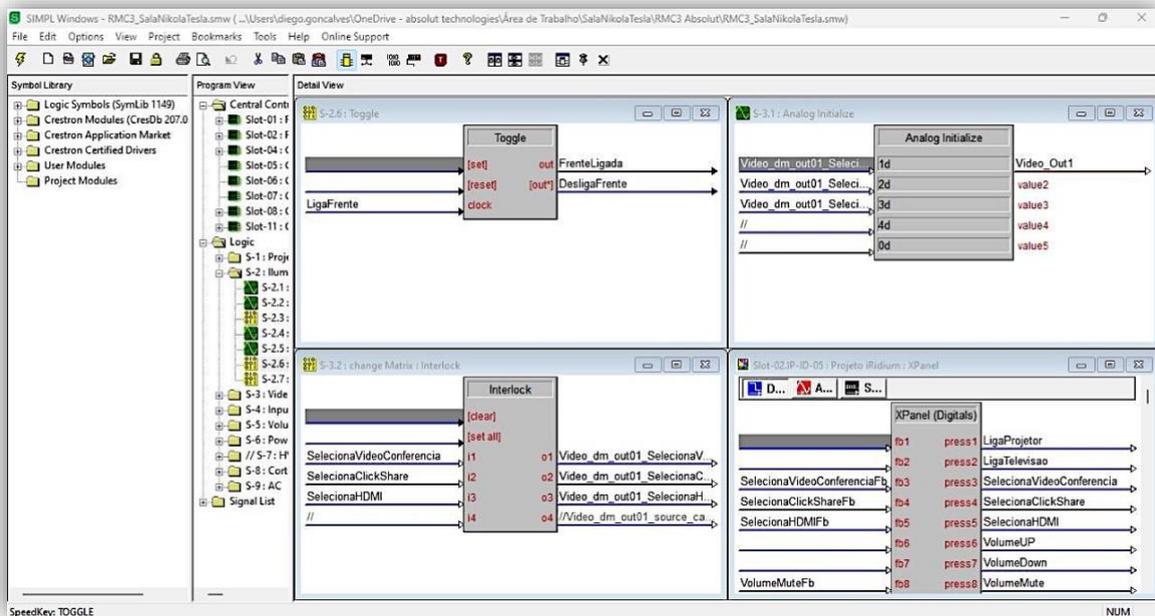


O controle dos equipamentos da sala de reunião foi realizado através de um controlador RMC3, da empresa Crestron, que possui memória DDR3 SDRAM de 256 MB, memória flash de 4 GB e armazenamento externo via porta USB, cinco meios de comunicação que são: Ethernet; Cresnet; USB; RS-232/422/485; e IR/Serial. Também possui duas entradas digitais e dois relés de saída. Sua programação foi realizada através do *software* SIMPL Windows, onde os blocos lógicos foram inseridos e, também, onde se definiu por qual barramento os equipamentos iriam se comunicar com os dispositivos. Com isso, criou-se as variáveis que interligam os blocos de modo a disparar os comandos que controlam os equipamentos. A Figura 3 ilustra um exemplo da tela de programação mencionada.

Com a programação dos equipamentos implementada, importou-se o arquivo gerado no programa SIMPL Windows para o Iridium Studio que, por sua vez, extraiu todas as variáveis criadas para o programa da criação da *interface* de controle e, nesse caso, o *gateway* de comunicação com a nuvem da plataforma TagoIO. Uma vez desenhada a *interface* de controle (Figuras 4 e 5) alocou-se cada variável para o seu respectivo botão.

Com a importação do arquivo do SIMPL Windows, cria-se um *driver* Crestron no *software* Iridium Studio que, por sua vez, precisa das credenciais de comunicação (autenticação SSL) para estabelecer a conexão entre *interface* de usuário e controladora Crestron RMC3. As credenciais necessárias para se estabelecer uma conexão entre o i3 pro e a controladora Crestron são: endereço IP; porta de comunicação; IP ID; modo de comunicação (CIP ou SCIP); Telnet Login; Telnet password; e Telnet SSL.

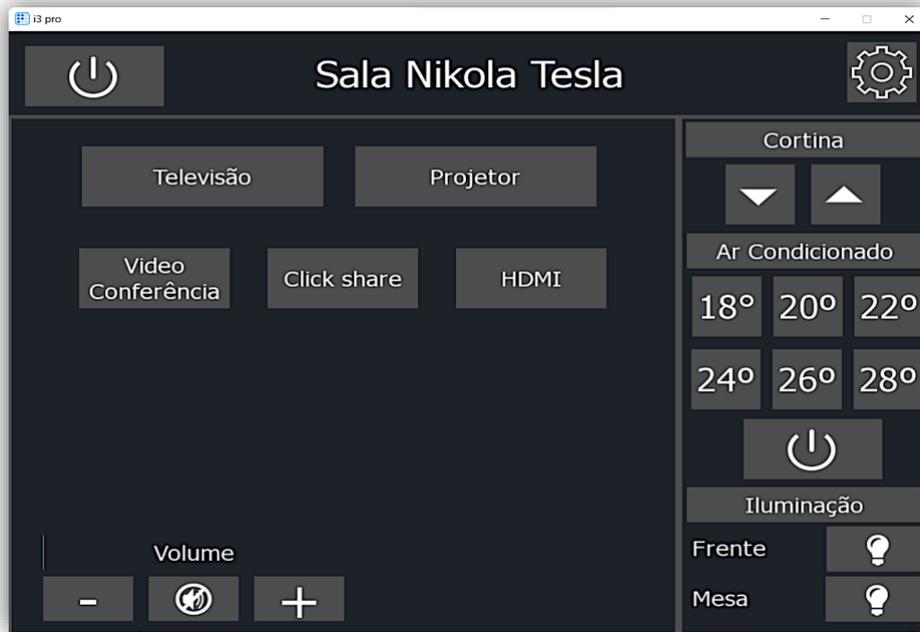
Figura 3. Exemplo da tela de programação Crestron



Fonte: Autores



Figura 4. Interface de controle



Fonte: Autores

Figura 5. Abertura de chamado



Fonte: Autores

Dado o meio de comunicação escolhido que, neste projeto, foi o HTTP, para essa comunicação utilizou-se o método POST, como indicado na documentação disponibilizada no site da empresa, com pequenas mudanças para se adequar à linguagem de programação Java Script utilizada no Iridium Studio que, como mencionado anteriormente, foi responsável pela interface de controle e, também, como gateway de comunicação com a TagIO. Para



essa conexão ocorrer precisa-se inserir dentro do Iridium Studio as credenciais de autenticação que, nesse caso, são: URL (Uniform Resource Locator); porta; e *token*.

A seguir, é apresentado o código que executa a interceptação da informação referente ao estado da variável e a transfere para a nuvem, possibilitando a criação do *dashboard* de monitoramento do sistema.

//Função que envia os dados para a plataforma Tago.IO

```
function fnTagoSend(devicedata)
{
  IR.Log('Post Pressed');
  IR.GetDevice("TagoIO").SendEx({
    Type: "POST",
    Url: "/data",
    Data: devicedata,
    Headers: JSON.Parse({"Content-Type":"application/json","Device-Token": DeviceToken}),
    cbReceiveText: function(text, code, headers) {IR.Log("cbReceiveText "+text+code+headers);},
  }
);
};
```

//Esta função captura a mudança de status da variável e passa nome e //valor para a função que envia os dados para a Tago.IO

```
IR.AddListener(IR.EVENT_TAG_CHANGE, IR.GetDevice("Crestron"), function(name, value)
{
  IR.Log(name+": "+value); // Feedback 1: 100
  if(name=="SelecionaVideoConferenciaFb"){
    //IR.Log(name+": "+value)
    Input = "Videoconferência";
    fnTagoSend(['{"variable": "Input","value": '+'"+Input+'"}'])
  }
}
```

4. ANÁLISE E DISCUSSÃO

Com base nas tecnologias utilizadas neste trabalho foi possível emular o funcionamento de uma sala de reunião e aplicar o conceito de *dashboard* em um sentido mais amplo, onde essas interfaces funcionam como *front-ends* intuitivos e fáceis de usar a fim de monitorar, analisar e otimizar atividades críticas de negócios, permitindo que usuários, em todos os níveis hierárquicos, melhorem suas decisões (Malik, 2005). A partir desse princípio, priorizou-se a criação de um *dashboard* mais amigável, apenas com informações relevantes para os profissionais que atuam no *facilities* das empresas e que, normalmente, se importam mais com o funcionamento de um conjunto de equipamentos do que com as informações individuais de cada dispositivo, o que pode confundir o usuário final. Com isso, foi aplicada a metodologia GQM para criação de um *dashboard*, onde o objetivo se pautou na utilização da sala, as questões foram baseadas em perguntas sobre os equipamentos e as medidas consistiram de valores obtidos a partir do estado desses equipamentos. A Tabela 1 resume esses pontos:



Tabela 1. Metodologia GQM

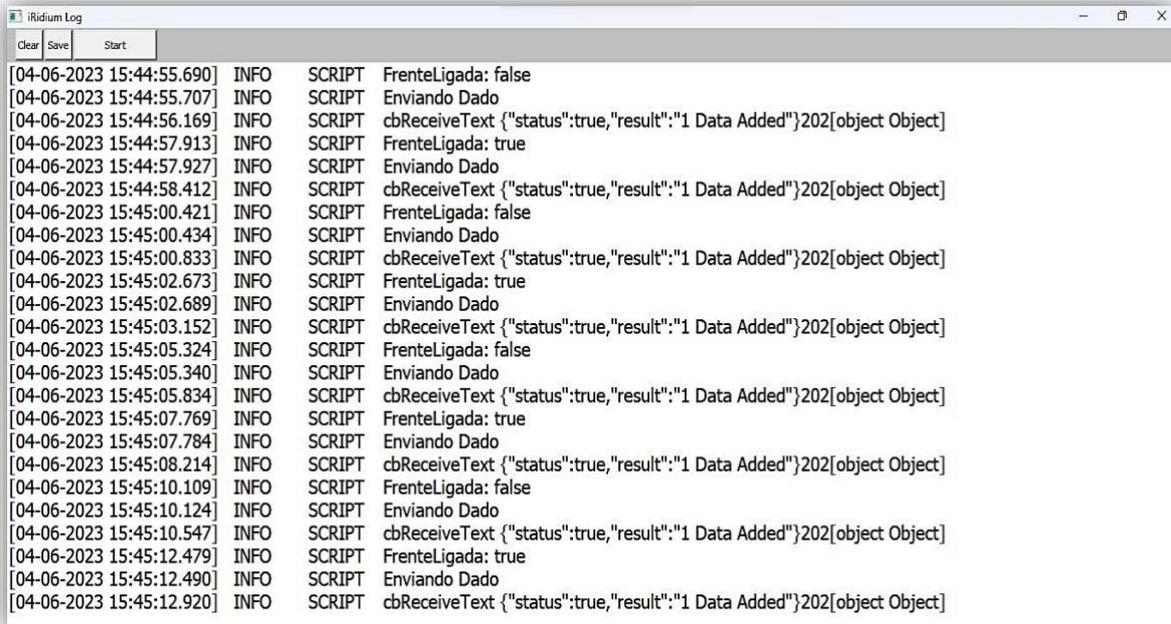
| OBJETIVOS |
|--|
| 1 - AVALIAR A UTILIZAÇÃO DA SALA DE REUNIÃO |
| PERGUNTAS |
| 1- A TELEVISÃO ESTÁ LIGADA? |
| 2 - O PROJETOR ESTÁ LIGADO? |
| 3 - O ÁUDIO ESTÁ MUTADO? |
| 4 - QUAL TEMPERATURA ESTÁ A SALA? |
| 5 - A CORTINA ESTÁ ABERTA OU FECHADA? |
| 6 - A LUZ ESTÁ LIGADA OU DESLIGADA? |
| 7 - O USUÁRIO REPORTOU ALGUM DEFEITO? |
| MEDIDAS |
| 1 - QUAIS DADOS RESPONDEM AS MINHAS PERGUNTAS? |

Fonte: Autores

Para analisar o tempo de resposta do sistema foram utilizados os dados produzidos no Iridium log (Figura 6) juntamente com a controladora RMC3, que foi a responsável por permitir a emulação de todo sistema. Para emular o sistema foi preciso compilar a programação feita no *software* SIMPL Windows, passar esse arquivo para a controladora e testar a programação através da ferramenta SIMPL Debugger (Figura 7) que permite alterar os valores das variáveis sem a necessidade dos equipamentos instalados fisicamente.

Foram organizados os valores de tempo na Tabela 2 e calculada a média aritmética simples que resultou em um valor de 463 ms para atualização do dado na nuvem.

Figura 6. Iridium Log



Fonte: Autores

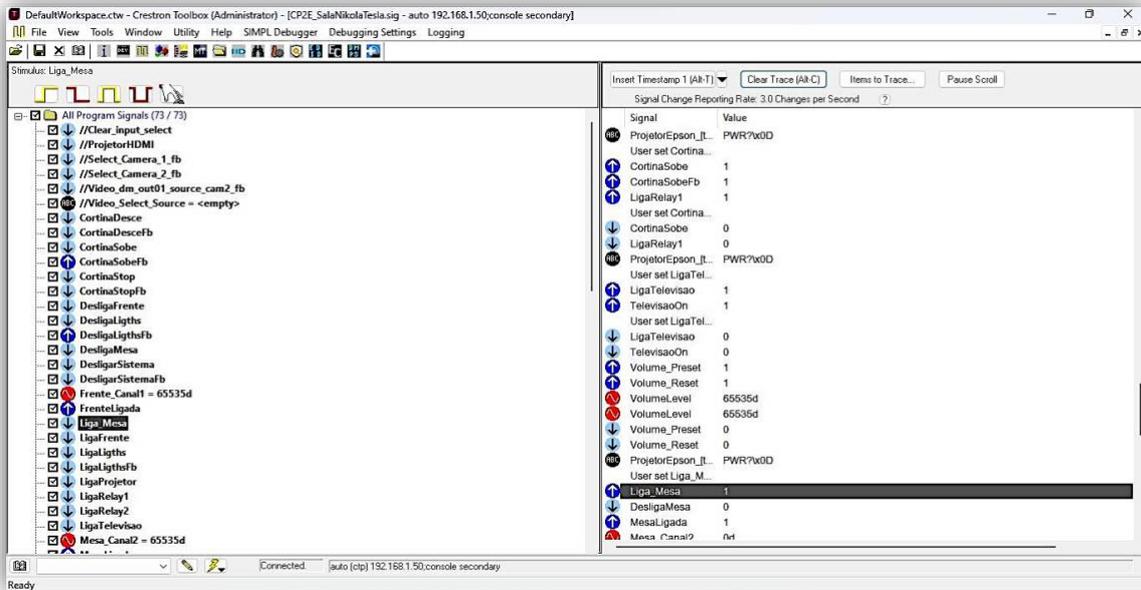


Tabela 2. Tempo médio

| Variável | Status | Tempo inicial | Confirmação do dado na nuvem | Tempo (s) |
|--------------------|--------|---------------|------------------------------|--------------|
| Frenteligada | false | 15:44:55.690 | 15:44:56.169 | 0,479 |
| | true | 15:44:57.913 | 15:44:58.412 | 0,499 |
| | false | 15:45:00.421 | 15:45:00.833 | 0,412 |
| | true | 15:45:02.673 | 15:45:03.152 | 0,479 |
| | false | 15:45:05.324 | 15:45:05.834 | 0,510 |
| | true | 15:45:07.769 | 15:45:08.214 | 0,445 |
| | false | 15:45:10.109 | 15:45:10.547 | 0,438 |
| | true | 15:45:12.479 | 15:45:12.920 | 0,441 |
| Tempo Médio | | | | 0,463 |

Fonte: Autores

Figura 7. SIMPL Debugger

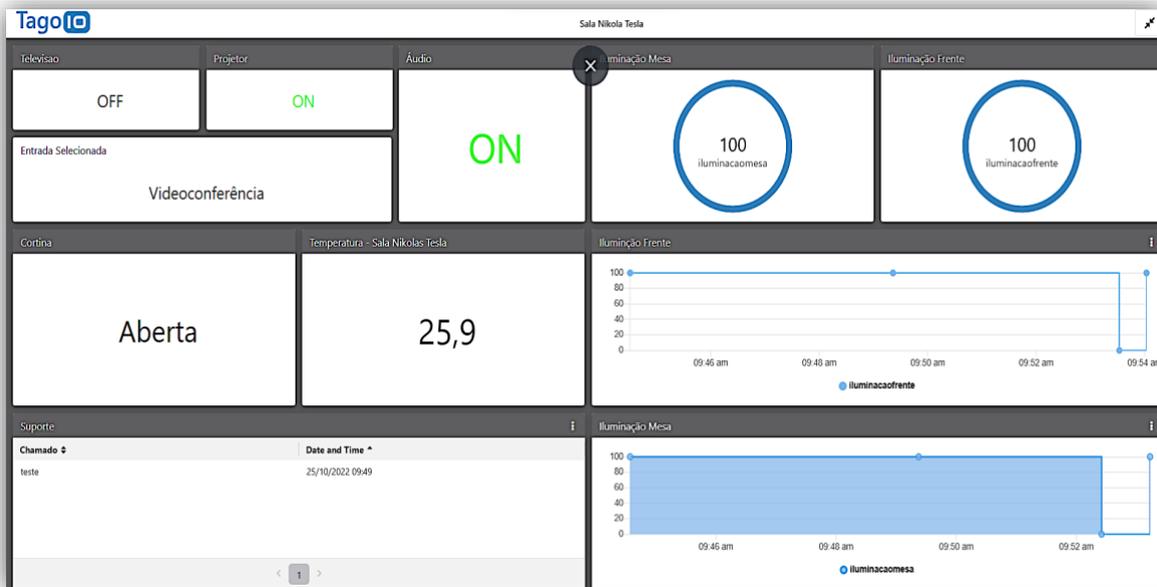


Fonte: Autores

Dessa forma, foi criado o *dashboard* da Figura 8, que apresenta o estado das variáveis em tempo real baseado no método *push*, que mostra na tela as informações de alguns equipamentos. Foram utilizados *widgets*, que são os elementos de interação, tal como um *display*, para mostrar o *status* da televisão, projetor, áudio, temperatura e cortina, que permite apenas visualizar o valor das variáveis que foram atribuídas a ele. Para iluminação foi utilizado o *widget* do tipo *dial*, como um botão giratório, que permite visualizar o percentual da iluminação, bem como ver a data e hora que ela foi ligada e/ou desligada no *widget* de área, através de um gráfico de área, onde o eixo Y refere-se o valor da variável e o eixo X refere-se ao tempo. Os dados podem ser exportados para um arquivo CSV ou XLSX. Também foi utilizado um *widget* do tipo tabela dinâmica, onde uma variável recebe um texto digitado pelo usuário, reportando um defeito encontrado na sala, e o *widget* associado dispara um e-mail para o suporte da empresa com os dados que o usuário digitou, abrindo um chamado para a equipe que fará a manutenção.



Figura 8. Dashboard de monitoramento final



Fonte: Autores

5. CONCLUSÃO

O presente trabalho se pautou na criação de um *dashboard*, capaz de mostrar de forma clara e precisa o uso dos equipamentos em uma sala de reunião, a fim de melhorar o processo de manutenção, abertura de chamados e experiência do usuário. O sistema desenvolvido e testado, de forma emulada, se mostrou eficaz quanto ao tempo de atualização (menor que 0,5s) dos dados na nuvem, na medida que o sistema detecta que uma variável mudou de estado. Foi possível criar um *layout* onde o cliente final consegue ter ideia do que está ocorrendo na sala de reunião, como: qual recurso visual está em utilização; se a iluminação está adequada; se a reunião é interna ou se existem participantes externos; se o áudio está funcionando; se a temperatura está de acordo com o programado; e se o usuário encontrou algum problema em sua utilização. A forma simples de integração com a nuvem, por meio do protocolo HTTP, motiva a integração, em trabalhos futuros, com as plataformas de gerenciamento de sistemas de videoconferência, como: Microsoft Teams; Logitech; Polycom; Zoom; entre outras – que já oferecem *dashboard* automáticos para seus sistemas. Outro ponto possível de ser explorado é o monitoramento de dispositivos de sinalização digital (*digital signage*), que oferecem *dashboard* para monitoramento remoto com Interface de Programação de Aplicação (Application Programming Interface, API) aberta para integração com plataformas de terceiros, permitindo concentrar os dados de todos os dispositivos em uma única plataforma.

REFERÊNCIAS

- Andrade, C. (1996). Novos Conceitos de Ocupação de Espaços de Escritórios Territoriais ou Não-territoriais. *Office*. (22)
- Janes A., Sillitti, A. & Succi G. (2016). *Effective Dashboard Design*
- Ashton, K. (2009). That 'internet of things' thing. *RFID journal*, 22(7), 97-114.
- Basili, V. R., Caldeira, Gianluigi., Rombach, H.D. (1994). *The goal question metric approach*. Recuperado de <https://www.semanticscholar.org/paper/The-Goal-Question-Metric-Approach-Basili-Caldiera/02e65151786574852007ecd007ee270c50470af0>



Iridium Mobile. <https://iridi.com> Acessado em: 8 abril 2022

Janes, A. & Succi, G. (2009), To Pull or Not to Pull. *Conference Paper*. <https://doi.org/10.1145/1639950.1640052>

Jurizato, L. A., & Pereira, P. S. R. (2003). Sistemas supervisorios. *Nova odessa, network Technologies*, 1, 2.

Lautier, François (1999). La gestion de la ressource spatiale. *Laboratoire Espaces Travail (LET)*. 1-25. Recuperado de <https://let.archi.fr/IMG/pdf/pub-let-article-ressource-spatiale.pdf>

Lopes, Y. M. & Moori, R. G. (2021). O papel da IoT na relação entre gestão estratégica da logística e desempenho operacional. *Revista de Administração Mackenzie*, 22(3), 1-27. <https://doi:10.1590/1678-6971/eRAMR210032>

Malik, S. (2005). *Enterprise dashboards: design and best practices for IT*. John Wiley & Sons.

Moço, L. P. L, (2015). *Análise do Desempenho de invocações à web services consumidos por um aplicativo móvel*. (Trabalho de Diplomação).

Universidade Tecnológica Federal do Paraná - UTFPR, Medianeira, PR, Brasil.

Okoh, C., Roy, R., & Mehnen, J. (2017). *Maintenance Informatics Dashboard Design for Through-Life Engineering Services*. Elsevier, 166-171

Paiola, C. E. G. (2018). O papel do sistema supervisorio no atual contexto tecnológico. *Revista InTech* 132, 6-18. Recuperado de https://www.aquarius.com.br/wp-content/uploads/2018/05/InTech132_artigo.pdf

Pleasant, B. (2023). *The New Crestron: Enterprise and UC Platforms for the Modern Workplace*. Crestron. Recuperado de

https://www.crestron.com/getmedia/c6af4f29-fc5b-4c98-b191-33208780f142/wm_evolution_of_crestron_whitepaper

Sasaki, Y. & Yokotani, T. (2019). Performance evaluation of MQTT as a communication protocol for IoT and prototyping. *Advances in Technology Innovation*, 4(1), 21-29.

