

## 15º Congresso de Inovação, Ciência e Tecnologia do IFSP - 2024

### SISTEMA WEB PARA VISUALIZAÇÃO DE IMAGENS TRANSMITIDAS POR MICROCONTROLADORES VIA GPRS

GUILHERME C. DA SILVA<sup>1</sup>, RODRIGO P. PANTONI<sup>2</sup>

<sup>1</sup> Graduando em Engenharia Elétrica, Campus Sertãozinho, carmona.silva@aluno.ifsp.edu.br

<sup>2</sup> Docente e Orientador do Projeto PIVICT, IFSP, Campus Sertãozinho, rpantoni@ifsp.edu.br

Área de conhecimento (Tabela CNPq): 1.03.04.04-5 Teleinformática.

**RESUMO:** Com os avanços tecnológicos das últimas décadas, o desenvolvimento de aplicações que integrem grandes sistemas e redes de dispositivos se torna cada vez mais essencial para acompanhar a crescente demanda por conectividade e automação. Nesse contexto, surge a definição de Internet das Coisas (IoT), que envolve a interconexão de dispositivos inteligentes capazes de coletar, transmitir e processar dados em tempo real, formando uma vasta rede de comunicação. Este trabalho tem como principal objetivo explorar e implementar essa tecnologia por meio do desenvolvimento de um aplicativo capaz de apresentar, de maneira eficiente e intuitiva, imagens armazenadas em um banco de dados centralizado, as quais são transmitidas a partir de um conjunto de microcontroladores via tecnologia GPRS. Para alcançar os resultados desejados, utilizou-se a linguagem de programação JavaScript, conhecida por sua versatilidade e capacidade de criar aplicações web interativas. Através dessa abordagem, foi possível obter resultados satisfatórios em bancada, que demonstram a abrangência e a aplicabilidade do conceito abordado neste estudo.

**PALAVRAS-CHAVE:** banco de dados na nuvem; sistema web; imagem; internet das coisas.

### WEB SYSTEM FOR VISUALIZATION OF IMAGES TRANSMITTED BY MICROCONTROLLERS VIA GPRS

**ABSTRACT:** With the technological advances of recent decades, the development of applications that integrate large systems and networks of devices becomes increasingly essential to keep up with the growing demand for connectivity and automation. In this context, the definition of the Internet of Things (IoT) emerges, involving the interconnection of intelligent devices capable of collecting, transmitting, and processing data in real-time, forming a vast communication network. This work aims to explore and implement this technology through the development of an application capable of efficiently and intuitively presenting images stored in a centralized database, which are transmitted from a set of microcontrollers via GPRS technology. To achieve the desired results, JavaScript was used as the programming language, known for its versatility and ability to create interactive web applications. Through this approach, it was possible to obtain satisfactory results in bench that demonstrate the scope and applicability of the concept addressed in this study.

**KEYWORDS:** cloud database; web system; image; internet of things.

## INTRODUÇÃO

Advindo das constantes evoluções tecnológicas nas últimas décadas, o conceito de Internet das Coisas (IoT) passou a ser amplamente discutido em nossa sociedade, sendo difundidas diversas perspectivas sobre o assunto. Desta forma, um dos pontos de vista a respeito do conceito coloca que IoT é um sistema de dispositivos físicos conectados à internet com a finalidade de coletar, analisar e compartilhar dados entre si e assim, prover uma melhora no estilo de vida humana (Masoodi *et al.*, 2024).

O desenvolvimento de uma aplicação básica usando IoT demanda um sistema de hardware que contenha conexão com a internet para o envio de informações. Geralmente são utilizados microcontroladores, sensores e algum protocolo de acesso à internet (Nivetha *et al.*, 2023). O cenário deste trabalho contempla aplicações em que não há acesso à internet por Wifi ou rede cabeada e, por isso, os dispositivos microcontroladores contam com um módulo GPRS, e assim enviar as imagens para um banco de dados na nuvem. Assim, este trabalho permite o acesso às imagens transmitidas pelos microcontroladores.

Neste contexto, o objetivo geral é desenvolver um sistema web capaz de acessar as imagens transmitidas por dispositivos que se conectam à internet via GPRS, de forma a organizar as imagens por dispositivos. Os objetivos específicos são: desenvolvimento do sistema (*front-end* e *back-end*) para visualização de imagens que seja hospedado num servidor web (1); desenvolvimento do banco de dados na nuvem para armazenamento e consulta das imagens (2); integração do método de envio de imagens via GPRS, desenvolvido por Costa e Pantoni (2023), utilizando transmissão de dados por HTTP em ambientes com conectividade limitada (3).

## MATERIAL E MÉTODOS

Para desenvolvimento deste trabalho, fora necessária a escolha de uma linguagem de programação que fosse versátil e tivesse uma integração com a web de forma simples, facilitando o trabalho tanto no *front-end*, responsável parte gráfica, unindo usuário e serviços de forma agradável e satisfatória, quanto no *back-end*, parte responsável por processar dados e executar as tarefas solicitadas pelo usuário (SGP-e Softplan, 2021). Tendo isso em vista optamos pela linguagem JavaScript.

Em relação ao objetivo específico 1, consistiu-se em criar classes e instanciar objetos referentes a cada microcontrolador e as imagens que cada microcontrolador enviará para o servidor. Outrossim, para traçar as rotas de cada solicitação, sejam elas HTTP *POST* ou *GET*, por exemplo, foi utilizada a biblioteca ElysiaJS (2024), que por sua vez direciona cada solicitação para seu respectivo serviço dentro da aplicação, seja ele a criação de um novo microcontrolador dentro do servidor ou o envio de uma imagem por um microcontrolador já existente dentro do sistema de unidades de armazenamento, que, no caso deste estudo, foi optado pelo uso de *buckets* dentro de um servidor web. *Buckets* são um sistema de armazenamento que não pode ser aninhado e servem para, além de armazenar, controlar dados dentro de um sistema ou serviço (Google Cloud, 2024).

Tendo em vista a segurança da informação e a conclusão de uma primeira etapa do objetivo específico 1, a comunicação entre o *back-end* e o *front-end*, se dá por meio de requisições do front para o back, que por sua vez remete ao banco de dados, tendo como retorno uma url que dá acesso a imagem no *bucket* dentro do servidor.

Em relação ao objetivo 2, o Sistema Gerenciador de Banco de Dados (SGBD) escolhido foi o PostgreSQL, que é gratuito e hospedado na nuvem, conectado a uma conta do Google. O serviço de desenvolvimento em nuvem adotado foi o Render, que possui um plano gratuito que supre as necessidades do projeto. Fora usada o Prisma, que é uma ORM (*Object-Relational Mapping*), ferramenta que facilita o gerenciamento entre objetos e tabelas dentro do banco de dados (Lorenz *et al.*, 2017), além de promover o uso de comandos em *Typescript* que facilitam o uso de SQL na aplicação.

O modelo de banco de dados utilizado segue o formato apresentado na Figura 1, onde há o relacionamento de um para muitos (1-N), ou seja, um microcontrolador terá muitas imagens. Cada microcontrolador será armazenado no banco de dados suas informações quanto ao número de série, ou seja, um nome único para diferenciá-lo dos demais, uma localização, que será enviada quando fizer a requisição utilizando o método POST para armazenamento da imagem no banco, e uma lista com todas as imagens enviadas por ele. Quanto a tabela relacionada às imagens, cada imagem terá armazenado no banco de dados um nome, uma url, que vem como resposta a uma requisição do

*front-end* para acessar as imagens relacionadas a determinado microcontrolador, e uma data e hora, que são a data e o horário em que a imagem for publicada no servidor.

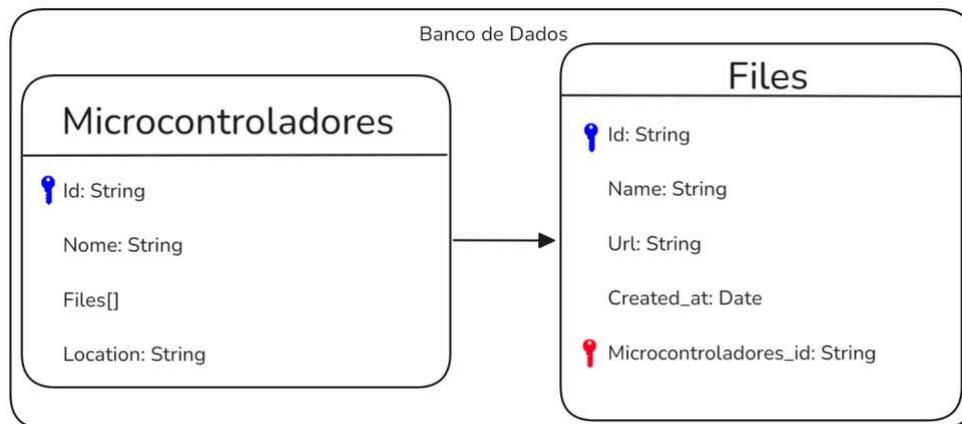


FIGURA 1. Modelo do Banco de Dados

O modelo de interação do sistema segue de acordo com a Figura 2. Cada microcontrolador da rede mencionada anteriormente faz uma requisição para a API (Interface de Programação de Aplicação) desenvolvida neste trabalho. A API se comunica com o banco de dados e insere as imagens em *buckets* dentro do servidor. Outra via é a parte do *front-end*. O usuário faz a requisição para o servidor, que serve a lista com todos os microcontroladores existentes no sistema e quando o usuário seleciona o microcontrolador desejado, lhe é apresentada todas as imagens referentes a esse microcontrolador utilizando de uma resposta do *back-end* e assim para o navegador.

FIGURA 2. Interação do sistema



Outrossim, em vista de satisfazer o objetivo específico 3, cada microcontrolador faz uma requisição do tipo HTTP *POST* para o servidor, enviando a imagem por meio de um *buffer* convertido para o formato base64, que é um método utilizado para transferência de dados MIME (*Multipurpose Internet Mail Extensions*). O servidor então recebe o *buffer* e converte a imagem para o formato original e, utilizando o ElysiaJS, a imagem é armazenada no banco de dados. Mais detalhes sobre a implementação realizada no microcontrolador é encontrada em Costa e Pantoni (2023).

## RESULTADOS E DISCUSSÃO

Os resultados obtidos podem ser observados através das imagens 3 e 4, onde é apresentado o *front-end* da aplicação. Na Figura 3, é apresentada uma lista com os microcontroladores registrados no sistema juntamente com seus nomes. Essa lista de dispositivos pode aumentar conforme forem instalados outros microcontroladores no sistema, sendo que cada microcontrolador realiza o acesso ao banco de dados com uma chave de acesso única para o servidor e, assim, é registrado no sistema como um novo microcontrolador. O comissionamento de um microcontrolador no sistema ocorre de forma automática quando envia uma mensagem específica via HTTP *POST* com as coordenadas geodésicas de sua localização. A partir deste momento, o sistema aloca as informações no banco de dados e fica à espera do envio de imagens do campo. Embora na Figura 3 os microcontroladores estejam com um

nome padrão, como “Microcontrolador 1”, “Microcontrolador 2” etc., é função do sistema fornecer a capacidade de renomeá-los, de forma a trazer mais facilidade para consultas do usuário.

### Visualizador de Imagens



FIGURA 3. Tela do Sistema Web com a lista de microcontroladores detectados

Na Figura 4, é mostrado um exemplo de imagem enviada por meio de um dos microcontroladores registrados no banco de dados. Assim, quando o usuário seleciona alguns dos microcontroladores mostrados na Figura 3, é aberta uma página web com as imagens recebidas do referido dispositivo.



FIGURA 4. Imagem transmitida pelo microcontrolador é mostrada no sistema

Assim, os seguintes casos de uso foram validados: o microcontrolador é inserido no sistema assim que é ligado de forma automática (1); o microcontrolador no sistema pode ser renomeado (2); as imagens capturadas são arquivadas no sistema e relacionadas ao microcontrolador apropriado (3); e as consultas para visualização dos microcontroladores e suas imagens (4).

### CONCLUSÕES

Dessa forma, constata-se que os objetivos propostos para o desenvolvimento do trabalho foram atingidos. A implementação de um sistema web para visualização de imagens se deu com sucesso, concluindo com êxito cada etapa dos objetivos específicos e apresentando a viabilidade do uso de sistemas em nuvem utilizando a tecnologia de banco de dados e *buckets* como forma de armazenar informações e transmitir dados de forma simples para usuários quando munidos de um *front-end* intuitivo.

As possíveis limitações relacionadas ao projeto podem ser vistas majoritariamente em forma de sobrecarga no armazenamento do servidor, no entanto, devido ao tamanho minimalista dos arquivos e dados salvos no mesmo, tal limitação se torna quase inexistente tendo em vista os sistemas de armazenamento dos servidores atuais.

Outrossim, possíveis melhorias no sistema podem ser realizadas por meio de um *front-end* mais intuitivo e harmonioso, facilitando a compreensão e navegação dos dados armazenados no banco de dados.

As aplicações do projeto são inúmeras, variando desde coletas de imagens como amostra em locais públicos para detecção de densidade de pessoas em determinados momentos, coleta de amostras para cálculo de vida útil em equipamento em campo, tais como motores elétricos e mecânicos, detecção de animais na natureza, entre outros.

Os testes e resultados foram conduzidos em bancada, o que encoraja os autores a prosseguirem com os testes de campo, de modo a instalar os dispositivos em lugares com conectividade limitada, tais como locais rurais, onde não se tem Wifi, a fim de compor uma aplicação de envio de informações para agricultura automatizada.

### **CONTRIBUIÇÕES DOS AUTORES**

Rodrigo P. Pantoni contribuiu com a concepção e estudo de viabilidade do projeto. Guilherme C. da Silva, por outro lado, desenvolveu o projeto em sua implementação, realização e testes completos. Assim, Rodrigo P. Pantoni e Guilherme C. da Silva contribuíram com a curadoria e análise dos dados e procederam com a metodologia e experimentos. Rodrigo P. Pantoni e Guilherme C. da Silva atuaram na redação do trabalho assim como na revisão do trabalho e aprovaram a versão submetida.

### **AGRADECIMENTOS**

Os autores agradecem ao Campus Sertãozinho do IFSP que está financiando a bolsa de iniciação científica da modalidade PIBIFSP do referido aluno participante do projeto. Também agradecem ao Instituto Federal de Educação, Ciência, Tecnologia de São Paulo que está financiando os recursos deste projeto por meio do Edital N° 329/2021 - "Edital de Fomento para Criação de Centros de Pesquisa e Inovação no IFSP".

### **REFERÊNCIAS**

COSTA, O. L.; PANTONI, R. P. Transmissão de imagem via GPRS para Internet das Coisas. In: CONGRESSO DE INOVAÇÃO, CIÊNCIA E TECNOLOGIA DO IFSP, 14., 2024, São Paulo. **Anais [...]**. São Paulo: IFSP, 2024.

ELYSIAJS. **Fast and Friendly Bun Web Framework**. Disponível em: <<https://elysiajs.dev>>. Acesso em: 27 ago. 2024.

GOOGLE CLOUD. 2024. Sobre os buckets do Cloud Storage. Disponível em: <<https://cloud.google.com/storage/docs/buckets?hl=pt-br>>. Acesso em: 19 ago. 2024.

LORENZ et al. Object-Relational Mapping Reconsidered. A Quantitative Study on the Impact of Database Technology on O/R Mapping Strategies. **CORE**, 2017. Disponível em: <<http://hdl.handle.net/10125/41754>>. Acesso em: 22 ago. 2024.

MASOODI et al. **Internet of Things Vulnerabilities and Recovery Strategies**. Auerbach Publications, 2024.

NIVETHA et al. 2023. IoT Application and Architecture. **International Journal of Scientific Research in Engineering and Management (IJSREM)**. Disponível em: <<https://ijsrem.com/download/iot-application-and-architecture>>. Acesso em: 22 set. 2024.

SGP-e Softplan. Front-end e back-end: o que são e quais as diferenças. 2021. Disponível em: <<https://sgpe.sea.sc.gov.br/capdoc/front-end-e-back-end-o-que-sao-e-quais-as/>>. Acesso em: 19 ago. 2024.