

# Desenvolvimento de uma Arquitetura IoT para Rastreamento e Adaptação de Recursos em Instituições de Ensino

Emanuel de F. Vieira<sup>1</sup>, Tiago A. Rizzetti<sup>1</sup>, Renato P. de Azevedo<sup>1</sup>

<sup>1</sup>Colégio Técnico Industrial de Santa Maria  
Universidade Federal de Santa Maria (UFSM)  
Av. Roraima nº 1000. Cidade Universitária. Camobi. Santa Maria - RS

emanuel.franceschi@acad.ufsm.br, tiago.rizzetti@ufsm.br,  
renato@redes.ufsm.br

**Resumo.** *Este trabalho apresenta o desenvolvimento de um sistema inteligente para rastreamento de presença e gerenciamento de recursos em ambientes de Educação Profissional e Tecnológica (EPT). A solução visa otimizar o controle de presença e a utilização dos espaços institucionais por meio de uma arquitetura modular que utiliza tecnologias IoT de baixo custo, como micro-controladores ESP32, Bluetooth Low Energy (BLE) e a plataforma Home Assistant. O núcleo do sistema é composto por um middleware, responsável por processar, persistir e disponibilizar os dados de localização de forma segura através de uma API RESTful. Testes parciais validaram a viabilidade da arquitetura a integração dos componentes, demonstrando o potencial da solução para aplicação prática em instituições de ensino. Com o objetivo de validar a solução em ambiente real, o projeto segue em desenvolvimento.*

## 1. Introdução

A gestão eficiente de ambientes educacionais é um desafio constante, especialmente em instituições caracterizadas pelo fluxo intenso de alunos e pela necessidade de otimizar o uso de recursos. A frequência em sala de aula é considerada um fator crítico para o desempenho acadêmico, levando muitas instituições a exigirem porcentagens mínimas de presença. Contudo, o controle manual de frequência, além de consumir tempo valioso dos docentes, é suscetível a falhas e não gera dados estruturados para uma análise mais aprofundada [Puckdeevongs et al. 2020].

A motivação para esta pesquisa origina-se dos desafios de gestão em ambientes de ensino, que são caracterizados pelo fluxo intenso de pessoas, pelo compartilhamento de salas e pela necessidade de otimizar o uso de recursos [Rao 2022]. Processos manuais, como o controle de frequência de estudantes ou o rastreamento de equipamentos, são ineficientes, suscetíveis a falhas e não geram dados estruturados para análises aprofundadas. O problema central é a carência de ferramentas flexíveis e acessíveis para instituições públicas, visto que soluções comerciais frequentemente não se adequam à realidade das instituições públicas de ensino, seja pelo custo elevado, pela infraestrutura necessária ou pela dificuldade de adaptação às especificidades de cada organização [Soares et al. 2020].

Nesse cenário, a abordagem inovadora deste trabalho é propor uma arquitetura de baixo custo baseada no conceito de Internet das Coisas (IoT), cujo diferencial é o desenvolvimento de um *middleware* próprio. Ao invés de focar em uma única aplicação, o projeto desenvolve uma camada de *software* intermediária que consome dados de localização

de uma rede de sensores ESP32 integrados à plataforma de automação residencial *Home Assistant*, tratando-os, persistindo-os em banco de dados e expondo-os de forma segura e padronizada através de uma API RESTful. Essa arquitetura desacopla a camada física da camada de aplicação, permitindo que diversos sistemas externos consumam os dados de localização para finalidades distintas, como gestão de presença, controle de inventário de ativos ou implementação de alertas de segurança.

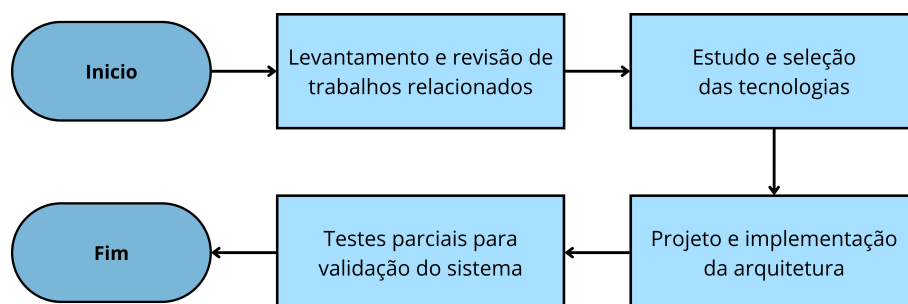
A partir deste contexto, a questão de pesquisa que norteia o estudo é: "Como projetar e desenvolver uma arquitetura modular, escalável, e de baixo custo que integre dados de localização em tempo real e os disponibilize de forma segura e flexível para múltiplas aplicações de gestão?" Para respondê-la, foi adotada uma metodologia estruturada em quatro etapas principais, detalhadas na Seção 2.

Como resultados parciais, a comunicação entre todos os componentes da arquitetura já foi validada, desde a detecção dos *beacons* até a persistência dos eventos de localização no banco de dados. As principais contribuições esperadas para a área de Informática na Educação são: (I) a disponibilização de uma arquitetura de referência, documentada e de código aberto, a fim de promover a replicação do sistema em outras instituições de ensino; (II) a criação de uma plataforma que disponibilize dados e sirva de base para o desenvolvimento de outras aplicações; (III) e por fim, o próprio sistema servir como um objeto de estudo e recurso didático para cursos técnicos e superiores nas áreas de Redes, IoT e Desenvolvimento de *Software*.

O restante do artigo está estruturado da seguinte forma: A Seção 2 detalha a metodologia empregada na realização deste estudo. Já a Seção 3 aborda os trabalhos relacionados. Em seguida, a Seção 4 apresenta a arquitetura proposta. Por fim, na Seção 5 são descritos os resultados parciais obtidos durante o desenvolvimento do sistema e os trabalhos futuros.

## 2. Materiais e Métodos

A metodologia adotada neste estudo emprega uma abordagem qualitativa e quantitativa, que combina um levantamento bibliográfico aprofundado com o desenvolvimento e a validação de um protótipo funcional. O processo foi estruturado em quatro etapas, conforme ilustrado na Figura 1.



**Figura 1. Fluxograma da metodologia empregada.**

A primeira etapa deste trabalho consistiu em um levantamento e revisão de trabalhos relacionados para fundamentar a pesquisa e identificar as tecnologias e métodos mais relevantes na área de localização *indoor*. Foi realizada uma pesquisa no ResearchGate,

utilizando a seguinte *string* de busca: ("Indoor Navigation" OR "Indoor Localization" OR "Indoor Positioning") AND ("Bluetooth Low Energy" OR "BLE"). A busca retornou um grande volume de artigos, permitindo selecionar os trabalhos que representam diferentes abordagens do problema e que mais se alinham ao contexto deste projeto, servindo como base para uma análise comparativa.

A segunda etapa foi dedicada ao estudo e à seleção das tecnologias para a arquitetura do sistema. Com base nos conhecimentos adquiridos na revisão bibliográfica e nos requisitos do projeto, optou-se pela utilização da plataforma *Home Assistant* como o ecossistema central, devido à sua flexibilidade, natureza de código aberto e ampla capacidade de integração com diversas tecnologias. Para os *beacons*, o microcontrolador ESP32 foi escolhido por ser reconhecido pelo baixo custo e conectividade *Bluetooth* nativa. Os dispositivos localizadores utilizam o *firmware* ESPresense, uma solução de código aberto especializada na detecção de presença por ambiente. Esse *firmware*, instalado em microcontroladores ESP32, é responsável por monitorar o sinal dos *beacons* e reportar os dados de proximidade (RSSI) ao *Home Assistant* via protocolo MQTT (*Message Queuing Telemetry Transport*), o que viabiliza a lógica de localização de forma eficiente e automatizada.

A terceira etapa concentrou-se no projeto e na implementação da arquitetura de comunicação e integração. Nessa fase, os componentes selecionados foram configurados e programados para operar de forma integrada. A Seção 4 apresenta a arquitetura resultante, detalhando o fluxo de dados e a interação entre os elementos.

Por fim, a quarta etapa consiste na realização de testes parciais para validação do sistema. Para isso, foi montado um cenário de teste em um ambiente real, composto por três salas de aula adjacentes, cada uma com um dispositivo localizador instalado no centro. O experimento consistiu no deslocamento controlado de dois *beacons*, cada um portado por um indivíduo. O objetivo principal foi avaliar a precisão do sistema na identificação da localização de cada *beacon*. Para isso, foram comparados os eventos persistidos no banco de dados do *middleware*, na tabela *LocationEvents*, com um roteiro de movimentação pré-definido, ilustrado na Figura 2, indo de *n1* para *n2* e assim por diante, a fim de representar seus deslocamentos.

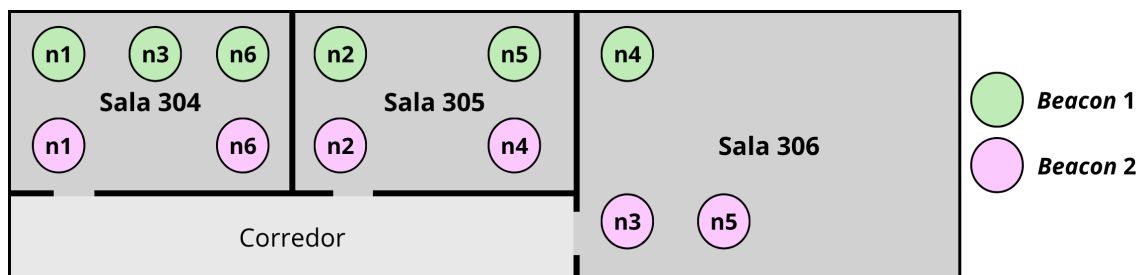


Figura 2. Roteiro de movimentação dos *beacons*.

### 3. Trabalhos Relacionados

O trabalho de [Dravya et al. 2020] propõe um sistema de navegação interna de baixo custo, utilizando microcontroladores ESP32 como receptores e o celular do usuário como emissor de sinal *Bluetooth Low Energy* (BLE). A localização é calculada convertendo

valores de RSSI em distância e aplicando o algoritmo de trilateração. O sistema, que alcançou uma acurácia média de 1,5 metros, se destaca por eliminar a necessidade de *beacons* dedicados.

[Chen et al. 2024] focam no rastreamento de ativos com precisão a nível de sala, empregando etiquetas BLE com colheita de energia. A inovação do estudo está no uso da teoria da evidência de *Dempster-Shafer* em áreas de sinal claro e de *fingerprinting* com algoritmos de aprendizado de máquina em zonas de sobreposição de sinal. Os experimentos demonstraram uma acurácia superior a 99% na identificação correta da sala.

Já [Puckdeevongs et al. 2020] desenvolveram um sistema para automatizar o controle de frequência em salas de aula. A solução utiliza o sinal BLE dos celulares dos alunos e técnicas de *fingerprinting* com redes neurais artificiais para estimar a posição. O sistema obteve um erro de posicionamento inferior a 1 metro em mais de 72% dos testes e integra os dados a uma aplicação web para gestão acadêmica.

Enquanto os trabalhos citados se concentram em técnicas para determinar a posição com precisão, este projeto se diferencia ao desenvolver uma camada de *software* intermediária. A solução opera sobre dados já processados pela plataforma *Home Assistant*, oferecendo uma API REST para gerenciamento e integração das informações de localização. O foco não é a precisão métrica, mas a identificação confiável do ambiente onde o usuário se encontra, permitindo que diversas outras aplicações consumam esses dados para fins como controle de acesso e geração de relatórios.

#### 4. Arquitetura Proposta

Esta seção descreve a arquitetura de comunicação proposta para o sistema. Os elementos do sistema foram organizados de forma modular, conforme ilustrado na Figura 3, a fim de garantir eficiência, escalabilidade e facilidade de manutenção.

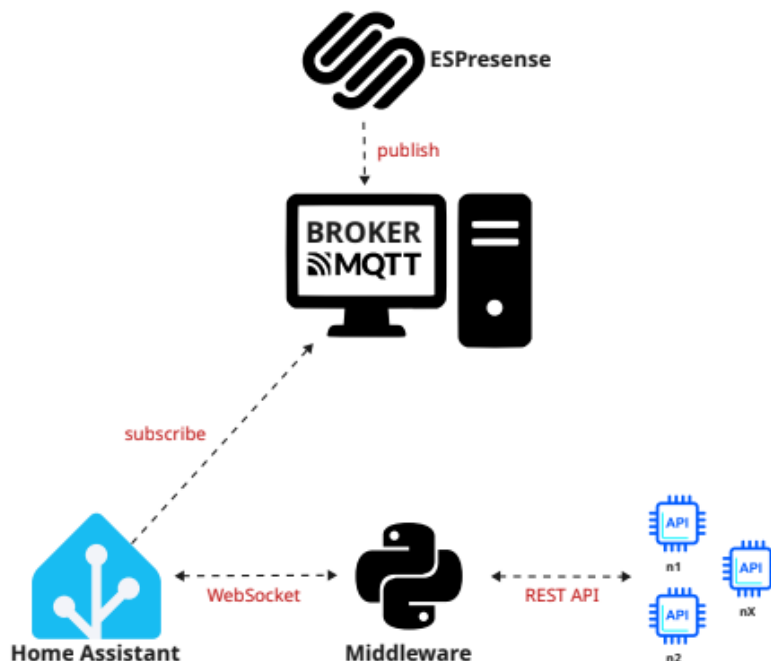
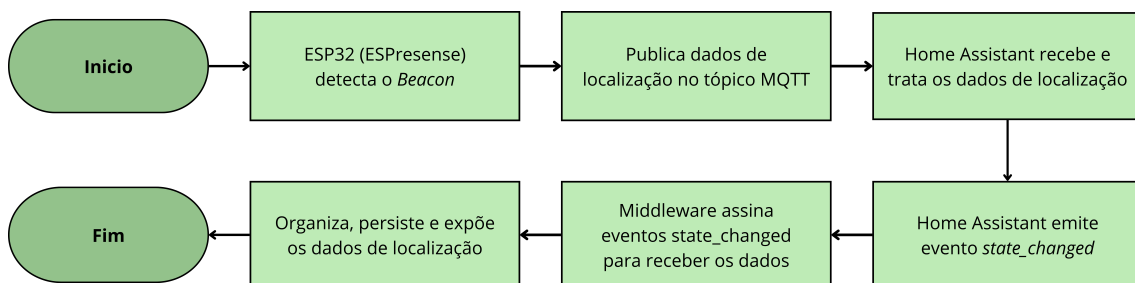


Figura 3. Arquitetura do Sistema.

A coleta de dados é realizada por dispositivos ESP32 com o *firmware* ESPresense, que atuam como localizadores. Instalados nos ambientes monitorados, esses dispositivos detectam a presença de *beacons* BLE e publicam os dados de localização, como os sinais RSSI, em tópicos de um *broker* MQTT. O *broker* funciona como um intermediário de mensagens, desacoplando os dispositivos localizadores dos consumidores, o que viabiliza a comunicação assíncrona entre os elementos do sistema.

O componente central da arquitetura é o *Home Assistant*, que se inscreve nos tópicos MQTT para receber os dados publicados pelos localizadores. Ele processa essas informações por meio do componente *mqtt\_room*, interpretando os sinais para atualizar o estado de presença das entidades monitoradas. Sempre que o estado de uma entidade é alterado, o *Home Assistant* dispara um evento do tipo *state\_changed*, disponibilizando essa informação em tempo real por meio de sua API.

Para o consumo externo e integração com outros sistemas, foi desenvolvido um *middleware* em Python. Ele estabelece uma conexão persistente via *WebSocket* com a API do *Home Assistant*, recebendo os eventos *state\_changed* assim que são emitidos. O *middleware*, por sua vez, expõe uma API REST, permitindo que sistemas de terceiros consultem as informações de presença de forma estruturada, seja para gerar relatórios ou para outras finalidades. Essa abordagem garante o desacoplamento entre os módulos e a segurança no acesso aos dados, que será reforçada pela implementação de um Controle de Acesso Baseado em Papéis (RBAC). Esse mecanismo permite definir permissões granulares para diferentes papéis de usuários no consumo das informações via API. O fluxograma da Figura 4 ilustra essa sequência de eventos, desde a detecção do *beacon* até o encaminhamento para o *middleware*.



**Figura 4. Fluxograma do processo de detecção de *beacons*.**

A escolha de utilizar o *Home Assistant* como intermediário foi fundamentada em critérios de modularidade e reusabilidade. Embora a conexão direta do *middleware* com o *broker* MQTT fosse viável, ela implicaria a necessidade de implementar manualmente toda a lógica de tratamento dos dados do ESPresense. Ao delegar essa tarefa ao *Home Assistant*, que possui integração nativa, a complexidade do *middleware* é reduzida, a consistência dos dados é aprimorada e o sistema se torna mais modular e flexível, permitindo que seja facilmente reutilizado em diferentes cenários.

## 5. Resultados Parciais e Trabalhos Futuros

O projeto, em desenvolvimento há 16 meses, encontra-se na fase de validação da arquitetura e integração dos seus componentes. Os testes realizados até o momento, em um cenário controlado composto por três salas de aula, confirmaram a viabilidade da

comunicação, desde a detecção de *beacons* até a persistência e exposição dos dados de localização pelo *middleware*. A seguir, são apresentados os resultados obtidos durante a realização do teste de movimentação de *beacons*.

A análise dos resultados confirmou que o sistema identificou e registrou com precisão a movimentação dos *beacons* entre os diferentes ambientes, demonstrando a correta integração e o fluxo de dados desde a detecção pelos dispositivos localizadores até o armazenamento final. A Figura 5 ilustra os registros na tabela *Location\_events* do banco de dados durante os testes.

```
mysql> SELECT * FROM LocationEvents WHERE Beacon_id = 1;
```

id	beacon_id	room_id	event_type	event_timestamp
1513	1	4	enter	2025-08-07 17:36:20
1515	1	4	leave	2025-08-07 17:36:52
1516	1	5	enter	2025-08-07 17:36:52
1519	1	5	leave	2025-08-07 17:37:35
1520	1	4	enter	2025-08-07 17:37:35
1523	1	4	leave	2025-08-07 17:38:06
1524	1	6	enter	2025-08-07 17:38:06
1527	1	6	leave	2025-08-07 17:39:46
1528	1	5	enter	2025-08-07 17:39:46
1532	1	5	leave	2025-08-07 17:40:26
1533	1	4	enter	2025-08-07 17:40:26
1535	1	4	leave	2025-08-07 17:43:27

```
mysql> SELECT * FROM LocationEvents WHERE Beacon_id = 2;
```

id	beacon_id	room_id	event_type	event_timestamp
1514	2	4	enter	2025-08-07 17:36:29
1517	2	4	leave	2025-08-07 17:36:56
1518	2	5	enter	2025-08-07 17:36:56
1521	2	5	leave	2025-08-07 17:37:56
1522	2	6	enter	2025-08-07 17:37:56
1525	2	6	leave	2025-08-07 17:39:01
1526	2	5	enter	2025-08-07 17:39:01
1531	2	5	leave	2025-08-07 17:39:51
1529	2	6	enter	2025-08-07 17:39:51
1530	2	6	leave	2025-08-07 17:40:06
1531	2	4	enter	2025-08-07 17:40:06
1534	2	4	leave	2025-08-07 17:43:27

**Figura 5. Eventos de localização registrados na tabela *LocationEvents*.**

Com previsão de conclusão para os próximos 7 meses, os trabalhos futuros concentram-se em duas frentes principais: (I) a expansão do ambiente de teste para um número maior de salas e usuários simultâneos, a fim de avaliar a escalabilidade da solução; e (II) o desenvolvimento de uma aplicação web de exemplo para consumir a API RESTful, gerando relatórios automatizados. Adicionalmente, será finalizada a implementação do Controle de Acesso Baseado em Papéis (RBAC) para o consumo dos dados. Ao final do projeto, espera-se consolidar a arquitetura, promovendo sua replicação e o desenvolvimento de novas aplicações.

## **Agradecimentos**

O presente trabalho foi realizado com apoio da Fundação Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - CAPES.

## **Referências**

- Chen, Y., Wang, Y., and Zhao, Y. (2024). A room-level indoor localization using an energy-harvesting ble tag. *Electronics*, 13(22).
- Dravya, Holla, U., and K N, P. (2020). Indoor navigation system using ble and esp32.
- Puckdeevongs, A., Tripathi, N. K., Witayangkurn, A., and Saengudomlert, P. (2020). Classroom attendance systems based on bluetooth low energy indoor positioning technology for smart campus. *Information*, 11(6).
- Rao, A. (2022). Attenface: A real time attendance system using face recognition.
- Soares, D. J. M., Soares, T. E. A., and dos Santos, W. (2020). Infraestrutura e desempenho escolar na prova brasil: aspectos e conexões. *Olhar de Professor*, 23:1–18.