

## **SISTEMA DE ESTACIONAMENTO INTELIGENTE: INTEGRAÇÃO DE IoT E PLATAFORMA DE CIDADES INTELIGENTES PARA GESTÃO SUSTENTÁVEL**

D.C. Galdez<sup>1</sup>, e-mail: daniel.cgm@discente.ufma.br. Orientador: L.H.N. Rodrigues<sup>2</sup>, e-mail: luiz.rodrigues@ecp.ufma.br.

<sup>1</sup>Universidade Federal do Maranhão. Centro de Ciência e Tecnologia, Bacharelado Interdisciplinar em Ciência e Tecnologia..

<sup>2</sup>Universidade Federal do Maranhão. Centro de Tecnologia, Departamento do Curso de Engenharia da Computação/CCET.

**RESUMO:** *Este trabalho propõe o desenvolvimento de um sistema de estacionamento inteligente para enfrentar o crescente desafio da gestão de vagas em áreas urbanas. A solução visa otimizar a utilização de espaços disponíveis, reduzir o tempo de busca por vagas e diminuir o tráfego e emissões de poluentes. A abordagem inclui o uso de microcontroladores de baixo custo e protocolos de comunicação leves, integrados a uma arquitetura de microsserviços, o que garante eficiência, flexibilidade e escalabilidade em cidades inteligentes e sustentáveis.*

**PALAVRAS-CHAVE:** microcontroladores; microsserviços; internet das coisas; cidades inteligentes; protocolos de comunicação.

### **1 INTRODUÇÃO**

A otimização do fluxo de veículos tornou-se uma questão crucial nas metrópoles contemporâneas, impulsionada pelo intenso tráfego e pela necessidade de deslocamento em horários específicos do dia, que resultam em congestionamentos e impactam diretamente a mobilidade urbana (TOMASIELLO et al., 2023). Adicionalmente, motoristas que buscam estacionar em grandes áreas urbanas desperdiçam tempo e combustível, o que contribui significativamente para o congestionamento do tráfego e poluição do ar (POLYCARPOU et al., 2013).

A busca por soluções tecnológicas inovadoras, como as cidades inteligentes, é de suma importância, pois visa não apenas aprimorar a experiência dos motoristas e elevar a qualidade de vida urbana, mas também transformar a gestão dos recursos

em organizações públicas e privadas nas grandes cidades. Ao integrar de forma eficiente sistemas de transporte, energia, segurança e infraestrutura, essas cidades promovem uma gestão mais eficiente e sustentável (RAO et al., 2022).

A Internet das Coisas (IoT) é um dos pilares mais importantes das cidades inteligentes. A IoT integra dispositivos físicos, sensores e programas para promover um monitoramento completo, a automação de processos e a tomada de decisões em tempo real. Esses dispositivos facilitam a análise de dados e a resposta imediata a eventos críticos (CNN, 2023).

Para viabilizar uma conectividade eficiente entre dispositivos, é fundamental utilizar protocolos que otimizem as principais características da IoT. Um desses protocolos é o MQTT (Message Queuing Telemetry Transport), projetado especificamente para garantir a troca rápida e confiável de mensagens entre "emissores" (publicadores) e "receptores" (subscritores) (SEREJO, 2018).

Este projeto propõe desenvolver e implementar um sistema de estacionamento inteligente. Para alcançar esse objetivo, tecnologias apropriadas para ambientes urbanos inteligentes serão selecionadas, o que inclui microcontroladores de baixo custo e baixo consumo de energia, protocolos de comunicação leves e uma plataforma de código aberto para garantir flexibilidade e escalabilidade da solução proposta.

## **2 METODOLOGIA**

Para desenvolver este projeto, foram criados programas em Python para análise e tratamento de dados, além de firmwares de monitoramento e protótipos eletrônicos que simulam a dinâmica funcional das vagas de estacionamento reais.

### **2.1 Sensores de Veículo**

Segundo Lin (2017), diversos tipos de sensores são utilizados em sistemas de estacionamento inteligente. Os sensores ultrassônicos, por exemplo, detectam a presença de veículos pela emissão de ondas sonoras, enquanto magnetômetros são

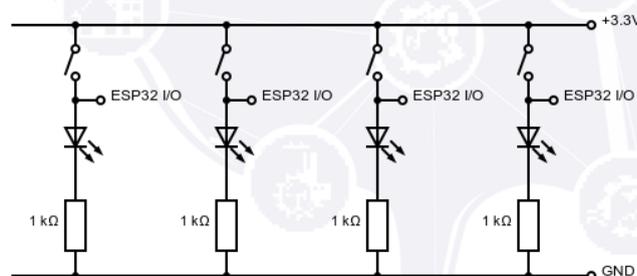
empregados para identificar variações no campo magnético causadas por veículos, ambos oferecendo precisão, mas a um custo financeiro elevado.

Nesse contexto, optou-se pelo uso de interruptores, os quais, apesar de não serem utilizados em soluções reais e de larga escala, são acessíveis e funcionalmente análogos em termos de detectar a ocupação das vagas, já que eles também sinalizam claramente dois estados de “ocupado” ou “livre”. Portanto, o uso desses componentes possibilitam uma abordagem mais direta e eficaz para ambientes experimentais.

## 2.2 Implementação em *Hardware*

O hardware é centrado no microcontrolador ESP 32, que coleta informações do ambiente por meio de suas entradas digitais conectadas a interruptores e LEDs vermelhos. Quando o interruptor é acionado, o LED acende, sinalizando que a vaga está ocupada; caso contrário, permanece apagado, indicando vaga disponível. Resistores *pull-down* mantêm a entrada em baixa tensão (lógica zero) quando o circuito está aberto e, ao acionar o interruptor, passa um sinal de alta tensão (lógica um) (ELETROGATE, 2019). O diagrama elétrico do projeto (Figura 1) foi planejado para replicar essa estrutura quatro vezes, simulando quatro vagas de estacionamento.

Figura 1 – Diagrama elétrico de interruptores pull-down em conexão paralela.

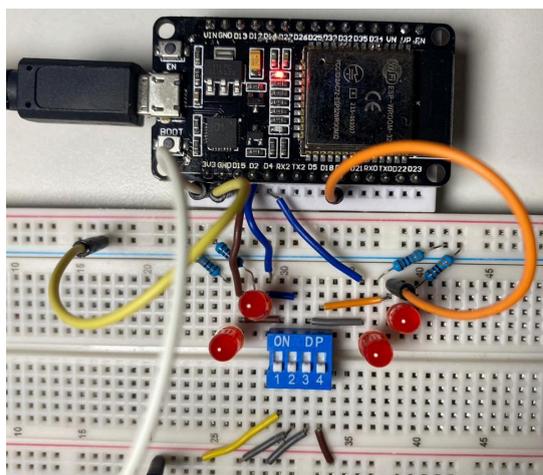


Fonte: Autoral (2024).

Após a montagem do circuito em placas de ensaio, ilustrada na Figura 2, utiliza-se a referência de terra e a alimentação de 3,3 volts fornecidas pela própria

placa.

Figura 2 – Protótipo de um sistema de estacionamento inteligente.



Fonte: Autoral (2024).

### 2.3 Comunicação MQTT

Conforme descrito por Yuan (2021), “o broker é um servidor intermediário que recebe todas as mensagens dos clientes e as encaminha para os destinatários relevantes”. No contexto deste projeto, a ESP 32 se conecta à rede Wi-Fi e ao broker de código aberto Mosquitto, que realiza essa intermediação. Após a conexão, o código monitora os pinos digitais da ESP 32 e publica as informações coletadas, de modo que os dados fiquem acessíveis remotamente.

### 2.4 Plataforma de Cidades Inteligentes

Desenvolveu-se uma aplicação em Python na plataforma de nuvem Google Colab, que recebe as mensagens enviadas pela ESP 32 por meio de um módulo MQTT disponibilizado gratuitamente pelo projeto Paho. O programa processa essas informações e calcula dados adicionais, como data, hora e tópico, então armazena todo material, ainda de forma privada, em uma planilha no Google Drive.

Por fim, todas as informações salvas são convertidas ao formato JSON e compartilhadas através de requisições HTTP com o microserviço de catálogo da

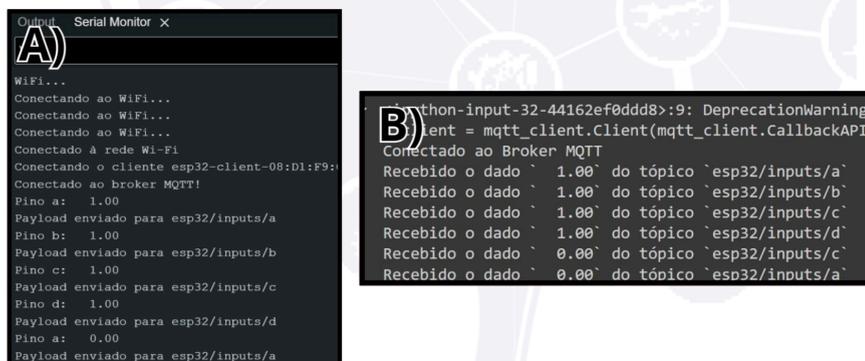
plataforma de cidades inteligentes, que é operada pelos servidores do Laboratório de Sistemas Distribuídos Inteligentes (LSDi) da UFMA. Assim, os dados serão acessíveis para uso em diversas interfaces de usuário e sistemas.

### 3 RESULTADOS E DISCUSSÕES

Os resultados demonstram que o circuito experimental foi capaz de abstrair funcionalmente um estacionamento, com LEDs que acendem e apagam individualmente e com níveis de tensão esperados nas entradas digitais. Entretanto, a utilização de interruptores em vez de sensores mais precisos, como os ultrassônicos e magnéticos descritos por Lin (2017), acaba por limitar demais a exatidão da coleta de informações. Esse fato abre margem para pesquisas futuras que explorem a integração de sensores mais sofisticados em projetos semelhantes.

Mesmo assim, a detecção dos estados das vagas e a transmissão das mensagens pelo microcontrolador foram satisfatórias, com todas as variações nas entradas digitais detectadas e publicadas no broker quase instantaneamente. As leituras obtidas após a ativação sequencial dos interruptores são apresentadas na Figura 3.

Figura 3 – Comunicação entre microcontrolador e aplicação em Python via MQTT. **A)** Saída da ESP32 com as mensagens publicadas; **B)** Saída do *script* Python com as mensagens recebidas.



```
Output Serial Monitor X
A)
WiFi...
Conectando ao WiFi...
Conectando ao WiFi...
Conectando ao WiFi...
Conectado à rede Wi-Fi
Conectando o cliente esp32-client-08:D1:F9:
Conectado ao Broker MQTT!
Pino a: 1.00
Payload enviado para esp32/inputs/a
Pino b: 1.00
Payload enviado para esp32/inputs/b
Pino c: 1.00
Payload enviado para esp32/inputs/c
Pino d: 1.00
Payload enviado para esp32/inputs/d
Pino a: 0.00
Payload enviado para esp32/inputs/a

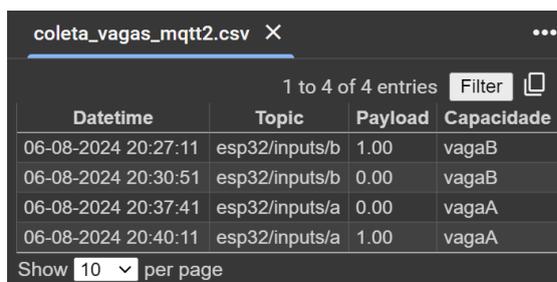
Python-input-32-44162ef0ddd8>: DeprecationWarning
Client = mqtt_client.Client(mqtt_client.CallbackAPI
Conectado ao Broker MQTT
Recebido o dado 1.00 do tópico `esp32/inputs/a`
Recebido o dado 1.00 do tópico `esp32/inputs/b`
Recebido o dado 1.00 do tópico `esp32/inputs/c`
Recebido o dado 1.00 do tópico `esp32/inputs/d`
Recebido o dado 0.00 do tópico `esp32/inputs/c`
Recebido o dado 0.00 do tópico `esp32/inputs/a`
```

Fonte: Autoral (2024).

Na Figura 4, observa-se que a aplicação desenvolvida no Google Colab associou precisamente todas as informações adicionais da mensagem recebida em

um arquivo CSV.

Figura 4 – Planilha com Informações das Vagas de Estacionamento.



Datetime	Topic	Payload	Capacidade
06-08-2024 20:27:11	esp32/inputs/b	1.00	vagaB
06-08-2024 20:30:51	esp32/inputs/b	0.00	vagaB
06-08-2024 20:37:41	esp32/inputs/a	0.00	vagaA
06-08-2024 20:40:11	esp32/inputs/a	1.00	vagaA

Fonte: Autoral (2024).

O estacionamento simulado, suas vagas e os valores dessas vagas ao longo do tempo foram registrados na instância da plataforma de cidades inteligente gerenciada por servidores da UFMA (Figura 5).

Figura 5 – Saída do script em Python após conversão dos dados e requisições com a API.

```

Convertendo planilha em JSON...
{"data":[{"vagaA1":1.0,"timestamp":"2024-08-01 12:00:00"}, {"vagaA2":1.0,"timestamp":"2024-08-01 12:00:00"}]}
Adicionando dados das capacidades ao recurso 'Estacionamento CCET' da API
OK!
Exibindo dados do recurso 'Estacionamento CCET'...
{
  "resources": [
    {
      "capabilities": {
        "environment_monitoring": [
          {
            "date": "2024-08-01T12:00:00.000Z",
            "vagaA1": 1.0
          },
          {
            "date": "2024-08-01T12:00:00.000Z",
            "vagaA2": 1.0
          },
          {
            "date": "2024-08-01T12:00:00.000Z",
            "vagaA3": 1.0
          },
          {
            "date": "2024-08-01T12:00:05.000Z",
            "vagaA4": 0.0
          }
        ]
      }
    }
  ],
  "uuid": "4dd1ec51-0210-4b10-a1f8-2b68708730d4"
}

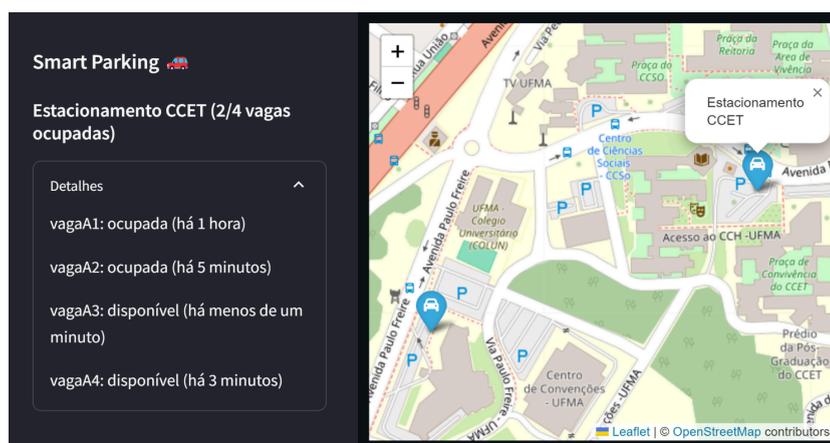
```

Fonte: Autoral (2024).

Estas informações compartilhadas com o microsserviço de catálogo da plataforma podem ser recuperadas pela API de coleta de dados. Para testar esse fato, foi desenvolvido um *dashboard* em Python com o *framework* Streamlit. Ao clicar

em cada ícone do mapa, é possível visualizar a quantidade total de vagas disponíveis, a ocupação de cada vaga e o tempo decorrido desde a última atualização. A interface do *dashboard* pode ser vista na Figura 6.

Figura 6 – Informações das vagas de um estacionamento selecionado no *dashboard*.



Fonte: Autoral (2024).

## 4 CONSIDERAÇÕES FINAIS

O projeto desenvolvido demonstrou que o sistema de estacionamento inteligente é uma solução eficiente para a gestão de vagas em organizações e áreas urbanas, integrando IoT, serviços em nuvem e protocolos de comunicação. Outrossim, o ESP32 provou ser uma opção acessível e eficaz para o monitoramento em tempo real, enquanto a integração com plataformas de código aberto, como o InterSCity, garantiu o armazenamento e compartilhamento dos dados.

Ainda que este protótipo utilize sensores simples e acessíveis, ele demonstra o potencial e a importância da integração de tecnologias para otimização da gestão de recursos, como observado por Rao et al. (2020) e CNN (2023). O trabalho, portanto, oferece uma solução prática e abre caminho para pesquisas futuras em mobilidade inteligente e sustentável.

## 5 REFERÊNCIAS

- [1] CNN. 2023. **Internet das Coisas: o que é, como funciona e exemplos de uso.** Disponível em: <<https://www.cnnbrasil.com.br/tecnologia/internet-das-coisas/>>. Acesso em: 26 set. 2024.
- [2] ELETROGATE. **Entendendo os Resistores de Pull-Up e Pull-Down.** Disponível em: <<https://blog.eletrogate.com/entendendo-os-resistores-de-pull-up-e-pull-down/>>. Acesso em: 25 ago. 2024.
- [3] Lin, T., Rivano, H., & Le Mouël, F. (2017). **A survey of smart parking solutions.** IEEE Transactions on Intelligent Transportation Systems, 18(12), 4-5. Disponível em: <<https://doi.org/10.1109/TITS.2017.2685143>>. Acesso em: 30 set. 2024.
- [4] Polycarpou, E., Lambrinos, L., & Protopapadakis, E. (2013). **Smart parking solutions.** IEEE 14th International Symposium (pp. 1-6). IEEE. Disponível em: <<https://ieeexplore.ieee.org/document/6583499>>. Acesso em: 30 set. 2024.
- [5] Rao, B. N., et al. (2022). **Smart traffic management system using IoT.** 2022 IEEE International Symposium (pp. 6-8). Warangal, India: IEEE. Disponível em: <<https://doi.org/10.1109/iSES54909.2022.00138>>. Data de acesso: 29 set. 2024.
- [6] Serejo, F. D. A. M. (2018). **Desenvolvimento de um sistema de monitoramento e controle para o processo de mistura de líquidos utilizando Internet das Coisas.** Curso de Engenharia Elétrica Industrial - IFMA, São Luís. Disponível em: <[https://www.sba.org.br/open\\_journal\\_systems/index.php/cba/article/download/71/5](https://www.sba.org.br/open_journal_systems/index.php/cba/article/download/71/5)>. Data de acesso: Data de acesso: 30 set. 2024.
- [7] Tomasiello, D. B., Pereira, R. H. M., & Nadalin, V. G. (2023). **Os impactos desiguais do congestionamento urbano no acesso a empregos.** Instituto de Pesquisa Econômica Aplicada (IPEA), Brasília, pp. 9-11. Disponível em: <<https://doi.org/10.38116/td2922-port>>. Data de acesso: 29 set. 2024.
- [8] YUAN, Michael. **Conhecendo o mqtt.** IBM, 2017. Disponível em: <<https://www.ibm.com/developerworks/br/library/iot-mqtt-why-good-foriot/index.html>>. Acesso em: 23 ago. 2024.