



## **DESENVOLVIMENTO DE UMA API PARA ENSINO DE IOT COM ESP32 EM DISCIPLINAS DE LABORATÓRIO DE ENGENHARIA ELÉTRICA**

DOI: 10.37702/2175-957X.COBENGE.2022.3837

Otávio Cosme Matias - otaviocmatias@gmail.com  
Instituto Federal de Educação Ciência e Tecnologia de Minas Gerais

Lucas Frederico Jardim Meloni - lucas.meloni@ifmg.edu.br  
Instituto Federal de Educação Ciência e Tecnologia de Minas Gerais

Felipe de Sousa Silva - felipe.silva@ifmg.edu.br  
Instituto Federal de Educação Ciência e Tecnologia de Minas Gerais

**Resumo:** *A evolução tecnológica promovida pelo desenvolvimento da eletrônica e dos meios de comunicação trouxe novas perspectivas para os meios de fabricação, o que criou o conceito de Indústria 4.0, caracterizada pela ampla integração de tecnologias e dispositivos conectados à Internet (IoT). O perfil exigido para os novos profissionais que ingressarão nessa indústria também mudou, exigindo autonomia no aprendizado, capacidade de se adaptar a novidades e domínio de amplas tecnologias. Isso traz novos desafios às instituições de ensino, tornando fundamental o desenvolvimento de novas ferramentas didáticas. Propõe-se nesse trabalho o desenvolvimento de uma Interface de programação de aplicações didáticas, baseada em módulos de desenvolvimento ESP32, que poderá ser utilizada como ferramenta de ensino de IoT para cursos de graduação em Engenharia Elétrica. Através deste ambiente virtual, o usuário terá acesso a exemplos, roteiros e outros materiais didáticos que facilitarão o desenvolvimento e teste de diversos sistemas diversos, sendo também capaz de alterar e monitorar variáveis e configurações de periféricos do ESP32 remotamente.*

**Palavras-chave:** ESP32, Embarcados, Laboratório



## DESENVOLVIMENTO DE UMA API PARA ENSINO DE IOT COM ESP32 EM DISCIPLINAS DE LABORATÓRIO DE ENGENHARIA ELÉTRICA

### 1 INTRODUÇÃO

A Internet das Coisas (IoT) é definida como a integração entre diferentes objetos, não necessariamente apenas computadores, realizada através da Internet, o que possibilita configurações ou monitoramentos remotos, promovendo a modularização de sistemas capazes de operar de forma autônoma (CARRION; QUARESMA, 2019). Esta é considerada o alicerce para a Indústria 4.0, um novo modelo de produção, com intensa automatização e que une os processos físicos e digitais através de redes de comunicação reduzindo ao mínimo a necessidade de intervenção humana (FREITAS, 2018). Assim, a formação de novos profissionais em engenharia, capazes de se adaptarem a diversas mudanças tecnológicas e também integrar várias tecnologias emergentes (SILVA; DUARTE; GUILOW, 2020) torna-se um novo desafio para as instituições de ensino.

Através da plataforma Arduino foi possível iniciar o desenvolvimento de parte destas novas competências, como mostrado em cursos de graduação em Física (MOREIRA et al., 2018) e em engenharias eletrônica e elétrica (VILAR et al., 2018), (MATOS et al., 2019). O baixo custo das placas de desenvolvimento Arduino UNO e Arduino Mega, programáveis através de uma interface de desenvolvimento (IDE) amigável (GEDDES, 2017) motivam o surgimento de oficinas e cursos de extensão, como (CAMPOS; SANTOS; OLIVEIRA, 2019) e possibilitam que os alunos tenham contato com diversos sensores, atuadores e também desenvolvam aplicações simples conectadas à Internet.

O sucesso da plataforma Arduino trouxe novas plataformas, como o *Micro:bit*, com uma placa de desenvolvimento *OpenSource* criada exclusivamente para fins educacionais, que pode ser programada em linguagens gráficas como o Microsoft MakeCode ou textuais como Python (MICRO:BIT, 2020). Esta placa não conta com conexão WiFi própria e necessita de módulos externos. Por isso, seu custo é muito superior ao das placas Arduino Uno e Mega, o que dificulta sua aquisição por instituições de Ensino.

Também se destacam as placas *Raspberry Pi* (PI, 2021a), que são computadores portáteis capazes de rodar sistemas operacionais embarcados, fato que possibilita o desenvolvimento de aplicações avançadas. Entretanto, as mesmas também apresentam um custo maior e, apesar de existirem modelos mais baratos como o *Raspberry Pi Pico* (PI, 2021b), também exigem módulos externos para conexão com a Internet.

Em 2016 foram lançados os módulos ESP-32, com rádio para comunicação IEEE 802.11b/g/n e Bluetooth 4.2 BLE, maior capacidade de memórias e núcleo de processamento DualCore (ESPRESSIF, 2020), dedicados ao desenvolvimento de dispositivos conectados à Internet (IoT). Estes módulos utilizam o mesmo desenvolvimento Arduino IDE, através das linguagens de programação C/C++, sem a necessidade de circuitos de programação externos. Com o ESP32 foi possível desenvolver transmissores sem fio inteligentes, para aplicações em sistemas de supervisão, controle e aquisição de dados (AGHENTA; IQBAL, 2019), sistemas de automação residencial (PEREIRA et al., 2020) e até medidores de energia (ALEXANDRE; FILHO; SANTANA, 2019) já foram desenvolvidos com ESP32, todos com capacidade de se comunicarem à Internet e possibilitarem monitoramento e alteração de parâmetros de forma remota.

Através destes diversos recursos opensource surgiram também plataformas de desenvolvimento IoT, como o AdaFruit IO (ADAFRUIT, 2021) e o Thingier.io (THINGER.IO,

2021) que são portais com ferramentas de desenvolvimento de aplicações embarcadas próprias para a Internet. Entretanto, apesar de disponibilizarem interface de programação gratuita, estas ferramentas apresentam uma interface relativamente complexa para disponibilizar seus recursos, o que pode dificultar o primeiro contato de novos alunos.

Assim, neste artigo é apresentada uma Interface de Programação de Aplicações (API) dedicada ao ensino de IoT em cursos de Laboratório de Sistemas Embarcados em Engenharia Elétrica. Propõe-se o desenvolvimento de um site, hospedado de forma local nos laboratórios, que servirá como ambiente virtual de ensino de recursos aos alunos que serão capazes de compreender aplicações de monitoramento e ações remotas, servidor/cliente, para realização de projetos e aprendizagem. A API conta com manuais de instrução, guias sobre programação e até vídeos explicativos que ensinam conceitos básicos de programação Web, como as linguagens HTML5, CSS, JavaScript, pouco exploradas em cursos de graduação em Engenharia Elétrica e que auxiliarão os estudantes a capacitarem-se para atuar no novo cenário da Indústria 4.0.

Na seção 2 será apresentada a metodologia estabelecida para desenvolver os códigos, através da seleção de aplicações básicas mais comuns a estudantes de Engenharia Elétrica. Na seção 3 será discutida a criação da API e dos materiais didáticos presentes nela. Por fim, na seção 4 serão discutidos os resultados obtidos e as conclusões.

## 2 REFERENCIAL TEÓRICO

O desenvolvimento tecnológico promoveu grandes mudanças sociais e econômicas através da integração de diversos dispositivos, flexibilização na tomada de decisões, modularização de sistemas e monitoramento remoto de processos de fabricação. Além de reduzir custos de produção, aumentar a economia de energia e garantir a preservação do meio ambiente, tais tecnologias também criaram o novo conceito de Indústria 4.0, cujos pilares são definidos, de acordo com (CARVALHO; FILHO, 2018) e (BRETTEL et al., 2018), pela capacidade de operação em tempo real, virtualização, descentralização de tomada de decisões, orientação de serviços e modularização de sistemas.

A ponte entre estas diversas tecnologias é a Internet das Coisas (IoT), definida segundo (CASAGRAS, 2009) e (MIOLA, 2019) como uma infraestrutura de rede global entre dispositivos capazes de se comunicar, capturar ou produzir dados com elevada autonomia. Nesta rede coexistem tanto os objetos físicos (computadores, sensores, atuadores, instrumentos, máquinas ferramentais, etc.) quanto os virtuais (softwares, inteligências artificiais, serviços promovidos por protocolos de comunicação, aplicativos móveis, etc.) e todos são capazes de interagir, produzir e trocar informações, além de operar de forma independente. Há uma previsão de que até 2030 existirão mais de 12 trilhões de equipamentos conectados (SPANULESCU, 2020) e através das definições apresentadas verifica-se que a infraestrutura formada por dispositivos IoT é o alicerce para a Indústria 4.0, como mostrado na Figura 1.

Figura 1 – Evoluções tecnológicas na Indústria.



Fonte: imagem retirada de (FILHO, 2018).

Dominar as mais variadas tecnologias da Indústria 4.0 exigem um perfil de profissional diferente, com elevada capacidade de aprendizagem, muito conhecimento técnico e também formação ativa e empreendedora (SILVA; DUARTE; GUILLOW, 2020; SILVA, 2018). Com isso, o cenário mundial e a educação sofreram consideráveis mudanças. De acordo com (BAUMAN, 2009), a educação mundial passou de um antigo formato sólido - onde os conhecimentos adquiridos na escola e nas universidades eram considerados suficientes para solucionar quase todos os possíveis problemas a serem encontrados – para um formato líquido, caracterizado por incertezas, imprevisibilidade e constantes atualizações.

Neste contexto, o professor e as instituições de ensino devem procurar novas ferramentas de ensino (DIESEL; BALDEZ; MARTINS, 2017), baseadas em metodologias ativas, onde o docente deve dá enfoque ao protagonismo dos alunos, desenvolver a autoaprendizagem e a curiosidade destes, como indicado nos princípios ilustrados na Figura 2.

Figura 2 – Princípios que constituem as metodologias ativas de ensino.



Fonte: imagem retirada de (DIESEL; BALDEZ; MARTINS, 2017).

Além da metodologia representada na Figura 2, também é necessário desenvolver novas plataformas de ensino, com materiais didáticos de fácil compreensão e que possam ser acessados de diferentes ambientes e de forma clara e organizada (CARDOSO et al., 2020). Por exemplo, em (CARVALHO; FILHO, 2018) é proposto um sistema de aprendizagem móvel, que consiste em um aplicativo para celular desenvolvido para que o aluno consiga estudar conceitos sobre IoT utilizando para isso plataformas móveis, tornando-o capaz de aproveitar qualquer oportunidade para aprender e dedicar-se de forma continuada. Há também a tendência em incorporar no ensino as novas tecnologias presentes na Indústria 4.0 como mostrado na tentativa de (PASZKIEWICZ et al., 2021) em utilizar realidade virtual como ferramenta de ensino.

As bases para compreender a IoT são, ainda segundo (SPANULESCU, 2020), as diversas linguagens de programação como C/C++, HTML/CSS, Javas, JavaScript, completadas por tecnologias como Ajax, MQTT e Websockets. Logo, o desenvolvimento de novas ferramentas deve lidar com todas estas linguagens de forma direta, através de códigos textuais, ou de forma gráfica.

Uma maneira de facilitar o ensino de IoT e apresentar estas diversas linguagens de forma única é através da criação de Interfaces para programação de Aplicações (APIs) personalizadas, que constituem um conjunto de bibliotecas de *software* contendo funções, protocolos e regras de comunicação, arquivos de configuração, ferramentas gráficas para controle e monitoramento (botões, caixas de seleção, objetos gráficos, campos para entradas de texto, etc.), além de exemplos práticos de aplicação que auxiliarão os programadores e novos estudantes a desenvolver seus próprios projetos (HERYANDI, 2018). Geralmente as APIs são disponibilizadas de forma organizada, em sites ou aplicativos que contêm documentações como manuais de instrução, esquemas elétricos, diagramas de conexão, etc. Um exemplo de imagem de uma API é mostrado em (ADAFRUIT, 2021) e (THINGER.IO, 2021) que fornecem ferramentas para desenvolvimento IoT baseadas em Arduino, Raspberry Pi e Micro.bit.

Um exemplo de API pode ser encontrado em (NATHOO et al., 2020), onde foi proposto o uso de interfaces de usuário tangíveis (TUI) para o ensino de IoT, onde os estudantes eram capazes de acessar ferramentas virtuais através da integração com objetos reais, utilizando sensores de movimento e captura de imagens. Outro exemplo de interface de programação para ensino de IoT é mostrado em (TAN et al., 2018), no qual foi proposto um conceito de sistema de ensino em que os estudantes utilizam ferramentas leitores RFID e leitura de códigos QR para acompanhar aulas e interagir. Nesta concepção, o professor é capaz de monitorar e cadastrar as interações dos alunos, promover questões e jogos educativos. Além de propor um novo conceito para sala de aula virtual, várias tecnologias utilizadas em IoT também puderam ser ensinadas nesta abordagem.

Também são encontrados esforços para o desenvolvimento de novas ferramentas de programação para sistemas embarcados, como a mostrada em (CHACZKO; BRAUN, 2017), para programação de placas *Raspberry Pi* através de diagramas de blocos. Foi relatado que os estudantes não universitários que testaram esta plataforma conseguiram rapidamente familiarizar-se com a ferramenta e desenvolver aplicações de média complexidade, além de ganhar experiência em programação de linguagem JavaScript.

Outra plataforma de programação proposta em (BAK; CHANG; CHOI, 2020) utiliza blocos gráficos para desenvolvimento de sistemas de automação residencial. Esta ferramenta conta com capacidades de verificação de redundâncias, inconsistências e circularidade, sendo avaliada por 33 participantes que a descreveram como intuitiva e compreensível para os usuários.

Entre os vários recursos disponíveis para ensino de IoT, encontram-se os módulos microcontrolados ESP32, de baixo custo e dedicados ao desenvolvimento de aplicações WiFi e *Bluetooth 4.0 BLE* (ESPRESSIF, 2020). Estes módulos contam com ferramentas próprias e gratuitas para programação e desenvolvimento de aplicações, disponibilizadas em uma API fornecida pelo próprio fabricante (ESPRESSIF, 2021). Porém, seu uso mais popular é em conjunto com a plataforma *Arduino IDE*, já utilizada em ambientes de ensino (CARDOSO et al., 2020).

O ESP32 apresenta um grande potencial para ensino de IoT, como demonstrado em (BENITEZ; SYMONDS; ELGUEZABAL, 2020), onde foi desenvolvido um braço robótico educacional, de baixo custo, capaz de ser controlado através de dispositivos móveis como *smartphones* e que foi utilizado para auxiliar estudantes no ensino remoto de robótica, durante a pandemia causada pelo COVID-19. Outros projetos como sistemas detectores de interrupções momentâneas de energia (MATIAS et al., 2020), sistemas IoT para monitoramento e aquisição de dados (AGHENTA; IQBAL, 2019) ou controladores de micro fluídos com capacidade de aquisição de dados sem fio (WATSON; SENYO, 2019).

Neste contexto, a proposta para este projeto artigo é apresentar uma API desenvolvida para ensino de IoT com módulos ESP32, personalizada para uma instituição de Ensino Federal. Para isso, foram desenvolvidos códigos, bibliotecas próprias e manuais de instrução para criação e utilização de ferramentas gráficas de controle e supervisão de variáveis (botões, caixas de seleção, exibição de gráficos, entradas de texto e números, etc.) utilizando linguagens como C/C++, HTML5/CSS3 e Javascript. O desenvolvimento desta API visa minimizar dificuldades encontradas em ensinar o IoT dentro da grade de Engenharia Elétrica, como pouca carga horária, dificuldade encontrada pelos professores em preparar apostilas e roteiros de laboratório, além da carência de disciplinas específicas ao ensino de linguagens de programação Web. Além disso, este trabalho também pretende contribuir com subsídios para adequar o perfil de profissionais recém formados pelo IFMG ao exigido pela Indústria 4.0, bem como atuar como um facilitador para implementação de metodologias ativas de ensino como Aprendizagem Baseada em Projetos.

### 3 CRIAÇÃO DA API

Inicialmente foi estabelecido como pré-requisito que os alunos que utilizariam a API a ser desenvolvida já tivessem contato prévio com os módulos ESP32. A partir disto, após ter sido feita uma pesquisa bibliográfica sobre aplicações IoT com ESP32, foram verificados três métodos para a aplicação: arquitetura Cliente/Servidor; utilização do protocolo MQTT e *Websockets*. No primeiro método, o módulo ESP32 hospeda uma pequena página HTML/CSS e conecta-se a clientes através de Javascript. Através do protocolo MQTT, o ESP32 comporta-se como um *publisher* ou *subscriber*, que acessará um servidor *broker* para publicar e acessar dados em nuvem. O último método investigado utiliza controles mais avançados através de bibliotecas *Websockets*.

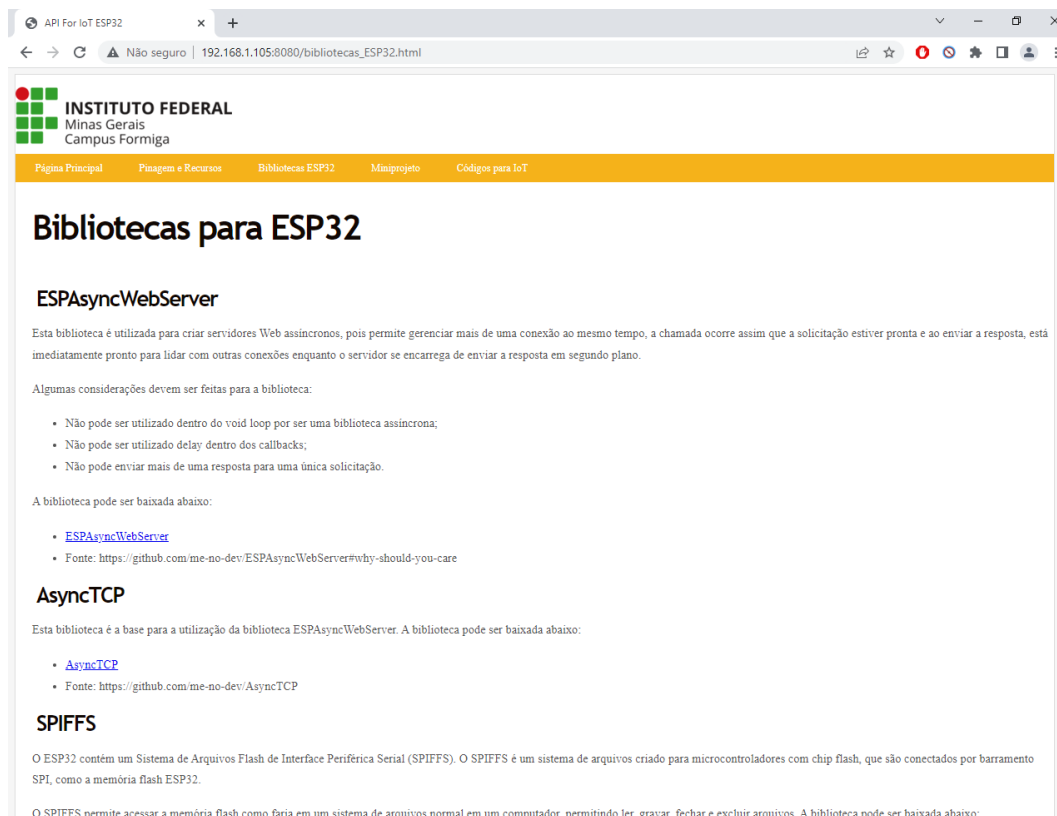
Após a pesquisa, optou-se pela topologia Cliente/Servidor como a utilizada para desenvolver os códigos devido a sua maior facilidade de configuração e ampla possibilidade de criação. Dessa forma, foi iniciado o desenvolvimento de controles básicos utilizando a topologia selecionada, em que um módulo ESP32 opera como um servidor de serviços e hospeda uma página HTML e, um *WebBrowser* poderá requisitar serviços como cliente.

Posteriormente, foram criados quatro tipos de aplicações IoT que possam ser utilizadas em laboratório e que os alunos utilizem ao longo de sua formação e vida profissional, tal como no Trabalho de Conclusão de Curso (TCC) e programação em

Javascript. Para elaboração da API foram determinadas 4 etapas desenvolvidas em sequência.

A API seria acessada através de um site, hospedado localmente, sendo estruturado e criado o site utilizando as linguagens de programação HTML e CSS, bem como um *layout* próprio para o site. Foram inclusas neste site, as bibliotecas necessárias para a execução dos códigos juntamente com uma breve descrição de instalação e funcionalidade destas. A página do site em que se encontram as bibliotecas é apresentada na "Figura 3".

Figura 3 – Página do site com as bibliotecas.



Fonte: imagem elaborada pelos autores.

Os primeiros códigos consistem em um acionamento de entradas e saídas digitais. É possível controlar saídas digitais ou analógicas do ESP32 pelo *Browser* enviando requisições. A página HTML do primeiro código é mostrada na "Figura 4 (a)". O segundo código trata-se de uma barra gráfica que permite variar a intensidade luminosa de um LED pelo *Browser*, a variação é feita através de Modulação por Largura de Pulso (PWM) em uma saída digital do ESP32. A "Figura 4 (b)" apresenta a página HTML segundo código.

Figura 4 – Páginas dos primeiros códigos. (a) página do primeiro código. (b) página do segundo código.



(a)

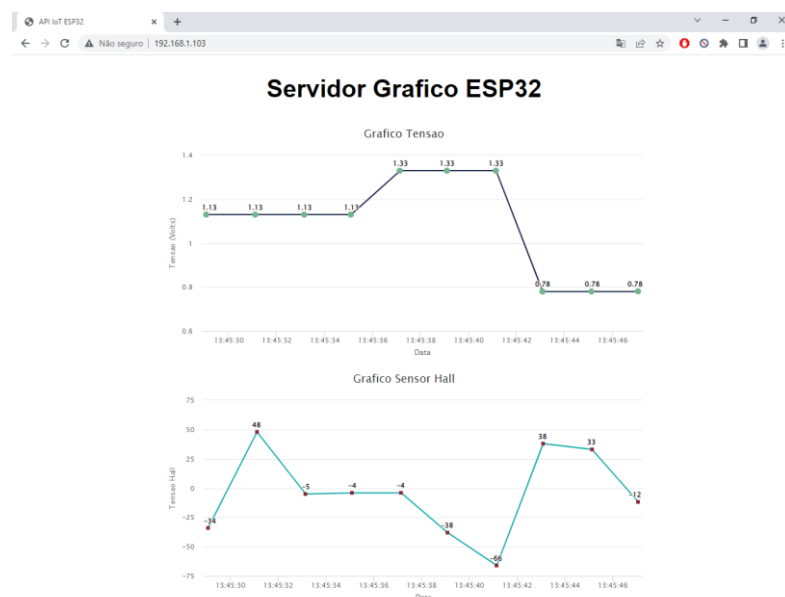


(b)

Fonte: imagem elaborada pelos autores.

O terceiro código é um pouco mais complexo, consistindo em um sistema para supervisão dos dados obtidos através da leitura de uma entrada analógica do ESP32. O último consiste na exibição da leitura de uma entrada analógica do ESP32 na forma de gráfico, sendo possível modificar as escalas e o *layout* do gráfico. Todos os códigos elaborados foram comentados passo-a-passo com a explicação das funcionalidades. A “Figura 5” mostra a página gerada para exibir a informação de leitura em um gráfico.

Figura 5 – Página HTML do terceiro código.



Fonte: imagem elaborada pelos autores.

Por fim, notou-se a necessidade de um miniprojeto que unisse todos os códigos criados, sendo elaborado um Ohmímetro a partir da entrada analógica do ESP32. A fim de





exemplificar a aplicabilidade dos conceitos, linguagens de programação e dispositivos IoT. Na “Figura 6” e “Figura 7” são mostradas as duas páginas HTML’s criadas para a execução do miniprojeto Ohmímetro. Na “Figura 6” a variação da observada no gráfico ocorreu devido a variação intencional realizada por um potenciômetro.

Figura 6 – Miniprojeto Ohmímetro.



Fonte: imagem elaborada pelos autores.

Figura 7 – Miniprojeto Ohmímetro.

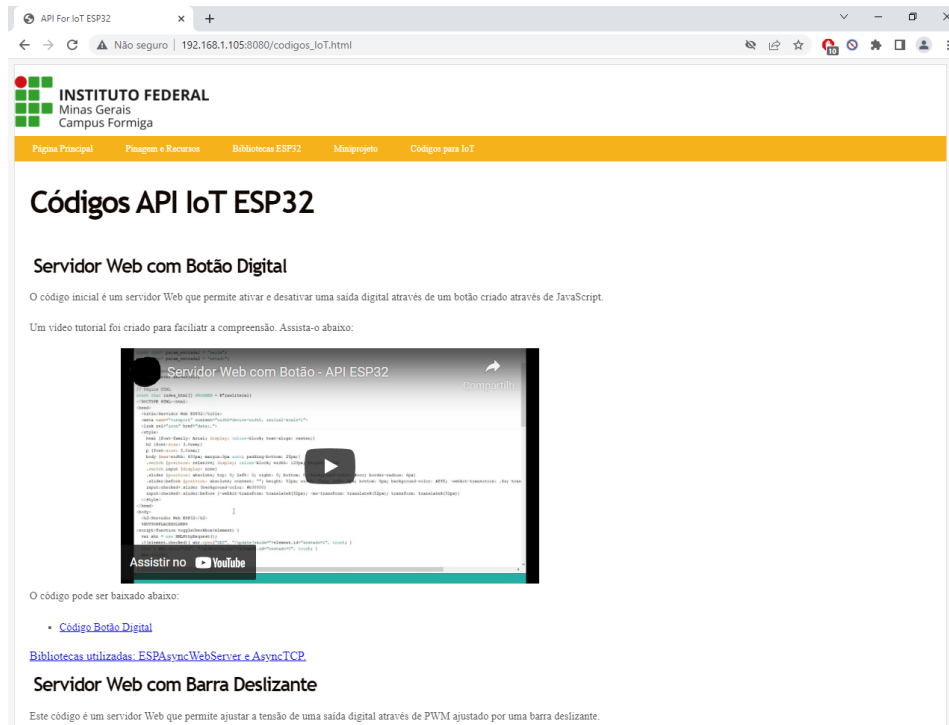


Fonte: imagem elaborada pelos autores.

A terceira etapa trata da inserção dos códigos e do miniprojeto elaborados ao *site*, assim como a criação dos vídeos explicativos armazenados no *YouTube*®, diagramas de montagem e aspectos básicos para programação em Javascript. Para elaboração dos diagramas de montagem, recorreu-se ao uso do *software* Fritzing®. A página do *site* com os vídeos explicativos e o esquemático de montagem do miniprojeto são mostrados na “Figura 8” e “Figura 9”, respectivamente.

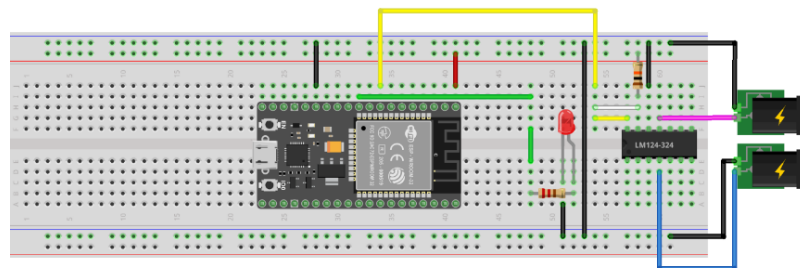


Figura 8 – Página do site com os vídeos explicativos.



Fonte: imagem elaborada pelos autores.

Figura 9 – Esquemático de montagem.



- PINAGEM:**
- Fio Preto: GND
  - Fio Vermelho: +3,3 V
  - Fio Amarelo: GPIO33 (ADC)
  - Fio Verde: GPIO26 (PWM)
  - Fio Verde: GPIO2 (LED)
  - Fio Rosa: -12 V
  - Fio Azul: +12 V
  - Fio Branco: Inserir o resistor a ser medido nos terminais do fio (100 a 10k ohms)

fritzing

Fonte: elaborado pelos autores, com o auxílio com do software Fritzing®.

Na última etapa, foi estudada as formas de hospedagem do *site* da API e verificou-se que a melhor alternativa foi hospedar em um computador na rede local de um laboratório. Dessa forma, basta um computador conectado à rede local hospedar a API que os demais computadores conectados à mesma rede terão acesso a ela. Para a hospedagem da API foi utilizado um servidor local criado por Node.js® e suas bibliotecas necessárias, sendo sua instalação explicada em um vídeo também criada nessa etapa.

## 4 RESULTADOS E DISCUSSÕES

Ao fim do processo de desenvolvimento, a API possui os seguintes recursos: códigos e um miniprojeto para ensino de IoT, três esquemáticos de montagem e pinagem das placas ESP32, vídeos didáticos sobre explicação dos códigos e uso das linguagens de programação, bibliotecas e tutoriais para execução do *site* na rede local de um laboratório.

A API finalizada foi testada no laboratório de eletrônica do IFMG *Campus* Formiga, sendo verificada sua funcionalidade por docentes responsáveis pelas disciplinas laboratoriais do curso de Engenharia Elétrica. A hospedagem do *site* foi realizada na rede local do laboratório e foi possível acessar todos os conteúdos da API, bem como reproduzir as montagens e utilizar os códigos propostos.

Os docentes avaliaram que a API propicia a integração do ensino de IoT nas disciplinas laboratoriais, pois os recursos oferecidos suprem dificuldades tais como, desinteresse dos alunos, falta de materiais didáticos sobre IoT destinados a estudantes de engenharia elétrica e o pouco conhecimento dos alunos em linguagem de programação *Web*.

Os resultados apresentados referem-se ao desenvolvimento e funcionalidades da API, necessitando ainda de mais verificações e contribuições para a avaliação total da API. Dessa forma, notou-se a viabilidade de explorar novas possibilidades de utilização em outras disciplinas relacionadas a indústria 4.0 e expandir o desenvolvimento do ensino de IoT.

## 5 CONSIDERAÇÕES FINAIS

O presente trabalho teve como objetivo apresentar a elaboração e os resultados parciais de projeto de pesquisa em desenvolvimento em iniciação científica, que foi iniciado em 2021. Destaca-se que este projeto teve como uma de suas motivações proporcionar uma interface didática para melhorar o ensino de IoT para discentes em aulas laboratoriais do Curso de Engenharia Elétrica em uma instituição Federal de Ensino.

Objetiva-se futuramente expandir o uso desta API nas disciplinas laboratoriais, coletar e aplicar a mesma em sala de aula para a partir do *feedback* dos docentes estabelecer melhorias, novos recursos e atualizar os materiais didáticos. Além disso, pretende-se disponibilizar a API em domínio público para que outras instituições possam utilizá-la em suas disciplinas laboratoriais do curso de Engenharia Elétrica com elementos de indústria 4.0.

Por fim, destaca-se que a API também tem por característica estimular a aprendizagem ativa, auxiliando os alunos a aprenderem por si sós, dentro do laboratório, auxiliando seu desenvolvimento próprio.

## AGRADECIMENTOS

Os autores deste projeto agradecem a todos os membros do IFMG *Campus* Formiga, do Grupo de Soluções em Engenharia (GSE) e da Pró-Reitoria de Extensão do IFMG que contribuíram para realização deste trabalho.

## REFERÊNCIAS

ADAFRUIT. **Adafruit IO**. Disponível em: <<https://io.adafruit.com/>>. Acesso em: 7 maio. 2021.

AGHENTA, L. O.; IQBAL, M. T. Low-Cost, Open Source IoT-Based SCADA System Design Using Thinger.IO and ESP32 Thing. **MDPI Electronics**, v. 8, n. 822, 2019.

ALEXANDRE, G. B.; FILHO, M. DÉMERY A.; SANTANA, V. B. **Projeto e montagem de um medidor de energia elétrica didático e de baixo custo**. XLVII Congresso Brasileiro de Educação em Engenharia e II Simpósio Internacional de Educação em Engenharia da ABENGE. **Anais...**Fortaleza: 2019

BAK, N.; CHANG, B.-M.; CHOI, K. Smart Block: A visual block language and its programming environment for IoT. **Journal of Computer Languages**, v. 60, p. 1–19, 2020.

BAUMAN, Z. Os desafios da educação: aprender a caminhar sobre areias movediças. **Cadernos de Pesquisa**, v. 39, n. 137, p. 661–684, 2009.

BENITEZ, V. H.; SYMONDS, R.; ELGUEZABAL, D. E. Design of an affordable IoT open-source robot arm for online teaching of robotics courses during the pandemic contingency. **HardawreX**, v. 8, p. 1–15, 2020.

BRETTEL, M. et al. How virtualization, Decentralization and Network Building change the manufacturing Landscape: An Industry 4.0 Perspective. **International Journal of Information and Communication Engineering**, v. 8, n. 1, 2018.

CAMPOS, G. L.; SANTOS, A. P. L. DOS; OLIVEIRA, P. S. DE. **Arduino: uma nova ferramenta para a aprendizagem, estudo de caso no IFMG Campus Formiga**. XLVII Congresso Brasileiro de Educação em Engenharia e II Simpósio Internacional de Educação em Engenharia da ABENGE. **Anais...**Fortaleza: 2019

CARDOSO, O. E. et al. **Desenvolvimento de uma plataforma de aprendizagem de sistemas embarcados baseada em ESP32**. XLVII Congresso Brasileiro de Educação em Engenharia e III Simpósio Internacional de Educação em Engenharia da ABENGE. **Anais...**2020

CARRION, P.; QUARESMA, M. Internet da Coisas (IoT): Definições e aplicabilidade aos usuários finais. **HFD**, v. 8, n. 15, p. 49–66, 2019.

CARVALHO, E. DOS S. DE S.; FILHO, N. F. D. Proposta de um sistema de aprendizagem móvel com foco nas características e aplicações práticas da indústria 4.0. **Revista ibérica de Sistemas e Tecnologias de Informação**, v. 27, n. 6, 2018.

CASAGRAS, E. F. P. **RFID and Inclusie Model for the Internet of Things - Final Report**. [s.l.] Eu Fip7 Project, 2009.

CHACZKO, Z.; BRAUN, R. **Learning data engineering: Creating IoT apps using the node-RED and the RPI technologies**. 16th International Conference on Information Technology Based Higher Education and Training. **Anais...**Ohrid: 2017

DIESEL, A.; BALDEZ, A. L. S.; MARTINS, S. N. Os princípios das metodologias ativas de ensino: uma abordagem teórica. **Revista Thema**, v. 14, n. 1, p. 268–288, 2017.

ESPRESSIF. **ESP32 Series Datasheet - Version 3.4.** Disponível em:  
<<https://www.espressif.com/en/products/devkits/esp32-devkitc/overview>>. Acesso em: 7 ago. 2020.

ESPRESSIF. **ESP32-IDF Programming Guide.** Disponível em:  
<<https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/index.html>>. Acesso em: 27 jan. 2021.

FILHO, P. N. **Indústria 4.0: o que é e quais os benefícios?** Disponível em:  
<<https://www.harbor.com.br/harbor-blog/2018/01/04/industria-4-0/>>. Acesso em: 6 maio. 2021.

FREITAS, A. DE P. P. **Análise bibliométrica da produção científica sobre indústria 4.0.** Uberlândia: Trabalho de Conclusão de Curso Universidade Federal de Uberlândia, 2018.

GEDDES, M. **Manual de Projetos do Arduino: 25 Projetos Práticos para Começar.** Primeira ed. São Paulo: Novatec, 2017.

HERYANDI, A. **Developing Application Programming Interface (API) for Student Academic Activity Monitoring using Firebase Cloud Messagint (FCM).** International Conference on Informatics, Engineering, Science and Technology. **Anais...**Bandung: 2018

MATOS, B. T. M. et al. Ensino da Robótica: O Arduino como Ferramenta Didática. In: ATENA (Ed.). **Educação: Políticas, Estrutura e Organização 6.** 6. ed. Ponta Grossa: [s.n.]. p. 263–272.

MICRO:BIT. **Introduction.** Disponível em: <<https://microbit.org/get-started/first-steps/introduction/>>. Acesso em: 10 fev. 2021.

MIOLA, M. **Desenvolvimento de componentes para App Inventor e servidor para dispositivos IoT.** Florianópolis: Trabalho de Conclusão de Curso da UFSC, 2019.

MOREIRA, M. P. C. et al. Contribuições do Arduino no ensino de Física: uma revisão sistemática de publicações na área do ensino. **Caderno Brasileiro de Ensino de Física**, v. 35, n. 3, p. 721–745, 2018.

NATHOO, A. et al. Using tangible user interfaces for teaching concepts of internet of things. **Interactive Technology and Smart Education**, v. 17, n. 2, p. 133–158, 2020.

PASZKIEWICZ, A. et al. Methodology of Implementing Virtual Reality in Education for Industry 4.0. **MDPI Sustainability**, v. 13, n. 5049, p. 1–25, 2021.

PEREIRA, D. A. T. et al. **Automação Residencial: comando remoto de cargas residenciais através de plataforma própria com comunicação Wireless.** XVIII Conferência de Estudos em Engenharia Elétrica. **Anais...**2020

PI, R. **Raspberry Pi.** Disponível em: <<https://www.raspberrypi.org/>>. Acesso em: 7 maio. 2021a.

PI, R. **Raspberry Pi Pico.** Disponível em:  
<<https://www.raspberrypi.org/products/raspberry-pi-pico/>>. Acesso em: 7 maio. 2021b.

SILVA, A. E. DA; DUARTE, E. R.; GUILOW, G. **Formação de Engenheiros para a Indústria 4.0.** XLVIII Congresso Brasileiro de Educação em Engenharia e III Simpósio

Internacional de Educação em Engenharia da ABENGE. **Anais...2020**

SILVA, R. N. DE A. **Sistema de monitoramento de fornecimento de energia elétrica para uma residência**. Brasília: Trabalho de Conclusão de Curso Centro Universitário de Brasília, 2018.

SPANULESCU, S. **ESP32 programming for the Internet of Things**. 2. ed. [s.l.] SEVER SPÂNULESCU, 2020.

TAN, P. et al. Teaching Management system with Applications of RFID and IoT Technology. **MDPI Education Sciences**, v. 8, n. 26, p. 1–13, 2018.

THINGER.IO. **Thinger.io**. Disponível em: <<https://thinger.io/>>. Acesso em: 7 maio. 2021.

VILAR, S. R. et al. Utilização da plataforma Arduino para a solução de problemas por alunos da disciplina de introdução à Engenharia Elétrica e algoritmos e lógica de programação do IFPB. **Revista Principia - Divulgação científica e tecnológica do IFPB**, n. 39, p. 72–78, 2018.

WATSON, C.; SENYO, S. All-in-one automated microfluids control system. **Hardwarex**, v. 6, p. 1–10, 2019.

## **DEVELOPMENT OF AN IOT TEACHING API BASED ON THE ESP32 PLATFORM FOR LABORATORY SUBJECTS IN ELECTRICAL ENGINEERING UNDERGRADUATE COURSES**

**Abstract:** *The technological evolution promoted by the development of electronics and the means of communication brought new perspectives to the means of manufacturing, which created the concept of Industry 4.0, characterized by the broad integration of technologies and devices connected to the Internet (IoT). The profile required for new professionals who will enter this industry has also changed, requiring autonomy in learning, the ability to adapt to new developments and mastery of broad technologies. This brings new challenges to educational institutions, making the development of new teaching tools essential. This work proposes the development of a didactic application programming interface, based on ESP32 development modules, which can be used as an IoT teaching tool for undergraduate courses in Electrical Engineering. Through this virtual environment, the user will have access to examples, scripts and other teaching materials that will facilitate the development and testing of several different systems, being also able to change and monitor variables and peripheral configurations of ESP32 remotely.*

**Keywords:** *ESP32, Embedded, Laboratory, IOT*