



**FACULDADE DE TECNOLOGIA E CIÊNCIAS SOCIAIS APLICADAS – FATECS**  
**Engenharia de Computação**

Lucas Athayde Dutra  
21606533

**IOT COMO PROPOSTA DE SEGURANÇA RESIDENCIAL**

BRASÍLIA  
2022



Lucas Athayde Dutra

## **IOT COMO PROPOSTA DE SEGURANÇA RESIDENCIAL**

Trabalho de Conclusão de Curso (TCC) apresentado como um dos requisitos para a conclusão do curso de Engenharia de Computação do UniCEUB– Centro Universitário de Brasília

Orientador: **Prof. MSc Francisco Javier de Obaldia Diaz.**

BRASÍLIA  
2022



Lucas Athayde Dutra

## **IOT COMO PROPOSTA DE SEGURANÇA RESIDENCIAL**

Trabalho de Conclusão de Curso (TCC) apresentado como um dos requisitos para a conclusão do curso de Engenharia da Computação do UniCEUB – Centro Universitário de Brasília

Orientador: **Prof. MsC Francisco Javier de Obaldia Diaz.**

Brasília, 2022.

### **BANCA EXAMINADORA**

---

Prof. MsC Francisco Javier de Obaldia Diaz  
Orientador (a)

---

Prof. Sidney Cerqueira Bispo dos Santos  
Examinador (a)

---

Prof. Luís Cláudio Lopes de Araújo  
Examinador (a)

# IOT COMO PROPOSTA DE SEGURANÇA RESIDENCIAL

## IOT AS A HOME SECURITY PROPOSAL

Lucas Athayde Dutra<sup>1</sup>, Francisco Javier de Obaldia Diaz<sup>2</sup>, Sidney Cerqueira Bispo dos Santos<sup>3</sup>, Luís Cláudio Lopes de Araújo<sup>4</sup>

### RESUMO

Com o crescimento populacional, é evidente o surgimento e aumento de criminalidade em locais urbanos e residenciais e, com isso, tem-se uma preocupação cada vez maior com a segurança do patrimônio privado. Observa-se uma busca por meios e alternativas para contornar-se essa situação e, com o avanço da tecnologia e dos microcontroladores, é possível realizar tal ação por um valor economicamente acessível. Tendo isso em vista, neste projeto, será construído um sistema de segurança residencial, contendo módulos de câmera, sensor de movimento, sensor de campo magnético para monitorar a entrada de uma residência e um alarme que irá disparar em um determinado tempo. Isso será feito com a utilização de um Arduino e aplicativo Blynk, que mostrará em tempo real as informações dos sensores e do stream de vídeo.

**Palavras-chave:** IoT. Segurança residencial. Tecnologia.

**Abstract:** As the population grows, it's evident the increase and emergency of crimes happening in public and residential spaces, as these states we can note a higher concern with security on private property. As observed, we have seen a search for means and alternatives to circumvent this situation, with technology and microcontrollers improving over time, it's possible to make this as an economically accessible value. Having this in mind, we're building a home security project, containing camera, movement sensor, magnetic sensor, buzzer modules, in which we're going to monitor house entrances and make an alarm that will set off after a given period. This will all be done with the Arduino and Blynk app, which will show us real-time information about those sensors and video stream.

**Keywords:** IoT. Home Security. Technology.

---

<sup>1</sup> Graduando do curso de Engenharia da Computação pelo Centro Universitário de Brasília (UniCEUB). E-mail: lucas.ad97@hotmail.com.

<sup>2</sup> UniCEUB, orientador.

<sup>3</sup> UniCEUB, primeiro examinador.

<sup>4</sup> UniCEUB, segundo examinador.

## 1 INTRODUÇÃO

Com o avanço da humanidade e do crescimento populacional, é possível observar um aumento significativo da pobreza ao redor do globo, principalmente nos centros urbanos, já que muitos migram como uma tentativa de ascensão econômica e mudança de vida.

Decorrente desse processo, observa-se o surgimento da criminalidade, caracterizada por furtos, roubos, assaltos, homicídios, latrocínio, dentre outras formas de crimes a pessoas e patrimônios.

De acordo com a Secretaria de Estado de Segurança Pública do Distrito Federal (SSP-DF), em 2020 foram realizadas 370 ocorrências de roubo em residência, assim como 900 em comércios. Já em 2021 obteve-se uma pequena redução de casos nessas duas categorias (340 casos residenciais e 888 comerciais), ou seja, uma redução de 7% e 1,35%, respectivamente. (GOVERNO DO DISTRITO FEDERAL, 2021; GOVERNO DO DISTRITO FEDERAL, 2022)

Apesar da redução de casos em função do aumento de empresas privadas de segurança domiciliar, da tecnologia e da melhoria do sistema público de segurança, considera-se ainda ser necessária uma busca por alternativas de segurança residencial que sejam notavelmente mais econômicas.

É perceptível o avanço que a tecnologia teve nos últimos 20 anos. A lista é grande, mas pode-se citar tanto o surgimento e a popularização da Internet nos anos 90 e no começo dos anos 2000 quanto o envio do Rover Perseverance à Marte, em 2021. Com a evolução da tecnologia, foi-se criando componentes menores, mais rápidos e mais eficientes nesse processo, e hoje vemos o resultado disso pela quantidade de equipamentos disponíveis que estão interligados via rede e conversam um com o outro. Esse fenômeno de comunicação e interação entre dispositivos foi nomeado de IoT, abreviação de *Internet of Things*, em tradução livre para “Internet das Coisas”.

Com o avanço da tecnologia, a curiosidade e o desejo por automatizar e tornar mais simples os processos do dia a dia tornaram-se um desafio, fazendo com que as pessoas deste polo de desenvolvimento começassem a tentar novas coisas, principalmente com o surgimento da Internet, em 1989.

Em 1990, pouco tempo depois do surgimento da Internet, John Romkey criou o primeiro dispositivo IoT, uma torradeira que podia ser ligada e desligada através da Internet, abrindo um mundo de possibilidades.

O termo Internet das Coisas foi criado em 1999 por Kevin Ashton, em um desenvolvimento de tag RFID para controle de produtos em uma esteira de produção, e desde então o termo ficou sendo referido como a integração de equipamentos, utensílios e gadgets uns com os outros, podendo conversar entre si e operar de forma independente por meio da Internet. (VISION OF HUMANITY, [s.d.]; FIA UK, 2021)

Em 2010, tinha-se, estimadamente, 800 milhões de dispositivos IoT conectados à rede; em 2019, observou-se que esse valor saltou para 10 bilhões e se estima que até 2025 haja mais de 30 bilhões. (STATISTA, 2022)

Com esses dispositivos se estendendo para todas e quaisquer áreas, observa-se o surgimento do termo “Domótica”, que une os termos “Domus”, que significa casa, com “Robótica”, em referência à automatização e ao controle dos equipamentos dentro de uma residência.

Nesse sentido, tendo em vista as informações sobre segurança pública e violência no Distrito Federal, o avanço das tecnologias de IoT nos últimos 20 anos juntamente com os conhecimentos adquiridos no curso, tem-se uma proposta de unir esses fatores em uma solução de segurança residencial, a partir da disponibilização de informações em tempo real sobre o que acontece na residência do usuário, mesmo que à distância, por um valor acessível.

Neste projeto, utiliza-se a implementação

de um dispositivo Arduino ligado a diferentes componentes que avisam ao residente se sua porta está aberta ou fechada, notificando caso tenha sido esquecida aberta, assim como uma câmera conectada a um aplicativo e um sensor de movimento. O sistema contempla a programação adequada para esses eventos.

## 2 REVISÃO BIBLIOGRÁFICA

### 2.1 Placa Arduino UNO

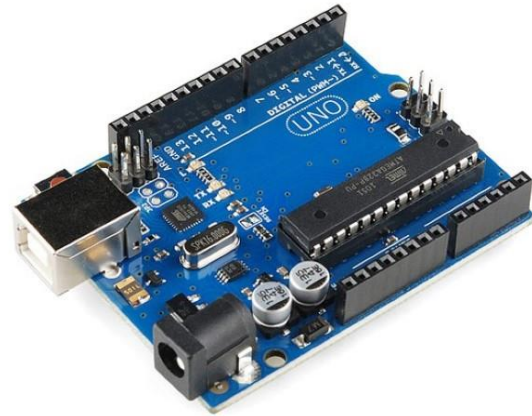
Arduino é uma plataforma de dispositivos eletrônicos open-source de baixo custo, sendo utilizada para uma vasta gama de prototipagem, de projetos simples até mais avançados e integrados.

Baseada no chip ATmega328P, o Arduino UNO é a placa mais robusta e qualificada para iniciantes no meio dos códigos e projetos que não demandam altas cargas de processamento e desempenho.

Utilizando Arduino Programming Language, baseada em Wiring, para código e execução de comandos compilados, o Arduino permite criar uma grande variedade de projetos e protótipos. (ARDUINO, c2022a; BIT DEGREE, 2021)

Essa plataforma possui 14 entradas/saídas digitais, sendo seis destas PWM, e outras seis entradas analógicas, totalizando 20 portas, um ressonador de cerâmica de 16 Mhz, uma memória flash de 32 KB, conexão USB e uma entrada de energia<sup>3</sup>, conforme mostra a Figura 1.

Figura 1. Arduino UNO.



Fonte: FilipeFlop (2022).

### 2.2 Módulo WI-FI ESP8266 ESP-01

O ESP8266 ESP-01 é um módulo WiFi de baixo custo, permitindo acoplar-se a qualquer microcontrolador e dar acesso à rede WiFi, utilizando o protocolo TCP/IP.

Possui um microprocessador L106 de 32-bit e uma memória interna, podendo, assim, salvar e executar funções de rede de forma independente. (WIA.IO, [s.d.]

Por ser de baixo custo, fácil utilização e alta compatibilidade com demais dispositivos, este módulo será utilizado para garantir que o Arduino possa se comunicar com a rede sem fio.

### 2.3 Sensor PIR HC-SR501

O Sensor Passive Infrared (PIR) HC-SR501 é um dispositivo de baixo custo que detecta a movimentação emitida pelo calor do ambiente em seu campo de visão. É composto por dois sensores piroelétricos, informando quando há uma diferença de

<sup>3</sup> Informações retiradas de Arduino.cc. Disponível em: <https://docs.arduino.cc/hardware/uno-rev3>. Acesso em: 29 mai. 2022.

deteção entre ambos. Pode operar entre 5V e 20V e é composto por três pinos de contato, sendo estes o GND, VCC e INPUT. (COMPONENTS 101, 2021)

Possui dois modos de operação definidos pelos pinos localizados na parte inferior do sensor, sendo um deles o modo “L”, que informa se houve um movimento em seu campo de visão e se desativa depois de um tempo, e o modo “H”, que se mantém ativo enquanto detectar movimento em seu campo de visão, desligando assim que para de receber essa informação. (Ibidem)

O PIR tem dois reguladores na frente, sendo um deles para a sensibilidade de movimento, presente na função “H”, e o outro de delay, para a função “L”. A Figura 2 apresenta o sensor PIR HC-SR501.

Figura 2. Sensor PIR.

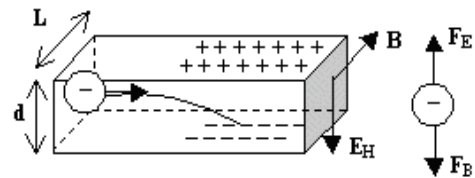


Fonte: Components 101 (2021).

## 2.4 Sensor de efeito Hall KY-003

O módulo KY-003 é um sensor de efeito Hall que, ao detectar a presença de um campo magnético no sensor, faz com que os elétrons se acumulem em uma das faces do dispositivo, permitindo detectar uma tensão de saída. A Figura 3 mostra um exemplo de funcionamento do sensor. (ENCYCLOPEDIA BRITANNICA, c2022)

Figura 3. Funcionamento dentro do sensor Hall.



Fonte: If.Ufrgs, [s.d].

Esse sensor é comumente utilizado em carros e dispositivos em que é necessário ter mais precisão da rotação, orientação e intensidade dos campos magnéticos no sistema, podendo enviar sinais analógicos ou digitais.

Para o projeto, foi utilizado o sensor de polaridade de campo magnético juntamente a um ímã, que notifica se a porta se encontra aberta ou fechada. Na Figura 4, está presente o módulo do sensor KY-003.

Figura 4. Sensor Hall KY-003.



Fonte: Arduino Modules (2021).

## 2.5 Buzzer passivo

Módulo buzzer passivo é utilizado em placas-mãe, emitindo som quando o circuito é fechado. A Figura 5 apresenta o buzzer e os

seus cabos positivo e negativo. Ele será utilizado como um alarme dependendo do estado dos outros módulos.

Figura 5. Buzzer de placa-mãe.



Fonte: Components101 (2017).

## 2.6 ESP32-CAM

O módulo ESP32-CAM é um microcontrolador ESP32 integrado com câmera OV2640, bluetooth, WiFi, cartão de memória microSD e algumas funcionalidades de software internas, incluindo reconhecimento facial, captura de fotos e streaming de vídeo (DRONEBOT WORKSHOP, 2020). A Figura 6 mostra o módulo ESP32-CAM isolado, com a câmera OV2640 ao lado direito.

Por ser um microcontrolador isolado, necessita-se de um adaptador *Future Technology Devices International*, fabricante do adaptador (FTDI) para que se possa configurar e inserir o código no módulo, fazendo com que se possa operar de forma indireta com os demais dispositivos.

Esse módulo opera em 5V, tornando-o altamente compatível com dispositivos IoT e projetos *Do it Yourself* (DIY).

Figura 6. ESP32-CAM com câmera OV2640.



Fonte: Dfrobot, [s.d.].

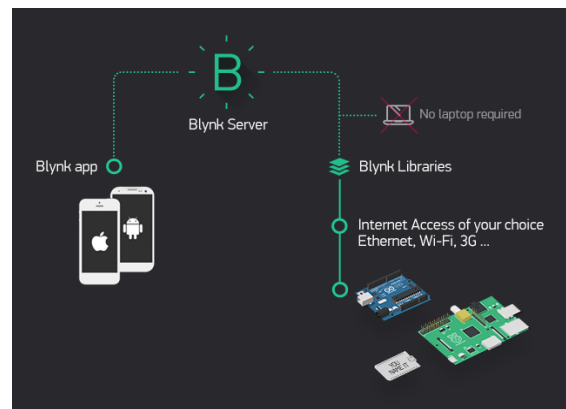
## 2.7 Blynk

Blynk é uma aplicação móvel que permite integrar dispositivos IoT com smartphones, fazendo com que se possa monitorar diferentes tipos de sensores, operar e alterar o estado de diferentes dispositivos pelo próprio celular. (TECH EXPLORATIONS, [s.d])

Possui dois tipos de planos, sendo um plano pago, voltado para grandes empresas, e outro livre, para pequenos projetos IoT.

Será utilizado o Blynk para fazer uma aplicação Mobile que possa integrar e dar feedback dos dispositivos ligados à rede pelo Arduino.

Figura 7. Esquema de conexão do App Blynk.



Fonte: Docs.blynk, [s.d.].



Os conceitos de tecnologia, rede e lógica de programação abordados na Figura 7 serão aplicados no projeto, conforme será visto a seguir.

### 3 METODOLOGIA DO TRABALHO

Tem-se como objetivo deste projeto apresentar uma solução acessível para as questões de segurança residencial por meio da montagem de um sistema contendo sensor de porta, de movimento, buzzer, câmera de segurança e um aplicativo para celular. Neste sentido, as análises de implementação do sistema foram feitas à luz da pesquisa aplicada ao problema de roubos e invasões a residências privadas, sendo abordadas e analisadas sob um olhar qualitativo, visando a observar as funcionalidades, descrever e analisar a aplicação dos sistemas Arduino, ESP32, ESP8266 e Blynk.

Quanto aos objetivos, o projeto fundamentou-se em uma pesquisa descritiva baseada em procedimentos técnicos, recorrendo a informações técnicas que constam em sites e documentações, demonstrando de que forma se dá a montagem e implementação dos módulos em um ambiente.

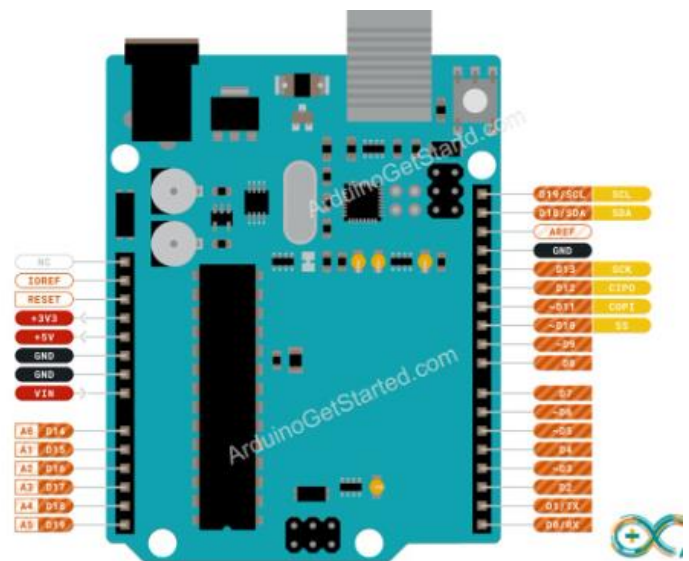
A montagem do sistema consiste em quatro etapas fundamentais para que o projeto tenha êxito em seu funcionamento que serão divididas em: configuração do Arduino, configuração dos módulos, configuração do ESP32-CAM e, por último, a integração com o aplicativo Blynk.

#### 3.1 Arduino, buzzer e KY-003

Em primeira instância, deve-se ligar o Arduino ao computador para executar o código que será escrito. O Arduino UNO possui 20 portas lógicas de input, sendo 14 dessas de input Digital contendo seis que suportam PWM (Pulse-Width Modulation), e outras seis portas de input Analógico. (ARDUINO GET STARTED, c2021). Para

esta implementação, serão utilizadas apenas as portas digitais. A Figura 8 mostra o esquema físico do Arduino Uno, apresentando seus pinos e as características de cada um deles.

Figura 8. Arduino Uno Pinout.



Fonte: Arduino Get Started (c2021).

É feita a implementação com o sensor KY-003, do LED vermelho e um buzzer. Com isso, é possível ter um breve teste de funcionalidade dos módulos em questão.

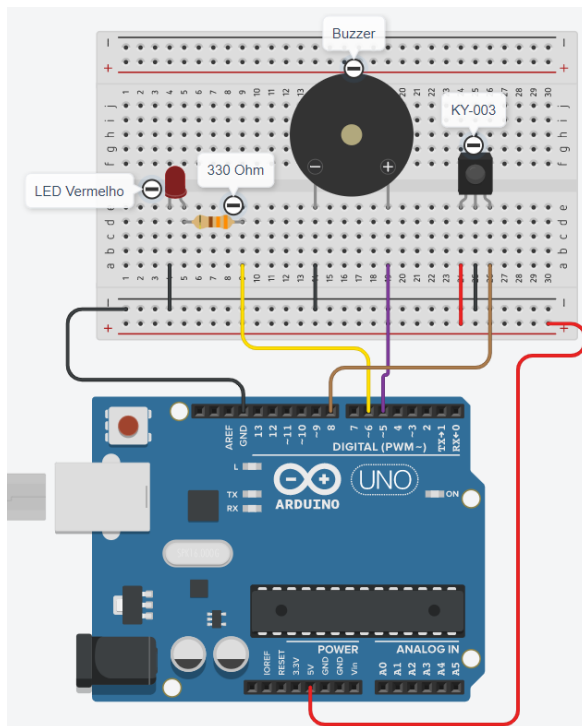
Por questão de praticidade, liga-se o GND do Arduino na fileira com sinal negativo do protoboard, podendo, então, aterrar toda a fileira e facilitar as conexões.

Insere-se o diodo LED vermelho no protoboard, que tem, por padrão, a perna mais curta representando o polo negativo e a perna mais longa como positivo. Então, liga-se a perna mais curta do diodo ao GND e a mais longa em série com um resistor de 330 Ohms, conectando, em seguida, a saída do resistor à porta D6 do Arduino.

Posteriormente, é feita a conexão do buzzer, que, por padrão, possui um pino positivo, indicado na parte superior por um sinal de +, e outro negativo. Conecta-se o pino negativo ao GND e o positivo, com um jumper, à porta D5 do Arduino.

Por último, liga-se o sensor Hall KY-003 ao 5V, GND, e a porta D8 no Arduino. Por operar na tensão do Arduino, não há necessidade de ligar em série a um resistor. A Figura 9 mostra como ficará o circuito em questão.

Figura 9. Circuito montado com LED, buzzer e sensor de porta Hall KY-003 ligados ao Arduino.



Fonte: Adaptado de TinkerCAD (2022).

Com o circuito montado, será implementado um código que acenderá o LED assim que KY-003 não estiver em contato, acionando um contador que irá verificar seu estado a todo momento. Uma vez não identificada alteração por N segundos, o buzzer será acionado, dando uma resposta tanto visual como audível.

### 3.2 ESP32-CAM

O módulo ESP32-CAM é uma placa com microprocessador ESP32 e uma câmera embutida, tornando-o um dispositivo bem eficiente e de baixo custo.

Este é utilizado em conjunto com a sua

ESP32-CAM-MB, que nada mais é que um adaptador micro-USB do módulo, fazendo com que se possa configurá-lo sem a necessidade de um módulo FTDI.

A câmera funcionará de forma independente dos demais módulos. Para isso, é necessário implementar código dentro do seu firmware e definir uma conexão para que seja posteriormente capturada pelo aplicativo Blynk.

Figura 10. ESP32-CAM à direita, ESP32-CAM-MB à esquerda e câmera OV2640 abaixo.



Fonte: Random Nerd Tutorials, c2013-2022c.

#### 3.2.1 Instalando driver do chipset CH340G

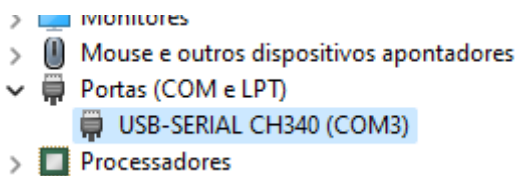
Para fazer a ESP32-CAM-MB ser reconhecida pelo computador como um dispositivo USB, deve-se instalar o driver do seu chipset embutido. Uma vez realizada a instalação, o Arduino IDE reconhece o dispositivo e possibilita implementar código em seu Firmware.

É possível encontrar os drivers do chipset

no site do fabricante WCH-IC. Neste<sup>4</sup>, encontram-se os drivers para os chipsets de prefixo CH340 e CH341.

Após baixar e instalar o driver do chipset, pode ser verificado se ele é reconhecido pelo computador por meio do Gerenciador de Dispositivos do Windows (Figura 11).

Figura 11. Gerenciador de Dispositivos do Windows.

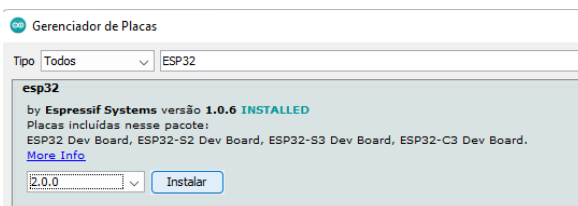


Fonte: Elaborado pelo autor (2022).

Em seguida, para garantir que o código seja enviado de forma correta para o dispositivo, deve-se instalar a biblioteca das placas ESP32 do Arduino IDE para que se possa implementar o código em seu firmware.

Para isso, vá em Ferramentas → Placas → Gerenciador de Placas no Arduino IDE. Neste menu, pesquise por ESP32 e instale a versão 2.0.0 da biblioteca (Figura 12).

Figura 12. Biblioteca ESP32 no Gerenciador de Placas.



Fonte: Elaborado pelo autor (2022).

Selecionando a placa Ai-Thinker como dispositivo, pode-se, então, ligar o módulo na entrada USB do computador e colocá-lo em funcionamento.

### 3.2.2 Implementando código de transmissão

Para que se possa tornar o dispositivo operante e visualizar um stream de vídeo pelo aplicativo, deve-se configurar um IP estático para a transmissão.

Utilizando o exemplo de CameraWebServer presente dentro do Arduino IDE, altera-se o código para que seja utilizado um IP fixo. (RANDOM NERD TUTORIALS, c2013-2022a.; OLINTO, 2020)

Com isso feito, é necessário compilar e carregar o código no firmware do dispositivo. Ao clicar em “carregar”, deve-se colocar o dispositivo em modo Flash, segurando o botão IO0 e apertando o botão RST em seguida (Figura 13).

Figura 13. Botões IO0 e RST no ESP32-CAM.



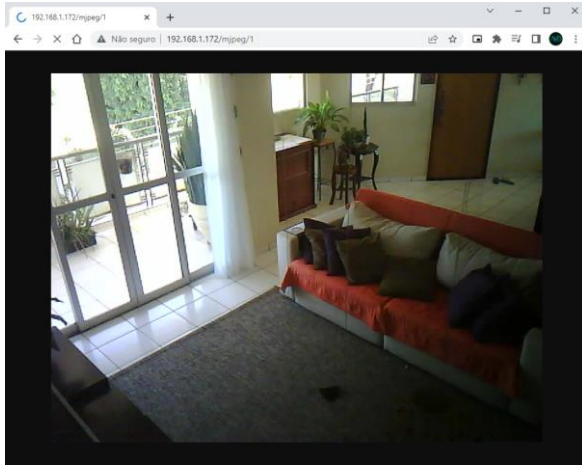
Fonte: Adaptado de Random Nerd Tutorials (c2013-2022d).

Deve-se receber uma confirmação no console de que o código foi enviado ao módulo.

Para executar a operação gravada dentro do módulo, deve-se apertar novamente o botão “RST”.

<sup>4</sup> Drivers disponíveis em: [http://www.wch-ic.com/downloads/CH341SER\\_ZIP.html](http://www.wch-ic.com/downloads/CH341SER_ZIP.html). Acesso em: 2 jun. 2022.

Figura 14. Webserver com extensão MJPEG rodando no IP definido.



Fonte: Elaborado pelo autor (2022).

Com isso, encerra-se a configuração do módulo ESP32-CAM e pode-se acessar o IP de streaming dentro da rede, como pode-se observar pela Figura 14 (RANDOM NERD TUTORIALS, c2013-2022c).

### 3.3 ESP8266 ESP-01S

A fim de que seja possível enviar informações dos módulos para a rede e receber em outra plataforma, deve-se instalar o módulo de Rede ESP8266 ESP-01S no Arduino. Para isso, será utilizado o adaptador USB e o adaptador de protoboard para uma melhor configuração e implementação do módulo ao projeto.

Em primeiro lugar, deve-se incluir a biblioteca do ESP8266 a IDE do Arduino e, em seguida, acoplar o módulo ao seu adaptador USB, conforme a Figura 15.

Figura 15. ESP-01 acoplado ao adaptador USB.



Fonte: Elaborado pelo autor (2022).

Para que se possa implementar código dentro do dispositivo, deve-se inseri-lo na porta USB do computador com o botão localizado na parte inferior do adaptador pressionado, que nada mais é que um botão fechando curto entre o GND e GPIO. Assim, é possível entrar no modo Boot e enviar código para o dispositivo. Na figura 16, é possível ver o botão localizado na parte inferior.

Figura 16. Botão localizado no adaptador USB do ESP-01.



Fonte: Elaborado pelo autor (2022).

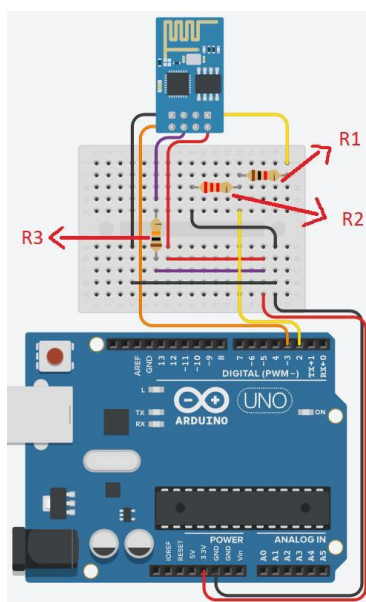


Para que o dispositivo funcione como uma ponte Wi-Fi entre o Arduino e o aplicativo, deve-se fazer a atualização de Firmware para que ele receba comandos AT (ATtention) e possa encaminhar as informações entre os dispositivos.

Após a atualização do Firmware, conecta-se o ESP-01 a uma placa Breakout para que se encaixe na protoboard e se possa conectar ao Arduino com mais praticidade.

O módulo ESP-01 opera em 3.3V, logo será ligado o GND no GND da placa, o CH\_PD a um resistor de 10K(R3) Ohm e, então, a corrente de energia 3.3V, TX, ao pino D3 do Arduino, RX a um divisor de tensão feita por um resistor de 1K Ohm(R1) e 2,2K Ohm(R2) e, depois, ao pino D2, e, por fim, o VCC diretamente ao 3.3V no Arduino. A Figura 17 mostra a montagem do ESP-01S ao Arduino. (O tutorial seguido consta no site Blog Master Walker<sup>5</sup>)

Figura 17. Montagem do ESP-01 ao Arduino.



Fonte: Adaptado de TinkerCAD (2022).

Dessa forma, finaliza-se a configuração e

<sup>5</sup> Disponível em: <https://blogmasterwalkershop.com.br/embarcados/esp8266/upgrade-de-firmware-do-wifi-esp8266-esp-01->

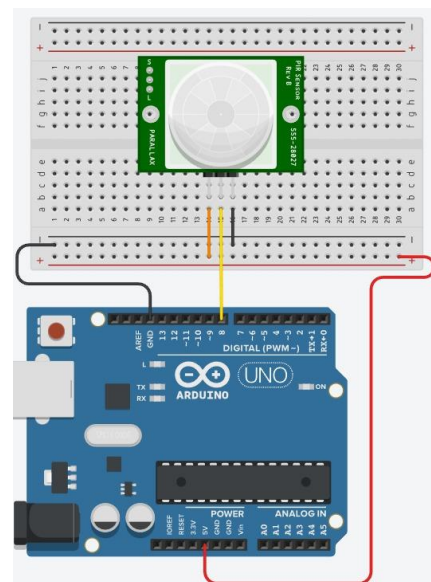
implementação do módulo ESP-01 ao Arduino e consegue-se enviar as informações para o App.

### 3.4 Sensor de movimento PIR HC-SR501

O sensor PIR HC-SR501 apresenta esquema de ligação bem simples, sendo um pino para o envio de informações para o Arduino e os outros dois para energizar o dispositivo. (COMPONENTS 101, 2021)

Liga-se o pino central à porta D8 do Arduino, o VCC ao 5V e GND no GND. Coloca-se o jumper amarelo na parte traseira do HC-SR501 no modo “H”, para que, sempre que seja detectado movimento, se tenha o envio do sinal. Com isso feito, ao programar o código no Arduino IDE, o sensor retornará HIGH ao receber qualquer movimento em seu campo de visão, e LOW caso contrário. A Figura 18 apresenta o circuito de montagem do PIR HC-SR501.

Figura 18. Montagem do PIR HC-SR501 ao Arduino.



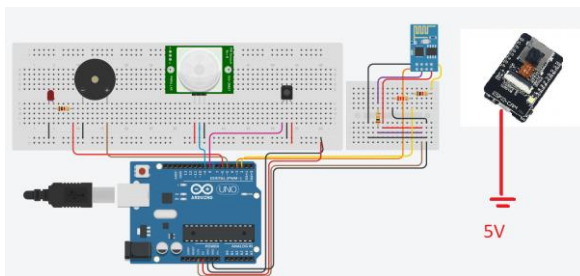
Fonte: Adaptado de TinkerCAD (2022).

atraves-do-arduino-e-conversor-usb-serial. Acesso em: 25 mai. 2022.

### 3.5 Circuito completo

Com todos os componentes ligados ao Arduino, o sistema ficará completo e funcional, podendo, então, colocar em prática a aplicação. Na figura 19 vemos o circuito completo.

Figura 19. Circuito completo montado.



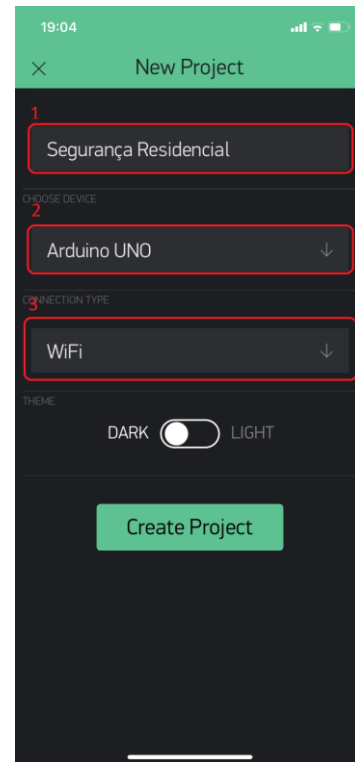
Fonte: Adaptado de TinkerCAD (2022).

### 3.6 Blynk

Com os dispositivos configurados e conectados na rede, deve-se configurar o App para que ele possa mostrar as informações recebidas e se comportar de acordo.

Deve-se criar um novo projeto no App, preenchendo o nome do projeto no campo indicado pelo número 1, o dispositivo que será feita a conexão no campo 2, e o tipo de conexão efetivamente no campo de número 3. A Figura 20 mostra os campos presentes a serem preenchidos para que se possa criar um projeto no app Blynk.

Figura 20. Criação do projeto no Blynk.



Fonte: Elaborado pelo autor (2022).

O App apresenta uma diversidade de botões, painéis, gauges, sliders, podendo informar ao usuário em tempo real o estado de seus dispositivos e sensores.

Neste projeto, serão utilizados os widgets de streaming de vídeo, display LCD 16x2, botões que alternam entre estados e notificação de PUSH para o dispositivo.

## 4 APRESENTAÇÃO E ANÁLISE DOS RESULTADOS

Após a montagem e codificação completa do sistema, foi posto em prática sua usabilidade, posicionando o sistema onde o sensor KY-003 tivesse alcance à porta, o PIR HC-SR501 direcionado para a sala de estar, e a câmera no canto superior esquerdo da sala, com visibilidade da porta de entrada e varanda.

#### 4.1 Montagem e configuração da ESP 32-CAM

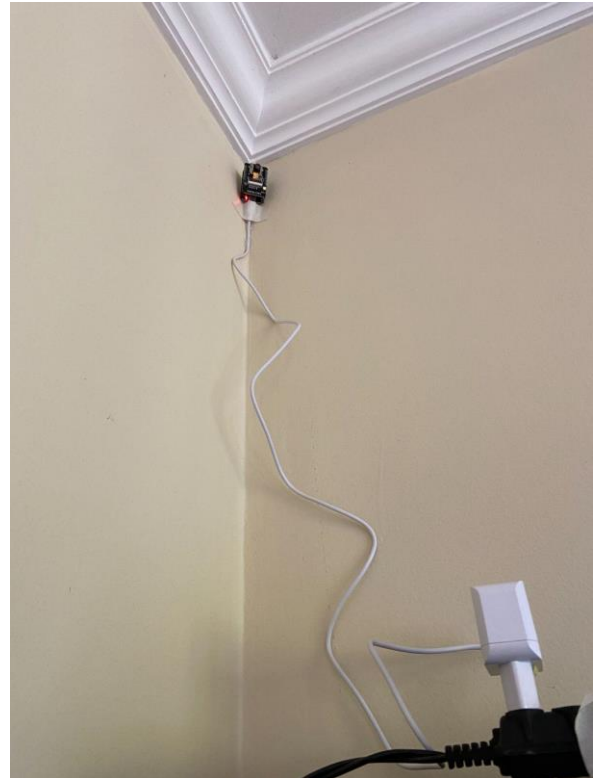
Para a configuração da câmera ESP32-CAM, foi utilizado um cabo USB. Nota-se que, para que seja feito o upload do código, é necessário um cabo que realize transferência de dados, e há cabos que não realizam tal ação.

Encontrou-se um obstáculo ao tentar realizar o upload de código para o dispositivo, em que era necessário curto-circuitar os pinos GPIO e GND e apertar o botão RST presente na placa em um momento específico em que se é feito o upload do código para o firmware do dispositivo. Após entendimento de seu funcionamento, a ação foi realizada com sucesso.

Ao configurar o IP e realizar o acesso ao endereço de transmissão, notou-se uma qualidade de imagem bem inferior à que se havia visto anteriormente nos testes. Após estudo e pesquisa, foi realizada a troca do FRAME\_SIZE de UXGA para SVGA, apresentando menor latência e uma qualidade de transmissão muito mais nítida. (RANDOM NERD TUTORIALS, c2013-2022b)

Depois das correções, a câmera foi ligada a uma fonte de 5V e um cabo USB e montada em uma posição com visibilidade ampla do ambiente, permitindo uma transmissão de vídeo pelo aplicativo pelo IP que foi configurado anteriormente no ESP32-CAM. (HOW TO ELECTRONICS, 2021)

Figura 21. Local de Montagem do módulo ESP32-CAM.



Fonte: Elaborado pelo autor (2022).

#### 4.2 ESP8266 ESP-01

Em primeira instância, tentou-se o upload de firmware do módulo WiFi ESP-01 através do Arduino, com este último servindo de ponte para o computador ao fechar curto entre GND e RESET. Contudo, após seguir instruções e documentações de conectividade com o Arduino<sup>6</sup>, não se obteve êxito, fazendo-se necessário adquirir um adaptador USB para realizar este passo.

Um outro problema encontrado com o dispositivo deu-se pela sua pouca utilização em conjunto com o Arduino no cenário atual, fazendo com que as bibliotecas mais recentes do módulo não apresentassem as devidas referências, sendo necessário baixar versões anteriores.

<sup>6</sup> Disponível em:  
<https://blogmasterwalkershop.com.br/embarcados/esp8266/upgrade-de-firmware-do-wifi-esp8266-esp-01->

[atraves-do-arduino-e-conversor-usb-serial](#). Acesso em: 25 mai. 2022.

Após entendimento e upload do firmware, havia uma divergência em várias documentações e tutoriais na Internet sobre qual posição se deveria ligar os pinos RX e TX, fazendo com que não fosse obtido êxito no processo de fazer o dispositivo se conectar ao WiFi. Realizando testes, ao inverter os pinos de RX e TX, o dispositivo conectou-se à rede com sucesso e executou a transmissão dos dados, eliminando demais impeditivos de comunicação via WiFi.

#### 4.3 Sensor PIR HC-SC501

A implementação do sensor HC-SC501 foi realizada de forma bem simples, sem grandes obstáculos nem divergências de resultados.

Em um primeiro momento durante sua implantação no local, o sensor estava bem instável, avisando continuamente pelo aplicativo que havia movimento.

O sensor possui ajustes de sensibilidade para seus diferentes modos de operação. Logo, após regular a sensibilidade de detecção, o sensor apresentou um funcionamento mais preciso de movimentação no ambiente.

Encontrou-se um problema no segundo dia de aplicação, em que o sensor retornava apenas o sinal HIGH. Após verificação, foi identificado que o jumper conectado ao VCC apresentava mau contato. Realizando a troca, o problema foi resolvido e o dispositivo voltou a operar normalmente.

#### 4.4 Sensor Hall KY-003, buzzer e LED.

A implementação do sensor KY-003 foi bem-sucedida e não apresentou nenhuma falha no processo, sendo um módulo de grande robustez, recebendo e transmitindo dados sem nenhuma falha em seus momentos de teste e usabilidade, detectando sempre que o ímã se aproximava e era afastado do sensor. O KY-003 foi colocado na parte superior da moldura, com o ímã grudado por uma fita na porta em si, fazendo com que ela se afastasse do sensor assim que fosse aberta, e se

aproximasse quando fechasse.

Para que houvesse um feedback visual e audível na forma de um alarme de porta, foi programado para que, sempre que a porta fosse aberta, o LED vermelho ligasse, informando que a porta se encontrava aberta, e desligasse, caso contrário.

Além disso, caso a porta fique aberta por um tempo superior ao programado, o buzzer dispara na frequência de 2000Hz contínuo, até que a porta venha a fechar novamente. A Figura 21 mostra como o sistema foi implementado.

Figura 22. Buzzer, KY-003 e LED vermelho montados ao lado da porta.



Fonte: Elaborado pelo autor (2022).

#### 4.5 Usabilidade do Blynk

Após todos os sistemas terem sido postos em prática e o ambiente no Blynk estabelecido com seus devidos Widgets, conseguiu-se monitorar com facilidade as informações dos sensores, com os estados



atuais sendo demonstrados via display LCD e botões, assim como o stream da câmera. Caso sejam detectados abertura da porta e movimento no ambiente, é enviada uma notificação via PUSH para o celular do usuário. Na Figura 22, tem-se o dashboard com todos os widgets utilizados.

Figura 23. Dashboard do Blynk App com Widgets.



Fonte: Elaborado pelo autor (2022).

## 5 CONSIDERAÇÕES FINAIS

Teve-se como objetivo do projeto realizar um sistema de segurança residencial que pudesse informar em tempo real ao usuário os estados dos sensores e as suas alterações, assim como um stream de vídeo pela rede, utilizando um Arduino e módulos pela sua praticidade e facilidade de uso. Os

sensores/módulos em se mostraram-se bem robustos, não apresentando falhas além das mencionadas durante o período de implementação e usabilidade. O projeto demonstrou ser uma solução financeiramente acessível, como pode ser observado pelos valores presentes na Tabela 1.

Tabela 1. Valores de componentes utilizados no projeto.

Material	Quantidade	Valor
Arduino	1	R\$ 59,90
Módulo PIR HC-SR501	1	R\$ 8,91
Sensor KY-003 de Efeito Hall	1	R\$ 4,41
ESP8266 ESP-01 com adaptadores	1	R\$ 48,49
ESP32-CAM com ESP32-CAM-MB	1	R\$ 89,99
Protoboard 830 pontos	1	R\$ 17,55
Protoboard Mini 170 Pontos	2	R\$ 9,46
Fonte 5V	2	R\$ 17,90
Ímã de Neodímio	1	R\$ 4,41
Jumpers	1	R\$ 34,74
<b>Total</b>		<b>R\$ 295,76</b>

Fonte: Elaborado pelo autor (2022).

No que se refere à utilização de módulos, o módulo ESP32-CAM e ESP-01 apresentaram um desempenho excelente, tendo apenas pequenas travas devido à instabilidade de conexão com a rede WiFi 2.4Ghz e pequenos delays para transmissão de informações para o aplicativo Blynk.

Durante o levantamento de equipamentos, houve uma dúvida quanto a que sensor utilizar na porta. No final das contas, o sensor

Hall KY-003 foi uma escolha indispensável, entregando respostas de forma rápida e eficiente para as alterações de estado.

O sensor HC-SC501 é módulo simples e de grande utilidade e, apesar dos problemas de sensibilidade, entregou o que foi esperado.

No entanto, apesar do êxito de implementação e seu valor relativamente baixo, foram identificados como pontos de possível melhoria uma redução de valor aderindo a outros módulos, como o NodeMCU ou um ESP8266.

O módulo ESP32-CAM possui uma câmera RGB, sendo pouco sensível à ausência de luz, tornando a visibilidade no escuro praticamente nula. Uma forma de contornar isso seria abrir a câmera e remover seu filtro infravermelho. Outra alternativa seria trocar a câmera por uma de modelo OV7725, que possui configurações de maior sensibilidade e exposição, tornando-o mais eficiente em ambientes com pouca ou nenhuma luz.

Outro ponto importante seria o acesso ao stream de vídeo. Para que se possa acessar de fora do ambiente em que o dispositivo está conectado, faz-se necessário abrir a porta 80 no roteador, que vem bloqueada por padrão pelos provedores de Internet.

Há um limite de widgets que se pode utilizar no Blynk, podendo ser expandido apenas por planos de pagamento, impedindo que o projeto fosse escalado em funcionalidades.

Tendo isso em vista, o desenvolvimento de um aplicativo independente para receber os dados do Arduino é uma excelente alternativa, saindo da limitação de planos pagos e documentações obsoletas.

A utilização de um RTC (Real time clock) para notificar o horário que foi realizada abertura da porta e para detecção de movimentos, assim como um banco de dados para armazenamento dessas informações deixariam o projeto muito mais redondo para seu propósito.

## AGRADECIMENTOS

Meus agradecimentos a minha mãe, a qual me auxiliou e apoiou durante toda a extensão do curso e me permitiu implementar o projeto em sua residência.

Aos professores Ivandro e William Malvezzi, do curso de Engenharia da Computação do UniCEUB, assim como aos amigos que realizei durante a graduação: Sávio, Michel, Arthur Alecsander e Arthur Cardoso, obrigado pelo auxílio e companheirismo.

A minha querida amiga Laura, que me auxiliou com dicas valiosas na elaboração deste projeto.

E por último, e não menos importante, ao professor Javier, pelo seu conhecimento, domínio e auxílio durante todo o desenvolvimento do projeto.

## REFERÊNCIAS

A history of the Internet of Things. **Fia.uk**, 2021. Disponível em: <https://www.fia.uk.com/news/history-of-iot.html>. Acesso em: 16 mai. 2022.

ACTIVE Passive Buzzer. **Components 101**, 2017. Disponível em: <https://components101.com/misc/buzzer-pinout-working-datasheet>. Acesso em: 30 mai. 2022.

ARDUINO Uno Pinout? **Arduino Get Started**, c2021. Disponível em: <https://arduinogetstarted.com/faq/arduino-uno-pinout>. Acesso em: 25 mai. 2022.

AUGUSTYN, A. Hall effect. **Encyclopedia Britannica**, c2022. Disponível em: <https://www.britannica.com/science/Hall-effect>. Acesso em: 25 mai. 2022.

CH341SER.ZIP. **Wch.ic**, c2002-2022. Disponível em: [http://www.wch-ic.com/downloads/CH341SER\\_ZIP.html](http://www.wch-ic.com/downloads/CH341SER_ZIP.html). Acesso em: 30 mai. 2022.

DIY Surveillance CCTV Camera using ESP32 CAM & Blynk. **How To Electronics**, 2021. Disponível em: <https://how2electronics.com/diy-surveillance-cctv->

camera-using-esp32-cam-blynk/. Acesso em: 11 jun. 2022.

DOCUMENTATION for Blynk, the most popular IoT platform for businesses. **Docs.blynk**, [s.d.]. Disponível em: <https://docs.blynk.cc/>. Acesso em: 10 jun. 2022.

ESP32-CAM Development Board. **Dfrobot**, [s.d.]. Disponível em: <https://www.dfrobot.com/product-1879.html>. Acesso em: 30 mai. 2022.

ESP32-CAM - Getting Started & Solving Common Problems. **DroneBot Workshop**, 2020. Disponível em: <https://dronebotworkshop.com/esp32-cam-intro/>. Acesso em: 30 mai. 2022.

ESP32 Static/Fixed IP Address. **Random Nerd Tutorials**, c2013-2022a. Disponível em: <https://randomnerdtutorials.com/esp32-static-fixed-ip-address-arduino-ide/>. Acesso em: 10 jun. 2022.

ESP32-CAM Troubleshooting Guide: Most Common Problems Fixed. **Random Nerd Tutorials**, c2013-2022b. Disponível em: <https://randomnerdtutorials.com/esp32-cam-troubleshooting-guide/>. Acesso em: 10 jun. 2022.

ESPRESSIF ESP8266. **Wia.io**, [s.d.]. Disponível em: <https://www.wia.io/things/esp8266>. Acesso em: 29 mai. 2022.

GOVERNO DO DISTRITO FEDERAL. Secretaria de Estado da Segurança Pública. **Balanco Criminal Distrito Federal – Comparativo Mensal 2020 – Por Natureza**. Subsecretaria de Gestão da Informação, 2021. Disponível em: <http://www.ssp.df.gov.br/wp-content/uploads/2017/11/DISTRITO-FEDERAL-1.pdf>. Acesso em: 9 mai. 2022

GOVERNO DO DISTRITO FEDERAL. Secretaria de Estado da Segurança Pública. **Balanco Criminal Distrito Federal – Comparativo Mensal 2021 – Por Natureza**. Subsecretaria de Gestão da Informação, 2022. Disponível em: <http://www.ssp.df.gov.br/wp-content/uploads/2017/11/DF-Balanco-criminal-2021.pdf>. Acesso em: 9 mai. 2022

HOW to Program /Upload Code to ESP32-CAM AI-Thinker (Arduino IDE). **Random Nerd Tutorials**, c2013-2022c. Disponível em: <https://randomnerdtutorials.com/program-upload-code-esp32-cam>. Acesso em: 7 jun. 2022.

HC-SR501 PIR Sensor. **Components 101**, 2021. Disponível em: <https://components101.com/sensors/hc-sr501-pir-sensor>. Acesso em: 25 mai. 2022.

INTRODUCTION. **Arduino**, c2022a. Disponível em: <https://www.arduino.cc/en/Guide/Introduction>. Acesso em: 25 mai. 2022.

IOT Technologies Explained: History, Examples, Risks & Future. **Vision of Humanity**, [s.d.]. Disponível em: <https://www.visionofhumanity.org/what-is-the-internet-of-things/>. Acesso em: 16 mai. 2022.

KY-003 Hall Magnetic Sensor Module. **Arduino Modules**, 2021. Disponível em: <https://arduinomodules.info/ky-003-hall-magnetic-sensor-module/>. Acesso em: 1 jun. 2022.

M, L. What is Arduino Language: Coding for Arduino Boards Explained. **Bit Degree**, 2021. Disponível em: <https://www.bitdegree.org/tutorials/what-is-arduino-language-coding-for-arduino-boards-explained/>. Acesso em: 29 mai. 2022.

O efeito hall. **If.Ufrgs**, [s.d.]. Disponível em: [https://www.if.ufrgs.br/tex/fis142/mod08/m\\_s03.html](https://www.if.ufrgs.br/tex/fis142/mod08/m_s03.html). Acesso em: 30 mai. 2022.

O que é Arduino, para que serve e primeiros passos. **FilipeFlop**, 2022. Disponível em: <https://www.filipeflop.com/blog/o-que-e-arduino/>. Acesso em: 29 mai. 2022.

OLINTO, G. R. F ESP32 CAM dicas: IP FIXO, Gateway, mascara e DNS. Como alterar porta STREAM/81 (video). **Lab de garagem**, 2020. Disponível em: <https://labdegaragem.com/profiles/blogs/esp32-cam-dicas-ip-fixo-gateway-mascara-e-dns-como-alterar-porta>. Acesso em: 10 jun. 2022.

OLIVEIRA, E. Upgrade de Firmware do WiFi ESP8266 ESP-01 através do Arduino e Conversor USB Serial. **Blog master walker shop**, [s.d.]. Disponível em: <https://blogmasterwalkershop.com.br/embarcados/esp8266/upgrade-de-firmware-do-wifi-esp8266-esp-01-atraves-do-arduino-e-conversor-usb-serial>. Acesso em: 25 mai. 2022.

UNO R3. **Arduino.cc**, c2022b. Disponível em: <https://docs.arduino.cc/hardware/uno-rev3>. Acesso em: 29 mai. 2022.

UPLOAD Code to ESP32-CAM using ESP32-CAM-MB USB. **Random Nerd Tutorials**, c2013-2022d. Disponível em: <https://randomnerdtutorials.com/upload-code-esp32-cam-mb-usb/>. Acesso em: 7 jun. 2022.

VAILSHERY, L. S. Global IoT and non-IoT connections 2010-2025. **Statista**, 2021. Disponível em: <https://www.statista.com/statistics/1101442/iot-number-of-connected-devices-worldwide/>. Acesso em: 16 mai. 2022.

WHAT is Blynk? **Tech Explorations**, [s.d]. Disponível em: <https://techexplorations.com/guides/blynk/1-what-is-blynk/>. Acesso em: 30 mai. 2022.