

**UNIVERSIDADE ESTADUAL PAULISTA “JÚLIO DE MESQUITA FILHO”**

**FACULDADE DE ENGENHARIA DE ILHA SOLTEIRA**

**DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

**EVANDRO MARTINS GREGOLETO**

**AUTOMAÇÃO RESIDENCIAL VIA IOT CONTROLANDO ESP8266 POR  
COMANDO DE VOZ**

Ilha Solteira

2022

EVANDRO MARTINS GREGOLETO

**AUTOMAÇÃO RESIDENCIAL VIA IOT CONTROLANDO ESP8266 POR  
COMANDO DE VOZ**

Trabalho de conclusão de curso apresentado à Faculdade de Engenharia do Campus de Ilha Solteira – UNESP como parte dos requisitos para obtenção do título de Engenheiro Eletricista.

Orientador: **Prof. Dr. Carlos Antônio Alves**

Ilha Solteira

2022

FICHA CATALOGRÁFICA

Desenvolvido pelo Serviço Técnico de Biblioteca e Documentação

G821a Gregoleto, Evandro Martins.  
Automação residencial via iot controlando ESP8266 por comando de voz /  
Evandro Martins Gregoleto. -- Ilha Solteira: [s.n.], 2022  
35 f. : il.

Trabalho de conclusão de curso (Graduação em Engenharia Elétrica) -  
Universidade Estadual Paulista. Faculdade de Engenharia de Ilha Solteira, 2022

Orientador: Carlos Antônio Alves  
Inclui bibliografia

1. Internet das coisas. 2. Comando de voz. 3. Automação residencial.

  
Raiane da Silva Santos

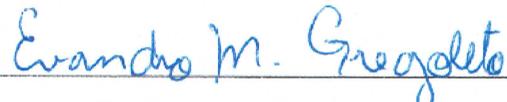
## ATA DE DEFESA DE TRABALHO DE GRADUAÇÃO

Aos quatro dias do mês de Julho do ano de dois mil e vinte e dois, o discente **Evandro Martins Gregoleto**, matriculada sob o nº 171052846, tendo como banca examinadora o seu orientador, o *Prof. Dr. Carlos Antonio Alves*, o *Prof. Dr. Jean Marcos de Souza Ribeiro* e o *Prof. Dr. Rodrigo Cardim*, apresentou o Trabalho de Graduação intitulado "AUTOMAÇÃO RESIDENCIAL VIA IOT CONTROLANDO ESP8266 POR COMANDO DE VOZ", obtendo a nota 9.0 ( NOVE ) e conceito APROVADO.



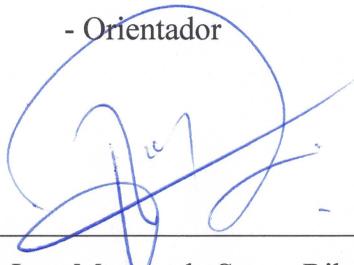
Prof. Dr. Carlos Antonio Alves

- Orientador



Evandro Martins Gregoleto

- Discente -



Prof. Dr. Jean Marcos de Souza Ribeiro

- Membro da Banca -



Prof. Dr. Rodrigo Cardim

- Membro da Banca -

## **AGRADECIMENTOS**

Agradeço a minha família por todo apoio, seja ele financeiro ou emocional. Durante toda minha jornada acadêmica, sempre estiveram apoiando minhas decisões, comemorando minhas vitórias e oferecendo consolo nos momentos de tristeza.

À minha namorada Camila Brito de Melo por ter me acompanhado durante grande parte do período de graduação, sendo paciente, me aconselhando e apoiando, independente de qual fosse a situação e sempre compartilhando momentos bons ao longo da vida.

Aos meus amigos de Dracena e de Ilha Solteira, que se fazem presente em minha vida por terem vivenciado momentos memoráveis de minha vida e terem tornado a experiência de graduação ainda mais incrível.

Ao meu orientador, Prof. Dr. Carlos Antônio Alves, pela confiança depositada em minha pessoa.

À Profa. Dra. Suely Cunha Amaro Mantovani, que marcou meu período acadêmico de forma positiva e me deu a honra de realizar meu primeiro trabalho na área de IoT e robótica como seu orientando.

À todos os professores que já fizeram parte de minha educação e àqueles que ainda o farão.

## RESUMO

Este trabalho aborda sobre o desenvolvimento de um sistema para automação de um ambiente residencial através do comando de voz. Foi utilizada a assistente virtual Alexa para que fosse feita a interface com o usuário e o reconhecimento dos comandos, além da utilização da plataforma Sinric Pro para estabelecer a conexão entre a assistente e o microcontrolador ESP8266. Também foram abordados os aspectos técnicos de configuração e integração dos serviços e dispositivos utilizados, juntamente à realização de testes que comprovaram a resiliência e desempenho do sistema em situações reais de uso.

**Palavras-chave:** Internet das Coisas, Comando de voz, Automação Residencial.

## **ABSTRACT**

This project aims to show the development of a system for automation of a residential environment through voice command. The virtual assistant Alexa was used to interface with users and recognize the commands, It was also used the Sinric Pro to establish the connection between the assistant and the ESP8266 microcontroller. The technical settings and integration of services and devices were also presented, together with tests that proved the resilience and performance of the system in real situations.

**Keywords:** Internet of Things, Voice command, Residential Automation.

## LISTA DE ILUSTRAÇÕES

Figura 1 – Características IoT.

Figura 2 – Arquitetura da internet das coisas.

Figura 3 – ESP8266.

Figura 4 – NodeMCU.

Figura 5 – Tabela de características do NodeMCU.

Figura 6 – Arduino IDE.

Figura 7 – Dashboard Sinric Pro.

Figura 8 – Fluxograma de operação Sinric Pro.

Figura 9 – Dispositivos Amazon

Figura 10 – Preferências Arduino IDE.

Figura 11 – Gerenciador de placas Arduino IDE.

Figura 12 – App Sinric Pro e Integração Alexa Sinric Pro.

Figura 13 – Protótipo de bancada.

Figura 14 – Teste de desempenho.

Figura 15 – Teste de resiliência.

Figura 16 – Esquema elétrico da aplicação.

Figura 17 – Esquema elétrico da fechadura eletrônica.

Figura 18 – Comparação de preço.

## LISTA DE ABREVIATURAS E SIGLAS

API – Application Programming Interface.

DC – Direct Current.

EEPROM – Electrically Erasable Programmable Read-Only Memory.

EXI – Efficient XML Interchange.

GPIO – General Purpose Input/Output.

GSM – Global System for Mobile.

HMI – Human-Machine Interface.

IA – Inteligência Artificial.

IDE – Integrated Development Environment.

IoT – Internet of Things.

IP – Internet Protocol.

LTE – Long Term Evolution.

MCU – Microcontroller Unit.

NFC – Near Field Communication.

OWL – Web Ontology Language.

PWM – Pulse Width Modulation.

RAM – Random Access Memory.

RDF – Resource Description Framework.

RFID – Radio Frequency Identification.

SDK – Software Development Kit.

USB – Universal Bus Serial.

Wi-Fi – Wireless Fidelity.

## Sumário

<b>1. Introdução</b> .....	8
<b>1.1 Objetivos</b> .....	9
<b>2. Fundamentação Teórica</b> .....	10
<b>2.1 Internet das Coisas</b> .....	10
<b>2.1.1 Conceito e Características</b> .....	10
<b>2.1.2 Arquitetura</b> .....	12
<b>2.1.3 Áreas de aplicação</b> .....	12
<b>2.2 Microcontroladores</b> .....	14
<b>2.2.1 ESP8266</b> .....	14
<b>2.2.2 NodeMCU</b> .....	15
<b>2.3 Softwares e Serviços</b> .....	17
<b>2.3.1 Arduino IDE</b> .....	17
<b>2.3.2 Sinric Pro</b> .....	18
<b>2.3.3 Alexa</b> .....	20
<b>3. Materiais e Métodos</b> .....	21
<b>4. Resultados</b> .....	25
<b>5. Conclusão</b> .....	28
<b>Referências</b> .....	29
<b>Apêndice A</b> .....	31

## 1. Introdução

Ainda no final do século XX é apresentado o conceito de Internet das coisas (em inglês, *Internet of Things* - IoT), um conceito onde diversos dispositivos eletrônicos são conectados à internet para que possam se comunicar entre si, trocando informações e comandos. Isso gerou um novo mercado de serviços que busca proporcionar maior conforto, acessibilidade, segurança e agilidade à vida das pessoas.

Esse conceito ganhou ainda mais força com a rápida evolução da eletrônica digital e das comunicações sem fio, como o Wi-Fi e o 4G LTE, proporcionando além de um maior número de dispositivos conectados, uma comunicação com maior velocidade e estabilidade neste fluxo de dados. Assim, tem-se um cenário onde sensores captam os dados do mundo físico, para que se tenha uma interpretação desses dados e uma resposta através de ações, tornando possível reinventar diversos setores da sociedade, como a manufatura, agricultura, habitação, locomoção e educação (LOPEZ RESEARCH LLC, 2013; ROCHA, 2019).

A domótica é a automação e o controle aplicados às residências, onde utiliza-se da tecnologia para conectar diversos dispositivos domésticos à internet e facilitar a realização de ações rotineiras de uma casa como mantê-la segura, sustentável, limpa e organizada, possibilitando uma melhor qualidade de vida para o usuário (MURATORI & DAL BÓ, 2011).

Esses recursos da domótica podem se tornar indispensáveis para a independência e acessibilidade de pessoas com deficiência. Assim, torna-se possível realizar tarefas cotidianas consideradas impossíveis e complexas em uma casa que não seja adaptada, tornando a vida de pessoas portadoras de deficiência física, idosos e obesos muito mais fácil e prazerosas (MARTINS, 2019).

A automação de aparelhos domésticos é uma aplicação que está se popularizando cada vez mais, porém há uma situação onde pouca ou quase nenhuma dessas tecnologias são desenvolvidas por empresas brasileiras e há a necessidade de importação de peças e do conhecimento para o desenvolvimento e aplicação em solo nacional. Porém, sabe-se que a integração de automação de portões, luzes, fechaduras, sistema de refrigeração e outros eletrodomésticos foi facilitada devido à rápida evolução e popularização da eletrônica digital e das placas de prototipagem que proporcionam um contato facilitado para os desenvolvedores, possibilitando uma

automação de baixo custo e que pode ser personalizada de acordo com a demanda de cada projeto.

Como exemplo, há o módulo ESP8266 que é um microcontrolador de baixo custo que tem a capacidade de comunicação via Wi-Fi, possibilitando o controle de suas portas digitais remotamente. O módulo possui uma interface USB-serial para comunicação e alimentação, além de um regulador de tensão em 3,3 V, enquanto a programação pode ser feita utilizando LUA ou a IDE do Arduino, utilizando linguagem C/C++ (CURVELLO, 2015).

## **1.1 Objetivos**

Este trabalho de graduação tem como objetivo apresentar o desenvolvimento de um sistema de automação de baixo custo para residências que será integrado à assistente virtual Alexa. A solução utilizará como base a plataforma NodeMCU para a conexão dos dispositivos de uma casa com a internet e realizar ações como controlar iluminação, liberar acesso em travas eletrônicas além do controle de tomadas inteligentes através do comando de voz.

Haverá a elaboração de protótipo para testes em bancada, com o propósito de estudar o desempenho do sistema e concluir se há a possibilidade de aplicação da solução em projetos reais.

## 2. Fundamentação Teórica

Como base teórica para a realização do trabalho de graduação, foi feito um estudo a respeito da origem e dos pilares da internet das coisas, além da sua aplicação em várias esferas da atividade humana.

Também apresenta-se estudos sobre as características do *hardware* e dos serviços utilizados na construção do protótipo.

### 2.1 Internet das Coisas

O termo internet das coisas surgiu no ano de 1999 e o pesquisador Kevin Ashton foi um dos primeiros a abordar este conceito que ganharia ainda mais força impulsionado pela evolução da eletrônica digital e da telecomunicação. A premissa é utilizar a tecnologia para conectar objetos cotidianos à internet e, assim, trazer informações do mundo físico para dentro do mundo digital e vice-versa (ASHTON, 2009).

#### 2.1.1 Conceito e Características

A internet das coisas é um conceito onde pessoas e objetos estariam conectados entre si através da internet para que possam receber e compartilhar informações, analisar dados e até atuar de forma remota. Quando esses dispositivos são capazes de se conectar à internet e podem processar informações, eles são chamados de dispositivos inteligentes, pois estes objetos podem agir de maneira inteligente quando comparado ao que não possuem essa conexão (CLARK, 2016).

As características do objeto inteligente é uma combinação de diversas tecnologias de forma complementar com o intuito de tornar viável a integração entre os objetos do ambiente físico e os serviços do mundo virtual.

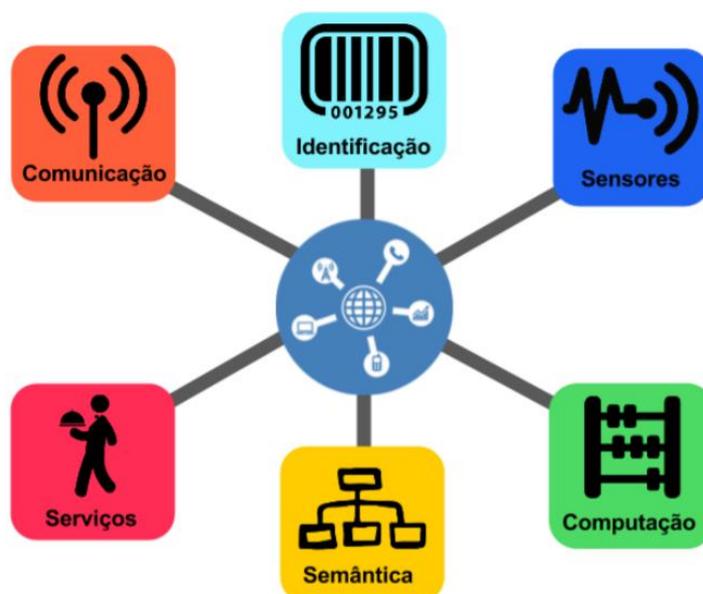
Assim, apresenta-se alguns blocos básicos para construção de um objeto inteligente para IoT:

- Identificação: refere-se à necessidade de identificar os objetos para se obter acesso às informações de maneira remota. Há disponível no mercado diversas tecnologias que podem ser usadas para identificar os objetos, como a identificação por rádio frequência (em inglês, *Radio Frequency Identification* – RFID) e o monitoramento por imagem.

- **Sensores/Atuadores:** Em plataformas IoT, tem-se os sensores para captar as informações do ambiente ao seu redor e, após coletada, fornecem os dados para que sejam processados ou armazenados. Já os atuadores são dispositivos físicos controlados pela plataforma IoT que podem manipular o ambiente, como válvulas solenóides, trancas eletrônicas, relés, motores e etc.
- **Comunicação:** refere-se às diferentes tecnologias utilizadas para estabelecer a conexão dos dispositivos inteligentes e a internet. Algumas das tecnologias são Wi-Fi, *Bluetooth Low Energy* (BLE) e GSM.
- **Computação:** refere-se ao processamento necessário para rodar o algoritmo e processar dados gerais, utilizando FPGAs, microcontroladores e processadores.
- **Serviços:** Com a evolução dos dispositivos inteligentes, surgiram diversos tipos de serviços que complementam o universo IoT, tais como: Serviço de Identificação, agregação de dados, colaboração e Inteligência.
- **Semântica:** refere-se à extração dos dados e conhecimento obtidos pelos objetos IoT, fazendo um uso eficiente dos recursos existentes na IoT. É possível citar a utilização de diversas técnicas como Resource Description Framework (RDF), Web Ontology Language (OWL) e Efficient XML Interchange (EXI) (Santos, 2016).

Na Figura 1 ilustra as características citadas acima.

Figura 1 – Características IoT.



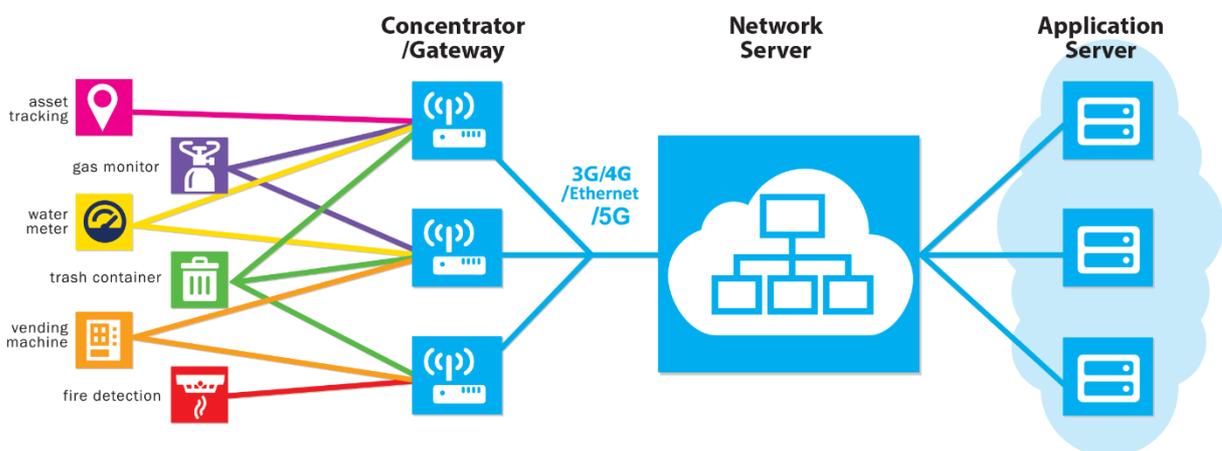
Fonte: (Santos, 2016).

### 2.1.2 Arquitetura

Para entendermos melhor o funcionamento de uma rede IoT, podemos fazer um estudo da arquitetura desta rede que é composta principalmente pelos dispositivos inteligentes como os sensores e atuadores que se comunicarão com os gateways, que são responsáveis por concentrar e transmitir as informações para os servidores da internet por meio das tecnologias 3G, 4G, 5G ou Ethernet. Uma vez que a informação alcança os servidores da internet, elas serão direcionadas aos servidores da aplicação utilizando protocolos que se encarregam de transmitir corretamente o conteúdo das mensagens até o destinatário correto, como o MQTT e o TCP/IP. Nos servidores da aplicação estarão alocados os parâmetros de configuração da aplicação e a inteligência artificial que fará a gestão das informações disponibilizadas pelos dispositivos e tomará as devidas decisões de acordo com o propósito da aplicação (SIMÕES, 2020).

Na Figura 2 é representada, de maneira simplificada, a arquitetura de uma rede da internet das coisas.

Figura 2 – Arquitetura da internet das coisas.



Fonte: (SIMÕES, 2020).

### 2.1.3 Áreas de aplicação

Com o avanço das tecnologias de comunicação sem fio e sensoriamento, existem diversas áreas que se beneficiam desse conceito de interconexão de dispositivos para proporcionar maior segurança, precisão, produtividade e até conforto para o usuário final. Algumas dessas áreas são:

- **Residencial**

Pode-se exemplificar a aplicação residencial da internet das coisas pelo uso dos smartphones e de assistentes virtuais para o controle de eletrodomésticos

conectados, câmeras, lâmpadas automatizadas e até para cortinas com acionamento automático. O foco desse uso é proporcionar mais segurança, qualidade de vida e conforto aos usuários, buscando uma economia de energia elétrica através do monitoramento do consumo individual dos dispositivos (CLARK, 2016).

- **Indústria**

No setor industrial, já é possível notar que as novas versões dos dispositivos de automação, além de controlarem os atuadores de máquinas industriais, conseguem se conectar à rede e transmitir dados de operação e de sensores dessas máquinas para *softwares* de rastreabilidade de produção, possibilitando identificação de falhas de processo e melhorando a gestão de produção e manutenção (MAGRANI, 2018).

- **Agricultura**

A agricultura é mais um dos setores que deve se beneficiar da utilização de dispositivos IoT, uma vez que estudos recentes mostram que a produção mundial de produtos agrícolas deve crescer até 60% nos próximos 25 anos, enquanto o crescimento de terras aráveis é de apenas 5%. O uso de máquinas autônomas e a introdução de tecnologias de análise e processamento de dados meteorológicos deve auxiliar a produção do campo a ser mais eficaz, diminuindo perdas e gastos enquanto há um aumento na qualidade e no faturamento das colheitas (ALBERTIN, 2017).

- **Transporte**

Ao citar o impacto da IoT no setor de transporte, é possível citar a existência de carros com direção autônoma, possibilitando que os carros sejam capazes de identificar objetos em sua trajetória e tomem decisões em tempo real de acordo com cada situação. A conexão de carros com a internet faz com que o *software* e os parâmetros de configuração do carro estejam sempre atualizados e o grande número de câmeras e sensores disponibilizam à inteligência artificial ainda mais informações sobre o mundo real para que se tenha uma navegação completamente autônoma com alto nível de segurança (TESLA, 2022).

Além da venda de veículos mais inteligente, há oportunidades também no setor de aluguel de veículos onde já é possível alugar e acessar um carro de forma totalmente remota, sem necessitar ter lojas físicas, longos contratos e deslocamentos desnecessários. Com apenas alguns toques na tela do celular você localiza e aluga o

carro mais perto da sua localização, sendo que o carro é aberto automaticamente e estará pronto para o uso assim que o usuário chegar no local (TURBI, 2018).

## **2.2 Microcontroladores**

O microcontrolador consiste em um circuito integrado programável que contém a estrutura básica de um microcomputador, isto é, há uma Unidade Central de Processamento, memórias RAM e EEPROM (Memórias de leitura e escrita volátil e não volátil, respectivamente), portas de entrada/saída.

Há a possibilidade de alguns modelos de microcontroladores incorporarem diferentes módulos complementares, como conversores analógico/digital (A/D), módulos PWM (em inglês, Pulse Width Modulation), módulos de comunicação serial entre outros. Como esse sistema possui diversos componentes dispostos em unidades diferentes, há a necessidade de interligá-los por meio de condutores paralelos constituindo um barramento para condução das informações através dos circuitos.

Os dispositivos que utilizam microcontroladores em seu sistema visam gerar e processar dados melhorando sua confiabilidade e conectividade além de diminuir seu tamanho, custo e consumo de energia. Alguns dos fabricantes de microcontroladores atuais são a Intel, Espressif Systems, Microchip, Atmel, Motorola, etc.

### **2.2.1 ESP8266**

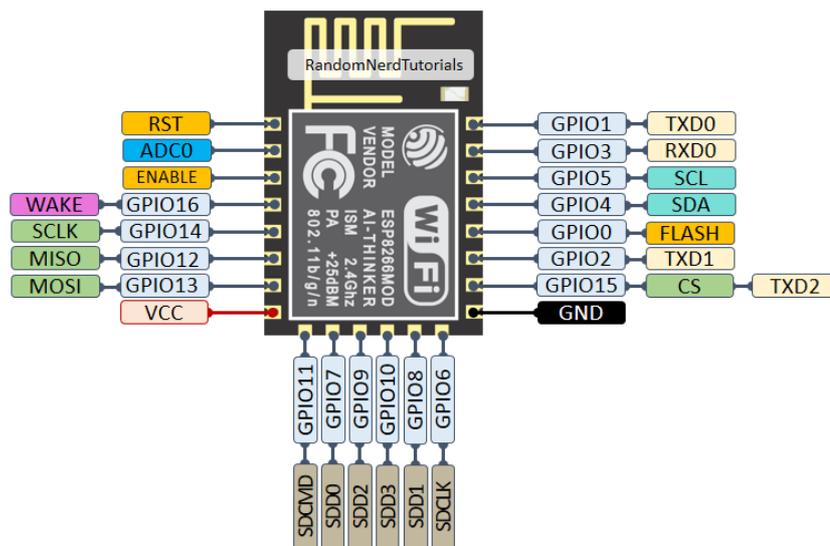
O ESP8266, da empresa chinesa Espressif Systems, é um módulo microcontrolador capaz de se conectar a redes Wi-Fi de 2.4 GHz e que pode ser utilizado tanto em uma aplicação *stand-alone*, onde ele não precisa de nenhum outro dispositivo para funcionar, como em uma aplicação de servidor escravo de outro microcontrolador. Assim, na condição de escravo, o ESP8266 funciona como um adaptador possibilitando a conexão Wi-Fi para outros microcontroladores, como, por exemplo, as placas da família Arduino (OLIVEIRA, 2017).

O ESP8266 pode operar em duas configurações, sendo ponto de acesso ou cliente. Na configuração ponto de acesso, o comportamento do módulo será semelhante ao de um roteador, possibilitando a criação de uma rede Wi-Fi restrita protegida por login e senha. Neste modo, é criado um servidor, com IP aleatório ou predefinido, que pode conter uma página web com informações dos componentes eletrônicos conectados ao ESP8266. Já no modo cliente, o

microcontrolador se conecta a uma rede Wi-Fi já existente, criando um servidor onde todos os dispositivos conectados nesta mesma rede Wi-Fi têm acesso a este servidor através do endereço IP. Ainda é possível mesclar os dois modos de operação, resultando em uma terceira configuração onde o ESP8266 atua simultaneamente como ponto de acesso e cliente (IKRABA, 2016).

Na montagem do protótipo, foi utilizada a configuração cliente para que fosse possível se conectar aos dispositivos da rede, tornando uma plataforma IoT com transmissão dos dados. É utilizado no projeto o SoC ESP8266-12, representado na Figura 3, acoplado à plataforma NodeMCU, que é um módulo de custo e tamanho reduzido quando comparado a outras plataformas de desenvolvimento IoT disponíveis no mercado.

Figura 3 – ESP8266.



Fonte: (RANDOM NERDS TUTORIALS, 2022)

Na figura 3 é possível observar a disposição física dos terminais de alimentação e comunicação do ESP8266.

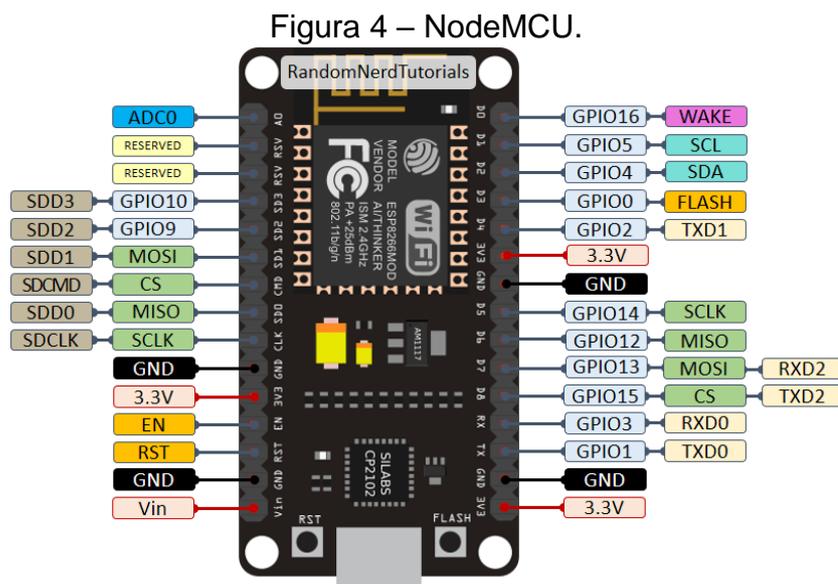
### 2.2.2 NodeMCU

A plataforma IoT NodeMCU foi desenvolvida para facilitar a prototipagem de projetos utilizando o ESP8266, pois ele já possui o microcontrolador acoplado a diversos componentes que possibilitam um uso mais versátil da plataforma. O NodeMCU surgiu pouco tempo após o lançamento do ESP8266, com o propósito de ser um dispositivo completo para o desenvolvimento de soluções voltadas para a IoT.

Uma grande vantagem dessa placa é a presença de uma interface USB-serial integrada do padrão Micro-USB, que pode ser utilizada tanto para a alimentação da plataforma quanto para a transmissão dos códigos do computador para as memórias da plataforma. O NodeMCU pode ser programado tanto através da linguagem LUA quanto da linguagem C++.

Dentre as características do NodeMCU, a faixa de tensão de operação da placa está entre 5 e 9 V, permitindo a utilização de baterias alcalinas de 9 V ou fontes DC para a alimentação do microcontrolador. A placa também possui 11 portas GPIO (Porta de entrada / saída de propósito geral), as quais podem ser utilizadas para conectar sensores, atuadores ou outros dispositivos eletrônicos (OLIVEIRA, 2017).

Na Figura 4 é apresentada a plataforma NodeMCU juntamente à informação de pinagem.



Fonte: (RANDOM NERDS TUTORIALS, 2022)

A maior distribuição de pinos aterrados e de alimentação 3,3 V, além da presença de um regulador de tensão e uma interface USB para comunicação integrados tornam o NodeMCU uma plataforma ainda mais completa para a elaboração de dispositivos IoT, permitindo a alimentação de outros componentes, como os sensores.

Na Figura 5 está representada uma tabela com outras características elétricas e eletrônicas do NodeMCU.

Figura 5 – Tabela de características do NodeMCU.

<b>Núcleos</b>	1
<b>Arquitetura</b>	32 Bits
<b>Frequência</b>	80 MHz
<b>Wi-Fi</b>	IEEE 802.11 b/g/n
<b>RAM</b>	160 KB
<b>Flash</b>	16 MB
<b>GPIO</b>	16
<b>Interfaces</b>	SPI/I2C/UART/I2S
<b>ADC</b>	1

Fonte: (OLIVEIRA, 2017).

### 2.3 Softwares e Serviços

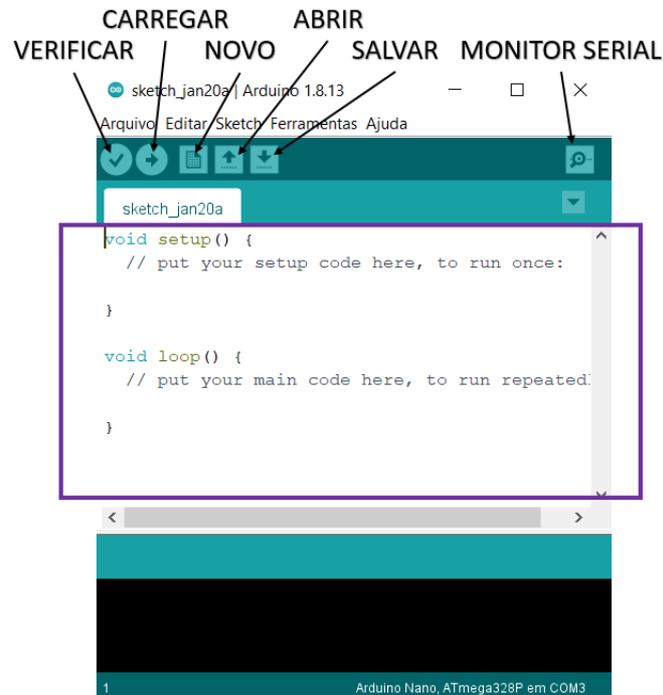
Nesta sessão serão apresentadas e explicadas mais informações sobre os *softwares* e serviços utilizados durante o desenvolvimento do protótipo.

#### 2.3.1 Arduino IDE

O *software* Arduino IDE é um ambiente de desenvolvimento integrado que permite a programação e o carregamento de códigos para placas da família Arduino e ESP8266/ESP32 com uma interface simples e amigável. Utilizar esse ambiente traz diversas vantagens ao desenvolvedor como: compilador integrado, monitor serial, reconhecimento de funções do código, ferramentas de gestão do projeto, entre outros. A linguagem utilizada pela IDE do Arduino é bem próxima à linguagem C++, o que facilita o desenvolvimento do código. Um código padrão feito nesse *software* utiliza duas funções principais que permite as placas funcionarem, a função “void setup” (função onde é realizada a declaração de parâmetros) e a função “void loop” (função onde os comandos são executados de maneira infinita pelo microcontrolador) (QUINTINO, 2021).

Na Figura 6 está representada a janela inicial da Arduino IDE, sendo possível observar como é a organização do ambiente de programação utilizado no projeto, sendo possível destacar os principais elementos necessários para a utilização do *software*, como os botões de verificação/compilação do código, o botão de carregamento para o microcontrolador, além da janela do monitor serial.

Figura 6 – Arduino IDE.



Fonte: (QUINTINO, 2021)

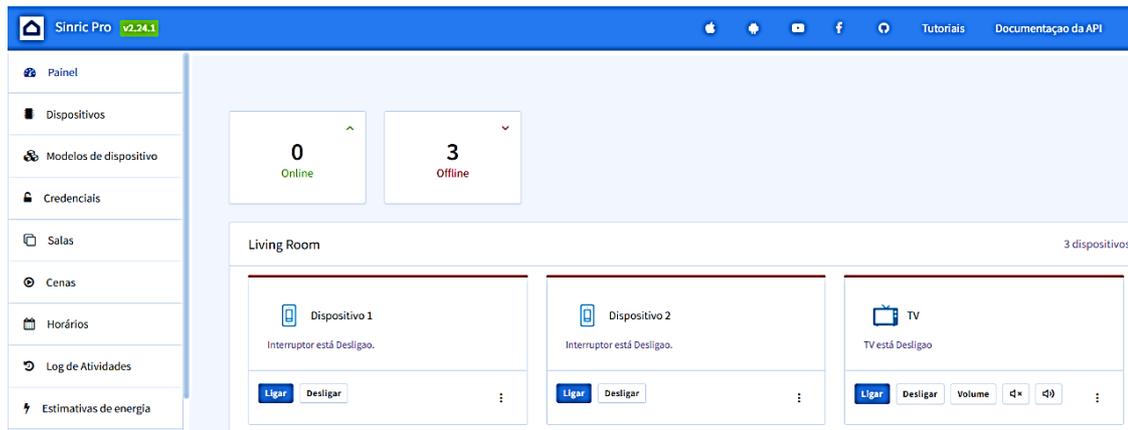
A janela “Monitor Serial” destacada na Figura 6 torna possível a visualização de mensagens enviadas do dispositivo para o computador, quando conectados pela interface USB-serial.

### 2.3.2 Sinric Pro

O Sinric Pro é um serviço gratuito feita para a possibilitar a integração de dispositivos IoT com as principais plataformas de automação residencial do mercado (Amazon Alexa, Google Home, Samsung SmartThings e IFTTT). Esta API pode ser utilizada em todos os dispositivos da família Arduino e ESP que tenham conexão com a internet, seja através de Shield ou nativo, além de ter SDKs disponíveis para Python e NodeJS.

Além de fornecer a conexão e comunicação entre as plataformas de automação e as placas de desenvolvimento IoT, o Sinric Pro disponibiliza uma página web em formato de *dashboard* para que o usuário possa criar, personalizar e acompanhar os objetos criados em seu projeto. Na Figura 7 observa-se a representação gráfica deste dashboard (SINRIC PRO, 2022).

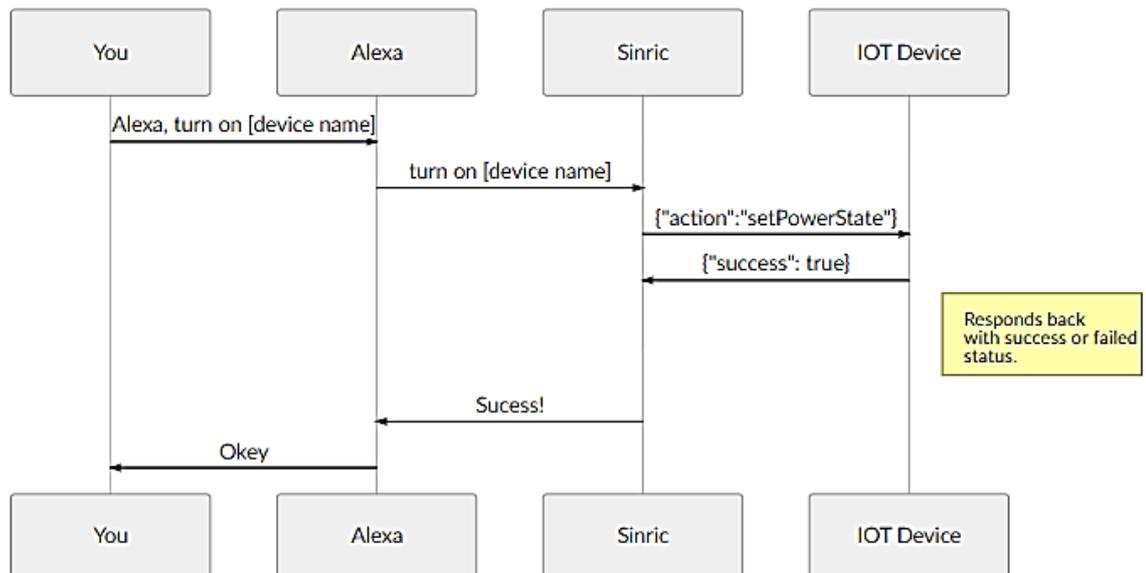
Figura 7 – Dashboard Sinric Pro.



Fonte: Próprio Autor.

Portanto, o comportamento do sistema a ser construído nesse trabalho utilizando o Sinric Pro deve ser semelhante ao fluxo disposto na Figura 8, onde há a interação do usuário com a Alexa e o comando é transmitido da assistente para o Sinric Pro, que será responsável pela comunicação com o dispositivo IoT, que por sua vez irá atuar e responder o comando com o status de sucesso ou falha da operação.

Figura 8 – Fluxograma de operação Sinric Pro.



Fonte: (SINRIC PRO, 2022).



### 3. Materiais e Métodos

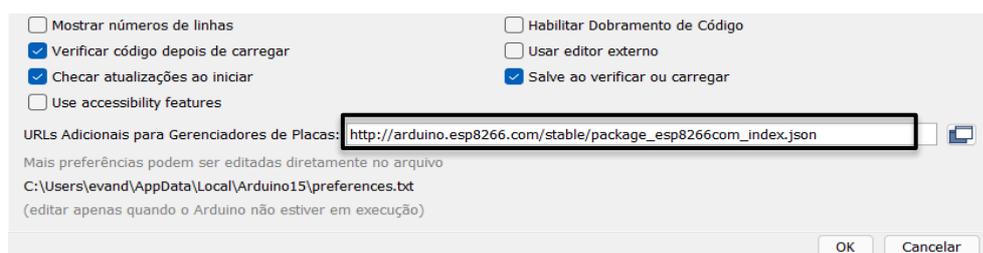
Os materiais utilizados na montagem do protótipo foram:

- Placa NodeMCU
- Fonte chaveada de 9 V
- Kit protoboard com fonte ajustável e jumper
- Módulos relé 5 V
- Cabo USB A - Micro USB
- Amazon Echo Dot 3ª Geração

Como primeiros passos, foram feitos os estudos sobre o funcionamento, programação e configuração do módulo ESP8266 juntamente com a plataforma Sinric Pro. As GPIO são pinos de entrada e saída de dados de propósito geral, que foram utilizados para fazer o controle dos módulos relés, sendo possível controlar se o circuito em questão deveria ser aberto ou fechado de acordo com o comando de voz dado pelo usuário. Assim, foram utilizados no projeto os pinos D0 e D3 do NodeMCU.

Para que o computador reconhecesse a placa NodeMCU 1.0 e ocorresse a transferência do código, foi feito um procedimento de configuração da IDE do Arduino. Nessa configuração foi expandido o número de placas reconhecidas pela IDE através de uma URL própria do ESP8266 preenchida no campo “*Additional Boards Manager URLs*” presente da tela de preferências. Na Figura 10 é representado o procedimento de configuração da IDE.

Figura 10 – Preferências Arduino IDE.

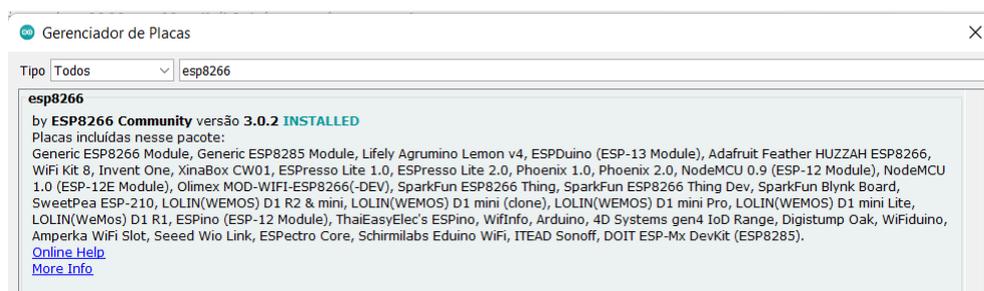


Fonte: Próprio Autor.

Após essa configuração, foi feita a instalação do pacote de placas da família ESP8266 disponível na janela Gerenciador de placas, tornando possível a

utilização da Arduino IDE para programar a plataforma NodeMCU. Na Figura 11 é representado o passo descrito acima e demonstrada a instalação do pacote de desenvolvimento das placas baseadas no ESP8266.

Figura 11 – Gerenciador de placas Arduino IDE.



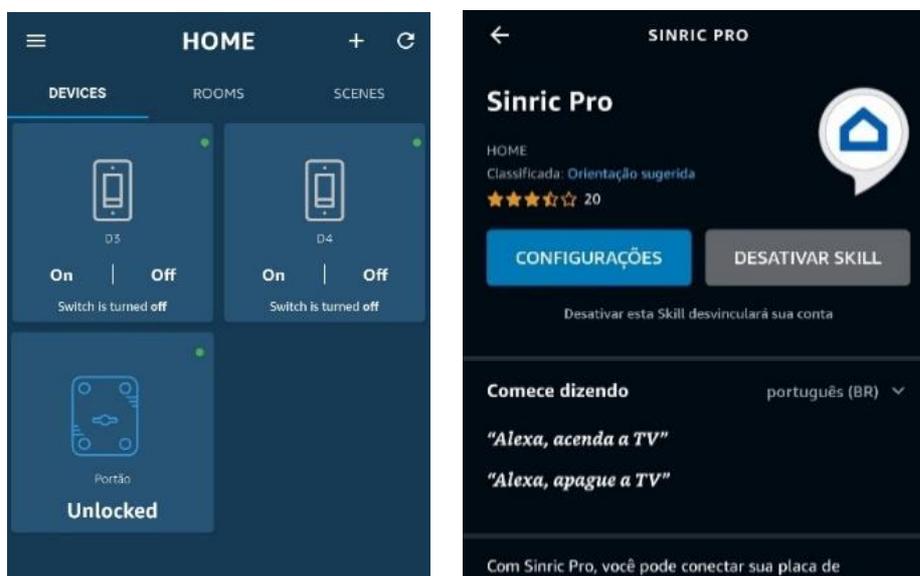
Fonte: Próprio Autor.

A plataforma Sinric Pro foi o serviço escolhido para fazer o gerenciamento dos dispositivos e controle do NodeMCU através da internet, possibilitando o acionamento remoto dos dispositivos via aplicativo ou comando de voz via integração com a assistente virtual Amazon Alexa. A configuração do serviço foi rapidamente feita ao criar uma conta gratuita no site do serviço para poder acessar o *dashboard*, que é um ambiente onde é feita a administração dos dispositivos, do serviço e das cenas criadas e está representado pela Figura 7. Com a criação dessa conta, foram gerados o *tokens* de identificação e proteção do sistema que serão utilizados no código carregado no microcontrolador.

Após a criação da conta no serviço, foram criados três dispositivos dentro do dashboard do Sinric Pro para que fizessem o acionamento de dois interruptores e o controle de acesso por meio do controle de uma fechadura eletrônica. Cada dispositivo criado também gerou um *token* de autenticação que também foi adicionado ao código assim como foi feito com o *token* do sistema.

A Alexa, por meio do login feito na Skill Sinric Pro, detectou automaticamente a existência dos dispositivos criados no dashboard e já foi capaz de enviar comandos para os dispositivos. Utilizar o Sinric Pro é interessante pois possibilita simular o comportamento de diversos aparelhos como uma lâmpada, um termostato, uma fechadura eletrônica além de permitir uma integração mais fácil com assistentes de voz além de disponibilizar um app para o sistema operacional Android para controle dos dispositivos. Na Figura 12 há duas representações, sendo do App Sinric Pro (a) e da integração da Alexa com o Sinric Pro (b).

Figura 12 – App Sinric Pro e Integração Alexa Sinric Pro.



(a)

(b)

Fonte: Próprio Autor.

Após a configuração inicial do Sinric e da Arduino IDE, o próximo processo seria a programação do algoritmo que será carregado ao ESP8266 e será responsável por estabelecer a conexão entre o microcontrolador, a internet e o servidor do Sinric Pro.

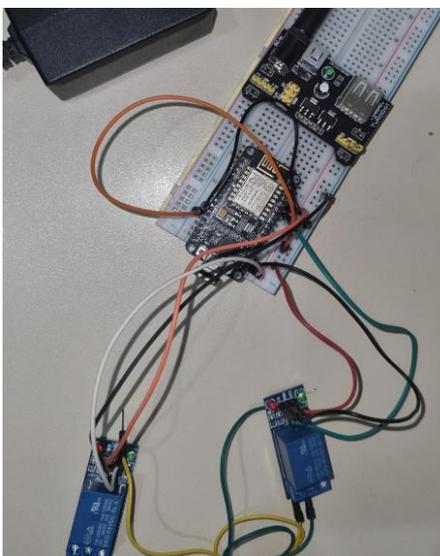
No código representado no Apêndice A, tem-se o código desenvolvido demonstrando a importação das bibliotecas e definição das variáveis que guardarão informações importantes do código como: SSID e senha do Wi-Fi, pinagem e identificação do *hardware*, além das chaves de segurança que identificam e protegem os dados de cada projeto realizado no Sinric Pro. Assim, a cada projeto e dispositivo criado, um novo número de identificação único também é gerado para que seja possível proteger e mapear os comandos dados a cada dispositivo via servidor. Isso torna possível a utilização de mais de uma placa NodeMCU além de aumentar a segurança e confiabilidade dos dados recebidos e enviados.

Após a configuração da rede e dos parâmetros, há as rotinas de programação que serão chamadas toda vez que um comando é recebido via Alexa ou Dashboard. Essas funções comparam as informações recebidas nas variáveis "*deviceId*" e "*state*". Assim, é possível identificar qual dispositivo deve ser acionado e qual estado ele deve

assumir de acordo com o comando dado pelo usuário, ligado e desligado (para o caso dos interruptores) ou aberto e fechado (para o caso da fechadura). O código irá fazer essas comparações das informações recebidas e decidir se deve assumir um estado lógico alto ou baixo para a porta referente ao dispositivo selecionado.

Com o código feito e transferido para o microcontrolador, houve a montagem do protótipo em bancada apresentado na Figura 13, onde a alimentação do módulo ESP8266 e dos módulos relés foram feitas por meio de uma fonte ajustável para protoboard alimentada por uma fonte chaveada bivolt de 9 V. Além disso, utilizou-se jumpers para a distribuição da energia elétrica pelas trilhas da protoboard.

Figura 13 – Protótipo de bancada.



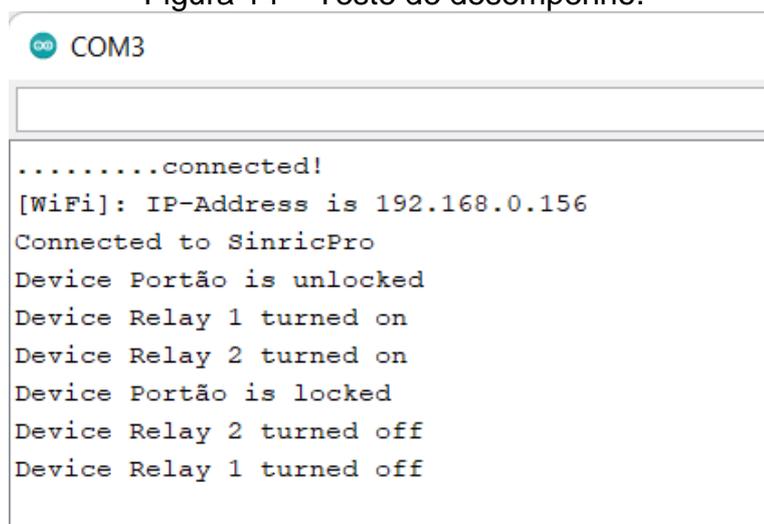
Fonte: Próprio autor.

#### 4. Resultados

Com a energização do sistema, representado na Figura 13, a primeira característica que chamou a atenção foi a conexão com a internet e com o servidor do Sinric Pro que foi rapidamente estabelecida. Assim, em menos de 20 segundos após a energização, o sistema já estava disponível para receber os comandos do usuário.

Interessante ressaltar a eficiência no reconhecimento dos dispositivos dentro do ambiente Alexa através da integração com o Sinric Pro, uma vez que o serviço da Amazon emitia uma sinalização sonora ao encontrar um novo dispositivo, poucos segundos após o processo de criação de um dispositivo ser concluído no dashboard. Na Figura 14 é representado o teste de desempenho onde aparecem as características destacadas acima.

Figura 14 – Teste de desempenho.

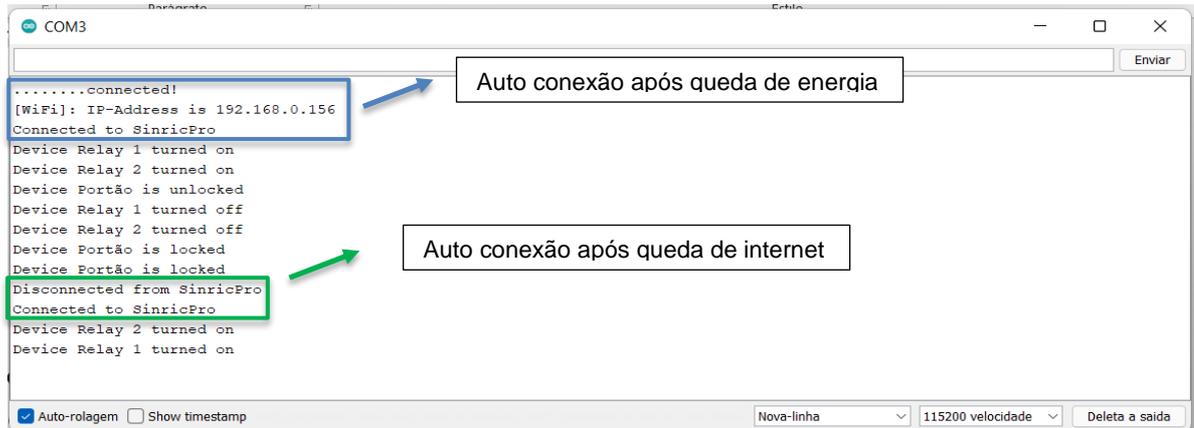
A screenshot of a terminal window titled 'COM3'. The terminal displays the following text: '.....connected!', '[WiFi]: IP-Address is 192.168.0.156', 'Connected to SinricPro', 'Device Portão is unlocked', 'Device Relay 1 turned on', 'Device Relay 2 turned on', 'Device Portão is locked', 'Device Relay 2 turned off', and 'Device Relay 1 turned off'.

```
.....connected!  
[WiFi]: IP-Address is 192.168.0.156  
Connected to SinricPro  
Device Portão is unlocked  
Device Relay 1 turned on  
Device Relay 2 turned on  
Device Portão is locked  
Device Relay 2 turned off  
Device Relay 1 turned off
```

Fonte: Próprio Autor.

O teste de desempenho demonstrou um excelente comportamento do sistema, onde todos os comandos dados para a Alexa foram devidamente captados e atendidos. Tornando possível observar o acionamento devido dos relés com um tempo de resposta muito baixo, um comportamento similar aos dispositivos que são referência no mercado de automação residencial. Nas simulações de quedas no sistema de alimentação e de conexão com a internet, obteve-se o resultado de que, em ambos os casos, o sistema iniciou e restabeleceu conexão com a internet e o servidor de forma automática após a normalização da situação. Os resultados desses testes estão representados na Figura 15.

Figura 15 – Testes de resiliência.

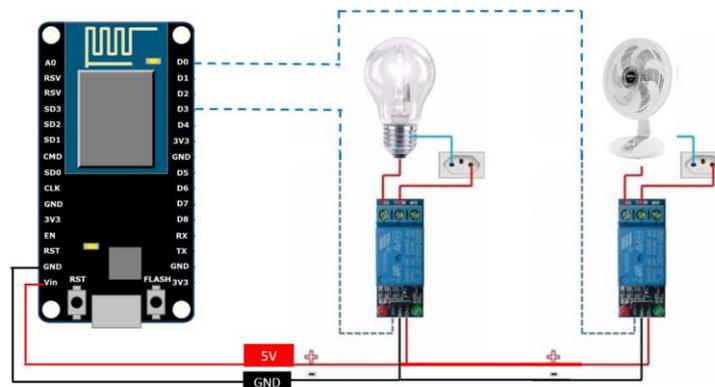


Fonte: Próprio autor

Após todos os testes de bancada, o projeto foi implementado em um ambiente que simulava o cômodo da casa de um usuário que possui uma das pernas amputadas, tendo o objetivo de estudar a viabilidade da aplicação deste sistema e como seria seu funcionamento em uma situação real. Na casa da pessoa voluntária, o projeto foi aplicado para o controle de uma luz, um ventilador e um portão eletrônico que não tinham nenhum tipo de comunicação com a internet.

Na Figura 16 está representada o esquema elétrico da instalação feita na residência do voluntário para o controle da luz e do ventilador.

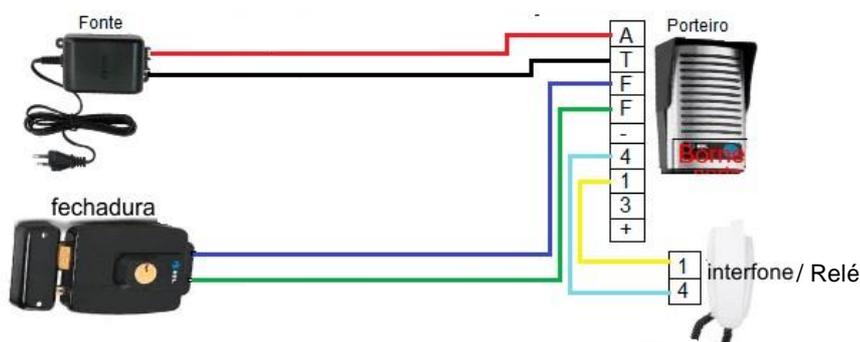
Figura 16 – Esquema elétrico da aplicação.



Fonte: Próprio Autor.

Na Figura 17 está representado o esquema elétrico de uma fechadura eletrônica idêntico ao esquema existente na casa do voluntário, sendo que o relé utilizado para controle de acesso por comando de voz foi instalado nos terminais identificados pelos números 1 e 4.

Figura 17 – Esquema elétrico da fechadura eletrônica.



Fonte: Próprio autor

Foi possível observar que o sistema foi facilmente integrado ao ambiente de aplicação, automatizando os dispositivos convencionais da casa e sua aplicação trouxe algum tipo de inteligência àqueles objetos, uma vez que eles poderiam ser acionados pelo comando de voz e poderiam ser programados para atuarem de forma automática. Outro ponto interessante foi a curva de aprendizagem rápida que o voluntário teve para aprender a interagir com o sistema e dar os comandos, dispensando treinamentos longos e complexos.

Na Figura 18 está representada a comparação de preço entre um dos principais kits de automação residencial do mercado com o preço gasto para a compra dos componentes envolvidos no desenvolvimento do projeto, onde é possível observar que o projeto custou menos de metade do preço do kit vendido comercialmente.

Figura 18 – Comparação de preço.

<p>Kit Casa Conectada Positivo Casa Inteligente, sua casa mais...</p> <p><b>R\$ 239,99</b></p> <p>Amazon.com.br</p>	<p>Conversor AC/DC - 90~240VAC - 9VDC / 1A - JP4</p> <p><b>R\$12,50</b></p> <p>QTD: 1</p>
	<p>Kit Protoboard</p> <p><b>R\$31,40</b></p> <p>QTD: 1</p>
	<p>Módulo Rele - 5v/10a - 4 Canais</p> <p><b>R\$25,50</b></p> <p>QTD: 1</p>
	<p>NodeMCU V3 - ESP8266 - ESP-12E - CH340</p> <p><b>R\$24,90</b></p> <p>QTD: 1</p>
<b>Total: R\$ 94,30</b>	

Fonte: Próprio Autor.

## 5. Conclusão

Com a conclusão dos testes e da aplicação do sistema, foi possível observar que a utilização do projeto para situações reais de automação em ambientes domiciliares é possível e viável. Apesar de necessitar de um conhecimento prévio em programação e eletrônica, para o desenvolvimento e aplicação do sistema da forma descrita neste trabalho de graduação, foi possível constatar que o projeto poderia ser facilmente escalado para versões *plug-&-play* do sistema, onde o usuário faria uma instalação e configuração guiada e facilitada do sistema sem a necessidade de integradores.

Outras características que reforçam o sucesso da operação do sistema foi a capacidade do sistema restabelecer a operação de forma automática após algum sinistro (perda de comunicação com a internet e de alimentação). Por fim, a simulação de aplicação do sistema na casa de um voluntário foi bem sucedida, uma vez que o projeto efetivou todas as tarefas designadas e as características de resiliência e desempenho do sistema, observadas nos testes de bancada, se mantiveram durante todo o período da aplicação prática.

Uma análise comparativa de preços também foi feita para estudar a viabilidade financeira do projeto desenvolvido quando comparado a um dos principais sistemas de automação residencial disponíveis no mercado, onde o total gasto para a montagem e desenvolvimento deste projeto foi de R\$ 94,30, enquanto um kit casa conectada da empresa Positivo, que realiza funções semelhantes, custa R\$ 239,99. Sendo possível concluir que com menos da metade do preço é possível desenvolver um sistema tão eficaz e confiável quanto os sistemas disponíveis no mercado.

## Referências

ALBERTIN, A. L.; ALBERTIN, R. M. d. M. **A internet das coisas irá muito além das coisas**. GV-Executivo, São Paulo, v. 16, p. 12–17, mar./abr. 2017.

AMAZON. **Alexa**. Disponível em: <https://developer.amazon.com/pt-BR/alexa>. Acesso em: 06 mar. 2022.

ASHTON, K. **That “internet of things” thing**. RFID Journal, p. 1, 2009

CLARK, J. **What is the Internet of Things?** 2016. Disponível em: <<https://www.ibm.com/blogs/internet-of-things/what-is-the-iot>>. Acesso em: 2 fev. 2022.

CURVELLO, André. **Apresentando o módulo ESP8266**. 2015. Disponível em: <https://www.embarcados.com.br/modulo-esp8266/>. Acesso em: 29 set. 2021.

IKRABA, Andrej *et al.* Streaming pulse data to the cloud with bluetooth LE or NODEMCU ESP8266. **2016 5Th Mediterranean Conference On Embedded Computing (Meco)**. Bar, Montenegro. jun. 2016.

LOPEZ RESEARCH LLC (San Francisco). **Uma introdução à Internet das Coisas (IoT)**. 2013. Disponível em: <[https://www.cisco.com/c/dam/global/pt\\_br/assets/brand/iot/iot/pdfs/lopez\\_research\\_an\\_introduction\\_to\\_iiot\\_102413\\_final\\_portuguese.pdf](https://www.cisco.com/c/dam/global/pt_br/assets/brand/iot/iot/pdfs/lopez_research_an_introduction_to_iiot_102413_final_portuguese.pdf) >. Acesso em: 29 set. 2021.

MAGRANI, E. **A Internet das Coisas**. 1. ed. Rio de Janeiro, RJ: FGV Editora, 2018.

MARTINS, Rafaela. **Conheça o poder da automação residencial para pessoa com deficiência**, Freedom, 4 de dezembro de 2019. Disponível em: <https://blog.freedom.ind.br/conheca-o-poder-da-automacao-residencial-para-pessoa-com-deficiencia/>. Acesso em: 29 set. 2021

MURATORI, J.R.; DAL BÓ, P.H. **Automação residencial: histórico, definições e conceitos**, p. 70- 77. Disponível em: [[http://www.osetoreletrico.com.br/wpcontent/uploads/2011/04/Ed62\\_fasc\\_automacao\\_capl.pdf](http://www.osetoreletrico.com.br/wpcontent/uploads/2011/04/Ed62_fasc_automacao_capl.pdf)]. Acesso em: 29 set. 2021.

**NODEMCU – ESP12: Guia completo – Introdução (Parte 1)**. 2018. Disponível em: <https://blog.eletrogate.com/nodemcu-esp12-introducao-1/>. Acesso em: 03 out. 2021.

RANDOM NERDS TUTORIALS. **ESP8266 Pinout Reference: Which GPIO pins should you use?** Disponível em: <https://randomnerdtutorials.com/esp8266-pinout-reference-gpios/>. Acesso em: 06 mar. 2022.

ROCHA, M. V **Artigos em destaque das redes sociais às coisas conectadas, tudo no seu smartphone**. 2019. Disponível em: <<https://abinc.org.br/das-redes-sociais-as-coisas-conectadas-tudo-no-seu-smartphone/>>. Acesso em: 29 set. 2021.

SANTOS, Bruno P. et al. **Internet das Coisas: da Teoria à Prática**. 2016. Salvador. Disponível em: <https://homepages.dcc.ufmg.br/~mmvieira/cc/papers/internet-das-coisas.pdf>. Acesso em: 2 fev. 2022.

SIMÕES, Marcos. **O Que Um Projeto Precisa Ter Para Ser Considerado IoT?** 2020. Disponível em: <https://www.dorotech.com.br/o-que-um-projeto-precisa-ter-para-ser-considerado-iot/>. Acesso em: 03 jul. 2022.

SINRIC PRO. **Sinric Pro**. Disponível em: <https://help.sinric.pro/>. Acesso em: 07 mar. 2022.

TESLA. **Autopilot**. 2022. Disponível em: <https://www.tesla.com/autopilot>. Acesso em: 06 mar. 2022.

TURBI (comp.). **Tecnologia: Turbi na Pequenas Empresas e Grandes Negócios**. 2018. Disponível em: <https://blog.turbi.com.br/tecnologia-turbi-na-midia/>. Acesso em: 06 mar. 2022.

OLIVEIRA, R.R. **Uso do microcontrolador ESP8266 para automação residencial**. 2017. 55 f. Trabalho de Conclusão de Curso (Graduação em Engenharia de Controle e Automação) – Universidade Federal do Rio de Janeiro, Escola Politécnica, 2017. Disponível em: <http://www.repositorio.poli.ufrj.br/monografias/monopoli10019583.pdf>. Acesso em: 06 mar. 2022.

QUINTINO, Eduardo de Castro. **O que é IDE Arduino?** 2021. Disponível em: <https://www.filipeflop.com/blog/o-que-e-ide-arduino/>. Acesso em: 07 mar. 2022.

## Apêndice A

```

#include <Arduino.h>
#include <ESP8266WiFi.h>
#include <SinricPro.h> //Bibliotecas de conexão com Sinric
#include <SinricProSwitch.h> //Bibliotecas de controle Interruptor
#include <SinricProLock.h> //Bibliotecas de controle Fechadura

/*****
 * Configuração dos Interruptores *
 *****/
#define RELAYPIN_1 D3
#define RELAYPIN_2 D0

struct RelayInfo {
  String deviceId;
  String name;
  int pin;
};

std::vector<RelayInfo> relays = {
  {"6221521c753dc5aab486153d", "Luz da Sala", RELAYPIN_1},
  {"62215296d0fd258c52ed81c6", "Ventilador do Teto", RELAYPIN_2},
/*
 * | | |
 * | | +----> GPIO
 * | +-----> Nome do dispositivo
 * +-----> ID do dispositivo
 *
 * Note: Some GPIO's are set to specific level when the board boots up
 * This might result in strange behavior if there are relays connected to those pins
 * Check your board documentation!
 */

//Configuração da fechadura
#define LOCK_ID "6233cd3bd0fd258c52fa4281" //
#define LOCK_PIN D0

```

```

//Definição de Wi-Fi e Token Sinric
#define WIFI_SSID      "Wifi"
#define WIFI_PASS      "12345678"
#define APP_KEY        "ad26d817-2c4f-41e9-a153-22b5c39da4d0"
#define APP_SECRET     "4b681562-e8b4-4894-9dd3-2116c54e13e3-65f6c36b-4347-4402-869b-06c7e7954d07"
#define BAUD_RATE      115200

//Subrotina de acionamento dos interruptores
bool onPowerState(const String &deviceId, bool &state)
{
  for (auto &relay : relays)
  {
    // Para cada relé configurado
    if (deviceId == relay.deviceId)
    { // check if deviceId matches

      Serial.printf("Device %s turned %s\r\n", relay.name.c_str(), state ? "on" : "off");
      // print nome do dispositivo e estado no monitor serial
      if(state)
      {
        digitalWrite(relay.pin, 0); // Estado da GPIO se comando = "ON"
      }
      else
      {
        digitalWrite(relay.pin, 1); // Estado da GPIO se comando = "OFF"
      }
      return true; // Sinal de Echo de sucesso
    }
  }
  return false; // Sinal de Echo de falha
}

//Subrotina de acionamento da fechadura
bool onLockState(String deviceId, bool &lockState)
{
  Serial.printf("Device %s is %s\r\n", "Portão", lockState?"locked":"unlocked");
  // print nome do dispositivo e estado no monitor serial
  digitalWrite(LOCK_PIN, 0);
  delay (800);
  digitalWrite(LOCK_PIN, 1);
  delay (500);
  digitalWrite(LOCK_PIN, 0);
  delay (800);
  digitalWrite(LOCK_PIN, 1);
  delay (500);
  return true;
}

```

```

void setupRelayPins()
{
  for (auto &relay : relays)
  {
    pinMode(relay.pin, OUTPUT);    // modo OUTPUT para todos os pinos
  }
}
void setupWiFi() //Subrotina de conexão com WI-Fi
{
  Serial.printf("\r\n[WiFi]: Conectando");
  WiFi.begin(WIFI_SSID, WIFI_PASS);

  while (WiFi.status() != WL_CONNECTED) {
    Serial.printf(".");
    delay(250);
  }
  Serial.printf("connected!\r\n[WiFi]: IP-Address is %s\r\n", WiFi.localIP().toString().c_str());
}
void setupSinricPro() //Subrotina de conexão com servidor Sinric
{
  SinricProLock &myLock = SinricPro[LOCK_ID];
  myLock.onLockState(onLockState);
  for (auto &relay : relays)
  {
    SinricProSwitch &mySwitch = SinricPro[relay.deviceId];
    mySwitch.onPowerState(onPowerState);
  }

  SinricPro.onConnected([]() { Serial.printf("Conectado ao SinricPro\r\n"); });
  SinricPro.onDisconnected([]() { Serial.printf("Desconectado do SinricPro\r\n"); });

  SinricPro.begin(APP_KEY, APP_SECRET);
}
void setup() {
  Serial.begin(BAUD_RATE);
  setupRelayPins();
  setupWiFi();
  setupSinricPro();
  digitalWrite(RELAYPIN_1,1);
  digitalWrite(RELAYPIN_2,1);
}
void loop() {
  SinricPro.handle();
}

```

---