

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA
E TECNOLOGIA DA PARAÍBA
COORDENAÇÃO DO CURSO SUPERIOR
DE TECNOLOGIA EM
AUTOMAÇÃO INDUSTRIAL



Júlio César Gonçalves de Sousa

PROTÓTIPO DE DOMÓTICA PARA AUTOMAÇÃO RESIDENCIAL

Orientador: João Pablo Santos da Silva

Coorientador: Caio Marco dos Santos Junqueira

Cajazeiras-PB

2021

Júlio César Gonçalves de Sousa

PROTÓTIPO DE DOMÓTICA PARA AUTOMAÇÃO RESIDENCIAL

Trabalho de Conclusão de Curso submetido à Coordenação do Curso de Tecnologia em Automação Industrial do Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, como parte dos requisitos para a obtenção do grau de Tecnólogo em Automação Industrial.

Orientador: João Pablo Santos da Silva

Coorientador: Caio Marco dos Santos Junqueira

Cajazeiras-PB

2021

Campus Cajazeiras
Coordenação de Biblioteca
Biblioteca Prof. Ribamar da Silva
Catalogação na fonte: Daniel Andrade CRB-15/593

S237p

Sousa, Júlio César Gonçalves de

Protótipo de domótica para automação residencial /Júlio César Gonçalves de Sousa; orientador João Pablo Santos da Silva; coorientador Caio Marco dos Santos Junqueira.- 2021.

70 f. : il.

Orientador: João Pablo Santos da Silva.

TCC(Tecnologia em Automação Industrial) – Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, Cajazeiras, 2021.

1. Automação Residencial 2. Domótica 3. Arduino 4.
Reconhecimentode voz. I. Título

CDU 681.5(0.067)

PROTÓTIPO DE DOMÓTICA PARA AUTOMAÇÃO RESIDENCIAL

Trabalho de Conclusão de Curso submetido à Coordenação do Curso de Tecnologia em Automação Industrial do Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, como parte dos requisitos para a obtenção do grau de Tecnólogo em Automação Industrial.

BANCA EXAMINADORA

João Pablo Santos da Silva

João Pablo Santos da Silva - (Orientador)
Unidade Acadêmica de Indústria do IFPB

Caio Marco dos Santos Junqueira

Caio Marco dos Santos Junqueira - (Coorientador)
Unidade Acadêmica de Indústria do IFPB

Marco D. de Sousa.

Marco Damasceno de Sousa - (Membro)
Unidade Acadêmica de Indústria do IFPB.

Cajazeiras, 27 de agosto de 2021

DEDICATÓRIA

Dedico este trabalho aos meus pais, em especial pela dedicação e apoio em todos os momentos difíceis.

AGRADECIMENTOS

Primeiramente a Deus por tornar tudo possível e ser essencial em nossas vidas.

Ao Instituto Federal da Paraíba, IFPB, pela oportunidade de realização de trabalhos em minha área de pesquisa.

Aos colegas do IFPB pelo seu auxílio nas tarefas desenvolvidas durante o curso e apoio na revisão deste trabalho.

RESUMO

Este trabalho apresenta um sistema de automação residencial chamado de domótica, ou mais precisamente casa automática, no qual foi testado o controle dos aparelhos elétricos residenciais de forma automática. O trabalho foi desenvolvido com o intuito de demonstrar o funcionamento da automação residencial, buscando dar início a uma tecnologia que, em breve, estará disponível à maioria da população. O trabalho apresenta um protótipo de custos mais baixos em relação aos sistemas semelhantes existentes no mercado tecnológico. Para o desenvolvimento, foram utilizados componentes eletrônicos e módulos microcontroladores integrados que possuem funcionalidades distintas com o objetivo de: acender ou apagar lâmpadas, utilizando o reconhecimento de voz ou o *smartphone*; informar as horas, temperatura e data e; interagir com o usuário de acordo com o comando de voz, mostrando algumas informações no *display*. A descrição de cada componente será dada de acordo com os *datasheets*, manuais e os resultados dos testes, mostrando as suas funcionalidades no protótipo. A rotina de tarefas que o sistema irá executar provém da linguagem desenvolvida por meio da IDE do Arduino. No módulo chamado de NodeMCU, uma rotina separada foi feita, mas com o módulo controlando o mesmo acionamento utilizando a porta lógica XOR. Um esquema da instalação da residência foi construído mostrando a localização do protótipo de domótica, das lâmpadas e dos cômodos usados nos testes. Após a construção do sistema, foi realizada uma avaliação do custo total do dispositivo instalado. Em seguida, uma comparação com tecnologias semelhantes foi avaliada. Ao final, o protótipo apresentou um custo menor, tornando mais acessível a maioria das pessoas. Os resultados obtidos foram satisfatórios com o protótipo funcionando conforme todos os recursos citados.

Palavras-Chave: Automação Residencial, Domótica, Arduino, Reconhecimento de voz.

ABSTRACT

This work presents a home automation system called home automation, or more precisely automatic house, in which the control of residential electrical appliances was tested automatically. The work was developed in order to demonstrate the functioning of home automation, seeking to start a technology that will soon be available to the majority of the population. The work presents a prototype of lower costs in relation to similar systems existing in the technological market. For development, electronic components and integrated microcontroller modules were used that have different functionalities with the objective of: turning on or off light bulbs, using voice recognition or smartphones; inform the time, temperature and date and; interact with the user according to the voice command, showing some information on the display. The description of each component will be given according to datasheets, manuals and test results, showing its functionalities in the prototype. The routine of tasks that the system will execute comes from the language developed through the Arduino IDE. In the module called NodeMCU, a separate routine was made, but with the module controlling the same drive using the XOR logic gate. A schematic of the installation of the residence was constructed showing the location of the home automation prototype, the lamps and the rooms used in the tests. After the construction of the system, an evaluation of the total cost of the installed device was carried out. Then a comparison with similar technologies was evaluated. In the end, the prototype had a lower cost, making it more accessible to most people. The results obtained were satisfactory with the prototype working according to all the resources mentioned.

Keywords: Home Automation, Home Automation, Arduino, Voice Recognition.

LISTA DE FIGURAS

Figura 1: Exemplos de assistentes virtuais de voz disponíveis no mercado (Cosmin L., 2017).....	20
Figura 2: Esquema de montagem do projeto com shield EasyVR e arduino-1, módulo relé serial-2, motor de passo-3, drive do motor de passo-4 (ZAMBERLAN, BERNARDO CERVO, 2016)	23
Figura 3: Lâmpadas da esquerda para direita: Inteligente, LED, Florescente.....	24
Figura 4: Principais componentes do Arduino UNO (TECH SUL ELETRÔNICOS, 2021)	24
Figura 5: Representação do diagrama do relê (LUIZ, S. J. S.; FARINELLI, F. A.).....	25
Figura 6: Módulo VoiceRecognitionV3 (SHEN, Wilson).....	26
Figura 7: DFPlayMini disponível no mercado (FILIPEFLOP).....	26
Figura 8 Módulo RTC DS3231 (FILIPEFLOP)	27
Figura 9: ESP8266 NodeMCU V3 ESP12E (MERCADOLIVRE)	27
Figura 10: Abrindo a biblioteca do VoiceRecognitionV3 (IDE do arduino)	28
Figura 11: A biblioteca base (IDE do Arduino)	29
Figura 12: Como encontrar a biblioteca para treinar as vozes (print IDE do arduino).....	30
Figura 13: Demonstração da conexão dos pinos (editada do fritzing)	30
Figura 14: LED no momento de funcionamento normal a esquerda e no momento do treinamento dos comandos de voz a direita (autoria própria).....	31
Figura 15: Comando 0 treinado (print da IDE do arduino)	32
Figura 16: Comandos utilizados no exemplo “vr_sample_train” (print da IDE do arduino).....	32
Figura 17: Módulo relê de 4 canais usado no protótipo mostrando a parte de cima com conexões dos pinos do arduino enquanto a parte de baixo será ligado a fase e os retornos das lâmpadas (autoria própria).....	33
Figura 18: Esquema para testes com o módulo relê (print do fritzing)	34
Figura 19: DFPLAYER MINI com identificação de cada pino (autoria própria).....	34
Figura 20: Demonstração das conexões dos pinos para o projeto da domótica (print do fritzing)	35
Figura 21: Utilizando a biblioteca “FullFunction” para controle do módulo DFPlayerMini	36
Figura 22: Exemplo de vários comandos para controle do DF Player Mini (DFRobot)	36
Figura 23: Pastas enumeradas no cartão micro SD (autoria própria).....	37
Figura 24: Nomes dos áudios nas pastas do cartão de memória micro SD (autoria própria).....	37
Figura 25: Aplicativo de voz do narrador no smartphone	38
Figura 26: RTC DS 3231 mostrando seus pinos e a bateria de lithium (autoria própria).....	39
Figura 27: Encontrando a biblioteca exemplo do RTC 3231 que serão incluídos comandos para controle do módulo (Autoria própria)	39
Figura 28: Esquema de ligação dos pinos do RTC 3231 no arduino (print do fritzing)	40
Figura 29: Exemplo de como acionar relês por tempo programado (autoria própria)	40
Figura 30: Pasta 01 áudios dos números de 0 a 59 (autoria própria).....	41
Figura 31: Programação para reproduzir as horas usando áudios das pastas 01 e 03 (autoria própria)....	41
Figura 32: Pasta 03 áudios extras (autoria própria).....	42
Figura 33: Programação do comando de voz: “data” reproduzindo a data atual (autoria própria).....	42

Figura 34: Arquivos da pasta 05, dias da semana (autoria própria).....	43
Figura 35: Arquivos da pasta 04, meses do ano (autoria própria)	43
Figura 36: Arquivos da pasta 06, áudios dos anos de 2020 a 2030 (autoria própria)	43
Figura 37: Programação para reproduzir o som dos anos colocada no comando “data”	44
Figura 38: Ajuste do Módulo RTC 3231 (print da IDE)	44
Figura 39: Módulo Wi-fi Esp8266 NodeMcu usado no protótipo (Autoria própria).....	45
Figura 40: Pinagem do ESP8266 (FREITAS, 2019)	45
Figura 41: Encontrando a biblioteca para usar no NodeMcu (print da IDE)	46
Figura 42: Habilitando à placa para gravar a programação (print da IDE).....	46
Figura 43: Passos para enviar o “Auth Token” (autoria própria)	47
Figura 44: Enviando o “Auth Token” (Autoria Própria).....	47
Figura 45: Programação a ser carregada no NodeMcu após inserir o “AuthToken”, nome e senha do wifi usado na residência (print da IDE)	48
Figura 46: Puxando pela direita encontramos os itens e selecionemos o item “button”, moveremos 4 e renomearemos e configuraremos as saídas “D0” a “D3” iniciando como nível alto (autoria própria)	48
Figura 47: Ao alimentar o NodeMcu, podemos ver no aplicativo Blynk a confirmação de conexão e ao pressionar o play, podemos usá-lo (autoria própria).....	49
Figura 48: NodeMcu e Arduino Uno para acionamento do módulo relê com CI de XOR (print do fritzing).	50
Figura 49: (a) Circuito XOR utilizando as portas NOT, AND e OR (b) Circuito tradicional para porta XOR (TOCCI, Ronald J., 2011)	50
Figura 50 Esquemático de montagem com Arduino, esp8266 NODEMCU, HD74LS86P e Módulo Relê (Fritzing)	51
Figura 51: Representação esquemática do CI HD74LS86P (RENESAS) com o real utilizado no protótipo (autoria própria).....	52
Figura 52: Esquema de conexão dos pinos do display OLED SSD1306 no arduino	53
Figura 53 Tela OLED SSD1306 do protótipo (autoria própria).....	56
Figura 54: Relê após controlar com o aplicativo Blynk pelo smartphone (Autoria própria)	56
Figura 55: Exemplo de programação de acionamento por voz (autoria própria)	57
Figura 56: Instalação do circuito de iluminação na domótica (magicplan).....	58
Figura 57: Instalação da domótica, representação em 3D (montagem no magicplan).	59
Figura 58: Esquema completo do Protótipo (fritzing).....	60
Figura 59: Protótipo montado com e sem o suporte (autoria própria)	68

LISTA TABELAS

Tabela 1 - Tabela-verdade da porta lógica XOR (TOCCI, Ronald J., 2011)	51
Tabela 2 - Custo geral do protótipo da domótica.....	61
Tabela 3 - Comparação de preços com sistemas domóticos semelhantes (Autoria própria).....	62

LISTA DE ABREVIATURAS

ADC	<i>Analog Digital Converter;</i>
CI	<i>Circuito integrado;</i>
CLP	<i>Controlador lógico programável;</i>
FAT	<i>File Allocation Table;</i>
GPIO	<i>General Purpose Input / Output;</i>
GHz	<i>GigaHertz;</i>
GND	<i>Ground;</i>
I2C ou IIC	<i>Inter-Integrated Circuit;</i>
IDE	<i>Integrated Development Environment;</i>
IBGE	<i>Instituto Brasileiro de Geografia e Estatística;</i>
LED	<i>Light emitting diode;</i>
Mp3	<i>MPEG 1 Layer-3;</i>
MCU	<i>Micro Controller Unit</i>
OLED	<i>Organic Light-Emitting Diode;</i>
PWM	<i>Pulse-Width Modulation;</i>
PIC	<i>Programmable Interface Controller;</i>
RX	<i>Receive;</i>
RTC	<i>Real Time Clock;</i>
SD	<i>Secure Digital;</i>
SCL ou SCK	<i>Serial Clock;</i>
SDA	<i>Serial Data;</i>
SPK	<i>Speaker;</i>
TF	<i>TransFlash;</i>
TX	<i>Transmit;</i>
TTL	<i>Transistor-transistor logic;</i>
UART	<i>Universal Asynchronous Receiver / Transmitter;</i>
USB	<i>Universal Serial Bus;</i>
CPU	<i>Central Processing Unit;</i>
VCC	<i>Tensão em Corrente Contínua;</i>
VIN	<i>Tensão de entrada;</i>
WI-FI	<i>Wireless Fidelity;</i>
WPA / WPA2	<i>Wi-Fi Protected Access.</i>

LISTA DE SÍMBOLOS

\oplus - Porta lógica ou-exclusivo (xor).

Ω - Ohm (ômega).

SUMÁRIO

1. INTRODUÇÃO	14
1.1 PROBLEMA DE PESQUISA	15
1.2 OBJETIVOS	15
1.2.1 Geral	15
1.2.2 Específicos.....	15
1.3 JUSTIFICATIVA	15
2. REVISÃO BIBLIOGRÁFICA	17
2.1 HISTÓRICO DA DOMÓTICA	17
2.2 DEFINIÇÃO DE DOMÓTICA.....	17
2.3 APLICAÇÕES DA DOMÓTICA	18
2.4 CLASSIFICAÇÃO DA AUTOMAÇÃO RESIDENCIAL.....	19
2.4.1 Autônomos	19
2.4.2 Integrados	19
2.4.3 Complexos.....	19
2.5 ASSISTENTES VIRTUAIS.....	19
2.6 DOMÓTICA APLICADA A IDOSOS E INCAPACITADOS	20
2.7 SISTEMAS DE AUTOMAÇÃO RESIDENCIAL DE BAIXO CUSTO	21
2.7.1 Acionamento por controle remoto.....	21
2.7.2 Acionamento por controle de voz.....	21
2.7.3 Sistemas de iluminação	23
2.7.4 Arduino UNO	24
2.7.5 Relê.....	25

2.7.6 Módulo VoiceRecognitionV3.....	25
2.7.7 Módulo DFPlayerMini	26
2.7.8 Módulo RTC DS3231.....	26
2.7.9 Módulo NodeMcu 1.0 ESP8266 ESP12E	27
3. METODOLOGIA.....	28
3.1 DOMÓTICA UTILIZANDO O MICROCONTROLADOR ARDUINO UNO.....	28
3.2 USANDO O VOICE RECOGNITION V3	29
3.2.1 Treinando as Vozes	31
3.3 FUNCIONAMENTO DOS RELÊS	33
3.4 USANDO O DFPLAYER MINI.....	34
3.4.1 Programa para funcionamento do DFPlayer Mini.....	35
3.5 USANDO O RTC (REAL TIME CLOCK)	38
3.5.1 Método para reprodução de áudios.....	40
3.6 USANDO O NODEMCU 1.0 ESP8266 ESP12E	45
3.7 HD74LS86P-XOR (OU-EXCLUSIVO)	50
3.8 USANDO O DISPLAY OLED SSD1306.....	52
3.8.1 Principais Especificações	52
3.9 COMPONENTES EXTRAS	53
4. ANÁLISE DOS RESULTADOS	55
4.1 FUNCIONAMENTO DO DISPLAY OLED SSD1306.....	56
4.2 FUNCIONAMENTO DO ESP8266 EM CONJUNTO COM O BLINK	56
4.3 FUNCIONAMENTO COM O CI-HD74LS86P-XOR.....	57
4.4 ESQUEMAS DE INSTALAÇÃO E PROGRAMAÇÃO	58
4.4.1 Instalação na Residência.....	58

4.4.2 Instalação dos Componentes do Protótipo de Domótica.....	59
4.4.3 Código de Programação.....	60
4.5 LEVANTAMENTO E COMPARAÇÃO DE CUSTOS.....	60
5. CONCLUSÃO.....	63
REFERÊNCIAS	64
APÊNDICE A – IMAGEM DO PROTÓTIPO.....	68

1. INTRODUÇÃO

Nos últimos anos, as novas tecnologias vêm ganhando cada vez mais espaço no ambiente residencial para as diversas classes sociais. Essas tecnologias facilitam a vida das pessoas, principalmente as portadoras de necessidades especiais com dificuldades físicas de deslocamento. Uma das formas tecnológicas de dar conforto no seu lar é por meio da automação residencial.

Dentre elas, destaca-se a domótica, um termo originado da Europa das palavras “Domus” (casa) e “imótica” (automática). Ou seja, a domótica é uma possibilidade de controle de forma automática de residências, que pode também ser conhecida por casas inteligentes (ALVES, J. A.; MOTA, J., 2003).

Uma domótica realiza funções lógicas de acordo com condições pré-estabelecidas. Em uma residência, por exemplo, utiliza os cenários de iluminação, climatização, som ambiente, reconhecimento de voz, vigilância etc., para satisfazer as preferências de um determinado usuário (ALVES, J. A.; MOTA, J., 2003).

Neste contexto, será proposto neste trabalho a criação de um protótipo de uma domótica inteligente de baixo custo com o objetivo de acrescentar mais modernidade, qualidade de vida, comodidade, agilidade, economia e segurança no ambiente residencial. O protótipo será composto por um conjunto de módulos e componentes eletrônicos que estarão conectados por meio de *jumpers*. O microcontrolador Arduino UNO funcionará como controle central, porém, será controlado por um módulo de reconhecimento de voz. Outro módulo que funcionará de forma paralela com o sistema já mencionado, usará a internet sem-fio (*WI-FI*) para controle da residência por smartphones. Dentre as funções que o protótipo irá executar, destacam-se os acionamentos de aparelhos eletrodomésticos, lâmpadas e demais aparelhos eletrônicos.

A tecnologia apresenta resultados semelhantes na atualidade com os chamados assistentes virtuais em produtos de grandes empresas famosas como: *Google Assistant* da *Google*, *Alexa* da *Amazon*, *Siri* da *Apple*, *Cortana* da *Microsoft* entre outras (COSMIN, L.). Contudo, somente a *Alexa* da *Amazon* possui a aplicação na automação residencial atuando em conjunto com aparelhos inteligentes.

Dentre as vantagens para criação desse protótipo em comparação aos assistentes virtuais existentes, podem-se citar: menor preço; maiores funcionalidades; simplicidade para modificação do funcionamento; capacidade de se integrar a diferentes módulos; maior aplicabilidade; capacidade de aplicar-se a diferentes formas de acionamento e; independência de internet para o funcionamento.

O projeto foi desenvolvido por meio da utilização dos conhecimentos e experiências adquiridos durante o curso de Automação Industrial. Desta forma, permitirá ao proponente deste trabalho de Conclusão de Curso as ferramentas necessárias para a construção de um produto que poderá ser fabricado em escala local, atendendo a região próxima à cidade de Cajazeiras.

1.1 PROBLEMA DE PESQUISA

O uso de uma domótica automática é fundamental para o conforto e agilidade no controle residencial e requer componentes eficazes com diversas funcionalidades, a maioria deles possui custo muito elevado. Diante disso como criar um dispositivo domótico automático de baixo custo utilizando microcontroladores e componentes eletrônicos?

1.2 OBJETIVOS

1.2.1 Geral

Projetar um protótipo de domótica automática de baixo custo para integração de sistemas residenciais. Verificar se é possível uma comercialização do dispositivo.

1.2.2 Específicos

- Projetar a estrutura do protótipo da domótica;
- Identificar os equipamentos de baixo custo que serão adicionados ao protótipo;
- Elaborar comandos de voz para serem treinados;
- Elaborar a rotina de programação com os comandos de voz;
- Elaborar arquivos de áudio em formato de Mp3 para respostas sonoras.
- Elaborar quais dispositivos serão controlados;
- Comparar o dispositivo desenvolvido com dispositivos comerciais;
- Analisar as possibilidades de redução de custo do projeto;
- Desenvolver uma interface mais amigável com o usuário;
- Desenvolver geração de pequenas informações para o usuário;
- Facilitar as tarefas diárias das residências;
- Trazer conforto, segurança e modernidade.

1.3 JUSTIFICATIVA

A ideia do desenvolvimento do protótipo se deu devido a necessidade de implantar um dispositivo tecnológico com baixo custo em uma região onde ocorre crescimento regular de

residências, favorecendo indivíduos que possuem dificuldades físicas de locomoção na vida habitacional e melhorando o dia-a-dia da população local nos sentidos de saúde mental e física, trazendo mais conforto, lazer e satisfação. Outra justificativa para a construção do protótipo é a economia, já que no dispositivo doméstico também pode ser programado o horário de acionamento de: lâmpadas, ar condicionado, irrigação, etc., podendo gerar uma gestão energética.

2. REVISÃO BIBLIOGRÁFICA

2.1 HISTÓRICO DA DOMÓTICA

Os sistemas Domóticos, atualmente vem ganhando popularidade como casas inteligentes (*Smart Home*). Porém, essa tecnologia teve suas primeiras ideias em 1894 com vários experimentos realizados por um cientista chamado Nikola Tesla que realizou testes que resultaram na origem do primeiro controle remoto em 1898 ao conseguir, por meio do sinal de rádio, controlar um barco (LUIZ, S. J. S.; FARINELLI, F. A.). Mais tarde essa tecnologia seria usada como referência em conjunto com várias outras com desenvolvimento da indústria informacional.

Em meados dos anos 70, com o desenvolvimento da automação industrial surgiu a ideia de levar essa tecnologia às residências, mas inicialmente foi implantada em prédios comerciais. Contudo, seus custos de instalação eram muitos elevados (ADMINWEBBIZZ). Na década de 1980, período de surgimento dos primeiros computadores pessoais, eles eram vistos como alternativas para controle centralizado da automação das indústrias, mas possuíam limitações: não podiam ficar ligados por longos períodos e sempre ocorriam problemas que influenciavam todo sistema industrial (VIANNA, Gabriel Pereira). Mais tarde, na década de 1990, o computador deixou de ser usado exclusivamente na indústria e passou a ser usado como ferramenta de lazer, com pouco proveito de seu uso em virtude de não possuir acesso à rede (DOMINGUES, Ricardo Gil).

No decorrer dos anos seguintes, várias tecnologias foram surgindo e implementadas na automação residencial com muitas limitações e custos cada vez mais elevados. Só a partir do ano de 2000, com o avanço tecnológico e surgimento da internet banda larga e dos primeiros *smartphones* que essa ideia de casas inteligentes ganhou força e se tornou cada vez mais comum nos dias atuais.

2.2 DEFINIÇÃO DE DOMÓTICA

A domótica, termo originário da Europa, é definida como casa automática ou mais especificamente controle automatizado de algo no ambiente doméstico. Outros termos que melhor definem a domótica são: o controle de forma centralizada e/ou remota e o controle de forma autônoma (LUIZ, S. J. S.; FARINELLI, F. A.). Essas definições foram possíveis devido ao grande número de tecnologias principalmente da área de eletrônica e telecomunicações presentes nos dias de hoje.

Uma característica da domótica é ser ágil, devido ao fato de possuir capacidade de atualização, modificação dos sistemas, dispositivos e funcionalidades.

Os sistemas domóticos podem ser divididos em 3 partes: sensores, atuadores e controladores.

- a) Sensores: dispositivos usados no sensoriamento e monitoramento do ambiente (LUIZ, S. J. S.; FARINELLI, F. A.). Ex.: sensores de temperatura, umidade, presença, luminosidade, infravermelho etc.
- b) Atuadores: dispositivos usados para executar alguma ação no ambiente residencial (LUIZ, S. J. S.; FARINELLI, F. A.). Ex.: Relés, aparelhos eletrodomésticos inteligentes etc.
- c) Controladores: dispositivos que recebem informações dos sensores e/ou usuários e controlam os atuadores de acordo com a rotina de programação criada (LUIZ, S. J. S.; FARINELLI, F. A.). Ex.: Microcontroladores, CLP's, PIC, sistemas embarcados etc.

2.3 APLICAÇÕES DA DOMÓTICA

No setor residencial a domótica possui várias aplicações dependendo da necessidade e escolha do usuário que são:

- a) Sistema de segurança: câmeras, alarmes, controles de acesso como: reconhecimentos de voz, leitor de impressão digital, reconhecimento facial, travas automáticas das portas etc (VIANNA, Gabriel Pereira).
- b) Sistemas de multimídia: música, áudios de auxílio e assistência (VIANNA, Gabriel Pereira).
- c) Sistemas ligados a atuadores: iluminação, abri ou fechar: portões, portas, cortinas persianas (VIANNA, Gabriel Pereira).
- d) Sistemas de comunicação: controle de tv, telefonia, pesquisa de internet, rádio AM ou FM (VIANNA, Gabriel Pereira).
- e) Sistemas utilitários: limpeza, climatização, irrigação de jardim ou plantas, abastecimento de água em reservatórios etc (VIANNA, Gabriel Pereira).

2.4 CLASSIFICAÇÃO DA AUTOMAÇÃO RESIDENCIAL

Podemos classificá-la em três sistemas: autônomos, integrados e complexos.

2.4.1 Autônomos

São sistemas de automação residencial que possuem o funcionamento independente, no qual não são controlados pelo usuário para executarem as tarefas domésticas. Eles são pré-programados para determinadas funções periódicas ou de acordo com dados recebidos de sensores como, por exemplo, um sistema de climatização automática, sistema de irrigação, sistema de incêndios etc (VIANNA, Gabriel Pereira).

2.4.2 Integrados

São sistemas que possuem funcionamento individual que se comunicam entre si e são controláveis por inteligência central.

2.4.3 Complexos

Sistemas com maiores níveis de interatividade com o usuário de acordo com sua necessidade, no qual todos os sistemas e subsistemas da residência são personalizados, integrados, controlados e gerenciados via internet, *bluetooth*, infravermelho, radiofrequência, computador, *smartphone*, controle remoto etc.

2.5 ASSISTENTES VIRTUAIS

Trata-se de *softwares* baseados em inteligência artificial programados para escutar comandos de voz ou textos de um determinado usuário e responder de forma que se torne mais próxima aos contatos humanos. Alguns se comunicam com “aparelhos inteligentes” enviando comandos para que sejam executados pedidos feitos pelo usuário. Alguns exemplos de comandos são: acender ou apagar luzes; ligar, desligar ou controlar aparelhos como: ar-condicionado, cafeteira, tv, aparelho de som; trancar/destrancar janelas ou portas; realizar pesquisas na internet etc.

Diferentemente de alguns dispositivos de baixo custo, os assistentes de voz fabricados por grandes empresas tecnológicas (Figura 1) não necessitam de treinamento da voz do usuário, tornando independente de treinamento. Isso acontece devido ao fato de a tecnologia já ter mapeado um número muito grande de vozes de usuários pelo mundo.



Figura 1: Exemplos de assistentes virtuais de voz disponíveis no mercado (Cosmin L., 2017).

2.6 DOMÓTICA APLICADA A IDOSOS E INCAPACITADOS

Os vários benefícios das domóticas como: maior comodidade, segurança, economia energética, bem-estar social, autonomia na execução de tarefas domésticas, entre outros, são de extrema importância no suporte às pessoas idosas, com incapacidades físicas ou dificuldades nas tarefas diárias. Segundo o IBGE (Instituto Brasileiro de Geografia Estatística) utilizando dados do censo de 2010, cerca de 8% da população era composta por idosos e 24% possuíam alguma incapacidade ou deficiência, o que gera maior importância do uso da domótica para esse público (DOMINGUES, Ricardo Gil).

A domótica pode ser aplicada na utilização de dispositivos de vigilância como câmeras e sensores de som para monitorar o sono e o estado de locomoção (DOMINGUES, Ricardo Gil). Nesses casos alarmes podem alertar os responsáveis ou proprietários a tomar providências em ações emergenciais ou anormais de acordo com a detecção dos dispositivos de monitoramento (DOMINGUES, Ricardo Gil).

Nos casos de pessoas incapacitadas com dificuldades de locomoção, a tecnologia pode acender ou apagar lâmpadas, ligar ou desligar aparelhos de som, fazer ligações, sem precisar sair do local de repouso e sem ajuda de terceiros, utilizando a voz, gestos, *smartphones* e *tablets* (DOMINGUES, Ricardo Gil). Para pessoas com deficiência visual, o áudio informaria o estado de um aparelho e o comando de voz garantiria seu controle.

Apesar de muitos benefícios da domótica, nos casos de acessibilidade a cômodos da residência, os aparelhos de locomoção possuem custos muito elevados, sendo utilizada quase que exclusivamente pelas classes mais ricas.

2.7 SISTEMAS DE AUTOMAÇÃO RESIDENCIAL DE BAIXO CUSTO

2.7.1 Acionamento por controle remoto

Dentre os objetivos do uso de controle remoto está a redução do esforço e acessibilidade sem sair do lugar.

Existem vários tipos de controle de forma remota: os que possuem infravermelho como controles de tv; os de radiofrequência, no qual tem-se exemplos como *bluetooth* e *wi-fi*. Os mais usados nas domóticas são os de radiofrequência, já que a maioria das pessoas estão constantemente usando aparelhos celulares ou *smartphones*.

O acionamento de relés utilizando controle remoto via infravermelho, os quais operam de 36 kHz a 38 kHz, tem o seguinte funcionamento: ao apertar qualquer tecla de um controle remoto simples, um sensor infravermelho “TSOP2236” fornece pulsos para um “CI 555 monoestável”; o sinal de sua saída é modificado, normalmente de nível baixo para nível alto e direcionado diretamente para um relé que ativa ou desativa um determinado aparelho da residência (CABRAL, M. M. A.; CAMPOS, A. L. P. de S.). A vantagem desse sistema é o número ilimitado de pessoas que poderia controlar a residência sem necessidade de internet. Sua desvantagem é que somente funciona no cômodo que o sistema se encontra. Esse sistema poderia ser integrado ao protótipo de domótica em casos de falta de internet. Outras vantagens são: evita deslocamentos afim de acender ou apagar lâmpadas; evita quedas ou esbarrões em objetos nos casos que o ambiente estiver escurecido ou com pouca visibilidade; ganho de tempo útil do usuário que esteja ocupado em alguma tarefa na residência; controle de aparelhos elétricos que não possuem controle remoto; diminuição de distrações e interrupções em alguns afazeres domésticos etc.

2.7.2 Acionamento por controle de voz

Os sistemas de reconhecimento de voz ou comandados por voz estão em constantes melhoria. Alguns dispositivos baseados em ditados necessitam de um pré-treinamento para que o sistema reconheça a voz do usuário. Nessa etapa, existem certos obstáculos que dificultam o mapeamento da voz a ser treinada como:

- a) Ruídos: barulhos, sons de música alta, animais etc.
- b) Ecos: sons refletidos que chegam ao microfone do dispositivo de reconhecimento de voz.
Ocorrem se o treinamento acontecer em lugares fechados e apertados, perto de objetos

grandes ou em cantos de parede.

- c) Falta de nitidez: neste momento a nitidez deve ser precisa, ou seja, o som deve ser treinado de perto para maior qualidade e não pode ser falada com tom baixo e muito rápida.

O funcionamento de um dispositivo de reconhecimento de voz se dá pelo fato das vibrações sonoras, no momento que o usuário fala, serem captadas e convertidas em sinais de ondas analógicas ou frequências, para serem lidas como dados digitais pela CPU do dispositivo (LUZ, J. G.; FRESSATTI, W). Assim, cada comando de voz treinado é diferenciado por critérios de intensidade, frequência, timbre, velocidade e distância.

Existe no mercado um dispositivo chamado “Shield EasyVR”, no qual pode trabalhar em conjunto com o Arduino. Ele utiliza um *software* “EasyVR Commander” disponibilizado pelo fabricante (ZAMBERLAN). Seu funcionamento é dividido em cinco partes:

- a) *Trigger* – grupo de comando inicial que após ser chamado inicia os próximos comandos de reconhecimentos de voz do *shield*, nenhum comando pode ser chamado antes desse (ZAMBERLAN).
- b) *Group* – grupo formado por 15 de comandos de voz que são usados após o comando *trigger*. É nesse grupo que são gravados os comandos que ativam lâmpadas, abrem ou fecham portas etc (ZAMBERLAN).
- c) *WordSet* – grupo de palavras, ensinadas pelos usuários, que serão utilizadas pelo dispositivo (ZAMBERLAN).
- d) *Password* – grupo que funciona como medida de segurança, no qual é criado uma senha para acesso aos demais comandos, restringindo o uso do dispositivo a pessoas não autorizadas (ZAMBERLAN).
- e) *SoundTable* – grupo de saída de áudio em que são acoplados os alto-falantes à *shield* que reproduzirá sons em resposta a comandos falados pelos usuários (ZAMBERLAN).

Um projeto utilizando o “Shield EasyVR” acoplado no arduino, junto com um módulo relé serial e um motor de passo, é capaz de abrir ou fechar portas, ligar ou desligar lâmpadas e diversos outros aparelhos da residência que o usuário assim desejar, pelo comando de voz (ZAMBERLAN). Seu número de comandos é reduzido, devido ao fato de que para acionar os dispositivos da residência com informações diferenciadas, como pedir para tocar músicas,

umentar ou diminuir volume, informar o tempo, orientar o usuário sobre a data etc., é necessário um número maior de comandos. Uma vantagem desse dispositivo é o menor número de *jumpers* devido ao barramento ser serial, sendo os dados enviados ao módulo de relés para serem transmitidos e recebidos por duas conexões. Abaixo na Figura 2 é apresentado o esquema de instalação.

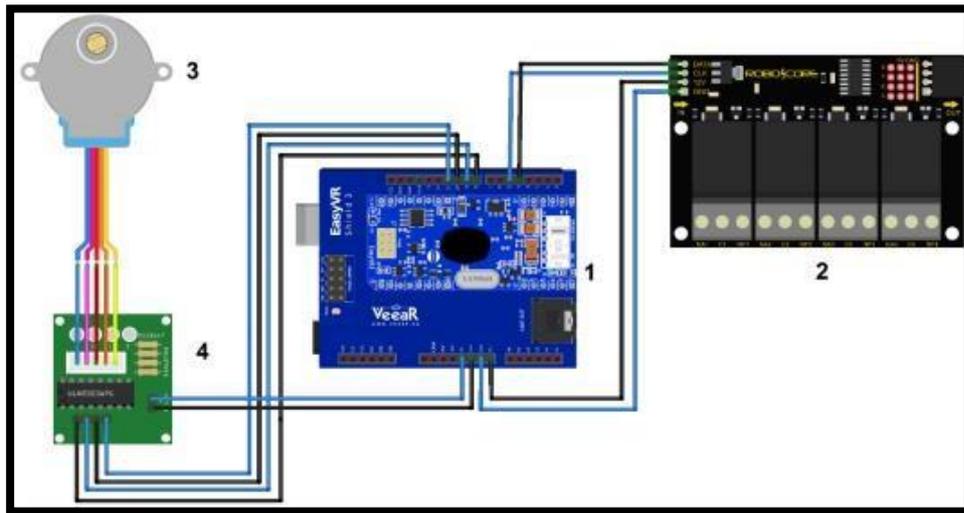


Figura 2: Esquema de montagem do projeto com *shield EasyVR* e *arduino-1*, módulo relé serial-2, motor de passo-3, drive do motor de passo-4 (ZAMBERLAN, BERNARDO CERVO, 2016).

2.7.3 Sistemas de iluminação

Existem dois tipos de sistemas de iluminação: os de lâmpadas inteligentes e as controladas por dispositivos controladores.

As lâmpadas inteligentes possuem funções de controle de intensidade, liga e desliga de forma automática ou controlada (SOUZA, U. R. dos S.; GOMES, F. F. B.). A sua forma de controle está relacionada a tecnologia sem fio por aplicativos de *smartphones*, porém, possuem custos altos se comparadas as lâmpadas comuns.

As lâmpadas comuns podem ser acionadas por atuadores que são ordenados pelos controladores e utilizam a rede elétrica da residência para os acionamentos. Os custos desse tipo de sistema são mais baixos se comparados as lâmpadas inteligentes, mas requerem uma instalação elétrica extra para integrar ao sistema domótico. São exemplos de lâmpadas comuns as incandescentes, fluorescentes e de LED, embora as lâmpadas incandescentes estejam saindo do mercado. Abaixo na Figura 3, observam-se diferentes lâmpadas no mercado atual.



Figura 3: Lâmpadas da esquerda para direita: Inteligente, LED, Florescente.

Fonte: amazon.com.br - kitled.com.br

2.7.4 Arduino UNO

A placa microcontrolada ou plataforma eletrônica de código aberto, chamada arduino, foi criada em 2005 por cinco pesquisadores: Tom Igoe, David Cuartielles, David Mellis, Massimo Banzi e Gianluca Martino (THOMSEN, Adilson). Porém, o modelo mais popular usado atualmente, o arduino UNO, foi lançado em 2010. A ideia era criar um dispositivo mais barato, de fácil programação e funcional (COSTA, Fabio.). Inicialmente, era fabricado para fins estudantis e projetistas amadores, os quais criaram o conceito de *hardware* livre, o que demonstrava que qualquer indivíduo poderia modificar e personalizar um código livremente (THOMSEN, Adilson). Na Figura 4, destaca-se o nome das principais entradas, saídas e alguns componentes.

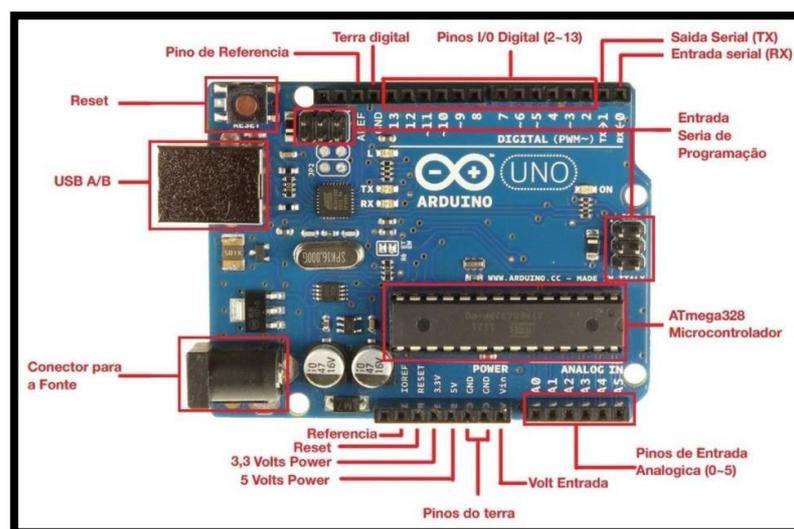


Figura 4: Principais componentes do Arduino UNO (TECH SUL ELETRÔNICOS, 2021).

Hoje em dia, ele é utilizado para fabricação de diversos dispositivos relacionados à automação industrial e residencial, projetos escolares, e referencial para sistemas de baixo custo. Ele possui seu próprio ambiente de desenvolvimento integrado (IDE), no qual é criada a programação utilizada para o seu funcionamento.

2.7.5 Relê

Os relês são interruptores eletromecânicos que são usados para ligar ou desligar dispositivos que estejam conectados à rede elétrica da residência. Seu funcionamento ocorre quando uma corrente circula pela bobina interna fazendo com que surja um campo magnético atraindo um contato ou uma série de contatos que fecha ou abre circuitos. Quando a corrente deixa de circular, o campo magnético é desativado e os contatos voltam ao estado inicial conforme a figura 5 abaixo (MULTILÓGICA SHOP).

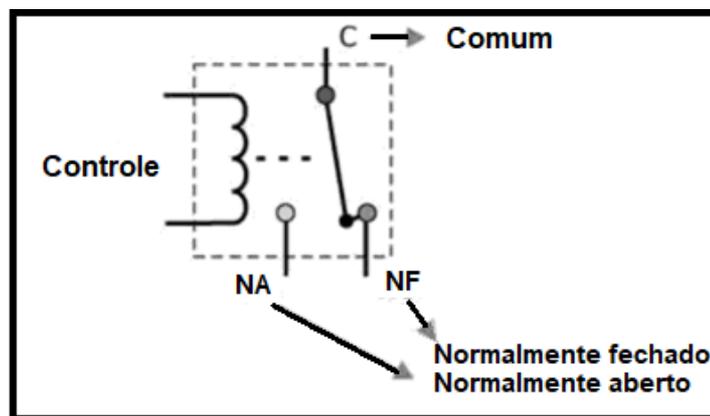


Figura 5: Representação do diagrama do relê (LUIZ, S. J. S.; FARINELLI, F. A.).

2.7.6 Módulo VoiceRecognitionV3

Define-se como uma placa de reconhecimento de voz criada pela empresa *ELECHOUSE*, podendo opcionalmente funcionar sem necessidade de arduino, porém para ser programada é dependente do mesmo. Ela suporta até 80 comandos curtos de voz, sendo que 7 funcionam ao mesmo tempo, ou seja, o usuário da domótica poderá alternar grupos de 7 dos 80 comandos disponíveis caso queira usar mais de 7 (SHEN, Wilson). A placa é demonstrada na figura 6 abaixo.

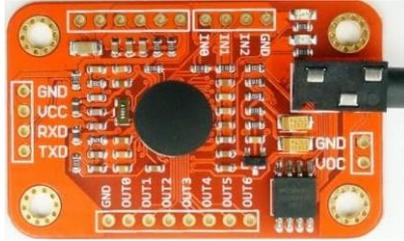


Figura 6: Módulo VoiceRecognition V3 (SHEN, Wilson).

A placa possui duas formas de controle: a primeira é através da porta serial e a segunda são os pinos de entrada geral. Ela funciona em tensões entre 4,5 a 5,5 V em correntes contínuas inferiores a 40mA. Sua precisão de reconhecimento é cerca de 99 % (em ambiente ideal) para ambientes sem muitos barulhos (SHEN, Wilson).

2.7.7 Módulo DFPlayerMini

É um módulo que funciona com 3,2V a 5,5V, leitor de cartão micro *Secure Digital* (SD) ou *TransFlash* (TF) com suporte de no máximo 32 Gb (*gigabytes*) de armazenamento. Os dados de áudio são classificados por pasta, com suporte de até 100 pastas. Cada pasta pode conter até 255 áudios. O volume é ajustável a 30 níveis e a equalização ajustável de 6 níveis. Ele fornece a decodificação de hardware *MP3* e *WMV* e os sistemas de arquivos suportados são *FAT16* e *FAT32* (AMPERKA). Na figura 7 abaixo observa-se o módulo mencionado.

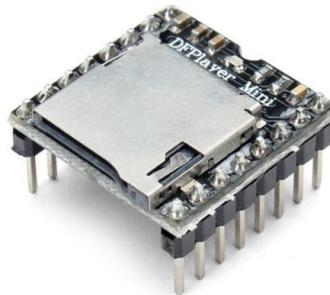


Figura 7: DFPlayMini disponível no mercado (FILIPEFLOP).

2.7.8 Módulo RTC DS3231

O relógio em tempo real (RTC) é um dispositivo de baixo custo que possui gerenciamento de funções de cronometragem, tem a função normal de um relógio em tempo real que conta horas, minutos, segundos, data do mês, mês, dia da semana e ano, com compensação de ano bissexto válida até 2100 (MAXIMINTEGRATED).

Ele é capaz de suportar um barramento I2C bidirecional e dados protocolo de transmissão, possui uma entrada de bateria de lithium que mantém os dados atualizados em caso de a alimentação ser interrompida, sua tensão de operação é de 3,3V a 5,5V DC, existe um sensor de temperatura de aproximadamente 3 °C de exatidão (MAXIMINTEGRATED). Observa-se na figura 8 abaixo um exemplo do modulo RTC.



Figura 8 Módulo RTC DS3231 (FILIFEFLOP).

2.7.9 Módulo NodeMcu 1.0 ESP8266 ESP12E

O módulo observado na figura 9, foi criado pela empresa Espressif Systems, é composto por um kit de desenvolvimento e *firmware* de código aberto integra GPIO, PWM, IIC, 1-Wire e ADC (ESPRESSIF INC). Possui entrada serial USB-TTL; a tensão da interface de comunicação é de 3,3V; *WiFi* a 2,4 GHz; suporte ao modo de segurança WPA / WPA2; sua entrada de energia por USB é entre 4,5V a 9V; suporte a interface de comunicação de dados UART / GPIO; opera a temperaturas entre -40°C a 125°C; seu flash é de 4 Megabyte (ESPRESSIF INC).



Figura 9: ESP8266 NodeMCU V3 ESP12E (MERCADOLIVRE).

3. METODOLOGIA

3.1 DOMÓTICA UTILIZANDO O MICROCONTROLADOR ARDUINO UNO

Na instalação da Domótica, a utilização do arduino foi fundamental para a execução do protótipo, nele foi gravado toda sequência de instruções a serem realizadas conforme o usuário desejar controlá-lo, seja por comando de voz ou via *wifi* com smartphone.

Para a criação da programação foram usadas, algumas bibliotecas criadas por fabricantes de alguns componentes integrados ao arduino. Os dados foram obtidos através de: *datasheets* ou manuais, busca na web e no gerenciador de bibliotecas da IDE do arduino.

O microcontrolador arduino UNO utilizará a comunicação serial para ser controlado pelo módulo de reconhecimento de voz chamado de “*VoiceRecognitionV3*” e controlará o módulo “*DF Player Mini*” usando a mesma comunicação. A comunicação I2C (circuito inter-integrado) será usada para o módulo “*RTC DS3231*” e para o módulo “*display OLED SSD1306*”. A placa de controle pelo smartphone foi o Esp8266 NodeMCU Esp12E que apesar de possuir a função de comunicação serial não foi integrada ao arduino, mas controla o mesmo aparelho da residência. Seguem os seguintes passos conforme a figura 10 abaixo, e então foi criada uma programação que melhor se adequava ao protótipo.

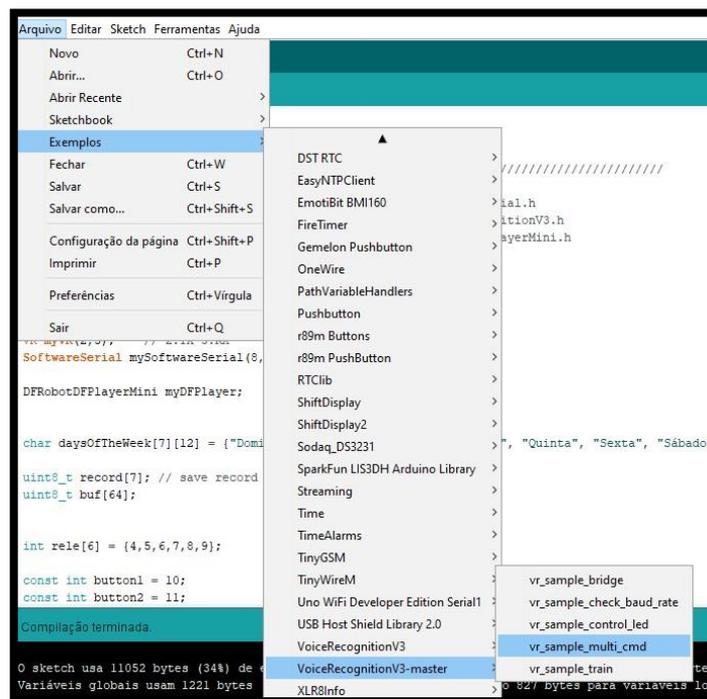
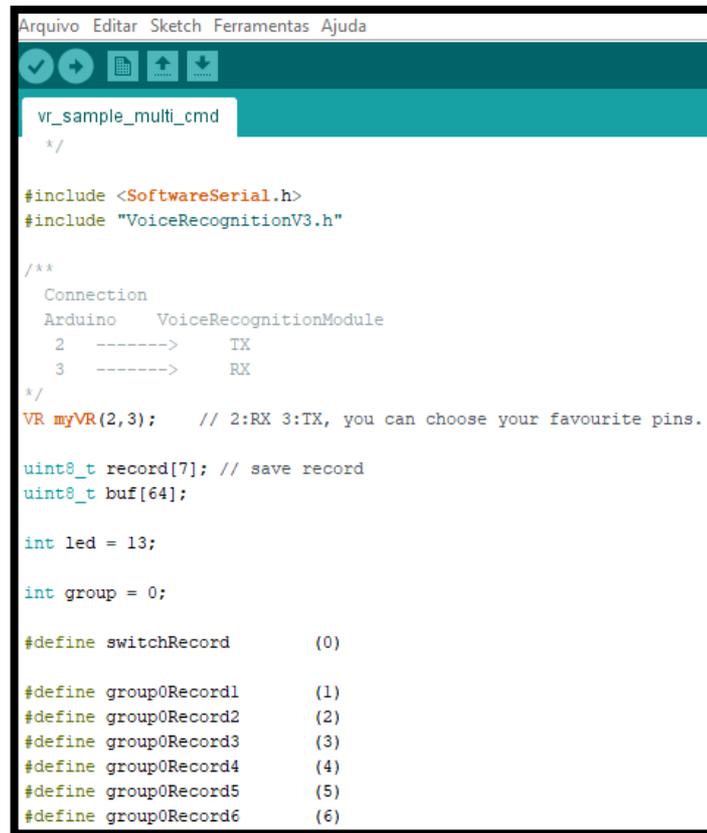


Figura 10: Abrindo a biblioteca do VoiceRecognitionV3 (IDE do arduino).

Observa-se a indicação dos pinos transmissor (TXD) e receptor (RXD) de dados da

placa de reconhecimento de voz para os pinos digitais 2 e 3 do arduino. São utilizadas sete variáveis definidas onde ficam armazenados os comandos treinados conforme a figura 11.



```

Arquivo  Editar  Sketch  Ferramentas  Ajuda
vr_sample_multi_cmd
*/
#include <SoftwareSerial.h>
#include "VoiceRecognitionV3.h"

/**
Connection
Arduino  VoiceRecognitionModule
2  ----->  TX
3  ----->  RX
*/
VR myVR(2,3); // 2:RX 3:TX, you can choose your favourite pins.

uint8_t record[7]; // save record
uint8_t buf[64];

int led = 13;

int group = 0;

#define switchRecord      (0)
#define group0Record1    (1)
#define group0Record2    (2)
#define group0Record3    (3)
#define group0Record4    (4)
#define group0Record5    (5)
#define group0Record6    (6)

```

Figura 11: A biblioteca base (IDE do Arduino).

3.2 USANDO O VOICE RECOGNITION V3

O reconhecimento da voz do usuário depende de treinamento para o determinado comando escolhido. Para isso na biblioteca disponível no *datasheet* do módulo *VoiceRecognitionV3* usar-se a opção de “vr_sample_train” encontrada na aba Arquivos e em exemplos (ELECHOUSE). Pode-se ver na figura 12 abaixo.

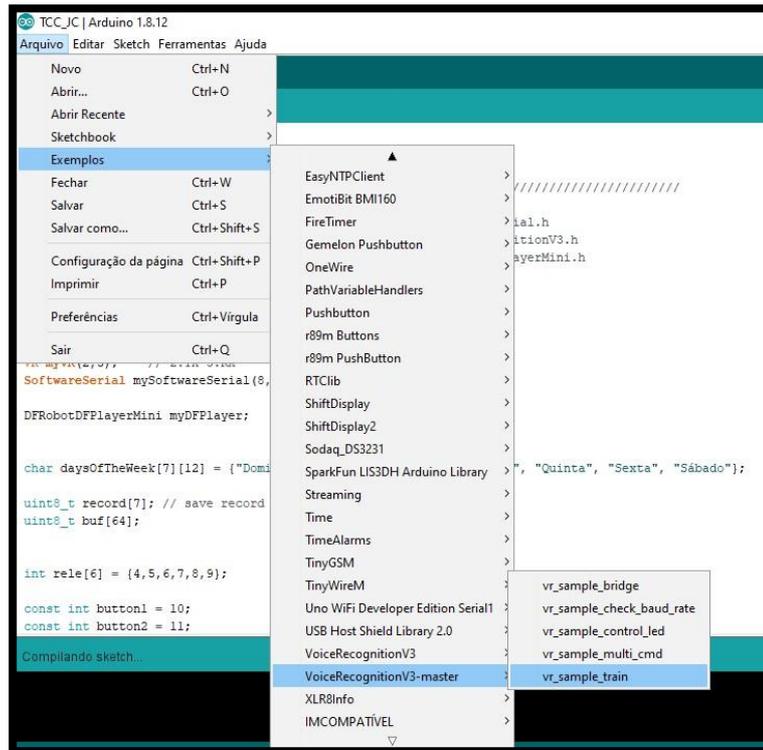


Figura 12: Como encontrar a biblioteca para treinar as vozes (print IDE do arduino).

Com o arduino integrado à placa *VoiceRecognitionV3* e conectado à entrada USB do computador, serão iniciados os testes de treinamento de vozes do usuário. Os pinos usados para transferências de dados seriais são respectivamente TXD e RXD do *VoiceRecognitionV3* conectados nos pinos digitais 2 e 3 do arduino UNO. Como o arduino está conectado ao computador pelo USB seus pinos 5V e GND alimentarão o módulo de reconhecimento de voz nos pinos VCC e GND como mostrado na figura 13 abaixo.

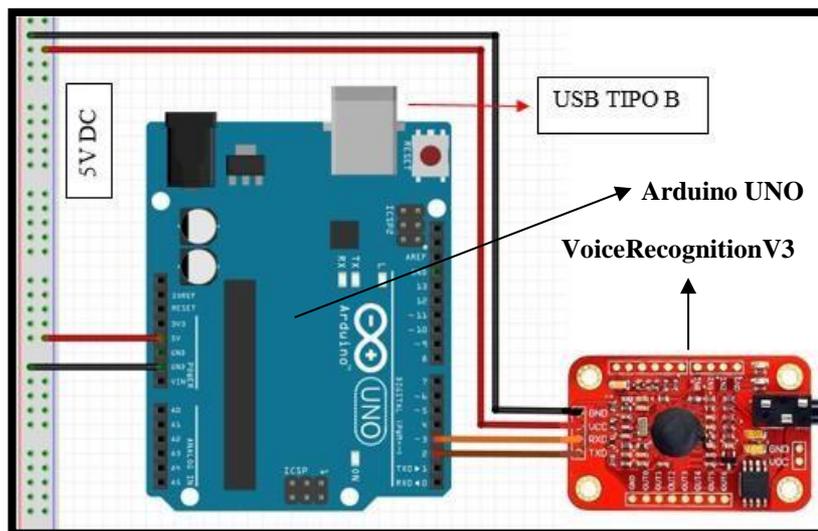


Figura 13: Demonstração da conexão dos pinos (editada do fritzing).

3.2.1 Treinando as Vozes

Ao carregar o código da biblioteca “vr_sample_train” no arduino, e abrir o monitor serial, localizado a direita representado como desenho de lupa, escreve-se o comando “train” mais o número correspondente aos 80 comandos para treinar. Como no protótipo será utilizado dois grupos de sete comandos cada, escolhe-se os números de 1 a 6 para o grupo 1 e de 7 a 12 para o grupo 2. O número 0 deverá ser o comando de alternar entre os grupos, ou seja, terá a função de substituir 6 comandos por outros 6 diferentes. Então ao digitarmos “train 0” e darmos ENTER no monitor serial, damos a ordem ao módulo de reconhecimento de voz para gravar a voz que será utilizada para ativar o comando 0. Geralmente o módulo reconhece em 2 tentativas de treinar a voz em condições ideais, ou seja, sem ruído ou barulho no ambiente de treinamento.

Observa-se na placa de reconhecimento de voz, figura 14, a indicação do LED caso esteja amarelo e piscando constantemente significa o funcionamento normal, após o ENTER em “train 0” o LED ficará vermelho, indicando o momento que deve ser falado. O usuário tem cerca de 3 segundos para falar e mais 3 para repetir a fala para que seja confirmado, ao ver o LED piscar e mudar a cor para amarelo significa que o reconhecimento de voz foi treinado com sucesso.

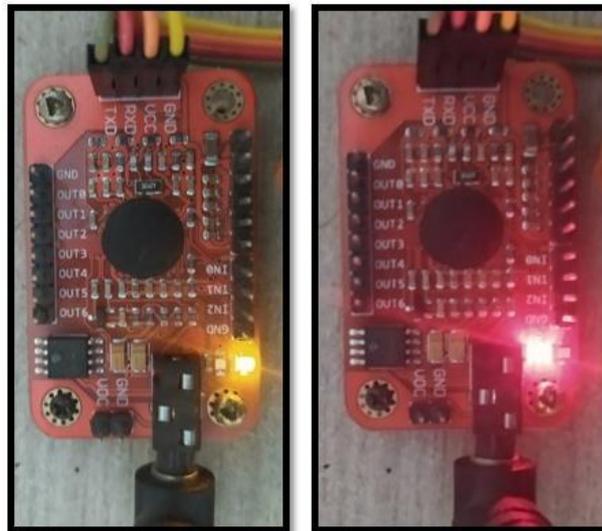


Figura 14: LED no momento de funcionamento normal a esquerda e no momento do treinamento dos comandos de voz a direita (autoria própria).

O comando deverá ser qualquer um, desde que dure cerca de 3 segundos. No protótipo foi usado “trocar” para o comando 0.

Após o registrar do comando 0, observa-se na Figura 15 a informação de confirmação

que a voz foi treinada com sucesso:

```

COM3
train 0
Enviar
vr          vr          vr          Check recognizer status
getsig      getsig (r)    getsig 0    Get signature of record
sigtrain    sigtrain (r) (sig)  sigtrain 0 ZERO  Train one record(r) with
settings    settings      settings    Check current system set
help        help          help        print this message
-----
train 0
-----
Record: 0   Speak now   ← Primeira fala
Record: 0   Speak again ← Segunda fala
Record: 0   Success
Train success: 1
Record 0    Trained    ← Voz treinada
-----
 Auto-rolagem  Show timestamp
Ambos, NL e CR  115200 velocidade  Deleta a saída

```

Figura 15: Comando 0 treinado (print da IDE do arduino).

Ao confirmar que a voz escolhida foi treinada deve-se carregá-la no módulo de reconhecimento de voz, então escrevemos o comando “*load 0*” após pressionar a tecla ENTER ou o botão enviar, o comando 0 estará carregado, o LED ficará piscando na cor amarela e já pode ser testado. Esse mesmo procedimento será feito para treinar e carregar os comandos restantes, mudando apenas o número correspondente aos comandos de 0 a 80.

O monitor serial na figura 16 já mostra as instruções de quais comandos devem ser utilizados.

```

COM3
train 0
Enviar
u\yyyyyElechouse Voice Recognition V3 Module "train" sample.
-----
Usage:
-----
COMMAND      FORMAT          EXAMPLE          Comment
-----
train        train (r0) (r1)...  train 0 2 45    Train records
load         load (r0) (r1) ...  load 0 51 2 3   Load records
clear        clear            clear           remove all records in Recognizer
record       record / record (r0) (r1)... record / record 0 79 Check record train status
vr           vr              vr             Check recognizer status
getsig       getsig (r)      getsig 0        Get signature of record (r)
sigtrain     sigtrain (r) (sig) sigtrain 0 ZERO  Train one record(r) with signature(sig)
settings     settings        settings        Check current system settings
help         help            help           print this message
-----
 Auto-rolagem  Show timestamp
Ambos, NL e CR  115200 velocidade  Deleta a saída

```

Figura 16: Comandos utilizados no exemplo “*vr_sample_train*” (print da IDE do arduino).

Depois que todos os comandos estiverem carregados, utiliza-se a biblioteca “vr_sample_multi_cmd” como base para controle do restante dos módulos. O próximo passo é definir uma programação para acionar os 4 relés pelas vozes treinadas no módulo de reconhecimento de voz.

3.3 FUNCIONAMENTO DOS RELÊS

O módulo relê, Figura 17, será usado no protótipo da domótica devido à sua simplicidade e utilidade para integrar a projetos com arduino. O módulo possui 4 canais, o qual estará conectado ao circuito de iluminação da residência para acionar as lâmpadas de 4 cômodos.

Os pinos digitais 4, 5, 6 e 7 do arduino serão utilizados para o acionamento dos relés. O módulo relê possui seus pinos de entrada em estado alto. Com isso, os pinos de saídas do arduino são declarados alto e para acionar o relé será criada uma programação que faça mudar o estado dos pinos de saída de alto para baixo, após o reconhecimento da voz do usuário.

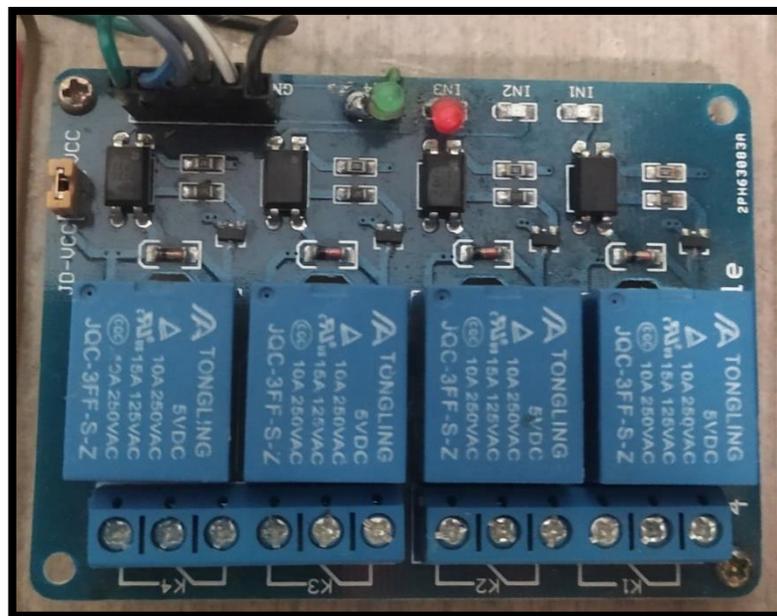


Figura 17: Módulo relê de 4 canais usado no protótipo mostrando a parte de cima com conexões dos pinos do arduino enquanto a parte de baixo será ligado a fase e os retornos das lâmpadas (autoria própria).

O esquema de conexão dos componentes para testes é apresentado na Figura 18, para uma alimentação de 5V nas três placas.

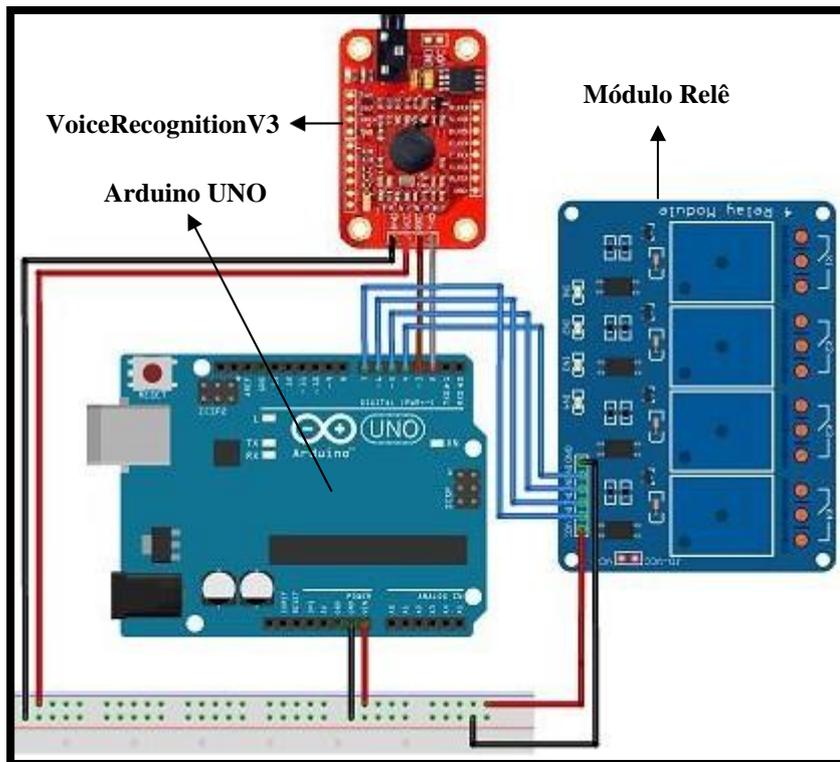


Figura 18: Esquema para testes com o módulo relê (print do *fritzing*).

3.4 USANDO O DFPLAYER MINI

O módulo foi escolhido devido ao baixo custo, simplicidade e facilidade em utilizar com Arduino. Ele é ideal para gerenciamentos de áudios de resposta a comandos de voz do usuário, facilidade de controle de transmissão e recepção de dados. Observa-se na Figura 19, o nome de cada pino abaixo.

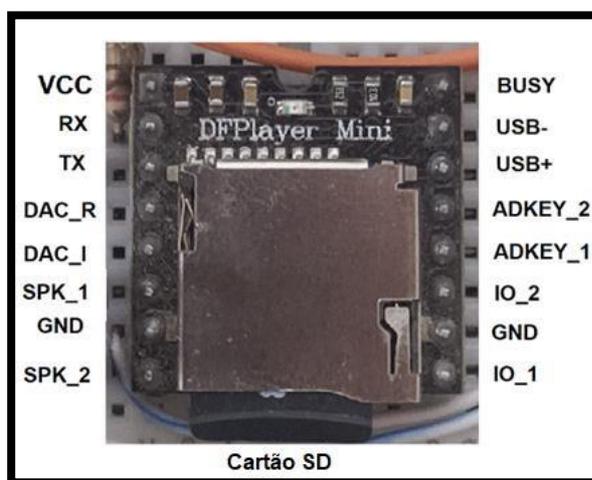


Figura 19: *DFPLAYER MINI* com identificação de cada pino (autoria própria).

Este componente funciona em conjunto a um alto-falante qualquer, onde os arquivos de

áudio são reproduzidos no mesmo, através dos pinos de saída SPK_1 e SPK_2 que, por sua vez, é ligado a uma saída P2 fêmea. O alto-falante deve conter alimentação própria e cabo P2 macho e o positivo e negativo respectivamente ser conectado aos pinos de saída SPK_1 e SPK2 do *DFPlayerMini*. Os pinos do *DFPlayerMini* ligados aos do arduino são respectivamente RX ao pino digital 9, deve existir um resistor de $230\ \Omega$ no RX para evitar ruídos na reprodução sonora, TX ao pino 8, VCC ao Vin e GND ao GND. Na figura 20 é apresentada a instalação para testes da reprodução de áudios.

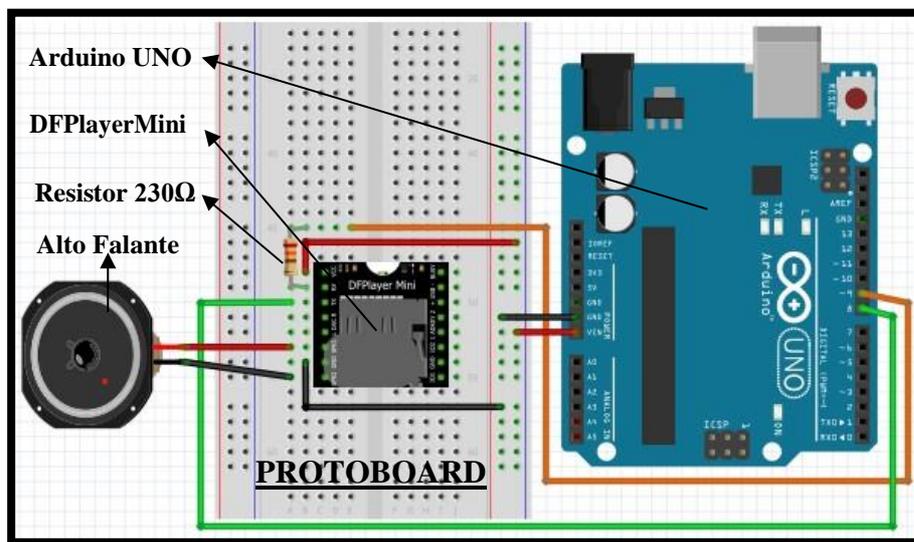


Figura 20: Demonstração das conexões dos pinos para o projeto da domótica (print do *fritzing*).

3.4.1 Programa para funcionamento do DFPlayer Mini

A programação referente a biblioteca *DFRobotDFPlayerMini* criada por *DFRobot*, encontrada no gerenciador de bibliotecas. Ao baixa-la pode-se encontrar nos exemplos diversas funções de programação do módulo na figura 21, usar as funções na figura 22, necessárias para o protótipo após inclui-la a “*vr_sample_multi_cmd*” que é a biblioteca base usada para a domótica (DFROBOT).

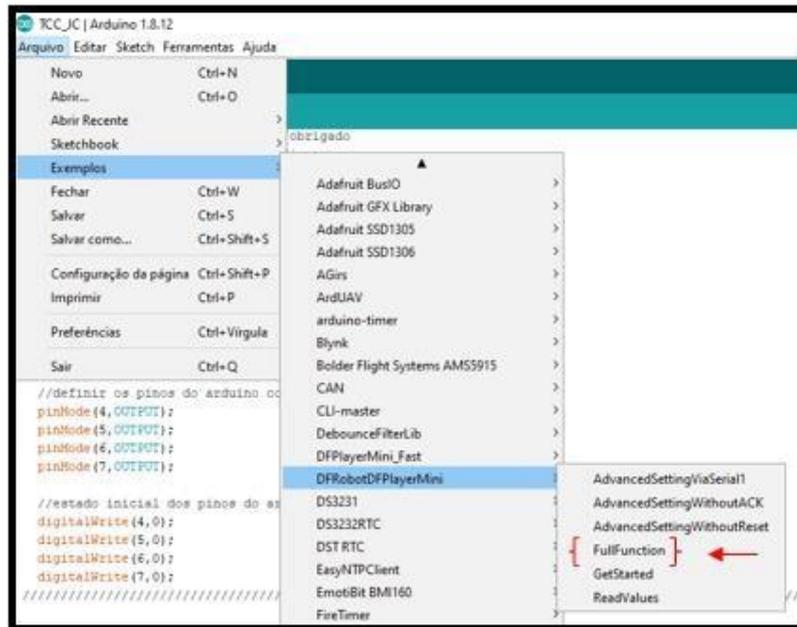


Figura 21: Utilizando a biblioteca “FullFunction” para controle do módulo *DFPlayerMini* (Print da IDE arduino).

```

FullFunction

myDFPlayer.setTimeout(500); //Set serial communication time out 500ms

//----Set volume----
myDFPlayer.volume(10); //Set volume value (0~30).
myDFPlayer.volumeUp(); //Volume Up
myDFPlayer.volumeDown(); //Volume Down

//----Set different EQ----
myDFPlayer.EQ(DFPLAYER_EQ_NORMAL);
// myDFPlayer.EQ(DFPLAYER_EQ_POP);
// myDFPlayer.EQ(DFPLAYER_EQ_ROCK);
// myDFPlayer.EQ(DFPLAYER_EQ_JAZZ);
// myDFPlayer.EQ(DFPLAYER_EQ_CLASSIC);
// myDFPlayer.EQ(DFPLAYER_EQ_BASS);

//----Set device we use SD as default----
// myDFPlayer.outputDevice(DFPLAYER_DEVICE_U_DISK);
myDFPlayer.outputDevice(DFPLAYER_DEVICE_SD);
// myDFPlayer.outputDevice(DFPLAYER_DEVICE_AUX);
// myDFPlayer.outputDevice(DFPLAYER_DEVICE_SLEEP);
// myDFPlayer.outputDevice(DFPLAYER_DEVICE_FLASH);

```

Figura 22: Exemplo de vários comandos para controle do *DF Player Mini (DFRobot)*.

Para que um áudio seja chamado no programa, deve ser digitado na IDE o comando: “myDFPlayer.playFolder(2,28);”, onde o primeiro número é a numeração da pasta “02” e o segundo número é o arquivo de áudio “028” encontrado na pasta 2.

As pastas onde ficam os áudios armazenados no cartão SD, devem ser renomeados da conforme a figura 23.

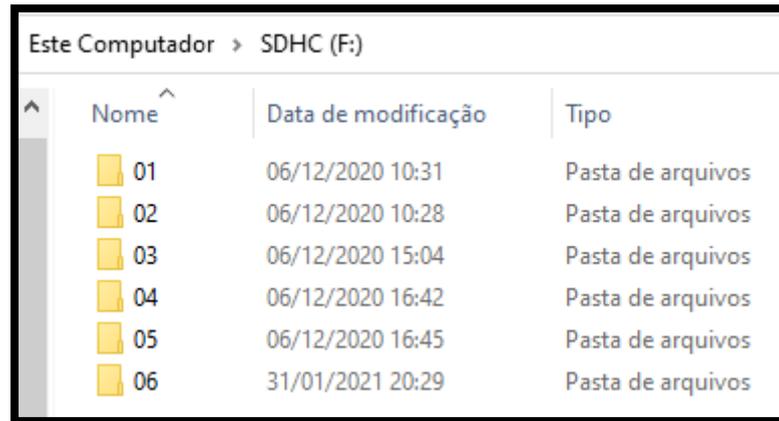


Figura 23: Pastas enumeradas no cartão micro SD (autoria própria).

Os áudios no formato *Mp3* são renomeados com os 3 primeiros dígitos numéricos conforme apresentado na Figura 24. Isso é necessário para que o módulo identifique qual áudio será utilizado, de acordo com a programação. O mesmo procedimento é válido para as outras pastas que serão utilizadas.

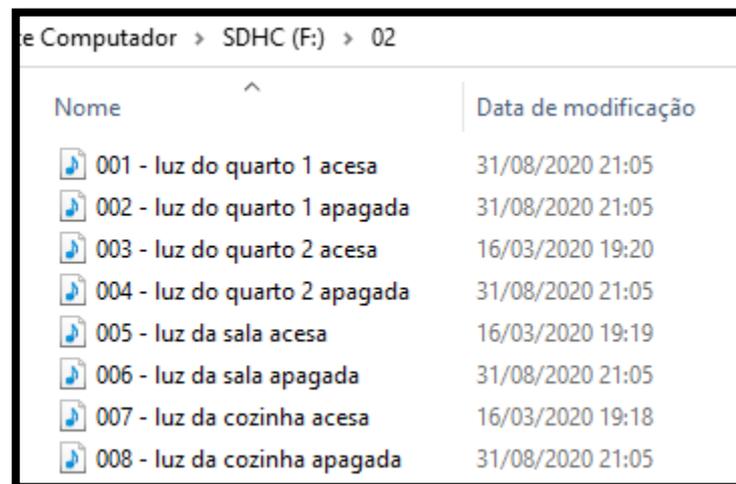


Figura 24: Nomes dos áudios nas pastas do cartão de memória micro SD (autoria própria).

Os áudios foram criados a partir de um aplicativo gratuito disponibilizado na *playstore*, chamado de “voz do narrador” observado na figura 25. Nele, o usuário digita o texto que será falado em áudio, considerando as seguintes opções: voz masculina ou feminina, velocidade de reprodução, voz grave ou agudas, etc. O áudio é obtido no formato de mp3 e enviado para as pastas do cartão micro SD, utilizando um computador. Os arquivos de áudios também podem ser criados a partir das gravações da própria voz ou obtidos de outros aplicativos.

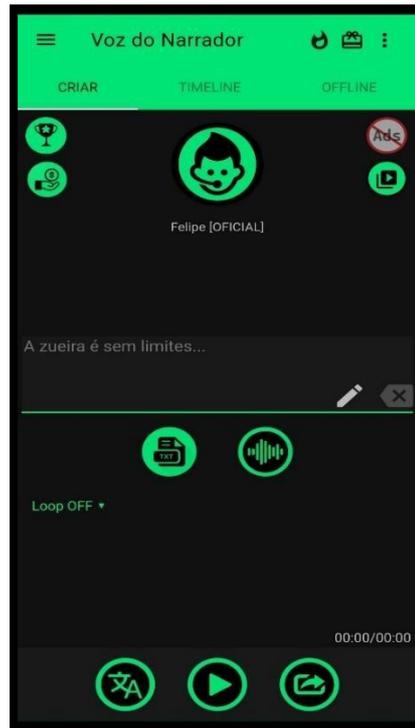


Figura 25: Aplicativo de voz do narrador no smartphone (HUGO, V.; COLADELO, R.; FERES, H., 2021).

3.5 USANDO O RTC (REAL TIME CLOCK)

O relógio de tempo real (RTC), mais precisamente RTC DS3231 visto na figura 26, foi escolhido com o intuito de orientar o usuário sobre o tempo real no momento que desejar, com informações de hora, data, dia da semana, e a temperatura.

Com essas funcionalidades ele poderia ser utilizado como temporizador para acionar aparelhos no tempo que o usuário preferir, por exemplo: Acender lâmpadas automaticamente as 15:22 horas e apagar as 15:23 horas conforme a figura 29, podendo evitar problemas como esquecer de apagar as luzes quando sair de casa ou até pegar no sono com as luzes acesas, gerando assim mais economia de energia elétrica; Acionar ar-condicionado quando a temperatura ficar mais alta, tornando-se um climatizador automático; Irrigar plantas do jardim no horário correto; Todas essas ideias fazem com que esse componente seja importante para integração com o protótipo de domótica deixando mais funcional.

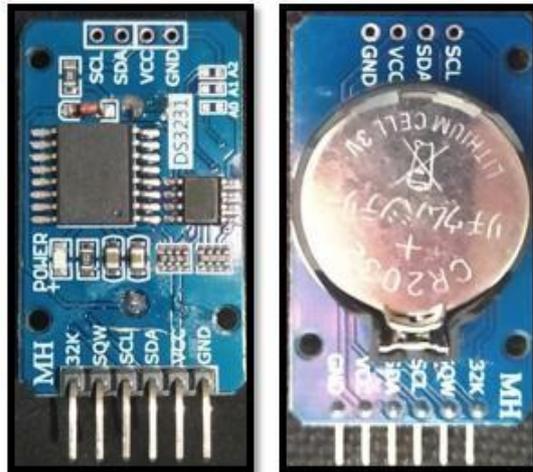


Figura 26: RTC DS 3231 mostrando seus pinos e a bateria de lithium (autoria própria).

Seguindo os passos para utilizar o módulo RTC no protótipo, inicialmente será baixada a biblioteca “RTClib” (ADAFRUIT) com o gerenciador de bibliotecas, e logo depois encontrase nos exemplos conforme a figura 27, serão introduzidos os comandos ao exemplo base: “vr_sample_multi_cmd” da biblioteca VoiceRecognitionV3 (ELECHOUSE). Serão usados alguns comandos necessários para obter o funcionamento planejado. As conexões usadas serão SCL conectada ao pino A5 (analógico 5) do arduino e SDA conectada em A4 (analógico 4), e a alimentação VCC em 5V e GND em GND, conforme a figura 28.

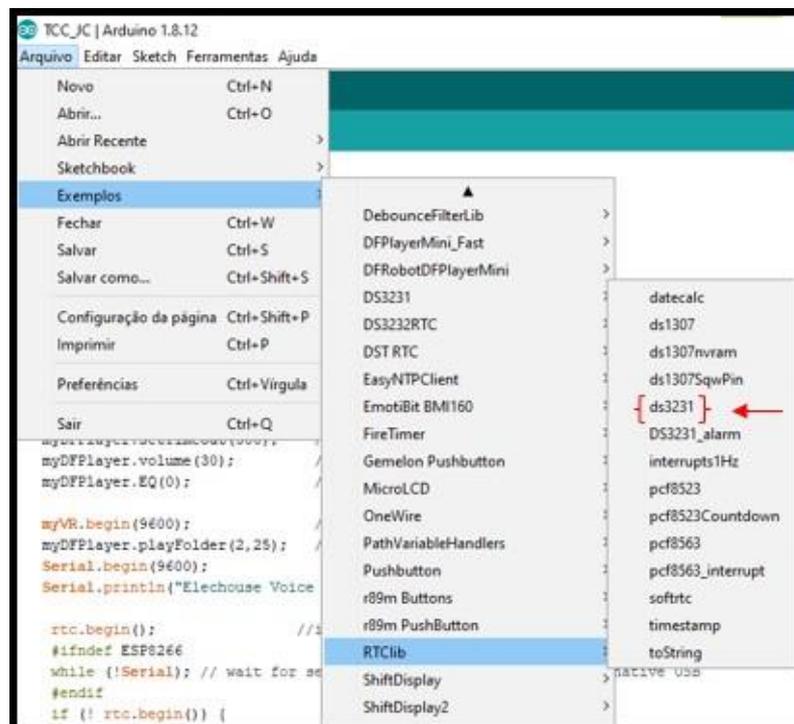


Figura 27: Encontrando a biblioteca exemplo do RTC 3231 que serão incluídos comandos para controle do módulo (Autoria própria).

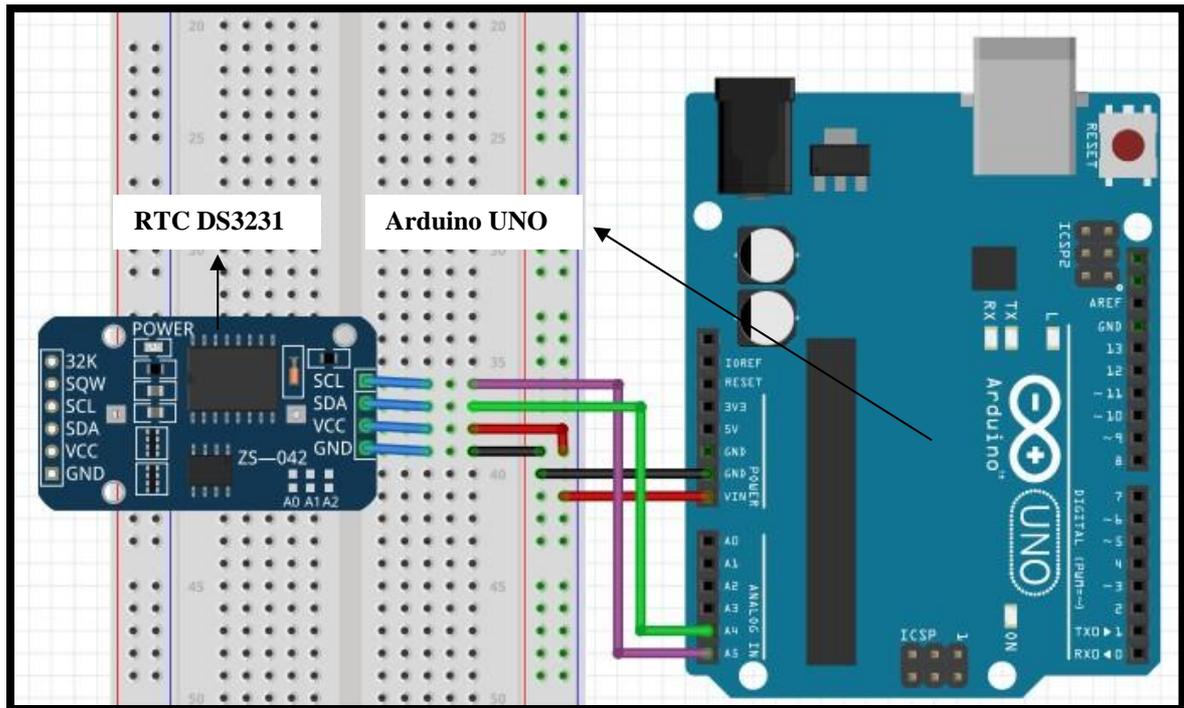


Figura 28: Esquema de ligação dos pinos do RTC 3231 no arduino (print do *fritzing*).

```

void loop()
{
    DateTime now = rtc.now();

    ///////////////acionamento programado //////////////////////

    const int HORA = 15;        //variavel de hora que o rele ira acionar
    const int MINUTO = 22;     //variavel de minuto
    const int SEGUNDO = 20;    //variavel de segundo

    const int HORA2 = 15;     //variavel 2 de hora que o rele ira acionar
    const int MINUTO2 = 23;   //variavel 2 de minuto
    const int SEGUNDO2 = 20;  //variavel 2 de segundo

    if((now.hour()==HORA)&&(now.minute()==MINUTO)&&(now.second()==SEGUNDO)){ //se a hora,minuto e segundo for igual a programada
        digitalWrite(rele[0], 1); //liga o rele 0
        digitalWrite(rele[1], 1); //liga o rele 1
        digitalWrite(rele[2], 1); //liga o rele 2
        digitalWrite(rele[3], 1); //liga o rele 3
    }
    if((now.hour()==HORA2)&&(now.minute()==MINUTO2)&&(now.second()==SEGUNDO2)){ //se a hora 2,minuto 2 e segundo 2 for igual a programada
        digitalWrite(rele[0], 0); //desliga o rele 0
        digitalWrite(rele[1], 0); //desliga o rele 1
        digitalWrite(rele[2], 0); //desliga o rele 2
        digitalWrite(rele[3], 0); //desliga o rele 3
    }
}

```

Figura 29: Exemplo de como acionar relês por tempo programado (autoria própria).

3.5.1 Método para reprodução de áudios

Para reproduzir o som de cada elemento foram criadas pastas com os áudios que são reproduzidos ao executar o comando do DFPlayerMini “myDFPlayer.playFolder(x,y);”, em que x é o número da pasta e y o número do arquivo de áudio na pasta. Os áudios da pasta são enumerados de 000 a 059 conforme a figura 30 e cada áudio representa o som do número. Exemplo o número 000 possui o áudio falado zero e assim cada número é o som do próprio

número. Dessa forma a programação para reproduzir a hora diante de um comando de voz pode ser observada na figura 31.

Nome	Data de modificação
 000-0	06/12/2020 11:22
 001 -1	28/11/2020 15:01
 002 -2	28/11/2020 15:01
 003 -3	28/11/2020 15:01
 004 -4	28/11/2020 15:01
 005 -5	28/11/2020 15:01

Figura 30: Pasta 01 áudios dos números de 0 a 59 (autoria própria).

```
// GRUPO 1
case group1Record1: //-----

    Serial.print("Comando 1 reconhecido - ");
    Serial.print("grupo:");
    Serial.println(group);

    Serial.print(now.hour(), DEC);
    Serial.print(':');
    hora=now.hour();
    myDFPlayer.playFolder(1, hora); // numero
    delay(1000);
    myDFPlayer.playFolder(3,4); //horas
    delay(1000);
    myDFPlayer.playFolder(3,2); // E
    delay(800);
    Serial.print(now.minute(), DEC);
    Serial.print(':');
    minu=now.minute();
    myDFPlayer.playFolder(1, minu); //numero
    delay(1500);
    myDFPlayer.playFolder(3,3); //minutos falando
    delay(1000);
    break;
```

Figura 31: Programação para reproduzir as horas usando áudios das pastas 01 e 03 (autoria própria).

Alguns áudios extras foram criados e armazenados na pasta 03 visto na figura 32, para reproduzir palavras diferentes dos números das horas que estão na pasta 01.

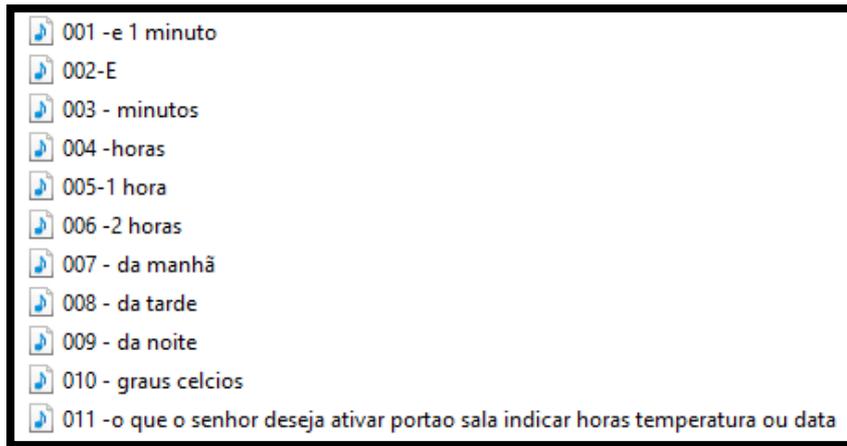


Figura 32: Pasta 03 áudios extras (autoria própria).

Para a reprodução da temperatura, o comando “`rtc.getTemperature();`” apresenta o número em tempo real da temperatura. Esse número deve ser correspondente ao que estiver na pasta 01 pela sequência de comando:

1. `temperatura=rtc.getTemperature();`
2. `myDFPlayer.playFolder(1, temperatura);`

Para reproduzir a data, foi criado um vetor que armazena os dias da semana:

`Char daysOfTheWeek[7][12]= {"Domingo","Segunda","Terça","Quarta","Quinta","Sexta", "Sabado"}`. Nas Figuras 33, 34, 35 e 36 observam-se cada comando de data e pastas correspondente à mesma:

```
Serial.print(now.day(), DEC);           //numero do dia no serial
Serial.print('/');

myDFPlayer.playFolder(1,now.day());     //numero do dia sonoro
delay(1000);

Serial.print(now.month(), DEC);         //numero do mes no serial
Serial.print('/');

myDFPlayer.playFolder(4,now.month());   //reproduz o som do mes atual
delay(1000);

Serial.print(now.year(), DEC);          //numero do ano atual
Serial.print('/');
// ano=now.year();
myDFPlayer.playFolder(6,6);             //de 2021
delay(1500);

//dia da semana
Serial.print(daysOfTheWeek[now.dayOfTheWeek()]);
Serial.print(" ");
semana=now.dayOfTheWeek();
```

Figura 33: Programação do comando de voz: “data” reproduzindo a data atual (autoria própria).

000 -Domingo	24/01/2021 21:09
001 - segunda feira	28/11/2020 15:08
002 -terça feira	28/11/2020 15:08
003 -quarta feira	28/11/2020 15:08
004 -quinta feira	28/11/2020 15:09
005 -sexta feira	28/11/2020 15:08
006 -sabado	28/11/2020 15:09
007 -Domingo	24/01/2021 21:09

Figura 34: Arquivos da pasta 05, dias da semana (autoria própria).

001 -de janeiro	28/11/2020 15:05
002 -de fevereiro	28/11/2020 15:05
003- de março	28/11/2020 15:05
004 - de abril	28/11/2020 15:05
005 -de maio	28/11/2020 15:05
006 -de junho	28/11/2020 15:05
007 -de julho	28/11/2020 15:05
008 -de agosto	28/11/2020 15:06
009 -de setembro	28/11/2020 15:06
010 -de outubro	28/11/2020 15:06
011 -de novembro	28/11/2020 15:06
012 -de dezembro	28/11/2020 15:06

Figura 35: Arquivos da pasta 04, meses do ano (autoria própria).

005 -de 2020	28/11/2020 15:06
006 -de 2021	28/11/2020 15:06
007 -de 2022	28/11/2020 15:07
008 -de 2023	28/11/2020 15:07
009-de 2024	28/11/2020 15:07
010 -de 2025	28/11/2020 15:07
011 -de 2026	28/11/2020 15:07
012 - de 2027	28/11/2020 15:07
013 -de 2028	28/11/2020 15:08
014 -de 2029	28/11/2020 15:08
015 -de 2030	28/11/2020 15:08

Figura 36: Arquivos da pasta 06, áudios dos anos de 2020 a 2030 (autoria própria).

Como o DFPlayerMini reconhece um arquivo com no máximo os 3 primeiros números, e os anos são compostos de 4 números, teve-se que escolher o número correspondente ao ano e colocar no comando: “myDFPlayer.playFolder(6,6);”, reproduzindo assim o áudio “006” da pasta “06” do ano 2021, conforme a Figura 37.

```

Serial.print(now.year(), DEC);           //numero do ano atual
Serial.print('/');
ano=now.year();
if(ano==2021){                           //se a data do ano for 2021
myDFPlayer.playFolder(6,6);              //reproduz o som: de 2021
delay(1500);
}
if(ano==2022){                           //se a data do ano for 2022
myDFPlayer.playFolder(6,7);              //reproduz o som: de 2022
delay(1500);
}
  if(ano==2023){                           //se a data do ano for 2023
myDFPlayer.playFolder(6,8);              //reproduz o som: de 2023
delay(1500);
}
if(ano==2024){                            //se a data do ano for 2024
myDFPlayer.playFolder(6,9);              //reproduz o som: de 2024
delay(1500);
}
}

```

Figura 37: Programação para reproduzir o som dos anos colocada no comando “data” (Print da IDE do arduino).

O RTC DS3231 deve ser calibrado para mostrar a informação correta atual. Na Figura 38, observa-se o código para ajuste, no qual o procedimento de programação deve ser carregado no Arduino. Caso houver necessidade de modificar outro comando, todo o comando de ajuste deve ser comentado com duas barras em cada linha de código, pois não é mais necessário a modificação. Assim, mesmo que o protótipo fique sem alimentação, as informações permanecerão memorizadas, já que o RTC 3231 possui bateria.

```

//      AJUSTE DO RTC ANO/MES/DIA/HORAS/MINUTOS/SEGUNDOS
if (rtc.lostPower()) {

    // rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));

    rtc.adjust(DateTime(2021, 06, 10, 20, 42,00));//----->////////local de ajuste/
}

```

Figura 38: Ajuste do Módulo RTC 3231 (print da IDE).

3.6 USANDO O NODEMCU 1.0 ESP8266 ESP12E

Planejado para controlar a iluminação por meio de internet sem-fio (Wi-fi) através de um celular smartphone, o módulo NodeMcu V3 na figura 39 e com descrição dos seus pinos na figura 40, foi fundamental para implementar o protótipo da domótica obtendo outra forma de

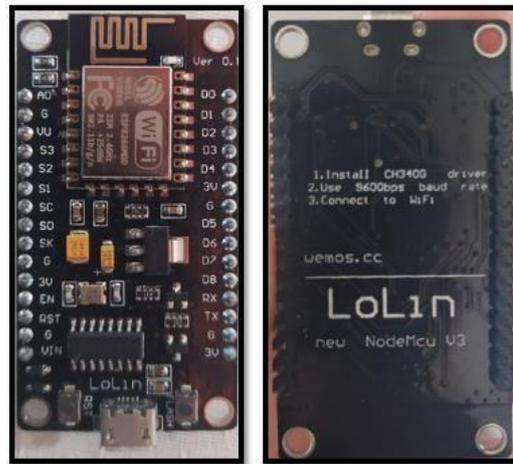


Figura 39: Módulo Wi-fi Esp8266 NodeMcu usado no protótipo (Autoria própria).

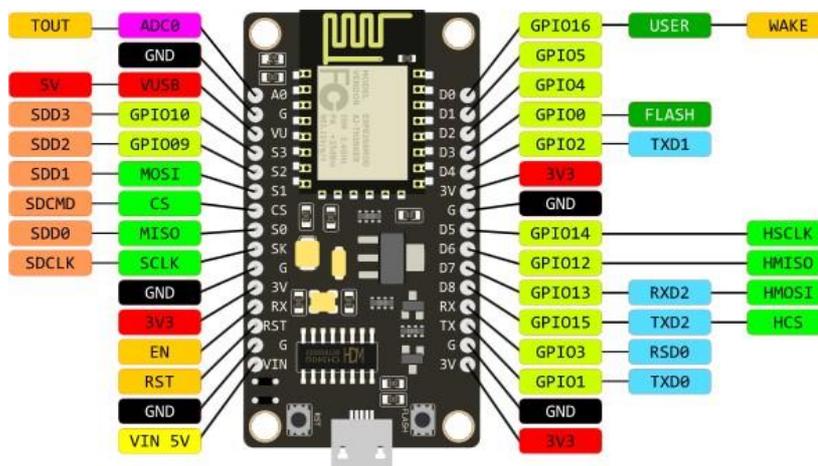


Figura 40: Pinagem do ESP8266 (FREITAS, 2019).

Para inserir a programação temos que baixar a biblioteca blynk (BLYNK IOT PLATFORM), encontrada no gerenciador de bibliotecas e abrir a opção “ESP8266_standalone” de acordo com a figura 41 e para gravar seleciona-se na placa destinada conforme a figura 42.

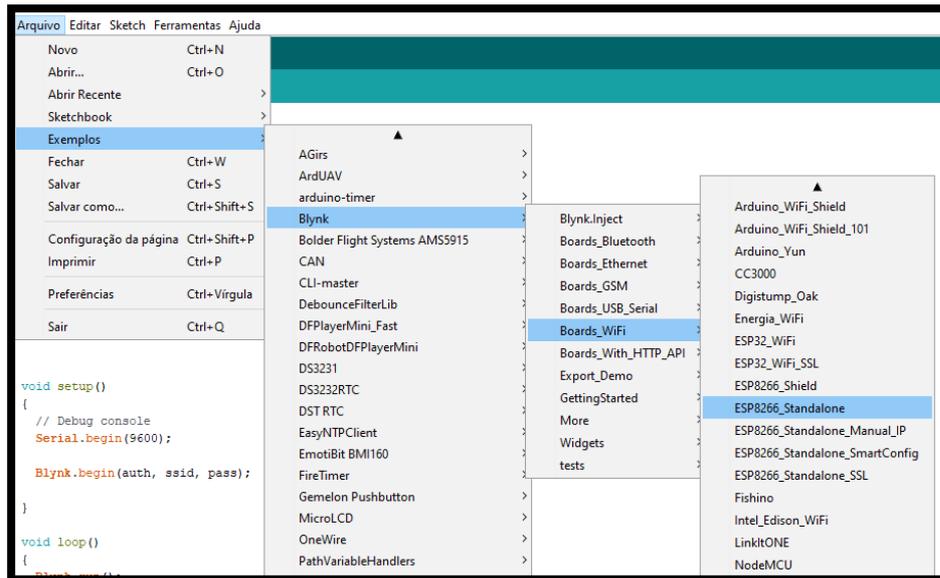


Figura 41: Encontrando a biblioteca para usar no NodeMcu (print da IDE).

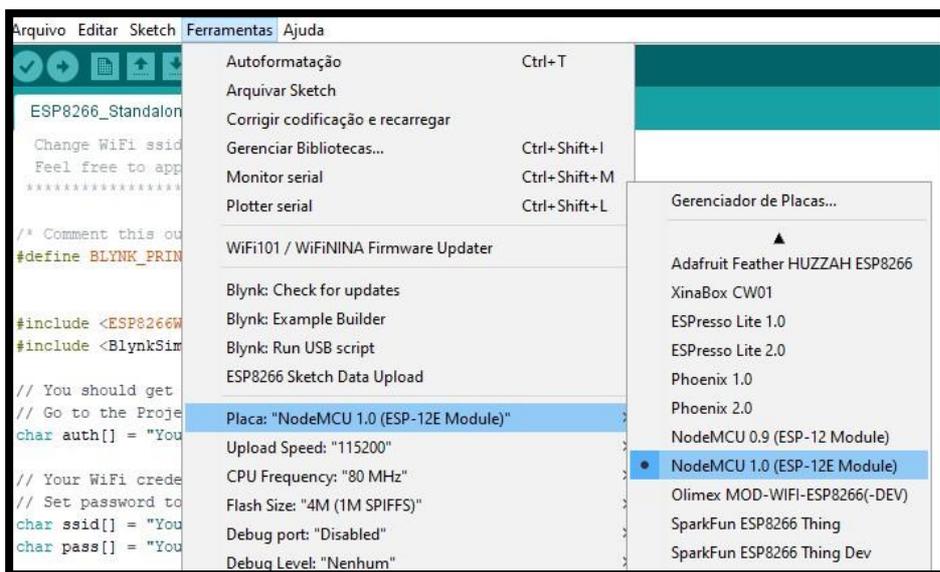


Figura 42: Habilitando à placa para gravar a programação (print da IDE).

Antes de enviar a programação para o módulo NodeMcu, temos que baixar o aplicativo *Blynk* da *playstory* para ter acesso a um registro em formato digital “*Auth Token*” que é uma sequência de caracteres única, como uma chave de acesso ao módulo NodeMcu. Após criar uma conta no aplicativo e enviar o “*Auth Token*” por e-mail iremos usá-lo na programação. Observam-se todos os passos nas figuras 43, 44, 45, 46 e 47.

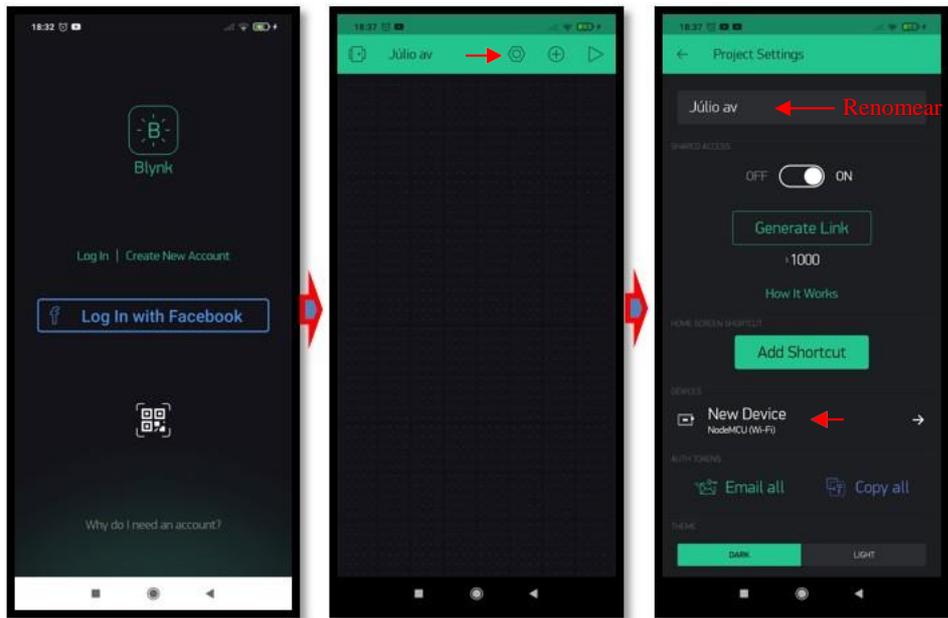


Figura 43: Passos para enviar o “Auth Token” (autoria própria).

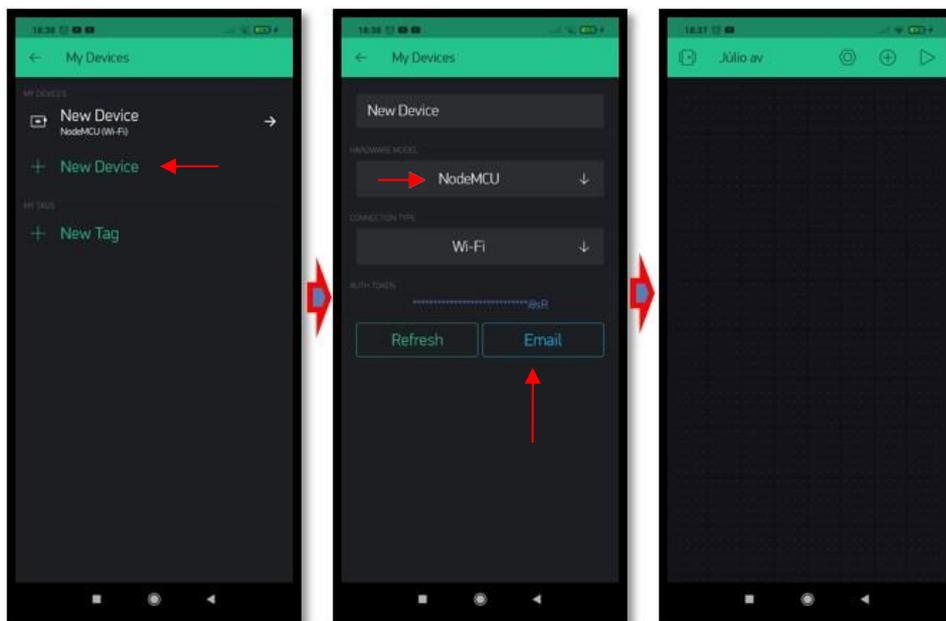


Figura 44: Enviando o “Auth Token” (Autoria Própria).

```

#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>

char auth[] = "*****"; //inserir o "Auth Token" recebido por e-mail

// Your WiFi credentials.
// Set password to "" for open networks.
char ssid[] = "*****"; // Nome do Wifi usado
char pass[] = "*****"; // Senha do Wifi usado

void setup()
{
  // Debug console
  Serial.begin(9600);

  Blynk.begin(auth, ssid, pass);
}

void loop()
{
  Blynk.run();
}

```

Figura 45: Programação a ser carregada no NodeMcu após inserir o “AuthToken”, nome e senha do wifi usado na residência (print da IDE).

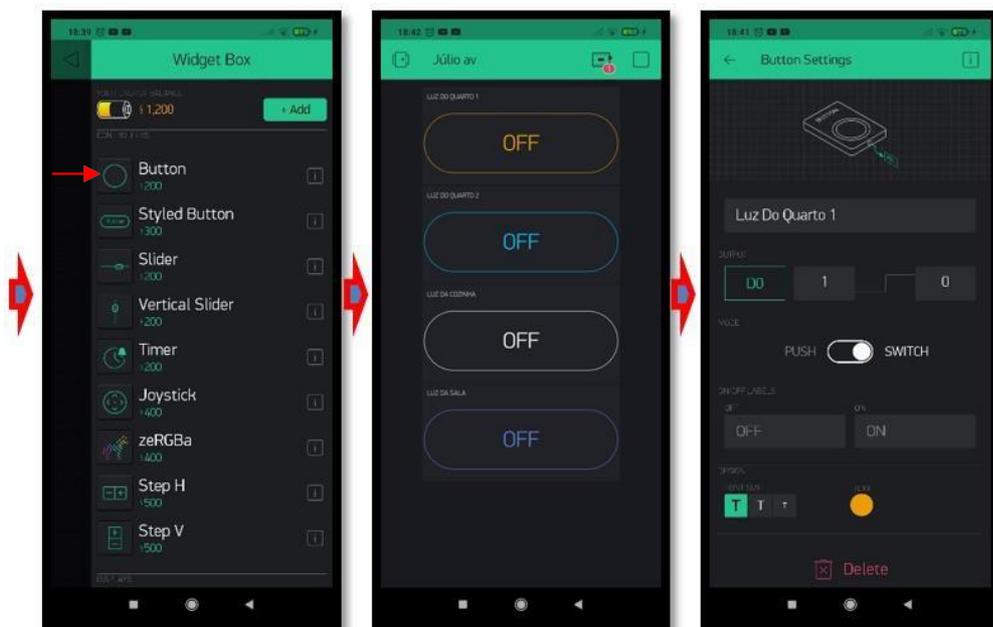


Figura 46: Puxando pela direita encontramos os itens e selecionamos o item “button”, moveremos 4 e renomearemos e configuraremos as saídas “D0” a “D3” iniciando como nível alto (autoria própria).

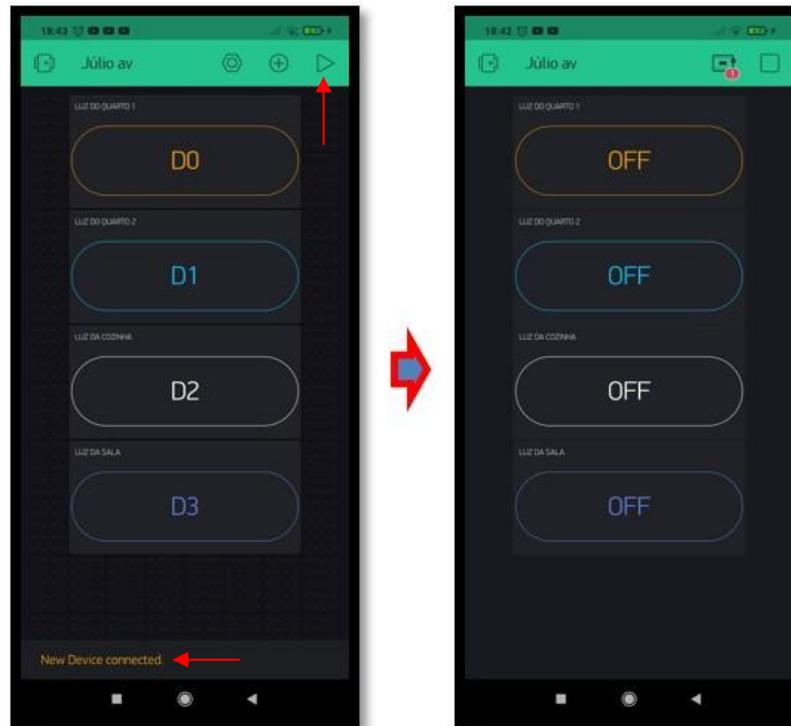


Figura 47: Ao alimentar o NodeMcu, podemos ver no aplicativo Blynk a confirmação de conexão e ao pressionar o play, podemos usá-lo (autoria própria).

A placa é alimentada nos pinos: 5V em VIN e GND em G, serão usados 4 pinos: D0, D1, D2, D3 definidos como saída para acionar os relês junto aos pinos de saída do arduino onde o modulo de reconhecimento de voz o controla. Para isso um CI que funcione como porta lógica “XOR” (OU exclusivo) é necessária para fazer o acionamento individual de cada microcontrolador em um único relê. Na figura 48 é visto o esquema para testes usando o CI HD74LS86P como porta XOR, arduino UNO, ESP8266 NodeMCU, relê e uma fonte de 5V corrente contínua.

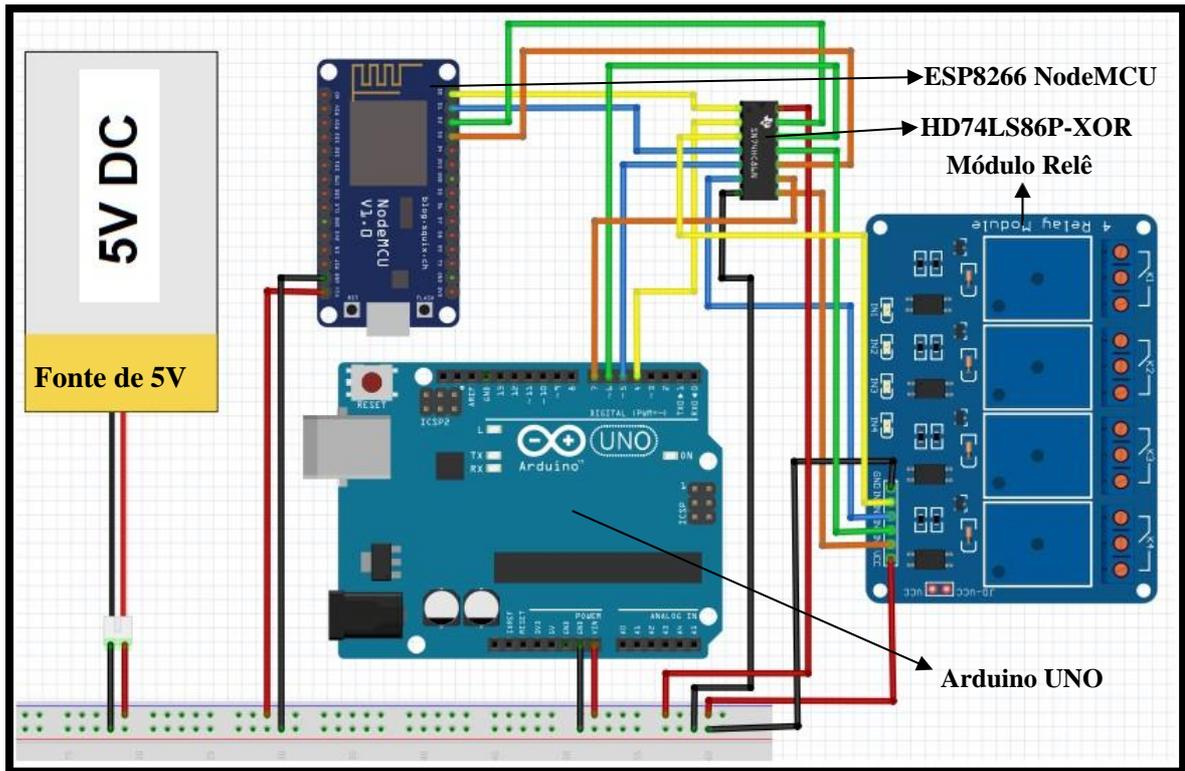


Figura 48: NodeMcu e Arduino Uno para acionamento do módulo relê com CI de XOR (print do *frtzing*).

3.7 HD74LS86P-XOR (OU-EXCLUSIVO)

O circuito integrado que funciona como XOR (ou exclusivo) é uma porta lógica que funciona com duas entradas e uma saída que são combinadas de modo que $x = \bar{A}B + A\bar{B}$ ou $x = A \oplus B$, onde x é a saída e A e B as entradas formando o diagrama da figura 49 abaixo:

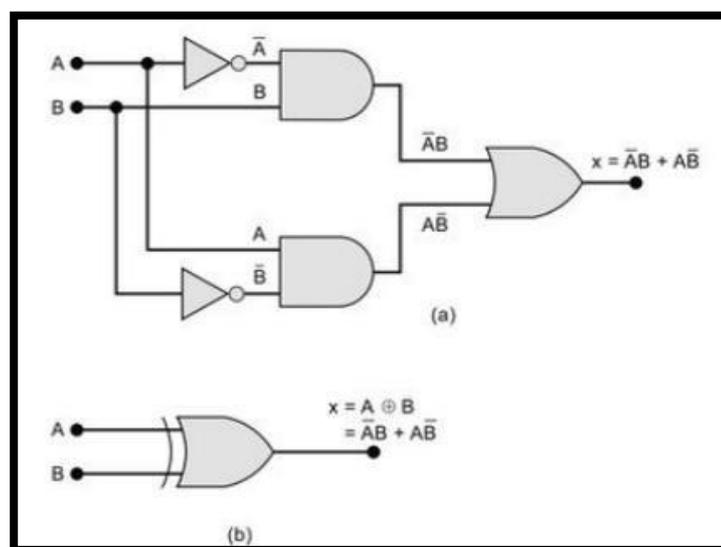


Figura 49: (a) Circuito XOR utilizando as portas NOT, AND e OR (b) Circuito tradicional para porta XOR (TOCCI, Ronald J., 2011).

Na porta lógica XOR, a saída será nível alto quando as duas entradas forem níveis diferentes e será nível baixo quando as duas entradas forem níveis iguais, conforme apresentado na Tabela 1.

ENTRADAS		SAÍDAS
A	B	X
0	0	0
1	0	1
0	1	1
1	1	0

Tabela 1 - Tabela-verdade da porta lógica XOR (TOCCI, Ronald J., 2011).

O CI “HD74LS86P” funciona com 8 entradas diferentes e 4 saídas foi muito importante para integrar o ESP8266 NodeMcu ao protótipo.

Enquanto os pinos de saída: D3, D2, D1 e D0 do NodeMCU ou ESP8266 serão conectados respectivamente em: 5, 2, 12 e 10 do HD74LS86P, os pinos digitais de saída do arduino UNO: D4, D5, D6 e D7 serão conectados respectivamente em: 9, 13, 1 e 4 e do HD74LS86P representada esquematicamente exterior e interior na figura 50 e 51 abaixo.

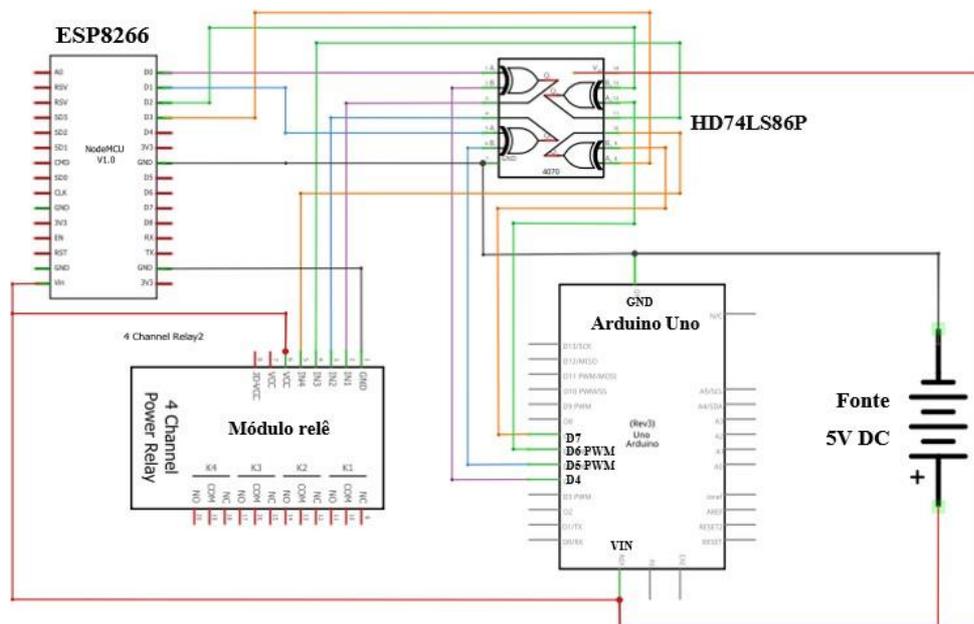


Figura 50 Esquematico de montagem com Arduino, esp8266 NODEMCU, HD74LS86P e Módulo Relê (Fritzing).

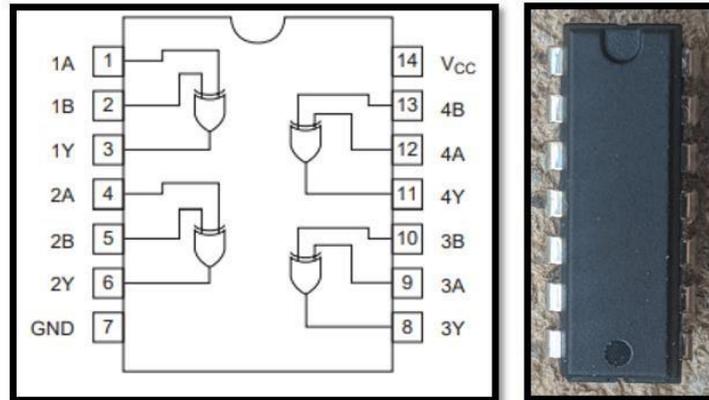


Figura 51: Representação esquemática do CI HD74LS86P (RENESAS) com o real utilizado no protótipo (autoria própria).

3.8 USANDO O DISPLAY OLED SSD1306

Foi necessário integra-lo ao protótipo afim de auxiliar o usuário com informações visuais de indicação do estado das lâmpadas (acesa ou apagada) e de horas, temperatura e data, mostradas em tempo real, principalmente a pessoas portadoras de deficiência auditiva, ou seja, incapaz de ouvir sons, podendo observar as informações no display.

Incluímos a biblioteca MultiLCD (HUANG), para controlarmos o Display OLED e criamos uma sequência de tarefas desejada.

3.8.1 Principais Especificações

- a) O *display* possui os pinos: SDA, SCK, VCC e GND ligados respectivamente em: A4, A5, 5 V e GND do Arduino UNO, conforme apresentado na Figura 52.
- b) VCC de 7 V a 15 V para painel de condução (SOLOMON SYSTECH).
- c) Corrente máxima da fonte do segmento de 100 μ A (SOLOMON SYSTECH).
- d) Corrente de dissipação máxima comum: 15 mA (SOLOMON SYSTECH).
- e) Ampla faixa de temperatura operacional: -40 °C a 85 °C (SOLOMON SYSTECH).
- f) Utiliza controlador para emissão de luz orgânica. Consiste em 128 segmentos e 64 comuns e resolução de 0,96 polegadas (SOLOMON SYSTECH).
- g) Possui interface de barramento I2C (Circuito Inter-integrado), possibilitando o uso de diversos componentes padronizados, permitindo a troca de dados de maneira mais eficaz

(SOLOMON SYSTECH).

h) Seu controle de brilho é de 256 etapas (SOLOMON SYSTECH).

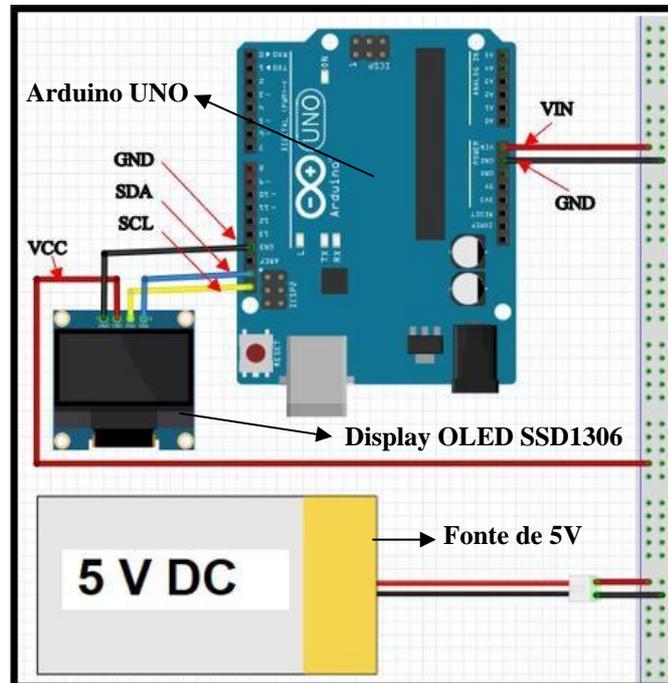


Figura 52: Esquema de conexão dos pinos do display OLED SSD1306 no arduino (montagem do *fritzing*).

3.9 COMPONENTES EXTRAS

Após a obtenção dos módulos que formam o protótipo, foram adicionados itens extras para complementar, dar apoio, melhorar a funcionalidade e a aparência do protótipo.

- a) **Cabo USB/Micro-USB** – barramento usado para: gravar dados desenvolvidos na IDE do Arduino utilizando um computador, no esp8266 NodeMCU com entrada Micro-USB.
- b) **Cabo USB A/B** – barramento usado para gravar toda rotina de programação no Arduino UNO.
- c) **Conector P2 fêmea** – usado para conexão de cabo P2 macho do (s) alto-falante (s) que será reproduzido os sons dos áudios. E conexão dos pinos “SPK_1” e “SPK_2” do módulo DFPlayerMini em positivo e negativo do conector fêmea.
- d) **Mini Chave Gangorra KCD11-101 2 Terminais** – mini interruptor manual usado para ligar ou desligar o módulo relé de 4 canais. Opera inicialmente aberto, evitando pulsos nas

lâmpadas no momento que o protótipo é energizado.

- e) **Mini Chave Alavanca SMTS102 2 Posições** – mini interruptor manual com a função de ligar ou desligar o módulo de reconhecimento de voz V3, afim de economizar energia elétrica da residência, caso o usuário deseje. Possui 3 terminais.
- f) **Chave Tecla KZJ-2x2 com Trava 6 Terminais** – interruptor manual com as mesmas funcionalidades dos anteriores, porém usado para ligar ou desligar o *display* OLED e para economia de energia.
- g) **Fonte de 5V** – carregador de celular simples modelo STA-U12ED da marca LG, com tensão de saída de 5 VDC e corrente de 0,7 A. Em sua saída USB, foi feita uma adaptação para ser utilizado dois fios: o positivo e o negativo que alimenta todos os módulos do protótipo.
- h) **Jumpers** – fios fabricados para projetos com microcontroladores, como arduino UNO. Os *jumpers* usados foram do tipo macho-fêmea e macho-macho, possuem conector de 2,54 mm, seção de 24 AWG e comprimento de 20 cm.
- i) **Protoboard** – placa de ensaio ou matriz de contato que possui 400 pontos compatíveis com os *jumpers* de conector macho e com os módulos possuidores de pinos macho. Foram usados os módulos: RTC3231, DFPlayerMini, HD74LS86P e os *jumpers*.
- j) **Resistor 230 Ω** – usado na entrada do pino RX do módulo DFPlayerMini para evitar ruídos na reprodução sonora.
- k) **Suporte de compensado** – criado com objetivo de fixar e cobrir os componentes evitando ficarem expostos à penetração de poeira ou detritos que se acumulam com o passar do tempo e para evitar impactos acidentais que podem gerar perda dos componentes, garantindo maior proteção e vida útil dos componentes eletrônicos.
- l) **Alto-falante** – trata-se de uma caixa de som usada para reproduzir os áudios criados após os comandos de voz. Tem alimentação no mesmo canal da alimentação dos módulos, com tensão de 5 VDC. Os sinais dos áudios serão transmitidos através dos pinos SPK_1 e SPK_2 do módulo DFPlayerMine ao cabo P2 fêmea, para ter opção que ser usado com qualquer alto-falante com cabo P2 macho.

m) Cartão SD –necessário o uso de um cartão de memória simples para armazenar os áudios em .mp3, podendo o usuário escolher o a capacidade que achar desejável.

4. ANÁLISE DOS RESULTADOS

Na montagem do protótipo de domótica os resultados foram bem-sucedidos, obtendo-se o funcionamento esperado para automatizar a residência. Contudo houveram várias tentativas de se obter uma rotina de tarefas esperadas, dentre elas temos os maus contatos nas conexões dos *jumpers* no protoboard gerando instabilidades nos relês e falhas no reconhecimento de voz. Uma das formas de resolver as falhas seria a montagem dos componentes em uma placa de circuito impresso ou o uso de cola-quente para que as conexões dos *jumpers* fiquem mais fixadas, com o uso da cola quente os problemas foram sanados como previsto.

Nos testes experimentais para funcionamento do NodeMCU simultaneamente com o arduino, primeiramente tinha-se o objetivo de fazê-lo funcionar em uma mesma programação, os pinos de saída do NodeMCU iriam para os pinos de entrada do arduino dessa forma foi trabalhado na IDE uma linha de comandos de uma porta lógica ou-exclusiva virtual, mas não se obteve êxito. Então para acelera o andamento do protótipo foi feita uma montagem externa utilizando o CI HD74LS86P-XOR possui um custo muito baixo o que gerou resultados satisfatórios.

Na realização dos testes do reconhecimento de voz, houve um problema relacionado com a reprodução do áudio do módulo DF Player Mine e a captação do áudio do usuário do módulo de reconhecimento de voz, pois, um dos comandos de voz criado era “quarto um” e a resposta sonora confirmava que a luz do “quarto um” estava acesa, no entanto, o som “quarto um” era repetido pelo próprio sistema e ele reconhecia a si próprio, isso gerou repetições de ativação e desativação. Para resolver esse problema existem três opções viáveis:

1. Diferenciar o comando de voz do da reprodução de resposta, ou seja, mudar o comando atual por um sinônimo ou mudar o som reproduzido de forma que o som fique diferente;
2. Baixar ou aumentar o volume de maneira que o volume fique diferente do comando treinado;
3. Mudar a voz de reprodução por uma mais grave ou aguda de maneira que diferencie da voz treinada no módulo de reconhecimento de voz pelo usuário.

4.1 FUNCIONAMENTO DO DISPLAY OLED SSD1306

No display foi programado a exibição das informações previstas em conjunto com o módulo RTC onde é originado as informações no momento em que o protótipo é ligado e iniciado. Ele exibe textos com duração de 4 segundos, como: “LUZ DO QUARTO 1 ACESA”, após comandos de voz através do módulo de reconhecimento de voz V3 que ativam os relês para acender ou apagar lâmpadas dos cômodos da residência, voltado a mostrar as informações iniciais logo depois, conforme a figura 53 a seguir.



Figura 53 Tela OLED SSD1306 do protótipo (autoria própria).

4.2 FUNCIONAMENTO DO ESP8266 EM CONJUNTO COM O BLINK

Na figura 54 observa-se o exemplo de funcionamento no acionamento do relê que acende ou apaga para ser usado na iluminação.

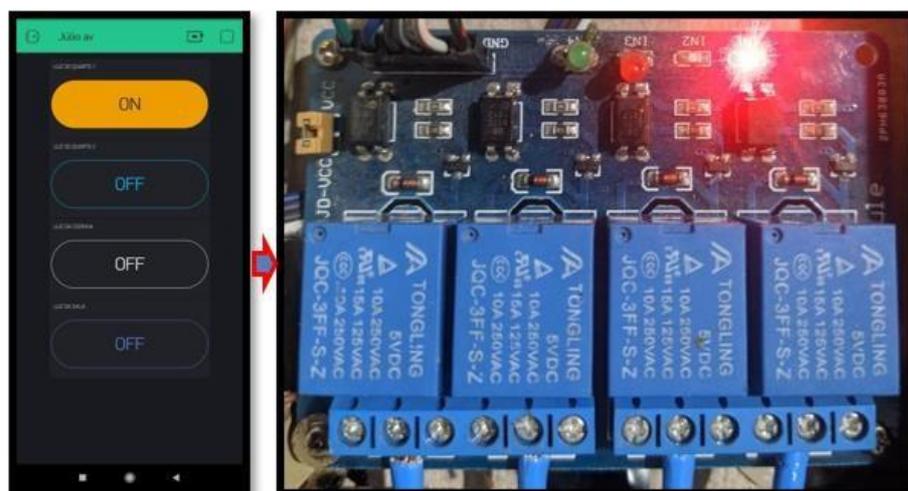


Figura 54: Relê após controlar com o aplicativo Blynk pelo smartphone (Autoria própria).

4.3 FUNCIONAMENTO COM O CI-HD74LS86P-XOR

Antes da obtenção do componente tentou-se criar uma lógica de ou-exclusivo na programação para poder integrar o NodeMcu no arduino UNO, foram criadas várias rotinas de programação e realizado testes sem resultados favoráveis.

Nos testes práticos o relê é ativado e desativado por voz, na rotina de programação do reconhecimento de voz, o estado do relê é utilizado como critério lógico, quando ele recebe sinal alto ele inverte o acionamento, ou seja, fica no estado baixo e o contrario acontece ao receber sinal nível baixo vindo das saídas do HD74LS86P. Abaixo figura 55 observa-se o exemplo do programa de um acionamento por voz relacionado ao estado do relê:

```

// GRUPO 0

case group0Record1:          //-----

myDFPlayer.playFolder(2,20); // ok
delay(1500);
Serial.print("Comando 1 reconhecido - ");
Serial.print("grupo:");
Serial.println(group);

if(digitalRead(rele[0])==1){ // se o relê estiver desativado
digitalWrite(rele[0], 0); // ativa o relê
myDFPlayer.playFolder(2,1); // luz do quarto 1 acesa
}
else{ // caso contrario
digitalWrite(rele[0], 1); // desativa o relê
myDFPlayer.playFolder(2,2); // luz do quarto 1 apagada
}

break;

```

Figura 55: Exemplo de programação de acionamento por voz (autoria própria).

Com a utilização do circuito integrado HD74LS86P foi possível usar os módulos simultaneamente, porém, em casos de reprodução sonora, só um módulo pode ser usado. Por exemplo: quando o relê é acionado somente utilizando o módulo de reconhecimento de voz, ele reproduzirá um som indicando que foi acionado, quando o relê for desativado por voz, um som indicará que foi desativado. Em caso de ativação por smartphone o arduino não indicará com reprodução sonora e, logo em seguida usar o módulo de reconhecimento de voz para ativar o relê novamente a reprodução indicará que foi acionado, quando na verdade estará desativando gerando um erro de reprodução. Dessa forma fica inviável usar a reprodução sonora para indicar que algum aparelho foi ligado ou desligado no caso de usar o reconhecimento de voz junto com o modulo ESP8266 via *wifi*, tornando inacessível para deficientes visuais. Esse problema

também gera erros quando a tela de mini display OLED é usada para mostrar informações do estado dos aparelhos tornando inviável também para deficientes auditivos.

4.4 ESQUEMAS DE INSTALAÇÃO E PROGRAMAÇÃO

4.4.1 Instalação na Residência

As instalações na residência foram feitas com fios condutores convencionais, conectados do retorno das lâmpadas aos relês, os condutores foram direcionados para o local onde o protótipo de módulos integrados se encontra, foi necessário um condutor fase para conectar nos relês do módulo de quatro canais. Como os interruptores das lâmpadas são simples, eles somente poderão ser usados quando o determinado relê estiver desativado ou desligado a lâmpada ou quando a domótica esteja desativada. Caso o interruptor usado seja paralelo “*three-way*” a lâmpada poderá ser ligada ou desligada tanto no interruptor “físico” quanto na domótica, neste caso não será possível ouvir o sinal sonoro correto já que o estado do interruptor é indefinido, neste caso fica inviável o uso sonoro dos áudios correspondentes aos estados ligado ou desligado. Observa-se nas figuras 56 e 57 o esquema de instalação na residência para testes em versões de planta baixa e 3d.

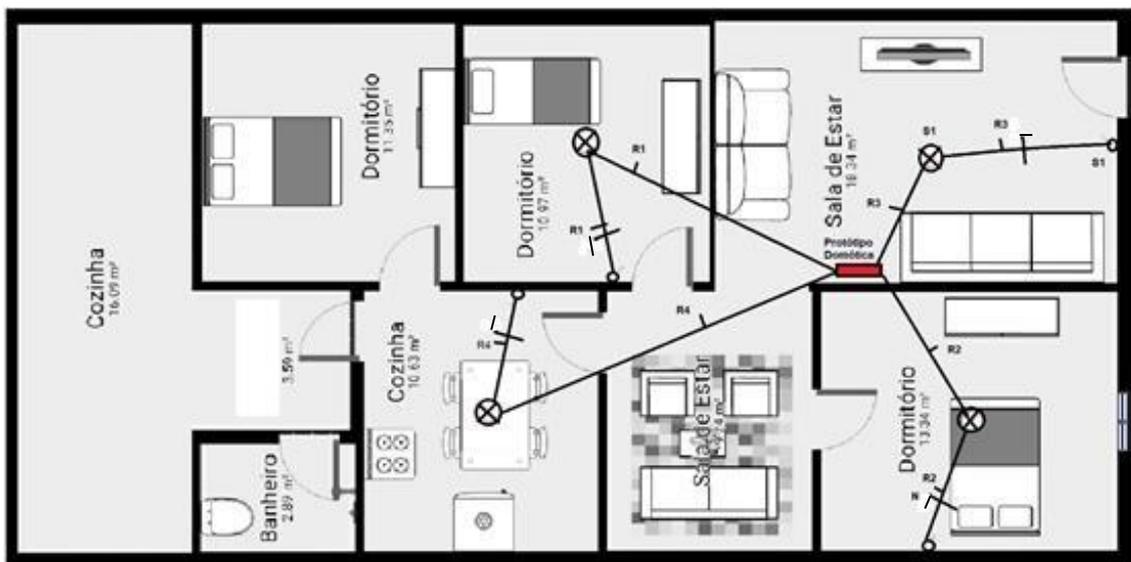


Figura 56: Instalação do circuito de iluminação na domótica (magicplan).

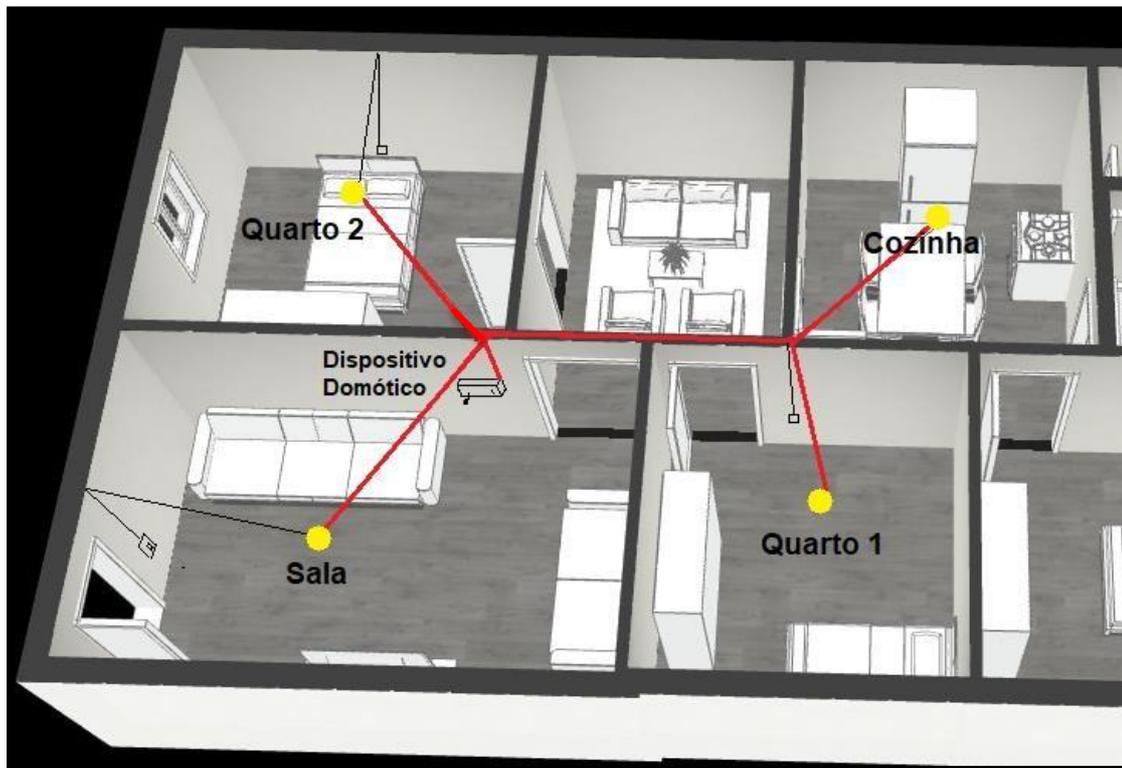


Figura 57: Instalação da domótica, representação em 3D (montagem no *magicplan*).

4.4.2 Instalação dos Componentes do Protótipo de Domótica

O esquema geral do protótipo de domótica com os principais componentes é mostrado na figura 58 a seguir.

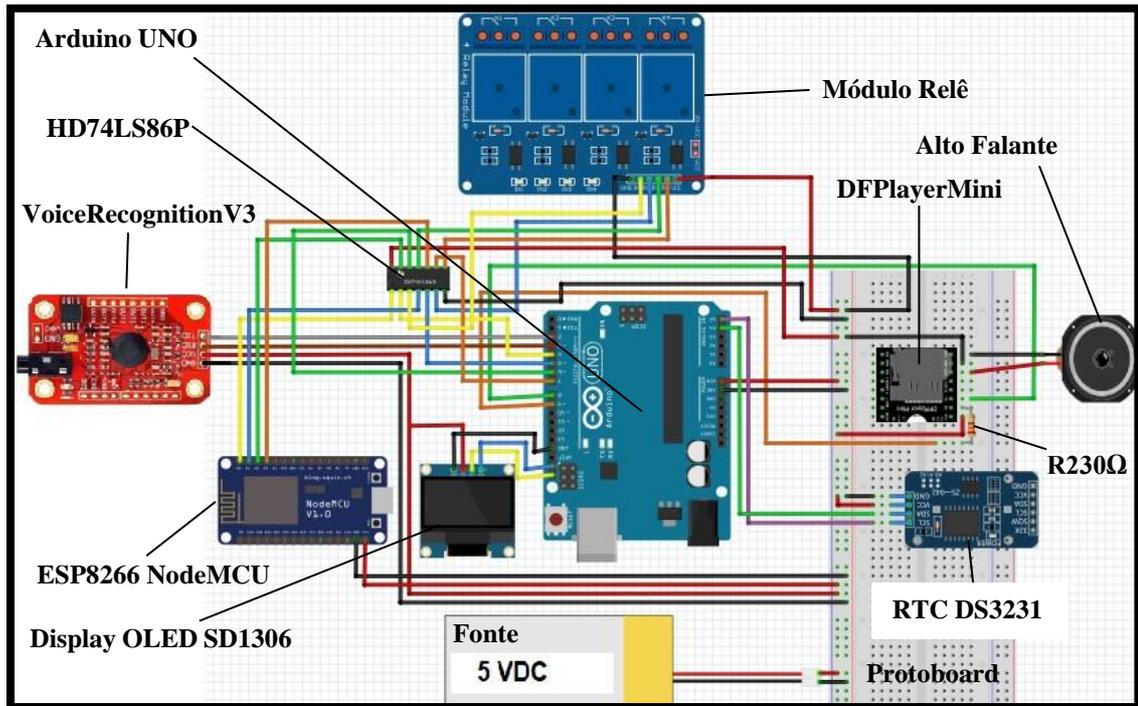


Figura 58: Esquema completo do Protótipo (*fritzing*).

4.4.3 Código de Programação

A programação em C++ foi dividida três partes: A geral gravada no arduino UNO; a de treinamento de vozes e a gravada no ESP8266 NodeMCU ESP12E para controlá-lo após configurar o *wifi* local e o endereço “AuthToken” do aplicativo “Blynk”.

Segue abaixo o link disponível para acesso:

<https://drive.google.com/file/d/1RnatKv_DFHBPsocIwx46X4oyMbgBQ_Ag/view?usp=sharing>.

4.5 LEVANTAMENTO E COMPARAÇÃO DE CUSTOS

O custo do protótipo da domótica depende de vários fatores, dois deles são a quantidade de aparelhos a ser conectados e do tamanho da residência, já que o método utilizado foi a conexão de condutores que vão do aparelho até o local do dispositivo domótico, e uma quantidade limitada de aparelhos definem o custo do protótipo.

Vários produtos já conhecidos no mercado diferem em alguns aspectos como: o reconhecimento de voz independente de treinamento, devido seu banco de dados possuir uma base capaz de reconhecer qualquer voz sem ser treinada, isso pode gerar uma certa desvantagem, já que o “protótipo” possui maior segurança, por reconhecer com precisão apenas a voz que foi treinada.

Alguns assistentes de voz necessitam de produtos que usam a mesma tecnologia para poder controlá-los, isso gera maiores custos. Um exemplo são as lâmpadas inteligentes que possuem tecnologia *wifi*, elas podem ser controladas remotamente por assistentes de voz e possuem preços mais elevados, se comparadas a lâmpadas normais de uma residência. Outra vantagem do “protótipo” é controlar lâmpadas por voz sem a necessidade de internet, isso leva maior facilidade de implantação do sistema em regiões sem acesso à internet.

Além das lâmpadas inteligentes também existem: tvs smart, aparelhos de som, cafeteiras, termômetros, alarmes, sistemas de hidromassagens em piscinas, entre outros, que funcionam por internet sem fio com custos muito elevados de obtenção.

Os preços de alguns componentes na tabela 2 abaixo são relacionados aos que estiveram disponíveis para a montagem do “protótipo” e não serem, necessariamente, as únicas opções econômicas disponíveis em mercado, podendo ser usadas outras opções a gosto do cliente ou usuário e na tabela 3 observa-se a comparação do custo comercial do protótipo com outras tecnologias com funcionalidades semelhantes nas mesmas condições de controle de 4 lâmpadas.

Componentes do Protótipo	Valor em reais
Módulo de reconhecimento de voz V3	R\$ 149,00
Módulo DFPlayerMini	R\$ 11,00
Arduino UNO	R\$ 40,00
Módulo RTC DS3231	R\$ 22,00
Módulo Esp8266 Esp12E NodeMCU V3 Ch340	R\$ 29,00
Módulo Display OLED 128x64 0.96	R\$ 35,00
Módulo Relê de 4 canais	R\$ 25,00
Fios Jumpers 40 Vias Macho X Fêmea	R\$ 17,00
Protoboard	R\$ 12,00
Cartão SD 2Gb	R\$ 14,00
Caixa de som alto-falante	R\$ 27,00
CI HD74LS86P-XOR	R\$ 2,43
Carregador de celular 5V	R\$ 26,90
Cabo USB/Micro-USB	R\$ 7,00
Conector P2 fêmea	R\$ 0,92
Mini Chave Gangorra KCD11-101 2 Terminais	R\$ 1,37
Mini Chave Alavanca SMTS102 2 Posições	R\$ 3,90
Chave Tecla KZJ-2x2 com Trava 6 Terminais	R\$ 0,90
Resistor 230Ω	R\$ 0,06
Fio 1,5 mm cabo paralelo 8 m	R\$ 16,32
Total	R\$ 440,08

Tabela 2 - Custo geral do protótipo da domótica.

Protótipo e produtos com funcionalidades semelhantes	Custo separado	Custo Total
Protótipo da domótica + construção do sistema	R\$ 440,08 + R\$ 40,00	R\$ 480,80
Smart Speaker 3ª geração com Alexa (REDAÇÃO – ZOOM) + 4 Lâmpadas WI FI Elgin	R\$ 330,00 + 4x R\$ 65,00	R\$ 590,00
Smart Speaker 2º geração Com Google Assistente (REDAÇÃO – ZOOM) + 4 Lâmpadas inteligentes Positivo	R\$ 269,23 + 4x R\$ 75,00	R\$ 569,23
Mibo Home HUB AGH 3001 INTELBRAS com alexa (REDAÇÃO – ZOOM) + 4 Lâmpadas inteligentes Positivo	R\$ 516,89 + 4x R\$ 75,00	R\$ 816,89
Smart speak com alexa (REDAÇÃO – ZOOM) + 1 interruptor wifi de 3 botões + 3 Lâmpadas WI FI Elgin	R\$ 399,00 + R\$ 139,99 + 3x R\$ 65,00	R\$ 733,99

Tabela 3 - Comparação de preços com sistemas domóticos semelhantes (Autoria própria).

5. CONCLUSÃO

A construção do protótipo de domótica inteligente trouxe uma nova experiência em construção de um dispositivo tecnológico, afim trazer maior acessibilidade à população local que não possui conhecimento de tal tecnologia e de exibir uma aplicação prática do curso de automação industrial.

Ao final dos testes e com o funcionamento conforme o planejado foi feita uma avaliação geral do custo total dos componentes e comparado a funcionamentos semelhantes de outros sistemas, obtendo maior vantagem econômica. Foi feita uma soma do custo da montagem e desenvolvimento do protótipo, o valor foi baseado no da formatação de computadores portáteis como *notebooks*.

No protótipo poderia ser programado outras funções que não foram oportunas ao escopo do presente trabalho. Funções como: comandos de voz extras para reproduzir músicas armazenadas, baixar ou aumentar volume, próxima música, repetir ou musica anterior podem ser adicionadas ao protótipo da domótica. Além disso, comandos programados para acionar aquecedores ou ar-condicionados quando a temperatura se encontrar fria ou quente, adicionar módulo *bluetooth* para controle sem fio e sem necessidade de internet, dentre outras ideias poderiam ser implementadas buscando a sua melhoria.

A tecnologia da domótica tem o objetivo de trazer benefícios como a substituição de tarefas manuais por automatizadas, gerando mais comodidade e menos necessidade de deslocamento do usuário e beneficiando pessoas portadores de necessidades especiais e as com mobilidade reduzida. Ao final, buscou-se neste trabalho, atrair o interesse da população para dar início a esse avanço tecnológico que se expande cada vez mais.

REFERÊNCIAS

ADAFRUIT. **RTClib**. github, [2020?]. Disponível em: <<https://github.com/adafruit/RTClib>> acesso em: 21 de abr. de 2021.

MAXIMINTEGRATED. **DS3231**. Datasheets, Extremely Accurate I2C-Integrated RTC/TCXO/Crystal, 2015. Disponível em: <<https://datasheets.maximintegrated.com/en/ds/DS3231.pdf>> acesso em: 21 de abr. de 2021.

ADMINWEBBIZZ. **Automação Residencial – o futuro já chegou**, 2017. Disponível em: <<https://innovhome.com.br/automacao-residencial-o-futuro-ja-chegou/#:~:text=A%20hist%C3%B3ria%20da%20automa%C3%A7%C3%A3o%20residencial%20A%20automa%C3%A7%C3%A3o%20residencial,j%C3%A1%20tomava%20impulso%20para%20revolucionar%20o%20mundo%20inteiro>> acesso em: 06 de jun. de 2021.

ALVES, J. A.; MOTA, J. **Casas Inteligentes**, 2003. Lisboa, Centro Atlântico.

MERCADOLIVRE. **Módulo Wifi Esp8266 Nodemcu V3 Lolin Com Chip Ch340g**, 2021. Disponível em: <https://produto.mercadolivre.com.br/MLB-814777718-modulo-wifi-esp8266-nodemcu-v3-lolin-com-chip-ch340g-_JM#position=1&search_layout=grid&type=item&tracking_id=9b3397b3-c4a0-4eeb-acc1-12b8ee7001f9> Acesso em 11 de set. de 2021.

AMPERKA. **DFPlayer Mini**. Datasheets, [2014?]. Disponível em: <http://files.amperka.ru/datasheets/DFPlayer_Mini.pdf> acesso em: 21 de abr. de 2021.

FILIPEFLOP. **Módulo MP3 DFPlayer Mini**, 2021. Disponível em: <<https://www.filipeflop.com/produto/modulo-mp3-dfplayer-mini/>> Acesso em 11 de set. de 2021.

FILIPEFLOP. **Real Time Clock RTC DS3231**, 2021. Disponível em: <<https://www.filipeflop.com/produto/real-time-clock-rtc-ds3231/>> Acesso em 11 de set. de 2021.

BLYNK IOT PLATFORM, **blynk-library**. Disponível em:

<<https://github.com/blynk/blynk-library>> acesso em 21 de abr. de 2021.

COSMIN, L. Geeks of Technology, sentrybrasil, 2017. **ESCOLHENDO O MELHOR ASSISTENTE DE VOZ PARA SUA CASA**. Disponível em:

<<https://sentrybrasil.com.br/cursos/escolhendo-o-melhor-assistente-de-voz-para-sua-casa/>> acesso em 12 de jun. de 2021.

IBGE. **PESSOAS COM DEFICIÊNCIA**. IBGE educa jovens. Disponível em:

<<https://educa.ibge.gov.br/jovens/conheca-o-brasil/populacao/20551-pessoas-com-deficiencia.html>> Acesso em 11 de set. de 2021.

DFROBOT. **DFRobotDFPlayerMini**. Disponível em

<<https://github.com/DFRobot/DFRobotDFPlayerMini>> acesso em: 21 de abr. de 2021.

DOMINGUES, Ricardo Gil. **A DOMÓTICA COMO TENDÊNCIA NA HABITAÇÃO: Aplicação em Habitações de Interesse Social com Suporte aos Idosos e Incapacitados**.

Orientador: Armando Carlos de Pina Filho, 2013 147f. Dissertação (Mestrado em Engenharia Urbana) – Universidade Federal do Rio de Janeiro, Escola Politécnica, Programa de Engenharia Urbana, Rio de Janeiro, 2013. Disponível em:

<<http://repositorio.poli.ufrj.br/dissertacoes/dissertpoli988.pdf>> acesso em: 14 de jun. de 2021.

ELECHOUSE, **VoiceRecognitionV3**, Disponível em:

<<https://github.com/elechouse/VoiceRecognitionV3>> acesso em: 21 de abr. de 2021.

ESPRESSIF INC. **ESP8266 Technical Reference**. Version 1.7 Copyright © 2020.

Disponível em: <https://www.espressif.com/sites/default/files/documentation/esp8266-technical_reference_en.pdf> acesso em 20 de maio de 2021.

HUANG, Stanley. **MultiLCD**, github, [2014?]. Disponível em:

<<https://github.com/stanleyhuangyc/MultiLCD/tree/master/MicroLCD>> acesso em: 21 de abr. de 2021.

HUGO, V.; COLADELO, R.; FERES, H. **Voz do Narrador**, Escolha tecnologia. Disponível

em: <<https://appstore.escolhatecnologia.com.br/app/voz-do-narrador/?ref=link-json>> acesso em: 05 de maio de 2021.

LUZ, J. G.; FRESSATTI, W. **RECONHECIMENTO DE VOZ UTILIZANDO ARDUINO**, Artigo científico, Universidade Paranaense (Unipar) Paranavaí-PR. Disponível em: <<https://docplayer.com.br/8888741-Reconhecimento-de-voz-utilizando-arduino.html>> acesso em: 20 de jun. de 2021.

LUIZ, S. J. S.; FARINELLI, F. A. **DOMÓTICA - Automação Residencial e Casas Inteligentes com Arduíno e ESP8266**, São Paulo, Érica, 2019.

FREITAS, Jota. **ESP8266-exemplos**, Pinagem do ESP8266, 2019, Disponível em: <<https://github.com/jotajr/esp8266-exemplos>> acesso em: 09 de ago. de 2021.

MULTILÓGICA SHOP, **Arduino Guia Iniciante**, versão 1.0. disciplinas. Disponível em: <https://edisciplinas.usp.br/pluginfile.php/3252633/mod_resource/content/1/Guia_Arduino_Iniciante_Multilogica_Shop.pdf> acesso em: 21 de abr. de 2021.

REDAÇÃO – ZOOM. **O guia da casa inteligente: como automatizar sua casa**. Zoom, 2020. Disponível em: <<https://www.zoom.com.br/smart-speaker/deumzoom/casa-inteligente-o-que-precisa-como-montar>> acesso em: 13 de maio de 2021.

RENESAS. **HD74LS86P Datasheet-Renesas**. Datasheetq, 2005. Site de busca e download gratuito de componentes eletrônicos. Disponível em: <<https://www.datasheetq.com/datasheet-download/389561/1/Renesas/HD74LS86P>> acesso em: 24 de abr. de 2021.

SHEN, Wilson. **Voice Recognition Module V3**. Elechouse 2014, Rev. - C V3. Disponível em: <https://www.elechouse.com/elechouse/images/product/VR3/VR3_manual.pdf> acesso em: 22 de abr. de 2021.

SOLOMON SYSTECH. **cdn-shop adafruit. SSD1306**. Advance Information 128 x 64 Dot Matrix OLED/PLED Segment/Common Driver with Controller, datasheets, 2008. Disponível em: <<https://cdn-shop.adafruit.com/datasheets/SSD1306.pdf>> acesso em: 03 de maio de 2021.

SOUZA, U. R. dos S.; GOMES, F. F. B. **Control House: Uma Plataforma para a Utilização de um Controle Universal para Ambientes Residenciais**. RCT - Revista de Ciência e Tecnologia, v. 6. Departamento de Educação – Centro Universitário Regional do Brasil (UNIRB), Salvador - BA, 2020. Disponível em:

<<https://revista.ufr.br/rct/article/view/5705/2857>> acesso em: 18 de maio de 2021.

TECH SUL ELETRÔNICOS. **Arduino UNO com ATmega328 – R3**. Disponível em:

<<https://techsuleletronicos.com.br/product/arduino-uno-com-atmega328-r3/>> acesso em: 20 de jun. de 2021.

COSTA, Fabio. **História do Arduino – como surgiu esta incrível plataforma de prototipagem eletrônica**. 16 DE SETEMBRO DE 2020. Disponível em

<<https://fabiocosta.net/arduino/historia-do-arduino/>> acesso em 20 de jun. de 2021.

THOMSEN, Adilson. **O que é arduino?**, 2014. Disponível em:

<<https://www.filipeflop.com/blog/o-que-e-arduino/>> acesso em 20 de jun. de 2021.

TOCCI, Ronald J. **Sistemas digitais: princípios e aplicações** / Ronald J. Tocci, Neal S.

Widmer, Gregory L. Moss; revisão técnica Renato Giacomini; tradução Jorge Ritter. – 11. ed. – São Paulo: Pearson Prentice Hall, 2011.

VIANNA, Gabriel Pereira. **DOMÓTICA: AUTOMAÇÃO RESIDENCIAL COM BAIXO CUSTO UTILIZANDO O ARDUINO**, Orientador: Prof. MSc. Silvio Moraes de Oliveira. 2018. 63f. TCC (Bacharelado) Engenharia Elétrica. Centro universitário UNIFACVEST, LAGES 2018. Disponível em:

<https://www.unifacvest.edu.br/assets/uploads/files/arquivos/ddb87-6_gabriel_viana.pdf> acesso em: 13 de jun. de 2021.

ZAMBERLAN, Bernardo Cervo. **AUTOMAÇÃO RESIDENCIAL POR COMANDOS DE VOZ COM ARDUINO**. Orientador: Prof. Ms. Fábio Sarturi Prass. 2016. 34f. TCC

(bacharelado) - CURSO DE SISTEMAS DE INFORMAÇÃO, ANTONIO MENEGHETTI FACULDADE - AMF, Restinga Seca – RS, 2016. Disponível em:

<http://repositorio.faculdadeam.edu.br/xmlui/bitstream/handle/123456789/81/TCC_SI_Berna

rdo_Cervo_Zamberlan_2016_AMF.pdf?sequence=1&isAllowed=y> acesso em: 19 de jun. de 2021.

CABRAL, M. M. A.; CAMPOS, A. L. P. de S. **SISTEMAS DE AUTOMAÇÃO RESIDENCIAL DE BAIXO CUSTO: UMA REALIDADE POSSÍVEL**, 2008. Disponível em < <https://www2.ifrn.edu.br/ojs/index.php/HOLOS/article/view/173/160> > Acesso em 19 de jun. de 2021.

APÊNDICE A – IMAGEM DO PROTÓTIPO

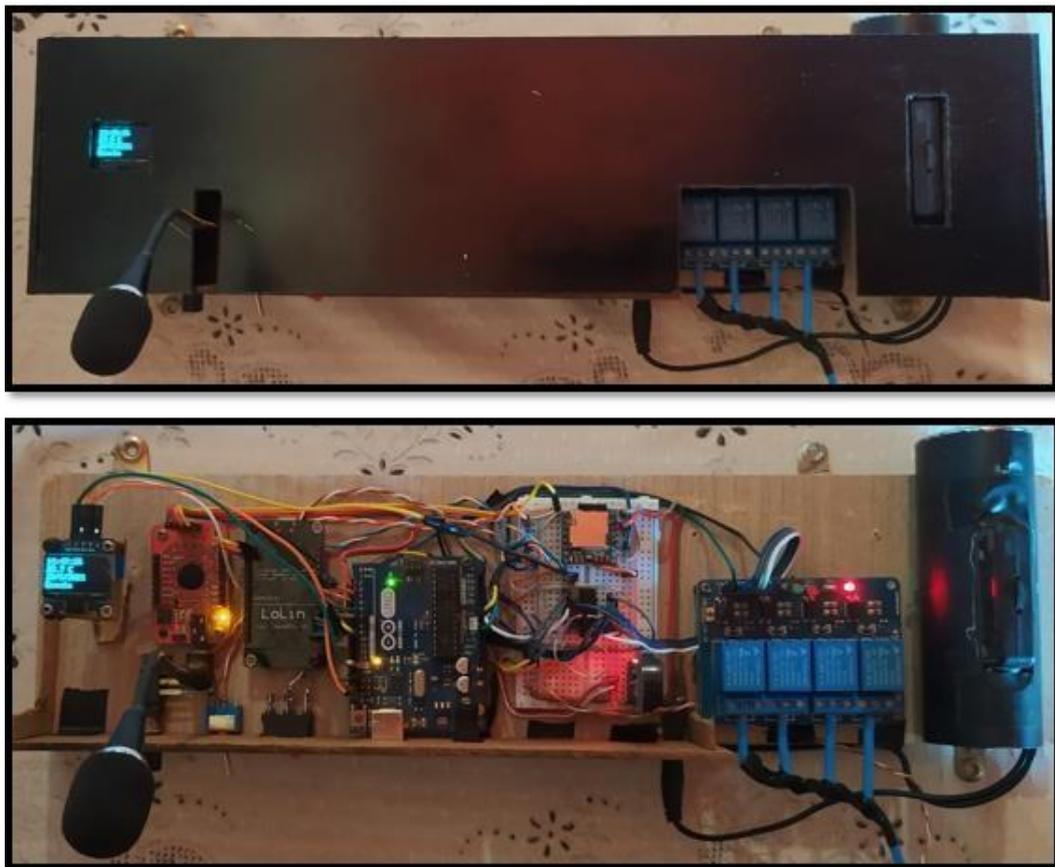


Figura 59: Protótipo montado com e sem o suporte (autoria própria).