

**CENTRO UNIVERSITÁRIO INTERNACIONAL UNINTER
ESCOLA SUPERIOR POLITÉCNICA
BACHARELADO EM ENGENHARIA DA COMPUTAÇÃO**

**ANDRÉ GUILHERME DOS SANTOS STOCCHERO
LUCAS FABRÍCIO STOCCHERO**

ROBO PET

TRABALHO DE CONCLUSÃO DE CURSO

CURITIBA

2021

ANDRÉ GUILHERME DOS SANTOS STOCCHERO
LUCAS FABRICIO STOCCHERO

ROBO PET

Trabalho de Conclusão de Curso
apresentado como requisito parcial à obtenção
do título de Bacharel em Engenharia da
Computação pela Escola Superior Politécnica
do Centro Universitário Internacional Uninter.

Orientador: Prof. Me. Charles Way Hun
Fung.

CURITIBA
2021

Este projeto é dedicado aos nossos familiares, amigos e professores que sempre nos incentivaram durante este período.

AGRADECIMENTOS

Agradecemos primeiramente a Deus por nos permitir estar sempre evoluindo como ser humano, nos dando sabedoria para enfrentar os desafios impostos e nos fortalecendo continuamente.

Também registramos o nosso reconhecimento aos professores do curso que sempre se mostraram dispostos a nos orientar nos mais diversos assuntos e principalmente naqueles pertinentes a este projeto, em especial ao nosso professor orientador Charles que detém um vasto conhecimento em diversas áreas, sendo muito solícito nas necessidades do grupo.

Aos nossos familiares e amigos que sempre estiveram ao nosso lado nos apoiando também a nossa gratidão, pois são nossos pilares e nos mantêm em pé diante das adversidades da vida.

“Para ser grande, às vezes é necessário
correr riscos enormes”. (GATES, Bill)

RESUMO

STOCCHERO, André Guilherme dos Santos; STOCCHERO, Lucas Fabricio. **ROBO PET**: Trabalho de Conclusão de Curso (Bacharelado em Engenharia da Computação) – Centro Universitário Internacional Uninter. Curitiba, 2021.

Desde o surgimento dos sistemas embarcados diversas áreas de estudos e aplicação puderam utilizar de seus benefícios. Uma forma para desenvolver aplicações para sistemas embarcados é utilizar componentes que venham a fornecer uma estrutura de hardware mais software. Os microcomputadores da Raspberry Pi conseguem fornecer essas características, sendo que suas placas suportam até versões compactas de sistemas operacionais. Com esse dispositivo de hardware é possível desenvolver diferentes tipos de projetos robóticos, entre eles os robôs que imitam o comportamento de animais de estimação, os quais, devido à pandemia em que se encontra o mundo atualmente, estão sendo cada vez mais procurados, visando o entretenimento de adultos e crianças e ajudando a deixar o isolamento ainda mais divertido.

Palavras-chave: Raspberry PI, animais de estimação, isolamento.

ABSTRACT

STOCCHERO, André Guilherme dos Santos; STOCCHERO, Lucas Fabrício. **PET ROBOT:** Course Completion Work (Bachelor of Computer Engineering) - Uninter International University Center. Curitiba, 2020.

Since the emergence of embedded systems, several areas of study and application have been able to use its benefits. One way to develop applications for embedded systems is to use components that will provide a hardware structure plus software. Raspberry Pi microcontrollers are able to provide these characteristics, and their cards support even compact versions of operating systems. With this hardware device it is possible to develop different types of robotic projects, including robots that mimic the behavior of pets, which, due to the pandemic in which the world is currently, are being increasingly sought after, aiming at the entertainment for adults and children and helping to make isolation even more fun.

Keywords: Raspberry PI, pets, isolation.

LISTA DE FIGURAS

Figura 1: Ponte H: Arranjo de chaves em H.	21
Figura 2: Motor DC.	24
Figura 3: Esquemático L298N.	25
Figura 4: Ponte H.	25
Figura 5: Sensor LDR e sua simbologia.	26
Figura 6: <i>Raspberry</i> Pi.	28
Figura 7: Pinagem do <i>Raspberry</i>	28
Figura 8: Display OLED.	29
Figura 9: Diagrama de blocos - Visão Geral do Sistema.	30
Figura 10: Diagrama de blocos do protótipo.	32
Figura 11: Base robótica.	33
Figura 12: Conexão entre ponte H e os motores DC.	34
Figura 13: Bateria 9V.	35
Figura 14: Powerbank 4400mAh.	35
Figura 15: Esquemático dos componentes.	36
Figura 16: Circuito LDR.	36
Figura 17: Esquemático do circuito.	37
Figura 18: Diagrama de caso de uso do sistema.	40
Figura 19: Diagrama de sequência do sistema.	41
Figura 20: Portas de comunicação I2C na <i>Raspberry</i>	42
Figura 21: Resultado com maior luminosidade.	43
Figura 22: Resultado com menor luminosidade.	43
Figura 23: Primeira etapa de movimentação.	44
Figura 24: Terceira etapa de movimentação.	46
Figura 25: Fluxograma do funcionamento do sistema.	47
Figura 26: Varredura do I2C.	48
Figura 27: Imagem display 1.	48
Figura 28: Imagem display 2.	48
Figura 29: Imagem display 3.	49
Figura 30: Imagem display 4.	49
Figura 31: Imagem display 5.	49
Figura 32: Imagem display 6.	49

Figura 33: Imagem display 7.	50
Figura 34: Imagem circuito com display.	50
Figura 35: Protótipo finalizado.	54
Figura 36: Protótipo finalizado.	54
Figura 37: Rosto inicial sem movimento.	55
Figura 38: Rosto triste mais movimento.	55
Figura 39: Rosto vergonhoso mais movimento.	56
Figura 40: Rosto duvidoso mais movimento.	56
Figura 41: Rosto bravo mais movimento.	57
Figura 42: Rosto feliz mais movimento.	57
Figura 43: Rosto manhoso mais movimento.	58

LISTA DE QUADROS

Quadro 1: Requisitos de Hardware.	31
Quadro 2: Requisito Funcional - Envio de comandos para o protótipo.....	36
Quadro 3: Requisito Funcional – Recebendo os dados do microfone.....	36
Quadro 4: Requisito Funcional – Movimentação do protótipo.	37
Quadro 5: Requisito Funcional - Recebendo dados do LDR.	37
Quadro 6: Requisito Funcional – Exibição das reações	37
Quadro 7: Requisito Não Funcional - Instalação do OpenCV	38
Quadro 8: Requisito Não Funcional - Instalação do Speech Recognitiom.	38
Quadro 9: Requisito Não Funcional – Biblioteca conversor AD	38

LISTA DE TABELAS

Tabela 1: Especificações do motor DC.....	22
Tabela 2: Características L298N.....	23
Tabela 3: Relação de entradas e saídas.....	24
Tabela 4: Tabela de característica da Raspberry.	26
Tabela 5: Parte móvel do sistema..	31
Tabela 6: Relação dos terminais de GPIO e da ponte H.....	32
Tabela 7: Relação dos terminais da ponte H com os motores.	33
Tabela 8: Relação das conexões entre sensor LDR e Raspberry.....	34
Tabela 9: Reações do robô.....	40
Tabela 10: Reação do robô ao não compreender um comando... ..	41
Tabela 11: Ações do robô com sua movimentação... ..	42
Tabela 12: Ação do robô não compreendida... ..	48
Tabela 13: Teste 1 – Ambiente isolado... ..	49
Tabela 14: Teste 2 – Ambiente isolado... ..	49
Tabela 15: Teste 3 – Ambiente isolado... ..	49
Tabela 16: Teste 4 – Ambiente com pessoas... ..	50
Tabela 17: Teste 5 – Ambiente com pessoas... ..	50
Tabela 18: Teste 6 – Ambiente com pessoas... ..	50

LISTA DE SIGLAS

AD	Conversor analógico/digital (Digital analog converter)
API	Interface de Programação de Aplicativos (<i>Application Programming Interface</i>)
IOT	Internet das Coisas (Internet of Things)
LCD	Display de Cristal Líquido (Liquid Crystal Display)
IOS	Sistema Operacional da Apple
DC	Corrente Direta/Contínua (<i>Direct Current</i>)
GPIO	Entrada/Saída de Propósito Geral (<i>General Purpose Input/Output</i>)
HDMI	Interface multimídia de alta definição (<i>High-Definition Multimedia Interface</i>)
SO	Sistema Operacional (Operational System)
USB	Porta Serial Universal (<i>Universal Serial Bus</i>)
OLED	Diodo Emissor de Luz (<i>Organic Light-Emitting Diode</i>)

LISTA DE ACRÔNIMOS

BIT	Dígito Binário (<i>Binary Digit</i>)
DOM	Modelo de Objeto de Documento (<i>Document Object Model</i>)
WIFI	Fidelidade sem Fio (<i>Wireless Fidelity</i>)

SUMÁRIO

1.	INTRODUÇÃO	16
1.0	PROBLEMA.....	17
1.1	JUSTIFICATIVA	17
1.2	OBJETIVO GERAL	18
1.3	OBJETIVOS ESPECÍFICOS	18
2.	FUNDAMENTAÇÃO TEÓRICA	19
2.1	RECONHECIMENTO DE VOZ	19
2.2	TECNOLOGIAS POR COMANDO DE VOZ.....	19
2.3	A LINGUAGEM DE PROGRAMAÇÃO PYTHON	20
2.4	PONTE H.....	21
2.5	SISTEMA OPERACIONAL.....	21
3.	MATERIAIS E MÉTODOS	22
3.1	MATERIAIS.....	23
3.1.1	HARDWARE	23
3.1.1.1	MOTOR DC	23
3.1.1.2	L298N.....	24
3.1.1.3	SENSOR LDR	26
3.1.1.4	<i>RASPBERRY PI</i>	26
3.2	METODOLOGIA	30
3.2.1	Requisitos de <i>Hardware</i>	31
3.2.2	Especificações de <i>Hardware</i>	32
3.2.3	<i>SOFTWARE</i>	38
3.2.3.1	Requisitos Funcionais	38
3.2.3.2	Requisitos Não Funcionais.....	40
3.2.3.3	Caso de Uso	40
3.2.3.4	Interface Gráfica.....	41
3.2.3.5	Instalação LDR.....	43
3.2.3.6	Movimentação do Protótipo	44
4	RESULTADOS E DISCUSSÕES	48
4.1.	TESTE OLED	49

4.2.	TESTE DE AUTONOMIA DA BATERIA	51
4.3.	TESTE DO MICROFONE	52
4.4.	PROTÓTIPO FINALIZADO.....	54
5	CONCLUSÃO	60
5.1	PRINCIPAIS DIFICULDADES	60
5.2	MELHORIAS FUTURAS	61

1. INTRODUÇÃO

As novas tecnologias surgem constantemente de forma a facilitar a vida dos usuários e aumentar o conforto nos diversos ambientes. Em conjunto a estas tecnologias, a Internet das Coisas (IoT, do inglês, Internet of Things) e sistemas embarcados vem ganhando cada vez mais espaço no mundo atual, permitindo a conexão entre ser humano e máquina (BRYNJOLFSSON e MCAFEE, 2014).

No último ano, o mundo tem passado por uma das fases mais difíceis já se enfrentadas, onde uma epidemia tem se alastrado mundialmente em todos os cantos, levando consigo infelizmente muitas mortes. Diante disso, o cuidado entre as pessoas tem sido cada vez maior, através de máscaras, álcool em gel, distanciamento social, fazendo com que muitas dessas pessoas tenham mudado suas rotinas, vindo a trabalharem em home office, por exemplo, ou estudarem através de aulas ao vivo com seus professores. Com essa situação, uma das alternativas encontradas para se ter uma boa companhia ou alguém que ajude a passar por tudo isso foi à adoção de animais de estimação. Houve um aumento significativo na busca pela adoção dos animaizinhos comparada a antes da pandemia, mostrando assim que esse foi um dos melhores caminhos a se seguir. Baseando nisso, a construção de um robô que venha imitar um animal de estimação veio a ser uma proposta, principalmente para aqueles que não podem adotar uma mascote de verdade, devido a não permissão de seus prédios, ou ainda problemas respiratórios, como alergias, por exemplo, ou também por terem filhos pequenos e não quererem que eles tenham um contato direto tão cedo.

PROBLEMA

Com o crescente desenvolvimento sobre sistemas com o qual as pessoas podem interagir com a tecnologia através de comandos de voz, gestos, entre outros, e com a miniaturização de sistemas convencionais, torna-se necessário uma tecnologia que possibilite uma melhor e mais rápida interação entre ser-humano e computador. Diante desse ano de 2021, agora em que uma pandemia se encontra criar uma tecnologia que ajude as pessoas no seu dia a dia, se distraindo e trazendo mais alegria para dentro de suas casas é uma boa alternativa, pois segundo pesquisas (encontradas em COVID-19 Resource The Center Of Lancet) podem surgir vários problemas de saúde, como depressão, estresse, o próprio medo de contrair o vírus, enfim, problemas esses que podem ser evitados ou diminuídos com o contato junto ao robô.

1.1 JUSTIFICATIVA

Cães, gatos e outros animais podem fazer bem para a saúde das pessoas, tanto a física quanto a mental. Afagar um animal de estimação parece ser um antídoto muito eficiente para encarar o dia a dia. Isso se faz ainda mais necessário e importante nos dias de hoje, onde se vive um tempo de distanciamento social devido à pandemia da corona vírus. Esses animaizinhos são capazes de elevar os níveis de hormônios e neurotransmissores responsáveis pela sensação de bem-estar e disposição e, com poucos minutos de interação já é capaz de reduzir o estresse e aumentar os níveis de hormônios relacionados ao prazer. Com isso, segundo entidades de proteção animal, a busca por adoções de pets aumentou durante o distanciamento social, cerca de 20% a 30% (GZH, 2020).

Com base nisso, foi realizada a construção de um robô que interage com o ser humano como um verdadeiro animal de estimação, demonstrando suas reações conforme o contato, ajudando a manter o convívio em isolamento. Este robô irá auxiliá-los, fazendo com que as crianças fiquem distraídas e se desenvolvendo através da curiosidade e contato com o objeto, além de evitarem que surjam algumas doenças, como alergias ao pelo do animal, por exemplo. Um estudo feito no Japão (EXAME, 2021) mostrou que esses robôs têm ajudado também universitários, os quais moram sozinhos e acabam tendo pouco contato com seus amigos e familiares, sendo então distraídos e confortados por esses novos parceiros.

1.2 OBJETIVO GERAL

O presente projeto propõe o desenvolvimento de um protótipo robótico que venha interagir com o usuário como um pet, ou seja, como um animalzinho de estimação, sendo possível a interação dos usuários por meio da voz mostrando reações em seu visor.

1.3 OBJETIVOS ESPECÍFICOS

- Desenvolver a montagem do circuito;
- Desenvolver a comunicação entre Raspberry e robô;
- Desenvolver uma interface de visualização da imagem formada;
- Desenvolver comunicação entre microfone e Raspberry;
- Integrar movimentação com interface gráfica (rosto do robô);
- Desenvolver a comunicação entre o Raspberry e o sensor LDR.

2. FUNDAMENTAÇÃO TEÓRICA

Para a utilização de robôs nas mais diversas tarefas é necessário que eles obtenham dados sobre o ambiente em que estão, como ruídos, vozes, entre outras. Desta forma as informações adquiridas são utilizadas como parâmetro para os cálculos programados.

2.1 RECONHECIMENTO DE VOZ

O processo de reconhecimento de voz requer fundamentos teóricos e práticos para sua realização. Falar deste processo de reconhecimento da fala em geral, resulta em uma vasta quantidade de temas, pois engloba matemática, informática, linguística, o idioma a ser trabalhado etc.

Vários são os tipos de sistemas que utilizam esse tipo de funcionamento, porém alguns deles são os mais conhecidos e usados atualmente, como por exemplo: Alexa da Amazon, que possui capacidade de interagir por voz, tocar músicas, fazer listas de tarefas, configurar alarmes, prover informações sobre tráfego, temperatura, entre outras informações, Google Assistant utilizado nos sistemas Google como, por exemplo: os celulares Android, e a Siri, sistema parecido com o assistente anterior, porém esse utilizados em sistemas IOS e Mac da Apple.

2.2 TECNOLOGIAS POR COMANDO DE VOZ

A área que faz estudo em tecnologias baseadas em comandos de voz está em constante crescimento, pois visa substituir, em certas aplicações, as interfaces tradicionais tais como teclados, controles, painéis, dentre outros dispositivos de mesma finalidade. Existe uma grande variedade de sistemas disponíveis no mercado com diferentes níveis de complexidade (DUBRIN, 2002). Algumas aplicações que esta tecnologia pode agregar, por exemplo, são: softwares de reconhecimento de voz que facilitam a acessibilidade ao computador ou a smartphones, no controle de ambiente acionado por voz (sistemas de iluminação, áudio, vídeo em automações residenciais) e até mesmo na mobilidade de pessoas que utilizam cadeiras de rodas comandadas através da fala.

Para a maior parte da população, este tipo de tecnologia poder ser ainda considerado como uma plataforma de lazer; para usuários com necessidades especiais e para pessoas que utilizam constantes tecnologias e atividades em excesso. O comando de voz aplicado a um assistente de tarefas pode facilitar muito a execução de atividades rotineiras (pesquisas na internet, notícias que envolvem interesses pessoais, dúvidas em relação a atividades de trabalho, inúmeras outras).

2.3 A LINGUAGEM DE PROGRAMAÇÃO PYTHON

Atualmente existem várias linguagens de programação, cada uma com seu foco, onde diferentes aplicações voltadas e diversas funções são realizadas. Ferramentas com uma estrutura de fácil aprendizado, e uma vasta gama de possibilidades de aplicações estão se tornando cada vez mais populares. Segundo (BORGES, 2014), Python é uma linguagem de programação de alto nível interpretada e orientada a objetos, onde os objetos possuem dados próprios e são definidos para estruturar o programa, sendo utilizada em desenvolvimento de aplicações web, jogos eletrônicos, big data, machine learning, IoT, entre muitas outras. A linguagem teve início em 1991, no Centro de Matemática e Computação (CWI), localizado na cidade de Amsterdam, na Holanda. Quem iniciou o desenvolvimento da linguagem foi o matemático Guido Van Rossum e o objetivo principal era a criação de uma linguagem interpretada com comandos simples de entender.

O Python é uma das linguagens mais populares e utilizadas por importantes empresas e renomados programadores da área de tecnologia da informação de todo o mundo. É uma linguagem de fácil aprendizado, pois apresenta-se como uma linguagem com um grande histórico de sucesso com simplicidade nos comandos e sintaxe intuitiva, possui uma documentação extensa de livros e manuais da linguagem e também a maioria das bibliotecas disponibilizadas são open source ou seja, permite uma inserção maior dos programadores em uma área e aplicação específica (MARQUES, 2011).

O Python possui uma grande capacidade de empacotamento que permite que módulos sejam reaproveitados e frameworks sejam construídos. Isso economiza tempo e o programador ganha qualidade de serviço ao usar módulos que já foram testados e validados por outros usuários. Esta linguagem funciona em diferentes plataformas e da mesma maneira em diferentes sistemas operacionais desde que seu interpretador seja instalado (PERKOVIC, 2016), sendo esta uma das razões da escolha desta linguagem como ferramenta deste trabalho.

2.4 PONTE H

A ponte H é um arranjo, em forma de “H”, de chaves que serve para inverter a polaridade de uma carga sem a necessidade de utilizar uma fonte simétrica. Tal arranjo é muito utilizado para acionamento de motores DC de modo bastante simples.

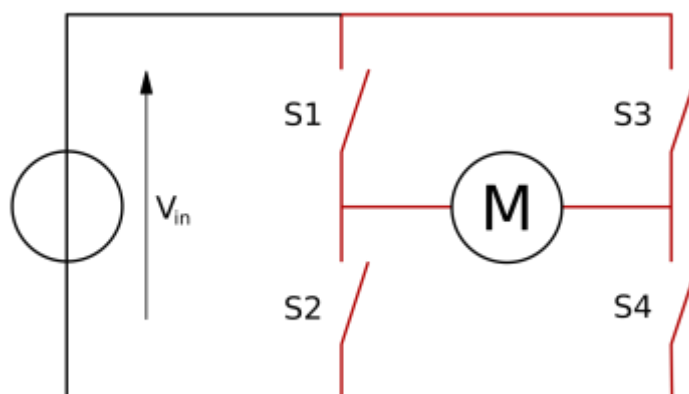


Figura 1: Ponte H: Arranjo de chaves em H.

Fonte: Embarcados.

O funcionamento da ponte H é bastante simples: fechando-se as chaves S1 e S4 tem-se o motor rodando em um sentido. Já com as chaves S3 e S2 acionadas, o sentido do fluxo da corrente sobre o motor é invertido, fazendo com que a rotação do motor também se inverta. Estes são os acionamentos convencionais.

Além destes dois acionamentos, há alguns estados bastante úteis: ligando-se S1 e S3 ou S2 e S4 pode-se gerar o efeito de freio motor, que permite reduzir o tempo de frenagem do motor sem necessidade de utilização de freios mecânicos convencionais. (EMBARCADOS, 2014).

2.5 SISTEMA OPERACIONAL

O sistema operacional pode ser entendido como um conjunto de softwares com o objetivo de administrar e gerenciar os recursos de um sistema, sendo eles componentes de hardware ou sistemas de arquivos de terceiros, estabelecendo assim uma interface entre a máquina e o usuário.

2.5.1 LINUX

Linux é um Sistema Operacional, assim como o Windows e o Mac OS, é baseado em Unix criado para desktops, mas também é usado em servidores, smartphones, tablets e outros tipos de dispositivos, incluindo caixas bancárias. Ao contrário de seus concorrentes mais famosos, o Linux não foi desenvolvido para fins comerciais e seu software e desenvolvimento são feitos em código aberto, o que significa que qualquer pessoa pode criar e distribuir aplicativos para ele.

A parte básica do Linux é composta de um kernel, software criado para fazer a comunicação de outros programas e traduzi-los em comandos para a unidade de processamento e outros componentes eletrônicos. Para funcionar, porém, também é necessário aplicativos e bibliotecas específicas para eles. (TECHTUDO, 2015).

Isto significa que um usuário de Linux pode escolher entre diversos aplicativos para executar a mesma função, sejam eles editores de texto, interfaces gráficas ou mesmo prompts de comando. (TECHTUDO, 2015).

2.5.2 RASPBIAN

O *Raspbian* é um sistema operacional livre baseado em Debian GNU Linux otimizado para o hardware *Raspberry Pi*. (PERICO *et al.* 2014). Possui um conjunto de programas básicos 41 e utilitários que possibilitam o funcionamento do *Raspberry Pi*. A construção inicial desses pacotes foi concluída em junho de 2012, contudo, o *Raspbian* ainda está em constante desenvolvimento, visando à melhoria da estabilidade e desempenho. O *Raspbian* é uma versão não oficial do *Debian* com configurações de compilação ajustadas para produzir código otimizado que é executado no *Raspberry Pi*. Isto proporciona um desempenho significativamente mais rápido para aplicativos que fazem uso intenso de operações aritméticas em ponto flutuantes. Todas as outras aplicações também ganham algum desempenho através do uso de instruções avançadas da CPU do *Raspberry Pi* (RASPBIAN, 2014).

3. MATERIAIS E MÉTODOS

Esta seção trata das informações referentes aos dispositivos de *hardware* e as tecnologias utilizadas no desenvolvimento do *software* da aplicação. Trata também da

organização do projeto, de como foi realizada a modelagem dos fluxos tendo como base os requisitos do sistema.

3.1 MATERIAIS

Para a construção física e lógica do protótipo alguns materiais foram necessários, os quais serão aqui apresentados.

3.1.1 *HARDWARE*

Esta seção trata de alguns componentes físicos do projeto: motores DC, ponte H, *Raspberry Pi* e o display.

3.1.1.1 MOTOR DC

O motor utilizado é o motor 3 a 6V com redução, possui 2 pinos de conexão e as seguintes especificações:

Tabela 1: Especificações do motor DC.

CARACTERÍSTICA	VALOR
Tensão nominal	3-6 V – DC
Velocidade sem carga	200 RPM (6V); 90 RPM (3V)
Corrente sem carga	200 mA (6V); 150 mA (3V)
Pino 1	VCC
Pino 2	GND

Fonte: Os Autores.



Figura 2: Motor DC.

Fonte: Os autores.

3.1.1.2 L298N

A ponte H é utilizada para determinar para qual sentido os motores rotacionarão, no protótipo é utilizada uma ponte, que é responsável pelo acionamento dos dois motores. Ela recebe sinais vindos da *Raspberry* e então realiza o chaveamento de acordo com os comandos recebidos.

Ela possui as seguintes características:

Tabela 2: Características L298N.

SÍMBOLO	PARÂMETRO	VALOR (máximo)
V_s	Tensão de alimentação	9 V
V_{ss}	Tensão de alimentação lógica em nível alto	3,3 V
	Tensão de alimentação lógica em nível baixo	1,5 V
V_i, V_{en}	Tensão de entrada e de ativação	-0,3 até 7 V
I	Corrente CC	4 A

Fonte: Adaptado de STMICROELECTRONICS.

Relação de entradas e saídas, juntamente com sua descrição:

Tabela 3: Relação de entradas e saídas.

NOME	FUNÇÃO
Sentido A e B	Possuem um resistor de carga conectado à terra para controlar a corrente
Input 1 e 2	Entradas do sentido A
Input 3 e 4	Entradas do sentido B
Out 1 e 2	Saídas do sentido A

Out 3 e 4	Saídas do sentido B
Vs	Tensão de alimentação nas saídas
Vss	Tensão de alimentação para os terminais lógicos
GND	Terra
EnA e EnB	Habilitam as entradas A e B respectivamente

Fonte: Adaptado de STMICROELECTRONICS.

O esquemático com a pinagem descrita na tabela anterior está na Figura 2 e a imagem do componente na Figura 3.

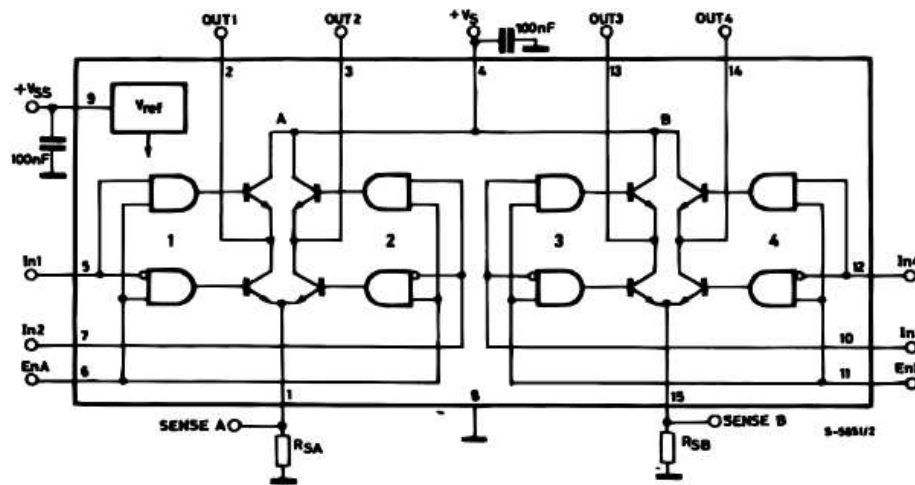


Figura 3: Esquemático L298N.

Fonte: STMICROELECTRONICS.

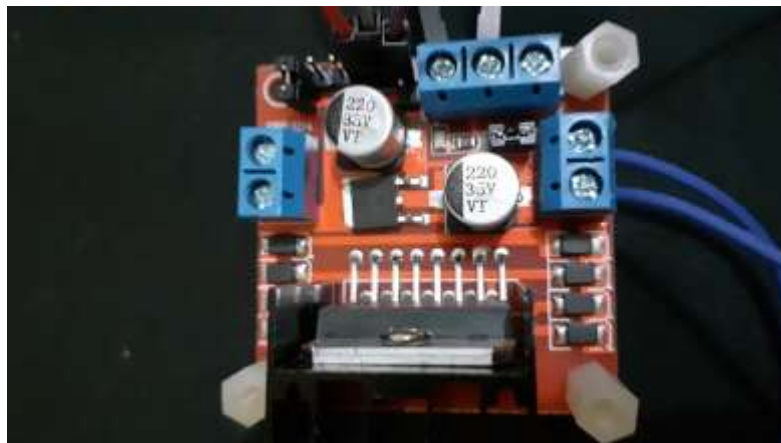


Figura 4: Ponte H.

Fonte: O Autores.

3.1.1.3 SENSOR LDR

O LDR, ou Resistor Dependente de Luz, é, basicamente, um resistor que varia sua resistência conforme a intensidade de luz que incide sobre ele. Os tipos mais comuns possuem maior sensibilidade à luz visível. Porém, existem variações que permitem maior sensibilidade à luz infravermelha. Possui apenas dois terminais e não tem polaridade. Além disso, um dos materiais presentes é um semicondutor de alta resistência. Uma informação importante, é que o LDR pode apresentar certos atrasos para variar a resistência, podendo demorar cerca de 10ms para detectar a mudança de um ambiente escuro para um claro. (MUNDO PROJETADO, 2017).

O funcionamento do LDR é relativamente simples! Quando as partículas de luz (fótons) incidem sobre a superfície do sensor, os elétrons que estão no material semicondutor são liberados, dessa forma a condutividade do LDR aumenta e a sua resistência diminui. (MANUAL DA ELETRÔNICA, 2017).

Em estado normal o material do LDR possui uma alta resistência, por isso quanto maior a incidência de luz sobre o sensor LDR, menor será a sua resistência! Ou seja, ao escurecer a resistência do LDR é máxima, e se o ambiente estiver claro a sua resistência será muito menor. (MANUAL DA ELETRÔNICA, 2017).



Figura 5: Sensor LDR e sua simbologia.

Fonte: (Wikipedia)

3.1.1.4 RASPBERRY PI

O Raspberry Pi é um minicomputador que possui o tamanho pouco maior de um cartão de crédito, foi desenvolvido pelo engenheiro britânico Eben Upton e sua equipe, então foi criada a Fundação Raspberry Pi. O intuito desse projeto é fornecer um meio de tecnologia de baixo

custo, sendo prático e acessível para que pessoas de diferentes idades pudessem aprender e desenvolver programas de modo intuitivo. (PORTUGAL, 2019).

O objetivo inicial do projeto era desenvolver e comercializar um computador de placa única, de tamanho reduzido e de fácil aquisição pelos interessados no assunto. Apesar do tamanho enxuto e das características pouco convencionais, o Raspberry PI é um computador como outro qualquer. Dependendo de sua finalidade, pode até ser utilizado como um computador de uso pessoal, tendo os mesmos recursos de computador convencional, como navegadores de internet, reprodução de conteúdo multimídia, ferramentas de edição de texto (Pacote LibreOffice). (RASPBERRY, 2019).

Os primeiros conceitos do Raspberry PI surgiram em 2006 por Upton e sua equipe, mas só em agosto de 2011 que os primeiros componentes começaram a ser distribuídos para o público em geral.

3.1.1.4.1 ARQUITETURA

O formato da arquitetura das placas do Raspberry Pi é do tipo SoC (System-on-a-Chip), integrando todos os componentes básicos de um computador e outros conectores eletrônicos. Ele possui processador, unidade de processamento gráfico (GPU), memória RAM, conectores para periféricos, como HDMI, USB, microUSB, jacks, conector de áudio, Ethernet, e módulo Wi-Fi. (RICHARDSON, WALLACE, 2013).

Conforme mencionado anteriormente, a *Raspberry* é um minicomputador que utiliza na maioria das vezes um sistema operacional *Linux*, e seus recursos são mais limitados que um computador comum. Algumas de suas características são:

Tabela 4: Tabela de característica da Raspberry.

ESPECIFICAÇÕES	CARACTERÍSTICA
Processador	Broadcom BCM2837B0, Cortex-A53 (ARMv8) SoC de 64 bits a 1.4GHz
Alimentação	5V / 2.5 ^a DC
Memória	SDRAM LPDDR2 de 1 GB
WIFI e Bluetooth	LAN sem fio de 2,4 GHz e 5 GHz IEEE 802.11.b / g / n / ac, Bluetooth 4.2
Portas USB	4 portas USB 2.0
GPIO	GPIO de 40 pinos

Fonte: Os Autores.

Na Figura 5 é exibido o *Raspberry Pi 3 Model B+*, que foi utilizado neste projeto. A Figura 6 mostra a numeração dos pinos, que vão de 1 a 40 e suas *labels*. Os pinos que podem ser programados para entrada ou saída de dados são os que possuem a *label* GPIO seguido de sua numeração específica.



Figura 6: *Raspberry Pi*.

Fonte: Os autores.

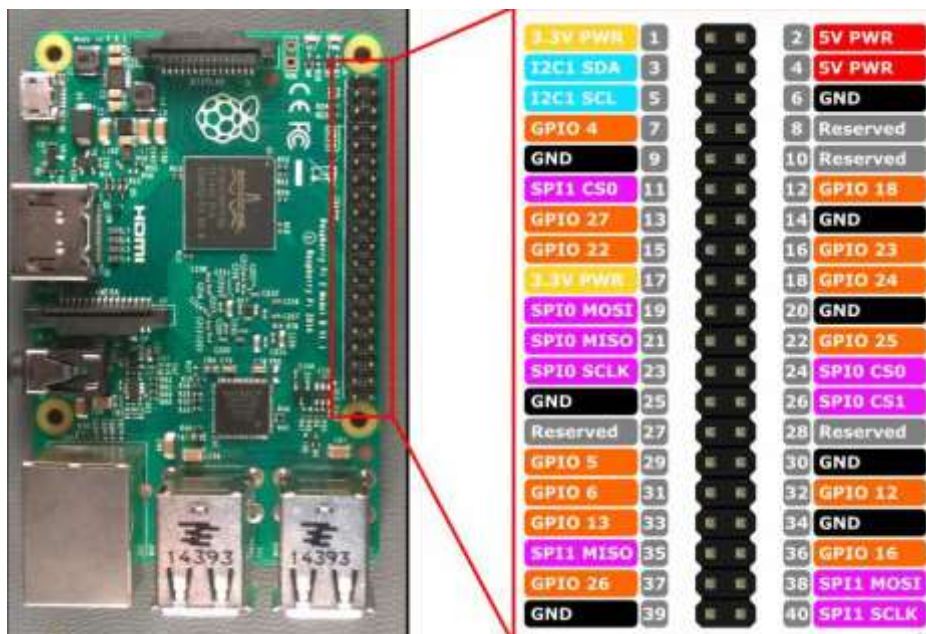


Figura 7: Pinagem do *Raspberry*.

Fonte: DO BIT AO BYTE.

3.1.1.5 DISPLAY OLED 0.96'

O OLED é um dispositivo de estado sólido composto por camadas finas de componentes orgânicos de espessura reduzida, cerca de 0,0001-0,0005 mm. O

funcionamento de um display OLED é baseado na eletroluminescência, a luz emitida por um material quando é irradiado a um campo elétrico.

A estrutura comum do OLED é baseada em cinco componentes. Na base, há um pedaço de vidro ou plástico transparente como substrato sobre o qual os dispositivos são depositados. Existem duas camadas orgânicas no meio, uma é chamada de camada orgânica condutora e a outra é chamada de camada orgânica emissora. Essas camadas estão localizadas em sequência entre dois eletrodos, um é o eletrodo positivo (ânodo) e o outro é o eletrodo negativo (cátodo).

Quando uma tensão é aplicada aos eletrodos do dispositivo, uma série de fenômenos físicos ocorrem, resultando na emissão de luz. A cor da luz gerada depende do tipo de plástico usado como camada orgânica emissora de luz. A intensidade e o brilho da luz emitida dependem da quantidade de tensão aplicada, quanto maior a voltagem, maior a intensidade e o brilho da luz (Oliveira & Falcão, 2015).

3.1.1.5.1. Especificações técnicas

O display apresenta as seguintes especificações técnicas:

- Interface: I2C;
- Tensão de operação: 3 á 5V;
- Nível lógico: 3,3 á 5V
- Cor: Azul e Amarela;
- Resolução: 128x64;
- Dimensões: 30 x 27 mm
- Pinos: GND, VCC, DAS e SCL.



Figura 8: Display OLED.

Fonte: Os autores.

3.2 METODOLOGIA

Após o embasamento histórico que ressaltou a influência e importância de robôs em nosso dia a dia, foram estudados os periféricos do projeto, visando entender o funcionamento de cada um dos componentes.

Esse projeto aborda um sistema com aspectos de *hardware* e *software* e todas as suas funções e ações dentro do protótipo.

Para o desenvolvimento de um *software*, independente da linguagem de programação ou da ferramenta de desenvolvimento, se faz necessário pré-definir todas as características do programa, de uma maneira simplificada contendo todas as suas funcionalidades, interação com o usuário e processamento das informações.

Após a definição dos parâmetros gerais do programa, foi escolhido a linguagem de programação *Python*, principalmente por possuir uma estrutura de código facilitada e de rápido aprendizado, também pela capacidade de executar tarefas complexas e integração com diversas bibliotecas de desenvolvimento.

A seguir está apresentado o diagrama de blocos que representa a visão geral do sistema:

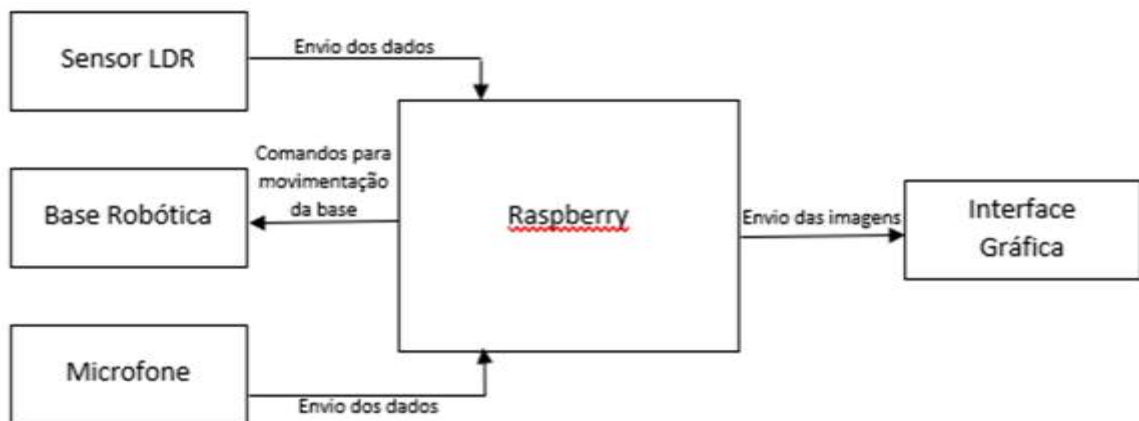


Figura 9: Diagrama de blocos - Visão Geral do Sistema.

Fonte: Os autores.

Cada bloco possui diferentes funcionalidades:

- O SENSOR LDR tem a responsabilidade de leitura de dados externos, processa e envia para o RASPBERRY;

- O MICROFONE também tem a função de receber dados de fonte externa e faz envio para o RASPBERRY;
- O RASPBERRY recebe e processa os dados do SENSOR e do MICROFONE, encaminha para a BASE ROBÓTICA e envia instruções para a INTERFACE GRÁFICA;
- A BASE ROBÓTICA recebe os comandos do RASPBERRY, com base nos dados que recebeu do SENSOR LDR e do MICROFONE.

3.2.1 Requisitos de *Hardware*

Para desenvolvimento do protótipo, alguns requisitos de hardware foram elaborados, conforme o quadro a seguir:

ID	Componente	Descrição
RH 01	Raspberry Pi 3 B	Funções: 1. Recebe informações do Microfone e do Sensor LDR; 2. Processa os dados coletados; 3. Envia os comandos para a base robótica executar o movimento.
RH 02	Sensor LDR	Captura os dados de luminosidade do ambiente externo.
RH 03	Base Robótica	Sustenta a parte física do protótipo.
RH 04	Microfone	Captura os dados de voz do ambiente externo.
RH 05	Motor DC	Movimenta o protótipo.

RH 06	Bateria 9V	Alimenta a Ponte H e os motores.
RH 07	Ponte H	Realizar o interfaceamento entre os motores e a Raspberry.
RH 08	Carregador Portátil (power bank)	Alimenta a Raspberry.
RH 09	Display OLED	Apresenta a interface gráfica do protótipo.
RH 10	Conversor AD	Responsável por fazer leitura do sensor LDR.

Quadro 1: Requisitos de Hardware.

Fonte: Os Autores.

Com base no Quadro 1, o protótipo é composto por: Raspberry, sensor LDR, base robótica, microfone, motor DC, bateria 9V, carregador portátil e o display OLED. A seguir o diagrama de blocos referente ao protótipo:

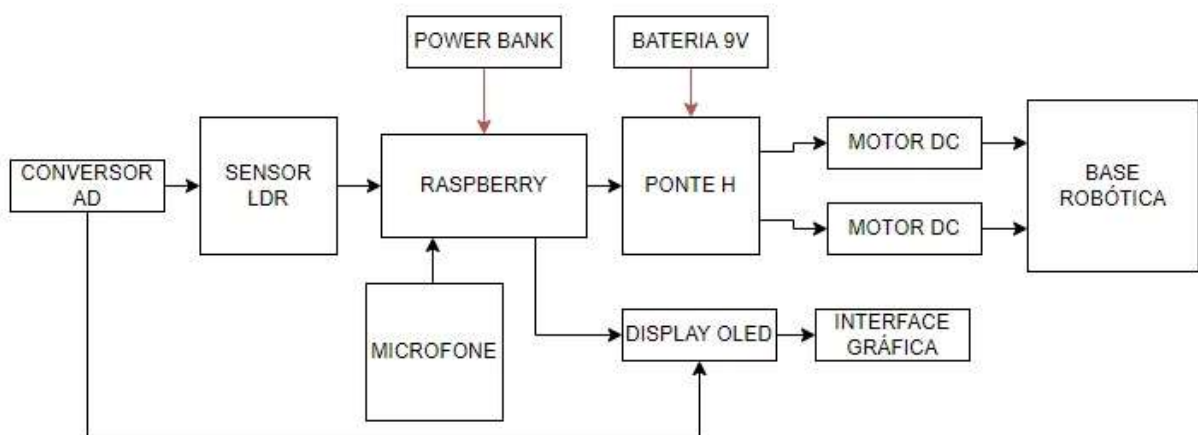


Figura 10: Diagrama de blocos do protótipo.

Fonte: Os Autores.

3.2.2 Especificações de *Hardware*

A base robótica representa a parte móvel do sistema, ela é composta pelos itens que estão relacionados na tabela a seguir:

Tabela 5: Parte móvel do sistema.

ITEM	FUNÇÃO
Plataforma de plástico	É utilizada como base para que os componentes que compõe o protótipo sejam acoplados.
Rodas	Duas rodas de borracha que reproduzem a rotação dos motores e por consequência movimentam o protótipo, e uma metálica que auxilia na direção da base.
Motores DC	Dois motores são utilizados para movimentar as rodas.

Fonte: Os Autores.

Através da imagem a seguir podemos ver a base robótica descrita na tabela anterior:



Figura 11: Base robótica.

Fonte: Os Autores.

Para que ocorra a interação do Raspberry com a ponte H, são utilizados os pinos do GPIO (Figura 7), eles são programados para enviar e receber dados. No caso deste projeto, são utilizados para controlar os comandos da ponte H.

A ponte H recebe os comandos que são enviados da Raspberry também pelos seus pinos de entrada (ENA, IN1, IN2, IN3, IN4 e ENB).

A tabela 6 mostra os pinos de entrada da Ponte H e os pinos de saída do Raspberry e suas respectivas relações.

Tabela 6: Relação dos terminais GPIO e da ponte H

NÚMERO DO GPIO	RASPBERRY TERMINAL	ENTRADA PONTE H
18	12	ENA
23	16	IN1
24	18	IN2
27	13	IN3
22	15	IN4
17	11	ENB

Fonte: Os Autores.

Após receber os comandos, o L298 (ponte H) realiza o chaveamento para o acionamento dos motores, resultando no movimento do protótipo.

Os motores são ativados através das saídas da ponte H: OUT1, OUT2, OUT3 e OUT4, como demonstrado na tabela 8 e na figura 11 a seguir:

Tabela 7: Relação dos terminais da ponte H com os motores.

SAÍDA PONTE H	PONTE H (TERMINAL)	MOTOR DC (TERMINAL)
OUT1	02	01
OUT2	03	02
OUT3	13	02
OUT4	14	01

Fonte: Os Autores.

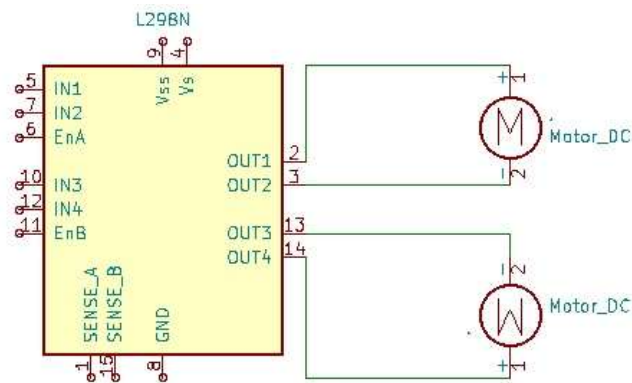


Figura 12: Conexão entre ponte H e os motores DC.

Fonte: Os Autores.

As saídas OUT1 e OUT4 são conectadas e ligadas na conexão positiva dos motores e as saídas OUT2 e OUT3 estão nas conexões negativas, tornando possível realizar a rotação em ambos os sentidos (horário e anti-horário).

Para alimentar os motores são necessários um valor mínimo de 3V e no máximo 6V, então foi utilizado uma bateria comum de 9V para alimentar toda a ponte H. Já na Raspberry são necessários 5V onde foi utilizado uma bateria portátil, também conhecida como Power Bank de 4400mAh.



Figura 13 - Bateria de 9V.

Fonte: Os Autores.



Figura 14 – Powerbank 4400mAh

Fonte: Os Autores.

Para atuar com o sensor LDR que possui seu funcionamento, a resistência alterada conforme a luminosidade sobre ele, quanto maior a quantidade de luz imposta menor a resistência elétrica e quanto menor for a luminosidade maior será a sua resistência.

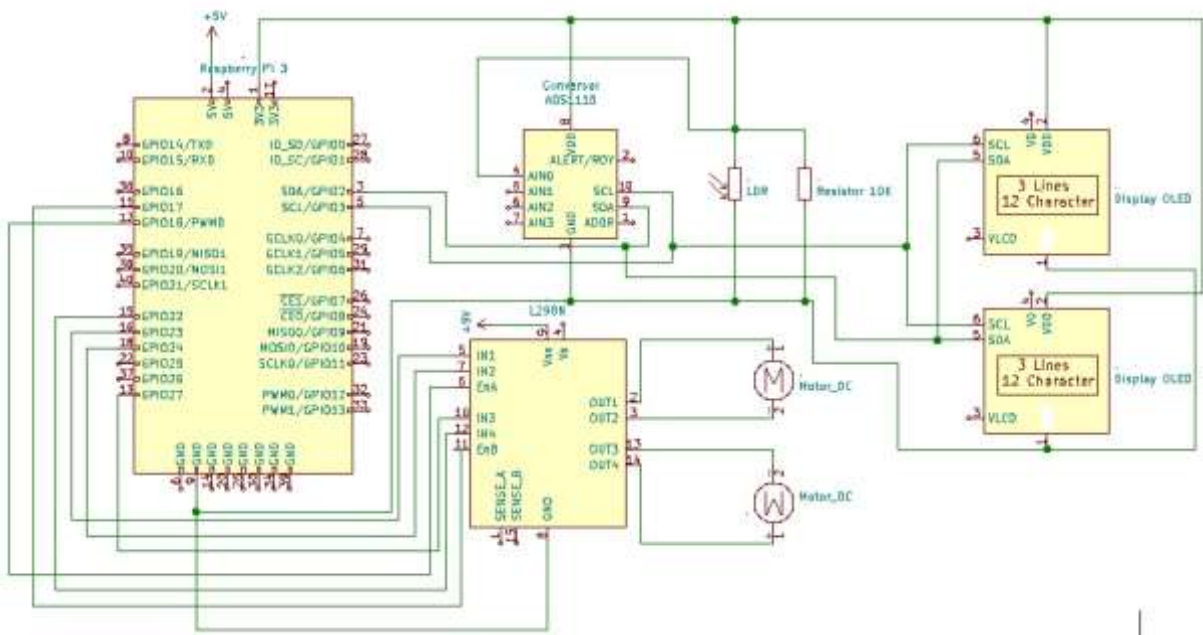
O sensor LDR é conectado em uma mini protoboard, onde ela tem suas conexões vindas da Raspberry pelos pinos GPIO, como demonstrado na tabela 8.

Tabela 8: Relação das conexões entre sensor LDR e Raspberry.

GPIO	RASPBERRY TERMINAL	ENTRADA CONVERSOR AD
3,3V	1	VDD
GND	9	GND
2	3	SCL
3	5	SDA

Fonte: Os Autores.

O esquemático que apresenta as conexões do sensor LDR, do conversor AD e do Raspberry está amostrado na Figura 15 abaixo:



2Figura 17 – Esquemático do circuito.

Fonte: Os Autores.

3.2.3 SOFTWARE

Nesta seção são abordados os itens relacionados aos *softwares* implementados no projeto, itens que nos mostram como é realizado o controle do protótipo e o processamento dos dados.

3.2.3.1 Requisitos Funcionais

Nesta etapa são apresentados os requisitos funcionais e não funcionais do sistema, através dos quadros a seguir:

Nº Requisito	RF 01	Título	Selecionar a direção do movimento.
Descrição	O usuário seleciona a direção para movimentação.		
Definição de Casos de Uso			
UC01	Título	Selecionar a direção do movimento.	
	Fluxo Principal	O usuário seleciona através do comando de voz a direção em que o protótipo deve se mover. “Para frente” – Frente; “Para trás” – Trás; “Para direita” – Direita; “Para esquerda” – Esquerda;	

	Fluxo(s) Alternativo(s)	Não realiza o movimento.
--	----------------------------	--------------------------

Quadro 2: Requisito Funcional - Envio de comandos para o protótipo.

Fonte: Os Autores.

Nº Requisito	RF 02	Título	Receber dados do microfone
Descrição	Recebendo dados do microfone		
Definição de Casos de Uso			
UC03	Título	Receber dados do microfone	
	Fluxo Principal	Sistema recebe os dados que são captados pelo microfone, a fim de realizar a movimentação do robô.	
	Fluxo(s) Alternativo(s)	Não realiza o movimento.	

Quadro 3: Requisito Funcional – Recebendo os dados do microfone.

Fonte: Os Autores.

Nº Requisito	RF 03	Título	Enviar o comando para o protótipo
Descrição	Envio do movimento selecionado pelo usuário		
Definição de Casos de Uso			
UC02	Título	Enviar o comando para o protótipo	
	Fluxo Principal	Sistema envia o comando para movimentação do protótipo com base na escolha do usuário.	
	Fluxo(s) Alternativo(s)	Não envia o comando.	

Quadro 4: Requisito Funcional – Movimentação do protótipo.

Fonte: Os Autores.

Nº Requisito	RF 04	Título	Receber dados do sensor LDR
Descrição	Recebimento dos dados vindos do sensor LDR		
Definição de Casos de Uso			
UC02	Título	Receber dados do sensor LDR	
	Fluxo Principal	Sistema recebe os dados que são captados pelo sensor LDR e reproduz a reação do robô.	
	Fluxo(s) Alternativo(s)	Não envia o comando.	

Quadro 5: Requisito Funcional – Recebendo dados do LDR.

Fonte: Os Autores.

Nº Requisito	RF 05	Título	Exibir a reação do robô
Descrição	Realizar a exibição das reações do robô		
Definição de Casos de Uso			
UC04	Título	Exibir a reação do robô	
	Fluxo Principal	Sistema exibe a reação do robô na interface gráfica conforme os dados que são recebidos.	
	Fluxo(s) Alternativo(s)	Não realiza a exibição.	

Quadro 6: Requisito Funcional – Exibição das reações.

Fonte: Os Autores.

3.2.3.2 Requisitos Não Funcionais

Nº Requisito	RNF 01	Título	Instalação do OpenCV Desktop
Descrição	Para o sistema funcionar, a versão da biblioteca OpenCV do python deve ser instalada.		

Quadro 7: Requisito Não Funcional - Instalação do OpenCV.

Fonte: Os Autores.

Nº Requisito	RNF 02	Título	Instalação do <i>Speech Recognition</i>
Descrição	Instalação do speech recognition para reconhecimento de voz.		

Quadro 8: Requisito Não Funcional - Instalação do Speech Recognition.

Fonte: Os Autores.

Nº Requisito	RNF 03	Título	Instalação da biblioteca do conversor analógico/digital
Descrição	Instalação da biblioteca para utilização do conversor analógico/digital.		

Quadro 9: Requisito Não Funcional – Biblioteca do conversor AD.

Fonte: Os Autores.

3.2.3.3 Caso de Uso

O diagrama de caso de uso a seguir nos mostra como o usuário interage com o sistema e a base robótica.

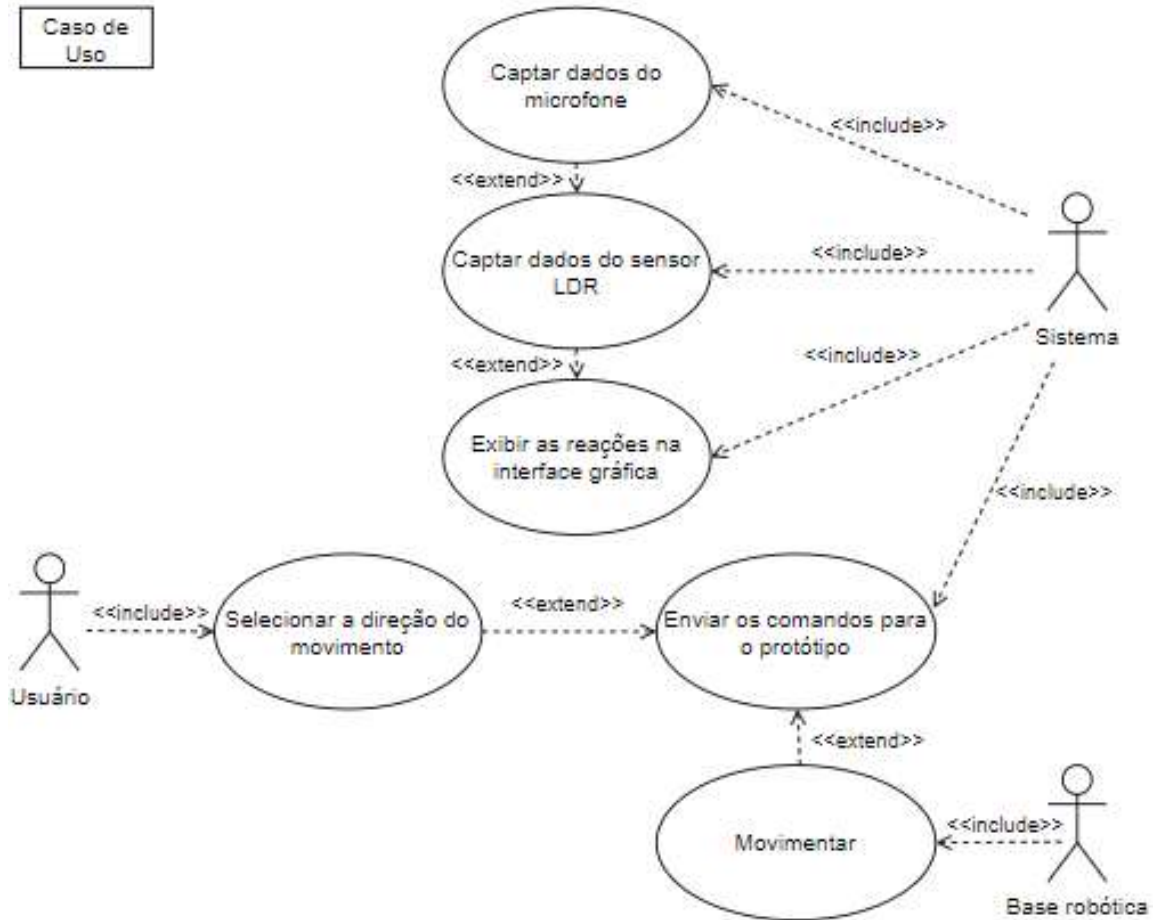


Figura 183: Diagrama Caso de Uso do sistema.

Fonte: Os Autores.

3.2.3.4 Interface Gráfica

A interface gráfica especificada aqui são os displays OLED, onde serão demonstradas as reações do robô, de acordo com as interações que ocorrerem com ele. Ou seja, funcionará como o rosto do robô.

O robô processa a informação vinda do microfone, faz a validação e compara com as funções programadas. Caso o comando recebido seja igual o programado na listagem de ações, o robô executa a ação vinculada e altera a face apresentada no display OLED. A seguir será demonstrado um diagrama de sequência para explicar esse funcionamento.

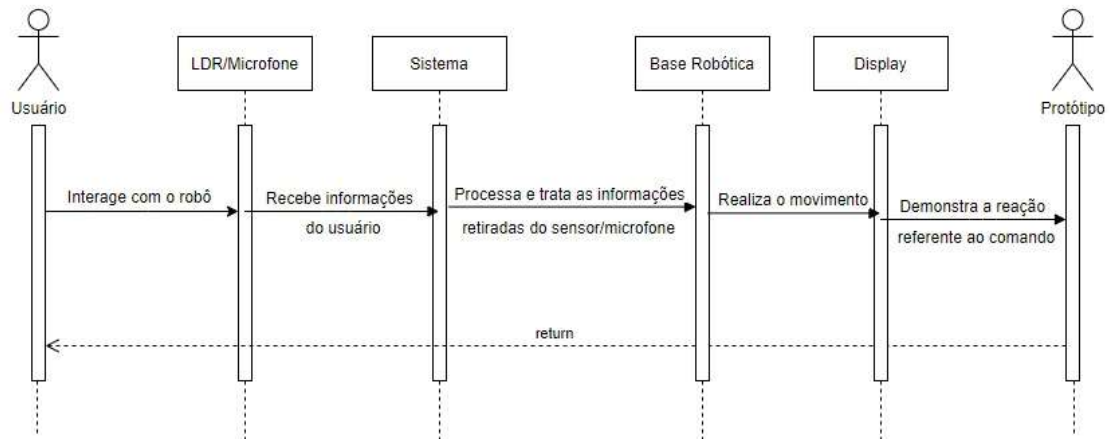


Figura 19: Diagrama de seqüência do sistema.

Fonte: Os autores.


Tabela 9: Reações do robô.

Comando de Voz	Interface Gráfica
“Frente” “Para frente”	
“Trás” “Para Trás”	
“Direita” “Para direita”	
“Esquerda” “Para esquerda”	
“Vem”	
“Vai”	
“Sai”	

Fonte: Os Autores.

E para o cenário de o robô não reconhecer o comando, a resposta do robô é:

Tabela 10: Reação do robô ao não compreender um comando.

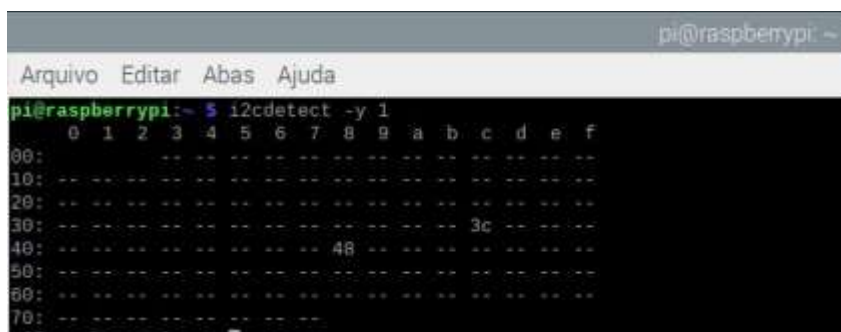
Interface Gráfica	Comando não compreendido
	Processa e demonstra a figura que corresponde ao duvidoso.

Fonte: Os Autores.

3.2.3.5 Instalação LDR

Para implementação do LDR, inicialmente é feita validação se o conversor está reconhecendo o componente. Além de saber as portas de comunicação I2C da *Raspberry* é de importante saber o endereço do dispositivo, visto que cada dispositivo possui um endereço para realizar a transmissão de dados.

A Figura 20, mostra a varredura I2C pelo *Raspberry* vindo do conversor AD coletado pelas entradas.



```

pi@raspberrypi: ~
Arquivo Editar Abas Ajuda
pi@raspberrypi:~$ i2cdetect -y 1
 0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00: -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- 3c -- -- -- --
40: -- -- -- -- -- -- -- -- 48 -- -- -- -- -- --
50: -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- -- -- -- -- -- -- -- --

```

Figura 20 – Portas de comunicação I2C na Raspberry.

Fonte: Os Autores.

Para o sensor de luz, após a captação, é necessário fazer a conversão A/D, pois os valores que são obtidos são valores analógicos, e para que o microcontrolador possa processá-los, é necessário que sejam convertidos em valores digitais. A partir daí, os valores que foram captados pelos sensores nas portas são atribuídos às variáveis para em seguida, ser feita a comparação. Toda essa situação foi tratada no script do projeto.

Nas Figuras abaixo, são amostrados os resultados quando o LDR recebe maior luminosidade e quando diminui a luminosidade no ambiente. Os valores alteram devido a alteração na resistência.

```
>>> %Run ldr.py
raw      v
1584    0.196
1568    0.194
1536    0.190
1392    0.174
1344    0.166
1216    0.152
1200    0.150
1040    0.130
560     0.066
592     0.072
544     0.070
```

Figura 21: Resultado com maior luminosidade.

Fonte: Os Autores.

```
>>> %Run ldr.py
raw      v
7648    0.956
7456    0.932
7536    0.946
7952    0.994
8016    1.002
8048    1.006
8032    1.004
```

Figura 22: Resultado com menor luminosidade.

Fonte: Os Autores.

3.2.3.6 Movimentação do Protótipo

O processo para movimentação da base robótica é baseado em três etapas, onde a primeira é a interação do usuário que deve realizar a ação de aproximar-se ao LDR para que altere a leitura da resistência feita pelo sistema. Assim que tiver a informação da resistência alterada, o protótipo irá realizar o movimento programado. O fluxograma a seguir demonstra a primeira etapa:

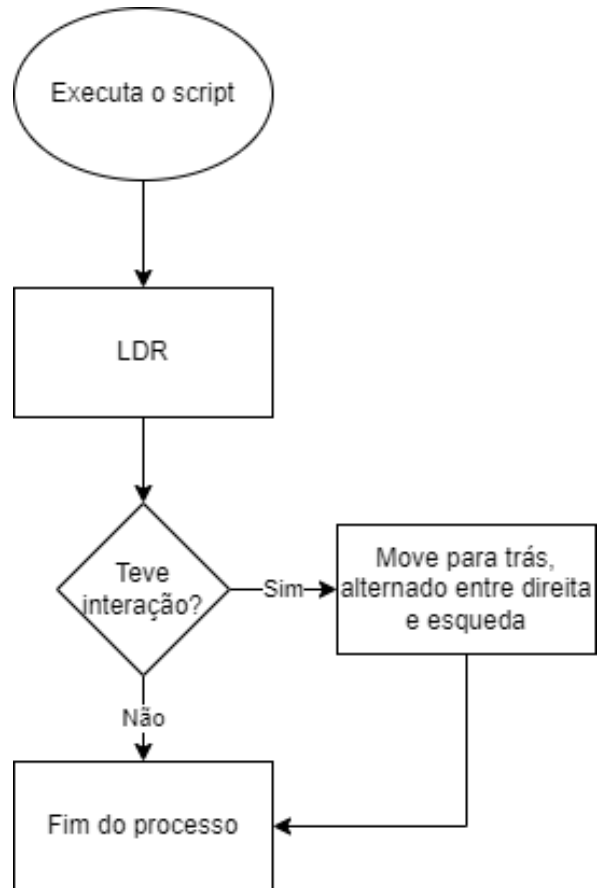


Figura 23: Primeira etapa de movimentação.

Fonte: Os autores.

A segunda etapa é a necessidade da interação do usuário que deve realizar a ação de fala referente ao movimento. Os comandos de voz que o sistema interpreta estão na Tabela 4.

Tabela 11 - Comandos e seus respectivos movimentos

Comando de Voz	Movimento
“Frente” “Para frente”	O protótipo se movimenta para frente.
“Trás” “Para Trás”	O protótipo se movimenta para trás.
“Direita” “Para direita”	O protótipo se movimenta para direita.
“Esquerda” “Para esquerda”	O protótipo se movimenta para esquerda.
“Vem”	O protótipo se movimenta para frente.

“Vai”	O protótipo se movimenta para trás.
“Sai”	O protótipo se movimenta para trás.

Fonte: Os autores.

A terceira etapa consiste na interpretação do sistema, onde o *script* principal, valida as informações do usuário coletadas através do microfone e caso comando seja um argumento válido ele executa o movimento de direção do protótipo, caso contrário ele retorna para um novo comando. Essa etapa está detalhada no fluxograma a seguir:

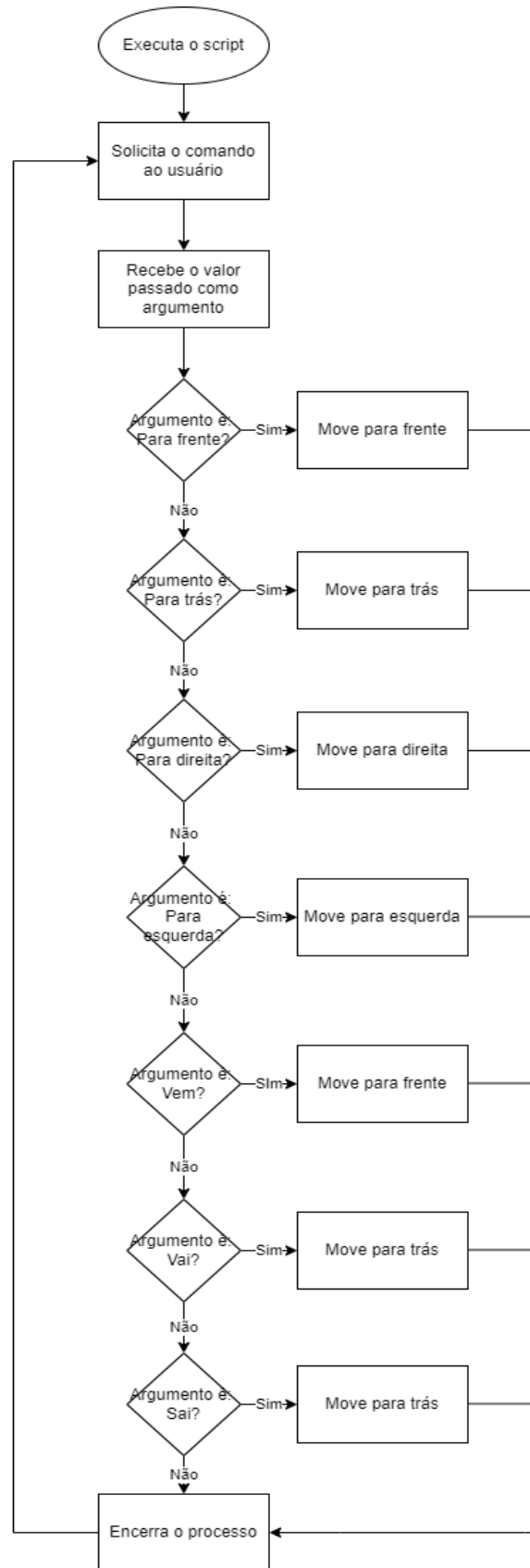


Figura 24: Terceira etapa de movimentação.

Fonte: Os autores.

4 RESULTADOS E DISCUSSÕES

O fluxograma a seguir mostra o funcionamento do sistema.

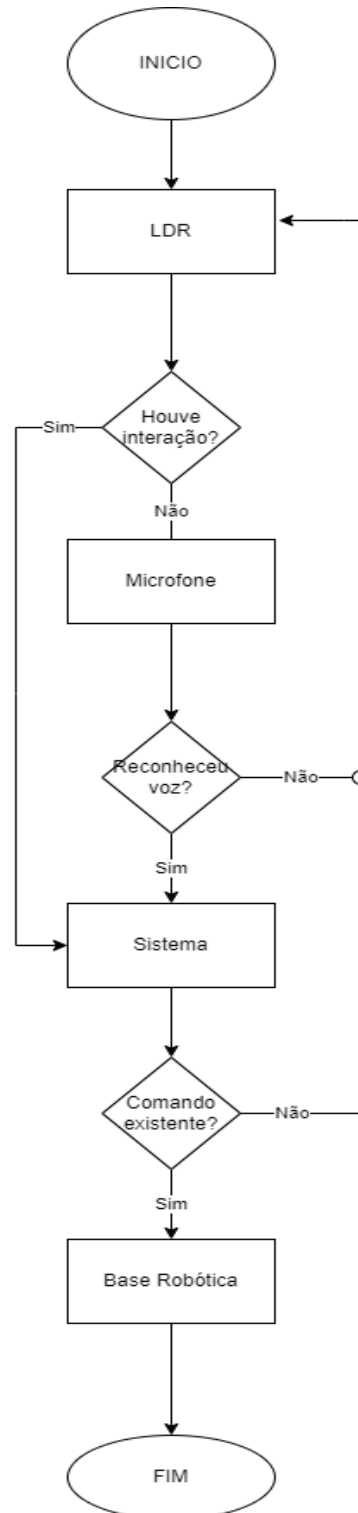


Figura 25: Fluxograma do funcionamento do sistema.

Fonte: Os autores.

4.1. TESTE OLED

Para o teste do OLED, ele foi conectado na *Raspberry* nos pinos de comunicação do tipo I2C, GPIO3 (SCL) e GPIO5 (SDA). Além de saber as portas de comunicação I2C da *Raspberry* é de suma importância saber o endereço do dispositivo, visto que cada dispositivo possui um endereço para transmissão de dados, de modo a viabilizar a comunicação com vários dispositivos pelo mesmo barramento. O endereço geralmente é encontrado na folha de dados do dispositivo. A figura 22 mostra a resposta da saída da varredura do dispositivo I2C.



```

Arquivo Editar Abas Ajuda
pi@raspberrypi:~$ i2cdetect -y 1
 00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
 10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
 20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
 30:  --  --  --  --  --  --  --  --  3c  --  --  --  --  --  --  --
 40:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
 50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
 60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
 70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --

```

Figura 26: Varredura do I2C.

Fonte: Os autores.

Na implementação, após requerer o endereço do dispositivo e a porta de comunicação, logo tendo esses dados foi realizado a configuração das telas necessárias para o projeto. As figuras abaixo mostram as imagens em dois displays, neste teste foi aplicado o tamanho da imagem e cores:

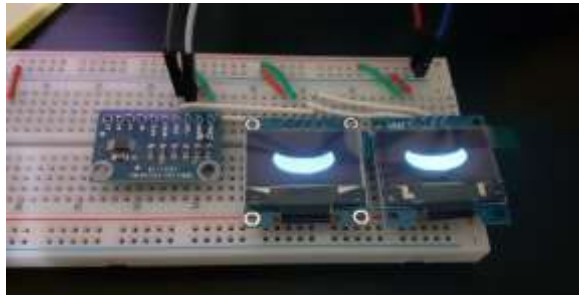


Figura 27: Imagem display 1.

Fonte: Os autores.

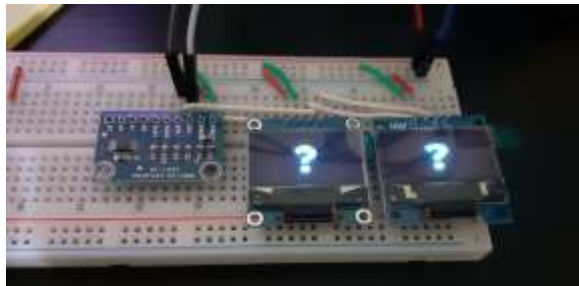


Figura 28: Imagem display 2.

Fonte: Os autores.

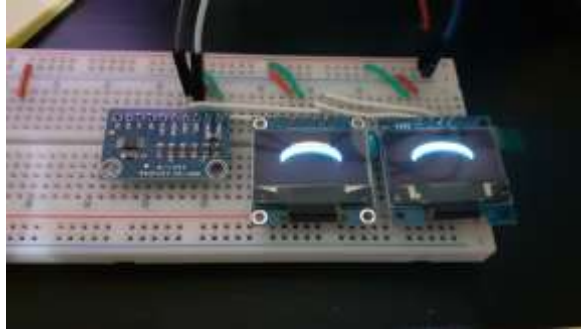


Figura 29: Imagem display 3.

Fonte: Os autores.

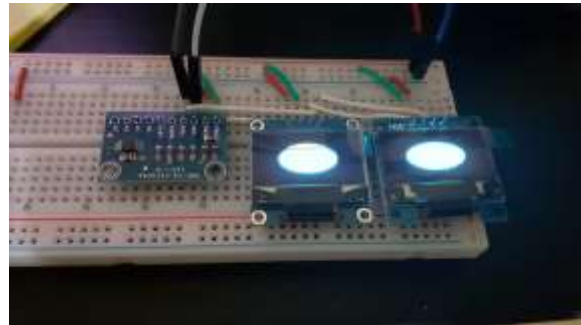


Figura 30: Imagem display 4.

Fonte: Os autores.

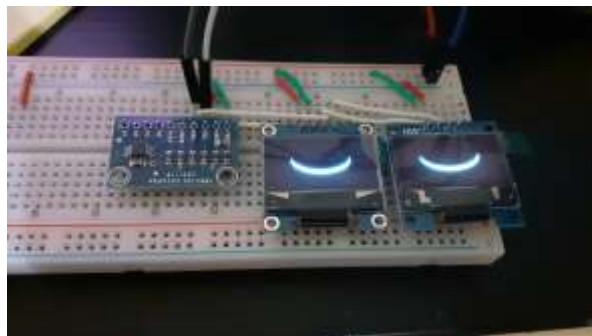


Figura 31: Imagem display 5.

Fonte: Os autores.

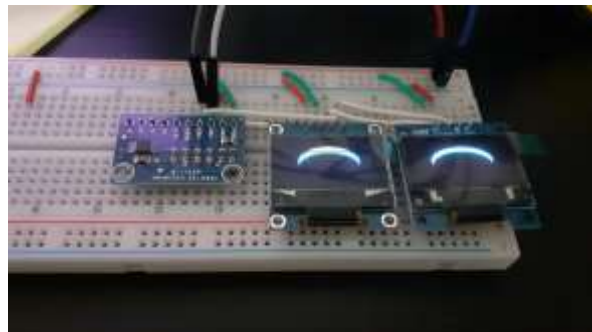


Figura 32: Imagem display 6.

Fonte: Os autores.

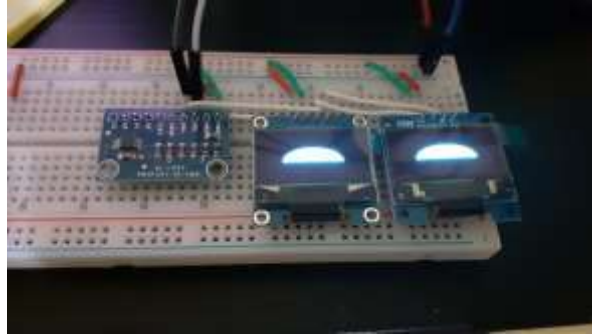


Figura 33: Imagem display 7.

Fonte: Os autores.

O circuito final do display está na Figura 34.



Figura 34: Imagem circuito com display.

Fonte: Os autores.

4.2. TESTE DE AUTONOMIA DA BATERIA

Em questão de autonomia é necessário considerar o tempo de uso do robô e evidenciar que a bateria utilizada é uma powerbank, que tem a capacidade de 4400mA/h, os consumos realizados pelos componentes da base:

Tabela 12: Autonomia do protótipo.

ITEM	mA/h
<i>Raspberry</i>	330

Fonte: Os Autores.

Realizando a divisão da capacidade da bateria, 4400mAH, pelo consumo total da *Raspberry*. Temos um total de treze horas e trinta e três minutos de autonomia.

4.3. TESTE DO MICROFONE

Foram realizados alguns testes com o microfone, com a intenção de validar o reconhecimento de voz. As palavras utilizadas foram: frente, para frente, traz, para trás, direita, para direita, esquerda, para esquerda, vem, vai, sai. Um total de cinco pessoas de idade e gênero diferentes foram convocadas, a fim de coletar dados representativos para o teste, sendo dividido em ambiente com pessoas e ambiente isolado. As tabelas a seguir mostram os resultados.

Tabela 13: Teste 1 – Ambiente isolado.

Ambiente isolado					
Teste 1		Palavras Testadas			
Gênero	Idade	Frente	Trás	Direita	Esquerda
Homem	50 anos	7	10	9	10
Mulher	40 anos	8	10	9	9
Homem	23 anos	8	9	10	10
Mulher	24 anos	8	9	10	10
Mulher	10 anos	7	9	9	10

Fonte: Os Autores.

Tabela 14: Teste 2 – Ambiente isolado.

Ambiente isolado					
Teste 2		Palavras Testadas			
Gênero	Idade	Para Frente	Para Trás	Para Direita	Para Esquerda
Homem	50 anos	6	7	8	9
Mulher	40 anos	7	8	8	8
Homem	23 anos	7	7	9	9
Mulher	24 anos	8	7	8	9
Mulher	10 anos	6	8	8	8

Fonte: Os Autores.

Tabela 15: Teste 3 – Ambiente isolado.

Ambiente isolado					
Teste 3		Palavras Testadas			
Gênero	Idade		Vem	Vai	Sai
Homem	50 anos		10	10	9
Mulher	40 anos		10	10	10
Homem	23 anos		10	10	10
Mulher	24 anos		10	10	10
Mulher	10 anos		9	10	10

Fonte: Os Autores.

Tabela 16: Teste 4 – Ambiente com pessoas.

Ambiente com pessoas					
Teste 4		Palavras Testadas			
Gênero	Idade	Frente	Trás	Direita	Esquerda
Homem	50 anos	5	5	6	5
Mulher	40 anos	6	5	6	6
Homem	23 anos	6	4	4	5
Mulher	24 anos	7	5	5	6
Mulher	10 anos	5	3	7	7

Fonte: Os Autores.

Tabela 17: Teste 5 – Ambiente com pessoas.

Ambiente com pessoas					
Teste 5		Palavras Testadas			
Gênero	Idade	Para Frente	Para Trás	Para Direita	Para Esquerda
Homem	50 anos	3	5	7	5
Mulher	40 anos	3	4	4	5
Homem	23 anos	6	4	4	4
Mulher	24 anos	3	5	4	4
Mulher	10 anos	3	3	3	4

Fonte: Os Autores.

Tabela 18: Teste 6 – Ambiente com pessoas.

Ambiente com pessoas.					
Teste 6		Palavras Testadas			
Gênero	Idade		Vem	Vai	Sai
Homem	50 anos		5	4	6
Mulher	40 anos		5	5	5
Homem	23 anos		6	4	4
Mulher	24 anos		5	4	4
Mulher	10 anos		6	4	4

Fonte: Os Autores.

Os dados mostrados nas tabelas representam dez amostras de cada palavra que foram testadas, sendo os números referentes à quantidade de vezes que o reconhecedor de voz interpretou corretamente. Por exemplo: na tabela 15, a palavra “vai” foi reconhecida todas as dez vezes pelos cinco participantes.

Já nos testes em ambientes com pessoas no caso um ambiente residencial, com aproximadamente 5 pessoas, o reconhecedor teve um pouco mais dificuldade, entendendo em menos vezes. Por exemplo: na tabela 17, a palavra “para frente” foi entendida apenas 3 vezes com quase todos os participantes.

4.4.PROTÓTIPO FINALIZADO

O protótipo pronto, com todos os componentes e suas funcionalidades integrados está exposto a seguir nas Figuras 35 e 36.



Figura 35 - Protótipo finalizado

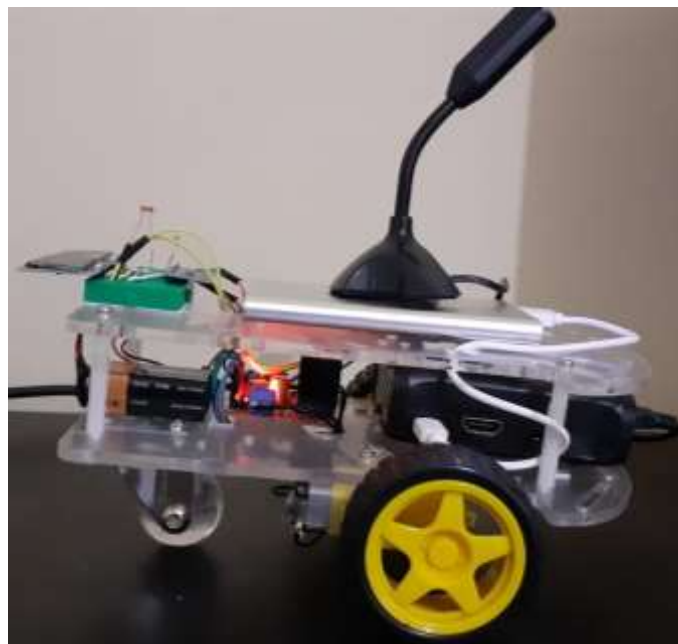


Figura 36 - Protótipo finalizado

O resultado de todas as ações do protótipo está apresentado nas figuras abaixo. Cada rosto tem sua respectiva reação através do movimento.

Na figura 37, temos no display apresentado o rosto inicial do robô quando o mesmo está aguardando o comando de voz ou ação com o LDR.



Figura 37 - Rosto inicial sem movimento



Figura 38 - Rosto triste mais movimento



Figura 39 – Rosto vergonhoso mais movimento



Figura 40 – Rosto duvidoso mais movimento



Figura 41 - Rosto bravo mais movimento



Figura 42 - Rosto feliz mais movimento



Figura 43 - Rosto manhoso mais movimento

5 CONCLUSÃO

Neste trabalho foi apresentado o desenvolvimento de um robô que assemelha o comportamento de um animal de estimação. Ele é constituído de uma integração da *Raspberry* que faz o controle de todos os componentes, fixado em uma base robótica em que nela está acoplado um sensor LDR para ativar as funcionalidades como se fosse um “toque”, um microfone para auxiliar na movimentação do protótipo e dois displays OLED para representar a expressões do robô.

Os resultados obtidos foram satisfatórios, uma vez que foi possível realizar a movimentação do robô com base nos dados recebidos do microfone e do LDR e mostrar as reações através da interface gráfica nos displays OLED.

O projeto nos ajudou em nosso desenvolvimento pessoal, profissional e acadêmico através da aplicação dos conteúdos ministrados nas aulas pelos professores e por todo o conhecimento que foi adquirido com os estudos realizados durante todo o período de graduação.

5.1 PRINCIPAIS DIFICULDADES

As maiores dificuldades enfrentadas estão relacionadas a parte de *software*, mais precisamente na integração entre o movimento da base e exibição da interface gráfica em displays.

Pode-se evidenciar algumas dificuldades ou empecilhos que mais tomaram tempo no momento da procura de solução, como:

- Movimentar o protótipo ao reconhecer o comando pela fala
 - O fato de o comando vir de um microfone e de o reconhecedor de comando de voz depender de conexão com a internet dificulta a interpretação do software e posteriormente o acionamento dos motores.
- Representar imagem simulando rosto do pet
 - Devido à complexidade da biblioteca *opencv*, realizar o tratamento de imagens em uma *Raspberry Pi* dificultou pelo motivo do poder de processamento do microcomputador que não possui um *hardware* tão potente.

5.2 MELHORIAS FUTURAS

Um próximo passo seria o uso de um *hardware* que possua suas especificações técnicas mais robustas do que o *Raspberry* utilizado, levando em consideração algumas lentidões e o pequeno *delay* da comunicação. Outros pontos seriam:

- Adicionar mais faces ao robô;
- Apresentar os rostos em um display de maior resolução;
- Adicionar falas e sons ao robô;
- Adicionar mais sensores para diferentes reações ao robô;
- Deixar o robô com uma aparência mais próxima a um animal de estimação.
- Adicionar aprendizado de máquina e conceitos de *machine learning* ao robô.

REFERÊNCIAS

BORGES, L.E. **Python para desenvolvedores**. Rio de Janeiro – Brasil: Editora Novatec, 2014.

BRYNJOLFSSON, E; MCAFEE, A. *The Second Machine Age: Work, Progress, and Prosperity in a Time of Brilliant Technologies*. Primeira edição. Editora W. W. Norton & Company, 2014.

DUBRIN, A. J. **Fundamentos do Comportamento Organizacional**. 2. ed. São Paulo: Thomson Learning, 2002.

FOUNDATION, RASPBERRY. **Raspberry Pi 3 Model B+**. 2019. Disponível em<<https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/>>. Acesso em 05/10/2019.

MARQUES, D. L.; COSTA, L. F. S. **Atraindo alunos do ensino médio para a computação: Uma experiência prática de introdução a programação utilizando jogos e python**. Departamento de Ciências Exatas-Universidade Federal da Paraíba-UFPB, 2011.

PERKOVIC, L.; **Introdução á computação usando Python: Um foco no desenvolvimento de aplicações**. Rio de Janeiro - Brasil: LTC, 2016.

PORTUGAL, C. R. P.; **Raspberry Pi**. 2019. Disponível em<<http://www.raspberrypiportugal.pt/raspberry-pi/>>. Acesso em 05/10/2019.

RABINER, L. R.; JUANG, B. H. *Fundamentals of speech recognition*. Englewood Cliffs, New Jersey, 1993.

RABINER, L. R.; SCHAFER, R. W. *Digital Processing of speech signals*. Englewood Cliffs, New Jersey, 1978.

RASPBERRY, P. **Raspberry pi.Raspberry Pi**, v. 1, p. 1, 2019.

RICHARDSON, M.; WALLACE, S. **Primeiros Passos com o Raspberry Pi**. Primeira edição. [S.l.]: Editora Novatec, 2013.

SEED, S.; **Kit ReSpeaker 6-Mic Circular Array para Raspberry Pi**. China, 2008-2019. Disponível em: <<http://wiki.seeedstudio.com>>. Acesso em: 21 de setembro 2019.

SRIRENGAN, K. Understanding UNIX. Nova Delhi: PHI Learning, 2006.

WESTER, INFO. **Linux** 2019. Disponível em<
https://www.infowester.com/historia_linux.php/>. Acesso em 05/10/2019.

GZH. **Pets e coronavírus: os benefícios de uma companhia animal em tempos de distanciamento social** 2020. Disponível em< <https://gauchazh.clicrbs.com.br/fique-bem/noticia/2020/05/pets-e-coronavirus-os-beneficios-de-uma-companhia-animal-em-tempos-de-distanciamento-social-ck9v19say008e015ntvxz6e2q.html>>. Acesso em 24 de março 2021.

BRAGA, INSTITUTO NEWTON C. **OpenCV - o que é, onde usar e como instalar na Raspberry Pi (MIC412)** 2020. Disponível em<
<https://www.newtonbraga.com.br/index.php/microcontroladores/143-tecnologia/17799-opencv-o-que-e-onde-usar-e-como-instalar-na-raspberry-pi-mic412.html>>. Acesso em 13 de maio 2021.

CEDROTECH. **OpenCV: Uma breve introdução à visão computacional com python** 2018. Disponível em< <https://blog.cedrotech.com/opencv-uma-breve-introducao-visao-computacional-com-python/>>. Acesso em 13 de maio 2021.

CASA DA ROBOTICA. **Motor DC 3 a 6V com redução**. 2021. Disponível em<
<https://www.casadarobotica.com/robotica/atuadores/motores-dc/motor-dc-3-a-6v-com-reducao/>>. Acesso em 24 de maio 2021.

STMICROELECTRONICS. *Datasheet* **L298N**. Disponível em:
<https://www.sparkfun.com/datasheets/Robotics/L298_H_Bridge.pdf> Acesso em: 31 de maio de 2021.

THE LANCET. *COVID-19 Resource Centre*. Disponível em: <<https://www.thelancet.com/coronavirus>> Acesso em: 28 de julho de 2021.

EXAME. *Robôs de companhia confortam japoneses durante a pandemia*. Disponível em: <<https://exame.com/casual/robos-de-companhia-confortam-japoneses-durante-a-pandemia/>> Acesso em: 28 de julho de 2021.

TECHTUDO. *Linux: Tudo o que você precisa saber antes de começar a usar*. Disponível em: <<https://www.techtudo.com.br/noticias/noticia/2015/03/linux-tudo-o-que-voce-precisa-saber-antes-de-comecar-usar.html>> Acesso em: 28 de julho de 2021.

DO BIT AO BYTE. *Vários modos de interagir com o GPIO do Raspberry Pi*. Disponível em: <<https://www.dobitaobyte.com.br/varios-modos-de-interagir-com-o-gpio-do-raspberry-pi/>> Acesso em: 28 de julho de 2021.

MUNDO PROJETADO. *LDR – O que é e como funciona*. Disponível em: <<https://mundoprojetado.com.br/ldr-o-que-e-e-como-funciona/>> Acesso em: 03 de agosto de 2021.

MANUAL DA ELETRÔNICA. *LDR – O que é e como funciona!* Disponível em: <<https://www.manualdaeletronica.com.br/ldr-o-que-e-como-funciona/>> Acesso em: 03 de agosto de 2021.

ATHOS ELETRONICS. *LDR – O que é e como efetuar leituras*. Disponível em: <<https://athoselectronics.com/ldr-o-que-e-como-funciona/>> Acesso em: 03 de agosto de 2021.

OLIVEIRA, V., & FALCÃO, R. *OLED - O que é?* E. d. Ciência, Editora: Correio do Minho, p. 30, 2015.