

CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS
Campus DIVINÓPOLIS
GRADUAÇÃO EM ENGENHARIA MECATRÔNICA

Felipe Figueiredo Varandas

DESENVOLVIMENTO DE UM PROTÓTIPO DE ALIMENTADOR
AUTOMÁTICO PARA ANIMAIS DOMÉSTICOS - *AUTOMATIC FEEDER*

Divinópolis
2019

Felipe Figueiredo Varandas

DESENVOLVIMENTO DE UM PROTÓTIPO DE ALIMENTADOR
AUTOMÁTICO PARA ANIMAIS DOMÉSTICOS - *AUTOMATIC FEEDER*

Monografia de Trabalho de Conclusão de Curso apresentada ao Colegiado de Graduação em Engenharia Mecatrônica como parte dos requisitos exigidos para a obtenção do título de Engenheiro Mecatrônico.

Áreas de integração: Eletrônica, Mecânica e Computação.

Orientador: Prof. M.Se Juliano de Barros Veloso e Lima

Co-orientador: Prof. M.Se Daniel Alves

Divinópolis
2019

Felipe Figueiredo Varandas

DESENVOLVIMENTO DE UM PROTÓTIPO DE ALIMENTADOR
AUTOMÁTICO PARA ANIMAIS DOMÉSTICOS - *AUTOMATIC FEEDER*

Monografia de Trabalho de Conclusão de Curso
apresentada ao Colegiado de Graduação em En-
genharia Mecatrônica como parte dos requisitos
exigidos para a obtenção do título de Engenheiro
Mecatrônico.

Áreas de integração: Eletrônica, Mecânica e
Computação.

Comissão Avaliadora:

Prof. M. Sc. Juliano de Barros V. e Lima
CEFET/MG *Campus V*

Prof. M. Sc. Alan Mendes Marotta
CEFET/MG *Campus V*

Prof. M. Sc. Márcio Alves de Aguiar
CEFET/MG *Campus V*

Divinópolis 2019

Dedico este trabalho à minha família,
pelos momentos de ausência.

Agradecimentos

Agradeço a Deus por ter me dado saúde e força para superar as dificuldades, por minha vida, família e amigos.

A esta universidade, seu corpo docente, direção e administração que oportunizaram a janela que hoje vislumbro um horizonte superior, eivado pela acendrada confiança no mérito e ética aqui presentes.

Agradeço a todos os professores por me proporcionar o conhecimento não apenas racional, mas a manifestação do caráter e afetividade da educação no processo de formação profissional, por tanto que se dedicaram a mim, não somente por terem me ensinado, mas por terem me feito aprender. A palavra mestre, nunca fará justiça aos professores dedicados aos quais sem nominar terão os meus eternos agradecimentos.

Ao meu orientador, pelo empenho dedicado à elaboração deste trabalho.

Aos meus pais, pelo amor, incentivo e apoio incondicional.

Agradeço a minha mãe Ana Maria, heroína que me deu apoio, incentivo nas horas difíceis, de desânimo e cansaço.

Ao meu pai Renan que apesar sempre me fortaleceu e que para mim foi muito importante.

Meus agradecimentos aos amigos, companheiros de trabalhos e irmãos na amizade que fizeram parte da minha formação e que vão continuar presentes em minha vida com certeza.

Agradeço finalmente ao meu filho Heitor, que foi a melhor coisa que já aconteceu em minha vida.

Enfim, a todos os que por algum motivo contribuíram para a realização desta pesquisa.

*“Não vos amoldeis às estruturas deste mundo,
mas transformai-vos pela renovação da mente,
mas transformai-vos pela renovação da mente,
a fim de distinguir qual é a vontade de Deus:
o que é bom, o que Lhe é agradável, o que é perfeito.”
(Bíblia Sagrada, Romanos 12, 2)*

Resumo

O projeto proposto visa o desenvolvimento de um protótipo de alimentador automático para animais domésticos, englobando três das grandes áreas da engenharia: mecânica, eletrônica e computação. A intenção é promover a comodidade na hora de alimentar os Pets, através do controle de vazão da ração, quantidade de vezes ao dia e monitoramento do consumo. Segundo dados do IBGE, os bichos de estimação estão presentes no cotidiano de quase metade da população brasileira e apesar do aumento desse número, muito pouco é investido em tecnologia e praticidade para os donos. Por isso, automatizar essa rotina diária é algo estratégico para aumentar a tranquilidade em se ter um animal de estimação. Muitas pessoas desejam um companheiro para se distrair dos problemas do cotidiano, entretanto, têm medo de não ter tempo ou mesmo se esquecer de alimentá-lo. Usualmente, a população utiliza-se do método manual para alimentar seus animais, sem nenhum tipo de monitoramento ou controle, portanto, sem conhecimento da quantidade de ração consumida pelo pet. Atualmente existem poucos modelos que proporcionam esse tipo de tecnologia e as que existem apresentam um valor elevado. Visto isso, o *Automatic Feeder* será projetado para atender essas demandas e outras possíveis, de acordo com análises a serem realizadas. O sistema será composto por sensores, atuadores, microcontroladores e circuitos eletrônicos que serão projetados para realizar o controle de vazão da ração. Outros componentes utilizados serão os sensores de nível e de vazão que monitora o consumo e quantidade restante de alimento que se encontra no reservatório, e em conjunto com o microcontrolador, fará a aquisição e tratamento de dados, gerando informações que serão traduzidas em valores de acordo com a escolha do usuário. Além disso, o protótipo apresentará um sistema de comunicação sem fio, que estará montado junto aos demais componentes, recebendo e enviando dados, tanto para coleta, quanto para a ação a ser realizada pelo sistema. Na sequência um aplicativo para *smartphones* será desenvolvido para promover a interação do sistema com o usuário, que receberá os comandos como quantidade de ração e quantas vezes ao dia. Além disso, terá uma tela acoplada ao equipamento apresentando os comandos para se comunicar e setar os parâmetros desejados. Esse *App* se comunicará, recebendo e enviando os dados provenientes do sistema embarcado do alimentador, via comunicação *wireless* com o microprocessador e fornecerá as informações sobre quantidade de ração restante e monitoramento do animal se alimentando. Por fim, será possível receber dicas de horários e quantidade de ração para cada tipo de raça. Espera-se com o desenvolvimento desse projeto, contribuir para a comodidade, conforto e praticidade do usuário, e tornar mais viável, economicamente, esse tipo de tecnologia, que envolve novas tendências de projetos embarcados.

Palavras-chave: Alimentador automático, automação, *IoT*, *App*, comodidade.

Resumo

The proposed project aims at the development of a prototype of automatic feeder for domestic animals, encompassing three of the major areas of engineering: mechanics, electronics and computing. The intention is to promote convenience when it comes to feeding the pets, by controlling feed flow, number of times a day and monitoring consumption. The IBGE data, the animals of today are present in the daily life of a Brazilian species and in the same number of the same, very little is invested in technology and practicality for the owners. Therefore, automating this daily routine is something strategic to increase the tranquility in having a pet. Many people want a companion to be distracted from the problems of daily life, however, they are afraid of not having time or even forget to feed it. Usually, the population uses the manual method to feed their animals, without any type of monitoring or control, therefore, without knowing the amount of feed consumed by the pet. Currently there are few models that provide this type of technology and those that exist have a high value. Seen this, the textit Automatic Feeder will be designed to meet these demands and others possible, according to analyzes to be performed. The system will be composed of sensors, actuators, microcontrollers and electronic circuits that will be designed to perform the flow control of the feed. Other components used will be the level and flow sensors that monitors the consumption and remaining quantity of food that is in the reservoir, and together with the microcontroller, will make the acquisition and processing of data, generating information that will be translated into values according to with the user's choice. In addition, the prototype will present a wireless communication system, which will be assembled along with the other components, receiving and sending data, both for collection and for the action to be performed by the system. Following an application for textit smartphones will be developed to promote the interaction of the system with the user, who will receive the commands as amount of ration and how many times a day. In addition, it will have a screen attached to the equipment presenting the commands to communicate and set the desired parameters. This App will communicate, receive, and send data from the feeder's embedded system via wireless communication with the microprocessor and provide information on the amount of feed remaining and monitoring of the feeding animal. Finally, it will be possible to receive tips on times and amount of ration for each type of breed. It is expected to develop this project, contribute to the comfort, comfort and practicality of the user, and make this technology more feasible, economically, which involves new trends of embedded projects.

Keywords: automatic, automation, textit IoT, textit App, convenience.

Sumário

Lista de Figuras	xiii
Lista de Tabelas	xiv
Lista de Acrônimos e Notação	xv
1 Introdução	1
1.1 Definição do problema e contextualização	1
1.2 Motivação	3
1.3 Objetivos	3
1.3.1 Objetivo Geral	3
1.3.2 Objetivos Específicos	3
2 Fundamentos	5
2.1 Revisão de Literatura	5
2.2 Metodologia	6
2.3 Fundamentação Teórica	6
2.3.1 Projeto e construção da estrutura	6
2.3.2 Controle e monitoramento de vazão da ração	7
2.3.3 Microprocessador para controle de vazão da ração	7
2.3.4 Sistema de alimentação	7
2.3.5 Circuito de comunicação	8
2.3.6 Aplicativo para <i>smartphone</i>	8
2.3.7 Resultados esperado com a construção do protótipo	9
3 Projeto Mecânico	10
3.1 Definição de como é realizada a vazão de ração	10
3.2 Definição da dimensão do recipiente	10
3.3 Projeto do controlador de vazão da ração	13
3.3.1 Hélice	13
3.3.2 Motor de Passo	16
3.4 Condutores de ração	16
3.4.1 Rampa	16
3.5 Estrutura	17
3.5.1 Estrutura externa	17
3.5.2 Estrutura interna	18

3.6	Definição do Atuador	22
3.6.1	Motor de Passo	22
3.6.2	Driver do Motor de Passo	22
3.7	Definição dos Sensores	23
3.7.1	Sensor infravermelho	23
3.7.2	Sensor de umidade e temperatura	24
3.7.3	Sensor de distância	24
3.7.4	Microprocessador	26
3.8	Definição da Fonte	27
3.8.1	Fonte Ajustável 3.3V 5V para Protoboard MB102	27
3.8.2	Fonte chaveada	28
3.9	Protoboard	29
3.10	Comunicação	31
3.10.1	ESP8266	31
3.10.2	Aplicativo	32
3.11	Construção parte Mecânica	32
3.11.1	Parafuso	33
3.12	Construção parte Elétrica	33
3.13	Programação	33
3.14	Blynk	34
3.14.1	Composição do Blynk	34
3.14.2	Requisitos para utilizar o Blynk	35
3.14.3	Download e instalação do Blynk	36
3.14.4	Configuração do Blynk no dispositivo móvel	36
3.14.5	Instalação do pacote de bibliotecas na IDE do Arduino	46
3.15	Controle do Sistema	47
3.16	Resultados e Discussão	51
3.17	Orçamento	53
4	Considerações Finais	54
4.1	Conclusões	54
5	Desenho Técnico	55
6	Programação Arduino IDE	61
	Referências	65

Lista de Figuras

1.1	População de animais e Faturamento no Brasil	2
1.2	Tecnologia para animais de estimação, Fonte: APT ¹	3
2.1	Fluxograma	6
3.1	Vista frontal do recipiente	11
3.2	Vista frontal do recipiente	12
3.3	Vista corte do recipiente	13
3.4	Vista panorâmica da Hélice	14
3.5	Vista corte da hélice	14
3.6	Vista frontal da hélice	15
3.7	Vista superior da hélice	15
3.8	Vista panorâmica do motor de passo	16
3.9	Vista panorâmica da rampa	17
3.10	Desenho esquemático da estrutura	18
3.11	Simulação-aplicação de tensão na estrutura	19
3.12	Simulação-deformação estrutura	20
3.13	Simulação-deslocamento estrutura	21
3.14	Motor de passo e driver	23
3.15	Sensor infravermelho	23
3.16	Sensor Umidade e Temperatura - DHT11	24
3.17	Sensor Ultrasonico HC-SR04	25
3.18	Arduino UNO	26
3.19	Fonte Ajustável 3.3V 5V para Protoboard MB102	28
3.20	Fonte chaveada	29
3.21	Protoboard	30
3.22	Protoboard	30
3.23	Módulo WiFi ESP8266 ESP-01	31
3.24	Parafuso de Aço para Madeira 3,5x30mm Auto Atarraxante Chata Philips	33
3.25	Arquitetura de funcionamento Blynk	35
3.26	Abertura app blynk	36
3.27	Aplicativo blynk	37
3.28	Configuração Projeto Aplicativo blynk	38
3.29	Token Aplicativo	38
3.30	E-mail com token	39
3.31	Opções do Aplicativo blynk	39

3.32	Configurações Aplicativo blynk	40
3.33	Projeto Aplicativo blynk	41
3.34	Projeto Aplicativo blynk	41
3.35	Opção criar mais projeto Aplicativo blynk	42
3.36	Configuração de projeto Aplicativo blynk	43
3.37	Opções de Widget Aplicativo blynk	44
3.38	Widget Aplicativo blynk	44
3.39	Limite Widget Aplicativo blynk	45
3.40	Opção compra Pontos Aplicativo blynk	45
3.41	Vericar de conectividade Aplicativo blynk	46
3.42	Site Blynk	47
3.43	Biblioteca Blynk Arduino IDE	47
3.44	Esquemático Projeto	49
3.45	Local para inserir o Token	49
3.46	Local para inserir nome rede e senha	49
3.47	Configuração velocidade para 9600 bauds	50
3.48	Seleção Arduino/Genuino Uno	50
3.49	Interface do Aplicativo Blynk	52
5.1	Desenho técnico Helice	56
5.2	Desenho técnico Motor de Passo	57
5.3	Desenho técnico Galão de Água	58
5.4	Desenho técnico da Estrutura	59
5.5	Desenho técnico do pode de ração	60

Lista de Tabelas

3.1	Dados referentes ao Motor de Passo	22
3.2	Dados referentes ao Driver Motor de Passo	22
3.3	Dados referentes ao Sensor Infravermelho	23
3.4	Dados referentes ao Sensor de umidade e temperatura	24
3.5	Dados referentes ao Sensor Ultrasonico HC-SR04	25
3.6	Dados referentes ao Arduino UNO	27
3.7	Dados referentes a Fonte Ajustável 3.3V 5V para Protoboard MB102	28
3.8	Dados referentes a Fonte chaveada	29
3.9	Dados referentes a Protoboard	31
3.10	Dados referentes a Módulo WiFi ESP8266 ESP-01	32
3.11	Orçamento	53

Lista de Acrônimos e Notação

ABINPET	Associação Brasileira da Indústria de Produtos para Animais de Estimação
SPC	Serviço de Proteção ao Crédito
CNDL	Confederação Nacional de Dirigentes Lojistas
IOT	Internet das Coisas(intenet of things)
PET	Animal de Estimação
TCP	Protocolo de Controle de Transmissão(Transmission Control Protocol)
IP	Protocolo de Internet(Internet Protocol)
MVP	Mínimo Produto Viável

Introdução

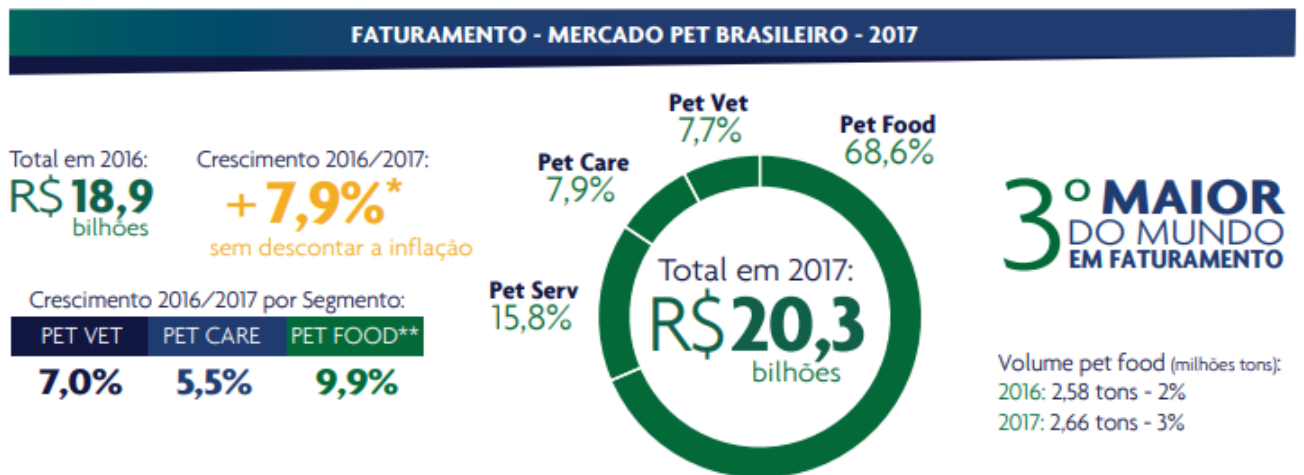
1.1 Definição do problema e contextualização

De acordo com os dados da ABINPET (2013), atualmente o Brasil tem a segunda maior população de cães, gatos e aves canoras e ornamentais em todo o mundo e é o quarto maior país em população total de animais de estimação. São 52.2 milhões de cães, 22.1 milhões de gatos, 18 milhões de peixes, 37,9 milhões de aves e mais 2,2 milhões de outros animais. O total é de 132,4 milhões de *pets*, sendo gastos em média 189,00 por mês, segundo o SPC BRASIL (2017) e CNDL (2017), com cada mascote. Visto isso, podemos afirmar a força potencial desse setor na economia brasileira. Tendo como objetivo o conforto e praticidade na hora de alimentar o animal de estimação, dispositivos mais sofisticados vêm sendo desenvolvidos, como, por exemplo, alimentadores eletrônicos para pequenas quantidades de ração. Contudo, os valores destes equipamentos são elevados e estes possuem pouco espaço de armazenamento para a ração, fazendo com que não suportem grandes estoques de alimento. Com isso, estes equipamentos não atendem animais de grande porte ou situações em que os donos precisem viajar. Com o avanço da tecnologia e a necessidade de interagir o mundo virtual com o físico, surgiu o termo Internet das Coisas, (*IoT*, do inglês *Internet of Things*), automatizando e, portanto, facilitando serviços do dia a dia. Com a grande propagação dos *smartphones*, o desenvolvimento de aplicativos controladores ficam cada vez mais viáveis. Obtendo como motivação o contexto acima, o projeto busca a oportunidade da união, de forma prática, da tecnologia de comunicação e monitoramento com a *IoT*. Propõe então, o desenvolvimento de um sistema que possa ser controlado de qualquer lugar do mundo, utilizando apenas um celular, combinado assim praticidade no auxílio que o usuário procura para ter o o monitoramento e controle da alimentação do animal de estimação.

Sendo assim, o projeto integrará três áreas da Engenharia Mecatrônica, sendo elas eletrônica, mecânica e computação.



Fonte: IBGE - Pesquisa quinzenal. Elaboração: Abinpet. Dados 2013. *Estimativa Abinpet para outros animais de estimação (répteis e pequenos mamíferos)



Fonte/Elaboração: Abinpet.

Pet Vet: Medicamentos Veterinários. Pet Care: Equipamentos, Acessórios, Produtos de Higiene e Beleza Animal. Pet Serv: Serviços.

*Descontando a inflação no ano de 2017 que foi de 2,95% o crescimento real do faturamento foi de 4,95%. Em todos os anos anteriores a 2017, não houve desconto de inflação.

**Representatividade.

Figura 1.1: População de animais e Faturamento no Brasil

1.2 Motivação

Atualmente, vê-se vários projetos e produtos tecnológicos que auxiliam e tornam a vida de donos e dos próprios animais mais confortável. Isso pode ser explicado pelo crescente número de equipamentos que utilizam o conceito "internet das coisas". Dessa forma, buscam com seu crescimento proporcionar aos objetos do dia-a-dia (quaisquer que sejam), mas com capacidade computacional e de comunicação se conectem à internet. A conexão com a rede mundial de computadores viabilizará, primeiro, controlar remotamente os objetos e, segundo, permitir que os próprios objetos sejam acessados como provedores de serviços. Estas novas habilidades, dos objetos comuns, geram um grande número de oportunidades tanto no âmbito acadêmico quanto no industrial. Todavia, estas possibilidades apresentam riscos e acarretam amplos desafios técnicos e sociais. Têm-se na Figura 1.2 , ilustração sobre a modernização em tal meio, um coleira que é um monitor de atividade vestível para cães.

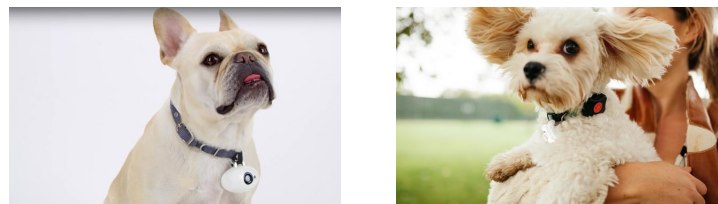


Figura 1.2: Tecnologia para animais de estimação, Fonte: APT¹

1.3 Objetivos

Na presente seção é descrito de forma concisa o objetivo geral do trabalho, o qual almeja-se alcançar, sendo apresentados também os objetivos específicos.

1.3.1 Objetivo Geral

Projetar um protótipo de alimentação automático para animais domésticos, capaz de estabelecer o controle de dosagem e vezes ao dia, monitorar o nível da ração, umidade e temperatura no silo e também, ser controlado remotamente, utilizando comunicação *wireless*, por meio de um aplicativo para *smartphones*.

1.3.2 Objetivos Específicos

- Construir a estrutura;
- Construir o controlador de vazão de ração;
- Construir o circuito eletrônico para regular a vazão;

- Construir o circuito para monitoramento;
- Construir o circuito de comunicação *wireless*;
- Desenvolver a comunicação entre o sistema e o app;

Fundamentos

No presente capítulo, situam-se revisão de literatura, metodologia e fundamentação teórica do trabalho.

2.1 Revisão de Literatura

A prática de domesticação de animais passou a fazer parte do dia a dia dos seres humanos e a convivência se transformou em amizade. Desde os tempos mais remotos, a domesticação está presente na vida dos seres humanos. Mesmo com o passar do tempo, com as várias invenções do homem, mesmo com a agitada vida dos dias atuais, ainda continua-se domesticando animais. Mas também, eles nos ajudaram na evolução, tanto na parte da mão de obra, quanto na parte de dar companhia para seu dono.

O cachorro foi o primeiro animal a ser domesticado e o que mais se adaptou com os humanos. Segundo ROSSI (2016), os cães que conhecemos hoje em dia, foram espécies de lobos no passado e que foram atraídos para as aldeias pela comida em abundância. O gato foi domesticado há mais de 5000 anos a.c, no antigo Egito. Por serem capazes de controlar pragas com ratos e outros animais menores dentro das casas, os gatos eram tratados como membro da família.

Na idade moderna, os animais eram utilizados também como cães de guarda, como meio para levar carroças e trenós e acompanhar tropeiros e agricultores (Berzins, 2000). Na idade do bronze e do ferro, os cavalos eram muito utilizados como meio de transporte, por serem considerados mais rápidos que outros meios (Levine, 1999). Essa interação com os animais apareceu também na mitologia, com deuses que tinham a composição de animais misturados com humanos, representando valores, proteção e esperança (Dotti, 2005). Atualmente, é possível verificar que os animais estão presentes e possuem papel muitas vezes ativo em desenhos animados, filmes, livros e propagandas (Garcia, 2009).

Idealizado no início dos anos 1980, os telefones capazes de executar várias tarefas foram criados pela companhia alemã Frog Desing em 1993 VOLTOLINI (2014), foi *smartphone*

considerado um dos primeiros dispositivos que congregam funções alternativas além de execução de chamadas.

Em 1991 começou a discussão sobre conexão de objetos, quando a conexão de TCP/IP e a Internet que conhecemos começou a se tornar acessível. E foi em 1999 que Kevin Ashton, do Massachusetts Institute of Technology, MIT, propôs o termo *Internet das Coisas* após dez anos de estudo e projetos, escreveu o artigo *A Coisa da Internet das Coisas* para o RFID Journal, e a partir daí o termo se popularizou.

Segundo ASHTON *et al.* (2009), a falta de tempo na rotina das pessoas fará com que necessitem se conectar à internet de várias maneiras. Com a mobilidade e tecnologia avançando, será possível acumular dados e até o movimento dos corpos com precisão.

Então, com o passar dos anos, com o aumento crescente da tecnologia, várias portas foram abertas. O que possibilita o desenvolvimento do projeto.POLI (2017)

2.2 Metodologia

Uma vez que o desenvolvimento do projeto integra diferentes áreas da Engenharia Mecatrônica, essa seção será subdividida a fim de simplificar e organizar as ideias do trabalho.

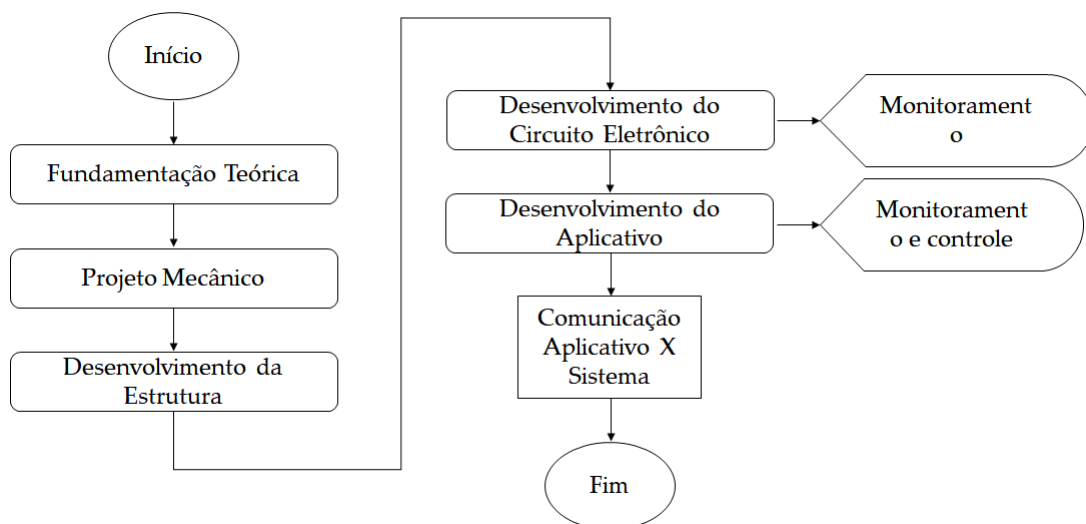


Figura 2.1: Fluxograma

2.3 Fundamentação Teórica

2.3.1 Projeto e construção da estrutura

Para construir uma estrutura que seja resistente e compacta, onde se consiga comportar todos os componentes de *hardware* que serão instalados. A estrutura do sistema

terá que ser projetada para suportar tantos esforços internos como externos, por isso, primeiramente será realizado uma análise de esforços no programa *SolidWorks*. Por se tratar de um alimentador de animais com diferentes tamanhos e pesos, a estrutura deve ser resistente o bastante para resistir a possíveis impactos externos, e justamente por tratar de diferentes animais, a quantidade máxima do reservatório de ração, (que no caso foi definida no projeto como no máximo 15kg), sofrerá um aumento no peso do sistema. A partir dos dados levantados anteriormente, com uma avaliação de força e momento feita no *software* Solidworks será possível fazer a escolha do material, dimensionamento e projetar a estrutura, seguindo os dados da resistência. mecânica. Foi utilizado o livro MECÂNICA DOS MATERIAIS, 2011 com instrumento de aprendizado para definir a estrutura do sistema.

2.3.2 Controle e monitoramento de vazão da ração

A quantidade de ração a ser despejada será escolhida pelo usuário, desse modo, o sistema para regulagem da vazão será em malha aberta, tendo em vista que o peso feito pela ração não terá influência na quantidade que a ser escolhida. Desse modo, os componentes necessários para essa montagem são um motor de passo com *driver*, que é usado para girar o sistema de hélice usada como mecanismo de vazamento da ração para os animais domésticos. Um sensor será utilizado para ver a quantidade restante de ração ainda no reservatório. OLIVEIRA, 2017

2.3.3 Microprocessador para controle de vazão da ração

Por não se tratar de uma estrutura totalmente mecânica, terá o microprocessador que controlará o motor de passo através do *driver*, este motor está acoplado as hélices, de modo que, a partir das escolhas do usuário, como por exemplo, a quantidade de ração, quantidade de vezes ao dia e o horário em que acontecerá o despejo do alimento. Desse modo, o circuito precisará operar de maneira precisa, contando com a ajuda de módulo de precisão, que funcione como um relógio. Dados do livro MONK, 2017 RODRIGO MAXIMIANO A. ALMEIDA CARLOS HENRIQUE V. MORAIS, 2016

2.3.4 Sistema de alimentação

Pelo fato dos dispositivos a serem usados possuírem diversos valores de tensão, o projeto deverá contar com um desenvolvimento de um sistema de alimentação. Por isso, o sistema possuirá um circuito conversor do tipo *Flyback* e reguladores, para executar os devidos ajustes de tensão. Deverá ser utilizado um conversor *Flyback* comercial, pois a construção do mesmo não é um dos objetivos do projeto.

2.3.5 Circuito de comunicação

O circuito de comunicação fará a troca de informações entre o dispositivo e o aplicativo para *smartphones*. Deseja-se utilizar uma comunicação do tipo *wireless*, pois como não é viável a utilização de cabos, por ser o principal almejo do projeto poder controlar o alimentador a distância, este tipo de sistema será de melhor aplicação. Dessa forma, será utilizado um módulo *wifi* operando com o microcontrolador, e em seguida, serão implementados os algoritmos para estabelecer a comunicação e fazer trocas de informações entre os dois sistemas.

2.3.6 Aplicativo para *smartphone*

O mercado de aplicativos cresceu muito nos últimos anos, segundo um estudo realizado pela Appnation, POLI (2017). Segundo levantamento, os *apps* movimentou cerca de 151 bilhões de dólares neste último ano, ou seja, é um mercado muito concorrido. Com o objetivo da praticidade em se alimentar o animal, e com o intuito de se obter uma interação entre o dispositivo e o usuário, será desenvolvido um aplicativo para *smartphone*, contendo todas as configurações necessárias para operação do alimentador. Além disso, será implementado uma opção com aviso para monitoramento do nível de ração e se o animal está comendo. A partir do *software* desenvolvido para a plataforma *Android*, será elaborado um app com layout simples, que cumprirá o objetivo.

E com popularização de *smartphones*, *tablets* e plataformas microcontroladas / embarcadas, tornou-se possível desenvolver os mais diversos tipos de projetos integrando várias tecnologias com a finalidade de controlar algo ou obter informações. O boom do Arduino fez com que até mesmo quem nunca foi muito fã de tecnologia, passasse a buscar conhecimento para desenvolver os mais diversos tipos de projetos. Dentre as possibilidades de projetos que podem ser desenvolvidos com Arduino e outras plataformas, o que mais se destaca é a automação residencial. A busca por comodidade fez com que o termo automação residencial ficasse muito comum no nosso dia a dia e despertou o interesse de diversas pessoas em controlar alguns processos diários de uma residência.

O controle destes processos diários através de um dispositivo móvel exige um aplicativo que possa executar as ações necessárias. Para quem não tem um conhecimento amplo de programação e nem prática, o desenvolvimento de um aplicativo para dispositivo móvel pode ser algo extremamente trabalhoso e o resultado nem sempre vai ser satisfatório. Logo, a frustração pode bater e aquele tão sonhado projeto pode ficar pelo caminho. Da mesma forma que existem pessoas interessadas em controlar tudo de uma forma simplificada, existem pessoas que estão trabalhando constantemente para oferecer soluções que torne possível o desenvolvimento de projetos sem que seja necessário um conhecimento avançado de programação. Uma ferramenta extremamente útil e desenvolvida por estas pessoas é

o Blynk.

2.3.7 Resultados esperado com a construção do protótipo

Deseja-se alcançar, com a construção do protótipo, um equipamento de fácil uso, que proporcione comodidade e praticidade ao usuário, além da segurança de saber que o animal doméstico foi alimentado. Portanto, espera-se: - Construir um alimentador automático que tenha um sistema de controle de vazão de ração, que atue com robustez e confiabilidade, de acordo com as necessidades requeridas pelo o usuário, através da escolha manual e do app. - Desenvolver um aplicativo que atenda o usuário, que seja de fácil uso, e que tenha uma conexão estável com o dispositivo. - Desempenhando as atividades acima, a partir desta, será desenvolvido um produto que atenda a demanda do mercado e que torne mais fácil ter um animal doméstico aos futuros usuários.

Projeto Mecânico

Neste capítulo, o projeto de uma planta piloto de alimentação automática para animais domésticos é apresentado. Nesse contexto, discute-se sobre o projeto do processo, a definição de sua estrutura e quais os sensores e atuadores que são utilizados. Além disso, é apresentado o projeto de um controlador de vazão de ração.

3.1 Definição de como é realizada a vazão de ração

Um recipiente é alocado ao topo de toda a estrutura, aonde será armazenada a ração. Como este recipiente tem uma abertura em sua parte inferior, a força da gravidade é responsável por conduzir a ração até o funil que direcionará a ração até os vãos da hélice que fica localizada logo abaixo, ligado ao recipiente onde ficará alocada a ração. Com a hélice conectada a um motor de passo, este que realizará um movimento giratório apenas no sentido horário, conduzindo então a ração para a rampa que por sua vez direcionará a ração até o pote, onde assim o animal irá se alimentar,

3.2 Definição da dimensão do recipiente

Como o projeto está definido para que o limite máximo de ração seja 15 kg, que ocupa um volume de 18000cm^3 , um recipiente que suportasse esse volume e peso, que atendesse as demais exigências para o armazenamento de alimento deveria ser encontrado. Como se trata da construção de um protótipo não comercial, definiu-se medidas que possibilitassem simular o ambiente em questão, então foi escolhido o galão que é utilizado para armazenar água, sendo ele ideal para representar um silo. Ainda por se tratar de um material que não causa nenhum risco de contaminação do alimento. Para atender a esses critérios, calculou-se a dimensão apresentada na Figura 3.1.

Como pode ser visto na Figura 3.1, a altura do recipiente é de 459 mm, por um diâmetro de 276 mm, onde serão preenchidos com o volume de ração. De acordo com a

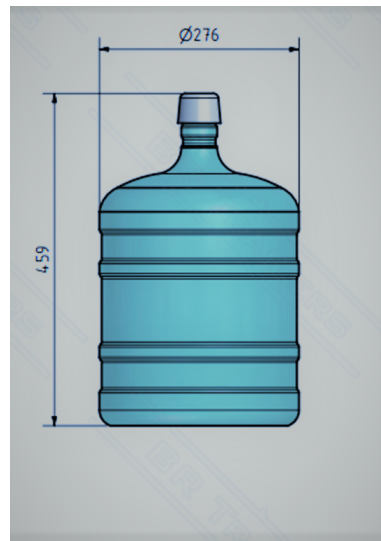


Figura 3.1: Vista frontal do recipiente

Equação 3.1, é possível calcular o volume máximo do recipiente, mas por se tratar de um galão que suporta 20l de água, e sabendo que o litro da água ocupa 1000 cm^3 . Então, a partir desse pressuposto sabemos que este ocupa um volume de 20000 cm^3 de água.

$$V_g = \pi \cdot r^2 \cdot h \quad (3.1)$$

Onde:

- $V_s = \text{Volume} \left(\frac{L}{\text{cm}^3} \right)$
- $r = \text{Raio do galão}(\text{cm})$;
- $h = \text{Altura do galão}(\text{cm})$.

De forma análoga, é possível calcular o volume de ração que o recipiente comportará. Com base em testes, após medir 1kg de ração, chegou-se ao valor de 1200 cm^3 . Dessa forma foi calculado o volume ocupado por 15kg, Equação 3.2.

$$V_r = 1200\text{ cm}^3 \cdot 15 = 18000\text{ cm}^3 \quad (3.2)$$

Onde:

- $V_r = \text{Volume de ração}$

Para definir qual o material de do galão, levou-se em consideração diversos itens, e como galões de água comercializados no Brasil são feitos de termoplástico, que é um polímero artificial que quando aquecido pode ser moldado. Os tipos de termoplástico utilizados para fabricação dos vasilhames de água mineral são o Polietileno Tereftalato , o

PP Polipropileno ou o PC Policarbonato. É um material muito resistente e propício para o armazenamento e conservação da água mineral por diversas razões. Uma delas é a alta rigidez do recipiente, outra é a capacidade de isolamento do plástico, que é uma barreira funcional contra contaminações externas do ar ou de outros corpos estranhos, assim como esse recipiente é ideal para a água, este atenderá os requisitos para armazenamento de ração. Devemos citar sua transparência que permite a entrada de luz, no qual ajuda na conservação da água mas não é ideal no armazenamento de ração, que deve ficar fora do alcance da luz. Mas isso será resolvido pelo fato de ter toda a estrutura em volta do recipiente, isolando totalmente contra a luz.

Com esses dados, definiu-se o material para o recipiente. Além de comportar-se como isolante contra contaminações externas do ar, há ainda a vantagem de viabilidade econômica apresentada por tal equipamento.

Tendo as dimensões do reservatório, foi então desenhado no programa solidworks para que assim fosse criado o projeto por inteiro. A vista frontal e com corte podem ser vistos nas Figuras 3.2 e 3.3. Onde este será utilizado nas demais partes do projeto.



Figura 3.2: Vista frontal do recipiente



Figura 3.3: Vista corte do recipiente

3.3 Projeto do controlador de vazão da ração

3.3.1 Hélice

O sistema de hélices no projeto são duas, a primeira é dosar a quantidade de ração a ser escolhida pelo usuário, e a segunda transportar a ração para a rampa de saída.

A dosagem da ração será feita de modo que a distância entre uma haste e outra da hélice tenha o volume de 50 gramas de ração, ou seja, o vão entre as hastes da hélice funcionará como parâmetro de peso da ração.

Desse modo, o movimento giratório da hélice fará com que uma quantidade de ração seja despejada, tendo assim, um melhor controle para a mecânica do processo. Na figura 3.4 abaixo podemos ver a hélice.

O material a ser utilizado na hélice será o *absacilonitrila butadieno estireno*, que é um material derivado do petróleo, ainda, por se tratar de um com grande aplicabilidade, ou seja, ele é um material leve, resistente a corrosão e oxidação, tem um baixo custo comparado com os demais materiais. E o mais importante, é um material que não contamina o alimento, dessa maneira, sua utilização não terá nenhum risco para o armazenamento da ração. Ainda, por se tratar de um protótipo para testes, a utilização da impressão 3d é de grande praticidade.

Como foi falado anteriormente o tamanho da hélice foi projetado para uma quantidade certa de ração, como podemos ver nas figuras 3.6 e 3.7. Para calcular o tamanho exato da

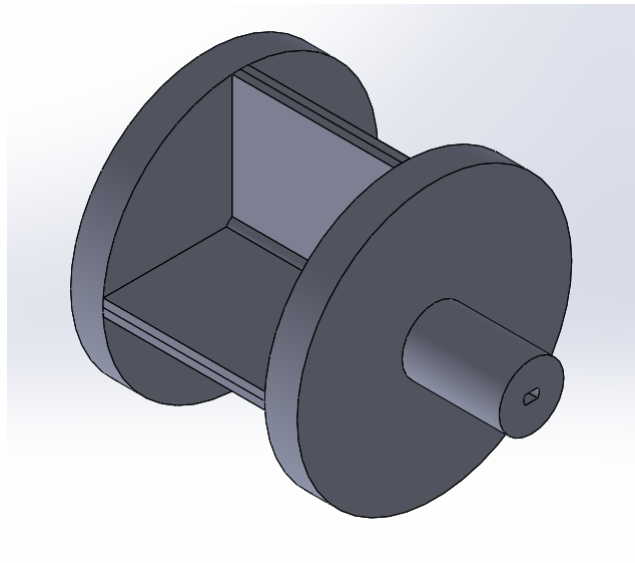


Figura 3.4: Vista panoramica da Hélice

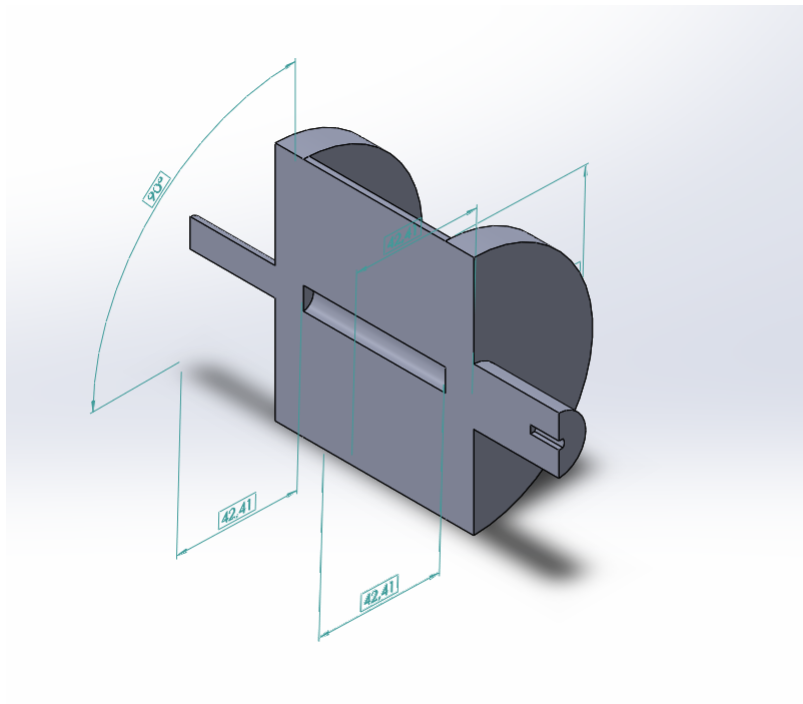


Figura 3.5: Vista corte da hélice

distância entre um pa e outra, foi feito um cálculo de volume . Sabendo que 50 gramas de ração ocupa um volume de:

$$60cm^3$$

, foi utilizada a equação de volume do tronco de cone.

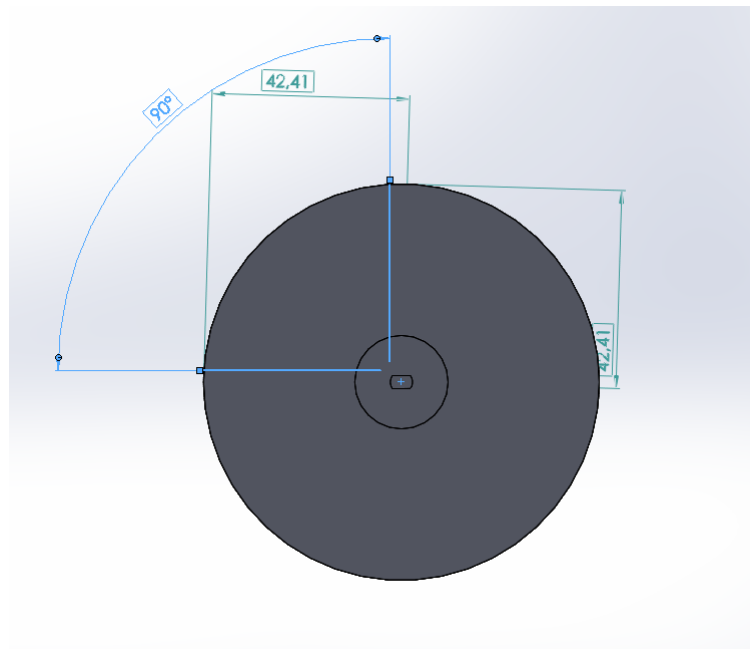


Figura 3.6: Vista frontal da hélice

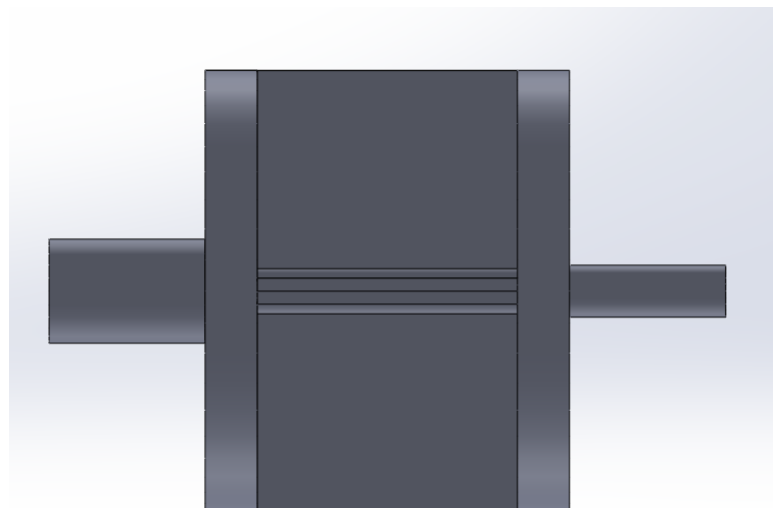


Figura 3.7: Vista superior da hélice

3.3.2 Motor de Passo

A utilização do motor de passo se dá, por ele ser um motor que tem um posicionamento muito preciso e rotação exata. Nesse tipo de motor a rotação do balancete (engrenagem) é controlado por uma série de campos eletromagnéticos que são ativados e desativados eletronicamente. Não usam escovas ou comutadores e possuem um número fixo de polos magnéticos que determinam o número de passos por revolução. O controle computadorizado de motores de passo é uma das formas mais versáteis de sistemas de posicionamento, particularmente quando digitalmente controlado como parte de um servo sistema. Na Figura 3.8 tem-se um esquemático do motor a ser utilizado.

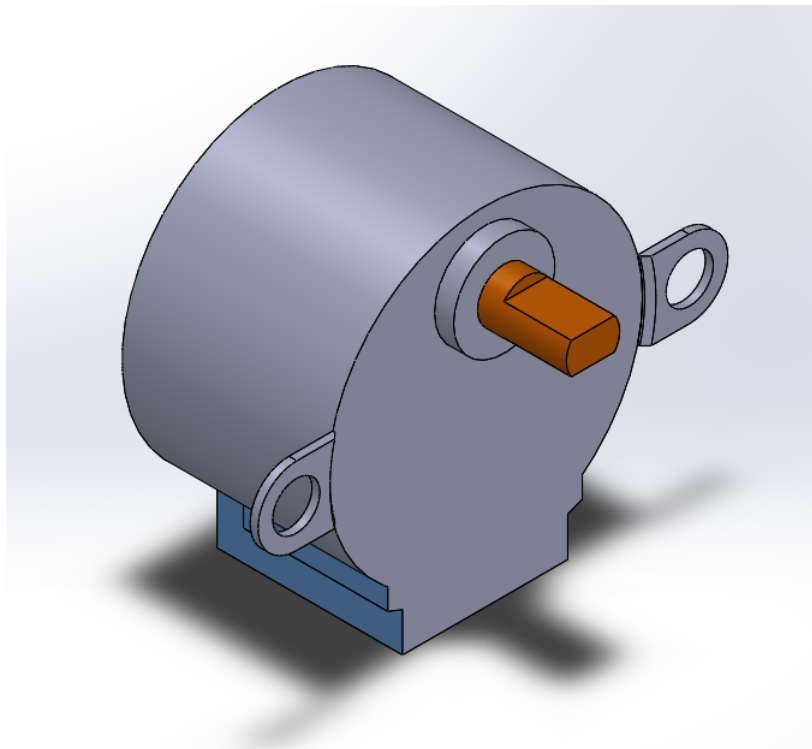


Figura 3.8: Vista panorâmica do motor de passo

3.4 Condutores de ração

3.4.1 Rampa

Uma rampa de metal posicionada estrategicamente em baixo da hélice, essa, que terá como função direcionar a ração até seu destino final, que é a vasilha.

Está será fixada no chão da estrutura.

Abaixo tem a Figura 3.9.

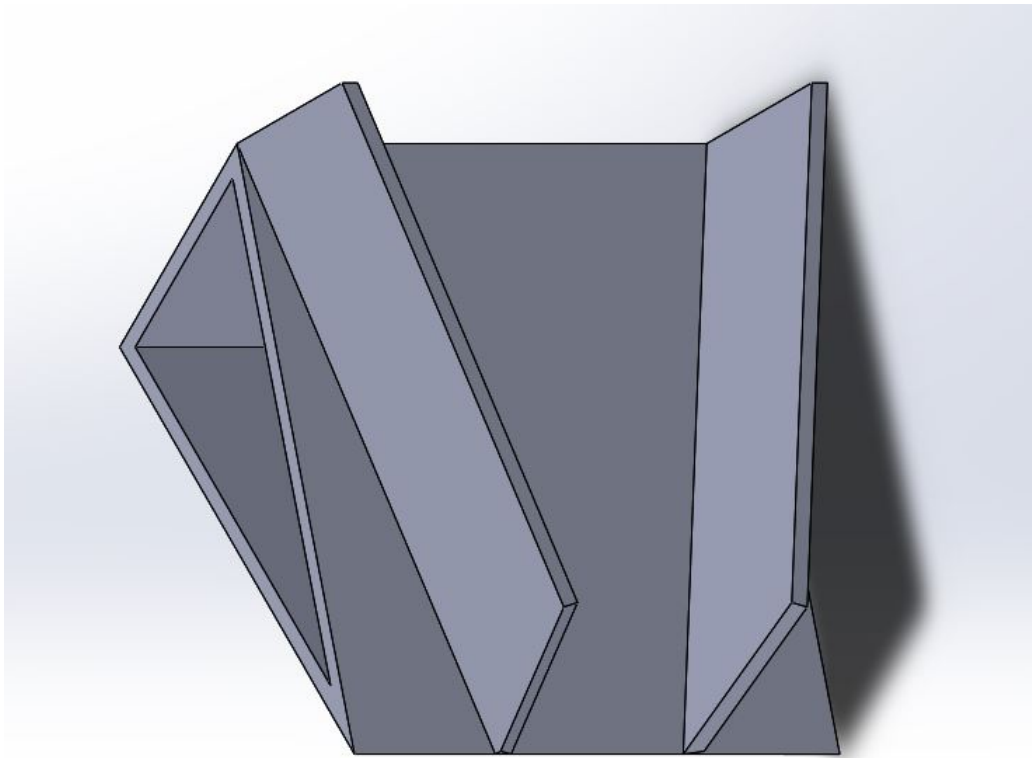


Figura 3.9: Vista panorâmica da rampa

3.5 Estrutura

3.5.1 Estrutura externa

Como se trata de um alimentador para animais, o projeto deve ter uma estrutura firme e estável, que resista a possíveis sobrecargas causada, muitas vezes, pelos próprios animais. Dessa forma, foi projetada uma caixa, que envolve todo o sistema da planta, contendo apenas duas aberturas.

Uma funcionará como *dispenser* da ração, onde este poderá se alimentar, e a outra é por onde de recarrega o recipiente de ração. A estrutura possuirá andares, que serão responsáveis por alocar e prender o recipiente e o motor de passo.

Para a confecção da estrutura optou-se por *Medium Density Fiberboard*, MDF, por ser um material de fácil acesso, por apresentar baixo custo de aquisição e por ser de fácil manuseio. Ainda sobre a escolha do MDF, um último fator foi considerado: o Laboratório de Protótipos do Cefet MG campus V tem as ferramentas e os maquinários necessários para a manufatura deste material.

Na figura 3.10 observa-se desenho esquemático do mecanismo.

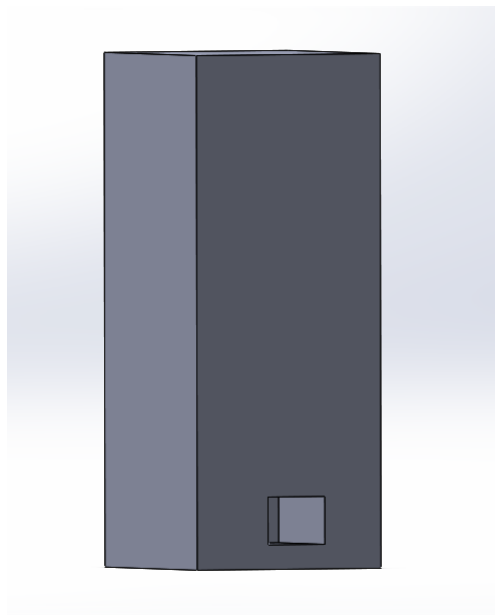


Figura 3.10: Desenho esquemático da estrutura

3.5.2 Estrutura interna

A estrutura interna foi projetada de forma a ser mais compacta possível, para que assim fique um equipamento compacto e que não ocupe muito espaço dentro de casa.

Foram feitos dois andares na estrutura, de forma que o segundo é responsável pelo suporte do recipiente de ração e o primeiro pelo conjunto de equipamentos eletrônicos e mecânicos.

Foi feita uma simulação no programa *SolidWorks* para simular as forças externas e interna em que a estrutura pode ser exposta. Na simulação foi aplicada uma força de 150N (peso máximo do recipiente de ração) na base onde este está fixado, uma força de 20N foi aplicada na base onde o conjunto de sensores e atuadores foram fixados.

No *software SolidWorks* foi feita toda a simulação sobre a estrutura. Para realizar este procedimento, é necessário aplicar uma malha sobre o objeto, está, é responsável por calcular o esforço em cada ponto do objeto.

Após o procedimento de malha, é necessário fixar o objeto, no caso, foi fixado nos locais onde se colocou os parafusos. Só assim então, é aplicada a força no ponto escolhido para simulação.

Tomando de base a simulação, nota-se que a estrutura resiste as forças aplicadas nesta. As imagens 3.11, 3.12 e 3.13, respectivamente de análise de tensão, deformação e deslocamento corroboram para a aplicabilidade da estrutura no trabalho.

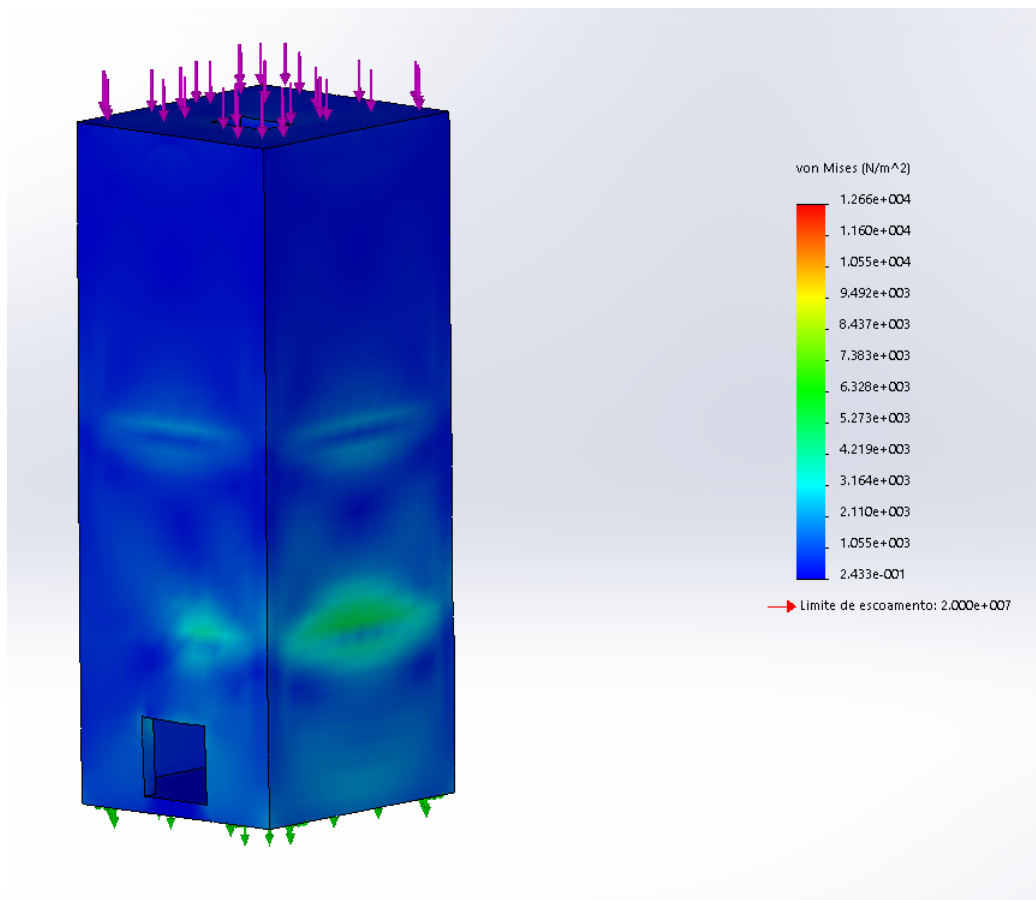


Figura 3.11: Simulação-aplicação de tensão na estrutura

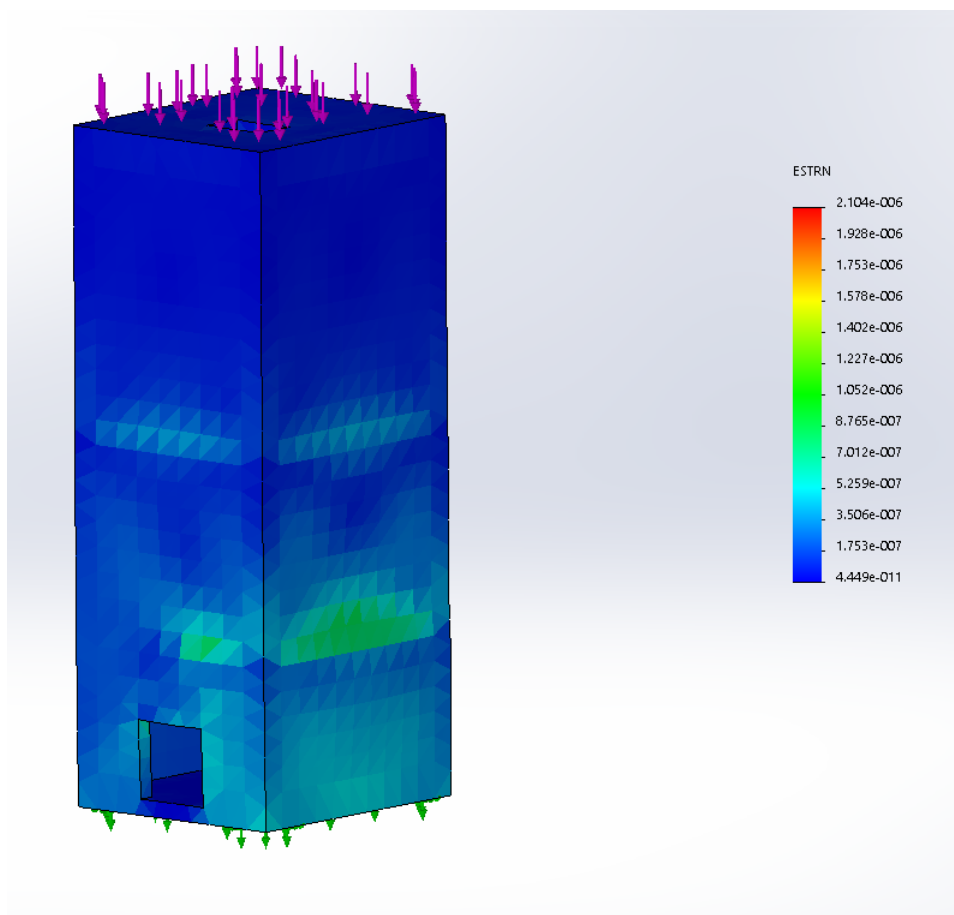


Figura 3.12: Simulação-deformação estrutura

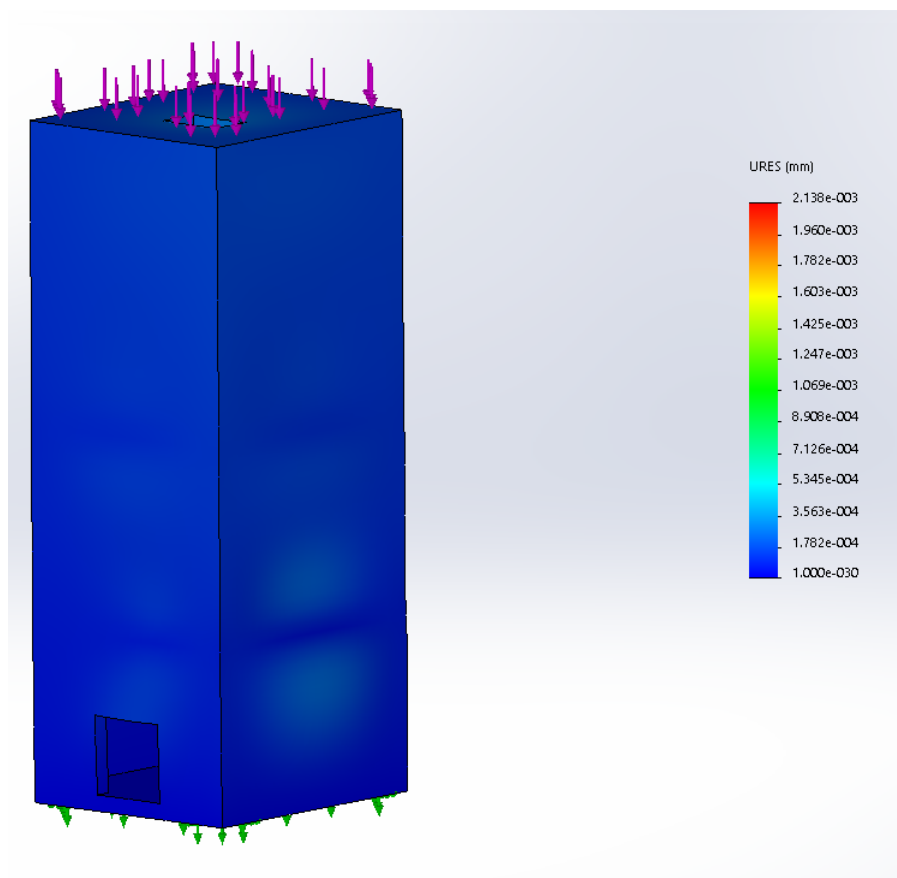


Figura 3.13: Simulação-deslocamento estrutura

3.6 Definição do Atuador

3.6.1 Motor de Passo

A Tabela 3.1 apresenta os dados do motor de Passo, de acordo com seu *datasheet*.

Tabela 3.1: Dados referentes ao Motor de Passo

ESPECIFICAÇÕES	Motor de Passo
Modelo	28BYJ-48
Tensão de operação	5 VDC
Ângulo do Passo (°)	5.625 / 64
Número da Fase L	4
Taxa de Variação de Velocidade	1/64
Frequência	100Hz
Resistência	DC 50 ± 7(25)
Frequência inativa da tração	1000Hz
Torque em tração	34,3mN.m (120Hz)
Torque de auto posicionamento	34.3mN.m
Torque de fricção	600-1200 gf.cm
Puxe o torque	300 gf.cm
Indutância Precisão	+ -20
Resistência isolada	80 10M? (500V)
Potência de eletricidade isolada	600VAC / 1mA / 1s
Grau de isolamento	0.5 A
Aumento de temperatura	40k (120Hz)
Ruído	35dB (120Hz, sem carga, 10cm)

3.6.2 Driver do Motor de Passo

Este Módulo Driver integra o CI ULN2003, que é composto por sete transistor *Darlington*s. Dessa fora é possível controlador um motor de passo unipolar com 4 fases e 5 fios. O Módulo suporta até 500 mA por canal e tem uma queda de tensão interna de cerca de 1V quando ligado. Ele também contém diodos de fixação internos para dissipar picos de tensão durante a condução de cargas indutivas. Na Figura 3.14 pode-se observar o motor de passo juntamente com o driver.

Tabela 3.2: Dados referentes ao Driver Motor de Passo

ESPECIFICAÇÕES	Driver Motor de Passo
Modelo	ULN2003
Tensão de operação do Módulo	3 5.5V
Tensão de operação do motor de passo suportado pelo Módulo	5 12V
Temperatura de funcionamento do Módulo	-25 °C 90 °C.



Figura 3.14: Motor de passo e driver

3.7 Definição dos Sensores

3.7.1 Sensor infravermelho

O sensor de obstáculo infravermelho 3.15 é um circuito composto por um emissor e receptor. Seu funcionamento é simples, quando algum obstáculo é colocado em frente ao sensor, o sinal infravermelho é refletido para o receptor. Quando isso acontece, o pino de saída *OUT* (fora) é colocado em nível baixo(0), no caso do projeto, o sensor infravermelho indicara ao programa, a real quantidade de ração presente no recipiente. Haverá três níveis, que são, cheio, médio e baixo. Dessa forma, o usuário terá conhecimento da real quantidade de ração restante.



Figura 3.15: Sensor infravermelho

Tabela 3.3: Dados referentes ao Sensor Infravermelho

ESPECIFICAÇÕES	Sensor Infravermelho
Sensor de obstáculo IR	
Tensão de operação	3.3 à 5V DC
Emissor e receptor IR	
Distância de detecção	2 à 80 cm
Potenciômetro para ajuste da distância	
Dimensões	37 x 14 x 6 mm

3.7.2 Sensor de umidade e temperatura

O Sensor de Umidade e Temperatura DHT11 3.16 é um dos componentes mais utilizados em projetos que envolva medição de temperatura e umidade ambiente. Este sensor faz medições de temperatura de 0° até 50° celsius e mede a umidade do ar nas faixas de 20% a 90%. A precisão (margem de erro) do sensor para medição de temperatura é de aproximadamente 2° celsius e para umidade é de 5%.

Este Sensor de Umidade e Temperatura DHT11 pode ser utilizado em diversos projetos com Arduino, PIC, Raspberry, NodeMCU ESP8266 e outros microcontroladores. Em projetos de automação residencial, por exemplo, pode ser utilizado em locais que seja necessário obter temperatura e umidade ambiente de forma constante e essas informações serão enviadas ao microcontrolador que está conectado a internet, para em seguida serem apresentadas na tela de um *smartphone* ou *tablet Android*.

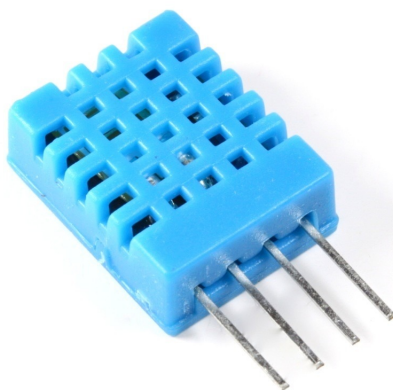


Figura 3.16: Sensor Umidade e Temperatura - DHT11

Tabela 3.4: Dados referentes ao Sensor de umidade e temperatura

ESPECIFICAÇÕES	Sensor de umidade e temperatura
Sensor de obstáculo IR	
Tensão de operação	3.3 à 5V DC
Emissor e receptor IR	
Distância de detecção	2 à 80 cm
Potenciômetro para ajuste da distância	
Dimensões	37 x 14 x 6 mm

3.7.3 Sensor de distância

O Sensor Ultrasonico HC-SR04 3.17 é um dos sensores mais conhecidos do mercado. Pode-se dizer que todos que estão iniciando com Arduino sempre procuram pelo Sensor

Ultrasonico HC-SR04 para desenvolver seus primeiros projetos medindo distâncias de obstáculos/objetos em relação ao sensor. O Sensor Ultrasonico HC-SR04 é aplicado com mais frequência em projetos de robótica, principalmente em chassis robóticos, robôs ou carrinhos. O sensor é capaz de medir com precisão (3mm de margem de erro) distâncias de 2cm até 4m. A composição do Sensor Ultrasonico HC-SR04 é feita de um emissor e um receptor ultrassônico, onde o sensor emite (emissor) sinais ultrassônicos que serão refletidos no obstáculo / objeto retornando (receptor) ao sensor. Com base no tempo que o sinal emitido levou para retornar ao sensor, o mesmo efetua o cálculo da distância. O sinal enviado pelo Sensor Ultrasonico HC-SR04 é equivalente a velocidade do som (340 metros por segundo), logo, o sensor estando a uma distância X do obstáculo/objeto, o sinal enviado vai percorrer uma distância que equivale a $2x$, pois o sinal percorreu o caminho de ida e em seguida fez o caminho de volta na mesma proporção. Além disso, é compatível com Arduino, NodeMCU ESP8266 e outros microcontroladores.



Figura 3.17: Sensor Ultrasonico HC-SR04

Tabela 3.5: Dados referentes ao Sensor Ultrasonico HC-SR04

ESPECIFICAÇÕES	Sensor Ultrasonico HC-SR04
Sensor de obstáculo IR	
Tensão de operação	3.3 à 5V DC
Emissor e receptor IR	
Distância de detecção	2 à 80 cm
Potenciômetro para ajuste da distância	
Dimensões	37 x 14 x 6 mm

3.7.4 Microprocessador

O microprocessador do sistema foi escolhido de acordo com os seguintes parâmetros: resolução das entradas analógicas, velocidade de leitura de dados, tensão e corrente de saída. De acordo com esses critérios, definiu-se o *Arduino UNO* para o desenvolvimento do trabalho, Figura 3.18 .



Figura 3.18: Arduino UNO

O Arduino Uno R3 com Cabo USB A/B é o que chamamos de plataforma de computação física ou embarcada, ou seja, um sistema que pode interagir com seu ambiente por meio de hardware e software.

O Arduino é um projeto de código aberto tanto em software como em hardware. Voltado para prototipagem eletrônica, tem como principais características a fácil utilização e flexibilidade. Foi desenvolvido em 2005 por Massimo Banzi (e outros colaboradores), na Itália, para auxiliar estudantes de design e artistas no aprendizado de eletrônica, tendo como ponto forte o baixo custo para que os estudantes pudessem ter fácil acesso a uma plataforma que futuramente viria a ser conhecida internacionalmente e utilizada em milhares de projetos em diversas áreas.

Cada versão do Arduino possui um microcontrolador que é responsável por gerenciar e executar todas as ações da placa. Dentre as versões oficiais que estão disponíveis no mercado, as que mais se destacam é o Arduino Uno R3 e o Arduino Mega 2560 R3. A placa é capaz de interagir com o software e hardware de forma direta, possibilitando a integração com diversos atuadores, sensores e demais eletrônicos.

Devido a sua fácil utilização e flexibilidade, o Arduino é aplicado em diversos projetos de robótica (robôs, carrinhos e chassis), automação industrial, automação residencial e na

construção de impressoras 3D (RepRap).

O Arduino Uno R3 oferecido é a versão *open hardware* (compatível), contudo possui o layout, a mesma estrutura de montagem e componentes da placa original italiana. Vale ressaltar que a placa é compatível com todos os produtos da linha arduino. A placa é de qualidade e não perde em nada para o original italiano.

Tabela 3.6: Dados referentes ao Arduino UNO

PARAMETROS	Arduino Uno R3 com Cabo USB A/B
Modelo da placa	Uno R3
Microcontrolador	ATmega328p
Tensão de operação	5VDC
Tensão de alimentação recomendada (externa)	7 - 12VDC
Limite de tensão de alimentação (externa)	6 - 20VDC
Portas digitais (I/O)	14 (das quais 6 oferecem PWM)
Entradas analógicas	6
Corrente DC por porta digital (I/O)	40mA
Corrente DC por entrada analógica	50mA
Memória Flash	32kb (ATmega328p) dos quais 0,5Kb
SRAM	2Kb (ATmega328p)
EEPROM	2Kb 1Kb (ATmega328p)
Velocidade do Clock	16MHz
Dimensões	53mm(L) X 13mm(A) X 68mm(C)
Peso	29g

3.8 Definição da Fonte

3.8.1 Fonte Ajustável 3.3V 5V para Protoboard MB102

A Fonte Ajustável 3.3V / 5V para Protoboard MB102 é um componente desenvolvido com a finalidade de alimentar as linhas laterais de protoboards (breadboards / matriz de contato) padrão MB102.

Esta Fonte Ajustável 3.3V / 5V para Protoboard MB102 é composta basicamente de um conector jack (P4) fêmea que aceita alimentação de 6,5 a 12VDC, um botão para ligar e desligar, um conector USB para alimentação, pinos 5VD, 3,3V e GND para ramificar alimentação, e pinos dispostos de forma que sejam encaixados nas laterais da protoboard (breadboard / matriz de contatos) padrão MB102.

Na saída dos pinos da placa terá 5V ou 3,3V. A Fonte Ajustável 3.3V / 5V para Protoboard MB102 é bastante utilizada em projetos com Arduino, PIC, Raspberry, NodeMCU ESP8266 e muitos outros microcontroladores.

Vale ressaltar que a tensão DC dos pinos que são inseridos na protoboard podem ser ajustados para 5V ou 3,3V através de jumpers que estão disponíveis na placa. Logo, você

pode ter 5V de um lado da protoboard e 3,3V do outro lado.

Desse modo, com a fonte ajustável, todos sensores e atuadores estão conectados na mesma placa, reduzindo assim espaço dentro da estrutura.

Tabela 3.7: Dados referentes a Fonte Ajustável 3.3V 5V para Protoboard MB102

ESPECIFICAÇÕES	Fonte Ajustável 3.3V 5V
Tensão de entrada	6,5 a 12 VDC / USB
Tensão de saída (ajustável por jumpers):	3,3VDC ou 5VDC
Corrente máxima de saída	700mA
Compatível com protoboard MB102	
Dimensões	32mm(L) X 24mm(A) X 53mm(C)
Peso	12g



Figura 3.19: Fonte Ajustável 3.3V 5V para Protoboard MB102

3.8.2 Fonte chaveada

A Fonte DC 9V 1A Bivolt Chaveada é ideal para fazer a alimentação externa do Arduino e outras plataformas microcontroladas que possuam plug P4 fêmea com positivo no centro.

Esta fonte faz o ajuste automático da tensão na entrada (100VAC até 240VAC) para uma tensão de saída de 9VDC.

Além disso, a fonte possui um circuito responsável por fazer o chaveamento da corrente e isso garante que a tensão na saída seja estável.

Dependendo da quantidade de entradas e saídas conectadas na plataforma microcontrolada, a alimentação vinda da porta USB não será o suficiente para fornecer a corrente necessária. Ao utilizar a Fonte DC 9V 1A Bivolt Chaveada você garante que seu projeto irá funcionar conforme o esperado e sem falhas devido a falta de corrente.

A Fonte DC 9V 1A Bivolt Chaveada também é bastante utilizada com a Fonte Ajustável 3.3V / 5V para Protoboard MB102.



Figura 3.20: Fonte chaveada

Tabela 3.8: Dados referentes a Fonte chaveada

ESPECIFICAÇÕES	Fonte chaveada
Tensão de entrada	bivolt 100 240V AC($\pm 10\%$)
Plug de saída:	P4
Corrente de saída (máxima)	1A
Tensão de saída	9VDC
Proteção	subtensão / sobretensão
Proteção	curto-circuito na saída
Proteção	sobrecarga
Proteção	transientes na rede elétrica
Proteção	ruídos na rede elétrica (rfi/emi)
Proteção	ruídos na carga (rfi/emi)
Isolação	entrada / saída
Dimensões	25mm(L) X 72mm(A) X 71mm(C)
Peso	49g

3.9 Protoboard

Uma protoboard (matriz de contato / breadboard) é uma placa com vários furos e conexões condutoras para montagem de circuitos elétricos / eletrônicos experimentais. A grande vantagem da protoboard na montagem de circuitos é a facilidade de inserção de componentes, uma vez que não necessita soldagem e você pode testar o circuito antes de finalizar o projeto.

Em universidades e escolas técnicas a Protoboard de 400 Pontos é utilizada em laboratórios de elétrica / eletrônica para que os estudantes possam desenvolver as práticas propostas de forma simples e objetiva. A Protoboard de 400 Pontos possui um adesivo de fixação em sua parte traseira para que possa ser fixada em uma base ou alguma outra superfície.

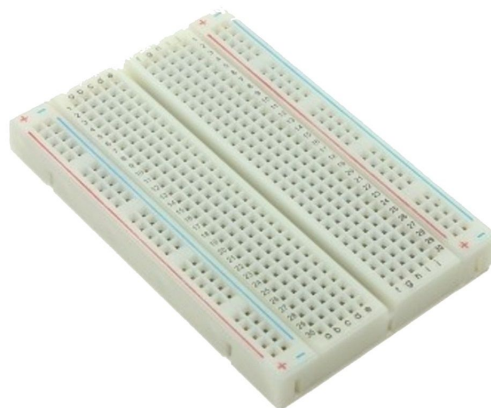


Figura 3.21: Protoboard



Figura 3.22: Protoboard

Tabela 3.9: Dados referentes a Protoboard

ESPECIFICAÇÕES	Protoboard
Pontos de conexão	400
Adesivo para fixação	
Dimensões	55mm(L) X 10mm(A) X 83mm(C)
Peso	46g

3.10 Comunicação

3.10.1 ESP8266

O Módulo ESP8266 é um dispositivo *IoT* (Internet das Coisas) que consiste de um microprocessador ARM de 32 bits com suporte embutido à rede WIFI e memória flash integrada. Essa arquitetura permite que ele possa ser programado de forma independente, sem a necessidade de outras placas microcontroladoras como o Arduino, por exemplo. Vale ressaltar que o ESP8266 além de ser programado em LUA (linguagem de programação desenvolvida no Brasil), também é compatível com o ambiente de programação (IDE) do Arduino.

Uma das grandes vantagens do Módulo WiFi ESP8266 ESP-01 é possuir WiFi nativo. A possibilidade de projetos na área de robótica e automação residencial é infinita. O Módulo WiFi ESP8266 ESP-01 também pode ser utilizado junto ao Arduino para agregar conexão sem fio a placa e tornar os projetos muito mais interessantes. O módulo tem suporte as redes WiFi 802.11 b/g/n e criptografia WPA e WPA2. Com a plataforma embarcada conectada a rede sem fio, você poderá integrar seus projetos com *smartphone* e *tablets*, seja Android ou iOS. Vale ressaltar que o nível lógico que o ESP8266 opera é de 3,3V e não é tolerante a 5V. Logo, ao efetuar as conexões do módulo é importante que essa questão do nível lógico seja feita da forma correta. Além disso, a fonte de alimentação do módulo deve ser de pelo menos 300mA para o correto funcionamento.

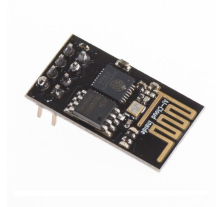


Figura 3.23: Módulo WiFi ESP8266 ESP-01

Tabela 3.10: Dados referentes a Módulo WiFi ESP8266 ESP-01

ESPECIFICAÇÕES	Módulo WiFi ESP8266 ESP-01
Modelo	ESP-01
Controlador	ESP8266
Tensão de operação:	3.3VDC
Suporte a redes	802.11 b/g/n
Potência de saída	+20dBm no modo 802.11b
Alcance	90m
Taxa de dados	250kbps a 2Mbit
Consumo em Standby	<1.0mW
Processador	Arm 32 bits
Comunicação	Serial
Comunicação	TCP / UDP
Criptografia	<i>OPEN/WEP/WPA_PSK/WPA2_PSK/WPA_WPA2_PSK</i>
Temperatura de operação	-40 a 125° celsius
Dimensões	14mm(L) X 10mm(A) X 24mm(C)
Peso	2g

3.10.2 Aplicativo

Um projeto detalhado e bem estruturado é primordial para ter um bom aplicativo. Com isso etapas foram criadas para que este saia de forma planejada.

- Para criar um aplicativo, deve primeiro ter uma ideia, os aplicativos servem para resolver problemas rotineiros da sociedade. Assim, o aplicativo em questão tem este como principal objetivo.
- O aplicativo vai funcionar no sistema operacional *mobile Android*. A estrutura de navegação do *app* será por menu imagem, ou seja, serão ícones de ilustrações, tendo assim uma forma intuitiva de navegar. Com o *app* online, qualquer parâmetro do sistema poderá ser alterado. Já em modo offline, o sistema manterá um modo offline escolhido pelo usuário. Toda ação do usuário deve acontecer em no máximo 3 cliques. Assim, o aplicativo deve ser rápido, dinâmico e oferecer uma experiência agradável ao usuário.
- Criar um *app* pensando em como ele pode ser completo pode atrapalhar o processo de criação deste. O objetivo é focar o *Minimum Viable Product*, MVP, as principais funcionalidades e os requisitos e recursos essenciais.

3.11 Construção parte Mecânica

Para construir a estrutura foi necessário apenas utilização de uma furadeira e uma chave Philips. Pelo fato de que todas as chapas de madeira foram encomendadas, seguindo

especificações projeto, foi necessário apenas fixá-las. Foi utilizado 28 parafusos em toda estrutura.

O mancal foi parafusado em um dos andares da estrutura, e alinhado com o eixo da hélice, de modo a ter um movimento livre a hélice, com controle de giro por apenas um dos lados.

O motor de passo foi fixado com uma abraçadeira D, foram utilizados 2 parafusos nas laterais. Duas dobradiças foram colocadas no topo da estrutura, e fixadas na tampa da estrutura, de modo que a tampa pode ser aberta para a reposição da ração.

Uma abertura foi feita na parte de saída do galão, para que aumentasse a vazão da ração, fazendo que não ocorresse entupimento.

3.11.1 Parafuso

O Parafuso 3,5x30mm é utilizado para fixação em madeira. Sua rosca Auto Atarraxante é maleável o que permite perfurar a madeira sem danificá-la e ao mesmo tempo é resistente a grandes cargas, o que diminuem o risco de quebrar. Ideal para aplicação em aglomerado e MDF.



Figura 3.24: Parafuso de Aço para Madeira 3,5x30mm Auto Atarraxante Chata Philips

3.12 Construção parte Elétrica

Para realizar a montagem do sistema de controle foi bem simples, foi utilizado apenas jumpers Macho e Fêmea. A montagem do sistema pode ser vista na Figura 3.44.

3.13 Programação

Foi utilizado o *software* do arduino. o Arduino IDE é a interface para fazer a programação para controlar o sistema. Este utiliza a linguagem de programação C++.

3.14 Blynk

O Blynk foi desenvolvido para ser utilizado em projetos *Iot*, que é um termo utilizado para descrever a forma como objetos do mundo real permanecem conectados em rede e podem ser acessados através da internet.

Uma das características do Blynk é permitir que a comunicação com uma plataforma microcontrolada / embarcada possa ser realizada sabendo o mínimo de programação e até mesmo sem criar uma linha de código. Ele permite que plataformas sejam controladas remotamente, de forma que dados de sensores e módulos possam ser obtidos e exibidos no aplicativo que fica instalado no dispositivo móvel. Permite também que cargas sejam acionadas, além de muitas outras funcionalidades que a ferramenta disponibiliza.

Com o Blynk tem a possibilidade de controlar a plataforma de qualquer lugar do mundo através do aplicativo instalado no dispositivo móvel. Para isto, basta que a plataforma esteja configurada e conectada ao servidor Blynk através da internet e que o aplicativo no dispositivo móvel também possua conexão com a internet. O Blynk é parcialmente *open source*: o servidor e bibliotecas possuem código aberto, enquanto o aplicativo mobile é de código fechado.

3.14.1 Composição do Blynk

App Blynk: possibilita criar interfaces de controle de forma simples, onde é necessário apenas arrastar os *widgets* e em poucos passos fazer a configuração.

Servidor Blynk: é responsável por todas as comunicações entre o dispositivo móvel e a plataforma. Você pode usar o Blynk Cloud ou executar um servidor Blynk em sua máquina local. O servidor pode trabalhar com diversos dispositivos, inclusive pode rodar em um Raspberry Pi.

Bibliotecas Blynk: há bibliotecas para todas as plataformas mais populares e compatíveis com o Blynk, permitindo a comunicação com o servidor na nuvem (cloud) ou local, processando todos os comandos de entrada e saída.

Na imagem abaixo podemos ver a arquitetura de funcionamento:

Além do aplicativo, servidor e bibliotecas, também é necessário uma conexão com a internet e uma plataforma (Arduino, ESP8266, NodeMCU, Raspberry ou qualquer plataforma compatível).

A seguir, pode ser visto algumas características do Blynk:

- A API e a interface é semelhante para as plataformas suportadas;
- A conexão com a cloud (nuvem) pode ser feita utilizando:
 - Ethernet;

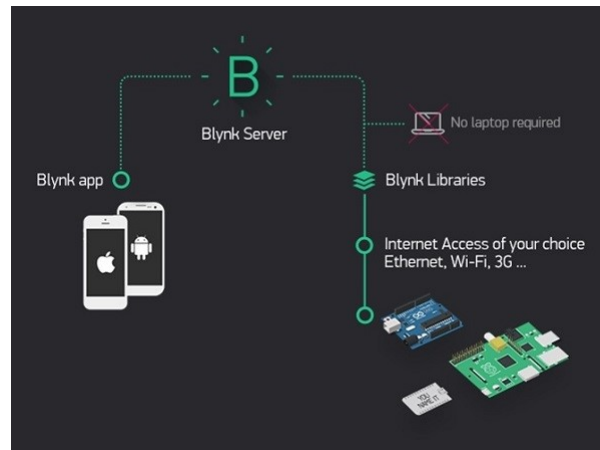


Figura 3.25: Arquitetura de funcionamento Blynk

- WiFi
- 3G;
- Bluetooth;
- USB (Serial);
- Conjunto de widgets de fácil utilização;
- Manipulação de pinos pode ser feita de forma direta sem nenhuma escrita de código na plataforma;
- Fácil de integrar e adicionar novas funcionalidades utilizando pinos virtuais;
- Monitoramento de dados através do widget Histórico Gráfico;
- Comunicação de dispositivo para dispositivo utilizando o widget Bridge (ponte);
- Envio de e-mails, tweets, notificações push e etc;

3.14.2 Requisitos para utilizar o Blynk

Plataforma (hardware): Arduino, ESP8266, NodeMCU, ESP32, Raspberry Pi ou alguma plataforma que figure na lista de dispositivos suportados.

Acesso à internet: O Blynk funciona através da internet, logo, a plataforma que você escolher deverá ser capaz de se conectar à internet.

Smartphone: o aplicativo Blynk possui uma interface bem intuitiva e o mesmo funciona em dispositivos móveis Android ou iOS.

Conta de usuário: após a instalação do aplicativo no dispositivo móvel é necessário a criação de uma conta de usuário para poder utilizar a ferramenta.

Biblioteca Blynk: para uso do Blynk será necessário efetuar a instalação de um pacote de bibliotecas.

3.14.3 Download e instalação do Blynk

Para fazer o download e instalação do app Blynk em seu dispositivo móvel (smartphone ou tablet), será necessário que acesse a loja de aplicativos do seu dispositivo. Se utiliza um dispositivo com sistema operacional Android, então basta acessar a Play Store diretamente do seu aparelho, fazer o download e a instalação do app Blynk. Caso utilize um dispositivo com sistema operacional iOS (Apple), basta acessar a iTunes App Store e fazer o download e instalação do Blynk.

Após o download e instalação do Blynk no dispositivo móvel, o usuário deverá clicar em *Login* para que ele possa entrar no aplicativo. No caso, um e-mail e uma senha será disponibilizado para o usuário.

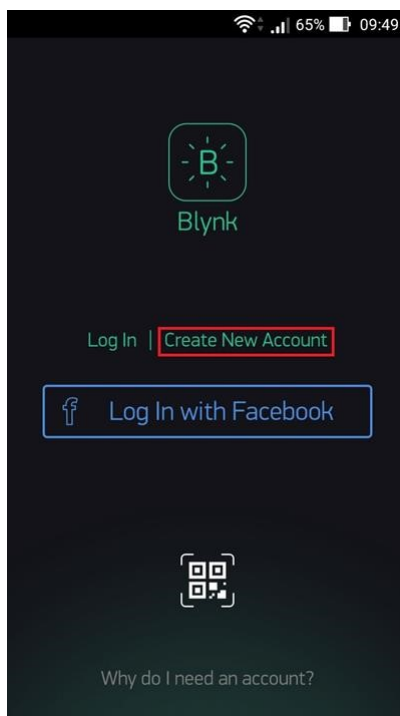


Figura 3.26: Abertura app blynk

Após entrar no aplicativo pela primeira vez, terá uma tela de criação do primeiro projeto. Para o usuário a tela já será direcionada para o projeto.

3.14.4 Configuração do Blynk no dispositivo móvel

Nesta primeira tela 3.27, é onde se inicia o projeto do sistema.

A: botão *Create New Project*. Criar o seu primeiro projeto Blynk.

B: botão *My Apps*. Ao clicar neste botão você será direcionado para uma outra tela que possui o botão *Create App Preview* que permite criar um app e dentro deste app disponibilizar um ou mais dos projetos Blynk. O app que foi criado ficará disponível nas apps stores para que outros usuários possam fazer o download.

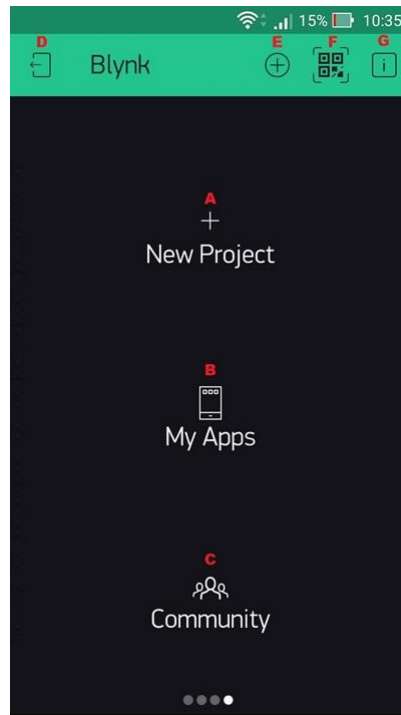


Figura 3.27: Aplicativo blynk

C: botão *Community*. Ao clicar neste botão será direcionado para o fórum do Blynk.

D: botão para fazer logout da conta, ou seja, retornar para a tela de login.

E: possui a mesma função do botão *Create New Project*.

F: permite ler um QRcode e copiar um projeto já existente e que foi compartilhado por algum usuário do Blynk.

G: traz algumas informações do aplicativo.

Com a finalidade de demonstrar e exemplificar a utilização do Blynk, a partir daqui as configurações serão feitas com base no uso do Arduino Uno em conjunto com o ESP8266.

Então, para criar o projeto, primeiro deve clicar em *Create New Project*. Em seguida, dar um nome para o projeto, em *CHOOSE DEVICE* selecione a plataforma que vai utilizar, em *CONNECTION TYPE* (Arduino) selecione o tipo de conexão que será utilizado (WIFI), em *THEME* selecione a cor do tema (se optar pelo DARK, o app manterá as cores padrão, contudo, se optar pelo LIGHT, o app mudará a cor preta para branca e vai manter a cor verde que é presente em ambos os temas) e por último clique no botão *Create*.

Após a criação do projeto será exibida uma mensagem, informando que o *Auth Token* do projeto foi enviado ao endereço de e-mail.

No e-mail que foi colocado no cadastro receberá o Token.

O *Auth Token* é uma chave de 32 caracteres alfanuméricos e ela possibilita que o servidor Blynk consiga direcionar as informações entre o aplicativo instalado no dispositivo

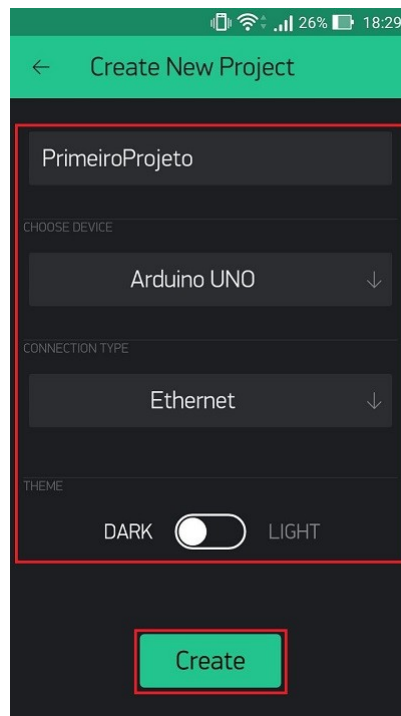


Figura 3.28: Configuração Projeto Aplicativo blynk

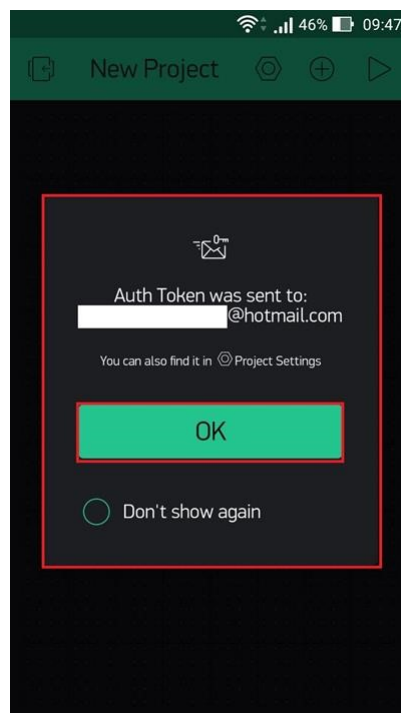


Figura 3.29: Token Aplicativo

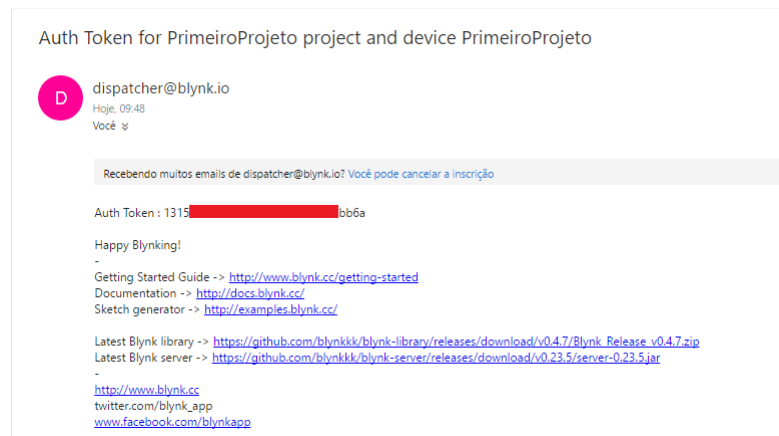


Figura 3.30: E-mail com token

móvel e a plataforma. O *Auth Token* deverá ser inserido nos códigos que serão carregados na plataforma.

Com o primeiro projeto criado, aparecerá um contêiner principal e mais algumas opções:

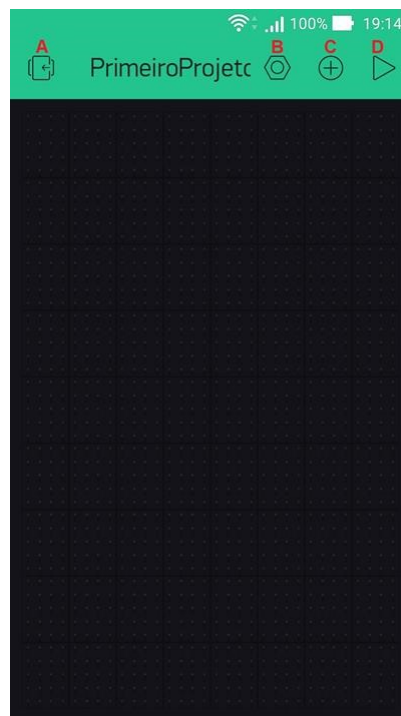


Figura 3.31: Opções do Aplicativo blynk

A: este botão já havia aparecido numa tela anterior, contudo, na tela atual ele tem uma segunda funcionalidade que no caso é a de voltar para a tela de selecionar / criar um novo projeto.

B: acessar as configurações do projeto. Na tela em que abrir verá o campo para alterar o nome do projeto, uma opção (*SHARED ACCESS*) para compartilhar o seu projeto com

outros usuários, uma opção (*Generate Link*) para gerar o link de compartilhamento (este compartilhamento será feito através de um QRcode que será gerado e você deverá repassar a quem deseja que tenha acesso ao seu projeto) e por fim uma opção *Add Shortcut* que tem como finalidade inserir na tela inicial do seu dispositivo móvel, um atalho do projeto para que o mesmo possa ser aberto de uma forma mais rápida. Mais abaixo há a opção *DEVICES*. Clique sobre esta opção para ser direcionado a tela *MY DEVICES*:

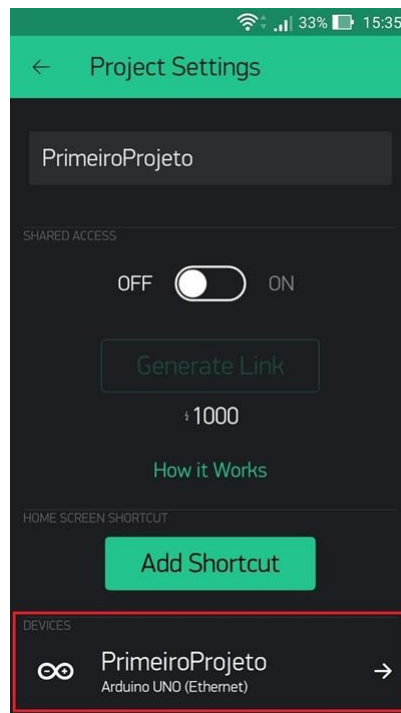


Figura 3.32: Configurações Aplicativo blynk

Na tela *MY DEVICES* (Meus Dispositivos) tem a opção de acessar as configurações do dispositivo registrado no projeto, pode adicionar um novo dispositivo através da opção *+ New device* e pode adicionar uma nova tag através da opção *+ New Tag*. Clique na primeira opção (seu dispositivo registrado):

Na tela que abrir verá as informações que selecionou quando estava criando o seu primeiro projeto:

Em *HARDWARE MODEL* pode ser alterado a plataforma que será utilizada, em *CONNECTION TYPE* pode ser alterado o tipo de conexão que será utilizada e em *AUTH TOKEN* temos a chave única do projeto e se clicar sobre o endereço da chave, a mesma será copiada pra o clipboard. Na opção *Refresh* pode atualizar *Auth Token*, contudo, essa opção só deve ser usada para desfazer do antigo, pois ele será substituído. Ao lado do *Refresh* há a opção E-mail e esta opção possibilita que você envie ao seu e-mail de cadastro no Blynk o *Auth Token* do projeto.

Abaixo do primeiro projeto há a opção *+ New Device* e esta opção possibilita criar um

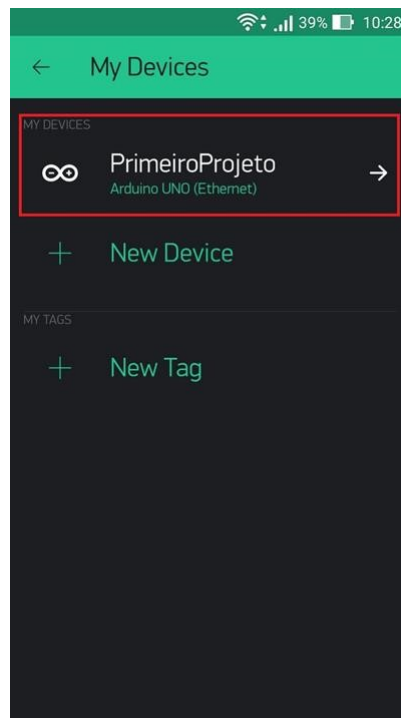


Figura 3.33: Projeto Aplicativo blynk

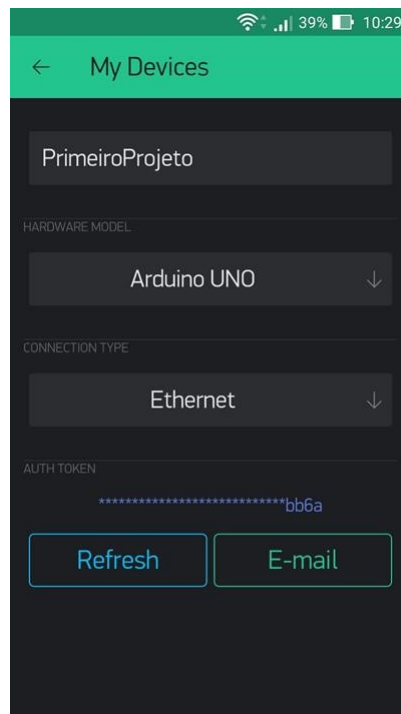


Figura 3.34: Projeto Aplicativo blynk

novo projeto, selecionando a plataforma e o tipo de conexão. Ao criar um novo dispositivo, será gerado um Auth Token. Na opção *+ New Tag* pode criar uma tag que pode unir dois ou mais projetos que foi criado, e a partir de apenas um comando pode controlar estes projetos (dispositivos):

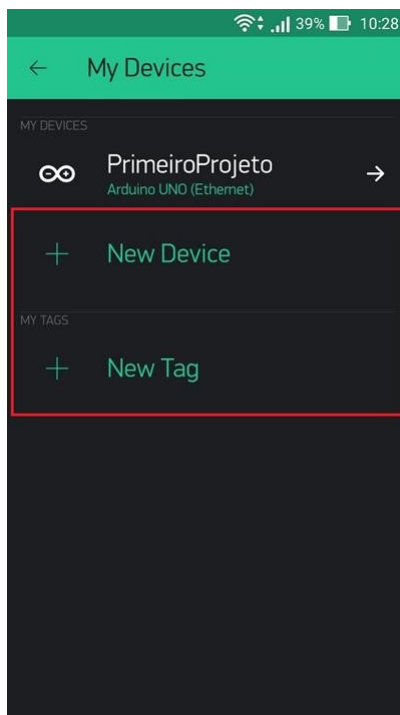


Figura 3.35: Opção criar mais projeto Aplicativo blynk

Mesmo que o segundo ou terceiro projeto, por exemplo, possua um tipo de plataforma e conexão diferente dos outros, é possível enviar um comando a partir do Blynk e controlar todos eles através da tag. Portanto, supondo que tenha um projeto com o Arduino Uno e Ethernet Shield, um segundo projeto com Arduino Mega 2560 e ESP8266 como módulo WiFi e um terceiro projeto com NodeMCU, caso reúna os três projetos em uma tag, poderá, por exemplo, com apenas um comando executar a mesma ação nas três plataformas e de forma independente.

Abaixo de *DEVICES* há a opção *AUTH TOKENS* e a mesma possui dois botões, sendo um deles o Email all (permite enviar ao e-mail todos os auth tokens dos dispositivos registrados) e Copy all (permite copiar para o clipboard (área de transferência) todos os auth tokens dos dispositivos registrados). Abaixo há a opção para escolha da cor do tema (DARK ou LIGHT), em seguida a opção KEEP SCREEN ALWAYS ON (Mantenha a tela sempre ativa) que você pode habilitar caso queira que a tela do dispositivo móvel permaneça ligada enquanto o Blynk estiver aberto. Depois há a opção SEND APP CONNECTED COMMAND que se ativada, informa quando o app Blynk é aberto e fechado no dispositivo móvel. Por último temos em *ACTIONS WITH PROJECT* as

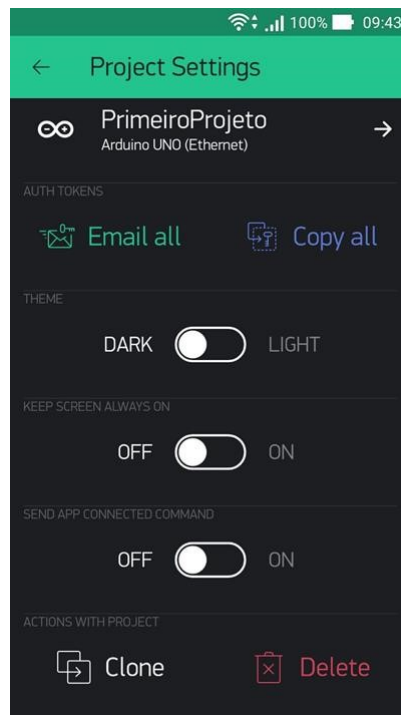


Figura 3.36: Configuração de projeto Aplicativo blynk

opções Clone que permite clonar o projeto atual e Delete que possibilita excluir o projeto atual definitivamente.

C: este botão com o símbolo de + permite que adicione widgets ao contêiner do projeto. Widgets são componentes de interface gráfica. Clique no botão e uma lista de widgets (Widget Box) será aberta (na tela principal do projeto, se deslizar a tela para a esquerda a lista de widgets também será mostrada):

Role a barra de rolagem para ver todos os widgets disponíveis. Note que eles estão divididos em categorias (CONTROLLERS / DISPLAYS / INTERFACE / NOTIFICATIONS / OTHER / SMARTPHONE SENSORS). Para adicionar um widget a tela principal do projeto basta clicar sobre o widget escolhido que o mesmo será inserido no contêiner. Para duplicar o widget basta pressioná-lo, aguardar habilitar a opção na parte superior do app e arrastar conforme mostrado na imagem abaixo. Caso queira deletar um widget, basta pressionar o mesmo e manter, que a opção vai surgir no topo da tela e então basta arrastar o widget para a opção e soltar, conforme mostrado na imagem abaixo:

Retorne a lista de widgets e veja que ao lado superior esquerdo da lista há uma bateria e acima dela está escrito YOUR ENERGY BALANCE (Seu balanço de energia):

O balanço inicial é de 2000 pontos (vamos chamar de pontos). Note que cada widget da lista possui um valor abaixo do nome. Como exemplo o widget Button : quando clicar sobre ele e o mesmo for para o contêiner de widgets (tela principal) a bateria de 2000 pontos vai cair para 1800 pontos, pois ele consome 200 pontos de energia. Logo,

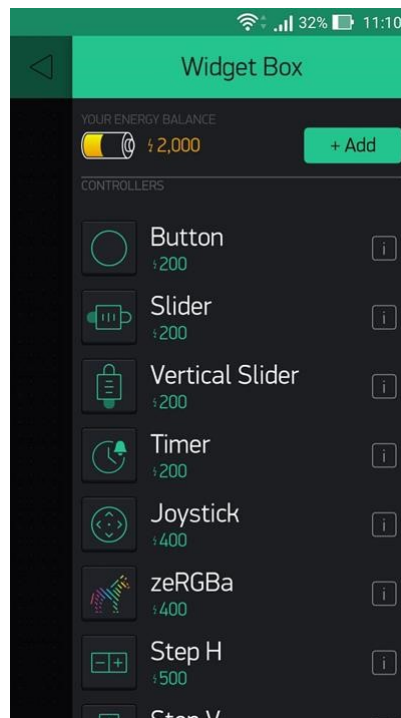


Figura 3.37: Opções de Widget Aplicativo blynk

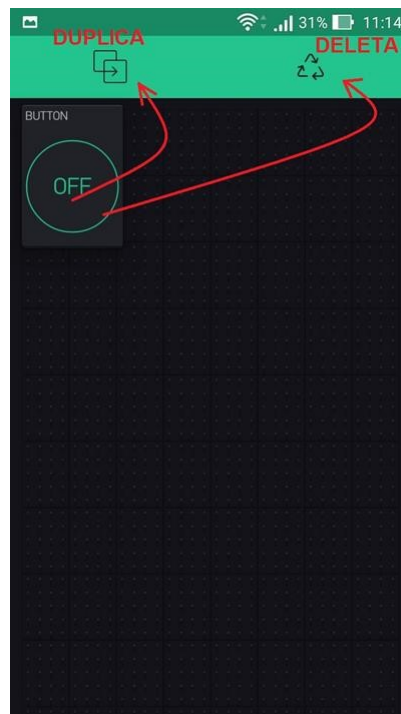


Figura 3.38: Widget Aplicativo blynk

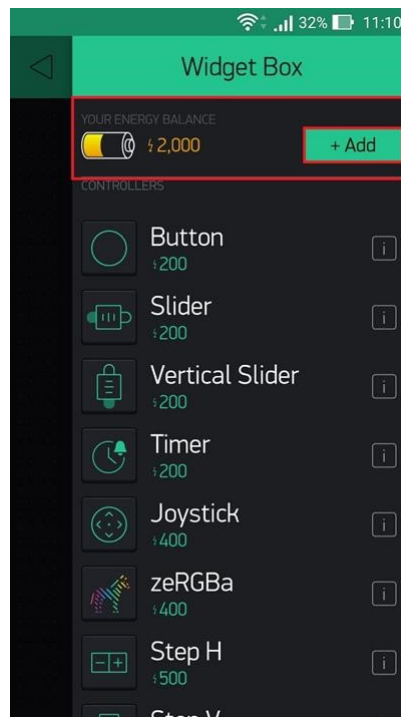


Figura 3.39: Limite Widget Aplicativo blynk

cada widget que adicionar ao contêiner irá aos poucos consumindo a energia da bateria. Retorne a lista de widgets e ao lado da bateria que tem o botão + Add. Clique sobre ele e a tela Add energy será aberta:

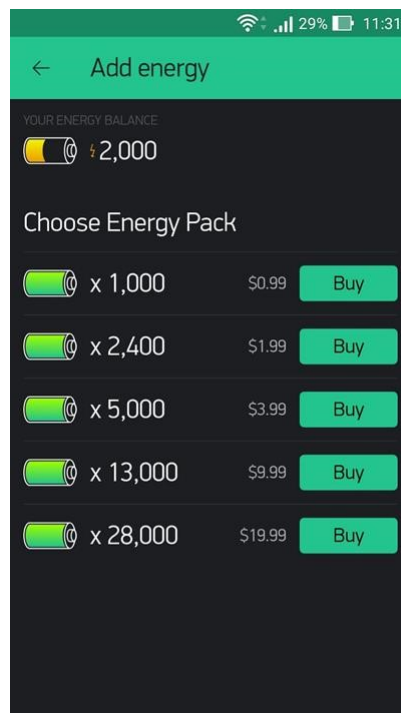


Figura 3.40: Opção compra Pontos Aplicativo blynk

basta acessar esta opção e comprar o pack.

No botão de voltar ao lado de Add energy. Na lista de widgets deslize o dedo para a direita na tela do dispositivo móvel para fechar a lista.

D: este botão com formato de Play é que vai habilitar a comunicação do aplicativo com a plataforma. Ele vai alternar o formato para Stop. Neste instante não pode editar nenhum widget que estiver na tela principal e nem poderá adicionar novos widgets. Pode ver que ao lado dele vai surgir um outro botão:

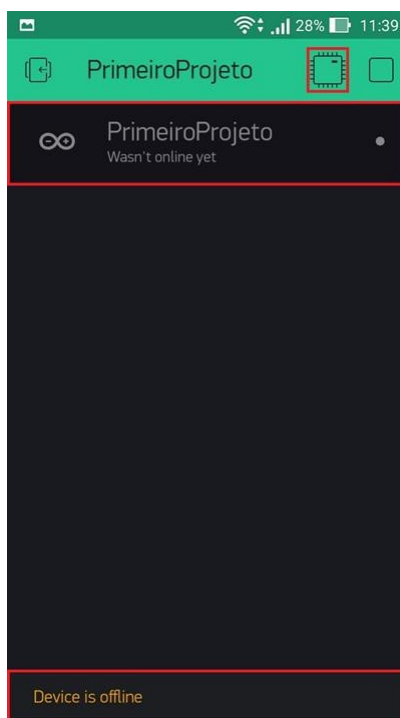


Figura 3.41: Verificar de conectividade Aplicativo blynk

No botão que surgiu, pode ver o status da conexão com a plataforma. Como ainda não temos conexão entre o app e a plataforma, a conexão estará offline e esta informação irá aparecer no rodapé do app (Device is offline). Clique sobre o botão Stop para parar a execução. Se tratando das configurações e opções que o app Blynk oferece, é basicamente o que foi mostrado nas imagens e nas explicações até aqui. Conforme mencionei anteriormente, é necessário um pacote de bibliotecas para que o Blynk possa funcionar.

3.14.5 Instalação do pacote de bibliotecas na IDE do Arduino

Para o download do pacote de bibliotecas do Blynk, deve acessar o site BLYNK (2013) e clicar na parte superior da página em GETTING STARTED e em seguida clicar em DOWNLOAD BLYNK LIBRARY:

O repositório a ser baixado deve ser a ultima versão, apos isso o arquivo.zip deve ser descompactado e copiado para a pasta do Arduino.

Após copiado os arquivos, pode conferir no próprio Arduino IDE se as pastas foram

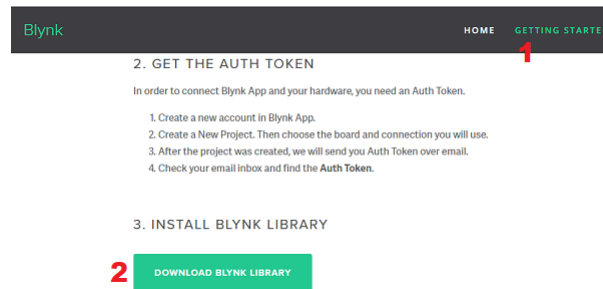


Figura 3.42: Site Blynk

colocadas em seus devidos lugares. Basta ir no menu Arquivo, Exemplos, role a lista de exemplos e veja se o Blynk está constando.

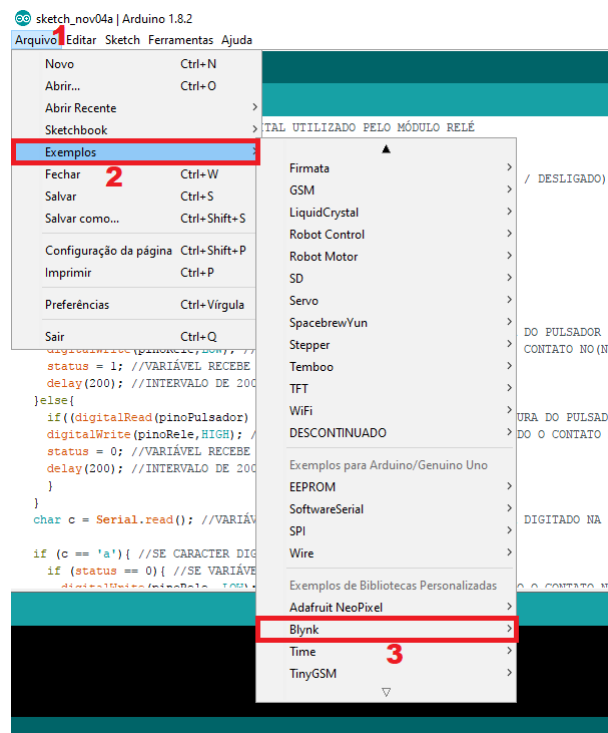


Figura 3.43: Biblioteca Blynk Arduino IDE

3.15 Controle do Sistema

A interação entre o Arduino e o ESP8266 permite que a placa possa acessar redes WiFi e com isso se torna possível desenvolver projetos bem interessantes, principalmente de automação residencial. O uso do Blynk em conjunto com o Arduino e o ESP8266 ESP-01 vai facilitar bastante a implementação do projeto de automação residencial ou IoT e vai permitir controlar o Arduino através do seu smartphone ou tablet.

A proposta para este trabalho é a integração entre o Arduino, o ESP-01 e o Blynk. O ESP-01 vai permitir que o Arduino se conecte a uma rede sem fio e se comunique

com o aplicativo Blynk instalado no dispositivo móvel. Se tratando de um projeto de automação, o Blynk possibilita controlar o motor a partir de um smartphone ou tablet que esteja conectado à internet pela rede WiFi ou até mesmo no 3G / 4G. Além disso, é possível obter informações dos sensores e demais módulos que estiverem conectados ao Arduino e mostrar estas informações na tela do dispositivo móvel.

Para que o ESP-01 possa se comunicar com o Arduino e com o Blynk, é necessário que o módulo esteja com um firmware que permita o envio de comandos AT. Nos testes que eu fiz, o firmware que funcionou perfeitamente foi a versão v0.9.5.2 AT Firmware.

O ESP-01 conectado a uma rede WiFi pode consumir até 300mA, logo, será necessário alimentá-lo com uma fonte que forneça a corrente necessária para o perfeito funcionamento. Utilizei uma fonte ajustável para protoboard, pois a mesma aceita de 6,5 a 12VDC na entrada, reduzindo a saída para 3.3VDC / 5VDC e fornece uma corrente de saída de aproximadamente 700mA. Desta forma, a fonte ajustável para protoboard será responsável por alimentar todas as peças que fazem parte do circuito, inclusive o Arduino através do pino VIN.

Observe no esquema 3.44 de ligação que o ESP-01 está sendo alimentado com a tensão de 3.3V que sai da fonte ajustável MB102, pois este é o nível lógico e a máxima alimentação que pode ser aplicada ao módulo de forma direta. Note também que, o módulo relé, o DHT11 e o Arduino estão sendo alimentados pelos 5V fornecidos na outra extremidade da protoboard. No esquema de ligação você pode ver que na fonte ajustável MB102 foi circulado em vermelho os jumpers que definem qual tensão será disponibilizada em cada lado da protoboard.

No código vá até a linha `char auth[] = YourAuthToken`; apague o texto `YourAuthToken` e adicione o seu `AuthToken`

Foi inserido o nome da rede WiFi na linha `char ssid[] = NOME DA SUA REDE WIFI`; e a senha deverá inserir na linha `char pass[] = SENHA DA SUA REDE WIFI`;

Para integrar o ESP-01 com o Arduino é necessário o uso de comunicação serial. O Arduino Uno possui apenas uma serial física e que é utilizada para comunicar a placa com o computador. Portanto, para comunicar o ESP-01 com o Arduino é necessário emular uma conexão serial através da biblioteca `SoftwareSerial` utilizando dois pinos digitais, sendo um o RX e o outro TX. A comunicação serial entre o ESP-01 e o Arduino flui sem problemas com a taxa de comunicação em 9600 bauds. No código, a linha define `ESP8266 BAUD 9600` já está definida com esta velocidade, mas o seu ESP-01 também deverá estar configurado em 9600.

No menu *Ferramentas* selecione a opção *Placa* e selecione o *Arduino/Genuino Uno*:

No meu caso a placa foi alocada na COM6, porém, em outros Arduinos podem ter sido alocado em uma COM de outro valor. Caso não saiba em qual porta COM a placa

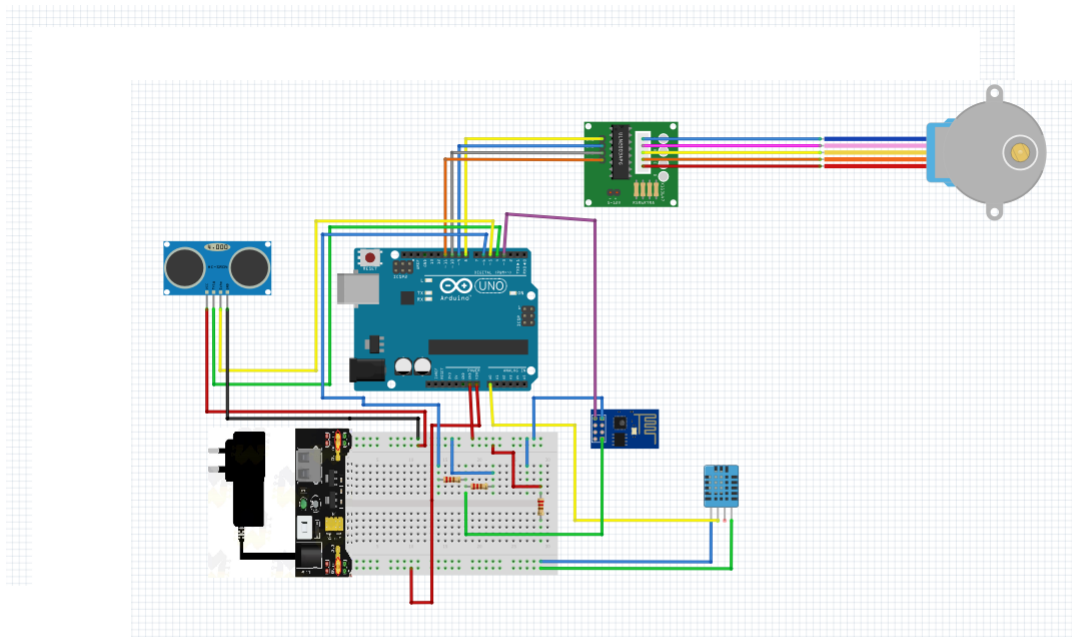


Figura 3.44: Esquemático Projeto

```

blynk_esp8266 | Arduino 1.8.5
Arquivo Editar Sketch Ferramentas Ajuda

blynk_esp8266

#include <ESP8266_Lib.h> //INCLUSÃO DE BIBLIOTECA
#include <BlynkSimpleShieldEsp8266.h> //INCLUSÃO DE BIBLIOTECA
#include <SoftwareSerial.h> //INCLUSÃO DE BIBLIOTECA
#include "dht.h" //INCLUSÃO DE BIBLIOTECA

SoftwareSerial EspSerial(10, 11); //PINOS QUE EMULAM A SERIAL (PINO 10 É O
char auth[] = "YourAuthToken"; //AUTH TOKEN (FORNECIDO PELO PROJETO NO BLY

```

Figura 3.45: Local para inserir o Token

```

blynk_esp8266 | Arduino 1.8.5
Arquivo Editar Sketch Ferramentas Ajuda

blynk_esp8266

#include <ESP8266_Lib.h> //INCLUSÃO DE BIBLIOTECA
#include <BlynkSimpleShieldEsp8266.h> //INCLUSÃO DE BIBLIOTECA
#include <SoftwareSerial.h> //INCLUSÃO DE BIBLIOTECA
#include "dht.h" //INCLUSÃO DE BIBLIOTECA

SoftwareSerial EspSerial(10, 11); //PINOS QUE EMULAM A SERIAL (PINO 10 É O
char auth[] = "YourAuthToken"; //AUTH TOKEN (FORNECIDO PELO PROJETO NO BLY

BlynkTimer timer; //OBJETO DO TIPO BlynkTimer

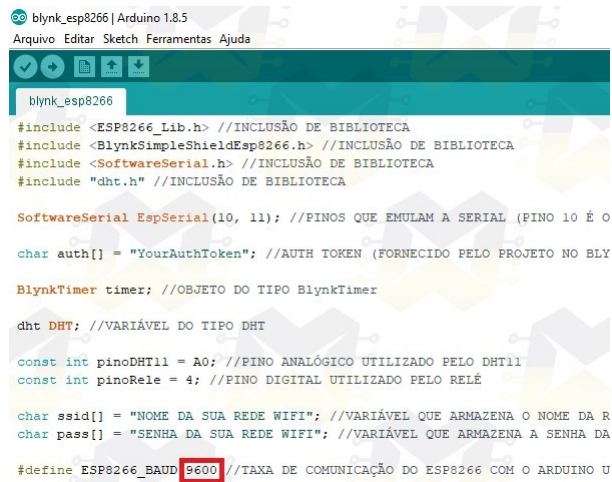
dht DHT; //VARIÁVEL DO TIPO DHT

const int pinoDHT11 = A0; //PINO ANALÓGICO UTILIZADO PELO DHT11
const int pinoRele = 4; //PINO DIGITAL UTILIZADO PELO RELÉ

char ssid[] = "NOME DA SUA REDE WIFI"; //VARIÁVEL QUE ARMAZENA O NOME DA R
char pass[] = "SENHA DA SUA REDE WIFI"; //VARIÁVEL QUE ARMAZENA A SENHA DA

```

Figura 3.46: Local para inserir nome rede e senha



```
blynk_esp8266 | Arduino 1.8.5
Arquivo Editar Sketch Ferramentas Ajuda

blynk_esp8266

#include <ESP8266_Lib.h> //INCLUSÃO DE BIBLIOTECA
#include <BlynkSimpleShieldEsp8266.h> //INCLUSÃO DE BIBLIOTECA
#include <SoftwareSerial.h> //INCLUSÃO DE BIBLIOTECA
#include "dht.h" //INCLUSÃO DE BIBLIOTECA

SoftwareSerial EspSerial(10, 11); //PINOS QUE EMULAM A SERIAL (PINO 10 É O
char auth[] = "YourAuthToken"; //AUTH TOKEN (FORNECIDO PELO PROJETO NO BLY
BlynkTimer timer; //OBJETO DO TIPO BlynkTimer

dht DHT; //VARIÁVEL DO TIPO DHT

const int pinoDHT11 = A0; //PINO ANALÓGICO UTILIZADO PELO DHT11
const int pinoRele = 4; //PINO DIGITAL UTILIZADO PELO RELÉ

char ssid[] = "NOME DA SUA REDE WIFI"; //VARIÁVEL QUE ARMAZENA O NOME DA R
char pass[] = "SENHA DA SUA REDE WIFI"; //VARIÁVEL QUE ARMAZENA A SENHA DA
#define ESP8266_BAUD 9600 //TAXA DE COMUNICAÇÃO DO ESP8266 COM O ARDUINO U
```

Figura 3.47: Configuração velocidade para 9600 bauds

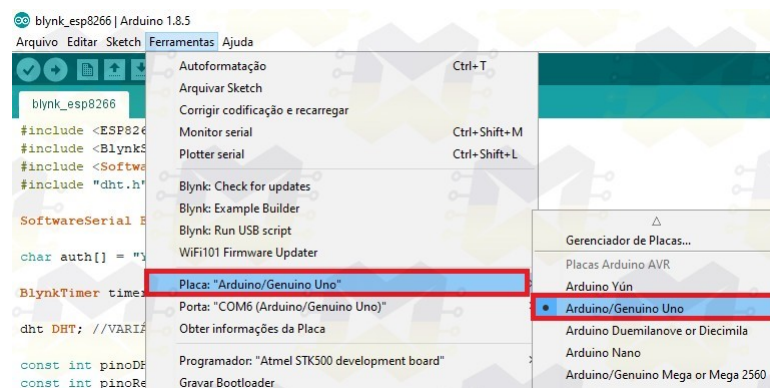


Figura 3.48: Seleção Arduino/Genuino Uno

foi alocada, basta retornar no menu Iniciar do Windows, acessar a opção Dispositivos e Impressoras , verificar a porta em que o Arduino está conectado, retornar na IDE e selecionar a porta COM. Em seguida, clique no botão para enviar o código ao Arduino e aguarde o carregamento. Terminado o carregamento do código, foi desconectado o cabo USB do Arduino, conectado a fonte de alimentação 9V na tomada e no plug jack da fonte ajustável para protoboard, e pressionado o interruptor para ligá-la. Abrindo o app Blynk no dispositivo móvel Android ou iOS, é só fazer o login e acessar o projeto.

3.16 Resultados e Discussão

Obtendo o resultado do trabalho realizado, temos então uma das primeiras dificuldades encontradas foi o fluxo de ração. O condutor que canalizava a ração estava impedindo seu fluxo, foi necessário então modificar a conexão, uma solução foi colocar uma abertura com diâmetro maior que a anterior.

O arduino jundo com o ESP se mostraram eficientes e supriram as necessidades propostas. Apresentaram uma boa velocidade de resposta. Um item muito importante utilizado foi a utilização da biblioteca *Stepper* que ajuda no controle de passos do motor de passo, melhorando assim o desempenho do hardware. Uma dificuldade encontrada foi que em toda reinicialização do arduino, perde-se o sinal do *wi-fi* por um tempo, até este reconectar novamente.

Para teste final, a ração foi coletada após uma reposição executada e feito medição do peso fornecido. O resultado foi de 54 gramas, sendo o grão da ração para animais de grande porte. Para o grão de ração para animais de pequeno porte, o resultado foi de 67 gramas, a diferença dos resultados se da pelo fato de que o espaço criado entre os grãos de ração difere um dos outros, sendo que os menores se acomodam com maior facilidade, deixando assim menos espaço vazio.

Na Figura 3.49 podemos ver a interface final do aplicativo.

- **A** Wigget de Displays - recebe os valores da umidade relativa do ar.
- **B** Wigget de Displays - recebe os valores da temperatura.
- **C** Wigget de controllers - envia sinal para despejar 50g de ração.
- **D** Wigget de controllers - envia sinal para despejar 100g de ração.
- **E** Wigget de Displays - recebe sinal quando algo se aproxima.
- **F** Wigget de Displays - recebe sinal do sensor de distancia, mostrando o nível da ração.

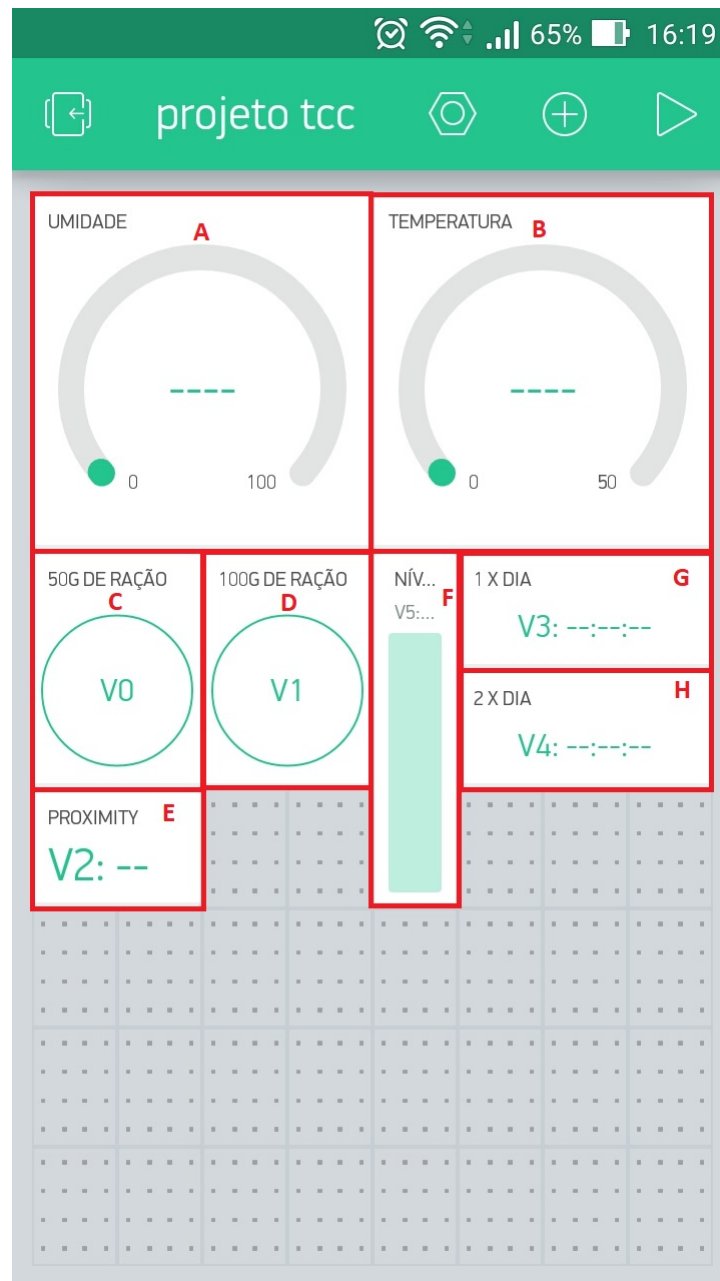


Figura 3.49: Interface do Aplicativo Blynk

- **G** Wigget de controllers - envia sinal para despejar ração no horário escolhido.
- **H** Wigget de controllers - envia sinal para despejar ração em outro horário escolhido.

Com os testes realizados, a interface pode sofrer mudanças. Novos Wiggets podem ser acionados ou retirados para que tenha assim o recebimento dos sinais e o controle necessário para concluir os o projeto.

3.17 Orçamento

Com todos os componentes listados acima, é possível calcular o custo total do projeto, apresentado na Tabela 3.11.

Por se tratar de um projeto com equipamentos de ponta, o valor do projeto ficou um pouco elevado, mas comparado com os valores de equipamentos similares disponíveis no mercado, ainda está dendo do previsto. O valor do custo do projeto será arcado por mim, já que é um investimento pessoal.

Tabela 3.11: Orçamento

COMPONENTE	PREÇO (reais)
Módulo WiFi ESP8266 ESP-01	32,90
Sensor de Umidade e Temperatura - DHT11	13,90
Protoboard de 400 Pontos	13,90
Resistor de 10K CR25 5	0,70
Galão de água	25,00
Cone	20,00
Rampa	50,00
Cabo Jumper Macho-Fêmea 20cm - KIT com 40pcs	14,90
Estrutura Mdf	120,00
Vasilha	20,00
Cabo Jumper Macho-Macho 20cm - KIT com 40pcs	14,90
Fonte Ajustável 3.3V / 5V para Protoboard MB102	13,90
Fonte DC 9V 1A Bivolt Chaveada	13,90
Arduino Uno R3 com Cabo USB A/B	59,90
Motor de Passo 28BYJ-48 com Driver ULN2003	21,90
Sensor Infravermelho Reflexivo de Obstáculo	7,90
Display LCD 16X2 Shield com Teclado	39,90
Sensor Ultrasonico HC-SR04	13,90
Parafuso de Aço para Madeira 3,5x30mm	1,00
Mancal	19,90
TOTAL	518,4

Considerações Finais

Neste capítulo, será apresentada a conclusão e proposta de trabalhos acerca do projeto desenvolvido.

4.1 Conclusões

O desenvolvimento do presente estudo e projeto possibilitou uma análise de como um desenvolver o projeto de modo a melhorar e criar uma nova maneira de que donos de animais de estimação possam criar estes. Além disso, também permitiu uma pesquisa de campo para obter dados mais consistentes sobre as etapas do processo, parte mais demorada do processo, grau de conhecimento que teve que ser buscado em vários artigos e livros.

Ainda baseado no estudo realizado, pode notar-se que o mercado de animais de estimação teve um aumento significativo podendo assim gerar ainda várias novas ideias no meio. E junto com a automação criar e com isso facilitar de maneira cada vez melhor a interação e o cuidado com os animais.

Com o estudo e a aplicabilidade da "internet das coisas", foi possível notar como a tecnologia em sistemas embarcados esta a melhorar a nossa vida, e pequenas coisas ja podem ser feitas com muio mais comodidade, como por exemplo, fazer um café ou ate mesmo abrir uma cortina. Desse modo, o mundo esta indo em direção a esta tecnologia.

Nesse sentido, a utilização de um alimentador automático permite aos usuários realizarem a ação de alimentar seus animais domésticos de forma mais rápida e eficiente. Além disso, diminui qualquer medo ou preocupação por não ter que estar presente todos os dias, podendo assim realizar outras tarefas e ate mesmo viajar sem problemas.

Capítulo **5**

Desenho Técnico

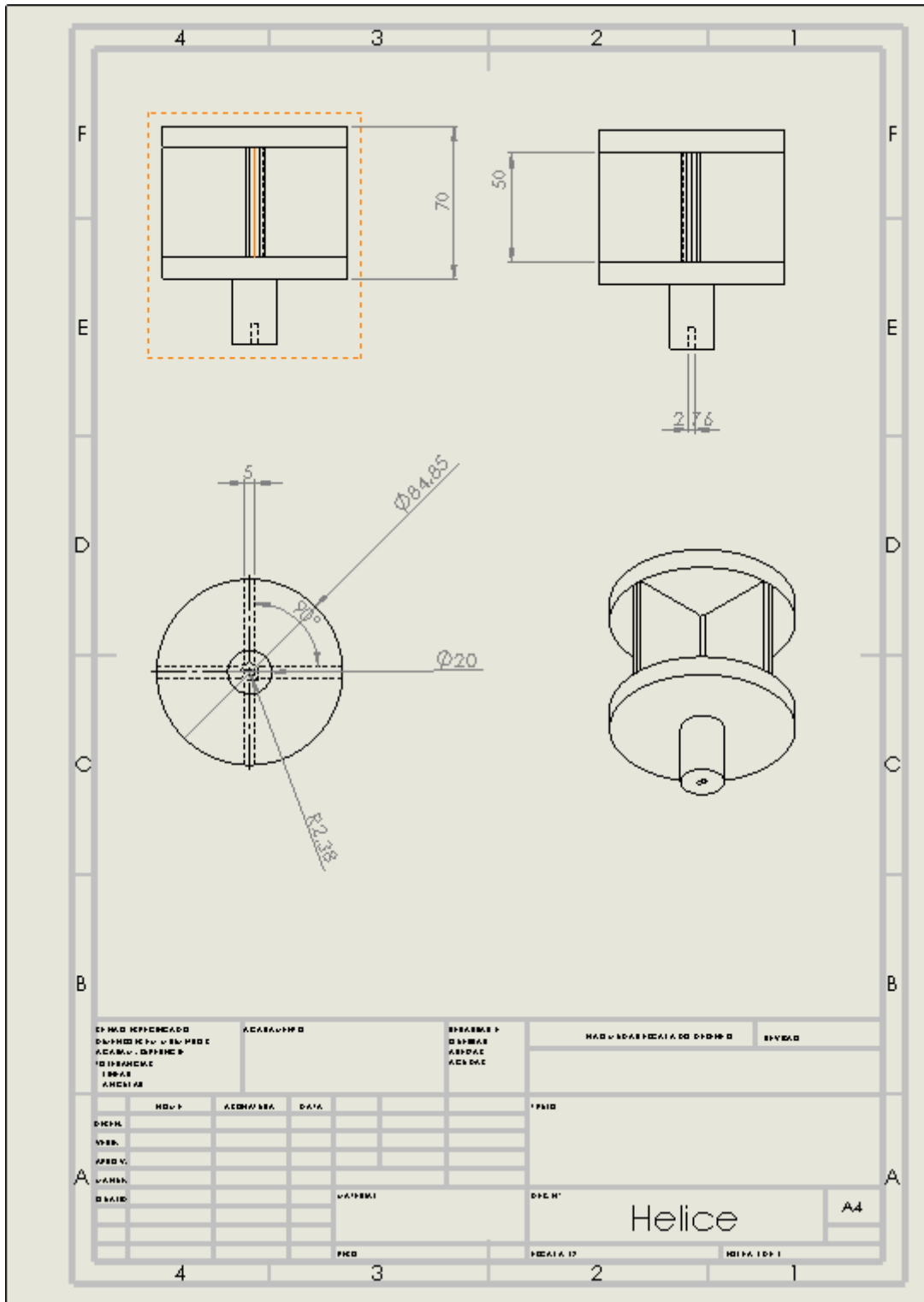


Figura 5.1: Desenho técnico Helice

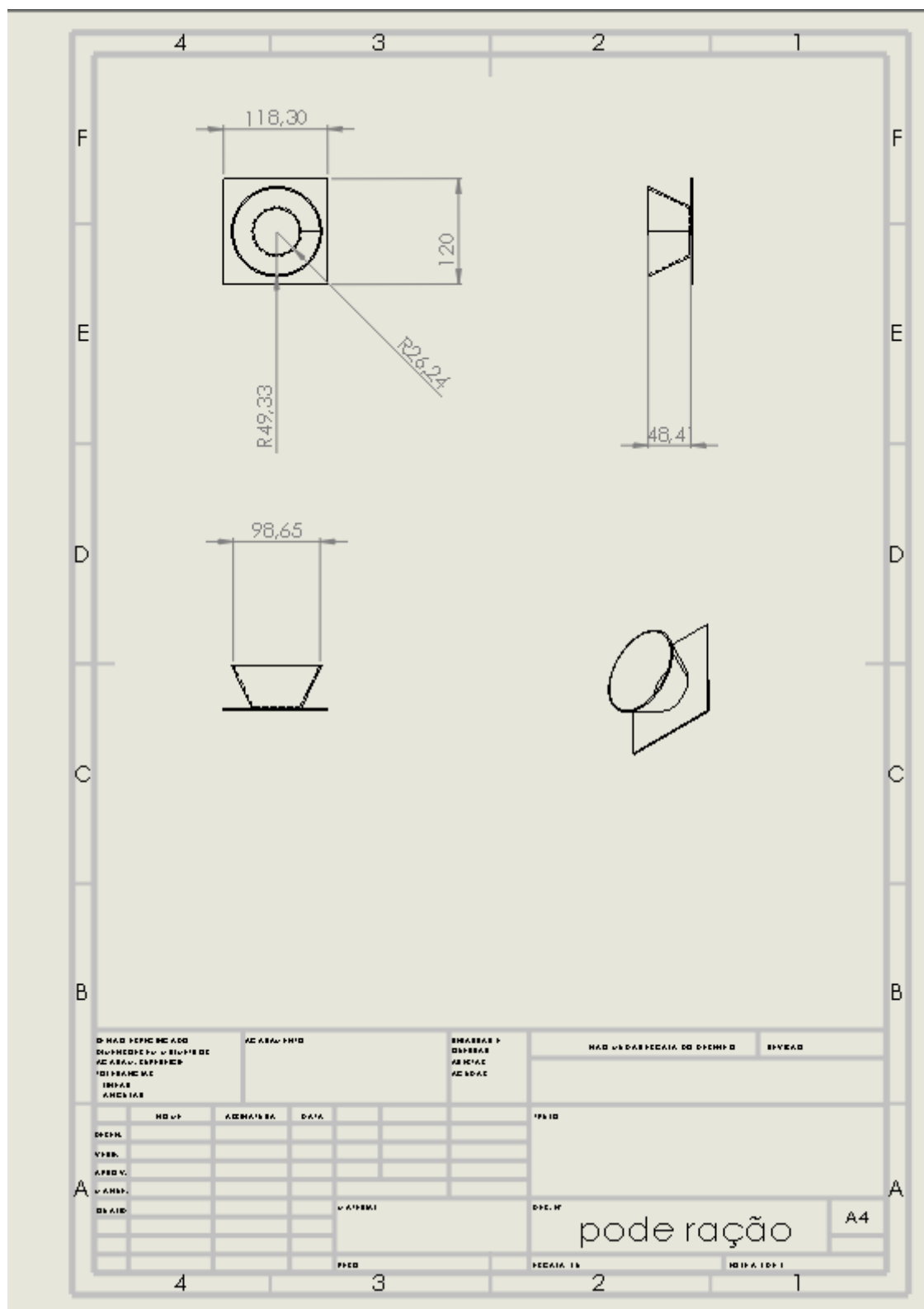


Figura 5.5: Desenho técnico do pode de ração

Capítulo 6

Programação Arduino IDE

```
#include <ESP8266_Lib.h> //INCLUSÃO DE BIBLIOTECA
#include <BlynkSimpleShieldEsp8266.h> //INCLUSÃO DE BIBLIOTECA
#include <SoftwareSerial.h> //INCLUSÃO DE BIBLIOTECA
#include "dht.h" //INCLUSÃO DE BIBLIOTECA
#include <Stepper.h>

int pinobotao_ah = V0
int leitura2 = 0;
Stepper myStepper(stepsPerRevolution, 8,10,9,11);
const int stepsPerRevolution = 500;
const int pinoSensor = 2; //PINO DIGITAL UTILIZADO PELO SENSOR
SoftwareSerial EspSerial(3, 6); // (PINO 3 É O RX E PINO 6 É O TX)

char auth[] = "YourAuthToken"; //(FORNECIDO PELO PROJETO NO BLYNK E POR E-
MAIL)

BlynkTimer timer; //OBJETO DO TIPO BlynkTimer

dht DHT; //VARIÁVEL DO TIPO DHT

const int pinoDHT11 = A0; //PINO ANALÓGICO UTILIZADO PELO DHT11
```

```

char ssid[] = "NOME DA SUA REDE WIFI"; //ARMAZENA O NOME DA REDE SEM FIO
char pass[] = "SENHA DA SUA REDE WIFI"; //ARMAZENA A SENHA DA REDE SEM FIO

#define ESP8266_BAUD 9600 //TAXA DE COMUNICAÇÃO DO ESP8266 COM O ARDUINO UNO
#define TRIGGER 4
#define ECHO 5

//Pin D1 > TRIGGER | Pin D2 > ECHO

ESP8266 wifi(&EspSerial); //PASSO OS PARÂMETROS PARA A FUNÇÃO

void setup(){
pinMode(pinoDHT11, INPUT); //DECLARA O PINO COMO ENTRADA
pinMode(TRIGGER, OUTPUT);
pinMode(ECHO, INPUT);
pinMode(BUILTIN_LED, OUTPUT);
pinMode(pinobotao_ah, OUTPUT);
pinMode(pinoSensor, INPUT); //DEFINE O PINO COMO ENTRADA

EspSerial.begin(ESP8266_BAUD); //INTERVALO DE 10 MILISSEGUNDOS)

Blynk.begin(auth, wifi, ssid, pass); //COMUNICAÇÃO BLYNK

timer.setInterval(1000L, sendUptime); //DEFINE O INTERVALO DE 1 SEGUNDO
//PARA EXECUÇÃO DA FUNÇÃO sendUptime
//Determina a velocidade inicial do motor
myStepper.setSpeed(60);

}

void sendUptime(){ //FUNÇÃO QUE OBTÉM A TEMPERATURA / UMIDADE
DHT.read11(pinoDHT11); //LÊ AS INFORMAÇÕES DO SENSOR
DHT.read11(pinoSensor); //LÊ AS INFORMAÇÕES DO SENSOR
Blynk.virtualWrite(1, DHT.temperature); //(PINO VIRTUAL 1) O VALOR DA TEMPERATURA
Blynk.virtualWrite(2, DHT.humidity); //(PINO VIRTUAL 2) O VALOR DA UMIDADE

```

```

Blynk.virtualWrite(3, DHT.distance); //(PINO VISTUAL 3) 0 OU 1
}

void loop(){
//Gira o motor no sentido ANTI-horario a 90 graus
leitura = digitalRead(pinobotao_Ah);
if (leitura2 != 0)
{
myStepper.step(512);
delay(2000);
}

long duration, distance;
digitalWrite(TRIGGER, LOW);
delayMicroseconds(2);

digitalWrite(TRIGGER, HIGH);
delayMicroseconds(10);

digitalWrite(TRIGGER, LOW);
duration = pulseIn(ECHO, HIGH);
distance = (duration/2) / 29.1;

if (distance <= 150) {
Blynk.virtualWrite(V0, 255);
}
else {
Blynk.virtualWrite(V0, 0);
}

if (distance <= 100) {
Blynk.virtualWrite(V1, 255);
}
else {
Blynk.virtualWrite(V1, 0);
}
}

```

```
if (distance <= 80) {
Blynk.virtualWrite(V2, 255);
}
else {
Blynk.virtualWrite(V2, 0);
}

if (distance <= 40) {
Blynk.virtualWrite(V3, 255);
}
else {
Blynk.virtualWrite(V3, 0);
}

if (distance <= 20) {
Blynk.virtualWrite(V4, 255);
}
else {
Blynk.virtualWrite(V4, 0);
}

Blynk.virtualWrite(V5, distance);
Blynk.run(); //INICIALIZA O BLYNK
timer.run(); //INICIALIZA O TIMER
}
```


Referências

ABINPET. <https://http://abinpet.org.br/site/mercado/> Acesso em abril de 2018.

APT. <https://www.theguardian.com/lifeandstyle/2017/feb/03/wearable-tech-for-pets#img-1> Acesso em março de 2018.

ASHTON, K. *et al.* That 'internet of things' thing. *RFID journal*, [S.l.], v.22, n.7, p.97–114, 2009.

BLYNK. <https://http://www.blynk.cc> Acesso em abril de 2018.

BRASIL, S. <https://https://www.spcbrasil.org.br/imprensa/noticia/3545> Acesso em abril de 2018.

CNDL. <https://http://site.cndl.org.br/61-dos-donos-de-animais-de-estimacao-vee> Acesso em abril de 2018.

MECÂNICA DOS MATERIAIS. 2011.

MONK, S. *Movimento Luz e Som com ARDUINO e RASPBERRY PI*. [S.l.: s.n.], 2017.

OLIVEIRA, S. de. 2017.

POLI, M. <https://exame.abril.com.br/carreira/mercado-pet-cresce-gracas-a-mudancas-no-comportamento-dos-donos-de-animais-c> Acesso em março de 2018.

RODRIGO MAXIMIANO A.ALMEIDA CARLOS HENRIQUE V.MORAIS, T. F. S. 2016.

ROSSI, A. <https://http://canaldopet.ig.com.br/curiosidades/2016-07-22/domesticacao-de-animais.html> Acesso em março de 2018.

VOLTOLINI, R. <https://https://www.tecmundo.com.br/celular/59888-conheca-primeiro-smartphone-historia-galerias.htm> Acesso em março de 2018.