

CENTRO PAULA SOUZA
ESCOLA TÉCNICA PROFESSOR MASSUYUKI KAWANO
Técnico em Redes de Computadores

Jaqueline da Silva Lombas

Liliane Maria da Silva Cardoso

Mayke Gustavo de Carvalho

Wagner Matheus da Silva Salatini

ALIMENTADOR AUTOMÁTICO PARA PET

Tupã-SP

2020

Jaqueline da Silva Lombas

Liliane Maria da Silva Cardoso

Mayke Gustavo de Carvalho

Wagner Matheus da Silva Salatini

ALIMENTADOR AUTOMÁTICO PARA PET

Trabalho de Conclusão de Curso apresentado ao Curso Técnico em Redes de Computadores da Etec PROFESSOR MASSUYUKI KAWANO, orientado pelo Prof. Caroline de Oliveira Ferraz, como requisito parcial para obtenção do título de técnico em Redes de Computadores.

Tupã-SP

2020

RESUMO

ALIMENTADOR AUTOMÁTICO PARA PET: UTILIZAÇÃO DE REDES PARA O DESENVOLVIMENTO DE ALIMENTADOR AUTOMATIZADO DE PET.

A internet tem oferecido grandes avanços positivos para a vida das pessoas nos mais variados meios, com o objetivo de facilitar, agilizar e atender as expectativas do usuário em geral. Um dos propósitos dessa avançada tecnologia também é aproximar o homem de suas atividades ou deveres diários, como por exemplo nas tarefas acadêmicas, na execução de trabalhos, nas compras, nas consultas, no controle de dados, dentre várias outras opções. E não distante a esta perspectiva, pode-se citar os cuidados aos animais à distância, controlando sua alimentação, monitorando-os por imagem, verificando a temperatura ambiente e como todo esse processo caminha para a execução de fato. O projeto auxiliará na automação da alimentação saudável e balanceada do animal de estimação, o alimentador será controlado remotamente através de um aplicativo web, que por sua vez irá fazer o trabalho de controlar e liberar o alimento. Além disso, o alimentador lhe permite através do agendamento controlar um plano de alimentação saudável para o seu animal de estimação, com base em seu comportamento alimentar e orientações de seu veterinário, que proporcionará um grande conforto deixando de lado as preocupações diárias com seu pet, e também contará com material reaproveitável e equipamentos baratos na sua produção. O trabalho como um todo visa mostrar como a automação pode ser projetada e construída de forma simples obedecendo aos conceitos de redes e segurança para o usuário, objetivo geral é facilitar a alimentação do pet a distância que auxiliará em um problema diário. O projeto utiliza o tipo de pesquisa aplicada, pois visa produzir um conhecimento que possa ser efetivamente aplicado na realidade, ajudando a alterar um problema do cotidiano das pessoas. Será realizado estudos teóricos sobre conexão usando protocolo MQTT, rede *WiFi*, técnicas na linguagem de programação C, linguagem de marcação HTML conjunto de CSS e *JavaScript* e introdução básica de eletrônica para microcontroladores, após a conclusão desses estudos será realizado os testes práticos unindo as tecnologias listadas acima e equipamentos eletrônicos que são utilizados na estrutura do projeto.

Palavras-chave: Automação; MQTT; Arduino; Alimentação; ESP32.

LISTA DE FIGURAS

Figura 1 ESP32 CAM	10
Figura 2 - Modulo FTDI.....	11
Figura 3 - Protoboard	11
Figura 4 - Arduino UNO	12
Figura 5 - Servo motor.....	13
Figura 6 - Par de LED IR e resistores.....	13
Figura 7 - Relé.....	14
Figura 8 - Bebedouro.....	15
Figura 9 - Conjunto rosca transportadora.....	15
Figura 10 - Motor de impressora	16
Figura 11 - Redutor de micro-ondas	16
Figura 12 - Modulo RFID.....	17
Figura 13 - Modulo Real time.....	18
Figura 14 - Dispensador de ração.....	18
Figura 15 - Código função publish.....	21
Figura 16 - Válvula tipo concha	22
Figura 17 - Ração impedindo o fechamento	22
Figura 18 - Extensão da concha com EVA	23
Figura 19 - Conjunto rosca transportadora.....	24
Figura 20 - Interior do reservatório.....	24
Figura 21 - Interface do aplicativo.....	25
Figura 22 - Demonstração captura de imagem.....	26
Figura 23 - Código aplicativo web	26
Figura 24 - Topologia da rede	27
Figura 25 - Código Arduino.....	28
Figura 26 - Ligações dos cabos	28
Figura 27 - Interior comedouro.....	29
Figura 28 - Servo motor e tampa	29
Figura 29 – Alimentador automático para pet.....	30
Figura 30 - Comparação de compactação.....	33
Figura 31 - Comparação MQTT e HTTP	34
Figura 32 - Comunicação múltiplas.....	35

LISTA DE ABREVIATURAS E SIGLAS

EVA	Espuma Vinílica Acetinada
FTDI	Future Technology Devices International
HTML	(HyperText Markup Language) - Linguagem de Marcação de Texto
HTTP	(Hypertext Transfer Protocol) – Protocolo de Transferência de Texto
IBM	International Business Machines
IDE Integrado	(Integrated Development Environment) - Ambiente de Desenvolvimento Integrado
IR	(Infrared) – Infravermelho
LED	LED (Light Emitting Diode) - Diodo Emissor de Luz
M2M	(Machine-to-Machine) - Máquina-para-Máquina
MDF	(Medium-Density Fiberboard) - Placa de Fibra de Média Densidade
MJPEG	Joint Photographic Experts Group
MPEG-4	Motion Picture Experts Group-4
MQTT	(Message Queuing Telemetry Transport) - Transporte de Telemetria de Enfileiramento de Mensagens
RFID	(Radio-Frequency IDentification) Identificação por Rádio Frequência
SSL	(Secure Socket Layer) - Camada Segura de Soquete
TLS	(Transport Layer Security) - Segurança da Camada de Transporte
TTL	(Transistor Transistor Logic) - Lógica transistor-transistor
USB	(Universal Serial Bus) - Porta Universal
DDNS Dinâmico	(<i>Dynamic Domain Name System</i>) - Sistema de Nome de Domínio Dinâmico

SUMÁRIO

1 INTRODUÇÃO	5
2 EMBASAMENTO TEÓRICO	7
3 METODOLOGIA.....	9
3.1 Recursos	20
4 RESULTADOS	21
4.1 Instruções técnicas.....	31
4.1.1 Configurar rede de conexão à internet.....	31
4.1.2 Alimentar seu pet.....	31
4.1.3 Capturar uma imagem	32
5 DISCUSSÃO	33
4.1 Comparação do protocolo MQTT e HTTP	34
6 CONCLUSÃO	37
REFERÊNCIAS.....	38

1 INTRODUÇÃO

A automação residencial conhecida também como domótica tem tido um papel essencial no que diz respeito às diversas tecnologias da atualidade, permitindo que uma gama diversos de usuários sejam beneficiados com as propostas da nova era tecnológica. A automação residencial consiste em um conjunto de tecnologias que ajudam na gestão e execução de tarefas domésticas cotidianas, proporcionando comodidade, conforto e o controle dos equipamentos que esteja capacitado com as tecnologias.

A rotina agitada e frenética da vida moderna do homem, como compromissos de trabalho, viagens e poucas pausas acaba afastando do seu animal de estimação, com isso é importante levantar potenciais problemas de comportamento e saúde que possam comprometer a o bem-estar dos animais. Não é recomendado que um animal tenha acesso à uma grande quantidade de alimento em uma única refeição diária, pois dificulta a correta alimentação e nutrição do animal, com a ausência do homem não é possível realizar esse controle sem um sistema automatizado.

O objetivo deste trabalho é desenvolver um protótipo de alimentador automático para pet que seja controlado a quantidade e horário da ração e o monitoramento por vídeo remotamente através de um *smartphone*, *tablet* ou computador usando recursos da internet como o protocolo MQTT para que evite que o animal coma todo alimento fornecido para ele, e com isso possibilitar um maior conforto ao homem e segurança ao animal.

Para desenvolver o projeto, propõem-se os seguintes objetivos específicos, estudar e compreender os componentes utilizados no projeto, estudar e compreender sobre protocolos de comunicação, levantamento bibliográfico de trabalhos relacionados, e produtos comerciais, aplicar conhecimento em eletrônica desenvolvendo um circuito elétrico, aplicar conhecimentos mecânicos na elaboração de uma estrutura do alimentador, aplicar conhecimento em programação para desenvolver um aplicativo web e programar microcontroladores e obter de conhecimentos necessários sobre bem-estar animal.

Observa-se que muitas pessoas encontram dificuldades ao saírem de casa, como por exemplo, quando vão trabalhar e ficam o dia todo fora da residência e o animal permanece por horas sozinho, ou em casos de uma viagem, onde o animal não tem aonde e nem com quem ficar durante um determinado período. Nota-se então que a importância desse trabalho é mostrar que é possível que a alimentação do animal seja realizada de forma gradativa, pois é essencial o cuidado com a saúde dele. Um alimentador automatizado fará com que as dificuldades encontradas, como a má alimentação ou a quantidade desregrada da ração, sejam sanadas, permitindo o dono possa alimentar seu animal no momento e na quantidade que ele julgar necessário, independentemente do local aonde ele esteja, desde que esteja conectado à internet.

2 EMBASAMENTO TEÓRICO

Os microcontroladores são dispositivos eletrônicos utilizados em praticamente todos os equipamentos eletrônicos digitais do nosso dia a dia como por exemplo, centrais de alarme, discos rígidos de computador, relógio de pulso, máquinas de lavar, forno de micro-ondas, aviões, impressoras, marca passos, carros etc. Também são muito utilizados na indústria uma das grandes vantagens é, sem dúvida a capacidade de monitoração remota de processos e máquinas, pois se torna muito mais fácil acompanhar as informações até mesmo pela internet como por exemplo nos controladores de processos, sensores inteligentes e inversores, desta forma favorecendo todos os envolvidos na atividade industrial, como investidores e colaboradores.

Microcontroladores “são circuitos integrados que possuem em seu interior todos os componentes necessários ao seu funcionamento dependendo unicamente da fonte de alimentação externa” (KERSCHBAUMER, 2019, p. 10). Ou seja, não são necessários outros componentes externos para a sua aplicação como por exemplo uma memória, pois todos os circuitos necessários já estão incluídos no mesmo, dentro de um único chip isso economiza tempo e espaço necessários para projetar um dispositivo e assim o torna um dispositivo de baixo custo.

Domótica nasceu da fusão da palavra em latim “Domus” que significa casa, com a palavra “Robótica” que se refere ao que é automatizado, isto é, realizar ações de forma automática, é uma tecnologia recente e é responsável pelo controle de recursos habitacionais do cotidiano. Capaz de proporcionar conexão e controle da sua casa e dos seus aparelhos eletrônicos estando você em qualquer lugar do mundo, basta estar conectado a internet, isso é o resultado da integração de seus aparelhos com comandos automáticos controlados na palma da sua mão por um smartphone ou computador.

A comodidade é o benefício mais evidente, os controles mais comuns e utilizados em casas são ar-condicionado, cortinas e persianas, câmeras, aquecedores, banheiras, sistemas de som, TV, luz, portas e portões de acesso, sensores de presença, irrigação de jardins e até mesmo alimentador para pets.

A domótica pode ser definida como um conjunto de tecnologias que ajudam na gestão e execução de tarefas domésticas cotidianas. A sua utilização tem

por objetivo proporcionar um maior nível de conforto, comodidade e segurança além de um menor e mais racional consumo de energia (AUTOMATICHOUSE, 2020).

O protocolo MQTT (*Message Queue Telemetry Transport*) foi inventado e desenvolvido inicialmente pela empresa de tecnologia IBM nos anos 90, a sua aplicação original era o uso na indústria de petróleo e sua função era vincular sensores a satélites, ele é um protocolo de mensagem com suporte para a comunicação assíncrona entre as partes com um modelo de publicação e assinatura, ideal para conectividade (M2M) máquina a máquina que faz o uso do protocolo TCP/IP, suas principais características são minimizar o uso de banda de rede e uso de recursos dos equipamentos enquanto garante confiabilidade e nível de garantia de entrega.

O broker é um servidor que recebe todas as mensagens dos clientes e, em seguida, roteia essas mensagens para os clientes de destino relevantes. Um cliente é qualquer coisa que possa interagir com o broker e receber mensagens (YUAN, 2020).

Sendo assim o Broker, é responsável por filtrar as mensagens e saber exatamente para quem enviar, já o publicador e assinante que são os clientes não precisam se conhecer diretamente somente precisam conhecer o broker, que é quem fará a notificação das mensagens e enviará essa informações para aqueles que tiverem inscritos no tópico referenciado.

3 METODOLOGIA

O projeto utiliza de diversos equipamentos e ferramentas para o seu desenvolvimento sendo eles:

- 1 Arduino
- 1 Esp32 CAM
- 1 Módulo RFID
- 1 Módulo real time
- 1 Módulo FTDI
- 2 Pares LED IR
- 4 Resistores
- 1 Relé
- 1 Servo motor
- 1 Conector de fonte
- 1 Motor de impressora
- 3 Parafusos
- 1 Redutor de micro-ondas
- 1 Rosca transportadora
- 1 Dispensador de ração
- 1 Comedouro
- 1 Bebedouro de pássaro
- 1 Protoboard
- 20 Cabos jumper
- 1 Botão
- 1 Fonte de alimentação
- 1 Placa de MDF
- 1 Placa de madeira
- 1 Chapa inox
- 1 Refil cola quente
- 1 Placa de plástico
- 1 Caixa de papelão
- Arduino IDE

- Visual Studio Code

O desenvolvimento do projeto se iniciou com uma pesquisa no Google em tutoriais de robótica e eletrônica sobre as características e a capacidade de cada equipamento previamente projetado para verificar se atenderia as necessidades do projeto, também foi pesquisado materiais didático em apostilas e tutoriais no Youtube de como utiliza-los, após verificação e confirmação que tais equipamento atenderia as necessidades do projeto foi realizado outra pesquisa em site de compras de eletrônicos para o orçamento de cada material a ser utilizado, e logo foi realizada a compra deles.

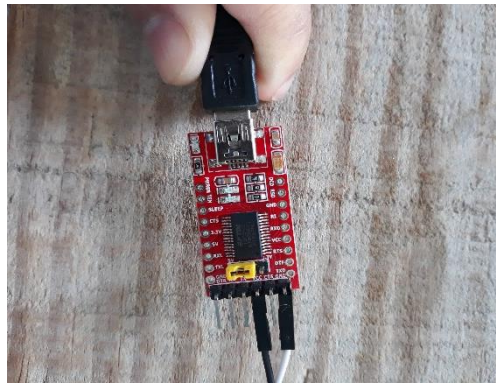
A ferramenta utilizada para o desenvolvimento da programação é o Arduino IDE, que é utilizado tanto para programar próprio Arduino como para o ESP32 ambos utiliza a linguagem C. Para programar o ESP32 foi adicionada os modelos das placas através das configurações da ferramenta, após este procedimento foi instalado o modelo específico da placa utilizada. Para fazer o upload do código compilado no Arduino IDE ao ESP foi usado o modulo FTDI que funciona como um conversor USB para serial TTL, permitindo a interface de dispositivos TTL para USB, ele é conectado à porta USB do computador e ligado por cabos jumper no microcontrolador.

Figura 1 ESP32 CAM



Fonte: Mayke Carvalho

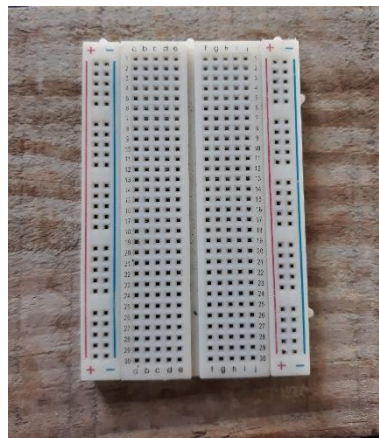
Figura 2 - Modulo FTDI



Fonte: Mayke Carvalho

Para realizar a conexão com a internet foi utilizado a WiFiManager, uma biblioteca que serve como um gerenciador de conexões wireless, pois com ela tem uma maior facilidade para configurar o ESP32 no modo Estação e Ponto de Acesso que é quando ele trabalha como um servidor para salvar as configurações de rede (nome da rede e senha) em memória não volátil para posteriormente trabalhar no modo Estação que é quando ele passa a utilizar as informações salvas e trabalhar como um cliente conectado ao roteador, esses dois modos são ativado através de um botão ligado a porta digital do ESP e fixado na caixa de proteção. Para fazer os testes práticos foi utilizado uma *protoboard*, que nada mais é uma placa de ensaio com furos (ou orifícios) e conexões condutoras utilizada para a montagem de protótipos e projetos em estado inicial, a grande vantagem da protoboard na montagem de circuitos eletrônicos é a facilidade de inserção de componentes, uma vez que não necessita soldagem.

Figura 3 - Protoboard



Fonte: Mayke Carvalho

Para a interação do usuário com o alimentador foi utilizado para a comunicação o protocolo MQTT, foi utilizado o serviço de broker gratuito da empresa HiveMQ, a comunicação do ESP32 com aplicativo web utiliza o protocolo que por sua vez faz o uso da biblioteca MQTT suas funções são as responsáveis em receber e enviar os dados ao Broker MQTT, e também se comunica com Arduino via comunicação serial para a troca de dados, como o recebimento do sensor LED IR que faz o monitoramento do nível da ração dentro do reservatório, a temperatura do ambiente, e informação sobre a presença de ração do comedouro, o ESP32 também é o equipamento responsável por fazer a captura da foto do ambiente, converter a imagem em base64, dividir em partes e enviar ao app fazendo o uso do MQTT.

O processo de upload do código no Arduino é mais simples ele não necessita de um modulo FTDI, basta conectar diretamente ao computador com um cabo USB/Serial, a programação dele também é mais simples ele fica responsável por pegar os dados enviado pelo ESP32 tais como, a hora para o agendamento e quantidade da ração a ser liberada, processar e controlar os motores e módulos, para isso foi usado uma conexão serial pelos pinos Rx e Tx de cada microcontrolador utilizando os cabos jumper.

Figura 4 - Arduino UNO



Fonte: Mayke Carvalho

Na parte eletromecânica foi usado um servo motor, são atuadores projetados para aplicações onde é necessário fazer o controle de movimento com posicionamento de alta precisão, reversão rápida e de alto desempenho, para seu funcionamento foi utilizado a biblioteca Servo na ferramenta Arduino IDE. Um dos motores é o responsável por movimentar a tampa de fechamento do dispensador de ração, juntamente conectado ao Arduino que controla o tempo em que o motor

permanece ligado, desta forma fazendo a dosagem de alimento escolhida pelo usuário, segundo motor é utilizado para movimentar a tampa do comedouro que faz parte do sistema RFID citado logo abaixo e ambos são controlados pelo Arduino.

Figura 5 - Servo motor



Fonte: Mayke Carvalho

Os sensores de LED IR são os componentes responsável para monitorar a quantidade de ração disponível dentro o reservatório do dispensador, e monitorar se a presença de ração dentro do comedouro. Para fazer uma melhor proteção para os LED foi utilizado tampas de garrafa pet para com furo central para fixá-los e para seu funcionamento foi utilizador resistores ligados em serie com a saída digital do Arduino.

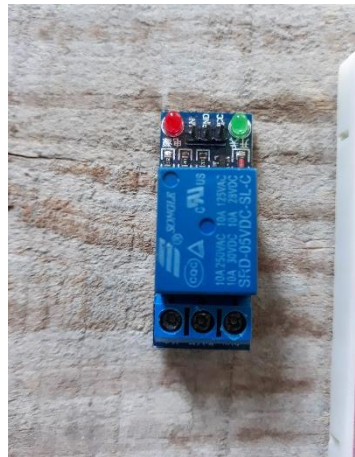
Figura 6 - Par de LED IR e resistores



Fonte: Mayke Carvalho

O relé é um interruptor eletromecânico, ou seja, seu acionamento ocorre quando a corrente elétrica percorre a bobina do relé criando um campo eletromagnético (igual a um ímã) que por sua vez atrai a alavanca responsável pela mudança do estado dos contatos, abrindo ou fechando. Ele foi utilizado para ligar e desligar o motor de impressora, pois se o motor fosse ligado diretamente nas saídas do Arduino correria um grande risco de danificá-lo pois a corrente do motor é superior a suportada pelo microcontrolador, desta forma o relé passou a ser o intermediador entre eles sendo conectado uns aos outros.

Figura 7 - Relé



Fonte: Mayke Carvalho

Para fixar o Arduino, a *protoboard*, o relé, e o conector da fonte de alimentação foi utilizado um pedaço de taboa de madeira maciça e para fazer a tampa do comedouro foi utilizado placa de madeira MDF 3mm e para fazer o forro de proteção dos cabos elétricos foi utilizado placas de papelão retirado de uma caixa.

Foi utilizado um bebedouro de pássaro reaproveitado para fazer uma caixa de proteção para o ESP32 e para poder fixá-lo melhor ao dispensador de ração, e para o pet se alimentar foi utilizado um comedouro de plástico.

Figura 8 - Bebedouro



Fonte: Mayke Carvalho

Para dispensar a ração que está no reservatório para o comedouro foi utilizado uma rosca transportadora feito em alumínio fundido, que é um mecanismo com um eixo girante na forma de uma rosca de parafuso para mover material de um ponto ao outro.

Figura 9 - Conjunto rosca transportadora



Fonte: Mayke Carvalho

E para movimentar a rosca foi utilizado um motor retirado de uma impressora inutilizada e fixado com resina plástica a um redutor do prato forno do micro-ondas que foi retirado de sucata, o redutor de velocidade é um conjunto de engrenagens que trabalham dentro de uma caixa metálica com objetivo de transmitir potência da entrada e mudar a velocidade de rotação em sua saída, com isso o

resultado final é uma velocidade reduzida, porém devido às leis da física, quando há redução da rotação, aumenta-se o torque disponível, ou seja aumenta-se a sua força.

Figura 10 - Motor de impressora



Fonte: Mayke Carvalho

Na figura 11 é possível visualizar o conjunto de engrenagens que forma o sistema de redução, elas são interligadas uma a outra desta forma transmitindo o movimento gerado pelo motor.

Figura 11 - Redutor de micro-ondas



Fonte: Mayke Carvalho

Foi utilizado um módulo RFID (*Radio-Frequency IDentification*, Identificação por Rádio Frequência) o sistema de comunicação para transferência de dados do sistema RFID utiliza ondas de rádio a uma frequência de 13,56 MHz, e uma etiqueta (tag) que é composta basicamente de uma antena, responsável por captar informações, e um microchip, que é capaz de armazenar dados quando for

aproximada do modulo. Nesse projeto ele tem a função de fazer a leitura da tag verificar a identificação para liberar a abertura da tampa do comedouro, ele foi conectado nos pinos digitais do Arduino, utilizando cabos jumper.

Figura 12 - Modulo RFID



Fonte: Mayke Carvalho

O projeto conta com sistema de segurança caso perca conexão com a internet e para controlar os horários da alimentação que foi agendada pelo proprietário do animal e para este fim foi utilizado o modulo *Real Time Clock* RTC DS1307 é um relógio de tempo real com calendário completo sendo capaz de fornecer informações como segundo, minutos, dia, data, mês e ano. Pois sem ele o Arduino não teria a capacidade de armazenar por um grande período o decorrer dos milissegundos em sua memória e qualquer falha de energia perderia essa contagem, já com o *Real Time Clock* pode ser programada para executar ações específicas e em data e horário específico, este modulo foi ligado ao Arduino com cabos jumper. A alimentação de todos os componentes é mantida por uma fonte de alimentação com capacidade de suprir a demanda de energia consumida pelo circuito.

Figura 13 - Modulo Real time



Fonte: Mayke Carvalho

Para o alimento ficar armazenado foi utilizado um dispensador de ração disponível no mercado, que tem a capacidade de armazenar alimento para um animal de pequeno porte por mais de uma semana, ele conta com uma tampa que mantém o sistema de reservatório totalmente fechado, para proteger o alimento de outros animais que não devem se alimentar com ração que ele armazena e também proteger de insetos e roedores.

Figura 14 - Dispensador de ração



Fonte: Mayke Carvalho

Para usar e gerenciar todos recursos presente nesse projeto foi desenvolvido um aplicativo *web* que utiliza uma tecnologia fundamental, a *WebSockets* é uma tecnologia avançada que torna possível abrir uma sessão de comunicação interativa entre o navegador do usuário e um servidor ela tem a capacidade de enviar mensagens para um servidor e receber respostas sem ter que

consultar o servidor para obtê-la. Outra tecnologia adotada foi a Paho JavaScript Client é uma biblioteca JavaScript de clientes MQTT baseada no navegador, que usa *WebSockets* para conectar-se a um MQTT *Broker*, foi utilizado a linguagem HTML e JavaScript junto as tecnologias citadas para o desenvolvimento do aplicativo utilizando a ferramenta VS Code como ambiente de produção.

3.1 Recursos

No desenvolvimento do projeto foi dividido as tarefas para cada integrante do grupo, a escolha foi baseada de acordo com o conhecimento pessoal de cada membro. O tempo total gasto para chegar ao resultado do projeto foi de 6 meses de trabalho.

Foi utilizado de diversos recursos nos quais são os seguintes materiais listado abaixo.

Despesas e investimento do projeto				
Item	Quantidade	Preço	Total	Fonte Recurso
Arduino	1	R\$ 35,00	R\$ 35,00	Recurso do grupo
ESP32	1	R\$ 70,00	R\$ 70,00	Recurso do grupo
Modulo RFID	1	R\$ 18,00	R\$ 18,00	Recurso do grupo
Modulo Real Time	1	R\$ 10,00	R\$ 10,00	Recurso do grupo
Sensor LED IR	2	R\$ 2,00	R\$ 4,00	Recurso do grupo
Servo motor	1	R\$ 13,00	R\$ 13,00	Recurso do grupo
Dispensador de ração	1	R\$ 120,00	R\$ 120,00	Doação
Protoboard	1	R\$ 12,00	R\$ 12,00	Recurso do grupo
Kit cabos jumper	1	R\$ 10,00	R\$ 10,00	Recurso do grupo
Botão	1	R\$ 0,40	R\$ 0,40	Recurso do grupo
Fonte alimentação reciclado	2	R\$ 0,00	R\$ 0,00	Recurso do grupo
Redutor de micro-ondas reciclado	1	R\$ 0,00	R\$ 0,00	Doação
Rosca Transportadora	1	R\$ 40,00	R\$ 40,00	Recurso do grupo
Motor de impressora reciclado	1	R\$ 0,00	R\$ 0,00	Recurso do grupo
Comedouro reciclado	1	R\$ 0,00	R\$ 0,00	Recurso do grupo
Bebedouro reciclado	1	R\$ 0,00	R\$ 0,00	Recurso do grupo
Placa madeira MDF reciclado	1	R\$ 0,00	R\$ 0,00	Recurso do grupo
Taboa de madeira reciclado	1	R\$ 0,00	R\$ 0,00	Recurso do grupo
Tampa de garrafa reciclado	2	R\$ 0,00	R\$ 0,00	Recurso do grupo
Parafuso	3	R\$ 0,05	R\$ 0,15	Recurso do grupo
Placa plástico reciclado	1	R\$ 0,00	R\$ 0,00	Recurso do grupo
Chapa inox reciclado	1	R\$ 0,00	R\$ 0,00	Recurso do grupo
Caixa de papelão reciclado	1	R\$ 0,00	R\$ 0,00	Recurso do grupo
Relé	1	R\$ 11,00	R\$ 11,00	Recurso do grupo
Refil cola quente	1	R\$ 1,00	R\$ 1,00	Recurso do grupo
Conector fonte	1	R\$ 5,00	R\$ 5,00	Recurso do grupo
Total			R\$ 349,55	

Figura 16 - Válvula tipo concha



Fonte: Mayke Carvalho

Ela funcionou porem teve um problema quando ao fechar algumas vezes a válvula prendia um grão de ração impedindo o fechamento completo do sistema causando uma grande sobrecarga no redutor e motor ao tentar fechar completamente, com isso há grandes chances de todo sistema ser danificado.

Figura 17 - Ração impedindo o fechamento



Fonte: Mayke Carvalho

Para resolver este problema foi utilizado uma fita de EVA (Espuma vinílica acetinada) 2mm colada na borda da concha para estender seu tamanho que ao fechar mesmo que um grão de ração fique preso, pôr o EVA ser uma material flexível ele sofre uma deformação e não impede o fechamento completo da válvula e conseqüentemente não força o redutor e motor.

Figura 18 - Extensão da concha com EVA



Fonte: Mayke Carvalho

Outro problema surgiu em relação a segurança, quando a uma interrupção de energia inesperada no alimentador, a válvula aberta e com a falta de energia não pode se mover para fechar liberando todo fluxo de ração até que o prato do comedouro esteja totalmente completo não tendo o devido controle esperado além de travar totalmente a válvula pois os grãos de ração impede o movimento, danificando todo sistema de controle.

Para solucionar esse problema foi optado por mudar o sistema de controle de ração desta vez foi utilizado o sistema de rosca transportado, um sistema eficiente, mais seguro quanto a interrupções de energia pois na falta ele simplesmente para o fluxo de ração sem causar maiores problemas, e volta seu processo normal quando a energia for estabelecida, também reduziu o número de equipamentos eletrônicos como sensores óticos para detectar a posição da concha e módulo ponte H para inverter o sentido de rotação do motor.

Figura 19 - Conjunto rosca transportadora



Fonte: Mayke Carvalho

Na figura abaixo é possível observar o motor fixado no reservatório e o par de sensor IR e uma placa de papelão no fundo para proteger os cabos elétricos.

Figura 20 - Interior do reservatório



Fonte: Mayke Carvalho

Aplicativo web apesar de uma interface muito simples atendeu perfeitamente as necessidades do projeto ele foi o responsável pela interação com usuário com o alimentador, ele foi capaz de conectar ao broker MQTT, mostrar as opções de porção de alimento, receber um alerta sobre o nível do reservatório, um alerta sobre a presença do alimento no comedouro para saber se o animal já se alimentou e fazer uma captura de imagem.

Figura 21 - Interface do aplicativo



Fonte: Mayke Carvalho

A próxima figura demonstra uma captura de imagem realizada pela câmera do ESP32 e enviada e visualizada no aplicativo web.

Figura 22 - Demonstração captura de imagem



Fonte: Mayke Carvalho

Na figura 23 mostra algumas funções da biblioteca Paho responsável por fazer receber mensagens do servidor broker, identificar seu destino e processar de acordo com seu conteúdo recebido.

Figura 23 - Código aplicativo web

```

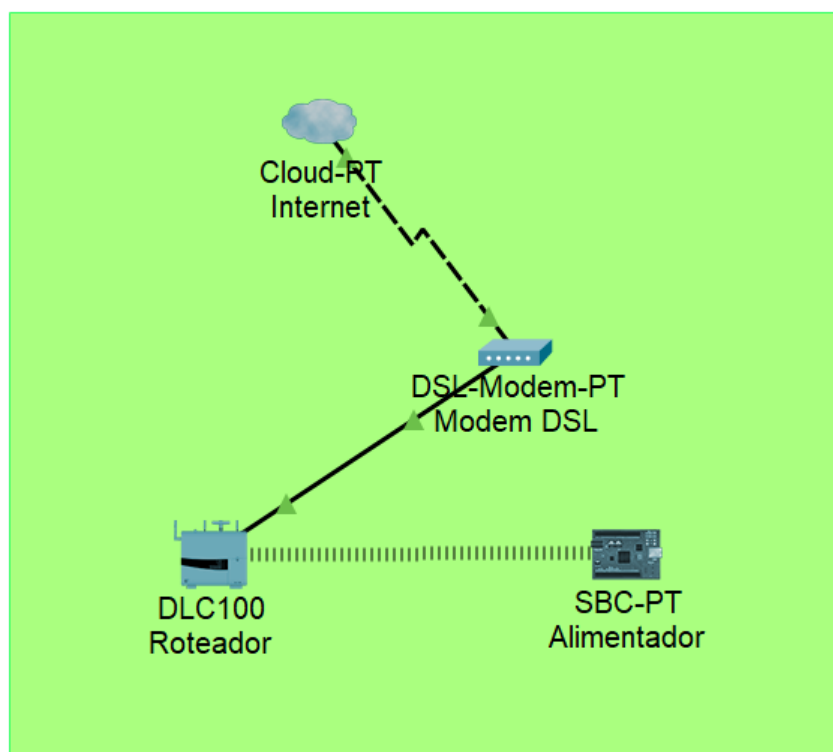
276
277     if(r_message.destinationName == "mk_nivel"){
278
279         v_racao = r_message.payloadString;
280
281         if(v_racao == 1)
282             alert("OPS! A ração está acabando");
283
284         console.log("nivel >>> "+v_racao);
285     }
286
287
288     if(r_message.destinationName == "mk_envia"){
289
290         foto = foto + r_message.payloadString;
291         console.log("img >>> "+foto);
292         console.log("comp: "+foto.length);
293
294         document.getElementById('preview').setAttribute('src', foto);
295         $("#preview").show();
296
297
298     }
299
300     if(r_message.destinationName == "mk_temp"){
301
302         console.log("temp >>> "+r_message.payloadString);
303         document.getElementById("viewTmp").innerHTML = r_message.payloadString + " °C";
304

```

Fonte: Mayke Carvalho

Seguindo com os testes o ESP32 obteve sucesso, suas funcionalidades ocorreram dentro do esperado, ou seja, conseguiu se conectar ao roteador doméstico tendo acesso à internet possibilitando sua comunicação com o broker MQTT, então passou receber e enviar dados como a imagem capturada pela câmera, os dados do nível de ração, a temperatura do ambiente e dados do sensor do comedouro e receber dados da quantidade de ração, e a opção de horário para o agendamento, na parte de comunicação serial entre o ESP e Arduino também funcionou com sucesso.

Figura 24 - Topologia da rede



Fonte: Mayke Carvalho

Como citado acima o Arduino conseguiu fazer a leitura do sensor reflexivo responsável pelo nível de ração, sensor do comedouro e temperatura ligado ao Arduino processa-las e repassá-la ao ESP via comunicação serial, com este mesmo método foi capaz de receber dados do ESP com a quantidade de ração processar e ativar/desativar o motor da rosca transportadora para que o alimento seja liberado do reservatório ao comedouro com a devida quantidade escolhida, e receber também a opção de horário.

Figura 25 - Código Arduino

```

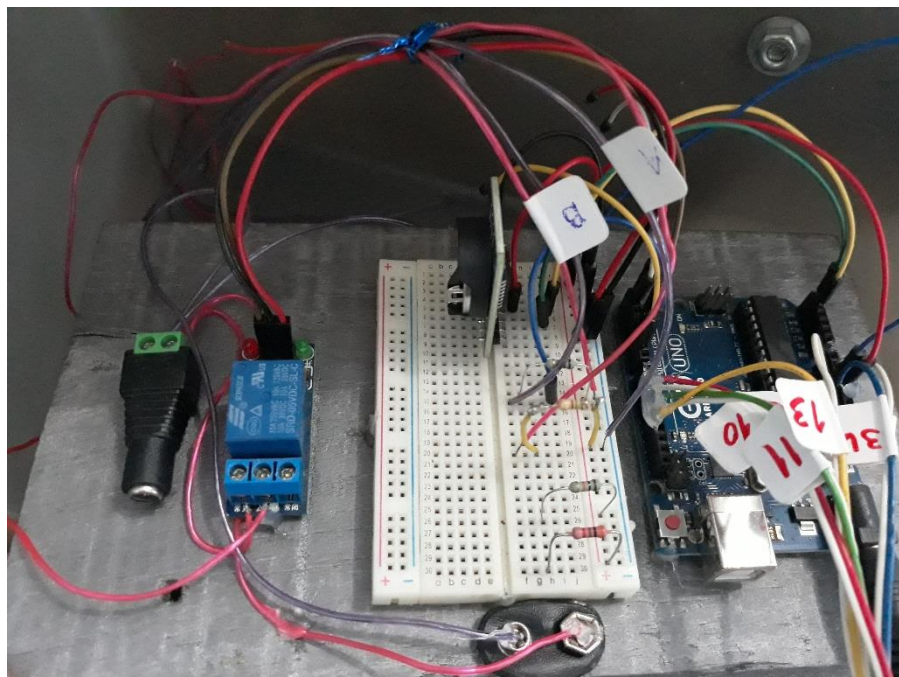
123
124 String conteudo = Serial.readString();
125 delay(100);
126
127
128 char orig = conteudo.charAt(0);
129
130 if(orig == 'x'){
131     estadoComed();
132     delay(1000);
133     temperatura();
134 }
135
136 if(orig == 'r'){
137     conteudo.remove(0,1);
138     numCtd = conteudo.toInt();
139
140     alimentar(numCtd);
141 }
142
143 if(orig == 'h'){
144     String r = conteudo.substring(2);
145
146     conteudo.remove(0,1);
147     numCtd = conteudo.toInt();
148     EEPROM.write(0,numCtd);
149
150     //qtd racao
151     r.remove(0,1);
152     numCtd = r.toInt();
153     EEPROM.write(1,numCtd);
154 }
155
156

```

Fonte: Mayke Carvalho

Na figura abaixo é possível ver alguns detalhes das ligações elétricas dos componentes utilizado para realizar os serviços descritos neste projeto.

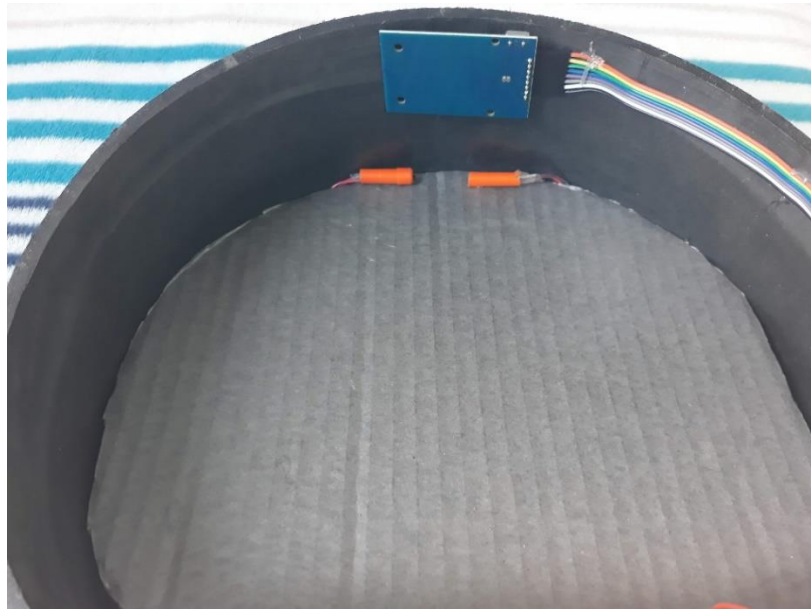
Figura 26 - Ligações dos cabos



Fonte: Mayke Carvalho

Na parte interna do comedouro o módulo RFID fixado e o par de sensor IR, que também conta com uma placa de papelão para proteção dos cabos elétricos.

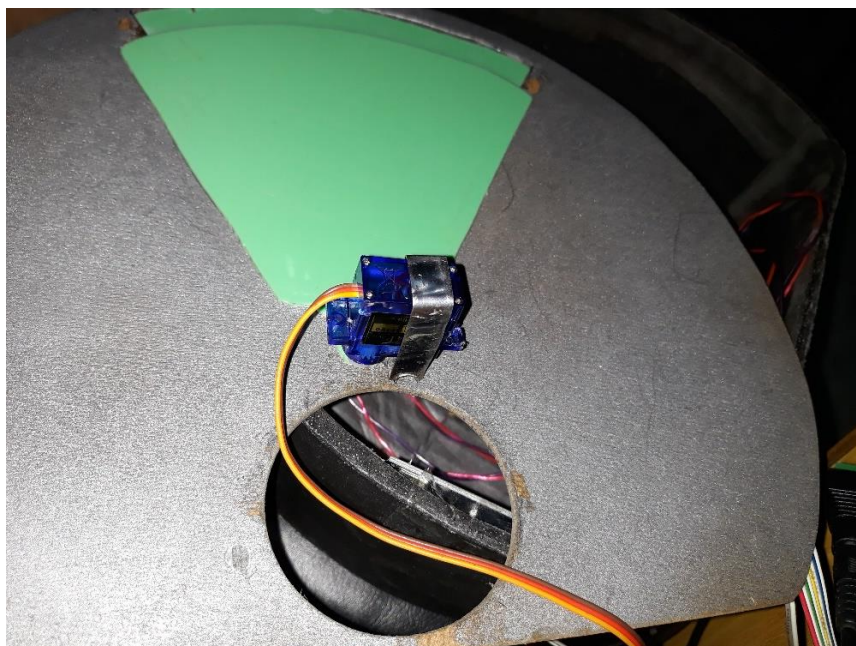
Figura 27 - Interior comedouro



Fonte: Mayke Carvalho

O servo motor fixado com uma pequena chapa de aço inox junto a placa MDF, e a tampa plástica fixada em seu eixo como mostra a figura 28, ambos formam a tampa do comedouro.

Figura 28 - Servo motor e tampa



Fonte: Mayke Carvalho

Na figura 29 é possível observar a presença de todas as partes da estrutura do alimentador utilizada para realizar este projeto, bem como o dispensador de ração, na parte inferior o comedouro e na parte superior o ESP32 fixado no topo.

Figura 29 – Alimentador automático para pet



Fonte: Mayke Carvalho

4.1 Instruções técnicas

4.1.1 Configurar rede de conexão à internet

Para configurar uma nova conexão realiza as seguintes etapas abaixo.

1. Ligue o alimentador na rede elétrica 127/220V.
2. Mantenha o botão reset pressionado por 2 segundos. Após esse procedimento o alimentador irá entrar em modo *Access Point* ficando visível para que seja possível se conectar a ele.
3. Realiza uma busca por redes *WiFi* disponível em seu smartphone, tablet ou computador, após encontrar a ALIMENTE conecte a ela.
4. Abra seu navegador de internet (browser) e na barra de endereços digite o seguinte IP: 192.168.4.1 e pressione *enter*.
5. Após carregar a página escolha a opção *Configure WiFi*. Esta etapa é responsável por exibir as redes disponíveis onde o alimentador vai se conectar para ter acesso a internet.
6. Nesse instante será listado as redes disponíveis, escolha a rede que seja conectar em seguida digite a senha da rede escolhida e clique e salvar.

Feito todos os procedimentos anteriores sua rede agora está salva, o alimentador já se conecta automaticamente a ela, caso não tenha obtido sucesso com a conexão revise as informações de rede escolhida e repita todo procedimento iniciando pela etapa 2.

4.1.2 Alimentar seu pet

Para alimentar seu pet siga as etapas abaixo.

1. Com o alimentador corretamente configurado e conectado à internet e o reservatório de ração do alimentador abastecido, abra o aplicativo Web e clique em conectar, será enviado um alerta sobre a situação do comedouro. Se houver problema revise as configurações de rede.
2. Escolha uma das opções de quantidade do alimento.
3. Após escolha clique em alimentar, se não houver nenhum problema em alguns segundos o alimentador iniciara o procedimento.

Se o nível do reservatório estiver abaixo de 30% do seu volume um alerta será enviado, abasteça com alimento o quanto antes.

4.1.3 Capturar uma imagem

Pode se tirar fotos do ambiente para um monitoramento visual para isso realiza as seguintes etapas.

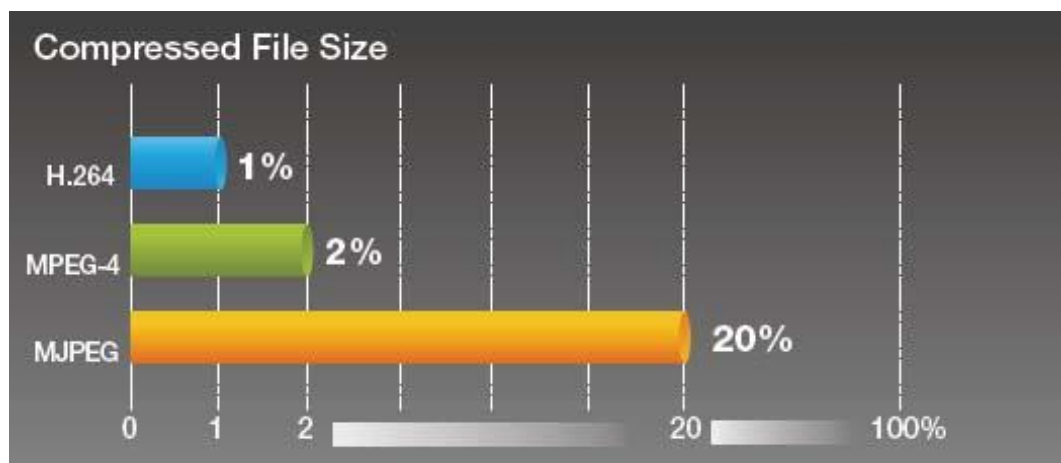
1. Com o alimentador corretamente configurado e conectado à internet abra o Aplicativo Web e clique em conectar e aguarde a mensagem. Se houver problema revise as configurações de rede.
2. Após a conexão bem sucedida clique em capturar e em alguns segundos a imagem será exibida. Para capturar outra imagem repita a etapa 2.

5 DISCUSSÃO

O MJPEG é um padrão simples de compactação de vídeo que comprime cada quadro, ou seja, cada foto antes de emitir a transmissão de vídeo, seu processamento é relativamente leve para o hardware porém seu desempenho permite menor compactação do que outros padrões como H.264 e MPEG-4, aumentando drasticamente o tamanho do arquivo ou a largura de banda durante o *streaming*.

Abaixo o gráfico apresentado mostra uma comparação com as três tecnologias realizando o trabalho de compactação com um arquivo do mesmo tamanho.

Figura 30 - Comparação de compactação



Fonte: (Likan, 2020)

Comparado a outra tecnologia de compactação, o H.264 é o mais recente algoritmo de compactação de vídeo, ele pode transmitir imagens de alta qualidade com tamanho de arquivo pequeno, reduzindo a 1% do seu tamanho original, portanto o *streaming* de vídeo com o H.264 ocupa menos largura de banda, o MPEG-4 também obtém um bom resultado reduzindo a 2% do seu tamanho.

Já o MJPEG fica bem atrás em comparação com as outras duas ele reduz a 20% do tamanho original do arquivo e por esse motivo o recurso de vídeo do ESP32 não funcionou bem em um ambiente com largura de banda restrita como no caso da rede móvel 3G ou 4G, no teste o vídeo apresentou com travamentos, ou ausência, só teve um melhor resultado quando a resolução da câmera foi colocada no

mínimo, mas a qualidade não foi satisfatória com pouca definição e mesmo assim não era sempre disponível acessar serviço de vídeo na rede móvel.

4.1 Comparação do protocolo MQTT e HTTP

A tecnologia usada no projeto para a comunicação foi bem sucedida apresentando bom resultado tanto para rede WiFi quando para uma rede móvel, isso graças ao protocolo MQTT, fácil de implementar, leve e eficiente pois não é necessários muito processamento e muita banda nem para que a comunicação ocorra.

Comparando o MQTT com o tradicional protocolo HTTP(*Hypertext Transfer Protocol*) Protocolo de Transferência de Hipertexto também usado para comunicação há várias vantagens, três ficam muito claras, baixa necessidade de processamento, o baixo consumo de memória para o envio de mensagem, isso é um ponto forte quando se trata de *hardware* e o baixo consumo de banda.

A tabela abaixo demonstra o resultado dos dois protocolos usados em diferente tipo de rede, na rede móvel 3G e rede *WiFi*.

Figura 31 - Comparação MQTT e HTTP

	3G		Wifi	
	HTTPS	MQTT	HTTPS	MQTT
% Battery / Hour	18.43%	16.13%	3.45%	4.23%
Messages / Hour	1708	160278	3628	263314
% Battery / Message	0.01709	0.00010	0.00095	0.00002
Messages Received	240 / 1024	1024 / 1024	524 / 1024	1024 / 1024

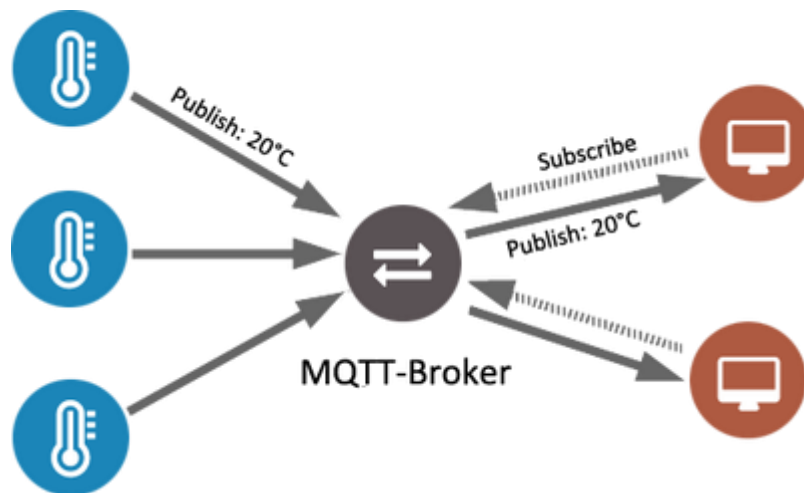
Fonte: (Nicholas, 2020)

O MQTT usa menos energia da bateria dos dispositivos, tendo a capacidade de enviar mais mensagens por hora, e sendo mais confiável que o HTTPS pois recebe todas as mensagens. O MQTT usa mais energia da bateria para iniciar uma conexão 4,23% usando o WiFi comparado aos 3,45% do HTTP, mas isso

é rapidamente compensado pelos ganhos à medida que a conexão permanece aberta por mais tempo.

MQTT tem um cabeçalho simples, usa apenas um cabeçalho binário de 2 bits para especificar o tipo de mensagem, um tópico baseado em texto e, em seguida, uma carga útil binária. Já no HTTP é um protocolo com muitos cabeçalhos e regras pesado chegando a 1KB. HTTP é um protocolo de um para um, o cliente faz uma solicitação e o servidor responde. É difícil e caro transmitir uma mensagem a todos os dispositivos na rede, diferente do MQTT que vários dispositivos podem se comunicar através do broker como aponta (YUAN, 2020).

Figura 32 - Comunicação múltiplas.



Fonte: (Page fault blog, 2020)

Segurança, o MQTT possui uma vantagem sobre o HTTP em relação à segurança dos dados transmitidos, ele usa SSL / TLS como um canal de transmissão de mensagens enquanto criptografa o conteúdo. Por outro lado, o HTTP não oferece nenhuma criptografia e os dados estão disponíveis no formato de texto simples não criptografado, por isso torna o protocolo penetrável por hackers.

Com uso do MQTT no projeto não foi necessário usar a configuração de *forwarding* do roteador que é processo de redirecionamento de portas, na prática é para qual porta e IP da rede local devem ser enviados pacotes recebidos da Internet em uma determinada porta de entrada aberta em conjunto com o serviço de DDNS (*Dynamic Domain Name System*) Sistema de Nome de Domínio Dinâmico que vai atribuir um e um nome de domínio para um endereço de IP de internet dinâmico a um *host* fixo no caso do projeto para o ESP32. Que com essas duas configurações no

roteador citadas acima, o alimentador ficaria disponível para ser acessado de uma rede externa, ou seja, pela internet. Porém com uma brecha na segurança porque com uma porta aberta, a conexão fica vulnerável, fazendo com que os dispositivos conectados na rede local sejam acessados com facilidade por pessoas mal intencionadas.

6 CONCLUSÃO

A automação residencial é uma área que está em constante evolução à medida que novas tecnologias são lançadas no mercado, resultando em benefícios as pessoas como a segurança, a praticidade e a comodidade.

O objetivo deste trabalho foi desenvolver um alimentador que fosse capaz de auxiliar as pessoas no dia-a-dia e mostrar o que a internet e a automação podem trazer de benefício a população.

A partir dos resultados, conclui-se que foi possível realizar a maioria dos objetivos deste projeto, os equipamentos utilizados responderam de forma satisfatória, os microcontroladores Arduino e o ESP32 mostrou serem ferramentas práticas de fácil utilização e com uma boa relação custo-benefício para a automação, assim como é importante destacar a utilização de materiais reaproveitados no processo de montagem do alimentador, pois sem isso deixaria o protótipo com custo maior.

Durante o desenvolvimento do projeto, várias dificuldades foram enfrentadas, principalmente na manipulação e envio de imagem obtida pelo ESP32, porém todo esforço serviu de aprendizado contribuindo para experiência profissional. A montagem prática do alimentador não pode ser realizada presencialmente com os integrantes do grupo, devido ao isolamento social causado pela pandemia do corona vírus (COVID-19).

Esta ideia apresentada neste projeto não visou a obtenção de lucro, e sim ajudar e incentivar a comunidade estudantil a fazer o uso da tecnologia, até mesmo na educação infantil como a robótica educacional que irá estimular o aluno a desenvolver o raciocínio lógico, ter um currículo escolar mais atrativo, a trabalhar em equipe e desenvolver a criatividade.

Tendo em vista uma maior eficiência do projeto para os usuários, algumas ideias de complemento deste projeto, é implementar um monitoramento com câmera tradicional usada no mercado, sensor de carga (peso) para monitorar a quantidade de ração no comedouro e salvar informação sobre a quantidade de ração ingerida pelo pet no banco de dados em nuvem, visando um controle alimentar mais preciso e equilibrando a dieta do animal.

REFERÊNCIAS

AUTOMATICHOUSE. **Automação Residencial como auxílio às necessidades do lar**. Disponível em: <https://www.automatchouse.com.br/automacao-residencial/o-que-e-automacao-residencial>. Acessado em :13 mar. 2020.

KERSCHBAUMER, Ricardo. **Engenharia de controle e Automação – Microcontroladores**. Luzerna, 2019. Apostila. Disponível em: https://cotemar.com.br/wp-content/uploads/2019/10/apostila-microcontroladores_compressed.pdf. Acessado em 21 de jun. 2020.

LIKAM. **H264 vs MPEG4 or MJPEG**. Disponível em: https://www.likam.it/index.php?option=com_content&view=article&id=81&Itemid=165&lang=it. Acessado em: 26 de jun.2020.

NICHOLAS.**MQTT**. Disponível em: <https://devopedia.org/mqtt>. Acessado em: 26 de jun.2020.

PAGE FAULT BLOG. **Using local MQTT broker for cloud and interprocess communication**. Disponível em: <https://pagefault.blog/2017/03/02/using-local-mqtt-broker-for-cloud-and-interprocess-communication/>. Acessado em: 26 de jun.2020.

YUAN, Michael. **Conhecendo o MQTT**. Por que o MQTT é um dos melhores protocolos de rede para a Internet das Coisas? Disponível em: <https://developer.ibm.com/br/articles/iot-mqtt-why-good-for-iot/>. Acessado em: 02 de jul. 2020.