UNIVERSIDADE DO ESTADO DO AMAZONAS
ESCOLA SUPERIOR DE TECNOLOGIA

BEATRIZ CORREIA SANTOS

DESENVOLVIMENTO DE UM SISTEMA PARA REGISTRO E CONTROLE DE RETIRADA DE OBJETOS EM ARMÁRIOS UTILIZANDO AS TECNOLOGIAS RFID E QR CODE

Orientador: Jozias Parente de Oliveira, Dr.

Manaus
2022

BEATRIZ CORREIA SANTOS

DESENVOLVIMENTO DE UM SISTEMA PARA REGISTRO E CONTROLE DE RETIRADA DE OBJETOS EM ARMÁRIOS UTILIZANDO AS TECNOLOGIAS RFID E QR CODE

Pesquisa desenvolvida durante a disciplina de Trabalho de Conclusão de Curso II e apresentada à banca avaliadora do Curso de Engenharia Elétrica da Escola Superior de Tecnologia da Universidade do Estado do Amazonas, como pré-requisito para a obtenção do título de Bacharel em Engenharia Elétrica.

Orientador: Jozias Parente de Oliveira, Dr.

Manaus

2022

Universidade do Estado do Amazonas – UEA
Escola Superior de Tecnologia - EST

Reitor:
André Luiz Nunes Zogahib
Vice-Reitora:
Kátia do Nascimento Couceiro
Diretora da Escola Superior de Tecnologia:
Ingrid Sammyne Gadelha Figueiredo
Coordenador do Curso de Engenharia Elétrica:
Israel Gondres Torné

Banca Avaliadora composta por: *Data da defesa: 14/10/2022.*
Prof. Dr. Jozias Parente de Oliveira *(Orientador)*
Prof. Dr. Raimundo Cláudio Souza Gomes
Prof. Dr. Israel Gondres Torné

Revisão Ortográfica: ***Jozias Parente de Oliveira***

CIP – Catalogação na Publicação

Santos, Beatriz Correia

Desenvolvimento de um sistema para registro e controle de retirada de objetos em armários utilizando as tecnologias RFID e QR Code / Beatriz Correia Santos; orientado por Jozias Parente de Oliveira. – Manaus: 2022.
65 p.: il.

Trabalho de Conclusão de Curso (Graduação em Engenharia Elétrica). Universidade do Estado do Amazonas, 2022.

1. Controle de estoque. 2. RFID. 3. QR Code. 4. IoT, 5. MQTT, 6. banco de dados, 7. SQLite I. Parente de Oliveira, Jozias, II. Universidade do Estado do Amazonas. III. Escola Superior de Tecnologia.

BEATRIZ CORREIA SANTOS

DESENVOLVIMENTO DE UM SISTEMA PARA REGISTRO E CONTROLE DE RETIRADA DE OBJETOS EM ARMÁRIOS UTILIZANDO AS TECNOLOGIAS RFID E QR CODE

Pesquisa desenvolvida durante a disciplina de Trabalho de Conclusão de Curso II e apresentada à banca avaliadora do Curso de Engenharia Elétrica da Escola Superior de Tecnologia da Universidade do Estado do Amazonas, como pré-requisito para a obtenção do título de Bacharel em Engenharia Elétrica.

Nota obtida: _____ (_____)

Aprovada em _____/_____/_____.

Área de concentração: Sistemas Embarcados

BANCA EXAMINADORA

_____
Orientador: Jozias Parente de Oliveira, Dr.

_____
Avaliador: Raimundo Cláudio Souza Gomes, Dr.

_____
Avaliador: Israel Gondres Torné, Dr.

Manaus
2022

DEDICATÓRIA

Dedico este trabalho a minha mãe, pois é graças ao seu esforço que hoje concluo o meu curso. À minha avó, Maria Odacy, que agora se encontra ao lado de Deus e sei que se estivesse aqui estaria muito orgulhosa e feliz por eu ter chego até aqui.

AGRADECIMENTOS

À Deus, por ter me concedido o dom da vida. À minha mãe, por ser meu porto seguro e minha razão de chegar até aqui. A minha família, por sempre me apoiar. Aos meus colegas da EST que de alguma forma contribuíram para eu chegar aqui, e principalmente a Jessamine que foi companheira e compartilhou os momentos mais difíceis da graduação. E também aos amigos do LSE em especial ao Gabriel e Renan. E por fim, aos professores da EST que contribuíram pela minha formação, em especial ao professor Jozias, por ter sido meu orientador e ter desempenhado tal função com dedicação e amizade.

Ninguém disse que seria fácil....
Mas consegui!

RESUMO

Um ambiente onde muitas pessoas têm acessos a objetos que são utilizados constantemente pode gerar uma perda no controle. Desta forma, alguns locais onde este processo acontece utilizam-se de planilhas de controle de entradas e saídas, normalmente feitas a mão onde se registram os horários das entradas e saídas dos objetos e quem os pegou. Encontrar uma maneira de fazer este controle de forma mais rápida e inteligente pode melhorar o nível de controle ao saber quem e o que foi retirado de um local, assegurando uma entrada e saída fácil e simplificada. O presente trabalho teve como objetivo o desenvolvimento de um sistema de registro e controle de objetos utilizando as seguintes tecnologias: RFID (Radio Frequency Identification) para a identificação das pessoas que terão acesso aos objetos e o QR Code, isto é, um código de barras, bidimensional, que é utilizado no monitoramento das saídas e entradas dos objetos. Os objetos são acondicionados em um ambiente fechado no qual o acesso é permitido somente por meio de algum cartão ou chaveiro com uma antena receptora que capte o sinal emitido por um leitor de rádio frequência que é utilizado para identificar a pessoa que fez o acesso ao local. Ao fazer a retirada do objeto, uma câmera verifica se há algum QR Code, e caso afirmativo o sistema registra a retirada do objeto no nome da última pessoa que obteve acesso ao ambiente. Desta forma, como resultados obteve-se o sistema com dois módulos, o RFID e o QR Code e o banco de dados no qual são registrados os acessos ao armário e os objetos retirados, podendo ser visualizados no banco de dados com data e hora dos eventos.

Palavras-chave: Controle de estoque, RFID, QR Code, IoT, MQTT, banco de dados, SQLite.

ABSTRACT

An environment where many people have access to objects that are used constantly can generate a loss of control. In this way, some places where this process takes place use input and output control spreadsheets, normally made by hand, where the times of entry and exit of objects and who took them are recorded. Finding a way to do this control faster and smarter can improve the level of control by knowing who and what has been removed from a location, ensuring easy and streamlined entry and exit. The present work aimed to develop a system for recording and controlling objects using the following technologies: RFID (Radio Frequency Identification) for the identification of people who will have access to objects and the QR Code, that is, a bar code, two-dimensional, which is used to monitor the outputs and inputs of objects. The objects are stored in a closed environment in which access is only allowed by means of a card or key fob with a receiving antenna that captures the signal emitted by a radio frequency reader that is used to identify the person who has accessed the place. When removing the object, a camera checks if there is a QR Code, and if so, the system records the removal of the object in the name of the last person who gained access to the environment. In this way, as a result, the system with two modules, the RFID and the QR Code and the database in which the accesses to the cabinet and the removed objects are registered, can be visualized in the database with date and time of the events.

Keywords: Inventory control, RFID, QR Code, IoT, MQTT, database, SQLite.

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

INTRODUÇÃO

O controle de entradas e saídas de objetos ou controle de estoque passou a ser importante dentro de indústrias, empresas que prestam serviços e até laboratórios, pois servem como regulador do fluxo dos objetos. Desta forma, alguns locais onde este processo acontece utilizam-se de fichas de prateleiras ou por fichas de controle, inclusive até hoje ainda existem empresas que trabalham com um desses sistemas, normalmente feitas a mão onde se registram os horários das entradas e saídas dos objetos (BALLOU, 2010). Ou seja, são feitos inventários de forma manual, e apesar do baixo custo operacional há o custo de paralisação do trabalho diário de funcionários para realizar o inventário causando o custo de horas extras (NOGUEIRA, 2018).

Com o avanço das tecnologias na área de IoT (Internet of Things) é possível encontrar uma maneira de fazer este controle de forma mais rápida e inteligente, assegurando uma entrada e saída fácil e simplificada. Tecnologias como o RFID, que utiliza a comunicação de rádio para identificar um objeto físico, e o QR Code que é um tipo de código de barras de matriz ou código bidimensional que pode armazenar informações de dados projetado para ser lido por câmeras, podem ser utilizados no desenvolvimento de um sistema de registro e controle de objetos que identificarão as pessoas e objetos, respectivamente, de modo que o sistema possa garantir os registros dos eventos de entrada e saída dos objetos. Assim sendo, o objetivo geral deste trabalho foi desenvolver um sistema baseado no microcontrolador ESP32 para registrar e controlar automaticamente a retirada de objetos de um armário de ferramentas por meio da leitura de dados enviados de etiquetas de RFID e QR Code.

Os objetivos específicos estabelecidos foram os seguintes: descrever as principais estratégias e soluções de mercado utilizadas para controle de estoque, realizar um levantamento bibliográfico acerca das tecnologias de RFID e QR Code para utilização na automatização do controle e registro de retirada de objetos em armários, e definir a arquitetura de hardware para a elaboração de um sistema para registro e controle de retirada de objetos em armários utilizando as tecnologias RFID e QR Code.

Como justificativa e motivação para o desenvolvimento deste trabalho destaca-se que em um centro de pesquisa e desenvolvimento, existe um laboratório de montagem e testes de hardware onde as ferramentas são armazenadas em armários e há uma dificuldade no controle dos objetos utilizados no dia a dia, pois são feitos de forma manual. Desta forma, torna-se importante o desenvolvimento de um sistema que melhore de forma significativa o controle de entradas e saídas de objetos de armário afim de eliminar as falhas ocasionadas pelo controle

manual, com a possibilidade de ser expandidos para ambientes maiores, como em indústrias. Ademais, com a revisão da literatura acerca das tecnologias, o presente trabalho poderá servir como ponto de partida para outras aplicações de controle de estoque.

O desenvolvimento desse trabalho se subdivide em quatro capítulos principais intitulados como: Referencial Teórico, Materiais e Métodos, Implementação do Projeto e Análise dos Resultados.

O primeiro capítulo está destinado ao Referencial Teórico, no qual é feito uma revisão dos assuntos necessários para a implementação do projeto. Inicialmente, é introduzido sobre a importância do controle de estoque dentro de uma empresa ou até laboratórios, também sobre o funcionamento das tecnologias que foram utilizadas no desenvolvimento do sistema, quais seja: RFID e o QR Code. Além disso, discorre-se sobre o que é a Internet das Coisas, microcontroladores, e especificamente sobre o ESP32, as camadas de rede o protocolo a ser utilizado no sistema e sobre o banco de dados utilizado no sistema. E, por fim, faz-se uma relação deste trabalho com outros já publicados utilizando ideias similares.

O segundo capítulo define os métodos, as técnicas e os procedimentos utilizados para a concepção do projeto. É explanado qual a modalidade da pesquisa, sua natureza seu objetivo e sua abordagem. Em seguida, dá-se ênfase nas descrições das atividades de forma sequencial para o desenvolvimento do projeto, dividindo-as em etapas.

O terceiro capitulo consiste na implementação do projeto. Primeiramente, foi feita a descrição de como foi realizada a preparação do ambiente onde seriam desenvolvidos os *firmwares* dos módulos, e em seguida a implementação do *firmware*. Também é descrito como foram desenvolvimentos os módulos do sistema e materiais utilizados. Outra etapa descrita, foi a integração dos códigos-fontes com os módulos. Em seguida, foi descrito sobre o desenvolvimento do banco de dados. Por fim, discorre-se sobre os testes de validação do sistema.

O quarto e último capítulo apresenta a análise dos resultados do projeto. Inicialmente, foi feita a instalação do sistema em um armário e o cadastrado dos nomes e ID’s de pessoas autorizadas a terem o acesso ao armário. E em seguida, o sistema passou por testes durante uma hora afim de obter um banco de dados com os registros das retiradas. Por fim, os dados coletados foram analisados.

1. REFERENCIAL TEÓRICO

1.1. CONTROLE DE ESTOQUE

O controle de estoque ou administração de recursos é um problema importante e de natureza igual a outros problemas encontrados em um laboratório ou até mesmo em uma indústria e deve também ter uma atenção para ser resolvido. Empresas armazenam grandes quantidades de materiais em estoque e administrar essa quantidade se torna algo muito difícil principalmente de forma manual usando planilhas como mostra a Figura 1, de forma que não é possível ter o mesmo controle e atenção para diferentes tipos de itens. Mas há itens que devem ser controlados separadamente com uma atenção especial devido à sua importância em termos do alto índice de investimento ou importância técnica do item na produção (KUNIGAMI, 2009). Desta forma, a automatização da gestão de estoques torna-se uma vantagem ao administrar estoques com pequenas e grandes quantidades de itens.

Figura 1. Exemplo de planilha para controle de estoque.

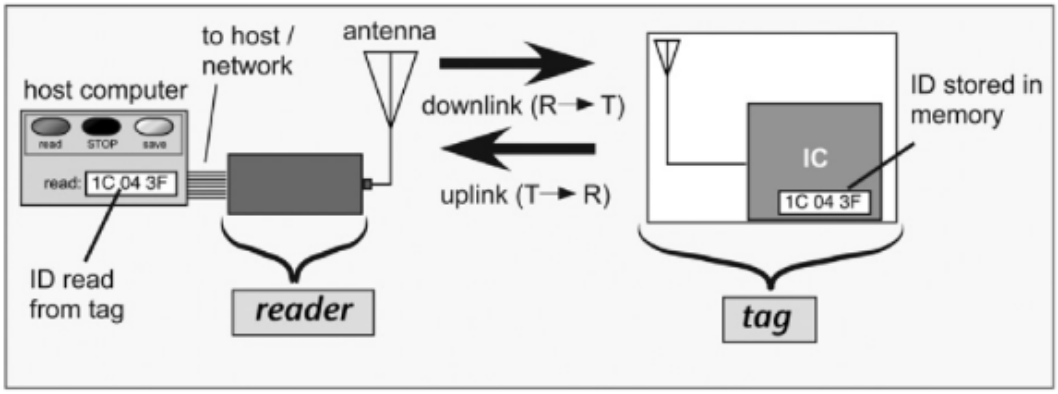
Fonte: (SMART PLANILHAS LTDA, 2020).

1.2. RFID

Um sistema RFID (figura 2) consiste principalmente em um transmissor-receptor (chamado de leitor) conectado a uma antena e um conjunto de transponders ou etiquetas, onde as informações são armazenadas. O transceptor se comunica com um computador por meio de uma aplicação, que gerencia os dados armazenados nas etiquetas. A antena estabelece a comunicação entre o transceptor e o transponder. O raio de ação do sistema também é

influenciado pelo tipo de etiquetas aderidas aos objetos. Estes dispositivos consistem em duas partes: um circuito integrado, que armazena e processa as informações, modula o sinal e coleta energia do transceptor, se necessário; e uma antena para transmitir e receber sinais (M. DOBKIN, 2013).

Figura 2. Visão geral do sistema de RFID.



Fonte: (M. DOBKIN, 2013, p. 22)

Existem três tipos de etiquetas: ativa, passiva e semi passiva. A etiqueta ativa integra a fonte de alimentação e transmite o sinal ao transceptor (K. KLAIR; CHIN; RAAD, 2010). Por outro lado, a etiqueta passiva obtém a energia necessária do leitor. A etiqueta passiva tem alcance de ação limitado (em média de 15 m), enquanto a ativa pode ser usada para distâncias de até 100 m. E por fim, a etiqueta semi-passiva pode transmitir, mas o retro espalhamento é usado. Além disso, essas etiquetas precisam ser ativadas por um sinal (LI; BECERIK-GERBER, 2011). A Tabela 1 mostra as diferenças entre os diferentes tipos de etiqueta.

Tabela 1. Diferenças entre os tipos de etiquetas RFID.

	Tag ativa	**Tag passiva**	**Tag semi passiva**
Faixa de distância	Até 100m	Até 15m	Até 60-80 m
Poder	Fonte de alimentação (Bateria)	Induzido de leitores	Ativado por um sinal
Custo relativo	>30 $	1 $	>20 $
Armazenamento de dados	Extensível e pode variar	512 bytes a 4 KB	Extensível e pode variar
Taxa de transferência de dados	Até 128 KB/s	Até 1 KB/s	Até 16 KB/s
Vida	Até 10 anos	Ilimitado	Mais de 6 anos

Fonte: (M. DOBKIN, 2013, p.7)

1.3. QR CODE

O código QR foi criado pela Denso Wave, subsidiária da Toyota, em 1994, e foi inicialmente usado para rastrear o estoque na fabricação de peças de veículos, e atualmente, tornou-se um método de codificação muito utilizado devido à sua conveniência de leitura, gravação e capacidade de correção de erros. A informação codificada pode ser texto, URL ou outros dados (M. DOBKIN, 2013). Um exemplo de QR Code pode ser observada na figura 3.

Figura 3. Exemplo de QR Code.

Fonte: Autoria Própria

1.4. INTERNET OF THINGS – IOT

Internet das coisas, do inglês Internet of Things, é uma rede de interconexão de dispositivos com acesso a rede local ou global de Internet (Figura 4). Através dessa conexão, pode haver uma troca de informações utilizando a estrutura da conexão para que haja, por exemplo, uma coleta de dados de diferentes sensores de forma remota, possibilitando o monitoramento das "coisas" por qualquer pessoa em qualquer lugar (STEVAN JUNIOR, 2018).

Figura 4. Exemplo de arquitetura de IoT.

Fonte: (TECLÓGICA, 2021, p.1)

1.5. MICROCONTROLADORES

Um microcontrolador é considerado um microcomputador que contém processador, memória e periféricos de entrada e saída em um único chip, de tal maneira que se resume a uma forma genérica da integração entre uma Unidade de Processamento Central, do inglês Central Processing Unit (CPU), e os periféricos que fazem a interface do microcontrolador com o mundo externo (OLIVEIRA JÚNIOR; DUARTE, 2010).

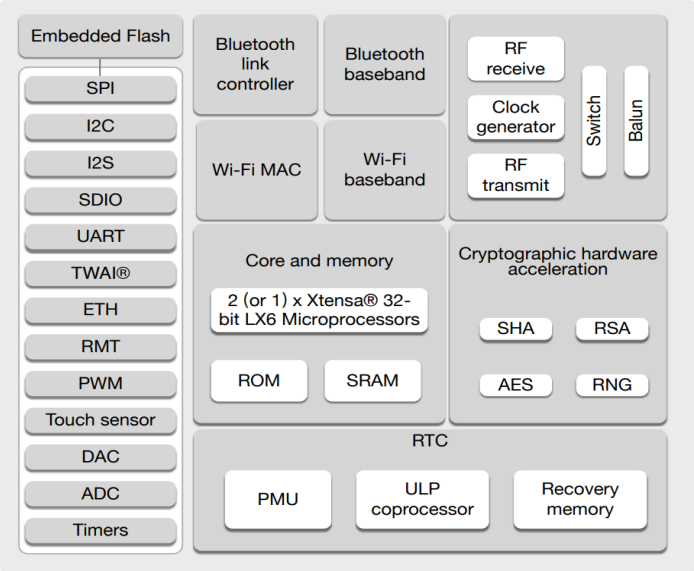
1.5.1. Módulo ESP32

O módulo ESP32 é um microcontrolador de "Sistema em um Chip", do inglês System on a Chip (SoC) capaz de realizar múltiplas tarefas e com poder de processamento bastante

elevado devido ao uso de dois núcleos de processamento de 32-bit. Devido à utilização já embarcada de dois módulos de comunicação bastante utilizados, que são os módulos de Wi-Fi e o Bluetooth, a escolha deste microcontrolador para aplicações IoT se tornou extensa, pois facilita a integração com diferentes dispositivos. As variadas interfaces periféricas do ESP32, tais como, SPI, I2C, I2S e UART, possibilitam a integração com diferentes sensores e módulos (ESPRESSIF, 2021).

O diagrama em blocos do ESP32 possibilita a visualização de seus periféricos e suas tecnologias de forma a facilitar e ilustrar a utilização e disposição deles, conforme ilustrado na Figura 5.

Figura 5. Diagrama em Blocos do ESP32.

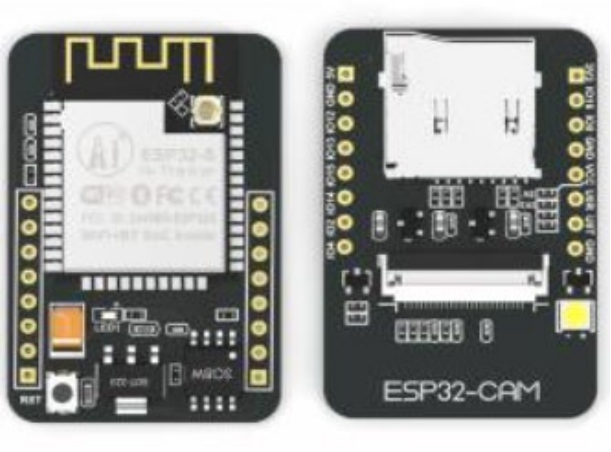


Fonte: (ESPRESSIF, 2022, p. 12).

1.5.2. Módulo AlThinker ESP 32 Cam

O módulo AlThinker ESP 32 Cam, visto na Figura 6, vem com chip ESP32-S (da mesma família do ESP32), possui um cabo especial para conectar a câmera OV2640, que possui uma lente grande angular com ângulo de visão de 160° e permite obter imagens com resolução de até 1600x1200 com taxa de atualização máxima de 15 FPS e um slot para cartão de memória microSD. O slot para cartão micro SD é usado para armazenar imagens no formato JPEG (AI-THINKER TECHNOLOGY CO., 2017).

Figura 6. ESP32-CAM.

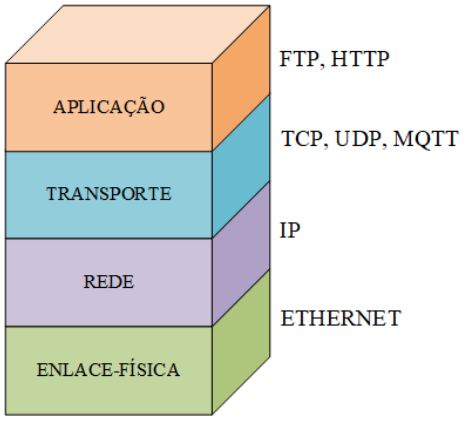


Fonte: (AI-THINKER TECHNOLOGY CO., 2017, p.1).

1.6. TCP/IP

O TCP/IP é um modelo de conjunto de camadas de rede nos quais oferecem recursos necessários para a conectividade. Esse modelo é dividido em 4 pilhas de protocolos, conhecidos como camadas, sendo eles: Camada de aplicação, Camada de transporte, Camada de rede e Camada de Enlace-Física (OLIVEIRA, 2017) (Figura 7). Na camada de transporte é por onde protocolos de troca de pacotes se situam, como por exemplo, o MQTT, e na camada de enlace-física é por onde se faz a conexão física com a rede, como por exemplo o Wi-Fi.

Figura 7. Camadas de comunicação TCP/IP.

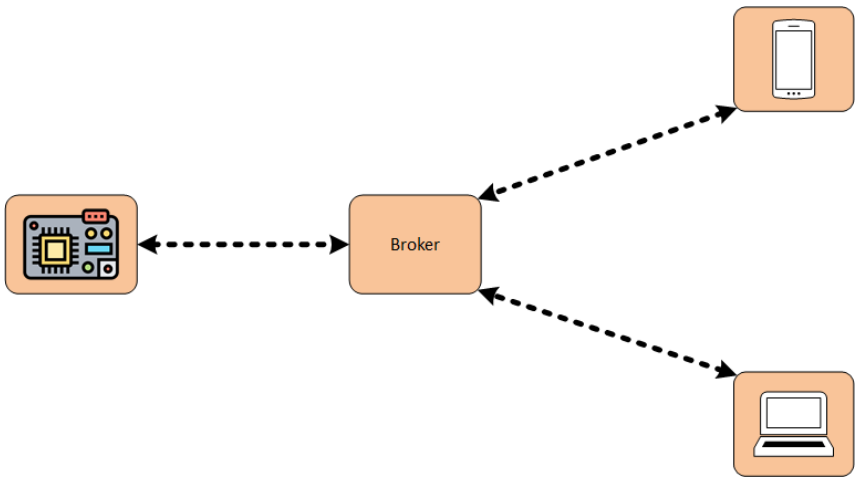


Fonte: Autoria Própria

1.7. MQTT

O MQTT é um protocolo simples e de baixa complexidade de transporte de mensagens que utiliza o modelo publicador/subscritor, do inglês publisher/subscriber, através de um gerenciador denominado corretor, ou broker, de forma que um produtor de dados publique uma mensagem para um destino chamado de tópico, e caso haja algum cliente inscrito neste tópico, este irá receber a mensagem enviada (MESNIL, 2014). Por ser um protocolo de reduzida complexidade, implicando em menor utilização de recursos computacionais, o MQTT é bastante utilizado em aplicações embarcadas utilizando microcontroladores para fazer a interface entre sensores e a rede de Internet, ou seja, a aplicação IoT se torna simplificada ao fazer o uso deste protocolo de transporte de mensagens. Na figura 8 é ilustrado um exemplo de arquitetura MQTT.

Figura 8. Exemplo de arquitetura MQTT.



Fonte: Autoria Própria.

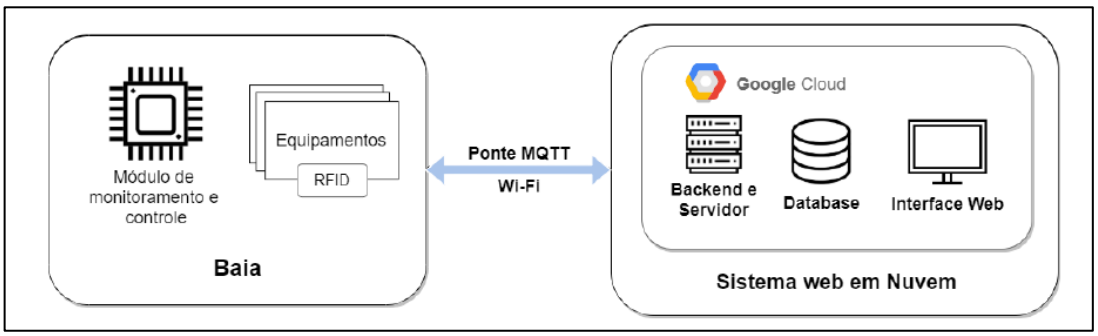
1.8. *SQLITE* - BANCO DE DADOS

SQLite é um sistema de banco de dados de domínio público útil para armazenamento de dados relacional, ou seja, usados para armazenar registros definidos pelo usuário em grandes tabelas. É simples de usar e operar, além de ser suportado por multiplataformas. Sendo o mais aplicado em sistemas de aplicativos baseados em Android, por exemplo. É amplamente utilizado devido à sua flexibilidade e plataforma independente. Ele contém tipos de dados simples, com capacidade de consulta dinâmica (A. KREIBICH, 2010).

1.9. TRABALHOS RELACIONADOS

O trabalho proposto por AMORIM (2020), tem como objetivo desenvolver um sistema de monitoramento para um laboratório de hardware, que controle a abertura automática de baias e monitore o uso dos equipamentos contidos nelas. A Figura 9, mostra a arquitetura utilizada pelo autor.

Figura 9. Arquitetura do Sistema implementado.

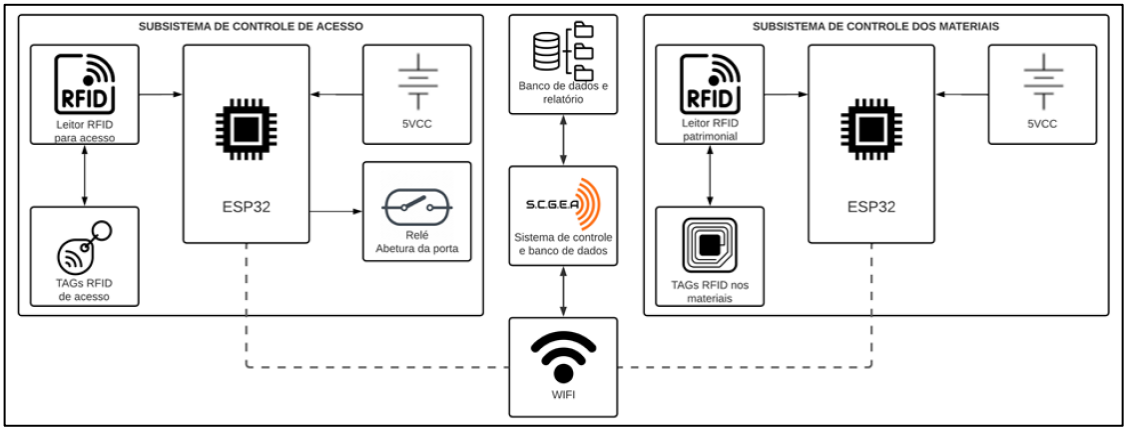


Fonte: AMORIM, 2020, p.32

O sistema desenvolvido permite aos alunos acessar as baias a qualquer hora do dia com seu crachá RFID e utilizar os equipamentos. O sistema gerencia a abertura da baia para usuários que tenham permissão, através da leitura do crachá RFID e da consulta do ID RFID do usuário no banco de dados. Também detecta, automaticamente, quando um equipamento é retirado para uso, quando é devolvido ao seu lugar ou possivelmente quando for extraviado, enviando todas as ações e informações ao sistema web em nuvem.

O trabalho proposto por Gross, (2022) teve como objetivo o desenvolvimento de um sistema que gerenciasse o controle de acesso ao almoxarifado, bem como o cadastro dos materiais com a finalidade de acompanhamento em tempo real as retiradas e o retorno dos equipamentos por meio da identificação por rádio frequência. O sistema foi dividido em dois subsistemas, como observado no diagrama em blocos visto na figura 10, o primeiro com o controle ao almoxarifado e o segundo para o controle dos materiais em si, sendo ambos integrados por meio de uma rede de internet Wi-Fi, que capta os dados enviados dos microcontroladores tratando e construindo um banco de dados que proporciona agilidade nas gestões logísticas do almoxarifado, e consequentemente o acompanhamento em tempo real dos acessos realizados no ambiente e dos materiais presentes ou não no estoque.

Figura 10. Diagrama de blocos do sistema geral do autor Gross.



Fonte: Gross, 2022, p.18

Ambos os trabalhos apresentados se utilizaram somente da tecnologia RFID para fazer o monitoramento de objetos, seja dos equipamentos de laboratório ou almoxarifado. Portanto, neste presente trabalho foi utilizado além da tecnologia de identificação por rádio frequência, a tecnologia QR Code para a identificação da retirada das ferramentas.

2. MATERIAIS E MÉTODOS

Neste capítulo serão abordados todos os aspectos metodológicos da pesquisa realizada, descrevendo-se os procedimentos necessários para desenvolver um sistema baseado no microcontrolador ESP32 para registrar e controlar automaticamente a retirada de objetos de um armário de ferramentas por meio da leitura de dados enviados de etiquetas de RFID e QR Code.

Esse trabalho teve por finalidade realizar uma pesquisa de natureza aplicada. Para alcançar os objetivos propostos e melhor apreciação deste trabalho, foi utilizada uma abordagem quantitativa. Com intuito de conhecer a problemática sobre a área de estudo foi realizada uma pesquisa exploratória acerca das tecnologias utilizadas para implementação do projeto. Para obtenção dos dados necessários, foi utilizada a pesquisa experimental como procedimento técnico.

Ademais, neste capítulo é retratada a sequência de eventos para a criação do projeto, desde a pesquisa, instalação e criação das bibliotecas no ambiente de desenvolvimento da IDE do Arduino ao posicionamento dos módulos no ambiente, processamento e envio de dados. O sistema consiste em um dispositivo que lê a informação contida num QR Code monitorando as saídas e entradas de objetos no armário, e um dispositivo que irá coletar o acesso de pessoas ao armário através do RFID de um cartão ou chaveiro, para processar e gerenciar cada uma dessas informações através de um banco de dados.

Para a construção do projeto, o mesmo foi dividido em um módulo RFID controlado pelo microcontrolador ESP32, e um módulo ESP32-Cam que contém uma câmera e também um microcontrolador ESP32.

2.1. MÉTODO PROPOSTO

Na primeira etapa foi realizado pesquisas sobre o controle de estoque, nas áreas de microcontroladores, em especial sobre o ESP32 da *Espressif* e da linguagem de programação C utilizada no IDE Arduino onde foi feita a programação do ESP32, e ainda sobre as tecnologias RFID e QR Code com base em artigos científicos.

Na segunda etapa, foi feita a aquisição dos módulos a serem utilizados durante o projeto para a validação do sistema de registro e controle automático, como o módulo leitor RFID MFRC522, o kit de desenvolvimento do ESP32 que foi utilizado para fazer o controle do

módulo leitor RFID que faz a leitura dos cartões e chaveiros RFID; o módulo ESP32-Cam que foi utilizado para ler o QR Code das ferramentas.

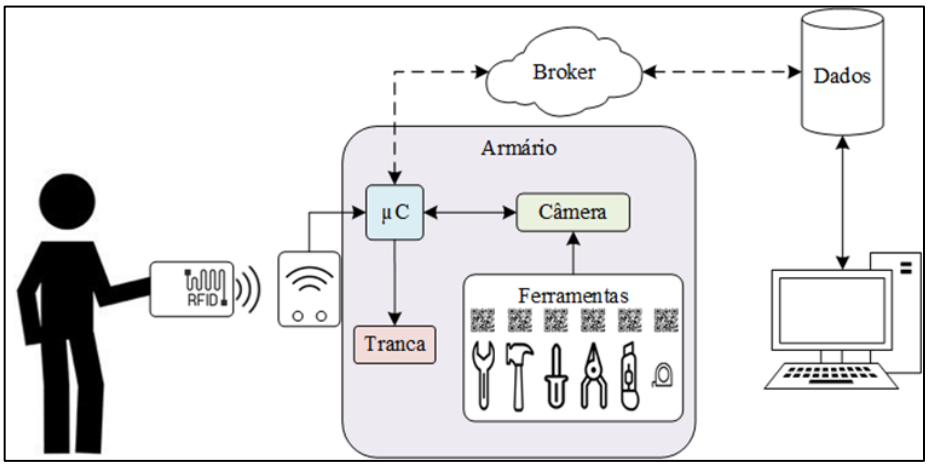
Na terceira etapa, foi desenvolvido o firmware utilizando a interface de desenvolvimento IDE Arduino versão 1.8.19. Inicialmente foram desenvolvidas bibliotecas para cada parte do código afim de deixar o código mais simples e organizado, dentro dessas bibliotecas criadas foram feitas as definições para que os módulos e os protocolos de comunicação fossem reconhecidos pelo microcontrolador, e também foram programados os comandos necessários para o desenvolvimento do sistema.

Na quarta etapa, foi feito o desenvolvimento dos protótipos utilizando placas de circuito impressos ilhadas, de forma que cada módulo possuísse alimentação externa. Em seguida, foi feita a montagem da placa de circuito impresso manualmente. E por fim, uma inspeção geral da placa por meio de testes com multímetro. Em seguida, fez-se a montagem do sistema, com a fonte e os módulos que foram acondicionados dentro de um case impresso em impressora 3D.

Na quinta etapa fez-se a validação do sistema, onde o firmware foi embarcado no hardware e o protótipo passou por testes em laboratório. Incialmente foram realizados testes individuais para validar as partes integrantes do sistema. Em seguida foi realizada a integração dos itens de hardware para validação das funcionalidades como um todo, e por fim o sistema foi posicionada em um armário para realização dos testes.

Na sexta etapa, foi desenvolvido um banco de dados utilizando SQLite, de forma a gerenciar de forma simplificada as informações. Por fim, realizou-se o cadastro dos usuários e a integração entre os dados transmitidos pelo microcontrolador e a base de dados. A Figura 11 ilustra um diagrama em blocos geral do sistema desenvolvido ao longo desta pesquisa.

Figura 11. Sistema a ser desenvolvido.



Fonte: Autoria Própria

2.2. MATERIAIS UTILIZADOS NA CONSTRUÇÃO DO SISTEMA

Para a confecção da placa do módulo RFID e módulo ESP32- CAM, foram utilizados os materiais listados abaixo:

- 1 módulo leitor RFID MFRC522;
- 1 kit de desenvolvimento do ESP32;
- 3 Cartões RFID;
- 2 Chaveiros RFID;
- 1 módulo ESP32-Cam;
- 1 LED de cor verde;
- 1 Placa de circuito impresso ilhada (Wire-Wrap) dupla face;
- 1 barra de soquete header fêmea;
- 1 barra de soquete header macho;
- Cabos Jumper Fêmea-Macho;
- Cabos Jumper Fêmea-Fêmea;
- 2 Resistor de 1k ohm;
- 2 conectores P4 Fêmea;
- 2 carregadores 5V.

3. IMPLEMENTAÇÃO DO PROJETO

Este capítulo apresenta os procedimentos da construção do protótipo, tais como: O desenvolvimento do firmware no ambiente da Arduino IDE; desenvolvimento de uma placa para o módulo RFID, e para o módulo ESP32-Cam, objetivando fazer o registro do acesso das pessoas ao armário e a identificação dos objetos retirados. Além disso, foi desenvolvido um banco de dados para o registro das pessoas autorizadas. São apresentados neste capítulo os seguintes tópicos:

a) preparação do ambiente de desenvolvimento do firmware;
b) implementação do *firmware;*
c) desenvolvimento dos protótipos dos módulos;
d) integração dos módulos com o *firmware*;
e) desenvolvimento do banco de dados;
f) testes de validação do protótipo do sistema.

3.1. PREPARAÇÃO DO AMBIENTE DE DESENVOLVIMENTO DO FIRMWARE

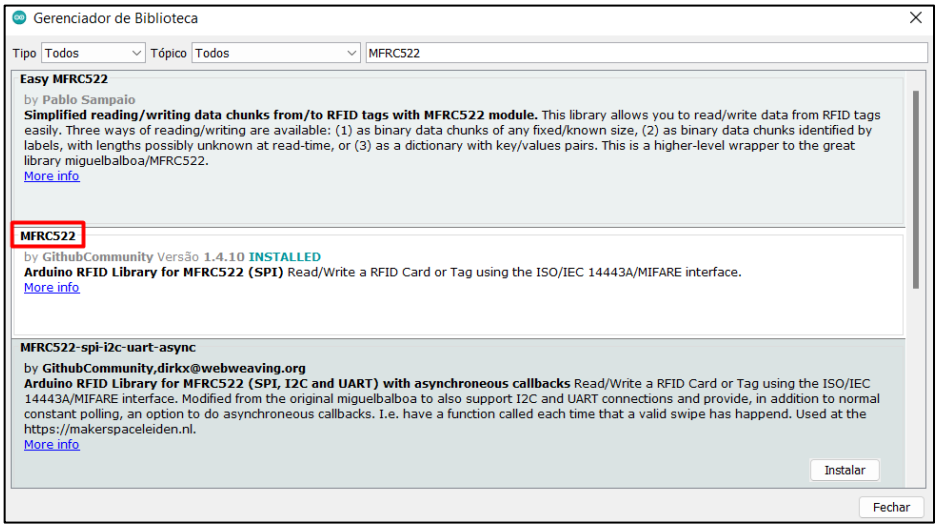
Para o desenvolvimento deste trabalho foram instaladas algumas bibliotecas específicas que serão usadas durante a programação. Primeiramente, foi adicionada a biblioteca "MFRC522" do módulo leitor RFID MFRC522, seguindo os passos a seguir:

a) Abriu-se o Arduino IDE e navegou-se até o Gerenciador de Bibliotecas através do caminho, Ferramentas>Gerenciar Bibliotecas;

b) No campo de pesquisa buscou-se por 'MFRC522', e apareceu uma opção *MFRC522 by Github Community Versão 1.4.10*, como mostra a figura 12;

c) Instalou-se a biblioteca clicando em "Instalar".

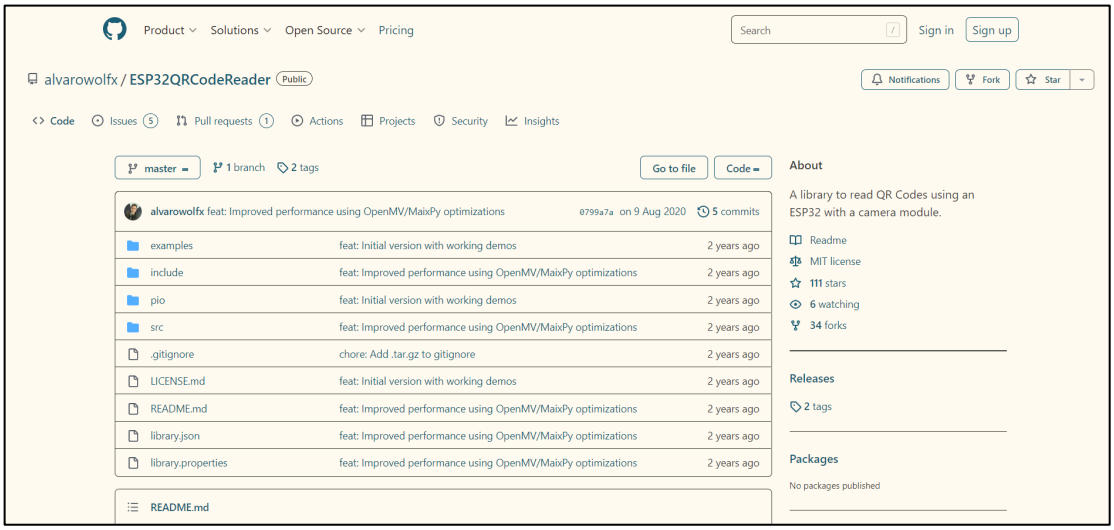
Figura 12. Instalação da biblioteca MFRC522.



Fonte: Autoria Própria.

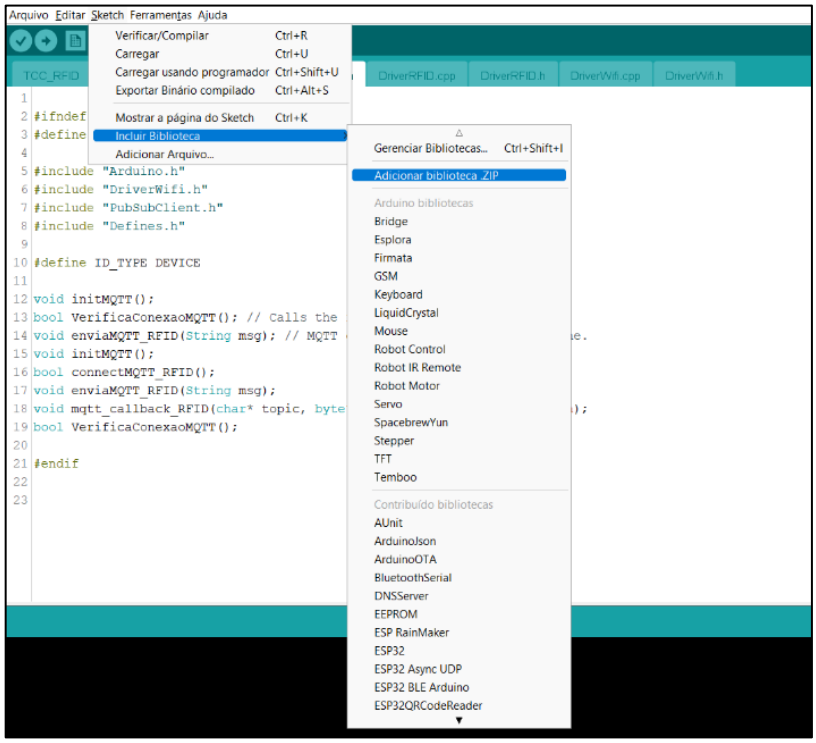
Outra biblioteca que foi utilizada para o desenvolvimento do firmware da placa que fará a leitura do QR Code é “ESP32QRCodeReader”, para esta ser adicionada o primeiro passo foi baixar a biblioteca desenvolvida por Álvaro Viebrantz disponível em seu GitHub no endereço *https://github.com/alvarowolfx/ESP32QRCodeReader*, como visto na Figura 13. Em seguida, carregou-se a biblioteca através do menu *Sketch*, e selecionando ‘Incluir Biblioteca’ e depois em ‘Adicionar Biblioteca .ZIP’, como visto na Figura 14. E por fim, selecionou-se a pasta *ESP32QRCodeReader-master*.

Figura 13. GitHub da biblioteca desenvolvida por Álvaro Viebrantz.



Fonte: Autoria Própria

Figura 14. Caminho para adicionar uma biblioteca no IDE Arduino.

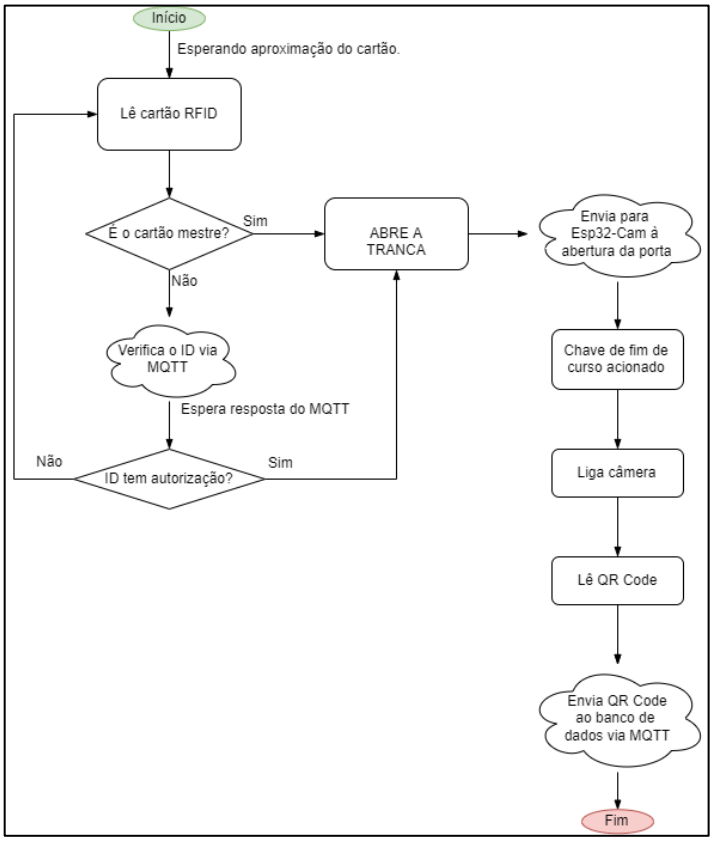


Fonte: Autoria Própria

3.2. IMPLEMENTAÇÃO DO *FIRMWARE*

Nesta etapa é descrito o procedimento para implementação do *firmware*, e os métodos abordados. O sistema foi dividido em duas partes, portanto foram desenvolvidos dois *firmwares*, uma para a implementação do RFID e o outro para o QR Code. A Figura 15 apresenta um fluxograma com a lógica do código-fonte do sistema completo.

Figura 15. Fluxograma da lógica do código-fonte.

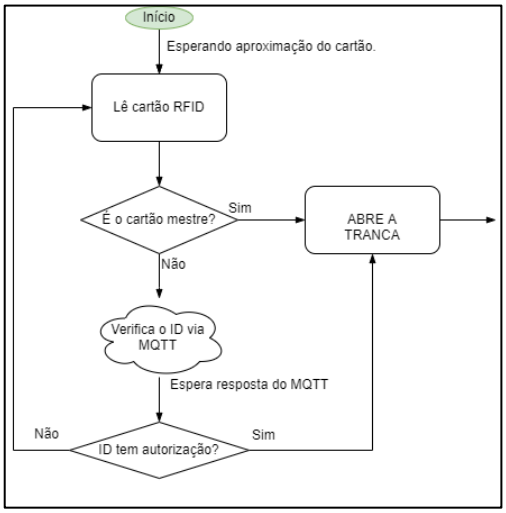


Fonte: Autoria Própria

3.2.1. *Firmware* do módulo leitor RFID

O fluxo do módulo leitor RFID é apresentado na Figura 16. O módulo leitor RFID é o responsável pela leitura do cartão RFID, e este módulo foi conectado ao microcontrolador ESP32, sendo ele o responsável pelo envio dos identificadores únicos dos cartões RFID ao banco de dados via protocolo MQTT.

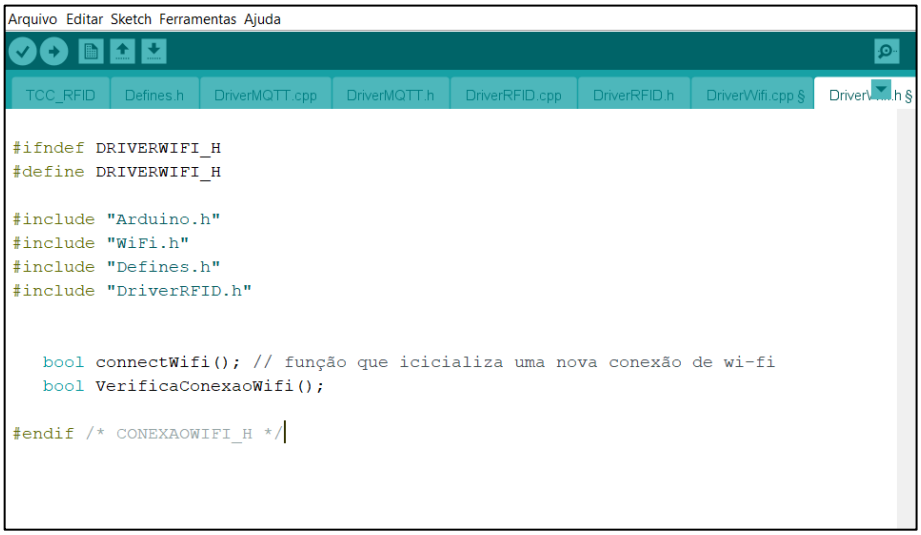
Figura 16. Parte do fluxograma referente ao leitor RFID.



Fonte: Autoria Própria

O código completo foi separado em *drivers* cada um responsável por uma parte do código afim de facilitar o entendimento e melhorar a organização do mesmo, o primeiro *driver* desenvolvido diz respeito a conexão de uma rede *Wi-Fi* local para que possa haver a comunicação com o MQTT. No código foram adicionados os parâmetros de login e senha para que o microcontrolador se conectasse corretamente no *Wi-Fi*. Na Figura 17 a seguir pode-se observar a inclusão das bibliotecas necessárias e na Figura 18 onde seriam feitas a tentativas de conexão.

Figura 17. Inclusão das bibliotecas para a conexão do *Wi-Fi.*



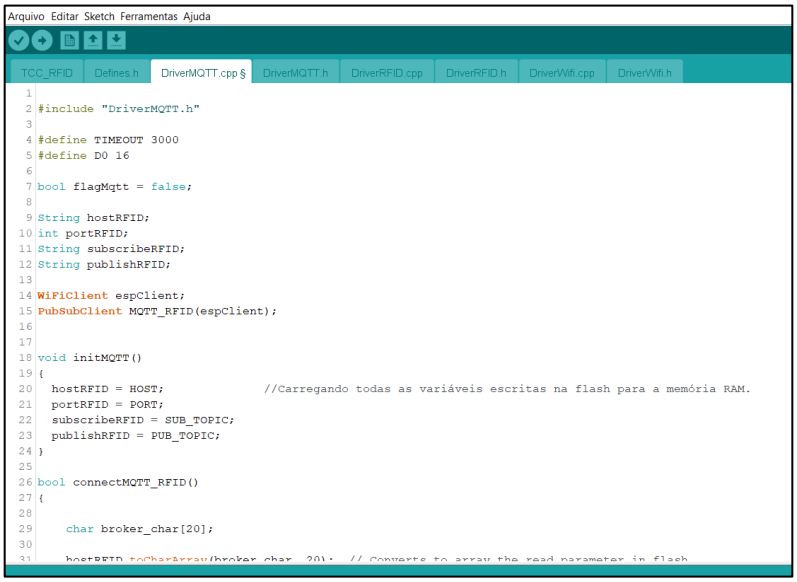
Fonte: Autoria Própria

Figura 18. Parte do código referentes a conexão *Wi-Fi*.

Fonte: Autoria Própria

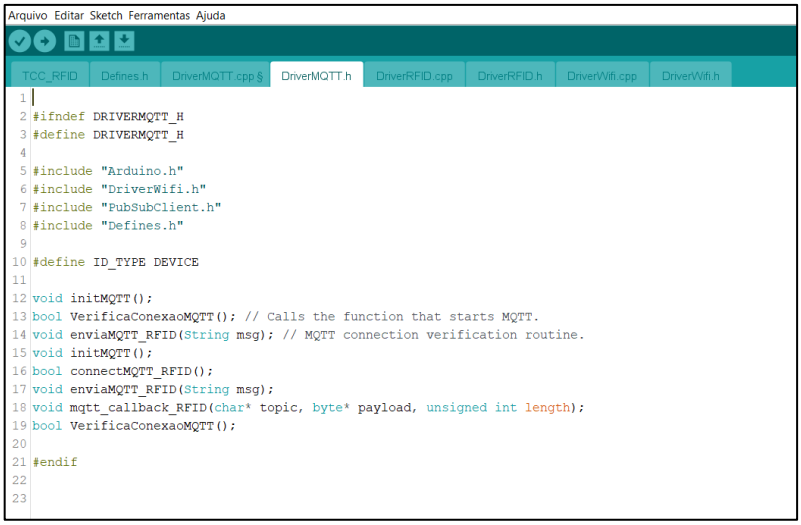
O segundo drive desenvolvido é referente ao protocolo MQTT. Para facilitar na implementação do sistema, foi utilizado um *broker* online, através de uma plataforma em nuvem MQTT gratuita e gerenciada pela HiveMQ Cloud que pode ser acessado através deste link *http://www.mqtt-dashboard.com/*, onde é recebido todas as mensagens e publicados para os clientes inscritos. Para ter acesso ao *broker* foram feitas as configurações dos parâmetros: o *host*, a porta, o tópico de publicação e o tópico de inscrição, todos inicializados no *driverMQTT.cpp*, como visto na Figura 19, para que o microcontrolador se conectasse ao MQTT online e os dispositivos pudessem se comunicar entre si. Na Figura 20, observa-se a inclusão das bibliotecas necessárias para a inicialização do *driver*.

Figura 19. Parte do código referentes a conexão do MQTT.



Fonte: Autoria Própria

Figura 20. Inclusão das bibliotecas referentes a conexão com o MQTT.



Fonte: Autoria Própria

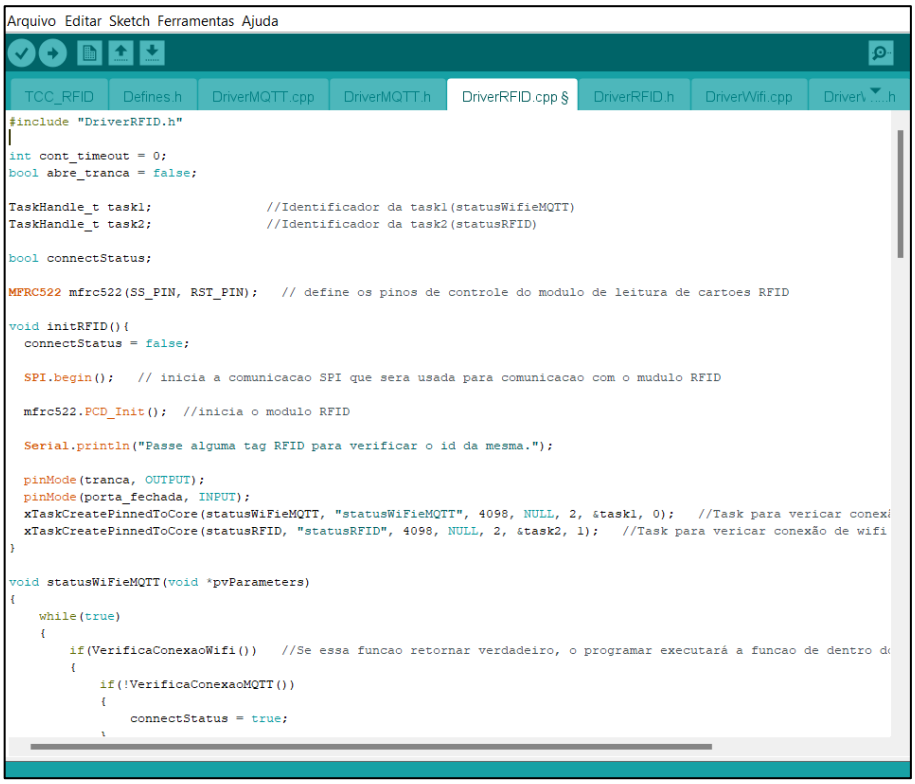
Posteriormente, foi desenvolvido o código do RFID onde são feitas as leituras dos cartões RFID e o processamento dos dados lidos pelo leitor RFID. Foram feitas a inclusão das bibliotecas necessárias para que o código funcione de maneira correta, como visto na Figura 21, e também parte do código desenvolvido pode ser visto na Figura 22. E nesta parte são lidas as informações contidas no cartão RFID através do leitor que está conectado ao ESP32 e então essas informações serão processadas pelo ESP32 e enviadas via MQTT. Dentro do código

desenvolvido foi adicionado um ID para que caso haja uma falha na comunicação com o MQTT ou com o banco de dados, possa liberar a tranca, garantindo a abertura do armário até que a conexão seja reestabelecida.

Figura 21. Inclusão das bibliotecas necessárias para o desenvolvimento do código do módulo RFID.

Fonte: Autoria Própria.

Figura 22. Código desenvolvido referente ao leitor RFID.

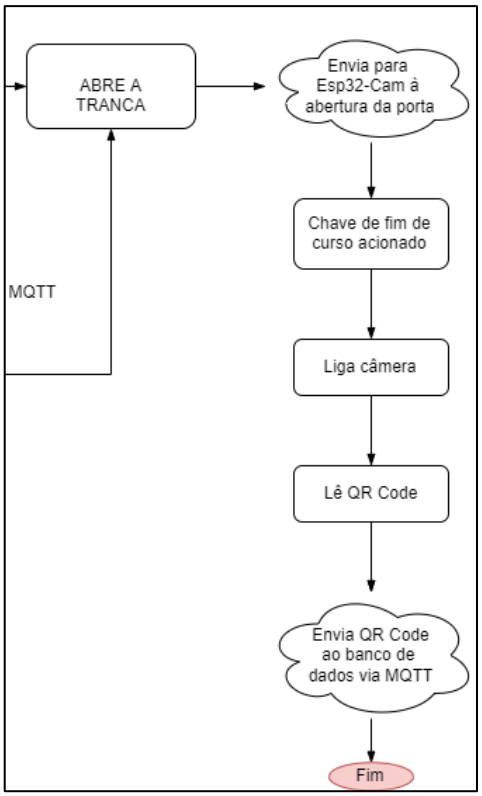


Fonte: Autoria Própria

3.2.2. O *firmware* do módulo QR Code

O fluxo de ações do módulo QR Code é apresentado na Figura 23. O módulo QR Code é o responsável pela leitura e decodificação dos QR Codes, o módulo já possui uma câmera acoplada juntamente com o microcontrolador ESP32. Como o módulo já vem conectado ao microcontrolador ESP32, é ele o responsável pelo envio dos textos existentes ao banco de dados via protocolo MQTT.

Figura 23. Parte do fluxograma referente ao módulo QR Code.



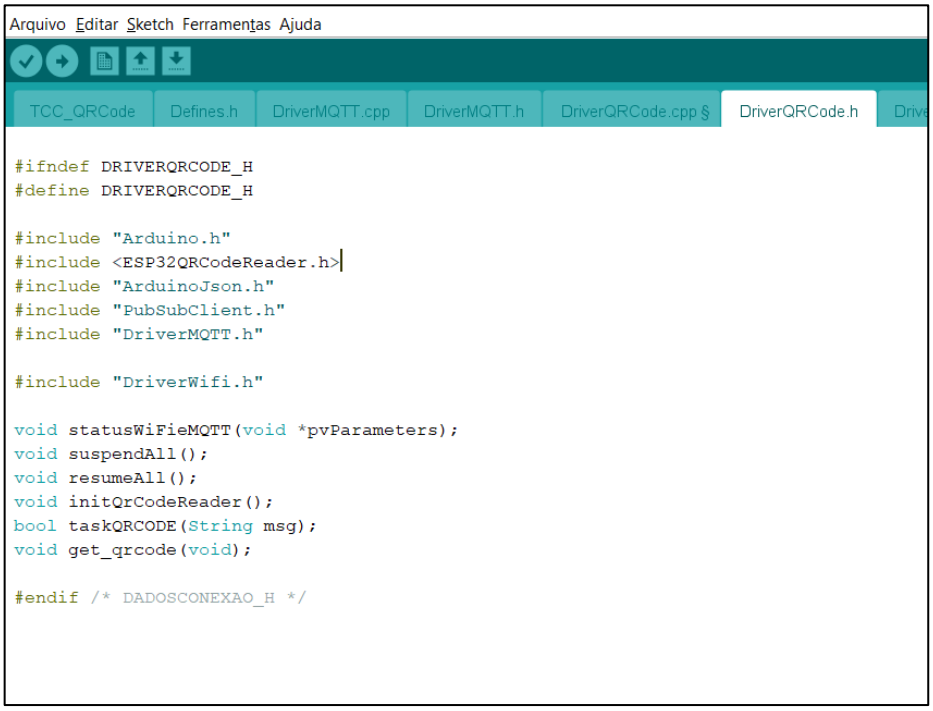
Fonte: Autoria Própria

O código do módulo QR Code também foi separado em *drivers,* da mesma forma que o módulo RFID. Dois dos drivers, o WI-FI e MQTT, também foram utilizados na arquitetura do módulo QR Code e eles estão descritos no tópico anterior.

Desta forma, foram adicionadas as bibliotecas necessárias para o desenvolvimento do código do módulo QR Code, como visto na Figura 24, onde são lidas as informações contidas nos códigos localizados abaixo de cada ferramenta, contendo uma informação do tipo texto,

que terá uma descrição pré-estabelecida. O firmware desenvolvido neste módulo só irá iniciar seu processo após o backend informar que houve a abertura da porta. A câmera esperará para começar a fazer a leitura, iniciando somente após a identificação do fechamento da porta, que neste caso será através do acionamento de uma GPIO associada a um sensor de fim de curso, e assim haverá a leitura dos possíveis QR Codes, caso haja a retirada de alguma ferramenta. Após a leitura a informação será transmitida ao banco de dados via MQTT, parte do código pode ser visualizada na Figura 25.

Figura 24. Inclusão das bibliotecas necessárias para o desenvolvimento do código do módulo QR Code.



Fonte: Autoria Própria

Figura 25. Parte do código desenvolvido para o módulo QR Code.

Fonte: Autoria Própria

3.3. DESENVOLVIMENTO DO PROTÓTIPO DOS MÓDULOS

Neste tópico é apresentado como foi realizada a implementação do projeto, abordando sobre a construção do circuito protótipo desenvolvido, onde são citados sobre os materiais utilizados, montagem e testes.

3.3.1. Módulo RFID

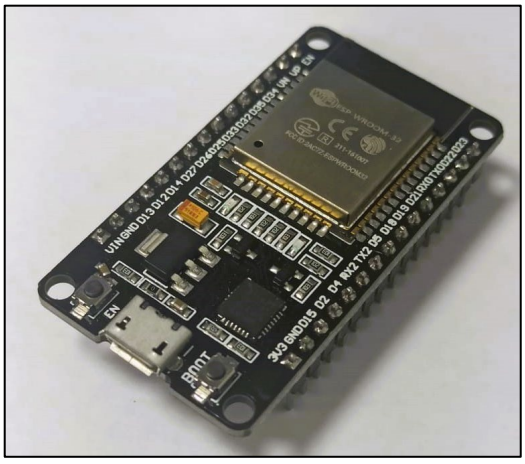
Para construir o circuito protótipo foi necessário o leitor RFID, o MFRC522, ilustrado na Figura 26, ele é utilizado na comunicação sem contato com alcance de 0 a 3 cm a uma frequência de 13,56 MHz, possui uma tensão de operação de 3,3 V, possui uma interface SPI (*Serial Peripheral Interface*) com uma taxa de transferência de até 10 Mbit/s.

Figura 26. Leitor MFRC522.

Fonte: Autoria Própria

O microcontrolador utilizado no projeto é o ESP32-DevKit-32D, ilustrado na Figura 27, sua tensão de alimentação pode ser de 3 V a 3,6 V, tensão de nível lógico é de 3,3 V, corrente de consumo típica é de 80 mA e máxima de 500 mA, conexão com rede *WI-FI* e *Bluetooth LE* (*Low Energy*/ Baixo consumo) 4.2 embutidos e é compatível com a *IDE* do Arduino.

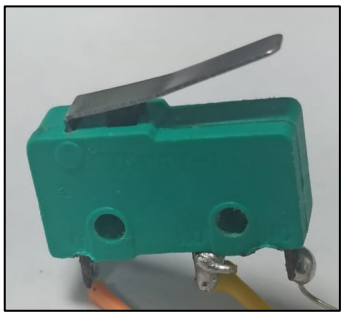
Figura 27. ESP32-DevKit-32D



Fonte: Autoria Própria

Foi utilizado também um sensor de fim de curso, como o mostrado na Figura 28, que é um dispositivo eletromecânico que consegue determinar que uma outra estrutura ligada ao seu eixo, chegou ao fim do seu campo de acionamento, ou seja, chegou ao fim do seu curso.

Figura 28. Sensor de fim de curso.

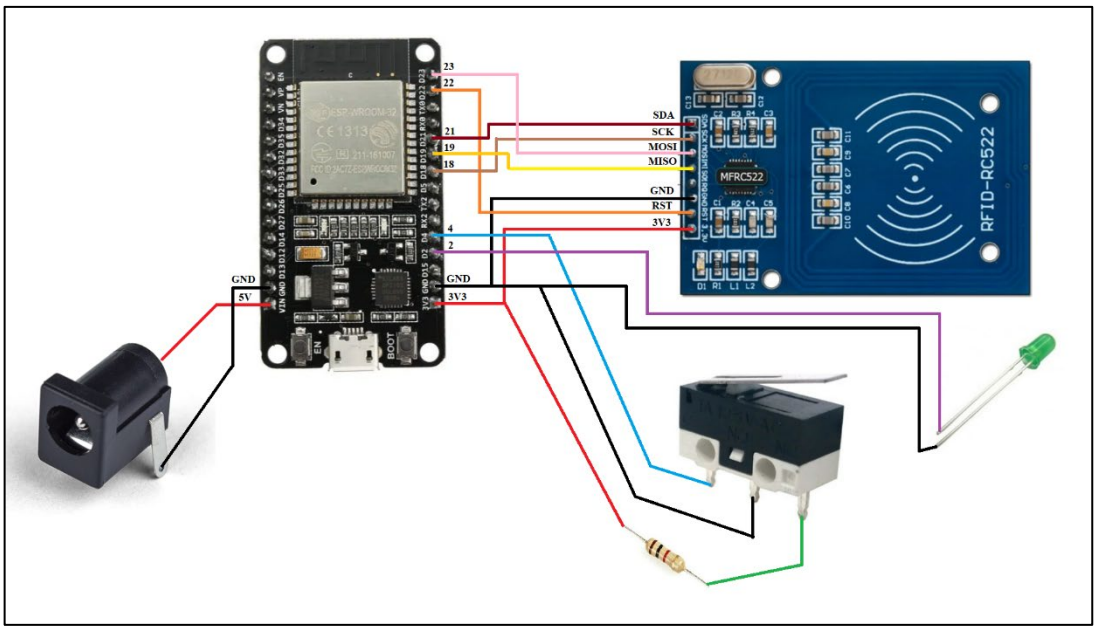


Fonte: Autoria Própria

O ESP32 é responsável por fazer o controle da leitura feita pelo leitor RFID, ativação e desativação do Led de indicação e sensor de fim de curso, logo estes periféricos estarão ligados ao ESP32.

Para o funcionamento do projeto é de fundamental importância saber em quais pinos do ESP32-DevKit-32D estarão conectados a esses periféricos, citados anteriormente. Portanto, na Figura 29 está ilustrada uma imagem que mostra a pinagem do ESP32 e os demais periféricos conectados, e as conexões entre os pinos deles foram ligadas conforme a Tabela 2. Foi ainda adicionado um conector Jack para conexões com plug P4 afim de conectar uma fonte externa de forma a alimentar todo o módulo.

Figura 29. Periféricos conectados ao ESP32.



Fonte: Autoria Própria

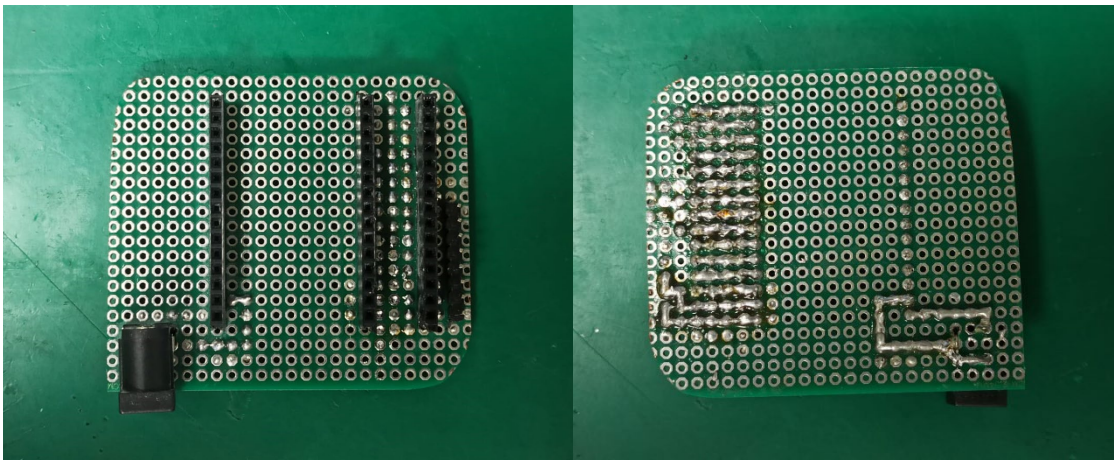
Tabela 2. Conexão dos pinos do ESP e periféricos.

Módulo ESP32	Módulo Leitor RFID
21	SDA
18	SCK
23	MOSI
19	MISO
Não conectado	IRQ
GND	GND
22	RST
3V3	3V3
	Sensor de Fim de Curso
4	C (Comum)
GND	NO (Normal Open)
3V3	NC (Normal Close)
	LED
2	Anodo
GND	Catodo
	Conector Jack P4 Fêmea
VIN	VIN
GND	GND

Fonte: Autoria Própria

Toda a estrutura foi montada em uma placa de circuito impresso ilhada dupla face, onde foram soldadas as barras de soquete *header* fêmea e macho, e também o conector Jack utilizando os seguintes itens: os componentes eletrônicos definidos anteriormente, pinça metálica, alicates, solda de estanho, estação de solda. Após a soldagem dos componentes eletrônicos, obteve-se pôr fim a PCI (Placa de Circuito Impresso) das Figuras 30a e 30b.

Figura 30. PCI montada.

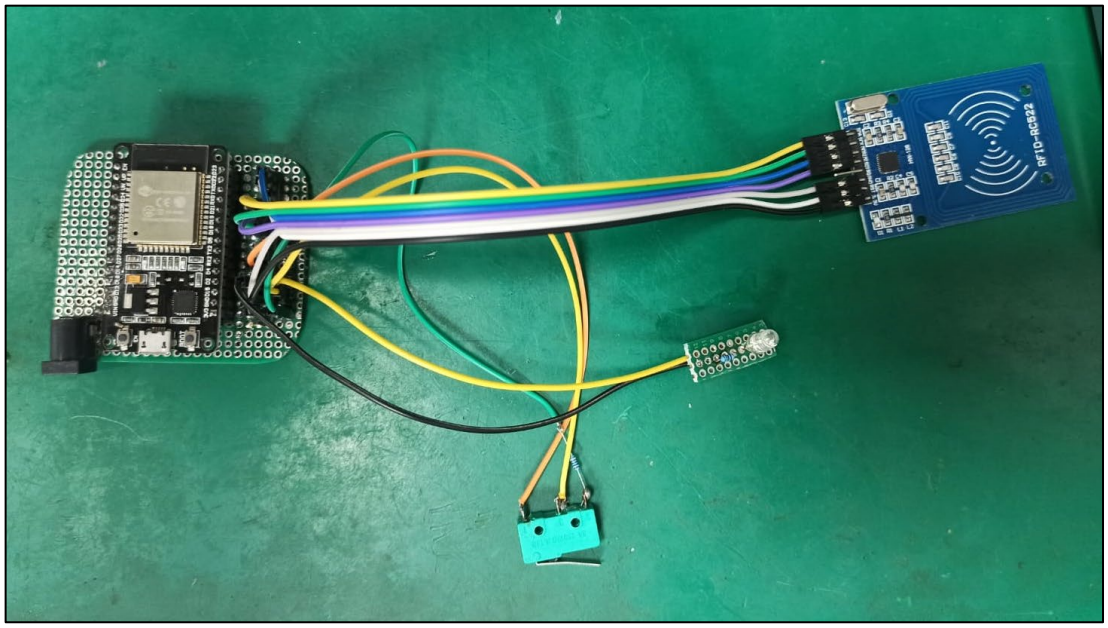


(a) (b)

Fonte: Autoria Própria

Finalizada a etapa de montagem, obtém-se a placa com as trilhas feitas com solda de acordo com a arquitetura apresentada do circuito, dando início a fixação do ESP32-DevKit-32D, e conexão dos jumpers para fazer a ligação com o módulo leitor RFID, sensor de fim de curso e Led, como visto na Figura 31.

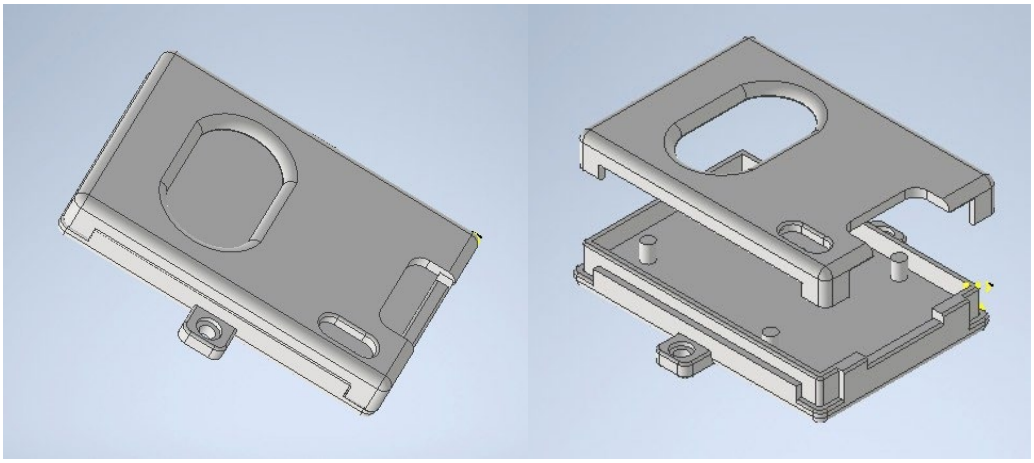
Figura 31. Módulo montado e conectado aos periféricos.



Fonte: Autoria Própria

Por fim foi feita a confecção da case para condicionar a placa de circuito e também para o leitor RFID, para que os mesmos não sofram danos físicos por processos mecânicos, tais como: quedas, contato humano e a interferência externa do ambiente. A case foi confeccionada em uma impressora de resina 3D. As Figuras 32a, 32b, 33a e 33b ilustram o fatiamento da case do módulo RFID e do leitor RFID, respectivamente. Nas Figuras 34a, 34b e 35a, 35b, mostram as cases já impressas e montados com o leitor e com o módulo.

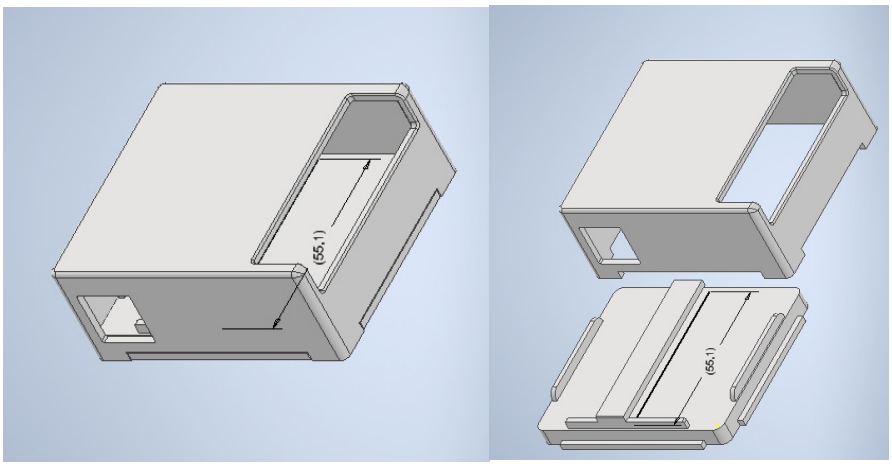
Figura 32. Fatiamento da case do leitor RFID.



(a) (b)

Fonte: Autoria Própria

Figura 33. Fatiamento da case do módulo RFID.



(a) (b)

Fonte: Autoria Própria

Figura 34. Módulo RFID montado na case.

(a) (b)

Fonte: Autoria Própria

Figura 35. Leitor montado na case impressa.

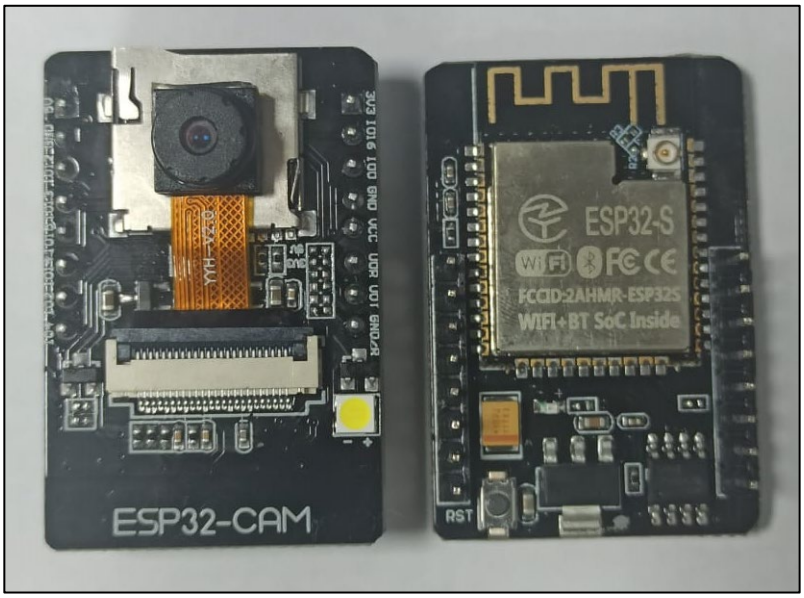
(a) (b)

Fonte: Autoria Própria

3.3.2. Módulo ESP32-CAM

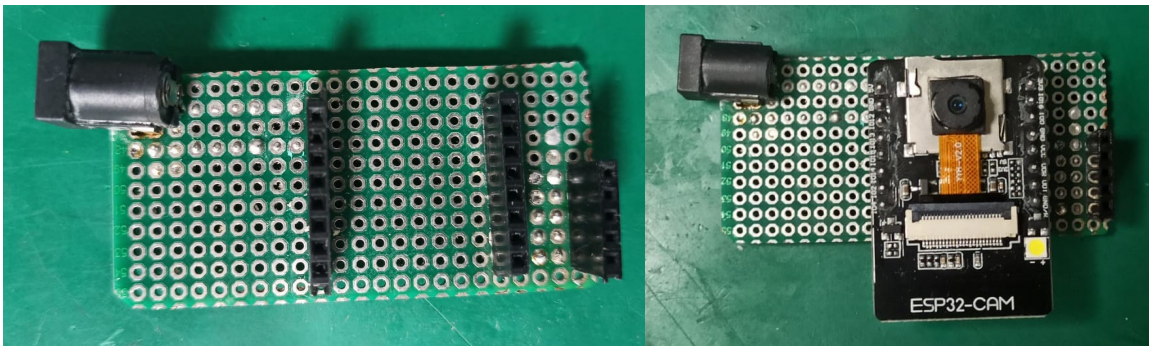
A construção desse módulo é mais simples pois o próprio já possui a câmera acoplada, como visto na Figura 36, que será a responsável pela captura dos QR Codes, desta forma foi soldadas as barras de soquete header fêmea na placa de circuito impresso ilhada para acoplar a módulo ESP32-CAM e também o conector Jack com plug P4 para que pudesse conectar uma fonte externa afim de alimentar o módulo externamente, como mostram as Figura 37a e 37b.

Figura 36. ESP32-Cam



Fonte: Autoria Própria

Figura 37. Módulo QR Code montado.

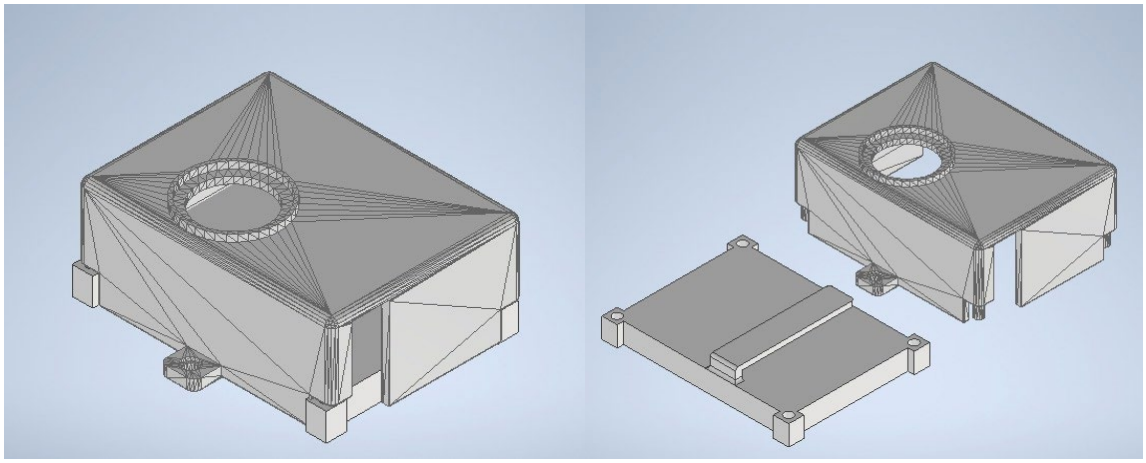


(a) (b)

Fonte: Autoria Própria

Com a finalização da montagem da placa onde ficará a ESP32-CAM. Além disso, foi confeccionado em uma impressora 3D a case de proteção do módulo QR Code, como é possível visualizar nas Figuras 38a e 38b. Após a soldagem dos *headers*, a placa foi inserida na case, conforme a Figura 39a e 39b.

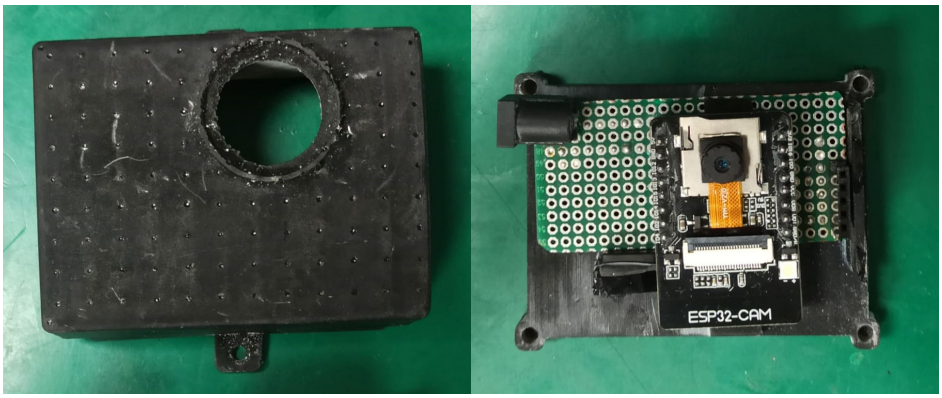
Figura 38. Fatiamento da case do módulo QR Code.



(a) (b)

Fonte: Autoria Própria

Figura 39 Módulo QR Code inserido dentro da case.



(a) (b)

Fonte: Autoria Própria

3.4. INTEGRAÇÃO COM O *FIRMWARE*.

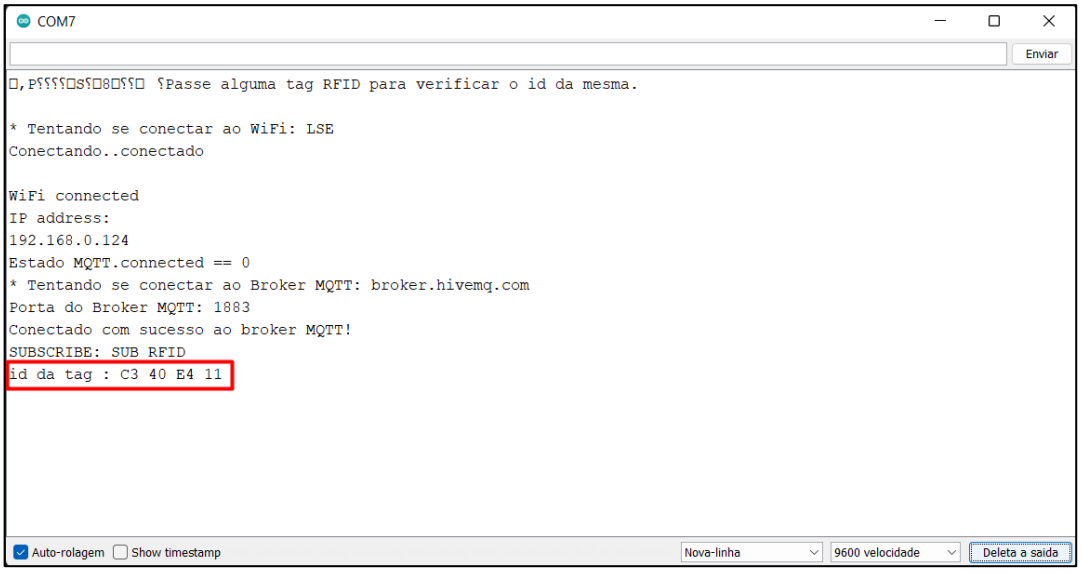
O sistema é dividido em duas etapas, cada um com uma resposta de acordo com as hipóteses estabelecidas. O primeiro dispositivo a atuar é o módulo RFID, responsável pela leitura dos cartões e chaveiros RFID. A segunda parte do sistema é a leitura do QR Code. Para o cumprimento de tais condições, foi necessário estabelecer uma comunicação entre os ESP32 de cada módulo, esta comunicação foi feita através do protocolo de comunicação *MQTT*, permitindo que os módulos enviem as mensagens através dos seus tópicos de publicação para

um cliente, sendo este cliente o banco de dados, que por sua vez realizará as ações enviando uma mensagem através dos tópicos onde ele estará inscrito.

3.4.1. Teste de leitura do RFID

A primeira parte do sistema faz a leitura do RFID, por tanto foi embarcado no ESP32 o código desenvolvido. No monitor serial do *IDE* Arduino, espera-se a inicialização microcontrolador, em seguida o módulo já está no modo a espera para identificar algum cartão ou chaveiro RFID. Após a inicialização, é feita a aproximação do cartão RFID, e na tela é exibido o *ID* único do cartão, como mostrado na Figura 40. As outras funcionalidades só são exibidas após a integração com o módulo QR Code.

Figura 40. Monitor Serial exibindo a leitura de um cartão.



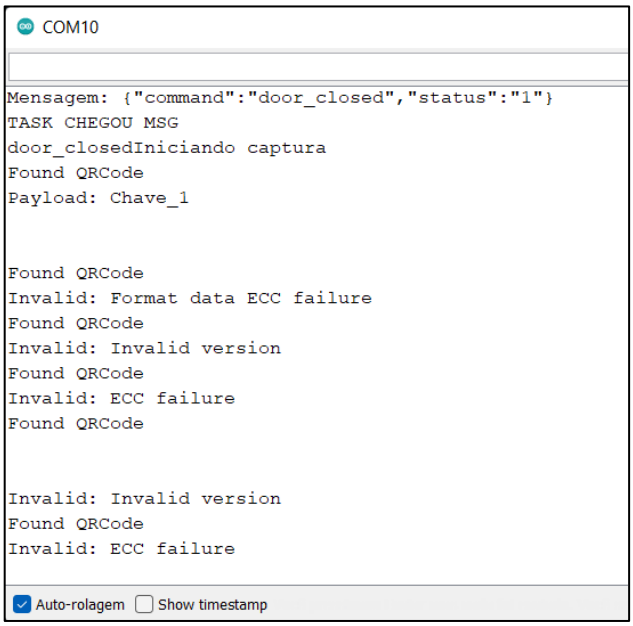
Fonte: Autoria Própria

3.4.2. Teste de identificação do QR Code

Para realizar o teste de identificação dos QR Code, foi embarcado o código desenvolvido no módulo. Em seguida, no monitor serial, esperou-se a visualização da inicialização do módulo, indicando suas conexões bem sucedidas com o *Wi-Fi* e *MQTT*. Para o módulo iniciar a leitura foi necessário enviar um comando que indica o fechamento da porta para a câmera iniciar a leitura dos QR Codes, o comando pode ser visto na Figura 41. Foram

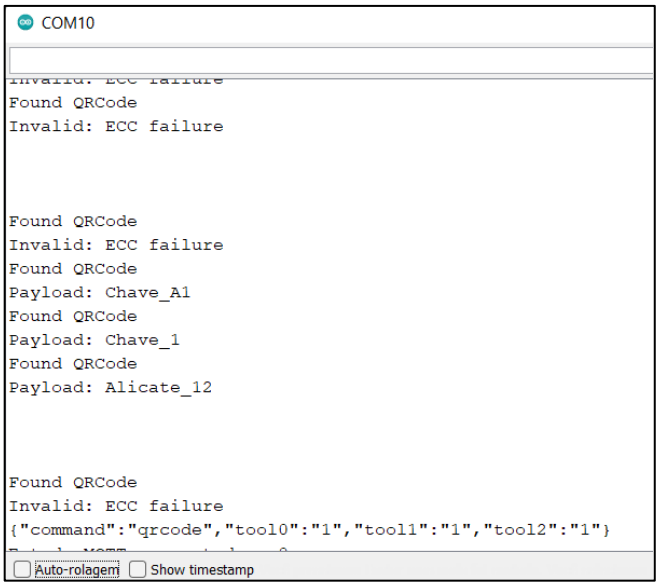
colocados três QR Codes para que pudessem ser identificados. Desta forma, pode-se observar que a câmera faz várias tentativas em um intervalo de 5 segundos. E conseguiu identificar os três QR Codes, o resultado das leituras é visto na Figura 42.

Figura 41. Monitor Serial exibindo as mensagens que chegam ao módulo QR Code.



Fonte: Autoria Própria

Figura 42. Monitor Serial exibindo a identificação dos QR Codes.



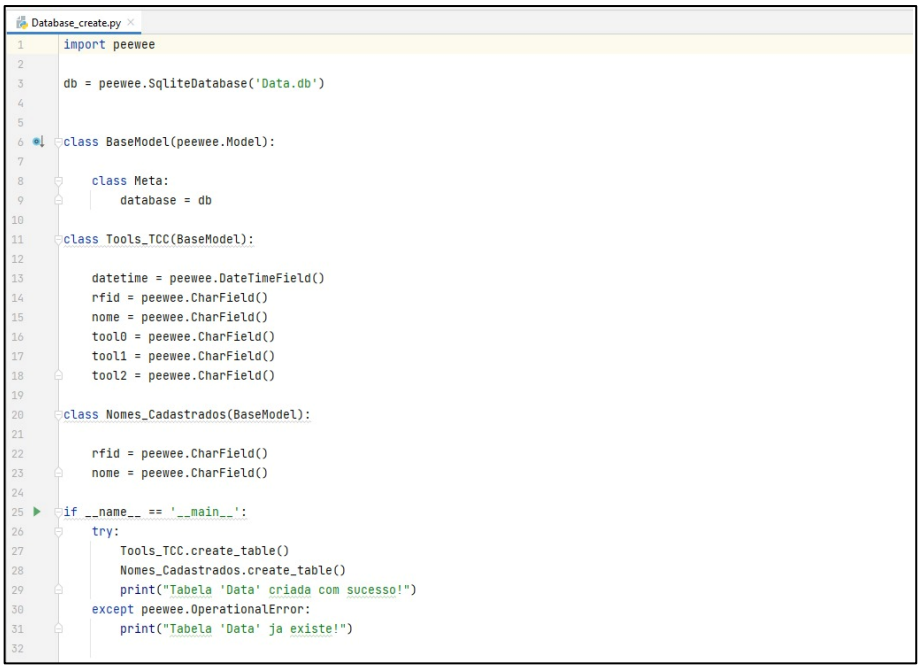
Fonte: Autoria Própria

Pode-se observar que é feito corretamente a leitura dos três QR Codes utilizados no teste, sendo assim validando o modulo leitor de QR Code.

3.5. DESENVOLVIMENTO DO BANCO DE DADOS

O desenvolvimento do banco de dados começou pela elaboração do *script* para o armazenamento dos dados coletados, desenvolvido em linguagem de programação *Python*, utilizando o *software PyCharm* 2022.1, neste *script* foi utilizado a biblioteca peewee. Para a criação da base de dados, primeiramente foi nomeado e criado o arquivo, e por se tratar de uma base de dados relacional, modelou-se as tabelas e os dados que posteriormente serão criados na base de dados. Para a implementação da base de dados, os dados enviados pelos módulos foram modelados de acordo com os dados enviados pelo *firmware*, nos quais são os *ID's* dos cartões e chaveiros RFID e as informações contidas nos QR Codes. Através do *script* "Database_create.py" foram modelados esses dados e gerada a base de dados, conforme demonstrado na Figura 43.

Figura 43. Código criado no *PyCharm* para a criação do banco de dados.



Fonte: Autoria Própria

Para se fazer o registro dentro do banco de dados, foi necessário também desenvolver um *script* que fizesse a interação entre o *firmware* e o banco de dados, sendo este o *backend*, que também foi desenvolvido no software PyCharm 2022.1. Este *script* é o responsável por autorizar ou não a abertura do armário, como se observa na Figura 44.

Figura 44. *Backend* desenvolvido no *PyCharm*.

Fonte: Autoria Própria

3.6. TESTES DE VALIDAÇÃO DO SISTEMA

Para validar o sistema foi feita a integração com os dois módulos juntamente com o banco de dados. Nesta etapa, foi analisado o comportamento do sistema perante aos cenários de teste aos quais foram submetidos, bem como as entradas e saída de dados do protótipo. Os testes foram divididos em quatro cenários diferentes:

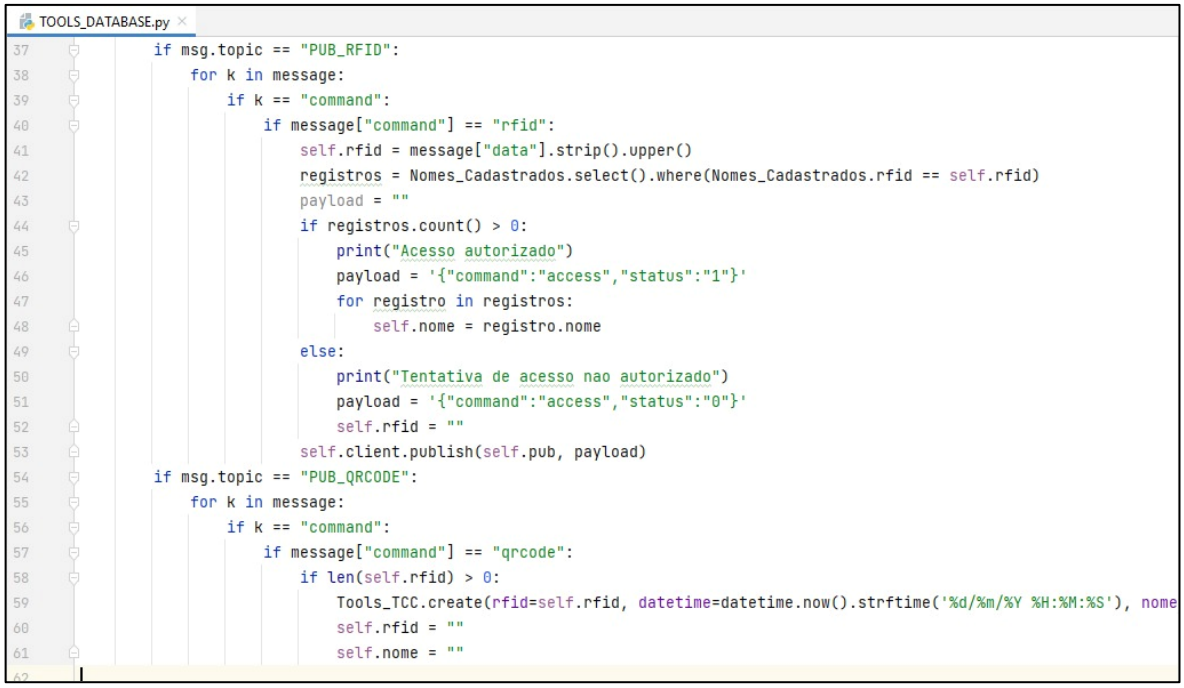
a) Identificação de pessoa autorizada;
b) Identificação de pessoa não autorizada;
c) Ferramenta retirada e devolvida;

3.6.1. Identificação de pessoa autorizada

Para validar esta parte do sistema, utilizou-se um cartão RFID já cadastrado no banco de dados. Na Figura 45, pode-se identificar que no código-fonte de autorização, que interage

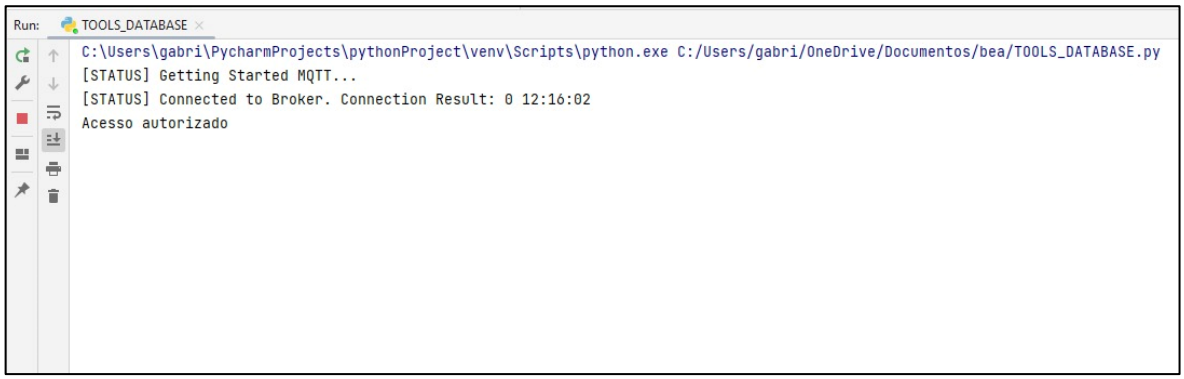
com o banco de dados, após a leitura do *ID* do cartão faz-se uma comparação com o banco de dados, e neste caso, é exibido no monitor serial, Figura 46, a mensagem “Acesso autorizado”. Neste momento o banco de dados, via *MQTT*, envia para o *subscriber* do RFID o comando ‘{“command”:“access”,“status”:“1”}’ onde o microcontrolador inscrito neste tópico recebe autorização para a abertura da tranca do armário, representado pelo led verde.

Figura 45. Código-fonte do *backend*.



Fonte: Autoria Própria

Figura 46. Monitor serial do *PyCharm* exibindo a mensagem “Acesso autorizado”.



Fonte: Autoria Própria

3.6.2. Identificação de pessoa não autorizada

Neste teste, foi utilizado um chaveiro RFID não cadastrado no banco de dados. De forma similar ao primeiro teste, foi feita a aproximação no leitor, no momento em que o código fonte de autorização não identificou o *ID* no banco de dados, foi exibida no monitor serial a mensagem “Tentativa de acesso não autorizado” como é mostrado na Figura 47, e este também enviou via *MQTT* ao microcontrolador o comando ‘{“command”:”access”,”status”:”0”}’ onde o microcontrolador inscrito neste tópico recebe a não autorização para a abertura da tranca do armário, desta forma o led não aciona.

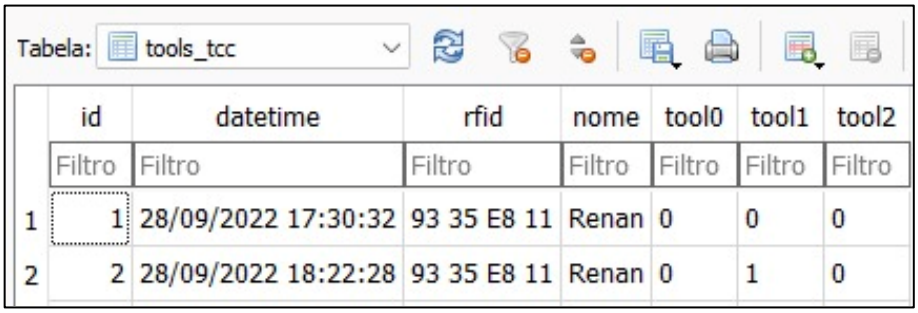
Figura 47. Monitor serial do *PyCharm* exibindo a mensagem “Tentativa de acesso não autorizado”.

Fonte: Autoria Própria

3.6.3. Ferramenta retirada e devolvida

Após o microcontrolador enviar o comando de autorização de abertura da tranca, o sistema fica à espera do comando de porta fechada ‘{“command”:“door_closed”,“status”:“1”}’ ser enviado. Ao receber a mensagem, o microcontrolador do módulo QR Code, que está inscrito no PUB_RFID, aciona a câmera para realizar a leitura do QR Code. Durante esse teste foram simulados a retirada de uma ferramenta. Deste modo, a câmera identificou um QR Code e após a leitura o microcontrolador enviou o pacote de informação ao banco de dados indicando a retirada de uma das ferramentas. O banco de dados, por sua vez registrou a retirada com nome e *ID* da pessoa que obteve acesso ao armário, com data e hora, como mostrado na Figura 48.

Figura 48. Registro do banco de dados da retirada de uma ferramenta.

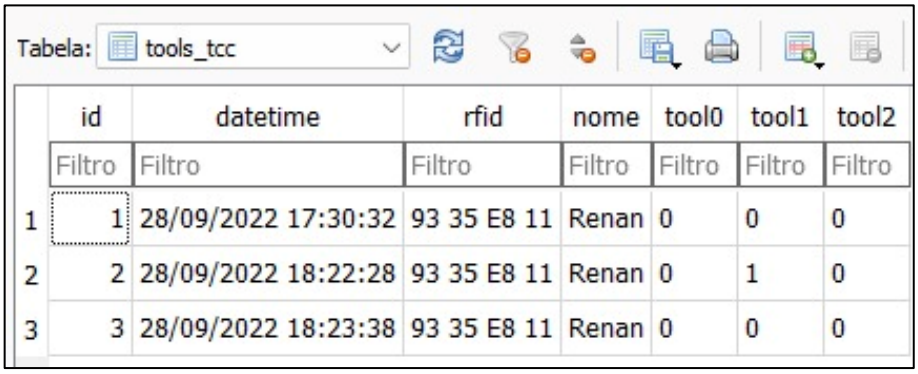


	id	datetime	rfid	nome	tool0	tool1	tool2
	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro
1	1	28/09/2022 17:30:32	93 35 E8 11	Renan	0	0	0
2	2	28/09/2022 18:22:28	93 35 E8 11	Renan	0	1	0

Fonte: Autoria Própria

Em seguida, o mesmo processo de abertura da tranca foi feito. E após o comando de porta fechada enviado, foi feita a leitura dos QR Code, e neste caso não foi encontrado nenhuma leitura, ou seja, houve a devolução da ferramenta. O banco de dados também fez o registro da devolução da ferramenta, como mostra a Figura 49.

Figura 49. Registro do banco de dados exibindo a devolução da ferramenta.



	id	datetime	rfid	nome	tool0	tool1	tool2
	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro
1	1	28/09/2022 17:30:32	93 35 E8 11	Renan	0	0	0
2	2	28/09/2022 18:22:28	93 35 E8 11	Renan	0	1	0
3	3	28/09/2022 18:23:38	93 35 E8 11	Renan	0	0	0

Fonte: Autoria Própria

4. ANÁLISE DOS RESULTADOS

O sistema foi instalado em um armário com três ferramentas, e cada ferramenta com um QR Code localizado abaixo delas, como mostrado na Figura 50. O módulo QR Code que contém a câmera responsável pela leitura, foi alocado acima das ferramentas a uma distância de 25 cm. Como o módulo RFID é dividida em duas partes foi alocado em dois locais diferentes. A primeira parte é o leitor e este foi alocado do lado de fora do armário juntamente com o led de indicação, que simula uma tranca, como visto na Figura 51. Na parte interna foi alocado a segunda parte do módulo que contém o microcontrolador e sensor de fim de curso, que identificará a abertura e fechamento da porta, como visto na Figura 52.

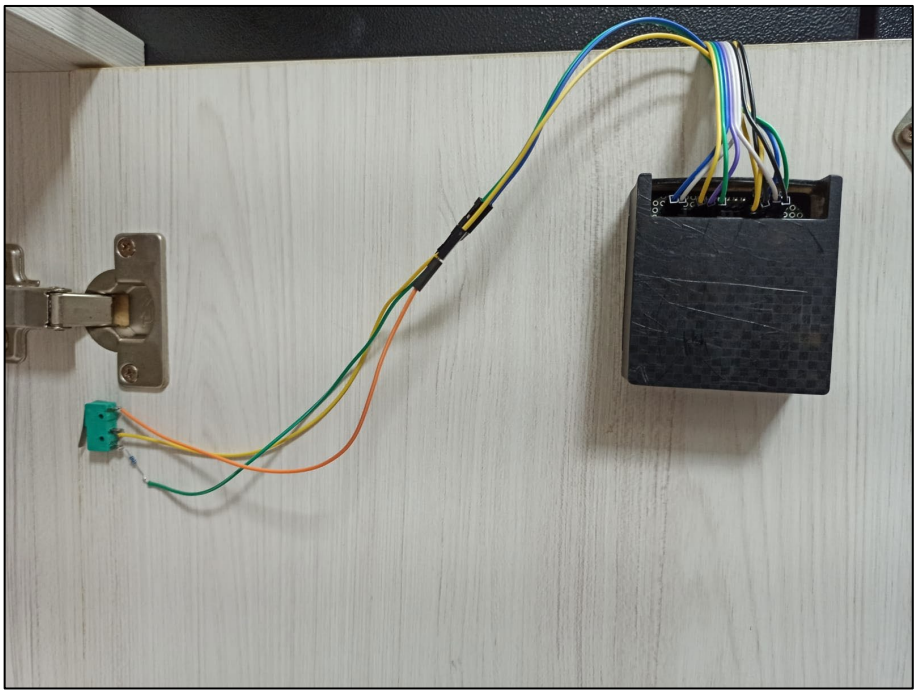
Figura 50. Ferramentas dentro do armário e câmera.

Fonte: Autoria Própria

Figura 51. Parte do módulo RFID alocado do lado de fora do armário.

Fonte: Autoria Própria

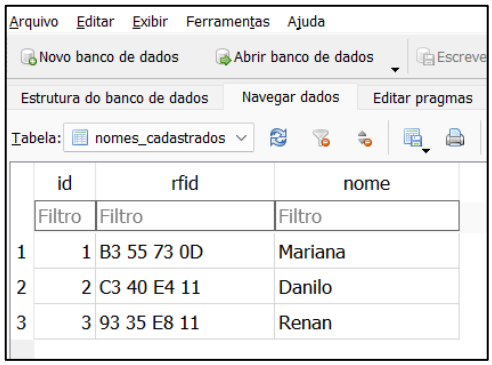
Figura 52. Parte do módulo RFID alocado na parte interna do armário.



Fonte: Autoria Própria

Com o sistema todo montado, foram feitos alguns processos de acesso e retirada de ferramentas, seguindo a arquitetura proposta. Primeiramente, no banco de dados foram adicionados três cadastros de *ID*'s, dois referentes aos cartões RFID e a um chaveiro RFID para que ao fazer os registros de acesso se pudesse associar os *ID*'s às pessoas e assim poder visualizar quem estaria acessando o armário. O cadastro dessas pessoas no banco de dados pode-se observar na Figura 53. Um chaveiro RFID não foi incluído ao banco de dados para que este simulasse uma *ID* não autorizado. Um outro cartão RFID que também não foi registrado é o chamado "chave-mestre", este utilizado apenas para caso o sistema perca a conexão com *WI-FI* ou com o *broker* e não bloqueie a tranca enquanto são feitas as tentativas de reconexão.

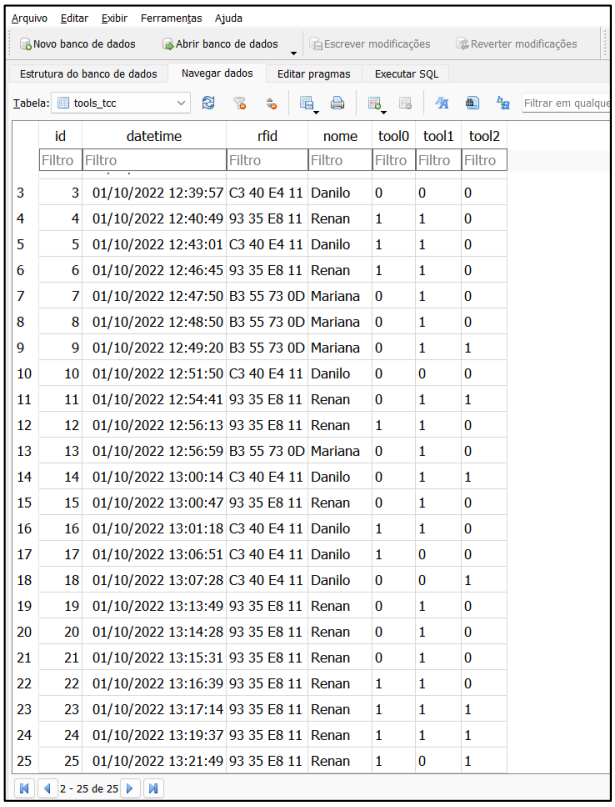
Figura 53. Cadastro feito no banco de dados.



Fonte: Autoria Própria

Após o cadastro dos nomes e *ID*'s, o sistema passou a funcionar e desta forma obteve-se um banco de dados com vários registros. Esses registros foram feitos durante uma hora, o banco de dados pode ser observado na Figura 54.

Figura 54. Registro feitos no banco de dados.



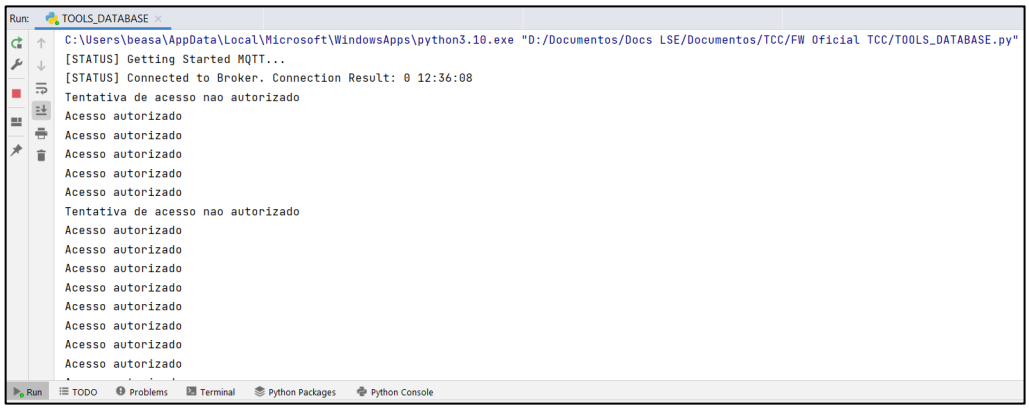
	id	datetime	rfid	nome	tool0	tool1	tool2
	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro
3	3	01/10/2022 12:39:57	C3 40 E4 11	Danilo	0	0	0
4	4	01/10/2022 12:40:49	93 35 E8 11	Renan	1	1	0
5	5	01/10/2022 12:43:01	C3 40 E4 11	Danilo	1	1	0
6	6	01/10/2022 12:46:45	93 35 E8 11	Renan	1	1	0
7	7	01/10/2022 12:47:50	B3 55 73 0D	Mariana	0	1	0
8	8	01/10/2022 12:48:50	B3 55 73 0D	Mariana	0	1	0
9	9	01/10/2022 12:49:20	B3 55 73 0D	Mariana	0	1	1
10	10	01/10/2022 12:51:50	C3 40 E4 11	Danilo	0	0	0
11	11	01/10/2022 12:54:41	93 35 E8 11	Renan	0	1	1
12	12	01/10/2022 12:56:13	93 35 E8 11	Renan	1	1	0
13	13	01/10/2022 12:56:59	B3 55 73 0D	Mariana	0	1	0
14	14	01/10/2022 13:00:14	C3 40 E4 11	Danilo	0	1	1
15	15	01/10/2022 13:00:47	93 35 E8 11	Renan	0	1	0
16	16	01/10/2022 13:01:18	C3 40 E4 11	Danilo	1	1	0
17	17	01/10/2022 13:06:51	C3 40 E4 11	Danilo	1	0	0
18	18	01/10/2022 13:07:28	C3 40 E4 11	Danilo	0	0	1
19	19	01/10/2022 13:13:49	93 35 E8 11	Renan	0	1	0
20	20	01/10/2022 13:14:28	93 35 E8 11	Renan	0	1	0
21	21	01/10/2022 13:15:31	93 35 E8 11	Renan	0	1	0
22	22	01/10/2022 13:16:39	93 35 E8 11	Renan	1	1	0
23	23	01/10/2022 13:17:14	93 35 E8 11	Renan	1	1	1
24	24	01/10/2022 13:19:37	93 35 E8 11	Renan	1	1	1
25	25	01/10/2022 13:21:49	93 35 E8 11	Renan	1	0	1

2 - 25 de 25

Fonte: Autoria Própria

Como o banco de dados é utilizado somente para registrar as ações de pessoas cadastradas, não se pode visualizar através dele as tentativas de acesso não autorizado. Porém, no monitor serial do *Pycharm* é possível observa que houve duas tentativas de acessos não autorizados durante o tempo de teste, como se observa na Figura 55, além de poder visualizar as tentativas em que o sistema autoriza a abertura do armário por pessoas cadastradas.

Figura 55. Monitor serial do *PyCharm* exibindo as mensagens do banco de dados.



Fonte: Autoria Própria

Quanto a análise do banco de dados, pode-se observar que enquanto uma ferramenta permanece dentro do armário ela recebe o valor 0 e este *status* só muda quando a ferramenta é retirada, recebendo o valor 1. Após a ferramenta ser retirada e seu *status* mudar para 1, ele permanece com esse valor até que seja devolvido, como visto na Figura 56. As ferramentas "tool0" e "tool1" são retiradas na linha 4, na linha 7 a ferramenta "tool0" é devolvida por outro usuário, diferente do usuário que a retirou. Porém, a ferramenta "tool1" permanece fora do armário. Com isso, se pode ter a conclusão de que esta ferramenta continua em posse da pessoa que a retirou, de acordo com o banco de dados.

Figura 56. Parte dos registros do banco de dados.

	id	datetime	rfid	nome	tool0	tool1	tool2
	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro
3	3	01/10/2022 12:39:57	C3 40 E4 11	Danilo	0	0	0
4	4	01/10/2022 12:40:49	93 35 E8 11	Renan	1	1	0
5	5	01/10/2022 12:43:01	C3 40 E4 11	Danilo	1	1	0
6	6	01/10/2022 12:46:45	93 35 E8 11	Renan	1	1	0
7	7	01/10/2022 12:47:50	B3 55 73 0D	Mariana	0	1	0
8	8	01/10/2022 12:48:50	B3 55 73 0D	Mariana	0	1	0

Fonte: Autoria Própria

Outro ponto importante já mencionado, é a altura que o módulo QR Code deve está alocado. Para encontrar a melhor distância foram feitos testes de altura para saber em que altura seria ideal para que a câmera pudesse ler todos os QR Codes existentes. As alturas testadas foram as seguintes: 10 cm, 15 cm, 20 cm, 25 cm e 30 cm.

Nas alturas de 10 cm e 15 cm houve uma maior quantidade de falhas nas leituras na maioria dos testes, e em poucos casos a câmera só conseguia coletar um QR Code. Na altura de 20 cm a câmera somente identificava dois QR Codes, apesar das falhas de leituras diminuírem. Na altura de 30 cm houve somente falhas nas leituras, então a partir desta altura não seria mais possível realizar a identificação correta dos QR Codes. Por isso a altura escolhida foi de 25 cm, pois nos testes é a altura em que houve maior quantidade de leituras corretas dos três QR Codes

utilizados. A Tabela 3 mostra de forma resumida os resultados obtidos dos testes de altura do módulo.

Tabela 3. Comparação entre as alturas testadas.

Altura (cm)	Muitas falhas na leitura	Leitura de 1 ou 2 QR Codes	Leitura de todos os QR Codes
10	X	X	
15	X	X	
20		X	
25			X
30	X		

Fonte: Autoria Própria

É importante lembrar que as leituras são feitas em um tempo de 5 segundos, porém não há um número exato de vezes em que o módulo QR Code irá fazer essas leituras. Outro ponto importante é que para confirmar a leitura de um ou mais QR Codes, a câmera deve lê-lo apenas uma vez para registrá-lo no banco de dados.

4.1. COMPARAÇÃO COM TRABALHOS RELACIONADOS.

Em comparação aos trabalhos citados anteriormente, pode-se ter duas principais comparações. A primeira delas é a utilização de uma outra tecnologia, o QR Code, para ajudar no monitoramento de objetos, pois é usado apenas a tecnologia de identificação por rádio frequência tanto na identificação de pessoas como na identificação de objetos.

A segunda comparação é a utilização do banco de dados. O trabalho proposto por Gross (2022), utilizou um banco de dados semelhante ao utilizado neste trabalho, já no trabalho proposto por AMORIM (2020), utilizou um banco de dados em nuvem, um serviço da *Google Cloud Platform* que pode proporcionar uma diminuição da infraestrutura, porém este é um serviço pago e é proporcional a quantidade de dados armazenados em nuvem, o que pode elevar o custo de um projeto.

CONCLUSÃO

Neste trabalho foi desenvolvido um sistema de controle de entradas e saídas de ferramentas onde pode-se identificar também quem realizou a retirada delas, utilizando as tecnologias RFID e QR Code. Para atingir os objetivos específicos, foram realizadas revisões bibliográficas sobre os assuntos referentes as tecnologias utilizadas, e sobre as principais estratégias e soluções de mercado utilizadas para controle de estoque. Além disso, foram abordadas pesquisas referentes às principais características do microcontrolador ESP32, e um estudo sobre os protocolos de comunicação MQTT e *WI-FI*, bem como o uso de bancos de dados.

A implementação da arquitetura do projeto foi realizada de forma satisfatória, de acordo com as etapas descritas no capítulo de materiais e métodos. Possibilitando realizar os testes de integração de sistema.

Referente a hipótese e com base nos resultados obtidos, conclui-se que é possível desenvolver um sistema de controle de entrada e saída de objetos utilizando as tecnologias RFID e QR Code, possuindo um banco de dados para armazenar informações referentes ao usuário de forma a registar os eventos com data e hora, possibilitando identificar quem foi a última pessoa a ter acesso aos objetos.

A escolha do módulo ESP32 foi essencial para o projeto, tendo em vista que ele conseguiu atender todas as necessidades no que diz respeito a processamento de dados recebidos, controle de abertura da tranca, conectar-se a uma rede *Wi-Fi* local, assim estabelecendo uma comunicação entre ele e o banco de dados através do MQTT, que permitiu que o mesmo enviasse e recebesse mensagens do banco de dados e processasse esses dados recebidos.

Apesar de ser possível o desenvolvimento do sistema, têm-se algumas limitações. A primeira a ser observada é a quantidade de QR Codes a serem lidos por uma câmera OV2640, pois esta é configurada para uma resolução em *SVGA*, ou seja, possui uma resolução de 800x600 pixels, não proporcionando qualidade no momento da leitura. Esta limitação se dá pelo processamento que o ESP32 possui, pois, o mesmo não permite que a resolução seja maior para esta aplicação. Contudo, para continuar a utilizar o ESP32-Cam uma sugestão seria a utilização de uma maior quantidade de câmeras para monitorar um armário com mais ferramentas. Outra sugestão, seria a troca do microcontrolador por um mais poderoso em

relação a processamento, ou mesmo um minicomputador como a *Raspberry Pi*, porém com um custo maior.

A segunda limitação a ser observada, foi a falta de uma interface para o cadastro e visualização dos dados. Como visto na implementação do projeto, o cadastro dos nomes e *ID*'s foi feito diretamente no banco de dados, tornando menos intuitivo. Caso o sistema tivesse um *frontend*, este possuiria telas onde os usuários que fossem utilizá-lo, poderiam usufruir de forma intuitiva tendo uma melhor aplicabilidade.

Portanto, para trabalhos futuros, a necessidade do desenvolvimento de uma interface completa contendo as informações das retiradas dos objetos e também a possibilidade do cadastro dos usuários sem que isto fosse feito diretamente no banco de dados se torna um ponto essencial para aprimorar o projeto, facilitando o acesso as informações.

REFERÊNCIAS BIBLIOGRÁFICAS

AI-THINKER TECHNOLOGY CO. (Shenzhen). **Módulo ESP32-CAM**. V1.0. [S. l.], 2017. Disponível em: <https://docs.ai-thinker.com/en/esp32-cam>. Acesso em: 7 maio 2022.

AMORIM, LUCAS ELISEU DE. **Sistema de Monitoramento do Uso dos Equipamentos no Laboratório de Hardware baseado em RFID**. 2020. Trabalho de Conclusão de Curso (Bacharelado em Engenharia da Computação) - Centro de Informática (CIN) da Universidade Federal de Pernambuco (UFPE), Recife, 2020.

KREIBICH, Jay. **Using SQLite: Small. Fast. Reliable. Choose Any Three**. [*S. l.*]: O'Reilly Media, Inc, 2010. 530 p. ISBN 978-0-596-52118-9.

BALLOU, Ronald H. **Logística empresarial: transportes administração de materiais distribuição física**. Tradução: Hugo T. Y. Yoshizaki. São Paulo: Atlas, 2010.

DANDARO, Fernando; MARTELLO, Leandro Lopes. **Planejamento e controle de estoque nas organizações**. Revista Gestão Industrial, v. 11, n. 2, 2015. DOI: 10.3895/gi.v11n2.2733. Disponível em: <https://periodicos.utfpr.edu.br/revis*tag*i/article/view/2733>. Acesso em: 14 abr. 2022.

ESPRESSIF SYSTEMS. **ESP32 Series Datasheet**. [S. l.], 2021. Disponível em: <https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf>. Acesso em: 1 maio 2021.

GROSS, L. H. A. da R. **Desenvolvimento de um sistema de controle e monitoramento de materiais do almoxarifado do Centro de Ensino e Instrução do CBPMPR, baseado na perspectiva de internet das coisas (IOT) com a utilização de microcontroladores ESP32, sensores de rádio frequência (RFID) e gerenciamento por software WEB**. Brazilian Journal of Development, [S. l.], v. 8, n. 3, p. 18495–18514, 2022. DOI: 10.34117/bjdv8n3-199. Disponível em: https://brazilianjournals.com/ojs/index.php/BRJD/article/view/45226. Acesso em: 28 sep. 2022.

KLAIR, Dheeraj K.; CHIN, Kwan-Wu; RAAD, Raad. **A Survey and Tutorial of RFID Anti-Collision Protocols**. IEEE Communications Surveys & Tutorials, [*s. l.*], v. 12, n. 3, p. 400-421, 26 abr. 2010. DOI 10.1109/SURV.2010.031810.00037. Disponível em: <https://ieeexplore.ieee.org/abstract/document/5455790/authors>. Acesso em: 7 maio 2022.

KUNIGAMI, Fabio Jun; OSÓRIO, Wislei Riuper. **GESTÃO NO CONTROLE DE ESTOQUE: ESTUDO DE CASO EM MONTADORA AUTOMOBILÍSTICA**. Revista Gestão Industrial, Ponta Grossa, p. 24-41, 3 dez. 2009. DOI 10.3895/S1808-04482009000400002. Disponível em: <https://periodicos.utfpr.edu.br/revis*tag*i/article/view/500>. Acesso em: 14 abr. 2022.

LI, Nan; BECERIK-GERBER, Burcin. **Life-cycle approach for implementing RFID technology in construction: Learning from academic and industry use cases**. Journal of Construction Engineering and Management, [*s. l.*], v. 137, n. 12, p. 1089-1098, 2011. DOI 10.1061/(ASCE)CO.1943-7862.0000376. Disponível em: <https://ascelibrary.org/doi/abs/10.1061/(ASCE)CO.1943-7862.0000376>. Acesso em: 7 maio 2022.

M. DOBKIN, Daniel. **THE RF IN RFID: UHF RFID in Practice**. 2. ed. USA: Newnes, 2013. ISBN 978-0-12-394583-9.

NOGUEIRA, LIDIANE DE PAULA et al. **ANÁLISE DA GESTÃO DE ESTOQUE DE UM COMÉRCIO VAREJISTA PET NO RAMO DE AQUARISMO**. Logística, [s. l.], ed. XV, 2018. Disponível em: https://www.aedb.br/seget/artigos2018.php?pag=269. Acesso em: 14 maio 2022.

OLIVEIRA, S. **Internet das Coisas com ESP8266, Arduino e Raspberry Pi**. São Paulo: Novatec, 2017.

SMART PLANILHAS LTDA (Brasil). **Ficha de Controle de Estoque em Excel**. [S. l.], 2020. Disponível em: <https://smartplanilhas.com.br/planilha-gratuita/ficha-de-controle-de-estoque-em-excel/>. Acesso em: 14 maio 2022.

STEVAN JUNIOR, S. L. **Internet das Coisas: Fundamentos e aplicações em Arduino e NodeMCU**. São Paulo: Saraiva, 2018.

TECLÓGICA. **4 BENEFÍCIOS DE UTILIZAR TECNOLOGIA IOT NA INDÚSTRIA.** Blumenau, SC, Brasil, 10 jun. 2021. Disponível em: <https://www.teclogica.com.br/iot-na-industria/>. Acesso em: 14 abr. 2022.

TIWARI, Sumit. **An introduction to QR code technology**. In: 2016 international conference on information technology (ICIT). IEEE, 2016. p. 39-44. Disponível em: <https://ieeexplore.ieee.org/abstract/document/7966807>. Acesso em: 14 abr. 2022.