

**INSTITUTO FEDERAL DE EDUCAÇÃO CIÊNCIA E TECNOLOGIA**  
**ENGENHARIA DE COMPUTAÇÃO**

**João Victor N. da Silva, Lohan Y. O. Pinheiro, Lucas Bivar F. Tavares,**  
**Lucas C. Vieira, Luís Henrique L. Santos**

**Uso dos microcontroladores ESP32, ESP32 CAM e Raspeberry PI 3 para automação de**  
**cancela eletrônica através do reconhecimento facial**

**Relatório Final - Internet of Things - Sistema de Segurança**

**CAMPINA GRANDE - PB**

**Mar/23**

## 1 INTRODUÇÃO

Em 15 de Novembro de 2022 a população mundial ultrapassou a marca de 8 bilhões. Com este espantoso número, questiona-se: com tamanha quantidade de pessoas, seria possível realizar o controle de tráfego em vias públicas e privadas? Uma das respostas para esse impasse vem sendo o uso da tecnologia. Por exemplo, mas que aplicado ao tráfego de veículos é o uso de semáforos e radares de velocidades inteligentes, que já fazem parte do cenário do tráfego do país, no entanto, ainda é necessário garantir esse mesmo nível de controle em ambientes mais restritos, como shows, locais onde haja pessoas com algum patamar mais elevado em sociedade, ambientes como empresas privadas, de maneira a controlar a corrente de pessoas, ou até mesmo para tornar conhecido a passagem de alguém por determinado ambiente.

O controle de tráfego é de suma importância, pois ajuda a manter aspectos sociais como segurança, organização e eficiência de fluxo. Entretanto, a demanda de fluxo é tão alta, que não pode ser vista de maneira manual, mas espera-se que seja feita de maneira automática, segura, além de precisar ser barata, eficiente e que seja de fácil uso. Neste contexto, os microcontroladores têm se destacado devido a estes dispositivos apresentarem componentes de um sistema microprocessado em um único invólucro, diminuindo sensivelmente o custo final do sistema. Outro motivo para a utilização de microcontroladores são suas dimensões, que proporcionam a fabricação de sistemas mais compactos. Exemplos disso são equipamentos eletrodomésticos, que executam tarefas pré-determinadas, porém com certo número de opções por parte do usuário.

Além de sistema de hardware que atenda a todos os requisitos citados, e que foram contemplados pelo microcontrolador, é necessário também uma lógica para que o hardware funcione. Há diversas formas de checagem de tráfego, seja através de verificações por cartões perfurados, impressões digitais ou detecção facial. Esta última, tem se destacado no mercado devido a sua potencialidade em segurança de pessoas, principalmente em órgãos públicos de alto escalão.

Portanto, para fins do projeto, adota-se como problema de investigação, entender como realizar o controle de tráfego de pessoas em locais públicos e/ou privados através do reconhecimento e detecção de faces, com finalidade didática e voltada aos alunos. A proposta

principal deste trabalho é, através dos microcontroladores ESP32 CAM, RASPBERRY PI 3 e ESP WROOM 32, criar um artefato que simule uma cancela eletrônica para controle de pessoas em espaços públicos e privados.

## **2 OBJETIVO**

Através deste trabalho, têm-se como objetivo entender os processos e etapas da construção de um sistema IoT. E este dispositivo, por sua vez, visa emular o controle do fluxo de pessoas através de uma cancela eletrônica inteligente usando reconhecimento facial.

### **2. 1. OBJETIVOS ESPECÍFICOS**

Para tornar factível o objetivo principal, elenca-se como objetivos específicos:

- a. Comunicar o ESP32 e a RASPBERRY através do protocolo MQTT enviando sinal de acesso liberado ou não, junto com o nome no payload se for liberado. Além disso, se for liberado, exibir mensagem com o nome da pessoa no display, acender led verde e movimentar micro servo motor. Se não for liberado, manter o led vermelho e pedir para o usuário se identificar;
- b. Comunicar o ESPCAM e a RASPBERRY através do protocolo HTTP, onde o RASPBERRY vai ficar solicitando os frames para o Webserver criado no ESPCAM.
- c. Comunicar o RASPBERRY com a Nuvem para através do protocolo HTTP enviar a foto para o backend identificar se a face identificada na câmera tem acesso liberado ou não no local;
- d. Implementar um algoritmo de reconhecimento facial para detecção de faces e a quem pertence a face;
- e. Implementar uma aplicação mobile para cadastrar novos usuários e exibir logs de acesso;
- f. Implementar rotas no backend na nuvem para cadastrar usuário, listar logs de acesso e identificar através da foto da face se o usuário tem acesso permitido ou não;

## **3 RECURSOS DAS PLACAS**

Módulo ESP32-CAM:

- Características:
  - 160 MHz de clock, podendo alcançar até 600 Milhões de instruções Dhrystone por segundo;
  - 520 KB de SRAM interna + 4 MBSRAM embutida na placa;
  - Protocolos UART, I2C e SPI;
  - Saídas digitais, PWM e DAC;
  - Suporte para câmeras OV2640 (2 MP);

- Suporte a comunicação WiFi tanto como estação (STA) quanto como (AP) e Bluetooth;
- 9 portas I/O;
- Consumo mínimo de corrente: 6 mA;
- Consumo máximo de corrente: 310 mA
- Pinout:
  - RXD e TXD: Pinos de comunicação serial;
  - ADC: Entradas analógicas;
  - CLK: Pinos que podem ser configurados como saída de clock;
  - Touch: Pinos conectados ao sensor de toque;
  - RTC: Pinos para comunicação com Real Time Clock;
  - VCC OUT: Saída de tensão (varia de acordo com a tensão de alimentação);
  - HSPI: saídas SPI (o H corresponde a Hardware).

### Raspberry PI 3

- Características:
  - Quad Core 1.2GHz Broadcom BCM2837 64bit CPU
  - 1GB RAM
  - BCM43438 wireless LAN and Bluetooth Low Energy (BLE) on board
  - 100 Base Ethernet
  - 40-pin extended GPIO
  - 4 USB 2 ports
  - 4 Pole stereo output and composite video port
  - HDMI

### ESP32-Wroom

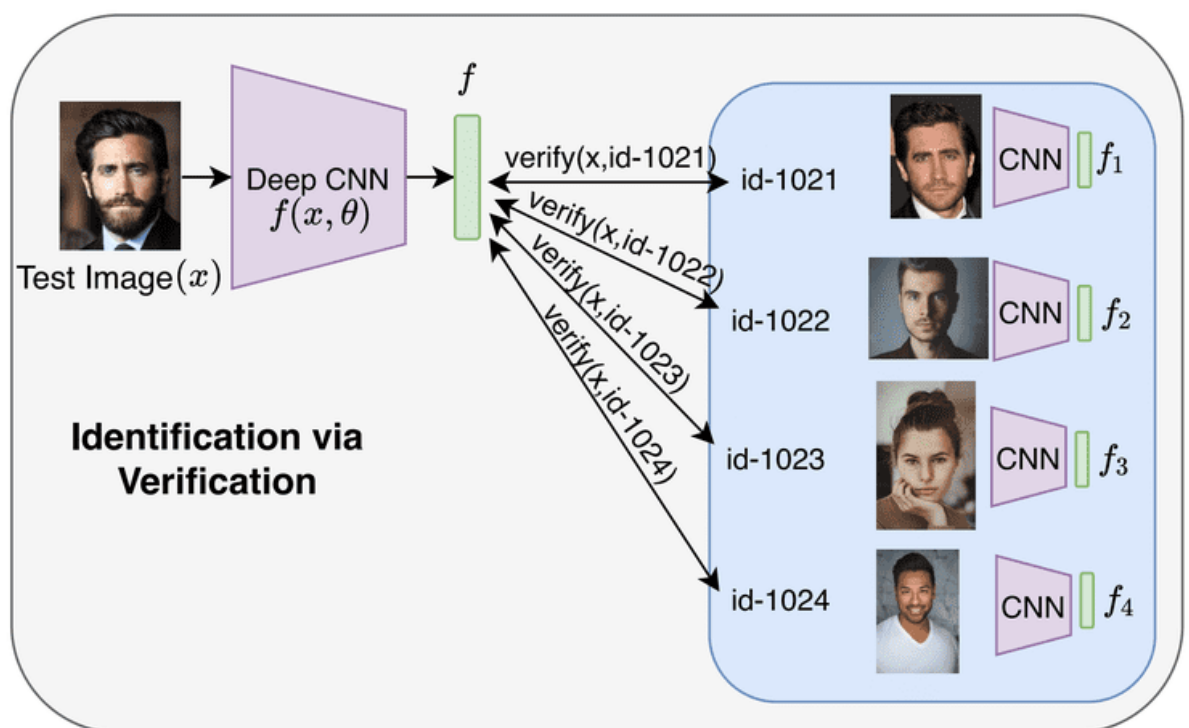
- Características:
  - CPU: Xtensa® Dual-Core 32-bit LX6
  - ROM: 448 KBytes
  - RAM: 520 Kbytes
  - Flash: 4 MB
  - Clock: 80 MHz~240 MHz (Ajustável)
  - WiFi 802.11 b/g/n: 2.4GHz~2.5 GHz

## 4 DO RECONHECIMENTO FACIAL

Com a finalidade de reconhecer faces, é feito o uso de uma Siamese Neural Network, que é definida como uma arquitetura de rede neural que recebe esse nome por ser composta por duas redes idênticas (gêmeas), que compartilham os mesmos pesos e arquitetura. Essa

estrutura foi proposta por Bromley em 1994 e é frequentemente utilizada em tarefas de reconhecimento de padrões, como reconhecimento de faces, detecção de objetos, entre outras.

A ideia por trás da Siamese Neural Network é comparar dois objetos ou imagens e determinar se eles são semelhantes ou diferentes. Para isso, a rede recebe como entrada duas imagens ou representações de objetos e as processa em paralelo pelas duas redes gêmeas. As saídas das duas redes são então comparadas para determinar a similaridade entre as entradas, dessa forma, não há a necessidade de re-treinar o modelo a cada nova face adicionada, economizando processamento e por isso, sendo escolhida como arquitetura para a detecção facial.



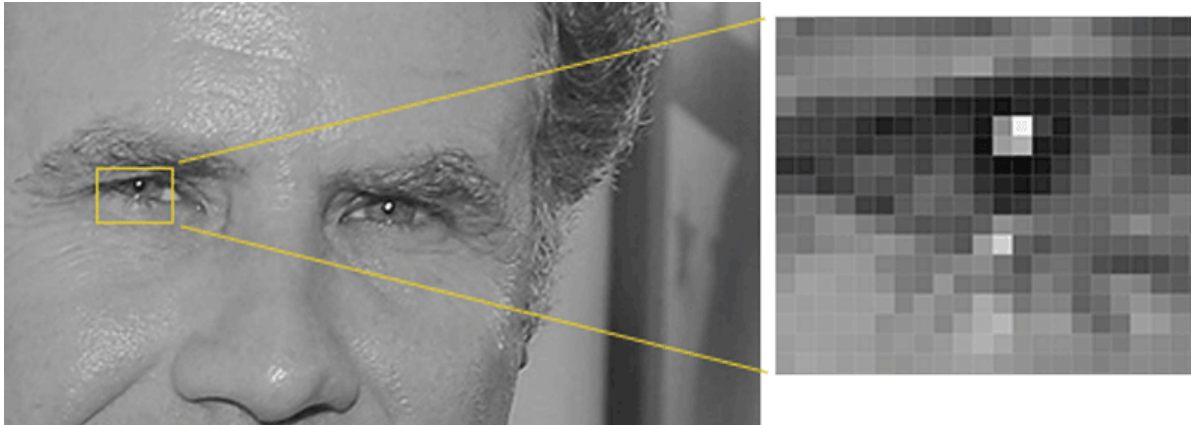
Funcionamento da rede neural Siamese Neural Network

Entretanto, a construção e treinamento do modelo utilizando esse tipo de rede neural mostrou-se custoso computacionalmente, tomando um longo tempo de treinamento. Para contornar o problema, utilizou-se então a API Face Recognition. A API possui uma acurácia de 99.38% e também utiliza redes neurais, assim como a rede neural siamesa.

O algoritmo construído na API funciona de maneira diferente, conforme descrito abaixo:

## 1º Passo) Identificar uma face em uma imagem

Para identificar um rosto em uma imagem, é aplicado o método de Histogramas de Gradientes Orientados (HOG). Tornamos a imagem preta e branca e em seguida observa-se cada pixel um por um, e para cada um desses pixels observador, identifica-se quais são os pixels que o cercam diretamente



O objetivo neste momento é descobrir o quão escuro é o pixel atual em comparação com os pixels ao seu redor, e então, desenhar uma seta mostrando em que direção a imagem está ficando mais escura. No final, ao aplicar todo este processo em cada um dos pixels da imagem, teremos que cada um dos pixels será substituído por um seta, chamadas de gradientes.

Após termos os gradientes de cada pixel, quebra-se a imagem em quadrados de 16x16 pixels e para cada quadrado, conta-se os gradientes apontando para cada direção. As setas com mais quantidade substituirão o quadro na imagem. Para encontrar rostos nesta imagem HOG, tudo o que precisamos fazer é encontrar a parte de nossa imagem que se parece mais com um padrão HOG conhecido que foi extraído de vários outros rostos de treinamento.

## 2º Passo) Posing and Projecting Faces

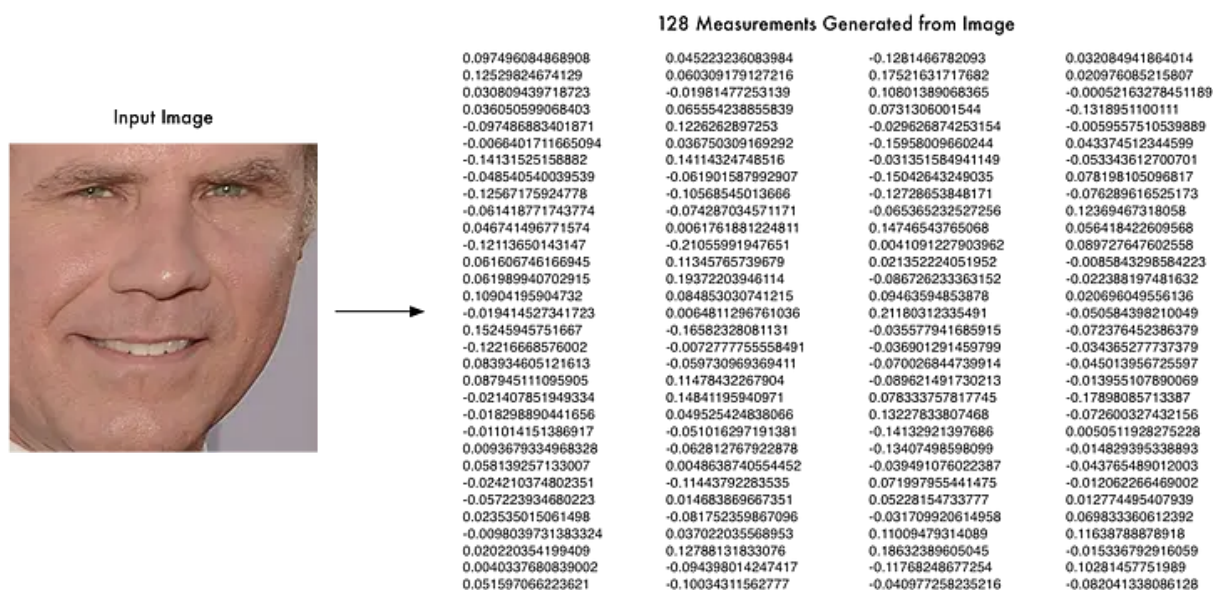
É utilizado o algoritmo de Estimativa de Ponto de Referência, que consiste em selecionar 68 pontos específicos do rosto e então treinar um algoritmo de machine learning capaz de identificar cada um destes pontos.

### 3º Passo) Encoding Faces

Para esta etapa, é treinado um modelo de rede neural convolucional profunda a fim de termos uma maneira mais confiável de medir um rosto. Deve-se entender que a medida de um rosto são medidas básicas, como tamanho da orelha, tamanho do nariz, cor do olho, etc. O trabalho de identificação dessas características é passada a essa rede neural, com 3 etapas:

1. Carregar uma imagem de um rosto para treinamento de uma pessoas conhecida;
2. Carregar uma outra foto da pessoa conhecida;
3. Carregar uma foto de uma pessoa totalmente desconhecida;

Após o treinamento, aplicando o algoritmo em uma face obtemos um vetor de 128 medidas, conforme a foto abaixo



Para cada uma das imagens teremos esse vetor com 128 valores. Para cada uma das 3 imagens é aplicada o algoritmo de rede neural, obtendo-se diferentes vetores com 128 medidas de face. Por fim, basta aplicar um algoritmo de classificação.

### 4º Passo) Encontrar o nome da pessoa a partir da codificação

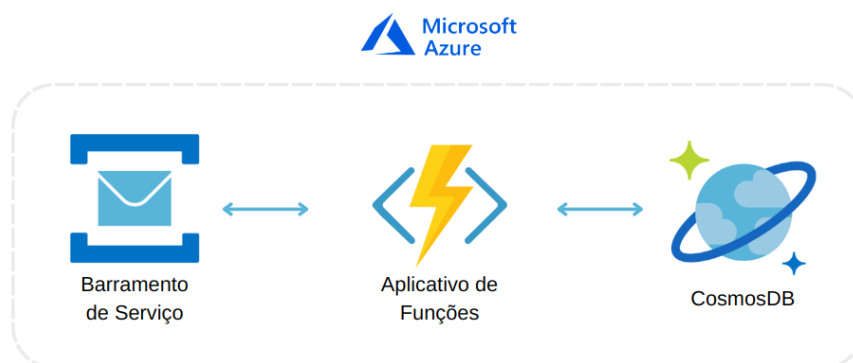
É utilizado um algoritmo de classificação *Support Vector Machine* (SVM). Treina-se o algoritmo de classificação e em milissegundos temos o reconhecimento de faces.

## 5 DOS REQUISITOS DO PROJETO

Quantidade	Descrição
1	ESP32 CAM
1	ESP WROOM 32
1	RASPBERRY PI 3 MODEL B
1	SERVIDOR AZURE
1	APP MOBILE
~	JUMPERS
2	LEDs 3mm
1	MICRO SERVO MOTOR

## 6 DA APLICAÇÃO BACKEND E MOBILE

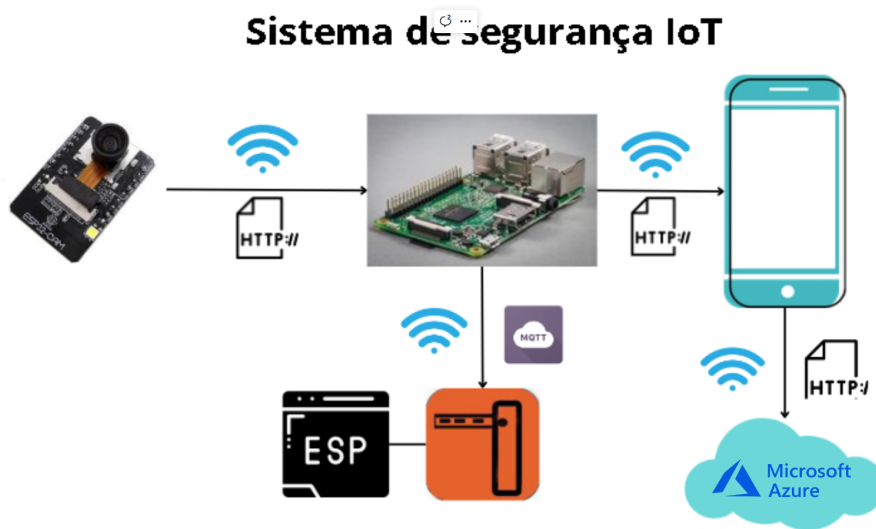
A arquitetura da aplicação será provisionada utilizando o serviço de computação em nuvem da Microsoft, a Azure. Tendo em vista a baixa complexidade de código no lado do servidor, serão utilizados poucos recursos, os quais são: Aplicativo de funções, para hospedar o código principal utilizando a tecnologia *serveless*, CosmosDB, banco de dados NoSQL altamente escalável e performático, e Barramento de Serviço, utilizado para gerenciar as filas de mensageria da aplicação. A aplicação ficará responsável por validar as imagens enviadas de cada usuário e retornar uma resposta informando se o acesso individual foi liberado ou bloqueado.



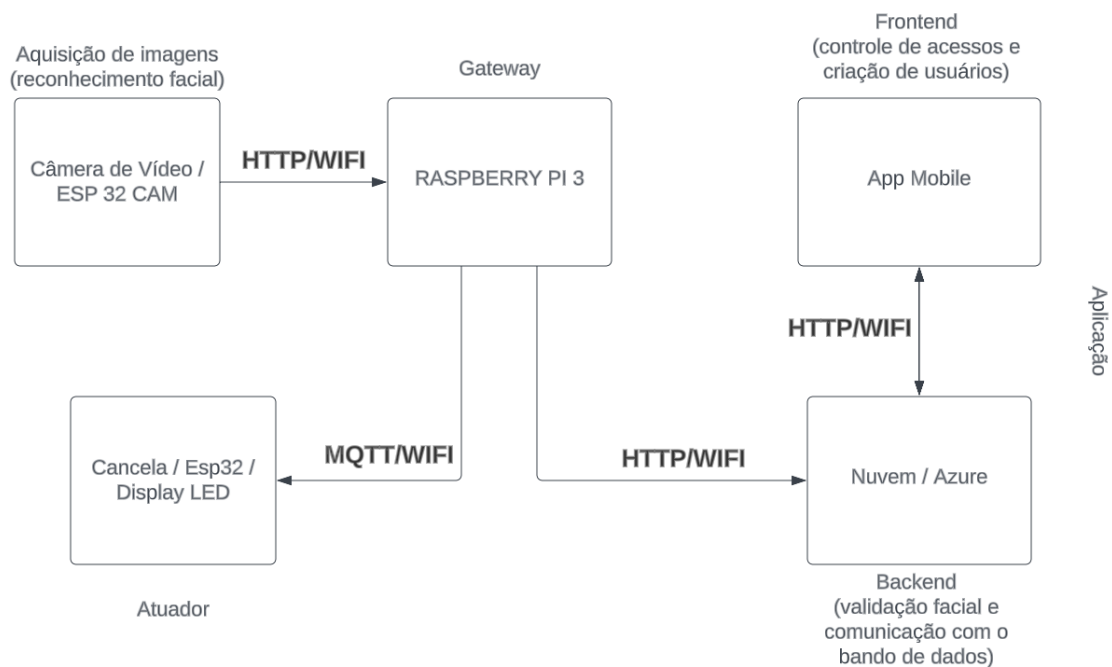
Além dos recursos em nuvem, também será desenvolvido um aplicativo para dispositivos móveis com o intuito de fazer o gerenciamento do sistema, seja para cadastrar novos clientes na base de dados, ou para analisar os *logs* de acesso individuais.



## 7 ILUSTRAÇÃO DO SISTEMA



## 8 DIAGRAMA DE BLOCOS



## 9 CONCLUSÃO

O presente estudo buscou criar uma tecnologia capaz de realizar a segurança de locais privados por meio do reconhecimento facial fazendo-se uso de técnicas de deep learning e machine learning, uma ESPCAM para a captura de vídeo, e da automação de uma cancela controlada pela ESP32 que recebe as ações enviadas pelo RASPBERRY PI 3 através do MQTT.

Dado o objetivo principal e os objetivos específicos, entende-se que o projeto atende de maneira completa seus requisitos, possibilitando o uso em produção.

## 10 REFERÊNCIAS BIBLIOGRÁFICAS

TRIGGS, Bill; DALAL, Navneet. **Histograms of Oriented Gradients for Human Detection**. Disponível em: <https://lear.inrialpes.fr/people/triggs/pubs/Dalal-cvpr05.pdf>. Acesso em: 16 jun. 2023.

PORTAL MEDIUM. **One-Shot Learning (Part 2/2): Facial Recognition Using a Siamese Network**. Disponível em: <https://heartbeat.comet.ml/one-shot-learning-part-2-2-facial-recognition-using-a-siamese-network-5aee53196255#:~:text=Facial%20recognition%20using%20the%20siamese%20network&text=Both%20the%20networks%20that%20return,should%20show%20a%20larger%20distancehttps://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html>. Acesso em: 16 jun. 2023.

PORTAL HOW TO ELECTRONICS. **ESP32 CAM Based Face & Eyes Recognition System**. Disponível em: <https://how2electronics.com/esp32-cam-based-face-eyes-recognition-system/>. Acesso em: 16 jun. 2023.

PORTAL MOSQUITTO. **Mosquito man page**. Disponível em: <https://mosquitto.org/man/mosquitto-8.html>. Acesso em: 16 jun. 2023.

PORTAL ESP-IDF *Programming Guide. API Reference. Application Protocols. ESP-MQTT. Configuration*. Disponível em: <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/protocols/mqtt.html#configuration>. Acesso em: 16 jun. 2023.

PORTAL SWA. **ESP32 MQTT client: Publish and Subscribe. HiveMQ and BME280 example**. Disponível em: <https://www.survivingwithandroid.com/esp32-mqtt-client-publish-and-subscribe/>. Acesso em: 16 jun. 2023.

PORTAL *STEVE'S GUIDE INTERNET*. ***Mosquitto Username and Password Authentication -Configuration and Testing***. Disponível em:

<http://www.steves-internet-guide.com/mqtt-username-password-example/>. Acesso em: 16 jun. 2023.

KAZEMI, Vahid; SULLIVAN Josephine. **One Millisecond Face Alignment with an Ensemble of Regression Trees**. Disponível em:

<https://www.csc.kth.se/~vahidk/papers/KazemiCVPR14.pdf>. Acesso em: 16 jun. 2023.