

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE ELETRÔNICA

THIAGO SEIJI MIYASAWA
ROBSON DE ALMEIDA SANTANA JUNIOR
ADRYAN CASTRO FERES

FECHADURA MODULAR INTELIGENTE:

RELATÓRIO FINAL

CURITIBA

2023

THIAGO SEIJI MIYASAWA
ROBSON DE ALMEIDA SANTANA JUNIOR
ADRYAN CASTRO FERES

FECHADURA MODULAR INTELIGENTE:

modular smart lock:

Relatório final apresentado(a) como requisito para obtenção do título(gra) de Bacharel em Engenharia de Computação, do DEPARTAMENTO ACADÊMICO DE ELETRÔNICA, da Universidade Tecnológica Federal do Paraná (UTFPR).

Ronnier Frates Rohrich: Prof(a). Dr(a).
Ronnier Frates Rohrich)

CURITIBA

2023



[4.0 Internacional](https://creativecommons.org/licenses/by/4.0/)

Esta licença permite compartilhamento, remixe, adaptação e criação a partir do trabalho, mesmo para fins comerciais, desde que sejam atribuídos créditos ao(s) autor(es).

Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.

RESUMO

MIYASAWA, thiago seiji; SANTANA, Robson de Almeida; FERES, Adryan Castro.

Fechadura modular inteligente. 2023. 18 f. Relatório final (Bacharelado em Engenharia de Computação) – Universidade Tecnológica Federal do Paraná. Curitiba, 2023.

Este trabalho tem como objetivo desenvolver uma fechadura eletrônica controlada por um microcontrolador, que oferece três tipos de acesso: login e senha, reconhecimento facial e código QR. Através do uso de microcontroladores, foi construída a estrutura física da fechadura e um banco de dados desenvolvido para armazenar as informações de acesso dos usuários. As informações recebidas pela fechadura são analisadas e comparadas com os dados do banco de dados. Além disso, o trabalho propõe a implementação de uma interface web para atualização do banco de dados, monitoramento do acesso à fechadura e facilitar o cadastro de novos usuários.

Ao final do trabalho, foi obtido um protótipo funcional de uma fechadura eletrônica inteligente, que oferece diferentes formas de acesso e é controlada por meio de microcontroladores. Essa solução visa aumentar a comodidade e a segurança dos usuários, tornando o processo de abertura de portas mais automatizadas.

Palavras-chave: Segurança. Reconhecimento Facial. Interface Web. Automatização. Integração.

SUMÁRIO

1	INTRODUÇÃO	4
1.1	OBJETIVOS	4
1.2	MARCOS	4
2	HARDWARE	5
2.1	COMPONENTES	5
2.2	UTILIZAÇÃO	6
3	SOFTWARE	7
3.1	A FECHADURA	7
3.2	PROTOCOLO DE COMUNICAÇÃO	7
3.3	FUNCIONAMENTO DO SOFTWARE	9
3.4	RECONHECIMENTO FACIAL	10
3.5	APLICAÇÃO WEB	11
4	RESULTADOS	15
4.1	HARDWARE	15
4.2	SOFTWARE	16
5	CONCLUSÃO	17
	REFERÊNCIAS	18

1 INTRODUÇÃO

Em um mundo cada vez mais automatizado, muitos objetos estão passando por transformações para permitir seu controle eletrônico, proporcionando maior comodidade aos usuários. Nesse contexto, o projeto em questão se trata de uma fechadura eletrônica controlada por um microcontrolador, que oferece três formas de acesso distintas, visando garantir a segurança e facilitar a utilização. Essas formas de acesso incluem login e senha, reconhecimento facial e código QR.

1.1 OBJETIVOS

O projeto tem como objetivos principais:

- Construir a estrutura física da fechadura
- Desenvolver um banco de dados para guardar as informações de acesso dos usuários
- Analisar as informações recebidas pela fechadura e comparar com as informações do banco de dados
- Atualizar o banco de dados e monitorar o acesso da fechadura por uma interface web

Com isso em mãos, integrar tudo à fechadura por meio dos microcontroladores , afim de produzir um protótipo da fechadura inteligente

1.2 MARCOS

- Projeto do sistema elétrico e mecânicos, inicio da construção do sistema elétrico e implementação do acesso via login e senha.
- Finalização do sistema elétrico, implementação do reconhecimento facial e da interface web para maior facilidade ao cadastrar novos usuários.
- Finalização da montagem da estrutura física do projeto, implementação da leitura de código QR e de um dashboard das liberações da fechadura na interface web para maior controle do acesso de pessoas.

2 HARDWARE

O hardware em si é pouco complexo, composto por dois microcontroladores que desempenham funções específicas no sistema. Além disso, são utilizados sensores para o envio e controle da fechadura, um relé para acionamento da mesma, bem como um teclado matricial e um display para aprimorar a interação com o usuário, como podemos ver na figura 1.

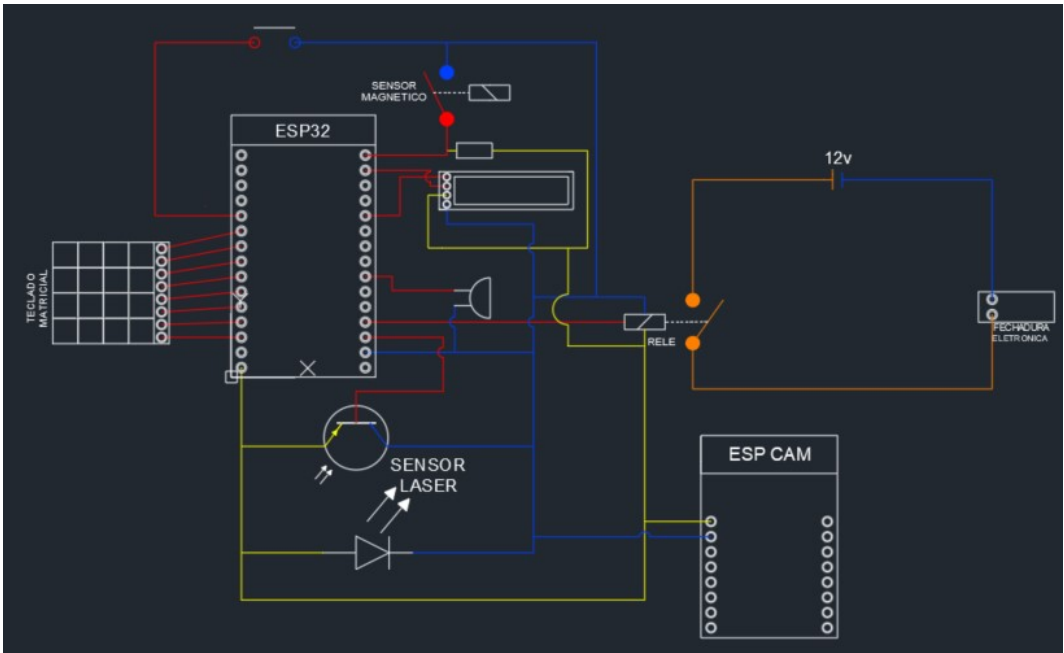


Figura 1 – Representação do sistema elétrico

2.1 COMPONENTES

Tabela 1 – Tabela de componentes.

componente	quantidade	valor
ESP32	1	35,00
ESPCAM	1	40,00
Display OLED 128x64 Px - 0.96	1	15,00
Teclado Matricial de Membrana	1	7,50
Fechadura Solenoide Eletrônica 12V	1	38,80
Módulo Relê 5V	1	8,00
Módulo Receptor Laser	1	16,00
Módulo Laser 5V	1	5,80
Buzzer	1	3,00
Resistores	*	0,05
Total		169,15

Fonte: Autoria própria.

2.2 UTILIZAÇÃO

- O ESP32 foi responsável pelo contato do usuário com a aplicação, recebendo os comandos advindos do teclado matricial, exibindo informações no display, comandando as requisições ao servidor e controlando a abertura ou bloqueio da fechadura, além da contagem de quantas pessoas passaram pela porta.
- O ESPCAM foi designado para registrar as fotos da face.
- O sensor magnético foi usado para se ter um controle do estado da porta, aberta ou fechada.
- Sensor e receptor laser realizam a contagem de quantas pessoas adentraram no local através da porta.
- O relê foi responsável por acionar a fechadura eletrônica.
- Um buzzer para indicação sonora de alguma atividade importante, como o abrir da porta.
- O display OLED e o teclado matricial para facilitar e tornar mais interativa a comunicação entre o projeto e o usuário.

3 SOFTWARE

3.1 A FECHADURA

Para que a fechadura funcione com as diferentes formas de acesso é necessário que por trás dos componentes mecânicos e eletrônicos tenha um software controlando os diferentes estados do dispositivo, gerenciando as comunicações, garantindo a integração e bom funcionamento de todo o sistema.

Em virtude disso, foi pensado e desenvolvido duas frentes para serem trabalhadas no software. Primeiramente, o controle sobre os embarcados e dispositivos acoplados ao mesmo, os quais o usuário final tem contato direto, dando início ao processo de acesso desejado e coletando informações para efetuar a identificação e liberação. Posteriormente, o desenvolvimento do servidor responsável por receber as solicitações provenientes do embarcado, fazer o processamento e consultar se consta no banco de dados, efetuando a liberação ou não. Além disto, oferece ao administrador do ambiente uma aplicação web que realiza o registro e remoção de usuários ao sistema, uma lista com usuários correntes e o número de pessoas que adentraram no local.

3.2 PROTOCOLO DE COMUNICAÇÃO

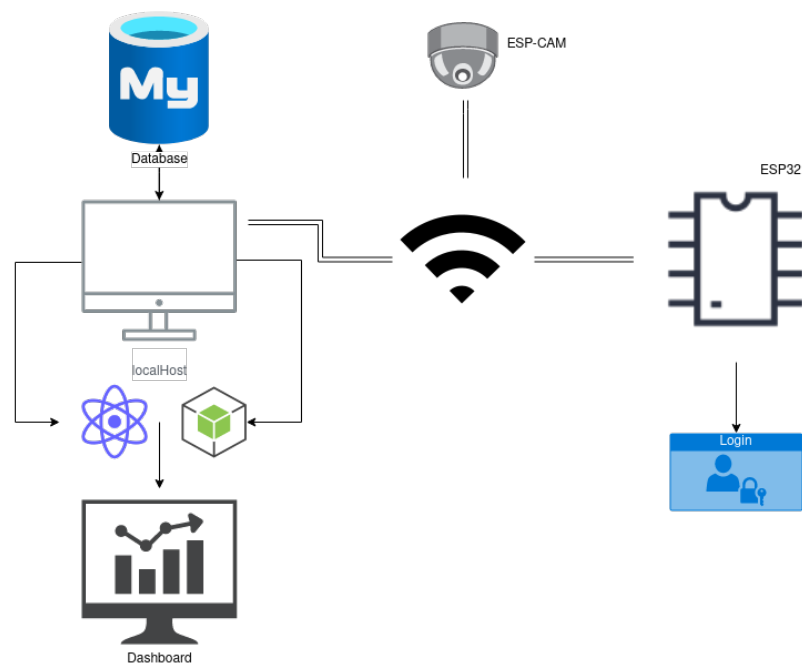


Figura 2 – Representação da comunicação entre os diferentes dispositivos que compõem o projeto.

Todos os dispositivos eletrônicos que compõem a fechadura estavam conectados à rede Wi-Fi, o que permite a transmissão de dados usando ondas de rádio entre os mesmos, como visto na figura 2. Isso só foi possível pelo fato deles possuírem um adaptador Wi-Fi embutido. Esse adaptador é responsável por receber e transmitir os sinais de rádio necessários para a comunicação sem fio.

Tendo isso posto, com uma rede para comunicação estabelecida é necessário um protocolo para a comunicação de dados entre as diferentes entidades que compõem o projeto, assim, foi escolhido o protocolo de Transferência de Hipertexto (HTTP) que se trata de um protocolo de comunicação usado para transferir dados na World Wide Web (WWW), Operando no topo da camada de transporte da pilha de protocolos TCP/IP o cliente envia uma solicitação HTTP ao servidor. Quando uma conexão HTTP é estabelecida, a solicitação é composta por um método (como GET, POST, PUT, DELETE, etc.), eles indicam para o servidor qual a ação que o cliente deseja realizar.

Assim, para a construção desta aplicação foram utilizados apenas dois métodos de solicitação entre cliente e servidor, sendo elas:

- GET – usada quando o cliente deseja obter recursos do servidor
- POST – usada quando o cliente deseja enviar dados para processamento ao servidor, como os dados de um formulário, por exemplo.

No funcionamento do TCP/IP, os dados são divididos em pacotes, numerados e enviados através da rede. O TCP estabelece uma conexão confiável entre as máquinas de origem e destino, garantindo a entrega correta dos pacotes. E o IP lida com o endereçamento e o roteamento dos pacotes, enquanto a camada de rede de acesso trata da transmissão física dos dados. Na comunicação entre o embarcado e servidor foi amplamente utilizado o método POST, uma vez que, enviava dados coletados do usuário para o servidor. Já no próprio servidor web a requisição GET foi predominante, em virtude de, a parte visual da aplicação (Front-End) apenas fazia um requisição ao servidor, que controlava e processava as informações(Back-end) esperando os dados correspondentes para cada tipo de requisição, de modo geral, para receber a lista de informações do banco de dados.

3.3 FUNCIONAMENTO DO SOFTWARE

O funcionamento do software que efetua a liberação pode ser descrito de forma lúdica e estruturada com o fluxograma 3 abaixo: (ADRYAN; ROBSON, 2023)

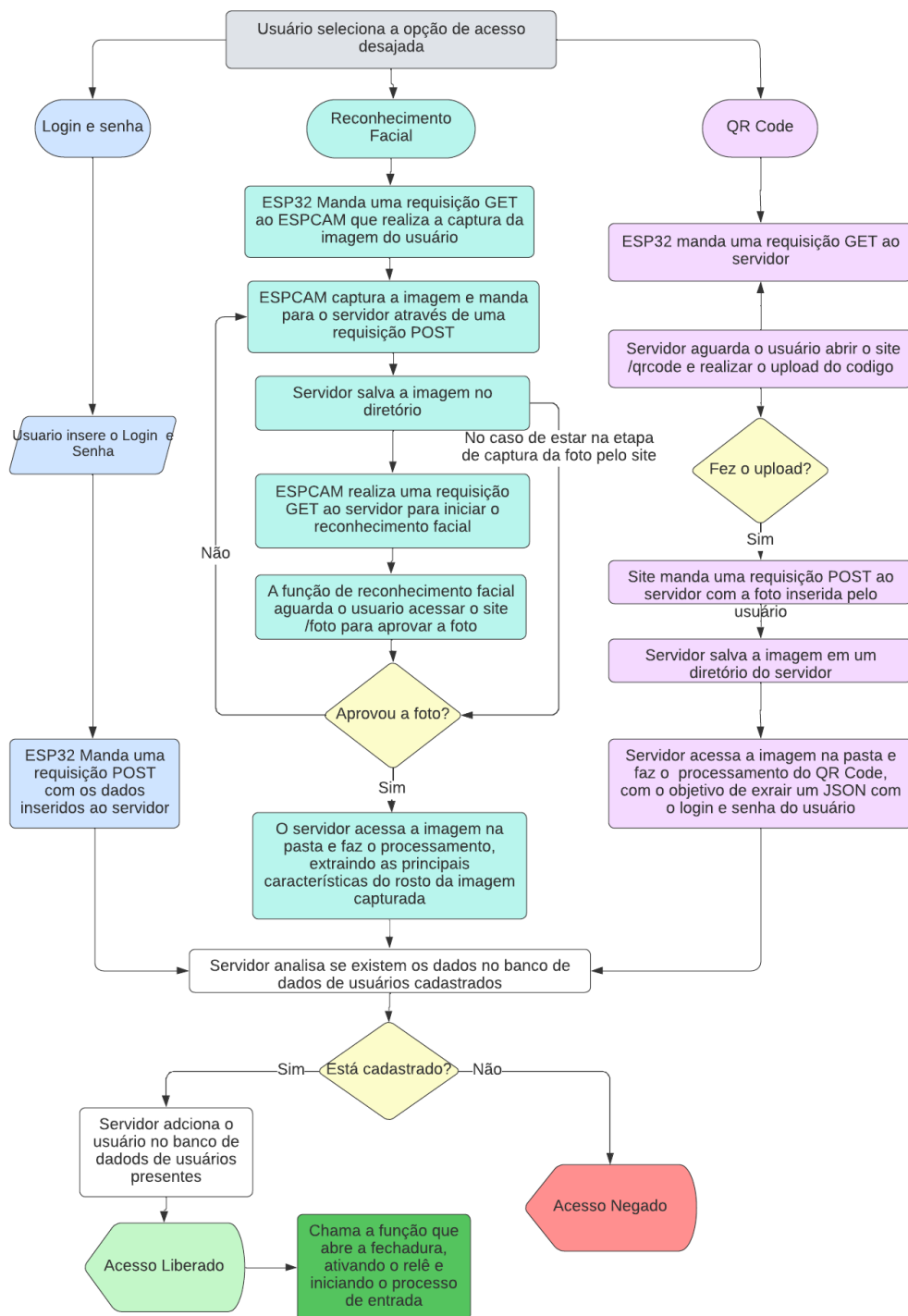


Figura 3 – Fluxograma do funcionamento do software esquematicamente .

3.4 RECONHECIMENTO FACIAL

Para realizar a liberação por reconhecimento facial, é necessário utilizar um algoritmo treinado para essa tarefa. No caso do servidor desenvolvido em JavaScript, a biblioteca escolhida foi a 'face-api.js', que é escrita na mesma linguagem, facilitando sua integração. Essa biblioteca é construída sobre a API principal do 'tensorflow.js', outra biblioteca que permite a execução de tarefas de aprendizado de máquina e inferência em tempo real diretamente no navegador ou em ambientes JavaScript.(SARAVANAN, 2021)

A biblioteca 'face-api.js' utiliza o algoritmo LBPH (Local Binary Patterns Histograms) para realizar o reconhecimento facial. Se trata de um método popular para identificar e reconhecer rostos em imagens. Sua abordagem envolve as seguintes etapas descritas na figura 4:

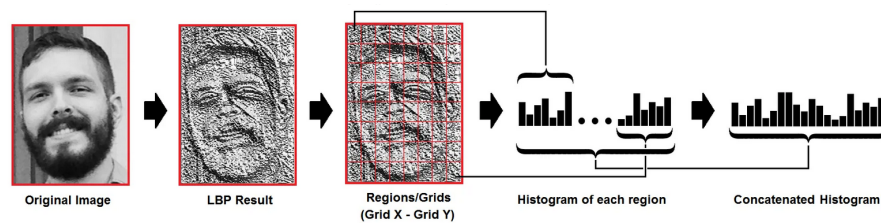


Figura 4 – Processo de tratamento de imagem com o algoritmo LBPH .

- Detecção de rosto: O algoritmo inicia localizando as regiões faciais em uma imagem.
- Pré-processamento: Efetua a equalização de histograma para melhorar o contraste e normalização da iluminação para reduzir variações de iluminação entre as imagens.
- Extração de características: O LBPH opera na imagem em escala de cinza e segmenta a região do rosto em pequenas sub-regiões . O algoritmo compara os pixels em cada sub-região com o pixel central e gera um código binário baseado nas comparações
- Criação do histograma: Os códigos binários gerados para cada sub-região são combinados para formar um histograma de padrões locais binários. O histograma captura a distribuição de diferentes padrões na imagem do rosto.
- Treinamento: O algoritmo de reconhecimento facial LBPH requer um conjunto de imagens de treinamento para construir um modelo.
- Reconhecimento: compara o histograma de LBPH do rosto desconhecido com os histogramas armazenados das imagens de treinamento. É utilizado a equação da distância

Euclidiana para identificar o rosto, uma vez que o descritor da face é um vetor de 128 caracteres.(PRADO, 2017)

$$D = \sqrt{\sum_{i=1}^n (hist1_i - hist2_i)^2}$$

Figura 5 – Fórmula da distância Euclidiana usada pelo algoritmo para encontrar o descritor de face correspondente.

3.5 APLICAÇÃO WEB

- Front-End

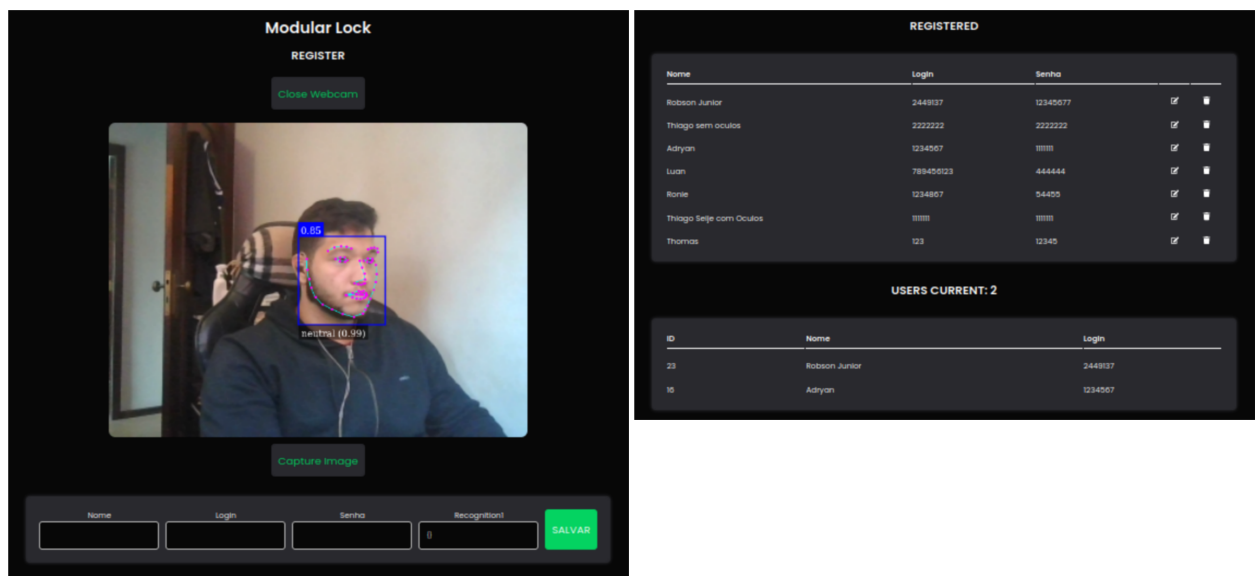


Figura 6 – Página Web da aplicação responsável por realizar o cadastro e exibir os usuários presentes e cadastrados .

Front-end refere-se à parte de uma aplicação web que lida com a interface do usuário e a interação com ela. É a camada que os usuários veem e com a qual interagem diretamente. Para a construção da mesma, foram utilizadas as tecnologias HTML e CSS, para construção e estilização e JavaScript e React para permitir a interação e funcionalidade com os componentes que o compõem. (CHAVAN, 2017)

A parte visual nesta aplicação tem como funcionalidade efetuar o cadastro e remoção de usuários, bem como, permitir o controle dos presentes no ambiente controlado, como mostrado na figura 6. Nesta camada que o usuário do sistema vê os elementos visuais, como os botões,

os formulário, a captura da câmera e uma ideia de como o algoritmo de reconhecimento facial descreve sua face, além da funcionalidade que permite interagir com esses elementos.

- Back-End/Servidor

Em uma aplicação web, o Back-End é à parte da aplicação que lida com a lógica de negócios, processamento de dados, armazenamento e comunicação com o banco de dados. É responsável por fornecer os dados e funcionalidades necessários para a interface, o Front-End. Foi construído usando Node.js(um ambiente de tempo de execução) e Express, um framework para Node.js. Ele fornece uma camada de abstração sobre o Node.js, facilitando a criação de aplicativos e APIs web. O Express simplifica o processo de criação de rotas, manipulação de solicitações e respostas, gerenciamento de sessões e muito mais. (EXPRESS, 2017)

O Back-End foi responsável por processar as solicitações feitas tanto pela aplicação web quanto pelos embarcados, executar a lógica necessária para atender a essas solicitações e retornar os resultados apropriados. Ele lidou com a segurança, autenticação de usuários e outras tarefas relacionadas ao controle de entradas e saídas. A figura 7 mostra as rotas de comunicação estabelecidas possibilitando o bom funcionamento da fechadura.

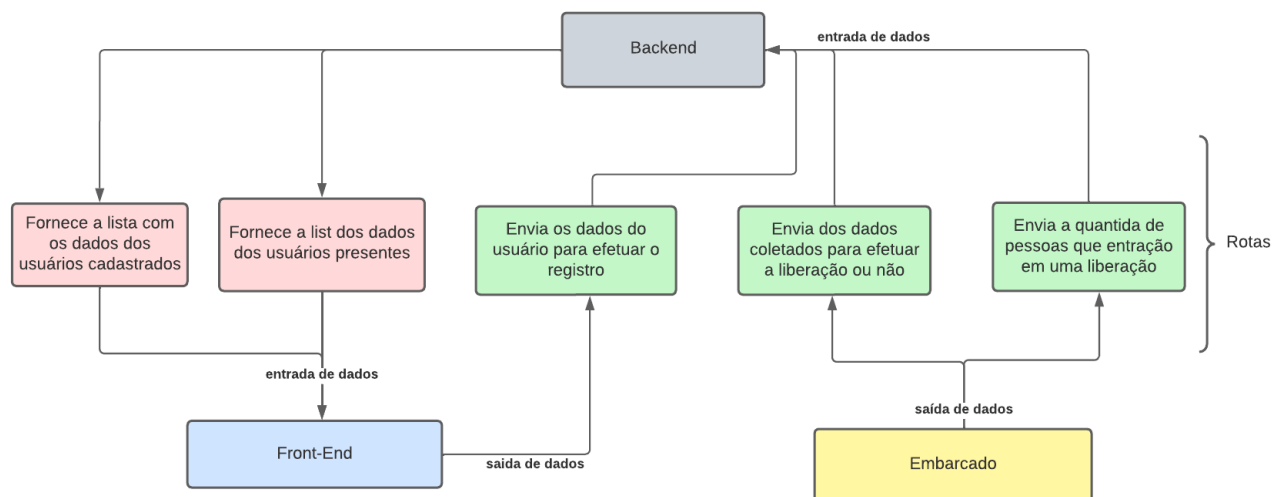


Figura 7 – Fluxograma da comunicação do tipo servidor e cliente no projeto.

- Página Iterativas

Com o intuito de otimizar o controle e a interação dos usuários com o projeto, foi elaborada uma solução composta por duas páginas web distintas, cada uma desempenhando um papel fundamental.

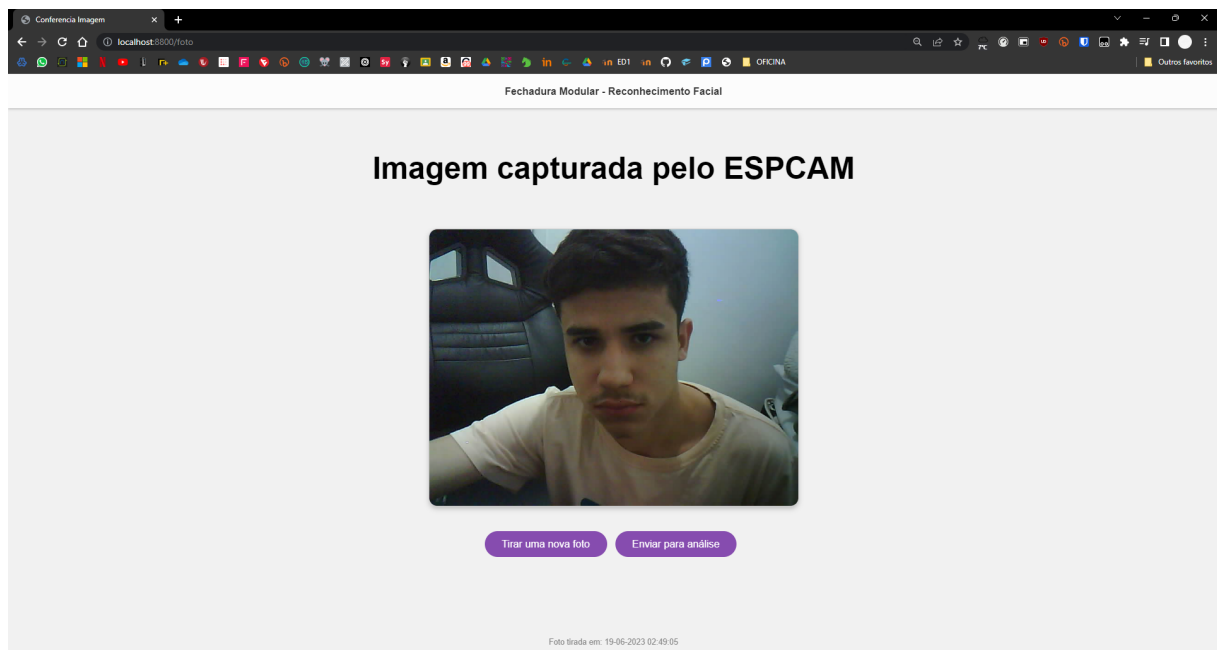
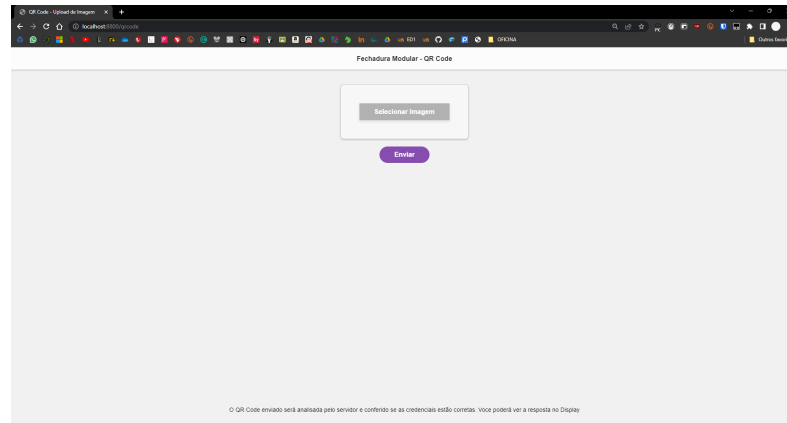


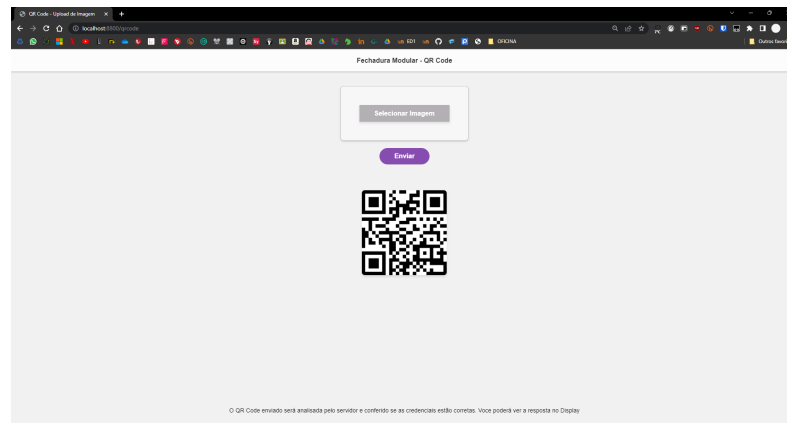
Figura 8 – Página para conferência da foto para o reconhecimento facial.

A figura 8 mostra a primeira Página Web permite que os usuários visualizem a fotografia capturada pelo dispositivo ESPCAM e verifiquem se a imagem está adequada para ser enviada ao sistema de reconhecimento facial. Por meio dessa página, os usuários têm a oportunidade de garantir que o rosto esteja claramente visível e bem posicionado na fotografia, o que é crucial para uma análise precisa e confiável.

Caso a fotografia não atenda aos critérios estabelecidos, o usuário tem a opção de realizar novamente a captura da foto, até que se encontre uma adequada para se continuar o processo, como se pode ver na figura ???. Essa abordagem oferece aos usuários um controle mais preciso e eficiente sobre o processo de autenticação facial, ao mesmo tempo em que garante que apenas imagens de boa qualidade sejam utilizadas para análise e comparação com as referências armazenadas no banco de dados.



(a) Página web sem nenhuma imagem



(b) QR CODE já carregado na página

Figura 9 – Página web para Upload do QR CODE

A segunda página web foi projetada para permitir que o usuário faça o upload do código QR obtido durante o processo de cadastro. Nessa etapa, é possível visualizar a imagem selecionada antes de enviá-la para o servidor, como pode ser visto na figura 9(b). Essa funcionalidade proporciona uma autenticação rápida e confiável, garantindo a integridade e proteção das informações sensíveis do usuário.

A interface da página foi desenvolvida com foco na facilidade de interação, buscando eliminar quaisquer obstáculos que possam comprometer a usabilidade. Dessa forma, os usuários desfrutam de uma experiência tranquila ao utilizar a página, tornando o processo de autenticação via QR-Code intuitivo e eficiente.

Ambas as páginas web foram concebidas para funcionar em conjunto com o Back-End. No processo de reconhecimento facial e acesso via QR-Code, o servidor aguarda a resposta do usuário em ambas as páginas para prosseguir com as etapas necessárias.

4 RESULTADOS

4.1 HARDWARE

Devido ao posicionamento dos componentes, a parte eletrônica foi alocada em em duas placas universais diferentes, uma que fica locada na tampa do dispositivo outra que fica na base, como pode ser visto nas imagens reais do protótipo e na figura 10 abaixo:

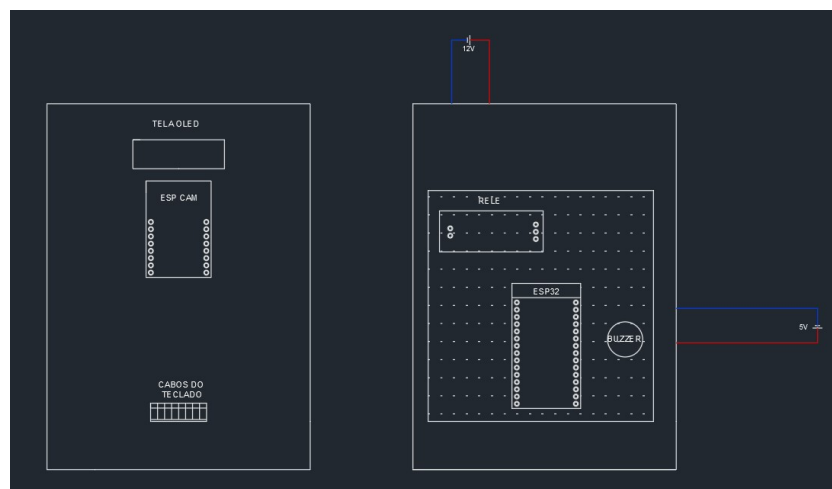
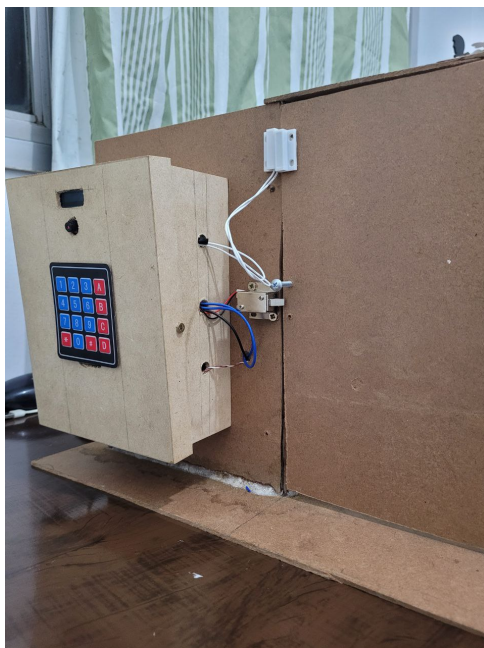
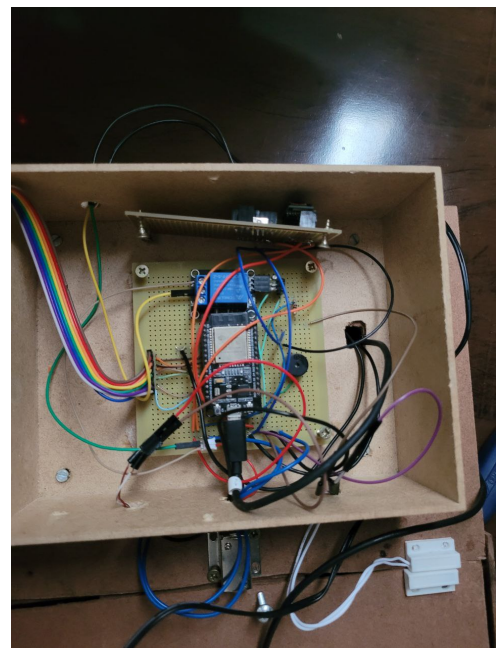


Figura 10 – representação da disposição dos componentes



(a) Caixa aonde estão os componentes



(b) Componentes

Figura 11 – Hardware da fechadura

Outra parte crucial no desenvolvimento do projeto foi a comunicação entre os dois ESPs, houveram alguns problemas ao longo do desenvolvimento até que funcionasse como esperado. Caso tivesse sido usado um único ESP com o módulo para câmera e disponibilidade de portas analógicas e digitais para atender a demanda do projeto, teria tido menos problema e trabalho, uma vez que toda atuação e solicitações estaria centralizado em um único dispositivo.

A utilização do ESPCAM como parte do sistema enfrentou alguns desafios adicionais devido à instabilidade na captura e no envio das fotos para análise. Esses problemas, destacaram a necessidade de uma abordagem cuidadosa e de refinamentos contínuos.

A instabilidade na captura das fotos pelo ESPCAM gerou situações em que as imagens capturadas não correspondiam ao esperado, resultando até mesmo no envio repetido da mesma foto. Essa inconsistência na captura comprometeu o reconhecimento facial e a extração das informações essenciais para a autenticação. Diante desses desafios, foram desenvolvidas páginas iterativas, oferecendo aos usuários finais maior autonomia para capturar a foto corretamente e permitindo que o processo fosse repetido várias vezes, se necessário.

4.2 SOFTWARE

Por parte do Software, ao longo do desenvolvimento houveram algumas dificuldades devido a falta de documentação e exemplos de aplicação, exceto na liberação por login e senha. No reconhecimento facial, por ter usado a biblioteca face-api.js que tem uma comunidade e adoção pequena, foi encontrado poucos artigos e exemplos práticos. Já no QR code, inicialmente foi pensado que uma única foto tirada pelo ESPCAM seria necessário para o Back-End interpretar, no entanto, era encontrado diversos padrões e informações no QR impossibilitando a leitura, mas no final, foi implementado uma solução para o problema.

5 CONCLUSÃO

Este trabalho apresentou o desenvolvimento de uma fechadura eletrônica controlada por um microcontrolador, com três tipos de acesso: login e senha, reconhecimento facial e código QR. Ao longo do projeto, foram alcançados os principais objetivos propostos, que incluíram a construção da estrutura física da fechadura (figura 11(b), figura 11(a) e 12 o desenvolvimento de um banco de dados para armazenar as informações de acesso dos usuários, a análise e comparação das informações recebidas pela fechadura com os dados do banco de dados, bem como a implementação de uma interface web para atualização e monitoramento do acesso.

Ao final do trabalho, apesar de ter um protótipo funcional de uma fechadura eletrônica inteligente, é importante ressaltar que este projeto é um protótipo e que existem possibilidades de melhorias e otimizações futuras. Como o caso do uso de múltiplos controladores, é notável que um único mini-microcomputador (Raspberry Pi ou um ESPCAM com mais entradas digitais, por exemplo) poderia realizar todo o trabalho. Sem contar que poderia ser melhorado os protocolos de segurança e buscar aprimoramentos na usabilidade da interface web, uma vez que armazena dados sensíveis dos usuários.

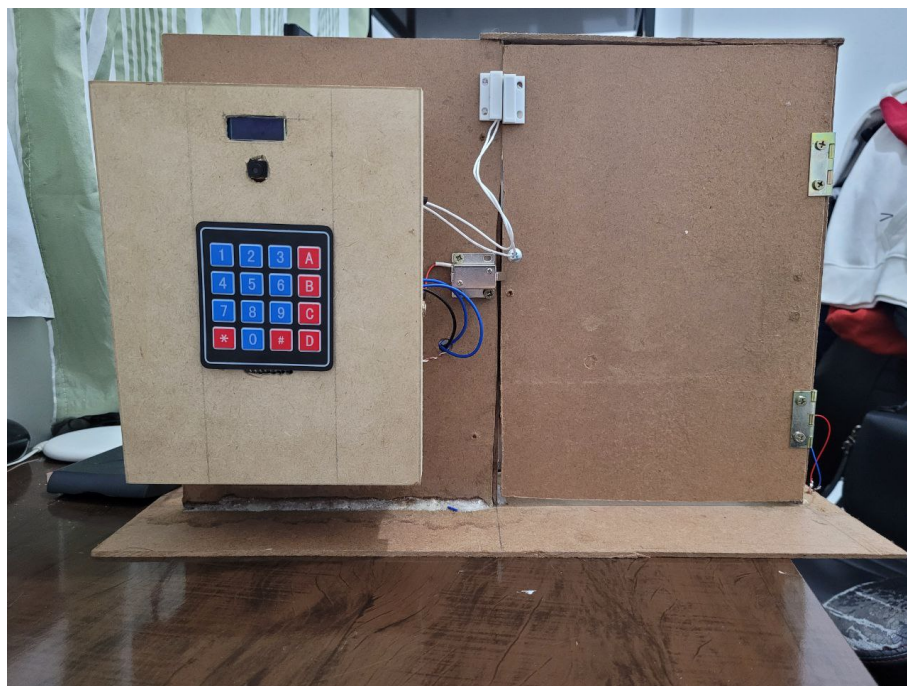


Figura 12 – Protótipo final.

REFERÊNCIAS

ADRYAN, Feres; ROBSON, Santana. **Código do projeto de Oficina de Integração 1**. 2023. Disponível em: <https://github.com/rjrobsonjunior/OficinadeIntegracao1>.

CHAVAN, Yogesh. **React CRUD App Tutorial – How to Build a Book Management App in React from Scratch**. 2017. Disponível em: <https://www.freecodecamp.org/news/react-crud-app-how-to-create-a-book-management-app-from-scratch/>.

EXPRESS. **Routing**. 2017. Disponível em: <https://expressjs.com/en/guide/routing.html>.

PRADO, Kelvin Salton do. **Face Recognition: Understanding LBPH Algorithm**. 2017. Disponível em: <https://towardsdatascience.com/face-recognition-how-lbph-works-90ec258c3d6b>.

SARAVANAN, Jeeva. **face-api.js : A way to build a Face Recognition system in the browser**. 2021. Disponível em: <https://medium.com/theleanprogrammer/face-api-js-a-way-to-build-face-recognition-system-in-browser-c1f4ac922657>.