Relatório Final
Sistema de controle de acesso com três fatores de autenticação

Mateus Felipe Massa, Victor Bastos I. Oliveira
Programa de Graduação em Engenharia Eletrônica, Faculdade Gama
Universidade de Brasília
Gama, DF, Brasil
email: mateusmassa7@gmail.com, victorbio101@gmail.com

I. RESUMO

Fazendo uso de uma placa Raspberry Pi, este projeto consiste no desenvolvimento de um sistema de controle de acesso com três fatores de autenticação, sendo estes a verificação de senha, checagem por RFID e reconhecimento facial por câmera. O sistema visa ser empregado em salas onde o acesso deve ser extremamente restrito, o que é relativamente comum no ramo de empresas laboratoriais, empresas de segurança, órgãos governamentais de alto escalão e forças armadas, por exemplo.

II. INTRODUÇÃO

Controle de acesso é definido como a prática de permitir a entrada e saída de um local apenas para indivíduos autorizados, a fim de garantir a segurança do conteúdo localizado na área. Este tipo de tecnologia tem aplicação em qualquer tipo de local que necessite da filtragem no acesso das pessoas, seja por questões de confidencialidade ou questões de segurança. A aplicação desse sistema pode ser exemplificada através de uma usina termonuclear, onde o acesso a certas partes críticas da planta deve ser estritamente controlado para evitar que pessoal não autorizado, geralmente sem conhecimento técnico e licenciamento, cause algum distúrbio e leve uma catástrofe à acontecer. Outro exemplo advém das aplicações governamentais e militares, onde segredos de Estado, arquivos de alto sigilo e equipamentos devem ser guardados com extrema cautela, com acesso liberado a apenas um seleto grupo de pessoas.

A motivação por trás desta proposta de projeto é realizar o controle eletrônico através de um dispositivo de segurança, mostrando que é possível criar um sistema complexo de autenticação e alerta que garante a eficiente filtragem do acesso de pessoal, a partir da combinação de diversos componentes e técnicas em eletrônica.

Alguns sistemas já propostos foram analisados para embasar o projeto em questão. Estes sistemas eram responsáveis por implementar uma trava inteligente à portas, a fim de garantir mais segurança ao usuário. No sistema proposto em [1] foi criado uma trava de segurança baseada na placa raspberry pi, utilizando câmeras e teclado matricial, para prover um sistema de alarme que tem capacidade de notificar o dono e reconhecer as faces dos usuários cadastrados. Em outro projeto, proposto em [2], foi feito o design e implementação de um sistema de monitoramento remoto de uma porta, a partir de um algoritmo de reconhecimento facial feito através da biblioteca OpenCV associado à checagem de senha, utilizando a raspberry pi como processador central. Em [3] foram implementados três métodos em sequência para validação da entrada, utilizando tecnologia RFID, uma senha PIN de seis dígitos e um código aleatório de único-acesso enviado como mensagem para o celular do usuário. Em todos os projetos tomados como base, nenhum implementou autenticação por três fatores propriamente dito, já que nenhum deles checou informações de tipos específicos na ordem necessária para ser classificado como autenticação por três fatores, como descrito em [4].

III. DESENVOLVIMENTO

A. Descrição de Hardware

Na montagem do sistema foram utilizados cinco periféricos principais conectados a placa de controle que rodava a lógica principal. A placa Raspberry Pi 3 modelo B foi utilizada como unidade de processamento central, responsável por controlar os periféricos de acordo com o código embarcado, além de executar a primeira parte do controle de acesso. Para compilar e executar os arquivos/códigos necessários na placa foi utilizado acesso remoto através do programa VNC Viewer.

Conectado aos pinos da Raspberry Pi, o módulo RFID-RC522 foi utilizado como periférico para validação de acesso, agindo como o segundo fator em um sistema com três fatores de autenticação. Este módulo foi interfaceado com a placa através do protocolo de comunicação SPI, ocupando um total de cinco pinos, desconsiderando alimentação e aterramento. Duas tags padrão MIFARE1 S50, uma do tipo chaveiro e outra do tipo cartão, foram utilizadas para testar a funcionalidade do módulo.

Nas portas USB da placa foram conectados uma webcam Logitech C270, de resolução máxima de 720p em 30 FPS e um teclado do tipo numpad. A câmera foi utilizada para a checagem do último fator de autenticação, dado através do reconhecimento facial do usuário e o teclado do tipo numpad foi utilizado para que o usuário conseguisse introduzir apenas números quando a senha fosse requisitada.

Um display LCD 16x02, interfaceado com a placa através do protocolo de comunicação I2C, foi utilizado para repassar

informações necessárias do sistema para o usuário. Por fim, foi utilizado um circuito isolador de potência constituído de um acoplador óptico 4N35 e um transistor de potência IRF540 para fazer o acionamento da tranca magnética, como descrito na figura a seguir:

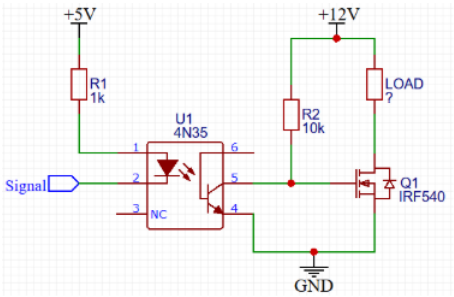

Fig. 1. Circuito isolador de potência.

O esquemático geral de conexão entre os hardwares pode ser visualizado na imagem a seguir:

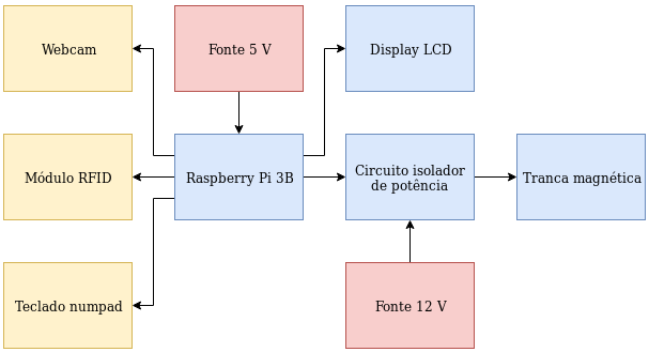

Fig. 2. Esquemático de conexão do hardware utilizado.

Por fim, tem-se a BOM (Bill of Materials) do projeto onde são descritos todos os componentes utilizados, qual o sistema em que estão inseridos, o fornecedor e preço de cada um. A Tabela se encontra no Apêndice no final do documento.

B. Descrição de Software

O software desenvolvido foi separado em dois blocos principais: sistema de cadastro e sistema de segurança. O bloco de cadastro foi desenvolvido a fim de permitir uma maior flexibilidade e facilidade no manejo de usuários para o administrador do bloco de segurança. A lógica principal deste bloco é descrita pela figura Fig.3 .

A partir de um menu principal, onde a opção inserida pelo usuário foi validada para aceitar apenas uma das opções disponíveis, apresentam-se os rumos que o programa pode seguir. Na opção de cadastrar um novo usuário, o programa primeiramente recolhe a identificação numérica (ID) escolhida para o novo usuário, a qual foi validada para aceitar apenas números e ter uma tamanho total de cinco caracteres. Na próxima etapa o programa requisita o nome do novo usuário, validando para que sejam inseridos apenas letras e para que o campo não esteja vazio. Ao pegar o nome, o programa faz uma formatação para que todas as letras sejam

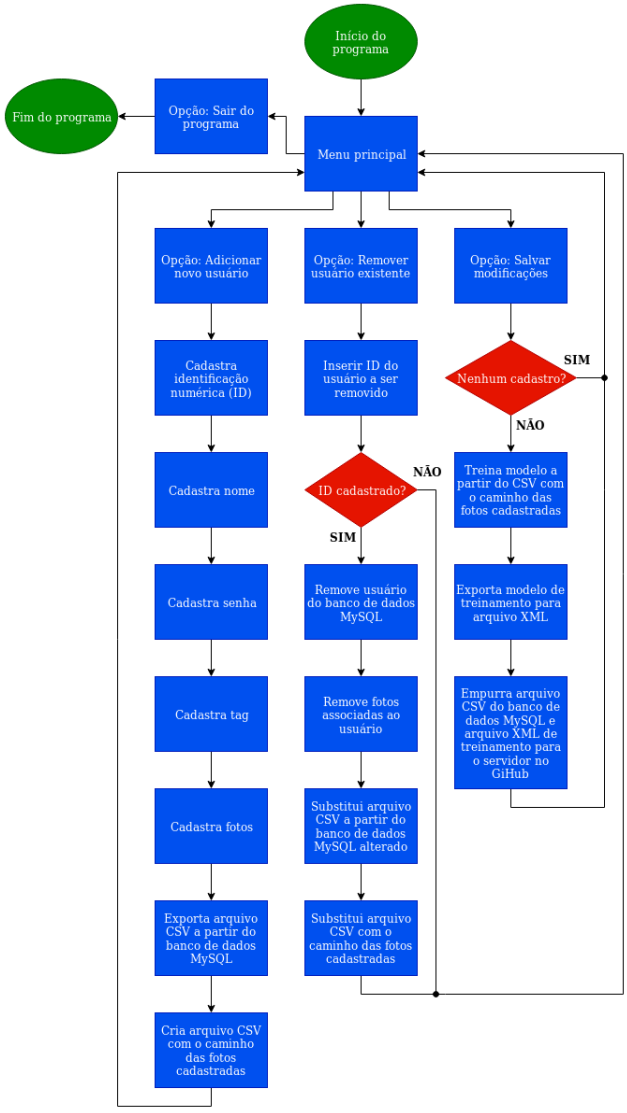

Fig. 3. Diagrama de blocos do sistema de cadastro.

maiúsculas, criando um padrão de nomes dentro do banco de dados. A seguir, o programa requisita a senha para o usuário, a qual foi validada para aceitar apenas números e ter um tamanho total de seis caracteres. Na próxima etapa o programa pede o padrão da tag, onde a aquisição foi validada para aceitar números e letras. Após pegar a tag, o programa formata o dado inserido para que todas as letras fiquem maiúsculas, respeitando o padrão de código dos cartões tipo MIFARE. Por fim, são retiradas e armazenadas vinte fotos do usuário, a fim de serem usadas no treinamento do modelo posteriormente.

Após retiradas as fotos, o programa envia o comando de exportar CSV para o banco de dados MySQL, a fim de gerar um arquivo CSV com todos os usuários atualmente cadastrados. A partir desse CSV gerado, é criado um outro CSV com a localização das fotos dentro do computador e o ID associado a cada foto, terminando assim a adição do novo usuário.

Para cada foto retirada, o programa realiza uma etapa de pré-processamento utilizando a biblioteca openCV 4 para armazenar as fotos em um formato adequado para o treinamento utilizando o algoritmo de reconhecimento facial Fisher Faces. Este algoritmo pode ser visualizado na figura a seguir, que explica a sucessão de tratamentos para se obter o resultado final:

Fig. 4. Diagrama de blocos para o pré-processamento das fotos.

Na opção de remover usuário existente, o programa requisita o ID do usuário a ser removido e em seguida checa se esse ID existe no banco de dados. Caso o ID não exista, o programa volta para o menu inicial, mas se este existir, é mandado o comando ao banco de dados MySQL para remover o usuário. Em seguida o programa exclui todas as fotos cadastradas associadas ao ID do usuário, para após gerar novamente o CSV com os dados de todos os usuários do banco de dados e gerar o CSV com o caminho das fotos associadas ao ID.

Na última opção, intitulada como salvar mudanças, o programa checa se há algum usuário cadastrado no banco de dados antes de prosseguir. Isso foi necessário já que o modelo de treinamento de reconhecimento facial utilizando o método Fisher Faces pela biblioteca OpenCV necessita de duas amostras de faces antes de treinar o modelo. Como no banco de dados das fotos foram inseridas diversas imagens de pessoas aleatórias para criar uma categoria de "desconhecidos" para o sistema, foi necessário realizar a checagem para no mínimo apenas uma pessoa ser cadastrada. Caso algum usuário esteja cadastrado, o programa treina o modelo a partir do arquivo CSV com o caminho das fotos associadas ao ID, gerando um arquivo com extensão XML que pode ser exportado. Por fim, o programa empurra o arquivo CSV do banco de dados MySQL e o arquivo de modelo treinado XML para o servidor criado no GitHub, para que o sistema de segurança possa usar os dados atualizados.

O bloco de segurança é constituído de basicamente de duas partes principais: a primeira parte que é um script que é executado intermitentemente fazendo o download dos arquivos de dados do usuário do servidor no GitHub. A segunda parte é o controle de acesso com a autenticação por três fatores (senha, RFID e reconhecimento facial).

O funcionamento do script é dado a partir de um loop infinito onde é utilizada uma função do git chamada pull que realiza uma verificação no servidor para saber se houve alguma mudança nos arquivos, se sim, ele realiza o download e encaminha para uma pasta selecionada dentro da Raspberry Pi, caso contrário ele informa ao sistema que não houve nenhuma mudança.

O controle de acesso foi desenvolvido utilizando como base a biblioteca MFRC522, uma biblioteca desenvolvida para o uso do módulo RFID na Raspberry Pi, funções prontas de uma biblioteca para o display LCD, desenvolvida em C com base na biblioteca wiringPi, uma função a parte de reconhecimento facial, desenvolvida a partir da biblioteca OpenCV, funções prontas com base na biblioteca String e Fstream de C++ para validação e busca em arquivo, biblioteca Time para recolhimento de data e hora, comandos do pacote Mutt para envio de e-mails e a biblioteca wiringPi para acionamento de pinos.

O fluxograma da figura Fig.5 descreve visualmente o programa principal, informando o fluxo das informações dentro do programa:

O loop principal começa realizando o mapeamento do pino GPIO1 que será usado para acionamento da tranca magnética. Depois é iniciado a autenticação da senha, onde a senha é requerida para o usuário e assim que é inserida o programa compara o dado de entrada, procurando no arquivo CSV que contém os dados do usuário, comparando com os dados da coluna 'senha' desse arquivo. A lógica se resume em recolher todos os dados dessa coluna em um vetor e comparar cada posição do vetor com o dado inserido. Caso essa comparação seja válida, o programa salva o identificador

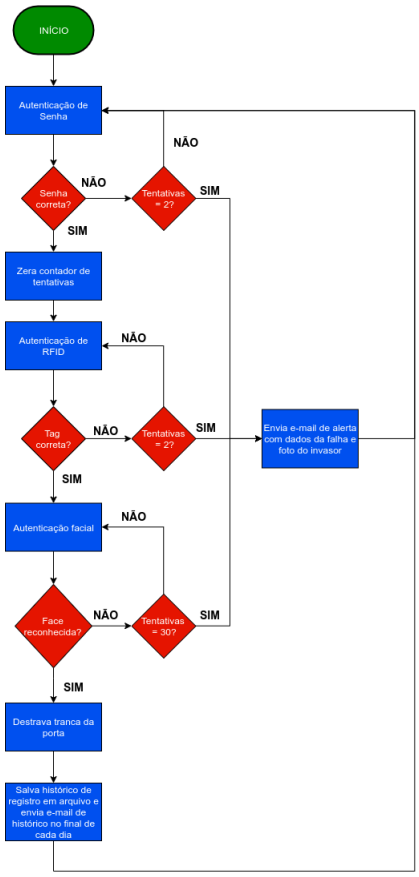

Fig. 5. Diagrama de blocos do sistema de controle de acesso.

de arquivo, retorna um booleano verdadeiro retornando para o loop principal. Caso contrário, o programa retorna um booleano falso, voltando para o começo da validação, pedindo a senha para o usuário novamente, enquanto o número de tentativas erradas for menor que 2 (começando em 0, ou seja, 3 tentativas). Caso o número de tentativas seja igual a 2 o programa aciona um programa de alerta que envia um email contendo dados (hora, data, etapa de verificação errada e foto do invasor) e envia para o email cadastrado do sistema. Após o programa reinicia, retornando para a validação inicial.

Em caso de sucesso na validação anterior, o usuário é direcionado para a validação de tag, no qual é requerida a tag do usuário. Após recolhido, o dado é comparado com os dados do arquivo de dados do usuário, agora refinando a procura a partir do identificador de arquivo que vai indicar em qual posição a validação anterior deu certo, procurando agora pelo número de tag correspondente desse identificador. Caso a tag inserida seja igual a tag correspondente da validação anterior, a validação atual é válida e o programa retorna um booleano verdadeiro, retornado para o loop principal. Caso contrário, o programa retorna um booleano falso, voltando para o começo da validação, requerindo a tag novamente, enquanto o número de tentativas for menor que 2. Caso o número de tentativas se iguale a 2, o programa aciona o programa de alerta que envia o email citado anteriormente. Após o programa reinicia, voltando para a validação inicial de senha.

Em caso de sucesso na validação da tag, o usuário é conduzido para a validação de reconhecimento facial, no qual é acionada a função criada à parte que realiza o reconhecimento facial, citada anteriormente. Na validação o usuário é direcionado a se posicionar em frente a webcam para o recolhimento da foto da face, onde essa face gerará um número de predição com base nas imagens do banco de dados e esse número é comparado com o identificador do usuário acessado a partir do identificador de arquivo citado. Esse processo é repetido enquanto o contador de sucesso, que é atrelado ao grau de confiabilidade da predição feita na imagem, ou o contador de tentativas não alcancem um valor determinado. Caso a autenticação seja válida, o programa retorna um booleano verdadeiro voltando para o loop principal, onde é acionada a tranca de segurança liberando a passagem para o usuário, bem como é feita a atualização do arquivo de histórico de acesso, que é enviado a cada 10 minutos para o email cadastrado do sistema. Caso a autenticação seja inválida, o programa retorna um booleano falso voltando para o loop principal onde o programa de alerta é acionado e a entrada do usuário é barrada.

O fluxograma abaixo demonstra o funcionamento da função de reconhecimento facial descrita acima:

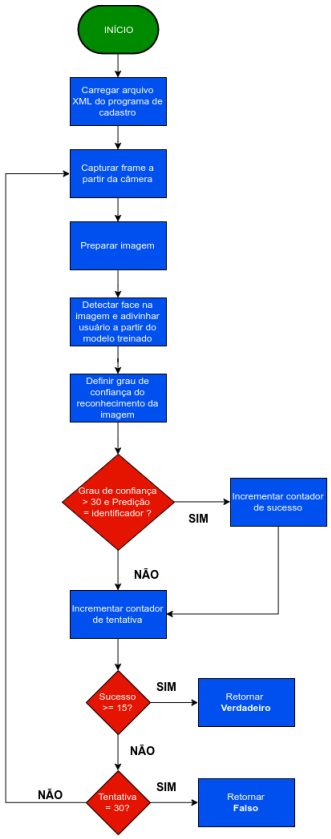

Fig. 6. Diagrama de blocos do programa de reconhecimento facial.

A função de reconhecimento demonstrada acima é uma função a parte, desenvolvida utilizando a biblioteca OpenCV. O funcionamento é dado a partir de um loop de tentativas

onde o retorno é booleano indicando se a verificação foi verdadeira ou falsa. Nesta função têm-se a captura do quadro pego pela webcam e um processamento para comparação com as imagens do banco de dados, que consiste em colocar a imagem em escala de cinza, detectar face na imagem, cortar imagem em torno da face detectada e redimensionar imagem para uma resolução gráfica definida. A partir desse processamento é feito uma predição, que reconhece a face na imagem, gerando também um grau de confiabilidade dessa predição. A partir desses dois valores são feitas verificações, na qual, caso o identificador da predição seja igual a um identificador definido e o grau de confiabilidade seja maior que 50%, um contador de sucesso é incrementado, caso contrário o programa segue e um contador de tentativas é incrementado. A verificação final é baseada nesses dois contadores, onde, caso o contador de sucesso seja igual a quinze, a função retorna booleano verdadeiro, indicando que a autenticação foi um sucesso. Caso o contador de tentativas seja maior ou igual a vinte e a verificação anterior for falha, a função retorna booleano falso, indicando que a autenticação falhou.

IV. Resultados

O funcionamento do programa de cadastro foi atestado através da visualização dos cadastros no banco de dados MySQL, assim como através dos arquivos gerados. Ao cadastrar um novo usuário, este foi tabelado dentro do banco de dados com todas as informações corretas, inseridas através do programa. Os arquivos CSV do banco de dados e o XML do modelo treinado foram gerados ao final do cadastro, onde também foi possível verificar a existência do novo usuário abrindo o arquivo com um editor de texto. Conferindo o repositório no GitHub, percebeu-se que as mudanças foram corretamente empurradas para o servidor através da opção de salvar mudanças, já que os dois arquivos necessários para o funcionamento do programa de segurança estavam presentes. A remoção de usuário também funcionou como esperado, já que quando foi selecionada, o usuário foi removido do banco de dados MySQL e os arquivos finais foram gerados novamente, a fim de atualizar a lista atual sem o usuário.

No programa de segurança, foram testados os periféricos e a funcionalidade de todos os códigos utilizados. Utilizando o teclado tipo numpad, foi inserida a senha numérica, que foi corretamente verificada a partir do arquivo CSV advindo do repositório. O mesmo efeito foi observado com o módulo RFID, o qual respondeu positivamente quando a tag associada ao usuário que introduziu a senha estava correta. A última etapa de verificação, o reconhecimento facial, também foi validada com sucesso, mas algumas ressalvas podem ser feitas. O método utilizado se apresentou extremamente sensível a condições de luz no ambiente, o que ficou evidente quando o reconhecimento facial foi testado em um local diferente do local que as fotos de treinamento foram tiradas.

As funcionalidades de alerta também foram validadas com sucesso. Ao errar pelo menos três vezes em qualquer etapa dos processos de verificação, a câmera era ativada e uma foto do invasor era tirada e mandada ao email cadastrado do projeto, com as informações de hora, data e em qual etapa do processo o invasor falhou. O relatório de acesso também foi gerado com sucesso, sendo mandado para o email cadastrado de dez em dez minutos (para fins de teste) com os usuários que conseguiram entrar, assim como a data e horário da entrada dos mesmos.

V. Conclusão

Com o desenvolvimento do projeto verificou-se que a utilização de sistemas embarcados na área de segurança é uma alternativa extremamente viável devido a capacidade de processamento de dados e robustez do sistema que é dedicado à uma aplicação exclusiva. Fazendo o uso da placa Raspberry Pi verificou-se sua funcionalidade, juntamente com os periféricos citados e unindo a funcionalidade de cada um através do software foi capaz de se desenvolver um sistema embarcado de segurança baseado em três fatores de autenticação. Os resultados encontrados foram satisfatórios tendo em vista o desenvolvimento a nível de protótipo do projeto, no qual visaram atestar o funcionamento em conjunto do sistema considerando tanto a parte de cadastro como a parte de controle de acesso.

Referências

[1]Vivek MISHRA, Shwetank. SONI. Smart door system for home security using raspberry pi 3. 2019. [3]SWITCHED ON NETWORK. Build a raspberry pi smart door lock security system for your smart home. 2019. [4]Margaret ROUSE. Three factor authentication (3fa). 2019. [2]Nasir et al ROY, Sourav. UDDIN. Design and implementation of the smart door lock system with face recognition method using the linux platform raspberry pi. 2018.

APÊNDICE

A. Primeiro Apêndice

Sistema de controle de acesso com três fatores de autenticação							
Item	**Sistema**	**Componente**	**Descrição**	**Fornecedor**	**Qtd**	**Preço unitário**	**Preço**
#1	Alimentação	Fonte 12V 2.5A	Fonte 12VDC 110/220V 2,5A C/PLUG P4	Hu Infinito	1	R$ 18,90	R$ 18,90
#2	Alimentação	Fonte 5V 3.0A	Fonte p/ Raspberry PI – 5VDC 110/220V 3A c/ MicroUSB	TechSul Eletrônicos	1	R$ 23,90	R$ 23,90
#3	Interação	Teclado NumPad	Teclado numérico usb preto 0134 Bright	Kalunga	1	R$ 33,50	R$ 33,50
#4	Interação	Display LCD 16x2	Display Cristal Líquido (LCD 16X02 - VD/PT)	Hu Infinito	1	R$ 13,90	R$ 13,90
#5	Interação	Módulo I2C	Módulo Serial I2C p/ Display LCD	TechSul Eletrônicos	1	R$ 8,50	R$ 8,50
#6	Interação	Módulo RFID	Módulo RFID-RC522 com Kit de Tags 13,56MHz	Hu Infinito	1	R$ 15,90	R$ 15,90
#7	Interação	WebCam	Webcam HD 720p C270 Logitech	Mercado Livre	1	R$ 120,00	R$ 120,00
#8	Interação	Tranca Magnética	Mini Fechadura Solenóide Eletrônica 12v	Mercado Livre	1	R$ 33,99	R$ 33,99
#9	Processamento	Raspberry Pi 3B	Raspberry Pi modelo 3B com case e cooler de resfriamento	Ncam	1	R$ 440,00	R$ 440,00
#10	Controle	Transistor	Transistor IRF	TechSul Eletrônicos	1	R$ 3,50	R$ 3,50
#11	Controle	Optoacoplador	Optoacoplador	TechSul Eletrônicos	1	R$ 1,29	R$ 1,29
#12	Controle	Resistores	Resistor de 1k e 10K	Hu Infinito	2	R$ 0,04	R$ 0,08
#13	Interno	Jumpers	Jumper para protoboard 200mm 40 peças	Hu Infinito	1	R$ 8,90	R$ 8,90
#14	Interno	Protoboard	Protoboard 400 pontos	Hu Infinito	1	R$ 9,00	R$ 9,00
#15	Estrutura	Cortes em MDF	8 cortes personalizados em MDF	I9	1	R$ 120,00	R$ 120,00
#16	Estrutura	Puxador	Puxador de ferro para a porta	Nova Arte	1	R$ 2,20	R$ 2,20
#17	Estrutura	Parafusos	Parafusos 2.5M/14mm de ferro para conexão da madeira	Tozetti	16	R$ 0,04	R$ 0,64
#18	Estrutura	Parafusos	Parafusos 3.5M/45mm de ferro para conexão da madeira	Tozetti	9	R$ 0,06	R$ 0,54
#19	Estrutura	Dobradiça	Dobradiça para caixa e porta	Tozetti	4	R$ 2,60	R$ 10,40
			Preço Total				R$ 865,14