



UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE
CENTRO DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA DE COMPUTAÇÃO E AUTOMAÇÃO
CURSO DE ENGENHARIA DE COMPUTAÇÃO E AUTOMAÇÃO

RELATÓRIO DO PROJETO DA UNIDADE 1 SISTEMA DE CONTROLE RESIDENCIAL

Discentes:

Jaderson de Oliveira
Joanderson Lucas Laurentino
Vitor Araújo de Almeida

Orientador: Professor Sérgio Natan Silva

Natal

21 de agosto de 2019

Sumário

| | |
|---|-----------|
| 1Introdução..... | 3 |
| 1.1 Situação problema..... | 3 |
| 2Abordagem Teórica..... | 4 |
| 2.1Python..... | 4 |
| 2.2Arduíno..... | 5 |
| 2.3Sockets..... | 5 |
| 2.4Thread..... | 6 |
| 3Descrição da Proposta..... | 6 |
| 4Desenvolvimento e Resultados..... | 7 |
| 5Conclusão..... | 12 |
| Referências..... | 12 |

1 Introdução

A tecnologia vem aos longos dos anos evoluindo e a segurança continua sendo uma das áreas onde ela é mais empregada, mas diferente do que muitos pensam, o emprego maciço dela não é somente no setor empresarial, é também no residencial e vem aumentando ano após ano resultado da insegurança.

Investimentos em infraestrutura de comunicações aliado ao crescimento do acesso à internet tem impulsionado o setor de construção civil e a sociedade como um todo a acrescentar soluções para controle residencial.

O controle de acesso e sensor de presença são umas dessas soluções para quem busca segurança através da tecnologia. O controle de quem entra e sai da sua casa ou em qual ambiente se encontra uma possível ameaça é essencial na hora de assegurar a proteção de quem mora ali e foi pensando nisso que esse projeto foi desenvolvido.

No desenvolvimento deste relatório vamos tratar sobre quais foram as soluções impregnadas para solucionar a proposta de projeto apresentada, as técnicas e linguagens de programação utilizadas, dificuldades que foram encontradas e o resultado final.

1.1 Situação problema

A segurança em um país com uma casa invadida a cada uma hora só no estado de São Paulo, segundo dados da SSP (Secretaria da Segurança Pública) é fundamental para uma melhor qualidade de vida. Isso tem levado não só vítimas como também quem preza pela sua segurança, a procurar ajuda de profissionais na área para investir na aplicação de sistemas de segurança ou a sofisticar um já existe.

Assim sendo, este projeto tem como objetivo apresentar uma proposta de solução para controle residencial utilizando tecnologias de baixo custo baseado na Internet das Coisas (Internet of Things - IoT) , utilizando Arduino, Python e etc.

2 Abordagem Teórica

2.1 Python

Python é uma linguagem de propósito geral de alto nível, multiparadigma, suporta o paradigma orientado a objetos, imperativo, funcional e procedural. Possui tipagem dinâmica e uma de suas principais características é permitir a fácil leitura do código e exigir poucas linhas de código se comparado ao mesmo programa em outras linguagens. Devido às suas características, ela é principalmente utilizada para processamento de textos, dados científicos e criação de CGIs para páginas dinâmicas para a web. Pode ser utilizada também, para implementar projetos que utilize diversas tecnologias, hardware (Arduino) e software (Web), por exemplo. Esta linguagem é considerada uma das 5 linguagens mais populares, de acordo com uma pesquisa conduzida pela RedMonk.

Para a implementação do projeto descrito nesse documento, foram utilizados funções e métodos de algumas bibliotecas da linguagem Python. quais foram importas para o projeto:

Socket: Através dessa biblioteca, o Python implementa a interface de rede utilizando os fundamentos da API de Socket, com ela, é possível criar uma estrutura de comunicação em rede entre um servidor e um cliente. Nessa aplicação, foi utilizado o protocolo TCP.

Thread: Este módulo fornece primitivas de baixo nível para trabalhar com vários threads (também chamados de processos ou tarefas leves) - vários threads de controle que compartilham seu espaço de dados global. Para sincronização, são fornecidos bloqueios simples (também chamados de mutexes ou semáforos binários).

Time: O módulo time oferece várias funções relacionadas a tempo, acesso de tempo e conversões. Com essa biblioteca, podemos implementar por exemplo, contadores.

Serial: Este módulo encapsula o acesso à porta serial. Ele fornece backends para Python executando no Windows, OSX, Linux, BSD. O módulo chamado “serial” seleciona automaticamente o back-end apropriado.

2.2 Arduíno

O Arduino é uma plataforma de prototipagem eletrônica de hardware livre e de placa única, projetada com um microcontrolador Atmel AVR com suporte de entrada e saída, uma linguagem de programação padrão, a qual tem origem em Wiring, e é essencialmente C/C++. O propósito do projeto do arduíno foi de desenvolver ferramentas que são acessíveis, que seja de baixo custo, flexíveis e fáceis de se utilizada por iniciantes e profissionais. Principalmente para aqueles que não teriam alcance aos controladores mais sofisticados e ferramentas mais complicadas.

Pode ser usado para o desenvolvimento de objetos interativos independentes, ou ainda para ser conectado a um computador hospedeiro. Uma típica placa Arduino é composta por um controlador, algumas linhas de E/S digital e analógica, além de uma interface serial ou USB, para interligar-se ao hospedeiro, que é usado para programá-la e interagi-la em tempo real. A placa em si não possui qualquer recurso de rede, porém é comum combinar um ou mais Arduinos deste modo, usando extensões apropriadas chamadas de shields. A interface do hospedeiro é simples, podendo ser escrita em várias linguagens. A mais popular é a Processing, mas outras que podem comunicar-se com a conexão serial são: Max/MSP, Pure Data, SuperCollider, ActionScript e Java.

2.3 Sockets

Network socket é um ponto final de um fluxo de comunicação entre processos através de uma rede de computadores. Atualmente, parte da comunicação entre computadores é baseada no Protocolo de Internet, portanto a maioria dos sockets de rede são sockets de Internet.

Uma API de sockets (API sockets) é uma interface de programação de aplicativos (API), normalmente fornecida pelo sistema operacional, que permite que os programas de aplicação controlem e usem soquetes de rede. APIs de soquete de Internet geralmente são baseados no padrão Berkeley sockets.

Um endereço de soquete (socket address) é a combinação de um endereço de IP e um número da porta, muito parecido com o final de uma conexão telefônica que é a combinação de um número de telefone e uma determinada extensão. Com base nesse endereço, soquetes de internet entregam pacotes de dados de entrada para o processo ou thread (computação) de aplicação apropriado.

2.4 Thread

um processo dividir a si mesmo em duas ou mais tarefas que podem ser executadas concorrentialmente. O suporte à thread é fornecido pelo próprio sistema operacional no implementada através de uma biblioteca de uma determinada linguagem, no caso de uma User-Level Thread (ULT). Uma thread permite, por exemplo, que o utilizador de um programa utilize uma funcionalidade do ambiente enquanto outras linhas de execução realizam outros cálculos e operações.

Em hardwares equipados com uma única CPU, cada thread é processada de forma aparentemente simultânea, pois a mudança entre uma thread e outra é feita de forma tão rápida que para o utilizador, isso está acontecendo paralelamente. Em hardwares com múltiplos CPUs ou multi-cores, as threads são realizadas realmente de forma simultânea.

Os sistemas que suportam uma única thread (em real execução) são chamados de monothread enquanto que os sistemas que suportam múltiplas threads são chamados de multithread.

3 Descrição da Proposta

Esse trabalho tem como finalidade fazer um sistema de controle residencial doméstico que permite a entrada mediante da senha correta e um computador que irá receber as informações de alguma tentativa de acesso ou movimentações no ambiente.

Para esse projeto planejamos utilizar o sensor de distância para ligar a lampada do ambiente e também para habilitar utilização do teclado para o recebimento da senha. Com o teclado habilitado e após a inserção da senha o sistema irá avaliar se a senha está correta, caso a senha estiver correta abrir a porta, e acionar um toque através de um Buzzer e uma luz verde para mostrar sucesso, com a senha errada além de não abrir a porta vai ser emitido um outro toque através de um Buzzer e uma luz vermelha implicando em senha errada, acesso negado. E durante esse processo as informações estarão sendo passadas para o computador.

O computador vai atuar como um servidor que irá mandar essas informações colhidas do sistema para os clientes (outros computadores via Sockets), essas informações vão compor o log do sistema, contendo quando foi feito o acesso, e qual foi o resultado do acesso.

4 Desenvolvimento e Resultados

O primeiro passo após a definição de qual projeto iríamos fazer foi o levantamento do que seria necessário para a implementação daquilo que nós tínhamos planejados. O material utilizado foi:

- Resistores
- Leds
- Jumpers
- Protoboard
- Buzzer
- Micro Servo Motor 9g SG90
- Sensor de Distância Ultrassônico HC-SR04
- Teclado Touch Capacitivo TTP229
- Rele 12V 1 Canal

Com os materiais necessários em mãos, decidimos então pensar sobre a lógica de funcionamento do circuito utilizando diagrama de blocos, como pode ser visto na figura 1, com isso feito partimos para a implementação do código Arduino. Começamos incluindo todas as bibliotecas essenciais para o funcionamento dos módulos, após isso fizemos as definições das portas que seriam utilizadas.

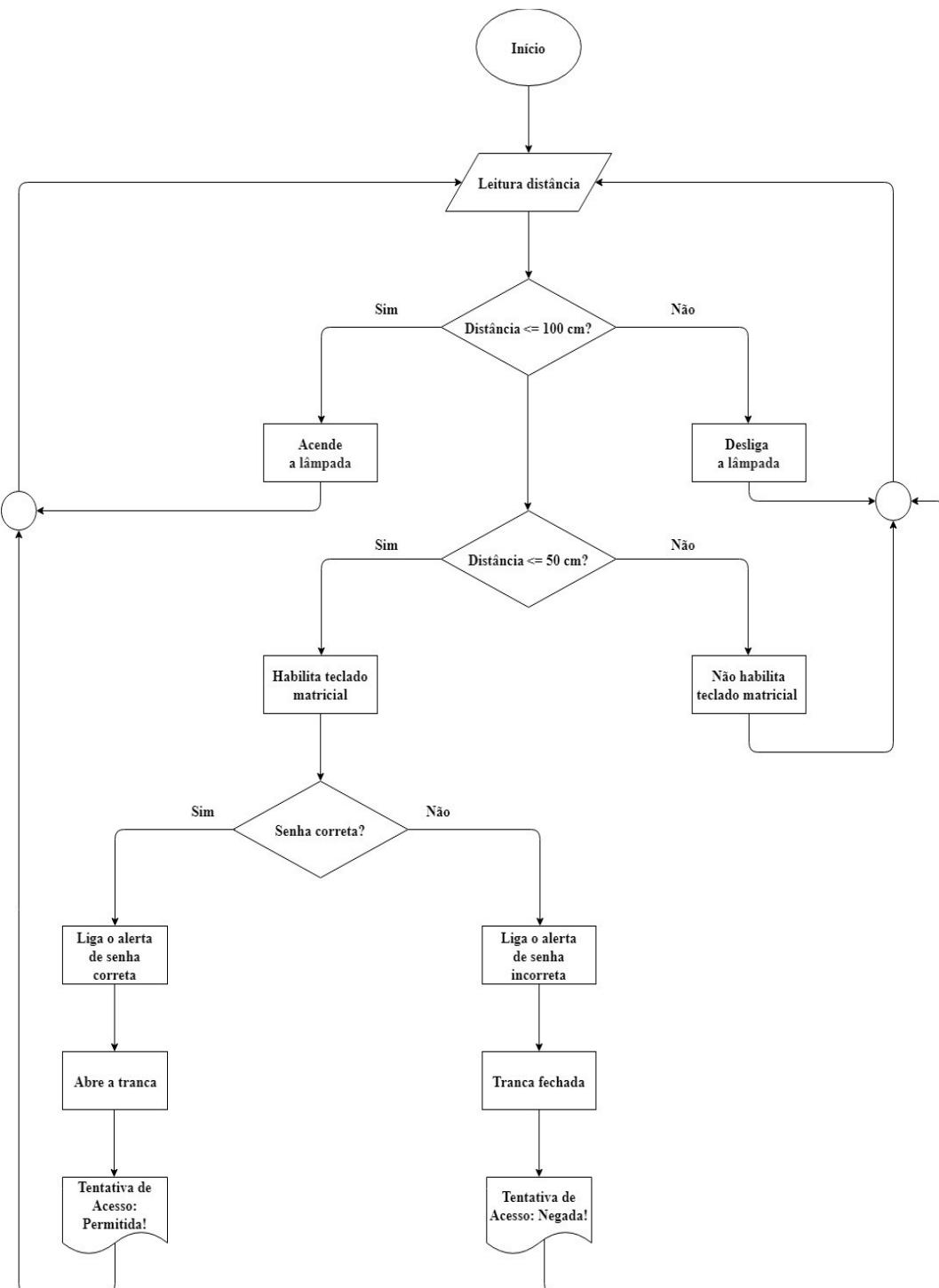


Figura 1

Depois das definições das portas, foi feito funções para o controle do sistema. As funções são as seguintes:

- void função_Control(): que tem como objetivo acender a lâmpada do ambiente e liberar o acesso ao teclado., tudo isso de acordo com uma distância, que é passada como parâmetro.

- void servoMotor(): faz o motor girar 90 graus e depois retornar para posição original.
- long sensorUltrassonico(): é responsável por receber o valor de leitura do sensor e retornar na unidade de medida centímetro.
- void tecladoMatricial(): função utilizada para receber a senha digitada no teclado e verificar se bate com a senha armazenada, se sim, acionar abre a fechadura, representado pelo servo motor.
- void ligarAlarme(): função para mostrar que a senha digitada no teclado seja correta, utilizando led e buzzer.
- void desligarAlarme(): função para mostrar que a senha digitada no teclado foi errada, utilizando led e buzzer.

Após a implementação do código Arduino, fizemos a montagem dos circuitos elétricos, como pode ser visto na figura 2 abaixo, tem apenas uma modificação em relação ao real, o teclado mostrado aí é diferente do que estamos utilizando.

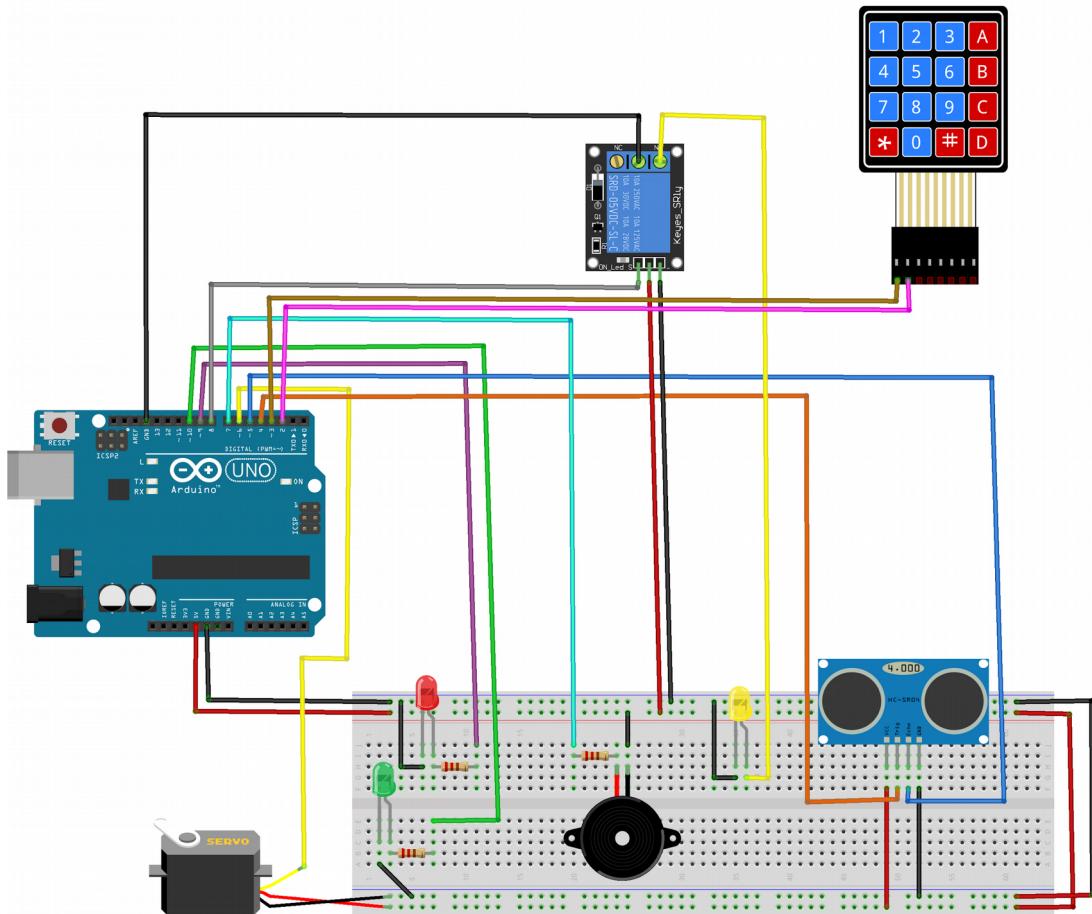


Figura 2

Já no código Python, a implementação do servidor e do cliente foi bem simples. No servidor foi definido uma função chamada “conectado”, para quando um cliente se conectar ao servidor e fazer essa comunicação. Quando o cliente conecta ao servidor, uma thread é iniciada e quando o cliente termina a comunicação essa thread é encerrada. Além disso, o servidor é o responsável por receber a comunicação serial, ele envia os dados para o cliente poder fazer a análise dele, no nosso caso, ele envia os dados para o cliente saber o que está acontecendo, se há alguma movimentação no ambiente, se alguém tentou ter acesso a outro ambiente através da senha no teclado e etc. Já o contador de tempo foi utilizado de seguinte maneira, implementamos uma função do python que envia um dado para o arduino, após esse recebimento o arduino envia os seus dados de para o python poder processar, isso tudo a cada “x” segundos, definidos no código.

Os resultados foram satisfatórios, gerando o que já era esperado. Primeiro nós fizemos uma análise através do osciloscópio do sinal PWM do servo motor utilizado, como pode ser visto nas figuras 3, 4, 5 abaixo, fizemos também uma análise do PWM do sensor ultrassônico, como pode ser visto nas figuras 6, 7 abaixo:

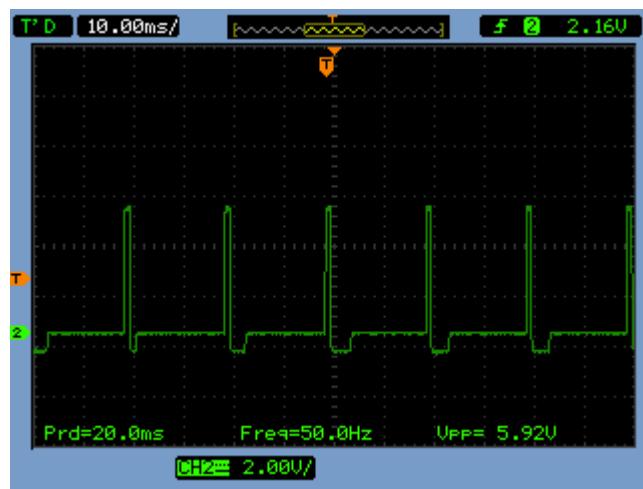


Figura 3

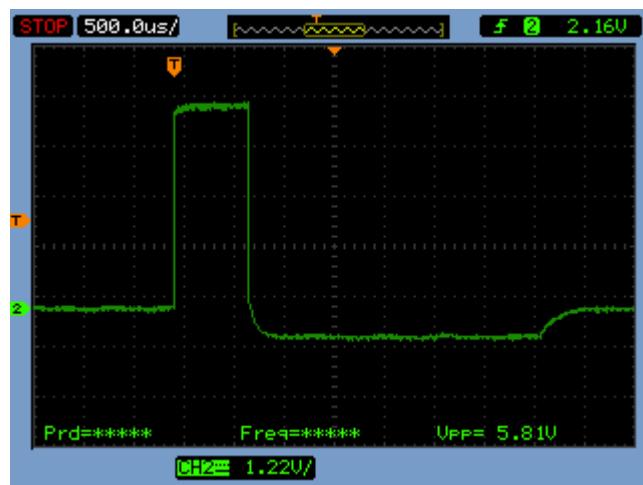


Figura 4

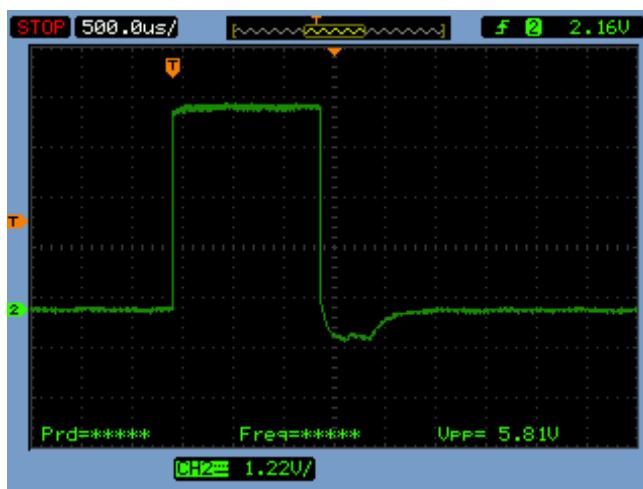


Figura 5

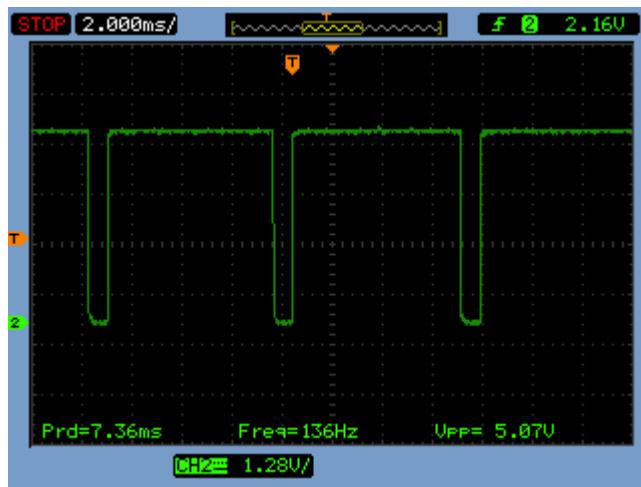


Figura 6

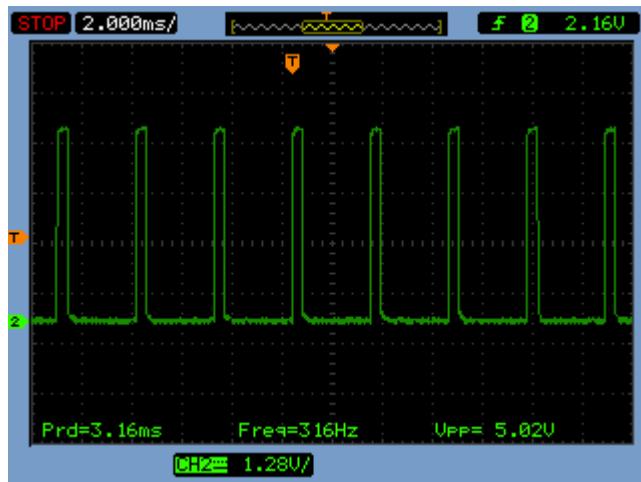


Figura 7

Os resultados práticos podem ser visto também em vídeos onde o links se encontram logo abaixo:

<https://drive.google.com/open?id=1LMiBi8HB1rEc2GQDzCrD-8aPES7mTTDI>

<https://drive.google.com/open?id=1E9I0vVKi9U-KYVqotZvbIMZ1teOZHObR>

5 Conclusão

Com a execução do projeto e resultados coletados pela implementação do projeto de sistema de controle residencial, é possível concluir que a utilização das tecnologias de hardware e software em um uso unificado, neste caso, fazendo uso das bibliotecas, dos módulos, dos componentes e sensores e a placa de arduino, assim como, a linguagem de alto nível Python, para construir a solução ao longo deste documento apresentada, constitui a viabilidade para execução de projetos de aplicações para utilização comercial e doméstica, haja visto o custo da implementação.

Referências

MACHADO, Alisson. **Socket em Python.** Disponível em: <>. Acesso em: 21/08/2019.

ARDUINO. Disponível em: <>. Acesso em: 25/08/2019.

ORTEP, Grupo. **A Importância do Controle de Acesso em Condomínios Residenciais.**

Disponível em: <>. Acesso em: 29/08/2019.

ANTONIO, Marcos. **A cada hora, uma casa é invadida em SP; o número de roubos a imóveis é o maior em 3 anos.** Disponível em: <>. Acesso em: 29/08/2019.

Python 2.7.16 documentation. Disponível em: <>. Acesso em: 30/08/2019.