

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

**ROBERTO GONÇALVES
MIGUEL LUIZ DE AZEVEDO**

**DOMÓTICA: CONTROLE DE ILUMINAÇÃO, CLIMATIZAÇÃO E SEGURANÇA DE
UMA RESIDÊNCIA.**

**CURITIBA
2021**

**ROBERTO GONÇALVES
MIGUEL LUIZ DE AZEVEDO**

Domótica: Controle De Iluminação, Climatização E Segurança De Uma Residência.

Home Automation: Lighting Control, Climate Control and Security of a Home.

Trabalho de conclusão de curso de graduação apresentada como requisito para obtenção do título de Tecnólogo em Sistemas de Telecomunicações da Universidade Tecnológica Federal do Paraná (UTFPR).
Orientadora: Prof^a. Dra. Simone Crocetti.

CURITIBA

2021



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

Esta licença permite download e compartilhamento do trabalho desde que sejam atribuídos créditos ao(s) autor(es), sem a possibilidade de alterá-lo ou utilizá-lo para fins comerciais. Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.

**ROBERTO GONÇALVES
MIGUEL LUIZ DE AZEVEDO**

**DOMÓTICA: CONTROLE DE ILUMINAÇÃO, CLIMATIZAÇÃO E SEGURANÇA DE UMA
RESIDÊNCIA.**

Trabalho de Conclusão de Curso de Graduação
apresentado como requisito para obtenção do título de
Tecnólogo em Sistemas de Telecomunicações da
Universidade Tecnológica Federal do Paraná (UTFPR).

Data de aprovação:

Tânia Monteiro
Doutorado
Universidade Tecnológica Federal do Paraná

Daniel Rossato de Oliveira
Mestrado
Universidade Tecnológica Federal do Paraná

Simone Crocetti
Doutorado
Universidade Tecnológica Federal do Paraná

**CURITIBA
2021**

Este trabalho é dedicado à Deus, que encheu nossos corações de paz e foi essencial para a nossa parceria. É dedicado também aos familiares, por todo amor e paciência, e por entenderem os momentos de ausência.

AGRADECIMENTOS

Diante de todos os desafios encontrados e do sucesso no resultado final, uma página apenas é pouco para agradecer aos envolvidos direta ou indiretamente neste trabalho.

Em primeiro lugar agradecemos a Deus, que nos capacitou para desenvolver este trabalho e, nos fortaleceu diante de todo o cansaço físico e emocional encontrados nesta longa e difícil caminhada. Obrigado por não nos deixar desistir e, ao mesmo tempo, nos ensinar a linda lição da persistência.

Agradecemos as nossas esposas, filhas e filhos, que tiveram paciência em muitas de nossas ausências, nos consolaram em momentos de abatimento por insucessos, nos animaram em momentos em que o desânimo apareceu e sempre acreditaram em nós, em situações em que nós mesmos perdemos as esperanças. Sem vocês nada deste sucesso teria acontecido.

Aos nossos pais que, sempre direcionaram suas orações e boas energias em nosso favor. Sempre nos deram a melhor educação que lhes era possível e, em muitas vezes, passaram necessidades para nos suprir.

Aos irmãos, irmãs, e demais familiares, que compreenderam muitas de nossas ausências em festividades ou outros encontros, pois entenderam que precisávamos nos ausentar para obter sucesso nesta empreitada.

Somos gratos a Prof^a Dra. Simone Crocetti, que no momento em que estávamos desacreditados, foi ela, uma das poucas, que acreditou e depositou sua confiança em nós. Nos deu total apoio e auxílio no que era preciso. Colocou todo o seu enorme conhecimento ao nosso dispor e, com isso, foi um sucesso. Aos demais professores da universidade, que nos ensinaram, dedicaram seu tempo e conhecimento em nosso benefício.

Enfim, agradeço a todos que fizeram parte da nossa vida nesta etapa de muitas alegrias e aprendizados.

RESUMO

Os constantes avanços pelos quais o conhecimento humano vem passando, têm favorecido para que a indústria dos eletrodomésticos e eletroeletrônicos residenciais estejam gradativamente sendo modernizados, oferecendo a opção de serem acessados pela internet, e até mesmo, através de um pequeno aplicativo de aparelho celular. Alguns elementos não são contemplados por essa modernidade, como por exemplo: janelas, cortinas e iluminação. A tendência em um futuro próximo é que uma casa esteja totalmente conectada, tornando-se, então, em uma residência tecnologicamente inteligente. O trabalho realizado, traz como alvo a apresentação de domótica, ou seja, uma automação residencial. Neste trabalho é proposta a automação de janelas, iluminação e tomadas, sendo acessados por um dispositivo conectado à internet com a segurança de dados do usuário. Esta solução foi demonstrada em um protótipo com controle real da aplicação proposta. Nessa perspectiva, destaca-se um sistema de criptografia em multicamadas ligado por meio de login, senha e acesso por VPN (*Virtual Private Network*). O *hardware* é composto por um servidor que controla os módulos, hospeda uma página *web* e todo o sistema de segurança da informação. Os módulos controlam os elementos do cômodo em que está alocado, devolvendo a informação sobre o estado desses elementos para o servidor, sendo que esses estados são visualizados pelo usuário na página *web*.

Palavras-chave: Casa inteligente; Automação residencial; Internet; Segurança.

ABSTRACT

The constant advances through which human knowledge has been passing have favored the industry of household appliances and home appliances being gradually modernized, offering the option of being accessed over the internet, and even through a small mobile device application. Some elements are not covered by this modernity, such as: windows, curtains and lighting. The trend in the very near future is for a home to be fully connected, thus becoming a technologically smart home. The work carried out has as its target the presentation of a home automation, that is, a home automation. This work proposes the automation of windows, lighting and sockets, being accessed by a device connected to the internet with the user's data security. This solution was demonstrated in a prototype with real control of the proposed application. In this perspective, a multilayer encryption system connected through login, password and VPN access stands out. The *hardware* is made up of a server that controls all the modules, hosts a *web* page and the entire information security system. The modules control the elements of the room in which they are allocated, returning information about the status of these elements to the server, and these statuses are viewed by the user on the *web* page.

Keywords: Smart home; Home automation; Internet; Safety.

LISTA DE ILUSTRAÇÕES

Figura 1 – Planta baixa de um exemplo do ambiente real.....	28
Figura 2 – Vista externa do protótipo.....	29
Figura 3 – Vista interna do protótipo.....	29
Figura 4 – Ambiente do usuário.....	30
Figura 5 – Estado dos elementos.....	31
Quadro 1 – Tensões de funcionamento dos elementos.....	32
Figura 6 – Esquema elétrico do módulo auxiliar.....	33
Figura 7 – Transmissor e receptor RF.....	34
Figura 8 – Diagrama em blocos geral.....	34
Figura 9 – Transporte da chave de criptografia.....	35
Figura 10 - Topologia da aplicação.....	35
Quadro 2 – Resumo da avaliação dos resultados.....	38
Quadro 3 – Comparativo entre os Raspberry Pi.....	39

LISTA DE ABREVIATURAS E SIGLAS

ASP	Active server pages
Aureside	Associação Brasileira de Automação Residencial e Predial
API	Application Programming Interface
CSS	Cascading Style Sheets
DDOS	<i>Distributed Denial of Service</i>
FTP	File Transfer Protocol
GPIO	General Purpose Input/Output
HTML	HyperText Markup Language
ICMP	Internet Control Message Protocol
IoT	Internet of Things
IP	Internet Protocol
LED	Light-Emitting diode
MHz	Megahertz
NAT	Network Address Translation
PIC	Peripheral Interface Controller
PHP	HyperText Preprocessor
RF	Rádio frequência
SD	Secure Digital
SEO	Search Engine Optimization
SGBD	Sistema de Gerenciamento de Banco de Dados
SSH	Secure Socket Shell
VCA	Voltage Current Alternate
VDC	Voltage Direct Current
VM	Virtual Machine
VPN	Virtual Private Network

SUMÁRIO

1 INTRODUÇÃO	11
1.1 Problema	11
1.2 Objetivos	12
1.2.1 Objetivo Geral.....	12
1.2.2 Objetivos Específicos	12
1.3 Justificativa	13
1.4 Procedimentos Metodológicos	13
1.5 Estrutura Do Trabalho	14
2 FUNDAMENTAÇÃO TEÓRICA	16
2.1 Domótica	16
2.2 Python	18
2.3 HTML	19
2.4 Raspberry Pi	20
2.5 Rádio Frequência (RF)	21
2.6 Banco De Dados	22
2.7 Segurança Da Informação	22
2.7.1 Conceito De Segurança Da Informação.....	23
2.7.2 VPN.....	25
3 ESTRUTURA DA SIMULAÇÃO EM MAQUETE	27
3.1 Ambiente Físico	27
3.2 Protótipo em Maquete	28
3.3 Ambiente do Usuário	29
3.4 Estado dos Elementos	30
3.5 Servidor	31
3.6 Módulo Físico	31
3.7 Transmissão dos Dados	33
3.8 Segurança dos Dados	34
3.9 SGBD	36
4 APRESENTAÇÃO E ANÁLISE DOS RESULTADOS	37
4.1 Avaliação dos Resultados	37
4.2 Dificuldades Encontradas no Trabalho	38
5 CONSIDERAÇÕES FINAIS	40
APÊNDICE A – PARTE DE CÓDIGO FONTE DO SERVIDOR	46

APÊNDICE B – PARTE DE CÓDIGO FONTE DO MÓDULO.....48

1 INTRODUÇÃO

Com o avanço tecnológico da atualidade, as residências não poderiam ficar de fora das novas tendências. Um avanço já é a própria Internet, porém, em se tratando de robótica e automação que estão atrelados à *web*, surge um novo elemento conhecido como IoT (*Internet of Things*), que veio para ser um grande aliado desse segmento. A automação residencial recebeu o nome de Domótica que é a fusão de duas palavras, Domus que significa casa e robótica que é o ato de automatizar. Com o advento dos *smartphones*, essa automação tornou-se uma tendência muito forte e o conceito de casas inteligentes tem uma boa perspectiva de crescimento. Segundo Muratori (2018), a expectativa é que em dois anos a tecnologia chegue a dois milhões de imóveis no País. Por enquanto, são 300 mil residências, de um total de 60 milhões com algum tipo de automação inteligente. Esse crescimento se dá pela procura por imóveis personalizados e inteligentes que facilitem a rotina e aumentem o conforto, e nessa perspectiva, casas conectadas significam mais conforto, segurança, sustentabilidade, economia financeira e de recursos naturais. O trabalho desenvolvido tem como objetivo atender a seguinte demanda: Como controlar uma residência à distância? Para simular o ambiente proposto, foi desenvolvido um protótipo em MDF (*Medium Density Fiberboard*), que realizou as atividades de abrir e fechar uma janela, acender e apagar uma lâmpada, ligar e desligar uma tomada.

1.1 Problema

Uma das dificuldades evidenciadas no dia a dia da população, está relacionada com as constantes variações climáticas e, devido aos compromissos fora de casa, fazem com que o indivíduo esteja sujeito à algumas dificuldades no que diz respeito a residência. A ventilação dos ambientes residenciais são importantes, independentemente da estação do ano, pois no inverno há uma predominância do tempo seco e do aparecimento de microorganismos nocivos, sendo necessário a ventilação para se evitar que doenças transmissíveis fiquem pelo ar. Essa afirmação é confirmada pela arquiteta Isabel Moratelli (2018) que diz:

“Esse é o lema contra a umidade. Abra as janelas da casa e deixe o ar e o sol circular pelas peças pelo menos duas vezes por semana. Assim, há menos chances da umidade se acumular e de aparecerem bolhas, mofo e manchas em sua casa. Não deixe tudo fechado por estar frio. É a maior

arma. Se a casa não ventilar, independentemente da peça, a umidade toma conta. Não há produto que adiante.” (MORATELLI, 2018)

Já no verão, esse controle se faz mais necessário pois é o período mais chuvoso do ano, isso é o que afirma o site Brasil Escola (2018): “o verão caracteriza-se por apresentar temperaturas mais elevadas e altos índices pluviométricos”. Não sendo viável abrir as janelas sem ter pessoas em casa, haja vista que uma chuva forte e as janelas estando abertas poderão trazer grandes transtornos.

Diante das variações climáticas e dos problemas abordados, a automação residencial vem também para auxiliar o indivíduo. As cortinas conectadas fazem o conjunto com as janelas, mas podem trabalhar independentemente. O acionamento remoto das tomadas podem acionar eletrodomésticos e eletroeletrônicos trazendo conforto aos moradores. Lâmpadas ligadas sem necessidade acabam gerando um gasto desnecessário de energia elétrica, além da redução da vida útil da lâmpada.

Diante de todas estas situações surge uma pergunta: Como ter acesso ou controle das instalações de uma residência à distância? Isso é possível por meio de elementos automatizados, controlados à distância por qualquer dispositivo móvel conectado à internet, previamente cadastrado, evitando fraudes e invasões cibernéticas.

1.2 Objetivos

Nesta seção são apresentados os objetivos gerais e específicos do trabalho relativos ao problema anteriormente apresentado.

1.2.1 Objetivo Geral

Controlar uma residência por dispositivo móvel, conectado à internet, utilizando os métodos para controle de acesso de usuário. Com atenção à segurança da informação.

1.2.2 Objetivos Específicos

Nesta subseção estão apresentados os objetivos específicos. Ao todo são quatro objetivos que permeiam todo o trabalho. Por meio da simulação em uma maquete é possível:

- Acionar, pela internet, os elementos de uma residência, como abrir e fechar uma janela, acender e apagar uma lâmpada, ligar e desligar uma tomada.
- Implantar um sistema de segurança de dados do usuário por VPN (Virtual Private Network), autenticação por usuário e senha.
- Modularizar o sistema de automação para os vários ambientes de uma residência fazendo com que cada cômodo tenha um controlador.
- Elaborar um sistema que proporcione maior conforto para os habitantes residenciais, oferecendo uma interface amigável de controle.

1.3 Justificativa

A tecnologia residencial vem crescendo em ritmo acelerado. Esse avanço se deve às preocupações com a segurança, conforto e sustentabilidade. O trabalho proposto foi motivado por entender a realidade das famílias que, não conseguem controlar suas janelas, tomadas e lâmpadas estando fora de casa o dia inteiro. As famílias sentem-se seguras deixando as janelas fechadas com receio de chuvas que podem molhar seus cômodos, mas não conseguem prover ventilação em suas casas. Outro problema é esquecer uma lâmpada acesa ou aparelho ligado desnecessariamente o dia inteiro, pois é um gasto de energia elétrica em vão.

A proposta é simular esse ambiente em escala menor, com uma maquete que estarão instalados os elementos de janela, lâmpada e tomada inteligente. Após a simulação na maquete serão vistas as dificuldades encontradas e as prováveis soluções para a resolução dos problemas, para posteriormente ser implantada em um ambiente real, onde a proposta estaria completa.

O sistema de segurança desta implantação alerta sobre qualquer ação nas janelas da residência, ou anomalias do mesmo elemento, enviando um aviso ao usuário do sistema, para que o mesmo possa efetuar as devidas ações. A segurança de dados do cliente neste projeto está segura, pois neste caso, o invasor teria que ultrapassar camadas robustas de validação como login, senha e acesso ao servidor por uma VPN.

1.4 Procedimentos Metodológicos

Nesta seção, será exposta a metodologia do trabalho, subdividida em 4 partes: Objetivo, Abordagem, Natureza e Procedimentos adotados. O objetivo do

trabalho é de cunho exploratório, ou seja, compreender um problema proposto, tornando-o mais conhecido e familiarizado à automação residencial dos leitores. A tecnologia de *hardware* e software detalhada ao longo do texto foi amplamente explorada. Muitos elementos foram estudados e analisados com o objetivo de encontrar uma interface amigável e um sistema funcional ao usuário.

A abordagem do trabalho é quantitativa. A pesquisa é quantitativa pelo fato de haver comparações representadas numericamente, graficamente e por imagens entre diferentes *hardwares* e *software* que serão facilmente visualizadas, principalmente na seção de apresentação e análise dos resultados no intuito de encontrar soluções práticas, simples e que possam satisfazer os objetivos propostos.

A natureza da pesquisa é aplicada, pois gera conhecimento para as aplicações práticas voltadas à solução do problema proposto no trabalho. O conhecimento citado está especificado principalmente na seção de fundamentação teórica que detalha todas as aplicações do trabalho.

O procedimento adotado foi o estudo de caso. A escolha desse procedimento foi essencial diante da necessidade de estudos profundos de alguns objetos adotados no trabalho, os quais estão relacionados a *hardware* e *software*, havendo uma seção específica de estudo de caso, dada sua importância para o trabalho.

1.5 Estrutura Do Trabalho

O trabalho está dividido em capítulos e subcapítulos: A estrutura está abaixo apresentada:

Capítulo 1 - Introdução: São apresentados, o problema, os objetivos da pesquisa, a justificativa, os procedimentos metodológicos, as indicações para o embasamento teórico, e a estrutura geral do trabalho. Estão descritos de forma resumida, a fim de se especificar brevemente o conteúdo geral do trabalho.

Capítulo 2 – Fundamentação teórica: São abordadas as linguagens HTML (HyperText Markup Language) e Python, a segurança da informação aplicada, o *hardware* utilizado que é o Raspberry pi, a transmissão via rádio frequência e o banco de dados. Toda a teoria está embasada em artigos, publicações científicas, livros especializados, entre outros.

Capítulo 3 – Estrutura da Simulação em Maquete: É abordada a parte prática do trabalho, explanando e ilustrando com imagens da página de acesso e o protótipo do trabalho entre outras. Neste capítulo estão apresentados detalhadamente cada fase e divisão do trabalho.

Capítulo 4 – Apresentação e Análise dos Resultados: São descritos os resultados obtidos e feitas as devidas análises relacionados a automação residencial. São descritas também algumas dificuldades enfrentadas ao longo da execução do trabalho.

Capítulo 5 – Considerações finais: São retomadas a pergunta de pesquisa e os seus objetivos, apontado como eles foram solucionados, respondidos, atingidos, por meio do trabalho realizado. Além disto, serão sugeridos trabalhos futuros que poderão ser realizados a partir do estudo realizado.

2 FUNDAMENTAÇÃO TEÓRICA

Para uma explanação desse tema deve ser considerado o *hardware* e *software* da aplicação proposta. Em relação ao software devem ser consideradas as linguagens de programação utilizadas, e para o *hardware* os dispositivos escolhidos, considerando suas configurações na perspectiva de um melhor desempenho e visando os menores custos.

2.1 Domótica

O conceito de domótica está detalhada neste sub capítulo. Segundo Mattede (2020):

“é a fusão de duas palavras: Domus que vem da palavra residência e a palavra robótica que, é o ato de realizar ações automáticas, e tem como objetivo controlar e automatizar equipamentos elétricos de forma local ou remota. A solução tem como objetivo trazer mais conforto, segurança e otimização de recursos da residência.” (MATTEDE, 2020)

É necessário conhecer como iniciou-se a tecnologia para entender a evolução da mesma. Em um breve relato da história, Adminwebbizz (2017) revela que a domótica,

“surgiu nos anos 70 quando a automação industrial dava um impulso e mudar a história industrial. Na época os engenheiros eletricitas começaram a pensar em formas de se levar essa automação para as residências e prédios comerciais. Os primeiros avanços foram mais tímidos e voltados para os prédios comerciais, com custos elevados e poucas empresas que ofereciam o serviço, que basicamente controlavam a circulação do ar e a intensidade das luzes, chamada de dimerização. Esses avanços prometiam transformar as casas em ambientes inteligentes, mas essa promessa foi se cumprir nos anos 2000, com a evolução e popularização da internet banda larga e dos smartphones”. (ADMINWEBBIZZ, 2017)

Após apreciar o surgimento da tecnologia, o ideal é entender em qual patamar se encontra a domótica no cenário de hoje. Atualmente, ao considerar o assunto Cordella afirma que,

“a domótica procura uma melhor integração através da automação nas áreas da segurança, comunicação e controle e gestão de fluídos, utilizando alguns elementos como sensores, transdutores, detectores e redes domóticas. Dentre as áreas de abrangências tem as funções domóticas:

- As necessidades de segurança, estão relacionadas com:
- A qualidade do ar.
- A prevenção de acidentes físicos e materiais.

A assistência à saúde.

A segurança anti-intrusos.

As necessidades de conforto ambiental, implicam na criação de um meio ambiente agradável:

Conforto térmico.

Conforto acústico.

Conforto visual.

Conforto olfativo.

Conforto espacial.

As necessidades de conforto de atividades, vêm facilitar os hábitos cotidianos:

Para dormir.

Para alimentar-se.

Para cuidar-se.

Para manutenção (dos locais e dos materiais).

Para comunicar-se.

Para divertir-se.

Para trabalhar. ” (CORDELLA, 2012)

Como exemplo de projeto de domótica, Fazano Jr (2013) produziu um protótipo de madeira, simulando um ambiente real com o objetivo de monitorar a temperatura ambiente, a umidade relativa do ar, o monitoramento do estado das janelas, o acionamento das luzes, a abertura e fechamento do portão de entrada, tudo isso acessado de forma local. Foi utilizado um arduíno como *hardware* principal com um shield ethernet acoplado para hospedagem da página *web*, sendo conectado a um roteador *wifi* para fazer uma ponte entre o dispositivo de acesso sendo um *notebook* ou um *smartphone* e o *hardware* principal. Foram utilizados sensores magnéticos para monitoramento do estado das janelas. Para monitoramento da umidade relativa do ar e temperatura, foi instalado um sensor NTC (Negative Temperature Coefficient) o qual possui um componente resistivo responsável por medir umidade e temperatura, sendo básico e de baixo custo. Para a movimentação do portão foi utilizado um servo motor que rotaciona entre 0 a 180°. Para o acionamento das lâmpadas foi instalado um relé em paralelo com o interruptor. Todos os dispositivos de entrada e saída foram gerenciados pelo *shield ethernet* acoplado no arduíno. O arduíno foi utilizado como servidor da página *web* desenvolvida em HTML (*Hypertext Markup Language*) que gerencia os elementos da residência.

2.2 Python

Há um consenso afirmando que esta linguagem é versátil e simples, e por este motivo está sendo explorada. Python, segundo Stumm (2013), “é uma linguagem muito versátil, podendo ser utilizada para vários fins. Um dos nichos em que Python mais tem sido usada ultimamente é no desenvolvimento *web*. E quando se trata de *web*, temos várias ferramentas disponíveis para usar com Python”. Essa linguagem de programação tem pontos positivos a serem avaliados. No site Pyscience Brasil são avaliadas algumas vantagens:

“Python é uma linguagem expressiva, em que é fácil traduzir o raciocínio em um algoritmo. Em aplicações científicas, o raciocínio é essencialmente complicado — essa é a natureza das ciências. Python é legível. Isso significa que é fácil compreender programas escritos há algum tempo. É comum que os programas em atividades científicas sejam criados a partir da evolução de algoritmos anteriores. Portanto, é importante ser capaz de entender o que foi feito antes.” (PYCIENCE BRASIL, 2018)

A participação da comunidade de desenvolvedores é que chama a atenção em Python. O que o site pontua é que:

“Python tem uma comunidade ativa e vibrante, espalhada por todo o mundo. E, sendo uma linguagem livre, todos os seus usuários estão dispostos a contribuir (o site é um exemplo disso, já que todos seus contribuidores são voluntários). Isso faz com que a documentação seja abundante e existam módulos para executar virtualmente qualquer tarefa necessária. Isso é importante: não há tempo para reinventar a roda, então poder contar com módulos prontos é ótimo. Mas, mais que isso, uma vez que os programas em Python são distribuídos na forma de código-fonte, qualquer pessoa pode alterar, corrigir e melhorar os algoritmos. Isso faz com que os módulos sejam maduros e seguros, testados contra diversas situações e diversas vezes. A robustez alcançada é um fator importante.” (PYCIENCE BRASIL, 2018)

Alguns detalhes são relevantes para a integração entre as linguagens. Para o site Devfuria (2016),

“Para que páginas *web* sejam servidas por scripts Python devemos ter o servidor *web* devidamente configurado e também precisaremos entender os diferentes protocolos que fazem a ponte entre os scripts Python e o Servidor *web*. Pois, sozinho, o servidor não saberá como fazer a ligação entre a requisição e a linguagem Python”. Muito importante também é a interação de Python com outras tarefas: buscar dados em um banco de dados remoto, ler uma página na internet, exibir graficamente os resultados, criar uma planilha, etc. Linguagens de cunho especificamente científico têm um sério problema aí, mas, uma vez que Python é utilizada em praticamente todo tipo de tarefa, encontram-se módulos prontos para realizar essas tarefas que podem ser tornar complicadas. Novamente, é uma preocupação a menos para quem está desenvolvendo aplicações científicas.” (DEVFURIA, 2016)

Para facilitar a integração com HTML, comunicação por rádio frequência e banco de dados, há a necessidade de se utilizar os *frameworks*. O site (Djangogirls 2020) esclarece que:

“*framework* é um conjunto de componentes que ajuda a desenvolver um site de forma mais rápida e fácil, pois os desenvolvedores de frameworks já deixaram algumas atividades prontas para serem usadas, aliviando parte do trabalho extra na construção de um site. Há dois frameworks conhecidos em programação Python, que são o Django e o Bottle.” (DJANGOGIRLS, 2020)

Visando um melhor desempenho, foi escolhido o *Bottle*, pois o mesmo utiliza menos memória e processamento, evitando uma sobrecarga e e travamento do sistema. Os resultados serão apresentados nas seções posteriores.

2.3 HTML

Esta linguagem é utilizada para construção do layout da página. O Python será integrado com o HTML para executar as atividades de baixo nível. De acordo com Eis (2011),

“ O HTML é uma das linguagens que utilizamos para desenvolver websites. O HTML é a linguagem base da internet. Foi criada para ser de fácil entendimento por seres humanos e também por máquinas”. (EIS, 2011)

Ao longo do tempo o HTML passou por muitas melhorias no intuito de uma interação entre o homem e a máquina. Segundo Marques (2017),

“A versão 5 do HTML foi desenvolvida para aperfeiçoar a experiência da *World Wide web* para os desenvolvedores e usuários finais. Ela trouxe grandes avanços para a linguagem. O HTML 5 fornece o suporte de áudio e vídeo em alto nível que, até então, não existia nas versões anteriores sem o uso de plug-ins. Essa nova versão também nasceu com uma proposta marcante: ela não é uma versão final, ou seja, seguirá recebendo mudanças ao longo do tempo. Anteriormente, as versões não eram padronizadas para criação de seções comuns e específicas como rodapés, cabeçalhos, menus, artigos, entre outros. Com a introdução de tags como <header>, <footer>, <section>, <article>, <nav>, etc, o HTML5 trouxe uma forma semântica de se trabalhar com a linguagem. Com esse avanço, foi possível aumentar a interatividade sem a necessidade de instalação de plug-ins. É um código pronto para futuros dispositivos, facilitando a reutilização da informação de diferentes maneiras. Portanto, possibilita diversas funcionalidades que antes só eram possíveis com plugins, sem que isso sobrecarregasse o website, mantendo-o leve e rápido. Com isso, as novas tags semânticas trouxeram grandes avanços para técnicas de SEO e para a leitura, pois informam ao navegador sobre o significado do conteúdo”. (MARQUES, 2017)

De acordo com o que está mencionado acima, sob o ponto de vista de (EIS, 2011) “há também as linguagens chamadas Linguagens Server-Side, que são linguagens como PHP (HyperText Preprocessor), Python, Ruby, ASP (Active Server Pages) e etc. Essas linguagens fazem tudo funcionar. Elas fazem os cálculos nos servidores e dão a resposta para o navegador do usuário”. (EIS, 2011).

“Mesmo que o HTML seja uma linguagem poderosa, não é totalmente suficiente para criar um site profissional e responsivo. Ele pode ser utilizado apenas para adicionar elementos de texto e criação de estruturas de conteúdo. Porém, o HTML funciona muito bem com outras duas linguagens de front-end: CSS (Cascating Style Sheets) e JavaScript. Juntos eles proporcionam a implementação de funcionalidades avançadas e uma ótima experiência ao usuário. O CSS é responsável pelo estilo como background, cores, layouts, espaçamentos, e animações. O JavaScript permite a adição de funcionalidades dinâmicas como sliders, pop-ups, e galerias de fotos. (L 2019)”.

2.4 Raspberry Pi

Havia duas opções de *hardware*: Arduíno e Raspberry pi. Foi escolhido o Raspberry pi visto que havia uma gama maior de ferramentas disponíveis. Este é um dispositivo bem compacto, mas muito poderoso na realização das atividades do trabalho. De acordo com RPF16RG (2017),

“Raspberry pi é um computador do tamanho de um cartão de crédito, que se conecta a um monitor de computador ou Televisor, e usa um teclado e um mouse padrão, desenvolvido no Reino Unido pela Fundação Raspberry pi. Todo o *hardware* é integrado numa única placa. O principal objetivo é promover o ensino em Ciência da Computação básica em escolas. É um pequeno dispositivo que permite que as pessoas de todas as idades possam explorar a computação para aprender a programar em linguagens como Python. É capaz de fazer tudo que você esperaria de um computador desktop, como navegar na internet, reproduzir vídeo de alta definição, fazer planilhas, processamento de texto, e jogar jogos”. Com isso: “ As placas Raspberry pi também executam uma versão especialmente projetada do sistema operacional Linux. Devido a isso, o software Linux é muito fácil de instalar e, com um pouco de esforço, você pode usá-lo como um emulador de streaming de mídia ou de videogames. O Raspberry pi não possui armazenamento interno, mas você pode usar cartões SD para este propósito. Isso lhe dá muita flexibilidade para testar diferentes sistemas operacionais ou atualizações, especialmente para fins de depuração e, em alguns casos, instalar até mesmo o Windows. Com o Pi, você pode executar vários programas ao mesmo tempo e até mesmo usar a API do Linux . O Raspberry pi permite que a interação com seus programas, utilizando um teclado e mouse, com isso é possível visualizar as ações do Raspberry pi através de um monitor. O Raspberry pi pode ser configurado para acesso via SSH ou transferência de arquivos por FTP devido à sua conectividade de rede independente. Você pode até usá-lo como uma ponte de rede. (VICTOR, 2018,).

Ainda tratando de Raspberry pi, há os pinos GPIO (General Purpose Input/Output), que são usados como entrada e saída. Para ARAÚJO (2014),

“O GPIO (General Purpose Input/Output), é basicamente um conjunto de pinos responsável por fazer a comunicação de entrada e saída de sinais digitais. Ele é composto por 26 pinos no Raspberry pi B, e 40 pinos no Raspberry pi B+. Com estes pinos é possível acionar LEDs, Motores, Relês, fazer leitura de sensores e botões, entre outros”. (ARAÚJO, 2014)

Dentre as placas disponíveis temos os modelos Raspberry pi 2 e o Raspberry pi 3, em que há diferenças significativas. Conforme Thomsen (2018),

As diferenças básicas dessa placa [Raspberry pi 3] para a sua antecessora [Raspberry pi 2] são a presença do Wifi e Bluetooth integrados, além do clock de 1.2 GHz. Vamos começar falando do processador. No Raspberry pi 3, temos um processador quad-core de 64 bits (Broadcom BCM2837) e no Pi 2 temos também um quad-core, mas de 32 bits (Broadcom BCM2836). Esse detalhe por si só já faz um grande diferença, mas também temos a questão do clock: 1.2 GHz no Pi 3, contra 900 MHz no Pi 2. Em um cálculo bem simples, são 300MHz à mais no processador do Raspberry pi 3. Conta ainda com uma arquitetura mais avançada: Cortex-A53 contra o Cortex-A7 no Pi 2.

2.5 Rádio Frequência (RF)

Transmissão de dados via RF têm se tornado comum em aplicações recentes. Essa importância é ressaltada por Oelinton (2018), “As ondas eletromagnéticas são decodificadas para captar informação. Em diversos momentos fazemos uso de ondas de rádio sem percebermos. Em resumo, ao usar a transmissão de dados via rádio podemos assistir televisão, ouvir rádio, acessar à internet e se comunicar”. A transmissão por rádio frequência foi utilizada para evitar a passagem de cabos, reduzindo o custo, conforme STRAUB (2018):

“O sistema de comunicação por radiofrequência é classificado como um sistema de comunicação sem fio que possui como base a utilização de ondas eletromagnéticas em uma faixa de frequência estabelecida de acordo com o equipamento, neste nosso caso, 433Mhz. Quando pretende-se desenvolver um trabalho de comunicação sem fio sempre encontramos problemas quando o assunto é instalação e implementação dos mesmos, ou até mesmo tem vezes que desejamos realizar um simples trabalho de acionamento para portas e portões através de sistemas que possibilitam o acionamento à distância. Para estes processos citados, em grande parte há resistência na hora de desenvolver os mesmos devido à falta de equipamentos capazes de executar tais atividades e os disponíveis para venda são de custo elevado ou de alta complexibilidade para a implementação em seus trabalhos. Pensando nesta dificuldade que muitos encontram durante o desenvolvimento de seus trabalhos, trouxemos uma opção barata e de fácil implementação que trabalha com sistemas de comunicação de rádio frequência, utilizando-se de um módulo transmissor e outro receptor que permitem a comunicação entre dois pontos distintos.” (STRAUB, 2018)

2.6 Banco De Dados

Esse ambiente é muito comum no ramo de programação para armazenamento e organização das informações. Segundo o site Oracle (2019),

“Um banco de dados é uma coleção organizada de informações - ou dados - estruturadas, normalmente armazenadas eletronicamente em um sistema de computador. Um banco de dados é geralmente controlado por um sistema de gerenciamento de banco de dados (SGBD). Juntos, os dados e o SGBD, juntamente com os aplicativos associados a eles, são chamados de sistema de banco de dados, geralmente abreviados para apenas banco de dados.” (ORACLE, 2019)

Pela simplicidade da aplicação, foi determinado o uso do SQLite. Essa simplicidade e a escolha desta aplicação foi baseada no artigo de Estrella (2019):

“São necessárias apenas poucas configurações com a SQLite, que é uma alternativa muito fácil de configurar. Já a MySQL requer uma quantidade maior de configurações, se comparada com a outra opção. Existem certos momentos em que a SQLite pode ser mais efetiva do que usar a MySQL, principalmente quando se desenvolve pequenos aplicativos independentes, trabalhos menores que não exigem muita escalabilidade, quando tem a exigência de ler e escrever direto da unidade de armazenamento e desenvolvimento básico e testes. A SQLite é Baseada em arquivos e fácil de configurar e usar, adequada para desenvolvimento e testes básicos, facilmente transportável, usar sintaxe SQL padrão com alterações mínimas, fácil de usar.” (ESTRELLA, 2019)

2.7 Segurança Da Informação

Na área de segurança da informação há a necessidade de aprofundamento em alguns assuntos, dentre eles será comentado o conceito de segurança da

informação e VPN. Neste caso a subseção está dividida em 2 partes: Conceito de segurança da informação e VPN.

2.7.1 Conceito De Segurança Da Informação

Um sistema organizado de informação é definido por Cunha (2013) como “um conjunto organizado de pessoas, *hardware*, *software*, redes de comunicação e recursos de dados que coleta, transforma e dissemina informações em uma organização.” e também dá uma segunda definição que é,

“um conjunto de componentes inter-relacionados que coleta (ou recupera), processa, armazena e distribui informações destinadas a apoiar a tomada de decisões, a coordenação e o controle de uma organização. Além de dar suporte ao processo decisório, à coordenação e ao controle, sistemas de informação podem também auxiliar gerentes e trabalhadores a analisar problemas, visualizar situações complexas, e criar produtos.” (CUNHA, 2013)

Para o Portal GSTI (2020), segurança da informação são “medidas necessárias para garantir a confidencialidade, integridade e disponibilidade das informações, tanto de empresas como de pessoas físicas, minimizando os riscos de perdas da confidencialidade, integridade e a disponibilidade”. Conforme Hintzbergen (2018) “O hexagrama parkeriano complementa a segurança da informação com mais três atributos: posse ou controle, autenticidade e utilidade”.

Incertezas, ameaças e vulnerabilidades são pontos que devem ser observados com atenção. Um risco no que diz respeito a segurança da informação, Hintzbergen (2018) define como:

“Efeito das incertezas sobre os objetivos. É muitas vezes expressos em termos de uma combinação entre os eventos da segurança da informação e a sua probabilidade de ocorrência. O risco à segurança da informação está associado ao potencial de ameaças explorarem as vulnerabilidades de um ativo de informação ou grupo de ativos de informações, desse modo causando danos a organização.” (HINTZBERGEN, 2018)

De acordo com o mesmo portal, GSTI (2020), “a confidencialidade é o princípio de segurança que garante que os dados, ou informações, devam ser acessados somente por pessoas autorizadas”. Segundo Hintzbergen (2018),

“o conceito de confidencialidade busca prevenir a divulgação intencional ou não intencional do conteúdo de uma mensagem. A perda da confidencialidade pode ocorrer de diversas maneiras, tais como a divulgação intencional ou não intencional de uma informação privada de uma empresa ou pelo mau uso das credenciais de acesso à rede. A confidencialidade assegura que o nível necessário de sigilo seja aplicado em cada elemento de processamento de dados que impede a divulgação não autorizada. Esse nível de confidencialidade deve prevalecer enquanto as informações estiverem no sistema, na transmissão e na chegada ao destino. Essa confidencialidade é fornecida por criptografia dos dados ao serem armazenados e transmitidos nos dispositivos ou na rede”.

Segundo o portal GSTI (2020), “a integridade garante que os dados acessados serão modificados somente por pessoas autorizadas”. Hintzbergen afirma que

“a integridade se refere a ser correto e consiste com o estado ou a informação pretendida. A modificação não autorizada, de forma acidental ou não acidental é uma forma de violação da integridade dos dados. A informação pode ser incorreta ou não autêntica, mas possuir integridade, ou ser correta e autêntica, mas faltar integridade. Ambientes que reforçam atributos de segurança asseguram que atacantes, ou erros de usuários, não comprometam a integridade das informações ou dos sistemas. A integridade pode ser garantida em grande parte por meio de técnicas de criptografia, que protege a informação de mudanças não autorizadas. Esses princípios de criptografia podem ser definidos por políticas e gestão de segurança, devidamente documentadas”. (HINTZBERGEN, 2018)

O portal GSTI (2020) conceitua “a disponibilidade como a garantia que, os dados que serão acessados por pessoas autorizadas, estejam disponíveis no momento que for requisitada”. Hintzbergen (2018) coloca três características de disponibilidade:

“Oportunidade, que é a disponibilidade da informação, quando necessário. Continuidade, que é quando a equipe consegue realizar as atividades mesmo com uma falha. Robustez, que garante capacidade suficiente para que toda a equipe trabalhe no sistema. A disponibilidade do sistema pode ser afetada pela falha de um dispositivo ou software. Por isso os dispositivos de backup podem suprir rapidamente sistemas críticos de forma temporária, desde que os funcionários sejam treinados para atuar e realizar os ajustes necessários com a intenção de restaurar o sistema. Fatores ambientais como frio, calor, eletricidade, estática, umidade e contaminantes também podem afetar o sistema. Sistema de estabilização de energia elétrica, temperatura e umidade são essenciais para um funcionamento adequado do sistema. Os ataques de negação de serviço são fatores que podem causar instabilidade e indisponibilidade dos sistemas, que podem ser evitados com configuração de roteadores e firewalls, reduzindo a incidência destes ataques”. (HINTZBERGEN, 2018)

O atributo de posse e controle é exemplificado como um ladrão que roube um envelope com um cartão de débito ou crédito, e junto a este envelope há a senha, mesmo que o ladrão não abra esse envelope, o proprietário ficará preocupado se o ladrão usará este cartão de forma fraudulenta, a qualquer momento sem o controle do proprietário. Essa situação ilustra a falta de controle ou posse da informação, mas não envolve a quebra do sigilo. A autenticidade é tratada por Hintzbergen (2018) como “a veracidade da alegação de origem ou a autoria das informações. Para informações eletrônicas, uma assinatura digital pode ser usada para verificar a autoria de um documento digital usando criptografia de chave pública”.

Hintzbergen (2018) ainda ressalta sua utilidade como “a capacidade de uso, que muitas vezes é confundida com disponibilidade, pois as falhas, tais como descritas, também podem requerer tempo para solucionar as alterações de formato ou alteração de dados. Entretanto, o conceito de capacidade de uso é diferente de disponibilidade.”

2.7.2 VPN

Muito comum na área de tecnologia da informação, a VPN deve ser estudada com atenção especial. A VPN (*Virtual Private Network*), segundo Branquinho (2014),

“é uma rede de comunicações privada normalmente utilizada por uma empresa ou um conjunto de empresas e/ou instituições, construída sobre uma rede de comunicações pública. As VPNs usam protocolos de criptografia por tunelamento que fornecem confidencialidade, autenticidade e integridade necessárias para garantir a privacidade das comunicações requeridas”. (BRANQUINHO, 2014)

Para complementação da teoria da VPN há uma explicação didática sobre esta ferramenta. Souza (2011),

“uma VPN disponibiliza na rede um caminho que se comporta como uma linha de comunicação entre dois pontos. Essa linha privativa é chamada de túnel e faz com que os dados fiquem em sigilo entre os dois pontos, por meio de criptografia. Os pacotes são encapsulados, criptografados e enviados dentro de outro pacote dentro da rede até o destino final. As VPNs podem ser implementadas dentro dos roteadores, em conjunto com as configurações de *firewall* e Proxy”. (SOUZA, 2011)

A solução utilizada para VPN no trabalho é a OpenVPN. Delfino (2018) explica a OpenVPN:

“O OpenVPN é um pacote repleto de funcionalidades que, quando devidamente configurado, estabelece comunicação segura entre computadores usando a Internet como instrumento de ligação — como se fosse um longo cabo abstrato conectando duas máquinas. Para que a conexão entre os clientes aconteça de maneira segura, o OpenVPN opera por meio da comunicação cliente-servidor, a qual é composta de dois níveis. Esse modo de operação requer duas instâncias — server e client — que interagem entre si. A melhor maneira de se conectar a qualquer VPN é usando o seu próprio cliente de software, visto que um programa específico executa, automaticamente, o complexo trabalho para fazer a conexão acontecer. Em poucas palavras, o usuário somente instala e usa. Entretanto, soluções desse tipo nem sempre atendem às necessidades de uma empresa, por exemplo, que enxergam problemas com interfaces pouco amigáveis, configurações que incluem elementos irrelevantes e mau aproveitamento dos recursos de *hardware*. Um meio eficiente para superar esses obstáculos e continuar usufruindo dos benefícios da VPN é utilizar um serviço com o cliente de código aberto, como o OpenVPN (publicado pela General Public License – GPL), bastando a rede suportar o protocolo em questão para ter condições de criar e gerenciar suas conexões.” (DELFINO, 2018)

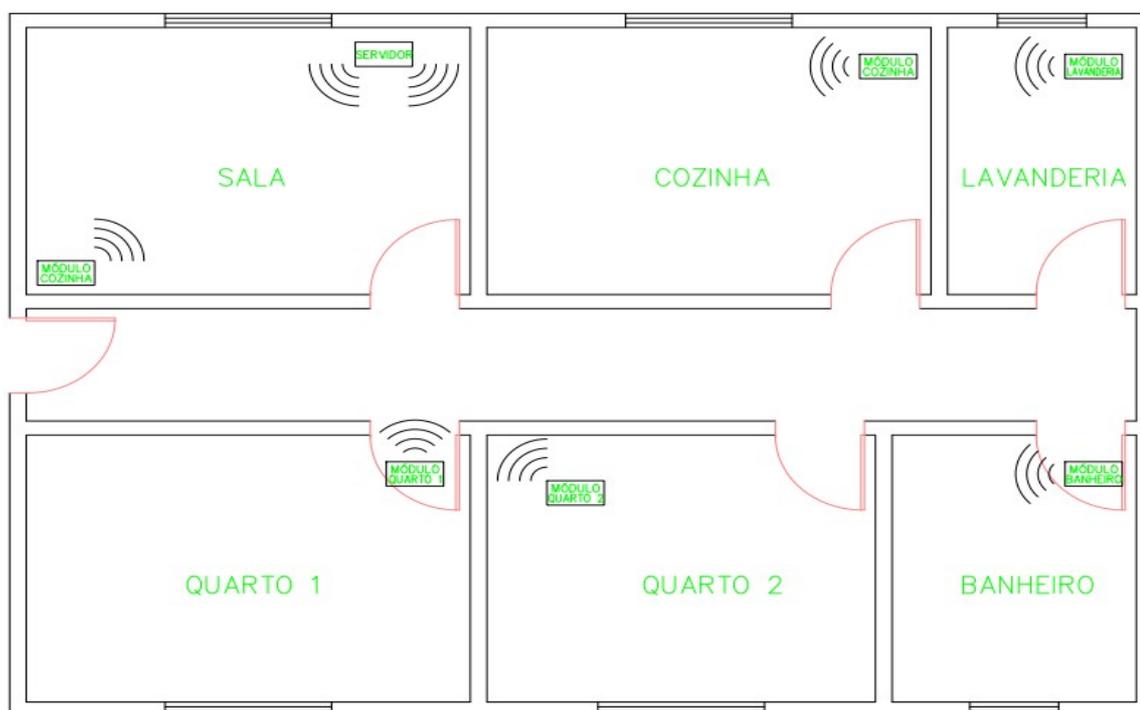
3 ESTRUTURA DA SIMULAÇÃO EM MAQUETE

Nesta seção será abordado o trabalho prático, o qual foi feito por meio da simulação em maquete de: lâmpada, tomada e janela. Serão apresentados, através de imagens e descrições, os elementos de *hardware* e *software* que compõe o trabalho, dentre eles estão as interfaces do usuário, dispositivos eletrônicos, protótipo e segurança da informação.

3.1 Ambiente Físico

Em um ambiente real os elementos, como janelas, lâmpadas e tomadas, estão conectados a um módulo por meio de cabos elétricos, como representado na Figura 1. Este módulo é composto de um circuito eletrônico ligado a um Raspberry pi e fica instalado nos cômodos da residência. Pode ser instalado um ou mais módulos no mesmo cômodo da residência, dependendo da quantidade de elemento a ser controlado, sendo que é necessário ter pelo menos um módulo no cômodo para que o servidor reconheça o local. Nos módulos estão instalados dispositivos de rádio frequência que se comunicam com o servidor. Estes dispositivos de rádio frequência recebem e enviam informações ao servidor. O servidor, que comporta a página *web*, envia o comando dado pelo usuário ao módulo, o qual através do dispositivo de rádio frequência, recebe o estado dos elementos, processa a informação e mostra o resultado ao usuário.

Figura 1 – Planta baixa de um exemplo do ambiente real



Fonte: Autoria própria (2021)

3.2 Protótipo em Maquete

A maquete foi construída em MDF fino cortado a laser, a qual está mostrada na Figura 2, que mostra a vista externa do protótipo. No protótipo foi instalada uma lâmpada e uma tomada real, com tensão elétrica de 127 VCA, simulando uma residência comum. O motor de acionamento da janela é de 12 VCC, podendo ser deslocado para trás, simulando um descarrilhamento da cremalheira, gerando o alarme de falha. Estes elementos podem ser vistos na Figura 3. Abaixo do protótipo da casa há uma gaveta onde está o *hardware*, para melhor organização dos elementos físicos. Assim somente levantando a tampa, é possível acessar o *hardware*.

Figura 2 – Vista externa do protótipo



Fonte: Autoria própria (2021)

Figura 3 – Vista interna do protótipo



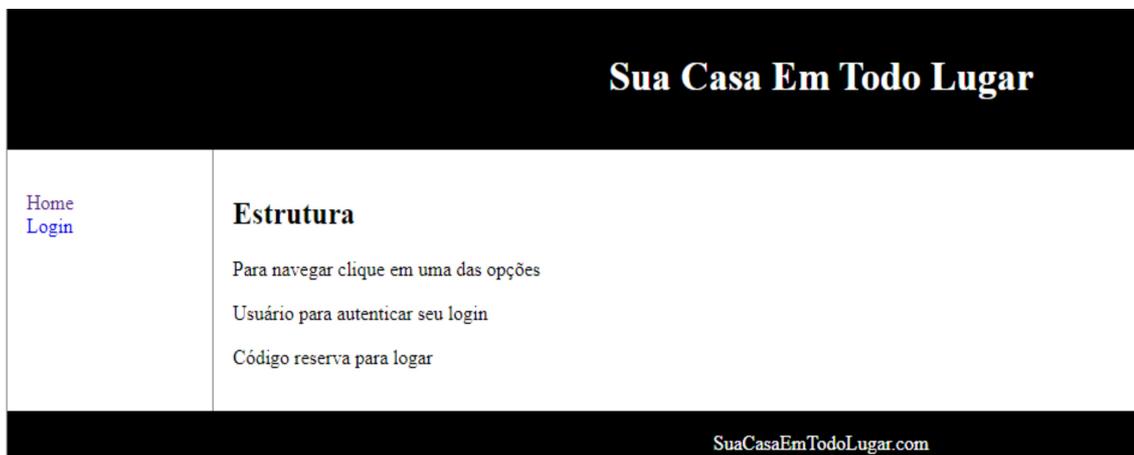
Fonte: Autoria própria (2021)

3.3 Ambiente do Usuário

Na tela inicial é possível acessar o ícone de login e senha, conforme mostrado na Figura 4. O proprietário irá realizar o cadastro dos usuários que serão autorizados no sistema. Os dados são armazenados em um banco de dados. Todos os usuários do serviço terão, obrigatoriamente, um login e senha para o acesso. Sem isso o usuário não conseguirá acessar os serviços. A tela de autenticação permite ao usuário inserir o usuário e a senha cadastrados previamente, com isso terá maior segurança em relação á sua residência no que diz respeito a sua automação.

O ambiente está alocado no servidor, que estará instalado dentro do próprio protótipo. Foi desenvolvido em linguagem HTML e CSS, e estas estão em conexão com o SQLite.

Figura 4 – Ambiente do usuário



Fonte: Autoria própria (2021)

3.4 Estado dos Elementos

Nesta tela, mostrada na Figura 5, ao acessar, o usuário pode acender a lâmpada desejada, poderá controlar a energização de uma tomada de algum eletrodoméstico ou eletroeletrônico que esteja conectado, abrir ou fechar as janelas de um determinado cômodo. Para executar as ações o usuário deve clicar e marcar o campo do elemento desejado, após isso o usuário confirma a ação no botão enviar. O servidor envia o sinal através do dispositivo de RF para o módulo, este por sua vez processa o sinal recebido do servidor e executa a ação, devolvendo o resultado para o servidor por sinais de RF. Esta tela está alocada dentro do servidor e, é desenvolvida em HTML, CSS e Python.

Figura 5 – Estado dos elementos

The screenshot shows a web application interface. At the top, there is a black header with the text "Sua Casa Em Todo Lugar" in white. Below the header, on the left, there is a navigation menu with links: "Home", "Login", and "Painel De Controle". The main content area is titled "Estrutura" and contains a heading "#Editar essa parte para sair". Below the heading is a table with four columns: "Local", "Dispositivos", "Estado", and "Alterar". The table contains three rows of data. Below the table, there are four buttons: "Limpar", "Carregar Dados", "Atualizar", and "Enviar". At the bottom of the page, there is a black footer with the text "SuaCasaEmTodoLugar.com".

Local	Dispositivos	Estado	Alterar
Sala	Lampada	Inativa	<input type="checkbox"/>
Sala	Tomada	Inativa	<input type="checkbox"/>
Sala	Janela	Aberto	<input type="checkbox"/>

Fonte: Autoria própria (2021)

3.5 Servidor

Foi utilizado um Raspberry pi 3 para o servidor do trabalho. Neste servidor está instalada a VPN, que será explicada mais detalhadamente nos próximos tópicos, conectando a um dispositivo móvel, dando mais segurança aos dados do usuário. No servidor está alocada a página *web*, que é a interface visível ao usuário para o acesso das funções que deseja executar. O banco de dados foi instalado para que seja armazenado o estado dos elementos e a sequência de bits a ser transmitida e recebida do módulo através do dispositivo de RF, também instalado no servidor. A linguagem de *backend* utilizada foi o Python, com o *framework Bottle*, tornando mais simples a programação e a execução das atividades. Houve a necessidade da instalação de dissipadores de calor pois a temperatura eleva-se muito durante a execução das atividades, podendo danificar o equipamento

3.6 Módulo Físico

O módulo físico é composto de um controlador lógico, um módulo auxiliar e dois transmissores de RF. O controlador lógico recebe os sinais do receptor RF, vindo do transmissor RF do servidor, interpreta a função a ser realizada e por meio das portas de IO envia um sinal lógico para o módulo auxiliar. Por sua vez o módulo auxiliar transforma o sinal lógico em sinais elétricos através dos transistores e relés. Os relés funcionam como interruptores automáticos, com isso eles convertem baixa

tensão em tensões mais altas, como 12 VCC, 24 VCC e 127 VCA. No caso particular da janela, há dois sensores de fim de curso, instalado no protótipo, que tem a função de monitorar o estado da janela, se está aberta ou fechada, e enviar esse estado diretamente ao controlador lógico. O controlador lógico por sua vez vai enviar a resposta das ações ao servidor através do transmissor RF, para que o servidor receba esse sinal pelo receptor e execute as ações necessárias.

O módulo auxiliar se comporta como um intermediário entre o controlador lógico e os elementos a serem controlados. A função principal do módulo auxiliar é transformar os sinais lógicos em sinais elétricos. O controlador lógico não tem a capacidade de operar com tensões elétricas mais elevadas, com isso foi necessário desenvolver um módulo auxiliar. Este controlador lógico não consegue enviar tensões maiores que 5 VCC em suas portas IO. Já nas portas de 5 VCC, se estiver em operação com algum elemento, travam o sistema operacional do Raspberry pi. O controlador lógico tem outro agravante que é o de não conseguir fornecer mais do que 20mA de corrente elétrica em suas portas, com isso alguns elementos não funcionariam corretamente.

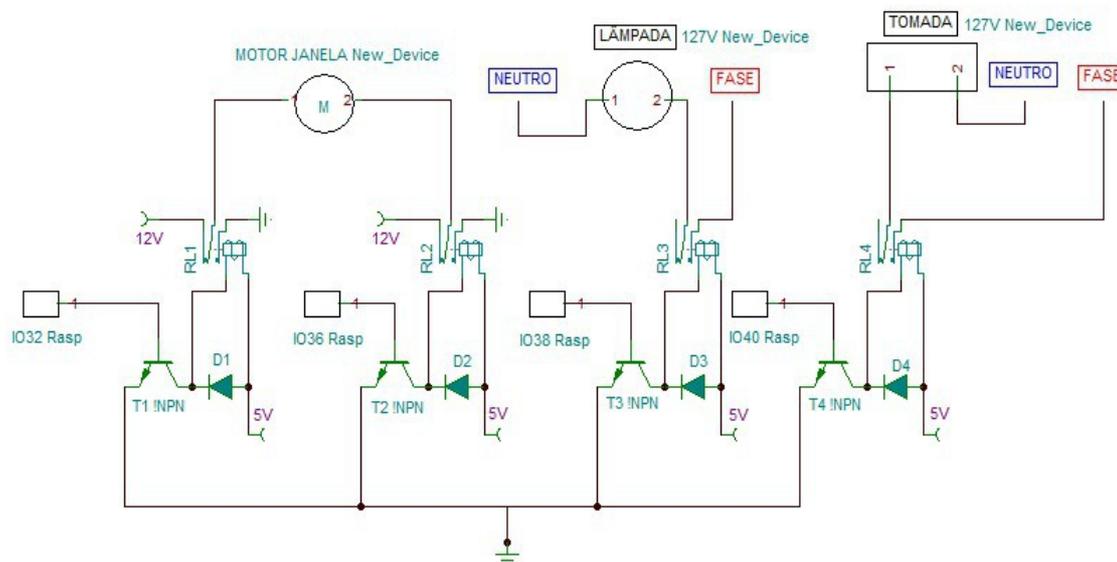
O controlador lógico envia aproximadamente 2,3 VCC em suas saídas, com isso, para transformar em tensões de trabalho, foram instalados transistores com a função de chave automática, para fornecer 12 VCC quando solicitado. Como os transistores não suportam uma tensão maior que 24 VCC foi necessário introduzir relés nesse circuito. Essa tensão de saída nos transistores acionam os relés cuja a bobina é da mesma tensão, e por seus terminais é possível passar até 250 VCA, que são para o funcionamento da tomada e lâmpada. No caso da janela são necessários dois relés, um para abrir e outro para fechar, com tensão de 12 VCC. As tensões de saída dos elementos estão apresentadas no Quadro 1. O esquema elétrico do módulo auxiliar está mostrado na Figura 6:

Quadro 1 – Tensões de funcionamento dos elementos

Janela	12 VCC
Tomada	127 VCA
Lâmpada	127 VCA

Fonte: Autoria Própria (2021)

Figura 6 – Esquema elétrico do módulo auxiliar



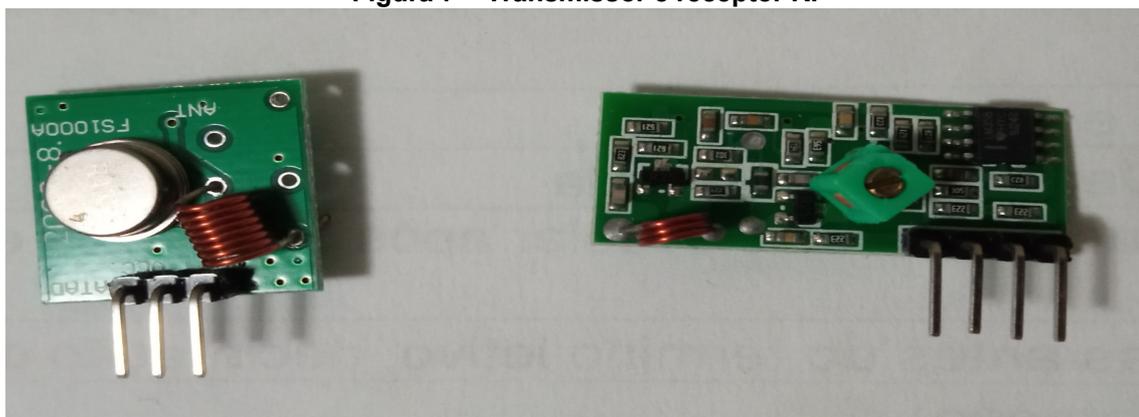
Fonte: Autoria própria (2021)

3.7 Transmissão dos Dados

A transmissão dos dados entre o servidor e o módulo é realizada entre os dispositivos de rádio frequência (RF). Há no servidor um transmissor e um receptor de RF. Estes dispositivos são módulos de RF, que trabalham com frequência de 433 MHz, sendo que o transmissor opera com tensões entre 3,3VCC e 12VCC, já o receptor opera com tensão fixa de 5VCC.

No módulo também há um transmissor e um receptor de RF. O servidor envia informações para o módulo como, abrir ou fechar a janela, apagar ou acender a lâmpada ou ativar e desativar a tomada, através do transmissor RF que, envia uma sequência numérica ao receptor do módulo que indica qual é o elemento a ser acionado. Ao receber essa sequência numérica, o módulo identifica qual é a ação e o elemento que deve ser acionado, busca o estado atual e executa a ação requerida. Após atuar, o módulo através do transmissor de RF, envia a sequência numérica que indica a conclusão da ação e o estado atual do elemento e, essa informação é recebida pelo receptor de RF do servidor que, indica ao usuário qual é o estado do elemento. Os transmissores de RF trabalham com uma tensão de 12VCC, vindo de uma fonte externa, e estão ligados nas portas GPIO 11 (GND), 13 do servidor e 15 do módulo. Os receptores de RF trabalham com tensão de 5VCC, vindo de uma fonte externa, e estão ligados nas portas GPIO 11(GND), 15 do servidor e 13 do módulo. O receptor e transmissor estão mostrados na Figura 7:

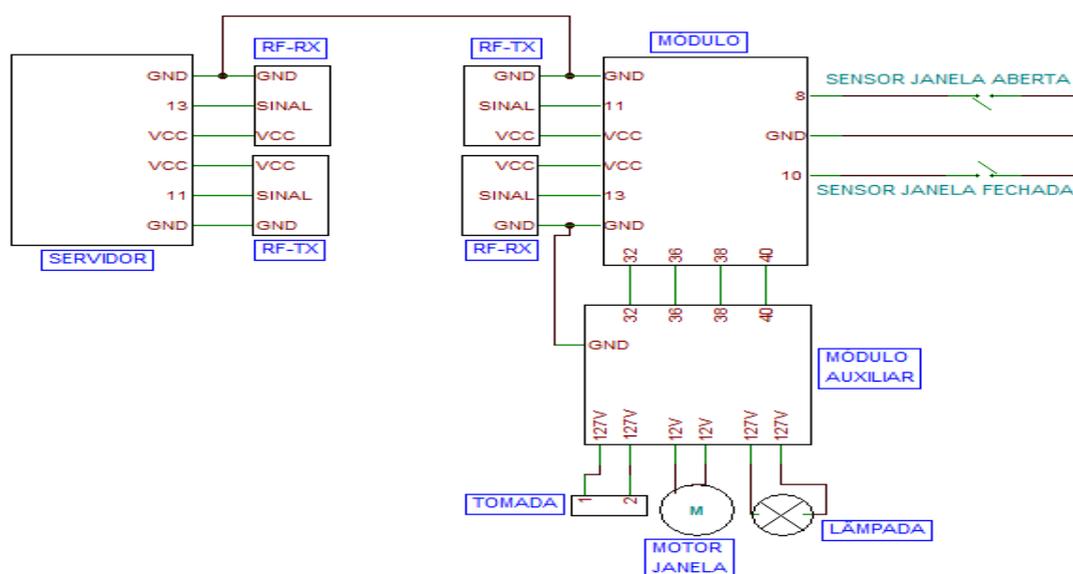
Figura 7 – Transmissor e receptor RF



Fonte: Autoria própria (2021)

Na Figura 8 é apresentada uma visão geral do sistema.

Figura 8 – Diagrama em blocos geral



Fonte: Autoria própria (2021)

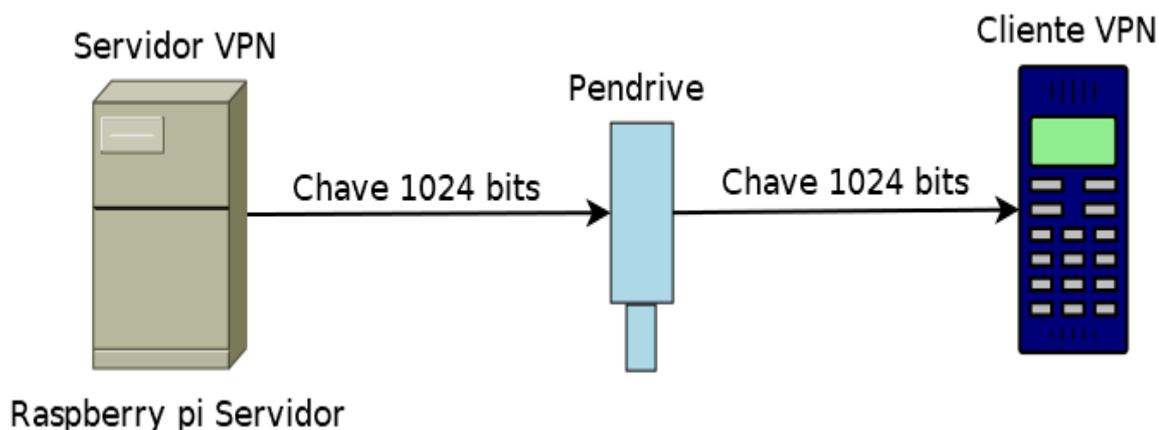
3.8 Segurança dos Dados

Neste trabalho foi instalada uma VPN para o acesso direto e seguro do dispositivo do usuário ao serviço, esse tunelamento dificulta a ação invasora de pessoas mal intencionadas, porque esse túnel é criptografado. Para essa aplicação foi utilizado o OpenVPN por ser mais simples de configurar e eficaz para a aplicação, sendo que o servidor foi instalado no Raspberry pi e o cliente foi instalado no *smartphone*. Houve a necessidade de configuração do modem da Huawei pois o mesmo não estava preparado para aceitar o serviço de VPN, mas não houve a

necessidade de liberação de portas pela operadora Oi. Foi utilizada uma chave de 1024 bits, que é suficiente para a aplicação. Para uma maior segurança a chave foi armazenada em um pendrive e descarregada diretamente no *smartphone*, com isso reduziu-se a possibilidade de interceptação e furto dessa chave se, por exemplo, comparado a um envio por e-mail. O DNS utilizado foi o da Google pois, possivelmente, é mais rápido.

Uma outra camada de proteção de dados foi a configuração de um iptables, dentro de um *firewall*, cujo o objetivo é de ocultar o IP do servidor no caso de um atacante realizar uma varredura, através do protocolo ICMP, na internet com o objetivo de descobrir alguma vulnerabilidade no servidor. Nesse caso o atacante não obtém resposta do IP configurado no servidor. Na Figura 9 é mostrado o transporte da chave de criptografia.

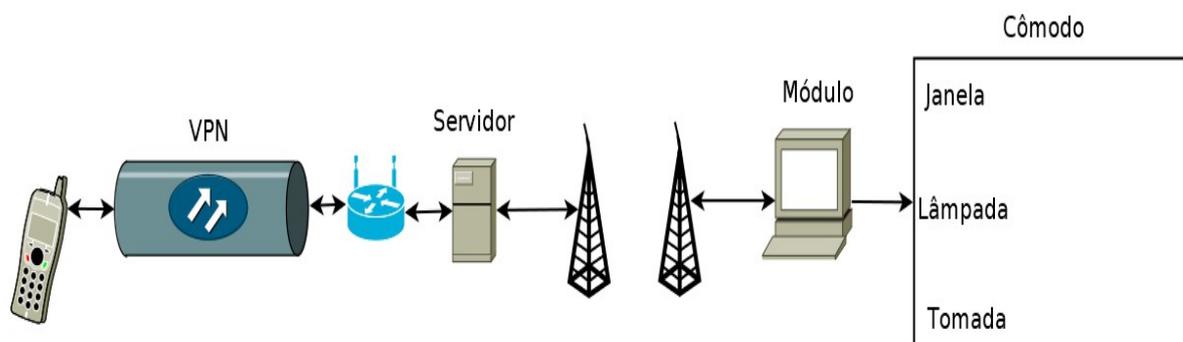
Figura 9 – Transporte da chave de criptografia



Fonte: Autoria própria (2021)

Na Figura 10 a topologia da aplicação.

Figura 10 - Topologia da aplicação



Fonte: Autoria própria (2021)

3.9 SGBD

O SQLite foi o sistema utilizado para o trabalho, pois ele é mais leve e atende a necessidade atual. O SQLite armazena os dados de usuário e senha, bem como os códigos de transmissão e recepção dos elementos que serão enviados pelos dispositivos de RF. A partir do momento que o usuário executa uma ação, o sistema, no servidor, busca o código adequado para a requisição e envia pelo transmissor ao módulo. A partir do momento que esse código chega ao módulo o sistema do mesmo busca o que lhe foi pedido, substitui a condição anterior e executa a ação. Após a execução da ação, o módulo busca o código de confirmação da ação no banco de dados e envia ao servidor através do transmissor de RF. O servidor recebe esse código, substitui o estado anterior, armazena no banco de dados o estado atual e demonstra ao usuário o resultado da ação solicitada. Toda essa programação de *insert* e *delete* no banco de dados está dentro do *script* principal do servidor e do módulo.

4 APRESENTAÇÃO E ANÁLISE DOS RESULTADOS

A seção de apresentação e análise de resultados é sub dividida em duas seções: (1) a avaliação dos resultados que mostra, de forma estatística, por comparação ou observação, os dados obtidos em todos os elementos e matérias envolvidas no trabalho e, (2) as dificuldades encontradas no trabalho que trata das dificuldades que foram enfrentadas ao longo da execução do trabalho.

4.1 Avaliação dos Resultados

O Raspberry pi 3, como o *hardware* principal do trabalho, teve um desempenho muito bom e sem falhas. Sendo executadas as mesmas atividades, o Raspberry pi 3 atingiu o máximo de 60% de processamento, enquanto o Raspberry pi 2 atingiu 100% de processamento, travando o sistema, sendo necessária, em algumas vezes, reiniciar o sistema. Um diferencial encontrado foi o dispositivo wireless integrado no *hardware* do Raspberry pi 3, pois no Raspberry pi 2 é necessário um adaptador wireless usb, e foi notado que esse adaptador influenciou no desempenho de velocidade da internet, pois o Raspberry pi 3 acessou a página *web* mais rapidamente do que o Raspberry pi 2.

Os receptores de rádio frequência conseguiram captar os sinais em um raio de até 6 metros de distância, sem sofrer interferência de outros dispositivos com a mesma tecnologia. Em contrapartida os transmissores não tiveram um bom desempenho, houve a necessidade de deixá-los com 5 mm de distância do receptor, pois a corrente de trabalho que o Raspberry pi 3 ofereceu não atingiu os 0,26A, que é o mínimo necessário para um bom funcionamento dos transmissores, que, em condições adequadas, podem enviar sinais até 10 metros de distância dos receptores.

O Python como a linguagem de baixo nível, apesar de sua simplicidade, atendeu as expectativas e cumpriu rigorosamente com o que lhe foi proposto, que é de executar as ações ordenadas pelo usuário. O *framework* de Python usado foi o *Bottle*, que cumpriu o que foi proposto. A outra opção de *framework*, o Jango, é mais completo, e poderá ser usado em trabalhos futuros.

O HTML teve um papel importante no layout da página, e em conjunto com Python trabalharam sem nenhuma intercorrência. O SGBD usado, que foi o SQLite, teve um desempenho razoável, sendo que cumpriu o que foi proposto. O *hardware*

secundário foi projetado e desenvolvido pelos membros da equipe, e executou muito bem o que foi submetido.

Na parte de segurança da informação foi implementada uma VPN que, foi facilmente configurada. Por algumas vezes foi necessário reconfigurá-la pois, como o ip residencial é do tipo dinâmico, por várias vezes foi alterado pela operadora, com isso houve a necessidade de reconfiguração. No *firewall* foi configurado um iptables, que não permitiu ao atacante o acesso do IP no servidor. Sem as regras de iptables o atacante descobriu com facilidade o ip da servidor, mas a partir do momento que as regras foram implementadas o servidor parou de responder as requisições. Com isso fica inviável o invasor tentar atacar um sistema que não responde as solicitações. Há um resumo da avaliação dos resultados no Quadro 2:

Quadro 2 – Resumo da avaliação dos resultados

Raspberry pi	Adequado o modelo 3
Receptores	Shield, 433MHZ
Python	Versão 3
HTML	Versão 5
Segurança de dados	VPN, IPTables, autenticação

Fonte: Aatoria própria (2021)

4.2 Dificuldades Encontradas no Trabalho

Durante a realização do trabalho, foi realizado de forma constante o monitoramento do processamento dos dispositivos, pois havia muita atividade sendo executada de forma simultânea. No início houve muitas falhas por essa razão. Por exemplo, no Raspberry pi 2 havia uma saturação do processamento. Para resolver trocamos o Raspberry pi 2 para o Raspberry pi 3, pois este tem uma capacidade e desempenho melhor. O Raspberry pi 2 tem um processador ARM cortex-A7, enquanto no Raspberry pi 3 o processador teve uma evolução para o ARM cotex-53, tendo as mesmas atividades ocupando, no máximo 40% de processamento. Em contrapartida o Raspberry pi 3 teve um aquecimento superior ao Raspberry pi 2, enquanto no Raspberry pi 2 o aquecimento chegou a 40°, no Raspberry pi 3 a temperatura chegou a atingir 58°, ambas medições feitas no auge do processamento. Nessa condição de temperatura um dos Raspberry pi 3 acabou estragando, sendo necessária a aquisição de outro dispositivo. Para resolver foram sendo instalados dissipadores de calor no processador, diminuindo os efeitos da

temperatura no Raspberry pi 3. De forma resumida, as informações do Raspberry pi estão descritas no Quadro 3:

Quadro 3 – Comparativo entre os Raspberry Pi

	Processador	Memória RAM	Tensão	Corrente	Processamento máximo	Temperatura máxima
Raspberry-pi 2	Quad-core 900MHz	1 Gb	5 VCC	2A	100,00%	58°
Raspberry-pi 3	Quad-core 1,2GHz	1 Gb	5,1 VCC	2,5A	40,00%	40°

Fonte: Autoria própria (2021)

No campo de energia elétrica foi utilizada uma fonte externa para a alimentação dos vários dispositivos, apesar do Raspberry pi 3 ter saídas de tensão de 5 VCC e 3,3 VCC. Ao executar as atividades, o Raspberry pi 3 deixava mais lenta a execução das atividades, e em alguns momentos reiniciava involuntariamente o sistema operacional, adicionalmente, foi notado o sobreaquecimento da placa. A fonte de alimentação utilizada para os dispositivos foi de 2.5 A, pois correntes menores, principalmente no Raspberry pi 3, com as atividades executadas, desarmavam o dispositivo por falta de energia. Foram testadas fontes de 1, 1,5 e 2 A, mas em algum momento haviam quedas de energia e, após medições e testes constatamos essa falha de energia. Portanto, descartadas do projeto, sendo utilizada a fonte de 2,5A.

Sobre a comunicação por rádio frequência foram realizados testes durante 24 horas em busca de sinais que poderiam interferir no funcionamento do trabalho, mas nenhum sinal captado foi capaz de interferir na aplicação. Caso apareça um código igual ao que for transmitido haverá interferência no sistema, podendo acionar algum elemento. O transmissor e o receptor tiveram que ficar muito próximos um do outro porque o Raspberry pi, nas suas portas GPIO, fornecem no máximo 0,20 A de corrente e, o transmissor funciona adequadamente com corrente a partir de 0,26 A no mínimo.

Houve a necessidade de projetar um circuito eletrônico pois, o Raspberry pi não foi capaz de fornecer tensão necessária para o acionamento dos dispositivos, e para um bom funcionamento dos elementos, principalmente os transmissores por rádio frequência, que funcionam satisfatoriamente com tensão de 12 VCC.

O SQLite é um SGBD (sistema gerenciador de banco de dados) simplificado e, por ser simplificado gera um único arquivo facilitando o backup e otimizando a busca de dados.

5 CONSIDERAÇÕES FINAIS

O trabalho de domótica (automação residencial) foi importante na formação em sistemas de telecomunicações, pois houve uma integração de várias disciplinas que tivemos durante o curso com este trabalho. As matérias envolvidas foram essenciais para o sucesso da empreitada.

A parte de segurança da informação teve uma participação fundamental na elaboração do trabalho e uma relevância satisfatória, pois foi elaborada e testada, gerando resultados positivos, dentre os testes o de invasão foi realizado e o sistema foi bem sucedido. Em complemento na segurança da informação, a VPN também foi importante para garantia desta segurança.

Na construção do sistema tivemos que aprender linguagens não abordadas no curso, como: HTML, CSS e Python. As linguagens utilizadas serviram para garantir uma interação entre usuário e sistema.

No que diz respeito ao *hardware* dessa implantação, o escolhido foi o Raspberry Pi pois, ele é um mini computador, mas muito compacto, melhorando a questão do espaço físico, principalmente se comparado a um computador de mesa, ou mesmo ao *notebook*, com a vantagem de se ter saídas IO de fácil acesso, que são essenciais para a montagem dos elementos da residência. Ainda para o *hardware*, uma parte foi adquirida e na outra parte surgiu por meio de um desenvolvimento realizado pela equipe. Houve um desempenho adequado, mas para sistemas mais detalhados cabe continuar buscando aperfeiçoamento, e uma possibilidade de aprimoramento que poderia ser a utilização de um microcontrolador PIC que é menos sujeito a interferências elétricas. Para implementação futura o sistema deveria ter uma alternativa de acionamento manual dos elementos, visando a independência destes, não gerando a dependência de um *smartphone* para o acionamento dos elementos dentro do próprio ambiente.

Para futuros trabalhos, a sugestão seria a incorporação com a Amazon Alexa. Este equipamento trabalha com comando de voz para o controle interno da residência, podendo representar uma inovação para a automação residencial.

Na questão de comunicação entre servidor e módulos foi utilizado o sistema de RF, pois tem um baixo consumo de energia e um longo alcance, podendo ter uma boa comunicação entre os cômodos da residência. Em outros casos podem ser implementados os sistemas de comunicação cabeado, *bluetooth*, *zig bee* ou

wireless, para cômodos distantes, ou onde não há uma visada direta do módulo ao servidor. Esta gama de aplicações pode gerar melhorias significativas na comunicação entre servidor e módulo.

A plataforma de programação em Python também oferece várias opções, como o Django, sendo esse um *framework* completo, principalmente em banco de dados, pois é possível utilizar o MySQL nas aplicações. Neste caso o Python foi utilizado por ser a linguagem padrão do Raspberry Pi, não sendo necessário instalar mais nada para utilizar. Outro motivo de se utilizar Python é a simples programação, sendo que sua manutenção futura fica mais facilitada e, por deixar o processamento do sistema mais leve. O *framework* escolhido foi o *bottle* pois ele se mostrou mais leve no sistema, pouco sobrecarregando o sistema. Para o banco de dados foi escolhido o SQLite, pois apesar de ser mais simples, atendeu todas as necessidades do sistema, não sobrecarregando em demasia a aplicação.

Em se tratando de programação, pode ser utilizado o Java, que é uma linguagem de programação robusta e disponibiliza várias opções, até mesmo no layout da página e aplicativos para dispositivos móveis.

Diante de todos os testes e implementações chegou-se a um protótipo funcional que atendeu aos objetivos propostos.

REFERÊNCIAS

ADMINWEBBIZZ. **Automação Residencial – o futuro já chegou.** Disponível em: <<https://innovhome.com.br/automacao-residencial-o-futuro-ja-chegou/>>. Acesso em: 02 jan. 2021.

ANTONIO, Adriano Martins. **As principais funcionalidades do Nmap.** Disponível em: <<https://www.pmgacademy.com/funcionalidades-do-nmap/>>. Acesso em: 18 out. 2020.

ARAÚJO, Thayron. **Raspberry pi B+: Introdução à porta GPIO.** Disponível em: <<https://blog.fazedores.com/raspberry-pi-b-introducao-porta-gpio/>>. Acesso em: 22 jul. 2019.

AZURE. **O que é uma máquina virtual?** Disponível em: <<https://azure.microsoft.com/pt-br/overview/what-is-a-virtual-machine/>>. Acesso em: 18 out. 2020.

BRANQUINHO, Marcelo Ayres; SEIDL, Jan; MORAES, Leonardo Cardoso de; BRANQUINHO, Thiago Braga; JUNIOR, Jarcy de Azevedo. **Segurança de Automação Industrial e SCADA.** 1.ed. Rio de Janeiro: Elsevier, 2014.

BRASIL ESCOLA. **Estações do ano.** Disponível em: <<https://brasilecola.uol.com.br/geografia/estacoes-ano.htm> >. Acesso em: 21 nov. 2019.

BRITO, Edivaldo. **Coisas que você deve saber antes de usar o Kali linux.** Disponível em: <<https://www.edivaldobrito.com.br/coisas-que-voce-deve-saber-antes-de-usar-o-kali-linux>>. Acesso em: 18 out. 2020.

CARAPEÇOS, Nathália. **Saiba como amenizar os problemas causados pela umidade em casa.** Disponível em: <<https://gauchazh.clicrbs.com.br/comportamento/noticia/2015/06/saiba-como-amenizar-os-problemas-causados-pela-umidade-em-casa-4787528.html>> Acesso em: 21 nov. 2019.

CORDELLA, Prof.. **Domótica.** Disponível em: <http://www.profcordella.com.br/unisanta/textos/fqa16_domotica_edificios_inteligentes.htm> Acesso em: 02 jan. 2021.

DJANGOGIRLS. **O que é Django?** Disponível em: <<https://tutorial.djangogirls.org/pt/django/>>. Acesso em: 18 out. 2020.

ESTRELLA, Carlos. **SQLite vs MySQL – Qual a Diferença e Qual Usar.** Disponível em: <<https://www.hostinger.com.br/tutoriais/SQLite-vs-mysql/#Pros-e-Contras-SQLite-vs-MySQL>> Acesso em: 28 nov 2019.

COSTA, Ricardo Ferreira. **Como proteger seu servidor linux contra ataques automatizados usando a ferramenta fail2ban.** Disponível em: <<https://br->

linux.org/2016/01/como-proteger-seu-servidor-linux-contrataques-automatizados-usando-a-ferramenta-fail2ban.html >. Acesso em: 30 ago. 2019.

DELFINO, Pedro. **OpenVPN: Conheça Uma Das Soluções VPN Mais Utilizadas Do Mundo**. Disponível em: <<https://e-tinet.com/linux/openvpn/>>. Acesso em: 08 nov. 2021.

EIS, Diego. **O básico: O que é HTML?**. Disponível em: <<https://tableless.com.br/o-que-html-basico/>>. Acesso em: 01 set. 2019.

FAZANO JR., Pedro Vicente Prata. **Projeto Domótico Para Ambientes Inteligentes Baseado Nas Tecnologias Arduino E Google Android**. Fundação Educacional do Município de Assis – FEMA – Assis, São Paulo, 2013. Disponível em: <https://cepein.femanet.com.br/BDigital/arqTccs/1011330638.pdf>. Acesso em: 01 ago 2020.

GOULART, Angelito. **Acesso remoto via RDP no Raspberry pi**. Disponível em: <<http://angelitomg.com/blog/acesso-remoto-via-rdp-no-raspberry-pi/>>. Acesso em: 04 set. 2019.

HINDZBERGEN, Jule; HINDZBERGEN, Kees; SMULDERS, André; BAARS, Hans. **Fundamentos da segurança da informação: Com base na Iso 27001 e na Iso 27002**. Tradução: Alan de Sá. 3.ed. Rio de Janeiro: Brasport, 2018.

JULIO, Karina Balan; RIBEIRO, Igor; ROGENSKI, Renato; FARIAS, Taís. **Imobiliário: Convivência digital**. Disponível em: <<http://nextnow.meioemensagem.com.br/imobiliario-convivencia-digital/>>. Acesso em: 11 set. 2019.

JUNIOR, Valdir. **Programando para web em Phyton**. Disponível em: <<https://pythonhelp.wordpress.com/2013/03/29/programando-para-a-web-em-Python/>>. Acesso em: 10 ago. 2019.

KALI. **Sobre o kali linux**. Disponível em: <<https://www.kali.org/about-us/>>. Acesso em: 18 out. 2020.

KASPERSKY. **O que são ataques de DDos**. Disponível em: <<https://www.kaspersky.com.br/resource-center/threats/ddos-attacks>>. Acesso em: 18 out. 2020.

L., Andrei. **O que é HTML: Guia básico para iniciantes**. Disponível em: <<https://www.hostinger.com.br/tutoriais/o-que-e-html-conceitos-basicos/>>. Acesso em: 28 nov 2019.

MATTEDE, Henrique. **Domótica - O que é e quais as vantagens!**. Disponível em: <<https://www.mundodaeletrica.com.br/domotica-o-que-e-quais-as-vantagens/>>. Acesso em: 02 jan 2021.

MELO, Sandro. **Exploração de vulnerabilidades em redes TCP/IP**. 3.ed. Rio de Janeiro: Alta books, 2017.

MURATORI, José Roberto. **O que pode ser feito para dinamizar o mercado de Automação Residencial no Brasil?** Disponível em: <<http://aureside.blogspot.com/2018/12/o-que-pode-ser-feito-para-dinamizar-o.html>>. Acesso em: 28 nov 2020.

OELINTON, Eder. **Como funciona a transmissão de dados via rádio?** Disponível em: <<https://suprimatec.com/como-funciona-a-transmissao-de-dados-via-radio/>>. Acesso em: 01 ago. 2019.

ORACLE **O Que É um Banco de Dados?** Disponível em: <<https://www.oracle.com/br/database/what-is-database.html>>. Acesso em: 04 out. 2020.

PORTAL GSTI. **O que é segurança da informação?** Disponível em: <<https://www.portalgsti.com.br/seguranca-da-informacao/sobre/>>. Acesso em: 18 out. 2020.

PYSCIENCE. **Python: O que é? Por que usar?** Disponível em: <<http://pyscience-brasil.wikidot.com/Python:Python-oq-e-pq>>. Acesso em: 28 nov. 2019.

MARQUES, Rafael. **O que é HTML? Entenda de forma descomplicada.** Disponível em: <<https://www.homehost.com.br/blog/tutoriais/o-que-e-html/>>. Acesso em: 28 nov 2019.

REIS, Fabio dos. **firewall iptables no Linux.** Disponível em: <<http://www.bosontreinamentos.com.br/linux/firewall-iptables-no-linux-parte-01-nocoas-basicas>>. Acesso em: 18 out. 2020.

ROOTS, Pablo T.. **Routers: Cinco formas de fechar a porta aos criminosos.** Disponível em: <<http://suporteninja.com/routers-cinco-formas-de-fechar-a-porta-aos-criminosos/>>. Acesso em: 30 ago. 2019.

SAB, Gabriel Augusto Amim; FERREIRA, Rafael Cardoso; ROZENDO, Rafael Gonsalves. **Negação de serviço, negação de serviço distribuída e botnets.** Disponível em: <https://www.gta.ufrj.br/grad/13_1/dos/ataques.html>. Acesso em: 18 out. 2020.

SANTOS, Alex. **Ferramentas para bloqueio de ataques DDOS.** Disponível em: <<https://www.vivaolinux.com.br/topico/Servidores-Linux-para-iniciantes/Ferramentas-para-bloqueio-de-ataques-DDOS>>. Acesso em: 30 ago. 2019.

SOUZA, Lindeberg Barros de. **Gerenciamento e segurança de redes.** 1.ed. São Paulo: SENAI-SP, 2011.

SPADRI, Ana. **O protocolo ICMP.** Disponível em: <<https://br.ccm.net/contents/267-o-protocolo-icmp>>. Acesso em: 18 out. 2020.

STRAUB, Matheus Gebert. **Comunicação sem fio arduino transferindo dados com transmissor receptor rf 433mhz RF 433MHZ STX882RF.** Disponível em:

<<https://www.usinainfo.com.br/blog/comunicacao-sem-fio-arduino-transferindo-dados-com-transmissor-receptor-rf-433mhz/>> Acesso em: 28 nov 2018.

THOMSEN, Adilson. **Review Raspberry pi 3: Tudo que você precisa saber**

. Disponível em: <<https://www.filipeflop.com/blog/raspberry-pi-3-review/>>. Acesso em: 16 out. 2021

VICTOR, Felipe. **Raspberry pi e Arduino: Qual é a diferença e qual é o melhor para o seu projeto.** Disponível em: <<https://portaldosaber.net/2018/02/raspberry-pi-e-arduino-qual-e-diferenca-e-qual-e-o-melhor-para-o-seu-projeto/>> Acesso em: 29 nov 2019.

VIEIRA, Vinicius. **Como proteger sua privacidade no Linux.** Disponível em: <<https://sejalivre.org/como-proteger-sua-privacidade-no-linux/>>. Acesso em: 01 set. 2019.

WIKIHOW. **Como criar um servidor web Raspberry pi.** Disponível em: <<https://pt.wikihow.com/Criar-um-Servidor-web-Raspberry-Pi>>. Acesso em: 27 mai. 2019.

APÊNDICE A – Parte de Código Fonte do Servidor

```
if("miguel" == CookLogin or "roberto" == CookLogin):
```

```
with
```

```
sqlite3.connect('/home/pi/Desktop/SERVIDOR/BD_SQLITE/BD_Estado_Atual.db') as  
conn:
```

```
    cursor = conn.cursor()
```

```
    cursor.execute("""SELECT Janela_Atual FROM Tabela_1""")
```

```
    for Janela_Atual in cursor.fetchone():
```

```
        BD_Atual_HTML_Janela = Janela_Atual
```

```
    cursor.execute("""SELECT Tomada_Atual FROM Tabela_1""")
```

```
    for Tomada_Atual in cursor.fetchone():
```

```
        BD_Atual_HTML_Tomada = Tomada_Atual
```

```
    cursor.execute("""SELECT lampada_Atual FROM Tabela_1""")
```

```
    for Lampada_Atual in cursor.fetchone():
```

```
        BD_Atual_HTML_Lampada = Lampada_Atual
```

```
if(ID_Usuario_Logado=='miguel' or ID_Usuario_Logado == 'roberto'):
```

```
    if (ID_Usuario_Logado=='miguel'):
```

```
        ID_Usuario = '567'
```

```
    elif (ID_Usuario_Logado=='roberto'):
```

```
        ID_Usuario = '765'
```

```
    print(ID_Usuario_Logado)
```

```
if(Boolean_HTML_Estado_Enviar==True):
```

```
    print(str(HTML_Estado_Enviar) + "Encontrado enviar")
```

```
print(ID_Verificador)
```

```
print (ID_Dispostivo)
```

```
print(ID_Usuario)
```

```
#print(Entrar_No_laco)
```

APÊNDICE B – Parte de Código Fonte do Módulo

```

IO.setmode (IO.BOARD)
IO.setup(Motor_Fechar,IO.OUT)
IO.setup(Motor_Abrir,IO.OUT)
IO.setup(Sensor_Janela_Fechada,IO.IN, pull_up_down=IO.PUD_UP)
IO.setup(Sensor_Janela_Aberta,IO.IN, pull_up_down=IO.PUD_UP)
IO.setwarnings(False)

```

```
try:
```

```
    while True:
```

```

        conn=sqlite3.connect("/home/pi/Desktop/MODULO/BD_SQLITE/B
D_Estado_Proximo.db")
        cursor = conn.cursor()
        cursor.execute("SELECT * FROM Tabela_1 cursor WHERE
EXISTS(SELECT 1 FROM Tabela_1 WHERE ID_Verificador =
cursor.ID_Verificador)")

```

```
        if cursor.fetchone():
```

```
            print("Existem Dados!")
```

```
with
```

```

sqlite3.connect("/home/pi/Desktop/MODULO/BD_SQLITE/BD_Estado_Proximo.db")
as conn:

```

```
    cursor = conn.cursor()
```

```
    cursor.execute("""SELECT ID_Verificador FROM Tabela_1""")
```

```
    for ID_Verificador in cursor.fetchone():
```

```
        ID_Verificador = ID_Verificador
```

```
        print("Aqui ID_Verificador " + ID_Verificador)
```

```
    cursor.execute("""SELECT Lampada_Proximo FROM Tabela_1""")
```

```
    for Lampada_Proximo in cursor.fetchone():
```

```
        Lampada_Atual = Lampada_Proximo
```

```
        print("Aqui Lampada_Proximo " + Lampada_Atual)
```

```
cursor.execute("""SELECT Tomada_Proximo FROM Tabela_1""")
for Tomada_Proximo in cursor.fetchone():
    Tomada_Atual = Tomada_Proximo
    print("Aqui Tomada_Proximo " + Tomada_Atual)
```

```
cursor.execute("""SELECT Janela_Proximo FROM Tabela_1""")
for Janela_Proximo in cursor.fetchone():
    Janela_Atual = Janela_Proximo
    print("Aqui Janela_Proximo " + Janela_Atual)
    print("")
```

```
if (Lampada_Atual=='0'):
    print("Lampada Desativada")
    IO.setmode (IO.BOARD)
    IO.setwarnings(False)
    #tira os avisos de erro
    IO.setup(Proximo_Estado_Lampada,IO.OUT)
    IO.output(Proximo_Estado_Lampada,0)
elif(Lampada_Atual=='1'):
    print("Lampada Ativa")
    IO.setmode (IO.BOARD)
    IO.setwarnings(False)
    #tira os avisos de erro
    IO.setup(Proximo_Estado_Lampada,IO.OUT)
    IO.output(Proximo_Estado_Lampada,1)
else:
    print("Erro na Lampada")
```

```
if (Tomada_Atual=='0'):
    print("Tomada Desativada")
    IO.setmode (IO.BOARD)
    IO.setwarnings(False)
    #tira os avisos de erro
```

```

IO.setup(Proximo_Estado_Tomada,IO.OUT)
IO.output(Proximo_Estado_Tomada,0)
elif(Tomada_Atual=='1'):
    print("Tomada Ativa")
    IO.setmode (IO.BOARD)
    IO.setwarnings(False)
    #tira os avisos de erro
    IO.setup(Proximo_Estado_Tomada,IO.OUT)
    IO.output(Proximo_Estado_Tomada,1)
else:
    print("Erro na Tomada")

if (Janela_Atual=='0'):#0 -> Abrir
    for x in range(Tempo_Abertura, Tempo_Total):
        time.sleep(1.0)
        #print("Entrda: " + str(Janela_Atual) )
        print("Tempo contando " + " " + str(x) )
        print(" ")
        if(IO.input(Sensor_Janela_Fechada) == True and
IO.input(Sensor_Janela_Aberta)== True ):
            print("Entro no laço " + str(Janela_Atual) + " " +
str(IO.input(Sensor_Janela_Fechada)) + " " + str(IO.input(Sensor_Janela_Aberta)))
            print("Neutro Abrindo " +str(Janela_Atual))
            IO.setmode (IO.BOARD)
            IO.setwarnings(False)
            #M1
            IO.setup(Motor_Fechar_32,IO.OUT)
            IO.output(Motor_Fechar_32,0)
            #M2
            IO.setup(Motor_Abrir_36,IO.OUT)
            IO.output(Motor_Abrir_36,1)
            #S1
            IO.setup(Led_Sensor_Fechada_26,IO.OUT)
            IO.output(Led_Sensor_Fechada_26,0)

```

```

#S2
IO.setup(Led_Sensor_Aberta_24,IO.OUT)
IO.output(Led_Sensor_Aberta_24,0)
    if(IO.input(Sensor_Janela_Fechada) == True and
IO.input(Sensor_Janela_Aberta)== False ):
        print("Entro no laço " + str(Janela_Atual) + " " +
str(IO.input(Sensor_Janela_Fechada)) + " " + str(IO.input(Sensor_Janela_Aberta)))
        print("Fechado Abrindo" +str(Janela_Atual))
        IO.setmode (IO.BOARD)
        IO.setwarnings(False)
#M1
IO.setup(Motor_Fechar_32,IO.OUT)
IO.output(Motor_Fechar_32,0)
#M2
IO.setup(Motor_Abrir_36,IO.OUT)
IO.output(Motor_Abrir_36,1)
#S1
IO.setup(Led_Sensor_Fechada_26,IO.OUT)
IO.output(Led_Sensor_Fechada_26,1)
#S2
IO.setup(Led_Sensor_Aberta_24,IO.OUT)
IO.output(Led_Sensor_Aberta_24,0)
    if(IO.input(Sensor_Janela_Fechada) == True and
IO.input(Sensor_Janela_Aberta)== False ):
        print("Entro no laço " + str(Janela_Atual) + " " +
str(IO.input(Sensor_Janela_Fechada)) + " " + str(IO.input(Sensor_Janela_Aberta)))
        print("Aberto " +str(Janela_Atual))
        IO.setmode (IO.BOARD)
        IO.setwarnings(False)
#M1
IO.setup(Motor_Fechar_32,IO.OUT)
IO.output(Motor_Fechar_32,0)
#M2
IO.setup(Motor_Abrir_36,IO.OUT)

```

```

IO.output(Motor_Abrir_36,0)
#S1
IO.setup(Led_Sensor_Fechada_26,IO.OUT)
IO.output(Led_Sensor_Fechada_26,0)
#S2
IO.setup(Led_Sensor_Aberta_24,IO.OUT)
IO.output(Led_Sensor_Aberta_24,1)
Janela_Atual = "0"
break
        if(IO.input(Sensor_Janela_Fechada) == False and
IO.input(Sensor_Janela_Aberta)== False ):
            print("Entro no laço " + str(Janela_Atual) + " " +
str(IO.input(Sensor_Janela_Fechada)) + " " + str(IO.input(Sensor_Janela_Aberta)))
            print("Erro nos Sensores 0" +str(Janela_Atual))
            IO.setmode (IO.BOARD)
            IO.setwarnings(False)
            #M1
            IO.setup(Motor_Fechar_32,IO.OUT)
            IO.output(Motor_Fechar_32,0)
            #M2
            IO.setup(Motor_Abrir_36,IO.OUT)
            IO.output(Motor_Abrir_36,0)
            #S1
            IO.setup(Led_Sensor_Fechada_26,IO.OUT)
            IO.output(Led_Sensor_Fechada_26,0)
            #S2
            IO.setup(Led_Sensor_Aberta_24,IO.OUT)
            IO.output(Led_Sensor_Aberta_24,0)
            Janela_Atual = "3"
            break
if(str(x) == str(Tempo_limite)):
    print("Erro no tempo ")
    #Janela_Atual = "3"
    for x in range(Tempo_Retorno, Tempo_Total):

```

```
time.sleep(1.0)
#print("Entrada: " + str(Janela_Atual) )
print("Tempo contando " + " " + str(x) )
print(" ")
        if(IO.input(Sensor_Janela_Fechada) == True and
IO.input(Sensor_Janela_Aberta)== True ):
            print("Entro no laço " + str(Janela_Atual) + " " +
str(IO.input(Sensor_Janela_Fechada)) + " " +
```