

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE GOIÁS
ESCOLA POLITÉCNICA E DE ARTES
GRADUAÇÃO EM ENGENHARIA DE COMPUTAÇÃO



**AUTOMAÇÃO RESIDENCIAL: SISTEMA DE SEGURANÇA UTILIZANDO
RASPBERRY PI**

PEDRO GABRIEL REIS PIRES

GOIÂNIA
2023

PEDRO GABRIEL REIS PIRES

**AUTOMAÇÃO RESIDENCIAL: SISTEMA DE SEGURANÇA UTILIZANDO
RASPBERRY PI**

Trabalho de Conclusão de Curso apresentado à Escola Politécnica e de Artes da Pontifícia Universidade Católica de Goiás, como parte dos requisitos para obtenção do título de Bacharel em Engenharia de Computação.

Orientador: Me. Gustavo Siqueira Vinhal

GOIÂNIA
2023

AGRADECIMENTOS

Agradeço primeiramente a Deus, nosso senhor, por todas benções derramadas em nós, e pela realização de todos objetivos e sempre estar ao meu lado diante de todas as dificuldades.

Aos meus familiares e amigos, em especial aos meus pais, Revelino Pires da Silva e Eliana Silva Reis, que sempre estiveram ao meu lado, me apoiando e incentivando durante toda a minha jornada acadêmica e nunca deixou faltar nada em casa. Suas palavras de encorajamento e apoio foram fundamentais para que eu pudesse perseverar nos momentos mais desafiadores.

Gostaria de expressar minha sincera gratidão a todas as pessoas que contribuíram e apoiaram-me ao longo deste processo de elaboração do meu Trabalho de Conclusão de Curso. Sem a ajuda e o encorajamento de vocês, não teria sido possível alcançar este marco importante em minha vida. Portanto, gostaria de dedicar este espaço para expressar minha profunda apreciação a todos os seguintes:

Meu orientador, Gustavo Siqueira Vinhal, por sua orientação perspicaz, sua paciência e dedicação em me auxiliar no desenvolvimento deste trabalho. Seus conselhos e conhecimentos foram fundamentais para o sucesso deste projeto.

A todos os meus professores, que compartilharam seu conhecimento e experiência ao longo dos anos de estudo. Suas aulas foram essenciais para a minha formação acadêmica e para a construção das bases deste trabalho e todos integrantes da Coordenação da Escola Politécnica e de Artes da Pontifícia Universidade Católica de Goiás pela disposição e prontidão a ajudar.

Agradeço também a todos os amigos e colegas de classe e demais membros da comunidade acadêmica que estiveram presentes ao longo desta jornada. A troca de ideias, debates e discussões enriqueceu meu conhecimento e me incentivou a buscar sempre mais.

Por fim, gostaria de expressar minha gratidão a todas as fontes bibliográficas e materiais consultados durante a pesquisa. As contribuições dos pesquisadores e autores nesta área de estudo foram fundamentais para a construção do embasamento teórico deste trabalho.

Muito obrigado a todos!

RESUMO

Avanços tecnológicos permitiram criar novas oportunidades de conforto, economia e, acima de tudo, segurança. O conceito de automatizar uma residência surgiu visando trazer para o ser humano esses benefícios. Além do conforto e modernidade, a automação residencial também traz um benefício prático para as pessoas. desfrutar a redução do trabalho físico e, conseqüentemente, a melhoria do bem-estar proporcionado pelos seus recursos, maximiza o tempo gasto em tarefas rotineiras. Conectar objetos digitais à rede global de computadores permite que esses objetos sejam controlados remotamente e usados como provedores de serviços. Dito isso, este trabalho tem como objetivo o desenvolvimento de um protótipo para um sistema que auxilie na automação residencial com foco na segurança. Para isso, será utilizado o *Home Assistant*. Ele é um aplicativo gerenciado por um supervisor que é executado em um contêiner do Docker. Ele pode ser executado em computadores de placa única com o sistema operacional dedicado Hass.io. Isso permite a portabilidade e versatilidade do sistema implementado.

PALAVRAS-chave: Automação Residencial, Home Assistant, Domótica, Raspberry Pi.

ABSTRACT

Technological advances have created new opportunities for comfort, savings and, above all, safety. The concept of automating a residence arose in order to bring these benefits to humans. In addition to comfort and modernity, home automation also brings a practical benefit to people. enjoy the reduction of physical work and, consequently, the improvement of well-being provided by their resources, maximizing the time spent on routine tasks. Connecting digital objects to the global computer network allows these objects to be remotely controlled and used as service providers. That said, this work aims to develop a prototype for a system that assists in home automation with a focus on security. For this, the Home Assistant will be used. It is a supervisor-managed application that runs in a Docker container. It can run on single board computers with Hass.io dedicated operating system. This allows the portability and versatility of the implemented system.

KEYWORDS: Home automation, Home Assistant, Home Automation, Raspberry PI.

LISTA DE ILUSTRAÇÕES

Figura 1 - A Internet das Coisas” nasceu” entre 2008 e 2009.....	14
Figura 2 - Tipos de arquitetura IoT.....	14
Figura 3 - Raspberry Pi 3 modelo B+.....	15
Figura 4 – Sensor de movimento PIR.....	17
Figura 5 – Buzzer.....	18
Figura 6 – Módulo Relé 2 canais.....	19
Figura 7 - Arquitetura do Hass.io.....	20
Figura 8: Módulos ADD-on.....	21
Figura 9 - Módulos do Home Assistant.....	21
Figura 10 - Home Control Core.....	22
Figura 11 - Componentes(atuadores/ sensores) Home Assistant.....	23
Figura 12 - Visão geral da interface web do Home Assistant.....	23
Figura 13: Visão geral da interface na criação de automação.....	26
Figura 14 - Local de instalação para integrações	26
Figura 15 - Local de instalação para integrações	26
Figura 16 – Circuito ilustrativo ideal.....	27
Figura 17 – Esquema ilustrativo definido.....	28
Figura 18 – Circuito físico.....	28
Figura 19 – Fluxograma sistema de segurança casa vazia.....	28
Figura 20 – Fluxograma automação dispara alarme.....	29
Figura 21 – Fluxograma sistema de segurança casa cheia.....	29
Figura 22 – Chave na interface do sistema.....	30
Figura 23 – Notificação de ativação do alarme.....	30
Figura 24 – Notificação de desativação do alarme.....	31
Figura 25 – Chave representado um sensor	31
Figura 26 – Notificação sensor detectou movimento.....	31

LISTA DE TABELAS

Tabela 1 – Especificações técnicas do Raspberry Pi

Sumário

1 INTRODUÇÃO	10
1.1 OBJETIVO GERAL	11
1.2 OBJETIVO ESPECÍFICO	11
1.3 ESTRUTURAS DO TRABALHO	11
2 AUTOMAÇÃO	12
2.1 AUTOMAÇÃO RESIDENCIAL	12
2.1.1 DOMÓTICA	13
2.2 INTERNET DAS COISAS	13
3 MATERIAIS E MÉTODOS	16
3.1 RASPBERRY PI	16
3.1.2 SENSORES	18
3.1.2.1 SENSORES PIR (<i>Passive Infrared Sensor</i>)	18
3.1.3 ATUADORES	19
3.1.3.1 BUZZER	19
3.1.3.2 MODULO RELÉ	20
3.2 SOFTWARE	21
3.2.1 HOME ASSISTANT	21
3.2.2 DEFINIÇÕES IMPORTANTES	26
3.3 Método de pesquisa	27
4 DESENVOLVIMENTO	29
4.1 CONFIGURAÇÃO DO SOFTWARE	29

4.1.1 IDE (Ambiente de desenvolvimento integrado).....	29
4.2 CONFIGURAÇÃO SISTEMA DE VIGILÂNCIA.....	31
4.2.1 CIRCUITO SISTEMA DE VIGILÂNCIA.....	32
4.2.2 FLUXOGRAMA.....	33
5 TESTES E RESULTADOS.....	36
5.1 EXPERIMENTO 1 – AUTOMAÇÃO “ALARME – ATIVO”	36
5.2 EXPERIMENTO 2 – AUTOMAÇÃO “Vigilância (casa vazia)”	37
5.3 EXPERIMENTO 3 – AUTOMAÇÃO “Vigilância (casa cheia)”	38
6 CONCLUSÃO.....	40
REFERÊNCIAS BIBLIOGRÁFICA.....	41
APÊNDICE A: CODIGO DO ARQUIVO CONFIGURATION.YAML.....	44
APÊNDICE B: CODIGO DO ARQUIVO AUTOMATIONS.YAML	45

1 INTRODUÇÃO

Todos os dias a tecnologia avança criando novas oportunidades de conforto, economia e, principalmente, segurança. Com isso, a automação residencial surgiu para simplificar e facilitar diversas atividades do dia a dia. Para isso, são necessárias algumas ferramentas, com baixo custo, que se conectem aos dispositivos de automação residencial por meio de um servidor web.

O mercado global de automação residencial, também conhecido como "domótica", passa atualmente por um período turbulento caracterizado pelo crescimento físico e aumento da cobertura da mídia sobre o tema. Segundo a Associação Brasileira de Automação Residencial (AURESIDE), a partir de 2019, a expectativa é crescer 30% ao ano, com financiamento projetado superior a R\$ 3,7 bilhões (ARAÚJO et al., 2018).

Apesar de ser uma realidade em algumas residências, a domótica é fortemente idealizado como uma tecnologia futurista. Apesar do alto custo, o planejamento cuidadoso é fundamental para garantir a integridade estrutural. A tecnologia nacional se tornará mais difundida, trazendo mudanças e inovações em projetos residenciais. Como resultado, o indivíduo experimentará inúmeras melhorias em conforto, segurança, confiabilidade, conveniência, acessibilidade e eficiência energética. (CABRAL, CAMPOS 2008).

A automação de tarefas, como sistemas de alarme e climatização, entre outras aplicações de tarefas automatizadas, tem a vantagem de trazer bem-estar e oferecer acessibilidade às pessoas. Considerando esse fato, as tarefas executadas por máquinas costumam ser executadas muito mais rapidamente do que as realizadas por humanos. Isso resulta em maior confiabilidade e menos esforço, para que os usuários possam se concentrar em outras tarefas mais difíceis, aumentando a produtividade (CABRAL, CAMPOS, 2008).

1.1 OBJETIVO GERAL

O objetivo deste projeto é desenvolver um protótipo para um sistema que auxilie na automação residencial com foco na segurança residencial.

1.2 OBJETIVO ESPECÍFICO

Desenvolver um sistema de automação residencial responsável pela vigilância da residência.

O sistema deverá cumprindo as seguintes atividades:

- Notificação: caso o sensor de presença detecte algum movimento deverá enviar mensagem para o usuário via um smartphone;
- Sirene: Acionar a sirene de segurança, caso o sistema de vigilância esteja ativo, caso os sensores de presenças detectem algum movimento.

1.3 ESTRUTURAS DO TRABALHO

Este trabalho foi dividido em quatro capítulos. O primeiro capítulo fornece uma breve visão geral do assunto, bem como os objetivos declarados e os resultados esperados. Com a descrição detalhada da automação residencial no Capítulo 2 dá ao leitor uma compreensão mais profunda do tema. A lista de tecnologias e componentes eletrônicos utilizados neste trabalho é apresentada no Capítulo 3, juntamente com informações sobre a funcionalidade de cada uma em relação à sua aplicação. O Capítulo 4 apresenta o testes realizados. Por fim, o Capítulo 5 apresenta a conclusão do trabalho.

2 AUTOMAÇÃO

Aguirre (2006) define automação como sistemas ou sistemas autônomos que fornecem soluções sofisticadas e ajustam automaticamente suas funções, principalmente para reduzir ou mesmo eliminar o envolvimento humano nos processos industriais (SILVA, 2016). Segundo Sousa (2018), os sistemas autônomos incluem:

Capazes de realizar uma tarefa sozinho, tomando suas próprias decisões, com base nas informações obtidas pelos sensores os quais são equipados. Essa autonomia é um desafio de grande interesse em muitas aplicações (SOUSA, 2018).

Existem muitas ferramentas distribuídas pela casa e utilizadas no ambiente doméstico para atender às necessidades do usuário, e podemos agrupar essas ferramentas em três grandes categorias (TAKIUCHI, MELO e TONIDANDEL, 2004):

- **Atuadores:** Controle de eletrodomésticos como iluminação, automação e irrigação.
- **Sensores:** Registre informações ambientais como umidade e presença.
- **Controladores:** Ajuste todos os sensores e atuadores para atender aos requisitos dos ocupantes para equipamentos de automação residencial

2.1 AUTOMAÇÃO RESIDENCIAL

Um processo ou sistema conhecido como automação residencial, ou domótica, prioriza a melhoria da qualidade de vida das pessoas, conforto, segurança e economia doméstica por meio do controle centralizado de funções como sistemas de água, luz, telefone, segurança, entre outros (ANGEL, 1993 e NUNES, 2002).

O princípio básico da domótica é aplicar a tecnologia de tal forma que seu objetivo central seja melhorar a qualidade de vida do usuário por meio de seu uso. Retenha a redução do trabalho físico e, inexoravelmente, a melhoria do bem-estar proporcionada pelos seus recursos (ANGEL, 1993).

A lista de recursos da casa inteligente é extensa. Você pode operar eletrodomésticos, condicionadores de ar, máquinas de café, fornos de micro-ondas, TVs, rádios, máquinas de lavar, smartphones, além de poder, usar o controle remoto para acender e apagar todas as luzes, ativar sistemas de segurança e abrir e fechar

portas, até mesmo um celular dispositivo. A ideia da "Internet das Coisas" (IoT) nasceu nesse contexto. Seu objetivo é conectar os dispositivos eletrônicos que usamos diariamente à internet ou bancos de dados, processar suas informações e devolvê-las ao usuário na forma de benefícios. (Wanzeler, 2016).

A automação residencial (AR) já oferece muitos benefícios em relação aos sistemas eletrônicos e eletromecânicos isolados e ineficientes. Que sistemas operacionais autônomos pressionam a infraestrutura dedicada torna isso óbvio. De fato, nos últimos anos, a AR desperta o interesse de todos, não apenas daqueles com formação técnica na área. As principais razões pelas quais podemos facilmente discutir tecnologia em casa em uma linguagem antes disponível apenas para analistas de sistemas é a computação pessoal e a internet (Wanzeler, 2016).

2.1.1 DOMÓTICA

O termo, "domótica", refere-se à automatização de uma residência através da instalação de equipamentos eletrônicos com capacidades de processamento, gestão e aprendizagem. Outros termos para domótica podem incluir "construção inteligente", "casa inteligente", "ambiente inteligente", entre outros. (ANGEL, 1993).

Devido aos inúmeros processos que estão sendo realizados, os sistemas robóticos domésticos exigem uma variedade de dispositivos de gerenciamento. Isso permite o maior controle do equipamento sem sacrificar o conforto adicional. (BOLZANI, 2004b).

2.2 INTERNET DAS COISAS

Em 1991, Mark Weiser (WEISER, 1991) apresentou o conceito de "Internet das Coisas" ao mundo pela primeira vez. Ele fez isso provocando uma discussão sobre a tendência tecnológica em direção ao que ele chamou de computação ubíqua, um termo que descreve um computador invisível que é imperceptível para as pessoas. Esses computadores estariam conectados mutualmente por meio de uma rede, criando um sistema inteligente e onipresente que auxilia as pessoas em seu dia a dia sem o seu conhecimento.

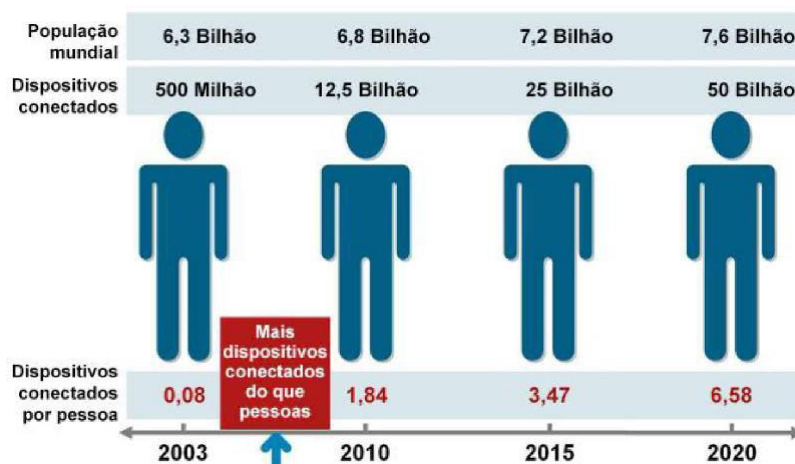
Depois de algum tempo, o termo "IoT", ou "Internet das Coisas", foi criado pelo

pesquisador britânico Kevin Ashton no início do ano 2000 ("UMA...", 2013 "). Em 1999, ele usou a frase pela primeira vez enquanto trabalhava em um projeto de integração de RFID (Identificação por Radiofrequência) com a Internet no laboratório AutoID do Massachusetts Institute of Technology (MIT).

Segundo (SANTOS et al., 2016), a tecnologia da Internet das Coisas (IoT) surgiu a partir de avanços na computação subatômica e áreas afins como microeletrônica, comunicação e sensoriamento. Em decorrência de seu potencial de uso nas mais variadas esferas da atividade humana, atraiu um interesse significativo da academia e da indústria. De acordo com (SANTOS et al., 2016), conectar objetos digitais à rede global de computadores permitirá que esses objetos sejam controlados remotamente em primeiro lugar e usados como provedores de serviços em segundo.

Como pode ser visto na Figura 1, em 2003 havia cerca de 6,3 bilhões de pessoas no Planeta Terra e 500 milhões de dispositivos conectados à Internet. Com base na definição do Cisco IBSG, a Internet das Coisas não existia em 2003 porque não havia muitas coisas conectadas e dispositivos não tradicionais, como celulares, ainda não haviam sido lançados (EVANS, 2011).

Figura 1: A Internet das Coisas "nasceu" entre 2008 e 2009.

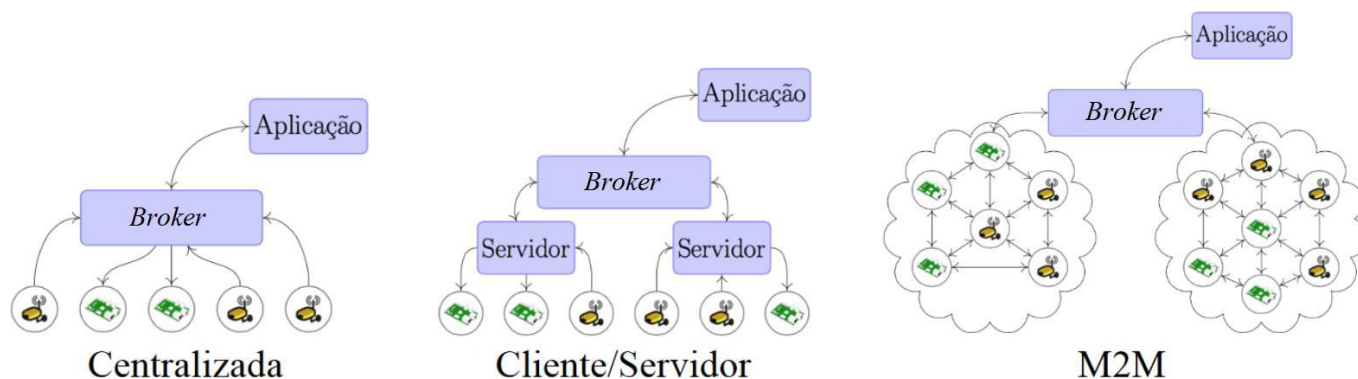


Fonte: Adaptado de (EVANS, 2011).

O Cisco IBSG estima que em algum momento entre os anos de 2008 e 2009, quando o número de dispositivos conectados à Internet ultrapassou o número de pessoas no planeta, a Internet das Coisas "nasceu". A Cisco IBSG prevê que haverá 50 bilhões de dispositivos conectados à Internet até o ano de 2020. (EVANS, 2011).

Em sua dissertação, (NEVES, sd) afirma que o paradigma da IoT pode ser resumido como uma rede heterogênea que conecta “coisas” à Internet, provocando mudanças tanto na comunicação quanto nos serviços prestados pela Internet. A cooperação permite a coleta e compartilhamento de informações do mundo real, o que possibilita o desenvolvimento de serviços individualizados. A troca de informações é centrada na comunicação M2M entre objetos (NEVES, sd). Alguns exemplos de arquiteturas de IoT envolvendo dispositivos e aplicativos são mostrados na Figura 2.

Figura 2: Tipos de arquitetura IoT.



Fonte: Adaptado de (MUSSIO; MAIA; LOPES, 2015).

Conforme ilustra a Figura 2, o único componente sobre o qual a arquitetura centralizada é construída, o Broker, é responsável por enviar as informações dos dispositivos para a aplicação. Uma arquitetura cliente/servidor é baseada na centralização, com a ressalva de que a informação é pré-processada por servidores e que, ao final, uma arquitetura máquina a máquina utiliza seus próprios dispositivos como distribuidores de informação (MUSSIO; MAIA; LOPES, 2015).

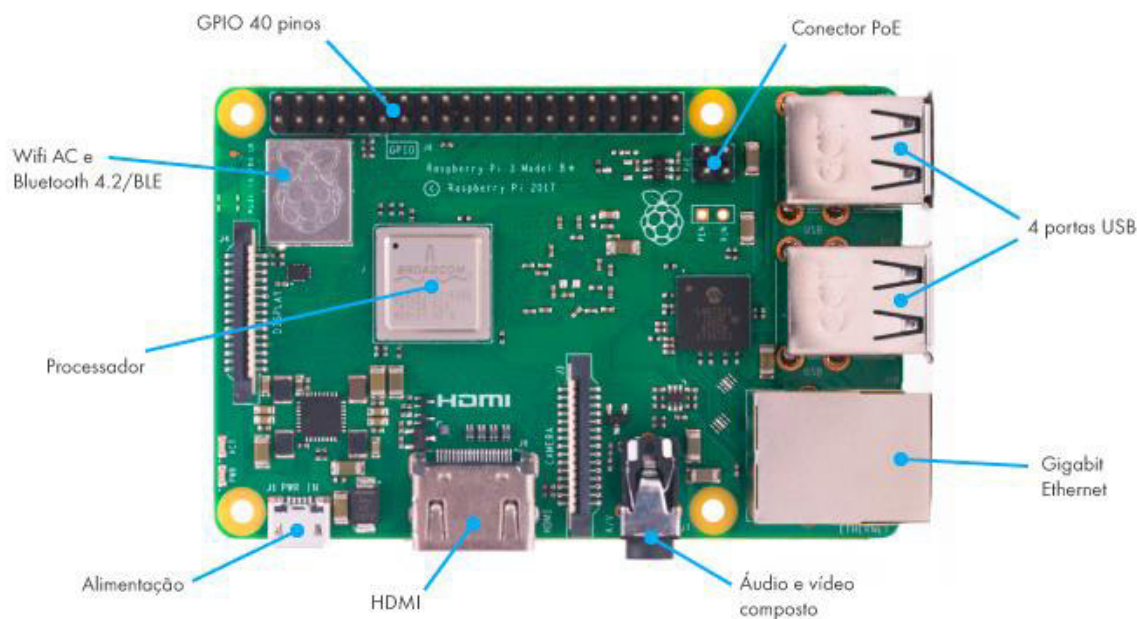
3 MATERIAIS E MÉTODOS

As tecnologias utilizadas para desenvolver este projeto (*hardware e software*) e a metodologia de pesquisa serão apresentadas neste capítulo.

3.1 RASPBERRY PI

O Raspberry Pi é um microcomputador de baixo custo projetado para possibilitar o aprendizado de programação sem incorrer em custos significativos. Além disso, tem GPIOs que permitem a criação de projetos eletrônicos, bem como projetos utilizando microcontroladores. Existem inúmeros modelos Raspberry Pi, com destaque para a versão 3, modelo B+ (RASPBERRYPI, 2021). Na Figura 3 é mostrada uma foto de um Raspberry Pi 3 modelo B+.

Figura 3: Raspberry Pi 3 modelo B+



Fonte: Fundação Raspberry Pi, 2021.

O Raspberry Pi usa um processador ARM (*Advanced RISC Machine*). Tal como o nome indica, RISC (Conjunto Reduzido de Instruções de Computador) são instruções que um processador que foi abrandado executa eficazmente para desempenhar as suas funções, como, por exemplo, abrir uma aplicação. Isso o torna mais eficaz e quente menos, exigindo menos energia.

O Raspberry Pi 3 Modelo B+ possui um *chip* Broadcom BCM2837, com quatro processadores de 64 *bits*, aumentando o desempenho em até 50% em relação ao seu antecessor. Embora a quantidade de memória não tenha mudado, o *clock* foi aumentado para 400 MHz. Dado que o relógio sofreu modificações, foi necessário que a corrente elétrica aumentasse em função da proporção de circuitos inseridos no *display*. Além disso, um *chip* de antena WI-FI e *Bluetooth* foi adicionado a esta versão, negando a necessidade de uma conexão de antena externa (RASPBerryPI, 2018).

Em contraste com os microcontroladores, o Raspberry roda Linux como seu sistema operacional, assim como um computador pessoal faria. Ele possui uma interface Ethernet, e também é possível usar um adaptador Wi-Fi que está conectado a uma de suas quatro portas USB. O armazenamento de dados é feito por um cartão MicroSD, que funciona para o Raspberry de maneira prática semelhante a um disco rígido para um computador. Esse microcomputador possui pinos que permitem a conexão com outros dispositivos eletrônicos, possibilitando o controle de atuadores e a leitura de sensores como resultado (MONK, 2016).

Pinos com propósitos gerais são chamados de GPIO (*General Purpose Input Output*). O Raspberry Pi tem 40 pinos, incluindo 28 pinos programáveis GPIO e 12 pinos não programáveis. Juntamente com os 28 GPIOs, há também pinos para a terra e a linha de alimentação de 5V. Como o Raspberry é um computador funcional, ele também possui duas portas micro-USB, uma porta HDMI, um slot para cartão de memória e um conector de vídeo RCA para uso com televisores mais antigos. Uma fonte de 2,5 A e 5 V é necessária para a operação (RASPBerryPI, 2018).

A Tabela 1 mostra outras informações a respeito das especificações do hardware.

Tabela 1 – Especificações técnicas do Raspberry Pi

Processador	Broadcom BCM2837B0
Memoria	1GB LPDDR2 SDRAM
Conectividade	2.4GHz e 5GHz IEEE 802.11.b/g/n/ac wireless LAN, Bluetooth
Acesso	Extended 40-pin GPIO header
Vídeo e Som	1 × full size HDMI MIPI DSI display MIPI CSI Câmera porta 4
Multimídia	H.264, MPEG-4 decode (1080p30)
Suporte de Cartão SD	Formato Micro SD para carregar o sistema operacional
Entrada de alimentação	5V/2.5A DC via micro USB 5V DC via GPIO header
Ambiente	Temperatura operacional, 0–50 ° C

Fonte: RASPBERRYPI, 2021.

3.1.2 SENSORES

Os sensores são dispositivos eletrônicos úteis em ambientes automatizados. Eles são responsáveis por identificar os sinais físicos do ambiente designado e converter esses sinais em informações que os sistemas de computador possam manipular (ACCARDI, 2012). Esses detalhes podem ser analógicos ou de natureza digital.

3.1.2.1 SENSORES PIR (*Passive Infrared Sensor*)

A Figura 4 apresenta o sensor PIR. Esse tipo de sensor é chamado de passivo porque não emite luz infravermelha. Pelo contrário, eles se informam sobre as mudanças na infraestrutura do ambiente. A radiação infravermelha está presente no espectro eletromagnético e é invisível para os seres humanos. Materiais que geram calor também emitem radiação infravermelha. Quando o sensor é acionado, ele faz a leitura da radiação infravermelha que está voltando do ambiente sem movimento (BEGHINI, 2013).

Figura 4 – Sensor de movimento PIR



Fonte: ELETROGATE, 2023.

3.1.3 ATUADORES

Os atuadores são responsáveis por realizar determinadas atividades em um ambiente físico. Esses dispositivos eletrônicos conseguem realizar ou controlar uma ação, ou movimento específico.

Um atuador pode trabalhar de forma ativa que precisa de uma constante alimentação externa como a energia ou passiva que utiliza a energia gerada pela ação da mecânica dos efetadores em sua interação ao ambiente exposto (SOUSA, 2018).

3.1.3.1 BUZZER

Uma sirene de 12V é a melhor escolha por ter alto volume de som e precisar de poucos componentes, como um relé e um chaveamento da bateria de 12V. No entanto, uma de 3V foi escolhida por ter o menor valor, 5 reais, e atender à necessidade por ser um protótipo.

A Figura 5 exibe um *buzzer*. É um dispositivo simples que produz ruídos audíveis quando ligado. Trabalhe com várias frequências que produzem tonalidades distintas. Projetos de campanha e sistemas de alarme funcionam melhor com um *buzzer*.

Figura 5 – Buzzer



Fonte: ELETROGATE, 2023.

3.1.3.2 MODULO RELÉ

Relés são aparelhos elétricos que têm como uma de suas funções a capacidade de passar corrente de alta tensão (110v ou 220v) utilizando um sinal de baixa tensão (5v). Ele seja um interruptor de baixa tensão ativado. A Figura 6 descreve um módulo vinculado com um canal.

Figura 6 – Módulo Relé 2 canais



Fonte: ELETROGATE, 2023.

Conforme a Figura 6, cada canal é um Relé, permitindo a integração com diversos dispositivos controláveis. A versatilidade de cada módulo permite sua

utilização em diversos projetos. No protótipo foi colocado ao lado do Raspberry Pi para as execuções de teste de inserção.

3.2 SOFTWARE

O *software* selecionado para fazer a implementação desse projeto foi o “Home Assistant”.

3.2.1 HOME ASSISTANT

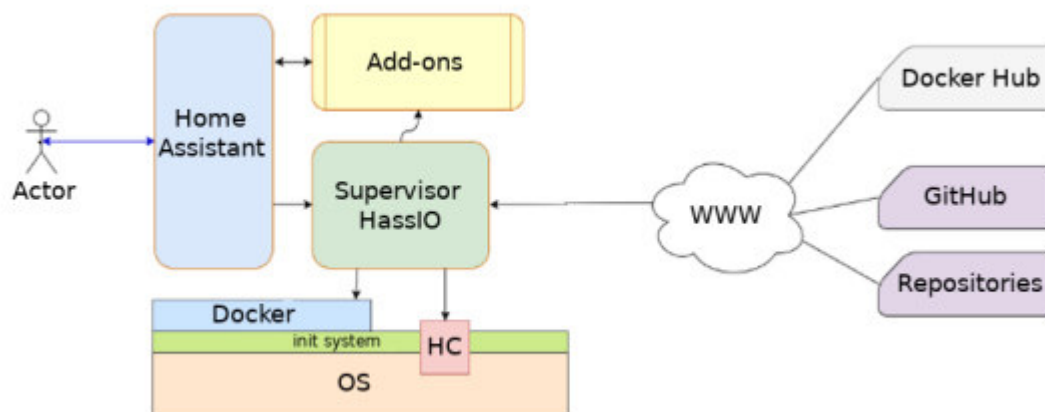
O *Home Assistant* é uma alternativa gratuita e de código aberto utilizada para desenvolver sistemas residenciais de automação descentralizada que controlam atuadores, avaliam dados coletados por sensores, colocam em prática regras de automação e gerenciam a comunicação entre dispositivos. O *Home Assistant* pode ser executado em computador Linux, diretamente em computadores de placa única com o sistema operacional dedicado Hass.io ou em qualquer outro ambiente que tenha o Python e suas dependências instaladas. Segundo Assistant (2019e), dentre as vantagens do Home Assistant pode-se destacar:

- É grátis e *Open Source*;
- É otimizado para rodar em SBCs;
- Permite que todas as automações sejam implantadas localmente;
- É de Fácil instalação;
- Possui interface web interativa para controle;
- Oferece rollback para versões antigas e
- É escalável, permitindo trabalhar com add-ons

O sistema operacional Hass.io foi criado para facilitar a instalação e o uso do *Home Assistant*. Como é possível ver na Figura 7, o *Home Assistant* é uma aplicação gerenciada por um supervisor que roda em um contêiner Docker e que, por sua vez, está sendo executado por um sistema operacional criado especialmente para SBCs. Com o supervisor, temos uma API que, além de gerenciar os processos do *Home Assistant*, também se encarregará de realizar as atualizações do sistema. Todas as configurações de rede do sistema são gerenciadas através da interface HC (Host Control), permitindo fazer configurações de rede e hardware

diretamente através do aplicativo *Home Assistant* (ASSISTANT, 2019a).

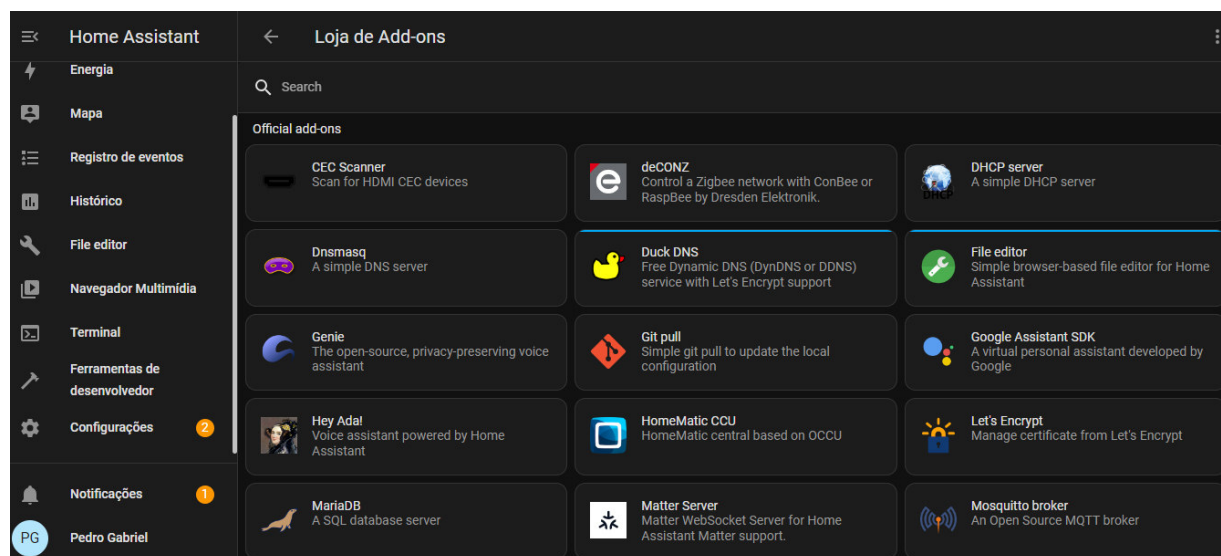
Figura 7: Arquitetura do Hass.io



Fonte: Developers Home Assistant, 2023.

Como o *Home Assistant* pretende ser um aplicativo escalável que opera sob demanda, o módulo add-on é responsável por adicionar recursos ao nosso sistema, assim mostrado na Figura 8. A loja virtual de complementos do *Home Assistant* é muito extensa e permite adicionar componentes de terceiros via github além dos complementos oficiais que dão ao sistema funcionalidades adicionais, como o Home Broker MQTT, DNS (*Domain Names System*), arquivo servidor e SSH (*SecureShell*), entre outros.

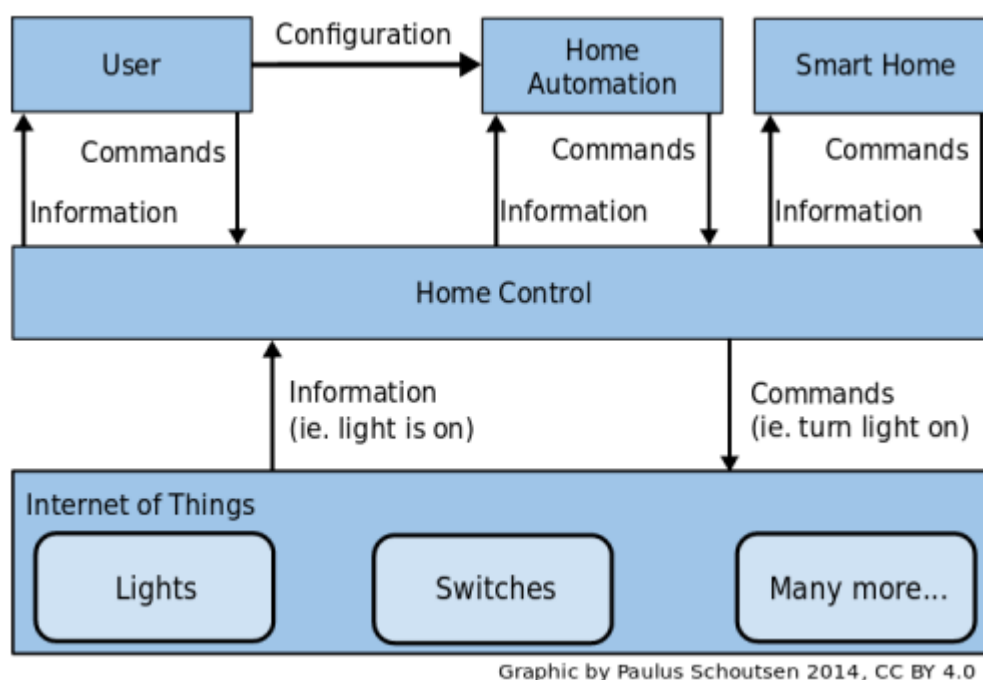
Figura 8: Módulos ADD-on



Fonte: Developers Home Assistant, 2023.

Como pode ser visto na Figura 9, o *Home Assistant* possui uma arquitetura de cinco partes, sendo o *Home Control* responsável pela coleta de dados e gerenciamento dos atuadores, user gerenciando todos os usuários do sistema, *Home Automation* ativando comandos com base nas configurações fornecidas pelo usuário, *Smart Home* ativando comandos com base nos estados de outros atuadores e sensores, e a “Internet das Coisas” sendo os atuadores e sensores reais.

Figura 9: Módulos do Home Assistant



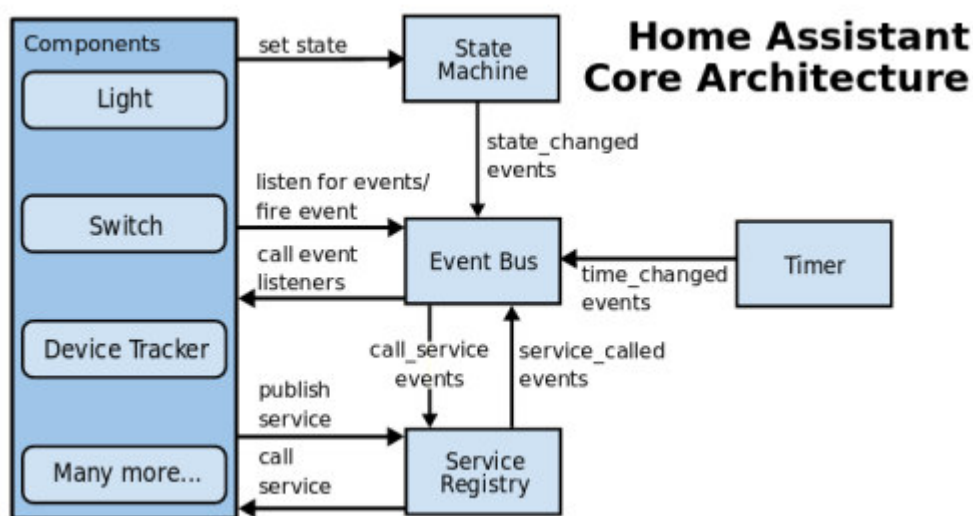
Fonte: Developers Home Assistant, 2023.

Para entender melhor como funciona o *Home Assistant*, é necessário definir os conceitos de evento, ação e serviço. Uma ação é um evento que termina depois que todas as condições necessárias para que ele aja sejam satisfeitas, e serviços são métodos usados para realizar ações. Um evento é qualquer coisa que nosso sistema pode observar que altera o estado de qualquer atuador ou sensor.

Ao examinar de perto o *Home Control*, pode-se dividi-lo em seções menores para entender melhor como o *Home Assistant* lida com as ações. Como mostra na Figura 10, o *Home Control* é dividido em quatro módulos separados. O módulo *Event*

Bus é responsável por ouvir e acionar eventos para os atuadores e manter o sistema em sincronia com o mundo real. Isso ocorre porque o *Bus Event* é o que desencadeia ações quando um atuador ou sensor precisa ser trocado. O módulo conhecido como *State Machine* mantém o controle de todos os estados dos atuadores e notifica o *Event Bus* sempre que um estado muda. O *Service Registry* cria uma lista de serviços que, por conta própria, realizam ações para os usuários. O componente conhecido como *Timer* é responsável por atualizar o horário no *Event Bus* e enviar um evento de mudança de horário para o *Home Control* uma vez a cada segundo (ASSISTANT, 2019b).

Figura 10: Home Control Core



Fonte: Developers Home Assistant, 2023.

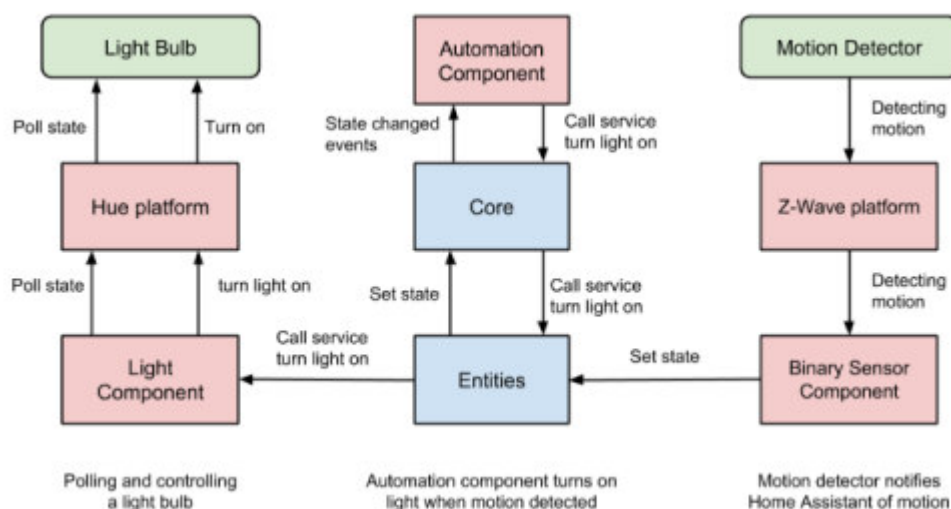
Essa arquitetura distribuída torna implícita a alternativa do *Home Assistant* para lidar com falsos positivos/negativos. Quando um serviço é invocado pelo *Service Registry*, mas nenhuma alteração é feita na *State Machine*, isso indica que houve uma falha de comunicação ou algum outro erro durante a execução da ação invocada.

É necessário que o *Home Assistant* forneça um tratamento, pois existem inúmeros fabricantes de dispositivos de automação inteligente, e cada um deles utiliza seu próprio conjunto de protocolos e padrões de comunicação. Isso permitirá que os dispositivos se comuniquem com a mais ampla gama de outros dispositivos possíveis e permitir que eles cooperem uns com os outros.

Como é possível ver na Figura 11, para comunicar com o *Home Assistant*, são

necessários atuadores e sensores, como a Lâmpada e os Detectores de Movimento (representados na imagem como *Light Bulb* e *Motion Detectos*), além do módulo de *software* responsável por fornecer inteligência aos atuadores e sensores, a *Hue Plataforma* e a *Z-Wave Plataforma*, ambas executadas pelo respectivo atuador ou sensor. Além disso, é necessário um conjunto de *softwares* que se encarregue de implementar os protocolos utilizados pelos atuadores e sensores e que opere de acordo com suas regras, conforme mostra a imagem do *Light Component*. Por fim, a interação dos sensores e atuadores compatíveis com o Home Assistant, é realizada utilizando os eventos, ação e serviço. Na ilustração, esta camada é representada por *Entities*.

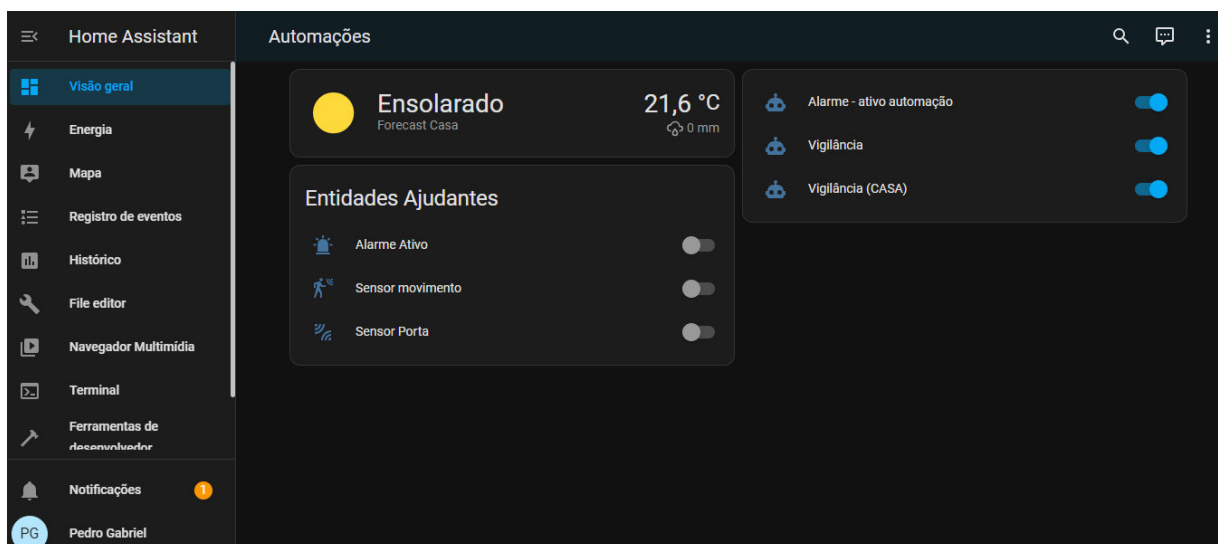
Figura 11: Componentes(atuadores/ sensores) Home Assistant



Fonte: Developers Home Assistant, 2023.

Conforme visto na Figura 12, além do editor de automação, a interface *online* do *Home Assistant* também fornece uma visão geral dos dispositivos e automações que fazem parte do sistema, dando aos usuários acesso rápido ao controle prático da entidade. A página do *site* também facilita o acesso a quase todos os recursos do sistema, incluindo uma loja virtual para complementos. A porta 8123 do dispositivo de instalação do sistema pode ser usada para acessar a interface *web*.

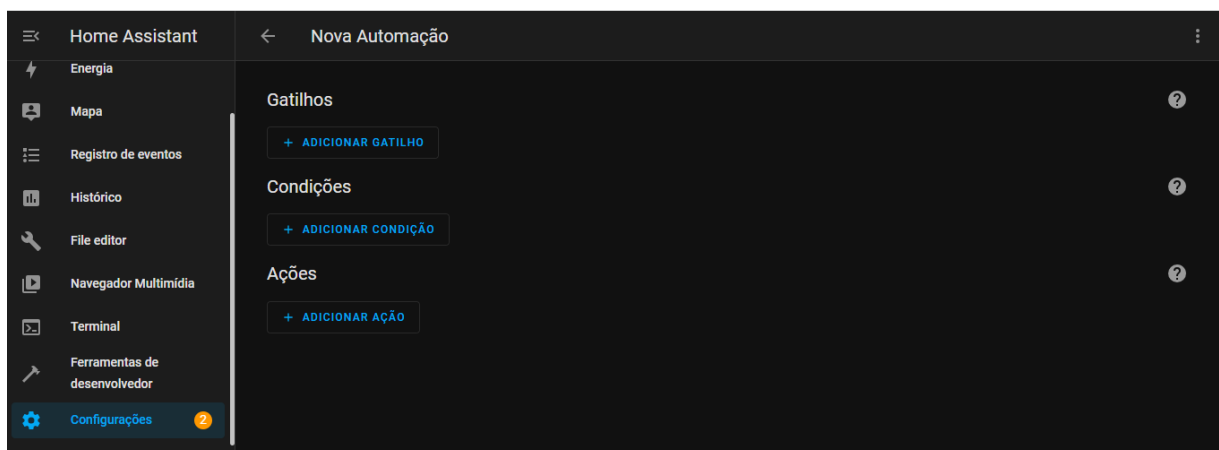
Figura 12: Visão geral da interface web do Home Assistant



Fonte: Autoria própria, 2023

3.2.2 DEFINIÇÕES IMPORTANTES

Figura 13: Visão geral da interface na criação de automação



Fonte: Autoria própria, 2023

Na figura 13 mostra a parte responsável por criar as automações, e em seguida está algumas definições importantes na de criar uma automação.

- **Entity:** Cada dispositivo no *Home Assistant* é representado como uma entidade separada. O trabalho interno do Home Assistant é divulgado por uma entidade. Um desenvolvedor de componente integrador não precisa se preocupar com o funcionamento dos serviços ou da máquina de estado. Em vez disso, ele estende uma classe de Entidade e implementa os atributos e procedimentos necessários para o tipo de dispositivo que está integrando;

- **Trigger:** Gatilho que irá iniciar a ação, "Action", podem haver vários gatilhos em uma automação, cada gatilho necessita que uma plataforma seja declarada;
- **Condition:** Requisito adicional ao Trigger que deverá ser cumprido para iniciar a ação, também necessita que uma plataforma seja declarada, mas possui variáveis diferentes do Trigger para cada plataforma;
- **Action:** Ação que será realizada, necessita que um ou mais serviços sejam declarados, uma lista de serviços disponíveis pra cada domínio pode ser encontrada no painel principal em ferramentas de desenvolvedores.

3.3 Método de pesquisa

A metodologia de pesquisa foi construída sobre o uso de video aulas, bibliotecas, livros e artigos acadêmicos, os quais forneceram a base para o desenvolvimento deste projeto. De forma que a mesma pode ser descrita como pesquisa bibliográfica que deriva suas tirar conclusões a partir de materiais já preparados para uso e a partir daí gerar novas hipóteses, como aconteceu neste caso particular.

O ambiente residencial e suas atividades autonômicas passivas foram observados de forma geral. Pela facilidade de uso e economia do projeto, então seria possível entregar benefícios. Para a escolha dos componentes, foi considerada a relação custo-benefício, tendo em vista o menor valor orçamentário global.

Os componentes foram escolhidos após a realização de inúmeros estudos sobre o assunto, a fim de possibilitar a seleção desses recursos, bem como seu uso racional naquele momento. Desta forma, foi escolhido o microprocessador Raspberry Pi. Tendo em conta a sua eficácia em atingir os objetivos do projeto. O fato de ter suporte de inúmeras bibliotecas facilita muito seu desenvolvimento consoante o protótipo. Juntamente com software Home Assistant, que possibilita fácil integração entre diversos dispositivos, independentemente da sua marca, o software opera para que

eles trabalhem em conjunto.

De forma geral, foi determinado que automatizar sistema com ênfase em segurança, já que tinha poucos projetos que implementava o mesmo. O acionamento foi acionado por um smartphone. A prototipagem se fez com alguns jumpers conectados ao GPIO e estabelecendo a troca de mensagem entre os sensores e os atuadores e sendo gerenciados pelo microprocessador.

4 DESENVOLVIMENTO

Nesta seção, serão detalhada como foi feito as devidas configuração e implementações para poder ser realizado os devidos testes do sistema.

4.1 CONFIGURAÇÃO DO SOFTWARE

O Raspberry deve estar conectado à fonte de alimentação e ao controlador via cabo Ethernet. Feito isso, o microcontrolador começará a projetar um servidor *web* que pode ser acessado em "http://homeassistant.local:8123" usando um navegador em um computador conectado a rede LAN, WiFi ou Ethernet do mesmo roteador. A página deve exibir uma mensagem indicando que uma atualização está em andamento.

Dependendo da velocidade da internet, o sistema atualizado abrirá a interface de configuração principal após cerca de 40 minutos.

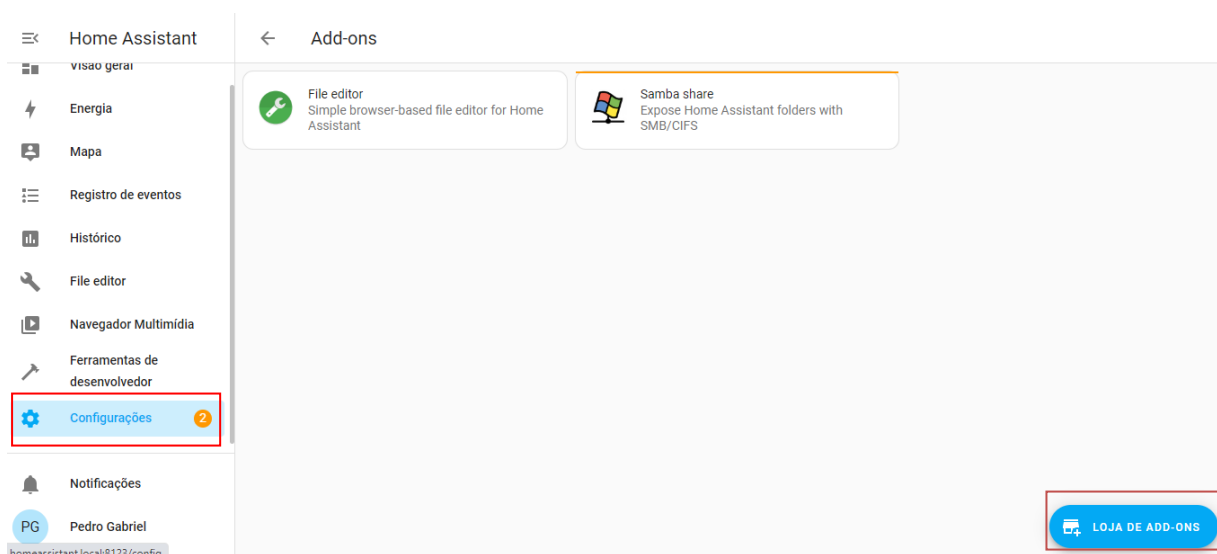
4.1.1 IDE (Ambiente de desenvolvimento integrado)

A instalação de Add-ons como "SAMBA SHARE" ou "FILE EDITOR" que possibilitar o acesso aos arquivos de configuração interna do sistema é necessária para a configuração do sistema. Os arquivos de configuração interna do sistema são necessários para a configuração do sistema. Por meio de dispositivos acessíveis pela rede, o SAMBA SHARE oferece facilidade de acesso ao processo de registro de acesso. Os documentos são disponibilizados por pelo FILE EDITOR em um ambiente de editor arquivo código com uma interface de navegador em um ambiente de editor de código com uma interface de navegador.

O FILE EDITOR foi escolhido para o projeto porque inclui detecção automática de erros de código, automática de erros de código, uma lista de todas as entidades, plataformas, serviços e eventos, um link para a biblioteca de informações do componente e a capacidade de usar botões para reiniciar as seções do sistema uma lista de todas as entidades, plataformas, serviços e eventos, um link para a biblioteca de informações de componentes e a capacidade de usar botões para reiniciar as seções do sistema.

Para a instalação, deve-se acessar o menu principal do Home Assistant, devendo-se selecionar a opção "Hass.io", conforme Figura 14. A lista de Add-ons instalados e atualizações disponíveis é exibida no menu "Configurações", e o menu "ADD ON STORE" deve ser selecionado. No momento da escrita, existem 20 Add-ons oficiais e 14 Add-ons primários criados pela comunidade "Github", com a possibilidade de Add-ons adicionais semelhante ao FILE EDITOR, qualquer Add-on pode ser instalado simplesmente clicando no botão loja de ADD-ONS e "instalar". Após a instalação, o complemento aparecerá na lista de itens instalados, onde o botão "iniciar" deve ser pressionado.

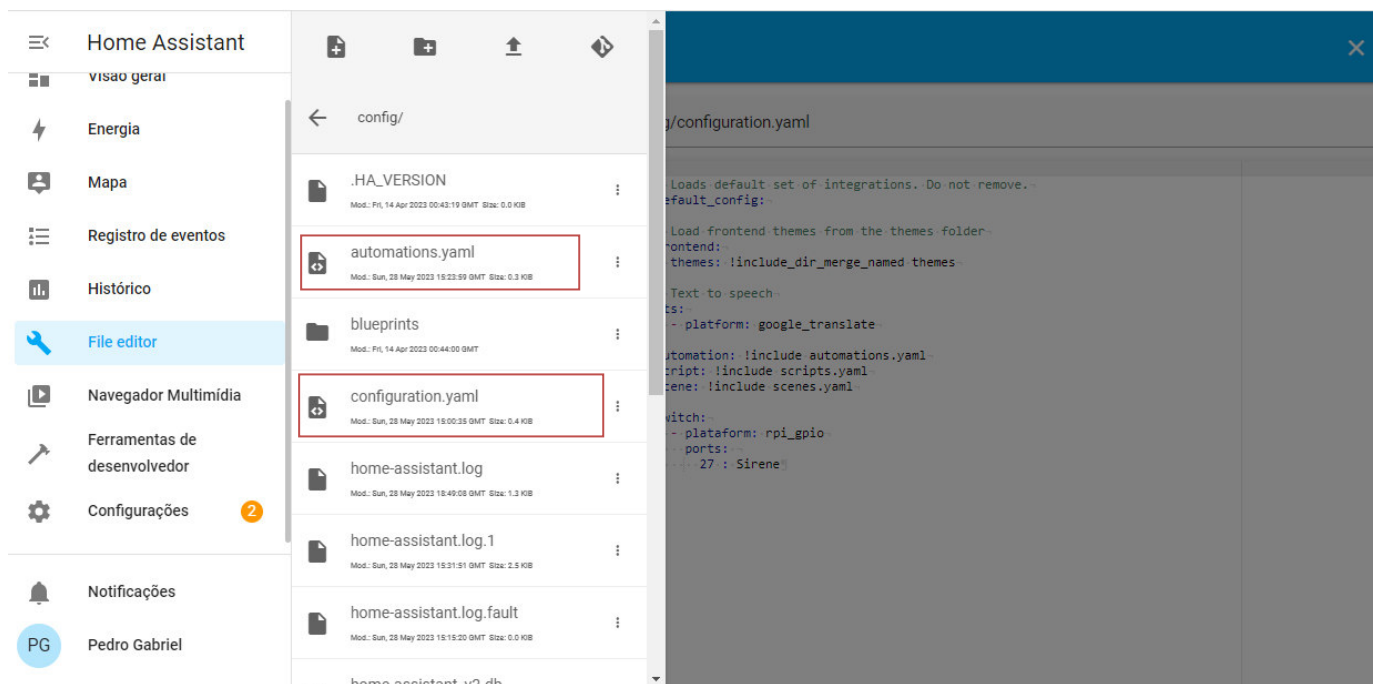
Figura 14 - Local de instalação para integrações



Fonte: Autoria própria, 2023

A Figura 15 apresenta o local dos arquivos principais que serão editados no trabalho assim como o local do ícone de detecção de erros no código.

Figura 15 - Local de instalação para integrações



Fonte: Autoria própria, 2023

Para que uma alteração ao arquivo “configuration.yaml” tenha efeito, o sistema deve ser reiniciado abrindo o “painel principal”, selecionando “Configuration”, depois “General” e depois o botão “RESTART”. Agora o editor irá aparecer no “painel principal” e irá abrir na mesma aba do navegador.

4.2 CONFIGURAÇÃO SISTEMA DE VIGILÂNCIA

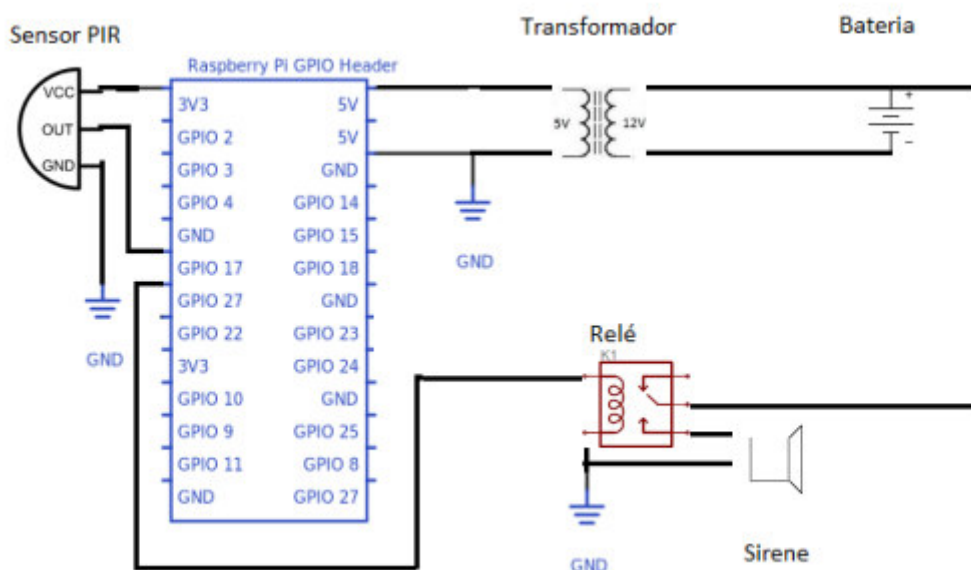
Para adicionar a função de sistema de vigilância primeiramente os sensores e atuadores necessários devem ser adicionados ao sistema adicionando algumas linhas de código ao registro "configuration.yaml", apenas copiando-os no final do documento.

O código contido nas linhas 17 a 21 do Apêndice A adiciona um sensor binário ao GPIO 11, também conhecido como pino 23. O código contido nas linhas 23 a 27 do apêndice A adiciona o atuador ao GPIO 17, também conhecidos como pinos 11.

4.2.1 CIRCUITO SISTEMA DE VIGILÂNCIA

A Figura 16 ilustra o esquema de ligação com o uso da sirene de 12V.

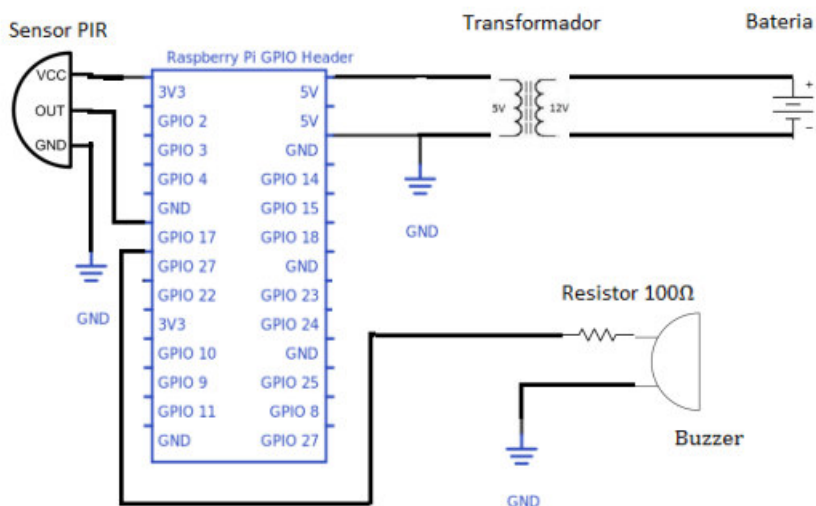
Figura 14 – Circuito ilustrativo ideal



Fonte: Autoria própria, 2023

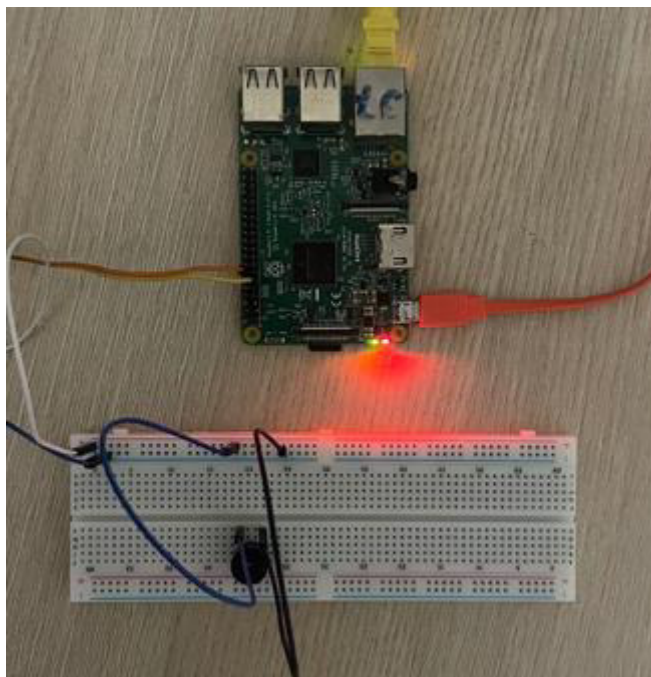
A sirene de 12V foi substituída por uma de 3,3V como definido no item 3.1.4.1 logo a Figura 17 ilustra o circuito implementado e na figura 18 mostra o circuito físico.

Figura 17 – Esquema ilustrativo definido



Fonte: Autoria própria, 2023.

Figura 18 – Circuito físico



Fonte: Autoria própria, 2023.

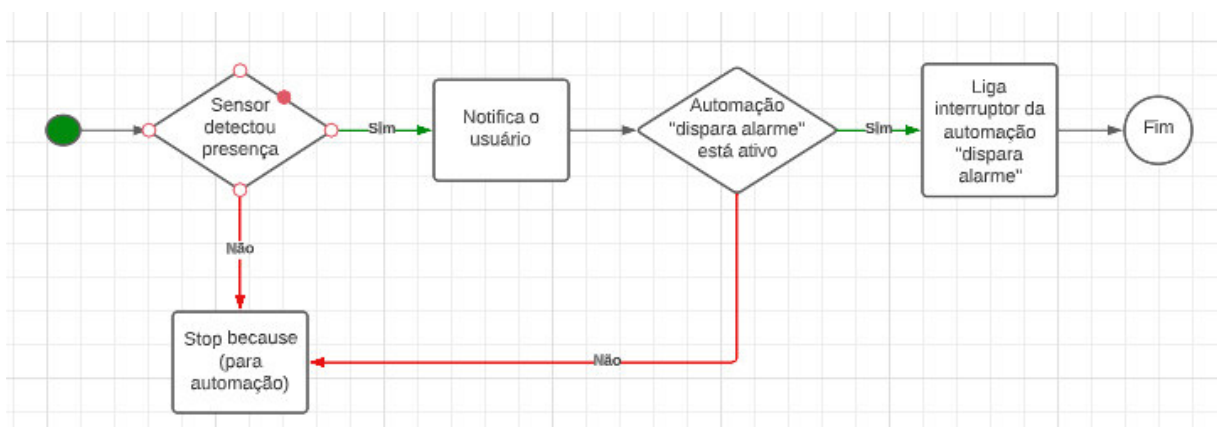
4.2.2 FLUXOGRAMA

Para realização do sistema de segurança foi dividido a em três automações, na qual a automação de disparar o alarme irá trabalhar em conjunto, com dois possíveis cenários:

- Automação alarme ativo;
- Automação de vigilância (quando estiver fora de casa);
- Automação de vigilância (quando estiver em casa);

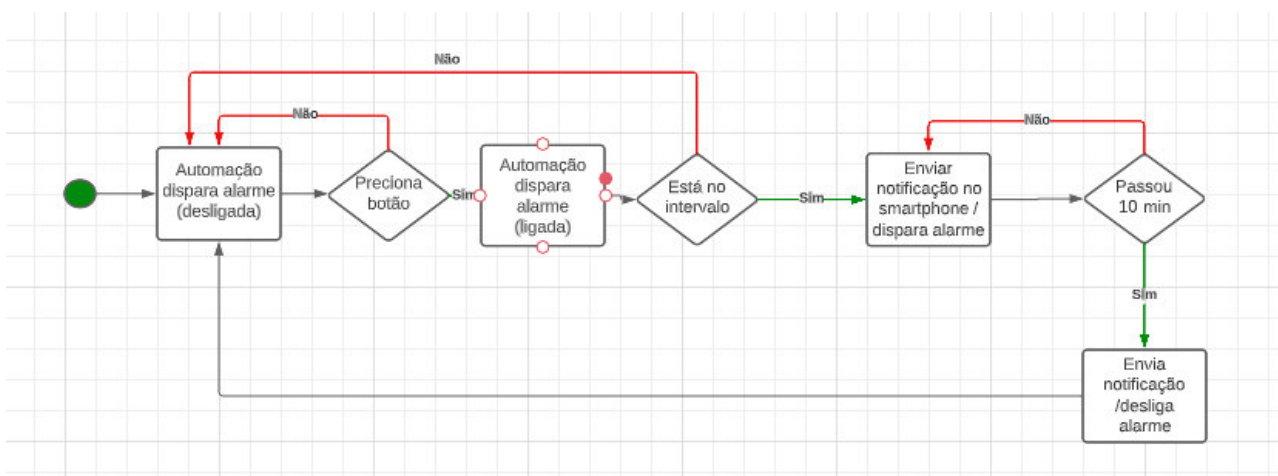
O primeiro cenário será caso o usuário não esteja em casa e o sensor detecta alguma presença, representado a automação de vigilância (quando estiver fora de casa) mostrado na Figura 19, e na Figura 20 mostra o funcionamento da automação dispara alarme.

Figura 19 – Fluxograma sistema de segurança casa vazia



Fonte: Autoria Própria, 2023

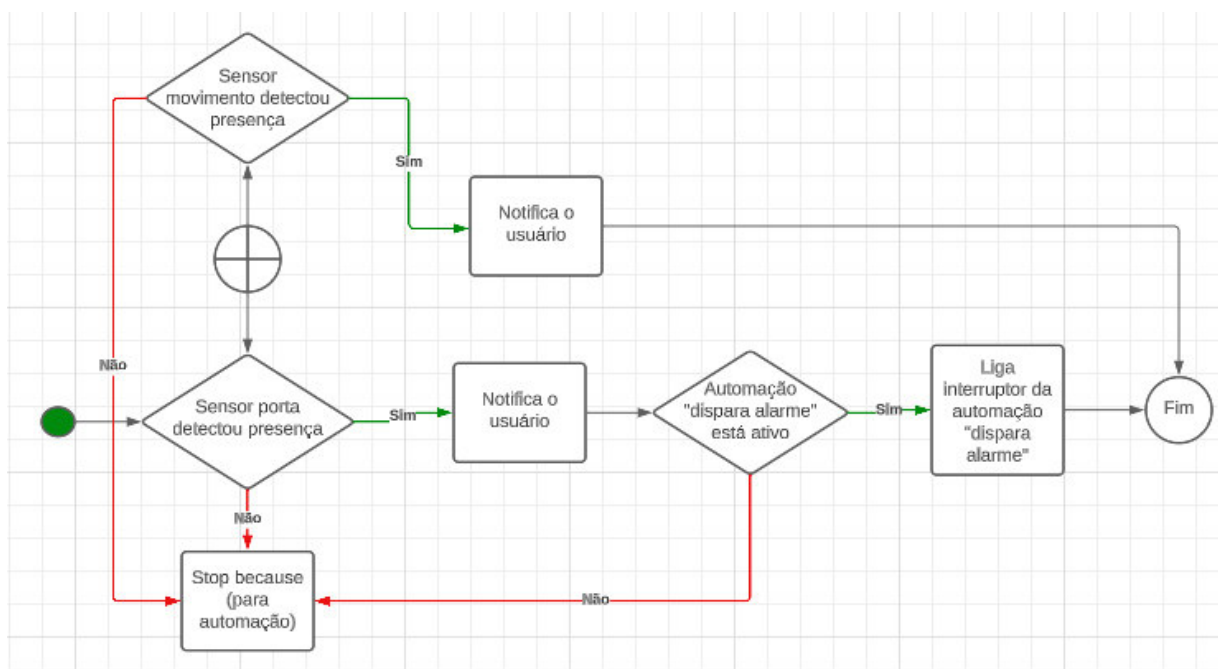
Figura 20 – Fluxograma automação dispara alarme



Fonte: Autoria Própria, 2023.

A automação de vigilância (quando estiver em casa), representado na Figura 21, estará sendo representada o segundo cenário caso o usuário esteja em casa e para o alarme não fica disparando toda vez que o sensor detectar algo, os sensores foram divididos em classes diferentes.

Figura 21 – Fluxograma sistema de segurança casa cheia



Fonte: Autoria Própria, 2023.

5 TESTES E RESULTADOS

Os testes são essenciais porque ajudam a avaliar a eficácia das respostas recebidas comparando-as com os resultados esperados. Testes práticos em bancada foram realizados para determinar a eficácia da operação e validar a solução proposta para o problema. Primeriamente foi testada a automação de disparar o alarme, isoladamente, depois as outras duas automações que trabalharão em conjunto com a de disparar o alarme.

5.1 EXPERIMENTO 1 – AUTOMAÇÃO “ALARME – ATIVO”

Essa automação é responsável por disparar o alarme e notificar o usuário que o alarme foi disparado, para isso foi realizada a criação de uma entrada virtual, um switch, que foi implementado para ligar e desligar o alarme, ou seja, o alarme só irá disparar caso essa entrada virtual esteja ligada, assim como mostra na Figura 22.

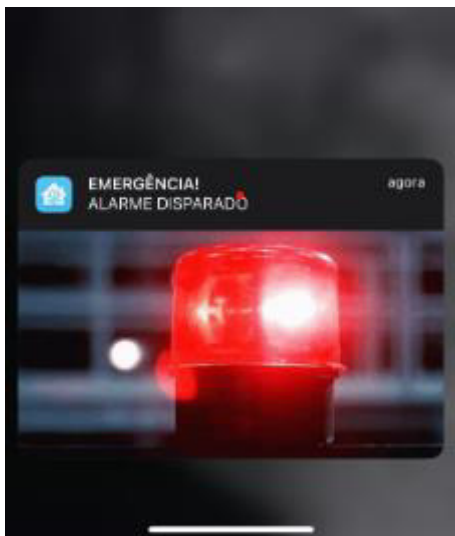
Figura 22 – Chave na interface do sistema



Fonte: Autoria Própria, 2023.

Sempre que o *switch* era ligado, a automação entra em uma condição de tempo, caso esteja no tempo estabelecido uma notificação era enviada ao *smartphone* que operava em uma rede 3G e tinha o aplicativo em execução no modo minimizado, conforme ilustra a Figura 23. Mesmo quando o telefone estava em espera, todas as notificações chegavam sem erros com atraso de um a dois segundos, com o som selecionado tocando e a informação que o alarme estava ativo.

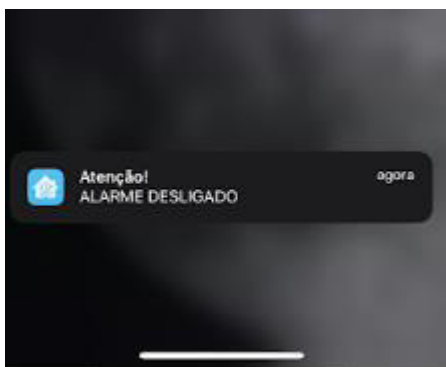
Figura 23 – Notificação de ativação do alarme



Fonte: Autoria própria, 2023.

Com decorrer do tempo estabelecido na implementação, o alarme era desligado e novamente uma notificação era enviada ao *smartphone*, assim como mostra na Figura 24.

Figura 24 – Notificação de desativação do alarme



Fonte: Autoria Própria, 2023.

5.2 EXPERIMENTO 2 – AUTOMAÇÃO “Vigilância (casa vazia)”

Dependente do Experimento 1, para acionar o alarme, também foi criada uma entrada virtual, uma chave no lugar do sensor, mostrada na Figura 25, para descartar possíveis erros nos sensores. A chave simulava o sensor de presença, seu acionamento foi realizado repetidamente 12 vezes.

Figura 25 – Chave representado um sensor

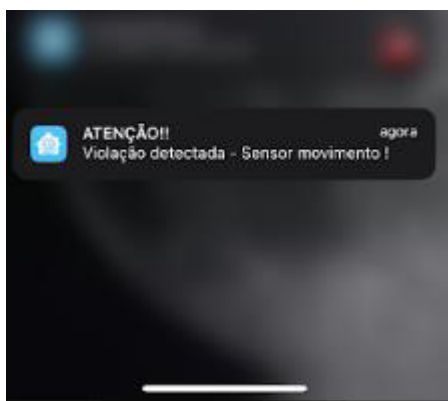


Fonte: Autoria Própria, 2023.

Sempre que a chave é acionada, a automação informava através de uma notificação, mostrado na Figura 25, e entrava em uma condição para verificar se a automação “Alarme ativo” estava habilitada, caso estivesse então o *switch* “Alarme ativo” (Figura 18) era acionado, para realizar o disparo do alarme, mostrado na figura 26. Caso contrário a automação para e nada mais acontece.

Implementação da condição está na linha 73 ao 83 do apêndice B.

Figura 25 – Notificação sensor detectou movimento



Fonte: Autoria Própria, 2023.

5.3 EXPERIMENTO 3 – AUTOMAÇÃO “Vigilância (casa cheia)”

A automação “vigilância (casa cheia)” trabalha tbm em conjunto com a automação do experimento 1, porém tem uma particularidade na implementação dessa automação, os sensores foram separados em categorias diferentes, sendo assim apenas alguns sensores caso detecte algo possa disparar o alarme, isso foi feito para que quando alguém estiver em casa e ligue o sistema de vigilância o alarme não fique disparando a todo momento.

Assim mostrado na linha 94 e 99 do apêndice B, foi estabelecido o ID de cada sensor, dessa forma, caso o sensor da categoria “MOVIMENTO” detecte algo a automação apenas irá enviar a notificação ao smartphone, logo o alarme irá acionar

apenas se o sensor da categoria “TRAVA” for acionado.

Caso não entre nessa condição a automação apenas irá para por estar no modo casa, assim como mostra na linha 122 do apêndice B.

6 CONCLUSÃO

Este projeto abordou a importância de um sistema de automação residencial com ênfase em segurança. Foi constatado que apenas 3% das residências no Brasil possuíam sistemas automatizados em 2015, de acordo com a Aureside.

Embora atualmente haja poucas residências com automação, espera-se que esse número aumente com o tempo devido ao conforto, valorização ambiental e segurança proporcionados por esses sistemas. O projeto demonstrou que é possível automatizar tarefas domésticas específicas usando microcontroladores, como o Raspberry Pi, o que beneficia especialmente pessoas com mobilidade reduzida.

Durante a execução do projeto, foram enfrentadas dificuldades na aquisição de equipamentos, como o módulo sensor, e na integração dos dispositivos ao GPIO do Raspberry Pi.

No entanto, esses obstáculos foram superados com esforço e busca por soluções alternativas. Em conclusão, o projeto alcançou seus objetivos, proporcionando uma solução viável para a automação residencial e seus benefícios relacionados ao conforto e segurança.

Para trabalhos futuros sugere-se que visem automatizar outras tarefas explorando ainda mais o potencial da tecnologia. Além disso, é possível buscar a integração com assistentes inteligentes como a Alexa para deixar o sistema ainda mais interativo e amigável ao usuário.

A integração do assistente inteligente permite que os usuários gerenciem e controlem tarefas automatizadas usando comandos de voz, resultando em uma experiência mais conveniente e eficaz.

A utilização de dispositivos Sonoff, como o relé WiFi Sonoff, que funciona como um interruptor controlado por um microcontrolador conectado à Internet, é outra sugestão intrigante para trabalhos futuros. Esses dispositivos permitem uma ampla gama de opções de automação residencial, incluindo controle de dispositivos eletrônicos e sistemas elétricos.

REFERÊNCIAS BIBLIOGRÁFICA

ACCARDI, Adonis; DODONOV, Eugeni. Automação residencial: elementos básicos, arquiteturas, setores, aplicações e protocolos. Revista TIS, v. 1, n. 2, 2012.

AGUIRRE, Luis antonio. Enciclopédia de Automática: Controle & Automação. São Paulo; Blucher, 2007. V.1.

ANGEL, P. M. Introducción a la domótica; Domótica: controle e automação. Escuela Brasileño-Argentina de Informática. EBAI. 1993

ASSISTANT, H. Hass.io. [S.l.], 2019. Disponível em: <https://www.home-assistant.io/getting-started>.

ASSISTANT, H. Arquitetura do Hass.io. [S.l.], 2019. Disponível em: https://developers.home-assistant.io/docs/en/architecture_hassio.html

Araújo, I. B. Q.; Souto, F. V.; Junior, A. G. C. Desenvolvimento de um protótipo de automação predial utilizando a plataforma de prototipagem eletrônica Arduino. In: Congresso Brasileiro de Educação em Engenharia (COBENGE), 2018. p. 1-9.

BOLZANI, C.A.M. Desenvolvimento de um simulador de controle de dispositivos residenciais inteligentes: uma introdução aos sistemas domóticos. São Paulo, Dissertação (mestrado), Universidade de São Paulo. 2004.^a

CABRAL, Michel Madson Alves; DE SIQUEIRA CAMPOS, Antonio Luiz Pereira. Sistemas de automação residencial de baixo custo: uma realidade possível. HOLOS, v. 3, p. 26-32, 2008.

EVANS, D. The Internet of Things: How the Next Evolution of the Internet Is Changing Everything. In:

Monk, S. (2013). Programação com Arduino >>começando com sketches. Porto

Alegre, RS: BOOKMAN EDITORA LTDA. MONK, S. Movimento, luz e som com Arduino e Raspberry Pi. [S.l.]: Novatec Editora, 2016.

MUSSIO, L. D.; MAIA, R. F.; LOPES, G. W. Um Estudo de Arquiteturas IoT para Dispositivos Embarcados. Sao Bernardo do Campo, v. 5, 2015.

NEVES, J. P. Uma abordagem para gerenciamento de fluxo de dados da Internet das Coisas. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal do Parana, Curitiba, 2017.

NUNES, Renato Jorge Caleira. Arquitectura de um Controlador Domótico Flexível e de Baixo Custo. INESC-ID. Lisboa – Portugal, 2002.

RASPBERRY PI FOUNDATION. raspberry pi, 2018. Decomratização de Tecnologia. Disponível em. Acesso em: 30 de nov. de 2021.

SANTOS, B. P. et al. Internet das Coisas: da Teoria à Prática. 2016. Universidade Federal de Minas Gerais, Belo Horizonte.

SILVA, Mauricio César; GAMBARATO, Vivian Toledo Santos. Domótica e tecnologias utilizadas na automação residencial. Tekhne e Logos, v. 7, n. 2, p. 56-67, 2016.

SOUSA, Leandro Magno Gomides de et al. Modelagem e compensação de erro de sensores e atuadores baseados em arduino. 2018.

TAKIUCHI, M., MELO, É., & TONIDANDEL, F. (2004). DOMÓTICA INTELIGENTE: AUTOMAÇÃO BASEADA EM COMPORTAMENTO. Centro Universitário da FEI – UniFE, São Bernardo do Campo – SP.

WANZELER, THIAGO (2016). DESENVOLVIMENTO DE UM SISTEMA DE AUTOMAÇÃO RESIDENCIAL DE BAIXO CUSTO ALIADO AO CONCEITO DE INTERNET DAS COISAS (IOT). XXXIV simpósio brasileiro de telecomunicações – sbRt2016.

WEISER, M. The computer for the 21st century. In: 3. v. 265, p. 94–105.

APÊNDICE A: CÓDIGO DO ARQUIVO CONFIGURATION.YAML

```
1  ~
2  # Loads default set of integrations. Do not remove.
3  default_config: ~
4  ~
5  # Load frontend themes from the themes folder
6  frontend: ~
7  - themes: !include_dir_merge_named themes
8  ~
9  # Text to speech
10 tts: ~
11 - - platform: google_translate
12 ~
13 automation: !include automations.yaml
14 script: !include scripts.yaml
15 scene: !include scenes.yaml
16 ~
17 binary_sensor: ~
18 - - platform: remote_rpi_gpio
19   - host: 192.168.0.123
20   - ports: ~
21     - 11: PIR Sensor
22     - ~
23 switch: ~
24 - - platform: remote_rpi_gpio
25   - host: 192.168.0.123
26   - ports: ~
27     - 17: buzzer
```

APÊNDICE B: CÓDIGO DO ARQUIVO AUTOMATIONS.YAML

```

1 | id: '1685304548103'
2 | alias: Alarme - ativo automação ..... # Nome da automação
3 | description: automação que dispara o alarme
4 | trigger: ..... # Gatilho que dispara a automação
5 | - platform: state
6 |   entity_id:
7 |     - input_boolean.alarme ..... # Entrada virtual
8 |     to: 'on'
9 | condition: []
10 | action: ..... # Ação realizada após o gatilho
11 | - if:
12 |   - condition: time
13 |     after: '17:00:00'
14 |     before: '16:59:59'
15 |     weekday:
16 |       - sun
17 |       - mon
18 |       - tue
19 |       - wed
20 |       - thu
21 |       - fri
22 |       - sat
23 |     then:
24 |       - service: automation.turn_on
25 |         entity_id: switch.sirene
26 |         alias: Acionar sirene
27 |       - service: notify.mobile_app_pedro_pia
28 |         data:
29 |           message: ALARME DISPARADO
30 |           title: EMERGÊNCIA!
31 |           data:
32 |             subtitle: ''
33 |           push:
34 |             sound: US-EN-Alexa-Motion-At-Front-Door.wav

```

```

35 ▾ ..... attachment:~
36 ..... url: https://tenor.com/pt-BR/view/police-siren-siren-gif-14993722.gif~
37 ..... content-type: gif~
38 ▾ ..... - delay:~
39 ..... hours: 0~
40 ..... minutes: 0~
41 ..... seconds: 15~
42 ..... milliseconds: 0~
43 ▾ ..... - service: automation.turn_off~
44 ▾ ..... data:~
45 ..... stop_actions: true~
46 ..... alias: Desligar sirene~
47 ..... entity_id: switch.sirene~
48 ▾ ..... - service: notify.mobile_app_pedro_pia~
49 ▾ ..... data:~
50 ..... message: ALARME DESLIGADO~
51 ..... title: Atenção!~
52 ▾ ..... - service: input_boolean.turn_off~
53 ..... data: {}~
54 ▾ ..... target:~
55 ..... entity_id: input_boolean.alarme~
56 ..... mode: single~
57 .....
58 ▾ - id: '1685305866006'~
59 ..... alias: Vigilância ..... # Nome da automação~
60 ..... description: Sistema de vigilância casa vazia~
61 ..... trigger: ..... # Gatilho que dispara a automação~
62 ▾ ..... - platform: state~
63 ..... entity_id:~
64 ..... - input_boolean.sensor~
65 ..... to: 'on'~
66 ..... condition: []~

```

```

67 --action:..... # Ação realizada após o gatilho
68 ▾ --service: notify.mobile_app_pedro_pia
69 ▾ --data:
70 ▾ --message: Violação detectada -- {{ state_attr(trigger.entity_id, 'friendly_name')}}
71 ▾ --!
72 ▾ --title: ATENÇÃO!!
73 ▾ --if:
74 ▾ --condition: state
75 ▾ --entity_id: automation.automation_1
76 ▾ --state: 'on'
77 ▾ --then:
78 ▾ --service: input_boolean.turn_on
79 ▾ --data: {}
80 ▾ --target:
81 ▾ --entity_id: input_boolean.alarme
82 ▾ --else:
83 ▾ --stop: Automações de disparar alarme está desligadas
84 --mode: single
85 --
86 ▾ --id: '1685424610772'
87 --alias: Vigilância (CASA)..... # Nome da automação
88 --description: 'Sistema de vigilância quando está em casa'
89 --trigger:..... # Gatilho que dispara a automação
90 ▾ --platform: state
91 ▾ --entity_id:
92 ▾ --input_boolean.sensor
93 ▾ --to: 'on'
94 ▾ --id: MOVIMENTO
95 ▾ --platform: state
96 ▾ --entity_id:
97 ▾ --input_boolean.sensor_porta
98 ▾ --to: 'on'
99 ▾ --id: TRAVA

100 --condition: []
101 --action:..... # Ação realizada após o gatilho
102 ▾ --service: notify.mobile_app_pedro_pia
103 ▾ --data:
104 ▾ --message: Violação detectada -- {{ state_attr(trigger.entity_id, 'friendly_name')}}
105 ▾ --!
106 ▾ --if:
107 ▾ --condition: trigger
108 ▾ --id: TRAVA
109 ▾ --then:
110 ▾ --if:
111 ▾ --condition: state
112 ▾ --entity_id: automation.automation_1
113 ▾ --state: 'on'
114 ▾ --then:
115 ▾ --service: input_boolean.turn_on
116 ▾ --data: {}
117 ▾ --target:
118 ▾ --entity_id: input_boolean.alarme
119 ▾ --else:
120 ▾ --stop: Automações de disparar alarme está desligadas
121 ▾ --else:
122 ▾ --stop: Modo Casa
123 --mode: single
124

```



PONTIFÍCIA UNIVERSIDADE CATÓLICA DE GOIÁS
GABINETE DO REITOR

Av. Universitária, 1069 • Setor Universitário
Caixa Postal 86 • CEP 74605-010
Goiânia • Goiás • Brasil
Fone: (62) 3946.1000
www.pucgoias.edu.br • reitoria@pucgoias.edu.br

RESOLUÇÃO nº 038/2020 – CEPE

ANEXO I

APÊNDICE ao TCC

Termo de autorização de publicação de produção acadêmica

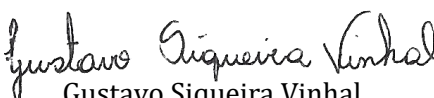
O(A) estudante **Pedro Gabriel Reis Pires** do Curso de **Engenharia de Computação**, matrícula **2017.2.0033.0091-1**, telefone: **(62) 99800-3589** e-mail **20172003300911@pucgo.edu.br**, na qualidade de titular dos direitos autorais, em consonância com a Lei nº 9.610/98 (Lei dos Direitos do Autor), autoriza a Pontifícia Universidade Católica de Goiás (PUC Goiás) a disponibilizar o Trabalho de Conclusão de Curso intitulado **Automação Residencial: Sistema de Segurança Utilizando Raspberry**, gratuitamente, sem ressarcimento dos direitos autorais, por 5 (cinco) anos, conforme permissões do documento, em meio eletrônico, na rede mundial de computadores, no formato especificado (Texto(PDF); Imagem (GIF ou JPEG); Som (WAVE, MPEG, AIFF, SND); Vídeo (MPEG, MWV, AVI, QT); outros, específicos da área; para fins de leitura e/ou impressão pela internet, a título de divulgação da produção científica gerada nos cursos de graduação da PUC Goiás.

Goiânia, 20 de junho de 2023.

Assinatura do autor:
Nome completo do autor:


Pedro Gabriel Reis Pires

Assinatura do professor – orientador:
Nome completo do professor – orientador:


Gustavo Siqueira Vinhal