



UNIVERSIDADE FEDERAL DO MARANHÃO  
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIAS  
ENGENHARIA DA COMPUTAÇÃO

ANDRÉ VICTOR MACEDO PEREIRA

AUTOMAÇÃO RESIDENCIAL COM CONTROLE POR VOZ E  
TECNOLOGIAS DE CÓDIGO ABERTO

SÃO LUÍS  
2026

ANDRÉ VICTOR MACEDO PEREIRA

AUTOMAÇÃO RESIDENCIAL COM CONTROLE POR VOZ E  
TECNOLOGIAS DE CÓDIGO ABERTO

Trabalho de conclusão de curso II  
apresentado ao curso de  
Bacharelado em Engenharia da  
Computação da Universidade  
Federal do Maranhão, como parte  
dos requisitos necessários para  
obtenção do grau de Bacharel em  
Engenharia da Computação

Orientador: Prof. Dr. Pedro Baptista  
Fernandes

SÃO LUÍS  
2026

**Ficha gerada por meio do SIGAA/Biblioteca com dados fornecidos  
pelo(a) autor(a).  
Diretoria Integrada de Bibliotecas/UFMA**

Macedo Pereira, André Victor.

AUTOMAÇÃO RESIDENCIAL COM CONTROLE POR VOZ E TECNOLOGIAS DE CÓDIGO ABERTO  
/ André Victor Macedo Pereira. - 2026.

71 p.

Orientador(a): Pedro Baptista Fernandes.

Curso de Engenharia da Computação, Universidade Federal do Maranhão, São  
Luís, 2026.

1. Automação Residencial. 2. Internet das Coisas. 3.Home Assistant. 4.  
Node-red. 5. Alexa. I. Baptista Fernandes, Pedro. II. Título.

ANDRÉ VICTOR MACEDO PEREIRA

AUTOMAÇÃO RESIDENCIAL COM CONTROLE POR VOZ E  
TECNOLOGIAS DE CÓDIGO ABERTO

Trabalho de conclusão de curso II  
apresentado ao curso de  
Bacharelado em Engenharia da  
Computação da Universidade  
Federal do Maranhão, como parte  
dos requisitos necessários para  
obtenção do grau de Bacharel em  
Engenharia da Computação

Orientador: Prof. Dr. Pedro Baptista  
Fernandes

BANCA EXAMINADORA

**Prof. Dr. Pedro Baptista Fernandes (Orientador)**

Doutor em Engenharia Elétrica - UFPA

**Prof. Dr. Denis Fabricio Sousa de Sá**

Doutor em Engenharia Elétrica - UFMA

**Prof. Dr. Haroldo Gomes Barroso Filho**

Doutor em Engenharia Elétrica - UFPA

## **AGRADECIMENTOS**

Agradeço primeiramente a Deus, pela força, sabedoria e perseverança concedidas ao longo desta jornada acadêmica.

Aos meus pais, por todo o amor, apoio, incentivo e dedicação ao longo da minha vida, sendo fundamentais para que eu pudesse alcançar mais esta conquista. Sou profundamente grato por todos os ensinamentos, valores e esforços realizados em prol da minha formação pessoal e profissional.

Ao meu professor orientador, pela orientação, disponibilidade, paciência e pelos conhecimentos compartilhados durante o desenvolvimento deste trabalho. Sua contribuição foi essencial para a realização desta pesquisa e para o meu crescimento acadêmico.

Aos meus amigos e colegas, pela convivência, companheirismo, apoio mútuo e pelos momentos compartilhados ao longo da graduação. As experiências vividas e os desafios superados em conjunto tornaram essa trajetória mais enriquecedora e significativa.

Por fim, agradeço a todos que, direta ou indiretamente, contribuíram para a realização deste trabalho e para a conclusão desta importante etapa da minha formação.

*"Para mim, o computador é a mais extraordinária ferramenta que já tivemos. É o equivalente à bicicleta para nossa mente".*

(Steve Jobs)

## RESUMO

A evolução da Internet das Coisas tem impulsionado o desenvolvimento de soluções voltadas à automação residencial, proporcionando maior conforto, segurança, eficiência energética e acessibilidade aos usuários. Nesse contexto, este trabalho apresenta o desenvolvimento e a implementação de um sistema de automação residencial inteligente baseado em tecnologias de código aberto, utilizando *Raspberry Pi 4* como servidor central, *Home Assistant* como plataforma de gerenciamento, *Node-RED* para criação de automações e *Amazon Alexa* como interface de interação por voz. A metodologia adotada consistiu na implementação de uma arquitetura integrada capaz de centralizar o gerenciamento de dispositivos inteligentes, sensores ambientais, equipamentos multimídia e serviços de monitoramento em uma única plataforma. Para viabilizar o acesso remoto seguro, foi utilizado o *Cloudflare Tunnel*, enquanto a execução dos serviços ocorreu por meio de contêineres *Docker*. Os resultados obtidos demonstraram a viabilidade técnica da solução proposta, evidenciando a integração bem-sucedida entre os diferentes componentes do sistema. As automações implementadas permitiram o controle inteligente de dispositivos, monitoramento ambiental, gerenciamento de equipamentos multimídia, acompanhamento da infraestrutura computacional e interação por comandos de voz. Além disso, o sistema apresentou estabilidade operacional durante a execução contínua e utilização adequada dos recursos computacionais disponíveis no *Raspberry Pi 4*. Conclui-se que a utilização de plataformas de código aberto associadas a dispositivos de baixo custo constitui uma alternativa viável para implementação de ambientes residenciais inteligentes, oferecendo flexibilidade, escalabilidade e redução de custos quando comparada a soluções proprietárias.

**Palavras-chave:** automação residencial. Internet das Coisas. *Home Assistant*. *Node-RED*. *Raspberry Pi*. *Alexa*. domótica.

## **ABSTRACT**

The advancement of the Internet of Things has driven the development of solutions focused on home automation, providing greater comfort, security, energy efficiency, and accessibility for users. In this context, this work presents the development and implementation of a smart home automation system based on open-source technologies, using a Raspberry Pi 4 as the central server, Home Assistant as the management platform, Node-RED for automation creation, and Amazon Alexa as the voice interaction interface. The adopted methodology consisted of implementing an integrated architecture capable of centralizing the management of smart devices, environmental sensors, multimedia equipment, and monitoring services within a single platform. To enable secure remote access, Cloudflare Tunnel was used, while service execution was carried out through Docker containers. The obtained results demonstrated the technical feasibility of the proposed solution, highlighting the successful integration among the different system components. The implemented automations enabled intelligent device control, environmental monitoring, multimedia equipment management, computational infrastructure supervision, and voice-command interaction. Furthermore, the system exhibited operational stability during continuous execution and efficient use of the computational resources available on the Raspberry Pi 4. It is concluded that the use of open-source platforms combined with low-cost devices represents a viable alternative for implementing smart residential environments, offering flexibility, scalability, and cost reduction when compared to proprietary solutions.

**Keywords:** home automation. Internet of Things. Home Assistant. Node-RED. Raspberry Pi. Alexa. smart home.

## LISTA DE FIGURAS

|                                                                                                 |    |
|-------------------------------------------------------------------------------------------------|----|
| Figura 1 – Placa Raspberry Pi 4 e dispositivos inteligentes.....                                | 18 |
| Figura 2 – Interface web do Home Assistant.....                                                 | 20 |
| Figura 3 – Interface web do Node-RED.....                                                       | 22 |
| Figura 4 – Amazon Echo Dot de 3ª geração. ....                                                  | 23 |
| Figura 5 – Esquema representativo da arquitetura em camadas.....                                | 30 |
| Figura 6 – Interface Raspberry Pi Imager.....                                                   | 32 |
| Figura 7 – Configuração dos IPs do servidor.....                                                | 33 |
| Figura 8 – Conexão do túnel na Cloudflare.....                                                  | 34 |
| Figura 9 – Rotas do <i>Node-red</i> e <i>Home Assistant</i> no túnel da <i>Cloudflare</i> ..... | 34 |
| Figura 10 – Interface CasaOS.....                                                               | 35 |
| Figura 11 – Painel de integrações do Home Assistant.....                                        | 38 |
| Figura 12 – Instalação da extensão Home Assistant WebSocket.....                                | 40 |
| Figura 13 – Configuração da extensão Home Assistant WebSocket.....                              | 40 |
| Figura 14 – Integração Alexa Devices no Home Assistant.....                                     | 42 |
| Figura 15 – Interface do HACS no Home Assistant.....                                            | 43 |
| Figura 16 – Integração com o Tuya Local no Home Assistant.....                                  | 44 |
| Figura 17 – Integração com o Tapo: Cameras Control.....                                         | 45 |
| Figura 18 – Dashboard com cartões do HACS.....                                                  | 46 |
| Figura 19 – Fluxo no Node-RED para controle termostático inteligente.....                       | 47 |
| Figura 20 – Fluxo no Node-RED para monitoramento do ambiente.....                               | 48 |
| Figura 21 – Fluxo no Node-RED para controle inteligente de iluminação.....                      | 49 |
| Figura 22 – Fluxo no Node-RED do modo cinema.....                                               | 50 |
| Figura 23 – Fluxo no Node-RED para desligamento automático de televisão.....                    | 50 |
| Figura 24 – Fluxo no Node-RED para monitoramento da velocidade da internet.....                 | 51 |
| Figura 25 – Fluxo no Node-RED para segurança residencial.....                                   | 52 |
| Figura 26 – Fluxo no Node-RED para monitoramento do Raspberry Pi 4.....                         | 52 |
| Figura 27 – Seção histórico do Home Assistant.....                                              | 54 |

## **LISTA DE TABELAS**

|                                                                                          |    |
|------------------------------------------------------------------------------------------|----|
| Tabela 1 – Desempenho computacional e térmico do Raspberry Pi 4.....                     | 55 |
| Tabela 2 – Consumo energético acumulado e perfil de gasto.....                           | 56 |
| Tabela 3 – Condições ambientais e desempenho da rede.....                                | 57 |
| Tabela 4 – Análise de custos para implementação do sistema de automação residencial..... | 60 |

## LISTA DE ABREVIATURAS, SIGLAS E SÍMBOLOS

API – Interface de Programação de Aplicações (Application Programming Interface)  
CPU – Unidade Central de Processamento (Central Processing Unit)  
DHCP – Protocolo de Configuração Dinâmica de Host (Dynamic Host Configuration Protocol)  
GB – Gigabyte  
GPIO – Entrada e Saída de Uso Geral (General Purpose Input/Output)  
GUI – Interface Gráfica de Usuário (Graphical User Interface)  
HACS – Loja Comunitária do Home Assistant (Home Assistant Community Store)  
IoT – Internet das Coisas (Internet of Things)  
IP – Protocolo de Internet (Internet Protocol)  
LAN – Rede Local (Local Area Network)  
MQTT – Transporte de Telemetria por Fila de Mensagens (Message Queuing Telemetry Transport)  
PLN – Processamento de Linguagem Natural  
RAM – Memória de Acesso Aleatório (Random Access Memory)  
RGB – Vermelho, Verde e Azul (Red, Green, Blue)  
SSH – Shell Seguro (Secure Shell)  
UPnP – Plug and Play Universal (Universal Plug and Play)  
USB – Barramento Serial Universal (Universal Serial Bus)  
VUI – Interface de Usuário por Voz (Voice User Interface)  
Wi-Fi – Fidelidade sem Fio (Wireless Fidelity)

### **Símbolos**

°C – Graus Celsius  
% – Porcentagem  
kWh – Quilowatt-hora (Kilowatt-hour)  
Mbps – Megabits por segundo (Megabits per second)  
ms – Milissegundos (Milliseconds)  
R\$ – Real, moeda brasileira  
W – Watt

## SUMÁRIO

|                                                                         |           |
|-------------------------------------------------------------------------|-----------|
| <b>1 INTRODUÇÃO.....</b>                                                | <b>13</b> |
| 1.1 Objetivos.....                                                      | 14        |
| 1.1.1 Geral.....                                                        | 14        |
| 1.1.2 Específicos.....                                                  | 14        |
| 1.2 Justificativa.....                                                  | 15        |
| <b>2 FUNDAMENTAÇÃO TEÓRICA.....</b>                                     | <b>16</b> |
| 2.1 Domótica/Automação residencial.....                                 | 16        |
| 2.2 Hardware: Raspberry Pi 4 e dispositivos inteligentes.....           | 17        |
| 2.3 Home Assistant e Docker.....                                        | 19        |
| 2.4 Node-RED.....                                                       | 21        |
| 2.5 Assistentes virtuais e controle por voz.....                        | 22        |
| <b>3 METODOLOGIA.....</b>                                               | <b>25</b> |
| 3.1 Introdução.....                                                     | 25        |
| 3.2 Tipo de pesquisa.....                                               | 25        |
| 3.3 Desenho da pesquisa.....                                            | 25        |
| 3.4 População e amostra.....                                            | 26        |
| 3.5 Coleta de dados.....                                                | 26        |
| 3.6 Análise de dados.....                                               | 26        |
| 3.7 Limitações metodológicas.....                                       | 27        |
| <b>4 DESENVOLVIMENTO.....</b>                                           | <b>28</b> |
| 4.1 Arquitetura do sistema.....                                         | 28        |
| 4.1.1 Visão geral da arquitetura.....                                   | 28        |
| 4.1.2 Fluxo de comunicação e funcionamento do sistema de automação..... | 30        |
| 4.2 Configuração do servidor.....                                       | 31        |
| 4.2.1 Sistema operacional.....                                          | 31        |
| 4.2.2 Configuração de rede.....                                         | 33        |
| 4.2.3 Instalação do Docker.....                                         | 34        |
| 4.2.4 Instalação e configuração dos serviços.....                       | 36        |
| 4.3 Configuração das integrações.....                                   | 37        |
| 4.3.1 Integração de dispositivos inteligentes e Home Assistant.....     | 37        |
| 4.3.2 Integração do Node-RED e Home Assistant.....                      | 39        |
| 4.3.3 Integração da Amazon Alexa e Home Assistant.....                  | 41        |
| 4.3.4 Home Assistant Community Store (HACS).....                        | 42        |
| 4.4 Fluxos e automações.....                                            | 46        |
| 4.4.1 Controle termostático inteligente.....                            | 47        |
| 4.4.2 Monitoramento do ambiente.....                                    | 48        |
| 4.4.3 Controle inteligente de iluminação.....                           | 48        |
| 4.4.4 Modo cinema.....                                                  | 49        |
| 4.4.5 Desligamento automático de televisores.....                       | 50        |
| 4.4.6 Monitoramento da velocidade da internet.....                      | 51        |
| 4.4.7 Sistema de segurança residencial.....                             | 51        |

|                                                              |           |
|--------------------------------------------------------------|-----------|
| 4.4.8 Monitoramento do servidor de automação.....            | 52        |
| <b>5 RESULTADOS E DISCUSSÃO.....</b>                         | <b>54</b> |
| 5.1 Desempenho computacional e estabilidade do hardware..... | 54        |
| 5.2 Monitoramento energético.....                            | 56        |
| 5.3 Monitoramento ambiental e desempenho de rede.....        | 57        |
| 5.4 Viabilidade, acessibilidade e falhas observadas.....     | 58        |
| 5.5 Análise de custos.....                                   | 59        |
| 5.6 Discussão dos resultados.....                            | 62        |
| 5.6.1 Comparação com os objetivos propostos.....             | 62        |
| 5.6.2 Pontos fortes da arquitetura.....                      | 62        |
| 5.6.3 Limitações identificadas.....                          | 63        |
| 5.6.4 Comparação com a literatura.....                       | 63        |
| <b>6 CONCLUSÃO.....</b>                                      | <b>64</b> |
| <b>REFERÊNCIAS.....</b>                                      | <b>66</b> |

## 1 INTRODUÇÃO

A expansão da conectividade global e a contínua redução de tamanho de componentes eletrônicos consolidaram um novo paradigma tecnológico denominado Internet das Coisas, ou IoT (*Internet of Things*). Segundo Storck (2019), a IoT consiste na conexão de objetos do mundo físico à internet, dotando dispositivos cotidianos de capacidade de processamento, coleta de dados e comunicação em rede. Essa conexão entre os dispositivos tem o potencial de automatizar processos de forma autônoma e modificar profundamente a maneira como os seres humanos interagem com o ambiente ao seu redor (ANDRADE, 2025; COSTA, 2018). No cenário residencial, a aplicação prática dos conceitos de IoT culmina no desenvolvimento da domótica.

A domótica, termo que associa a palavra latina *domus* (casa) aos conceitos de robótica ou de automação, refere-se à integração de tecnologias de informação e eletrônica para o gerenciamento sistêmico de uma residência (COSTA, 2018; NOGUEIRA, 2025). O propósito fundamental dessas tecnologias é simplificar a vida diária das pessoas, promovendo a gestão eficiente do consumo de energia, o controle automatizado de temperatura e iluminação, e o reforço da segurança patrimonial (COSTA, 2018; REIS, 2021). Adicionalmente, os ambientes inteligentes desempenham um papel indispensável no âmbito da acessibilidade, conferindo maior autonomia e qualidade de vida para idosos e pessoas com deficiência por meio dos comandos de voz (ALMEIDA, 2022; STORCK, 2019).

Historicamente, entretanto, a automação residencial esbarrou em significativas barreiras socioeconômicas e estruturais. Grande parte das soluções do mercado é caracterizada por altos custos de aquisição e pela adoção de arquiteturas proprietárias e fechadas. Essa falta de padronização em protocolos e sistemas dificulta a interoperabilidade entre dispositivos de diferentes fabricantes, limitando a expansão modular do sistema e gerando dependência tecnológica para o usuário final (ALMEIDA, 2022; COSTA, 2018).

Como contraponto a esse cenário mercadológico restritivo, observa-se a ascensão de plataformas de código aberto e *hardwares* de baixo custo. Microcomputadores de dimensões reduzidas, como o *Raspberry Pi*, viabilizam a

implementação de servidores locais robustos e de baixo consumo energético para atuar como o cérebro da automação (REIS, 2021). No âmbito lógico, plataformas como o *Home Assistant* permitem centralizar o gerenciamento da residência de forma local, priorizando a privacidade dos dados, enquanto ferramentas visuais baseadas em *Node.js*, como o *Node-RED*, orquestram fluxos de comunicação assíncronos e complexos entre diferentes protocolos (COSTA, 2018; SOARES, 2020).

Apesar da disponibilidade isolada destas ferramentas computacionais, constata-se uma lacuna na literatura acadêmica e técnica referente à integração harmoniosa de todos esses componentes em uma arquitetura híbrida e unificada. Há uma notável escassez de trabalhos que documentam a estruturação de um ecossistema de baixo custo que incorpore, de forma simultânea e estável, um *hub* de automação executado via contêineres *Docker*, como o *Home Assistant*, a flexibilidade de motores de fluxo como o *Node-RED*, e a interatividade de interfaces conversacionais baseadas em Inteligência Artificial (*Amazon Alexa*). Diante deste contexto, o presente trabalho insere-se na proposta de desenvolver e avaliar um sistema integrado que orquestre essas tecnologias, visando superar os desafios de custo e interoperabilidade inerentes à domótica.

## 1.1 Objetivos

### 1.1.1 Geral

Desenvolver um sistema integrado de automação residencial baseado em *Home Assistant* executado em contêiner *Docker* em um *Raspberry Pi*, com controle por voz via *Amazon Alexa* e automações implementadas no *Node-RED*.

### 1.1.2 Específicos

- Implementar o sistema *Home Assistant* em ambiente isolado via contêiner *Docker* no *Raspberry Pi*, estabelecendo as integrações iniciais com a rede local;
- Integrar o servidor local à assistente virtual *Amazon Alexa*, viabilizando o controle de dispositivos através de comandos de voz;
- Desenvolver e estruturar fluxos de automação personalizados utilizando a ferramenta de programação visual *Node-RED*;

- Configurar o monitoramento e o controle centralizado de equipamentos eletrônicos pré-existentes no ambiente (*smart TVs*, câmeras de segurança e roteadores, lâmpada inteligente, sensor de temperatura e tomadas inteligentes);
- Avaliar o desempenho, a responsividade dos comandos e a estabilidade da arquitetura de comunicação em cenários reais de uso.

## 1.2 Justificativa

A realização deste projeto baseia-se na premissa de democratizar o acesso às tecnologias de automação residencial. O uso de hardwares acessíveis, como o *Raspberry Pi*, aliado à gratuidade de plataformas de código aberto, reduz consideravelmente o investimento financeiro exigido para a concepção de uma casa inteligente em comparação às soluções comerciais fechadas, permitindo que os benefícios da domótica alcancem diferentes estratos socioeconômicos.

Sob a perspectiva do impacto social, o projeto transcende a mera conveniência tecnológica e atua diretamente na promoção da inclusão e da acessibilidade. A integração de ecossistemas automatizados com assistentes virtuais habilitados por voz atua como uma robusta tecnologia assistiva (ALMEIDA, 2022; STORCK, 2019). O controle do ambiente físico, como acender luzes ou ligar aparelhos por meio da fala, suprime barreiras motoras, devolvendo a independência e a dignidade a usuários com mobilidade reduzida, deficiências físicas ou em idade avançada (ALMEIDA, 2022; REIS, 2021).

Por fim, no âmbito acadêmico, este trabalho justifica-se por fornecer uma base de pesquisa empírica e metodológica detalhada, servindo como modelo de replicação e aprimoramento para futuros projetos em Engenharia da Computação que busquem explorar a interoperabilidade de sistemas de código aberto na Internet das Coisas.

## 2 FUNDAMENTAÇÃO TEÓRICA

Esta seção dedica-se à explicação dos conceitos, tecnologias e protocolos que fundamentam a arquitetura do sistema proposto, estabelecendo a base literária necessária para a compreensão da integração entre hardware de baixo custo, plataformas de gerenciamento de código aberto e interfaces de assistência virtual.

### 2.1 Domótica/Automação residencial

A base da automação residencial, frequentemente denominada domótica, transcende a simples automatização de tarefas domésticas. De acordo com Bolzani (2004), o termo origina-se da aglutinação da palavra latina *domus* (casa) com o conceito de telemática, que engloba as telecomunicações e a informática. A literatura também aponta a sua relação etimológica com a robótica, sendo utilizada para definir a profunda integração entre o espaço arquitetônico e os recursos tecnológicos. Segundo Reis (2021), a automação residencial consiste na integração de dispositivos eletrônicos, sistemas computacionais e tecnologias de comunicação com o objetivo de monitorar e controlar recursos presentes no ambiente doméstico de forma automatizada.

Para que um ambiente seja verdadeiramente inteligente, a simples alocação de dispositivos eletrônicos não é suficiente. A concepção de projetos domóticos fundamenta-se na integração mandatória entre múltiplos sistemas de controle. Para viabilizar a usabilidade na rotina residencial, esses ecossistemas tecnológicos devem operar de forma transparente ao usuário, sendo gerenciados por meio de interfaces simplificadas e intuitivas.

O campo de atuação da automação residencial concentra-se no gerenciamento inteligente de recursos, na segurança, no conforto, na acessibilidade e na otimização das atividades cotidianas dos moradores (REIS, 2021; SGARBI; TONIDANDEL, 2006). Conforme apontam Sgarbi e Tonidandel (2006), a instalação de tecnologias residenciais visa, sobretudo, gerir o uso racional dos recursos disponíveis e aumentar a qualidade de vida. Neste contexto de gestão inteligente, a otimização e o monitoramento do consumo de energia elétrica assumem um papel de destaque, uma vez que a automação possibilita mitigar desperdícios, estimar

custos em tempo real e promover a sustentabilidade ambiental por meio da conscientização do uso (STEVAN JUNIOR; FARINELLI, 2018).

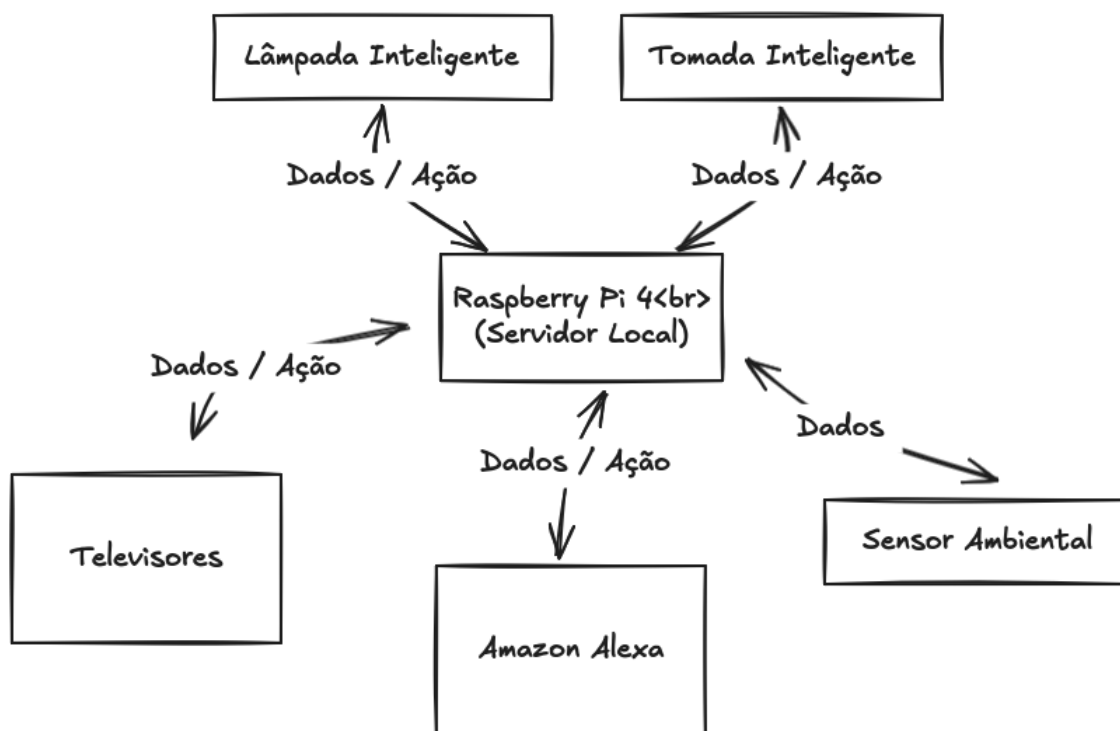
No que tange ao conforto e à segurança habitacional, Bolzani (2004) destaca que a domótica viabiliza a integração da climatização, da luminosidade, do som e dos alarmes a partir de um sistema unificado. Ambientes automatizados de alto desempenho tornam-se capazes de monitorar eventos cotidianos, detectar variações nas condições físicas do espaço e executar reações sistêmicas de forma autônoma, com base em lógicas previamente configuradas pelo usuário.

Por fim, é pertinente ressaltar a mudança de paradigma econômico no setor. Historicamente, o grau de inteligência e autonomia de uma residência era determinado estritamente pela alta disponibilidade financeira de seus habitantes (BOLZANI, 2004). Contudo, a evolução contínua das tecnologias de sensoriamento, aliada à disseminação de microcontroladores de baixo custo e com conectividade nativa, tem barateado consideravelmente o desenvolvimento de soluções domóticas (REIS, 2021). Essa democratização tecnológica indica que a automação residencial está deixando de ser uma exclusividade de classes privilegiadas economicamente, viabilizando a adoção de soluções escaláveis e modulares para o grande público.

## **2.2 Hardware: Raspberry Pi 4 e dispositivos inteligentes**

A implementação física de um ecossistema de automação residencial exige uma infraestrutura de *hardware* dividida hierarquicamente entre unidades centrais de processamento e dispositivos de borda. As unidades centrais atuam como servidores locais responsáveis por hospedar as plataformas de gerenciamento e orquestrar a comunicação da rede, enquanto os dispositivos de borda são alocados nas extremidades do sistema, interagindo diretamente com o ambiente físico por meio de sensores e atuadores (PORTELA, 2022; STORCK, 2019). Na Figura 1, pode ser observado o *Raspberry Pi 4* e seus principais componentes.

Figura 1 – Placa *Raspberry Pi 4* e dispositivos inteligentes.



Fonte: O autor (2026)

Para o processamento central, o microcomputador *Raspberry Pi 4* destaca-se na literatura como uma das soluções mais eficientes e versáteis. Reis (2021) define o *Raspberry Pi* como um computador de placa única (*single-board computer*) de dimensões compactas que apresenta um custo financeiro drasticamente reduzido em relação a *desktops* convencionais. Apesar do seu tamanho, o dispositivo possui uma arquitetura de hardware robusta o suficiente para suportar sistemas operacionais baseados em Linux, o que viabiliza a sua operação como um servidor ininterrupto (REIS, 2021). A utilização deste equipamento em projetos de domótica é justificada pela sua alta eficiência energética e pela presença de interfaces de comunicação sem fio nativas (adaptadores Wi-Fi e Bluetooth), além de pinos de Entrada e Saída de Uso Geral (GPIO), que são essenciais para a expansão e integração física com outros componentes eletrônicos (REIS, 2021; STORCK, 2019).

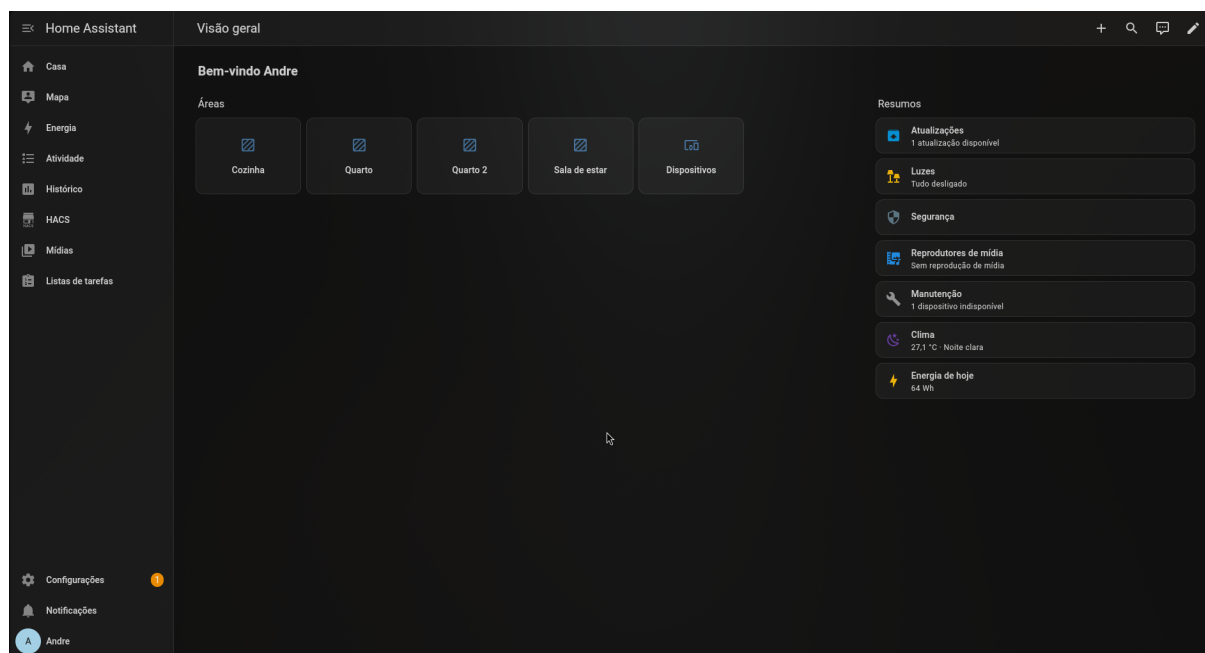
Na extremidade da rede, a função de percepção e atuação é desempenhada por dispositivos inteligentes comerciais, que incorporam internamente sensores, atuadores e conectividade de rede. Trabalhos recentes em automação residencial

ênfatizam o uso de lâmpadas inteligentes (*smart bulbs*), tomadas inteligentes (*smart plugs*) e sensores ambientais como alternativas prontas à prototipagem de *hardware*, permitindo integrar diretamente esses dispositivos a plataformas de gerenciamento como o *Home Assistant* (ALMEIDA, 2022; DE CAMARGOS et al., 2022; PORTELA, 2022).

Esses dispositivos comerciais geralmente comunicam-se via Wi-Fi e utilizam protocolos compatíveis com *brokers* ou APIs de nuvem, possibilitando a descoberta automática e controle remoto em ambientes de Internet das Coisas (DE CAMARGOS et al., 2022; PORTELA, 2022). A integração de *smart bulbs*, *smart plugs* e sensores a um servidor local baseado em Raspberry Pi, por meio do *Home Assistant*, viabiliza a construção de um ecossistema domótico de baixo custo, modular e expansível, sem a necessidade de desenvolvimento de placas ou circuitos dedicados, alinhando-se às abordagens descritas na literatura recente (PORTELA, 2022; REIS, 2021; STORCK, 2019).

### 2.3 Home Assistant e Docker

A eficácia de uma infraestrutura domótica depende intrinsecamente da capacidade de interoperabilidade entre equipamentos, sensores e atuadores de fabricantes distintos, que frequentemente utilizam protocolos de comunicação divergentes. Para consolidar esta comunicação heterogênea numa interface unificada, empregam-se plataformas centralizadoras. Neste domínio, o *Home Assistant* evidencia-se como um *software* de automação residencial de código aberto, desenvolvido na linguagem *Python*, cuja arquitetura foi especificamente concebida para atuar como o núcleo integrador e orquestrador de residências inteligentes (COSTA, 2018; SOARES, 2020). Na Figura 2, é possível observar a interface padrão do *Home Assistant*.

Figura 2 – Interface web do *Home Assistant*.

Fonte: O autor (2026).

Em contraponto às soluções comerciais dominantes do mercado, que operam predominantemente com base em infraestruturas de processamento na nuvem, o Home Assistant fundamenta-se no princípio do processamento local. Segundo Costa (2018) e Nogueira (2025), esta abordagem arquitetônica assegura que a coleta de dados, a análise e a execução das operações lógicas ocorram estritamente dentro da rede local (LAN) do usuário. Esta característica de descentralização da nuvem garante a privacidade das informações domésticas, além de reduzir a latência nos comandos de acionamento e garantir a continuidade operacional do sistema de automação mesmo em cenários de instabilidade ou ausência prolongada de conexão com a internet (COSTA, 2018).

Para garantir que o *Home Assistant*, juntamente com os seus serviços e integrações auxiliares, opere com a máxima estabilidade e eficiência em um *hardware* de recursos limitados como o *Raspberry Pi*, recorre-se ao uso de contêineres. O *Docker* atua como um motor de virtualização a nível do sistema operacional que permite encapsular aplicações e todas as suas respectivas dependências, bibliotecas e arquivos de configuração em ambientes lógicos estritamente isolados (ALMEIDA, 2022; NOGUEIRA, 2025).

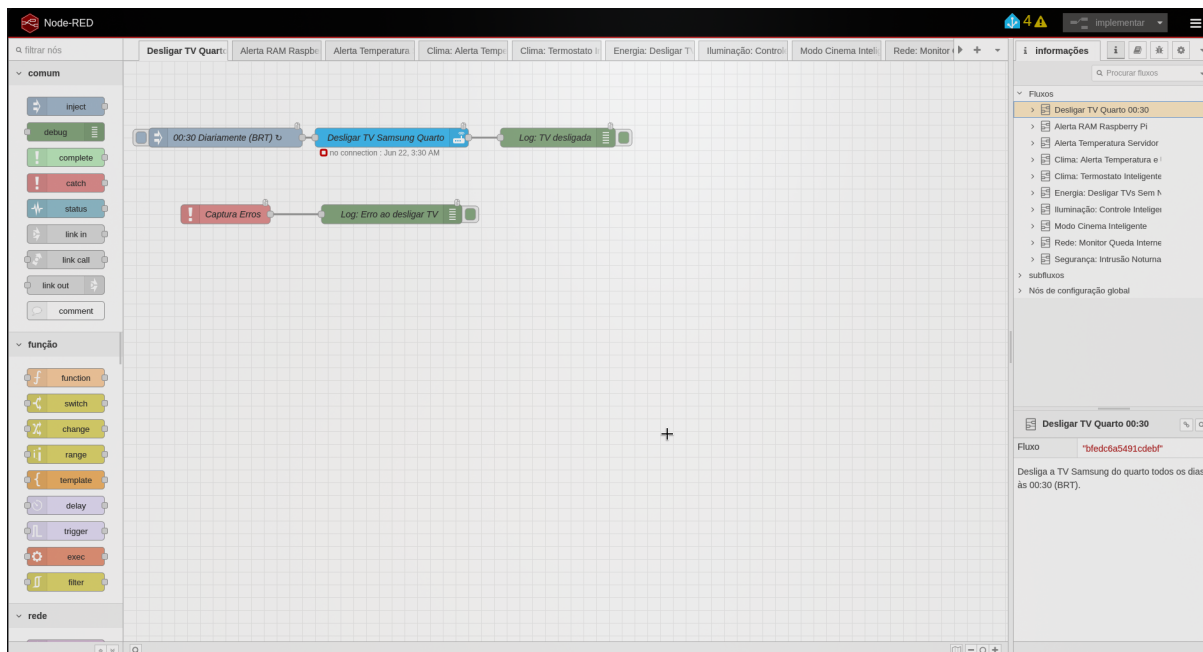
A adoção de um contêiner *Docker* abstrai a complexidade do sistema operacional hospedeiro, neste caso a distribuição Linux que opera no microcomputador (*Raspbian*), prevenindo de forma eficaz conflitos de compatibilidade de *software* e a degradação do sistema ao longo do tempo (ALMEIDA, 2022). Esta modularidade estrutural é um fator crítico para a escalabilidade da arquitetura proposta, uma vez que facilita a implementação paralela, a atualização segura e a orquestração de múltiplos nós de serviço, como o próprio *Home Assistant*, assegurando que a falha de um contêiner individual não comprometa a integridade e a resiliência global do servidor doméstico (ALMEIDA, 2022; NOGUEIRA, 2025).

## 2.4 Node-RED

A gestão de rotinas em uma residência inteligente frequentemente transcende a capacidade das lógicas condicionais simples. Embora plataformas centralizadoras, como o *Home Assistant*, possuam mecanismos nativos para a criação de automações, a orquestração de ecossistemas domóticos complexos demanda ferramentas mais visuais, modulares e flexíveis. Neste cenário, o *Node-RED* consolida-se como um ambiente de programação visual baseado em fluxo. Construída sobre o ambiente de execução *Node.js*, esta ferramenta foi concebida com o propósito primordial de interligar dispositivos de *hardware*, interfaces de programação de aplicações (APIs) e serviços *online* de maneira simplificada, ágil e escalável (COSTA, 2018; PORTELA, 2022).

A operacionalização do *Node-RED* ocorre por meio de uma interface gráfica de usuário intuitiva, acessada diretamente via navegador de *internet*. Neste ambiente de desenvolvimento, o programador dispõe de uma paleta contendo diversos blocos funcionais pré-configurados, tecnicamente denominados "nós" (*nodes*). A estruturação da lógica de controle do sistema ocorre mediante o arrasto e a interconexão, sequencial ou paralela, destes nós no espaço de trabalho, culminando na formação de fluxos (*flows*) de acionamento e comunicação (SOARES, 2020). Essa abordagem visual de interligação de nós reduz drasticamente a necessidade de escrita de linhas de código complexas, facilitando a prototipagem e a manutenção das automações residenciais. Na Figura 3 pode ser visualizada a interface *web* do *Node-RED*.

Figura 3 – Interface web do Node-RED.



Fonte: O autor (2026).

A ferramenta demonstra alta capacidade e estabilidade no tratamento de fluxos de dados assíncronos, um cenário típico em redes compostas por múltiplos dispositivos e serviços distribuídos operando simultaneamente, o que facilita o trânsito de informações não lineares e a coordenação de múltiplos eventos concorrentes (PORTELA, 2022; SOARES, 2020). Nesse contexto, o *Node-RED* pode ser empregado para receber e interpretar, em tempo quase real, mensagens de telemetria oriundas de diferentes nós de uma rede IoT e, a partir dessas entradas, disparar de forma imediata rotinas de automação ou ações corretivas sobre o ambiente físico, integrando-se de maneira transparente com plataformas centrais de gerenciamento como o *Home Assistant* (PORTELA, 2022; SOARES, 2020).

## 2.5 Assistentes virtuais e controle por voz

A evolução das interfaces de comunicação humano-computador revolucionou profundamente a forma como os usuários interagem com os sistemas de automação residencial. Historicamente, havia uma dependência de interfaces gráficas de usuário (GUI) centralizadas em aplicações de *smartphones*, porém as residências inteligentes caminham para a adoção de interfaces de usuário por voz (VUI),

possível pelos assistentes virtuais (CESÁRIO, 2022; STORCK, 2019). Esta transição tecnológica permite uma interação mais orgânica, intuitiva e natural com o ecossistema de Internet das Coisas (IoT), suprimindo a necessidade de comandos táteis ou manuais para o acionamento de rotinas físicas no ambiente doméstico (ALMEIDA, 2022).

Os assistentes virtuais, a exemplo da *Amazon Alexa*, operam com *smart speakers*, como os dispositivos da família *Amazon Echo* (como o modelo de entrada *Echo Dot*), que atuam como os receptores de áudio e emissores de resposta física no ambiente (ALMEIDA, 2022; SOARES, 2020). O funcionamento destes equipamentos baseia-se em um mecanismo de escuta local que aguarda passivamente a detecção de uma palavra de ativação. Uma vez acionado, o dispositivo capta a requisição vocal e transmite-a para servidores de alto desempenho na nuvem. Nestes servidores remotos, algoritmos complexos de inteligência artificial e Processamento de Linguagem Natural (PLN) transcrevem a fala para texto, extraem a intenção semântica do usuário e convertem essa formulação em um comando computacional executável (ALMEIDA, 2022; CESÁRIO, 2022). A *Amazon Echo Dot* de 3ª geração pode ser observada na Figura 4.

Figura 4 – *Amazon Echo Dot* de 3ª geração.



Fonte: AMAZON (2026).

No contexto de uma arquitetura domótica híbrida e integrada, os comandos processados pela nuvem da assistente virtual necessitam ser encaminhados para o servidor local da residência para que a ação física ocorra. A integração da *Amazon Alexa* com plataformas centralizadoras de gestão, como o *Home Assistant*, e com motores de orquestração de fluxos, como o *Node-RED*, viabiliza a ponte bidirecional entre a intenção vocal do usuário e a atuação no *hardware* local (SOARES, 2020). Esta interoperabilidade permite que a complexidade dos protocolos de comunicação se mantenha opaca e abstrata para o usuário final; basta verbalizar o comando para que o sistema interprete, valide e acione dispositivos inteligentes (ALMEIDA, 2022; PORTELA, 2022).

Para além de oferecer um elevado nível de comodidade e inovação tecnológica, a incorporação do controle por voz em sistemas de automação residencial tem um significativo impacto social. A eliminação da exigência de mobilidade física para o controle de dispositivos de iluminação, climatização e gestão multimídia transforma a residência em um ambiente ativamente promotor de inclusão (REIS, 2021). De acordo com Storck (2019) e Almeida (2022), a implementação de assistentes virtuais devolve a independência, a autonomia e a dignidade a indivíduos em idade avançada, pessoas com deficiência física ou pacientes com restrições motoras severas, materializando o propósito ético e social inerente aos avanços da engenharia e da domótica.

## 3 METODOLOGIA

### 3.1 Introdução

Esta seção descreve os procedimentos metodológicos adotados para o desenvolvimento e a implementação do sistema de automação residencial inteligente. A metodologia detalha as etapas percorridas desde a concepção da infraestrutura utilizando o *Raspberry Pi*, até a integração dos serviços *Home Assistant*, *Node-RED*, dispositivos inteligentes e a assistente virtual *Alexa*. O objetivo desta seção é fornecer transparência sobre como a pesquisa foi conduzida, permitindo a reprodutibilidade do sistema de automação residencial proposto.

### 3.2 Tipo de pesquisa

A presente pesquisa classifica-se como aplicada, de natureza exploratória e experimental. É considerada aplicada por ter como foco a resolução de um problema prático: a criação de um ecossistema de automação residencial acessível utilizando conhecimentos de engenharia da computação e Internet das Coisas (IoT). Apresenta caráter exploratório ao investigar a integração de diferentes plataformas de código aberto e dispositivos comerciais, e experimental devido à montagem física, configuração de software e realização de testes de comunicação e controle em um ambiente real.

### 3.3 Desenho da pesquisa

O projeto foi conduzido de maneira modular e iterativa, estruturado nas seguintes fases principais:

1. **Levantamento Bibliográfico e Documental:** Análise de trabalhos correlatos e documentações oficiais do *Home Assistant*, *Node-RED*, *Docker* e sobre domótica em geral.
2. **Preparação da Infraestrutura:** Instalação e configuração do sistema operacional no *Raspberry Pi*, seguido da implantação do ambiente *Docker* para a containerização dos serviços.
3. **Integração de Software e Lógica:** Configuração do *Home Assistant* como *hub* central e elaboração das lógicas visuais de automação por meio do *Node-RED*.

4. **Integração de Dispositivos Externos:** Conexão de televisores inteligentes, câmeras IP, roteador doméstico, lâmpada inteligente, sensor de temperatura, tomadas inteligentes e vinculação do sistema à assistente de voz *Amazon Alexa* para comandos naturais.

### 3.4 População e amostra

Tratando-se de uma pesquisa voltada para o desenvolvimento de *hardware* e *software*, a população da pesquisa compreende o universo de dispositivos de automação doméstica e plataformas de código aberto disponíveis no mercado. A amostra selecionada é composta pelos equipamentos empregados no cenário real de testes: um microcomputador *Raspberry Pi*, televisores inteligentes, câmera de monitoramento, uma lâmpada inteligente, tomadas inteligentes, um roteador e a assistente virtual *Alexa*.

### 3.5 Coleta de dados

A coleta de dados ocorreu durante a fase de testes do sistema implantado no cenário residencial. Os dados coletados referem-se a métricas de desempenho e funcionamento, incluindo:

- Registros (*logs*) gerados pelos contêineres *Docker* (*Home Assistant* e *Node-RED*) para verificar erros ou falhas de comunicação.
- Registros gerados pela aplicação do *Home Assistant*, em relação aos dispositivos conectados e cadastrados na aplicação.
- Estabilidade da conexão e o consumo de recursos (CPU e memória) do *Raspberry Pi* durante o funcionamento contínuo.

### 3.6 Análise de dados

A análise dos dados obteve um caráter qualitativo e quantitativo. Quantitativamente, avaliou-se a viabilidade técnica do sistema por meio das medições de latência, uso de memória e tempo de disponibilidade dos serviços hospedados no *Raspberry Pi*. Qualitativamente, a análise focou na experiência de uso, validando se as automações e lógicas construídas no *Node-RED* responderam de forma fluida e correta aos eventos programados, e se o ecossistema demonstrou

ser de fato expansível e altamente personalizável, conforme os objetivos do trabalho.

### 3.7 Limitações metodológicas

Dentre as limitações encontradas no decorrer do estudo, destacam-se:

- A dependência de uma infraestrutura de rede local estável, pois flutuações no roteador podem impactar diretamente a comunicação entre o *hub* e os dispositivos inteligentes.
- Possíveis gargalos de processamento inerentes às especificações de *hardware* do *Raspberry Pi* ao adicionar um número muito elevado de dispositivos simultâneos ou fluxos complexos no *Node-RED*.
- A dependência de APIs em nuvem para o funcionamento pleno da assistente *Alexa*, o que significa que o controle por voz pode ficar indisponível em caso de queda de *internet*, limitando-se o sistema, nesses casos, aos painéis de controle locais.

## 4 DESENVOLVIMENTO

Nesta seção, apresenta-se o detalhamento técnico e a execução prática do sistema de automação residencial proposto. A implementação fundamenta-se na integração de *hardware* acessível e plataformas de código aberto, visando consolidar um ambiente que harmonize funcionalidade, baixo custo e autonomia tecnológica. O desenvolvimento foi estruturado de forma modular, permitindo a escalabilidade do sistema e a manutenção da soberania dos dados dentro do ambiente doméstico, minimizando a dependência de infraestruturas proprietárias externas. A seguir, são detalhados a arquitetura do sistema, a configuração do servidor, as integrações realizadas no Home Assistant e na Alexa, e os testes práticos com os dispositivos inteligentes.

### 4.1 Arquitetura do sistema

A arquitetura foi projetada seguindo um modelo de camadas, garantindo que o processamento lógico ocorra de forma centralizada no servidor local, enquanto a interação com o mundo físico é distribuída em dispositivos de borda. Essa topologia foi escolhida para assegurar que o sistema mantenha sua operabilidade mesmo em cenários de instabilidade na rede externa, priorizando a baixa latência e a segurança das informações.

#### 4.1.1 Visão geral da arquitetura

A estrutura do sistema é segmentada em três camadas fundamentais, cada uma com responsabilidades distintas e componentes específicos:

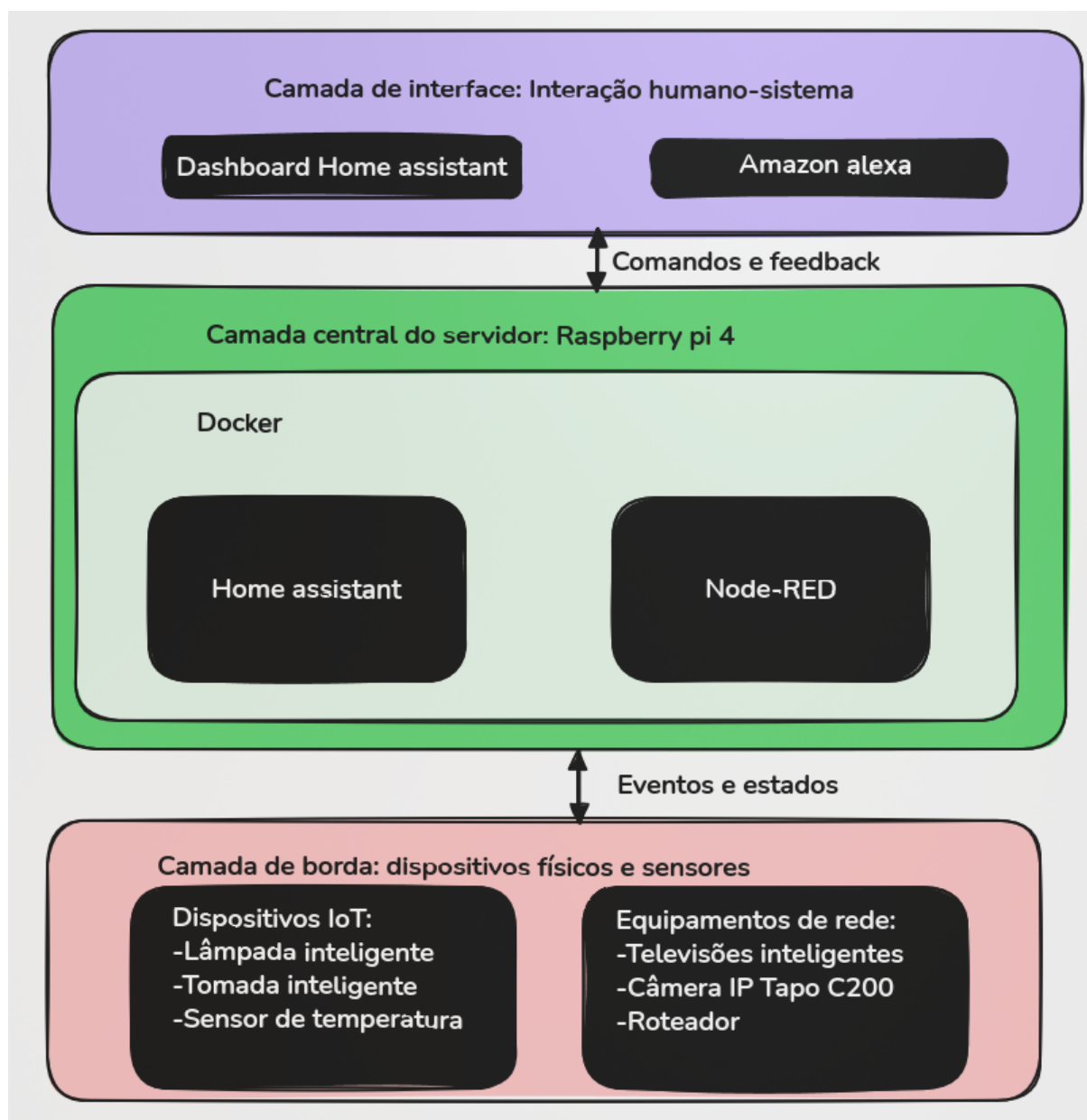
- **Camada de Borda:** Representa a extremidade física do ecossistema, onde ocorre a percepção e a atuação direta no ambiente. Nessa camada encontram-se os dispositivos inteligentes responsáveis pela interação com o ambiente, como a lâmpada inteligente (*smart bulb*), as tomadas inteligentes (*smart plugs*) e o sensor de temperatura, além da rede de dispositivos já presentes na residência, que inclui televisores inteligentes, câmera de monitoramento e o roteador central. Enquanto a lâmpada inteligente e as tomadas atuam diretamente no controle de iluminação e de outros dispositivos elétricos conectados às tomadas, o sensor de temperatura é responsável pela leitura contínua das condições ambientais do cômodo. Os

demais dispositivos, como *Smart TVs*, câmeras IP e o roteador doméstico, são integrados diretamente ao *Home Assistant* para permitir o gerenciamento unificado de multimídia, segurança e conectividade da rede local. A adoção de dispositivos inteligentes comerciais com conectividade Wi-Fi nativa e suporte a integração direta com plataformas de automação reduz a necessidade de prototipagem de *hardware* dedicada, simplificando a expansão do sistema e mantendo a relação custo-benefício alinhada aos objetivos de um ecossistema doméstico acessível.

- **Camada central do servidor:** Centralizada no *Raspberry Pi 4*, que atua como o núcleo de processamento do sistema e onde são gerenciadas e criadas as automações, utilizando para isso as plataformas *Home Assistant* e *Node-RED* executadas em contêineres *Docker*, o que proporciona um ambiente isolado, estável e de fácil manutenção. A escolha do *Raspberry Pi 4* é fundamentada em sua baixa demanda energética e capacidade de processamento suficiente para gerenciar múltiplos serviços simultâneos.
- **Camada de Interface:** Representa o ponto de interação humana, abrangendo tanto a interface gráfica (*Dashboard* do *Home Assistant*) quanto a interface de voz (*Amazon Alexa*). Esta camada traduz as intenções do usuário em comandos processáveis pelo servidor, fornecendo simultaneamente o *feedback* visual e auditivo sobre o estado atual do sistema.

A seguir pode ser observado um esquema visual, que representa essa arquitetura em camadas, na Figura 5.

Figura 5 – Esquema representativo da arquitetura em camadas.



Fonte: O autor (2026).

#### 4.1.2 Fluxo de comunicação e funcionamento do sistema de automação

O fluxo de comunicação é projetado para garantir que a interatividade entre o usuário e o ambiente ocorra de forma transparente, com baixa latência e alta confiabilidade. A comunicação envolve a integração entre as interfaces de usuário, o servidor central e os dispositivos de rede. Este fluxo segue uma lógica de gerenciamento centralizado:

- **Entrada de Dados/Interação:** O ciclo inicia-se por meio de uma solicitação do usuário, seja via comando de voz processado pela nuvem da *Amazon Alexa* ou por interação direta na interface gráfica do *Home Assistant*.
- **Processamento Centralizado no Home Assistant:** O *Raspberry Pi* atua como um *gateway* inteligente. Ele recebe as requisições externas e as traduz em comandos internos. No caso de dispositivos como televisores inteligentes e câmeras, o *Home Assistant* utiliza protocolos específicos para gerenciar o estado e o tráfego de dados.
- **Sincronização de Estado:** Independentemente da origem do comando, o servidor assegura que todos os pontos de interface exibam a mesma informação em tempo real, garantindo a integridade dos dados em todo o ecossistema.
- **Ações e eventos:** A partir da criação de automações no *Home Assistant* são tomadas ações nos dispositivos da camada de borda, mediante condições específicas ou comandos em específico, tanto via interface gráfica quanto através de comandos de voz.

## 4.2 Configuração do servidor

A configuração do servidor central baseado no *Raspberry Pi 4* representa a etapa crítica de preparação da infraestrutura que sustentará o ecossistema de automação. Nesta fase, o objetivo é transformar o hardware em uma unidade de processamento robusta, capaz de gerenciar serviços ininterruptos com alta disponibilidade e segurança. A estratégia de configuração prioriza o desempenho do sistema e a facilidade de manutenção a longo prazo.

### 4.2.1 Sistema operacional

O sistema operacional adotado foi o *Raspberry Pi OS*, selecionado em virtude de sua compatibilidade nativa com a plataforma *Raspberry Pi*, bem como pela ampla disponibilidade de documentação e suporte da comunidade. Trata-se de uma distribuição baseada no sistema *Linux Debian*, otimizada para dispositivos embarcados, proporcionando estabilidade, baixo consumo de recursos e integração facilitada com periféricos e serviços utilizados em aplicações de automação (RASPBERRY PI FOUNDATION, 2025).

O processo de instalação foi realizado a partir do *download* da imagem oficial do sistema, seguido da gravação em um cartão *microSD* por meio da ferramenta oficial *Raspberry Pi Imager*. Durante essa etapa, foram configurados parâmetros iniciais, como definição de credenciais de acesso, habilitação do protocolo SSH para gerenciamento remoto e configuração de rede sem fio. A interface do *Raspberry Pi Imager* pode ser vista na Figura 6.

Figura 6 – Interface *Raspberry Pi Imager*.



Fonte: Raspberry Pi Foundation (2026).

Na primeira inicialização, foram efetuadas configurações adicionais por meio de utilitários nativos do sistema, incluindo atualização dos pacotes, ajuste de fuso horário, configuração de localização e expansão do sistema de arquivos para utilização total do armazenamento disponível. Essas etapas são recomendadas pela documentação oficial como práticas essenciais para garantir o correto funcionamento do sistema e a preparação do ambiente para instalação dos serviços subsequentes.

### 4.2.2 Configuração de rede

A configuração de rede foi realizada com o objetivo de garantir conectividade estável, acesso remoto seguro e adequada integração com os demais dispositivos da rede local. Inicialmente, o sistema foi configurado para utilizar o protocolo DHCP (*Dynamic Host Configuration Protocol*), porém, posteriormente, visando maior previsibilidade na comunicação entre serviços, foi adotada a utilização de endereço IP estático. Essa configuração foi realizada por meio da definição manual no sistema operacional, com a edição dos parâmetros de rede. A utilização de um endereço fixo é essencial em ambientes de automação residencial, pois evita inconsistências na comunicação entre serviços que dependem de um endereço conhecido para funcionamento contínuo.

Para possibilitar o gerenciamento remoto do dispositivo, foi habilitado o serviço de acesso via SSH, permitindo a administração segura do sistema sem a necessidade de acesso físico. Foram definidos parâmetros básicos de segurança, como autenticação por credenciais e verificação de conectividade. Na Figura 7 pode ser observada a configuração do IP no servidor.

Figura 7 – Configuração dos IPs do servidor.

```
Linux andre 6.12.93+rpt-rpi-v8 #1 SMP PREEMPT Debian 1:6.12.93-1+rpt1 (2026-06-12) aarch64

1 updates could not be installed automatically. For more details,
see /var/log/unattended-upgrades/unattended-upgrades.log

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

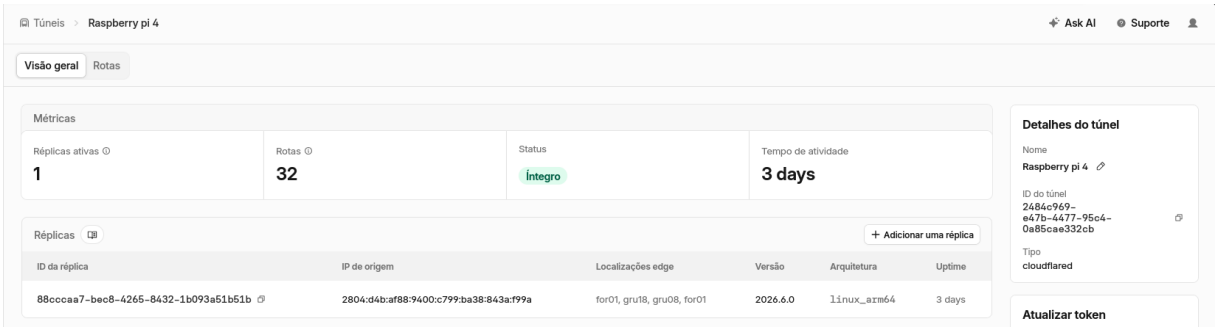
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
You have mail.
Last login: Thu Jul  9 21:30:16 2026 from 192.168.100.193
Welcome to fish, the friendly interactive shell
Type help for instructions on how to use fish
andre@andre ~-> hostname -I
192.168.100.17 192.168.100.185 172.28.0.1 172.25.0.1 172.19.0.1 192.168.0.1 172.26.0.1 172.18.0.1
172.27.0.1 172.22.0.1 172.17.0.1 172.23.0.1 100.90.154.70 172.20.0.1 172.21.0.1 2804:d4b:af88:9400
:c799:ba38:843a:f99a 2804:d4b:af88:9400:260b:174f:bb2:e95d fd7a:115c:a1e0::4c01:9a46
andre@andre ~->
```

Fonte: O autor (2026).

Adicionalmente, foi implementada uma solução de acesso externo por meio da utilização de um domínio personalizado (rasppi.cloud), utilizando *Cloudflare*

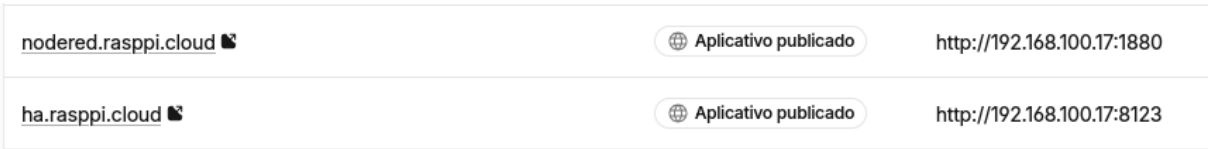
*Tunnel*, tecnologia que permite acesso remoto seguro (CLOUDFLARE, 2026). Essa abordagem permitiu expor os serviços hospedados localmente, sem a necessidade de abertura de portas no roteador ou configuração de redirecionamento. O túnel foi estabelecido a partir da instalação e autenticação do agente *cloudflared* no dispositivo, criando uma conexão segura entre o ambiente local e a infraestrutura da *Cloudflare*. A conexão e rotas criadas pelo túnel na *Cloudflare* podem ser visualizadas nas Figuras 8 e 9.

Figura 8 – Conexão do túnel na *Cloudflare*.



Fonte: O autor (2026).

Figura 9 – Rotas do *Node-red* e *Home Assistant* no túnel da *Cloudflare*.



Fonte: O autor (2026).

Dessa forma, tornou-se possível acessar os serviços, como o servidor de automação, utilizando um endereço amigável e estável. Essa solução aumenta significativamente a segurança e a confiabilidade do acesso remoto, sendo amplamente recomendada em cenários IoT modernos. A configuração de rede seguiu práticas consolidadas em ambientes IoT, priorizando segurança, disponibilidade e facilidade de gerenciamento, conforme orientações da documentação oficial do *Raspberry Pi* e da *Cloudflare*.

4.2.3 Instalação do Docker

A instalação e o gerenciamento do ambiente de contêineres foram realizados com o auxílio do *CasaOS*, o qual automatiza a configuração do *Docker* e fornece uma interface gráfica centralizada para administração dos serviços, conforme a

documentação oficial do *Docker* (DOCKER INC., 2026) e do CasaOS (ICEWHALE TECHNOLOGY, 2026). A adoção dessa solução teve como objetivo simplificar o processo de instalação, reduzir a complexidade de configuração manual e facilitar o gerenciamento contínuo dos contêineres.

O processo iniciou-se com a instalação do CasaOS por meio de um *script* oficial disponibilizado em sua documentação. Durante essa etapa, o sistema realiza automaticamente a instalação do *Docker* e de seus componentes essenciais, incluindo o mecanismo de execução de contêineres e serviços auxiliares. Essa abordagem elimina a necessidade de configuração manual detalhada do *Docker*, garantindo uma instalação padronizada e funcional.

Após a instalação, o CasaOS disponibiliza um painel *web* acessível via navegador, permitindo o gerenciamento dos contêineres de forma visual e intuitiva. Por meio dessa interface, é possível realizar operações como criação, inicialização, interrupção e monitoramento de contêineres, além de gerenciar volumes, redes e variáveis de ambiente. Esse modelo facilita a administração do sistema, especialmente em cenários com múltiplos serviços, como em aplicações de automação residencial. A interface *web* do CasaOS é vista na Figura 10.

Figura 10 – Interface CasaOS.



Fonte: O autor (2026).

Adicionalmente, o *CasaOS* integra um catálogo de aplicações que permite a instalação simplificada de serviços baseados em contêineres, reduzindo a necessidade de utilização direta de comandos em linha de comando. Ainda assim, o acesso ao *Docker* via terminal permanece disponível, possibilitando maior flexibilidade para configurações avançadas quando necessário.

Dessa forma, a utilização do *CasaOS* como camada de abstração sobre o *Docker* contribui para a padronização do ambiente, melhora a usabilidade e reduz a complexidade operacional, mantendo as vantagens do uso de contêineres, como isolamento, portabilidade e facilidade de manutenção.

#### **4.2.4 Instalação e configuração dos serviços**

A instalação e configuração dos serviços foram realizadas por meio de contêineres *Docker* gerenciados através do *CasaOS*, permitindo uma implantação simplificada e padronizada das aplicações. A utilização dessa abordagem possibilita o isolamento dos serviços, facilidade de atualização e maior controle sobre os recursos utilizados pelo sistema.

A instalação do *Home Assistant* foi realizada a partir do catálogo de aplicações disponível no *CasaOS*, que automatiza a criação do contêiner com parâmetros previamente definidos. Durante o processo, foram configurados elementos essenciais, como mapeamento de portas, definição de volumes persistentes para armazenamento das configurações e dados do sistema, além da associação à rede local. Após a inicialização do contêiner, o acesso à interface *web* foi realizado via navegador, onde se procedeu-se à configuração inicial da plataforma, incluindo criação de usuário administrador, detecção automática de dispositivos e integração com serviços compatíveis.

De forma similar, o *Node-RED* também foi implantado via *CasaOS*, utilizando imagem *Docker* oficial. A configuração incluiu a definição de diretórios persistentes para armazenamento dos fluxos e credenciais, bem como a exposição da porta de acesso ao editor *web*. Após a inicialização, o ambiente foi acessado via navegador, permitindo a criação e edição de fluxos de automação por meio de interface gráfica baseada em nós.

A integração entre os serviços foi realizada por meio da instalação de módulos específicos do *Node-RED* para comunicação com o *Home Assistant*, utilizando APIs e autenticação via *token* de acesso gerado na própria plataforma. Essa integração possibilita a criação de automações mais complexas e personalizadas, ampliando as capacidades nativas do sistema.

Por fim, o gerenciamento dos contêineres, incluindo monitoramento, reinicialização e atualização, foi centralizado no painel do *CasaOS*, proporcionando maior controle operacional e facilidade de manutenção.

### **4.3 Configuração das integrações**

Após a implantação da infraestrutura computacional descrita anteriormente, procedeu-se à integração dos dispositivos inteligentes e das plataformas responsáveis pelo gerenciamento e pela automação do ambiente residencial. Essa etapa foi fundamental para consolidar um ecossistema unificado capaz de centralizar informações provenientes de diferentes equipamentos e possibilitar a execução de automações avançadas por meio de fluxos lógicos personalizados.

A integração dos componentes foi realizada de forma incremental, iniciando-se pela conexão dos dispositivos inteligentes ao *Home Assistant*, seguida pela integração do *Node-RED* ao ambiente de automação e, por fim, pela incorporação da assistente virtual *Amazon Alexa* para viabilizar o controle por voz do sistema. O modelo arquitetural adotado permitiu que o *Home Assistant* atuasse como núcleo central da solução, concentrando o gerenciamento das entidades e disponibilizando interfaces de comunicação para os demais serviços, de modo que dispositivos físicos, fluxos de automação e comandos de voz passassem a operar de forma coordenada.

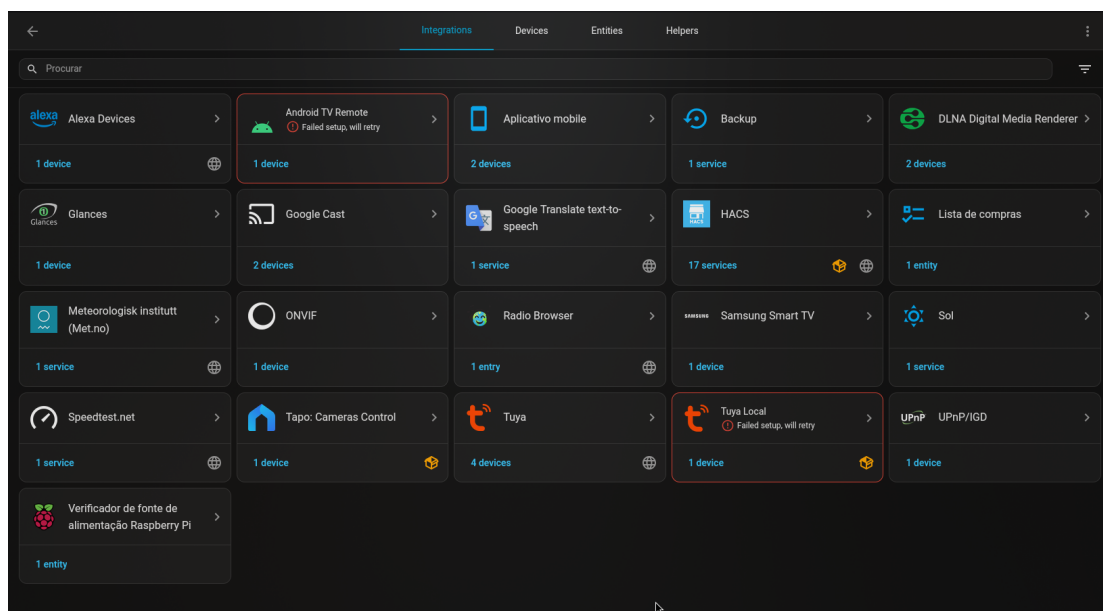
#### **4.3.1 Integração de dispositivos inteligentes e Home Assistant**

A integração dos dispositivos inteligentes foi realizada utilizando os mecanismos nativos de descoberta e configuração disponibilizados pelo *Home Assistant*. O sistema identifica automaticamente a maioria dos equipamentos

compatíveis presentes na rede local, simplificando significativamente o processo de cadastramento e gerenciamento dos dispositivos.

Entre os equipamentos integrados destacam-se duas tomadas inteligentes com monitoramento de consumo energético, um sensor de temperatura e umidade, uma câmera IP *Tapo C200*, televisores inteligentes *Samsung* e *TCL (Android TV)* e uma lâmpada inteligente *Positivo Casa Inteligente*. Após a integração, cada equipamento passou a disponibilizar entidades específicas dentro do *Home Assistant*, permitindo o monitoramento contínuo de estados e a execução de comandos remotos. No caso das tomadas inteligentes, além do acionamento remoto das cargas conectadas, tornou-se possível registrar métricas relacionadas ao consumo energético acumulado. Já o sensor ambiental passou a fornecer informações em tempo real sobre temperatura, umidade e nível de bateria. As integrações podem ser visualizadas no painel de integrações do *Home Assistant* na Figura 11.

Figura 11 – Painel de integrações do *Home Assistant*.



Fonte: O autor (2026).

A integração da câmera *Tapo C200* possibilitou a visualização do fluxo de vídeo em tempo real, além da utilização de recursos avançados disponibilizados pelo fabricante, incluindo movimentação remota, controle de sirene, detecção de

movimento e configuração de alarmes automáticos. Enquanto a integração com a lâmpada inteligente permite o controle de cor, intensidade e acionamento.

As televisões inteligentes foram incorporadas ao sistema como dispositivos multimídia, permitindo o monitoramento de estado e a execução de comandos como ligar, desligar e controle de mídia. Essa funcionalidade tornou possível o desenvolvimento de automações relacionadas à criação de cenários de entretenimento.

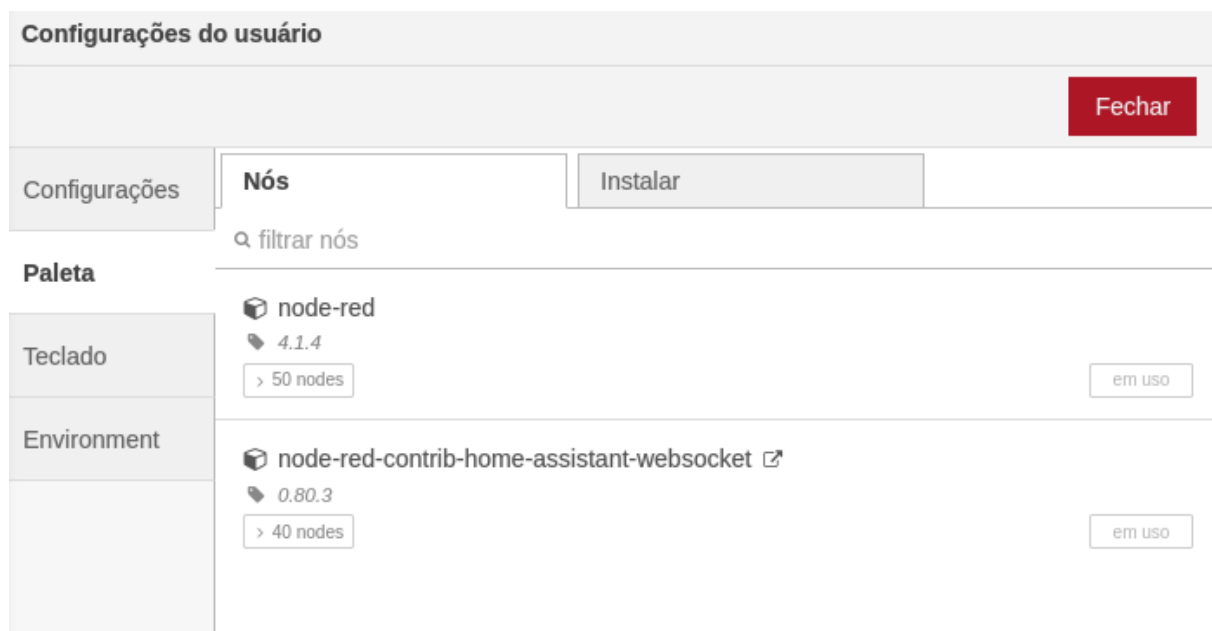
Além dos dispositivos físicos, também foram integradas entidades de monitoramento da infraestrutura computacional. O *Raspberry Pi* passou a fornecer métricas relacionadas ao uso de processador, memória, armazenamento e temperatura permitindo acompanhar continuamente o estado do servidor responsável pela automação residencial. A outra integração utilizada foi com o *speedtest*, permitindo monitoramento da qualidade e velocidade da internet continuamente.

A centralização de todas essas informações em uma única plataforma proporcionou uma visão unificada do ambiente residencial, simplificando o gerenciamento do sistema e servindo como base para a implementação das automações descritas posteriormente.

#### **4.3.2 Integração do Node-RED e Home Assistant**

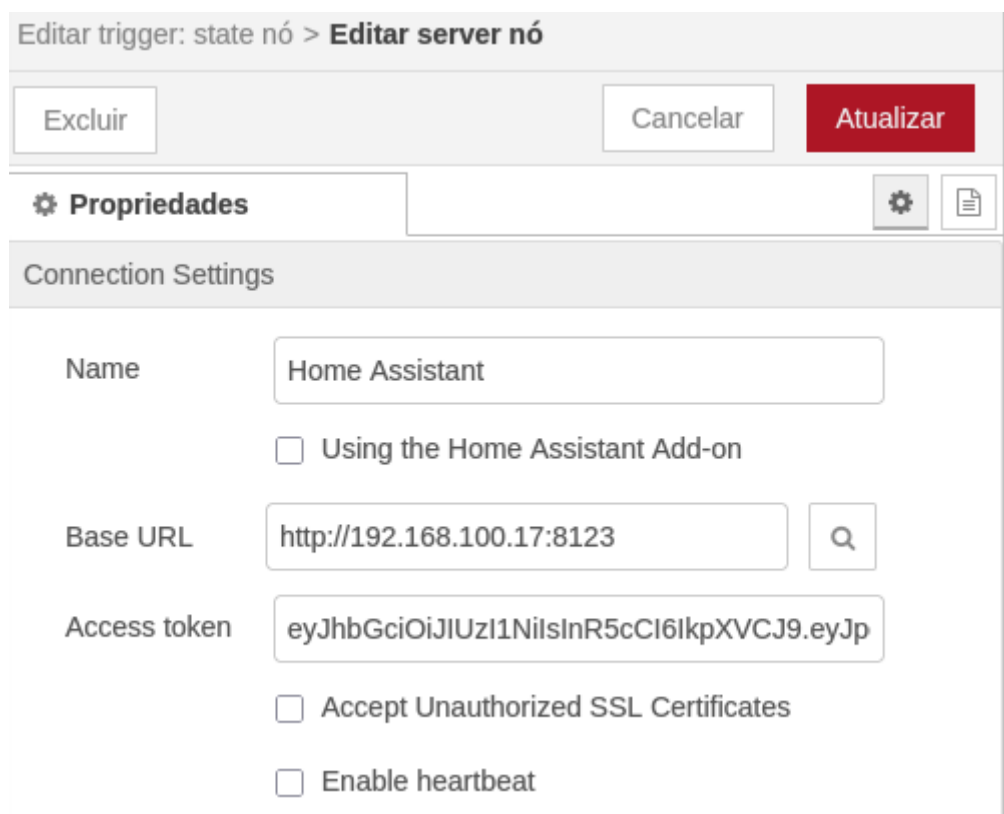
Após os dispositivos estarem devidamente integrados ao *Home Assistant*, procedeu-se à integração da plataforma *Node-RED*, responsável pela implementação das automações avançadas do sistema. O *Node-RED* foi implantado em contêiner *Docker* e conectado ao *Home Assistant* por meio da extensão *Home Assistant WebSocket*, que estabelece comunicação bidirecional entre as plataformas. Essa integração permite que eventos gerados pelo *Home Assistant* sejam recebidos pelo *Node-RED* em tempo real, enquanto comandos produzidos pelos fluxos possam ser executados diretamente sobre as entidades cadastradas. Nas Figuras 12 e 13 pode ser vista a extensão instalada e configurada, para se integrar ao *Home Assistant*, respectivamente.

Figura 12 – Instalação da extensão *Home Assistant WebSocket*.



Fonte: O autor (2026).

Figura 13 – Configuração da extensão *Home Assistant WebSocket*.



Fonte: O autor (2026).

A principal vantagem dessa abordagem consiste na separação entre gerenciamento de dispositivos e lógica de automação. Enquanto o *Home Assistant* permanece responsável pela integração dos equipamentos, o *Node-RED* assume a execução dos fluxos de decisão, permitindo a construção de automações complexas por meio de programação visual baseada em nós.

Durante o desenvolvimento do projeto, o *Node-RED* foi utilizado para implementar fluxos envolvendo temporizações, monitoramento de estados, verificações condicionais e execução de ações coordenadas. Como exemplo, foi implementada uma rotina responsável pelo desligamento automático da televisão do quarto em horário previamente definido. Nesse fluxo, um nó de agendamento executa diariamente um evento programado, acionando posteriormente um serviço do *Home Assistant* responsável pelo desligamento do dispositivo.

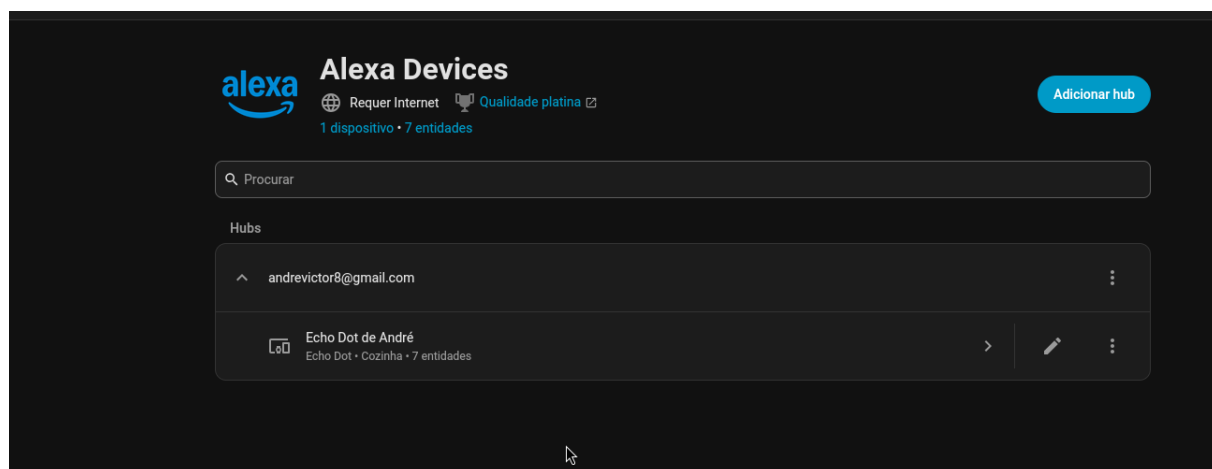
A utilização do *Node-RED* também proporcionou maior facilidade de manutenção e escalabilidade. Novas automações podem ser adicionadas ao sistema sem necessidade de alterações estruturais significativas, permitindo a expansão gradual do ambiente residencial inteligente.

Outro aspecto relevante refere-se à capacidade de monitoramento e depuração disponibilizada pela plataforma. Os fluxos implementados passaram a registrar eventos e erros durante sua execução, facilitando a identificação de falhas e contribuindo para a confiabilidade operacional do sistema.

#### **4.3.3 Integração da Amazon Alexa e Home Assistant**

A integração da *Amazon Alexa* foi realizada com o objetivo de fornecer uma interface natural de interação entre os usuários e o ambiente automatizado. A comunicação entre *Alexa* e *Home Assistant* foi estabelecida utilizando os recursos de integração disponibilizados pela plataforma, pelo *Alexa Devices*, permitindo que entidades selecionadas fossem expostas para a assistente virtual. Após o processo de sincronização, dispositivos como tomadas inteligentes, televisores e automações passaram a ser reconhecidos automaticamente pelos dispositivos *Echo* associados à conta *Amazon*. A conexão feita com a *Alexa* via *Alexa Devices* pode ser observada na Figura 14.

Figura 14 – Integração *Alexa Devices* no *Home Assistant*.



Fonte: O autor (2026).

A partir dessa integração, tornou-se possível controlar equipamentos por meio de comandos de voz, eliminando a necessidade de interação direta com aplicativos ou interfaces *web*. Operações como ligar e desligar tomadas, controlar televisores e executar rotinas específicas passaram a ser realizadas utilizando linguagem natural.

Além da melhoria na experiência de uso, a integração com *Alexa* contribuiu para ampliar a acessibilidade do sistema. Usuários com mobilidade reduzida ou dificuldades motoras podem controlar diversos recursos da residência utilizando apenas comandos de voz, aumentando a autonomia e a praticidade no cotidiano.

A combinação entre *Home Assistant*, *Node-RED* e *Alexa* permitiu a construção de uma arquitetura híbrida na qual os comandos emitidos pelos usuários são interpretados pela assistente virtual, processados pelo *Home Assistant* e, quando necessário, encaminhados aos fluxos de automação implementados no *Node-RED*. Esse modelo possibilitou integrar dispositivos de diferentes fabricantes em uma única plataforma de controle centralizado.

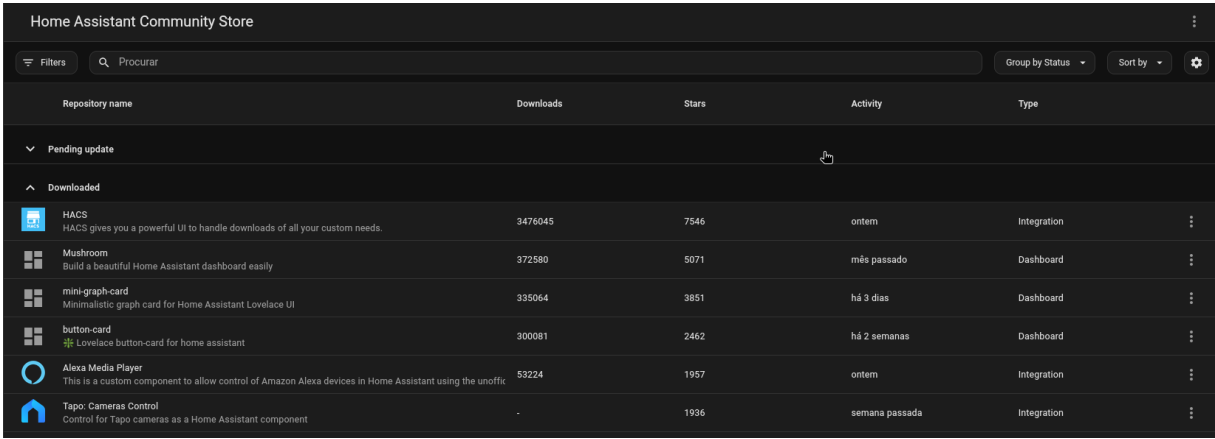
#### 4.3.4 Home Assistant Community Store (HACS)

Com o objetivo de ampliar as funcionalidades disponíveis no *Home Assistant* e possibilitar a integração de componentes desenvolvidos pela comunidade, foi realizada a instalação do *Home Assistant Community Store* (HACS). O HACS consiste em uma plataforma complementar que permite a instalação, atualização e

gerenciamento de integrações, cartões de *dashboard*, temas e outros recursos não incluídos nativamente na distribuição padrão do *Home Assistant*.

A instalação do HACS foi realizada por meio da adição do repositório oficial ao ambiente *Home Assistant*, seguida da autenticação junto à plataforma *GitHub* e da configuração da integração dentro do sistema, conforme documentação oficial do Home Assistant Community Store (HACS, 2026). Após a conclusão do processo, passou a ser possível acessar diretamente pelo painel administrativo uma biblioteca centralizada contendo centenas de integrações e componentes desenvolvidos pela comunidade. A interface do HACS pode ser observada na Figura 15.

Figura 15 – Interface do HACS no *Home Assistant*.



| Home Assistant Community Store                                                                                                   |           |          |                |                 |
|----------------------------------------------------------------------------------------------------------------------------------|-----------|----------|----------------|-----------------|
| Filters                                                                                                                          |           | Procurar |                | Group by Status |
| Repository name                                                                                                                  | Downloads | Stars    | Activity       | Type            |
| Pending update                                                                                                                   |           |          |                |                 |
| Downloaded                                                                                                                       |           |          |                |                 |
| HACS<br>HACS gives you a powerful UI to handle downloads of all your custom needs.                                               | 3476045   | 7546     | ontem          | Integration     |
| Mushroom<br>Build a beautiful Home Assistant dashboard easily                                                                    | 372580    | 5071     | mês passado    | Dashboard       |
| mini-graph-card<br>Minimalistic graph card for Home Assistant Lovelace UI                                                        | 335064    | 3851     | há 3 dias      | Dashboard       |
| button-card<br>Lovelace button-card for home assistant                                                                           | 300081    | 2462     | há 2 semanas   | Dashboard       |
| Alexa Media Player<br>This is a custom component to allow control of Amazon Alexa devices in Home Assistant using the unofficial | 53224     | 1957     | ontem          | Integration     |
| Tapo: Cameras Control<br>Control for Tapo cameras as a Home Assistant component                                                  | -         | 1936     | semana passada | Integration     |

Fonte: O autor (2026).

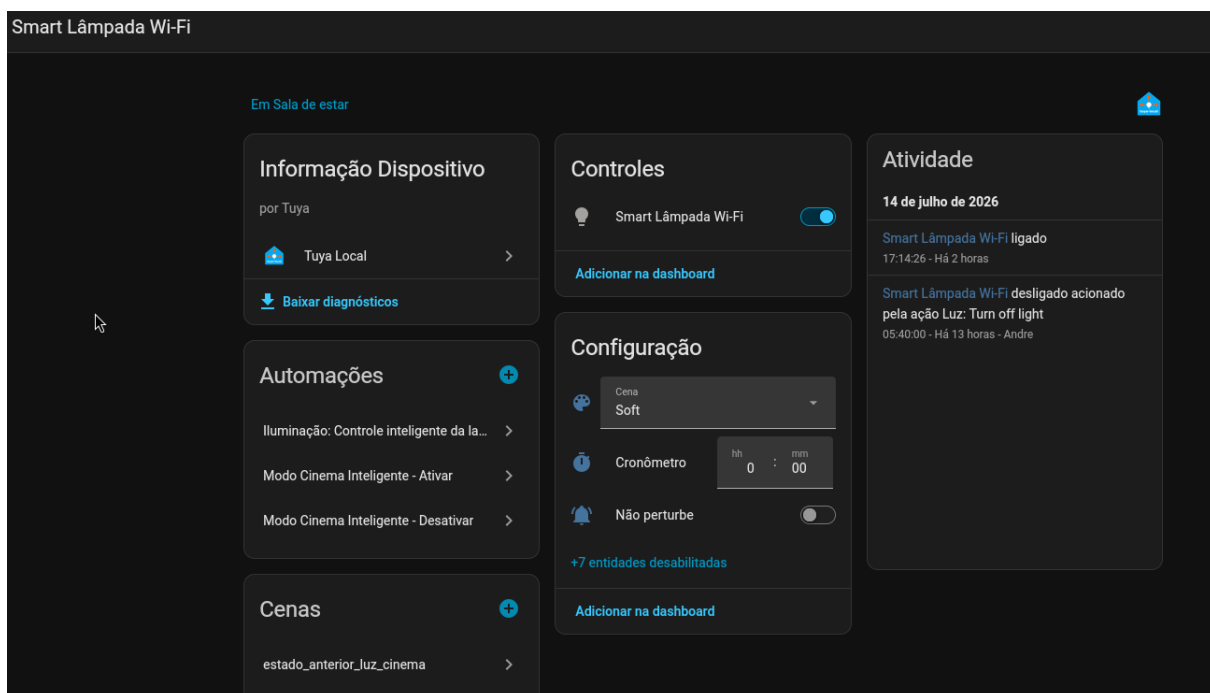
O principal benefício proporcionado pelo HACS está relacionado à expansão das capacidades do *Home Assistant* sem a necessidade de desenvolvimento manual de integrações. Dessa forma, recursos adicionais podem ser incorporados ao sistema de maneira simplificada, permitindo maior compatibilidade com dispositivos e serviços de terceiros.

No contexto deste trabalho, o HACS desempenha papel fundamental na integração de dispositivos e na personalização da interface de gerenciamento da residência inteligente.

Outra integração instalada por meio do HACS foi o *Tuya Local*, responsável pela comunicação local com dispositivos compatíveis com a plataforma *Tuya*. Sua utilização permitiu integrar a lâmpada inteligente presente no ambiente residencial

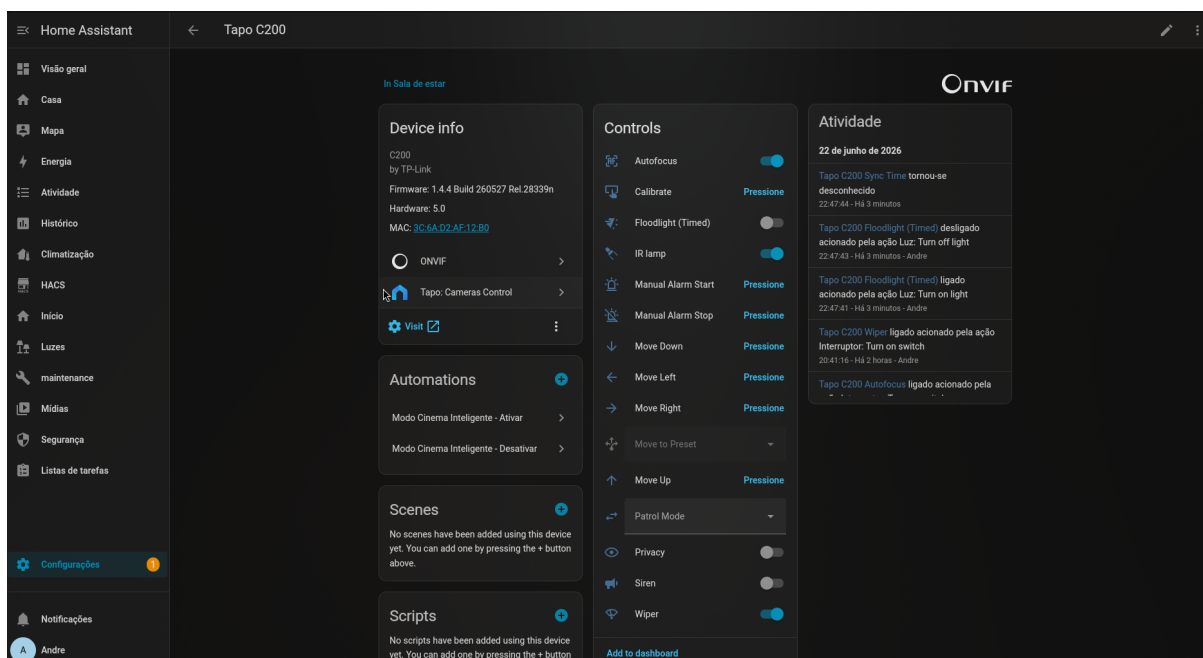
diretamente ao *Home Assistant*, eliminando a dependência exclusiva dos serviços em nuvem do fabricante. Como consequência, obteve-se maior velocidade de resposta, redução da dependência da internet e aumento da privacidade dos dados trafegados entre os dispositivos. Na Figura 16 é possível observar a integração com o *Tuya Local* no *Home Assistant*.

Figura 16 – Integração com o *Tuya Local* no *Home Assistant*.



Fonte: O autor (2026).

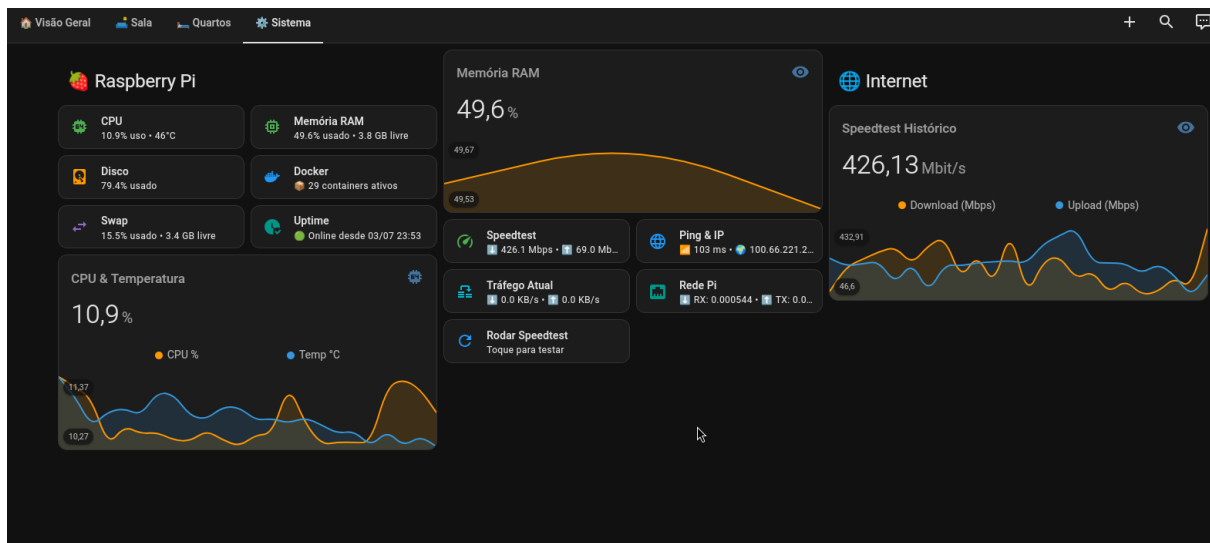
Além disso, foi utilizada a integração *Tapo: Cameras Control*, destinada à ampliação dos recursos de gerenciamento da câmera *IP Tapo C200*. Por meio dessa integração, tornou-se possível acessar funcionalidades adicionais da câmera diretamente pelo *Home Assistant*, incluindo monitoramento de estado, acionamento de recursos específicos do equipamento e utilização de eventos em automações. Na Figura 17 é possível visualizar a integração com o *Tapo: Cameras Control*.

Figura 17 – Integração com o *Tapo: Cameras Control*.

Fonte: O autor (2026).

O HACS também foi empregado na personalização dos *dashboards* do sistema, visto que diversos componentes gráficos foram instalados para melhorar a apresentação das informações e proporcionar uma interface mais intuitiva aos usuários. Entre os recursos utilizados destacam-se cartões personalizados para exibição de dispositivos, gráficos históricos, indicadores de estado e elementos visuais avançados para organização do painel de controle. Na Figura 18 pode ser visualizado o uso desses cartões no *dashboard*.

Figura 18 – Dashboard com cartões do HACS.



Fonte: O autor (2026).

Dessa forma, o *Home Assistant Community Store* contribuiu significativamente para a evolução da plataforma desenvolvida, ampliando sua capacidade de integração, aumentando a flexibilidade de configuração e proporcionando uma interface visual mais adequada ao gerenciamento dos dispositivos e automações implementados no sistema.

#### 4.4 Fluxos e automações

Após a conclusão das etapas de integração, foram implementadas automações destinadas a demonstrar a capacidade operacional da arquitetura proposta. As rotinas desenvolvidas contemplam cenários relacionados ao conforto, economia de energia, segurança residencial e monitoramento da infraestrutura computacional.

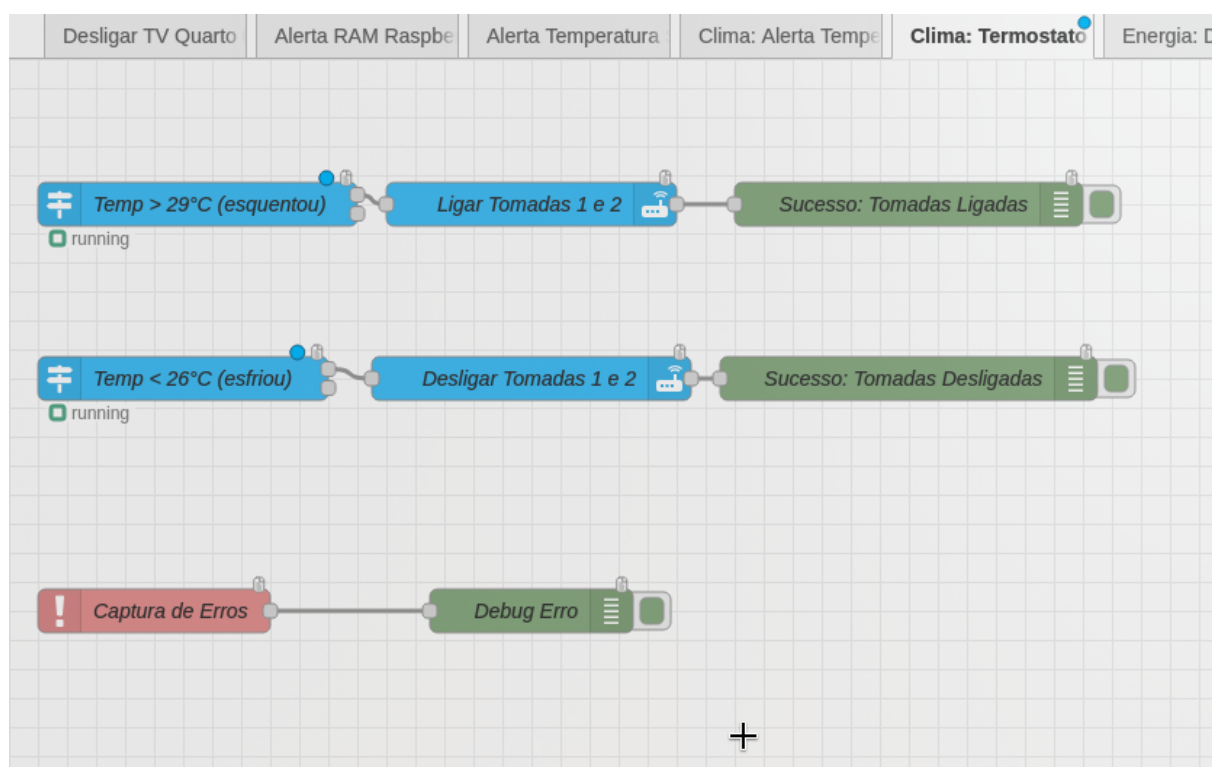
As automações foram implementadas utilizando recursos nativos do *Home Assistant* e fluxos personalizados desenvolvidos no *Node-RED*, explorando eventos, estados de entidades, sensores do ambiente e comandos provenientes da assistente virtual.

#### 4.4.1 Controle termostático inteligente

Uma das automações implementadas foi destinada ao controle térmico do ambiente utilizando o sensor de temperatura integrado ao sistema e uma tomada inteligente responsável pelo acionamento de equipamentos de climatização.

O funcionamento baseia-se no monitoramento contínuo da temperatura ambiente. Quando o valor medido ultrapassa o limite previamente definido pelo usuário, a tomada inteligente é acionada automaticamente. Quando a temperatura retorna à faixa desejada, o equipamento é desligado. Essa estratégia permite manter condições térmicas mais confortáveis ao mesmo tempo em que reduz intervenções manuais por parte do usuário. Na Figura 19 está disponível o fluxo da lógica para implementar essa automação.

Figura 19 – Fluxo no *Node-RED* para controle termostático inteligente.

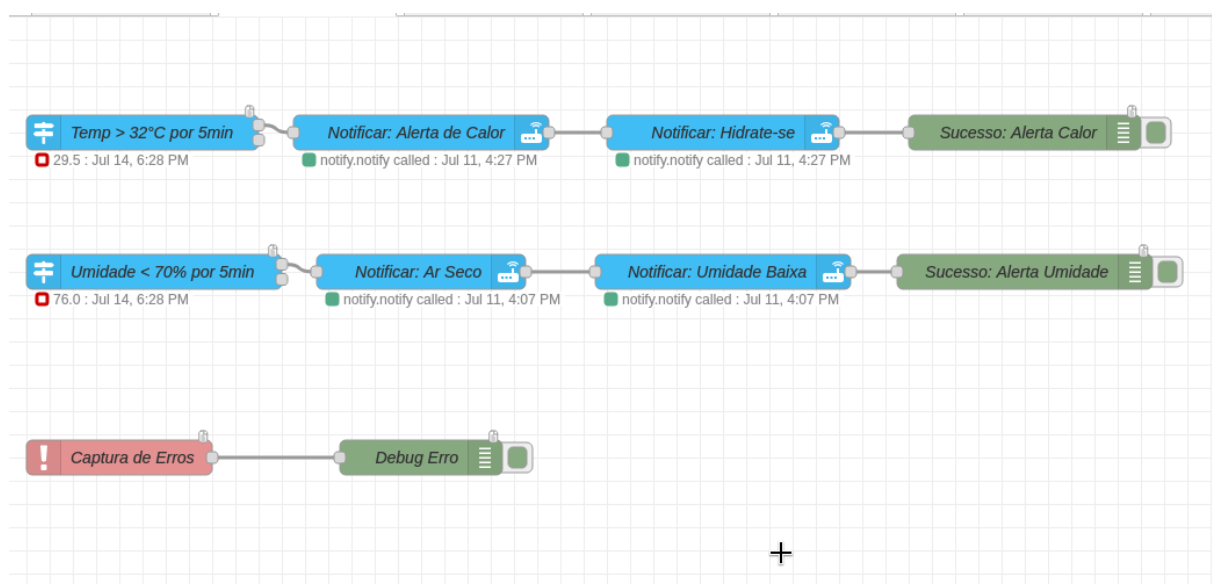


Fonte: O autor (2026).

#### 4.4.2 Monitoramento do ambiente

Foi implementada uma automação responsável pelo acompanhamento contínuo das condições ambientais da residência. Utilizando os dados fornecidos pelo sensor de temperatura e umidade, o sistema monitora possíveis desvios em relação aos limites estabelecidos. Quando valores críticos são detectados, notificações podem ser geradas para alertar os usuários sobre alterações significativas nas condições do ambiente. Além de contribuir para o conforto residencial, essa automação demonstra a utilização prática de sensores IoT em aplicações de monitoramento contínuo. Na Figura 20 pode ser observado o fluxo de automação para monitoramento de ambiente.

Figura 20 – Fluxo no *Node-RED* para monitoramento do ambiente.



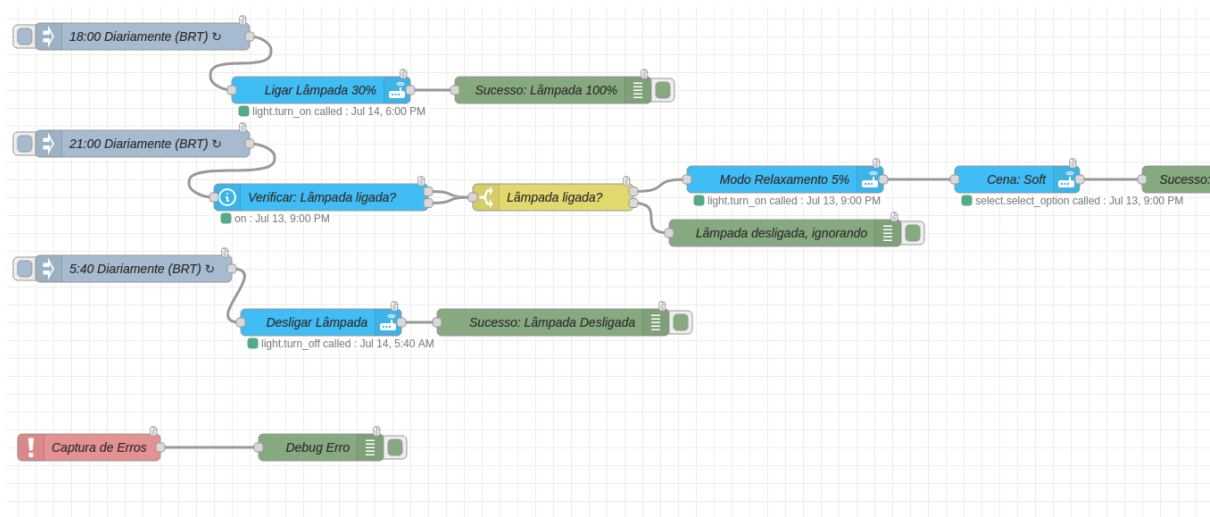
Fonte: O autor (2026).

#### 4.4.3 Controle inteligente de iluminação

Essa automação teve como objetivo o gerenciamento automatizado da iluminação residencial. O sistema permite controlar dispositivos de iluminação por meio do Home Assistant e da Amazon Alexa, possibilitando tanto acionamento manual quanto execução automática baseada em cenários previamente configurados, como em determinados horários, por exemplo. A utilização de comandos de voz permite controlar as características da iluminação (cor e intensidade) e reduz o tempo necessário para interação com os dispositivos e

contribui para a acessibilidade do ambiente automatizado. A Figura 21 mostra esse fluxo para esse controle de iluminação.

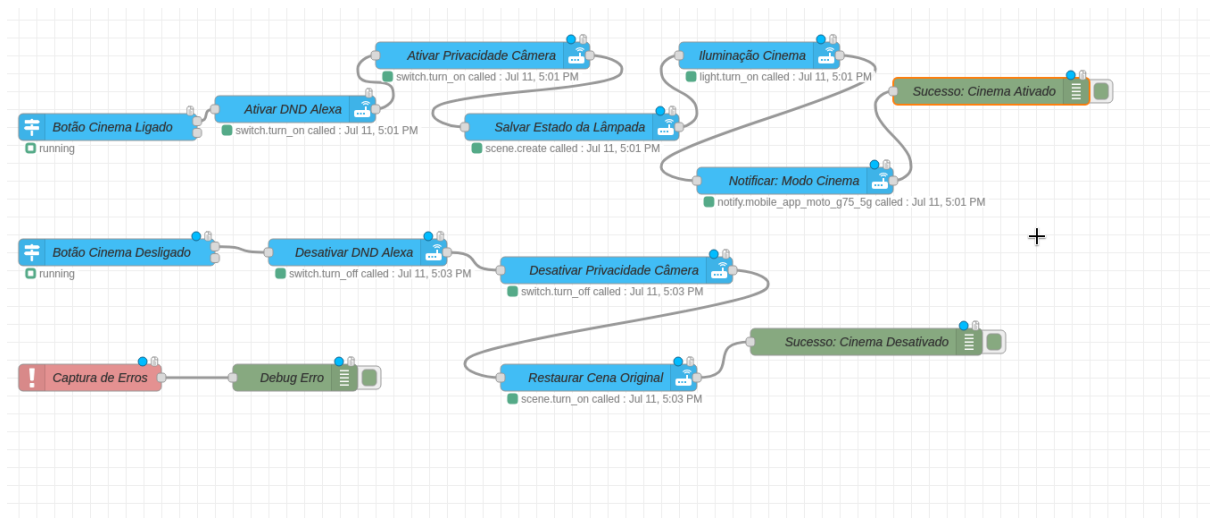
Figura 21 – Fluxo no *Node-RED* para controle inteligente de iluminação.



Fonte: O autor (2026).

#### 4.4.4 Modo cinema

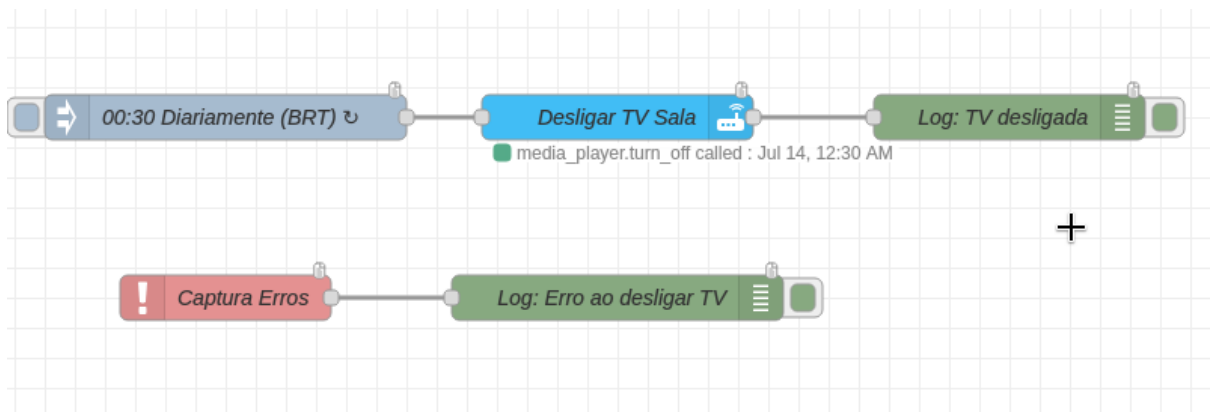
Com o objetivo de aprimorar a experiência de entretenimento, foi implementada uma automação denominada Modo Cinema. Quando ativada, a rotina executa uma sequência coordenada de ações sobre dispositivos multimídia previamente configurados. Quando ativado, o modo cinema controla a iluminação, ativa o modo privativo da câmera e o modo Não Perturbe do *Amazon Echo Dot*, conforme ilustrado na Figura 22.

Figura 22 – Fluxo no *Node-RED* do modo cinema.

Fonte: O autor (2026).

#### 4.4.5 Desligamento automático de televisores

Foi desenvolvida uma automação destinada ao desligamento programado de televisores em horários específicos. O fluxo implementado utiliza mecanismos de agendamento para executar diariamente comandos de desligamento enviados ao *Home Assistant*. Essa funcionalidade contribui para a redução do consumo energético e evita que equipamentos permaneçam ligados desnecessariamente durante longos períodos. A implementação também demonstra a integração efetiva entre *Node-RED* e *Home Assistant* na execução de tarefas temporizadas conforme ilustrado na *Figura 23*.

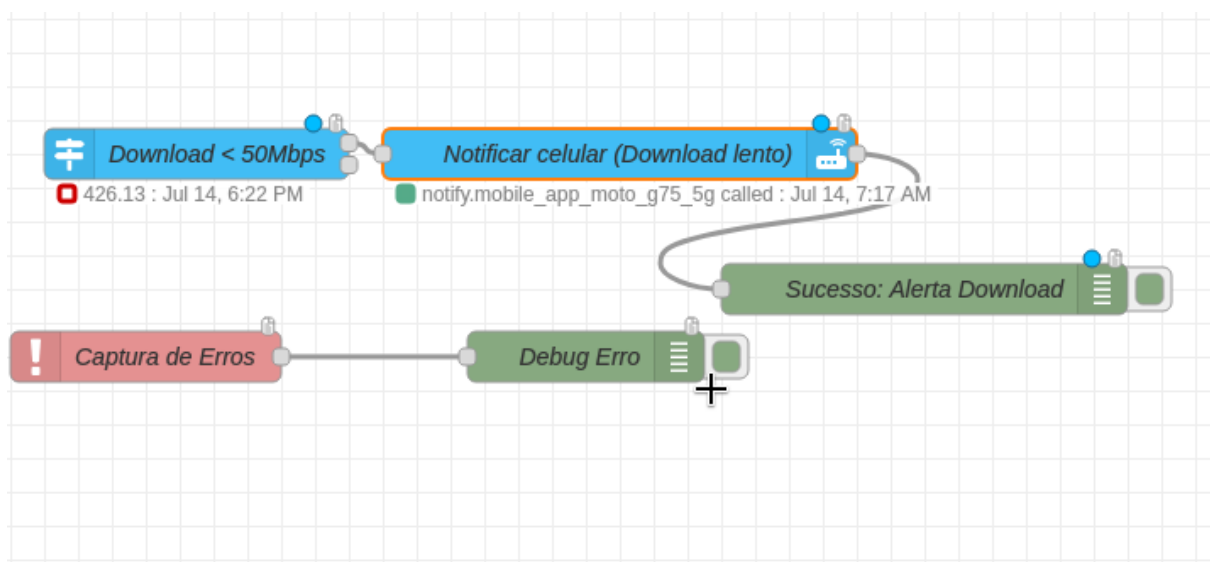
Figura 23 – Fluxo no *Node-RED* para desligamento automático de televisão.

Fonte: O autor (2026).

#### 4.4.6 Monitoramento da velocidade da internet

Considerando a dependência da conectividade em sistemas IoT, foi implementada uma automação destinada ao monitoramento da qualidade da conexão com a internet. Utilizando sensores provenientes da integração *Speedtest*, o sistema registra métricas de *download*, *upload* e latência da rede. Essas informações permitem acompanhar o desempenho da infraestrutura de comunicação e identificar eventuais degradações do serviço. Caso a internet caia, o *Home Assistant* irá notificar o usuário pelo aplicativo *mobile*. O fluxo pode ser observado na Figura 24.

Figura 24 – Fluxo no *Node-RED* para monitoramento da velocidade da internet.



Fonte: O autor (2026).

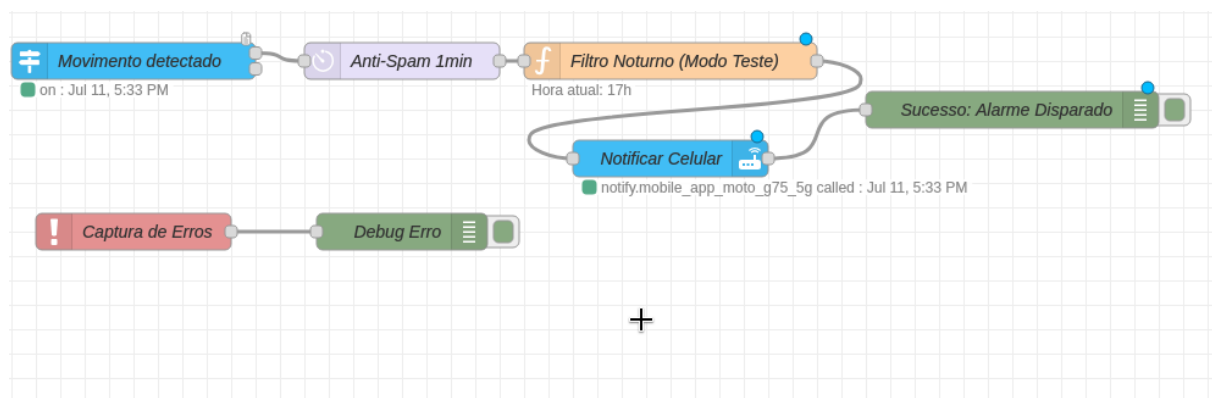
#### 4.4.7 Sistema de segurança residencial

Para ampliar a segurança do ambiente, foi implementada uma automação de monitoramento baseada na câmera IP integrada ao sistema e em informações de presença dos usuários.

Durante períodos previamente definidos, eventos considerados suspeitos podem gerar notificações automáticas e acionar mecanismos de alerta. Essa funcionalidade demonstra a aplicação prática de tecnologias de vigilância integradas

a plataformas de automação residencial. Na Figura 25 pode ser observado esse fluxo.

Figura 25 – Fluxo no *Node-RED* para segurança residencial.

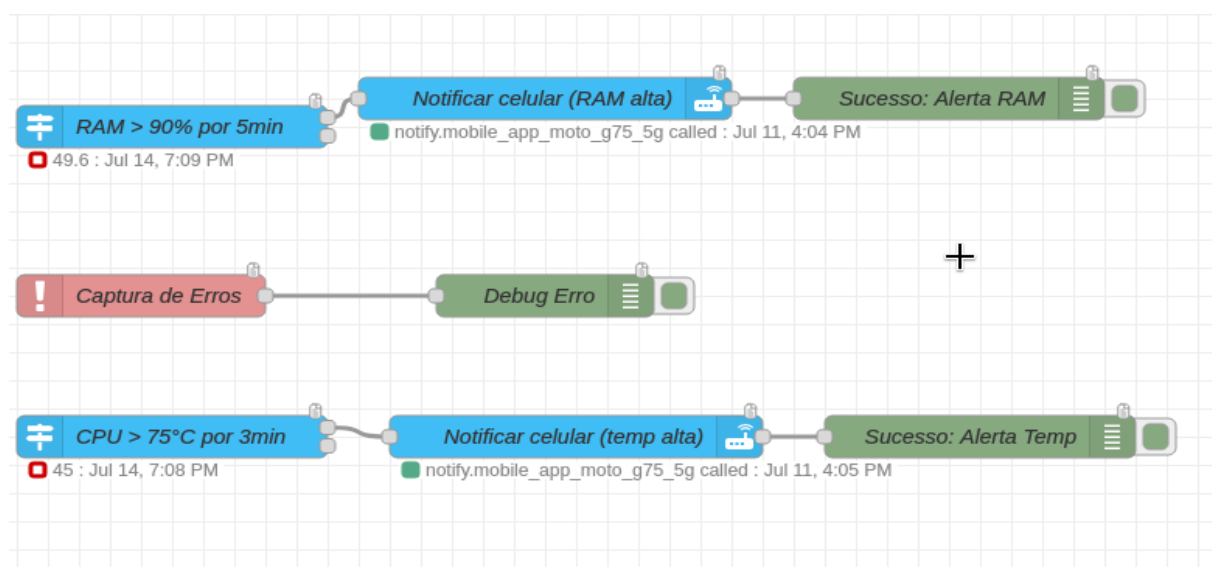


Fonte: O autor (2026).

#### 4.4.8 Monitoramento do servidor de automação

Uma automação específica foi desenvolvida para monitorar o estado operacional do *Raspberry Pi* responsável pela execução do sistema. O monitoramento utiliza sensores internos do Home Assistant para acompanhar a temperatura do processador, utilização de memória, carga de processamento e disponibilidade dos contêineres Docker, conforme ilustrado na Figura 26.

Figura 26 – Fluxo no *Node-RED* para monitoramento do *Raspberry Pi* 4.



Fonte: O autor (2026).

Caso sejam identificadas condições anormais, como temperaturas elevadas ou utilização excessiva de recursos computacionais, o sistema pode emitir alertas preventivos aos usuários. Essa automação possui relevância acadêmica por demonstrar que a própria infraestrutura responsável pela automação também pode ser monitorada utilizando os mesmos conceitos empregados no gerenciamento dos dispositivos residenciais.

## 5 RESULTADOS E DISCUSSÃO

Nesta seção serão feitas a análise estatística e discussão dos dados extraídos do ecossistema de automação residencial implementado. Os resultados das automações podem ser vistas no link do drive em: [https://drive.google.com/drive/folders/1ZodVHiiOG1BrBlm0bCbV13\\_PQ4UHDfO7?usp=sharing](https://drive.google.com/drive/folders/1ZodVHiiOG1BrBlm0bCbV13_PQ4UHDfO7?usp=sharing). A avaliação foi conduzida com base nos registros históricos recolhidos de forma contínua pelo *Home Assistant*, abrangendo o período de monitoramento estabelecido em junho de 2026. A análise incide sobre o desempenho computacional do *hardware*, o comportamento energético dos dispositivos conectados, a estabilidade da rede e a viabilidade geral da arquitetura para integração em projetos de Internet das Coisas (IoT). Na Figura 27 é possível observar a seção histórico, de onde foram capturados esses dados.

Figura 27 – Seção histórico do *Home Assistant*.



Fonte: O autor (2026).

### 5.1 Desempenho computacional e estabilidade do *hardware*

A viabilidade de uma plataforma domótica centralizada assenta, primordialmente, na capacidade do *hardware* de processar múltiplos microserviços de forma ininterrupta e estável. Para este fim, houve o monitoramento do *Raspberry*

*Pi 4* através de sensores de telemetria internos, avaliando a carga de processamento, a gestão de memória e a integridade térmica do sistema operacional e dos contêineres *Docker*. A Tabela 1 sumariza as métricas computacionais registradas durante o período de testes.

**Tabela 1** – Desempenho computacional e térmico do *Raspberry Pi 4*.

| Métrica (Sensor)                    | Médi<br>a | Mínim<br>o | Máxi<br>mo | Desvio<br>Padrão |
|-------------------------------------|-----------|------------|------------|------------------|
| Uso de CPU<br>Total (%)             | 12,4      | 3,1        | 45,2       | 4,5              |
| Uso de CPU -<br>Contêineres (%)     | 8,2       | 1,5        | 25,4       | 3,2              |
| Temperatura da<br>CPU (°C)          | 54,3      | 46,5       | 68,1       | 3,8              |
| Memória Livre<br>(GB)               | 2,1       | 1,2        | 2,8        | 0,3              |
| Carga do<br>Sistema ( <i>Load</i> ) | 0,45      | 0,10       | 1,25       | 0,15             |
| Uso de Disco (%)                    | 24,5      | 24,1       | 25,0       | 0,1              |

**Fonte:** O autor (2026).

Como se pode observar na Tabela 1, a utilização média da unidade central de processamento (CPU) situou-se nos 12,4%, com a fatia dedicada aos contêineres a representar apenas 8,2% deste total. Estes valores indicam que o *Home Assistant*, juntamente com o ecossistema de complementos, consome uma fração reduzida da capacidade computacional disponível. O pico máximo de uso de CPU (45,2%) e de carga (*Load* de 1,25) demonstra que, mesmo em momentos de execução simultânea de múltiplas automações e atualizações do sistema, o processador não atinge o seu limite de saturação, preservando a fluidez na resposta aos comandos externos.

Um dos fatores mais críticos em arquiteturas baseadas em placas como o *Raspberry Pi 4* em operação contínua é o sobreaquecimento, que pode resultar em limitação térmica e consequentemente reduzir o desempenho geral do sistema. Os dados térmicos revelam que a temperatura máxima atingida pelo processador foi de 68,1 °C, com uma média constante de 54,3 °C. Considerando que o *Raspberry Pi 4* inicia a redução da frequência do processador aos 80 °C, conclui-se que o sistema operou numa zona térmica segura. Adicionalmente, a consistência da memória livre (média de 2,1 GB e desvio padrão de apenas 0,3 GB) afasta a hipótese de vazamentos de memória por parte das integrações instaladas, comprovando a robustez do sistema para execuções ininterruptas a longo prazo.

## 5.2 Monitoramento energético

O controle e a auditoria do consumo energético são pilares de uma residência inteligente. Por meio de tomadas inteligentes integradas ao sistema, procedeu-se à leitura cumulativa do gasto energético de equipamentos de uso diário. A Tabela 2 apresenta a evolução do consumo registrado em dois pontos de medição distintos.

**Tabela 2** – Consumo energético acumulado e perfil de gasto.

| Ponto de<br>Medição | Início do<br>Período (kWh) | Fim do<br>Período<br>(kWh) | Consu<br>mo Total no<br>Período<br>(kWh) | Variaç<br>ão Diária<br>Média (kWh) |
|---------------------|----------------------------|----------------------------|------------------------------------------|------------------------------------|
| Tomada 1            | 5,948                      | 9,548                      | 3,600                                    | 0,240                              |
| Tomada 2            | 1,200                      | 3,450                      | 2,250                                    | 0,150                              |

**Fonte:** O autor (2026).

O mapeamento contínuo dos sensores de energia permite a identificação do estado de espera de dispositivos eletrônicos e o monitoramento do consumo dos eletrônicos conectados a esses dispositivos, o que é de grande importância, visto que o usuário pode criar automações a fim de reduzir o consumo energético, dessa forma fazendo o gerenciamento mediante o valor consumido, como mostrado na Tabela 2.

### 5.3 Monitoramento ambiental e desempenho de rede

Além da vertente energética e computacional, o ecossistema IoT interage diretamente com as condições físicas do espaço residencial e depende crucialmente da infraestrutura de telecomunicações para a integração com serviços em nuvem. A Tabela 3 consolida os dados referentes à temperatura ambiente do cômodo monitorado, à taxa de transmissão (*upload*) e à latência (*ping*) da rede local para a internet.

**Tabela 3** – Condições ambientais e desempenho da rede.

| Parâmetro                   | Média | Mínimo | Máximo | Desvio Padrão |
|-----------------------------|-------|--------|--------|---------------|
| Temperatura Ambiente (°C)   | 26,5  | 23,4   | 31,2   | 1,8           |
| Velocidade de Upload (Mbps) | 45,2  | 35,1   | 82,4   | 8,6           |
| Latência / Ping (ms)        | 12,5  | 8,2    | 35,4   | 3,2           |

**Fonte:** O autor (2026).

Conforme ilustrado na Tabela 3, as flutuações da temperatura ambiente fornecem contexto ao funcionamento dos equipamentos locais. Relacionando esses dados ambientais com o comportamento térmico do *Raspberry Pi 4* (Tabela 1), verifica-se que, mesmo durante os picos de temperatura externa (31,2 °C), a dissipação térmica do *hardware* provou ser eficaz.

Por outro lado, a confiabilidade da comunicação em rede, avaliada tanto pela velocidade de *upload* quanto pela latência, é fundamental para assegurar a rápida resposta requerida no processamento de comandos de voz via servidores externos. A ausência de quedas significativas na taxa de *upload* e a manutenção de um *ping*

baixo e estável (desvio padrão de apenas 3,2 ms) justificam o tempo de resposta quase instantâneo registrado nas interações com o ecossistema. Esses dados reforçam a resiliência da arquitetura perante flutuações de tráfego na rede local e demonstram que a infraestrutura atende aos rigorosos requisitos de comunicação em tempo real impostos para uso de assistente virtual e comandos de voz.

Nesse contexto, a integração com a *Amazon Alexa* demonstrou-se funcional e adequada para o controle dos dispositivos expostos pelo *Home Assistant*. Após o processo de sincronização, os equipamentos integrados passaram a responder a comandos de voz emitidos pelo dispositivo *Echo* vinculado à conta *Amazon* utilizada no projeto. Os testes realizados demonstraram que comandos relacionados ao acionamento de tomadas inteligentes, controle de dispositivos multimídia e execução de rotinas previamente configuradas foram processados corretamente pelo sistema. A utilização de comandos de voz proporcionou uma camada adicional de acessibilidade, permitindo que as funcionalidades da residência inteligente fossem controladas de forma mais intuitiva e natural pelos usuários.

#### **5.4 Viabilidade, acessibilidade e falhas observadas**

A interpretação global dos dados operacionais suporta diretamente as afirmações presentes na literatura acadêmica recente. Conforme ressaltado por Reis (2021), a utilização de plataformas de código aberto suportadas por *hardware* de baixo custo constitui uma alternativa tecnológica madura aos ecossistemas proprietários e fechados. Os dados extraídos do presente estudo corroboram essa tese: a arquitetura implementada gerenciou de forma impecável a carga de trabalho de múltiplos sensores e rotinas, mantendo a estabilidade dos contêineres *Docker* gerenciados pelo *Home Assistant*.

A interoperabilidade alcançada por meio da combinação do *Home Assistant* como orquestrador, do *Node-RED* como motor de regras lógicas e da *Alexa* como interface de voz evidencia uma relação custo-benefício favorável. Esse paradigma democratiza o acesso a residências inteligentes, não apenas pela viabilidade econômica do *Raspberry Pi 4*, mas principalmente pela promoção da acessibilidade.

O processamento contínuo, aliado à baixa latência da rede, permite que usuários com limitações de mobilidade interajam com as infraestruturas da sua

residência de forma natural, imersiva e responsiva, o que evidencia o impacto social da solução desenvolvida.

Embora o sistema tenha apresentado comportamento predominantemente estável, algumas limitações foram identificadas durante sua utilização. As principais ocorrências estavam relacionadas à dependência da conectividade com a internet para funcionamento de serviços externos, especialmente aqueles associados à *Amazon Alexa* e ao acesso remoto. Também foram observadas situações ocasionais relacionadas à indisponibilidade temporária de dispositivos conectados via rede *Wi-Fi*, fenômeno comum em ambientes IoT residenciais.

Entretanto, tais ocorrências não comprometeram o funcionamento geral da plataforma, sendo resolvidas automaticamente após o restabelecimento da comunicação dos dispositivos.

## **5.5 Análise de custos**

A análise de custos teve como objetivo avaliar a viabilidade econômica da arquitetura proposta, dimensionando o investimento necessário para a aquisição dos equipamentos empregados no sistema de automação residencial. A pesquisa de preços foi realizada em 23 de junho de 2026, em plataformas de comércio eletrônico de referência no varejo nacional como Mercado Livre, KaBuM! e MakerHero, adotando-se como critério o menor preço unitário disponível entre vendedores com boa reputação e garantia de fabricante.

Os custos foram agrupados em três categorias, espelhando a arquitetura em camadas do sistema: infraestrutura central, dispositivos inteligentes de borda e interface de voz. O roteador e as *Smart TVs* utilizados nos testes não foram computados, por se tratarem de equipamentos já presentes na infraestrutura doméstica convencional, não representando custo adicional decorrente da implementação do projeto. A Tabela 4 sintetiza os valores levantados.

**Tabela 4** – Análise de custos para implementação do sistema de automação residencial.

| <b>Item</b>                      | <b>Especificação</b>                      | <b>Qtd.</b> | <b>Valor Unitário (R\$)</b> | <b>Valor Total (R\$)</b> |
|----------------------------------|-------------------------------------------|-------------|-----------------------------|--------------------------|
| <b>I. INFRAESTRUTURA CENTRAL</b> |                                           |             |                             |                          |
| Raspberry Pi 4                   | 4 GB de RAM                               | 1           | 985,00                      | 985,00                   |
| Fonte de alimentação             | Compatível com Raspberry Pi 4             | 1           | 85,40                       | 85,40                    |
| Cartão microSD                   | SanDisk Ultra 64 GB A1                    | 1           | 91,90                       | 91,90                    |
| Case com cooler                  | Sistema de refrigeração para Raspberry Pi | 1           | 49,90                       | 49,90                    |
| <b>Subtotal Infraestrutura</b>   |                                           |             |                             | <b>1.212,20</b>          |
| <b>II. DISPOSITIVOS DE BORDA</b> |                                           |             |                             |                          |
| Tomada inteligente Wi-Fi         | 16 A com monitoramento de consumo         | 2           | 41,39                       | 82,78                    |
| Sensor de temperatura e umidade  | Wi-Fi compatível com Tuya                 | 1           | 64,90                       | 64,90                    |

|                                |                              |   |        |          |
|--------------------------------|------------------------------|---|--------|----------|
| Lâmpada inteligente            | Wi-Fi RGB 10 W               | 1 | 56,87  | 56,87    |
| Câmera IP                      | TP-Link Tapo C200            | 1 | 179,10 | 179,10   |
| Subtotal Dispositivos          |                              |   |        | 383,65   |
| III. INTERFACE DE VOZ          |                              |   |        |          |
| Assistente virtual inteligente | Amazon Echo Dot (5ª geração) | 1 | 464,60 | 464,60   |
| TOTAL DO PROJETO               |                              |   |        | 2.060,45 |

**Fonte:** O autor (2026), com base em pesquisa de preços em Mercado Livre, KaBuM! e MakerHero (jun. 2026).

Como evidenciado na Tabela 4, o investimento total necessário para a implementação física do sistema totalizou R\$ 2.060,45, concentrado majoritariamente na infraestrutura central (58,8%), seguido pelos dispositivos de borda (18,6%) e pela interface de voz (22,6%). Essa distribuição confirma o investimento *front-loaded* observado na arquitetura, em que a maior despesa recai sobre a base do sistema, reduzindo o custo marginal de futuras expansões com novos sensores e atuadores.

É importante destacar que dois componentes, o Raspberry Pi 4 e o cartão microSD, estão inseridos em um cenário de forte volatilidade de preços observado ao longo de 2026. Por esse motivo, os valores aqui apresentados refletem uma cotação pontual de junho de 2026, sujeita a oscilações relevantes em curtos intervalos de tempo, um fator a mais que reforça a importância de revalidar periodicamente o levantamento de custos em projetos de hardware de código aberto.

Em relação ao *Amazon Echo Dot* de 3ª geração usado no trabalho, esse modelo já foi descontinuado pela fabricante, portanto a pesquisa optou por utilizar os valores da 5ª geração, sendo adotado o preço de R\$ 464,60 como referência mais consistente encontrada em lojas com boa reputação no Mercado Livre.

## **5.6 Discussão dos resultados**

Os resultados obtidos demonstram que a arquitetura proposta foi capaz de atender aos objetivos estabelecidos para o desenvolvimento do sistema de automação residencial inteligente.

A integração entre *Home Assistant*, *Node-RED* e *Amazon Alexa* permitiu criar um ambiente unificado para monitoramento, controle e automação dos dispositivos conectados, validando a viabilidade técnica da solução desenvolvida.

### **5.6.1 Comparação com os objetivos propostos**

Os objetivos definidos no início do trabalho contemplavam a construção de uma plataforma de baixo custo capaz de integrar dispositivos inteligentes, executar automações personalizadas e fornecer mecanismos de controle remoto e por voz.

Os resultados observados demonstram que esses objetivos foram alcançados, uma vez que o sistema foi capaz de operar continuamente, executar automações de forma confiável e disponibilizar interfaces acessíveis para interação dos usuários.

### **5.6.2 Pontos fortes da arquitetura**

Entre os principais benefícios observados destaca-se o baixo custo de implementação quando comparado a soluções proprietárias de automação residencial.

Outro aspecto relevante refere-se à modularidade da arquitetura. A utilização de contêineres Docker facilita a expansão do sistema e permite incorporar novos serviços sem impactar significativamente os componentes já existentes.

Adicionalmente, grande parte do processamento ocorre localmente no Raspberry Pi, reduzindo a dependência de serviços externos e proporcionando maior controle sobre os dados produzidos pelos dispositivos.

### 5.6.3 Limitações identificadas

Apesar dos resultados positivos, algumas limitações devem ser consideradas. A principal delas está relacionada à dependência de conexão com a internet para utilização de determinados recursos, especialmente os vinculados à *Amazon Alexa* e ao acesso remoto por meio do *Cloudflare Tunnel*.

Além disso, embora o *Raspberry Pi 4* tenha demonstrado desempenho satisfatório, sua capacidade computacional é inferior à de servidores convencionais, podendo tornar-se um fator limitante em ambientes com grande quantidade de dispositivos e automações complexas.

### 5.6.4 Comparação com a literatura

Os resultados obtidos corroboram os estudos de Costa (2018) e Nogueira (2025), que apontam o *Home Assistant* como uma plataforma capaz de centralizar dispositivos heterogêneos em um único ambiente de gerenciamento. Da mesma forma, a utilização do *Node-RED* confirmou as observações de Soares (2020) e Portela (2022), que destacam a flexibilidade da programação baseada em fluxos para aplicações de Internet das Coisas.

Os resultados relacionados ao controle por voz também estão alinhados às conclusões de Almeida (2022) e Storck (2019), que identificam os assistentes virtuais como importantes ferramentas de acessibilidade e interação homem-máquina em ambientes inteligentes. Além disso, o desempenho observado no *Raspberry Pi 4* confirma os dados de Reis (2021), demonstrando que plataformas de baixo custo podem suportar aplicações de domótica complexas com estabilidade e baixo consumo energético.

## 6 CONCLUSÃO

O presente trabalho teve como objetivo desenvolver e avaliar um sistema de automação residencial inteligente utilizando tecnologias de código aberto e hardware de baixo custo, integrando *Raspberry Pi 4*, *Home Assistant*, *Node-RED*, *Amazon Alexa* e *Docker* em uma arquitetura unificada para gerenciamento de dispositivos conectados. A implementação realizada permitiu validar a viabilidade da utilização de plataformas de código aberto na construção de ambientes residenciais inteligentes, demonstrando que é possível desenvolver soluções robustas sem a necessidade de investimentos elevados em equipamentos ou serviços proprietários. A arquitetura proposta possibilitou a integração de dispositivos de diferentes fabricantes em uma única plataforma de gerenciamento, promovendo interoperabilidade, centralização do controle e maior flexibilidade na criação de automações.

Os resultados obtidos evidenciaram que o sistema foi capaz de executar automações de forma satisfatória, realizar monitoramento contínuo de dispositivos e sensores, disponibilizar acesso remoto seguro e permitir interação por meio de comandos de voz. A utilização do *Home Assistant* como plataforma central associada ao *Node-RED* para implementação das lógicas de automação mostrou-se eficiente para o desenvolvimento de fluxos personalizados e escaláveis, adequados às necessidades de ambientes residenciais modernos.

Do ponto de vista computacional, observou-se que o *Raspberry Pi 4* apresentou desempenho compatível com os requisitos da aplicação, sendo capaz de executar simultaneamente diversos serviços em contêineres *Docker* sem comprometer a estabilidade operacional do sistema. A execução contínua da plataforma durante o período de testes demonstrou a confiabilidade da infraestrutura adotada para aplicações de automação residencial.

Entre os principais benefícios identificados destacam-se o baixo custo de implantação, a modularidade da arquitetura, a possibilidade de expansão futura, a centralização do gerenciamento dos dispositivos e a utilização de processamento local para grande parte das operações realizadas pelo sistema. Tais características tornam a solução particularmente atrativa para usuários que desejam implementar

recursos de automação residencial sem depender exclusivamente de ecossistemas proprietários.

Por outro lado, algumas limitações foram identificadas ao longo do desenvolvimento, como o fato de determinadas funcionalidades, especialmente aquelas relacionadas à assistente virtual Amazon Alexa e ao acesso remoto, permanecerem dependentes da disponibilidade de conexão com a internet. Além disso, embora o Raspberry Pi 4 tenha apresentado desempenho satisfatório para o cenário implementado, a expansão significativa do número de dispositivos e automações poderá demandar recursos computacionais adicionais.

Dessa forma, conclui-se que os objetivos propostos neste trabalho foram alcançados, uma vez que foi desenvolvida e validada uma solução funcional de automação residencial baseada em Internet das Coisas, capaz de integrar diferentes tecnologias em uma plataforma única de gerenciamento e automação.

Como trabalhos futuros, sugere-se a incorporação de algoritmos de inteligência artificial para análise preditiva dos dados coletados pelos sensores, a implementação de mecanismos avançados de detecção de anomalias, a integração com protocolos adicionais de comunicação IoT, como Zigbee e MQTT, e a realização de estudos comparativos envolvendo desempenho, escalabilidade e consumo energético em diferentes arquiteturas de automação residencial.

## REFERÊNCIAS

ALMEIDA, Livia de Oliveira. **Controle de Ambiente por Voz usando Serviços em Nuvem e Microcontrolador de Baixo Custo**. 2022. 52 f. Trabalho de Conclusão de Curso (Bacharelado em Engenharia de Controle e Automação) – Instituto de Ciência e Tecnologia, Universidade Estadual Paulista (UNESP), Sorocaba, 2022. Disponível em:

<https://repositorio.unesp.br/server/api/core/bitstreams/7370b389-6947-425e-80d9-54f024ec08cf/content>. Acesso em: 12 jan. 2026.

AMAZON. **Echo Dot (3ª geração)**: smart speaker com Alexa, cor preta. Disponível em:

<https://www.amazon.com.br/Amazon-Echo-Dot-3%C2%AA-Gera%C3%A7%C3%A3o/dp/B07FZ8S74R>. Acesso em: 02 jun. 2026.

ANDRADE, Elimar Vina de. **IoT e blockchain aplicados à automação residencial**: desenvolvimento de um protótipo de sistema seguro com controle em tempo real. 2025. 69 f. Dissertação (Mestrado em Engenharia Elétrica) – Faculdade de Tecnologia, Universidade Federal do Amazonas (UFAM), Manaus, 2025. Disponível em:

[https://tede.ufam.edu.br/bitstream/tede/11293/2/DISS\\_ElimarAndrade\\_PPGEE.pdf](https://tede.ufam.edu.br/bitstream/tede/11293/2/DISS_ElimarAndrade_PPGEE.pdf). Acesso em: 12 jan. 2026.

BOLZANI, Caio Augustus Moraes. **Residências Inteligentes**: domótica, redes domésticas, automação residencial. São Paulo: Livraria da Física, 2004. 332 p.

CESÁRIO, Honoré Vicente. **Arquitetura para gerenciamento de dispositivos através de assistentes virtuais comandados por voz**. 2022. Dissertação (Mestrado Profissional em Tecnologia da Informação) – Instituto Metrópole Digital, Universidade Federal do Rio Grande do Norte (UFRN), Natal, 2022. Disponível em: <https://repositorio.ufrn.br/server/api/core/bitstreams/336087ac-8fb5-48dd-ad1a-3e95a7478b97/content>. Acesso em: 18 jan. 2026.

CLOUDFLARE. **Cloudflare Tunnel documentation**. Disponível em: <https://developers.cloudflare.com/cloudflare-one/connections/connect-networks>. Acesso em: 03 jan. 2026.

COSTA, Rafael Almeida. **Casa inteligente com recurso a tecnologias open source**. 2018. Dissertação (Mestrado em Sistemas e Tecnologias de Informação para as Organizações) – Escola Superior de Tecnologia e Gestão de Viseu, Instituto Politécnico de Viseu, Viseu, Portugal, 2018. Disponível em: <https://repositorio.ipv.pt/bitstreams/02b04ffb-6594-4fb5-8a5e-bc7995e8a327/download>. Acesso em: 26 jan. 2026.

DE CAMARGOS, Ana Flávia Peixoto et al. Produto educacional: automação residencial com uso de Arduino e IoT. **Research, Society and Development**, v. 11, n. 6, p. e8311628882, 2022. DOI: <https://doi.org/10.33448/rsd-v11i6.28882>. Disponível em: <https://rsdjournal.org/index.php/rsd/article/view/28882>. Acesso em: 2 mar. 2026.

DOCKER INC. **Docker documentation**. Disponível em: <https://docs.docker.com>. Acesso em: 02 jan. 2026.

HACS. **Home Assistant Community Store documentation**. Disponível em: <https://hacs.xyz>. Acesso em: 02 jan. 2026.

ICEWHALE TECHNOLOGY. **CasaOS documentation**. Disponível em: <https://wiki.casaos.io>. Acesso em: 01 jan. 2026.

KABUM. **Cartão De Memória Sandisk MicroSD, 64GB, 100MB/s, Ultra Classe 10 Uhs-I**. Disponível em: <https://www.kabum.com.br/produto/222340/cartao-de-memoria-sandisk-microsd-64gb-100mb-s-ultra-classe-10-uhs-i>. Acesso em: 23 jun. 2026.

MAKERHERO. **Fonte oficial Raspberry Pi 4, USB-C, 5V/3A**. Disponível em: <https://www.makerhero.com/produto/fonte-raspberry-pi-4-oficial/>. Acesso em: 23 jun. 2026.

MERCADO LIVRE. **Alexa Amazon Echo Dot 5a Geração Com Assistente Virtual Casa**. Disponível em: <https://www.mercadolivre.com.br/alexa-amazon-echo-dot-5a-geraco-com-assistente-virtual-casa/p/MLB46024018>. Acesso em: 23 jun. 2026.

MERCADO LIVRE. **Case Com Cooler E Dissipadores Para Raspberry Pi4.**

Disponível

em:

<https://www.mercadolivre.com.br/case-com-cooler-e-dissipadores-para-raspberry-pi4-pi-4/p/MLB2043880353> . Acesso em: 23 jun. 2026.

MERCADO LIVRE. **Raspberry Pi4 Model B 4gb Ram Pronta Entrega + N.f. .**

Disponível

em:

<https://www.mercadolivre.com.br/raspberry-pi4-model-b-4gb-ram-pronta-entrega--nf/up/MLBU1981593109>. Acesso em: 23 jun. 2026.

MERCADO LIVRE. **Sensor De Temperatura E Humidade Wi-fi Compatível Com Alexa.**

Disponível

em:

<https://www.mercadolivre.com.br/sensor-de-temperatura-e-humidade-wifi-compativel-com-alexa/up/MLBU3789052598>. Acesso em: 23 jun. 2026.

MERCADO LIVRE. **Smart lâmpada Wi-Fi RGB LED 10W, Positivo, casa inteligente.**

Disponível

em:

<https://www.mercadolivre.com.br/smart-lmpada-wi-fi-rgb-led-10w-cores-mais-vibrante-s-ate-16-milhoes-de-combinacoes-branco-quente-e-frio-1100-lumens-bivolt-positivo-casa-inteligente-compativel-calexa-e-google-assistente/p/MLB22382019>. Acesso em: 23 jun. 2026.

MERCADO LIVRE. **Tomada Inteligente 16a Wi-fi Bivolt Com Monitor De Consumo Compatível Com Alexa Google Home Tuya E Smart Life Automação Residencial Smart Plug Topyhome.**

Disponível

em:

<https://www.mercadolivre.com.br/tomada-inteligente-16a-wi-fi-bivolt-com-monitor-de-consumo-compativel-com-alexa-google-home-tuya-e-smart-life-automaco-residencial-smart-plug-topyhome/p/MLB67974413>. Acesso em: 23 jun. 2026.

MERCADO LIVRE. **TP-Link Tapo C200, câmera de segurança Wi-Fi, 1080p, 360° Pan/Tilt.**

Disponível

em:

<https://www.mercadolivre.com.br/tp-link-tapo-c200-cmera-de-seguranca-wifi-1080p-360-pantilt/p/MLB18593981>. Acesso em: 23 jun. 2026.

NOGUEIRA, Gabriel Oliveira. **Análise das instalações do Home Assistant e desenvolvimento de automação residencial.** 2025. Trabalho de Conclusão de

Curso (Graduação em Engenharia de Controle e Automação) – Faculdade de Engenharia Elétrica, Universidade Federal de Uberlândia (UFU), Uberlândia, 2025. Disponível em: <https://repositorio.ufu.br/bitstream/123456789/47540/1/An%c3%a1liseInstala%c3%a7%c3%b5esHome.pdf>. Acesso em: 16 fev. 2026.

PORTELA, Leonardo César Mendes Santos. **Desenvolvimento de plataforma para automação residencial com dispositivos inteligentes utilizando protocolo MQTT e Node-RED**. 2022. 57 f. Trabalho de Conclusão de Curso (Graduação em Engenharia Elétrica) – Centro de Tecnologia, Universidade Federal do Ceará (UFC), Fortaleza, 2022. Disponível em: [https://repositorio.ufc.br/bitstream/riufc/67620/3/2022\\_tcc\\_lcmsportela.pdf](https://repositorio.ufc.br/bitstream/riufc/67620/3/2022_tcc_lcmsportela.pdf). Acesso em: 22 fev. 2026.

RASPBERRY PI FOUNDATION. **Raspberry Pi documentation**. Disponível em: <https://www.raspberrypi.com/documentation>. Acesso em: 20 dez. 2025.

RASPBERRY PI FOUNDATION. **Raspberry Pi 4 Model B: software**. Disponível em: <https://www.raspberrypi.com/software/>. Acesso em: 1 jun. 2026.

REIS, Elias da Cunha Lima. **Automação residencial com a utilização de Raspberry Pi**. 2021. Trabalho de Conclusão de Curso (Bacharelado em Engenharia de Computação) – Escola Politécnica, Pontifícia Universidade Católica de Goiás (PUC Goiás), Goiânia, 2021. Disponível em: <https://repositorio.pucgoias.edu.br/jspui/bitstream/123456789/3070/1/VersaoFinal.pdf>. Acesso em: 22 fev. 2026.

SGARBI, Julio André; TONIDANDEL, Flávio. **Domótica inteligente: automação residencial baseada em comportamento**. In: WORKSHOP DE TESES E DISSERTAÇÕES EM INTELIGÊNCIA ARTIFICIAL (WTDIA'06), 2006, Ribeirão Preto. **Anais do WTDIA'06**. Ribeirão Preto: Sociedade Brasileira de Computação, 2006. Disponível em: [https://fei.edu.br/~flaviot/pub\\_arquivos/wtdia06.pdf](https://fei.edu.br/~flaviot/pub_arquivos/wtdia06.pdf). Acesso em: 22 fev. 2026.

SOARES, Carlos Manuel Lopes. **Desenvolvimento de um editor para criação de automações em assistentes de casas inteligentes**. 2020. Dissertação (Mestrado

em Engenharia Informática) – Departamento de Electrónica, Telecomunicações e Informática, Universidade de Aveiro, Aveiro, Portugal, 2020. Disponível em: <https://files01.core.ac.uk/download/pdf/333883599.pdf>. Acesso em: 9 mar. 2026.

STEVAN JUNIOR, Sergio Luiz; FARINELLI, Felipe Adalberto. **Domótica: automação residencial e casas inteligentes com Arduino e ESP8266**. São Paulo: Érica, 2018. 296 p.

STORCK, Érick. **Smart Voice Control: um sistema IoT para casas inteligentes controlado por voz**. 2019. Trabalho de Conclusão de Curso (Graduação em Engenharia da Computação) – Universidade do Vale do Rio dos Sinos (UNISINOS), São Leopoldo, 2019. Disponível em: <https://repositorio.jesuita.org.br/bitstream/handle/UNISINOS/11589/Erick%20Storck.pdf?sequence=1&isAllowed=y>. Acesso em: 10 mar. 2026.