

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

Diseño de un sistema domótico centralizado con comunicación inalámbrica mediante el uso de sistemas embebidos, con pantalla de operador y tecnología IoT, para el proyecto de laboratorio de domótica e Inmótica de la FIEC

PROYECTO INTEGRADOR

Previo la obtención del Título de:

Ingeniero en Electrónica y automatización

Presentado por:

Limberg Javier Angulo Casanova

Kevin Ricardo Rivera Moreira

GUAYAQUIL - ECUADOR

Año: 2023

DEDICATORIA

El presente proyecto de grado se lo dedico a toda mi familia. Principalmente, a mis padres, Edison Rivera Morán y Piedad Moreira Zambrano que me apoyaron y contuvieron los momentos buenos y malos en todo mi proceso académico. Gracias por enseñarme a cómo enfrentar los desafíos sin ceder a la presión ni rendirme en el camino.

He aprendido a convertirme en la persona que soy en la actualidad gracias a las enseñanzas que he recibido: mis convicciones, mis éticas, mi tenacidad y mi dedicación, todo esto acompañado por una generosa dosis de afecto y sin esperar nada en retorno.

También quiero dedicarle este trabajo a mis hermanos, abuelos y amigos con quienes conté con su respaldo y confiaron plenamente en ofrecer recomendaciones para concluir este trayecto hacia la excelencia.

-Kevin Rivera Moreira

DEDICATORIA

Este proyecto de grado va dedicado a todo aquel que creyó en mí y a su vez a todo aquel que no lo hizo. A mis sobrinos, para que vean que para todo hay tiempo, y deseo que sean mejores que yo.

-Limberg Angulo Casanova

AGRADECIMIENTOS

Quiero empezar agradeciendo a mi madre Piedad Moreira Zambrano y a mi padre Edison Rivera Morán quienes han sido el pilar fundamental para este gran logro en mi formación profesional. De igual manera a mis hermanos Edison, Jefferson, Yamilex y Salome que de alguna u otra manera me han apoyado para llegar a este objetivo tan anhelado.

También quiero agradecer a mis amigos universitarios en especial a Ernesto Jurado y Julio Bejarano por siempre ayudarnos en todo el recorrido estudiantil. De la misma forma agradecer a mi pareja Josselyn Tapia quien me ayudo en las últimas instancias de la titulación con un soporte emocional incondicional.

Finalmente, quiero agradecer a mi compañero de tesis, tutor y profesores quienes conforman la ESPOL por brindar lo mejor para la excelencia académica.

Siempre Gracias

-Kevin Rivera Moreira

AGRADECIMIENTOS

Doy gracias a Dios porque Él es quien permite todo lo que pasa en mi vida. Agradezco a las mujeres de mi vida, mi madre Janeth, mis hermanas Dina y Narcisa. A mis referentes, mi padre Limberg, mis cuñados Edwin y Roberto. También agradezco a mi primo Freddy que más que un primo es mi hermano. Todos han formado parte de este proceso y por el apoyo de todos ellos he llegado hasta aquí.

Finalmente agradezco a mi compañero de tesis y a mi tutor con quienes fue un gusto trabajar en este proyecto.

-Limberg Angulo Casanova

DECLARACIÓN EXPRESA

“Los derechos de titularidad y explotación, nos corresponde conforme al reglamento de propiedad intelectual de la institución; *Limberg Angulo Casanova, Kevin Rivera Moreira* y damos nuestro consentimiento para que la ESPOL realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual”



Limberg Javier Angulo Casanova



Kevin Ricardo Rivera Moreira

EVALUADORES

.....
Ing. Efrén Vinicio Herrera Muentes, Ph.D.

PROFESOR DE LA MATERIA



.....
Msc. Alexander Prieto

PROFESOR TUTOR

RESUMEN

Este proyecto se centra en diseñar y desarrollar un sistema domótico integral que permita la automatización y el control inteligente de diversos dispositivos y sistemas en un hogar a través de una pantalla de operador, para ello se usan los conocimientos en microcontroladores y comunicaciones, ya que parte fundamental del sistema es la comunicación inalámbrica a través de Wi-Fi.

Para el desarrollo del sistema domótico centralizado se utilizaron dos microcontroladores que son la Raspberry pi 4 cuya función fue de ser el servidor donde se aloja el software Home Assistant y el otro microcontrolador fue la ESP32 que era la encargada de administrar los sensores y actuadores, es decir ambas tarjetas se comunican inalámbricamente vía Wi-Fi para lograr el funcionamiento del sistema.

Con esa metodología se diseñó el sistema con una pantalla táctil que contiene la interfaz gráfica de control y se realizaron pruebas con sensores típicos de uso en la domótica, como resultado se obtuvieron dos tableros donde se evidencian todos los componentes del sistema y constan de automatizaciones funcionales para su demostración. Además, se integró una función de comando de voz a través de Amazon Alexa y así el sistema es más completo para el usuario.

El proyecto es aplicable no solo para el aprendizaje de los estudiantes del laboratorio de la materia de Domótica e Inmótica de FIEC, sino que es un sistema para implementación real en hogares e inmuebles ya que cumple con los estándares de otros sistemas en el mercado, siendo este una opción viable.

Palabras Clave: Domótica, Inmótica, Microcontroladores, Home Assistant

ABSTRACT

This project focuses on designing and developing a comprehensive home automation system that allows the automation and intelligent control of various devices and systems in a home through an operator screen, for which knowledge in microcontrollers and communications is used, since part of Fundamental to the system is wireless communication via Wi-Fi.

For the development of the centralized home automation system, two microcontrollers were used, which are the Raspberry pi 4, whose function was to be the server where the Home Assistant software is hosted, and the other microcontroller was the ESP32, which oversaw managing the sensors and actuators. In other words, both cards communicate wirelessly via Wi-Fi to achieve system operation.

With this methodology, the system was designed with a touch screen that contains the graphical control interface and tests were carried out with sensors typical of use in home automation, as a result, two boards were obtained where all the components of the system are evidenced and consist of functional automations. for your demonstration. In addition, a voice command function was integrated through Amazon Alexa and thus the system is more complete for the user.

The project is applicable not only for the learning of the students of the FIEC Domotics and Inmotics laboratory, but it is a system for real implementation in homes and buildings since it complies with the standards of other systems in the market, being This is a viable option.

Keywords: home automation, building automation, microcontrollers, home assistant

ÍNDICE GENERAL

EVALUADORES	VII
RESUMEN.....	I
ABSTRACT	II
ÍNDICE GENERAL.....	III
ABREVIATURAS.....	VI
SIMBOLOGÍA.....	VII
ÍNDICE DE FIGURAS	VIII
ÍNDICE DE TABLAS	XI
CAPÍTULO 1.....	1
1. INTRODUCCIÓN	1
1.1 Descripción del problema	2
1.2 Justificación del problema.....	3
1.3 Objetivos	4
1.3.1 Objetivo General	4
1.3.2 Objetivos Específicos.....	4
1.4 Marco teórico	5
1.4.1 Estado del arte	5
1.4.2 Sistema Domótico e Inmótico Centralizado.....	6
1.4.3 Microcontroladores.....	7
1.4.4 Sensores	8
1.4.5 Actuadores	8
1.4.6 Periféricos.....	9
1.4.7 Interfaz Gráfica.....	9

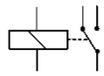
1.4.8	Redes Inalámbricas.....	10
1.4.9	Internet de las cosas	12
CAPÍTULO 2.....		13
2.	METODOLOGÍA	13
2.1	Fase de análisis de requerimientos	13
2.1.1	Centralita	13
2.1.2	Pantalla de operador	15
2.1.3	Sensores y actuadores.....	17
2.1.4	Home Assistant	25
2.2	Esquema de solución.....	25
2.3	Instalación de software	26
2.3.1	Raspberry Pi 4 con Home Assistant	26
2.3.2	ESP32 con ESPHome.....	32
2.4	Diagrama de conexiones	41
2.5	Programación en ESPHome.....	43
2.6	Dashboard en Home Assistant	49
2.7	Home Assistant en IoT.....	54
2.8	Base de Datos.....	59
CAPÍTULO 3.....		61
3.	ANÁLISIS Y RESULTADOS	61
3.1	Tableros del sistema domótico	61
3.3	Pruebas del sistema	67
3.3.1	Sensor DHT11.....	67
3.3.2	Sensor de movimiento PIR	68
3.3.3	Encendido de luz.....	70

3.3.4	Apertura de cerradura eléctrica.....	71
3.3.5	Comandos de voz con Amazon Alexa	73
3.3.6	Comando de Voz con Assist	75
3.4	Análisis de costos	77
CAPÍTULO 4.....		80
4.	CONCLUSIONES Y RECOMENDACIONES	80
4.1	Conclusiones.....	80
4.2	Recomendaciones.....	81
Referencias.....		82
Libros Físicos		82
Artículos en Línea		83
Papers académicos.....		83
Anexos		84

ABREVIATURAS

ESPOL	Escuela Superior Politécnica del Litoral
FIEC	Facultad de ingeniería en electricidad y computación
IoT	Internet de las cosas
HA	Home Assistant
PIR	Passive Infra Red - sensor infrarrojo pasivo
Wi-Fi	Wireless Fidelity – Fidelidad Inalámbrica
IEEE	Instituto de Ingenieros Eléctricos y Electrónicos
LDR	Resistor dependiente de luz
CPU	Unidad central de procesamiento
ADC	Convertidor analógico digital
DAC	Convertidor digital analógico
RAM	Memoria de acceso Aleatorio
ROM	Memoria de solo lectura
E/S	Entradas/Salidas
SoC	Centro de operaciones de seguridad
IDE	Entorno de Desarrollo Integrado

SIMBOLOGÍA



Relé doble contacto



Movimiento



Buzzer



Temperatura



Humedad



Casa Inteligente



Microcontrolador



Cerradura Eléctrica



Pantalla Táctil

cm Centímetro

m Metro

A Amperio

mA Miliamperio

dB Decibelio

V Voltios

VCC Voltaje de Corriente Continua

GND Ground – Tierra

S+ Señal Positiva

S- Señal Negativa

GHz Gigahertz

ÍNDICE DE FIGURAS

<i>Figura 2.1 Raspberry Pi 4 Modelo B</i>	14
<i>Figura 2.2 Raspberry Pi 4 Modelo B Esquema de Pines</i>	15
<i>Figura 2.3 Tablet YOBANSE Tab 7</i>	16
<i>Figura 2.4 Microcontrolador ESP32</i>	17
<i>Figura 2.5 Esquema de Pines del Microcontrolador ESP32</i>	19
<i>Figura 2.6 Microcontrolador ESP 32 con Soporte de Batería</i>	20
<i>Figura 2.7 Módulo del Sensor PIR</i>	20
<i>Figura 2.8 Composición del Modulo del Sensor PIR</i>	21
<i>Figura 2.9 Módulo del sensor DHT11</i>	22
<i>Figura 2.10 Módulo Relé 5VDC</i>	23
<i>Figura 2.11 Módulo buzzer pasivo</i>	24
<i>Figura 2.12 Logo de ESPHome</i>	25
<i>Figura 2.13 Esquema de Solución</i>	26
<i>Figura 2.14 Raspberry Pi Imager</i>	28
<i>Figura 2.15 Estructura de Docker</i>	29
<i>Figura 2.16 Lista de dispositivos inalámbricos conectados en la configuración del Router</i>	31
<i>Figura 2.17 Página de inicio de sesión de Home Assistant</i>	31
<i>Figura 2.18 Página de tienda de complementos</i>	32
<i>Figura 2.19 Configuración de ESPHome</i>	32
<i>Figura 2.20 Botón de añadir nuevo dispositivo en ESPHome</i>	33
<i>Figura 2.21 Información de Instalación de un nuevo dispositivo</i>	34
<i>Figura 2.22 Nombre del dispositivo que se va a agregar</i>	34
<i>Figura 2.23 Selección de dispositivo</i>	35
<i>Figura 2.24 Código de Encriptación</i>	35
<i>Figura 2.25 Instalar esp32.yaml Paso 1</i>	36
<i>Figura 2.26 Instalar esp32.yaml Paso 2</i>	37
<i>Figura 2.27 Instalar esp32.yaml Paso 3</i>	38
<i>Figura 2.28 Instalar esp32.yaml Paso 4</i>	38
<i>Figura 2.29 Instalar esp32.yaml Paso 5</i>	39

<i>Figura 2.30 Instalar esp32.yaml Paso 6</i>	<i>39</i>
<i>Figura 2.31 Conexión Wi-Fi</i>	<i>39</i>
<i>Figura 2.32 Credenciales de la red Wi-Fi</i>	<i>40</i>
<i>Figura 2.33 Finalización de la conexión Wi-Fi</i>	<i>40</i>
<i>Figura 2.34 ESPHome con el nodo creado</i>	<i>41</i>
<i>Figura 2.35 Diagrama de conexiones de los sensores y actuadores a la ESP32.....</i>	<i>42</i>
<i>Figura 2.36 Diagrama de flujo sensor PIR y buzzer</i>	<i>44</i>
<i>Figura 2.37 Diagrama de flujo relé y foco.....</i>	<i>44</i>
<i>Figura 2.38 Diagrama de flujo relé y cerrojo eléctrico</i>	<i>44</i>
<i>Figura 2.39 Diagrama de flujo sensor DHT11</i>	<i>45</i>
<i>Figura 2.40 Entrar al IDE de ESPHome en Home Assistant.....</i>	<i>45</i>
<i>Figura 2.41 IDE de ESPHome en Home Assistant.....</i>	<i>48</i>
<i>Figura 2.42 Compilación de la programación</i>	<i>49</i>
<i>Figura 2.43 Panel inicial en Home Assistant</i>	<i>50</i>
<i>Figura 2.44 Lista de Entidades</i>	<i>51</i>
<i>Figura 2.45 Tarjeta de Temperatura</i>	<i>52</i>
<i>Figura 2.46 Tarjeta de Humedad</i>	<i>53</i>
<i>Figura 2.47 Home Assistant Cloud</i>	<i>54</i>
<i>Figura 2.48 Ventana de IoT en Home Assistant.....</i>	<i>55</i>
<i>Figura 2.49 Integración Amazon Alexa.....</i>	<i>56</i>
<i>Figura 2.50 Ventana de Integración Amazon Alexa</i>	<i>56</i>
<i>Figura 2.51 Exponer entidades.....</i>	<i>57</i>
<i>Figura 2.52 Habilitador de Entidades.....</i>	<i>57</i>
<i>Figura 2.53 Configuración en la App de Amazon Alexa</i>	<i>58</i>
<i>Figura 2.54 Tecnologías de integración.....</i>	<i>58</i>
<i>Figura 2.55 Ejemplo de visualizador de base de datos SQLite.....</i>	<i>59</i>
<i>Figura 3.1 Tableros del sistema domótico</i>	<i>61</i>
<i>Figura 3.2 Tablero # 1</i>	<i>62</i>
<i>Figura 3.3 Tablero # 2</i>	<i>63</i>
<i>Figura 3.4 Vista Posterior del Tablero # 1</i>	<i>64</i>
<i>Figura 3.5 Vista Posterior del Tablero # 2</i>	<i>64</i>

<i>Figura 3.6 Credenciales del Sistema</i>	<i>65</i>
<i>Figura 3.7 Interfaz gráfica del sistema.....</i>	<i>66</i>
<i>Figura 3.8 Pantalla táctil con interfaz gráfica del sistema.....</i>	<i>66</i>
<i>Figura 3.9 Valor actual e Histórico de Temperatura</i>	<i>67</i>
<i>Figura 3.10 Valor actual e Histórico de Humedad.....</i>	<i>67</i>
<i>Figura 3.11 Temperatura y Humedad sensada desde la consola de ESPHome Logs..</i>	<i>68</i>
<i>Figura 3.12 Panel con sensor PIR activado.....</i>	<i>68</i>
<i>Figura 3.13 Pantalla con buzzer activado.....</i>	<i>69</i>
<i>Figura 3.14 Notificaciones de Alerta de Movimiento</i>	<i>69</i>
<i>Figura 3.15 Panel con Luz Activada</i>	<i>70</i>
<i>Figura 3.16 Pantalla con Luz Activada</i>	<i>70</i>
<i>Figura 3.17 Luz Apagada.....</i>	<i>71</i>
<i>Figura 3.18 Luz Encendida</i>	<i>71</i>
<i>Figura 3.19 Panel con Cerradura Activada.....</i>	<i>71</i>
<i>Figura 3.20 Pantalla con Cerradura Activada.....</i>	<i>72</i>
<i>Figura 3.21 Cerradura Cerrada.....</i>	<i>72</i>
<i>Figura 3.22 Cerradura Abierta</i>	<i>72</i>
<i>Figura 3.23 Echo Dot 3 en Espera.....</i>	<i>73</i>
<i>Figura 3.24 Encendido de Luz con Amazon Alexa.....</i>	<i>73</i>
<i>Figura 3.25 Apertura de cerradura con Amazon Alexa</i>	<i>74</i>
<i>Figura 3.26 Encendido de Sirena con Amazon Alexa</i>	<i>75</i>
<i>Figura 3.27 Comando de Voz con Assist.....</i>	<i>76</i>

ÍNDICE DE TABLAS

<i>Tabla 2.1 Características de la Raspberry Pi 4</i>	14
<i>Tabla 2.2 Características Tablet YOBANSE Tab 7</i>	16
<i>Tabla 2.3 Características del Microcontrolador ESP32</i>	17
<i>Tabla 2.4 Características del Sensor DHT11</i>	22
<i>Tabla 2.5 Características del Módulo Relé</i>	23
<i>Tabla 2.6 Características Buzzer Pasivo</i>	24
<i>Tabla 2.7 Tabla de asignaciones de pines de la ESP32</i>	43
<i>Tabla 3.1 Costos del Sistema Domótico Centralizado</i>	77
<i>Tabla 3.2 Costos de dispositivos domóticos en el mercado</i>	78

CAPÍTULO 1

1. INTRODUCCIÓN

Los sistemas domóticos son un campo que ha experimentado un crecimiento significativo y se ha transformado en una parte integral de casas e inmuebles debido a las ventajas que ofrece al usuario como son: seguridad, comodidad e incluso ahorro energético.

Una de las tendencias más importantes en la domótica actual es la integración de dispositivos y sistemas a través de plataformas y protocolos de comunicación estandarizados. Esto permite que diferentes dispositivos de distintos fabricantes se conecten y funcionen juntos de manera más fluida.

Algunas conectividades inalámbricas, como Wi-Fi, Bluetooth y Zigbee, también permiten rapidez y sencillez. Su principal ventaja es la movilidad de los dispositivos sin necesidad de modificar sus conexiones, además de ser muy extendidas lo cual lo hace muy asequible y económico. Esto significa que los usuarios pueden controlar y monitorear sus hogares desde cualquier red conectada a internet, incluidos sus Smartphone o tabletas, incluso cuando no están físicamente presentes.

Ahora utilizando la tecnología IoT (Internet de las cosas) tenemos la capacidad de vincular muchos de nuestros dispositivos de manera colectiva a una red en común como lo es el internet. Los sistemas embebidos no son la excepción para esta tecnología, ya que gran parte de ellos pueden conectarse a la red y enviar o recibir datos, para luego almacenar esos datos en la nube integrada al internet, y de esta forma el usuario decida qué acción realizar acorde a los datos recibidos comúnmente por sensores, a esto lo conocemos como automatización, que orientado a los hogares se transforma en domótica.

Al integrar todas las tecnologías mencionadas en una residencia la convertimos en un hogar inteligente, ya que tendremos un control de la mayoría de los dispositivos y sistemas que se encuentren en él, como son: Electrodomésticos, sistema de luces, calefacción, alarmas de seguridad, etc.

1.1 Descripción del problema

En la ESPOL, en la facultad de ingeniería en electricidad y computación actualmente se dicta la materia de Domótica e Inmótica como una materia de itinerario la cual no cuenta con su respectivo laboratorio, pero se encuentra en proyecto su desarrollo. Esta materia se ha venido actualizando en los últimos años para tener una mayor cantidad de actividades prácticas, por lo que se requieren prototipos de aplicaciones domóticas reales para el aprendizaje efectivo de los estudiantes, dado que la materia cuenta con pocas horas de clases a la semana y que uno de los aspectos fundamentales para el estudio y la comprensión de la domótica es la experiencia práctica en un entorno de laboratorio adecuado.

Para poder implementar el proyecto de laboratorio de Domótica e Inmótica de la FIEC se requieren diseños domóticos reales implementados que cubran la demostración de la mayoría de los temas abarcados en el curso, como sensores, actuadores, interfaces y comunicaciones. Esto representa una limitación significativa para los estudiantes, ya que impide que adquieran habilidades prácticas y conocimientos sólidos sobre la implementación de sistemas domóticos.

Una de las aplicaciones típicas de sistemas domóticos propietario es el uso de centralitas que utilicen la red Wi-Fi de la propia casa para las comunicaciones de la red de control de los dispositivos en la vivienda. A pesar de ser un sistema clásico y muy comúnmente usado no se posee ninguno en la FIEC puesto que la mayoría son de alto costo.

1.2 Justificación del problema

La implementación de este proyecto es ideal para la demostración y aprendizaje de los estudiantes dentro del laboratorio de la materia de Domótica e Inmótica ya que se podrá observar de manera física el diseño y funcionamiento de un sistema automatizado orientado a infraestructuras como casas, locales u oficinas.

El proyecto proporcionaría a los estudiantes un entorno de aprendizaje en el que puedan experimentar y practicar con tecnologías y dispositivos domóticos reales. Esto les permitiría abordar los diferentes aspectos de la domótica de manera integrada y permitirá ir adquiriendo habilidades prácticas que serán fundamentales para su formación como ingenieros en el campo de la automatización y el control residencial.

Utilizando sistemas embebidos como Raspberry, ESP32 y una pantalla de operador se puede lograr implementar una centralita domótica típica con comunicación inalámbrica y tecnología IoT que sea más económica que los sistemas propietarios comercializados, además de que se pueda reprogramar completamente, sirviendo como prototipo demostrativo de varias materias, principalmente la de Domótica e Inmótica.

1.3 Objetivos

1.3.1 Objetivo General

Diseñar un sistema domótico centralizado con comunicación Wi-Fi mediante el uso de sistemas embebidos, una pantalla de operador y tecnología IoT, con el fin del desarrollo de un prototipo que pueda proporcionar a los estudiantes de la materia de Domótica e Inmótica una experiencia práctica y efectiva.

1.3.2 Objetivos Específicos

1. Comparar las diferentes tecnologías del mercado para la selección de los componentes más idóneos.
2. Diseñar la arquitectura del sistema domótico, definiendo los componentes necesarios, su interconexión y las funciones específicas que cada uno debe cumplir para lograr una integración efectiva y un control centralizado, incluyendo la comunicación inalámbrica y tecnología IoT.
3. Configurar los diferentes sensores y actuadores en el sistema domótico, asegurando su correcto funcionamiento y compatibilidad con la arquitectura diseñada.
4. Elaborar una centralita domótica con un interfaz de usuario intuitiva y amigable, que permita a los usuarios controlar y monitorear las funciones del sistema domótico de manera sencilla y eficiente desarrollando IoT.
5. Validar el sistema domótico diseñado, en un entorno de laboratorio, realizando pruebas reales y simulaciones para comprobar su desempeño, confiabilidad y eficacia.

1.4 Marco teórico

1.4.1 Estado del arte

Según (Chinmay B., Sudipto B., Animesh K., Farhad H., 2019) en su artículo Sistema Avanzado de Domótica del Hogar Utilizando Raspberry – Pi y Arduino, hace referencia que los sistemas de control centralizados pueden controlar una variedad de dispositivos y sensores del hogar, como los sistemas de iluminación, calefacción y seguridad. En este proyecto se utilizó la Raspberry Pi como servidor web incorporado para alojar un panel de control de dispositivos mediante Dataplicity. Mientras que el Arduino recibe órdenes del controlador principal para efectuar los cambios en el manejo de los actuadores. El sistema da mejor rendimiento porque se puede conectar al dispositivo sin necesidad de internet, además de que garantiza una mejor seguridad informática por su almacenamiento y manejo de los datos única y exclusivamente a través de la Raspberry Pi.

De acuerdo con (Sandesh D., Jitendra M., 2019) en su publicación Automatización del Hogar con Raspberry Pi, demuestra que el sistema se puede integrar a casas o edificios donde se requiera controlar luces, ventiladores, neveras, extractores de aire u otro tipo de electrodomésticos que estén conectados a la red automatizada, de esta forma podrán ser controlados mediante comando de voz o simplemente de forma automática. El sistema de automatización utiliza varios sensores como PIR (sensor infrarrojo pasivo), sensor de humo MQ-2, LDR, entre otros. También utiliza programación condicionada para realizar la toma de decisiones de forma automática teniendo una conexión entre Arduino y Raspberry Pi para la lectura de datos de los sensores.

Las investigaciones antes mencionadas nos deja en constancia que el proyecto es viable para casas inteligentes con el uso de Raspberry Pi, Arduino, comunicaciones inalámbricas, interfaz de usuario y tecnología IoT, donde se incorpore diferentes sensores distribuidos por todo el hogar así mismo como sus actuadores siendo estos dos últimos controlados por un mismo dispositivo, priorizando la facilidad de uso y la conservación de la energía eléctrica, además de que resulta ser de gran utilidad para los usuarios incluidos adultos mayores y con discapacidad, manejando el sistema desde una unidad de control centralizada que cuente con pantalla de operador.

1.4.2 Sistema Domótico e Inmótico Centralizado

Los sistemas domóticos centralizados comprenden un conjunto de dispositivos y tecnologías interconectados que permiten automatizar y controlar diferentes equipos que conforman el hogar, como la iluminación, la climatización, la seguridad y los electrodomésticos. Todo esto es posible gracias a la unidad central de control quien manipula todo el sistema puesto que también se lo puede monitorear mediante interfaces fáciles de entender, como paneles táctiles o dispositivos móviles. Por otra parte, la Inmótica se encarga de exactamente lo mismo, pero en edificios comerciales, industriales o institucionales (Nuno M. Garcia, 2014).

Para tener un mayor enfoque de lo que son sistemas domóticos e Inmóticos centralizados detallamos algunas referencias:

- **Control4**

Sistema domótico centralizado que ofrece soluciones para el control de iluminación, audio, video, seguridad y climatización. Permite controlar todos estos aspectos de forma integrada a través de una interfaz intuitiva.

- **Crestron**

Ofrece soluciones para la gestión centralizada de la iluminación, la climatización, los sistemas de audio y video, la seguridad y mucho más.

- **Lutron**

Conocido por sus soluciones de control de iluminación, pero también ofrece sistemas de automatización y control centralizados para otros aspectos del hogar, como la climatización y las persianas.

- **Savant**

Empresa especializada en soluciones de automatización y control doméstico. Sus sistemas centralizados permiten la gestión de iluminación, audio, video, seguridad, entretenimiento y mucho más.

Los sistemas domóticos e Inmóticos mencionados nos dan un punto de partida para la realización de nuestro sistema domótico, ya que podemos ver las fortalezas y debilidades con forme a sus antecedentes. Dichos sistemas centralizados tienen al menos una unidad de control que mayormente son basadas en microcontroladores.

1.4.3 Microcontroladores

Un microcontrolador es un circuito integrado programable que en el interior de su chip contiene una unidad central de procesamiento (CPU), memoria, periféricos de entrada/salida, temporizadores y contadores, convertidor analógico-digital (ADC) y convertidor digital-analógico (DAC) con el fin de cumplir las tareas específicas que se escriben en su programación (Ayala, 2013). Sus componentes se detallan a continuación:

1. La unidad central de procesamiento, o CPU, es el componente principal de un microcontrolador encargado de llevar a cabo las operaciones de procesamiento y control. Es responsable de ejecutar las instrucciones almacenadas en la memoria para realizar diversas tareas (Uruguay, 2017).
2. El microcontrolador está equipado con una memoria de programa (ya sea ROM o Flash) y una memoria de datos (RAM). Durante la operación, la memoria de programa guarda tanto el código del programa como las instrucciones que la CPU lleva a cabo. Paralelamente, la memoria de datos retiene variables y resultados temporales (Canarias, 2018).
3. Con el propósito de establecer conexión con el entorno externo, los microcontroladores poseen múltiples puertos E/S. Estos puertos habilitan la conexión de dispositivos externos, como sensores y actuadores, con el fin de adquirir información de entrada y regular las señales de salida (Yáñez, 2020).
4. Por lo general, los microcontroladores cuentan con temporizadores y contadores integrados, los cuales tienen la función de medir el tiempo, generar señales de sincronización y supervisar eventos que ocurren de manera periódica (Meza, 2018).
5. Los convertidores analógico-digital (ADC) y digital-analógico (DAC) son elementos que posibilitan al microcontrolador interpretar y transformar las señales analógicas provenientes de sensores en datos digitales, además de generar señales analógicas para el control de dispositivos de salida (Noriega, 2008).

Algunos ejemplos de placas de desarrollo con microcontroladores muy usados son de hardware y código abierto por lo que se mencionan a continuación.

- **Raspberry Pi4**

Una placa de desarrollo de computadora de bajo costo y alto rendimiento llamada Raspberry Pi 4 se lanzó en junio del 2019. Está diseñado para ayudar a los estudiantes a aprender informática y programación, así como para ser un dispositivo útil y poderoso en una amplia gama de proyectos de electrónica y computación (Upton E., 2020).

- **ESP32**

La tarjeta de desarrollo ESP32 integra un microcontrolador y un módulo de comunicación inalámbrica ESP32 basado en el sistema en chip (SoC). La capacidad de conectarse de manera eficiente y adaptable a Wi-Fi y Bluetooth lo convierte en una opción popular para proyectos de Internet de las cosas, automatización del hogar y desarrollo de aplicaciones embebidas (P., 2018).

- **Arduino UNO**

Una placa de desarrollo de hardware de código abierto llamada Arduino Uno utiliza un microcontrolador y el entorno de programación Arduino. Con su diseño que facilita la creación de prototipos y proyectos interactivos, permite a los usuarios controlar y monitorear fácilmente sensores, actuadores y otros dispositivos electrónicos (Banzi M., 2014).

1.4.4 Sensores

Un sensor es un artefacto que detecta y responde a estímulos físicos o químicos del entorno y convierte estas magnitudes en señales eléctricas. Los sensores son esenciales para los sistemas de medición y control porque permiten recopilar información del mundo físico y usarla para tomar decisiones en sistemas digitales, analógicos o ambos (J., 2010).

1.4.5 Actuadores

Un actuador es un mecanismo o dispositivo que transforma una señal de control en movimiento o acción. Los actuadores son esenciales para los sistemas de automatización y control porque permiten a los sistemas realizar acciones o tener efectos en el entorno físico (W., 2015).

1.4.6 Periféricos

Los periféricos son dispositivos externos que se conectan a una computadora u otro sistema informático para proporcionar funciones adicionales o facilitar la interacción del usuario con el sistema. Para cada periférico se necesita instalar un driver en el sistema informático el cual va a ser el responsable de comunicar el hardware con el software. Estos dispositivos incluyen a los teclados, ratones, impresoras, escáneres, monitores, cámaras, etc. (Tanenbaum A., 2014).

Drivers

Los controladores, o drivers, son aplicaciones de software que posibilitan la comunicación y la administración de los componentes físicos de una computadora por parte del sistema operativo. Estos controladores sirven como intermediarios entre el sistema operativo y el hardware y brindan una interfaz estándar que permite que el sistema operativo se comunice y utilice los dispositivos de manera adecuada (W. S., 2018).

Software

Los programas, datos y reglas informáticas que controlan el funcionamiento de los sistemas informáticos se denominan software. Incluye instrucciones lógicas codificadas que permiten a las computadoras realizar ciertas tareas, como procesamiento de datos, cálculos, comunicación, interfaces gráficas y control de dispositivos (I., 2016).

1.4.7 Interfaz Gráfica

Menciona la representación visual e interactiva de un programa o sistema informático que permite a los usuarios interactuar con él mediante componentes gráficos como botones, menús desplegables, ventanas y otros elementos visuales utilizando pantallas táctiles o dispositivos móviles (Tidwell, 2020). Hay varios tipos según su diseño:

- Interfaz de usuario basada en ventanas: este tipo de interfaz se compone de ventanas, botones, menús y otros elementos gráficos que el usuario puede interactuar para realizar tareas.
- Interfaz de usuario iconográfica: en esta modalidad de interfaz, los usuarios emplean iconos y elementos visuales para llevar a cabo diversas acciones mediante el uso de iconos.
- Interfaz de usuario basada en texto: este tipo de interfaz utiliza texto y comandos para interactuar con el sistema.

En la domótica lo más usual es utilizar una pantalla de más de 7 pulgadas como interfaz para el usuario, además debe estar ubicada en un lugar estratégico para la fácil accesibilidad con las redes inalámbricas y control del hogar.

1.4.8 Redes Inalámbricas

La transmisión de datos sin cables físicos entre dispositivos a través de la comunicación inalámbrica posibilita la movilidad y flexibilidad del sistema, convirtiéndose en una herramienta beneficiosa en múltiples áreas. En la actualidad cada vez es más popular debido a su facilidad de manejo con el objetivo de eliminar los cables en todas las telecomunicaciones. Existen diferentes tipos de redes inalámbricas como Wi-Fi, Bluetooth, Zigbee y otras para establecer conexiones y transmitir información de manera inalámbrica entre equipos (Stallings, 2007).

Entre las comunicaciones inalámbricas más usadas en dispositivos electrónicos orientados a la domótica, tenemos:

Wi-Fi

Se trata de la red inalámbrica más famosa debido a su conexión con Internet. Desde un punto de vista técnico, el estándar IEEE 802.11 lo define como un protocolo destinado a la comunicación con dispositivos inalámbricos habilitados para Wi-Fi, incluyendo enrutadores inalámbricos y puntos de acceso. Estos últimos son compatibles con múltiples estándares IEEE, cada uno de los cuales es una revisión aprobada en intervalos específicos. Estos estándares operan en distintas frecuencias, ofrecen

diversas capacidades de ancho de banda y admiten variados números de canales (Ohrman, 2003).

Bluetooth

El uso de esta red inalámbrica difiere del Wi-Fi, ya que opera en un rango muy limitado, permitiendo la conexión y el intercambio de datos entre dispositivos electrónicos cercanos y con conexión uno a uno. Emplea ondas de radio de baja intensidad, específicamente en la banda de frecuencia de 2.4 GHz, para establecer conexiones de comunicación y simplificar la transferencia de diversos tipos de información, como datos, audios y archivos, entre dispositivos que sean compatibles con este tipo de tecnología (Albert S. Huang, 2007).

ZigBee

Es uno de los más nuevos protocolos de comunicación inalámbrica de baja potencia y corto alcance diseñado específicamente para aplicaciones de redes de sensores y control, a pesar de esto ZigBee es ampliamente utilizado en aplicaciones de automatización del hogar debido a su pequeño tamaño y bajo costo, lo que permite que los módulos ZigBee se integren en productos de consumo y dispositivos domésticos como bombillas, electrodomésticos, alarmas y otros elementos de automatización del hogar (Elahi, 2009).

En conclusión, al comparar ZigBee con tecnologías comunes como Wi-Fi, encontramos que esta última proporciona velocidades de datos más altas. En contraste, el gasto energético y las dimensiones del hardware son significativamente amplificadas, lo cual podría plantear un problema importante en ciertas utilidades tales como la automatización del hogar o la instrumentación de mediciones. Respecto a Bluetooth Low Energy, aunque la tecnología es muy similar, la principal ventaja de ZigBee es su mayor alcance y número de dispositivos que permite la red (hasta 65.535 nodos). Lo cual es muy eficiente para integrarlo al internet de las cosas (IoT).

1.4.9 Internet de las cosas

En esencia IoT o Internet de las cosas, es la capacidad que tienen los sistemas y dispositivos como sensores o actuadores de comunicarse y recopilar datos entre sí a través del internet. En la actualidad, IoT permite que los objetos que antes eran comunes se transformen en inteligentes, los ejemplos más comunes son los electrodomésticos, sistemas de luces e incluso automóviles, que conectados al internet nos abren posibilidades de control y supervisión (Kranz, 2017).

En los primeros sistemas domóticos teníamos redes cerradas o limitadas dentro del hogar, es decir todo control y supervisión se podría hacer solo desde la misma ubicación, pero con la integración de IoT esto dio un giro, pero a la vez solo surge una desventaja ya que, debido a la gran cantidad de datos transmitidos y procesados por varios servicios a través de Internet y conexiones, los mismos datos se vuelven vulnerables. Las posibilidades de utilizar estos datos para otros fines, como análisis de rutina o de comportamiento, hace que esta información se difunda permitiendo que otras personas tengan conocimiento de nuestro diario vivir.

Finalmente, podemos decir que el IoT le ha brindado una gran evolución a la domótica, y mucho más a la posibilidad de controlar y monitorear nuestro hogar desde cualquier lugar que tenga conexión a internet.

CAPÍTULO 2

2. METODOLOGÍA

2.1 Fase de análisis de requerimientos

Para cumplir con los requerimientos del sistema domótico propuesto tenemos que dar prioridad a tres componentes que son: la centralita basada en un sistema embebido o microcontrolador, la pantalla como interfaz para el panel de control y la comunicación inalámbrica hacia los sensores y actuadores además del uso de tecnología IoT.

Todos estos elementos serán integrados a través de un software orientado a la domótica llamado Home Assistant, que a su vez consta con extensiones para trabajar con el microcontrolador ESP32, esta extensión es conocida como ESPHOME.

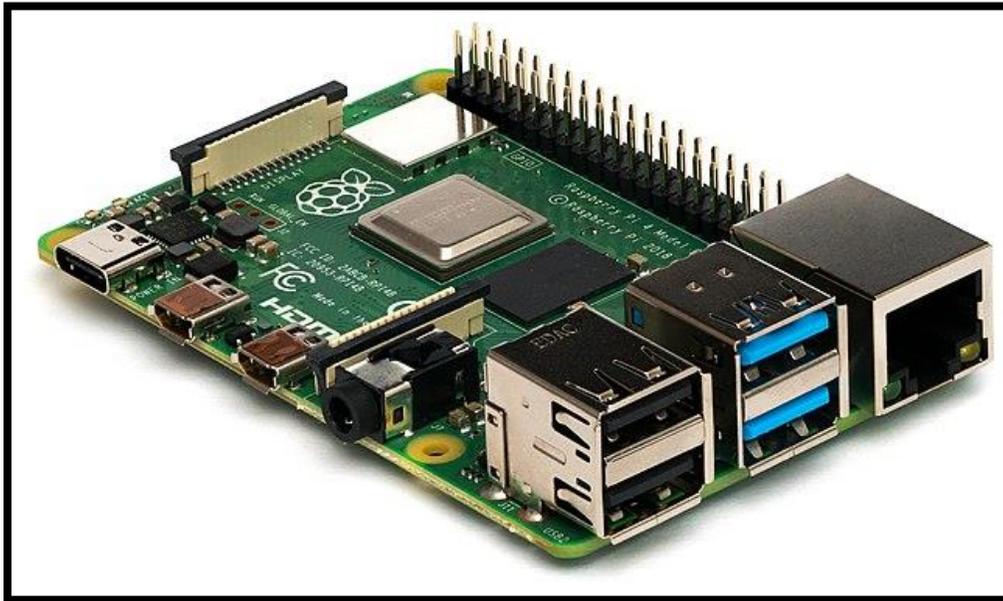
Se realizó una investigación y análisis de las tecnologías actuales en el mercado para así seleccionar las mejores opciones en cuanto a precio y características que satisfagan las necesidades del proyecto.

2.1.1 Centralita

La central estará basada en un microcontrolador, el cual debe tener buena capacidad de memoria y procesamiento de datos, además de conectividad inalámbrica. Este microcontrolador tendrá como función principal actuar como servidor, es decir recibirá y enviará datos de los dispositivos conectados en la red.

El microcontrolador seleccionado es la Raspberry Pi 4 Model B en la versión de 8GB de RAM, debido a que sus características van acorde a la función que va a ejecutar en nuestro sistema centralizado, sus características más destacables son:

Figura 2.1 Raspberry Pi 4 Modelo B



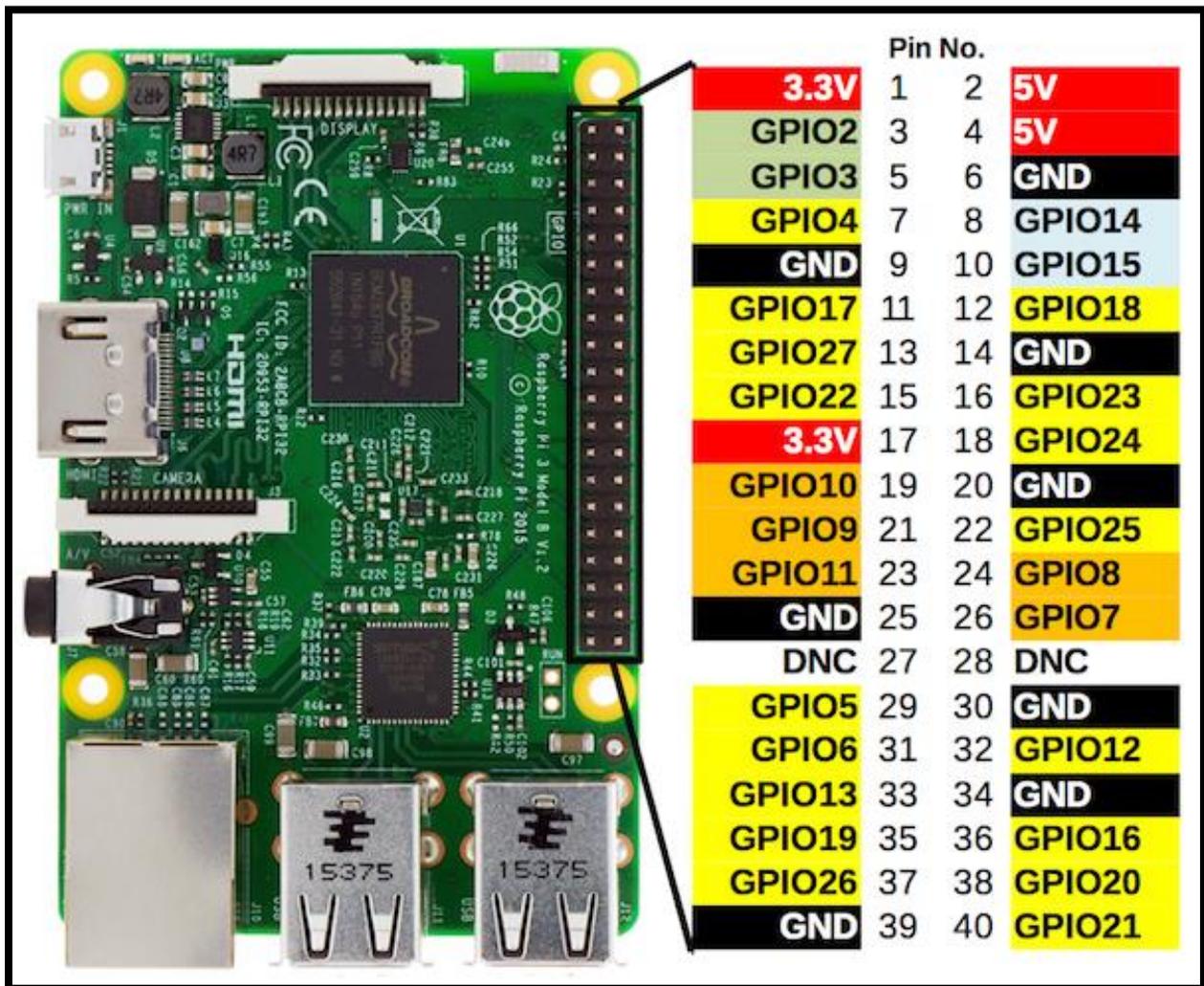
Nota. La figura 2.1 tomada de Rytronics (2023).

Tabla 2.1 Características de la Raspberry Pi 4

Procesador	ARM Cortex-A72
Frecuencia de reloj	1,5 GHz
Gpu	Unidad de procesamiento gráfico VideoCore VI (con capacidad para OpenGL ES 3.x)
Memoria	Memoria RAM LPDDR4 de 8 GB
Conectividad	Tecnologías de conectividad Bluetooth 5.0, Wi-Fi 802.11ac y Ethernet Gigabit
Puertos	<ul style="list-style-type: none"> ▪ Puede ser alimentado a través de Ethernet (PoE) ▪ Opción de alimentación mediante USB C a 5 Vcc (mínimo 3 A) ▪ Posibilidad de alimentación a 5 Vcc mediante encabezado GPIO (mínimo 3 A) ▪ Memoria RAM de 4 GB SDRAM LPDDR4-3200 ▪ Admite Wi-Fi de doble banda: 2.4 GHz / 5 GHz según el estándar IEEE 802.11.b/g/n/ac ▪ Bluetooth 5.0 BLE ▪ Puerto RJ45 para Ethernet Gigabit ▪ Dos puertos USB 3.0 y dos puertos USB 2.0 ▪ Ranura para tarjeta microSD ▪ Conector GPIO de 40 pines ▪ Salida de audio estéreo y video compuesto mediante conector de 4 polos ▪ Dos salidas micro HDMI (compatible con resolución de hasta 4K a 60 Hz)
Precio	Desde 55 dólares

Nota. La tabla 2.1 fue tomada de Raspberry Pi (2023).

Figura 2.2 Raspberry Pi 4 Modelo B Esquema de Pines



Nota. La figura 2.2 fue tomada de Raspberry Pi (2023).

2.1.2 Pantalla de operador

El sistema domótico tendrá una interfaz gráfica tipo panel, la cual debe estar en una pantalla táctil para que exista la interacción física con el usuario de manera sencilla y de rápido acceso. Se consideraron pantallas propias compatibles con el microcontrolador, a pesar de esto la opción más viable es utilizar la tecnología IoT y la comunicación inalámbrica para establecer nuestro panel en un dispositivo con pantalla como lo es una Tablet.

Esta opción tiene ventajas notables respecto a las pantallas de los microcontroladores tradicionales. Una de ellas es la posibilidad de cambiar la ubicación de la pantalla dentro

del hogar ya que su comunicación es inalámbrica, además de poder contar con más de una pantalla para acceder al panel, esto último es muy útil para casas de gran tamaño o de más de un piso. En comparación de precios también resulta más factible adquirir una Tablet que la pantalla táctil de siete pulgadas de la Raspberry, cuyo valor comienza desde los noventa dólares.

Un aspecto importante es que no requerimos de una Tablet muy avanzada para esta aplicación, ya que tendrá solo el uso de funcionar como panel, esto es viable en el ámbito económico. Debido a esto, el sistema cuenta con una Tablet marca Yobanse con las siguientes características:

Figura 2.3 Tablet YOBANSE Tab 7



Nota. La figura 2.3 fue tomada de YOBANSE (2023).

Tabla 2.2 Características Tablet YOBANSE Tab 7

Marca	YOBANSE
Nombre del modelo	TAB7
Capacidad de almacenamiento de memoria	32 GB
Tamaño de pantalla	7 pulgadas
Sistema operativo	Android 11
Precio	45 dólares

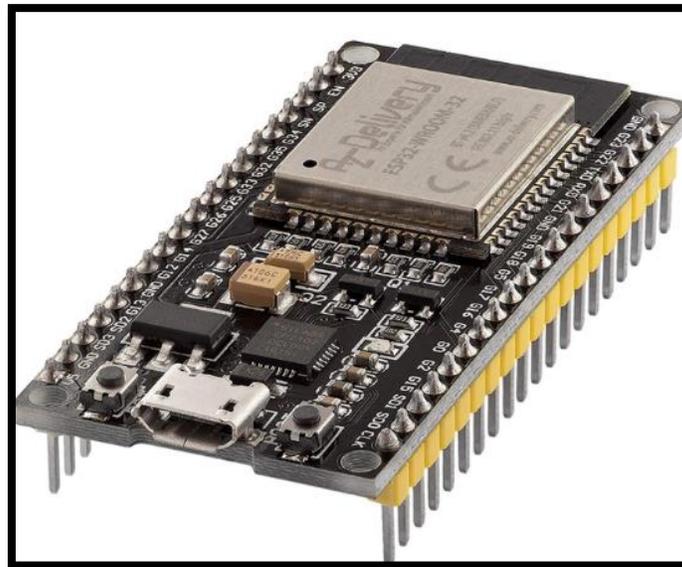
Nota. La tabla 2.2 fue tomada de YOBANSE (2023).

2.1.3 Sensores y actuadores

En esta sección se requiere integrar sensores y actuadores comandados por un microcontrolador de menor gama comparado al de la central, pero a pesar de ser de lo mencionado debe de incorporar comunicación inalámbrica Wi-Fi.

Para ello uno de los sistemas embebidos más adecuados es la ESP32 ya que además de su tamaño compacto cumple con dos de los requisitos más importantes para esta parte del sistema domótico, que son el costo y la comunicación inalámbrica. Es un microcontrolador muy potente con las siguientes características:

Figura 2.4 Microcontrolador ESP32



Nota. La figura 2.4 fue tomada de ORELLANA (2023).

Tabla 2.3 Características del Microcontrolador ESP32

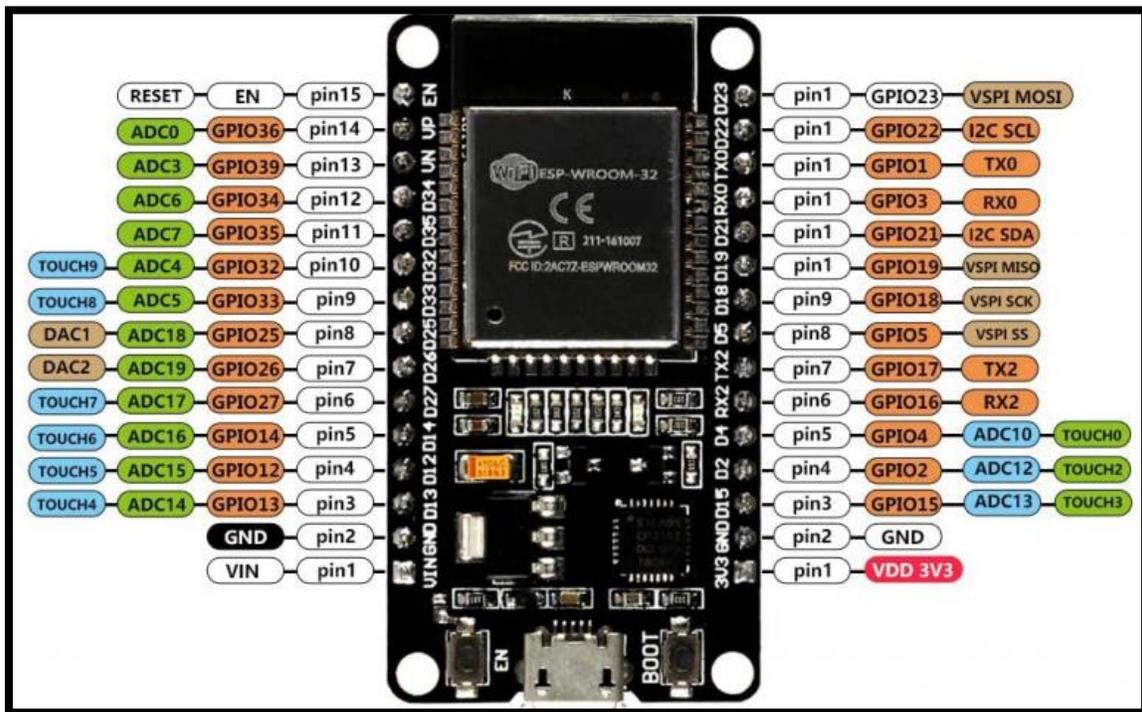
Procesador	Microprocesador de 32-bit Xtensa LX6
Memoria	520 KiB SRAM
Conectividad	Wi-Fi y Bluetooth
Interfaces periféricas	30 pines (Análogo – Digital- PWM)
Precio	12 dólares
Interfaces periféricas	<ul style="list-style-type: none">▪ Conversor Analógico-Digital SAR de 12 bits con la capacidad de hasta 18 canales.▪ Dos Convertidores Digital-Analógico (DAC) de 8 bits cada uno.▪ Conjunto de 10 sensores de tacto (sensores capacitivos utilizables como GPIOs).▪ Cuatro interfaces SPI.▪ Dos interfaces I²S.▪ Dos interfaces I²C.▪ Tres puertos UART.

	<ul style="list-style-type: none"> ▪ Controlador principal para SD/SDIO/CE-ATA/MMC/eMMC. ▪ Controlador secundario para SDIO/SPI. ▪ Interfaz Ethernet MAC con un DMA específico y apto para el protocolo IEEE 1588 Precision Time Protocol. ▪ Bus CAN 2.0. ▪ Controlador infrarrojo remoto (TX/RX) con capacidad para hasta 8 canales. ▪ Generador de señales PWM. ▪ Modulador PWM para LEDs (hasta 16 canales). ▪ Detector de efecto Hall. ▪ Preamplificador analógico de ultra baja potencia.
Administración de Energía	<ul style="list-style-type: none"> ▪ Fuente interna de regulación de voltaje con baja pérdida. ▪ Dominio de alimentación separado para el Reloj en Tiempo Real (RTC). ▪ Consumo de corriente en modo de suspensión profunda de 5µA. ▪ Activación a través de interrupción de GPIO, temporizador, lecturas del convertor Analógico-Digital (ADC) y activación del sensor de toque capacitivo.

Nota. La tabla 2.3 fue tomada de SIGMA (2023).

La ESP32 en comparación con Arduino que es un microcontrolador comúnmente usado, cuenta con varias ventajas a pesar de que ambos oscilan por los mismos precios, entre estas ventajas están: el CPU de la ESP32 es de 32 bits con doble núcleo, mientras que el del Arduino es de 8 bits con un solo núcleo. Además, la velocidad de reloj de la ESP32 va hasta los 240 MHz, es mucho mayor que la de Arduino UNO y MEGA que llegan hasta 8 MHz y 16 MHz respectivamente. Con esto podemos concluir que la ESP32 es más potente y económico que Arduino.

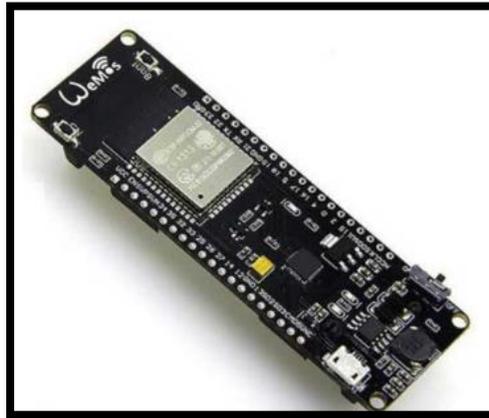
Figura 2.5 Esquema de Pines del Microcontrolador ESP32



Nota. La figura 2.5 fue tomada de SIGMA (2023).

Como fue mencionado, nuestra ESP32 será la que integre los sensores que detallaremos posteriormente, y será capaz debido a la cantidad numerosa de pines. Además, será muy factible debido al ahorro de energía ya que la ESP32 logra un consumo de energía ultra bajo con una combinación de varios tipos de software propietario como será ESPHOME y también incluye características de última generación, como activación de reloj de grano fino, varios modos de potencia y escalado dinámico de potencia. Inclusive existen versiones de este microcontrolador en el cual viene incluido un soporte para batería y de esta manera tendríamos un dispositivo autónomo.

Figura 2.6 Microcontrolador ESP 32 con Soporte de Batería



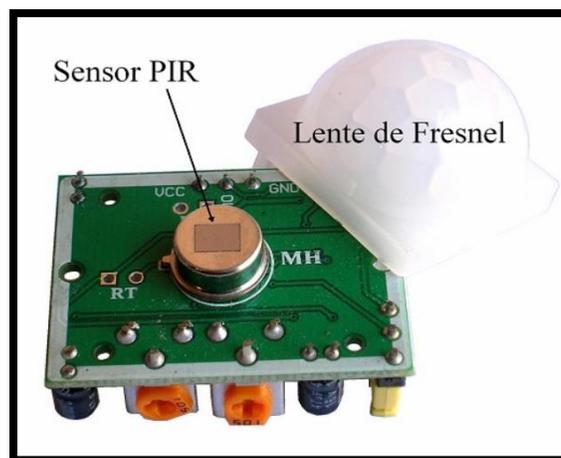
Nota. La figura 2.6 fue tomada de Mechatronic (2023)

Sensor PIR

La seguridad es un componente fundamental en el área de la domótica e Inmótica, es por esto por lo que se utilizan muchos sensores de presencia como el PIR, que es un sensor de tipo óptico que sirve para detectar en un rango determinado la presencia o movimiento de alguna persona u objeto.

El principio que utiliza este sensor es similar a los radares, pero este utiliza un lente para que recaigan los rayos incidentes sobre un punto, que consta con un cristal llamado lente de Fresnel, guiando los rayos hacia el punto donde se encuentra el sensor PIR, este a su vez mide el retardo entre dos sensores de luz para así obtener la distancia a la que se puede encontrar el objeto o persona como tal.

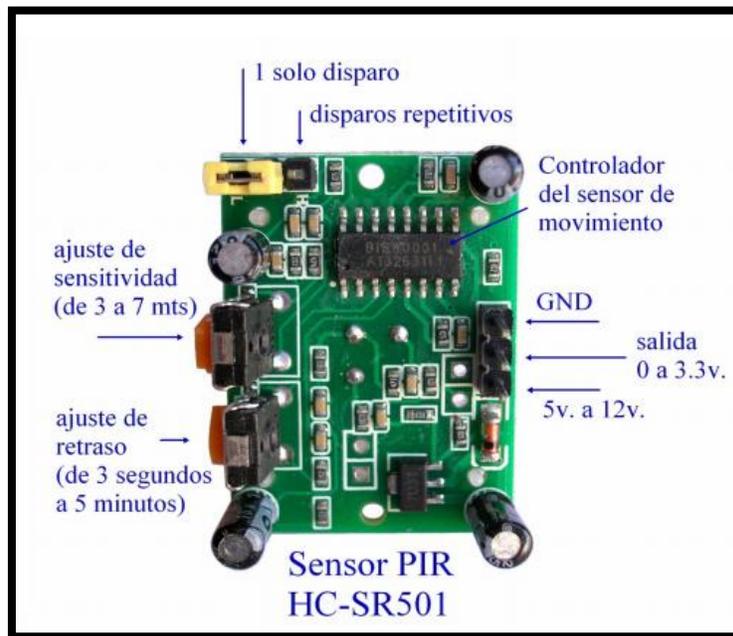
Figura 2.7 Módulo del Sensor PIR



Nota. La figura 2.7 fue tomada de 330OHM (2023).

Para trabajar con el microcontrolador ESP32 se eligió el módulo HC-SR501 ya que consta con las terminales adecuadas para la integración hacia la tarjeta controladora. Dentro del módulo tenemos la posibilidad de ajustar dos parámetros a través de dos potenciómetros, que son el nivel de sensibilidad y el tiempo de retardo, estos pueden variar de 5 a 7 metros y de 3 segundos a 5 minutos respectivamente.

Figura 2.8 Composición del Módulo del Sensor PIR



Nota. La figura 2. Fue tomada de 330OHM (2023).

Finalmente, las conexiones de entrada y salida son las de GND o tierra, la alimentación va de 5v a 12v y el pin de salida que será el que envíe el dato de activación del sensor que trabaja a 3.3v. El sensor tiene un precio en el mercado de \$ 2,50.

Sensor DHT11

El sensor digital más usado para recibir datos de temperatura y humedad en microcontroladores, debido a su simplicidad y bajo costo. Es comúnmente aplicado para el control de temperatura, ya que con los datos se puede tomar una decisión por ejemplo de encender un aire acondicionado, e incluso en otras áreas como la agricultura podría accionar un sistema de riego dependiendo del dato enviado por este sensor.

Entre las características de este sensor tenemos:

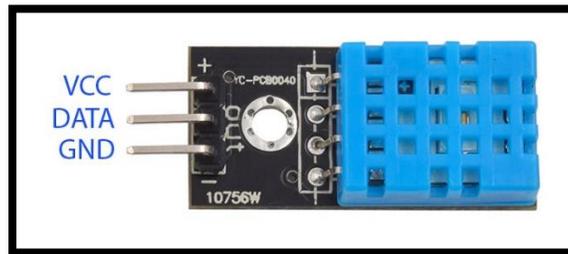
Tabla 2.4 Características del Sensor DHT11

Voltaje de funcionamiento	3V – 5V DC
Intervalo de medición de temperatura	0°C – 50°C
Exactitud de medición de temperatura	±2.0 °C
Resolución °C	0.1°C
Margen de medición de humedad	20% - 90% RH
Precisión de medición de humedad	5% RH
Resolución % RH	1% RH
Tiempo de muestreo	1 segundo
Interfaz de comunicación digital	Bus individual (bidireccional)
Modelo	DHT11
Dimensiones	16x12x5 mm
Peso	1 gramo
Material	Plástico Rígido

Nota. La tabla 2.4 fue tomada de Hardwarelibre (2023).

Los pines facilitan la integración de este sensor a nuestro microcontrolador ya que consta con cuatro terminales identificados como:

Figura 2.9 Módulo del sensor DHT11



Nota. La figura 2.9 fue tomada de TICmaker (2018)

- Alimentación: +5V (VCC)
- Señal (DATA)
- Tierra (GND)

A pesar de ser de los más básicos fue seleccionado porque trabaja dentro de los rangos normales de temperatura y humedad en una residencia, y será muy útil para integrarlo con demás dispositivos que influyen en la temperatura de un espacio en el hogar. El sensor tiene un precio en el mercado de \$ 2,2.

Relé 5VDC

El propósito del módulo de relé es cambiar la dirección de corriente de las cargas. Los puntos de conexión en el relé están concebidos para alternar corrientes de hasta 10 A y 250 V CA (o 30 V CC), aunque es aconsejable emplear repuestos que no superen estos límites. Este componente proporciona la electrónica necesaria para activar la bobina del relé. La señal de control tiene la capacidad de derivar de cualquier circuito de control TTL o CMOS, por ejemplo, un microcontrolador, como el caso de ESP32, que suministra una señal de salida de 3.3V a 5V. Las características más destacadas son:

Figura 2.10 Módulo Relé 5VDC



Nota. La figura 2.10 fue tomada de MEGATRONICA (2023).

Tabla 2.5 Características del Módulo Relé

Cantidad de canales de activación	1 canal
Tipo de Relevador	2 servicios con un solo polo
Voltaje requerido por la bobina del relevador	3.3 - 5 VDC
Indicador LED	Se ilumina cuando la bobina del relé está en funcionamiento
Consumo de corriente de activación	15 – 20 mA
Control	Se puede controlar de manera directa mediante circuitos lógicos
Conexiones	Se enlaza mediante terminales de tornillo
Tamaño de los puertos de entrada lógica	0.1 pulgadas
Carga máxima soportada	AC 250V / 10A, DC 30V / 10A
Consumo de corriente de activación	5 mA
Tamaño del módulo	46.5 x 12 x 18.5 mm (largo x ancho x alto)

DC +	Entrada de energía positiva (VCC)
DC-	Entrada de energía negativa (GND)
IN	Puede funcionar como relé de control de nivel alto o bajo
NO	Interfaz del relé normalmente abierto
COM	Punto de contacto común para los relés
NC	Interfaz del relé normalmente cerrado
Actúa como un disparador de nivel bajo cuando está conectado al pin LOW	
Actúa como un disparador de nivel alto cuando está conectado al pin HIGH	

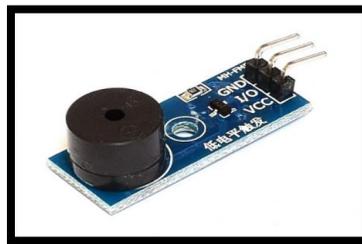
Nota. La tabla 2.5 fue tomada de SSDielect (2023).

Este dispositivo es muy usado en la domótica ya que es la conexión entre el controlador y los dispositivos que manejan potencias mayores a este, como lo son los focos, electrodomésticos, etc. El actuador tiene un precio en el mercado de \$ 2,25.

Módulo Buzzer Pasivo

De la mano de los sensores orientados a la seguridad como el sensor PIR e inclusive sensores de temperatura, van integrados los actuadores de sonido como lo es el módulo buzzer, que comúnmente se los utiliza como alarma.

Figura 2.11 Módulo buzzer pasivo



Nota. La figura 2.11 fue tomada de NAYLAMP (2023)

Tabla 2.6 Características Buzzer Pasivo

Voltaje de Operación	3.3VDC ~ 5VDC
Corriente Aproximada de consumo	30 mA
Pines	VCC, GND y Señal
Sonoridad	70 dB ~ 85 dB
Tipo de buzzer piezoeléctrico	Pasivo

Nota. La tabla 2.6 fue tomada de NAYLAMP (2023).

Dentro de nuestro sistema este módulo irá integrado al sensor de presencia PIR para simular una alarma típica de los sistemas domóticos, debido a que este módulo tiene

entre 70 y 85 decibelios que es un rango aceptable para alarmas en interiores. El módulo tiene un precio en el mercado de \$ 1,90.

2.1.4 Home Assistant

Home Assistant es un software de código abierto orientado a la automatización del hogar, y es adecuado para ejecutarse en una Raspberry pi como servidor local para comunicarse con un buen número de dispositivos en la red.

Una de las características más relevantes de Home Assistant son los Add-Ons, que son aplicaciones dentro del software que permiten integrar dispositivos de manera fácil y sacarle el mayor provecho. El Add-On que necesitamos es ESPHome, que permite crear nodos donde se conectarán las esp32 a nuestro servidor y en el mismo lugar realizar códigos para darle la aplicación que quisiéramos.

ESPHome

ESPHome es un sistema para controlar el microcontrolador ESP8266/ESP32 y RP2040 mediante archivos de configuración simples pero potentes y controlarlos de forma remota a través de sistemas de automatización del hogar.

Figura 2.12 Logo de ESPHome

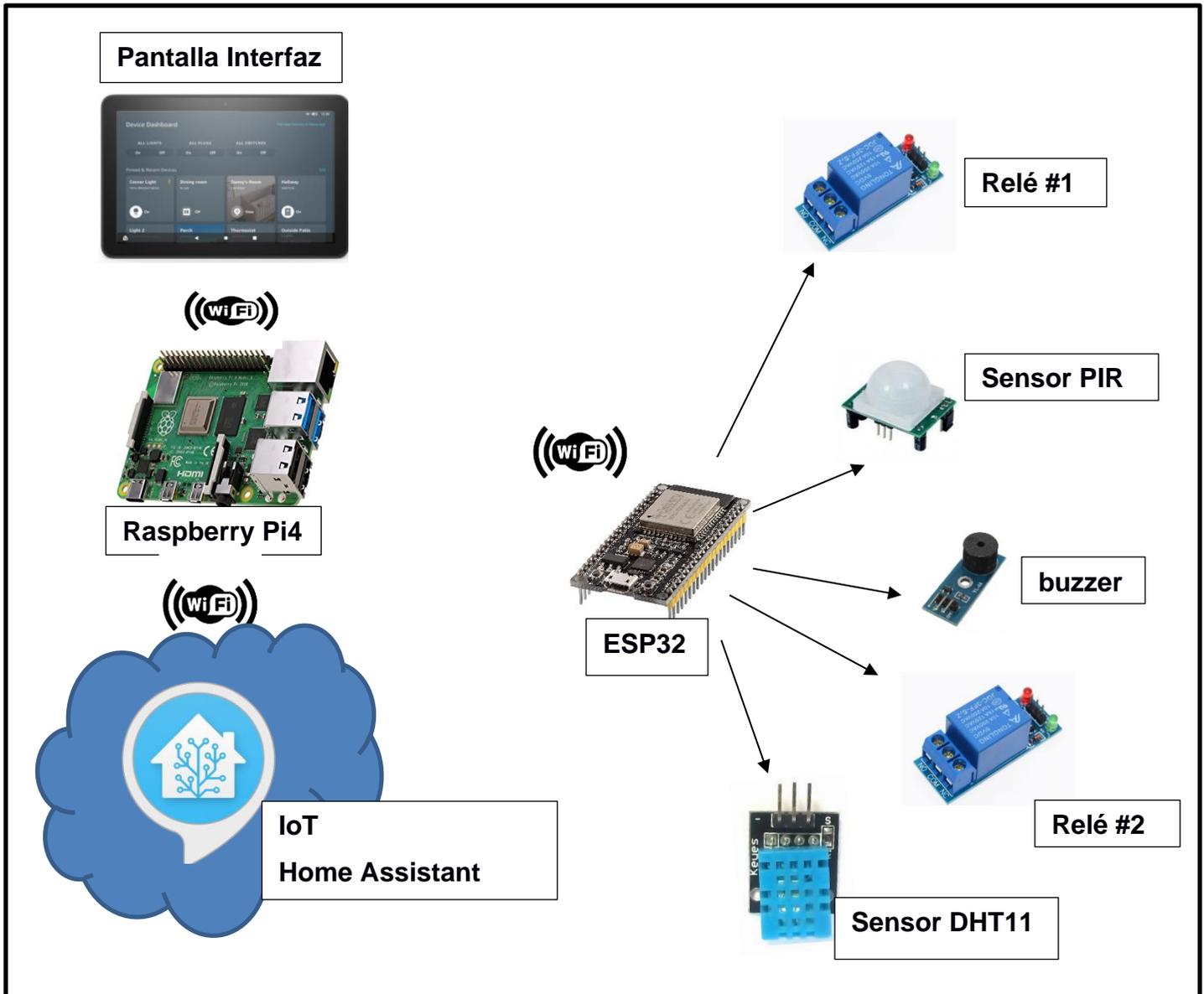


Nota. La figura 2.12 fue tomada de ESPHome (2023).

2.2 Esquema de solución

Una vez que hemos seleccionado los dispositivos que se ajustan a nuestros requerimientos, podemos formar nuestro sistema que es centralizado. De manera concreta, la Raspberry pi 4 es utilizada como la portadora de nuestro sistema operativo Home Assistant, es decir será el centro de toda la red actuando únicamente como servidor, enviando y recibiendo datos de todos los dispositivos conectados inalámbricamente por Wi-Fi.

Figura 2.13 Esquema de Solución



2.3 Instalación de software

2.3.1 Raspberry Pi 4 con Home Assistant

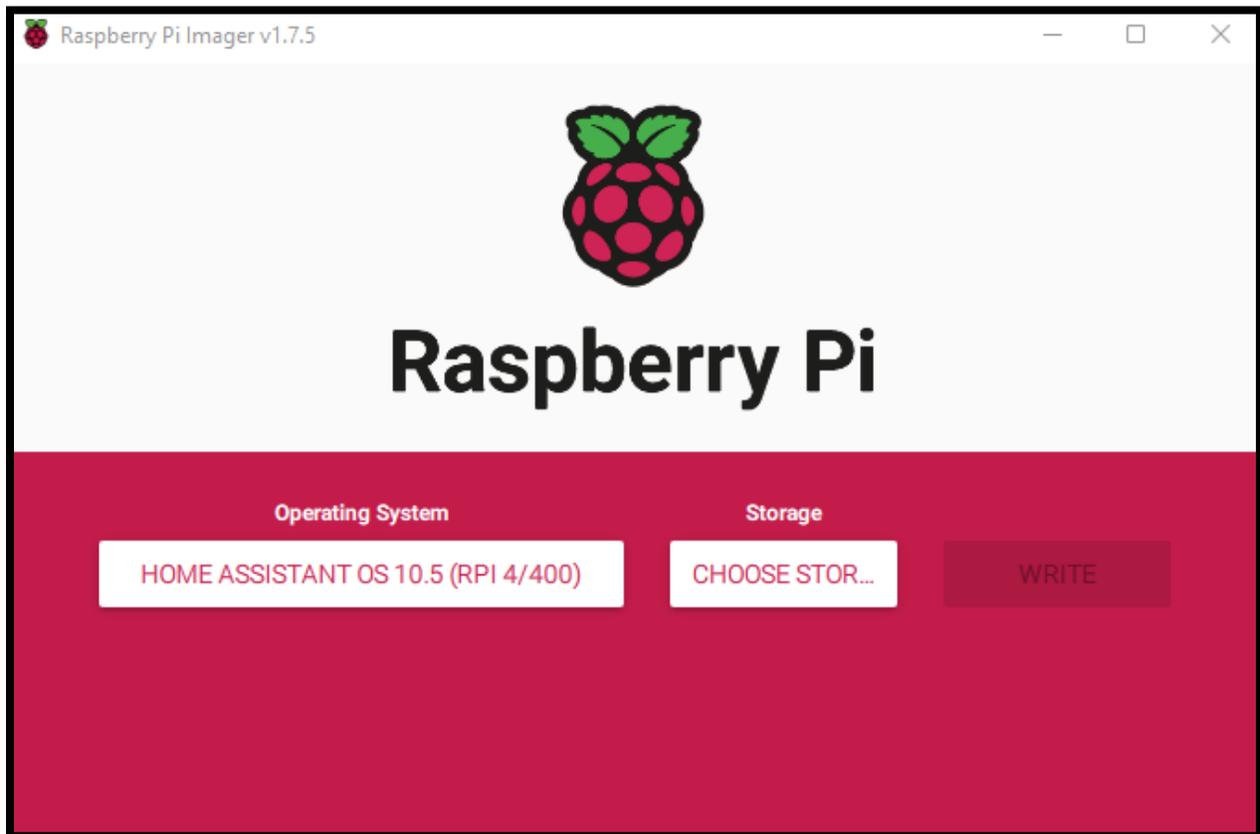
El Sistema domótico al ser centralizado necesita un servidor principal, el cual contendrá nuestro programa Home Assistant y se encarga de administrar los demás dispositivos enlazados en la misma red. La Raspberry Pi 4 es capaz de correr dicho programa por lo cual será utilizada para tal función. Existen varios procesos para la instalación de Home Assistant, de los cuales dos son aplicables en este caso, y son los siguientes:

- Sistema operativo Home Assistant: sistema operativo mínimo optimizado para potenciar Home Assistant. Viene con Supervisor para administrar Home Assistant Core y Add-ons. Método de instalación recomendado.
- Home Assistant Container: instalación independiente basada en contenedores de Home Assistant Core (por ejemplo, Docker).

Instalación de Home Assistant OS

1. Descargar e instalar Raspberry Pi Imager en la computadora como se describe en <https://www.raspberrypi.com/software/>.
2. Abrir el generador de imágenes Raspberry Pi.
3. Elegir el sistema operativo:
Seleccionar Elegir sistema operativo.
Seleccionar Otro sistema operativo específico > Asistentes domésticos y domótica > Home Assistant
Elegir el sistema operativo Home Assistant que coincida con el hardware (RPi 3 o RPi 4).
4. Elegir el almacenamiento: insertar la tarjeta SD en la computadora. Nota: se sobrescribirá el contenido de la tarjeta. Seleccionar su tarjeta SD.
5. Escribir el instalador en la tarjeta SD:
Para iniciar el proceso, seleccione Escribir.
6. Esperar a que el sistema operativo Home Assistant se escriba en la tarjeta SD.

Figura 2.14 Raspberry Pi Imager



7. Encender la Raspberry Pi, pero antes: Insertar la tarjeta SD en su Raspberry Pi.
8. Enchufar un cable Ethernet que esté conectado a la red.
9. Conectar la fuente de alimentación para poner en marcha el dispositivo.

Finalmente, ya tendremos instalado Home Assistant y podemos ir al navegador del sistema de escritorio, y al esperar unos minutos podremos acceder a nuestro Home Assistant en `homeassistant.local:8123`.

Instalación de Home Assistant container

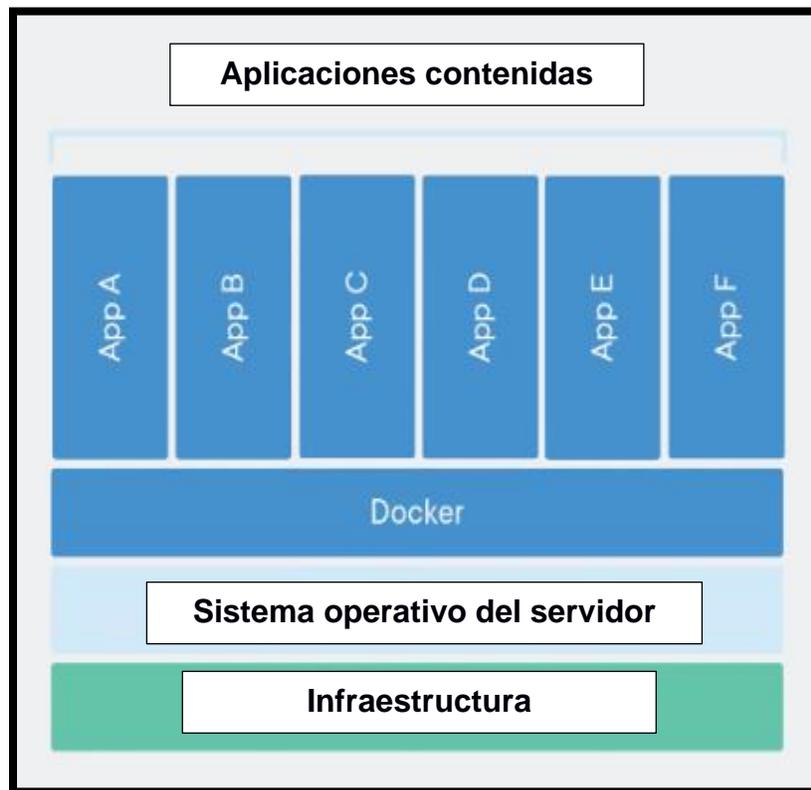
Este método de instalación se puede usar para tener las demás funciones de la Raspberry, es decir tener nuestro sistema operativo raspbian funcionando y correr varios programas a la vez por medio de contenedores, antes de eso tenemos que conocer que es Docker.

Docker

Es una tecnología que permite ejecutar un proceso de una manera totalmente aislada del resto de los procesos de la máquina y también permite que el proceso que se está ejecutando defina todas las dependencias que necesita para ser ejecutadas, es decir, cuando se cree el proyecto se va a requerir muchas dependencias o una herramienta pero que solo funcionan en una distribución concreta de Linux, entonces esta distribución que necesita para funcionar, las dependencias que necesita del sistema operativo para ejecutar y las dependencias del lenguaje de programación que necesita esa herramienta para funcionar con Docker se va a conseguir empaquetar en una imagen de tal manera que toda la información esté disponible al momento de ejecutar Docker.

Con esto se consigue que el programa empaquetado sea totalmente portable para diferentes tipos de sistemas operativos como, IOS, Ubuntu, Red Hat Os, etc y en cualquier máquina con estos sistemas operativos va a funcionar exactamente igual porque la imagen que se ejecuta en el contenedor ya tiene todas las dependencias lo único que se necesita es que en la máquina se tenga instalado Docker.

Figura 2.15 Estructura de Docker



Los contenedores aíslan el software de su entorno y aseguran su ejecución uniforme, incluso en medio de variaciones, como las que podrían surgir entre las etapas de desarrollo y las de implementación. Esto se aplica a los contenedores de Docker cuando se ejecutan en el Docker Engine.

Una vez que conocemos lo que es Docker tendremos que instalarlo y seguir con la instalación de Home Assistant container, para ello debemos de tener nuestro sistema operativo raspbian ya instalado en la Raspberry y luego seguir los pasos:

1. Abrir la ventana de comandos de la Raspberry.
2. Instalar Docker con el siguiente código:

#Actualizar el sistema

```
sudo apt update
```

```
sudo apt upgrade
```

#Obtenemos el script de Docker

```
curl -sSL https://get.docker.com | sh
```

Configurar docker

```
sudo usermod -aG docker pi
```

#Para verificar que está instalado correctamente

```
docker run hello-world
```

#Instalar Home Assistant escribiendo el siguiente código:

```
docker run -d \
```

```
  --name homeassistant \
```

```
  --privileged \
```

```
  --restart=unless-stopped \
```

```
  -e TZ=MY_TIME_ZONE \
```

```
  -v /PATH_TO_YOUR_CONFIG:/config \
```

```
  --network=host \
```

```
  ghcr.io/home-assistant/home-assistant:stable
```

En estas líneas debemos de cambiar dos parámetros, que son la carpeta donde se instalara el programa y la zona horaria correspondiente.

- Finalmente Podemos acceder desde <http://<host>:8123> donde host será la dirección IP de nuestra Raspberry, que será verificada entrando a las configuraciones del Router que se esté utilizando en este caso un Huawei B612s.

Figura 2.16 Lista de dispositivos inalámbricos conectados en la configuración del Router

ID	Dirección IP	Nombre del host	Dirección MAC	Duración	Operación
centralita: 3					
1	192.168.8.102	homeassistant	DC:A6:32:FD:5A:6A	00:01:10	Bloquear

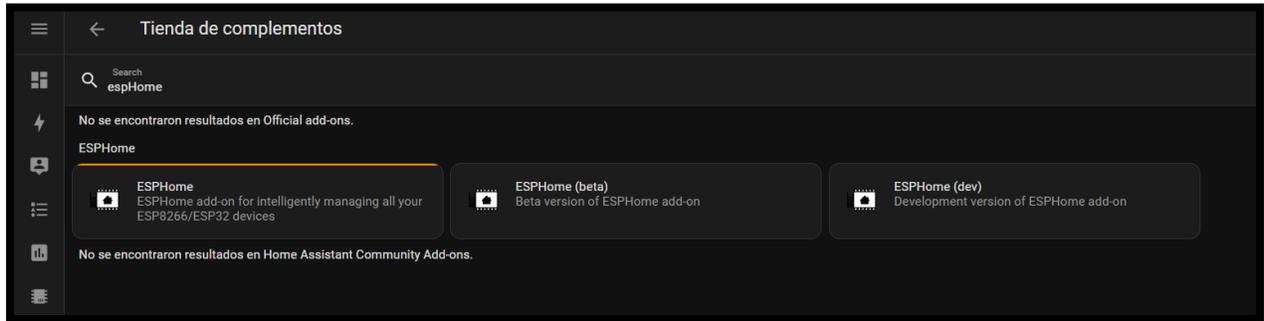
Utilizando cualquiera de los métodos expuestos en este documento podremos crear nuestro servidor, si todo se instaló adecuadamente nos aparecerá la ventana para crear nuestro usuario y contraseña del Home Assistant.

Figura 2.17 Página de inicio de sesión de Home Assistant

2.3.2 ESP32 con ESPHome

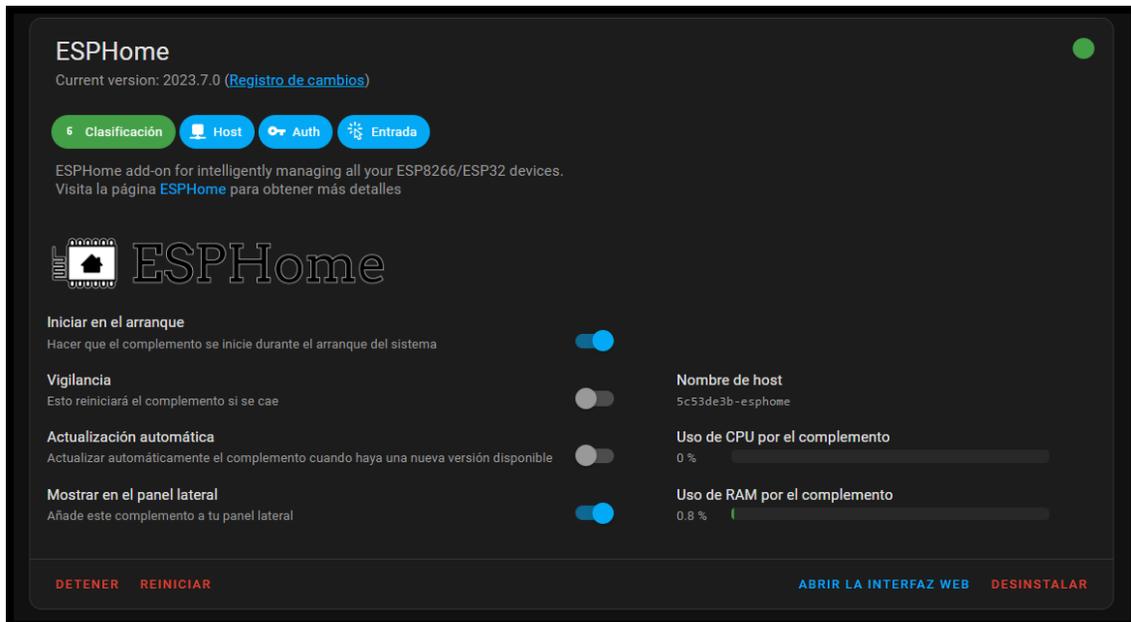
En la página principal de nuestro servidor de Home Assistant tendremos que agregar la extensión de ESPHome para poder integrar la ESP32 a nuestra red y así empezar a programarla en ese mismo entorno, para esto instalamos ESPHome en el apartado ubicado en “Add-ons” o tienda de complementos como vemos en la ilustración:

Figura 2.18 *Página de tienda de complementos*



Luego de instalarlo es recomendable tenerlo en la barra de tareas ubicada en la parte lateral izquierda, para ello activamos las opciones de mostrar que nos aparecen luego de la instalación como lo vemos en la ilustración. Ya con esto tendremos integrado ESPHome en nuestro Home Assistant y siempre es factible reiniciar el programa después de cualquier instalación

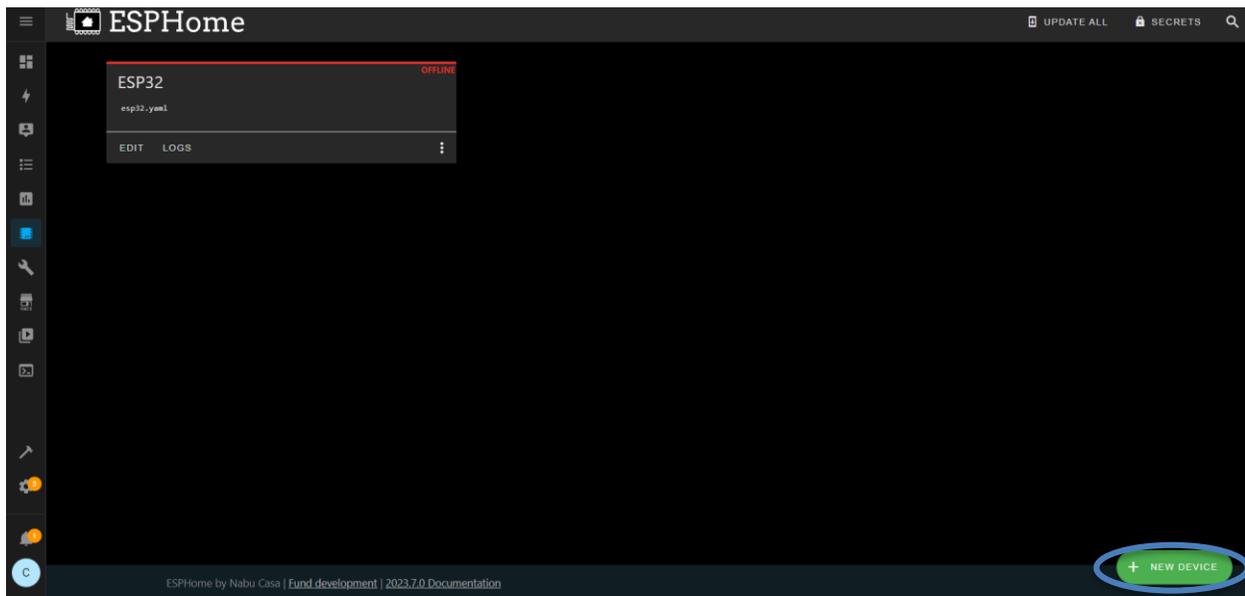
Figura 2.19 *Configuración de ESPHome*



Ya instalado abrimos el entorno de ESPHome para poder añadir un nodo, que en otras palabras será cada una de las ESP32 que deseemos ir agregando, para este caso agregaremos un solo nodo o microcontrolador en el cual podremos integrar varios sensores y actuadores gracias a la cantidad numerosa de pines.

Si el caso lo amerita se pueden añadir más nodos, por ejemplo, ya en un sistema domótico real es necesario ya que cada dispositivo se encontrará en un lugar o ubicación diferente.

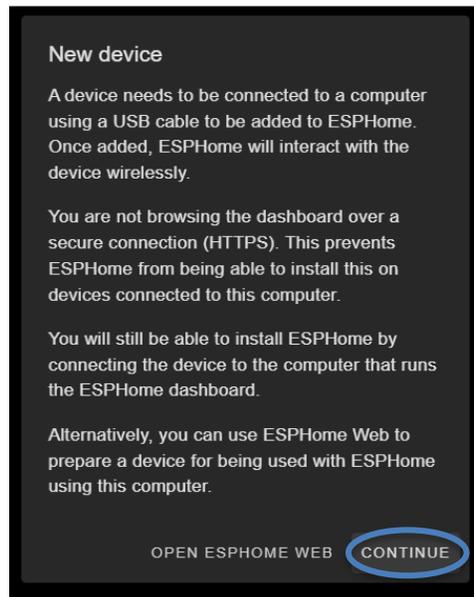
Figura 2.20 Botón de añadir nuevo dispositivo en ESPHome



Cuando le damos en añadir nos aparecerá un mensaje que nos indica que la ESP32 tiene que estar conectada a través de su cable USB, pero éste solo es para la primera vez, es decir instalar el firmware de ESPHome en el microcontrolador, posteriormente en la programación podremos aprovechar el Wi-Fi de la ESP32 para cargar cualquier código o modificación inalámbricamente.

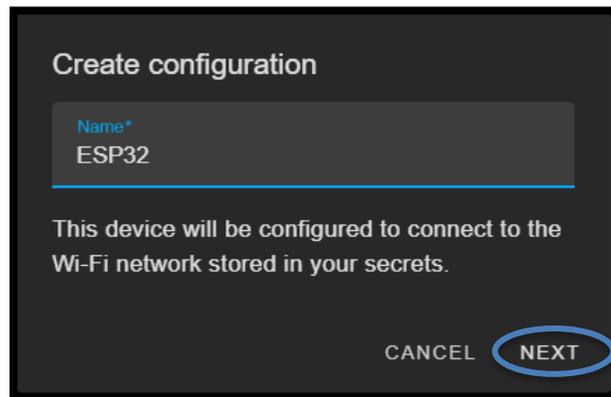
Además, tenemos dos opciones para realizar esta instalación, una es desde la misma plataforma de Home Assistant dándole clic en continuar como vemos en la figura. El otro método y más recomendable es instalarlo desde la misma página web de ESPHome.

Figura 2.21 Información de Instalación de un nuevo dispositivo



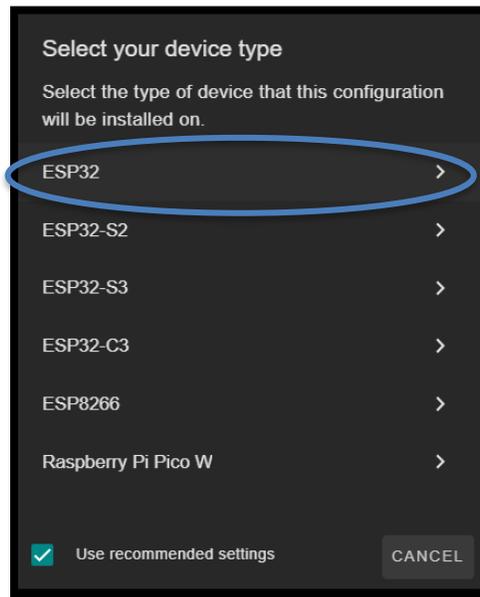
Con la ESP32 configurada con el firmware podemos empezar a colocar las credenciales, el nombre del nodo, y el nombre de la red de internet a la que estará funcionando nuestro sistema. Luego de esto seleccionamos el microcontrolador a usar que es la ESP32 ya que nos muestra varios modelos que podrían usarse.

Figura 2.22 Nombre del dispositivo que se va a agregar



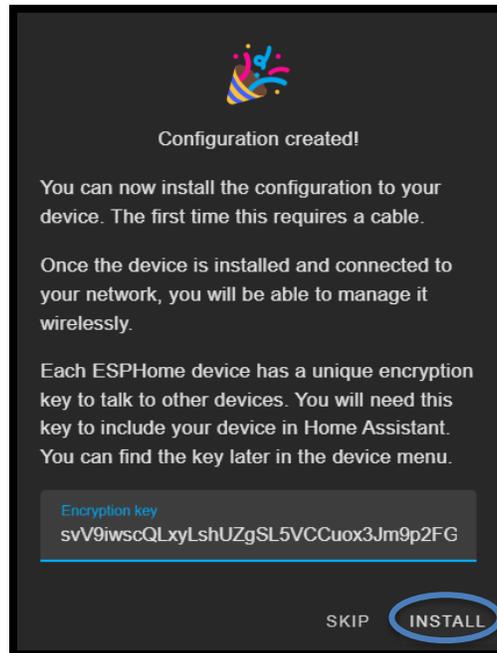
Nota. Se debe colocar un nombre de cómo quiere nombrar al nuevo dispositivo agregado.

Figura 2.23 Selección de dispositivo



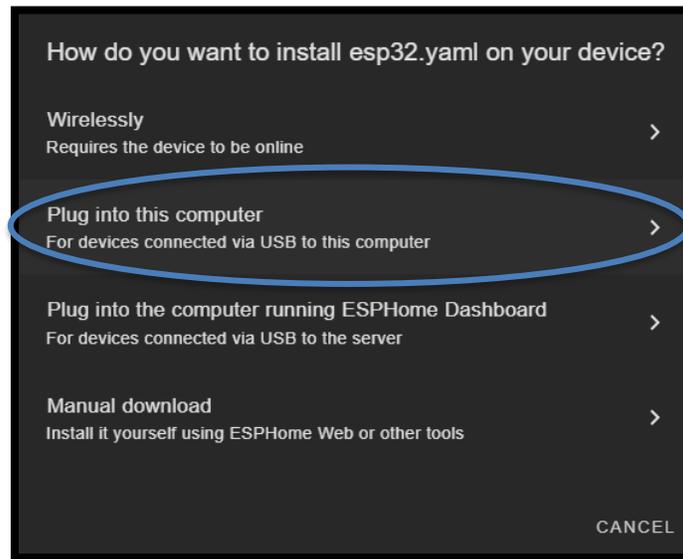
Nota. Se debe seleccionar el modelo del dispositivo que se quiere agregar en este caso se debe seleccionar ESP32.

Figura 2.24 Código de Encriptación



Nota. El código de encriptación brindado por ESPHome nos va a servir para integrar la ESP32 con Home Assistant. Para continuar con la instalación de esp32.yaml de damos clic en INSTALL.

Figura 2.25 Instalar esp32.yaml Paso 1



Nota. Seleccionar la segunda opción “conéctalo a esta computadora”.

El nodo está listo para ser instalado en la ESP32 y para esto existen cuatro alternativas:
Inalámbricamente vía Wi-Fi:

Después de la primera vez que instalamos ESPHome podremos usar esta opción y cargar toda la programación a la tarjeta, esta opción es recomendable una vez que este creado el nodo, pero no para la instalación de este.

Conectar la ESP32 a la computadora donde estamos viendo el Home Assistant:

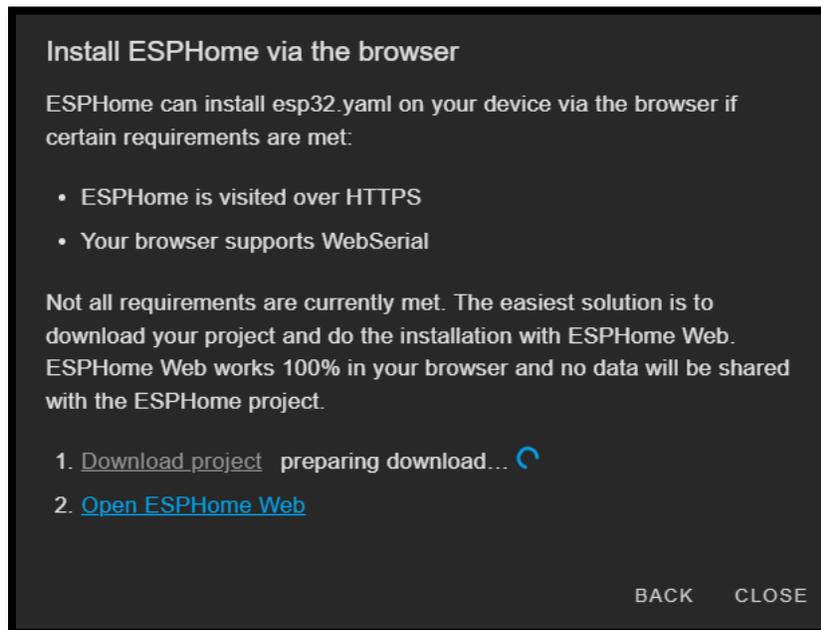
Para esta opción conectamos vía USB a la computadora donde estamos configurando todo, esta alternativa no sería útil si estuviésemos configurando todo a través de un teléfono móvil o Tablet.

Conectar la ESP32 al dispositivo (servidor) donde está instalado el Home Assistant:

Es la opción más recomendable ya que el servidor instalará vía USB a la ESP32, en nuestro caso la Raspberry Pi 4 cuenta con entradas USB y es una manera más rápida y viable de realizar la instalación.

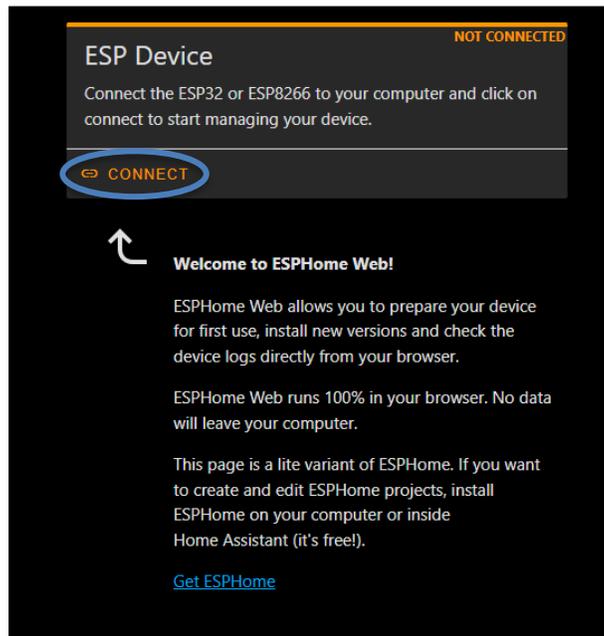
La última opción es instalar Manualmente, es decir flashear a través de otros programas la ESP32 con el código que nos da ESPHome, para esta opción se requiere un poco más de conocimiento en el manejo de la tarjeta en cuestión.

Figura 2.26 Instalar esp32.yaml Paso 2



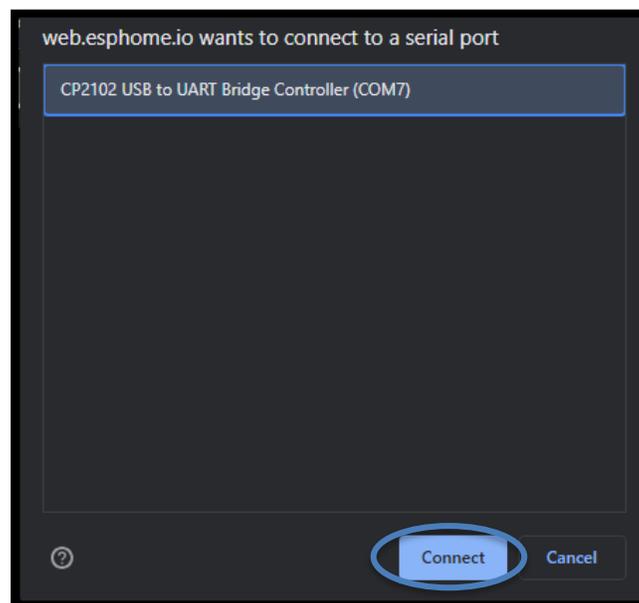
Nota. Se debe esperar hasta que el archivo “.bin” se genere con éxito, después se da clic en “Download Project”. Posterior a la descarga del archivo se da clic en “Open ESPHome Web” y esto nos va a redirigir hacia la página de ESPHome donde podremos cargar el archivo que descargamos.

Figura 2.27 Instalar esp32.yaml Paso 3



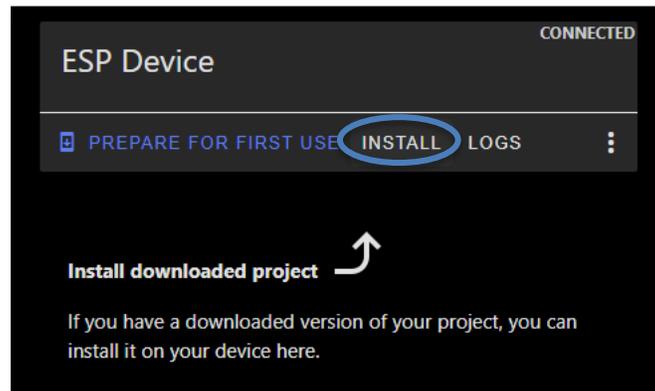
Nota. En la página de ESPHome nos aparece la ventana “ESP Device” dar clic en CONNECT. Además, conectar la ESP32 vía USB a la computadora donde se está realizando la configuración.

Figura 2.28 Instalar esp32.yaml Paso 4



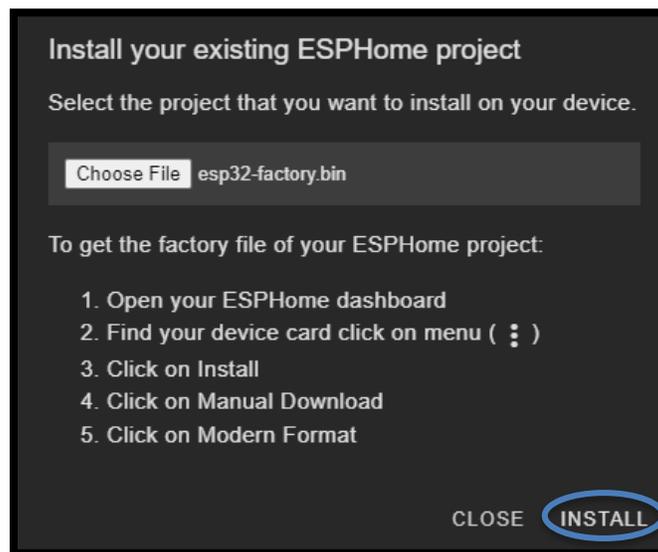
Nota. En la ventana de los puertos COM habilitados, seleccionar el puerto al que se encuentra conectado la ESP32 vía USB y dar clic en Connect.

Figura 2.29 Instalar esp32.yaml Paso 5



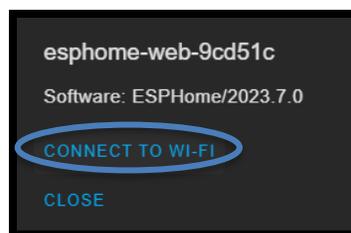
Nota. Se procede con la instalación haciendo en INSTALL.

Figura 2.30 Instalar esp32.yaml Paso 6



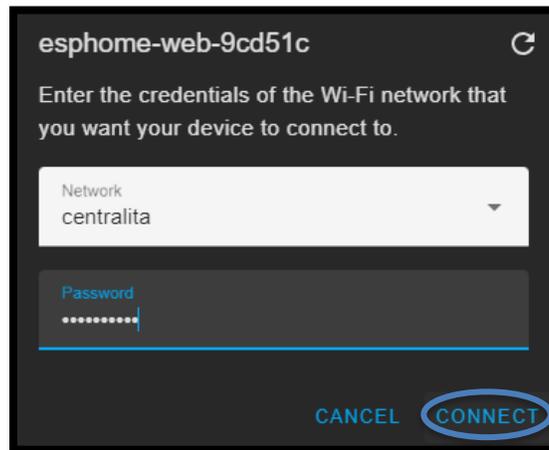
Nota. Es momento de seleccionar el archivo “.bin” que se descargó en el paso 2. Dar clic en Choose File y seleccionar el archivo descargado. Finalmente, dar clic en INSTALL y esperar alrededor de 2 minutos hasta que termine la instalación del archivo en la ESP32.

Figura 2.31 Conexión Wi-Fi



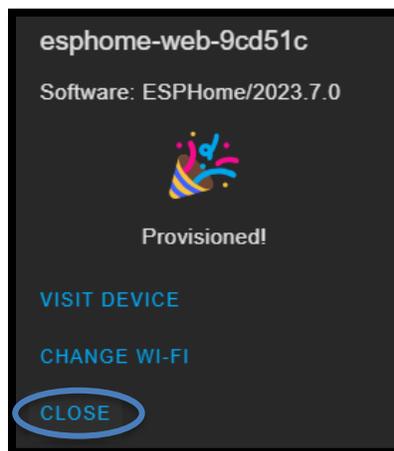
Nota. Para conectar la ESP32 a la red Wi-Fi dar clic en “CONNECT TO WI-FI”

Figura 2.32 Credenciales de la red Wi-Fi



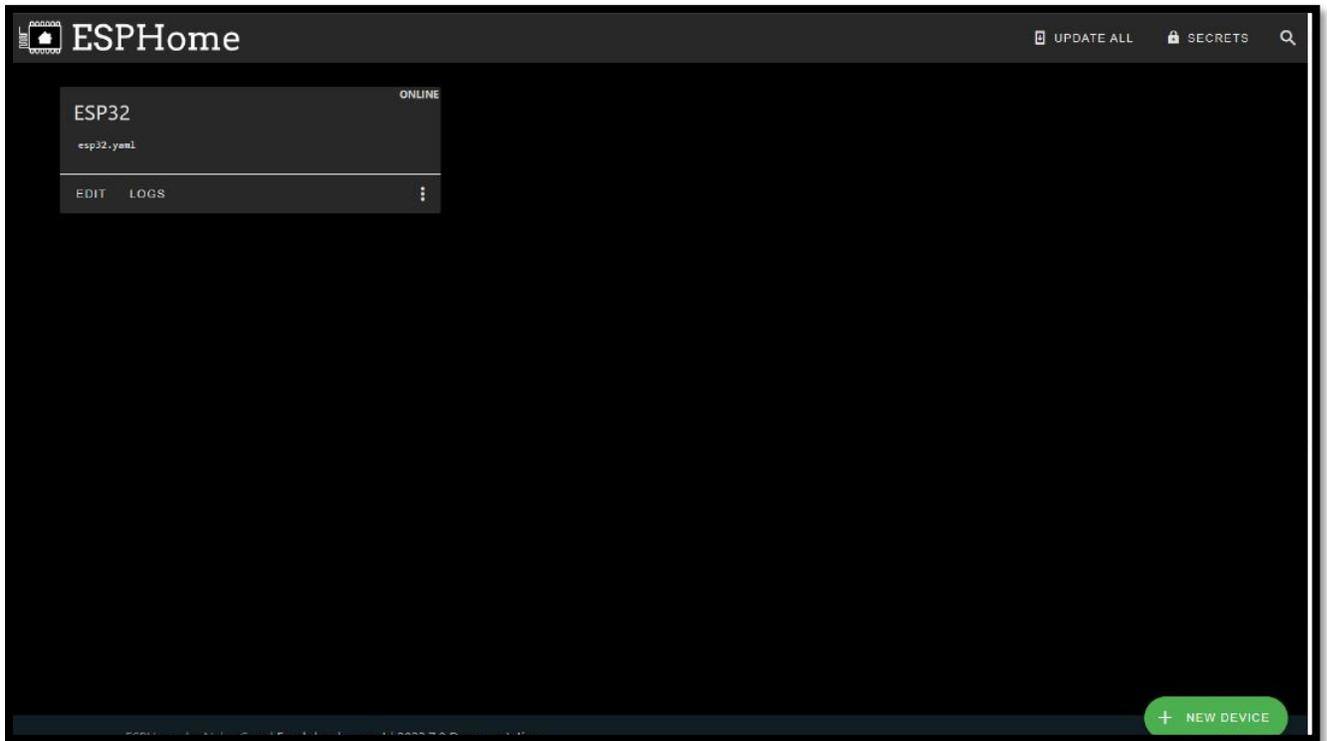
Note. Colocar las credenciales de la red Wi-Fi a la que se va a conectar la ESP32.

Figura 2.33 Finalización de la conexión Wi-Fi



Nota. Cuando aparezca esta ventana se ha finalizado con el proceso de instalar el código base y la conexión a la red Wi-Fi de la ESP32 dar clic en CLOSE, ahora es momento de ir a ESPHome de Home Assistant.

Figura 2.34 ESPHome con el nodo creado

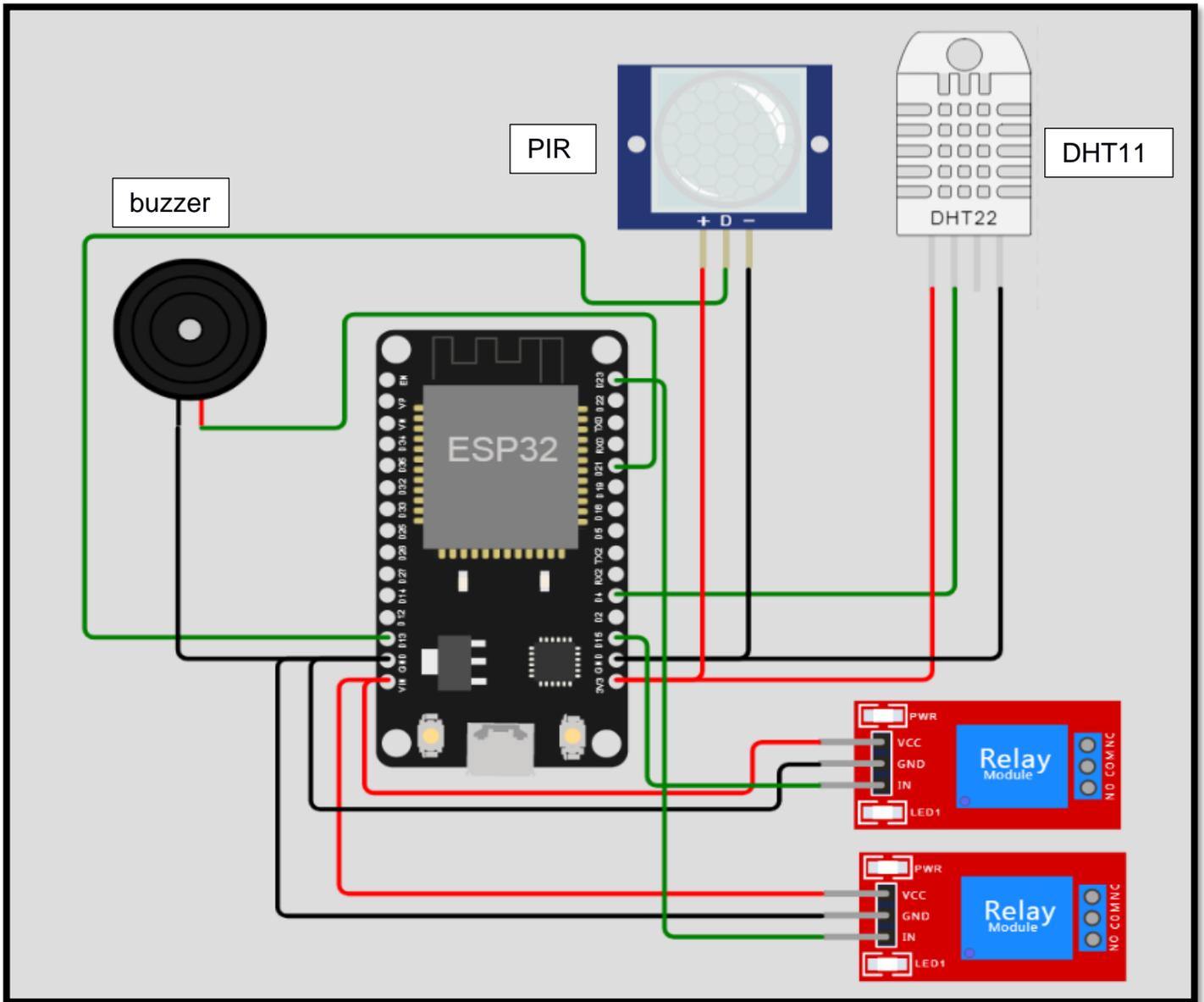


2.4 Diagrama de conexiones

En el siguiente diagrama se encuentran las conexiones físicas de todos los sensores que se van a utilizar y se describe la asignación de cada uno de los pines de la ESP32 que se utilizaron.

La mayoría de los módulos de sensores vienen con tres pines para su conexión, tal es el caso del sensor PIR, sensor DHT11 y el relé de 5v. Con anterioridad en la Fig.5 se mostró el esquema de la organización de los pines de este microcontrolador para la fácil comprensión de este esquema.

Figura 2.35 Diagrama de conexiones de los sensores y actuadores a la ESP32



Nota. La figura 2.35 muestra el diagrama de conexiones realizadas en el simulador WOKWI donde podemos identificar el sensor PIR, DHT11 y como actuador el buzzer y dos relés que están en la parte inferior de la imagen.

En la siguiente tabla podremos identificar los pines asignados a cada sensor y actuador en la ESP32 para de esta manera realizar la programación correcta ya que es necesario que la programación y las conexiones físicas estén coordinadas.

Tabla 2.7 Tabla de asignaciones de pines de la ESP32

Sensor/actuador	Pin
DHT11	3V3 GND GPIO4
Relé 1 (Foco)	VIN GND GPIO23
PIR	3V3 GND GPIO13
Relé 2 (cerradura)	VIN GND GPIO15
Buzzer	GND GPIO21

Nota. La tabla 2.7 muestra que el color de la letra en la abreviación del pin es igual al color del cable en la figura 2.35, de esta manera se garantiza una conexión segura.

2.5 Programación en ESPHome

Observaremos el código cargado en nuestra ESP32, previo a esto de manera explicativa se muestra un diagrama de flujo para entender la estructura de la programación.

Cabe mencionar que ESPHome nos brinda códigos genéricos para cada sensor y actuador compatible con la ESP32, pero es necesario tener conocimientos de programación ya que queremos dar aplicaciones reales a nuestros dispositivos.

Una ventaja que brinda Home Assistant es que podemos realizar la programación de los dispositivos desde la misma plataforma, y a su vez desde cualquier dispositivo donde podamos acceder a nuestro usuario del sistema domótico por ejemplo desde el mismo panel e inclusive desde la Raspberry.

Figura 2.36 Diagrama de flujo sensor PIR y buzzer

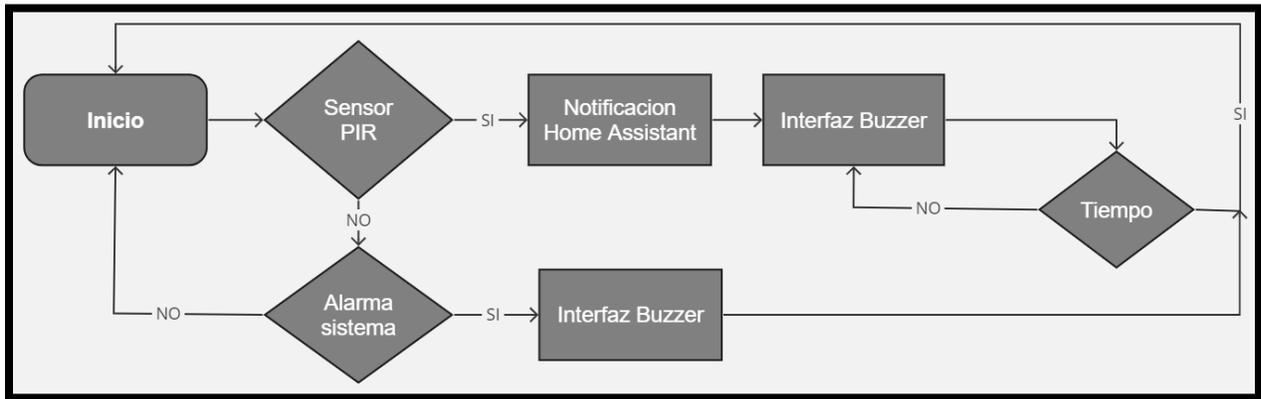


Figura 2.37 Diagrama de flujo relé y foco

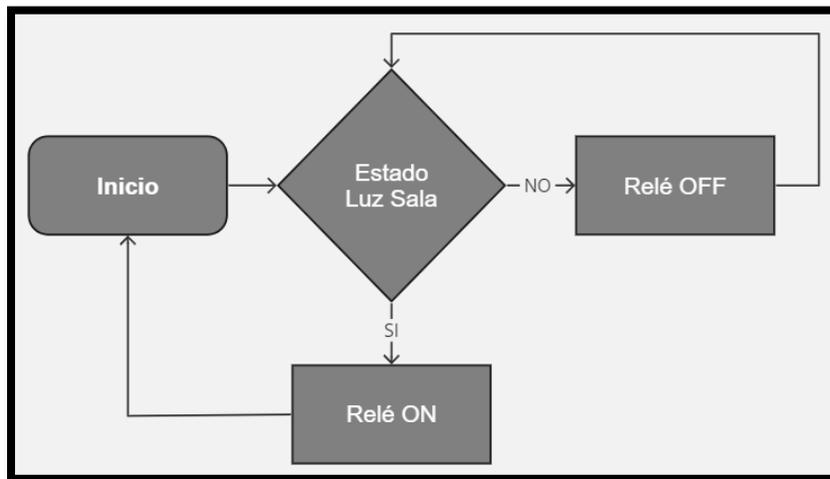


Figura 2.38 Diagrama de flujo relé y cerrojo eléctrico

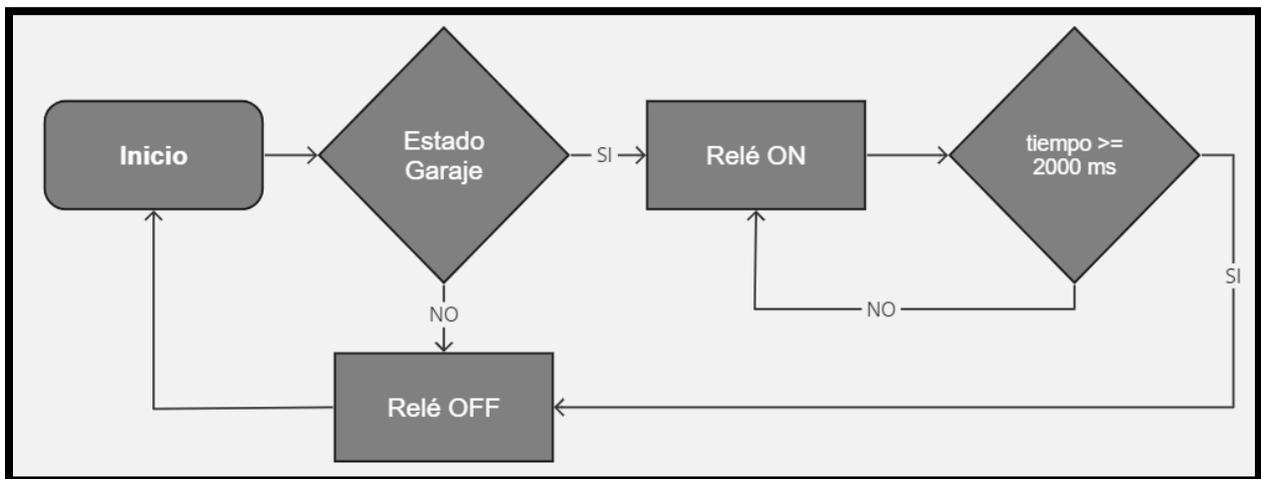
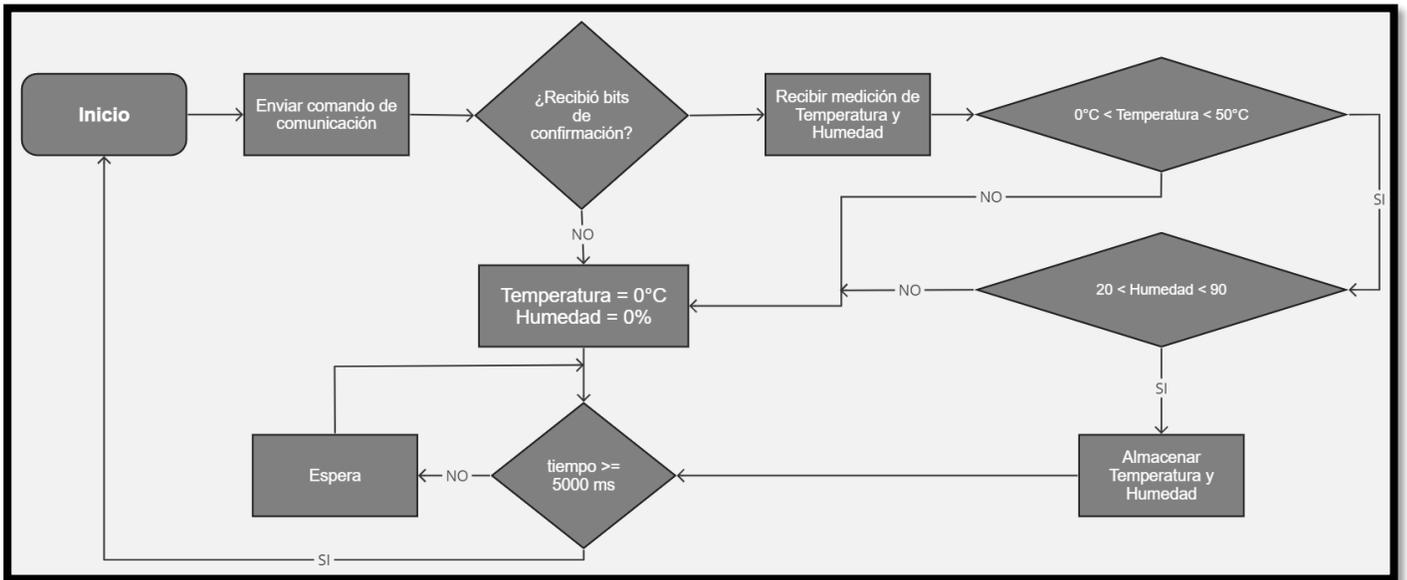


Figura 2.39 Diagrama de flujo sensor DHT11



Una vez realizado el diagrama de flujo de cada uno de los sensores y actuadores que van a interactuar en el sistema es momento de empezar a programar en el IDE que nos proporciona ESPHome.

Figura 2.40 Entrar al IDE de ESPHome en Home Assistant



Nota. Para empezar a programar las diferentes entidades en el IDE de ESPHome es necesario dar clic en EDIT y se abrirá la ventana del IDE.

ESPHome nos brinda un código preestablecido en el cual se encuentra inicializado automáticamente y a su vez genera las credenciales para conectarnos inalámbricamente. Es decir, en este código tendremos la clave y contraseña de nuestra

red inalámbrica, así como las credenciales de nuestro nodo creado para que se integre con Home Assistant.

```
esphome:
  name: esp32
  friendly_name: ESP32
esp32:
  board: esp32dev
  framework:
    type: arduino
# Habilitar el Registro
logger:
# Habilitar Home Assistant API
api:
  encryption:
    key: "ZCymG2nWxAQWhByuUMN8pM6XSWB7PGMxcNDxLZGKaeU="
ota:
  password: "8460fe7e9e690fcc696cdb1233a4350"
wifi:
  ssid: !secret wifi_ssid
  password: !secret wifi_password
# Active la opción de acceso secundario (portal cautivo) en situaciones donde la conexión Wi-Fi
# experimente problemas
ap:
  ssid: "Esp32 Fallback Hotspot"
  password: "Wjxe0ejn8xpr"
```

La siguiente etapa es programar cada dispositivo que vayamos a integrar, esto queda a disposición de cada usuario. ESPHome cuenta en su página web con una amplia lista de dispositivos con códigos base para su rápida programación, pero es recomendable tener conocimientos previos de programación de microcontroladores para darle una aplicación completa a nuestros dispositivos.

A continuación, se muestra la programación de cada sensor y actuador usado en este sistema domótico, los cuales ya fueron expuestos en secciones anteriores:

Sensor DHT11

En este código asignamos el pin número 4 del microcontrolador, y al indicarle que tipo de plataforma es nos brinda las dos secciones de temperatura y humedad a la cual se les asigna un nombre. Finalmente indicamos el tiempo de actualización o medición de 5 segundos.

sensor:

- platform: dht

pin: 4

temperature:

name: "Temperature"

humidity:

name: "Humidity"

update_interval: 5s

Alarma de movimiento (Sensor PIR + Buzzer)

Para realizar una aplicación de un sistema domótico tradicional se implementó una alarma, definiendo el pin número 13 para la entrada del sensor PIR y asociando su activación a la variable buzzer, que está definida como salida en el pin 21.

switch:

- platform: gpio

id: buzzer

name: buzzer

pin:

number: 21

binary_sensor:

- platform: gpio

pin: 13

name: "PIR Sensor"

device_class: motion

on_press:

then:

switch.turn_on: buzzer

on_release:

then:

switch.turn_off: buzzer

filters:

- delayed_on: 100ms

Cerradura eléctrica (Relé 5v)

Otra aplicación realizada fue una cerradura eléctrica para puertas, definiendo el pin 23 como salida para el relé de 5v y dándole parámetros de tiempo para la apertura.

switch:

- platform: gpio

pin: 23

id: relay

name: "Gate Remote"

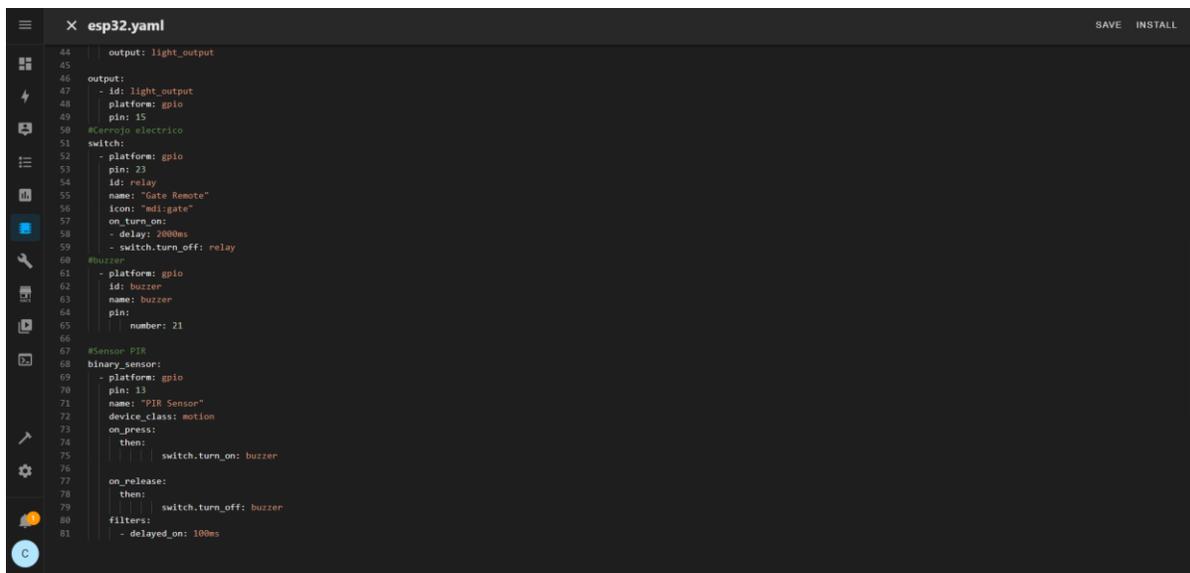
icon: "mdi:gate"

on_turn_on:

- delay: 2000ms

- switch.turn_off: relay

Figura 2.41 IDE de ESPHome en Home Assistant

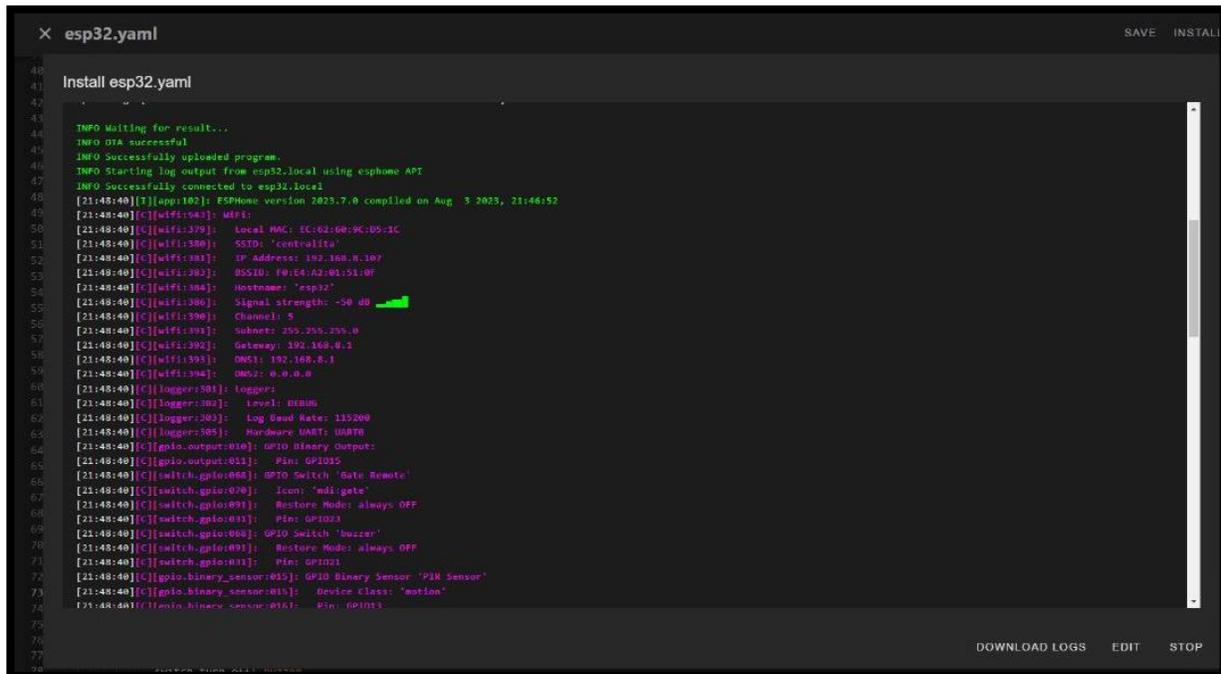


```
44   output: light_output
45
46   output:
47     - id: light_output
48       platform: gpio
49       pin: 15
50   #Cerrajo electrico
51   switch:
52     - platform: gpio
53       pin: 23
54       id: relay
55       name: "Gate Remote"
56       icon: "mdi:gate"
57       on_turn_on:
58         - delay: 2000ms
59         - switch.turn_off: relay
60   #buzzer
61   - platform: gpio
62     id: buzzer
63     name: buzzer
64     pin:
65       number: 21
66
67   #Sensor PIR
68   binary_sensor:
69     - platform: gpio
70       pin: 13
71       name: "PIR Sensor"
72       device_class: motion
73       on_press:
74         then:
75           - switch.turn_on: buzzer
76
77       on_release:
78         then:
79           - switch.turn_off: buzzer
80       filters:
81         - delayed_on: 100ms
```

Nota. Una vez que hemos escrito nuestros códigos dentro de la ventana donde se creó nuestro nodo, podremos guardar e instalar esta programación en el microcontrolador.

Cabe mencionar que una vez que hemos configurado por primera vez la tarjeta podremos cargar la información inalámbricamente las veces que necesitemos, es decir solo para la primera configuración necesitamos el cable de datos.

Figura 2.42 Compilación de la programación



```
Install esp32.yaml
INFO Waiting for result...
INFO OTA successful
INFO Successfully uploaded program.
INFO Starting log output from esp32.local using esphome API
INFO Successfully connected to esp32.local
[21:48:40][T][app:102]: ESPHome version 2023.7.0 compiled on Aug  3 2023, 21:48:52
[21:48:40][C][wifi:043]: WiFi:
[21:48:40][C][wifi:299]: Local MAC: EC:62:96:9C:05:3C
[21:48:40][C][wifi:300]: SSID: 'controlita'
[21:48:40][C][wifi:303]: IP Address: 192.168.8.107
[21:48:40][C][wifi:303]: BSSID: F0:E4:A2:91:51:9F
[21:48:40][C][wifi:304]: Hostname: 'esp32'
[21:48:40][C][wifi:306]: Signal strength: -50 dB
[21:48:40][C][wifi:300]: Channel: 5
[21:48:40][C][wifi:303]: Subnet: 255.255.255.0
[21:48:40][C][wifi:302]: Gateway: 192.168.0.1
[21:48:40][C][wifi:305]: DNS1: 192.168.0.1
[21:48:40][C][wifi:304]: DNS2: 0.0.0.0
[21:48:40][C][logger:301]: Logger:
[21:48:40][C][logger:303]: Level: DEBUG
[21:48:40][C][logger:303]: Log Read Rate: 135200
[21:48:40][C][logger:305]: Hardware UART: UART0
[21:48:40][C][gpio.output:010]: GPIO Binary Output:
[21:48:40][C][gpio.output:011]: Pin: GPIO5
[21:48:40][C][switch.gpio:008]: GPIO Switch 'Gate Remote'
[21:48:40][C][switch.gpio:070]: Icon: 'mdi:gate'
[21:48:40][C][switch.gpio:091]: Restore Mode: always OFF
[21:48:40][C][switch.gpio:011]: Pin: GPIO23
[21:48:40][C][switch.gpio:058]: GPIO Switch 'buzzer'
[21:48:40][C][switch.gpio:091]: Restore Mode: always OFF
[21:48:40][C][switch.gpio:011]: Pin: GPIO21
[21:48:40][C][gpio.binary_sensor:015]: GPIO Binary Sensor 'PIR Sensor'
[21:48:40][C][gpio.binary_sensor:015]: Device Class: 'motion'
[21:48:40][C][gpio.binary_sensor:076]: Pin: GPIO13
```

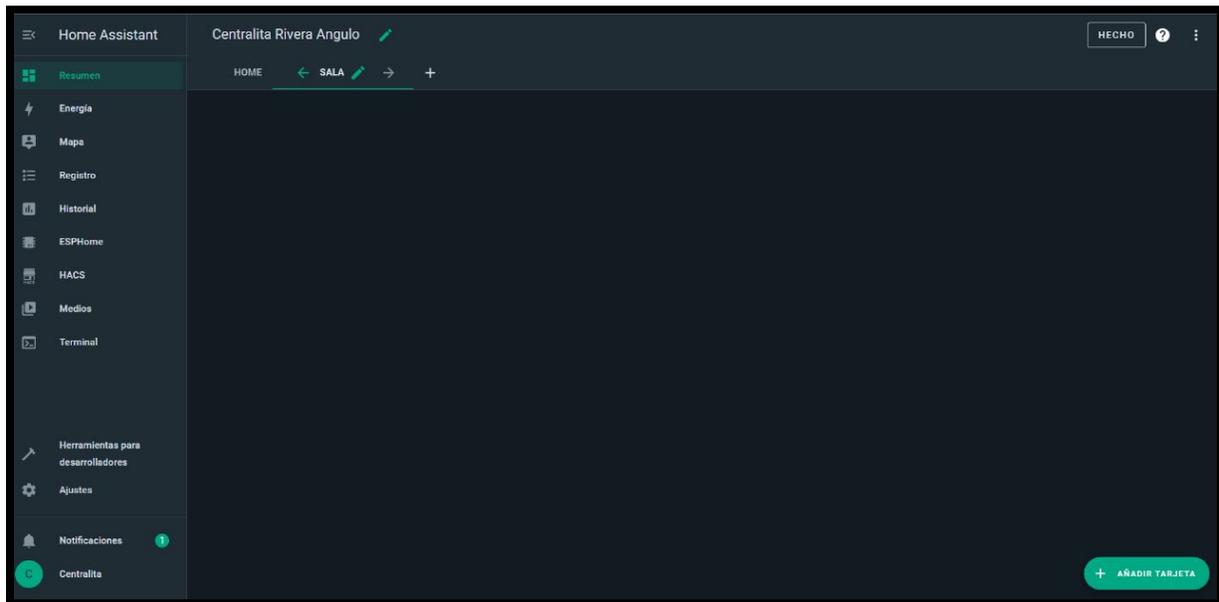
Si realizamos correctamente la programación debe aparecer esta ventana mostrando el estado de la red y mostrando los datos de cada sensor. En este caso se realizó la programación desde una computadora en la que ingresamos con nuestro usuario y contraseña a Home Assistant.

2.6 Dashboard en Home Assistant

En un sistema domótico es clave tener una buena interfaz gráfica para el usuario, ya que con esto garantizamos el fácil acceso a las funciones que brindamos como ver la información del clima dentro del hogar e interactuar con los dispositivos conectados a la red. Home Assistant brinda la opción de administrar nuestro panel de acuerdo con la necesidad y gusto de cada usuario.

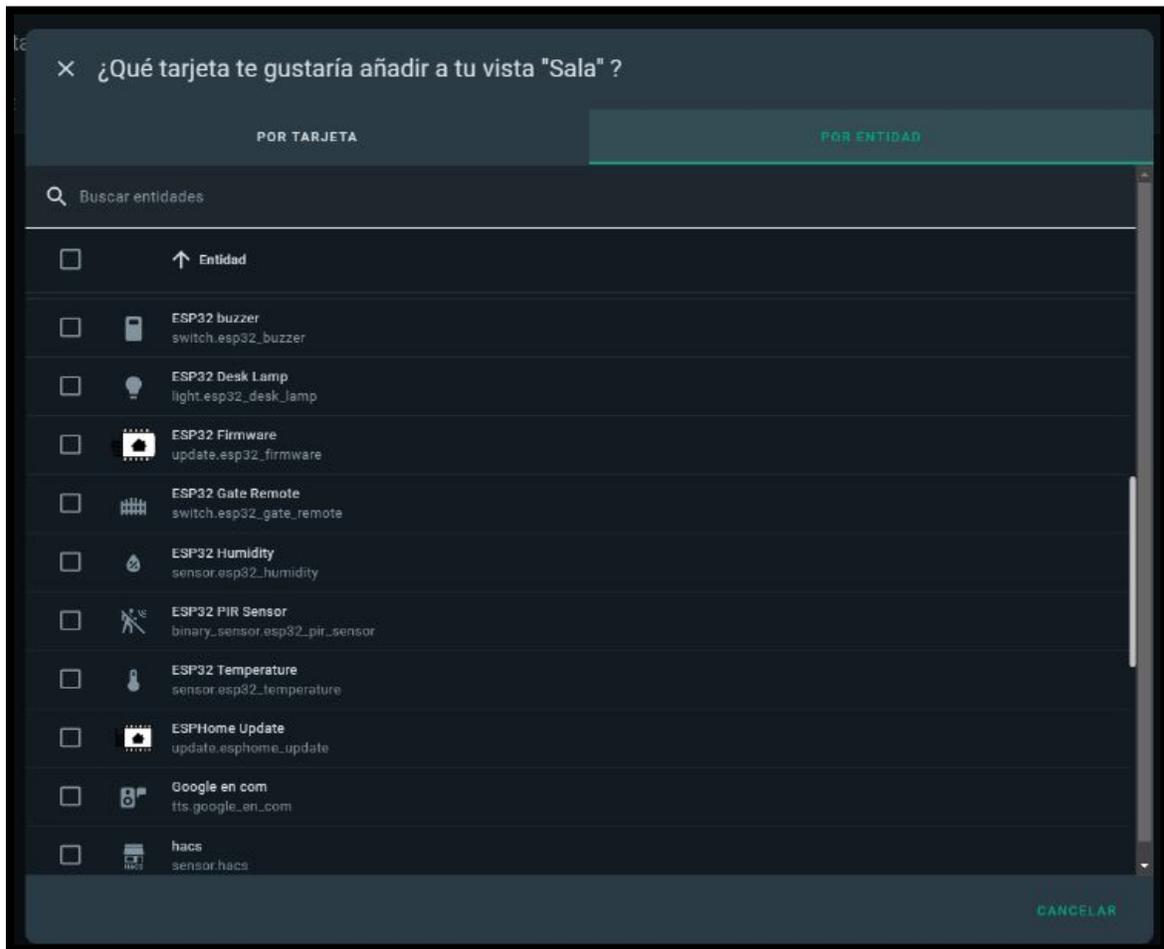
Para ello hemos realizado un panel con todos los dispositivos que hemos integrado, de esta manera podremos visualizar y controlar de manera rápida nuestro sistema domótico, a continuación, mostraremos como configurar este panel:

Figura 2.43 Panel inicial en Home Assistant



Como observamos en la figura tenemos vacío nuestro panel de control, para empezar a diseñarlo tenemos que haber compilado nuestra programación de los dispositivos en el microcontrolador y de esta manera, al añadir las tarjetas o widgets, el software reconocerá las entidades y las podremos asociar a una de ellas como vemos en la siguiente figura:

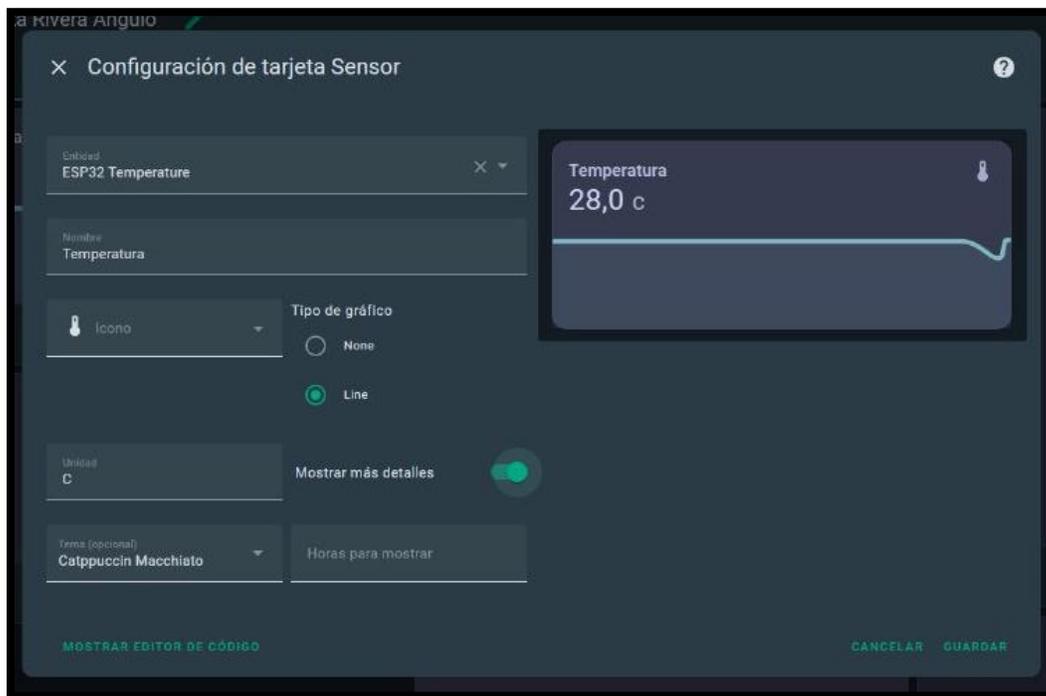
Figura 2.44 Lista de Entidades



En la lista de la figura 2.44 aparecen todas las entidades programadas y nos recomienda un tipo de tarjeta para agregar, aunque nosotros podríamos cambiarlo manualmente, a continuación, se muestra como configurarlo gráficamente y a través del editor de código a manera de ejemplo se va a diseñar la tarjeta para el sensor de temperatura y de humedad:

Configuración gráfica

Figura 2.45 Tarjeta de Temperatura



Configuración con código

graph: line

type: sensor

entity: sensor.esp32_temperature

detail: 2

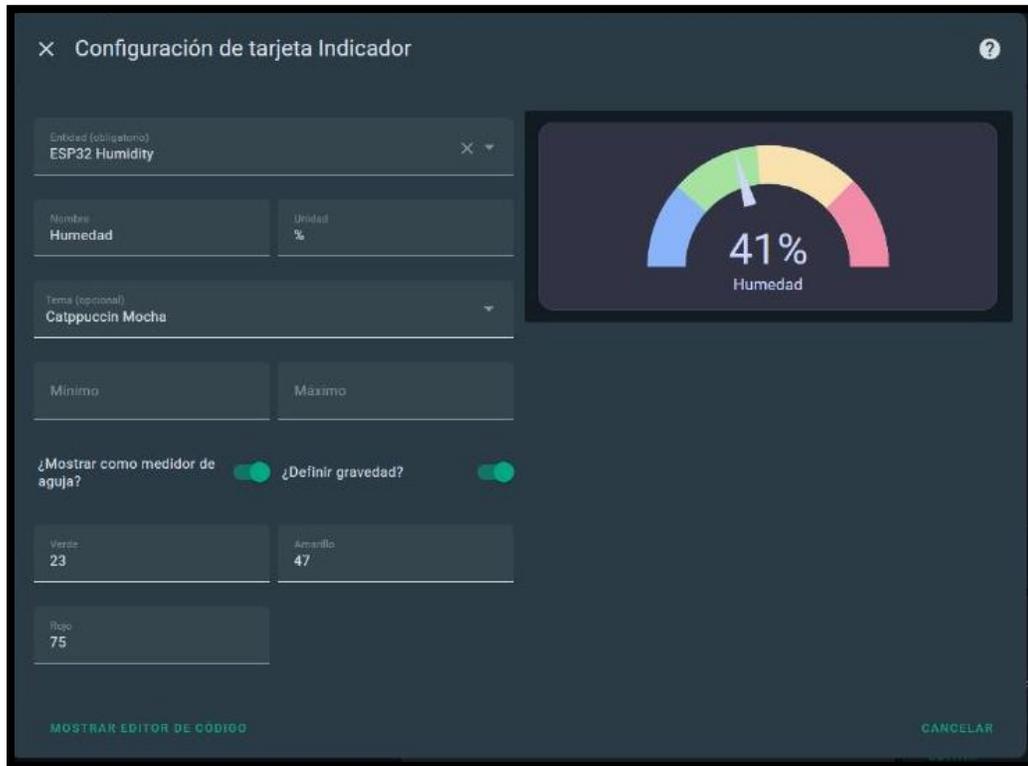
name: Temperatura

unit: C

theme: Catppuccin Macchiato

Configuración gráfica

Figura 2.46 Tarjeta de Humedad



Configuración con código

type: gauge

entity: sensor.esp32_humidity

name: Humedad

unit: '%'

needle: true

severity:

green: 23

yellow: 47

red: 75

theme: Catppuccin Mocha

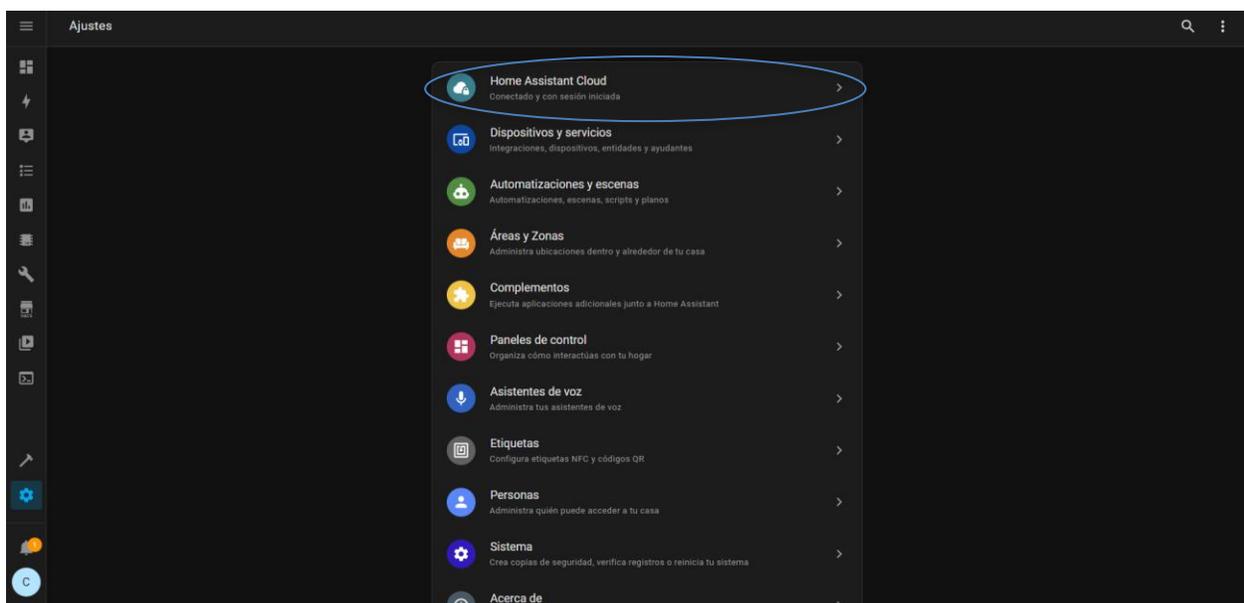
De esta manera se configura cada entidad para formar un panel de control amigable para el usuario, este panel será mostrado en la pantalla de la Tablet y desde ahí se podrá controlar el sistema.

2.7 Home Assistant en IoT

Una vez que tenemos el servidor local en funcionamiento tenemos en cuenta que el sistema domótico funciona con dispositivos que estén conectados a la red local o Wi-Fi a la que está conectado dicho servidor. Es aquí donde usaremos IoT para brindar la posibilidad de controlar y monitorear nuestro sistema desde cualquier parte del mundo donde tengamos internet.

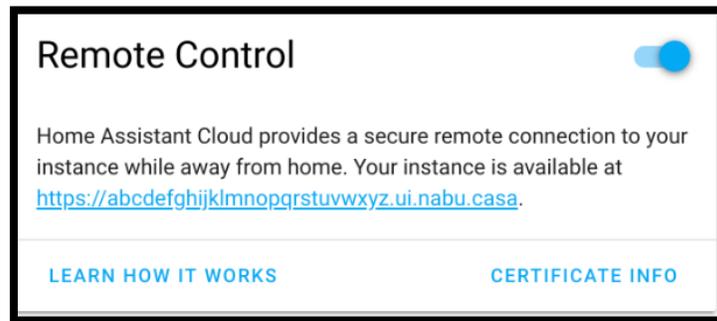
Home Assistant Cloud es el servicio que nos permite conectarnos a nuestro servidor local desde cualquier lugar. Para activar esta función debemos de crear una cuenta de Home Assistant Cloud, ya que este servicio no es gratuito, tiene un costo de 6 dólares mensuales con una prueba gratuita de un mes. Se considera un precio asequible para las funciones que brinda además de la seguridad en la red.

Figura 2.47 Home Assistant Cloud



Nota. Para empezar con la integración IoT se debe dar clic en Home Assistant Cloud y posterior a eso registrar una cuenta de correo electrónico.

Figura 2.48 Ventana de IoT en Home Assistant



Nota. Habilitar la opción de Remote Control para que se genere un link con conexión https consiguiendo de esta forma poder acceder a nuestro servidor de forma externa o desde cualquier lugar del mundo con acceso a internet.

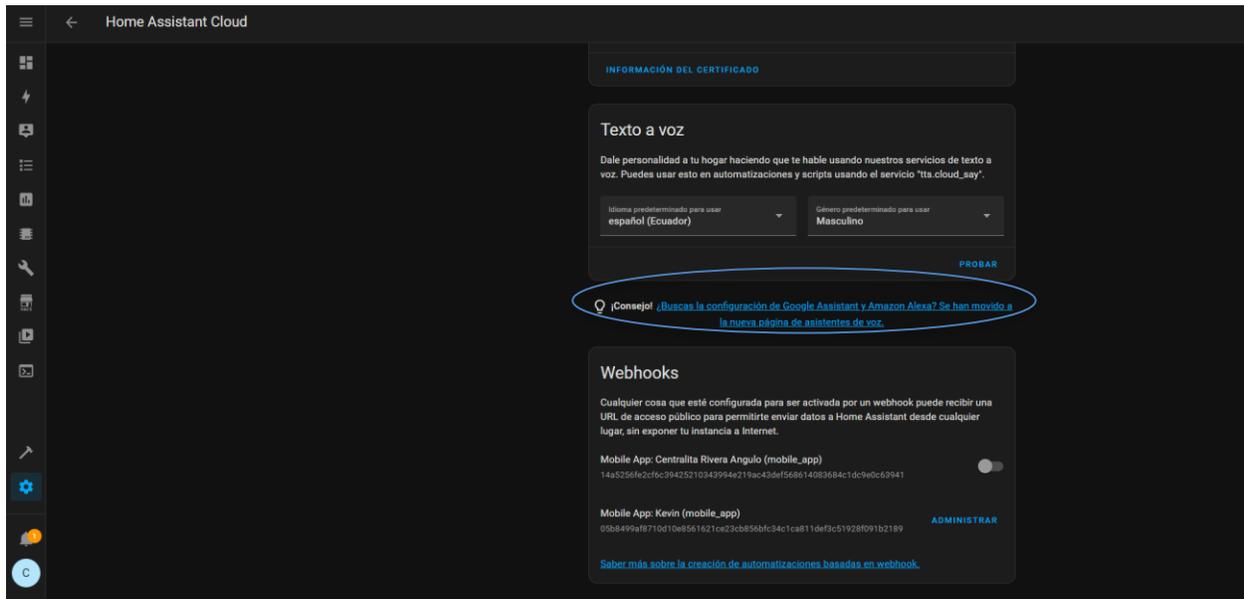
La IU remota encripta toda la comunicación entre nuestro navegador y la instancia local. El cifrado proporciona un certificado Let's Encrypt, ya que Let's Encrypt es la autoridad de certificación que hace posible configurar un servidor HTTPS y permitir que este genere automáticamente un certificado válido para navegadores. Debajo de todo eso, el servidor local de Home Assistant está conectado a uno de los servidores proxy de interfaz de usuario personalizados. Esos servidores proxy de interfaz de usuario funcionan a nivel de TCP y reenvían todos los datos cifrados a la instancia local.

La capacidad de enrutamiento se logra mediante la expansión de la señalización del nombre del servidor (SNI) en el protocolo de capa de transporte seguro (TLS). Contiene la información acerca de qué nombre de host está destinada una solicitud entrante y se reenvía esta información a la instancia local coincidente. Para poder enrutar varias solicitudes simultáneas, todos los datos se enrutarán a través de un multiplexor TCP. La instancia local de Home Assistant recibirá los paquetes TCP, los desmultiplexará, los descifrará con el certificado SSL y los reenviará al componente HTTP.

Una vez terminado el proceso anterior es momento de comenzar a integrar asistentes de voz para que los usuarios tengan una experiencia más sencilla al momento de monitorear los dispositivos integrados al panel.

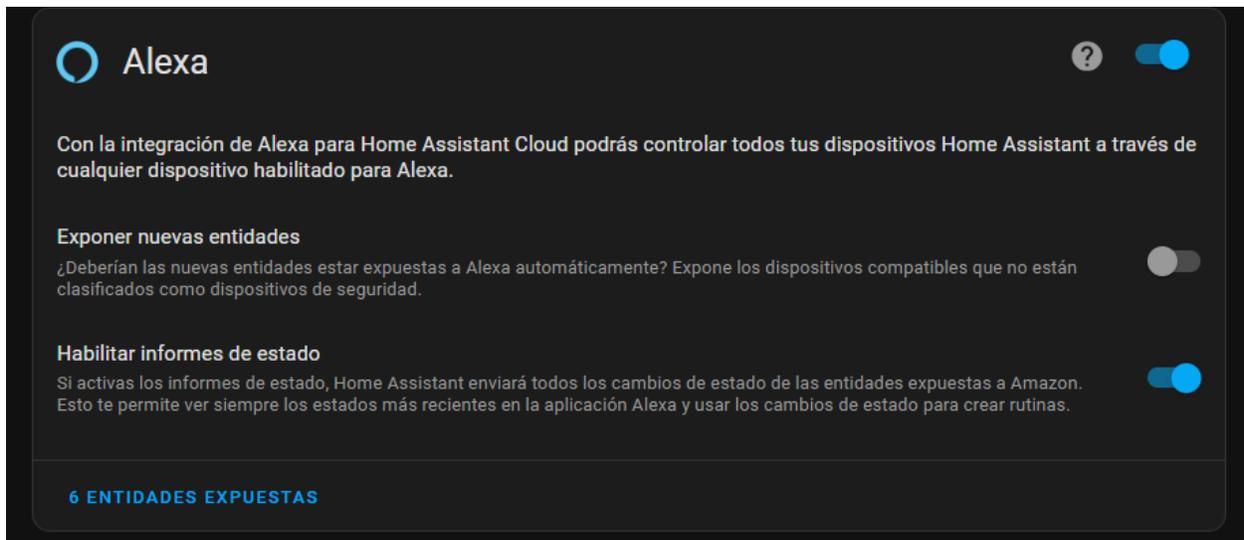
Amazon Alexa:

Figura 2.49 Integración Amazon Alexa



Nota. Dar clic en el lugar señalado de la figura 2.49 para empezar la integración con Amazon Alexa

Figura 2.50 Ventana de Integración Amazon Alexa



Nota. Habilitar las opciones como se muestra en la figura 2.5.

Para poder configurar las entidades que tenemos en nuestra ESP32 se debe ir a ENTIDADES EXPUESTAS como se muestra a continuación:

Figura 2.51 Exponer entidades

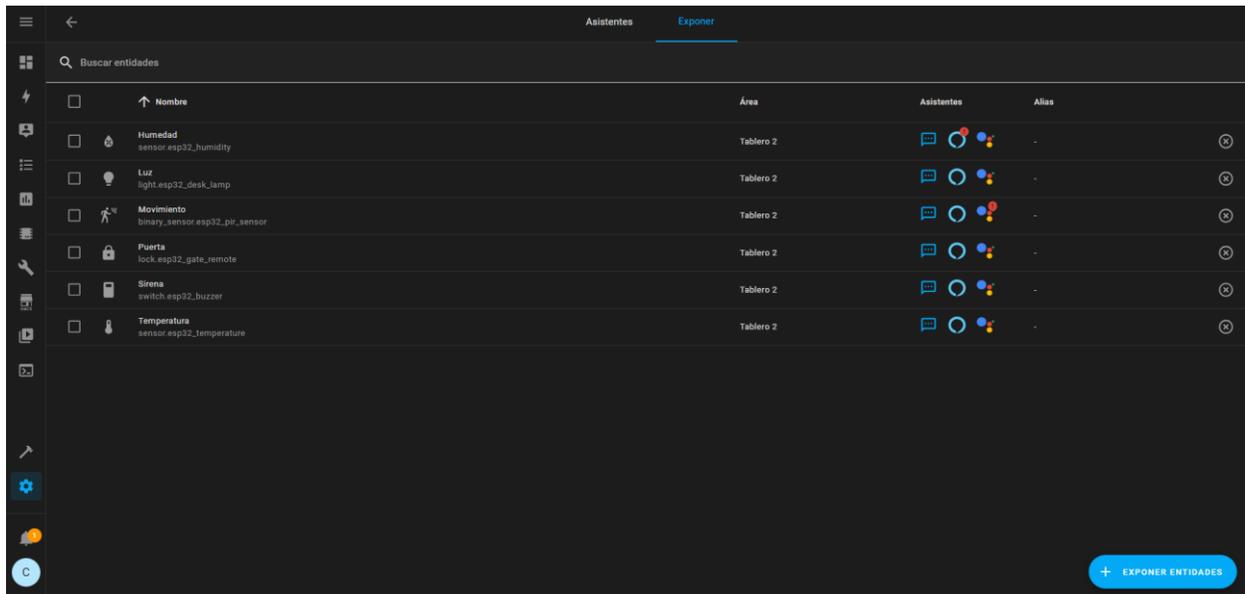
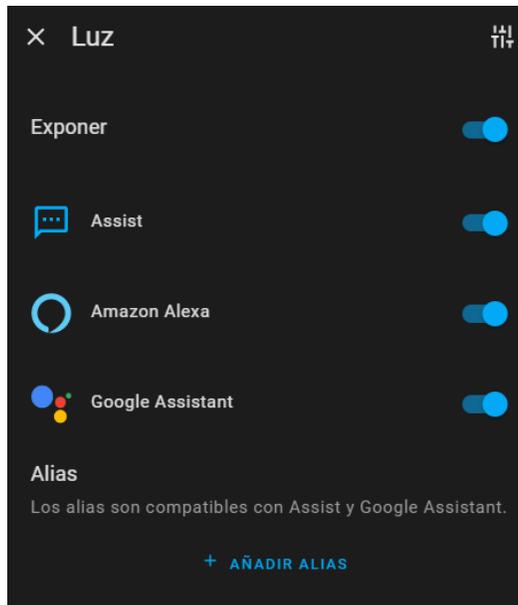
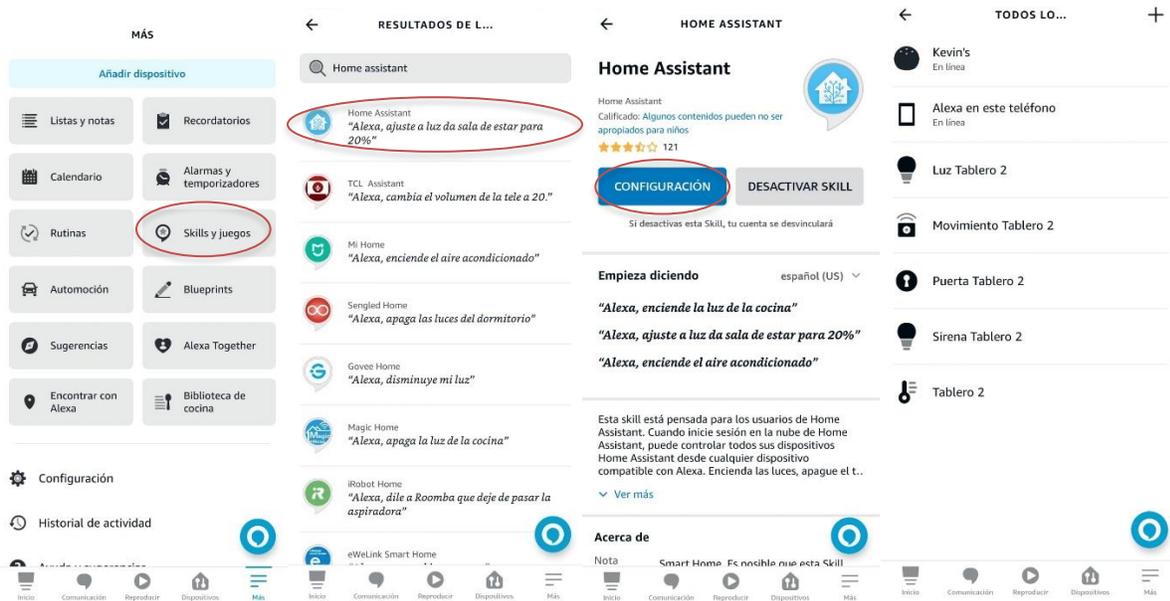


Figura 2.52 Habilitador de Entidades



Nota. Se asegura que todas las entidades estén habilitadas y soportadas por Amazon Alexa. Finalizado la configuración en la interfaz de home Assistant se procede a configurar en la aplicación de Amazon Alexa.

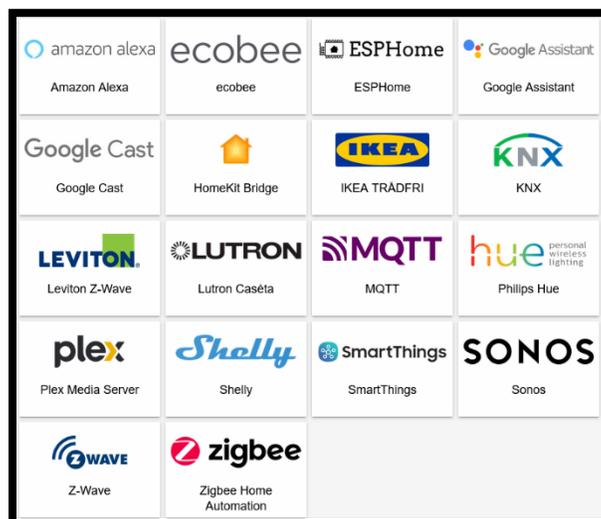
Figura 2.53 Configuración en la App de Amazon Alexa



Nota. Se debe agregar cada entidad como un dispositivo nuevo y configurando el dispositivo como corresponde a la entidad, es decir, si es interruptor, sensor, cerradura, etc.

Además, cabe recalcar que Home Assistant cuenta con muchas integraciones enlazadas a IoT con los cuales podríamos realizar un sistema muy completo para el hogar.

Figura 2.54 Tecnologías de integración



2.8 Base de Datos

Si se está utilizando la configuración preestablecida, los datos se almacenarán en el servidor en el que esté instalado Home Assistant. En este caso es la Raspberry Pi que tiene una tarjeta SD, al almacenar muchos datos afecta el tiempo de reacción del sistema y el tiempo de vida del medio de almacenamiento (la tarjeta SD). Por esta razón, es recomendable establecer el `commit_interval` en un valor más alto, limitar la cantidad de datos almacenados (por ejemplo, excluyendo dispositivos) o almacenar los datos en otro lugar (por ejemplo, otro sistema).

La Grabadora, es la responsable de almacenar los detalles en una base de datos, que luego son manejados por la "history", que es la integración que rastreará todo lo que sucede dentro de Home Assistant y permite al usuario navegar a través de él. Si se excluye alguna entidad del registro, no habrá historial disponible para estas entidades. Home Assistant emplea SQLAlchemy, que es un Mapeador Objeto-Relacional (ORM), es decir, que puede usar cualquier backend SQL para la Recorder que sea compatible con SQLAlchemy, como MySQL, MariaDB, PostgreSQL o MS SQL Server.

Figura 2.55 Ejemplo de visualizador de base de datos SQLite



El motor de base de datos preestablecido es SQLite, que no requiere ninguna configuración y almacena en un único archivo de disco multiplataforma. La base de datos se almacena en el directorio de configuración de Home Assistant ('/config/') con el nombre home-assistant_v2.db. Si desea modificar los ajustes por defecto de la grabadora en su instalación, simplemente añada el siguiente contenido a su archivo denominado "configuration.yaml".

Example configuration.yaml entry

recorder:

Mediante la implementación exitosa de los procedimientos mencionados se ha permitido lograr una integración altamente eficiente entre la pantalla de operador, los sensores, los actuadores, los microcontroladores y el uso de sistemas embebidos para la tecnología IoT. Este enfoque holístico ha posibilitado una operación armoniosa al unificar todas estas componentes en un único panel de control. La notable ventaja de esta solución radica en su versatilidad, ya que se ha logrado adaptar este sistema integrado para su utilización en dispositivos Android e IOS. Al permitir el acceso y monitoreo del sistema domótico desde estos dispositivos móviles ampliamente utilizados, se ha alcanzado un nivel de comodidad y control para los usuarios. En consecuencia, esta implementación promete no solo una mayor eficiencia operativa, sino también una experiencia de usuario mejorada y simplificada en el ámbito de la automatización residencial.

CAPÍTULO 3

3. ANÁLISIS Y RESULTADOS

3.1 Tableros del sistema domótico

Se implementaron dos tableros de 52x52 cm en el cual se encuentran los dispositivos del sistema domótico, se diseñó de esta manera para mostrar la comunicación inalámbrica entre ambos, ya que en el tablero #1 tenemos la centralita con la pantalla táctil que contiene la interfaz gráfica y en el tablero #2 el microcontrolador ESP32 con los sensores y actuadores.

Figura 3.1 Tableros del sistema domótico



Estos tableros permiten visualizar de manera clara y práctica la interacción y funcionalidad del sistema. Esta disposición física de los elementos ha brindado una representación tangible de cómo los dispositivos se comunican y cooperan para lograr la automatización y control inteligente de un entorno. En última instancia, esta configuración de tableros ha facilitado la comprensión de las capacidades y beneficios de la domótica,

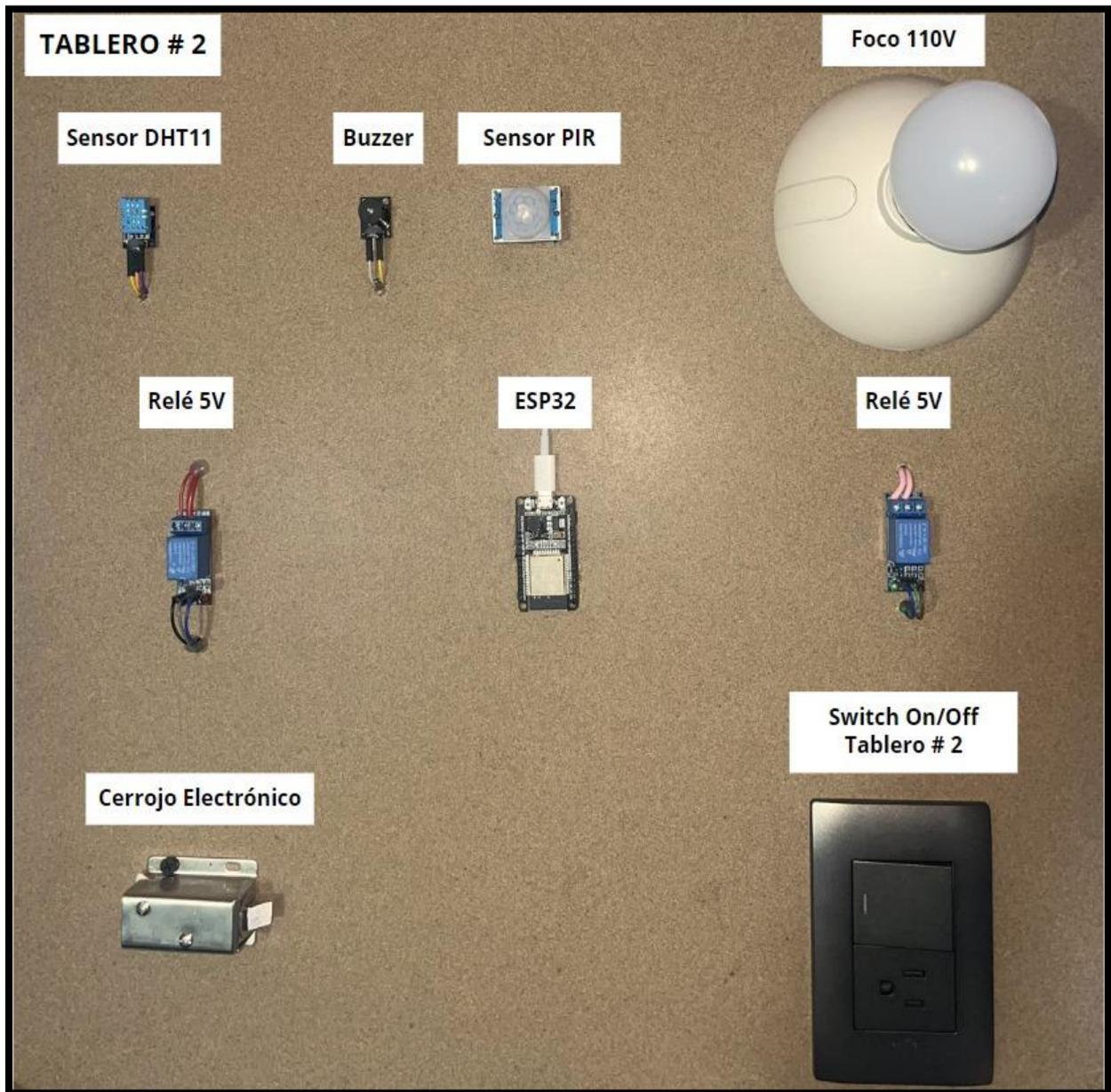
resaltando la importancia de la integración de la tecnología inalámbrica y la interfaz intuitiva en el diseño de sistemas para el hogar más eficientes y convenientes.

Figura 3.2 Tablero # 1



Nota. En la figura 3.2 se muestra el tablero # 1 detallando cada uno de sus componentes.

Figura 3.3 Tablero # 2



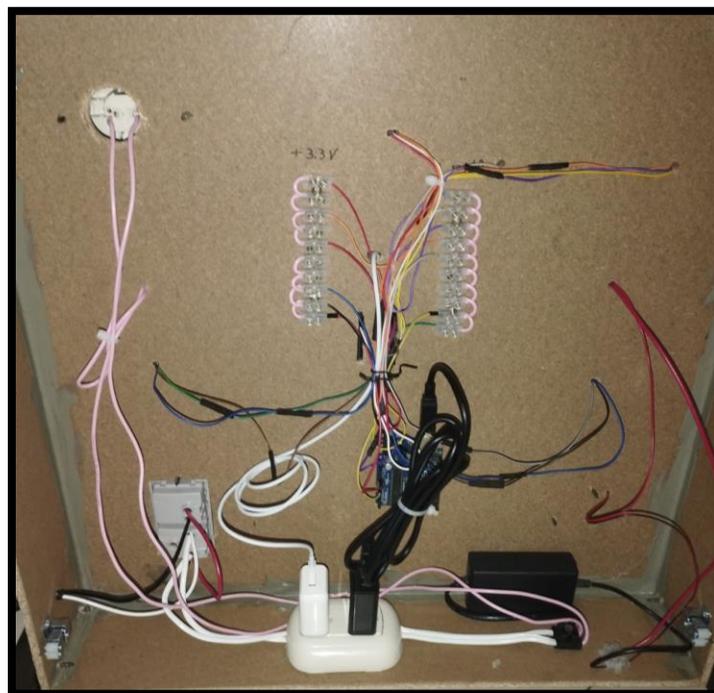
Nota. En la figura 3.3 se muestra el tablero # 2 detallando cada uno de sus componentes.

Figura 3.4 Vista Posterior del Tablero # 1



Nota. En las conexiones del tablero # 1 solo se emplean 3 cargadores que es para la Tablet, Router y Raspberry Pi 4

Figura 3.5 Vista Posterior del Tablero # 2

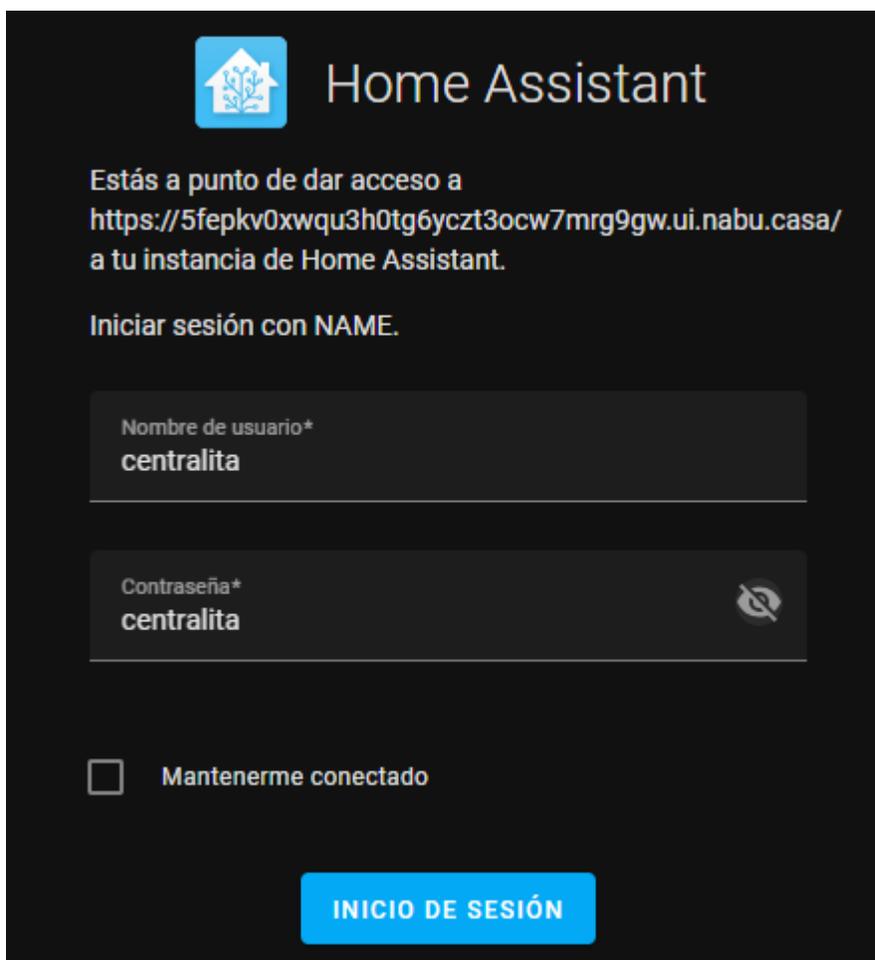


Nota. En las conexiones del tablero # 2 se realizó la conexión de los diferentes GPIO que controlan los diferentes sensores y actuadores, una alimentación de 3.3V, GND, cargador de 12V, cargador de 5V y la conexión AC para la boquilla.

3.2 Acceso a la Interfaz gráfica

Al encender la centralita automáticamente se inicia el servidor y podremos acceder a ingresar nuestras credenciales para visualizar nuestro panel de control y todo el menú de home Assistant escribiendo la dirección mostrada en la figura en cualquier navegador. Para acceder de manera rápida instalamos la aplicación de home Assistant en el dispositivo que sirve de panel de control, que es la Tablet.

Figura 3.6 Credenciales del Sistema



Home Assistant

Estás a punto de dar acceso a
<https://5fepkv0xwqu3h0tg6yczt3ocw7mrg9gw.ui.nabu.casa/>
a tu instancia de Home Assistant.

Iniciar sesión con NAME.

Nombre de usuario*
centralita

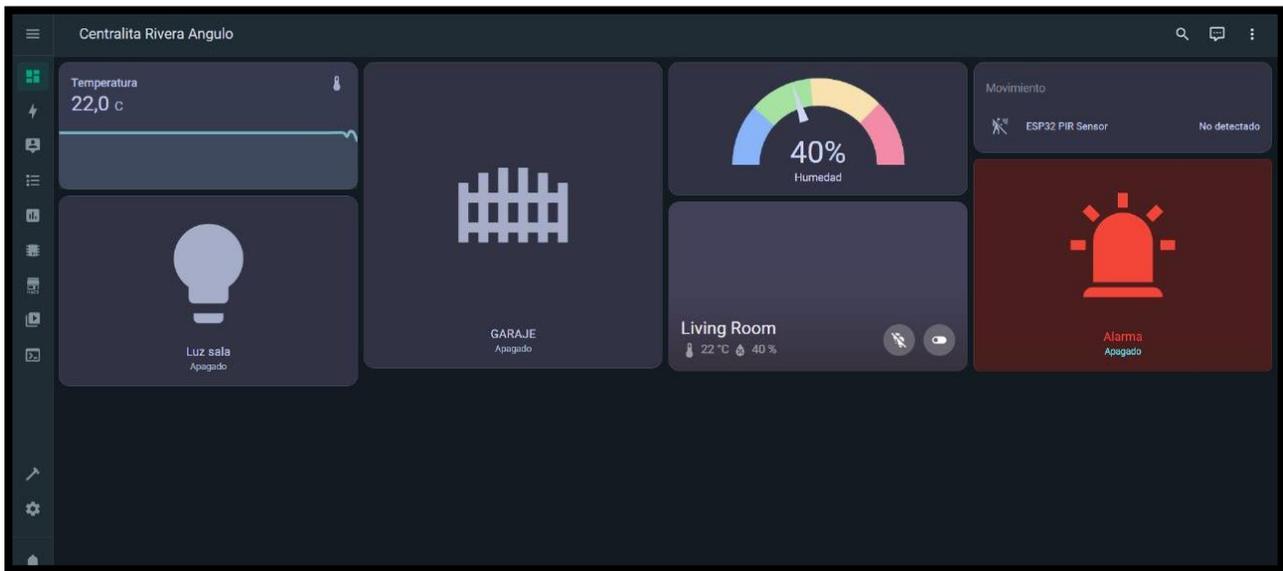
Contraseña*
centralita

Mantenerme conectado

INICIO DE SESIÓN

Luego de ingresar las credenciales se puede iniciar sesión, en nuestro panel de control no será necesario ingresarla cada vez, solo será necesario si se reinicia la centralita.

Figura 3.7 Interfaz gráfica del sistema



El diseño de la interfaz gráfica es agradable visualmente ya que muestra de manera simple los datos de medición en tiempo real de nuestros sensores, además de brindar la capacidad de interactuar de manera física con los botones que activan y desactivan los actuadores como son los relés que comandan luces y cerraduras.

Figura 3.8 Pantalla táctil con interfaz gráfica del sistema



3.3 Pruebas del sistema

3.3.1 Sensor DHT11

Se realizó la prueba del sensor de humedad y temperatura exitosamente, en el panel se logra observar en una tarjeta el valor en grados centígrados de la temperatura en una escalada de 0 – 50°C y el valor de la humedad en porcentajes del 20 al 90%. Esto lo podemos observar más detalladamente en la consola de programación de ESPHome ya que nos muestra el estado del sensor cada 5 segundos, que fue el intervalo de tiempo de censado programado con anterioridad.

Figura 3.9 Valor actual e Histórico de Temperatura

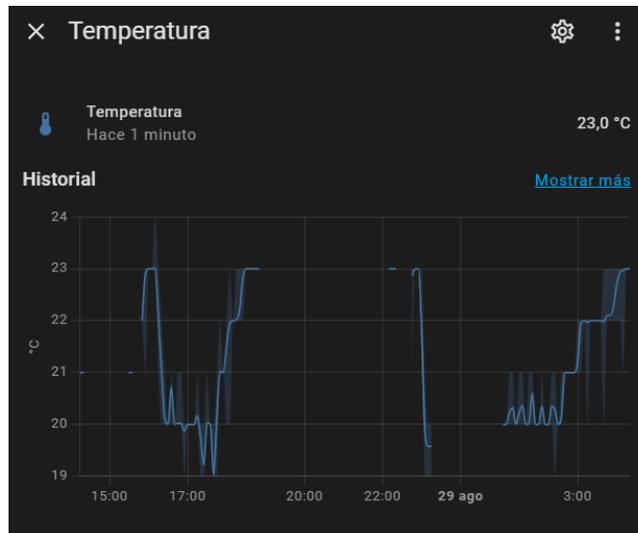


Figura 3.10 Valor actual e Histórico de Humedad



Figura 3.11 Temperatura y Humedad sensada desde la consola de ESPHome Logs

```
Logs esp32.yaml
[04:28:08][C][switch.gpio:031]: Pin: GPIO23
[04:28:08][C][switch.gpio:068]: GPIO Switch 'buzzer'
[04:28:08][C][switch.gpio:091]: Restore Mode: always OFF
[04:28:08][C][switch.gpio:031]: Pin: GPIO23
[04:28:08][C][gpio.binary_sensor:015]: GPIO Binary Sensor 'PIR Sensor'
[04:28:08][C][gpio.binary_sensor:015]: Device Class: 'motion'
[04:28:08][C][gpio.binary_sensor:016]: Pin: GPIO13
[04:28:08][C][light:103]: Light 'Desk Lamp'
[04:28:08][C][dht:017]: DHT:
[04:28:08][C][dht:018]: Pin: GPIO4
[04:28:08][C][dht:020]: Auto-detected model: DHT11
[04:28:08][C][dht:027]: Update Interval: 5.0s
[04:28:08][C][dht:029]: Temperature 'Temperature'
[04:28:08][C][dht:029]: Device Class: 'temperature'
[04:28:08][C][dht:029]: State Class: 'measurement'
[04:28:08][C][dht:029]: Unit of Measurement: '°C'
[04:28:08][C][dht:029]: Accuracy Decimals: 1
[04:28:08][C][dht:030]: Humidity 'Humidity'
[04:28:08][C][dht:030]: Device Class: 'humidity'
[04:28:08][C][dht:030]: State Class: 'measurement'
[04:28:08][C][dht:030]: Unit of Measurement: '%'
[04:28:08][C][dht:030]: Accuracy Decimals: 0
[04:28:08][C][capture_port:088]: Capture Portal:
[04:28:08][C][mdns:111]: mDNS:
[04:28:08][C][mdns:111]: Hostname: esp32
[04:28:08][C][ota:093]: Over-The-Air Updates:
[04:28:08][C][ota:094]: Address: esp32.local:3232
[04:28:08][C][ota:097]: Using Password.
[04:28:08][C][api:138]: API Server:
[04:28:08][C][api:139]: Address: esp32.local:6053
[04:28:08][C][api:141]: Using noise encryption: YES
[04:28:09][D][dht:048]: Got Temperature=23.0°C Humidity=42.0%
[04:28:09][D][sensor:094]: 'Temperature': Sending state 23.00000 °C with 1 decimals of accuracy
[04:28:09][D][sensor:094]: 'Humidity': Sending state 42.00000 % with 0 decimals of accuracy
```

Nota. De esta forma se comprueba que los valores de temperatura y humedad se van actualizando cada 5 segundos.

3.3.2 Sensor de movimiento PIR

La alarma programada mediante un sensor PIR y un buzzer fue puesta a prueba con la opción de notificación del sistema, es decir que cuando el sensor detecte presencia enviará una notificación al dispositivo en el que estemos en sesión del Home Assistant.

Figura 3.12 Panel con sensor PIR activado



Figura 3.13 Pantalla con buzzer activado

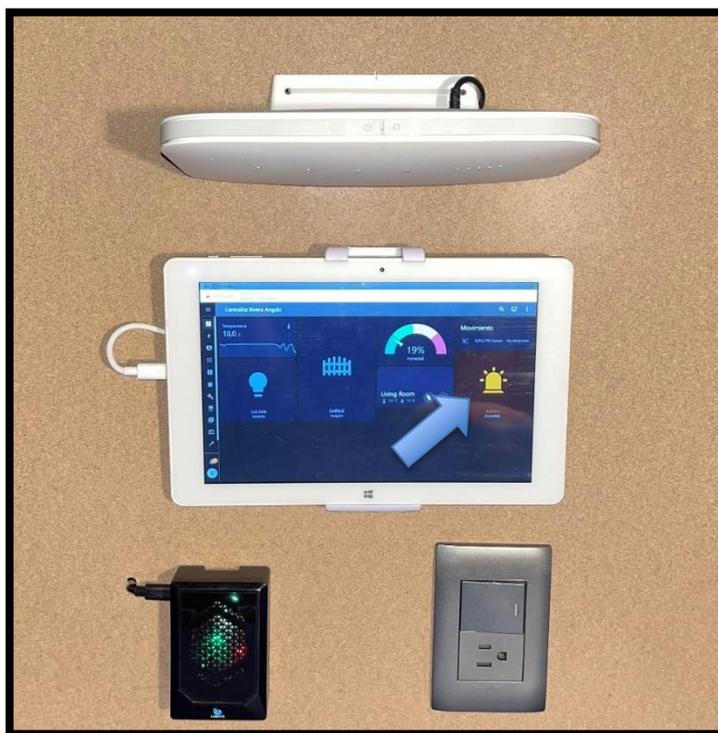
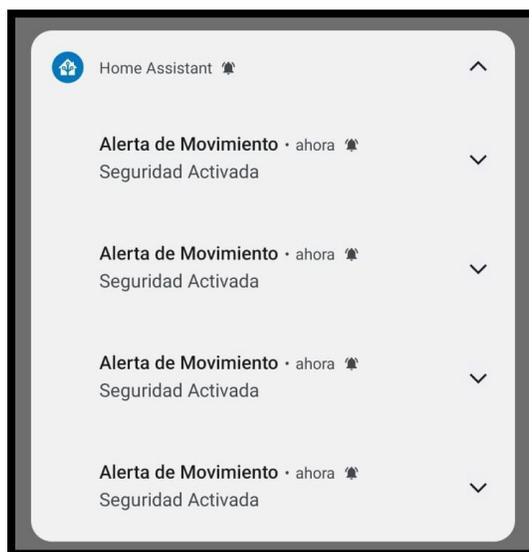


Figura 3.14 Notificaciones de Alerta de Movimiento



Nota. Cada vez que se registre un movimiento la automatización creada en Home Assistant va a notificar un mensaje de alerta al dispositivo vinculado.

3.3.3 Encendido de luz

Para realizar el control de encendido y apagado del foco basta con pulsar la tarjeta de luz en la pantalla táctil. La tarjeta tiene como propiedad notificarnos el estado del actuador a través del color amarillo para encendido y color gris cuando se encuentra apagado.

Figura 3.15 Panel con Luz Activada

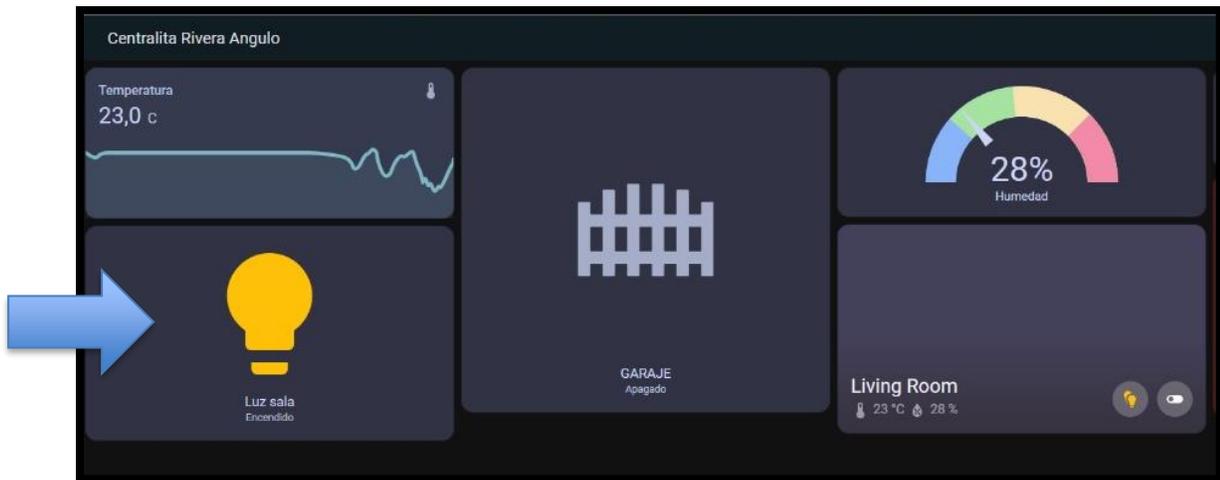


Figura 3.16 Pantalla con Luz Activada



Figura 3.17 Luz Apagada

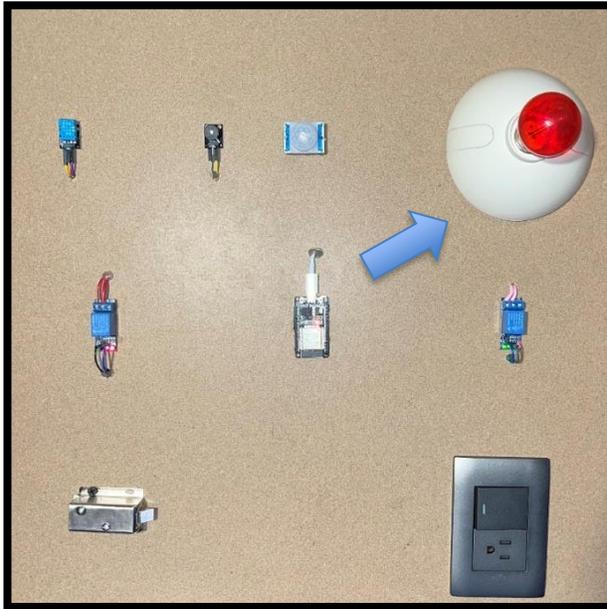
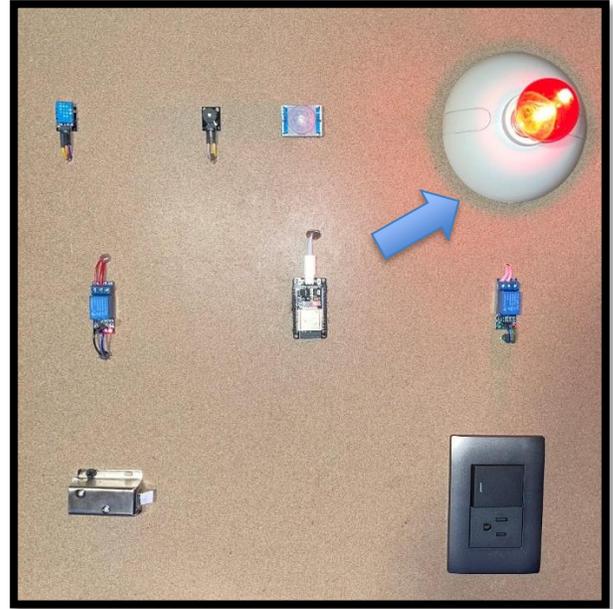


Figura 3.18 Luz Encendida



3.3.4 Apertura de cerradura eléctrica

Para activar y desactivar la cerradura eléctrica damos un toque a la pantalla en la tarjeta Garaje, la cual nos indica el estado de la cerradura a través de dos colores que son amarillo y gris, siendo el primero para indicar la cerradura abierta y el segundo para mostrar lo opuesto.

Figura 3.19 Panel con Cerradura Activada



Nota. Se nota que los objetos cuando están activados cambian a color amarillo.

Figura 3.20 Pantalla con Cerradura Activada

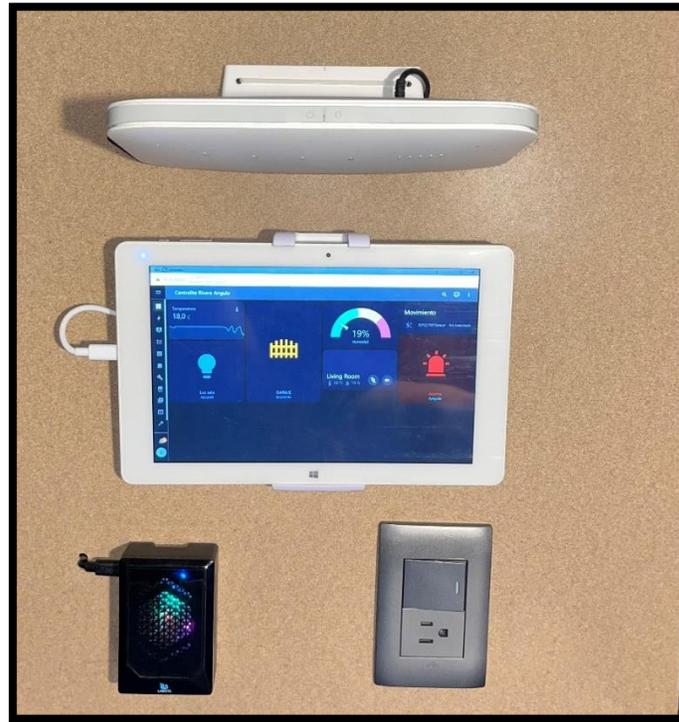
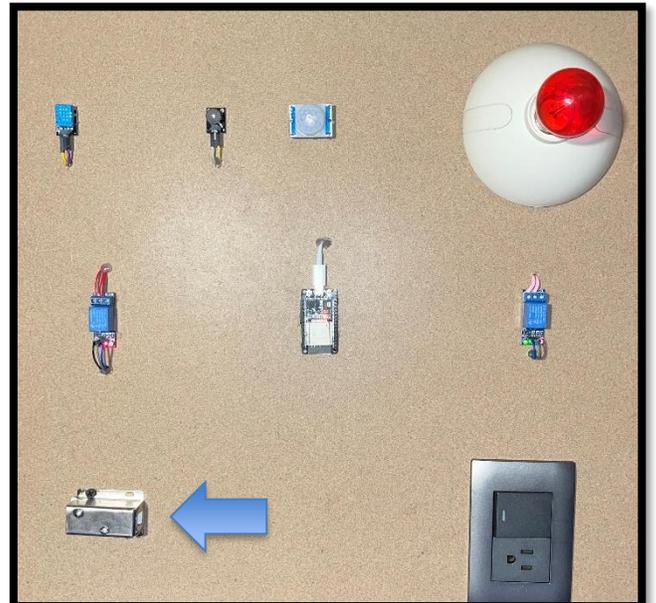


Figura 3.21 Cerradura Cerrada



Figura 3.22 Cerradura Abierta



3.3.5 Comandos de voz con Amazon Alexa

Home Assistant nos permite integrar diversos asistentes de voz, para esta prueba se utilizó el Amazon Alexa del dispositivo Echo Dot 3, el cual se vincula a nuestro sistema y recibirá los comandos de voz para activar actuadores y brindarnos información de los sensores.

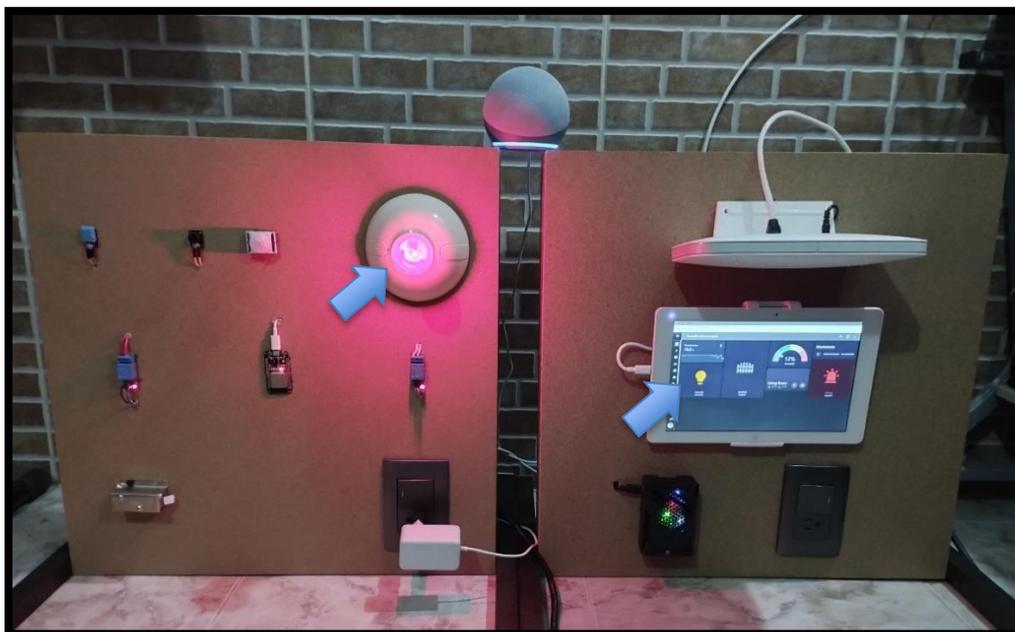
Figura 3.23 Echo Dot 3 en Espera



Nota. Cuando Echo Dot 3 enciende las luces después de decir “Alexa” es momento de dictar a Alexa un comando.

Encendido de luces con Amazon Alexa (comando de voz: “Alexa, encender luz tablero 2”), (Alexa: “Muy Bien”)

Figura 3.24 Encendido de Luz con Amazon Alexa



Apertura de cerradura con Amazon Alexa:

(comando de voz: "Alexa, abrir puerta tablero 2")

(Alexa: "Cuál es tu código de voz para puerta tablero 2")

(comando de voz: "1234")

(Alexa: "Abriendo, el dispositivo puerta tablero 2 esta desbloqueado")

Figura 3.25 Apertura de cerradura con Amazon Alexa



Nota. En la figura 3.25 se puede apreciar que al terminar de decir la contraseña para abrir la puerta Alexa envía el comando a Home Assistant y abre la cerradura.

Encendido de sirena con Amazon Alexa (comando de voz: "Alexa, encender sirena tablero 2")

Figura 3.26 Encendido de Sirena con Amazon Alexa



Para pedir la temperatura actual del sensor DHT11 basta con decir “Alexa, cual es la Temperatura de tablero 2” y Alexa va a responder el valor de la Temperatura en ese momento. Sin embargo, es necesario mencionar que para la humedad no sucede lo mismo ya que Amazon Alexa todavía no integra un dispositivo para agregarlo como humedad.

3.3.6 Comando de Voz con Assist

Assist es un asistente de voz que viene integrado en Home Assistant por lo que al igual que Amazon Alexa obedece instrucciones según sea el comando de voz configurado, pero a diferencia de Alexa con Assist también podemos escribir lo que queremos realizar y personalizar cualquier comando.

Figura 3.27 Comando de Voz con Assist



Nota. En la figura 3.27 podemos ver alguno de los comandos de voz que podemos interactuar con Assist, también se nota la diferencia en el comando de abrir puerta que no nos pide ninguna clave de autorización y esto es debido a que Assist es algo propio de Home Assistant que viene integrado en el panel de resumen.

Los comandos que se puede utilizar con Assist son:

{sensor de movimiento: ¿cómo está movimiento?}

{sirena: encender sirena; apagar sirena}

{luces: encender luz; apagar luz}

{temperatura: ¿cuál es la temperatura?}

{humedad: ¿cuál es la humedad?}

{puerta: abrir puerta}

De esta manera se comprueba que todas las operatividades propuestas en la metodología son válidas y funcionan de manera correcta dando como resultado un sistema viable en cuanto a sus funcionalidades, ahora vamos a analizar la viabilidad en cuanto al factor económico.

3.4 Análisis de costos

En la tabla 3.1 se detalla el costo de implementación de este sistema domótico para analizar la viabilidad con relación a otros sistemas existentes.

Tabla 3.1 Costos del Sistema Domótico Centralizado

Cantidad	Descripción	Precio (\$)
1	Raspberry Pi 4	\$ 185
1	ESP32	\$ 12
1	Sensor PIR	\$ 2,50
1	Sensor DHT11	\$ 2,20
2	Relé 5v	\$ 2,25
1	Módulo buzzer	\$ 1,90
1	Juego de cables Jumper	\$ 4
1	Tablet 7 pulgadas	\$ 50
2	Interruptor 110v	\$ 6,40
1	Boquilla de foco 110v	\$ 2,50
1	Cerradura eléctrica	\$ 3,00
1	Foco	\$ 1,5
1	Mano de Obra Ingeniería	\$ 81,98
TOTAL		\$ 355.23

El costo total de la implementación del sistema domótico es de \$ 231,72 para determinar si este valor es significativo es necesario revisar los beneficios que tendrá el inmueble donde estará instalado, como son:

- Monitoreo de climatización en el interior (automatizaciones a equipos de climatización).
- Seguridad con las alarmas que involucran sensores de presencia.
- Facilidad del control de luces y electrodomésticos en el inmueble.
- Control y acceso a todas estas funciones desde cualquier lugar donde esté disponible el internet a través del control remoto de Home Assistant.
- Integraciones a control por comandos de voz como Amazon Alexa o Google Assistant.

Dadas todas las funciones del sistema podemos decir que el precio con relación a las funciones es viable, además comparamos dicho valor con dispositivos existentes en el mercado.

Tabla 3.2 Costos de dispositivos domóticos en el mercado

Cantidad	Producto	Valor en el mercado
1	Interruptor inteligente marca Broadlink	\$ 25
1	Control de puerta marca Tuya	\$ 49
1	Interruptor wifi marca Tuya	\$ 16
1	Módulo relé dual marca Tuya	\$ 44
1	Centralita marca Z Wave	\$ 323
1	Termómetro / Higrómetro marca SensorPush	\$ 50
1	Bocina de sirena marca HORN	\$ 11
1	Sensor movimiento marca Smart Switch	\$ 25
1	Foco marca Sengled	\$ 6,5

1	Mano de obra de Ingeniería	\$ 164.85
TOTAL		\$ 714,35

Realizando la comparación entre lo que se realizó en este proyecto vs lo que ya existe en el mercado, se aprecia un ahorro dramático en cuanto a la inversión total de la implementación, economizando hasta un 67.56% en cuanto a las tecnologías del mercado actual.

Existen interruptores inteligentes que son comandados por señal Wi-Fi que van desde los \$ 15, esto en medianas instalaciones representa un costo elevado ya que solo nos brinda el servicio de apagar y encender luces inalámbricamente, mientras que el sistema domótico implementado cuenta con muchas integraciones manejadas desde una central, es decir, no es necesario instalar varios softwares para poder manipular los sensores o actuadores que se instalen en el hogar así sean de diferentes marcas.

Dentro del análisis de costos se incluye la mano de obra de ingeniería la cual es considerada acorde al alcance del proyecto, es decir se cobra por la implementación del sistema una tarifa base del 30% de la inversión total de los materiales, siendo una instalación de acuerdo al alcance que tiene el sistema en este documento. Se añaden valores de facturación adicionales por cada módulo ESP32 que el cliente desee añadir, este valor se consideró como el 30% del costo del módulo.

CAPÍTULO 4

4. CONCLUSIONES Y RECOMENDACIONES

4.1 CONCLUSIONES

- Se cumplió con el objetivo de combinar sistemas embebidos, comunicación Wifi y tecnología IoT para crear un sistema domótico centralizado mostrando un enfoque en la utilización de tecnologías de vanguardia. Esta integración permite a los estudiantes aprender sobre la convergencia de diferentes disciplinas tecnológicas y cómo se pueden aplicar en la automatización de hogares y edificios.
- Se logró integrar una pantalla de operador ya que el diseño del sistema no solo se centra en la automatización técnica, sino también en la experiencia del usuario. Los estudiantes pueden aprender sobre la importancia de la interfaz de usuario, la facilidad de uso y la capacidad de controlar y monitorear el sistema de manera eficiente.
- La comparación exhaustiva de las tecnologías disponibles en el mercado proporcionó una visión clara de las opciones disponibles para los componentes del sistema domótico. Esta fase permitió identificar las tecnologías más adecuadas para cumplir con los requisitos del proyecto y proporcionar una base sólida para la toma de decisiones informadas.
- La fase de validación en un entorno físico como lo son los tableros fue esencial para evaluar el desempeño, la confiabilidad y la eficacia del sistema. Además, de que el sistema domótico diseñado ha sido probado de forma real y puede ser aplicado no sólo en la enseñanza, sino que es una alternativa económica y efectiva para su aplicación en cualquier casa con tecnología Wi-Fi.

4.2 RECOMENDACIONES

- El sistema domótico al tener acceso a casi todo el hogar debe de ser lo más seguro, por esto se recomienda utilizar credenciales de usuario y contraseña de difícil acceso y que no sean compartidas con personas que no habiten en el hogar.
- Para realizar las conexiones entre el microcontrolador ESP32 con los diferentes sensores y actuadores se debe asegurar que en el cálculo de corriente para cada GPIO no debe superar los 40mA, caso contrario se debe incorporar una fuente externa para brindar el suministro de corriente necesario y cabe mencionar que para la extensión a un hogar se emplearían varios módulos de ESP32 con sus sensores y actuadores.
- Cuando realizamos la programación referente a la notificación de algún dispositivo vinculado a home Assistant, comprobar que la acción que va a desencadenar el mensaje de alarma supere los 500ms de duración, puesto que home Assistant no recibe valores inferiores.
- Si queremos un funcionamiento óptimo de Home Assistant es recomendable no instalarlo en un contenedor si no en la totalidad de la Raspberry Pi 4, esto hará que se dedique solo a actuar como servidor y no compartir más tareas gastando memoria y volviendo lento el sistema.

Referencias

LIBROS FÍSICOS

- Albert S. Huang, L. R. (2007). *Bluetooth Essentials for Programmers*. New York: Cambridge University Press.
- Ayala, K. J. (2013). *The 8051 Microcontroller: Architecture, Programming and Applications*. New York, Los Angeles, San Francisco: West Publishing Company.
- Banzi M., S. M. (2014). *Arduino Uno: A Hands-On Guide for Be*. U.S.A: MAKERMEDIA.
- Elahi, A. (2009). *ZigBee Wireless Sensor and Control Network*. U.S.A.: PEARSON.
- I., S. (2016). *Software Engineering*. Boston: PEARSON.
- J., F. (2010). *Handbook of Modern Sensors: Physics, Designs, and Applications*. San Diego: Advanced Monitors Corporation.
- Kranz, M. (2017). *Building the Internet of Things*. New Jersey: Wiley.
- Nuno M. Garcia, A. P. (2014). *Smart Homes and Health Telematics*. Denver: Revised Papers.
- Ohrman, F. (2003). *Wi-Fi Handbook: Building 802.11b Wireless Networks*. U.S.A.: McGraw-Hill Professional.
- P., L. (2018). *Exploring the ESP32: Get started with Internet of Things*. U.S.A.
- Stallings, W. (2007). *Wireless Communications and Networks*. U.S.A.: MK.
- Tanenbaum A., B. H. (2014). *Modern Operating Systems*. U.S.A: PEARSON.
- Tidwell, J. (2020). *Designing Interfaces: Patterns for Effective Interaction Design*. U.S.A: OREILLY.
- Upton E., H. G. (2020). *Raspberry Pi User Guide*. U.S.A.: WILEY.
- W., B. (2015). *Mechatronics: Electronic Control Systems in Mechanical and Electrical Engineering*. U.S.A.: PEARSON.
- W., S. (2018). *Operating Systems: Internals and Design Principles*. Boston: PEARSON.

ARTÍCULOS EN LÍNEA

- Canarias, G. d. (13 de 11 de 2018). *Gobierno de Canarias*. Obtenido de <https://www3.gobiernodecanarias.org/medusa/ecoblog/mgoncal/files/2018/11/unidad-1-hardware-tic-4-eso-ies-playa-honda-1.pdf>
- Meza, I. D. (15 de 4 de 2018). *TECSUP*. Obtenido de https://www.academia.edu/364440030/Lab_03_Temporizadores_y_contadores
- Noriega, S. (22 de 2 de 2008). *catedra.ing.unlp.edu.ar*. Obtenido de <https://catedra.ing.unlp.edu.ar/electrotecnia/islyd/Tema%2011%20Conversores%202008%20BYN.pdf>
- Uruguay, U. d. (15 de mayo de 2017). *FING*. Obtenido de <https://www.fing.edu.uy/tecnoinf/mvd/cursos/arqcomp/material/teo/arq-teo06.pdf>
- Yáñez, E. (5 de 2 de 2020). *eduteka*. Obtenido de Universidad ICESI: <https://eduteka.icesi.edu.co/proyectos.php/1/21608>

PAPERS ACADÉMICOS

- R. Meenakshi, S. Arulmurugan y S. Nallusamy. (2017) International Journal of Engineering Research & Technology. "A Survey on Centralized Home Automation Systems"
- M. H. Parvez, A. S. M. Mohsin y M. S. Hossain. (2014) International Journal of Engineering & Technology. "Design and Implementation of a Centralized Home Automation System"
- S. Kumar y V. Aggarwal. (2016) International Journal of Computer Applications "A Framework for Centralized Home Automation Systems Based on Internet of Things"
- S. Dwivedi, M. Bhatt y P. Joshi. (2015) International Journal of Advanced Research in Computer Science and Software Engineering. "Centralized Home Automation and Control using Raspberry Pi"
- N. S. Srinidhi y S. M. Pujar. (2016) International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering. "Centralized Home Automation System using IoT"

Anexos

Anexo1: Implementación Tablero # 1



Anexo 2: Implementación Tablero # 2



Anexo 3: Implementación de Tableros

