



**UNIVERSIDAD TECNOLÓGICA ISRAEL**

**TRABAJO TITULACIÓN EN OPCIÓN AL GRADO DE:  
INGENIERO EN ELECTRONICA DIGITAL Y  
TELECOMUNICACIONES**

**TEMA:**

PROTOTIPO DE UN SISTEMA DE UNA CASA INTELIGENTE, CONTRALADO A  
TRAVÉS DE COMANDOS DE VOZ PARA PERSONAS CON DISCAPACIDAD DE  
SUS EXTREMIDADES SUPERIORES

**AUTOR:**

**HÉCTOR DANIEL PACHACAMA SOTOMAYOR**

**TUTOR:**

**ING. RENÉ ERNESTO CORTIJO LEYVA, Mg.**

**QUITO, ECUADOR**

**2020**

## DECLARACIÓN

Yo, HÉCTOR DANIEL PACHACAMA SOTOMAYOR con C.I.N°1722511035, declaro que el presente proyecto de tesis de grado, denominado “PROTOTIPO DE UN SISTEMA DE UNA CASA INTELIGENTE, CONTRALADO A TRAVÉS DE COMANDOS DE VOZ PARA PERSONAS CON DISCAPACIDAD DE SUS EXTREMIDADES SUPERIORES”, es de mi autoría el cual se lo ha realizado de manera íntegra y respetando los derechos intelectuales de las personas y a la vez adjuntando conceptos mediante citas en las cuales se detallan de manera más completa en la bibliografía. Debido a lo expuesto en esta declaración, me responsabilizo del contenido, la intensidad y el alcance del proyecto por lo tanto cedo los derechos a la Universidad Tecnológica Israel para que el uso del mismo como materia de consulta o lectura.

Quito, 22 de agosto de 2020

AUTOR

.....

Héctor Daniel Pachacama Sotomayor

# UNIVERSIDAD TECNOLÓGICA ISRAEL

## APROBACIÓN DEL TUTOR

En mi calidad de tutor del trabajo de titulación certifico:

Que el trabajo de titulación **“PROTOTIPO DE UN SISTEMA DE UNA CASA INTELIGENTE, CONTRALADO A TRAVÉS DE COMANDOS DE VOZ PARA PERSONAS CON DISCAPACIDAD DE SUS EXTREMIDADES SUPERIORES.”**, presentado por el Sr. **Héctor Daniel Pachacama Sotomayor**, estudiante de la carrera de Electrónica Digital y Telecomunicaciones, reúne los requisitos y méritos suficientes para ser sometido a la evaluación del Tribunal de Grado, que se designe, para su correspondiente estudio y calificación.

Quito, 22 de agosto de 2020

TUTOR

.....

Ing. Rene Ernesto Cortijo Leyva, Mg.

## **AGRADECIMIENTO**

Quiero agradecer en primer lugar a Dios, a mis Padres por su sacrificio diario, hermanas, quienes supieron apoyarme para seguir adelante con la culminación exitosa de mis estudios.

A la Universidad Tecnológica Israel, por la culminación profesional académica y a los distintos profesores de la carrera que aportaron con nuevos conocimientos para la realización de este proyecto.

Héctor Daniel Pachacama Sotomayor

## TABLA DE CONTENIDO

DECLARACIÓN	ii
APROBACIÓN DEL TUTOR	iii
AGRADECIMIENTO	iv
INTRODUCCIÓN	1
Antecedentes de la situación objeto de estudio.	1
Presentación y justificación del problema.	2
Objetivo General	3
Objetivos Específicos	3
Alcance	3
Descripción de Capítulos	4
CAPÍTULO 1	5
FUNDAMENTACIÓN TEÓRICA	5
1.1    La Domótica	5
1.2    Casa inteligente	5
1.3    Sistema de Reconocimiento Automático del Habla (SRAH)	6
1.4    Reconocimiento de voz por computadora	6
1.5    Asistentes de voz inteligentes	7
1.5.1    Siri	8
1.5.2    Alexa	8
1.5.3    Asistente de Google	8
1.5.4    Alisa	8
1.6    Raspberry pi	9
1.6.1    Pines GPIO	9
1.7    Transductores y sensores	10

1.7.1	Transductor	11
1.7.2	Sensor	11
1.8	Lenguajes de programación	11
1.8.1	Python	11
1.8.2	HTML	12
1.8.3	PHP	12
1.9	Sistemas de alarma	12
1.10	Sistemas de control de acceso	13
CAPÍTULO 2		14
MARCO METODOLÓGICO		14
CAPÍTULO 3		16
3.1	Propuesta	16
3.2	Diseño general del sistema	16
3.2.1	Diseño del sistema de Alarma	18
3.2.2	Diseño del sistema de control de acceso	20
3.2.3	Diseño del sistema eléctrico (luces)	24
3.3	Elementos del sistema	25
3.3.1	Raspberry pi	25
3.3.2	Detector de movimiento	27
3.3.3	Lectora de proximidad	28
3.3.4	Contactos magnéticos	28
3.3.5	Placa de interfaz de relé 16 canales	28
3.3.6	Fuente de Alimentación	29

3.3.7	Micrófono inalámbrico	31
3.3	Diagrama de flujo del programa	33
3.4	Configuración y puesta en marcha de la Raspberry	34
3.5.1	Librería Annyang	34
3.5.2	Librería Pico2wave	35
3.5.3	Servidor web Apache	35
3.5.4	Funcionamiento de Apache	35
3.6	Presupuesto de la implementación.	36
CAPÍTULO 4		39
IMPLEMENTACIÓN		39
4.1	DESARROLLO	39
4.2	Cableado de sensores y cerraduras	40
4.3	Cableado para control de luces	40
4.4	Instalación y armado del tablero de control	40
4.5	Instalación del sistema de alarma.	42
4.6	Control de accesos.	43
4.6.1	Conexión de lectora de tarjetas RFID	44
4.6.2	Conexión de Cerradura electromagnética	45
4.6.3	Conexión de Micrófono Inalámbrico	45
4.7	Desarrollo de la programación	46
4.7.1	Instalación del sistema operativo Raspbian	46
4.7.2	Asignación de una IP estática	47
4.7.3	Conexión SSH	48
4.7.4	Entorno de trabajo e instalación de Actualizaciones	49

4.7.5	Instalación del servidor web Apache y PHP	50
4.7.6	Configuración de modo seguro HTTPS	50
4.8	Descripción de la programación.	51
4.9	Pruebas de funcionamiento	53
4.10	Análisis de resultados	57
CONCLUSIONES		59
RECOMENDACIONES		60
ANEXOS		61

## LISTA DE FIGURAS

Figura 1. Disposición de pines GPIO.....	10
Figura 2: Interacción de la Raspberry con Componentes Electrónicos .....	16
Figura 3: Esquema general de proyecto .....	18
Figura 4: Esquema conexión alarma.....	19
Figura 5: Esquema de conexión sensores PIR .....	20
Figura 7: Esquema de conexión cerradura.....	22
Figura 8: Circuito para disparo de relé.....	23
Figura 10: Plano eléctrico y control .....	24
Figura 11: Esquema de conexión focos .....	25
Figura 13: Detector de movimiento PIR .....	27
Figura 15: Módulo de Relés 16CH. ....	29
Figura 17: Micrófono inalámbrico VHF.....	32
Figura 18: Adaptador de Audio USB.....	32
Figura 19: Diagrama de flujo programa.....	33
Figura 20: Plano ubicación de elementos.....	39
Figura 21: Esquema de conexiones tablero de Control.....	41
Figura 22: Tablero de control.....	42
Figura 24: Instalación de sensor PIR .....	43
Figura 25: Instalación de cerradura eléctrica .....	44
Figura 27: Lectora de proximidad instalada.....	45
Figura 28: Instalación cerradura electromagnética .....	45

Figura 29: Conexión del micrófono.....	46
Figura 31: Verificación que se ha creado la carpeta SSH.....	47
Figura 32: Configuración IP estática .....	48
Figura 33: Conexión SSH a la Raspberry .....	48
Figura 34: Verificación de conexión a internet.....	49
Figura 35: Verificación de instalación del servidor Apache.....	50
Figura 36: Configuración HTTPS.....	51
Figura 37: Modificación del programa index2.html .....	54

## LISTA DE TABLAS

Tabla 1: Cálculo de la fuente de alimentación .....	30
Tabla 2: Rango de Frecuencias Espectro Radioeléctrico .....	31
Tabla 3: Equipos de implementación del sistema .....	37
Tabla 4 Conexión de dispositivos a placas de relés .....	42
Tabla 5: Comandos y acción que ejecuta el programa.....	52
Tabla 6: Pruebas de funcionamiento .....	55
Tabla 7: Prueba de reconocimiento hombre-mujer .....	56
Tabla 8: Resumen de pruebas .....	57

## RESUMEN

En el presente trabajo de titulación surge con el fin de facilitar a personas con discapacidad en sus extremidades superiores, a realizar algunas actividades cotidianas en el hogar, como encender y apagar luces, abrir puertas, activar alarma, que pueden ser muy insignificantes, pero tienen un nivel de dificultad mayor para personas con esta discapacidad. Para lo cual se plantea como solución la creación de un prototipo de una casa inteligente controlada a través de comandos de voz.

Esta investigación se basa en el uso de una tarjeta Raspberry PI para controlar y automatizar la programación de diferentes sistemas de control de iluminación, accesos y alarmas contra robos. La programación se basa en un algoritmo corriendo en un servidor web en este caso Apache, para el reconocimiento de comandos de voz se utiliza un micrófono inalámbrico, que permite el ingreso de datos a la Raspberry y una librería de comandos de java, que permitirá el control de los diferentes sistemas.

El proyecto incluye el control del sistema de iluminación: encendido y apagado de luces de tres ambientes sala, cocina y dormitorio; sistema de acceso apertura de puerta principal y una puerta interior, sistema de intrusión: está compuesto por sensores de movimiento PIR, contacto magnético, y una sirena de alarma.

El presente trabajo muestra el funcionamiento del prototipo, sensores y dispositivos que conforman, implementación pruebas de funcionamiento que se tuvo en la realización de este proyecto.

**Palabras Clave:** Raspberry, reconocimiento de voz, casa inteligente, domótica.

## ABSTRACT

The present title project is given in order to facilitate people with disabilities in their upper limbs, to perform some daily activities at home, such as turning on and off lights, opening doors, activating alarm, which can be very insignificant but have a higher level of difficulty for people with this disability. For which it is proposed as a solution the creation of a prototype of a smart house controlled by voice commands.

This research is based on the use of Raspberry PI cards to control and automate the programming of different lighting control, access and burglar alarm systems. The programming is based on an algorithm running on a web server, in this case Apache, for the recognition of voice commands a wireless microphone is used, which allows data entry to the Raspberry and a java command library, which will allow the control of the different systems.

The project includes the control of the lighting system: on and off of lights in three living room, kitchen and bedroom environments; Front door opening access system and an interior door, intrusion system: it consists of PIR motion sensors, magnetic contact, and an alarm siren.

This work shows the operation of the prototype, sensors and devices that make up, implementation of operational tests that was had in the realization of this project.

**Keywords:** Raspberry, voice recognition, smart home, home automation.

# INTRODUCCIÓN

## **Antecedentes de la situación objeto de estudio.**

El avance de la tecnología permite en la actualidad que la mayoría de equipos eléctricos y electrónicos puedan conectarse al internet, y estos a su vez puedan ser comandados a través de dispositivos móviles como celulares, tabletas inteligentes, logrando una interacción del usuario con las máquinas de una manera más simplificada.

Esto se logra a través del uso de diferentes plataformas, servidores y librerías, que realizan la lógica para el control de electrodomésticos o equipos electrónicos. La mayoría de dispositivos son creados en función de un ser humano común, pero no toman en cuenta que existen personas que tienen alguna discapacidad y que su manipulación de ciertos elementos se vuelve difícil.

Por esto es necesario desarrollar un prototipo de casa controlada a través de comandos de voz para personas con discapacidad en sus extremidades superiores mediante una tarjeta de control que en este caso es la Raspberry, junto a librería de internet que hacen posible el reconocimiento de voz y el control de diferentes sensores y actuadores, además se busca integrar varios subsistemas en un solo elemento de control, que su software sea libre y permita a desarrolladores que puedan desarrollar mejoras.

Según (Mejía, 2011) menciona en su trabajo el uso de un módulo de reconocimiento de voz denominado VR Stamp, desarrollado por la empresa Sensory Inc. y la utilización del programa Quick T2SI para el reconocimiento de voz, este sistema se basa en un sistema de varios micrófonos y altavoces. Este proyecto contribuye la utilización de un servidor web para interacción del humano con los dispositivos.

Otro documento presentado por (Tipán, 2018) en su trabajo titulado “Prototipo de una casa inteligente controlada a través de un celular android vía GSM”, propone un sistema de control de iluminación, control de temperatura, apertura y cierre de una persiana, puerta de garaje y un sistema de alarma, el control de los dispositivos lo realiza a través de un arduino que este a su vez se comunica con un celular a través de una aplicación web, con el fin de tener mayor

comodidad en nuestros hogares con la automatización de casas. Esta tesis aporta al presente trabajo en como considerar la arquitectura para sistemas de domóticas en casas.

Con esto se procede a desarrollar un prototipo de reconocimiento de voz basado en una librería java que permita al usuario interactuar con dispositivos electrónicos, el proyecto se llevara a cabo usando una plataforma computacional con la ayuda de lenguajes de programación de código abierto, que permitan que el sistema pueda seguir teniendo mejoras y así reducir costos.

### **Presentación y justificación del problema.**

En la rutina diaria existen cosas tan simples que se las realiza en el hogar como encender la iluminación, abrir y cerrar una puerta que son extremadamente sencillas de ejecutar, pero tales acciones para personas con discapacidad de sus extremidades es un poco más complejo realizar y a veces necesitan la ayuda de otras personas para realizar estas actividades.

En ecuador y en el mundo existe un porcentaje considerable de personas con discapacidad que tienen dificultad para maniobras de diferentes elementos, y la mayoría de sistemas eléctricos y electrónicos no son creados para que exista una inclusión de estas personas, y si existen tienen costos muy elevados, razón por la cual el motivo del presente proyecto.

Debido al avance de la tecnología esto es posible gracias a la domótica, ya que hoy en día disponemos de cientos de sensores que automatizan los procesos y que estos pueden ser controlados a través de comandos de voz.

Este proyecto trata de dar una solución a este problema, con la creación de un sistema que permita obtener datos de los sensores y poder procesar la información, actuar y tomar decisiones, esto es posible gracias a una Raspberry Pi y una librería de reconocimiento de voz, que permite interactuar con sensores, actuadores, a través de mensajes de voz.

Este proyecto se enfoca en solventar las necesidades de personas con discapacidad, mejorar sus condiciones de vida, dando mayor independencia al no requerir de otras personas para realizar las actividades antes mencionadas.

## **Objetivo General**

Implementar un prototipo domótico para una casa, contralado a través de comandos de voz para personas con discapacidad de sus extremidades superiores, que permita al usuario tener el control a través de comandos de voz.

## **Objetivos Específicos**

- Definir las principales variables a controlar.
- Diseñar el sistema electrónico de control, integrando sensores y actuadores.
- Desarrollar la programación para reconocer los comandos de voz y realizar el control de los actuadores del sistema a través de una Raspberry Pi.
- Implementar el prototipo del sistema en una casa.
- Verificar el funcionamiento del sistema mediante un protocolo de prueba.

## **Alcance**

Implementar el prototipo domótico en una casa con tres ambientes sala, cocina y un dormitorio, en el que permita validar el funcionamiento de los sensores y accionamientos del sistema.

El sistema estará compuesto por una serie de subsistemas que serán controlados a través de comandos de voz y se describe a continuación:

Sistema de iluminación. - Encendido y apagado de iluminación de tres compartimientos del hogar.

Sistema de Alarma. - A través de comandos de voz activará y desactivará el sistema de alarma antirrobo, este contiene dos sensores de movimiento, contactos magnéticos para la puerta principal y una sirena de aviso.

Desarrollar control para apertura de la puerta principal, esta tendrá una chapa eléctrica y se activará desde dentro de la casa por comandos de voz, para la apertura desde la calle se

instalará un lector de tarjetas RFID; y se realizará el control para la apertura de una puerta interna de la casa.

Procesamiento y reconocimiento de comandos de voz que permita controlar los diferentes sensores y actuadores a través de una tarjeta Raspberry PI y una librería de reconocimiento de voz y describir el modelo que utiliza la misma.

Funcionamiento: al ingresar a la casa deberá ingresar la clave a través de comandos de voz para desactivar la alarma, cuando el sistema de alarma este desactivado se podrán actuar sobre la iluminación, estos sistemas no funcionarán si la alarma esta activada.

El sistema realizará el reconocimiento simultáneo de máximo 3 palabras por comando ejemplo (Apagar luz sala), el número de comandos del sistema será máximo de 30 comandos y funcionan uno a la vez.

Crear un manual de usuario que describa el funcionamiento, uso y posibles fallas del sistema.

### **Descripción de Capítulos**

El presente proyecto consta de cuatro capítulos. El primer capítulo hace referencia a la fundamentación teórica sobre diferentes dispositivos que componen el proyecto y los fundamentos necesarios para la realización del mismo.

El segundo capítulo, se muestra el marco metodológico utilizado para esta investigación en la que describe la metodología utilizada para la elaboración del prototipo.

El tercer capítulo presenta la propuesta para el proyecto en el cual incluye diagramas de flujo de la lógica del programa, diseño electrónico de los subsistemas del prototipo, componentes del algoritmo.

Finalmente, el cuarto capítulo describe el proceso de implementación del sistema, así como el desarrollo de la programación desde la instalación del sistema operativo en nuestra Raspberry hasta el correcto funcionamiento del mismo, en la cual se detallan sus resultados y pruebas de funcionamiento.

# CAPÍTULO 1

## FUNDAMENTACIÓN TEÓRICA

### 1.1 La Domótica

“De hecho, la domótica es un campo técnico con una importancia social igualmente importante. Como primera aproximación, pero a costa de una sobre simplificación, confirmaremos que la domótica incluye la introducción de tecnologías de la información en el hogar para mejorar la calidad de vida de sus residentes y ampliar sus posibilidades de comunicación, automatizar los procesos del hogar y hacer que estos procesos se comuniquen con los residentes. Comunicarse entre sí y con la relación externa. Por tanto, la tecnología de la información es una herramienta, las personas son destinatarias y la satisfacción de determinadas necesidades es su objetivo” (Domínguez y Sáez, 2006).

La domótica es una alternativa diferente que representa un beneficio para las personas y se caracteriza por la complejidad que exista entre los dispositivos.

### 1.2 Casa inteligente

La vivienda es una de las opciones que son propensas a cambiar debido a la innovación tecnológica. El polémico término hace referencia al futuro de la sociedad de la información, estaremos inmersos en muchos dispositivos inteligentes, y estos dispositivos interactuarán de forma básica e intuitiva.

Las casas inteligentes incorporan tecnología capaz de actuar en función de la detección de presencia de algún individuo o responden a una consecuencia. Están conformados por una multitud de sistemas interconectados y estos a su vez se integran a objetos cotidianos.

El uso de la tecnología para dotar de inteligencia al espacio ambiental para que se adapte al individuo proporcionando memoria para el ambiente, para que aprenda de nuestras actividades diarias y simplifique la vida, ocultando tecnologías complejas.

### **1.3 Sistema de Reconocimiento Automático del Habla (SRAH)**

“El sistema de reconocimiento de voz automático puede gestionar las señales de voz enviadas por las personas. La señal se procesa digitalmente para obtener elementos de medida (muestras), que pueden expresar su comportamiento y realizar un proceso de procesamiento de señales enfocado a la identificación” (Oropeza, 2006).

“En la investigación de señales de voz, hay dos módulos importantes: formación y reconocimiento. La formación es una de las etapas más críticas en estos sistemas, y gran parte del éxito del sistema de reconocimiento de voz radica en esta etapa” (Oropeza, 2006).

“En la fase de formación del sistema, se pueden implementar de las siguientes formas:

- Bancos de filtro
- Codificación predictiva lineal
- Modelos ocultos de Markov
- Redes neuronales artificiales
- Lógica Difusa
- Sistema de reconocimiento híbrido” (Oropeza, 2006).

### **1.4 Reconocimiento de voz por computadora**

“El sistema de reconocimiento de voz por computadora es una tarea compleja de reconocimiento de patrones y sistema biométrico. La señal de voz que se puede muestrear está en el rango de frecuencia de 8 a 16 KHz. Una vez digitalizada la señal, se debe analizar para extraer la información relevante. Para ello existen muchos métodos, como el análisis de Fourier, la predicción de linealidad perceptual, etc. La característica básica de capturar la señal de voz es la frecuencia de la muestra, porque puede ser el factor limitante para distinguir la calidad de la señal” (Oropeza, 2006).

El estudio del reconocimiento de voz por computadora se empezó a inicios de los años 50's en los cuales lograron el reconocimiento de 10 sílabas monolocator, con esto se llevó a cabo estudios más especializados en los cuales se definen los fundamentos del sistema de reconocimiento, para los años 70's ya se tenía proyectos con reconocimiento con grandes vocabularios, para los años 80's se genera una expansión, algoritmos para el habla continua a través de modelos estadísticos en los cuales podemos mencionar los más importantes:

- “Modelos ocultos de Markov” (Oropeza, 2006).
- “Introducción a redes neuronales en el reconocimiento de voz” (Oropeza, 2006).

“En la década de 1990, los sistemas de reconocimiento de voz comenzaron a implementarse en computadoras y procesadores rápidos y económicos. Para el año 2000, el sistema de reconocimiento se hizo realidad, se implementó en el sistema operativo y por lo tanto en aplicaciones dedicadas a la gestión del reconocimiento de voz en teléfonos y sitios de Internet” (Oropeza, 2006, p.272).

### **1.5 Asistentes de voz inteligentes**

“El internet de las cosas ya es una realidad gracias a los nuevos avances de la tecnología que han permitido automatizar algunos procesos y nuestro objetivo es adaptarnos a ella. Se ha producido el cambio en como el usuario interactúa con las máquinas, buscando facilidad e inmediatez, esto ha favorecido a la utilización de asistentes virtuales de voz, ya que es posible dar una orden hablada y esperar que una maquina lo acate. Este es el escenario que da un inicio al uso de este tipo de asistentes inteligentes con el fin de facilitar procesos”. (Kanllli, Liege, & Lostalé, 2018).

“Los asistentes de voz son dispositivos que permiten realizar una búsqueda de información en internet a través de una petición verbal a un dispositivo inteligente como una computadora, teléfono o tabletas, que cada vez más está ingresando a nuestros hogares facilitando actividades cotidianas. Los asistentes hoy en día y gracias a “machine learning” y al “deep learning” los asistentes virtuales son capaces de reconocer el lenguaje con una precisión de

entre el 95% y el 97%, por lo que un asistente por un rango de comprensión menor no será fácilmente aceptado por los usuarios”. (Kanllli, Liege, & Lostalé, 2018).

Entre los más usados y conocidos tenemos a Siri-Apple, Alexa-Amazon, Asistente de Google, Alisa-Yandex, Cortana-Microsoft entre otros, los cuales permiten realizar actividades y tareas como controlar tu casa de forma inteligente, los cuales describiremos a continuación:

### **1.5.1 Siri**

Es un asistente de inteligencia artificial diseñado para comunicarse entre humanos y computadoras. Esta aplicación utiliza procesamiento de lenguaje natural. Al presionar el micrófono, lo que se dice se registra, comprime y envía al procesador del centro de datos de Apple. Convierta voz en texto emitiendo comandos relacionados con el proceso.

### **1.5.2 Alexa**

Es un servicio de voz ubicado en la nube de la empresa estadounidense Amazon, su nombre se basó en la librería denominada biblioteca de Alejandría, la cual contiene una gran cantidad de conocimiento, este servicio está activado por voz con una palabra clave (Alexa) con el fin de que la aplicación reconozca que es una orden, este tipo de servicio permite controlar dispositivos como luces, cámaras, termostatos entre otros.

### **1.5.3 Asistente de Google**

Es un asistente virtual de inteligencia artificial desarrollado por Google, utiliza voz natural y entrada a través de teclado este permite búsqueda en internet e ingreso a aplicaciones.

### **1.5.4 Alisa**

Es un asistente virtual inteligente ruso para sistemas operativos desarrollado por Yandex, la cual posee sus propios servidores, se basa en un motor de charla basado en redes neuronales entrenadas en conjuntos de datos.

## 1.6 Raspberry pi

“Es una computadora privada de todos los accesorios y se puede quitar sin afectar las operaciones básicas. Consiste en una placa que admite varios componentes necesarios en una computadora común y puede funcionar así. Raspberry Pi se define como un milagro en miniatura, mantiene una potencia informática importante en un volumen pequeño” (RASPBerry PI FOUNDATION UK, 2020).

El último modelo que se encuentra en el mercado es la Raspberry 4B con diferentes capacidades de memoria RAM 1GB, 2GB y 4GB. Este equipo ha ido evolucionando mejorando su potencia y prestaciones, al ser de código abierto toda la información acerca de su diseño se encuentra disponible, es por esto su excelente precio.

### 1.6.1 Pines GPIO

“Son las entradas y salidas de propósito general y algunos de funciones específicas de la tarjeta Raspberry en nuestro caso son 40 pines como se describen a continuación:

- Amarillo (2): Alimentación a 3.3V.
- Rojo (2): Alimentación a 5V.
- Naranja (26): Entradas / salidas de propósito general. Pueden configurarse como entradas o salidas. **El nivel alto es de 3.3V y no son tolerantes a tensiones de 5V.**
- Gris (2): Reservados.
- Negro (8): Conexión a GND o masa.
- Azul (2): Comunicación mediante el protocolo I2C para comunicarse con periféricos que siguen este protocolo.
- Verde (2): Destinados a conexión para UART para puerto serie convencional.
- Morado (5): Comunicación mediante el protocolo SPI para comunicarse con periféricos que siguen este protocolo” (RASPBerry PI FOUNDATION UK, 2020).

“Uno de los puntos que se debe tomar en consideración para su uso es la lógica interna de la Raspberry se basa en 3,3V por lo que es necesario que todos los componentes que se conecten en sus pines GPIO sean compatibles con esa tensión, ya que no disponen de protección por lo que un voltaje mayor puede quemar la placa” (RASPBerry PI FOUNDATION UK, 2020).

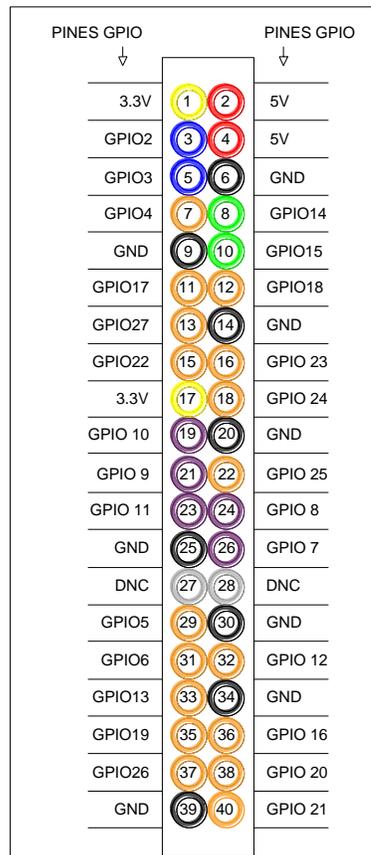


Figura 1. Disposición de pines GPIO. Fuente: (RASPBerry PI FOUNDATION UK, 2020)

## 1.7 Transductores y sensores

“La utilización de los sensores es indispensable en la automatización de procesos e innumerables aplicaciones de nuestra vida cotidiana” (Pallás, 2005, p.2).

### **1.7.1 Transductor**

“Un transductor es cualquier dispositivo que convierte una señal en forma física en una señal correspondiente pero la forma física es diferente, porque hay seis tipos de señales: señales mecánicas, térmicas, magnéticas, eléctricas, ópticas y moleculares, cualquier dispositivo que convierte una señal en una señal. Está bien. Otro tipo de señal se considera un transductor, pero los excelentes transductores son aquellos que proporcionan una señal de salida eléctrica” (Pallás, 2005, p.2).

### **1.7.2 Sensor**

“Un sensor es un dispositivo que proporciona una señal de salida conmutable basada en la energía del medio medido, que es una función de la variable medida. Además, se refiere a un dispositivo que proporciona una respuesta al generar señales eléctricas para estímulos o señales físicas o químicas. Los sensores y transductores a veces se denominan sinónimos, pero los sensores tienen un significado más amplio: los sentidos se expanden para adquirir conocimiento de cantidades físicas y la naturaleza o magnitud de las cantidades físicas no pueden ser percibidas directamente por los sentidos. El transductor, en cambio, la señal de entrada y la señal de salida no deben ser homogéneas” (Pallás, 2005, p.3).

## **1.8 Lenguajes de programación**

### **1.8.1 Python**

“Python es un lenguaje de programación potente y fácil de aprender. Tiene una estructura de datos de alto nivel eficiente y un método de programación orientado a objetos simple pero efectivo. La elegante sintaxis y escritura dinámica de Python y sus características de interpretación lo convierten en un lenguaje ideal para la creación de scripts y el rápido desarrollo de aplicaciones en varios campos y en la mayoría de las plataformas” (Van Rossum, 2009).

“El intérprete de Python y la extensa biblioteca estándar están disponibles de forma gratuita en formato binario y de código fuente en el sitio web de Python <http://www.python.org/> para todas las plataformas principales, y se pueden distribuir de forma gratuita. El mismo sitio

también contiene muchos módulos Python gratuitos de terceros, distribuciones y enlaces a programas y herramientas, y otros documentos” (Van Rossum, 2009).

### 1.8.2 HTML

“HTML es el lenguaje para escribir páginas web. Los usuarios pueden ver páginas web a través de una aplicación llamada navegador. Por lo tanto, podemos decir que HTML es el lenguaje utilizado por los navegadores para mostrar páginas web a los usuarios y es la interfaz más extensa de la web en la actualidad” (UNAM, 2020).

“Este lenguaje nos permite combinar texto, sonido e imágenes y combinarlos según nuestras preferencias. Además, esta es su ventaja frente a libros o revistas: HTML permite introducir referencias a otras páginas a través de enlaces de hipertexto.” (UNAM, 2020).

### 1.8.3 PHP

“PHP es un lenguaje de programación para páginas web del lado del servidor, y su característica principal es la independencia de la plataforma y su uso gratuito” (Gutiérrez, 2020).

“El lenguaje del lado del servidor es un lenguaje que se ejecuta en el servidor web antes de que la página web se envíe al cliente a través de Internet. La página que se ejecuta en el servidor puede realizar el acceso a la base de datos, la conexión a la red y otras tareas para crear la página final que verá el cliente. El cliente solo recibe la página que contiene el código HTML generado al ejecutar PHP” (Gutiérrez, 2020).

## 1.9 Sistemas de alarma

Son un elemento de seguridad pasivo, lo que quiere decir con esto, es que no evitan el suceso o problema en caso de intrusión a la casa o algún incendio, pero si son capaces de dar una alarma, que permita actuar oportunamente.

Los sistemas de alarma están compuestos por diferentes elementos:

**Central procesadora** es la que recibe las señales de los sensores y actúa en respuesta a cualquier estímulo;

**Elementos de detección** son los que detectan el movimiento cuando el sistema está activado y disparan la sirena;

**Sirena exterior** es el elemento audible que permite dar a viso de un evento;

**Teclado** es el elemento que permite armar y desarmar el sistema, y su principal función es permitir el ingreso de personas autorizadas.

### **1.10 Sistemas de control de acceso**

El objetivo de un sistema de control de acceso es permitir el acceso a personas autorizadas hacia una casa, oficina, almacén o edificios. Estos van desde sistemas más básicos como es una cerradura con llave hasta sistemas más sofisticados que permiten gestionar varios accesos.

El sistema consta de un panel de control con controlador inteligente, cierres eléctricos, lectores, pulsadores, servidores de control de acceso. La elección de los diferentes dispositivos depende del área a instalar y la infraestructura de las puertas para determinar el tipo y número de cierres eléctricos además del presupuesto con el que se cuenta.

## CAPÍTULO 2

### MARCO METODOLÓGICO

Este capítulo presenta los aspectos metodológicos utilizados para desarrollar un prototipo de una casa inteligente controlada por comandos de voz, dicho proyecto servirá de apoyo para personas con discapacidad de sus extremidades superiores para la cual se describe métodos y técnicas utilizadas en su desarrollo.

#### **2.1 Tipo de investigación.**

Para el presente proyecto se ha definido que el tipo de investigación es aplicada, la cual que busca aplicar los conocimientos adquiridos, con la creación de un prototipo de control a través de comandos de voz que mejore el desenvolvimiento dentro del hogar, enfocado a personas con discapacidad en sus extremidades superiores.

#### **2.2 Métodos**

Se ha utilizado un método histórico, del cual se recoge información de implementaciones anteriores en sistemas domóticas, se selecciona la información adecuada, verifica el grado de confiabilidad y principios generales, que sirven como pautas para generar nuestro proyecto, así como proporciona la información necesaria para la fundamentación teórica.

Con la información adquirida se procede a utilizar un método de experimentación directa para verificar el funcionamiento y comportamiento de los dispositivos del sistema.

A través de los resultados obtenidos se utiliza un método analítico el cual permite realizar el diseño de un diagrama de flujo de la lógica que debe seguir nuestro prototipo. Este sirve para el desarrollo de los algoritmos que realizarán el reconocimiento de voz y la interacción con dispositivos de control, para poder llegar a los resultados planteados en los cuales se logre integrar varios sistemas que el usuario pueda activarlos a través de comandos de voz.

#### **2.3 Técnicas**

Mediante la observación de sistemas domóticos comerciales se logró comprender el

funcionamiento de los mismos y cómo interactúan con dispositivos de control y actuadores, esto permite seleccionar y clasificar los elementos adecuados para llevar a cabo nuestro proyecto de una manera eficiente.

Por medio de la investigación se pudo recolectar información acerca de la programación y funcionamiento del elemento de control (Raspberry), y mediante experimentación con los elementos seleccionados se logró crear un prototipo acorde a las necesidades requeridas.

Con un protocolo de pruebas se procede a validar que los elementos seleccionados cumplan con la función necesaria y la programación cumpla los objetivos planteados.

## **METODOLOGÍA DE TRABAJO**

La metodología utilizada en la presente investigación consta de cuatro fases:

Fase I. Definir el problema. El problema se da por la dificultad que tienen las personas con discapacidad en sus extremidades superiores para el desenvolvimiento en el hogar, razón por la cual se decide realizar un prototipo que facilite al usuario a realizar acciones como abrir cerraduras, activar un sistema de alarma, encender o apagar focos del hogar a través de comandos de voz.

Fase II. Definir los requerimientos del sistema propuesto. Se recaba información acerca de dispositivos que puedan dar una solución práctica y económicamente factible para utilizar en nuestro prototipo, se definen los requerimientos mínimos y la tecnología a utilizar.

Fase III. Selección de la información. En este punto es esencial instruirnos acerca de las especificaciones técnicas de elementos seleccionados, como acoplarlos a nuestro proyecto para formar un conjunto, aprender la programación de nuestro elemento de control y dispositivos electrónicos, realizar un análisis para verificar que los elementos seleccionados sean los más adecuados para nuestro prototipo propuesto.

Fase IV. Generar diagramas lógicos, diseños electrónicos que permitan fabricar un sistema que pueda ingresar al mercado a competir con sistemas comerciales, se elabora la documentación en la cual incluyen cálculos, tablas y planos especificando sus características técnicas, para luego proceder a la implementación y la puesta en marcha del prototipo.

## CAPÍTULO 3

### 3.1 Propuesta

En este proyecto se ha considerado utilizar una tarjeta Raspberry Pi y además se ha considerado acoplar elementos comerciales de bajo costo que contribuyan un avance tecnológico que facilite las actividades cotidianas, que sea accesible y competitivos con sistemas comerciales, el prototipo consiste en automatizar acciones de nuestra vida cotidiana como encender luces, abrir puertas, desactivar alarma de seguridad a través de comandos de voz. Lo que se busca con este proyecto es ayudar a personas con discapacidad en sus extremidades superiores para que puedan tener independencia al poder interactuar con actuadores a través de comandos de voz.

### 3.2 Diseño general del sistema

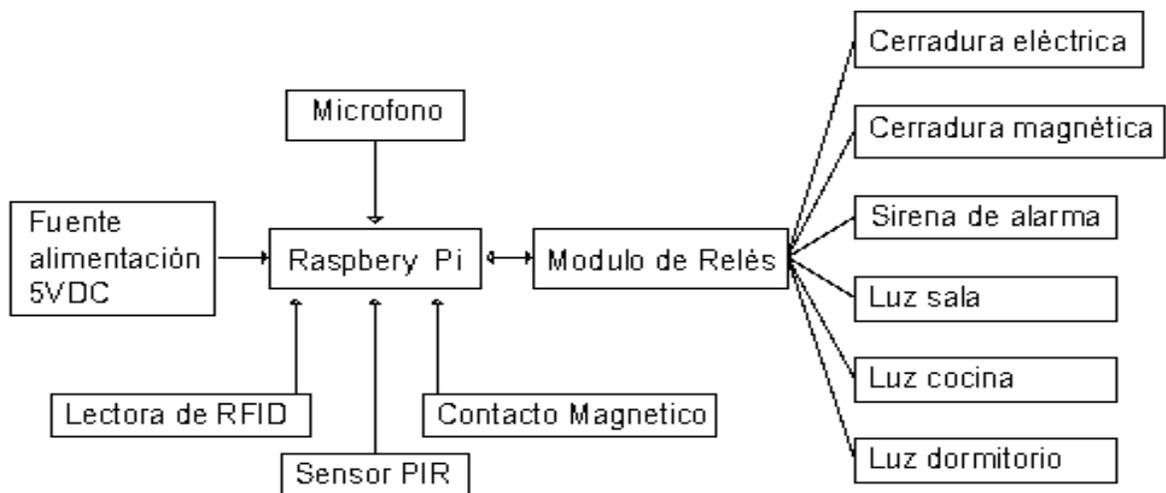


Figura 2: Interacción de la Raspberry con Componentes Electrónicos. Fuente: Elaborado por el autor

El sistema tiene como parte fundamental la Raspberry PI que va ser quien realice la lógica de control, acompañado de una librería de JavaScript para el reconocimiento de voz, a través de instrucciones (comandos), permitirá el accionamiento de los relés que estos a su vez actúan con las luces y cerraduras.

El sistema se basa en un servidor web en este caso Apache, en el cual se modificará el programa raíz de HTML para desarrollo del programa, Apache usa como directorio

/var/www/html como raíz de su sitio, es decir cuando llame la raspberry en el puerto 80 http, apache busca en el archivo en el sitio raíz mencionado.

Para dinamizar el sitio web se usa el servidor PHP para que el usuario envíe la información al servidor y este le devuelva al usuario la información procesada, en este caso el script .php será el encargado de comandos de voz y procesamiento para el funcionamiento de los actuadores.

El ingreso de los comandos de voz a la Raspberry se lo realizará mediante un micrófono inalámbrico, el algoritmo establece una comparación entre los comandos recibidos por el micrófono y los del programa, al existir similitud este procede a realizar la activación o desactivación de los pines GPIO.

El sistema de alarma funcionará como un sistema convencional de robo, es decir permite la detección de un intruso y notifica a través de una sirena sobre este evento. Está conformado por un contacto magnético en la puerta principal, y dos sensores de presencia, los cuales envían la señal para activar una sirena de alarma. La diferencia de este sistema en comparación a los convencionales, es que ya no se dispone de un teclado que permita activar y desactivar la alarma, si no que el sistema se lo activa o desactiva por una clave emitida por un comando de voz.

Sistema de control de iluminación para tres ambientes, para esto se controlará el encendido y apagado de los focos de la sala, cocina y dormitorio con comandos de voz. Para manejar la parte de fuerza se ha dispuesto usar un módulo de relés que funcionan como interruptores, reciben la señal que envía la Raspberry y producen la activación de los mismos, cada relé maneja el foco de un ambiente y este a su vez corresponderá al pin de salida de Raspberry.

En control de iluminación, se describe como una variable cualitativa dicotómica, la cual contempla dos opciones posibles en este caso la señal del controlador cambia de ON a OFF sin estados intermedios (encendido y apagado).

Sistema de control de acceso para esto se utilizara un lector RFID de tarjetas magnéticas ubicado en el ingreso a la casa, la misma que activa o desactiva la cerradura eléctrica y

permite el ingreso al hogar, para la apertura desde la parte interna de la casa se la realiza a través de comandos de voz.

Así mismo se ha provisto de colocar una cerradura electromagnética para una puerta interna, en este caso la apertura se la realiza por comandos de voz. Para el sistema de acceso se proveerá una fuente con respaldo de batería, para el caso de fallo de la energía eléctrica.

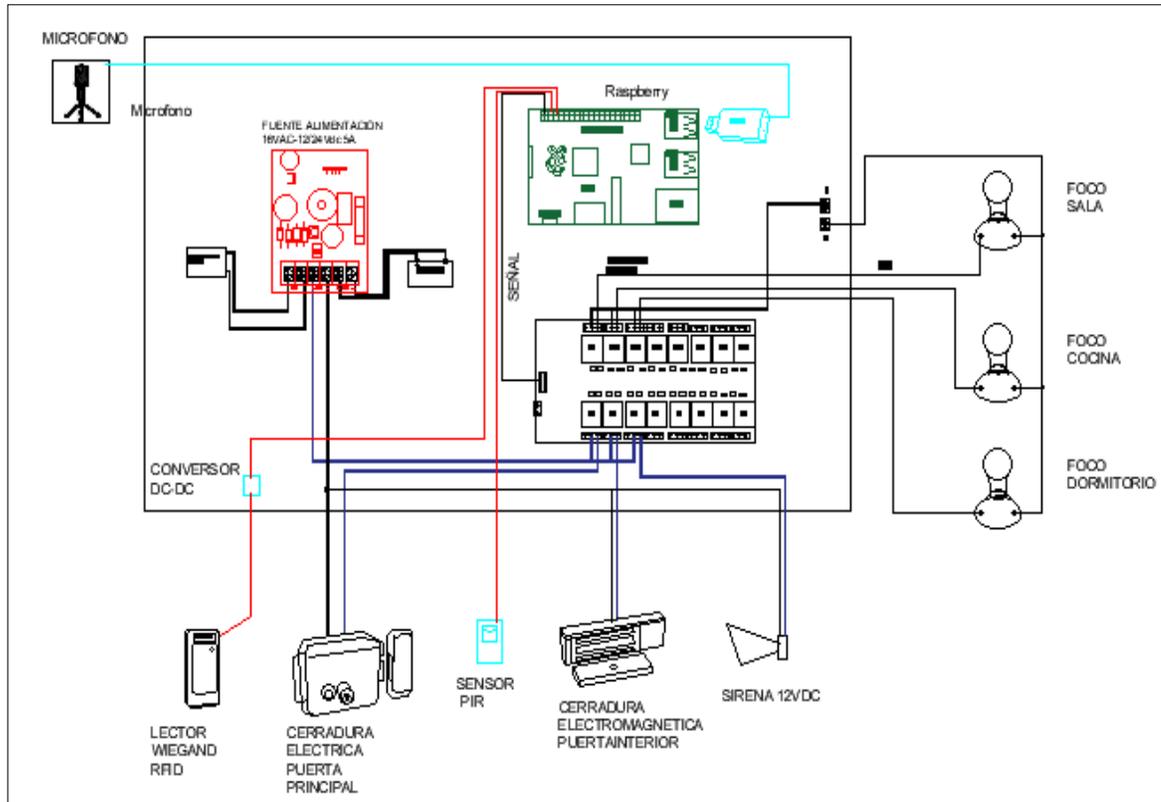


Figura 3: Esquema general de proyecto. Fuente: Elaborado por el autor

### 3.2.1 Diseño del sistema de alarma

El objetivo del sistema de alarma es proporcionar una señal con el fin de alertar en caso de robo y además pueda dar una señal de alarma en caso de una emergencia. Para el sistema de alarma se ha considerado colocar contactos magnéticos en las puertas, este será la primera zona de detección, la segunda zona está compuesta por dos sensores de movimiento PIR con los cuales se puede censar la presencia de personas no autorizadas en nuestro hogar, una sirena la cual proporciona una señal audible, para activar o desactivar la alarma, se lo realiza a través de comandos de voz con un código de ingreso, como se lo realiza con un teclado.

Para la conexión de los contactos magnéticos es necesario utilizar un par del cable UTP para enviar la señal, estos no necesitan alimentación. Esta señal se controla a través del pin GPIO 21.

Para la conexión de la sirena es necesario utilizar un relé de la placa de interfaz de relés, en este caso la señal para el encendido de la sirena es a través de pin GPIO 14. La conexión se la realizará según el siguiente esquema.

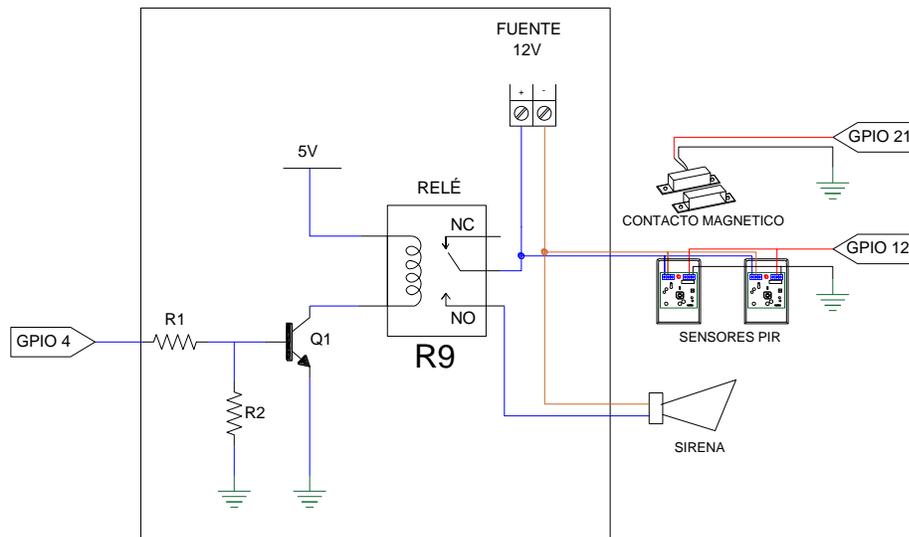


Figura 4: Esquema conexión alarma. Fuente: Elaborado por el autor

Para la conexión de los sensores PIR es necesario instalar dos cables para alimentación y dos para señal, los dos sensores PIR deberán conectarse en paralelo y la señal estará controlado a través del pin GPIO 12 de la Raspberry.

### Conexiones de los terminales

12VDC: Entrada de energía

ALARMA: Conmutador contacto NC

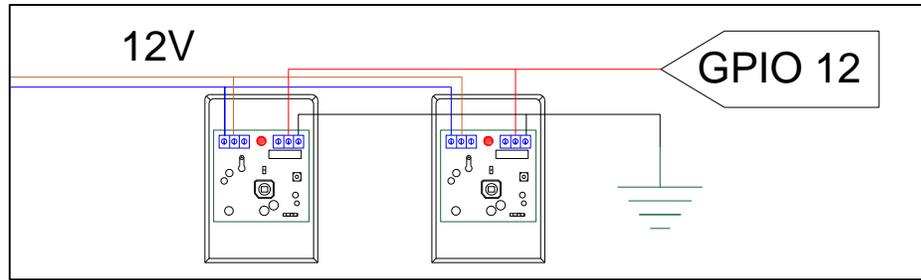


Figura 5: Esquema de conexión sensores PIR. Fuente: Elaborado por el autor

### 3.2.2 Diseño del sistema de control de acceso

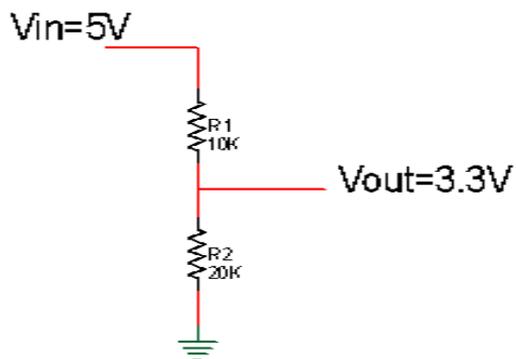
Para el sistema de control de acceso se ha previsto instalar una chapa eléctrica y una lectora de proximidad para apertura por la parte de afuera, la apertura por la parte de adentro será a través de comando de voz, mientras que para la puerta interna se ha previsto instalar una cerradura electromagnética, su apertura igualmente será por comandos de voz.

#### Conexión de la lectora

La lectora funciona con un voltaje de alimentación de 5-16VDC, además con dos cables para envío de la señal, para 0L y 1L, el voltaje que envían a la lectora es 5v por lo que es necesario antes del ingreso a los pines GPIO se ponga un regulador de tensión a 3.3V lo cual lo hemos realizado a través un divisor de tensión con 2 resistencias.

#### Divisor de tensión

$$V_{out} = V_{in} \frac{R_2}{R_1 + R_2}$$



Si tenemos la tensión de entrada es de 5V y las resistencias de 10K y 20K obtenemos un voltaje de 3.3V a la salida

$$V_{out} = 5V \frac{20K}{10K + 20K} = 3.3V$$

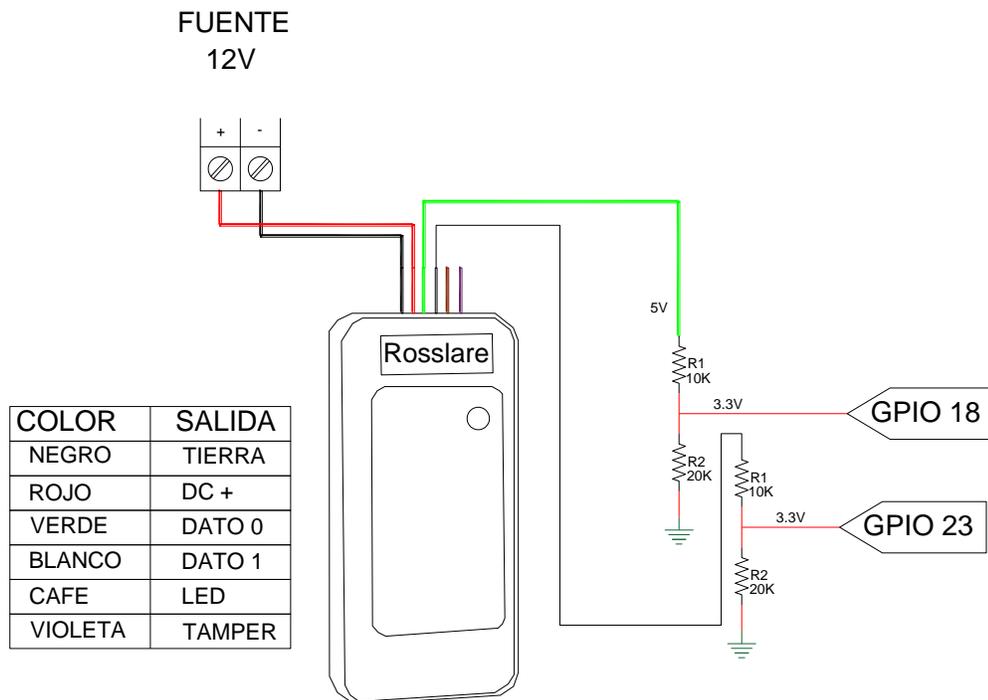


Figura 6: Esquema de conexión Lectora. Fuente: (Rosslare, 2019)

### Conexión de la cerradura eléctrica

En la puerta de entrada principal se ha previsto instalar cerradura eléctrica de 12V para el funcionamiento de este elemento es necesario 2 cables para alimentación, la misma que será activada a través de un relé. Para el disparo de los relés es necesario realizarlo a través de un transistor ya que el voltaje de salida de los pines GPIO es de 3.3V mientras que el del disparo de los relés es de 5V para lo cual se va a hacer funcionar al transistor como interruptor.

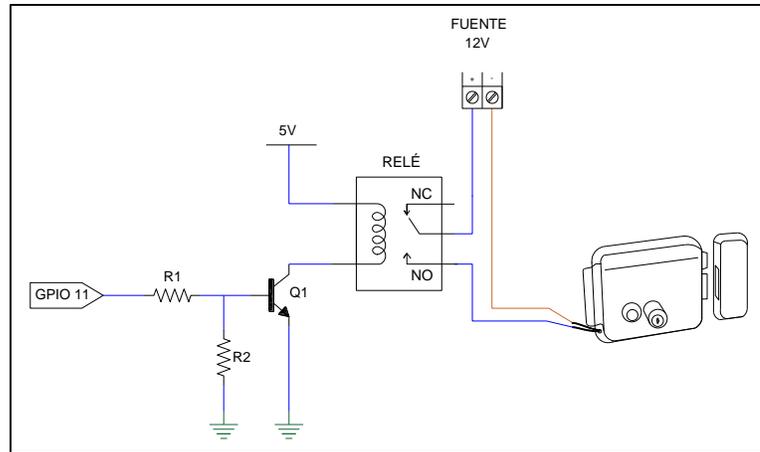


Figura 7: Esquema de conexión cerradura. Fuente: Elaborado por el autor

### Circuito de disparo de relé a través de pin GPIO utilizando un transistor NPN 2N3904

Para convertir el nivel de voltaje de la salida digital de la Raspberry Pi en nivel digital para los relés usaremos un inversor lógico convertidor de nivel utilizando un transistor.

Cuando el voltaje de entrada este en 0V el transistor estará en modo de corte (circuito abierto entre Emisor-Colector), mientras que cuando este en 3.3V el transistor trabajara en modo saturación (circuito cerrado entre Emisor-Colector), como se observa en la figura 8.

Se utilizará un transistor 2N3904, este tiene un voltaje colector-emisor máximo de 40V, considerando una ganancia del transistor de  $hFE=50$ , la corriente necesaria para activar el relé es de 50mA.

$$I_B = \frac{I_c}{hFE} = \frac{50mA}{50} = 1mA$$

$$V_{in} + I_B R_B - V_{BE} = 0 \quad \rightarrow \quad R_B = \frac{V_{in} - V_{BE}}{I_B} = \frac{3 - 0.7}{1mA} = 2.3K\Omega$$

Valor comercial de la resistencia  $R_1=R_B=2.2K\Omega$ , la resistencia R2 no es imprescindible pero es buena costumbre colocar una resistencia entre la base y tierra, esta sirve para que el transistor no se encienda de modo errático, cuando la Raspberry o cualquier microcontrolador está en fase de iniciación y sus salidas no se encuentran mapeadas, su valor no es crítico pero

podemos usar un valor grande en este caso  $R2=47K\Omega$  para no alterar el cálculo de la resistencia base.

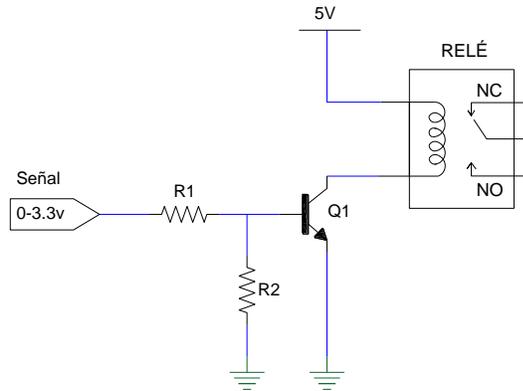


Figura 8: Circuito para disparo de relé. Fuente: Elaborado por el autor

### Conexión de la cerradura electromagnética

En la puerta interior se ha propuesto instalar una cerradura electromagnética de 270kg de fuerza, esta trabaja con un voltaje de 12VDC, para lo cual es necesario llevar la alimentación desde nuestro tablero de control hasta la cerradura, el control de la misma se la realizara a través de una señal que dispara un relé tal como se muestra abajo.

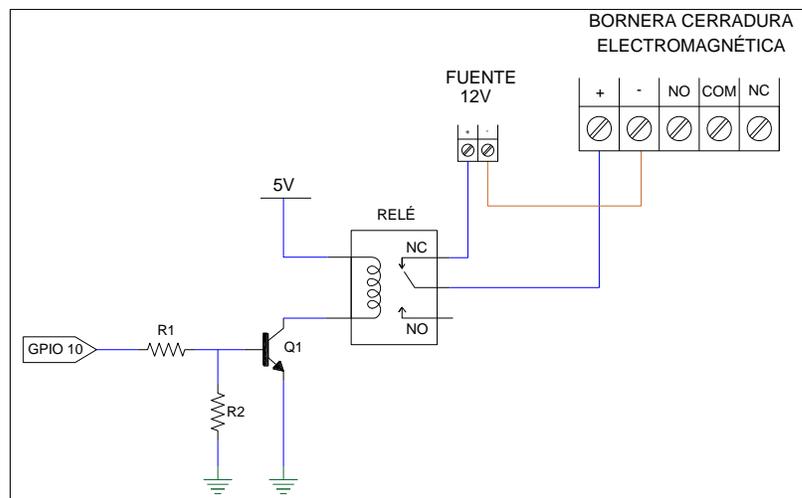


Figura 9: Conexión de cerradura electromagnética. Fuente: (SECO-LARM, 2020)



La conexión va de la siguiente manera la fase llega al terminal común de los relés y desde la salida del contacto normalmente abierto (NA) se lleva los retornos hacia los focos, y del otro extremo de los focos van hacia el neutro.

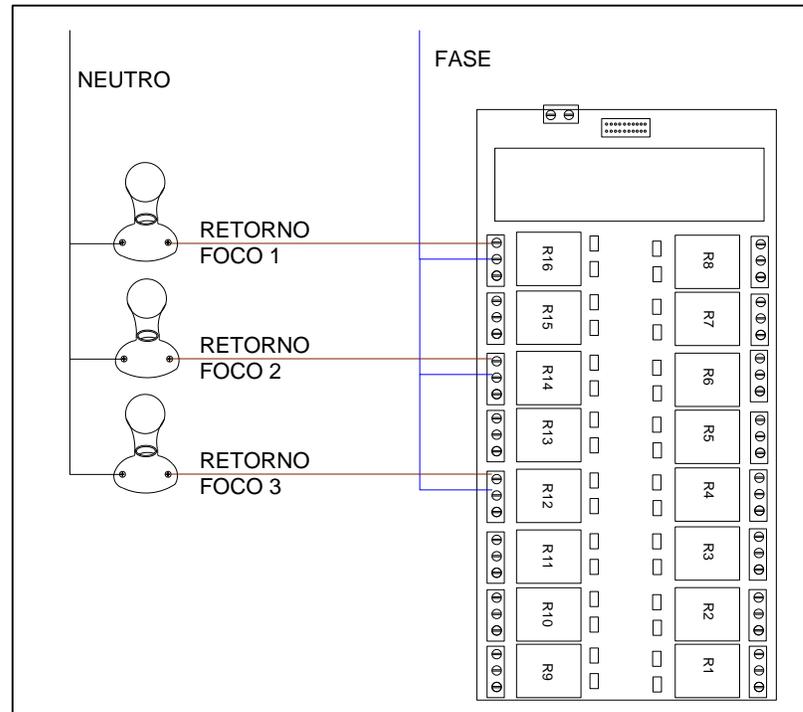


Figura 11: Esquema de conexión focos. Fuente: Elaborado por el autor

### 3.3 Elementos del sistema

#### 3.3.1 Raspberry pi

“La tarjeta utilizada en este proyecto es una Raspberry pi 4B con 4GB de RAM, es un software de código abierto y su sistema operativo oficial es una versión modificada de Debian llamada Raspbian, aunque permite otros sistemas operativos, incluidas las versiones de Windows. 10. Esta versión incluye procesador Broadcom BCM2711, SoC Cortex-A72 de cuatro núcleos (ARM v8) de 64 bits a 1,5 GHz, GPU, puerto USB, HDMI, conexión Gigabit Ethernet, 40 pines GPIO y un conector de cámara” (RASPBERRY PI FOUNDATION UK, 2020).

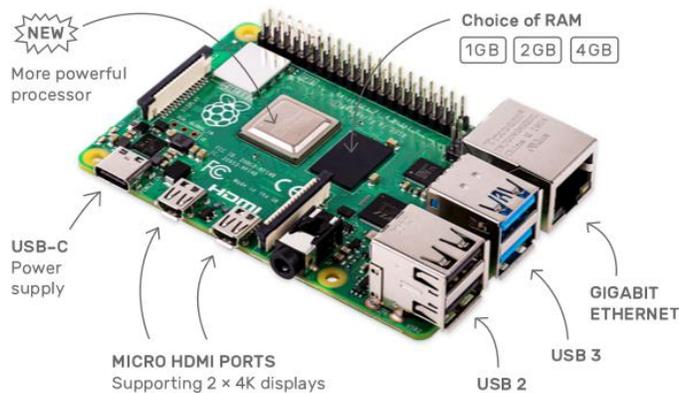


Figura 12: Tarjeta Raspberry Pi 4B Fuente: (RASPBERRY PI FOUNDATION UK, 2020).  
Obtenido de: <https://projects.raspberrypi.org/en>

### “Características Técnicas

- Broadcom BCM2711, Quad core Cortex-A72 (ARM v8) SoC de 64 bits a 1.5GHz
- SDRAM LPDDR4-3200 de 1 GB, 2 GB o 4 GB (según el modelo)
- 2.4 GHz y 5.0 GHz IEEE 802.11ac inalámbrico, Bluetooth 5.0, BLE
- Gigabit Ethernet
- 2 puertos USB 3.0; 2 puertos USB 2.0.
- Cabezal GPIO estándar de 40 pines de Raspberry Pi (totalmente compatible con placas anteriores)
- 2 puertos micro HDMI (hasta 4kp60 compatibles)
- Puerto de pantalla MIPI DSI de 2 carriles
- Puerto de cámara MIPI CSI de 2 carriles
- Puerto de audio compuesto y video compuesto de 4 polos
- H.265 (decodificación 4kp60), H264 (decodificación 1080p60, codificación 1080p30)
- Gráficos OpenGL ES 3.0

- Ranura para tarjeta micro SD para cargar el sistema operativo y el almacenamiento de datos
- 5 V CC a través del conector USB-C (mínimo 3 A \*)
- 5V DC a través del encabezado GPIO (mínimo 3A \*)
- Alimentación a través de Ethernet (Poe) habilitada (requiere un Poe HAT separado)
- Temperatura ambiente de funcionamiento: 0 - 50 grados C” (RASPBerry PI FOUNDATION UK, 2020)

### 3.3.2 Detector de movimiento

Para sensar la presencia utilizaremos un detector de movimiento pasivo infrarrojo. Este tipo de detectores reaccionan frente a fuentes de energía corporales, puede ser el calor del cuerpo humano, reciben la variación de las radiaciones infrarrojas del medio ambiente que la cubre. Este detector es llamado pasivo, debido a que no emite radiaciones, sino que las recibe es decir realiza una comparación entre la diferencia del calor emitido por el cuerpo humano y el espacio alrededor del mismo. Su componente primordial son los sensores piroeléctricos que es el encargado de detectar cambios en la radiación infrarroja.



Figura 13: Detector de movimiento PIR. Fuente: (Risco Group, 2020). Obtenido de: <https://www.riscogroup.com/spain/products/product/5474>

### 3.3.3 Lectora de proximidad

La lectora de tarjetas de proximidad AY-K12 y etiquetas de RFID cuenta con un formato de salida Wiegand 26 bits, modulación ASK (modulación de amplitud) a 125KHz, trabaja desde una distancia de 80mm, este tipo de lector posee una notificación audible para la lectura de la tarjeta, el control LED ofrece información al usuario final del estado de la lectora. Tiene un amplio intervalo de voltaje de funcionamiento de 5 a 16 VCC.



Figura 14: Lector de proximidad wiegand. Fuente: (Rosslare, 2019). Obtenido de: [http://jhbdiseñoingenieria.com/files/rosslare/Datasheet\\_AY-X12.pdf](http://jhbdiseñoingenieria.com/files/rosslare/Datasheet_AY-X12.pdf)

### 3.3.4 Contactos magnéticos

Estos funcionan como un interruptor en la que el imán es el que cierra o abre el circuito dependiendo de si están cerca uno del otro. Estos sensores son la primera alerta en un sistema de alarma contra robo son utilizados para protección de ventanas y puertas, cuando la puerta se abra el contacto será quien informe de un evento y active un dispositivo de alarma, para la alimentación de estos solo es necesario un par del cable UTP.

### 3.3.5 Placa de interfaz de relé 16 canales

Son un dispositivo eléctrico que se puede utilizar como interruptor para abrir y cerrar canales de corriente, pero son eléctricos, lo que permite que los electroimanes abran o cierren contactos. Para nuestro proyecto utilizaremos un módulo de interfaz 16 relés para Raspberry, estos se utilizan para asegurar el aislamiento entre el circuito de mando y el circuito de potencia externo porque están protegidos por optoacopladores. (Amazon, 2020)



Figura 15: Módulo de Relés 16CH. Fuente: (Amazon, 2020). Obtenido de: <https://www.amazon.com/-/es/SainSmart-M%C3%B3dulo-rel%C3%A9-16-canales/dp/B0057OC66U>

### 3.3.6 Fuente de Alimentación

Es el dispositivo electrónico con la capacidad de convertir el voltaje AC en voltaje DC regulado con el fin de proporcionar energía a nuestro sistema, están equipadas con elementos de protección contra sobrecarga y protección térmica, estas están disponibles según la capacidad de energía necesaria para cada sistema. En nuestro caso estamos utilizando una fuente regulada a 12v con una corriente de 5A, la misma que posee conexión para una fuente de respaldo (batería), el cálculo de la fuente de alimentación se muestra en la Tabla 1

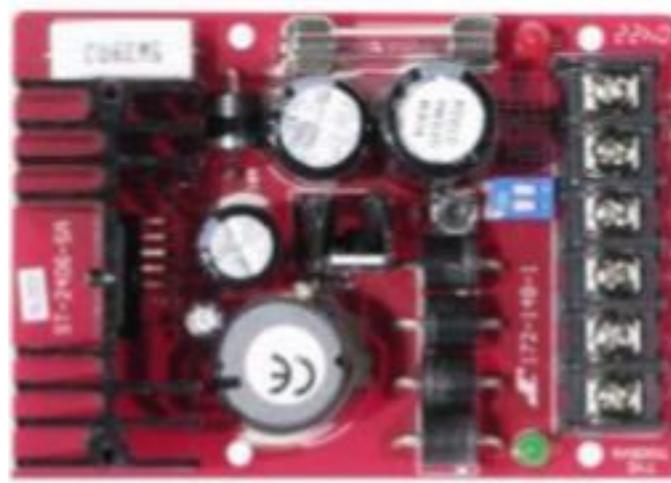


Figura 16: Fuente regulada a 12VDC. Fuente: (SECO-LARM, 2020). Obtenido de: <http://www.seco-larm.com/es/Security/security-power-supplies/ST-1206-1.5AQ>

**Tabla 1:**  
Cálculo de la fuente de alimentación

DESCRIPCION	CANT	VOLTAJE	CORRIENTE	POTENCIA UNITARIA	POTENCIA TOTAL
		(V)	(mA)	(W)	(W)
PLACA DE INTERFAZ DE 16 RELES	1	12	160	1,92	1,92
CERRADURA ELECTRICA	1	12	60	0,72	0,72
CERRADURA ELECTROMAGNETICA	1	12	500	6	6
SENSOR DE MOVIMIENTO PIR	2	12	12	0,144	0,288
LECTORA DE PROXIMIDAD	1	12	50	0,6	0,6
RASPBERRY PI	1	5	2500	12,5	12,5
SIRENA	1	12	2500	30	30
				TOTAL	52,03
			POTENCIA TOTAL=	52,03	Watios
			CORRIENTE=	4,34	Amperios
			FUENTE NORMALIZADA=	5	Amperios
			VOLTAJE DE FUNCIONAMIENTO=	12	Voltios

Nota: Tabla adaptada por el autor

### Batería de Respaldo

La batería de respaldo a utilizar será de 12V 7Ah y servirá para el sistema de alarma en caso de fallo de energía eléctrica, cuando la alarma se encuentre activada los sensores de movimiento y contacto magnético siguen funcionando al detectar algún intruso disparan la sirena.

Además, la batería permite que el usuario pueda ingresar a la casa cuando no haya energía eléctrica ya que el lector de proximidad y las cerraduras que encuentran activas y funcionan con normalidad independientes de los comandos de voz.

Para calcular la capacidad de la batería se ha considerado 90 minutos de respaldo y se la corriente de consumo de nuestro sistema de 4.34A la capacidad de la batería es:

Capacidad = 4.34 Amperios x 1.5 horas

Capacidad = 6.51Ah

Para lo cual se ha elegido una batería normalizada de 7Ah

### 3.3.7 Micrófono inalámbrico

Un micrófono es un traductor electroacústico, que puede convertir la energía acústica en energía eléctrica. En caso de los micrófonos inalámbricos el micrófono está compuesto por dos elementos, un transmisor integrado con un micrófono y un receptor con una o más antenas, este es el que procesa la señal enviada desde el micrófono.

Existen dos rangos de frecuencias regulados por la Comisión Federal de Comunicaciones los VHF y los UHF para el uso de micrófonos inalámbricos,

- VHF es un rango de frecuencia de radio que oscila entre 30 y 300Mhz que se usa para televisión y radio FM
- UHF es un rango de frecuencia que oscila entre 300Mhz y a 3Ghz y se utiliza para servicios de televisión y telefonía móvil.

**Tabla 2:**  
Rango de Frecuencias Espectro Radioeléctrico

Número de la banda	Símbolos (en inglés)	Gama de frecuencias (excluido el límite inferior, pero incluido el superior)	Subdivisión métrica correspondiente	Abreviaturas métricas para las bandas
3	ULF	300-3 000 Hz	Ondas hectokilométricas	B.hkm
4	VLF	3-30 kHz	Ondas miriamétricas	B.Mam
5	LF	30-300 kHz	Ondas kilométricas	B.km
6	MF	300-3 000 kHz	Ondas hectométricas	B.hm
7	HF	3-30 MHz	Ondas decamétricas	B.dam
8	VHF	30-300 MHz	Ondas métricas	B.m
9	UHF	300-3 000 MHz	Ondas decimétricas	B.dm
10	SHF	3-30 GHz	Ondas centimétricas	B.cm
11	EHF	30-300 GHz	Ondas milimétricas	B.mm
12		300-3 000 GHz	Ondas decimilimétricas	B.dmm
13		3-30 THz	Ondas centimilimétricas	B.cmm
14		30-300 THz	Ondas micrométricas	B.µm
15		300-3 000 THz	Ondas decimicrométricas	B.dµm

Nota: Tabla adaptada por (ITU-R, 2015)

El micrófono utilizado será uno que trabaje en la banda de frecuencia VHF se ha elegido en esta banda de frecuencia debido a que este contiene menos tráfico de ondas de radio,

también su rango de alcance es mayor y su costo es menor en comparación a un micrófono UHF. El receptor del micrófono se instalará en el tablero de control y el transmisor servirá para que el usuario mencione los comandos para el funcionamiento de los diferentes dispositivos.



Figura 17: Micrófono inalámbrico VHF. Fuente: (EB AUDIO, 2020)

Para conectar un micrófono a la Raspberry es necesario conectar adaptador de audio externo de USB a 3.5 ya que la tarjeta no dispone de entrada de audio, esta permite la conexión del micrófono y de un parlante para salida de audio. No necesita instalar ningún driver y trabaja con sistemas operativos Linux y Windows y no necesita alimentación externa.



Figura 18: Adaptador de Audio USB. Fuente: (Amazon, 2020)

### 3.3 Diagrama lógico del programa

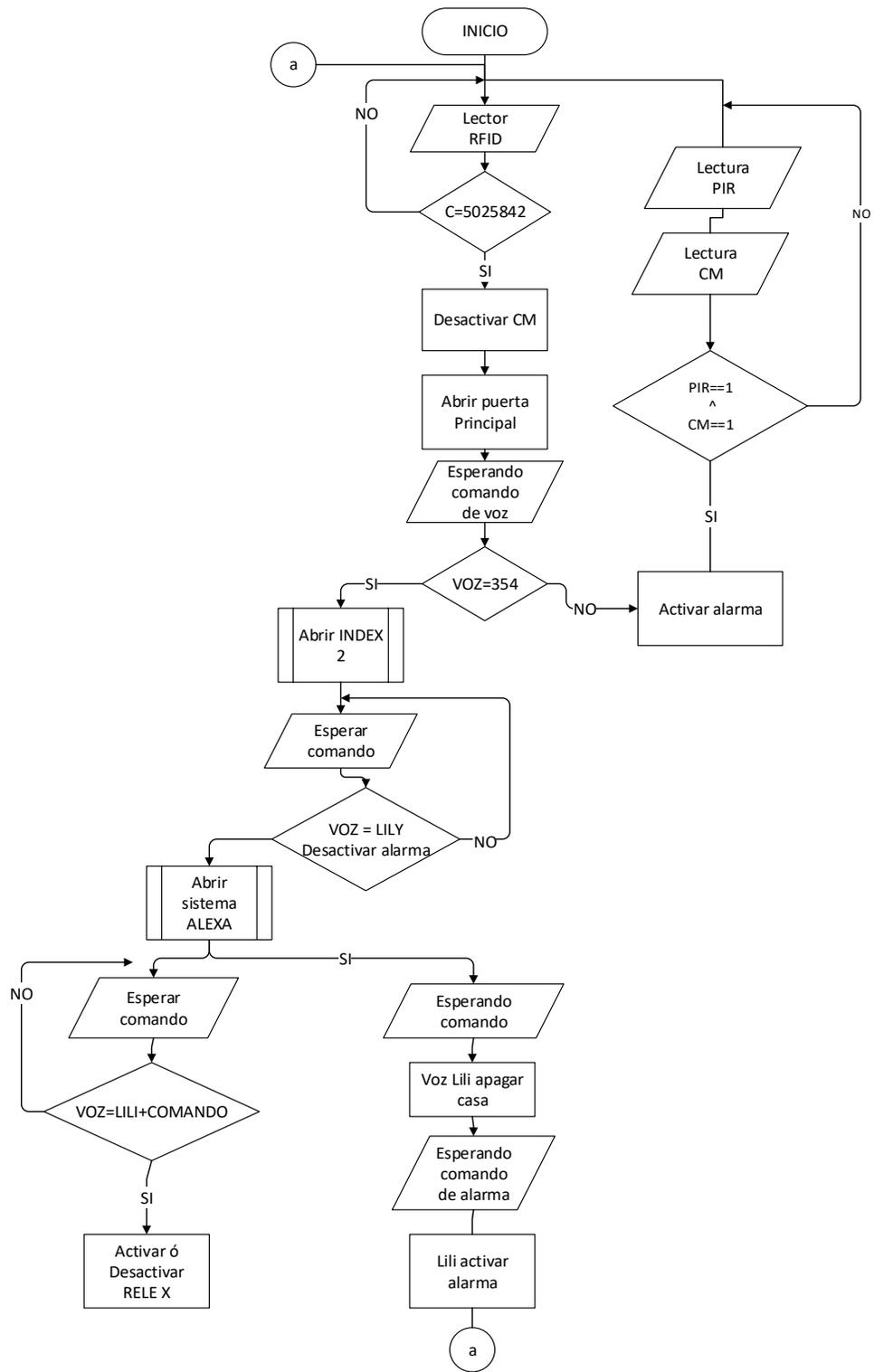


Figura 19: Diagrama de funcionamiento del algoritmo. Fuente: Elaborado por el autor

En la figura 3.14 se puede observar, el proceso para activar la alarma, en la cual se debe tomar los datos del estado del contacto magnético de la puerta principal y de los sensores de movimiento PIR, al igual que para el acceso se tiene como dato la lectura de una tarjeta RFID.

### 3.4 Configuración y puesta en marcha de la Raspberry

Para nuestro proyecto el elemento principal es la Raspberry, para ponerla en funcionamiento es necesario instalar el sistema operativo oficial, en este caso es Raspbian para lo cual es necesario descargar la imagen de la página oficial y copiarla en una SD de por lo menos 16GB. En la misma que se desarrolla un algoritmo que permita el reconocimiento y procesamiento de comandos de voz.

El reconocimiento de voz es un método que permite interactuar con sensores y estos a su vez con máquinas o actuadores a través de comandos de voz y se pueden ejecutar acciones que se las realizan habitualmente a través de botones.

Para la presente tesis se utilizará el reconocimiento de voz a través de una librería ejecutándose en un servidor web, llamada la librería annyang, la misma que permite realizar el reconocimiento y procesamiento de los comandos de voz, además se ha utilizado un servidor web local, en nuestro caso se utilizó Apache por ser un servidor de código abierto fácil de usar y configurar, además es inmune a amenazas porque se actualiza constantemente.

#### 3.5.1 Librería Annyang

“Es una pequeña biblioteca de JavaScript que permite a los usuarios controlar la aplicación web con comandos de voz, la ventaja de esta librería es que admite varios idiomas y no tiene dependencias, tiene un peso de 2kb y es una librería de uso gratuito” (Annyang, 2019).

Para empezar a utilizar esta librería procedemos a importarla desde su página oficial y esta es compatible con todos los navegadores.

```
<script src="//cdnjs.cloudflare.com/ajax/libs/annyang/2.6.0/annyang.min.js"></script>
```

“La licencia para esta biblioteca es la MIT (Massachusetts Institute of Technology) lo que significa que es de software libre la cual permite copiar, modificar, publicar, sublicenciar e integrar con otro software es decir sin restricciones” (Annyang, 2019).

### **3.5.2 Librería Pico2wave**

“Para tener una mejor interacción con el sistema se ha previsto instalar una herramienta en línea denominada pico2wave, la cual es compatible con sistemas linux y permite traducir el texto en un archivo .wav, con esto se puede tener una interacción más dinámica con el sistema al conectarlo a un parlante” (GitHub, 2020).

Con la ayuda de un pequeño parlante se podrá escuchar a nuestro sistema que responde cada vez que ejecuta alguna acción, esto es gracias a que nuestro algoritmo incluirá mensajes, los mismos que a través de esta librería se convierten en audio.

### **3.5.3 Servidor web Apache**

“Es un software de servidor web gratuito y de código abierto, y el 65% de los sitios web del mundo utilizan este software. El nombre oficial es Apache HTTP Server, mantenido y desarrollado por "Apache Software Foundation", que procesa las solicitudes a través de HTTP” (APACHE, 2019).

“Es uno de los servidores más antiguos y confiables, permite a los propietarios de sitios web servir contenido en la web, cuando alguien quiere visitar un sitio ingresa un nombre de dominio en la barra de direcciones de su navegador. Luego el servidor web envía los archivos solicitados actuando como un repartidor virtual” (Hostinger, 2019).

Para nuestro caso Apache se comunica a través del protocolo HTTPS es decir que se transmite las solicitudes y comandos a través del puerto 443. Es compatible con sistemas operativos como Linux, Windows, MAC entre otros.

### **3.5.4 Funcionamiento de Apache**

“Llamamos a Apache como servidor web, pero no es un servidor físico, sino un software que se ejecuta en el servidor. Su trabajo consiste en establecer una conexión entre el servidor

y el navegador del visitante del sitio web (Firefox, Google Chrome, Safari, etc.), mientras se envían archivos entre ellos (estructura cliente-servidor). Apache es un software multiplataforma, por lo que puede ejecutarse en servidores Unix y Windows” (Hostinger, 2019).

“Cuando un usuario desea cargar la página de su sitio web (como la página de inicio o la página, su navegador enviará una solicitud a su servidor y Apache devolverá el archivo que contiene todos los archivos solicitados (texto, imágenes, etc.). El servidor y el cliente se comunican a través del protocolo HTTP, y Apache es responsable de garantizar una comunicación fluida y segura entre las dos computadoras” (Hostinger, 2019).

“Apache tiene una estructura basada en módulos, por lo que es altamente personalizable. El módulo permite a los administradores del servidor habilitar y deshabilitar otras funciones. Apache tiene módulos de seguridad, almacenamiento en caché, reescritura de URL, verificación de contraseña, etc.” (Hostinger, 2019).

#### **“Ventajas de usar Apache.**

- Código fuente libre y abierto, incluso con fines comerciales.
- Software confiable y estable.
- Parches de seguridad que se actualizan periódicamente y con frecuencia.
- Estructura flexible basada en módulos.
- Fácil de configurar.
- Multiplataforma
- Si tiene alguna pregunta, gran comunidad y soporte disponible” (Hostinger, 2019).

### **3.6 Presupuesto de la implementación.**

Se han considerado todos los equipos tanto para el control como para el cableado de fuerza y se recalca que el software a utilizar es gratuito por lo que no se refleja este valor en los rubros como muestra en la Tabla 3.

Una vez definido los elementos y equipos a usar en nuestro sistema, se estima un presupuesto referencial de 700 USD.

**Tabla 3:**  
Equipos de implementación del sistema

ITEM	CANT	DESCRIPCION	P. Unitario	P. Total
1	1	Tablero metálico 40x40x20cm	63,53	63,53
2	1	Fuente de alimentación 120VAC/12VDC-5A Marca: Enforcer, Modelo: ST-2406-5AQ	45,00	45,00
3	1	Batería 12V-7Ah, Marca: Casil, Modelo: CA1270	18,00	18,00
4	1	Adaptador 120VAC-16,5VAC Modelo Vipertek	20,00	20,00
5	1	Raspberry Pi 4b	75,00	75,00
6	1	Adaptador Raspberry 5V-3A	8,50	8,50
7	1	Tarjeta micro SD 32GB clase 10	14,00	14,00
8	1	Tarjeta de interfaz de relés 16 CH equipados con relés de alta corriente capaz de soportar 10A-250VAC o 10A-30VDC	30,00	30,00
9	2	Sensor de movimiento PIR, Marca Risco, Modelo: RK210PR	20,50	41,00
10	1	Lectora de proximidad Marca Rossalare, Modelo: AY-K12	55,00	55,00
11	2	Tarjeta PVC 26 Bits	2,50	5,00
12	1	Micrófono inalámbrico, Marca: EB AUDIO	42,00	42,00
13	1	Tarjeta de sonido externa USB	5,00	5,00

ITEM	CANT	DESCRIPCION	P. Unitario	P. Total
14	1	Sirena 30W-12VDC Marca: Vipertek, Modelo:VP-S30	22,00	22,00
15	1	Cerradura electromagnética 270kg Marca: ZKTeco Modelo: AL-280(LED)	27,00	27,00
16	1	Cerradura eléctrica 12v Marca: Garen, Modelo:GA042	45,00	45,00
17	1	Contacto magnético	1,50	1,50
18	1	Breaker 10A-1P sobrepuesto Marca: Camsco	3,50	3,50
19	1	Tomacorriente doble 15A/127V	3,00	3,00
20	50	Cable flexible Nº14 THHN AWG	0,40	20,00
21	50	Cable UTP Cat 5E Marca: Hikvision	0,25	12,50
22	6	Canaleta plástica 20x10x2000mm	1,25	7,50
23	2	Caja Dexson 2x4 sobrepuesto	1,60	3,20
24	1	Accesorios de Instalación	20,00	20,00

Nota: Tabla adaptada por el auto

# CAPÍTULO 4

## IMPLEMENTACIÓN

### 4.1 DESARROLLO

En este capítulo se procede a realizar una descripción de la implementación de nuestro prototipo. Primero de acuerdo a nuestro plano de diseño podemos observar la ubicación de los puntos eléctricos de iluminación, sensores y cerraduras que permitirán realizar el montaje de cada uno de estos dispositivos.

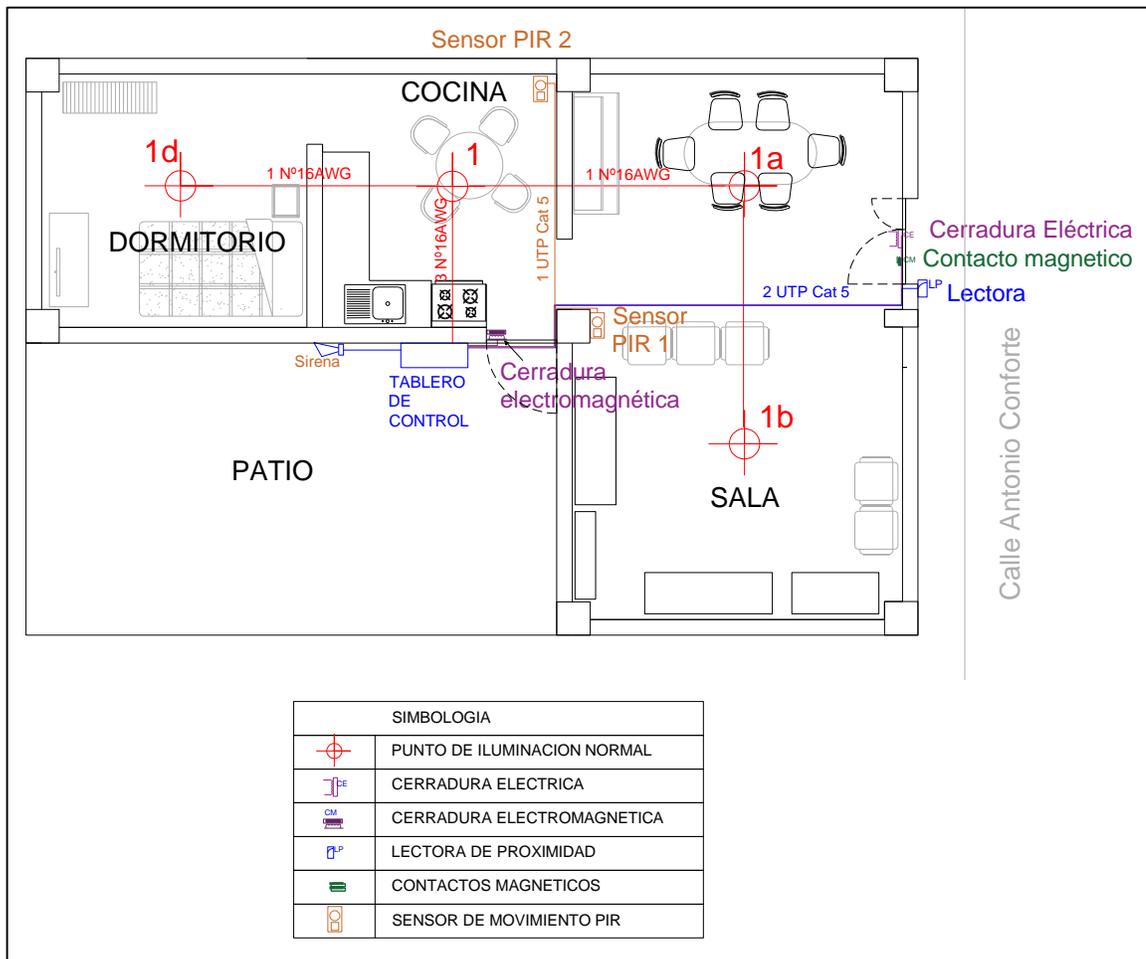


Figura 20: Plano ubicación de elementos. Fuente: Elaborado por el autor

## **4.2 Cableado de sensores y cerraduras**

Para el cableado primero se procede a instalar canaletas plásticas de 25x10mm para los cables UTP que alimentaran a los diferentes dispositivos en las trayectorias que indica el plano, es necesario etiquetar en las puntas de los cables para no confundir al momento de realizar las conexiones en el tablero de control.

Para la puerta principal instalamos 2 cables UTP, uno para la lectora de proximidad y otro para la cerradura eléctrica y el contacto magnético.

Para el lector PIR instalaremos un cable UTP desde el tablero hasta el primer sensor de movimiento y desde este otro cable UTP hasta el otro sensor PIR para realizar su conexión en paralelo.

## **4.3 Cableado para control de luces**

Luego se procede a cablear desde cada foco un cable N° 16 AWG hasta el tablero de control, que va a ser el retorno de cada foco y se lo conectará a los relés. Cabe indicar que se está instalando en una casa que ya se encontraba funcionando por lo que ya existe conexión de fase y neutro a los focos, lo único que se está haciendo es una modificación en la que se eliminan los interruptores y el control se lo realiza a través del módulo de relés que se encuentra en el tablero de control.

Una vez realizado el cableado de todos los dispositivos, se procede a instalar el tablero de control, este se ha previsto instalar junto a la caja de control eléctrica que está ubicada a la salida de la cocina.

## **4.4 Instalación y armado del tablero de control**

Está compuesto de los siguientes elementos:

- Adaptador AC/AC 120V-16.5VAC
- Fuente de alimentación 16VAC-12VDC 5A
- Batería Seca 12V-7AH

- Raspberry PI
- Módulo de 16 relés para cargas DC/AC cada uno de estos soporta una corriente de 10A
- Placa de protección de pines GPIO

El montaje de estos elementos se lo realiza de acuerdo al siguiente esquema de conexión el mismo que se instala en la placa doble fondo del tablero.

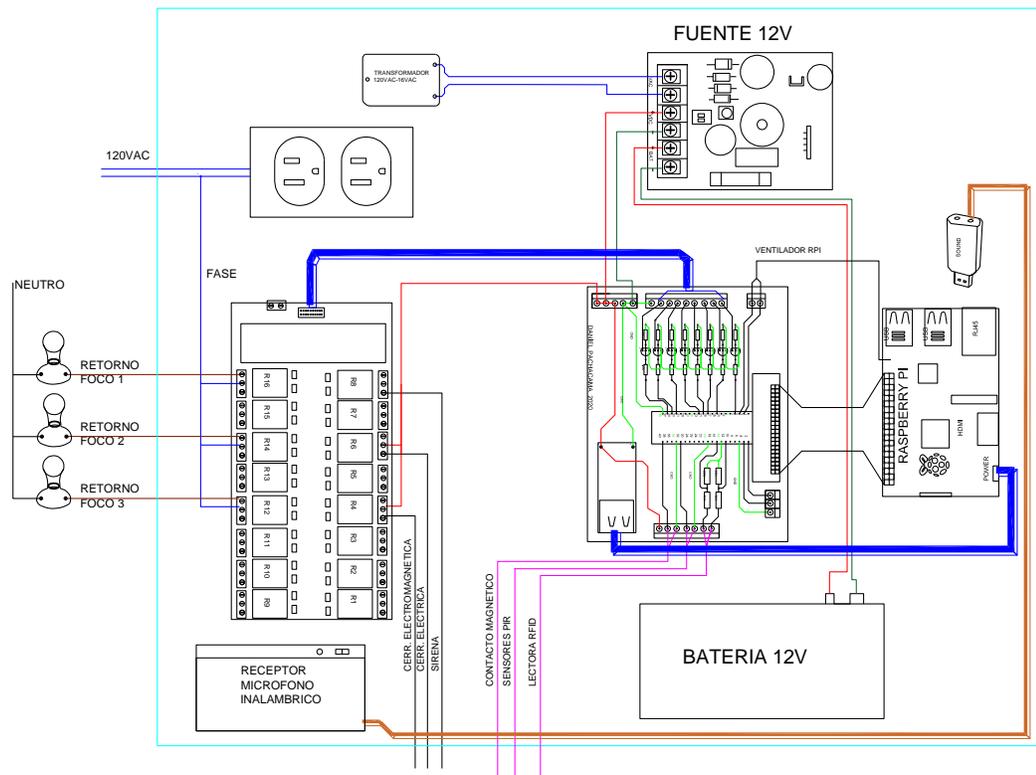


Figura 21: Esquema de conexiones tablero de Control. Fuente: Elaborado por el autor

El módulo de los relés servirá para alimentar las cerraduras electromagnéticas, la sirena, los focos de sala, cocina, y dormitorio. Por lo cual se ha dispuesto la conexión de la siguiente manera. Ver tabla 4

**Tabla 4:**  
Conexión de dispositivos a placas de relés

RELE	USO
Nº 16	FOCO SALA
Nº 14	FOCO COCINA
Nº 12	FOCO DORMITORIO
Nº 8	SIRENA
Nº 6	CERRADURA ELÉCTRICA
Nº 4	CERRADURA ELECTROMAGNÉTICA

Nota: Tabla adaptada por el autor



Figura 22: Tablero de control. Fuente: Elaborado por el autor

#### 4.5 Instalación del sistema de alarma.

Como su nombre lo indica el sistema alarma es aquel que a través de diversos sensores avise en caso de intrusión o intento y active una alarma.

El mismo está compuesto por contactos magnéticos que se instalaron en la puerta principal, dos detectores de movimiento infrarrojos pasivos que serán los que envíen una señal al controlador para activar una sirena de alarma. La activación y desactivación de la sirena se lo realizará desde el interior de la vivienda a través de comandos de voz.



Figura 23: Instalación contactos magnéticos. Fuente: (Amazon, 2020)

Para la instalación del sensor PIR se utiliza los cables de color azul y naranja para la alimentación y los cables color café para el envío de la señal.



Figura 24: Instalación de sensor PIR. Fuente: (Risco Group, 2020)

#### 4.6 Control de accesos.

Para la puerta principal se instaló una cerradura eléctrica que su apertura por la parte de afuera se la realizará a través de un lector RFID, y por el lado de adentro se abrirá a través de comandos de voz. Para la conexión de la cerradura utilizamos 2 pares del cable UTP, el par azul y el naranja.



Figura 25: Instalación de cerradura eléctrica. Fuente: Elaborado por el autor

#### 4.6.1 Conexión de lectora RFID

Para nuestro caso hemos utilizado una lectora de marca rosslare AY-k12 la cual utilizaremos 4 pares del cable UTP, para el control utilizaremos el par de color café para Dato 1, el par de color verde para los Datos 0, además para la alimentación utilizamos el par azul y naranja. Las salidas lógicas funcionan con un voltaje de 5V para la conexión a la Raspberry es necesario reducir este voltaje ya que terminales de la tarjeta trabajan con una tensión máxima 3.3V.

SALIDA	LECTOR	UTP
TIERRA	NEGRO	PAR NARANJA
DC +	ROJO	PAR ROJO
DATO 0	VERDE	PAR CAFE
DATO 1	BLANCO	PAR VERDE
LED	CAFE	
TAMPER	VIOLETA	

Figura 26: Conexión salida lectora con el cable UTP. Fuente: Elaborado por el autor



Figura 27: Lectora de proximidad instalada. Fuente: (Rosslare, 2019)

#### 4.6.2 Conexión de Cerradura electromagnética

Para la puerta interior se ha instalado una cerradura electromagnética, para la conexión de esta es necesario 2 pares del cable UTP el azul y el naranja, para la alimentación. Para la apertura de esta cerradura se necesita un corte de energía, esto lo realizaremos a través de un contacto cerrado de un relé de la placa de interfaz de relés.



Figura 28: Instalación cerradura electromagnética Fuente: (SECO-LARM, 2020)

#### 4.6.3 Conexión del Micrófono Inalámbrico

Para conectar del micrófono es necesario conectar el adaptador de sonido USB en la Raspberry esta tiene dos salidas una para audio y otro para el micrófono. En la salida del

micrófono conectamos el receptor del micrófono inalámbrico, y en la salida de audio conectamos un parlante

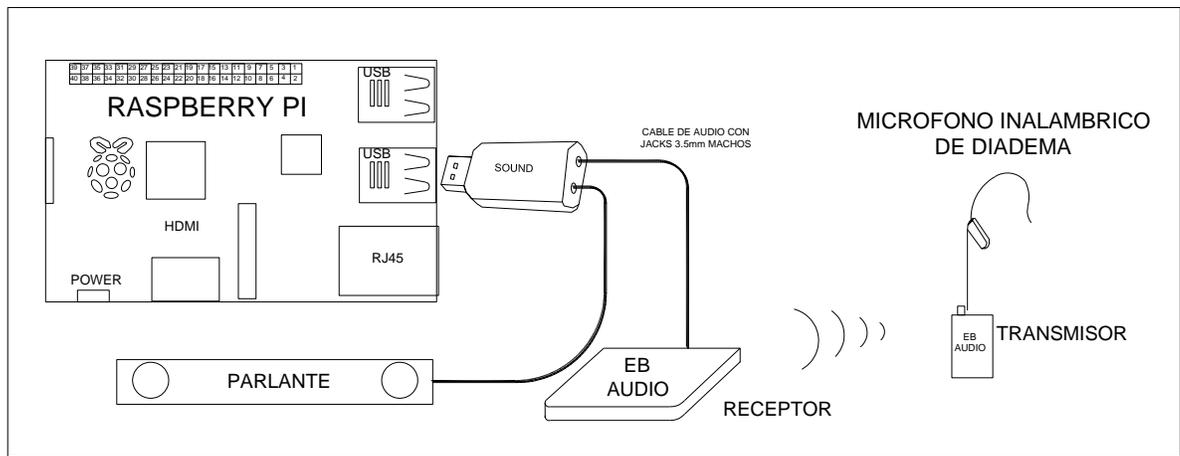


Figura 29: Conexión del micrófono. Fuente: Elaborado por el autor

## 4.7 Desarrollo de la programación

### 4.7.1 Instalación del sistema operativo Raspbian

1. Primero debemos formatear la memoria SD para esto utilizamos el programa SDFormater el mismo que puede descargar desde la siguiente página: <https://www.sdcard.org>.
2. Luego descargamos el programa “win32imager” que permite copiar y grabar imágenes. <https://sourceforge.net/projects/win32diskimager/>
3. Siguiendo descargamos la imagen .iso de la página oficial de Raspberry. <https://www.raspberrypi.org/downloads/>

En el programa win32imager, en la etiqueta “Imagen File” seleccionamos la ruta donde fue descargada la imagen, en la etiqueta “Device” seleccionamos la unidad y luego presionamos “Write” con esto se ha instalado el sistema operativo en la SD.

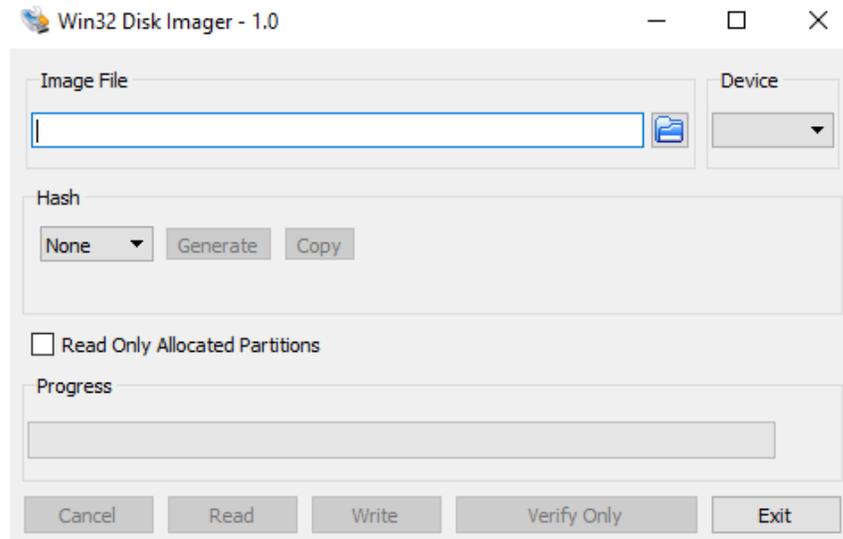


Figura 30: Programa win32imager Fuente: (Programa Win32 Disk Imager, 2020)

Una vez instalada la imagen en la memoria SD junto con el sistema operativo, se debe habilitar la conexión de acceso vía SSH:

Para esto ingresamos a CMD y digitamos el comando “**echo>D:\ssh**” y se crea un archivo SSH que habilita las conexiones remotas.

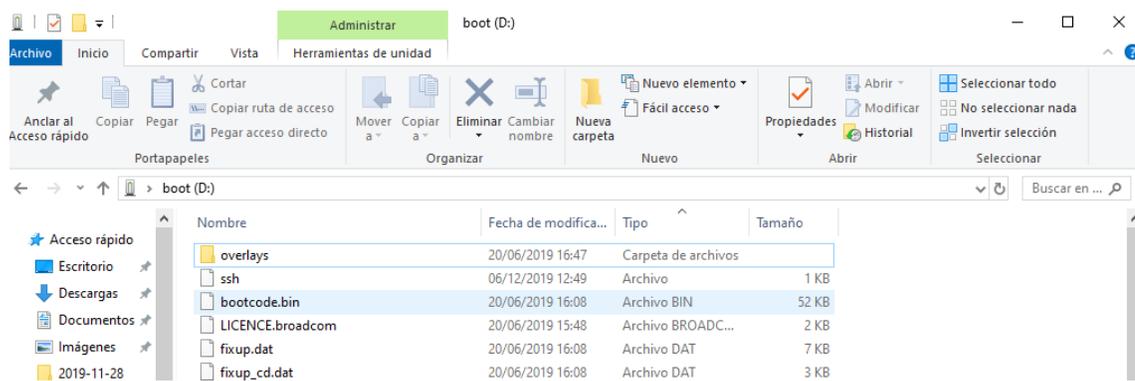


Figura 31: Verificación que se ha creado la carpeta SSH. Fuente: Elaborado por el autor

### 4.7.2 Asignación de una IP estática

Esto permite asignar una dirección estática a la tarjeta física de la PC con el fin de crear una red local con la tarjeta Raspberry, para esto ingresamos a Conexiones de Red inalámbrica, activamos la opción “Permitir que los usuarios de otras redes se conecten”.

Luego se asigna automáticamente la IP a nuestra tarjeta física. Con esto conseguimos que la IP del área local pueda compartir el internet a través de la red Wi Fi.

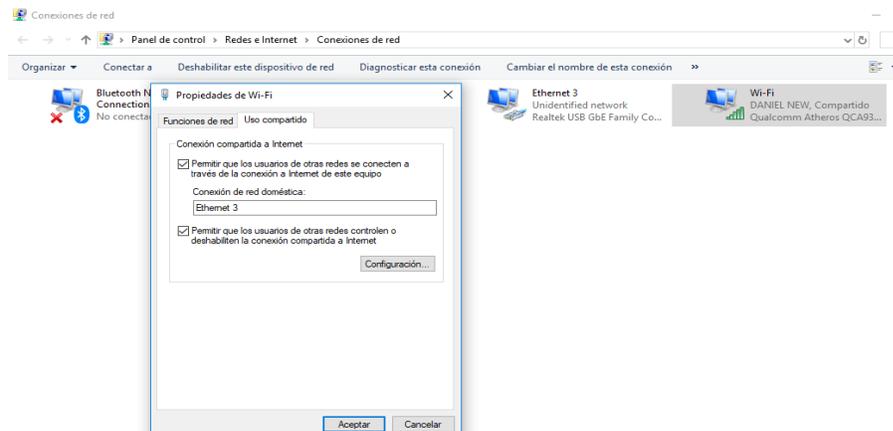


Figura 32: Configuración IP estática. Fuente: Elaborado por el autor

### 4.7.3 Conexión SSH

Con el comando `arp -a` en el CMD de la PC podemos verificar que IP fue asignada a la Raspberry y con esta dirección a través del programa Putty ingresamos a través de SSH a la tarjeta.

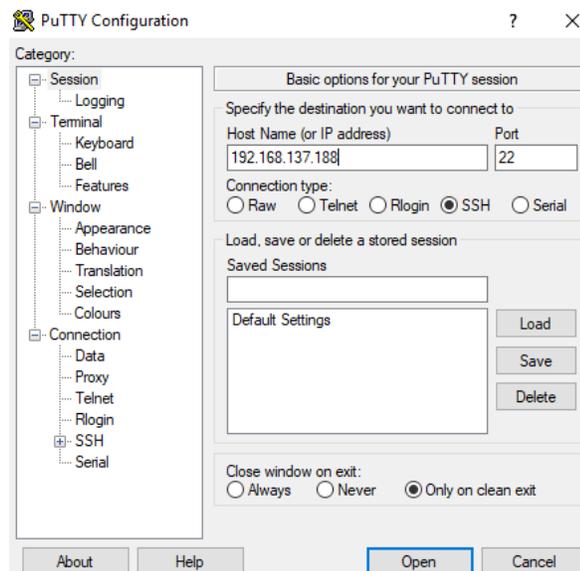


Figura 33: Conexión SSH a la Raspberry. Fuente: (Programa PuTTY, 2020)

#### 4.7.4 Entorno de trabajo e instalación de Actualizaciones

Una vez que ingresa se debe introducir usuario y contraseña los cuales por defecto son:

Usuario: pi

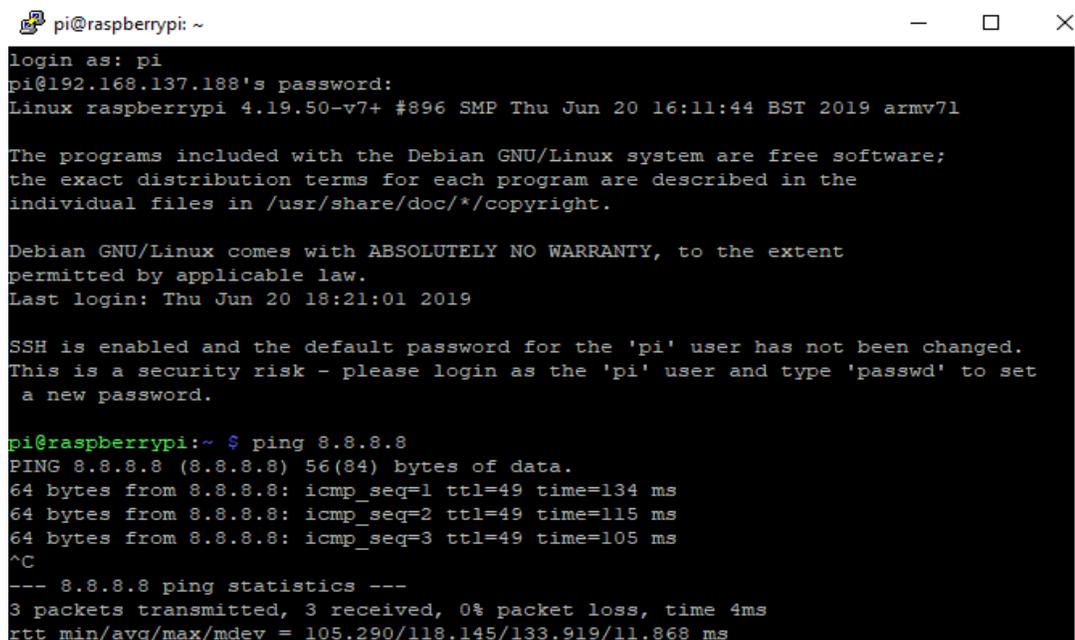
Contraseña: raspberry

Luego procedemos a verificar que tengamos conexión a internet con un comando Ping al servidor de google 8.8.8.8, comprobando que exista conexión procedemos a instalar y descargar las actualizaciones de los repositorios.

“Comandos:

**\$ sudo apt-get update** Permite buscar actualizaciones

**\$ sudo apt-get upgrade** Permite instalar las actualizaciones encontradas” (GitHub, 2020).



```
pi@raspberrypi: ~  
login as: pi  
pi@192.168.137.188's password:  
Linux raspberrypi 4.19.50-v7+ #896 SMP Thu Jun 20 16:11:44 BST 2019 armv7l  
  
The programs included with the Debian GNU/Linux system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
Last login: Thu Jun 20 18:21:01 2019  
  
SSH is enabled and the default password for the 'pi' user has not been changed.  
This is a security risk - please login as the 'pi' user and type 'passwd' to set  
a new password.  
  
pi@raspberrypi:~ $ ping 8.8.8.8  
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data:  
64 bytes from 8.8.8.8: icmp_seq=1 ttl=49 time=134 ms  
64 bytes from 8.8.8.8: icmp_seq=2 ttl=49 time=115 ms  
64 bytes from 8.8.8.8: icmp_seq=3 ttl=49 time=105 ms  
^C  
--- 8.8.8.8 ping statistics ---  
3 packets transmitted, 3 received, 0% packet loss, time 4ms  
rtt min/avg/max/mdev = 105.290/118.145/133.919/11.868 ms
```

Figura 34: Verificación de conexión a internet. Fuente: Elaborado por el autor

### 4.7.5 Instalación del servidor web Apache y PHP

“El servidor instalado es Apache, que es un servidor de HTTP de código abierto, para lo cual emitimos el comando” (APACHE, 2019):

```
$sudo apt-get install apache2
```

Cuando se termina la instalación se procede a verificar que se ha instalado correctamente para lo cual ingresamos al navegador web con la IP asignada.

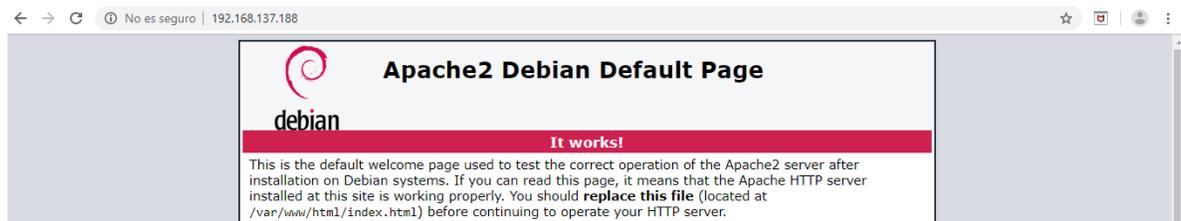


Figura 35: Verificación de instalación del servidor Apache. Fuente: Elaborado por el autor

Apache usa como directorio raíz de su sitio "/var /www /html" esto significa que cuando llama a la raspberry en el puerto 80 busca en el archivo "/var /www /html" esto servirá para crear un sitio HTML, JavaScript internamente. Pero para permitir interacciones entre el sitio y el usuario debe registrarse y para esto es necesario instalar PHP, esto sirve para que nuestras páginas soporten contenido dinámico, con PHP el usuario recibe la información ya procesada porque todo el código está en el servidor. Para esto instalamos el servidor PHP con el comando:

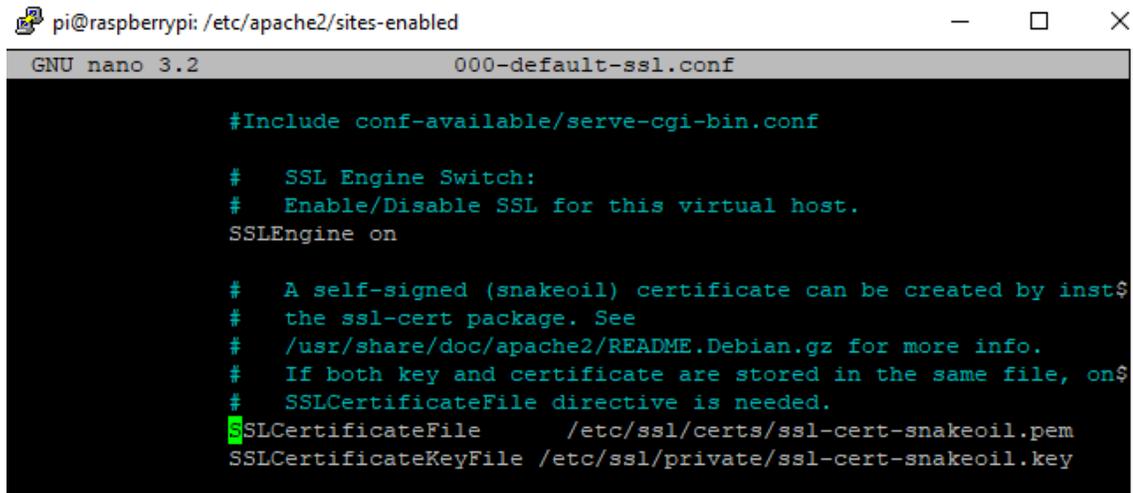
```
$ sudo apt-get install php7.0
```

### 4.7.6 Configuración de modo seguro HTTPS

Para ejecutar nuestro programa y que la página permita el acceso al micrófono, configuraremos un sitio seguro (https), el cual abriremos el directorio del servidor Apache y buscaremos el archivo de certificados denominado SSL, en el cual creamos el certificado y editaremos las líneas mostradas en la Figura 4.17 por las siguientes:

```
SSLCertificateFile /etc/apache2/ssl/server.crt
```

SSLCertificateKeyFile /etc/apache2/ssl/server.key



```

pi@raspberrypi: /etc/apache2/sites-enabled
GNU nano 3.2                                000-default-ssl.conf

#Include conf-available/serve-cgi-bin.conf

# SSL Engine Switch:
# Enable/Disable SSL for this virtual host.
SSLEngine on

# A self-signed (snakeoil) certificate can be created by installing
# the ssl-cert package. See
# /usr/share/doc/apache2/README.Debian.gz for more info.
# If both key and certificate are stored in the same file, the
# SSLCertificateFile directive is needed.
SSLCertificateFile /etc/ssl/certs/ssl-cert-snakeoil.pem
SSLCertificateKeyFile /etc/ssl/private/ssl-cert-snakeoil.key
  
```

Figura 36: Configuración HTTPS. Fuente: Elaborado por el autor

Luego reiniciaremos al servidor introduciendo el comando y estará listo.

```
sudo /etc/init.d/apache2 restart
```

#### 4.8 Descripción de la programación.

El programa está compuesta por diferentes subprogramas que se encuentran entrelazados entre sí, que se describen a continuación:

**Index.html:** Este contiene el programa para que se ejecute la librería annyang y en este archivo se configura el idioma y la clave para desactivar la alarma, (Clave 354).

**Index2.html:** En este archivo se ejecuta a través de un comando de voz que en este caso es la clave que desactiva la alarma, además se encuentra la programación del idioma que va utilizar en este caso Español ('es-MX') y se encuentra la programación de la palabra clave (LILI) esta es la que permite al programa reconocer que se está dando una instrucción al programa. En este archivo también se encuentra ligado al programa denominado Alexa.php.

**Alexa.php:** en este archivo se encuentra los comandos de voz para los accionamientos de los diferentes dispositivos de salida como son focos, cerraduras y alarma.

Todo esto lo puede realizar a través de comandos de voz los mismos que se describen a continuación:

**Tabla 5:**  
Comandos y acción que ejecuta el programa

<b>COMANDO DE VOZ</b>	<b>ACCION</b>
Lili 354	Clave de ingreso para desactivar alarma
Lili desactivar alarma	Desactiva la alarma y activa los comandos internos
Lili activar alarma	Activa el sistema de alarma sensores PIR, CM
Lili encender Sala	Enciende el foco de la sala
Lili apagar Sala	Apaga el foco de la sala
Lili encender cocina	Enciende el foco de la cocina
Lili apagar cocina	Apaga el foco de la cocina
Lili encender cuarto	Enciende el foco del dormitorio
Lili apagar cuarto	Apaga el foco del dormitorio
Lili encender casa	Enciende los focos de los tres ambientes
Lili apagar casa	Apaga los focos de los tres ambientes
Lili abrir puerta 1	Abre la puerta principal
Lili abrir puerta cocina	Abre la puerta de la cocina
Lili auxilio	Hace sonar la sirena en caso de emergencia
Lili 123	Apaga la sirena en caso de emergencia

Nota: Tabla adaptada por el autor

Cada uno de los comandos tiene asociado un programa que realiza la acción solicitada, al mencionar la palabra clave más el comando ejemplo (Lili encender foco sala) el programa envía a ejecutar el programa que está asociado al mismo, en este caso se ejecuta el programa denominado “efoco1.py” lo que hace es activar el pin de la raspberry asignado a el foco sala. Es así como se ejecuta cada uno de los comandos, a continuación se describe el funcionamiento de uno de estos.

**Efoco1.py:** este es el programa que se ejecuta una vez que se mencione la palabra clave y el comando, en este caso (Lili encender foco sala), el programa lo que hace es cambiar el estado de un pin GPIO de la raspberry de 0L a 1L, con esto se da la señal a un relé que accionará el foco de la sala, además para verificar que el programa está ejecutando la orden, se ha instalado la librería “pico2 wave” esta permite traducir el texto en señal de voz, para esto es necesario conectar a un parlante y este dará como respuesta (Encendiendo foco sala).

**wiegand.py:** Para esta tesis se está utilizando lectores RFID de 26 bits que son los comerciales, para lo cual es necesario realizar la lectura de las tarjetas, este programa es el que procesa los datos recibidos de la tarjeta y devuelve el número de la tarjeta.

**acceso.py:** Este programa consta de dos partes. La primera parte se encargada de verificar el estado de los sensores infrarrojos (PIR), contactos magnéticos, cuando el sistema está con la alarma activada, el cambio de estado de cualquiera de estos sensores hará que se dispare la sirena.

La segunda parte de este programa es que realiza una comparación del código recibido de la tarjeta RFID con el código almacenado en el programa si existe coincidencia se abrirá la puerta principal, una vez dentro de la casa tiene que decir la clave de acceso (354) y seguidamente (Lili desactivar alarma) y se encuentra listo para dar los comandos dentro de la casa.

#### 4.9 Pruebas de funcionamiento

Una vez instalado todos los componentes y realizado todas las conexiones se realiza las pruebas para verificar el funcionamiento del sistema.

Para lo cual se realiza la verificación de las palabras que está reconociendo el programa, para lo cual se procede a realizar una modificación en el programa index2, agregando la línea **alert(voz)**; para que muestre en el navegador los comandos que recibe por parte del micrófono.

```
function openPage(voz){
    alert(voz);
    window.open('alexa.php?voz='+voz, '_blank');
}
```

Figura 37: Modificación del programa index2.html. Fuente: Elaborado por el autor

Una vez realizada esa modificación abrimos el navegador web e introducimos la dirección de nuestra Raspberry en este caso <https://192.168.0.110/index2.html> y procedemos a mencionar los comandos y ver que palabras reconoce, del cual se obtiene los siguientes resultados.

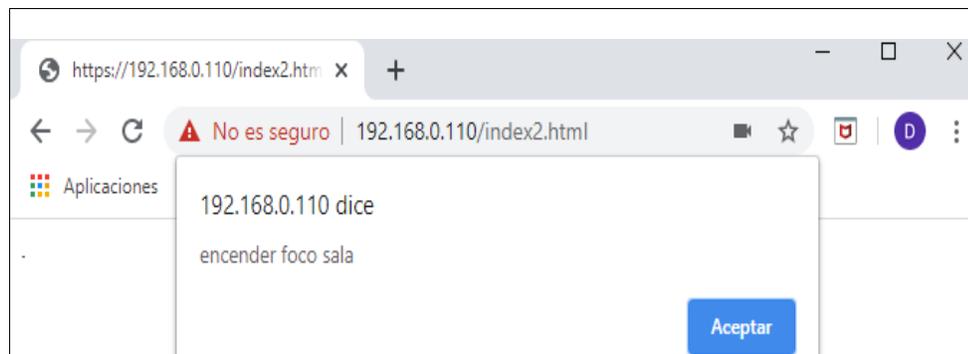


Figura 38: Comprobación de comandos reconocidos. Fuente: Elaborado por el autor

Con la comprobación de los resultados se obtuvieron los siguientes resultados (Tabla 6) en los cuales se valida que nuestro sistema reconozca el comando y que ejecute la acción a la cual fue diseñada.

Además de comprobar el reconocimiento se comprueba el funcionamiento del sistema de iluminación y de control de apertura de las puertas.

**Tabla 6:**  
Pruebas de funcionamiento

<b>COMANDO DE VOZ</b>	<b>RECONOCE EL COMANDO</b>	<b>EJECUTA EL PROGRAMA</b>
	<b>%</b>	<b>SI</b>
Lili 354	100%	X
Lili desactivar alarma	98%	X
Lili activar alarma	97%	X
Lili encender foco sala	98%	X
Lili apagar foco sala	99%	X
Lili encender foco cocina	98%	X
Lili apagar foco cocina	98%	X
Lili encender foco cuarto	97%	X
Lili apagar foco cuarto	98%	X
Lili encender casa	98%	X
Lili apagar casa	99%	X
Lili abrir puerta 1	97%	X
Lili abrir puerta cocina	97%	X
Lili auxilio	98%	X

Nota: Tabla adaptada por el autor

Mediante la siguiente tabla se verifica que nuestro prototipo tiene una precisión del 98% en cuanto al reconocimiento, este porcentaje se ha tomado en base a una tabla de 20 muestras por comando, y su diferencia radica en que se ha omitido alguna palabra ejemplo “Lili encender sala” en la cual se omitió la palabra foco por ende el sistema no realiza ninguna acción.

La tabla 7 muestra los resultados de una prueba de reconocimiento de los comandos tomadas en 10 muestras, la variable es el usuario (hombre y mujer) en la cual se verifica el reconocimiento de los comandos para las dos personas es satisfactorio.

**Tabla 7:**  
Prueba de reconocimiento hombre-mujer

COMANDO	HOMBRE	MUJER
Lili 354	10/10	10/10
Lili desactivar alarma	10/10	10/10
Lili activar alarma	10/10	10/10
Lili encender foco sala	9/10	8/10
Lili apagar foco sala	10/10	10/10
Lili encender foco cocina	9/10	10/10
Lili apagar foco cocina	10/10	10/10
Lili encender foco cuarto	8/10	9/10
Lili apagar foco cuarto	10/10	10/10
Lili encender casa	10/10	10/10
Lili apagar casa	10/10	10/10
Lili abrir puerta 1	10/10	10/10
Lili abrir puerta cocina	9/10	9/10
Lili auxilio	10/10	10/10

Nota: Tabla adaptada por el autor

- **Prueba de sistema de alarma**

Esta prueba consta de comprobar el funcionamiento de este subsistema, está compuesto por dos zonas de alarma, una que es el contacto magnético de la puerta principal y la segunda zona que son los sensores de presencia PIR. Para probar la zona uno, con la alarma activada

se procede a abrir la puerta principal obteniendo como resultado el sonido de la sirena, así también se procede a probar los sensores de movimiento PIR, con alarma activada provocamos un movimiento en el interior de la casa y se disparan los sensores que estos a su vez envían la señal que hacen accionar la sirena.

- **Prueba lectora de proximidad**

Se procede a verificar el funcionamiento de la lectora, para esto se hace la prueba de lectura con dos tarjetas magnéticas una con el código programado y otra con otro código diferente, para verificar que la tarjeta con el código válido produzca la apertura de la puerta principal, en la cual se obtiene un resultado favorable con el 100% de eficacia en la lectura.

#### 4.10 Análisis de resultados

Según las pruebas realizadas, se puede verificar la interacción del usuario con nuestro prototipo, que este a su vez permite la interacción con dispositivos eléctricos y electrónicos, de nuestro entorno cotidiano.

**Tabla 8:**  
Resumen de pruebas

Actividad	Cumple
Reconocimiento de comandos de voz	SI
Encendido y apagado de luces	SI
Activación del sistema de seguridad	SI
Detección de intrusos	SI
Notificación sonora en caso de alarma	SI
Lectura de tarjetas RFID	SI
Activación para apertura de puertas	SI

Nota: Tabla adaptada por el autor

En cuanto a los resultados de reconocimiento el sistema utiliza una palabra clave la cual permite reconocer cuando es una orden. Su margen de error es muy bajo y se ha producido porque el usuario no pronuncia bien el comando, pero no afecta al sistema ya que bastaría con repetir el comando para que se ejecute una orden.

Los datos de la tabla 8 resumen las pruebas realizadas, las cuales permiten evaluar nuestro prototipo. Se concluye que nuestro sistema cumple con los objetivos para los cuales fue diseñado, facilitando el uso de algunos dispositivos, enfocado a personas con discapacidad en sus extremidades.

## CONCLUSIONES

El prototipo está conformado de tres subsistemas iluminación; control del encendido y apagado de tres ambientes, control acceso; apertura de puertas, seguridad; detección de intrusos, activación y desactivación de alarma, estos manejan variables discretas ON-OFF que se manejan a través de relés, esto permitió el accionamiento de diferentes dispositivos de salida, como interruptores, cerraduras, sirena de alarma.

La arquitectura de la Raspberry permite una reducción de costos del sistema, una por ser de código abierto y segundo por tener una infraestructura con múltiples entradas y salidas que permiten controlar varios subsistemas en un dispositivo, la lógica de control se la implementó a través de una librería de internet de uso gratuito que se encuentra corriendo sobre un servidor web Apache, para enlazar el control con los dispositivos de salida se diseñó una placa PCB electrónica, que junto con una placa de interfaz de relés permite la interacción de la Raspberry con sensores y actuadores.

La programación realizada en la Raspberry permite reconocer comandos de voz que da el usuario mediante un micrófono inalámbrico, estos datos ingresan al sistema, este los compara con la librería de comandos programados si existe coincidencia entre ellos realiza la ejecución del comando, para la verificación de que el sistema recibió la orden se ha colocado un parlante que responde con la acción ejecutada si el proceso se realizó correctamente caso contrario el usuario deberá repetir el comando.

Con la implementación del prototipo en una casa se concluye que es funcional para los fines que fue desarrollado y al no ser un equipo mono marca permite adaptarse según las necesidades del usuario y en caso de daño de alguno de sus elementos pueda realizarse el cambio sin ningún problema.

Se valoró el rendimiento del sistema mediante un protocolo de pruebas en las que se verifica el reconocimiento y ejecución de los comandos, la compatibilidad del prototipo para hombres y mujeres, en la que se concluye que el prototipo con los algoritmos desarrollados, cumple con los objetivos desarrollados como es el de reconocer comandos de voz y permitir el accionamiento de sensores facilitando usuario.

## RECOMENDACIONES

- Mediante el presente prototipo puede ser de ayuda para futuras investigaciones en las cuales se pueda añadir otros sistemas, utilizar nuevas tecnologías y el objetivo es utilizar elementos comerciales de fácil acceso que no limiten al usuario a una sola marca.
- Probar insertar otros subsistemas en el prototipo, como circuito cerrado de televisión, detección de incendios con el fin de crear un sistema completo, con un desempeño de mayor alcance.
- El sistema trabaja mediante una librería en línea y por ende debe siempre estar conectado al a red para su funcionamiento, se debería desarrollar un algoritmo que no necesite dependencia de una red para su operación.
- La aplicación práctica del prototipo es aplicado para casas pequeñas ya que se encuentra limitado debido al radio de alcance del micrófono, se recomienda generar una solución que permita al usuario tener mayor cobertura.

## ANEXOS

- MANUAL DE SOPORTE TECNICO
- CRONOGRAMA DE ACTIVIDADES
- CIRCUITO ELECTRÓNICO PARA CONTROL PARA EL DISPARO DE RELÉS, CONEXIÓN DE LECTORA, SENSORES PIR Y CONTACTO MAGNÉTICO
- PLACA PCB PARA EL DISPARO DE RELÉS A TRAVÉS DE PINES GPIO
- ESQUEMA DE CONEXIÓN PLACA PCB
- ESQUEMA GENERAL DE CONEXIÓN EN TABLERO

### PROGRAMACION:

- ALGORITMO PARA RECONOCER COMANDOS DE VOZ, Y CLAVE DE ACCESO PARA DESACTIVACIÓN DEL SISTEMA DE ALARMA.
- ALGORITMO RECOGE LOS COMANDOS DE VOZ, LOS COMPARA CON NUESTRA LIBRERÍA Y EJECUTA LOS PROGRAMAS PYTHON.
- CÓDIGO PARA ACTIVAR Y DESACTIVAR EL SISTEMA DE ALARMA Y LECTURA DE TARJETA WIEGAND.
- LIBRERÍA DE COMANDOS, EL ALGORITMO DIRECCIONA Y EJECUTA LOS SCRIPTS DE PHYTON.
- PROGRAMAS PARA ACTIVACIÓN DE FOCOS, CERRADURAS Y ALARMA.

## REFERENCIAS BIBLIOGRÁFICAS

- Álvarez García, V. (s.f.). *Estándar VoiceXML*. Obtenido de <http://di002.edv.uniovi.es/~cueva/asignaturas/doctorado/2006/trabajos/VoiceXML.pdf>
- Annyang. (15 de 07 de 2019). *annyang*. Obtenido de <https://www.talater.com/annyang/>
- APACHE. (2019). *The Apache software foundation*. Obtenido de <https://www.apache.org/foundation/how-it-works.html>
- Domínguez, H. M., & Sáez, F. (Junio de 2006). *Domótica: Un enfoque sociotécnico*. (F. R. Segovia, Ed.) Recuperado el Julio de 2019, de [http://lsi.vc.ehu.es/pablogn/investig/dom%C3%B3tica/libro\\_domotica.pdf](http://lsi.vc.ehu.es/pablogn/investig/dom%C3%B3tica/libro_domotica.pdf)
- GitHub. (11 de 05 de 2020). *github*. Obtenido de <https://github.com/>
- Gutiérrez Olivares, J. (2020). *Taller de PHP*. Obtenido de [http://tutoriales.altervista.org/trabajos/Manual\\_php\\_completo.pdf](http://tutoriales.altervista.org/trabajos/Manual_php_completo.pdf)
- Hostinger. (15 de 07 de 2019). *Tutorial Hostinger*. Obtenido de <https://www.hostinger.es/tutoriales/que-es-apache/>
- ITU-R. (08 de 2015). *Nomenclatura de las bandas de frecuencias y de las longitudes de onda empleados en telecomunicaciones*. Obtenido de [https://www.itu.int/dms\\_pubrec/itu-r/rec/v/R-REC-V.431-8-201508-I!!PDF-S.pdf](https://www.itu.int/dms_pubrec/itu-r/rec/v/R-REC-V.431-8-201508-I!!PDF-S.pdf)
- Kanllli, Liege, J., & Lostalé, E. (2018). *La era de la voz: Asistentes virtuales y voice marketing*. Obtenido de <https://www.kanlli.com/wp->

content/uploads/2018/09/LA\_ERA\_DE\_LA\_VOZ\_ASISTENTES\_VIRTUALES\_  
Y\_VOICE\_MARKETING.pdf

Mejía Tamayo, F. X. (2011). *DISEÑO E IMPLEMENTACION DE UN SISTEMA DOMOTICO CONTROLADO POR VOZ PARA REALIZAR CONSULTAS RUTINARIAS Y EL CONTROL Y OPERACIÓN DE UN SISTEMA DE SEGURIDAD ANTI INTRUSOS, BASADO EN UN SITEMA DE MULTIPLES MICROFONOS Y PARLANTES*. Obtenido de

<https://bibdigital.epn.edu.ec/bitstream/15000/16823/1/CD-3531.pdf>

Oropeza Rodriguez, J. L. (15 de 12 de 2006). *Algoritmos y Métodos para el*

*Reconocimiento de Voz en Español Mediante*. Obtenido de

<http://www.scielo.org.mx/pdf/cys/v9n3/v9n3a7.pdf>

Pallás Areny, R. (2005). *Sensores y acondicionadores de señal*. Obtenido de

[https://www.academia.edu/25807463/Sensores\\_y\\_acondicionadores\\_de\\_senal\\_ramon\\_pallas\\_areny](https://www.academia.edu/25807463/Sensores_y_acondicionadores_de_senal_ramon_pallas_areny)

RASPBERRY PI FOUNDATION UK. (23 de 04 de 2020). *Raspberry Pi*. Obtenido de

<https://projects.raspberrypi.org/en>

Risco Group. (2020). *CoMET PIR*. Obtenido de

<https://www.riscogroup.com/spain/products/product/5474>

Rosslare. (15 de 08 de 2019). *Rosslare Security Products*. Obtenido de

[http://jhbdisenoeingenieria.com/files/rosslare/Datashee\\_AY-X12.pdf](http://jhbdisenoeingenieria.com/files/rosslare/Datashee_AY-X12.pdf)

SECO-LARM. (2020). *SECO-LARM*. Obtenido de [http://www.seco-](http://www.seco-larm.com/es/Security/security-power-supplies/ST-1206-1.5AQ)

[larm.com/es/Security/security-power-supplies/ST-1206-1.5AQ](http://www.seco-larm.com/es/Security/security-power-supplies/ST-1206-1.5AQ)

Tipán Bautista, E. M. (2018). *Universidad Israel*. Obtenido de PROTOTIPO DE UNA CASA INTELIGENTE, CONTROLADA A TRAVÉS DE UN CELULAR ANDROID VÍA GSM.: <http://repositorio.uisrael.edu.ec/handle/47000/1631>

UNAM. (Abril de 2020). *Manual HTML*. Obtenido de <http://profesores.fi-b.unam.mx/cintia/Manualhtml.pdf>

Van Rossum, G. (Septiembre de 2009). *El tutorial de Python*. Obtenido de <http://docs.python.org.ar/tutorial/pdfs/TutorialPython2.pdf>

**MANUAL**  
**DE SOPORTE TÉCNICO**

## **INTRODUCCIÓN**

El manual describe el funcionamiento y puesta en marcha del sistema, para que cualquier persona con pocos conocimientos en sistemas electrónicos pueda solucionar cualquier problema del PROTOTIPO DE UN SISTEMA DE UNA CASA INTELIGENTE, CONTRALADO A TRAVÉS DE COMANDOS DE VOZ PARA PERSONAS CON DISCAPACIDAD DE SUS EXTREMIDADES SUPERIORES.

## **OBJETIVOS**

- Ofrecer la información necesaria para su instalación y puesta en marcha del sistema.
- Describir la arquitectura del sistema y su correcta funcionalidad.
- Detallar las especificaciones y requerimientos necesarios para el funcionamiento del sistema.
- Describir la lógica de la programación con el fin de detectar posibles averías.

### **Requerimientos técnicos**

#### **Requerimientos mínimos de hardware**

- Raspberry Pi modelo 4B
- Fuente de poder con respaldo de batería 12VDC- 5A, Marca Enforcer ST-2406-5AQ
- Placa de interfaz de relé 12v-16canales equipados con relés de alta corriente capaz de soportar 10A-250VAC; 10A-30VDC.
- Batería 12V- 7Ah, Marca Casil CA1270

- Micrófono Inalámbrico, Marca EB sound
- Detectores de movimiento PIR, Marca Risco CoMET PIR RK210PR
- Lectora de proximidad, Marca Rosslare AY-K12

### **Requerimientos mínimos de software**

- Sistema operativo Raspbian
- Privilegios de administración

### **Herramientas utilizadas para el desarrollo**

#### **SD Card Formatter**

Programa de descarga gratuita que sirve para dar formato a las tarjetas de memoria compatible con memorias SD, SDHC y SCXC.

#### **Win32DiskImager**

Programa gratuito que permite grabar una imagen de disco sin formato en un dispositivo extraíble, además permite realizar copias de respaldo.

#### **Putty**

Es un programa que permite la conexión a través de SSH y Telnet a nuestro dispositivo en este caso la Raspberry, en la que podremos conectarnos de forma remota iniciando una sesión, este permita realizar la programación y ejecutar comandos.

## **Especificaciones Técnicas.**

El dispositivo que va realizar toda la lógica de control de nuestro sistema es la Raspberry Pi, la cual realiza la interacción con los dispositivos electrónicos y esta su vez la interacción con el usuario. Las especificaciones se describen a continuación.

### **Especificaciones técnicas Raspberry pi 4B.**

- Broadcom BCM2711, Quad core Cortex-A72 (ARM v8) SoC de 64 bits a 1.5GHz
- SDRAM LPDDR4-3200 de 1 GB, 2 GB o 4 GB (según el modelo)
- 2.4 GHz y 5.0 GHz IEEE 802.11ac inalámbrico, Bluetooth 5.0, BLE
- Gigabit Ethernet
- 2 puertos USB 3.0; 2 puertos USB 2.0.
- Cabezal GPIO estándar de 40 pines de Raspberry Pi (totalmente compatible con placas anteriores)
- 2 puertos micro HDMI (hasta 4kp60 compatibles)
- Puerto de pantalla MIPI DSI de 2 carriles
- Puerto de cámara MIPI CSI de 2 carriles
- Puerto de audio compuesto y video compuesto de 4 polos
- H.265 (decodificación 4kp60), H264 (decodificación 1080p60, codificación 1080p30)
- Gráficos OpenGL ES 3.0

- Ranura para tarjeta micro SD para cargar el sistema operativo y el almacenamiento de datos
- 5V DC a través del encabezado GPIO (mínimo 3A \*)
- Alimentación a través de Ethernet (PoE) habilitada (requiere un PoE HAT separado)
- Temperatura ambiente de funcionamiento: 0 - 50 grados C

### **Requerimientos de alimentación de AC**

Para la alimentación de este dispositivo es necesario una fuente de alimentación 110AC a 5VDC con puerto de entrada tipo USB-C de 15W, es recomendable usar una fuente original para Raspberry.

### **Requerimientos software**

Para el funcionamiento de la Raspberry es necesario instalar una tarjeta externa micro SD en la cual se deberá cargar la imagen con el sistema operativo Raspbian, el cual puede ser descargado de la página oficial.

### **Fuente de alimentación**

Para nuestro sistema es necesario tener una fuente que tenga la capacidad disponible para manejar todos los dispositivos electrónicos que compone el sistema, además esta deberá incluir una batería de respaldo, en caso de que falle la corriente eléctrica el sistema de control de acceso permita el ingreso a la casa.

## **Especificaciones técnicas**

- Convierte la energía de entrada CA en energía de salida de CC de bajo voltaje
- Interruptor selectivo para salida de voltaje 6-12-24V
- Consumo de corriente máxima 5A
- Voltaje de salida regulado y filtrado
- Protección contra cortocircuitos
- Protección por compensación térmica
- Cargador de batería de respaldo incorporado
- Cambia automáticamente en caso de pérdida de energía
- Salida con fusible para una carga regulada
- 2 indicadores de estado de entrada de Alimentación Alterna y Continua
- Compatible con baterías recargables de Plomo Acido o de tipo Gel
- Protección de polaridad inversa

## **Batería de respaldo**

La batería usada es de plomo ácido sellado 12V-7AH la cual funciona con fuente de respaldo para alimentar las cerraduras en caso de fallo de energía eléctrica.

<b>Nominal Voltage</b>		<b>12V</b>
<b>Rated Capacity 77<sup>0</sup>F(25<sup>0</sup>C) (20HR)</b>		<b>7.0Ah</b>
<b>Dimensions (mm/inch)</b>	<b>Length</b>	<b>151 (5.95)</b>
	<b>Width</b>	<b>65 (2.56)</b>
	<b>Height</b>	<b>95 (3.74)</b>
	<b>Total Height</b>	<b>101 (3.98)</b>
<b>Approx. Weight (kg/Ibs)</b>		<b>2.0 (4.41)</b>
<b>Terminal</b>		<b>T1/T2</b>

### **Placa de interfaz de relés 16 canales**

Para manejar la parte de la carga es decir las luces, sirena, cerraduras se ha previsto instalar un módulo de relés el mismo que puede manejar cargas que trabajan en corriente alterna con voltajes hasta 250VAC como en corriente continua hasta 30VDC.

### **Especificaciones Técnicas**

- Número de canales: 16Ch.
- Voltaje de control señal: 5V, bajo activo.
- Voltaje de carga: AC 250V/10A, AC125V/15A, DC 30V/10A.
- Activación de puerto: se activa en nivel bajo 0L
- Indicadores LED's para los relés estado de salida.
- Usa opto acopladores de protección para evitar daño en la placa.

## **Lectora de proximidad**

Para el control de acceso de la puerta principal se ha utilizado lectora Rosslare AY-K12 la misma que permite el acceso a través de una tarjeta magnética con las siguientes especificaciones:

### **Especificaciones técnicas:**

- Rango de voltaje de operación 5-16VDC
- Entrada máxima de corriente 35mA en standby y 50mA en lectura
- Distancia máxima del cable para el controlador 150m
- Rango de temperatura de operación -31°C a 63°C
- Rango de humedad de operación 0-95% (no condensado)

## **Detector de movimiento PIR (RK210PR)**

Para la detección de movimiento en caso de intrusión de personas no autorizadas se ha previsto instalar 2 Detector PIR marca Risco.

### **Especificaciones Técnicas**

- Voltaje de operación: 9 hasta 16 VDC.
- Contacto Alarma: normalmente cerrado (N.C.) 0.1Amp, 24VDC.
- Contacto Tamper: normalmente cerrado (N.C.) 0.1Amp, 24VDC.
- Compensación automática de temperatura.

- Filtración óptica: protección contra luz blanca, lentes pigmentadas.
- Alcance de detección: 12mx12m.
- Temperatura de operación -5°C hasta +50°C.

### **Micrófono inalámbrico**

Para que el usuario pueda dar los comandos de voz al sistema se utilizara un micrófono inalámbrico marca EB Sound con las siguientes especificaciones:

- Rango de Frecuencia: VHF 200-270Mhz
- Impedancia de salida 600Ω
- Alimentación: Transmisor-2 Pilas AA 1.5V, Receptor -2 Pilas AA 1.5V o Fuente 5V-500mA.
- Distancia de alcance 50m
- Salida de Audio receptor Jack 3.5
- 1 Canal
- Rango Dinámico: más de 110dB
- Respuesta de Frecuencia 80hz-18khz

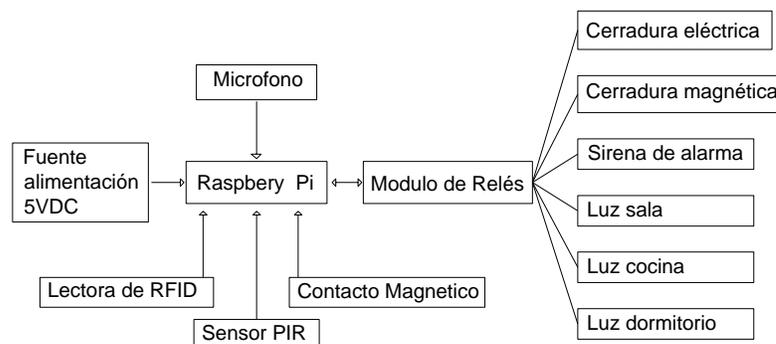
### **Requerimientos para instalación del sistema**

Para el funcionamiento de nuestro sistema es necesario instalar un circuito independiente 110V para alimentación de nuestro tablero el alimentador deberá ser 14(14)+14THHN desde el tablero de energía con un breaker de protección de 10A.

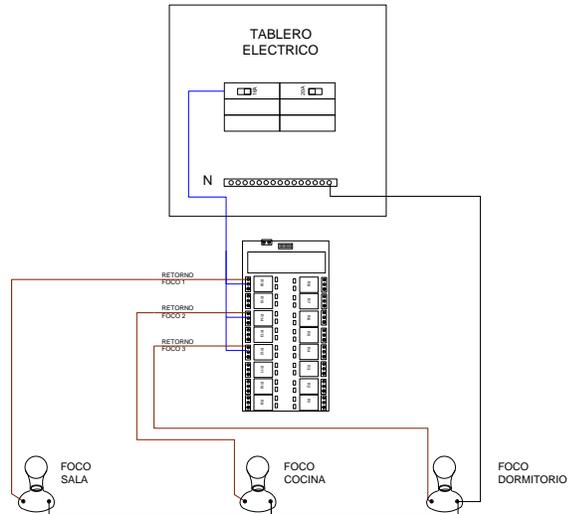
Además del punto eléctrico es necesario un punto de red para el funcionamiento de nuestro sistema, este cable va desde el switch de internet hasta la Raspberry podrá ser con un cable UTP cat 5.

### Cableado y conexión

Desde el tablero se procede a realizar el cableado hasta los diferentes elementos:



**Sistema de iluminación** primero es necesario identificar los circuitos en el tablero de iluminación para verificar que breaker está alimentando a cada foco. En nuestro caso hay un solo breaker para el circuito de iluminación, para lo cual es necesario cablear un solo cable desde la salida del breaker (fase) hacia el tablero de control, y para el control de los focos es necesario cablear un conductor N°16 THHN AWG desde cada foco hasta el tablero de control identificando en los extremos a que foco pertenece cada cable.

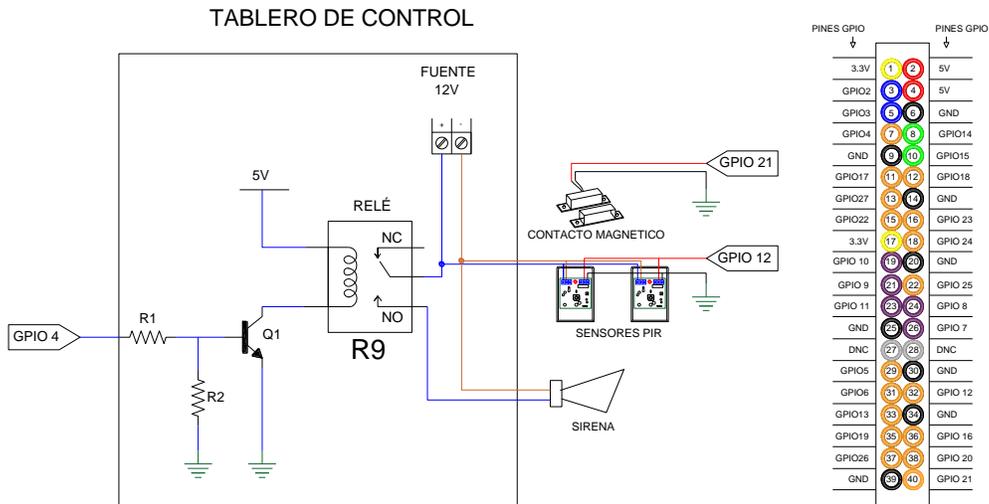


**Ilustración 1. Diagrama de conexión luces**

### **Sistema de alarma**

Para los elementos de alarma se utilizara cable UTP cat 5E para los cuales es necesario un cable hacia cada dispositivo desde el tablero de control hasta los sensores, para los Detectores PIR es necesario llegar con un cable UTP hasta el sensor más cercano en el cual se utiliza cuatro pares, dos para alimentación y dos para señal, la conexión entre PIR es en paralelo para esto es necesario un cable UTP entre los PIR.

Para el contacto magnético es necesario un cable UTP del cual se utiliza un par que es el que envía la señal del estado de la puerta este va hasta el tablero de control.



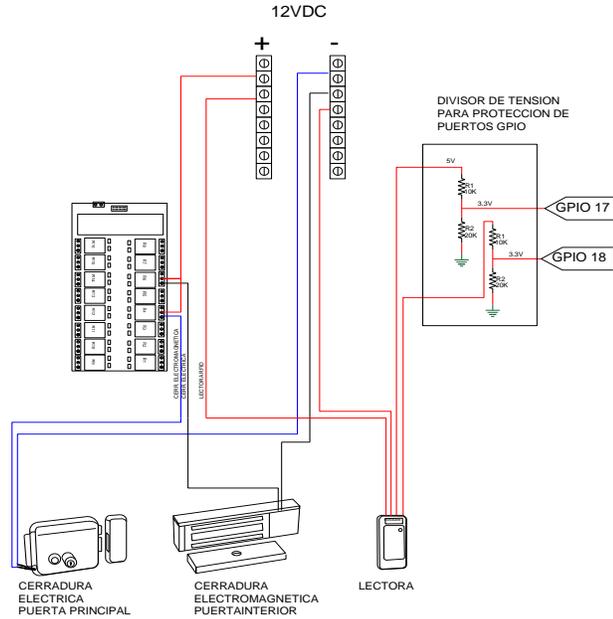
**Ilustración 2. Diagrama de conexión alarma**

**Distribución de pines GPIO**

En el esquema se indica la conexión en el tablero de control y la placa de interfaz en la cual se utilizara el relé N° 9 y los pines GPIO de la Raspberry los cuales se conectan cada dispositivo.

### **Sistema de control de Acceso**

Este sistema involucra la conexión de las cerraduras y de la lectora de proximidad para lo cual es necesario cablear desde el tablero de control un cable UTP para cada elemento.

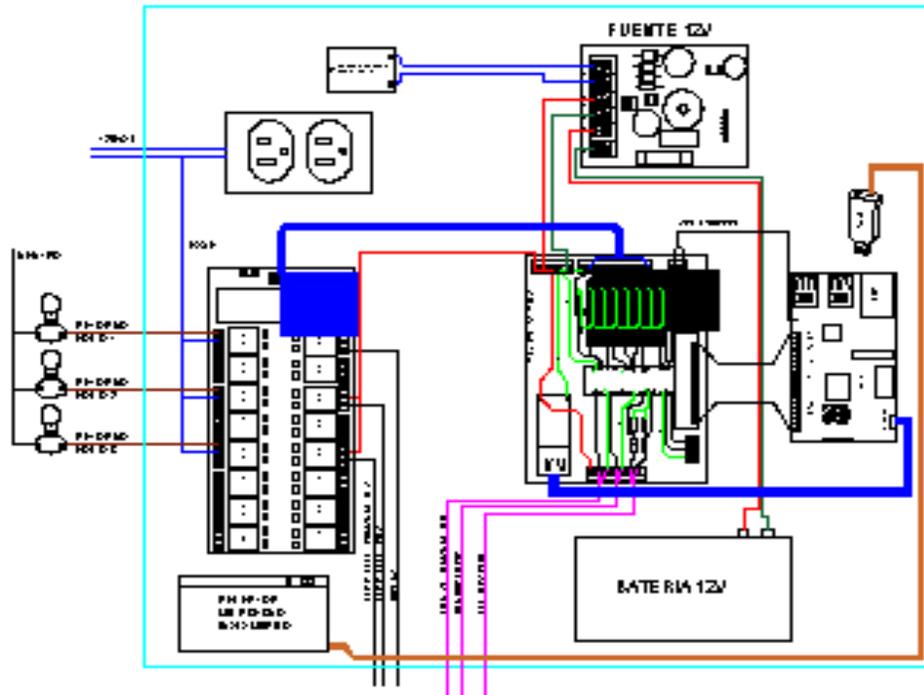


Al realizar el cableado no olvidar etiquetar los cables en las puntas para saber que dispositivo vienen y que pares del cable se están utilizando para señal y cuales para alimentación para luego proceder a la conexión. Para no cometer errores hemos definido los colores de los cables a usar para conexión del cable UTP a los dispositivos.

COLORES	DESCRIPCION
AZUL	ALIMENTACION TERMINAL POSITIVO
AZUL-BLANCO	ALIMENTACION TERMINAL POSITIVO
NARANJA	ALIMENTACION TERMINAL NEGATIVO
NARANJA-BLANCO	ALIMENTACION TERMINAL NEGATIVO
CAFÉ	SEÑAL
CAFÉ-BLANCO	SEÑAL

VERDE	SEÑAL
VERDE-BLANCO	SEÑAL

Para las conexiones en el tablero se tiene el siguiente esquema:



En el tablero tenemos un tomacorriente de 110v el mismo que será alimentado desde el tablero eléctrico como se describió anteriormente, este tomacorriente nos servirá para energizar la fuente de alimentación de 12v y para alimentar a la Raspberry.

Desde la fuente de 12v se conectara de los bornes denominados como BAT hacia la batería de respaldo tal como se indica en el esquema.

La fuente de alimentación se utiliza para alimentar de la placa de interfaz de relés que esta a su vez maneja las cerraduras y la sirena, además desde la barra de 12v se alimenta a la lectora de RFID y a los detectores PIR que trabajan a ese voltaje.

### Conexión del micrófono

Para la conexión del micrófono es necesario concertar la tarjeta de sonido USB en la Raspberry esta tiene dos salidas una para audio y otro para el micrófono. En la salida del micrófono conectamos el receptor del micrófono inalámbrico.

Tanto el receptor como el transmisor del micrófono funcionan con pilas normales AA 1.5V

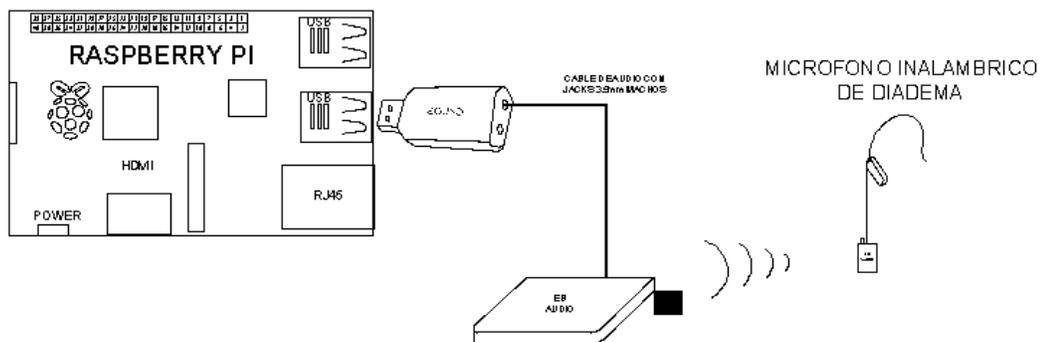


Ilustración 3. Conexión del micrófono

Una vez instalado y realizadas todas las conexiones indicadas se proceden a verificar que todas las conexiones se encuentren correctamente realizadas y que no existan cables flojos para evitar falsas alarmas.

Luego se procede a subir el breaker que alimenta a nuestro tablero y se debe verificar antes de conectar los adaptadores al tomacorriente con un multímetro que el voltaje sea el adecuado 110-120VAC.

Una vez verificado se procede a conectar los adaptadores de la fuente DC y la Raspberry.

El cable de la Raspberry tiene un botón de encendido y apagado, se deberá presionar una vez para encender la Raspberry esperar 2 minutos hasta que arranque el sistema operativo y cargue toda la lógica del programa.

Luego se procede a encender el transmisor y el receptor del micrófono los dos tienen un interruptor de encendido y apagado.

Y está listo para utilizar el sistema.

### **Funcionamiento del sistema**

Para su funcionamiento es necesario que el usuario se encuentre con el micrófono encendido tanto el receptor como el transmisor.

El sistema permite encender y apagar las luces de tres ambientes sala, cocina y dormitorio; activar y desactivar el sistema de alarma; abrir puertas y dar una señal de auxilio en caso de emergencia para que se encienda la alarma.

Todo esto lo puede realizar a través de comandos de voz los mismos que se describen a continuación detallando cual es la orden del comando y la acción que se produce.

<b>COMANDO DE VOZ</b>	<b>ACCION</b>
Lili 354	clave de ingreso para desactivar alarma
Lili desactivar alarma	desactiva la alarma y activa los comandos internos

Lili activar alarma	activa el sistema de alarma sensores PIR, CM
Lili encender Sala	Enciende el foco de la sala
Lili apagar Sala	Apaga el foco de la sala
Lili encender cocina	Enciende el foco de la cocina
Lili apagar cocina	Apaga el foco de la cocina
Lili encender cuarto	Enciende el foco del dormitorio
Lili apagar cuarto	Apaga el foco del dormitorio
Lili encender casa	Enciende los focos de los tres ambientes
Lili apagar casa	Apaga los focos de los tres ambientes
Lili abrir puerta 1	Abre la puerta principal
Lili abrir puerta cocina	Abre la puerta de la cocina
Lili auxilio	hace sonar la sirena en caso de emergencia
Lili 123	apaga la sirena en caso de emergencia

### **Sistema de alarma**

El sistema está compuesto por dos sensores PIR, un contacto magnético para la puerta principal y una sirena de aviso.

Cuando la alarma esta activada el sistema está censando el estado de los sensores PIR y del contacto magnético, en caso se encuentra en estado normalmente cerrado, cuando se

produzca un cambio de estado por presencia de un intruso o por apertura de la puerta principal el sistema activara la alarma.

Para desactivar la alarma el usuario deberá decir la clave y seguidamente el comando Lili desactivar alarma y se apagara la alarma.

En el caso de que la alarma esta desactivada el sistema lo único que hace es no activar la alarma en presencia de movimiento y apertura de la puerta.

Para activar la alarma el usuario deberá verificar que la puerta interior este cerrada y como consejo el usuario deberá decir el comando “**Lili apagar casa**” para asegurarse de que todos los focos estén apagados, luego deberá salir cerrar la puerta principal y decir el comando “**Lili Activar alarma**” y se activara la alarma.

### **Sistema de acceso**

Cuando el sistema de alarma este activada y el usuario quiera entrar a su hogar, en el ingreso se encuentra ubicado un lector de tarjetas RFID que permitirá la apertura de la puerta principal a través de una tarjeta magnética.

El usuario acerca la tarjeta sobre el lector y se produce la apertura de la puerta, el usuario ingresa a la casa, tiene 15 segundos para decir la clave y el comando Lili desactivar alarma, caso contrario se activara la sirena.

Cuando la alarma está desactivada, es decir, el usuario está en la casa, el usuario puede abrir cualquiera de las dos puertas mediante comando de voz.

## Sistema de Luces

Los comandos para encendido de los tres ambientes de la casa (sala, cocina y dormitorio) y apertura de puertas funcionan únicamente si la alarma esta desactivada, con los comandos antes descritos, en los cuales el usuario deberá mencionar cualquier comando según la acción que quiera hacer ejemplo: **“Lili encender foco sala”** enciende el foco de la sala.

## Ingresar al programa

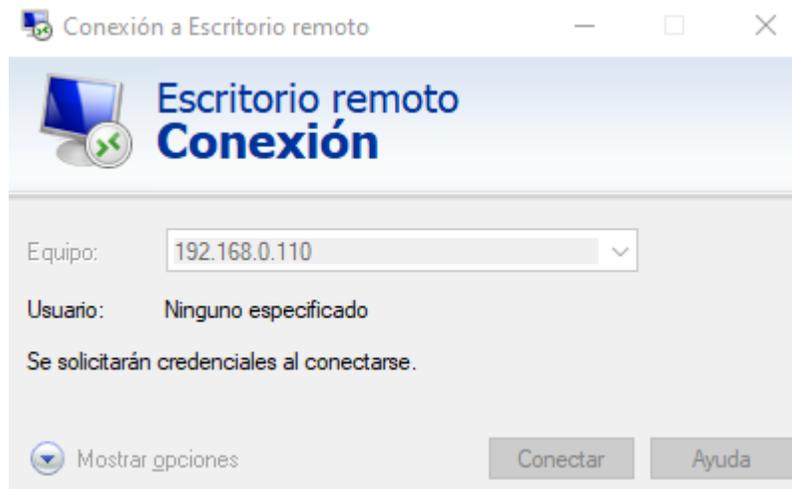
Para ingresar a la programación de nuestra tarjeta Raspberry es necesario tener una computadora ya que el ingreso lo haremos a través de una conexión remota SSH. Para lo que es necesario conectarse a la red local ya sea por cable o por WIFI, tener instalado el programa PuTTY, y necesitamos conocer cuál es la IP que está asignada la Raspberry.

La IP de nuestra Raspberry la podemos verificar en nuestro router en la opción **DHCP Client List** , la IP configurada es la 192.168.0.110

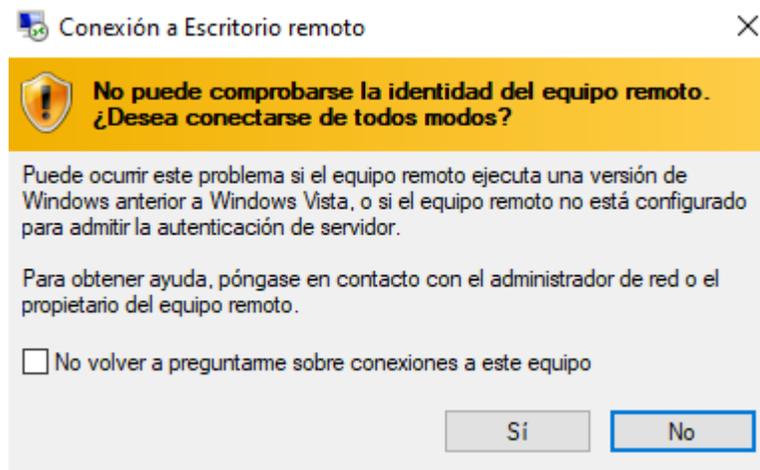
### DHCP Client List

ID	Client Name	MAC Address	Assigned IP	Lease Time
1	Galaxy-A20	F0-8A-76-34-C2-C8	192.168.0.103	01:45:08
2	Honor_8X-98c74128233b58e0	7C-94-2A-0F-91-CA	192.168.0.105	01:43:34
3	HUAWEI_P_smart-91ca7ee9e8	F0-0F-EC-4D-50-12	192.168.0.106	01:47:28
4	HUAWEI_P_smart-a235c515ba	B4-86-55-37-69-77	192.168.0.101	01:16:15
5	Windows-Phone	E8-15-0E-0B-21-04	192.168.0.104	01:43:25
6	WINDOWS-FEV9SHR	F4-06-69-AE-95-61	192.168.0.102	01:59:32
7	android-7db082586cf3f051	BC-B3-08-B2-68-23	192.168.0.100	01:40:35
8	LGSmarTV	3C-CD-93-F0-63-86	192.168.0.107	01:41:25
9	raspberrypi	DC-A6-32-1B-92-21	192.168.0.110	01:57:35

Para verificación de que nuestro programa está corriendo se debe verificar ingresando al entorno grafico de nuestra Raspberry a través de una conexión remota, para esto en el buscador de Windows Escribimos **Conexión a escritorio remoto** e ingresamos ya en esta ventana introducimos la dirección IP de nuestra Raspberry y damos clic en conectar.



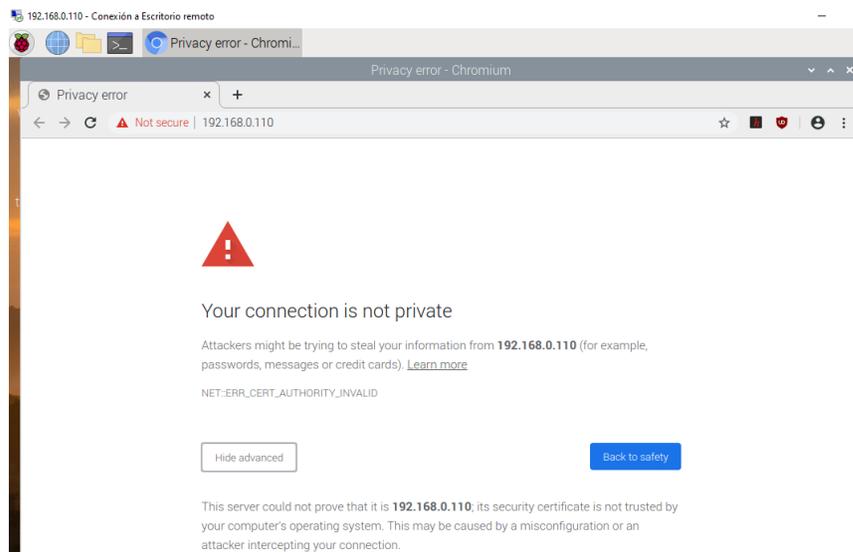
Luego nos aparece un mensaje de advertencia a lo que marcamos la casilla No volver a preguntarme sobre conexiones a este equipo y luego damos clic en la opción **Sí**.

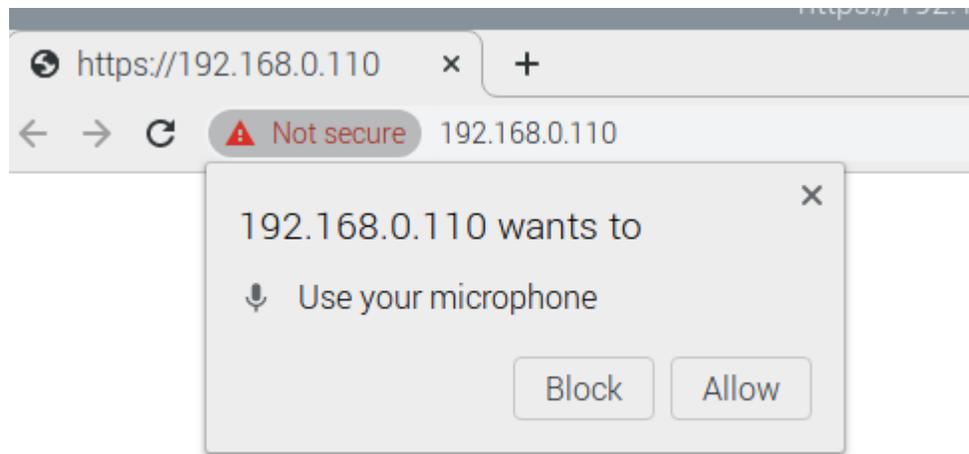


Luego se nos abrirá la pantalla de la conexión remota en la cual nos pedirá usuario y contraseña en la cual utilizaremos lo siguientes datos y daremos clic en Ok

- Usuario: pi
- Contraseña: Raspberry

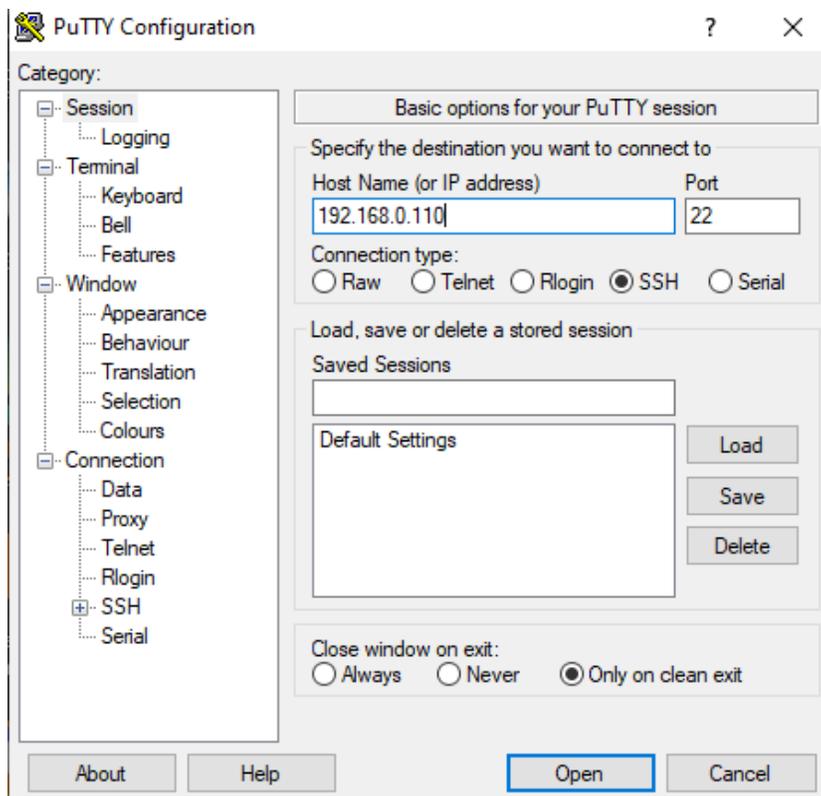
Para lo cual abrimos el buscador de internet, la primera vez de inicio de la Raspberry se deberá autorizar el ingreso a este sitio y permitir el acceso al micrófono en la opción ALLOW.



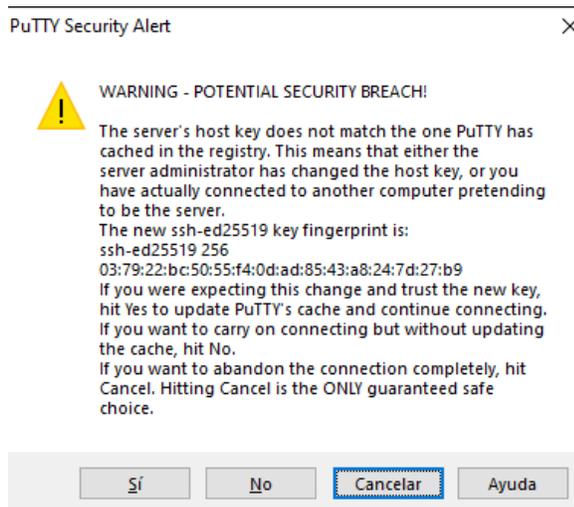


Con esto nuestro sistema está listo para recibir los comandos de voz y procesarlos.

Procedemos a abrir nuestro programa PuTTY y colocamos la dirección de nuestra Raspberry seleccionamos la opción SSH y damos clic en Open



Nos muestra este mensaje a lo que damos clic en **si**



Se nos abre la siguiente ventana en la cual nos pide usuario y contraseña

- Usuario: pi
- Contraseña: raspberry

```
pi@raspberrypi: ~
login as: pi
pi@192.168.0.110's password:
Linux raspberrypi 4.19.97-v7l+ #1294 SMP Thu Jan 30 13:21:14 GMT 2020 armv7l

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

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Apr 23 20:17:27 2020

SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set
a new password.

Wi-Fi is currently blocked by rfkill.
Use raspi-config to set the country before use.

pi@raspberrypi:~ $
```

Con esto ya estamos en el entorno de configuración de la Raspberry

Ingresamos en la dirección `cd /var/www/html/` y con el comando `ls` nos muestra cada uno de los programas para encender y apagar focos, abrir cerraduras, para verificar el funcionamiento podemos ejecutarlos de forma manual con el comando `sudo python + el programa` ejemplo `sudo python ecasa.py` en este caso al ejecutar el comando se va encender los focos de la casa.

```
pi@raspberrypi:~ $ cd /var/www/html/
pi@raspberrypi:/var/www/html $ ls
acasa.py  afoco3.py  auxilio.py  ecasa.py  efoco3.py  index.html
afocol.py alarmarobo.py cerradura1.py efocol.py index2.html info.php
afoco2.py alexa.php  cerradura2.py efoco2.py indexcp.html L.wav
```

Para ejecutar el programa principal denominado `acceso.py` este se encuentra en el directorio `cd Documents/tesis/acceso/` una vez dentro del directorio se procede a introducir el comando `sudo python3 acceso.py`

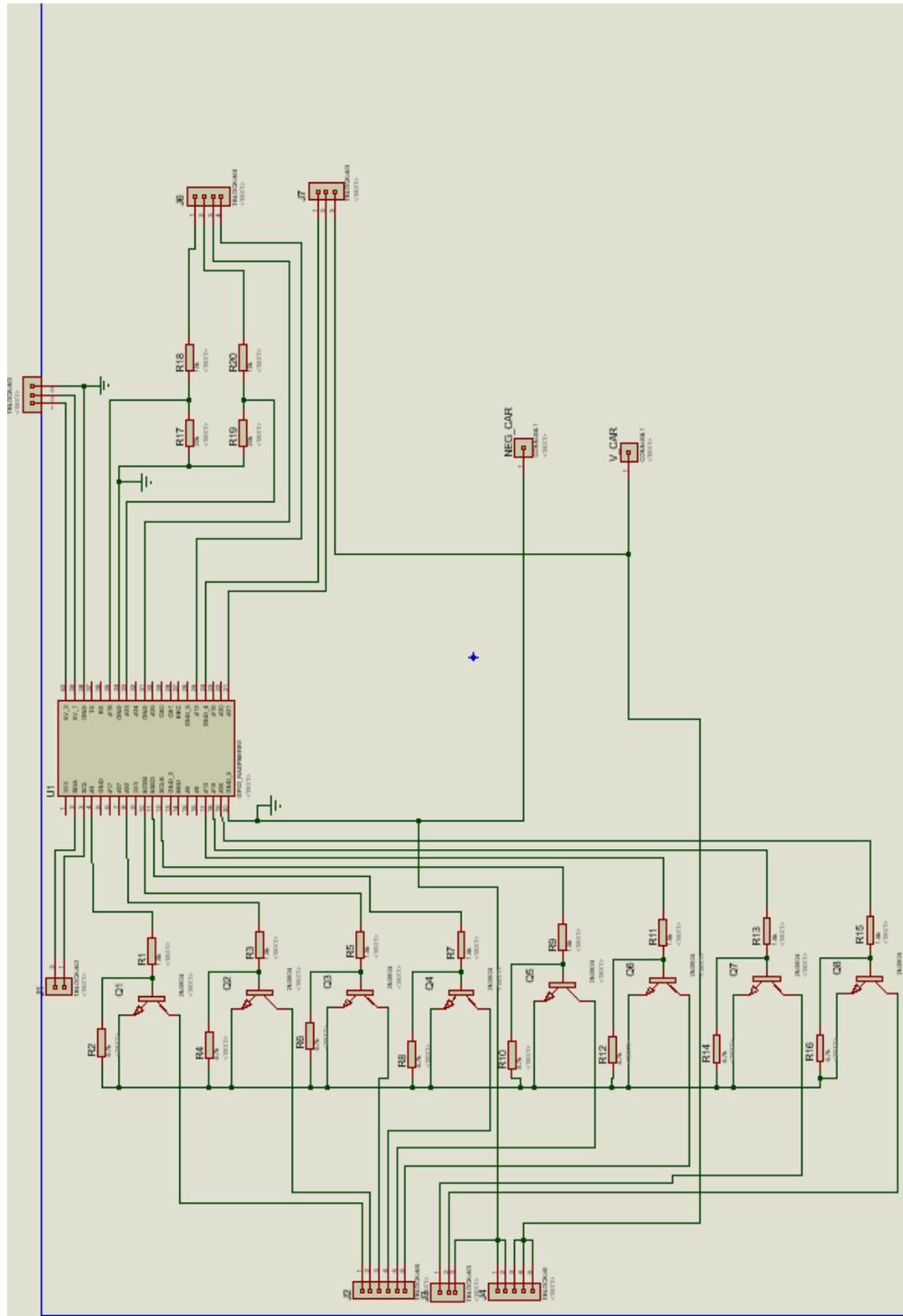
```
pi@raspberrypi:~ $ cd Documents/tesis/acceso/
pi@raspberrypi:~/Documents/tesis/acceso $ ls
acceso.py  act_des_sis  codigos  __pycache__  wiegand.py
pi@raspberrypi:~/Documents/tesis/acceso $ sudo python3 acceso.py
```

Corriendo este programa nos muestra el estado de la alarma para verificación en la cual nos indica si el sistema esta alarmado o sin alarma y la detección en caso de intruso.



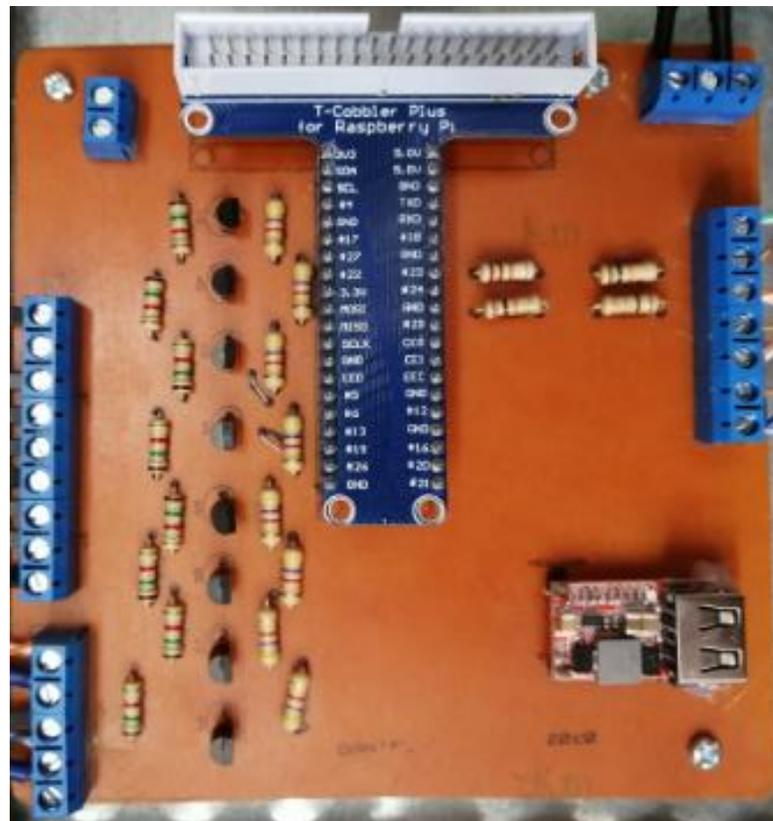
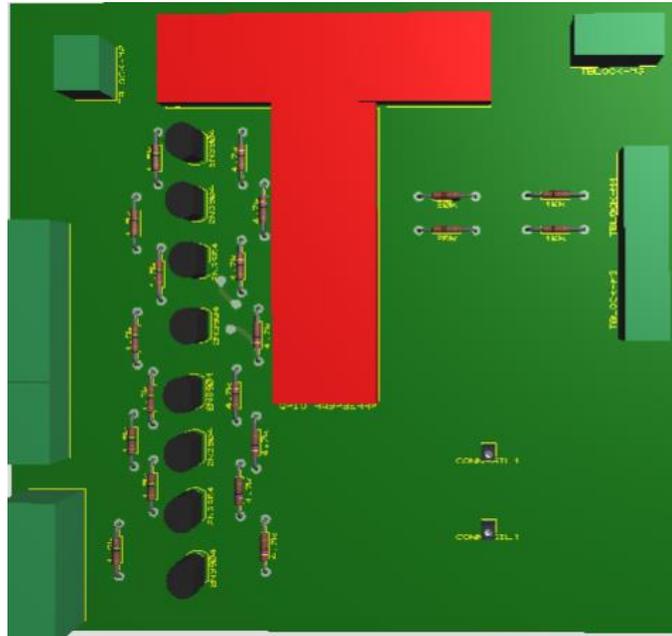
### ANEXO 3

Circuito electrónico para control para el disparo de relés, conexión de lectora, sensores PIR y contacto magnético.



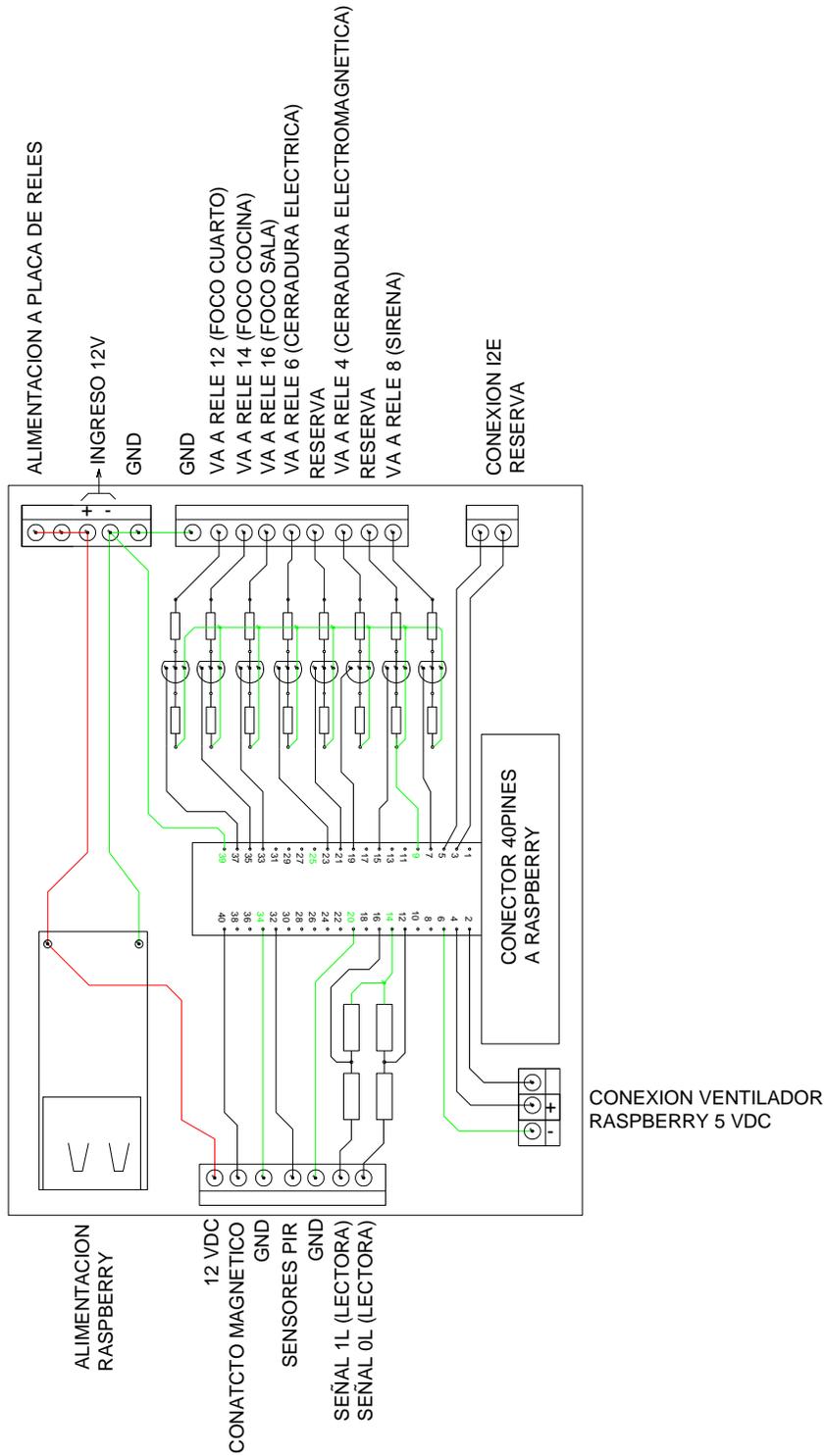
## ANEXO 4

### PLACA PCB PARA EL DISPARO DE RELÉS A TRAVÉS DE PINES GPIO



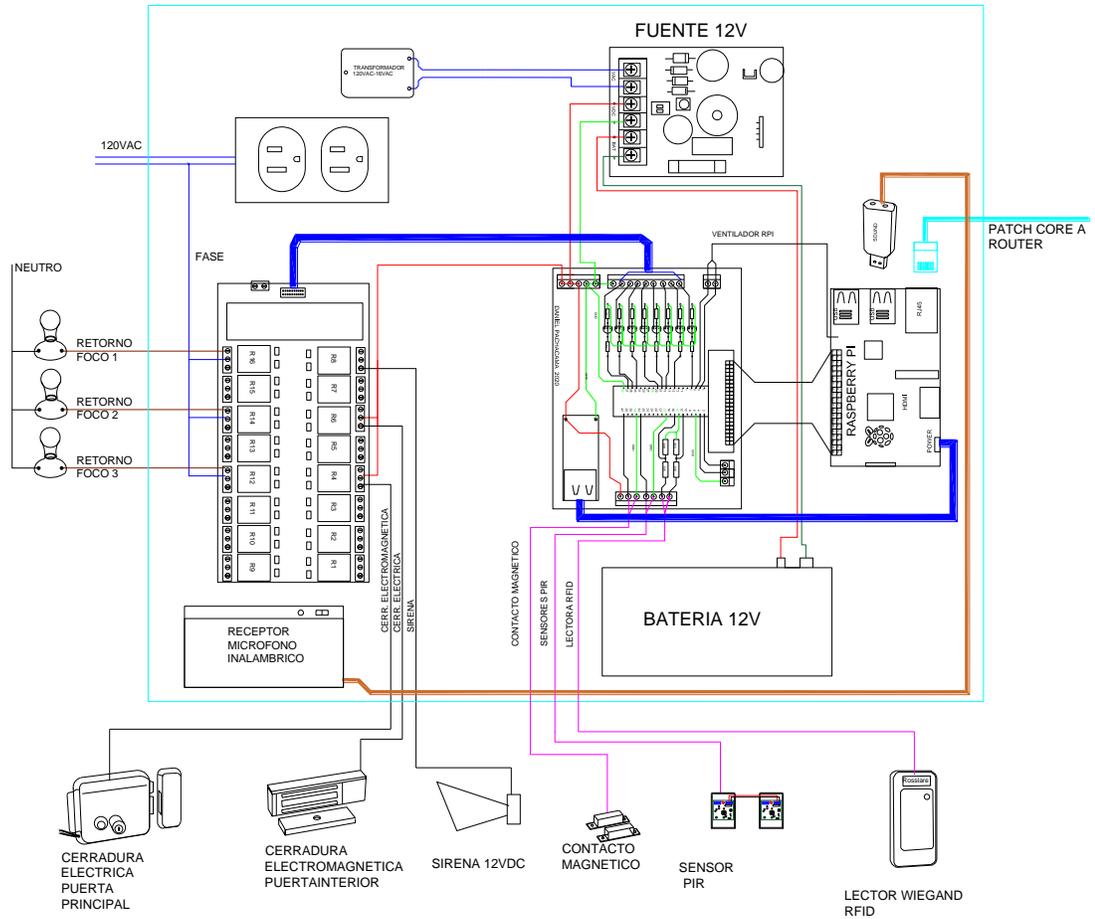
# ANEXO 5

## ESQUEMA DE CONEXIÓN PLACA PCB



# ANEXO 6

## ESQUEMA GENERAL DE CONEXIÓN EN TABLERO



## ANEXO 7

**Algoritmo para importar librería anyyang para reconocer comandos de voz, y clave de acceso para desactivación del sistema de alarma.**

```
<html>
<html>
<head>
  <meta charset="utf-8">
  <script src="//cdnjs.cloudflare.com/ajax/libs/anyyang/2.6.1/anyyang.min.js"></sc$
</head>
<body>
<script>
function openPage(){
  window.open('https://192.168.0.110/index2.html','_self');
}

if (anyyang) {
  anyyang.setLanguage('es-MX');
  var commands = {
    '354' : openPage,
    'tres cinco cuatro' : openPage,
  };
  anyyang.addCommands(commands);
  anyyang.start();
}
else{
  alert('ERROR: servidor no cargado');
}
</script>
</body>
</html>
```

## ANEXO 8

**Algoritmo recoge los comandos de voz, los compara con nuestra librería y ejecuta los programas python.**

```
<html>
<html>
<head>
<meta charset="utf-8">
<script src="//cdnjs.cloudflare.com/ajax/libs/annyang/2.6.1/annyang.min.js"></script>
</head>
<body>
<script>
function openPage(voz){
    window.open('alexa.php?voz='+voz,'_blank');
}
if (annyang) {
    annyang.setLanguage('es-MX');
    var commands = {
        'Lili *voz' : openPage,
    };
    annyang.addCommands(commands);
    annyang.start();
}
else{
    alert('ERROR: servidor no cargado');
}
</script>
</body>
</html>
```

## ANEXO 9

### Código para activar y desactivar el sistema de alarma y lectura de tarjeta wiegand

```
import wiegand
import time
import RPi.GPIO as GPIO
from wiegand import Wiegand
led=4
puerta=11
puls=21
pir=12
GPIO.setmode(GPIO.BCM)
GPIO.setup(led,GPIO.OUT)
GPIO.setup(puerta,GPIO.OUT)
GPIO.setup(puls,GPIO.IN,pull_up_down=GPIO.PUD_UP)
GPIO.setup(pir,GPIO.IN,pull_up_down=GPIO.PUD_UP)
cont=0
f=open("/home/pi/Documents/tesis/acceso/codigos/cod.txt","w")
f.write("0")
a=Wiegand("Wiegand1", 23 , 18, 2)
alarma=0

def monitor_alarm():
    estado=GPIO.input(puls)
    if estado==1 or pir==1:
        print ("alerta intruso")
        #encender sirena
        alarma=2
        GPIO.output(led,1)
    else:
        print ("alerta sin intruso")
        #apagar sirena
        alarma=5

    time.sleep(0.1)

    return alarma

def desactivar_alarm():
    f=open("/home/pi/Documents/tesis/acceso/codigos/desactivar.txt","r")
    c1=f.read()
    f.close()
    return c1
```

```

def activar_alarm():
    f=open("/home/pi/Documents/tesis/acceso/codigos/desactivar.txt","r")
    c1=f.read()
    f.close()
    return c1

while True:
    if activar_alarm()=="3174624":
        monitor_alarm()
    if desactivar_alarm()=="5025842":
        #descativar gpio
        print("sistema CASA ACTIVADA")

    a.binaryToIntWholeValue
    time.sleep(0.6)
    #print ("sistema en espera")
    f=open("/home/pi/Documents/tesis/acceso/codigos/cod.txt","r")
    c=f.read()
    f.close()
    if c=="5025842":
        print ("access activado")
        GPIO.output(puerta,1)
        time.sleep(0.1)
        GPIO.output(puerta,0)
        print ("esperando comando voz")
        time.sleep(20)
        f=open("/home/pi/Documents/tesis/acceso/codigos/cod.txt","w")
        f.write("0")
        f=open("/home/pi/Documents/tesis/acceso/codigos/cod.txt","w")
        f.write("0")

    else:
        print ("access desactivado")
        GPIO.output(puerta,0)

```

## ANEXO 10

### Librería de comandos, el algoritmo direcciona y ejecuta los scripts de python (alexa.php)

```
<html>
<body>
<?php
$valor = $_REQUEST['voz'];
// FOCO SALA
if($valor=='Encender foco sala' || $valor=='encender foco sala' || $valor=='Enciende foco
sala' || $valor=='enciende foco sala')
{
    exec("sudo python /var/www/html/efoco1.py");
}
if($valor=='Apaga foco sala' || $valor=='Apagar foco sala' || $valor=='apagar foco sala' ||
$valor=='apagar foco sala')
{
    exec("sudo python /var/www/html/afoco1.py");
}
// FOCO COCINA
if($valor=='Encender foco cocina' || $valor=='encender foco cocina' || $valor=='Enciende
foco cocina' || $valor=='enciende foco cocina')
{
    exec("sudo python /var/www/html/efoco2.py");
}
if($valor=='Apaga foco cocina' || $valor=='Apagar foco cocina' || $valor=='apagar foco
cocina' || $valor=='apagar foco cocina')
{
    exec("sudo python /var/www/html/afoco2.py");
}
// FOCO CUARTO
if($valor=='Encender foco cuarto' || $valor=='encender foco cuarto' || $valor=='Enciende
foco cuarto' || $valor=='enciende foco cuarto')
{
    exec("sudo python /var/www/html/efoco3.py");
}
if($valor=='Apaga foco cuarto' || $valor=='Apagar foco cuarto' || $valor=='apagar foco
cuarto' || $valor=='apagar foco cuarto')
{
    exec("sudo python /var/www/html/afoco3.py");
}
// APAGAR CASA
if($valor=='Apagar casa' || $valor=='apagar casa')
{
    exec("sudo python /var/www/html/acasa.py");
}
// ENCENDER CASA
if($valor=='Encender casa' || $valor=='encender casa')
```

```

{
  exec("sudo python /var/www/html/ecasa.py");
}
// APAGAR ALARMA EN CASO ROBO
if($valor=='354' || $valor=='tres cinco cuatro')
{
  exec("sudo python /var/www/html/alarमारobo.py");
}
// ALARMA EN CASO ROBO
if($valor=='Auxilio' || $valor=='auxilio')
{
  exec("sudo python /var/www/html/auxilio.py");
}
// CERRADURA

if($valor=='abrir puerta cocina' || $valor=='abre puerta 2' || $valor=='abre puertas 2' ||
$valor=='Abrir puerta cocina' || $valor=='Abre puerta 2' || $valor=='Abre pue$
{
  exec("sudo python /var/www/html/cerradura2.py");
}

//puerta de ingreso,salida
if($valor=='abrir puerta 1' || $valor=='abre puerta 1' || $valor=='Abre puerta 1' ||
$valor=='Abrir puerta 1' || $valor=='Abre puerta uno')
{
  exec("sudo python /var/www/html/cerradura1.py");
}
// ALARMA

if($valor=='Desactivar alarma' || $valor=='Desactiva alarma' )
{
  exec("sudo python3 /home/pi/Documents/tesis/acceso/act_des_sis/offalarm.py");
}

if($valor=='Activar alarma' || $valor=='activa alarma' )
{
  exec("sudo python3 /home/pi/Documents/tesis/acceso/act_des_sis/onalarm.py");
}

echo "<script>>window.close();</script>";
?>
</body>
</html>

```

## ANEXO 11

### Programas para activación de focos, cerraduras y alarmas.

#### Encender sala (efoco1.py)

---

```
import time
import RPi.GPIO as gpio
import os
led=13

gpio.setmode(gpio.BCM)
gpio.setup(13,gpio.OUT)
gpio.output(led,1)

os.system(' pico2wave -l es-ES -w L.wav "encendiendo sala " && aplay L.wav')
```

#### Encender cocina (efoco2.py)

---

```
import time
import RPi.GPIO as gpio
import os
led=19

gpio.setmode(gpio.BCM)
gpio.setup(5,gpio.OUT)
gpio.output(led,1)

os.system(' pico2wave -l es-ES -w L.wav "encendiendo cocina" && aplay L.wav')
```

#### Encender dormitorio (efoco3.py)

---

```
import time
import RPi.GPIO as gpio
import os
led=26

gpio.setmode(gpio.BCM)
gpio.setup(26,gpio.OUT)
gpio.output(led,1)

os.system(' pico2wave -l es-ES -w L.wav "encendiendo dormitorio " && aplay L.wav')
```

### **Apagar sala (efoco1.py)**

---

```
import time
import RPi.GPIO as gpio
import os
led=13
gpio.setwarnings(False)
gpio.setmode(gpio.BCM)
gpio.setup(13,gpio.OUT)
gpio.output(led,0)

os.system(' pico2wave -l es-ES -w L.wav "apagado sala " && aplay L.wav')
```

### **Apagar cocina (efoco2.py)**

---

```
import time
import RPi.GPIO as gpio
import os
led=5

gpio.setwarnings(False)
gpio.setmode(gpio.BCM)
gpio.setup(5,gpio.OUT)
gpio.output(led,0)
os.system(' pico2wave -l es-ES -w L.wav "apagando cocina" && aplay L.wav')
```

### **Apagar dormitorio (efoco3.py)**

---

```
import time
import RPi.GPIO as gpio
import os
led=6

gpio.setwarnings(False)
gpio.setmode(gpio.BCM)
gpio.setup(6,gpio.OUT)
gpio.output(led,0)

os.system(' pico2wave -l es-ES -w L.wav "apagando dormitorio " && aplay L.wav')
```

## **Encender los focos de los tres ambientes (ecasa.py)**

---

```
import time
import RPi.GPIO as gpio
import os
sala=13
cocina=19
cuarto=26

gpio.setmode(gpio.BCM)
gpio.setup(sala,gpio.OUT)
gpio.setup(cocina,gpio.OUT)
gpio.setup(cuarto,gpio.OUT)
gpio.output(sala,1)
gpio.output(cocina,1)
gpio.output(cuarto,1)

os.system(' pico2wave -l es-ES -w L.wav "Focos de casa encendidos" && aplay L.wav')
```

## **Apagar los focos de los tres ambientes (ecasa.py)**

---

```
import time
import RPi.GPIO as gpio
import os
sala=13
cocina=19
cuarto=26

gpio.setmode(gpio.BCM)
gpio.setup(sala,gpio.OUT)
gpio.setup(cocina,gpio.OUT)
gpio.setup(cuarto,gpio.OUT)

gpio.output(sala,1)
gpio.output(cocina,1)
gpio.output(cuarto,1)

os.system(' pico2wave -l es-ES -w L.wav "Focos de casa encendidos" && aplay L.wav')
```

---

### **Encender sirena en caso de emergencia (auxilio.py)**

---

```
import time
import RPi.GPIO as gpio
import os
led=4

gpio.setmode(gpio.BCM)
gpio.setup(led,gpio.OUT)
gpio.output(led,1)

os.system(' pico2wave -l es-ES -w L.wav "Activando sirena " && aplay L.wav')
```

### **Apagar sirena (alarmrobo.py)**

---

```
import time
import RPi.GPIO as gpio
import os
led=4
gpio.setmode(gpio.BCM)
gpio.setup(led,gpio.OUT)
gpio.output(led,0)

os.system(' pico2wave -l es-ES -w L.wav "Apagando sirena " && aplay L.wav')
```

### **Abrir cerradura 1 (cerradura1.py)**

---

```
import time
import RPi.GPIO as gpio
import os
led=11

gpio.setmode(gpio.BCM)
gpio.setup(led,gpio.OUT)
gpio.output(led,1)
time.sleep(0.5)
gpio.output(led,0)

os.system(' pico2wave -l es-ES -w L.wav "Abriendo puerta principal" && aplay L.wav')
```

## **Abrir cerradura 2 (cerradura2.py)**

---

```
import time
import RPi.GPIO as gpio
import os
led=10
```

```
gpio.setmode(gpio.BCM)
gpio.setup(led,gpio.OUT)
gpio.output(led,1)
time.sleep(0.5)
gpio.output(led,0)
```

```
os.system(' pico2wave -l es-ES -w L.wav "Abriendo puerta principal" && aplay L.wav')
```