



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E
INDUSTRIAL
CARRERA DE TELECOMUNICACIONES

Tema:

SISTEMA DOMÓTICO PARA PERSONAS CON DISCAPACIDAD
VISUAL MEDIANTE INTELIGENCIA ARTIFICIAL Y
RECONOCIMIENTO DE VOZ

Trabajo de titulación modalidad Proyecto de Investigación, presentado previo a la
obtención del título de Ingeniero en Telecomunicaciones

ÁREA: Electrónica

LÍNEA DE INVESTIGACIÓN: Tecnología de la Información y Sistemas de
control

AUTOR: Dennis Andrés Lascano Solís

TUTOR: Dr. Freddy Geovanny Benalcázar Palacios

Ambato – Ecuador

Agosto - 2023

APROBACIÓN DEL TUTOR

En calidad de tutor del trabajo de titulación con el tema: SISTEMA DOMÓTICO PARA PERSONAS CON DISCAPACIDAD VISUAL MEDIANTE INTELIGENCIA ARTIFICIAL Y RECONOCIMIENTO DE VOZ, desarrollado bajo la modalidad Proyecto de Investigación por el señor Dennis Andrés Lascano Solís, estudiante de la Carrera de Telecomunicaciones, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, me permito indicar que el estudiante ha sido tutorado durante todo el desarrollo del trabajo hasta su conclusión, de acuerdo a lo dispuesto en el Artículo 17 del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato y el numeral 6.3 del instructivo del reglamento referido.

Ambato, agosto 2023.

Dr. Freddy Geovanny Benalcázar Palacios

TUTOR

AUTORÍA

El presente trabajo de titulación titulado: SISTEMA DOMÓTICO PARA PERSONAS CON DISCAPACIDAD VISUAL MEDIANTE INTELIGENCIA ARTIFICIAL Y RECONOCIMIENTO DE VOZ es absolutamente original, auténtico y personal y ha observado los preceptos establecidos en la Disposición General Quinta del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato. En tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, agosto 2023



Dennis Andrés Lascano Solís

C.C. 180534159-9

AUTOR

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato para que reproduzca total o parcialmente este trabajo de titulación dentro de las regulaciones legales e institucionales correspondientes. Además, cedo todos mis derechos de autor a favor de la institución con el propósito de su difusión pública, por lo tanto, autorizo su publicación en el repositorio virtual institucional como un documento disponible para la lectura y uso con fines académicos e investigativos de acuerdo con la Disposición General Cuarta del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato.

Ambato, agosto 2023



Dennis Andrés Lascano Solís

C.C. 180534159-9

AUTOR

APROBACIÓN DEL TRIBUNAL DE GRADO

En calidad de par calificador del informe final del trabajo de titulación presentado por el señor Dennis Andrés Lascano Solís, estudiante de la Carrera de Telecomunicaciones, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, bajo la Modalidad Proyecto de Investigación titulado SISTEMA DOMÓTICO PARA PERSONAS CON DISCAPACIDAD VISUAL MEDIANTE INTELIGENCIA ARTIFICIAL Y RECONOCIMIENTO DE VOZ, nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo al Artículo 19 del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato y el numeral 6.4 del instructivo del reglamento referido. Para cuya constancia suscribimos, conjuntamente con la señora Presidente del Tribunal.

Ambato, agosto 2023.

Ing. Elsa Pilar Urrutia Urrutia Mg.

PRESIDENTE DEL TRIBUNAL

Ing. Santiago Mauricio Altamirano
Meléndez Mg.

PROFESOR CALIFICADOR

Ing. Mario Geovanny García
Carrillo Mg.

PROFESOR CALIFICADOR

DEDICATORIA

Este proyecto va dedicado especialmente para aquellas personas que estuvieron durante todo el transcurso de mi vida universitaria, mi madre Mariana Solís y mi padre Byron Lascano por ser mi sustento, mi aliento y mi inspiración ya que siempre han estado apoyándome incondicionalmente, a mi hermano Jorge Lascano y a toda mi familia y amigos que me han apoyado en cada momento de mis estudios, todos han sido mi motivación de seguir adelante y nunca dejar a un lado mis objetivos para así poder culminar mis estudios.

Andy

AGRADECIMIENTO

Al ver el resultado logrado con este ambicioso proyecto solamente se me ocurre decir ¡Gracias!, gracias a Dios por darme salud y vida para seguir de pie.

A la facultad de Ingeniería en Sistemas, Electrónica e Industrial por abrirme sus puertas, a los docentes que fueron parte de este grandioso proceso compartiendo sus conocimientos y su amistad, al ingeniero Freddy Benalcazar por ser mi guía en el desarrollo de este proyecto.

Agradecimientos a todos compañeros de universidad y trabajo por todos los momentos buenos y malos que hemos compartido, gracias por su amistad y apoyo.

Andy

ÍNDICE GENERAL DE CONTENIDOS

APROBACIÓN DEL TUTOR.....	ii
AUTORÍA.....	iii
DERECHOS DE AUTOR	iv
APROBACIÓN DEL TRIBUNAL DE GRADO	v
DEDICATORIA	vi
AGRADECIMIENTO	vii
ÍNDICE GENERAL DE CONTENIDOS.....	viii
ÍNDICE DE TABLAS	xi
ÍNDICE DE FIGURAS.....	xii
RESUMEN EJECUTIVO	xv
ABSTRACT.....	xvi
CAPÍTULO I.....	1
MARCO TEÓRICO.....	1
1.1. Tema de investigación.....	1
1.1.1. Planteamiento del problema.....	1
1.2. Antecedentes investigativos	3
1.3. Fundamentación teórica	5
1.3.1. Sistema Domótico	5
1.3.2. Inteligencia Artificial	12
1.3.3. Internet de las Cosas (IoT).....	18
1.3.4. Discapacidad	19
1.4. Objetivos	21
1.4.1. Objetivo general.....	21

1.4.2.	Objetivos específicos	21
CAPÍTULO II		22
METODOLOGÍA		22
2.1.	Materiales	22
2.2.	Métodos	23
2.2.1.	Modalidad de la investigación	23
2.2.2.	Población y muestra	24
2.2.3.	Recolección de información	24
2.2.4.	Procesamiento y análisis de datos.....	25
CAPÍTULO III		26
RESULTADOS Y DISCUSIÓN		26
3.1.	Análisis y discusión de resultados.....	26
3.2.	Desarrollo de la propuesta	26
3.2.1.	Análisis de la factibilidad.....	26
3.2.2.	Resultados y Tabulación de datos.....	27
3.2.3.	Análisis e interpretación de datos	28
3.3.	Requerimientos.....	36
3.3.1.	Requerimientos de las personas con discapacidad visual	36
3.3.2.	Requerimientos del sistema domótico	37
3.4.	Análisis del sistema domótico	37
3.5.	Arquitectura del sistema domótico.....	37
3.5.1.	Tipos de mando.....	38
3.6.	Diseño del prototipo	40
3.6.1.	Diagrama de bloques.....	40
3.7.	Análisis de las tecnologías móviles.....	41

3.8.	Análisis Técnico de Hardware requerido	42
3.8.1.	Selección de Hardware	42
3.9.	Diagrama de bloques del sistema domótico	49
3.10.	Diseño del circuito electrónico.....	50
3.11.	Software.....	53
3.11.1.	Reconocimiento de voz.....	53
3.11.2.	Instalación de la imagen Raspbian para Raspberry Pi 4.....	53
3.11.3.	Instalar las librerías	56
3.12.	Desarrollo de la programación del sistema domótico	62
3.12.1.	Programación del sistema para el reconocimiento de voz.....	62
3.12.2.	Implementación del Prototipo del sistema domótico.....	67
3.12.3.	Elaboración del case para la placa del circuito	71
3.12.4.	Elaboración del prototipo de vivienda	72
3.12.5.	Colocación de elementos electrónicos	74
3.13.	Pruebas de funcionamiento	79
3.13.1.	Reconocimiento de voz.....	79
3.14.	Presupuesto del prototipo	81
CAPÍTULO IV.....		84
5.1.	Conclusiones	84
5.2.	Recomendaciones	85
REFERENCIAS BIBLIOGRÁFICAS.....		86
ANEXOS		91

ÍNDICE DE TABLAS

Tabla 1.- Población de estudio tomada de la Escuela de Educación Básica Julius Doephner.	24
Tabla 2.- Tabulación de datos.	27
Tabla 3.- Tabulación de datos, pregunta 1.	28
Tabla 4.- Tabulación de datos, pregunta 2.	30
Tabla 5.- Tabulación de datos, pregunta 3.	31
Tabla 6.- Tabulación de datos, pregunta 4.	32
Tabla 7.- Tabulación de datos, pregunta 5.	34
Tabla 8.- Tipos de arquitecturas de sistemas domóticas.	38
Tabla 9.- Tipos de mandos para el control domótico.	39
Tabla 10.- Comparativa entre tecnologías móviles para el control del sistema domótico.	42
Tabla 11.- Características técnicas del micrófono para el reconocimiento de voz. ...	43
Tabla 12.- Cuadro comparativo de Raspberry PI.	44
Tabla 13.- Características técnicas de servomotores.	45
Tabla 14.- Características técnicas de ventiladores.	47
Tabla 15.- Características técnicas de los relés.	48
Tabla 16.- Comandos de voz.	63
Tabla 17.- Prueba de Funcionamiento de Reconocimiento de voz.	79
Tabla 18.- Resultados de las pruebas realizados para el reconocimiento de voz.	80
Tabla 19.- Presupuesto.	83

ÍNDICE DE FIGURAS

Figura 1.- Sistemas inteligentes en una vivienda	6
Figura 2.- Arquitectura centralizada	9
Figura 3.- Arquitectura descentralizada	10
Figura 4.- Arquitectura semicentralizado.....	11
Figura 5.- Inteligencia Artificial y sus categorías fundamentales.....	15
Figura 6.- Tabulación de datos, pregunta 1.....	29
Figura 7.- Tabulación de datos, pregunta 2.....	30
Figura 8.- Tabulación de datos, pregunta 3.....	31
Figura 9.- Tabulación de datos, pregunta 4.....	33
Figura 10.- Tabulación de datos, pregunta 5.....	34
Figura 11.- Diagrama de bloques del sistema domótico.....	40
Figura 12.- Diagrama de bloques para el sistema domótico.....	49
Figura 13.- Circuito electrónico para el control domótico.....	50
Figura 14.- Sistemas de control de iluminación.....	51
Figura 15.- Circuito para el control de servomotores.	51
Figura 16.- Circuito de sistema de ventilación.....	52
Figura 17.- Circuito de control de puertas de la vivienda.....	52
Figura 18.- Instalación de la imagen de Raspbian.	53
Figura 19.- Interfaz de Pi Imager.	54
Figura 20.- Flasheando la tarjeta SD.....	54
Figura 21.- Configuración de la Raspberry.....	55
Figura 22.- Configuración de sistema.	55
Figura 23.- Actualización del sistema operativo.....	56
Figura 24.- Instalación de la librería pyAudio.	57

Figura 25.- Instalación de la librería SpeechRecognition.	57
Figura 26.- Instalación de la librería RPi.GPIO.	58
Figura 27.- Comprobación de la instalación de librerías.	58
Figura 28.- Librerías necesarias para el entrenamiento de voz.	59
Figura 29.- Carga de archivos para el entrenamiento.	59
Figura 30.- Comprobación de audios.	60
Figura 31.- Puntuaciones del sistema.	60
Figura 32.- Testeo de la Red Neuronal.	61
Figura 33.- Comprobación de la Red Neuronal al ingresar un comando.	61
Figura 34.- Lazo principal del diagrama de flujo del Reconocimiento de Voz.	65
Figura 35.- Diagrama de flujo del Reconocimiento de Voz.	66
Figura 36.- Diseño de la placa PCB.	67
Figura 37.- Parte posterior de la placa PCB.	68
Figura 38.- Parte posterior de la placa PCB.	69
Figura 39.- Diseño de la placa PCB.	70
Figura 40.- Conexión de Circuitos del Sistema Domótico.	70
Figura 41.- Diseño del case para la placa del circuito.	71
Figura 42.- Case para la placa.	71
Figura 43.- Case para la placa.	72
Figura 44.- Plano 2D del Prototipo de la Vivienda.	72
Figura 45.- Plano 3D del prototipo de la vivienda.	73
Figura 46.- Armado del prototipo de la vivienda.	73
Figura 47.- Instalación de las luces leds de la vivienda.	74
Figura 48.- Instalación del servomotor para control de persiana de la sala.	75
Figura 49.- Instalación del servomotor para control de persiana del cuarto.	75

Figura 50.- Instalación del ventilador de la vivienda.	76
Figura 51.- Instalación de la cocineta de la vivienda.	76
Figura 52.- Instalación del sistema de control de acceso.	77
Figura 53.- Instalación completa de todos los elementos electrónicos.	77
Figura 54.- Prototipo del Sistema Domótico funcional.	78
Figura 55.- Prototipo del Sistema Domótico funcional.	78

RESUMEN EJECUTIVO

El presente proyecto de Investigación tiene como objetivo implementar un prototipo de un sistema domótico para personas con discapacidad visual mediante Inteligencia Artificial y reconocimiento de voz, para mejorar la calidad de vida de las personas con discapacidad visual dentro de su vivienda, para cumplir con dicho propósito se llevó a cabo un análisis del estado actual de las viviendas de dichas personas, mediante una encuesta dirigida a las personas con discapacidad visual registradas en la “Escuela de Educación Básica Especializada Cardenal Julius Doepfner” en el Cantón Ambato, donde la falta de accesibilidad, control de acceso, control de iluminación, control de artefactos electrónicos y seguridad en las viviendas fueron notorias, de esta manera se establece la utilización de los beneficios de la domótica y se plantea los principales requerimientos que necesitan para la gestión doméstica, hardware y software libre que ayudaran en la implementación del prototipo para que se reduzca el costo.

El proyecto de investigación desarrollado está compuesto por un sistema de reconocimiento de voz con Inteligencia Artificial mediante el uso del ordenador Raspberry PI 4, en el cual se desarrolló el sistema de aprendizaje automático en TensorFlow que permite el entrenamiento de redes neuronales para el reconocimiento de una voz específica para el control general del sistema, esto ayudará a solucionar los problemas con mejor calidad y con costos bajos sin necesidad de modificar la infraestructura de las viviendas que se deseen controlar.

Palabras Clave: Raspberry PI, TensorFlow, reconocimiento de voz, inteligencia artificial, control, seguridad, acceso.

ABSTRACT

This research project aims to implement a prototype of a home automation system for visually impaired people through artificial intelligence and voice recognition, to improve the quality of life of visually impaired people in their homes, to fulfill this purpose was carried out an analysis of the current state of the homes of such people, through a survey of visually impaired people registered in the "School of Specialized Basic Education Cardinal Julius Doephner" in the Canton Ambato, where the lack of accessibility, access control, lighting control, control of electronic devices and security in the homes were notorious, thus establishing the use of the benefits of home automation and the main requirements they need for home management, hardware and free software that will help in the implementation of the prototype to reduce the cost.

The research project developed is composed of a voice recognition system with artificial intelligence through the use of the Raspberry PI 4 computer, in which the automatic learning system was developed in TensorFlow that allows the training of neural networks for the recognition of a specific voice for the general control of the system, this will help to solve problems with better quality and low costs without the need to modify the infrastructure of the houses to be controlled.

Keywords: Raspberry PI, TensorFlow, voice recognition, artificial intelligence, control, security, access.

CAPÍTULO I

MARCO TEÓRICO

1.1. Tema de investigación

SISTEMA DOMÓTICO PARA PERSONAS CON DISCAPACIDAD VISUAL MEDIANTE INTELIGENCIA ARTIFICIAL Y RECONOCIMIENTO DE VOZ

1.1.1. Planteamiento del problema

A nivel mundial la Organización Mundial de la Salud (OMS), en un informe sobre la visión, al menos 2200 millones tienen una discapacidad visual total o parcial. El informe establece que todas las personas con discapacidad visual severa que no se puede tratar pueden llevar una vida independiente si tienen acceso a opciones como lupas ópticas, lectura Braille y sistemas de automatización del hogar. En un 50% de los casos se pudo haber tratado para ayudarlos con una forma de vida digna y adecuada al entorno tecnológico en el que estamos inmiscuido toda la sociedad [1]. Hoy en día se puede apreciar que a nivel mundial las personas con discapacidad visual total, en muchas ocasiones se les dificulta adaptarse a un entorno ocasionando muchos factores, tanto como la discriminación por parte de la sociedad o hasta de los propios familiares quienes en muchos casos los llevan a fundaciones e incluso abandonarlos ya sea por falta de recursos para el cuidado de ellos [2].

A nivel nacional Ecuador que es un país subdesarrollado con recursos limitados indican no cuentan con acceso a dispositivos y tecnologías de apoyo aquellas personas con discapacidad como son: silla de ruedas, prótesis, dispositivos de ayuda visual, equipos, entre otros. Según el CONADIS 455289 de las cuales el 11, 47% representan a personas con discapacidad visual severa. En lo que respecta a la Sierra ecuatoriana se registran 1371 personas con discapacidad visual, divididos de acuerdo con su grado de discapacidad. Las personas con deficiencia visual grave (75% a 84%) y ceguera (85% a 100%) con un total de 647, de sexo femenino 318 y masculino 329. Las personas con deficiencia visual grave representan la mayoría con un 81,14% y desde

luego es más que evidente una falta de abordaje de esta problemática; razón por la cual debe ser abordada de la manera más pronta posible [3].

Una incorporación de sistemas de automatización en el hogar, puede potenciar la realización de actividades de la vida diaria tanto las básicas como las que requieren de mayor autonomía personal, que pueden resultar difíciles para personas con algún tipo de discapacidad. Aun así, esta incorporación se está haciendo de manera muy lenta y con distintas repercusiones entre los diferentes grupos de edad, fundamentalmente dadas por un “rechazo tecnológico” [4]. Ese rechazo tecnológico viene dado fundamentalmente por personas tradicionales y va ligado a aspectos como: complejidad, miedo a lo desconocido, miedo al aislamiento social y a la pérdida del apoyo emocional, sensación de inseguridad, pérdida de control y el precio elevado.

A nivel local en Ambato según los datos del CONADIS en la ciudad se registran 933 personas con discapacidad visual con un grado de discapacidad oscilante del 85% a 100% siendo el caso más severo con un total de 140 personas. Restringiendo la posibilidad de trasladarse de un lugar a otro, manipulación de objetos para realizar tareas cotidianas en el hogar como la apertura y cierre de puertas y ventanas o encendido y apagado de luces y equipos, entre otros. Lo cual puede convertirse en labores complicadas para personas con limitantes en su movilidad, creando una condición de dependencia haciendo imprescindible la ayuda de otra persona. Debido a que las personas con discapacidad visual no siempre cuentan con recursos económicos para solventar gastos de personas auxiliares que lo cuide y los familiares no pueden estar siempre con ellas por cuestiones de trabajo, esto da lugar a que estén solas y se vean expuestas a sufrir algún accidente agravando su condición.

El presente proyecto tiene la visión de permitir que personas con discapacidad visual puedan realizar tareas cotidianas que antes no podían realizar; esto implica, abrir el seguro de puerta, controlar la temperatura, abrir grifo de ducha o lavaderos. Para su implementación se utilizó Inteligencia Artificial, dispositivos de transmisión y recepción de datos y programación en software libre.

1.2. Antecedentes investigativos

La última década ha visto muchos avances en los sistemas domóticos, al efectuar una investigación bibliográfica en repositorios digitales se logró encontrar proyectos afines al presente tema, con los cuales se pudo generar un soporte en el desarrollo del proyecto de investigación propuesto.

En el año 2022, Pascual Lourdes desarrolló la “Contribución al desarrollo de modelos de control domótico de usuario adultos mayores”, el cual busca explotar las tecnologías de la información y la comunicación (TIC) para ayudar a las personas y, específicamente, cómo diseñar sistemas Ambient Assisting Living (AAL) para monitorear y mejorar la independencia de las personas de avanzada edad. El envejecimiento puede conducir progresivamente de grados bajos a severos de discapacidad. A menudo, las personas mayores tienen múltiples discapacidades, que afectan tanto a los sistemas perceptivos como cognitivos [5].

En el año 2021, Chaparro David en su investigación bajo el tema “Desarrollo de una aplicación móvil para la comunicación de personas con discapacidad mediante la utilización de asistentes virtuales inteligentes de dispositivos IoT”, el mismo que consiste en la implementación de un sistema domótico controlado de forma remota y reconocimiento de voz, destinado principalmente con la discapacidad físico-motora y visual, utilizando una interfaz en LabVIEW y dos tecnologías Local Network Control (LCN) [6]. El internet ha llevado a una automatización del hogar más simple e inteligente, accesible en cualquier momento y en cualquier lugar, lo que permite casas inteligentes. Las casas inteligentes son muy valiosas para las personas con discapacidad ya que ofrecen la oportunidad de lograr la independencia. Esto abarca no solo la usabilidad, sino también el cuidado personal y la seguridad.

En el año 2020, Cahuana Aquino presenta un trabajo de fin de Master TFM bajo el nombre “Sistema domótico con la tecnología Arduino a través de un aplicativo móvil para el control de salud”. Los avances recientes en los sistemas portátiles de control

de la salud permiten una intervención temprana, lo que puede contribuir a aumentar la esperanza de vida: las redes de área corporal y los Health Smart Homes (HSH) incorporan sistemas de salud portátiles y fijos para ayudar a las personas mayores o discapacitadas. Los sistemas de monitorización del control corporal son esenciales para una intervención temprana y para prolongar la esperanza de vida. Los dispositivos inteligentes portátiles pueden monitorear de forma activa y constante los parámetros vitales (redes de área corporal). Sin embargo, una revisión sistemática reciente de hogares inteligentes y tecnologías de monitoreo de la salud en el hogar señala que el nivel de preparación tecnológica para estos sistemas aún es bajo y carece de evidencia sólida de su efectividad como herramienta de prevención para ayudar a la población anciana [7].

En el año 2019, Carvalho María en su investigación denominado como “Arquitectura de servicios B-live para domótica” presenta un sistema diseñado para personas con problemas de motricidad (tetraplégicos, parapléjicos, usuarios de sillas de ruedas) y personas mayores, destinado a ayudar a las personas en tareas comunes, como encender/apagar las luces de la habitación, abrir/cerrar persianas y puertas. Entre las formas naturales de interacción con la tecnología, la voz y la sensación están surgiendo cada vez más. Los autores observaron que la tecnología del habla tiene un gran potencial para facilitar la vida cotidiana, pero fomenta un estilo de vida perezoso, lo que provoca una rápida degeneración de la salud en las personas mayores. Este estudio indica que al diseñar para las personas mayores es crucial promover una forma de vida saludable. La tecnología móvil generalizada juega un papel fundamental en el control remoto de las personas mayores, especialmente cuando los familiares están involucrados en su integración social y estimulación física [8].

En el año 2018, los investigadores David Valencia y Luis Gonzáles realizaron el “Diseño e implementación de un prototipo de robot asistente para personas con discapacidad motriz y adultos mayores, basado en Inteligencia Artificial”, el cual se centra en diseño de software y hardware del prototipo de un robot asistente por medio de algoritmos de visión artificial Fisherfaces, webcam y la librería OpenCV para la detección y reconocimiento facial además se utilizó dispositivos electrónicos como el

Arduino para el control de los movimientos del robot, con la implementación de este proyecto se logró solventar las necesidades básicas que una persona adulta mayor requiere. Las soluciones mejoradas con tecnología diseñadas para ayudar a las personas con discapacidad en casas inteligentes son útiles para cualquiera que viva en condiciones físicas o ambientales difíciles. Por lo tanto, la usabilidad del diseño domótica tiene potencial para beneficiar a la sociedad en su conjunto [9].

1.3. Fundamentación teórica

1.3.1. Sistema Domótico

La domótica es un sistema que permite que los dispositivos se conecten y monitoreen de forma remota a través de Internet. En los últimos años, el concepto ha tenido una fuerte evolución, siendo utilizado actualmente en diversos dominios como casas inteligentes, telemedicina, entornos industriales, etc [10]. Es importante señalar que las tecnologías de red de sensores inalámbricos permiten una interconexión global de dispositivos inteligentes con funcionalidades avanzadas. Una red de domótica inalámbrica, compuesta por sensores y actuadores que comparten recursos y están interconectados entre sí, es la tecnología clave para hacer hogares inteligentes.

Los sistemas domóticos ofrecen una serie de beneficios; agregan seguridad a través del control de electrodomésticos e iluminación, aseguran el hogar a través de cerraduras de puertas automáticas, aumentan la conciencia a través de cámaras de seguridad, aumentan la comodidad a través del ajuste de temperatura, ahorran un tiempo precioso, brindan control y ahorran dinero.



Figura 1.- Sistemas inteligentes en una vivienda.

Fuente: [11].

Elementos de un sistema domótico

Un sistema domótico típico consta de sensores, actuadores y controladores de hardware. Básicamente, hay dos formas diferentes de controlar un sistema domótico, el control centralizado y el descentralizado, pero también es posible alguna combinación. La estructura del sistema determina dónde se ubica la inteligencia, lo que significa si la parte principal del software de control está alojada en un componente o en todos los componentes [12]. A continuación, se enlista una descripción de cada componente de un sistema domótico

Controladores

Se denominan controladores a los elementos encargados de asimilar la información emitida por el resto de los elementos del sistema domótico. Es necesario destacar que dentro de los controladores se incluyen adicionalmente las interfaces que permiten el control del usuario u otra aplicación [13].

Actuadores

Estos son dispositivos que reciben comandos de un determinado sistema y desde luego estas acciones son ejecutadas. Los actuadores empleados dentro del desarrollo de un modelo de automatización pueden operarse de forma neumática, hidráulica o eléctrica. Las principales propiedades que este tipo de componentes pueden impartir al sistema son: potencia, peso, volumen, precisión, velocidad y costo [13].

Sensores

Es un dispositivo que se permite detectar cantidades físicas o químicas como temperatura, luz, distancia, aceleración, inclinación, desplazamiento, presión, fuerza, torque, humedad, pH en otros parámetros y desde esa perspectiva convertirlas en variables eléctricas perfectamente medibles. Dentro del mercado se pueden encontrar una serie de sensores que comúnmente son empleados en la automatización de edificaciones, los más utilizados son los termostatos, detectores de gas, humo, calor, humedad y presencia [13].

Áreas de la Domótica

Las áreas que abarca la domótica tienen su origen en la capacidad que esta tiene para ofrecer y/o asimilar. Dentro de las más comunes tenemos el área de ahorro energético, confort, seguridad y comunicaciones.

Ahorro energético

Las aplicaciones orientadas hacia el ahorro energético están destinadas disminuir la cantidad de energía consumida con base en una serie de aspectos como; vivienda, tarifas energéticas, entre otras. La domótica ofrece a los usuarios un número ilimitado de aplicaciones que únicamente están limitadas a los intereses de los usuarios [14].

Confort

La domótica permite una serie de comodidades, dentro de las cuales las más comunes se enlistan a continuación.

- Regulación de la cantidad de iluminación
- Encendido y apagado de elementos y equipos
- Manejo de luces a distancia
- Cierre de puertas y elevación de persianas
- Control por voz de elementos [14].

Seguridad

En esta área, el uso de dispositivos de control, en particular cámaras de video y sensores, se aplica no solo para proteger contra robos e intrusiones en los hogares, sino también para detectar fugas de gas, robos, incendios, etc [14].

Comunicaciones

En esta tipología de área se incluyen el intercambio de información, tanto entre personas como personas y dispositivos domésticos, ya sea en interiores o exteriores. Dentro de los ejemplos más comunes existen telefonía por medio de centrales domésticas, mantenimiento de electrodomésticos e instalaciones a distancia [14].

Arquitectura

Dentro de un sistema domótico los principales tipos de arquitectura se describen como control centralizado, descentralizado y por último semicentralizado. A continuación, se describe cada uno de ellos.

Control centralizado

En un sistema centralizado, solo un componente contiene la inteligencia y genera acciones. Las señales de los sensores (interruptor, sensor de luminosidad, sensor de movimiento, sensor de temperatura, etc.) se envían al componente central y luego

decide qué acciones deben realizar ciertos actuadores (obturador, interruptor de relé, atenuador, etc.). El componente maestro es esencial para el sistema y si se rompe o se quita, ya nada funcionará. Una ventaja de este sistema es que los sensores y actuadores no necesitan mucha inteligencia y son de construcción sencilla. En los cambios de configuración, solo se debe cambiar el software en el componente maestro. Los costos de expansión son más bajos que con un sistema descentralizado porque los módulos de sensores y actuadores contienen menos inteligencia, pero los costos iniciales son más altos debido al controlador maestro [15].

En general, el componente maestro solicita a los sensores nuevos datos en intervalos de tiempo regulares, lo que también se denomina sondeo. En ese caso, el sistema sería un sistema maestro único. Cuantos más esclavos haya en el bus, más tiempo se necesitará para solicitar cada esclavo. La otra posibilidad es que los sensores puedan enviar datos automáticamente en un cambio. Este sería entonces un sistema multimaestro con la ventaja de que hay menos tráfico en el bus [15].

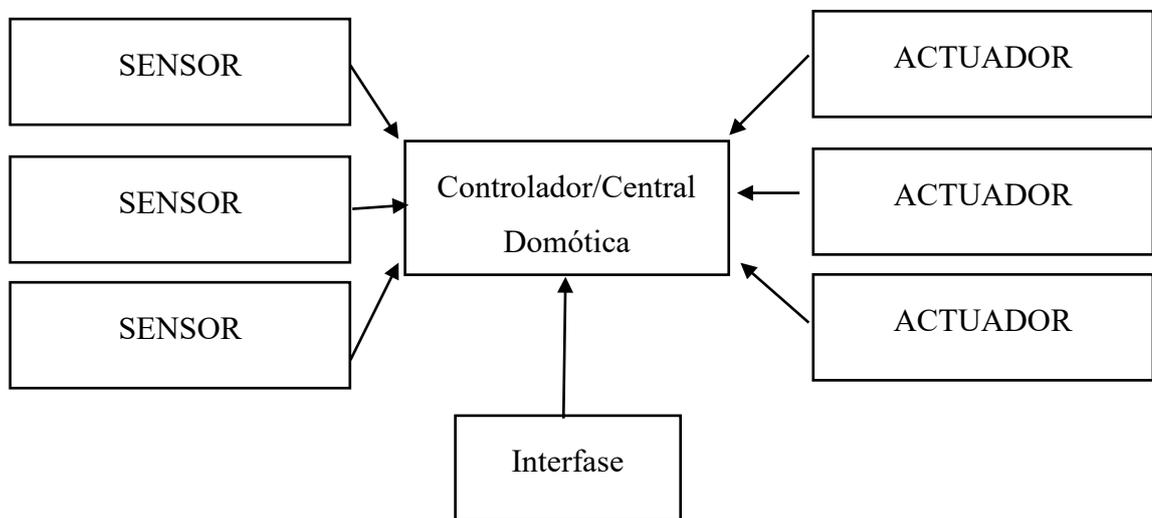


Figura 2.- Arquitectura centralizada

Fuente: Elaborado por el autor basado en [15].

Control Descentralizado

En un sistema descentralizado, cada componente contiene inteligencia y envía comandos directamente al bus. Los actuadores escuchan los comandos en el bus y deciden si un comando en particular les interesa para ejecutar una acción. Tal sistema consta de una fuente de alimentación y dispositivos de entrada y salida. Por lo tanto, el precio de un sistema pequeño es bajo, pero las expansiones generalmente son más costosas que con un sistema centralizado. Si un componente falla, no afectará el funcionamiento de los otros componentes. La actividad del bus depende de la cantidad de eventos y no de la cantidad de sensores en el bus [16].

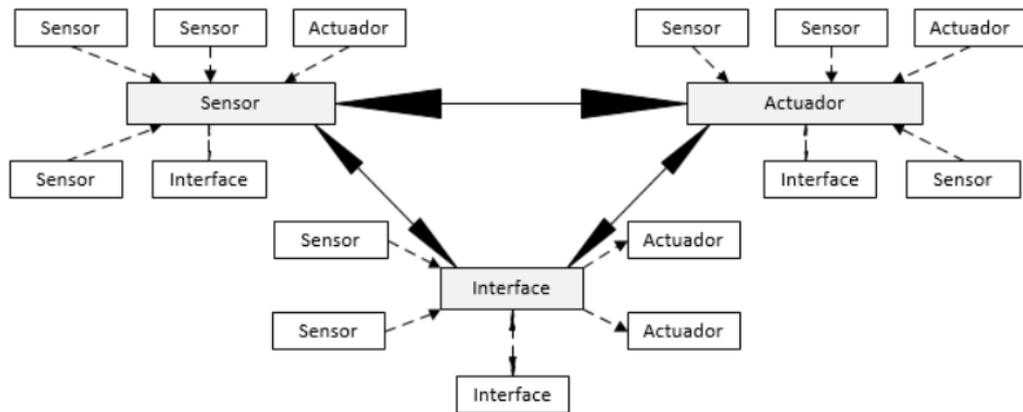


Figura 3.- Arquitectura descentralizada.

Fuente: [16].

Control Semicentralizado

Un sistema semicentralizado consta de unos pocos módulos (p. ej., módulo de conmutación de luces, módulo de atenuación, módulo de persianas, varios módulos de sensores, etc.) que son independientes y la inteligencia se aloja allí. Entonces, si estos módulos están conectados entre sí a través del bus, la inteligencia se distribuye entre los distintos módulos de salida y cada módulo de salida tiene solo la inteligencia para controlar sus propias acciones. El sistema consta de subsistemas con su propio controlador. Entonces, si un módulo falla, los otros módulos aún pueden funcionar [7].

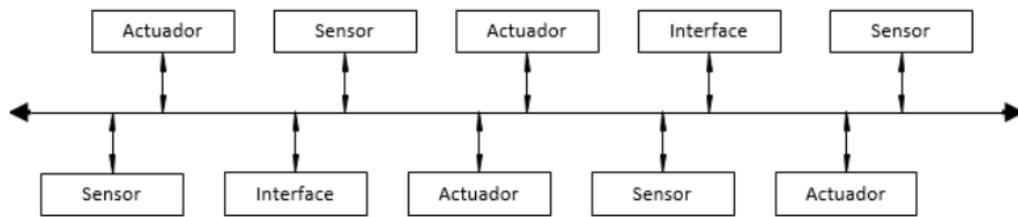


Figura 4.- Arquitectura semicentralizado.

Fuente: [7].

Sistema de control

Se denomina sistema de control al modelo que manipula salidas a través de entradas empleados en sistemas electrónicos denominados como microcontroladores. Dentro los sistemas más comunes tenemos sistema de control de lazo abierto y cerrado.

Sistemas de control de lazo abierto

Un sistema de lazo abierto es un tipo de sistema de control en el que la salida del sistema depende de la entrada, pero la entrada o el controlador es independiente de la salida del sistema. Estos sistemas no contienen ningún bucle de realimentación, por lo que también se conocen como sistemas sin realimentación [17].

Sistemas de control de lazo cerrado

Un sistema de control de bucle cerrado es un dispositivo mecánico o electrónico que regula automáticamente un sistema para mantener un estado o punto de consigna deseado sin interacción humana. Este sistema utiliza un sistema de realimentación o sensor. El control en bucle cerrado contrasta con el control en bucle abierto, en el que no existe un mecanismo de autorregulación y suele ser necesaria la interacción humana [17].

1.3.2. Inteligencia Artificial

La Inteligencia Artificial (IA) es una rama de la informática que persigue crear ordenadores y/o máquinas tan inteligentes como los seres humanos. John McCarthy, el padre de la Inteligencia Artificial, describió la IA como la ciencia y la ingeniería de crear máquinas inteligentes, especialmente programas informáticos inteligentes [18]. Por otro lado, la IA es una rama de la Ciencia que trata de ayudar a las máquinas a encontrar soluciones a problemas complejos de una forma más parecida a la humana [19]. En general, se trata de tomar prestadas características de la inteligencia humana y aplicarlas en forma de algoritmos de manera que resulten fáciles de utilizar para los ordenadores [20].

En función de los requisitos establecidos, puede adoptarse un enfoque más o menos flexible o eficiente, lo que influye en lo artificial que parezca el comportamiento inteligente. Lo artificial es definido en diferentes enfoques por diversos investigadores a lo largo de su evolución. La Inteligencia Artificial es el estudio de cómo hacer que los ordenadores hagan cosas que, de momento, hacen mejor las personas [11]. Hay otras definiciones posibles; la IA es una colección de problemas difíciles que pueden resolver los humanos y otros seres vivos; sin embargo, no existen los algoritmos para resolverlos, un claro ejemplo de este tema son la comprensión del lenguaje natural hablado, el diagnóstico médico, el diseño de circuitos, el aprendizaje, la autoadaptación, el razonamiento, el juego de ajedrez, la demostración de teorías matemáticas, etc [20].

Aplicaciones de la Inteligencia Artificial

En 1981, la Inteligencia Artificial se convirtió en una industria. Las teorías y técnicas propuestas en la literatura sobre Inteligencia Artificial fueron aceptadas por las industrias, que empezaron a invertir en el área [21]. Los conceptos de Inteligencia Artificial se aplican en diversos campos. Algunos de ellos son los siguientes:

Juegos

La IA desempeña un papel crucial en juegos estratégicos como el ajedrez, el póquer, el tres en raya, etc., en los que la máquina puede pensar en un gran número de posiciones posibles basándose en conocimientos heurísticos [22].

Reconocimiento de la voz

Algunos sistemas inteligentes son capaces de escuchar y comprender el lenguaje en términos de frases y sus significados mientras un humano le habla. Puede manejar diferentes acentos, palabras de argot, ruido de fondo, cambios en el ruido humano debido al frío, etc [10].

Comprensión del lenguaje natural

El ordenador ya puede entender las lenguas naturales y, por tanto, los humanos pueden interactuar utilizando lenguas habladas naturales. Para ello, es necesario que el ordenador comprenda el tema del que trata el texto, lo cual sólo es posible en un número muy limitado de ámbitos [23].

Visión por ordenador

La visión por ordenador permite al ordenador comprender las señales y actuar en consecuencia. Existen algunos sistemas como, por ejemplo; sistema de reconocimiento facial instalado en un aeropuerto y sobre todo aspecto el diagnóstico médico [21].

Robots inteligentes

Los robots disponen de sensores para detectar datos físicos del mundo real como la luz, el calor, la temperatura, el movimiento, el sonido, los golpes y la presión. Sin mencionar el hecho que disponen de procesadores eficientes, múltiples sensores y una memoria enorme, para exhibir inteligencia [24].

Machine Learning (ML)

Es un conjunto de técnicas relacionadas con la Inteligencia Artificial y se basa en algoritmos que intentan aprender de grandes conjuntos de datos. Una característica muy importante de estos algoritmos es la predicción de nuevos casos en base a la experiencia obtenida del conjunto de datos utilizado para el entrenamiento. El Machine Learning suele dividirse en dos tipos, el aprendizaje supervisado y no supervisado [25].

El aprendizaje supervisado cada observación o muestra de un conjunto de datos tiene asociada una variable o dato que indica qué sucedió. Es decir, que aspectos y/o características fueron analizadas y debidamente etiquetadas. Este tipo de aprendizaje se divide en clasificación y regresión [26]. En la clasificación, las salidas del sistema son finitas y discretas, y se interpretan en términos de la clase a la que pertenecen y, por otro lado, en regresión se denominan salidas continuas.

En el aprendizaje no supervisado hay datos en el conjunto de datos para el entrenamiento, pero el resultado es desconocido o falta, o se sabe poco al respecto. Es decir, no hay una variable objetivo y se buscan patrones. Se pueden encontrar estructuras en los datos para determinar qué predecir. Estas estructuras incluyen el agrupamiento (el proceso de dividir un conjunto de datos en un conjunto de subclases significativas llamadas grupos) y la asociación (un conjunto de atributos significativos) [27].

Deep Learning (DL)

El aprendizaje profundo es una rama del aprendizaje automático que utiliza varias estructuras de redes neuronales para aprender capas sucesivas de representaciones de datos cada vez más significativas. Se refiere a la cantidad de capas de representación utilizadas en el modelo de aprendizaje profundo. Por lo general, se utilizan decenas o cientos de capas de representación, que aprenden automáticamente a medida que el modelo se entrena en los datos. A continuación, se muestra un modelo representativo de lo anteriormente descrito [28].

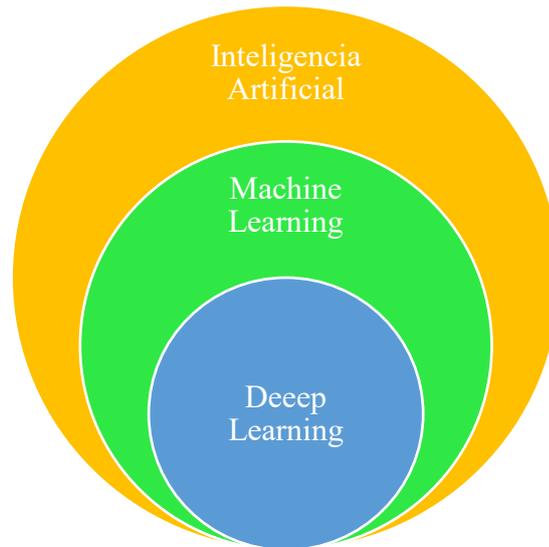


Figura 5.- Inteligencia Artificial y sus categorías fundamentales.

Fuente: [28].

En general, cualquier técnica de aprendizaje automático intenta asignar entradas a salidas de destino. El aprendizaje profundo asigna entradas a objetivos utilizando una Red Neuronal artificial compuesta por una gran cantidad de capas dispuestas jerárquicamente. La red aprende algo simple en la primera capa de la jerarquía y luego pasa esa información a la siguiente capa. La siguiente capa toma esa información simple, la combina en algo más complejo y la pasa a una tercera capa. Este proceso continúa para que cada capa de la jerarquía se base en algo más complejo a partir de las entradas recibidas de la capa anterior.

Tipos de Inteligencia Artificial

Existen varios tipos de Inteligencia Artificial, que se generalizan en base al avance de los tipos de IA. Estas se enlistan a continuación:

Máquinas reactivas

La génesis de la IA comenzó con el desarrollo de las máquinas reactivas, el tipo más fundamental de IA. Las máquinas reactivas son precisamente eso: reactivas. Pueden responder a peticiones y tareas inmediatas, pero no son capaces de almacenar memoria o aprender de experiencias pasadas [29].

Inteligencia Artificial Estrecha (Limitada)

La Inteligencia Artificial Estrecha (IAN), también conocida como IA describe las herramientas de IA diseñadas para llevar a cabo acciones o comandos muy específicos. Las tecnologías IAN están construidas para servir y sobresalir en una capacidad cognitiva, y no pueden aprender de forma independiente habilidades más allá de su diseño. A menudo utilizan algoritmos de aprendizaje automático y redes neuronales para completar estas tareas específicas [30].

Inteligencia Artificial General

La Inteligencia Artificial General (IAG) describe la IA que puede aprender, pensar y realizar una amplia gama de acciones de forma similar a los humanos. El objetivo del diseño de la IAG es poder crear máquinas capaces de realizar tareas multifuncionales y de actuar como asistentes de los seres humanos en la vida cotidiana [30].

Autoconciencia

La superInteligencia Artificial es material de ciencia ficción. Se teoriza que una vez que la IA haya alcanzado el nivel de inteligencia general, pronto aprenderá a un ritmo tan rápido que sus conocimientos y capacidades serán más fuertes incluso que los de la humanidad. La ASI actuaría como tecnología vertebradora de la IA completamente consciente de sí misma y de otros robots individualistas [29].

Reconocimiento de voz

De acuerdo con Barrios *et al.* [31], durante mucho tiempo ha existido la ilusión de hablar con un ordenador y ser comprendido. Aunque se ha producido un enorme desarrollo en los últimos años, incluso los mejores sistemas actuales están aún lejos de la comprensión real. Sin embargo, cada vez son más capaces de convertir el habla humana de una señal de audio en un texto. Recientemente, incluso han tenido cierto éxito con el habla natural y fluida [32]. Por ende, a continuación, se detallan algunas aplicaciones más puntuales que involucran la IA y el reconocimiento de voz.

- **Programas de dictado**

En la actualidad existen varios, como IBM ViaVoice o Dragon Systems Naturally Speaking, que pueden manejar vocabularios de varios cientos de miles de palabras. Generalmente, esto requiere formación del orador que va a utilizar el sistema y un micrófono de bajo ruido. Este último suele ser suministrado por el fabricante del software [33].

- **Navegación y telefonía**

Algunos grandes fabricantes de automóviles ofrecen ahora sistemas que controlan el teléfono o el sistema de navegación por voz. Una vez más, al igual que con los sistemas de dictado, la atención se centra en la cancelación del ruido. Afortunadamente, las respuestas al impulso de la sala en los coches son bastante cortas, por lo que no hay grandes problemas en este sentido [4].

- **Sistemas de información telefónica**

Un claro ejemplo de esta temática son los ferrocarriles alemanes, así como muchos bancos y otros proveedores. Estos permiten ahora a los usuarios hacer llamadas telefónicas con sistemas de diálogo por voz; este tipo de aplicaciones constituye un interesante experimento práctico [2].

- **Apoyo a discapacitados y comunicación**

Los sistemas de dictado y control por voz pueden sin duda facilitar la vida a las personas con deficiencias motrices. Sin embargo, lo que sigue faltando a este respecto suelen ser buenas ideas sobre cómo utilizar el control por voz de forma que se aprovechen al máximo sus ventajas. Esta es probablemente una de las interfaces más interesantes entre la ciencia de la comunicación, la ergonomía y la tecnología de software que se ha utilizado poco hasta ahora [34].

1.3.3. Internet de las Cosas (IoT)

El Internet de las cosas (IoT) se define como un sistema que está interconectado con sensores y actuadores que se encuentran en dispositivos y se utilizan para aprovechar los datos que se obtienen a través de sensores [12]. Se refiere a este sistema también como “internet de los objetos”, lo que implica el uso de objetos que se conectan a una red inalámbrica y un sistema de autoconfiguración, por ejemplo, un hogar.

A continuación, se explican los tres componentes principales de Internet de las Cosas que permiten la comunicación entre dispositivos interconectados: Hardware: incluye un grupo de herramientas de comunicación integradas, sensores y actuadores.

- **Middleware:** Hacen referencia a las herramientas de análisis de datos que se utilizan y los servicios de almacenamiento bajo demanda.
- **Presentación:** El resultado debe presentarse de una manera sencilla que sea fácil de visualizar, comprender e interpretar.

Se puede añadir que para el correcto desempeño del Internet de las Cosas (IoT) funcione correctamente y logre los resultados deseados, es crucial contar con las siguientes tecnologías:

- **Identificación por radiofrecuencia (RFID):** esta tecnología es clave para la comunicación integrada, ya que permite el diseño de microchips que se utilizan para la comunicación inalámbrica de datos. La misma tecnología también está

disponible en las aplicaciones de control de acceso 9, así como en las instalaciones de transporte, como el reemplazo de boletos y los boletos de registro.

- **Redes de Sensores Inalámbricos (WSN):** Se refiere a un grupo de sensores inteligentes que se encargan de recopilar, procesar y analizar información vital que se ha obtenido en diferentes entornos. Los datos obtenidos de los sensores se distribuyen entre varios nodos y se envían para su posterior análisis.
- **Esquemas de direccionamiento:** un sistema debe ser capaz de identificar 'cosas' para el éxito de los elementos de conexión de IoT para una identificación más fácil de la ubicación y otras funcionalidades.
- **Análisis y almacenamiento de datos:** la creación de datos es uno de los resultados que se registran cuando se usa IoT; por lo tanto, es fundamental que los datos se analicen y almacenen correctamente. Los sistemas de Inteligencia Artificial a menudo se desarrollan con algoritmos para facilitar este proceso y los datos se centralizan o distribuyen según sea necesario.
- **Visualización:** Este es un paso vital en IoT ya que facilita la interacción entre el usuario y el entorno. Las interfaces utilizadas para dichos dispositivos deben ser atractivas y fáciles de navegar para los usuarios [12].

1.3.4. Discapacidad

La discapacidad se conoce como algún tipo de restricción de una capacidad humana, la cual limita el buen desempeño de acciones o funciones específicas, provocando así una notoria desventaja al desenvolverse en la vida cotidiana, en su medio físico o social. Sus causas pueden ser físicas, psíquicas y sensoriales; a su vez pueden ser discapacidades permanentes o transitorias [35].

Por otro lado, la discapacidad se define como una alteración permanente y grave de la participación social y económica de una persona; está causada por la interacción de factores sociales u otros factores ambientales desfavorables (barreras) y las características de la persona afectada que dificultan o impiden superar las barreras. Por tanto, la discapacidad no se considera una enfermedad. Tanto los objetos e instalaciones cotidianos o la falta de ellos como las actitudes de otras personas tienen un efecto discapacitante en el entorno de la persona [36]. Las barreras a los objetos adquieren a menudo su cualidad de discapacitantes por la falta de un diseño universal que tenga en cuenta no sólo las necesidades de grupos de población numéricamente grandes o influyentes [37].

Tipos de discapacidad

La discapacidad se puede clasificar de acuerdo con el tipo de afectación que sufre una persona y se puede ver a continuación:

Discapacidad sensorial visual

La discapacidad sensorial visual se puede definir como el déficit en las funciones visuales y estructuras corporales del ojo o sistema nervioso, se caracteriza por la pérdida o alteración parcial o completa de su función y las limitaciones que presenta el individuo al realizar una tarea o acción en un contexto normalizado [37].

Discapacidad sensorial auditiva

La discapacidad sensorial auditiva se puede definir como las deficiencias en las funciones y estructuras corporales del sistema auditivo, es decir que no pueden escuchar normalmente y las limitaciones que presenta el individuo al realizar una tarea o acción en un contexto normalizado [37].

Discapacidad visceral

La discapacidad visceral se puede definir como las deficiencias en las funciones y estructuras corporales de los sistemas cardiovascular, hematológico, inmunológico, respiratorio, digestivo, metabólico, endocrino y genitourinarias, asociadas o no a otras funciones o estructuras corporales deficientes, y la imposibilidad de desarrollar actividades cotidianas con normalidad, por ejemplo: un diabético o un cardíaco [38].

1.4. Objetivos

1.4.1. Objetivo general

- Implementar un prototipo de un sistema domótico para personas con discapacidad visual mediante Inteligencia Artificial y reconocimiento de voz.

1.4.2. Objetivos específicos

- Identificar el estado actual de las viviendas de las personas con discapacidad visual y establecer los requerimientos del sistema de control domótico en la provincia de Tungurahua, cantón Ambato.
- Diseñar el sistema domótico para personas con discapacidad visual mediante Inteligencia Artificial y reconocimiento de voz.
- Evaluar el desempeño del prototipo desarrollado.

CAPÍTULO II

METODOLOGÍA

2.1. Materiales

Para la implementación del Sistema Domótico para personas con discapacidad visual se utilizó varios elementos electrónicos como: un micrófono Bluetooth que combina las funcionalidades de un micrófono y la tecnología inalámbrica Bluetooth para no utilizar cables y conectarse a un dispositivo, como un teléfono móvil, una computadora o un altavoz. Esto proporciona libertad de movimiento y flexibilidad al utilizar el micrófono, ya que no está limitado por cables. Como sistema central se utilizó la Raspberry Pi 4 que es una pequeña placa electrónica que contiene todos los componentes necesarios para funcionar como un ordenador completo en un formato compacto. Esta placa también incluye puertos USB 3.0, puertos USB 2.0, puerto Ethernet, salida HDMI, conector de audio, ranura para tarjeta microSD y conectores GPIO que serán las salidas de control domótico. La Raspberry Pi 4 es capaz de ejecutar una variedad de sistemas operativos, siendo el sistema operativo oficial llamado Raspberry Pi OS (anteriormente conocido como Raspbian), basado en Linux. Esto la hace adecuada para proyectos de automatización del hogar.

De igual manera se utilizó servomotores que son un tipo de motor utilizado para controlar con precisión el movimiento angular de un eje. A diferencia de los motores convencionales que giran continuamente, los servomotores están diseñados para moverse a posiciones específicas y mantenerse en ellas, proporcionando un control preciso sobre la posición angular. En el proyecto se utilizó los servomotores para abrir y cerrar las cortinas mediante un mecanismo de biela-manivela con ayuda del posicionamiento del eje del servomotor. También se utilizó un ventilador DC que consta de un motor de corriente continua (DC) para generar flujo de aire y proporcionar enfriamiento. Este ventilador funciona con una tensión de alimentación de 5 voltios, lo que hace que sea compatible con la fuente de energía del sistema domótico, se utilizó este dispositivo para simular el aire acondicionado dentro del hogar.

Además, se utilizó un relé de 5v el cual es un dispositivo electromecánico que se utiliza para controlar la conexión o desconexión de un circuito eléctrico a través de una señal de entrada. Funciona como un interruptor controlado por un electroimán. En el proyecto se utilizó este dispositivo para simular las cerraduras magnéticas y poder tener el control de las puertas para poder abrirlas y cerrarlas mediante el envío de una señal. Del mismo modo, se utilizó leds de alta luminosidad para la iluminación del hogar, los mismo que trabajan con un voltaje de entrada de 5v y son compatibles con la fuente del sistema domótico ya que no consumen mucha energía y permiten que el sistema trabaje de manera correcta.

Así mismo para el entrenamiento de voz se desarrolló con TENSORFLOW pues es una biblioteca de código abierto desarrollada por Google que se utiliza ampliamente en el campo del aprendizaje automático y la Inteligencia Artificial, para desarrollo de software se utilizó Python para implementar el modelo entrenado y gestionar el control de las salidas GPIO.

2.2. Métodos

2.2.1. Modalidad de la investigación

En primera instancia el presente trabajo se categorizó como una Investigación de Campo ya que se emplearon todas las habilidades adquiridas durante la carrera. Por otro lado, es necesario destacar que por medio de esta indagación se pretende mejorar la calidad de vida de las personas con discapacidad visual.

De igual manera se caracterizó por ser de carácter bibliográfico, ya que la información se basó en la búsqueda de información en artículos científicos, nacionales e internacionales, además páginas web sobre proyectos, artículos, revistas y libros con un análisis similar de variables. Es por ese motivo que la presente investigación se basa en una investigación experimental, puesto que se realizó pruebas en el transcurso del desarrollo del proyecto cumpliendo así con los objetivos trazados.

2.2.2. Población y muestra

Población

La población abordada dentro de la presente investigación son las personas con discapacidad visual del cantón Ambato provincia Tungurahua de la “Escuela de Educación Básica Especializada Cardenal Julius Doephner”. En la tabla 1 se detalla la información de cuantas personas con discapacidad visual se encuentran registradas en la Escuela de Educación Básica.

Tabla 1.- Población de estudio tomada de la Escuela de Educación Básica Julius Doephner.

Escuela	Tipo de Discapacidad	Edad	Genero	#
Escuela de Educación Básica Especializada Cardenal “Julius Doephner”	Visual	9 a 17 años	Femenino	9
			Masculino	12
Total				21

Fuente: Elaborado por Investigador.

Muestra

La población al ser menos de 100 individuos no requirió de la aplicación de la fórmula de población finita. Es decir, todo el trabajo se enfocó en un modelo no probabilístico de selección por conveniencia, tomando en cuenta así a todos los estudiantes registrados en la “Escuela de Educación Básica Especializada Cardenal Julius Doephner” que en total suman 21 estudiantes registrados.

2.2.3. Recolección de información

La recolección de la información se realizó a través de páginas web y fuentes bibliográficas con el propósito de ampliar la visión del proyecto por medio de revistas,

tutoriales y hojas de datos técnicos para un enfoque teórico-práctico y manuales de los dispositivos a utilizar para una mejor comprensión del tema de investigación. Por otro lado, es necesario acotar dicha información se caracterizó desde una perspectiva de índole netamente cuantitativa pues se analizaron los datos del instrumento aplicado a los sujetos de estudio. En adición, es menester resaltar que la investigación fue de corte transversal. Es decir, la toma de datos a los individuos de estudio se efectuó en un solo momento y no fue necesario réplicas. La data recabada tuvo el propósito de responder a los objetivos planteados.

2.2.4. Procesamiento y análisis de datos

El procesamiento y análisis de datos se constituye de una serie de pasos que se enlistan a continuación:

- Organización de la información recolectada con el propósito de sustentar las variables de estudio.
- Revisión de la información obtenida y categorizarla con el objetivo de establecer un análisis significativo.
- Análisis de la información que permita establecer una alternativa de solución del problema
- Planteamiento de la solución con base en una maqueta que contenga la propuesta de solución
- Pruebas de funcionamiento y validación.

Como se detalló con antelación la información se categorizó desde una perspectiva cuantitativa para su asimilación con base en tablas de frecuencia y porcentaje para de esa manera generar un análisis e interpretación de datos.

CAPÍTULO III

RESULTADOS Y DISCUSIÓN

3.1. Análisis y discusión de resultados

El sistema domótico para personas con discapacidad visual mediante Inteligencia Artificial y reconocimiento de voz cuenta con información de suma importancia que contribuyen a mejorar la calidad de vida de personas con discapacidad visual. Para ello primero se analizó la factibilidad con respecto al desarrollo de un sistema domótico para personas con discapacidad visual y los principales requerimientos de las personas, las tecnologías de reconocimiento de voz y los dispositivos electrónicos adecuados para el sistema domótico. En este aspecto es importante resaltar que cada uno de los elementos y el análisis que se realizó permitieron tener un alto porcentaje de eficiencia del sistema de 94,29% y un error mínimo del 5,71%. Es necesario destacar también que cada componente fue seleccionado adecuadamente con base a su aplicación en el proyecto; tal es el caso de la tecnología móvil. Este aspecto en particular funciona por medio de bluetooth, puesto que es más simple de aplicar, manejar y controlar. Cada uno de estos aspectos permitieron solventar cada una de las necesidades de las personas con discapacidad visual proporcionándoles seguridad, confort y desde esa perspectiva un sistema fácil de manipular con base en sus capacidades especiales.

3.2. Desarrollo de la propuesta

En este apartado se enlistan la consecución de los objetivos planteados en el desarrollo de la presente investigación.

3.2.1. Análisis de la factibilidad

Para realizar el análisis de factibilidad de este proyecto se debe considerar tanto la factibilidad técnica, bibliográfica y económica.

Factibilidad Técnica

El presente proyecto es factible técnicamente, pues los elementos electrónicos que se utilizan se pueden conseguir de manera fácil y rápida en el país.

Factibilidad bibliográfica

Es factible bibliográficamente, pues información acerca de control domótico, control por voz y entrenamiento de voz, se encuentra documentando en varios libros, tesis, revistas que se encuentran en la biblioteca y bases de dato de la Universidad Técnica de Ambato, de igual manera existe mucha información en la web.

Factibilidad económica

El investigador tiene la posibilidad de costear este proyecto, por lo cual es factible económicamente.

3.2.2. Resultados y Tabulación de datos

Una vez tomada la muestra de 21 personas con discapacidad visual registradas en la “Escuela de Educación Básica Especializada Cardenal Julius Doepfner” en el Cantón Ambato, se identificó el estado actual de las viviendas de las personas con discapacidad visual. En la tabla 2, se presenta la tabulación de las preguntas realizadas en la encuesta a las personas con discapacidad visual.

Tabla 2.- Tabulación de datos.

Ítem	Pregunta	Respuesta	N.º	%
1	¿La vivienda en la que usted habita es?	Arrendada	6	28.6%
		Propia	15	71.4%
2	¿Cómo considera usted la accesibilidad con la que cuenta una persona con discapacidad visual en su vivienda?	Excelente	0	0%
		Buena	2	9.5%
		Regular	5	23.8%
		Mala	14	66.7%

3	¿Las medidas de seguridad con las que cuenta su vivienda para una persona con discapacidad visual son?	Excelentes	0	0%
		Buenas	1	4.8%
		Regular	6	28.6%
		Malas	14	66.7%
4	¿Cómo estima usted la inclusión de la domótica en una vivienda para mejorar la calidad de vida de personas con discapacidad visual?	Excelente	13	61.9%
		Buena	7	33.3%
		Regular	1	4.8%
		Mala	0	0%
5	¿Qué aspectos tecnológicos considera usted deben solventarse para mejorar la calidad de vida de las personas con discapacidad visual dentro del hogar?	Control de dispositivos y artefactos electrónicos	5	23.8%
		Control de iluminación	3	14.3%
		Seguridad	2	9.5%
		Acceso	1	4.8%
		Todos	10	47.6%

Fuente: Elaborado por Investigador.

3.2.3. Análisis e interpretación de datos

Pregunta N°1. ¿La vivienda en la que usted habita es?

Tabla 3.- Tabulación de datos, pregunta 1.

1.- ¿La vivienda en la que usted habita es?				
	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Propia	15	71.4	71.4	71.4
Arrendada	6	28.6	28.6	100.0
Total	21	100.0	100.0	

Fuente: Elaborado por Investigador.



Figura 6.- Tabulación de datos, pregunta 1.

Fuente: Elaborado por Investigador.

Interpretación de resultados: El 71.43% de las personas encuestadas denotan que su vivienda es propia mientras que el 28.6% de los estudiantes aseveró vivir en una casa de arriendo.

Análisis: De acuerdo con el resultado de las encuestas realizadas se puede deducir que la mayor parte de las personas con discapacidad visual de la “Escuela de Educación Básica Especializada Cardenal Julius Doepfner” en el Cantón Ambato poseen un hogar propio, por lo tanto, cuentan con mayor posibilidad de tener instalaciones adecuadas en la misma para mejorar su calidad de vida.

Pregunta N°2. ¿Cómo considera usted la accesibilidad con la que cuenta una persona con discapacidad visual en su vivienda?

Tabla 4.- Tabulación de datos, pregunta 2.

2.- ¿Cómo considera usted la accesibilidad con la que cuenta una persona con discapacidad visual en su vivienda?				
	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Excelente	0	0	0	0
Mala	14	66.7	66.7	66.7
Regular	5	23.8	23.8	90.5
Buena	2	9.5	9.5	100.0
Total	21	100.0	100.0	

Fuente: Elaborado por Investigador.



Figura 7.- Tabulación de datos, pregunta 2.

Fuente: Elaborado por Investigador.

Interpretación de resultados: El 66.67% de las personas encuestadas denotan que la accesibilidad dentro de su vivienda mala. El 23.61% comentó poseer accesibilidad regular. Y, por último, el 9.52% detalló una accesibilidad buena.

Análisis: Según los resultados de la encuesta realizada es acertado mencionar que de la población de estudio el mayor porcentaje de estudiantes con discapacidad visual posee accesibilidad mala y la vivienda en la que habita no cuenta con una adecuada instalación para tener fácil acceso.

Pregunta N°3. ¿Las medidas de seguridad con las que cuenta su vivienda para una persona con discapacidad visual son?

Tabla 5.- Tabulación de datos, pregunta 3.

3.- ¿Las medidas de seguridad con las que cuenta su vivienda para una persona con discapacidad visual son?				
	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Excelente	0	0	0	0
Mala	14	66.7	66.7	66.7
Regular	6	28.6	28.6	95.2
Buena	1	4.8	4.8	100.0
Total	21	100.0	100.0	

Fuente: Elaborado por Investigador.



Figura 8.- Tabulación de datos, pregunta 3.

Fuente: Elaborado por Investigador.

Interpretación de resultados: El 66.67% de las personas encuestadas denotan que la seguridad dentro de su vivienda es mala. El 28.57% comentó poseer una seguridad regular dentro de su vivienda, sin embargo, el 4.76% aseguró que cuenta con una seguridad buena dentro de su hogar.

Análisis: Se puede verificar que existen porcentajes significativos de 66.67% y 28.57% de personas encuestadas expresan que las medidas de seguridad con las que cuenta su vivienda para una persona con discapacidad visual son malas, y por ende este punto es de suma importancia ya que en primer lugar hay que cuidar de la integridad física como mental de las personas con discapacidad visual.

Pregunta N°4. ¿Cómo estima usted la inclusión de la domótica en una vivienda para mejorar la calidad de vida de personas con discapacidad visual?

Tabla 6.- Tabulación de datos, pregunta 4.

4.- ¿Cómo estima usted la inclusión de la domótica en una vivienda para mejorar la calidad de vida de personas con discapacidad visual?				
	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Mala	0	0	0	0
Regular	1	4.8	4.8	4.8
Buena	7	33.3	33.3	38.1
Excelente	13	61.9	61.9	100.0
Total	21	100.0	100.0	

Fuente: Elaborado por Investigador.

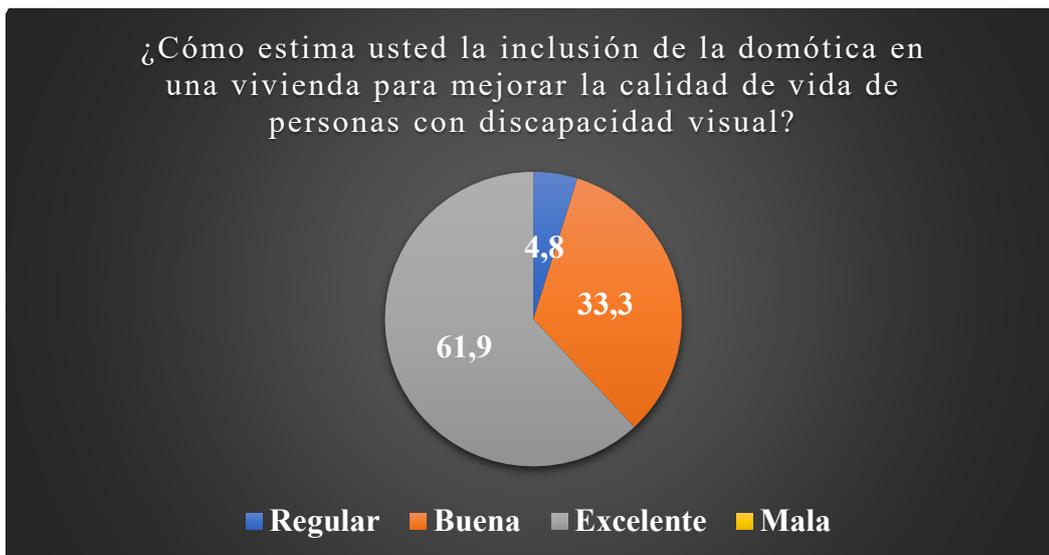


Figura 9.- Tabulación de datos, pregunta 4.

Fuente: Elaborado por Investigador.

Interpretación de resultados: El 61.90% de las personas encuestadas denotan que la domótica aplicada en su hogar sería una excelente idea. El 33.33% comentó que la idea de implementar la domótica en su hogar sería buena; sin embargo, el 4.76% detalló ser una idea regular.

Análisis: De acuerdo con los resultados obtenidos en las encuestas se puede verificar que la mayor parte de personas piensan que la inclusión de la domótica en su hogar sería una buena idea para mejorar la calidad de vida de las personas con discapacidad visual.

Pregunta N°5. ¿Qué aspectos tecnológicos considera usted que deben solventarse para mejorar la calidad de vida de las personas con discapacidad visual dentro del hogar?

Tabla 7.- Tabulación de datos, pregunta 5.

5.- ¿Qué aspectos tecnológicos considera usted que deben solventarse para mejorar la calidad de vida de las personas con discapacidad visual dentro del hogar?				
	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Control de dispositivos y artefactos electrónicos	5	23.8	23.8	23.8
Control de iluminación	3	14.3	14.3	38.1
Seguridad	2	9.5	9.5	47.6
Acceso	1	4.8	4.8	52.4
Todos	10	47.6	47.6	100.0
Total	21	100.0	100.0	

Fuente: Elaborado por Investigador.

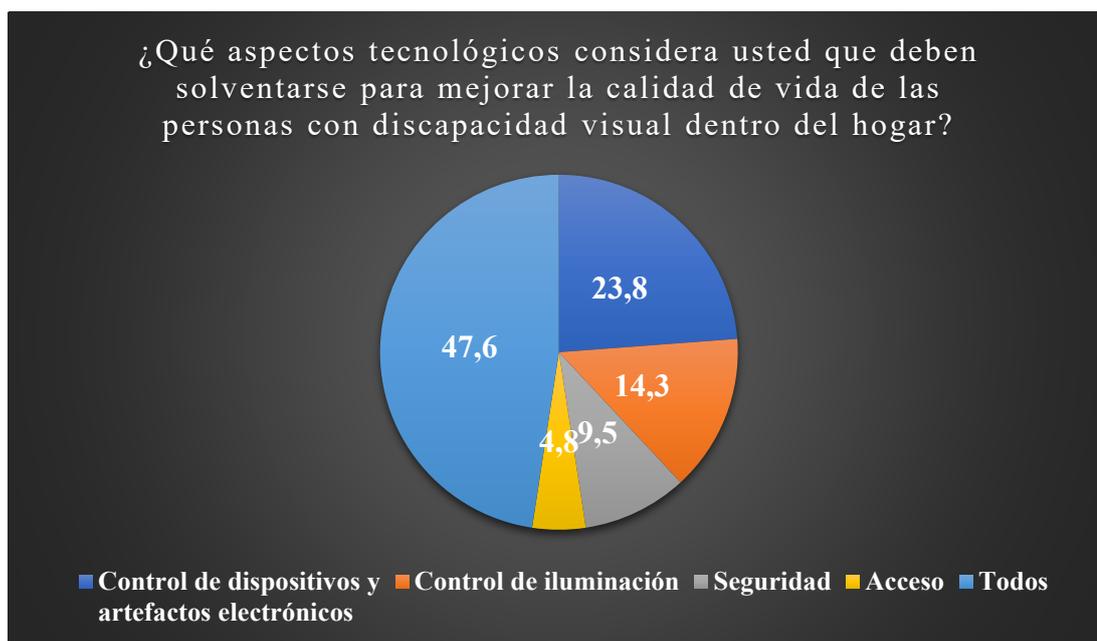


Figura 10.- Tabulación de datos, pregunta 5.

Fuente: Elaborado por Investigador.

Interpretación de resultados: El 23.8% de las personas encuestadas consideran que debe proveerse el control de dispositivos y artefactos electrónicos, el 14.3% piensan

que es necesario el control de iluminación, el 9.5% afirma que es fundamental la seguridad en el hogar, el 4.8% acceso y el 47.6% piensa que es sumamente necesario poder contar con todos los aspectos tecnológicos mencionados.

Análisis: En base a los datos obtenidos en las encuestas realizadas se logra determinar que mayoritariamente las personas consideran que las inclusiones de cada uno de los aspectos tecnológicos son indispensables para mejorar la calidad de vida de las personas con discapacidad visual.

Análisis global de la encuesta

En base a los datos obtenidos en la encuesta realizada se determina que la mayoría de las personas sienten que hay una falta de atención a las personas con discapacidad, por lo que estas personas piensan que la inclusión de la tecnología en sus viviendas como: el control de equipos electrónicos, iluminación, accesibilidad y seguridad son aspectos que contribuyen de gran manera a facilitar y mejorar la calidad de vida de las personas con discapacidad visual. Es necesario destacar de igual manera que como la mayoría de los indagados posee una vivienda propia el trabajo es totalmente factible.

Por otro lado, para el análisis de la estadística de fiabilidad se utilizó el programa estadístico Excel para analizar las preguntas realizadas a los estudiantes con discapacidad visual registrados en la “Escuela de Educación Básica Especializada Cardenal Julius Doephner” en el Cantón Ambato, el cual nos brinda el valor del alfa de Cronbach de 0,854; siendo el mínimo 0.8. En consecuencia, el análisis de la encuesta planteada presenta que las preguntas de la encuesta aplicada tienen una gran relación entre las mismas por ende el instrumento es fiable y puede ser aplicado sin mayor reparo.

3.3. Requerimientos

3.3.1. Requerimientos de las personas con discapacidad visual

De acuerdo con la realización de la investigación previa y una vez hecha la interpretación de los datos obtenidos en la encuesta realizada a los estudiantes con discapacidad visual de la “Escuela de Educación Básica Especializada Cardenal Julius Doepfner” en el Cantón Ambato, se determinó los principales requerimientos que necesita una persona con discapacidad visual dentro de su vivienda para mejorar y facilitar la calidad de vida y su rutina diaria, los cuales son:

- **Control de luces:** Se solicita que se realice el control de iluminación de encendido y apagado en las diferentes áreas del hogar tales como, luz del cuarto, luz de la sala, luz del baño y luz de la cocina de la vivienda.
- **Control de puertas:** De igual manera se necesita el control total de las puertas de la vivienda tales como: la puerta principal y la puerta del cuarto, que estas se abran y se cierren.
- **Control de persianas:** De igual manera se necesita el control total de las persianas de la vivienda tales como: la persiana del cuarto y la persiana de la sala, que estas se abran y se cierren.
- **Control de un dispositivo eléctrico:** También se requiere el control de encendido y apagado de dispositivos electrónicos de 110 V AC los cuales sean útiles para personas con discapacidad visual, como una radio, lámpara entre otros, en este caso se implementó el control de una cocina de inducción.
- **Sistema de seguridad:** Se requiere que el sistema una vez instalado en la vivienda solamente se active mediante la voz de la persona con discapacidad visual, es por ello por lo que se desarrolló el entrenamiento de una Red Neuronal con la voz de la persona que controlará el sistema domótico.

Tomando en consideración cada uno de los requerimientos de las personas con discapacidad visual se procede a realizar el diseño del sistema domótico para personas con discapacidad visual mediante Inteligencia Artificial y reconocimiento de voz.

3.3.2. Requerimientos del sistema domótico

Se deben tener en cuenta que al tratarse de un Sistema Domótico dirigido a personas con discapacidad visual en el cual se utilizará la Inteligencia Artificial y el reconocimiento de voz, es necesario el uso de un dispositivo con un procesador de capacidad de procesamiento de 1,5 GHz y un soporte de 64 bits que permita que fluya la información de manera rápida como también permita soportar varias palabras para los comandos de voz, ya que cada instrucción permitirá controlar las diferentes áreas de la vivienda.

3.4. Análisis del sistema domótico

Para el Análisis de los sistemas domóticos para personas con discapacidad visual se efectúa en función de su arquitectura y tipo de mando.

3.5. Arquitectura del sistema domótico

El análisis de los sistemas domóticos para personas con discapacidad visual se realiza a través del análisis de tipos de arquitecturas las cuales se detallan en la tabla 8, donde se muestra dos tipos de arquitecturas para sistemas domóticos, con el propósito de seleccionar el mejor.

Una vez realizado la comparación de las arquitecturas para sistemas domóticos, se selecciona la arquitectura centralizada porque solo se necesita un controlador. La ventaja principal de usar este sistema domótico centralizado es que es fácil de realizar cualquier mantenimiento, actualización o modificación sin tener que abrir las tapas de las diferentes cajas de distribución a lo largo de la vivienda. Esto también lo hace conveniente para usar en entornos residenciales, especialmente aquellos donde las distancias desde o hacia los cuadros de control son cortas.

Tabla 8.- Tipos de arquitecturas de sistemas domóticas.

Arquitectura de sistemas domóticos para personas con discapacidad visual		
	Arquitectura Centralizada	Arquitectura Distribuida
Ventajas	Mayor control y coordinación centralizados.	Mayor flexibilidad y escalabilidad.
	Simplificación de la instalación y configuración.	Menor dependencia de un único punto de fallo.
	Mayor eficiencia en el procesamiento y almacenamiento de datos.	Capacidad de recuperación y continuidad del servicio en caso de fallos locales.
	Facilidad para implementar actualizaciones y mejoras.	Menor latencia y tiempos de respuesta más rápidos debido a la distribución de los dispositivos y la capacidad de procesamiento.
Desventajas	Mayor dependencia de un único punto de control.	Mayor capacidad para adaptarse a cambios y expansiones en el sistema sin afectar su funcionamiento global.
	Menor flexibilidad y adaptabilidad a cambios futuros.	Mayor complejidad en la instalación, configuración y mantenimiento del sistema debido a la distribución de los dispositivos y el control.
	Mayor dificultad para realizar modificaciones y personalizaciones.	Posibilidad de conflictos y problemas de comunicación entre los dispositivos distribuidos.
	Posibilidad de sobrecarga en el sistema centralizado.	Mayor consumo de energía debido a la necesidad de mantener la comunicación constante entre los dispositivos distribuidos.
		Requiere una infraestructura de red sólida y confiable para garantizar la comunicación adecuada entre los dispositivos distribuidos.

Fuente: Elaborado por Investigador.

3.5.1. Tipos de mando

Para el análisis de los sistemas domóticos para personas con discapacidad visual se realizó una tabla comparativa de los tipos de mandos, los cuales se pueden emplear en los sistemas domóticos. En la tabla 9 se puede observar las ventajas y desventajas de los tipos de mandos para su mejor selección.

Tabla 9.- Tipos de mandos para el control domótico.

Tipo de Mando	Descripción	Ventajas
Control por voz	Utiliza comandos de voz para controlar el sistema domótico.	Accesibilidad para personas con discapacidad visual, ya que no se requiere interacción física o visual.
Control táctil	Utiliza pantallas y gestos táctiles para controlar el sistema.	Interacción táctil intuitiva y familiar para muchas personas.
Control remoto	Utiliza un dispositivo remoto, como un control remoto infrarrojo, para controlar los dispositivos del sistema.	Control físico a distancia conveniente.
Control mediante gestos	Utiliza sensores de movimiento para reconocer y traducir gestos en comandos para el sistema domótico.	Permite una interacción sin contacto físico directo.
Control mediante braille	Utiliza teclados o paneles con Braille para permitir la interacción con el sistema mediante el tacto.	Accesibilidad para personas con discapacidad visual que conocen el sistema Braille.
Control mediante sonidos	Utiliza tonos o patrones de sonido para activar o controlar los dispositivos del sistema.	Proporciona retroalimentación auditiva para las personas con discapacidad visual.
Control mediante pulsadores	Utiliza botones o pulsadores grandes y de fácil identificación para controlar los dispositivos del sistema.	Accesibilidad para personas con discapacidad visual, ya que se pueden identificar fácilmente los botones grandes y se proporciona retroalimentación táctil al presionarlos.

Fuente: Elaborado por Investigador.

Una vez realizada la comparación de los tipos de mandos existentes se selecciona el control por voz como mando para el sistema domótico, debido a los principales problemas a los que se enfrenta una persona con discapacidad visual, ya que principalmente sus deficiencias no le permiten movilizar con facilidad alguna parte de su cuerpo o realizar alguna tarea dentro del hogar. El control por voz ayudará a facilitar y mejorar su calidad de vida, como también el control por voz posee un lenguaje abierto y flexible para que la persona pueda interactuar de manera fácil con el sistema y de esta manera pueda hacer uso de este sin tener ningún problema.

3.6. Diseño del prototipo

3.6.1. Diagrama de bloques

Se hizo un diagrama de bloques como se muestra en la figura 11 para la implementación del prototipo del sistema domótico para personas con discapacidad visual mediante Inteligencia Artificial y reconocimiento de voz, con el propósito de tener un mejor entendimiento sobre cada una de las fases de este.

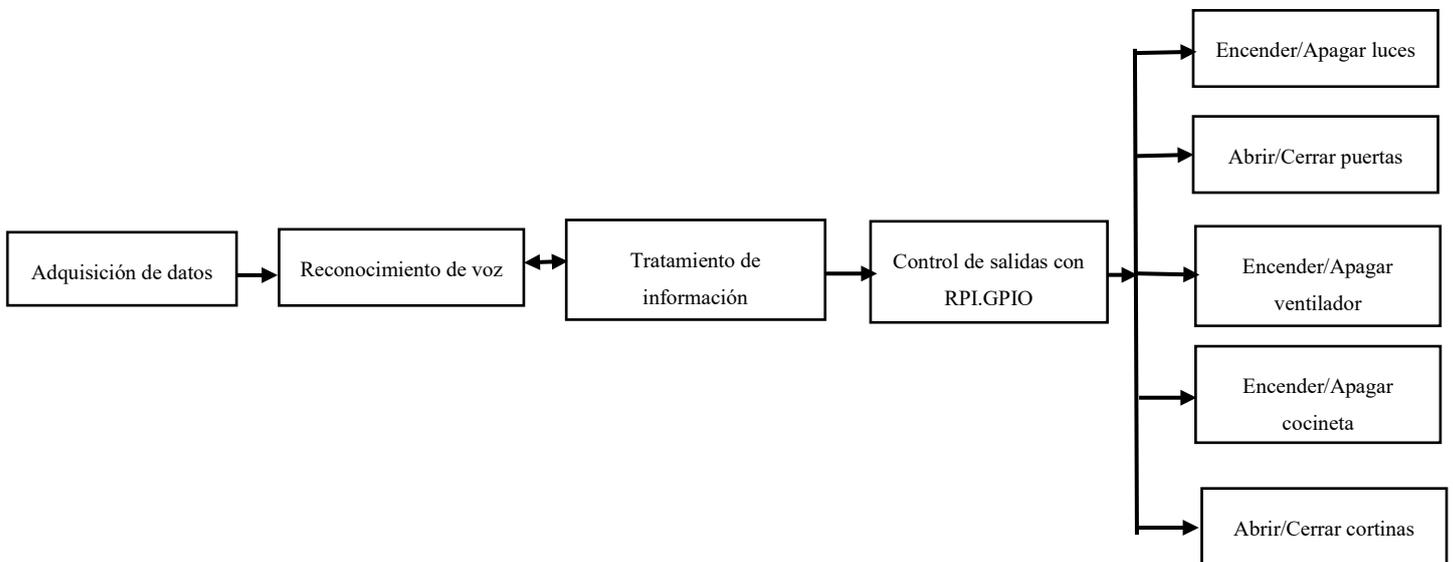


Figura 11.- Diagrama de bloques del sistema domótico.

Fuente: Elaborado por Investigador.

Como se puede observar en el diagrama de bloques del prototipo del sistema domótico, este está constituido por varias etapas que se describen a continuación:

Adquisición de datos: Por medio del Micrófono Bluetooth se adquiere los datos de voz, por el mismo se envían a la Raspberry Pi para el procesamiento de estos.

Reconocimiento de voz: En esta etapa se reconoce que la persona que habla es el usuario con el que se entrenó la voz para el control del sistema, si es la voz del usuario es la correcta continua con la etapa, en caso de que no sea, vuelve a la etapa de adquisición de datos.

Tratamiento de la información: Luego de reconocer el usuario, procede a reconocer los comandos, si es igual al comando determinado en el sistema continua con la siguiente etapa, caso contrario regresa a la etapa de adquisición de datos.

Control de salidas: Cada comando envía a una función que se programa en Python y se ejecuta con la ayuda de la Raspberry y la librería RPI.GPIO, finalmente se realiza la acción detallada en el diagrama de bloques.

El equipo remoto tiene un circuito en la cual están conectadas las entradas y salidas para el control de luces y dispositivos electrónicos, como también la conexión para los motores para el control de puertas y persianas de la vivienda.

3.7. Análisis de las tecnologías móviles

La tabla 10 muestra la comparativa de las distintas tecnologías que existen en el mercado al momento de decidir cuál es la mejor para el control del sistema domótico.

Una vez analizada la tabla 10 de tecnologías móviles para el control del sistema se elige el Bluetooth para el control del sistema domótico, debido a que posee una mayor velocidad de transmisión y un mayor alcance respecto a IRDA, además el Wi-Fi es mucho más complicado y tiene un hardware involucrando con tarjetas Wi-Fi, módems, routers y software; en cambio Bluetooth resulta ser más sencillo al momento de conectar ya que sólo requiere que los dispositivos se enciendan y se enlacen juntos. Además, Bluetooth también requiere un alcance menor que Wi-Fi y por lo tanto es más eficiente en cuanto al consumo de energía.

Tabla 10.- Comparativa entre tecnologías móviles para el control del sistema domótico.

Parámetros	Wifi	Bluetooth	IRDA
Velocidad de TX	11Mb – 248Mb	1Mb – 32Mb	4 Mb
Distancia	30m – 100m	10m – 100m	2m
Medio de TX	Radio Frecuencia	Radio Frecuencia	Infrarrojo
Número de dispositivos	-----	1 a la vez	1 a la vez
Topología	Estrella	Maestro-Esclavo	Maestro-Esclavo
Costo	Medio-Alto	Bajo	Bajo

Fuente: Elaborado por Investigador.

3.8. Análisis Técnico de Hardware requerido

3.8.1. Selección de Hardware

Micrófono Fifine M6

Para la implementación del sistema domótico en la parte del reconocimiento de voz se utilizará un micrófono para el ingreso de los comandos de voz y el entrenamiento de la Red Neuronal, para ello se realiza una comparación de las características técnicas de 2 micrófonos, lo cual se puede ver en la tabla 11.

Analizando la tabla 11 se selecciona el Micrófono Fifine M6 siendo una opción recomendada debido a sus características destacadas. Dentro de las más puntuales se consideran su calidad de sonido nítida y la conexión mediante USB. Por otro lado, está construido con materiales duraderos y es versátil en su uso. Además, su facilidad de conexión y uso lo hace adecuado para una variedad de situaciones, como grabaciones vocales.

Tabla 11.- Características técnicas del micrófono para el reconocimiento de voz.

Características	Micrófono Fifine M6	Micrófono Profesional Lavalier USB Tipo C
Imagen		
Tipo de micrófono	Micrófono de condensador	Micrófono de condensador
Conexión	USB	USB Tipo C
Patrón polar	Cardioide	Omnidireccional
Respuesta de frecuencia	20Hz - 20kHz	50Hz - 16kHz
Sensibilidad	-34dB ± 2dB	-30dB ± 3dB
Relación señal/ruido	≥ 78dB	≥ 70dB
Longitud del cable	Aproximadamente 1.8 metros	Aproximadamente 2 metros
Control de volumen	Sí	No
Monitorización de auriculares	Sí	No
Clip de sujeción	Sí	Sí
Accesorios incluidos	Trípode, filtro de espuma, cable USB	Adaptador USB-C a USB-A, clip de sujeción
Compatibilidad	Windows, Mac, Linux, PlayStation 4	Windows, Mac, Android, dispositivos USB-C
Precio aproximado	Alrededor de \$50	Alrededor de \$30

Elaborado por: Investigador basado en [39]

Modelo de placa Raspberry Pi para Reconocimiento de Voz

De igual manera para el desarrollo del reconocimiento de voz y el entrenamiento de la Red Neuronal del proyecto de investigación se analizó las características técnicas de la Raspberry Pi, la misma que actuará como microcontrolador genera. En la tabla 12 se puede observar algunos modelos de placas y sus características.

Tabla 12.- Cuadro comparativo de Raspberry PI.

Características	Raspberry Pi 3	Raspberry Pi 4 Modelo A	Raspberry Pi 4 Modelo B
Imagen			
CPU	Broadcom BCM2837B0, Cortex-A53	Broadcom BCM2711, Cortex-A72	Broadcom BCM2711, Cortex-A72
Núcleos de CPU	4 (Quad-core)	4 (Quad-core)	4 (Quad-core)
Velocidad de CPU	1.2 GHz	1.5 GHz	1.5 GHz
RAM	1 GB DDR2 SDRAM	2 GB LPDDR4 SDRAM	2 GB / 4 GB / 8 GB LPDDR4 SDRAM
GPU	VideoCore IV	VideoCore VI	VideoCore VI
Resolución de video	Hasta 1080p	Hasta 4K a 60Hz	Hasta 4K a 60Hz
Almacenamiento	Ranura para tarjeta microSD	Ranura para tarjeta microSD	Ranura para tarjeta microSD
Puertos USB	4 x USB 2.0	2 x USB 3.0, 1 x USB 2.0	2 x USB 3.0, 2 x USB 2.0
Conexión de red	Ethernet, Wi-Fi 802.11n	Ethernet, Wi-Fi 802.11ac	Ethernet, Wi-Fi 802.11ac
Bluetooth	Bluetooth 4.2	Bluetooth 5.0	Bluetooth 5.0
GPIO	40 pines GPIO	40 pines GPIO	40 pines GPIO

Salida de video	HDMI, RCA	HDMI, DSI	HDMI, DSI
Consumo de energía	2.5W	3.0W	3.0W
Precio	Menor	Medio	Mayor

Elaborado por: Investigador basado en [40]

Analizando la tabla 12, se puede observar que los avances se han ido dando a través del tiempo en la Raspberry PI, la Raspberry PI 3 es 10 veces superior a la tarjeta Raspberry original; sin embargo, se escogió la Raspberry PI 4 ya que cuenta con características acorde para un sistema domótico como también otro punto a favor es que su compatibilidad con distribuciones de Windows, cuenta con conexión wifi y bluetooth, además es la placa más potente y rápida en comparación con sus antecesoras, también cuenta con entradas USB necesarias para el desarrollo del sistema de reconocimiento de voz.

Servomotor

Para la realización del sistema de control de persianas se utilizó dos servomotores por lo cual se realizó una comparación de las características técnicas de dos tipos de servomotores que se detallan en la tabla 13.

Tabla 13.- Características técnicas de servomotores.

Características	Servomotor MG996R	Servomotor Sg90
Imagen		
Voltaje de operación	4.8V - 7.2V	4.8V - 5.0V

Torque máximo	0.9218 Nm	0.1765 Nm
Velocidad de rotación	0.17 s/60°	0.12 s/60°
Ángulo de rotación	180°	180°
Tipo de motor	Motor de engranajes	Motor de engranajes
Tipo de conexión	Cable conector de 3 pines	Cable conector de 3 pines
Dimensiones	40.7mm x 19.7mm x 42.9mm	22.2mm x 11.8mm x 31 mm
Peso	55g	9g
Aplicaciones	Robótica, domótica, modelismo, mecanismos	Domótica, pequeños proyectos, modelismo, juguetes

Elaborado por: Investigador basado en [41].

Analizando la tabla 13 se puede observar que el servomotor MG996R es mucho más grande y tiene un torque mucho más elevado que el servomotor Sg90 ya que sus aplicaciones básicamente es para proyectos más grandes en los cuales su función sea levantar más peso, es por ello que se eligió el servomotor Sg90 ya que es una opción recomendada por las siguientes razones; tiene un torque máximo de 0.1765 nanómetros suficiente para cumplir la función de biela manivela para poder abrir y cerrar las persianas de la vivienda. Es más pequeño, lo que le brinda mayor estabilidad y resistencia en proyectos domóticos. Por último, es un modelo común y fácil de encontrar en el mercado, compatible con diversos controladores y sistemas.

Ventilador DC

Para la realización del sistema de control de temperatura en la vivienda se utilizó un ventilador, por lo cual se realizó una comparación de las características técnicas de dos tipos de ventiladores que se detallan en la tabla14.

Tabla 14.- Características técnicas de ventiladores.

Características	Ventilador DC de 5V	Ventilador DC de 12V
Imagen		
Voltaje de operación	5V	12V
Consumo de energía	Menor	Mayor
Velocidad de rotación	Generalmente más baja	Generalmente más alta
Flujo de aire	Menor	Mayor
Ruido	Generalmente más silencioso	Generalmente más ruidoso
Compatibilidad	Ampliamente utilizado en dispositivos	Ampliamente utilizado en dispositivos
Tamaño y dimensiones	Varía según el modelo	Varía según el modelo
Uso común	Dispositivos portátiles, electrónica	Ordenadores, sistemas de refrigeración

Elaborado por: Investigador basado en [42].

Analizando la tabla 14 descrita con antelación se eligió el ventilador de 5V, pues dentro de la implementación del prototipo del sistema domótico se necesita de una cantidad reducida de energía ya que el sistema trabaja con una fuente de 5V, además este tiene un menor nivel de ruido, mayor seguridad y facilidad de alimentación.

Relé

Para la realización del sistema de control de puertas en la vivienda se utilizó relés para que cumplan la función de una cerradura magnética, por lo cual se realizó una

comparación de las características técnicas de 3 tipos de relés que se detallan en la tabla 15.

Tabla 15.- Características técnicas de los relés.

Características	Relevador 5V DC SRD-5VDC-SL-C	Relé 5V SPDT	Relés miniatura serie RT424
Imagen			
Tensión	5V	5V	5V
Corriente máx.	10A	5A	20A
Tipo de carga	Resistiva	Resistiva	Resistiva
Número de pines	4	6	5
Tipo de conexión	PCB	PCB	Conector
Señal de control	5V TTL	5V TTL	5V TTL
Consumo de energía	80mA	50mA	100mA
Tiempo de respuesta	10ms	5ms	15ms

Elaborado por: Investigador basado en [43].

Analizando la tabla 15 se escogió el relé de 5V SPDT, pues tiene un tiempo de respuesta rápido, el consumo de energía es mínimo en comparación a los otros dos y la corriente máxima que soporta es de 5A, es compatible con la fuente del sistema y suficiente para la implementación del sistema de acceso para la vivienda.

3.9. Diagrama de bloques del sistema domótico

Analizando la selección de hardware requerido para el sistema domótico se realiza el diagrama de bloques como se puede observar en la figura 12, en la cual se detalla cómo funciona el sistema domótico con los materiales descritos anteriormente.

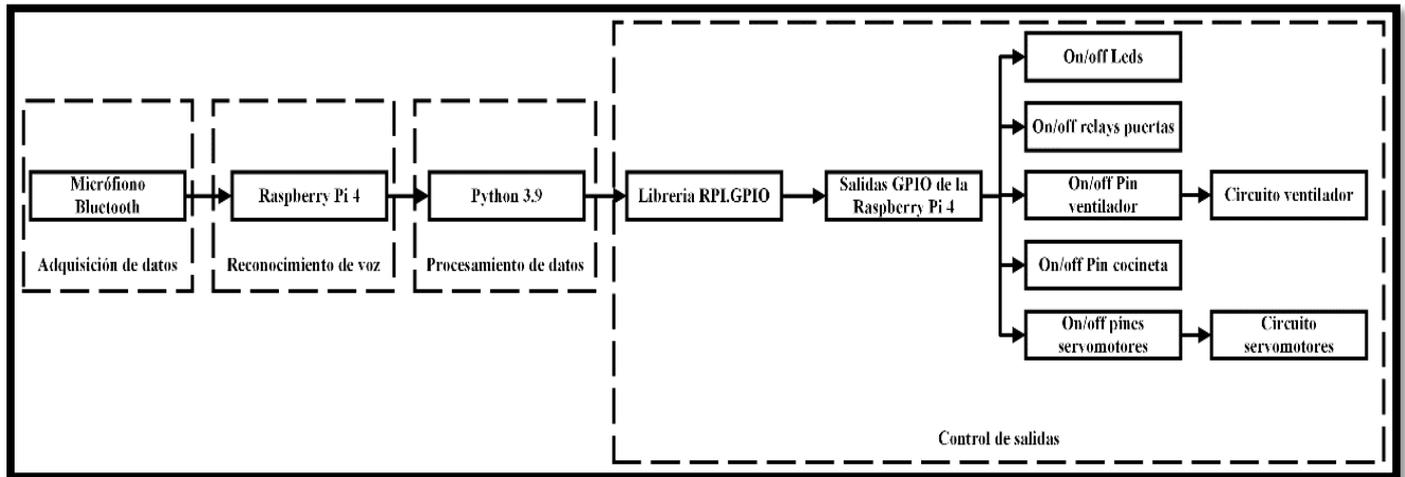


Figura 12.- Diagrama de bloques para el sistema domótico.

Fuente: Elaborado por Investigador.

En este diagrama de bloques se detalla cada uno de los dispositivos a utilizarse en cada etapa del sistema y que son las siguientes:

Para el ingreso de datos, se utiliza un micrófono fífine M6 que se conecta mediante un adaptador USB a la placa utilizada en el reconocimiento de voz.

Para la siguiente etapa que es de reconocimiento de voz se utiliza la placa de hardware libre que es la Raspberry PI 4. Esta placa se encarga de recibir los comandos de voz, compararlos con el modelo entrenado, verifica que sea el usuario correcto el que está ingresando los comandos y procede a compararlos con la base de datos.

Para la siguiente etapa de procesamiento de datos se utiliza de igual manera la Raspberry PI 4, que está encargada de la concentración de cada función que se desarrollará según las instrucciones recibidas de la etapa de reconocimiento de voz que está conectada mediante el módulo bluetooth propio de la placa hacia el micrófono, también recibirá datos de los sensores, motores y todos los dispositivos electrónicos implementados en el sistema domótico. Las instrucciones recibidas se procesan y la placa de control se encarga de activar el sistema por los puertos RPI.GPIO.

La etapa de control de salidas se enlaza con la librería RPI.GPIO, la misma que activa o desactiva los pines de la placa, la salida en high significa que están activas y salidas en low significan que están desactivadas.

3.10. Diseño del circuito electrónico

Se realizó el diagrama esquemático del circuito electrónico que se indica en la figura 13, donde se muestra la conexión de todos los circuitos del sistema domótico en el cual se puede observar cómo se comunica el reconocimiento de voz con el circuito de control a través de los pines de la Raspberry PI 4, este circuito también conecta los servomotores para el control de las persianas, el sistema de iluminación, el sistema de control de puertas y también el sistema de ventilación.

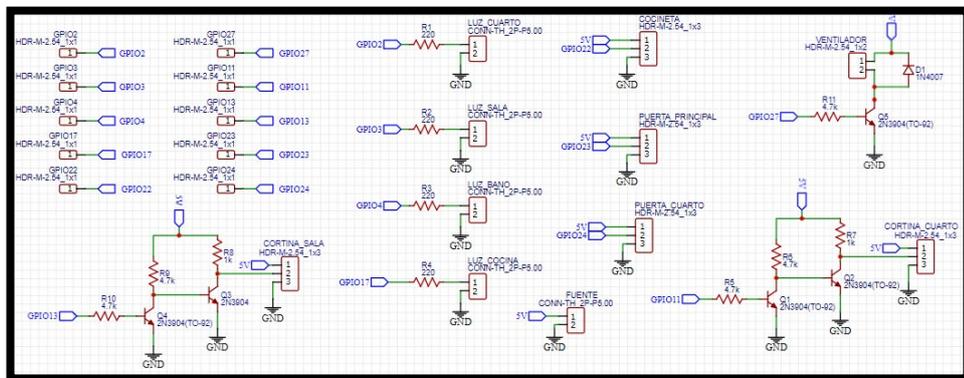


Figura 13.- Circuito electrónico para el control domótico.

Fuente: Elaborado por Investigador.

Como se puede observar en esta parte del circuito está el control de las luces que están representadas por leds, que van desde el pin GPIO conectadas cada una con una resistencia para no quemar cada led.

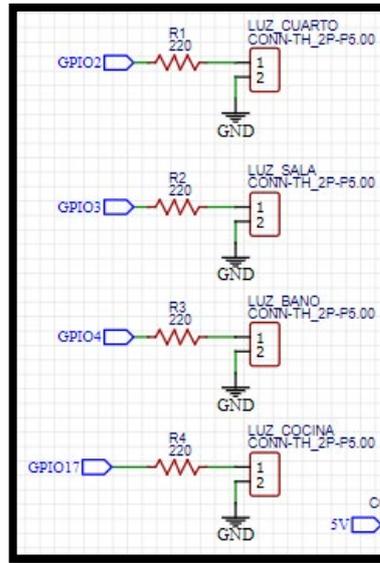


Figura 14.- Sistemas de control de iluminación.

Fuente: Elaborado por Investigador.

De igual manera se tiene un circuito para el control de las persianas, este circuito está constituido por dos servomotores que trabajan mediante un pulso PWM que se activa cuando se envía la señal desde la Raspberry PI.

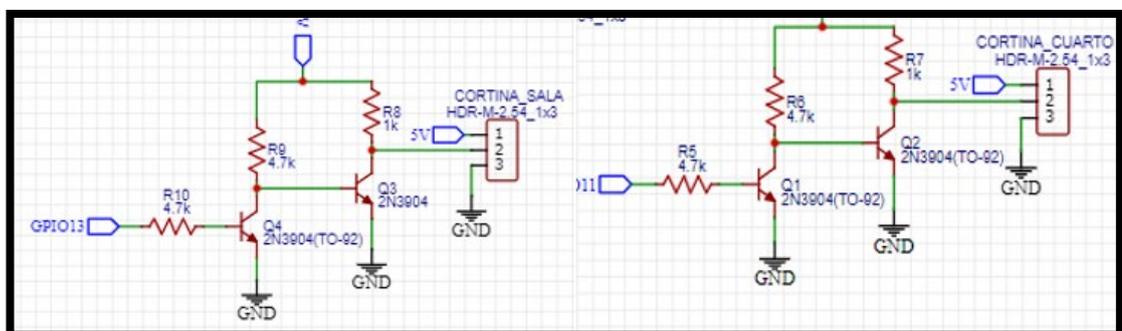


Figura 15.- Circuito para el control de servomotores.

Fuente: Elaborado por Investigador.

De la misma manera se tiene un circuito para el control de ventilación, este circuito está compuesto por un ventilador de 5V conectado a una fuente externa del sistema para que no exista una corriente de retorno y pueda quemar la Raspberry PI, para que el ventilador se active lo que se realiza es el envío de una señal desde la placa de control.

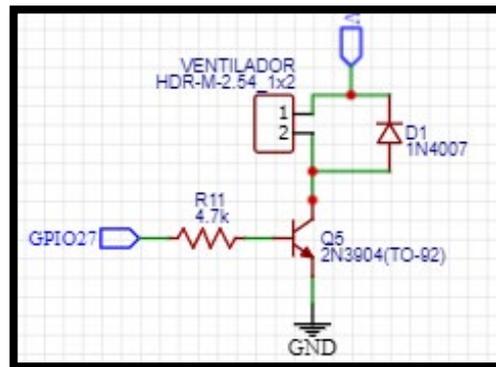


Figura 16.- Circuito de sistema de ventilación.

Fuente: Elaborado por Investigador.

Finalmente se realizó un circuito para el control de las puertas de la vivienda, este circuito está constituido por relés de 5V que simulan a las cerraduras magnéticas, para que el circuito se active se envía una señal desde la Raspberry Pi para que el relé genere el campo magnético y la puerta se mantenga cerrada.

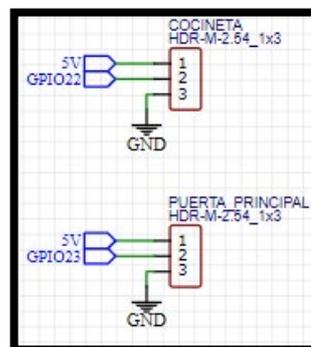


Figura 17.- Circuito de control de puertas de la vivienda.

Fuente: Elaborado por Investigador

3.11. Software

La instalación de las siguientes herramientas de software que se especifican a continuación son de suma importancia para el desarrollo del sistema de reconocimiento de voz.

3.11.1. Reconocimiento de voz

Para realizar el sistema de reconocimiento de voz se utiliza directamente la tarjeta de software libre la Raspberry Pi 4 a la cual se le instaló la imagen de Raspbian.

3.11.2. Instalación de la imagen Raspbian para Raspberry Pi 4

Raspbian es un sistema operativo para dispositivos como la Raspberry PI 4, este sistema está orientado para el desarrollo del Internet de las cosas IoT, y también el desarrollo de proyectos con Inteligencia Artificial. Además, cuenta con una interfaz accesible para poder trabajar fácilmente en el desarrollo de proyectos. Para la instalación del sistema operativo Raspbian se siguió los siguientes pasos:

Descarga de la imagen de Raspberry Pi desde el sitio web oficial de la Raspberry Pi (www.raspberrypi.org).

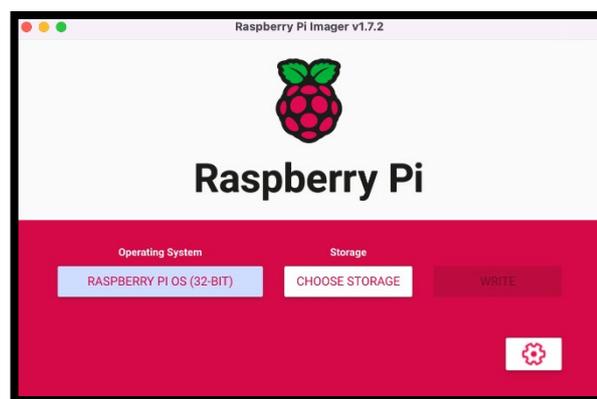


Figura 18.- Instalación de la imagen de Raspbian.

Fuente: Elaborado por Investigador.

Se procede a realizar un formateo de discos propio de la PI Imager después de conectar la tarjeta microSD a la computadora.

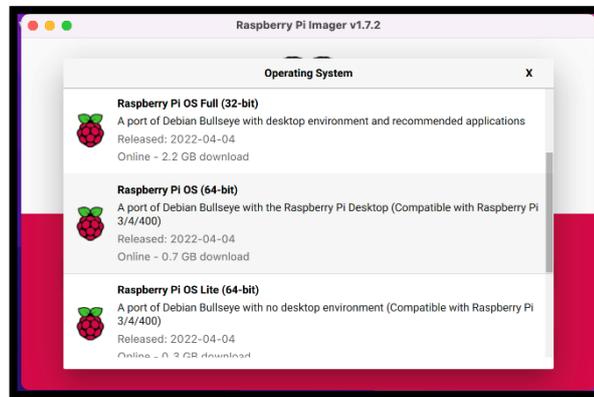


Figura 19.- Interfaz de Pi Imager.

Fuente: Elaborado por Investigador.

Después se procede a flashear la imagen de Raspbian, para ello se utiliza un programa como Etcher, que está disponible en etcher.io. Se selecciona el destino de la imagen descargada de Raspbian y la tarjeta microSD. Durante este proceso se eliminarán todos los datos de la tarjeta.



Figura 20.- Flasheando la tarjeta SD.

Fuente: Elaborado por Investigador.

Después de grabar la imagen en la tarjeta microSD, se procede a retirarla de la computadora y conectarla en la Raspberry PI 4.

Se procede a la activación de los puertos que se van a utilizar como son el monitor, el teclado y el mouse a la Raspberry Pi 4. Para poder utilizar los puertos los dispositivos deben ser compatibles con la Raspberry PI 4 como cables HDMI, USB y otros.

Luego se procede a encender la Raspberry PI 4 y comenzará a cargar el sistema operativo desde la tarjeta microSD.

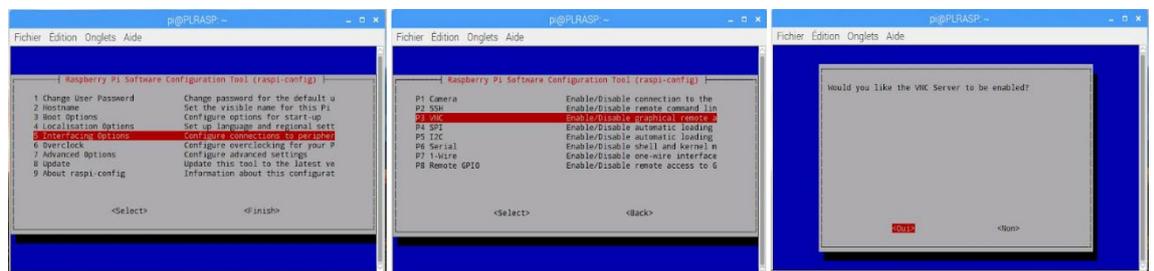


Figura 21.- Configuración de la Raspberry.

Fuente: Elaborado por Investigador.

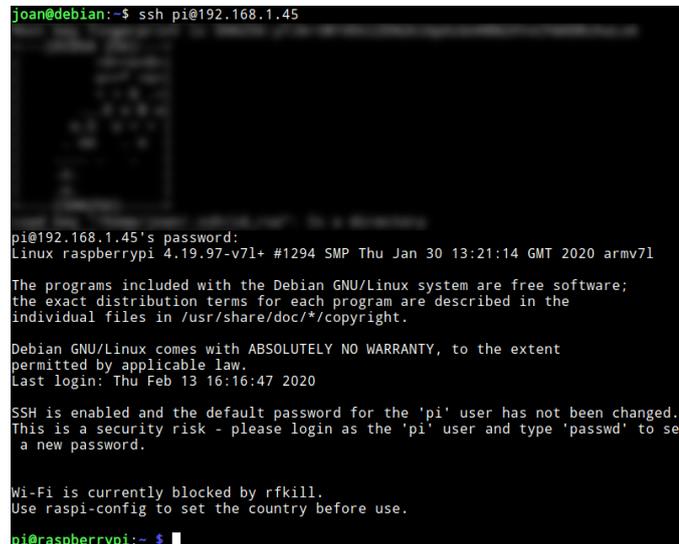
Después de encender la Raspberry PI 4 se continua con el procedimiento de configuración inicial. La configuración del idioma, el nombre de host, la contraseña del usuario y la configuración de la red que se encuentran en el sistema operativo.



Figura 22.- Configuración de sistema.

Fuente: Elaborado por Investigador.

A continuación, se realiza actualizaciones del sistema operativo, para ello se procede a abrir una terminal en Raspbian y ejecutar los comandos `sudo apt update` y `sudo apt upgrade`.



```
joan@debian:~$ ssh pi@192.168.1.45
pi@192.168.1.45'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 Feb 13 16:16:47 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:~$
```

Figura 23.- Actualización del sistema operativo.

Fuente: Elaborado por Investigador.

3.11.3. Instalar las librerías

En el sistema operativo Raspbian viene incluido lo que es Python 3.9, por lo que solo se necesita instalar las librerías de reconocimiento de voz y control de salidas de la Raspberry Pi 4, para ello hay que seguir los siguientes pasos:

Se procede a instalar las dependencias necesarias para poder utilizar las bibliotecas. Se ingresa desde la terminal del sistema operativo el siguiente comando “`pip install pyAudio`” para instalar la librería de pyAudio como se muestra en la figura 24.

```
BALAJI@DESKTOP-867PKRV MINGW64 ~/desktop/google-assistant
$ pip install pyAudio-0.2.11-cp38-cp38-win_amd64.whl
Processing c:\users\balaji\desktop\google-assistant\pyaudio-0.2.11-cp38-cp38-win_amd64.w
hl
Installing collected packages: pyAudio
Successfully installed pyAudio-0.2.11

BALAJI@DESKTOP-867PKRV MINGW64 ~/desktop/google-assistant
$ █
```

Figura 24.- Instalación de la librería pyAudio.

Fuente: Elaborado por Investigador.

A continuación, se ingresa el comando “pip3 install SpeechRecognition” desde la terminal del sistema operativo para instalar la librería de SpeechRecognition en la Raspberry PI 4 como se muestra en la figura 25.

```
PS C:\Users\User\AppData\Local\atom\app-1.45.0\Projects>
PS C:\Users\User\AppData\Local\atom\app-1.45.0\Projects> pip install SpeechRecognition
Using cached SpeechRecognition-3.8.1-py2.py3-none-any.whl (32.8 MB)
Installing collected packages: SpeechRecognition
Successfully installed SpeechRecognition-3.8.1
```

Figura 25.- Instalación de la librería SpeechRecognition.

Fuente: Elaborado por Investigador.

Después se ingresa desde la terminal del sistema operativo el siguiente comando “pip3 install RPi.GPIO” para instalar la librería RPi.GPIO en la Raspberry PI 4 como se muestra en la figura 26.

```
pi@raspberrypi ~$ sudo pip install rpi.gpio
Downloading/unpacking rpi.gpio
  Running setup.py egg_info for package rpi.gpio

Installing collected packages: rpi.gpio
  Running setup.py install for rpi.gpio
    building 'RPi.GPIO' extension
      gcc -pthread -fno-strict-aliasing -DNDEBUG -g -fwrapv -O2 -Wall -Wstrict-pro
totypes -fPIC -I/usr/include/python2.7 -c source/py_gpio.c -o build/temp.linux-ar
mv61-2.7/source/py_gpio.o
      gcc -pthread -fno-strict-aliasing -DNDEBUG -g -fwrapv -O2 -Wall -Wstrict-pro
totypes -fPIC -I/usr/include/python2.7 -c source/c_gpio.c -o build/temp.linux-ar
mv61-2.7/source/c_gpio.o
      gcc -pthread -shared -Wl,-O1 -Wl,-Bsymbolic-functions -Wl,-z,relro build/tem
p.linux-armv61-2.7/source/py_gpio.o build/temp.linux-armv61-2.7/source/c_gpio.o
-o build/lib.linux-armv61-2.7/RPi/GPIO.so

Successfully installed rpi.gpio
```

Figura 26.- Instalación de la librería RPi.GPIO.

Fuente: Elaborado por Investigador.

Finalmente, para asegurarse de que las bibliotecas se hayan instalado correctamente, se procede a ejecutar desde la terminal un script de prueba como se muestra en la figura 27. Si no se producen errores y se visualiza el mensaje "Bibliotecas instaladas correctamente" en la terminal, significa que las bibliotecas SpeechRecognition y RPi.GPIO se han instalado correctamente en la Raspberry Pi.

```
pi@raspberrypi:~$ sudo python
Python 2.7.9 (default, Sep 17 2016, 20:26:04)
[GCC 4.9.2] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import RPi.GPIO as GPIO
```

Figura 27.- Comprobación de la instalación de librerías.

Fuente: Elaborado por Investigador.

Programación de la Red Neuronal para el reconocimiento de voz

Para la programación de la Red Neuronal de igual manera se lo realizó en Python 3.9 propio de la imagen de Raspbian, para ello se comienza con la importación de las librerías necesarias como se muestra en la figura 28.

```
import os

from IPython import display
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd

import tensorflow as tf
import tensorflow_hub as hub
import tensorflow_io as tfio
```

Figura 28.- Librerías necesarias para el entrenamiento de voz.

Fuente: Elaborado por Investigador.

Después, se procede a cargar los audios previamente grabados con la persona con discapacidad visual quien será la que controle el sistema domótico como se observar en la figura 29.

```
@tf.function
def load_wav_16k_mono(filename):
    """ Load a WAV file, convert it to a float tensor, resample to 16 kHz
    file_contents = tf.io.read_file(filename)
    wav, sample_rate = tf.audio.decode_wav(
        file_contents,
        desired_channels=1)
    wav = tf.squeeze(wav, axis=-1)
    sample_rate = tf.cast(sample_rate, dtype=tf.int64)
    wav = tfio.audio.resample(wav, rate_in=sample_rate, rate_out=16000)
    return wav
```

Figura 29.- Carga de archivos para el entrenamiento.

Fuente: Elaborado por Investigador.

A continuación, se procede a comprobar que los audios estén subidos correctamente, como se observa en la figura 30.

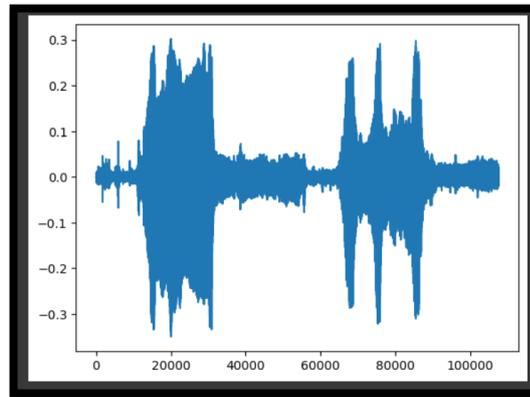


Figura 30.- Comprobación de audios.

Fuente: Elaborado por Investigador.

A continuación, se procede a utilizar YAMNet para la verificación de las puntuaciones a nivel de fotograma por clase. Las puntuaciones se agregan por clase a través de los fotogramas para determinar las predicciones a nivel de clip. Finalmente, se sumaron las mejores puntuaciones para determinar la clase con la mejor puntuación a nivel de clip como se muestra en la figura 31, esto se realiza para verificar que el sistema reconozca los comandos de voz cargados en el entrenamiento.

```
scores, embeddings, spectrogram = yamnet_model(testing_wav_data)
class_scores = tf.reduce_mean(scores, axis=0)
top_class = tf.math.argmax(class_scores)
inferred_class = class_names[top_class]

print(f'The main sound is: {inferred_class}')
print(f'The embeddings shape: {embeddings.shape}')
```

Figura 31.- Puntuaciones del sistema.

Fuente: Elaborado por Investigador.

Finalmente, se procede a realizar un testeo del modelo como se muestra en la figura 32, en el cual se muestra el nombre del audio al que se parece en los archivos de la base de la Red Neuronal.

```
# Run the model, check the output.
scores, embeddings, spectrogram = yamnet_model(waveform)
class_scores = tf.reduce_mean(scores, axis=0)
top_class = tf.math.argmax(class_scores)
inferred_class = class_names[top_class]
top_score = class_scores[top_class]
print(f'[YAMNet] The main sound is: {inferred_class} ({top_score})')

reloaded_results = reloaded_model(waveform)
your_top_class = tf.math.argmax(reloaded_results)
your_inferred_class = my_classes[your_top_class]
class_probabilities = tf.nn.softmax(reloaded_results, axis=-1)
your_top_score = class_probabilities[your_top_class]
print(f'[Your model] The main sound is: {your_inferred_class} ({your_top_score})')
```

Figura 32.- Testeo de la Red Neuronal.

Fuente: Elaborado por Investigador.

En la figura 33 se puede observar el resultado obtenido al momento de ingresar el comando para abrir la puerta del cuarto, como se puede apreciar en la imagen la red neuronal esta funcionando correctamente.

```
[YAMNet] The main sound is: abrir_puerta_cuarto (0.5838820338249207)
[Your model] The main sound is: abrir_puerta_cuarto (0.9897232055664062)
```

Figura 33.- Comprobación de la Red Neuronal al ingresar un comando.

Fuente: Elaborado por Investigador.

3.12. Desarrollo de la programación del sistema domótico

Para el desarrollo del sistema se procede a realizar la programación de las diferentes etapas del sistema domótico.

3.12.1. Programación del sistema para el reconocimiento de voz

Para la programación del sistema para el reconocimiento de voz se realizó un programa en Python que se divide en 5 etapas:

- Grabación de audio mediante el micrófono bluetooth.
- Análisis del audio.
- Comparación de la voz con la que se entrenó el sistema.
- Comparación de los datos convertidos a STRING para determinar si son los mismos que los comandos planteados.
- Envío de la orden a los puertos GPIO.

El programa recibe los comandos de voz que se envían mediante el micrófono bluetooth para después proceder a analizarlos con el sistema de entrenamiento de la Red Neuronal, la cual compara si corresponde a la voz entrenada para el sistema y después procede a convertirlos en texto para comparar cada palabra con la base de datos guardada, para proceder a verificar si todas las palabras enviadas coinciden y finalmente enviar los datos al puerto GPIO de la Raspberry PI, la cual es la encargada del control del sistema.

La tabla 16 muestra cada uno de los comandos entrenados para el sistema domótico las palabras claves y la acción que debe realizar el sistema al momento de ingresar cada comando.

Tabla 16.- Comandos de voz.

Palabra 1	Palabra 2	Palabra 3	Acción por acatar
Abrir	Puerta	Principal	Se abre la puerta principal de la casa.
		Cuarto	Se abre la puerta del cuarto de la casa.
	Cortina	Sala	Se abre la cortina de sala de la casa.
		Cuarto	Se abre la cortina del cuarto de la casa.
Cerrar	Puerta	Principal	Se cierra la puerta principal de la casa.
		Cuarto	Se cierra la puerta del cuarto de la casa.
	Cortina	Sala	Se cierra la cortina de la sala de la casa.
		Cuarto	Se abre la cortina del cuarto de la casa.
Encender	Luz	Cuarto	Se enciende la luz del cuarto de la casa.
		Sala	Se enciende la luz de la sala de la casa.
		Baño	Se enciende la luz del baño de la casa.
		Cocina	Se enciende la luz de la cocina de la casa.
	Ventilador	-----	Se enciende el ventilador de la casa.
	Cocineta	----	Se enciende la cocineta de la casa.
Apagar	Luz	Cuarto	Se apaga la luz del cuarto de la casa.
		Sala	Se apaga la luz de la sala de la casa.
		Baño	Se apaga la luz del baño de la casa.
		Cocina	Se apaga la luz de la cocina de la casa.
	Ventilador	-----	Se apaga el ventilador de la casa.

Cocineta	-----	Se apaga la cocineta de la casa.
----------	-------	----------------------------------

Fuente: Elaborado por Investigador.

Con la pronunciación de las palabras claves que se muestran en la tabla 16 se puede controlar las diferentes áreas de la vivienda, como por ejemplo el sistema de iluminación, el sistema de acceso, el sistema de persianas y electrodomésticos, siempre y cuando la voz que ingrese los comandos sea de la persona con la que se entrenó la Red Neuronal.

Diagrama de Flujo

El diagrama de flujo que se puede observar en la figura 24 detalla el flujograma de la programación del reconocimiento de voz que se realizó en la Raspberry PI 4 por medio de la imagen de Raspbian que viene con las librerías de Python.

El lazo principal se encarga de: abrir el archivo de audio, inicializar las variables, obtener datos del audio enviado por el micrófono bluetooth, después el programa empieza a analizar lo que está grabando, como se detalla en la figura 34. La librería Speech Recognizer convierte el archivo de audio en texto y la información se guarda en la variable "voz", luego se compara palabra por palabra con el programa de la base de datos para después enviar los códigos correspondientes a los comandos de voz de manera que las funciones definidas reciban los comandos para activar o desactivar los puertos GPIO, como se muestra en la figura 35, si las palabras mencionadas no corresponden o el orden de ellas no es el indicado el programa regresa y finaliza.

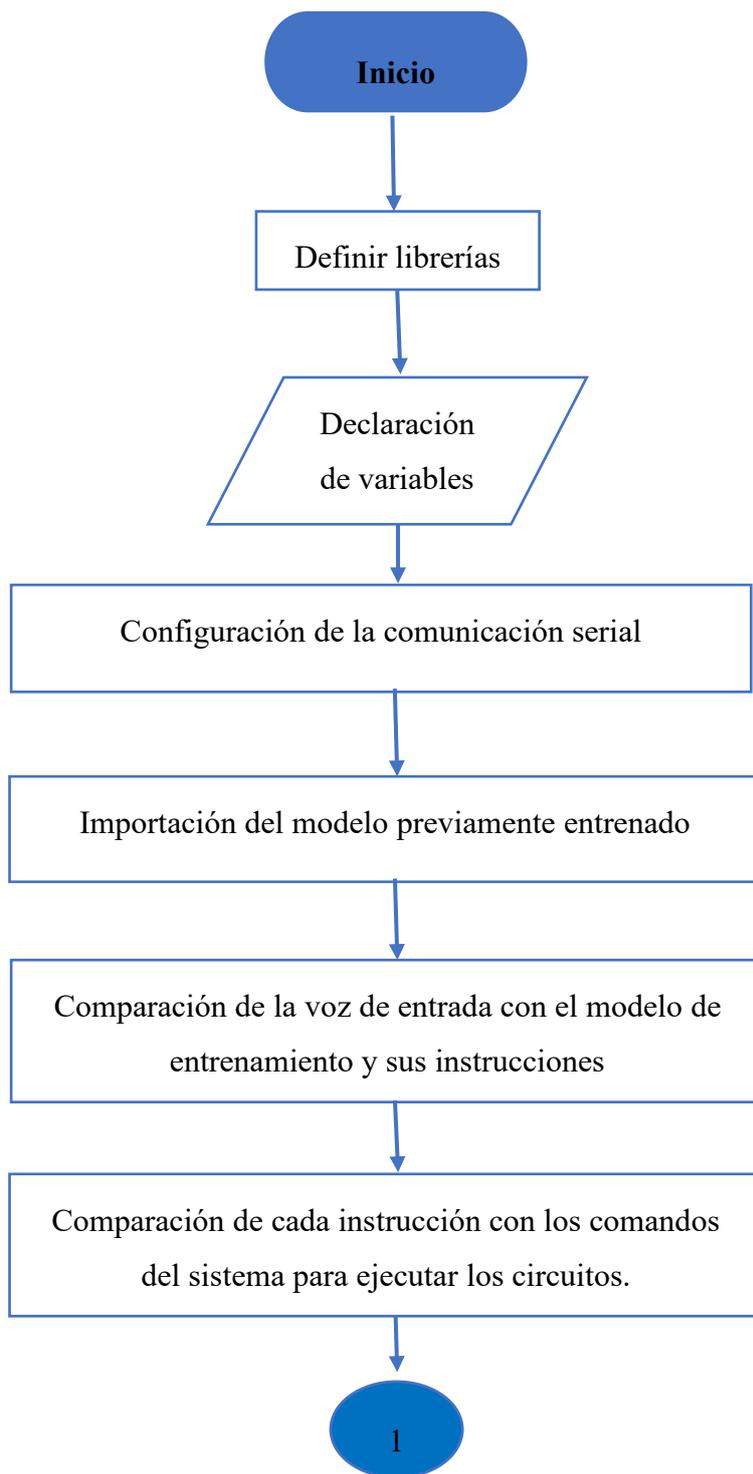


Figura 34.- Lazo principal del diagrama de flujo del Reconocimiento de Voz.

Fuente: Elaborado por Investigador

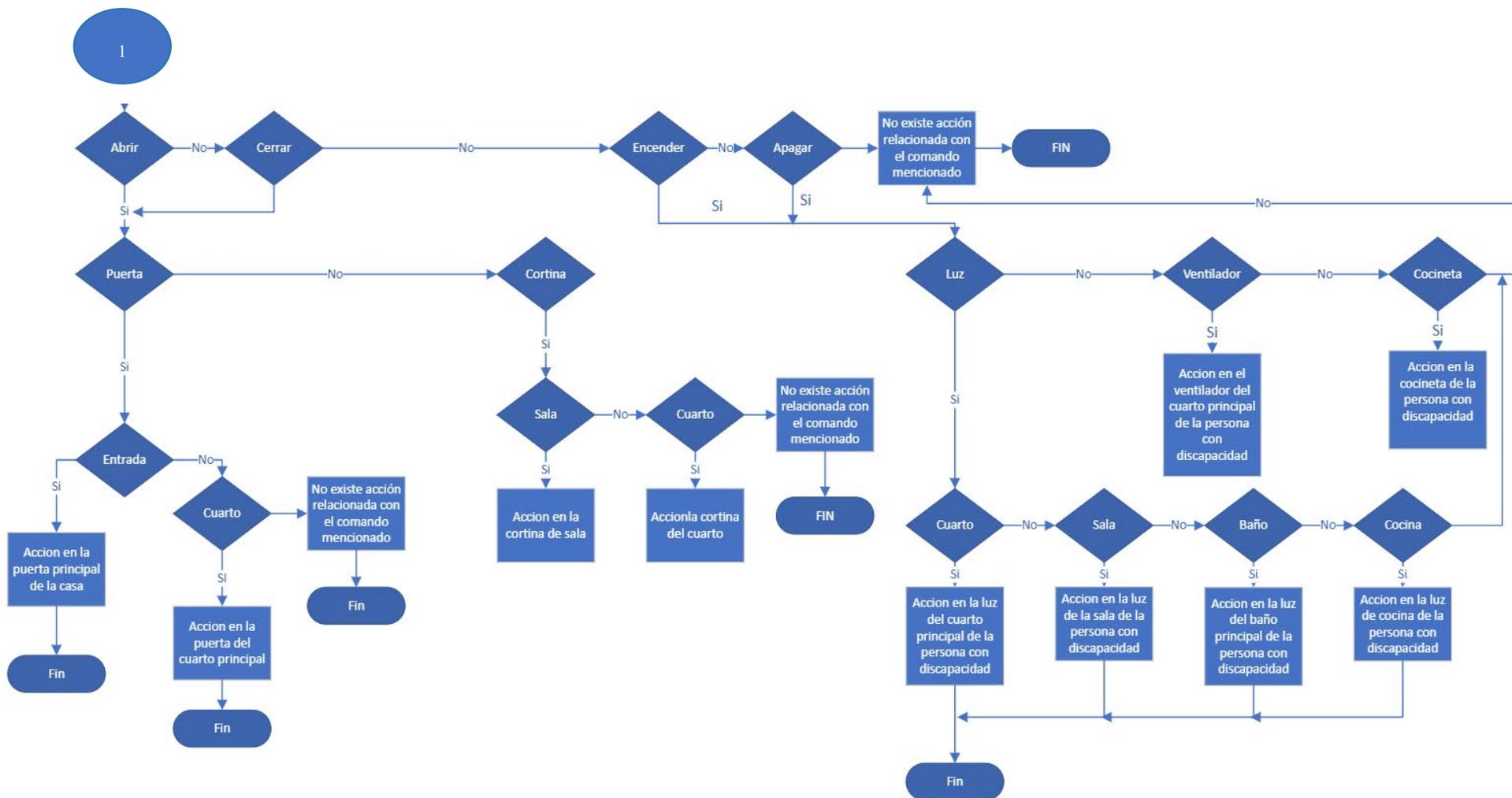


Figura 35.- Diagrama de flujo del Reconocimiento de Voz.

Fuente: Elaborado por Investigador.

3.12.2. Implementación del Prototipo del sistema domótico

Una vez ya programadas las diferentes etapas del sistema domótico para personas con discapacidad visual se procede a la implementación del prototipo del sistema domótico de la siguiente manera:

Implementación de la placa de circuitos

Para la implementación de los diferentes circuitos del sistema domótico se realizó el diagrama electrónico PCB en el software Proteus versión 8.12, como se muestra en la figura 36.

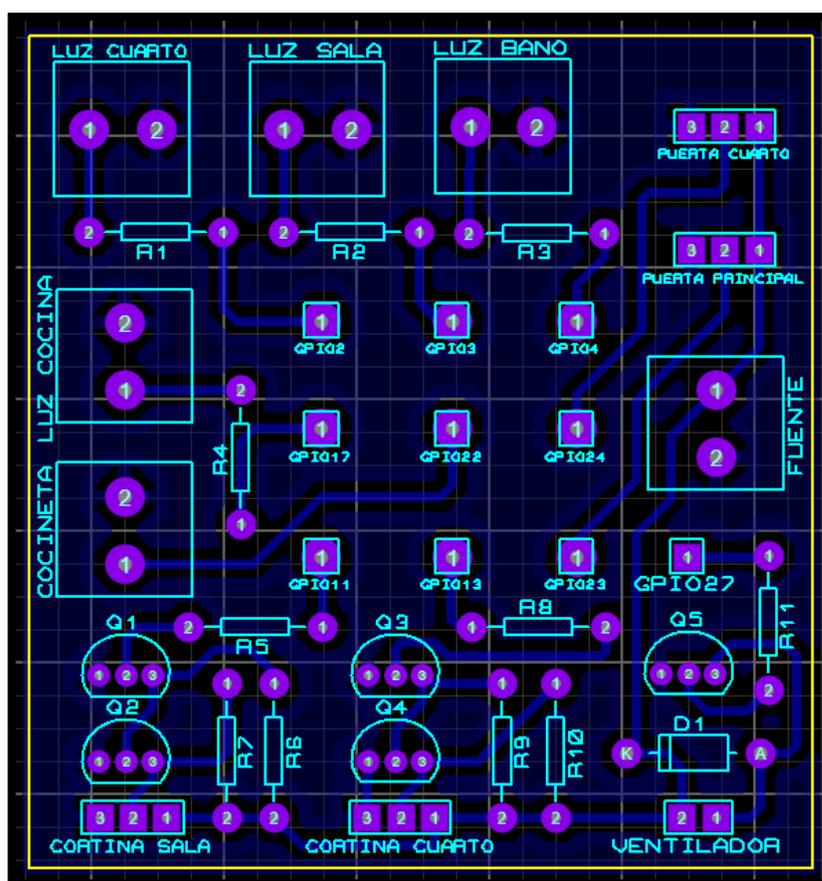


Figura 36.- Diseño de la placa PCB.

Fuente: Elaborado por Investigador.

En la figura 36 se muestra el diseño que se va a plasmar en la placa de baquelita para proceder a ensamblar el circuito integrado I/O. En este circuito consta el sistema de control de los servomotores para las persianas, el sistema de control de acceso para las puertas de la vivienda, el sistema de iluminación para el control de las luces de la vivienda y todos los equipos que conforman el sistema domótico.

La figura 37 muestra el esquema en 3D del circuito PCB realizado en el software Proteus 8.12, de esta manera quedará la placa una vez que se haya implementado todos los dispositivos electrónicos.

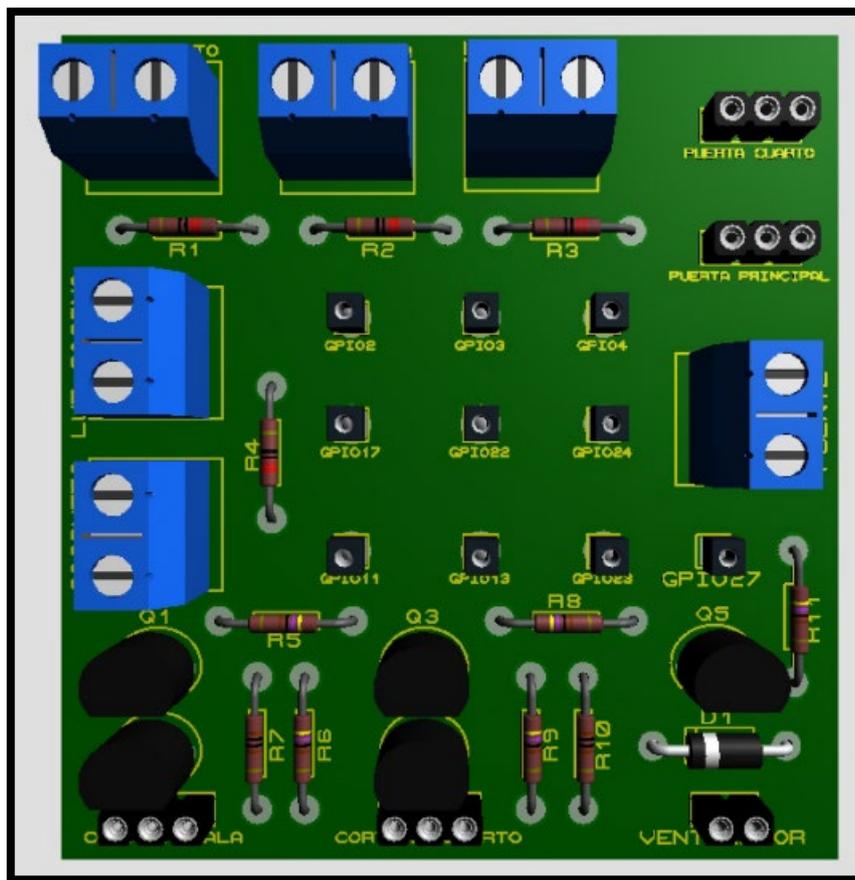


Figura 37.- Parte posterior de la placa PCB.

Fuente: Elaborado por Investigador.

Luego de diseñar el circuito PCB se procede a elaborar la placa de forma física, para lo cual se utilizó el método de planchado y ácido para quitar el exceso de cobre, el resultado se muestra en la figura 38.

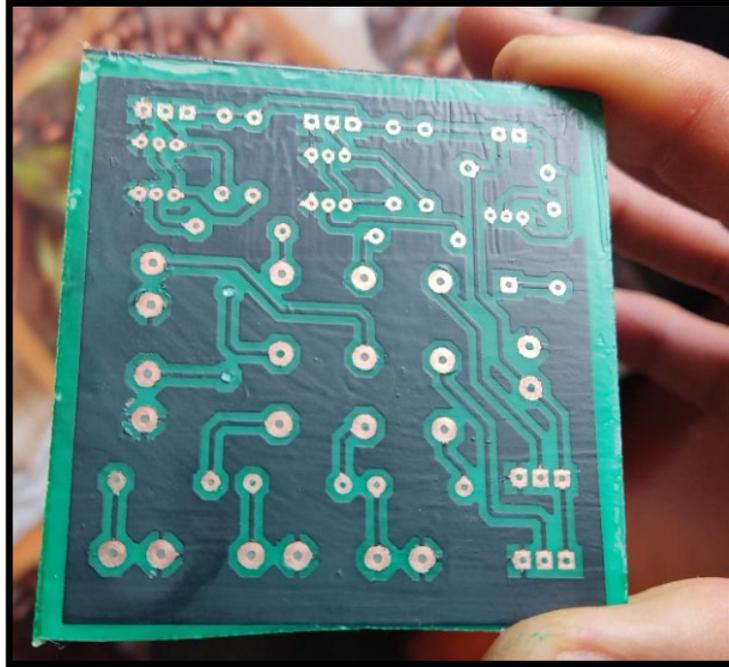


Figura 38.- Parte posterior de la placa PCB.

Fuente: Elaborado por Investigador.

Después de tener elaborado el circuito impreso y ya perforados los puntos donde se colocarán los dispositivos electrónicos, se procede a la colocación de estos, para de esta manera tener la placa lista para realizar las conexiones de todo el sistema domótico. En la figura 39 se puede observar la colocación de los dispositivos electrónicos en la placa impresa.

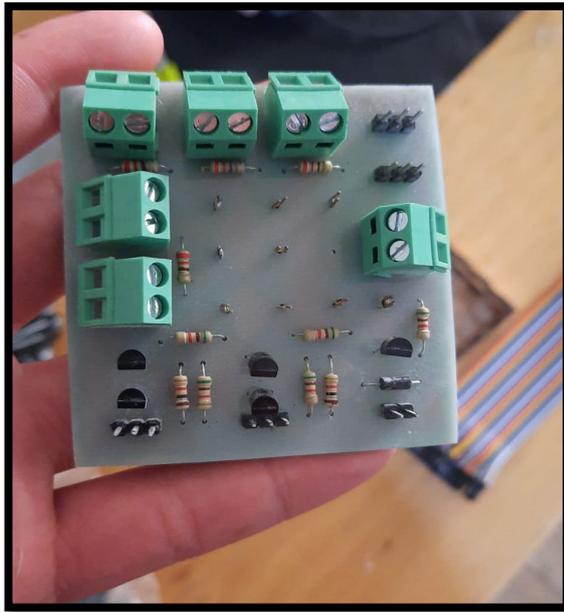


Figura 39.- Diseño de la placa PCB.

Fuente: Elaborado por Investigador.

Una vez terminada la implementación de los dispositivos electrónicos en la placa se procede a realizar la conexión de los mismos, en la figura 40 se puede observar la conexión de todos los dispositivos del sistema domótico.

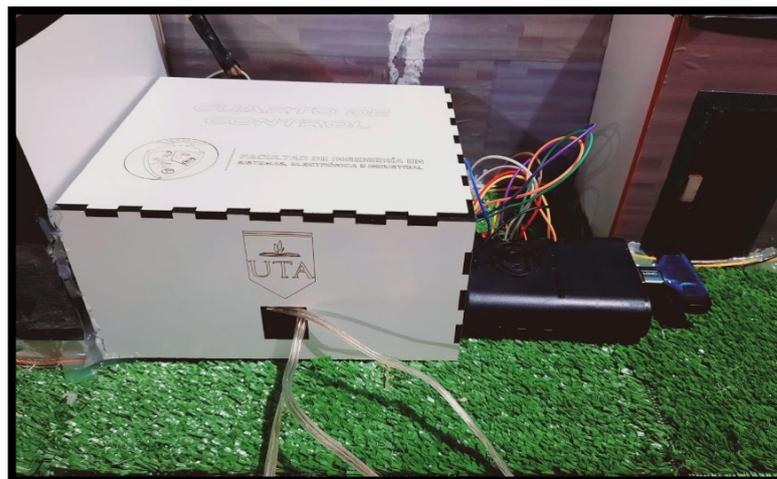


Figura 40.- Conexión de Circuitos del Sistema Domótico.

Fuente: Elaborado por Investigador.

3.12.3. Elaboración del case para la placa del circuito

Para desarrollar el case para la placa del circuito del sistema domótico, se comenzó con el diseño del esquema en el software LightBurn como se observa en la figura 41.

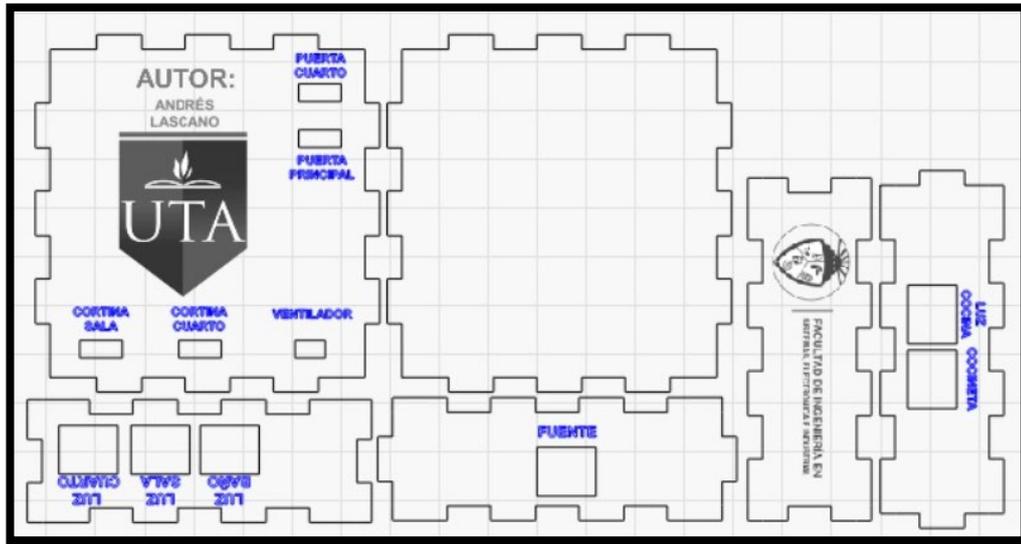


Figura 41.- Diseño del case para la placa del circuito.

Fuente: Elaborado por Investigador.

Una vez ya diseñado el esquema del case para la placa y su estructura, se procede a cortar las planchas de PBC, para después armarlo como se observa en las figuras 42 y figura 43.

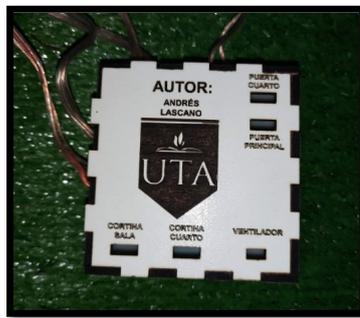


Figura 42.- Case para la placa.

Fuente: Elaborado por Investigador.



Figura 43.- Case para la placa.

Fuente: Elaborado por Investigador.

3.12.4. Elaboración del prototipo de vivienda

Para desarrollar el prototipo de vivienda, se tomó como referencia la casa de una persona con discapacidad visual de la “Escuela de Educación Básica Especializada Cardenal Julius Doephner”, la cual tiene una habitación para la persona con discapacidad visual, un baño, una sala, una cocina y el comedor; todo esto cuenta con el espacio suficiente para que no haya barreras para el desplazamiento, como se muestra en la figura 44.



Figura 44.- Plano 2D del Prototipo de la Vivienda.

Fuente: Elaborado por Investigador.

Se realizó también el plano en 3D de la vivienda en la cual se basará la elaboración del prototipo del sistema domótico para personas con discapacidad visual, como se muestra en la figura 45.



Figura 45.- Plano 3D del prototipo de la vivienda.

Fuente: Elaborado por Investigador.

Una vez diseñado el prototipo de la vivienda y su respectiva distribución, se creó el diseño en las planchas de MDF para proceder armar el prototipo como se muestra en la **Figura 46**.



Figura 46.- Armado del prototipo de la vivienda.

Fuente: Elaborado por Investigador.

3.12.5. Colocación de elementos electrónicos

Una vez armado el prototipo de la vivienda se procede a colocar cada uno de los elementos electrónicos en los lugares adecuados.

Se coloca las luces LED en la parte superior de la vivienda como se muestra en la figura 47, una luz por cada parte de la vivienda.



Figura 47.- Instalación de las luces leds de la vivienda.

Fuente: Elaborado por Investigador.

En las figuras 48 y 49 se observan los servomotores que van colocados cerca de las ventanas, estos tienen un sistema biela-manivela para convertir el movimiento circular en movimiento lineal para al momento de recibir la señal abrir y cerrar las persianas de la vivienda.



Figura 48.- Instalación del servomotor para control de persiana de la sala.

Fuente: Elaborado por Investigador.



Figura 49.- Instalación del servomotor para control de persiana del cuarto.

Fuente: Elaborado por Investigador.

En la figura 50 se muestra la instalación del ventilador, el cual está instalado en la parte superior de la vivienda, justamente en la parte de la sala para simular lo que sería el aire acondicionado de la vivienda.

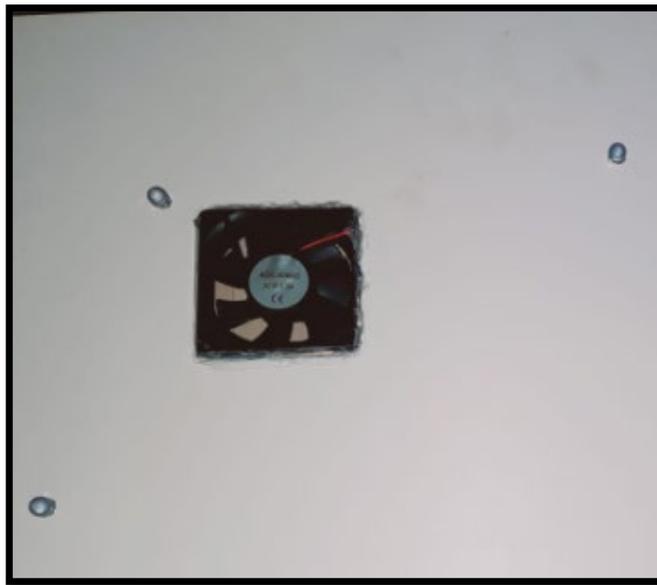


Figura 50.- Instalación del ventilador de la vivienda.

Fuente: Elaborado por Investigador

En la figura 51 se muestra la instalación de la cocineta que está representada por la activación de 2 luces leds, para simular el funcionamiento de una cocina a inducción.

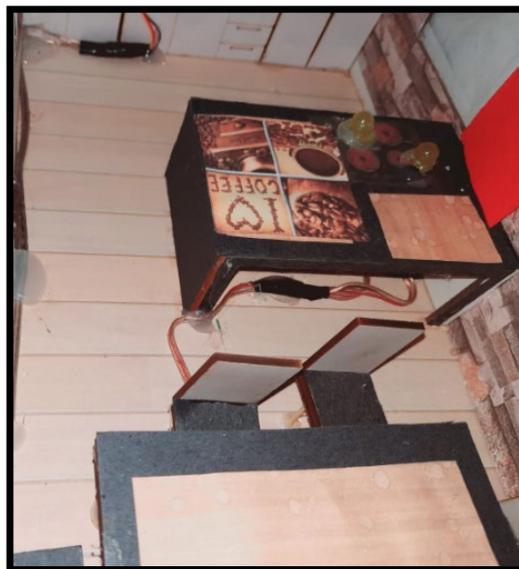


Figura 51.- Instalación de la cocineta de la vivienda.

Fuente: Elaborado por Investigador.

En la figura 52 se muestra la instalación de los relés de 5 V que simulan a las cerraduras magnéticas para el control de acceso de la vivienda.

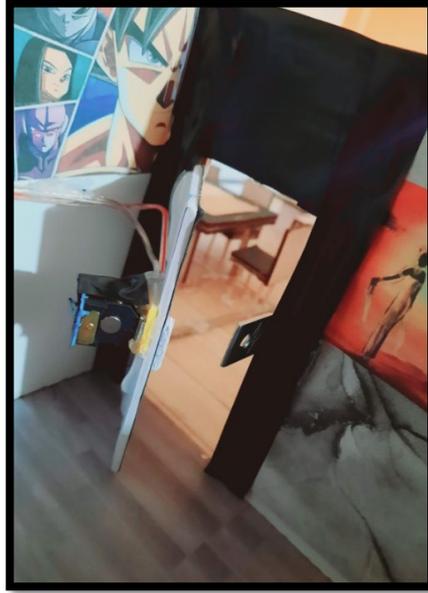


Figura 52.- Instalación del sistema de control de acceso.

Fuente: Elaborado por Investigador.

En la figura 53 se observa la colocación de todos los elementos electrónicos ya implementados en la vivienda.



Figura 53.- Instalación completa de todos los elementos electrónicos.

Fuente: Elaborado por Investigador.

En las figuras 54 y 55 se puede observar el prototipo de la vivienda con el sistema domótico funcional.



Figura 54.- Prototipo del Sistema Domótico funcional.

Fuente: Elaborado por Investigador.

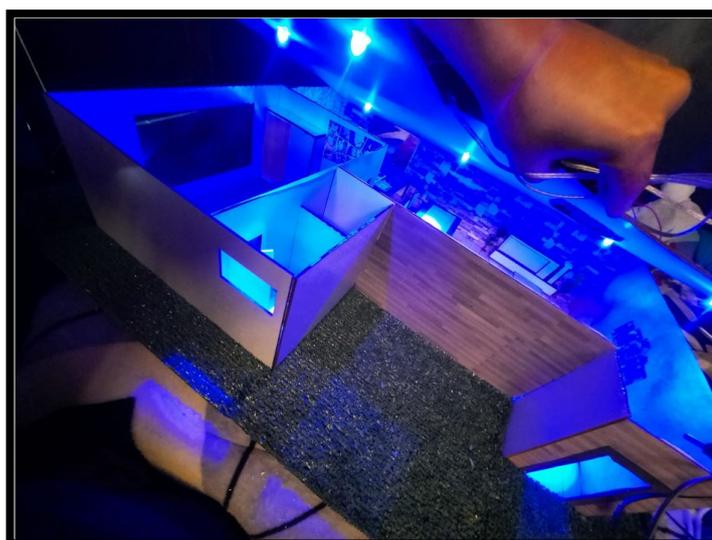


Figura 55.- Prototipo del Sistema Domótico funcional.

Fuente: Elaborado por Investigador.

3.13. Pruebas de funcionamiento

En el progreso de la implementación del prototipo del sistema domótico para personas con discapacidad visual mediante Inteligencia Artificial y reconocimiento de voz, fue de suma importancia realizar pruebas de funcionamiento con el fin de determinar la confiabilidad del sistema y verificar que el sistema funcione correctamente.

3.13.1. Reconocimiento de voz

Para realizar el análisis de confiabilidad del sistema de reconocimiento de voz se realizó pruebas de funcionamiento en 4 días diferentes con la persona con discapacidad visual (Pd) con la cual se entrenó la Red Neuronal para el sistema domótico. La tabla 17 muestra los resultados de las pruebas de funcionamiento, donde se marca un (1) cuando el comando de voz fue reconocido y un (0) cuando el comando de voz no fue reconocido satisfactoriamente.

Tabla 17.- Prueba de Funcionamiento de Reconocimiento de voz.

Fecha	Comandos	Pd	Pd	Pd	Pd	Pd	Pd
16/06/2023	Abrir-Cortina-Sala	1	1	0	1	1	1
	Abrir-Puerta-Principal	1	1	1	1	1	1
	Abrir-Cortina-Cuarto	1	1	1	1	1	1
	Encender-Luz-Cuarto	0	1	1	1	1	1
	Encender-Luz-Cocina	1	0	1	1	1	1
	Encender-Luz-Sala	1	1	1	1	1	1
	Encender-Ventilador	1	1	1	1	1	1
	Encender-Cocineta	1	1	1	1	1	1
	Abrir-Cortina-Sala	1	1	1	1	0	1
	Abrir-Puerta-Principal	1	1	1	1	1	1
Abrir-Cortina-Cuarto	0	1	1	1	1	1	
Encender-Luz-Cocina	1	1	1	1	1	0	
Encender-Luz-Cuarto	1	1	0	1	1	1	
Encender-Luz-Baño	1	1	1	1	1	1	

17/06/2023	Encender-Luz-Sala	1	1	1	1	1	1
	Encender-Ventilador	1	1	1	1	1	1
	Encender-Cocineta	1	1	1	1	1	1
18/06/2023	Abrir-Cortina-Sala	1	1	1	1	1	1
	Abrir-Puerta-Principal	1	1	1	0	1	1
	Abrir-Cortina-Cuarto	1	1	0	1	1	1
	Abrir-Puerta-Cuarto	1	1	1	1	1	1
	Encender-Luz-Cuarto	1	1	1	1	1	1
	Encender-Luz-Cocina	1	1	1	1	1	1
	Encender-Luz-Sala	1	1	1	1	1	1
	Encender-Ventilador	1	0	1	1	1	1
	Encender-Cocineta	1	1	1	1	1	1
	Encender-Luz-Baño	1	1	1	1	0	1
19/06/2023	Abrir-Puerta-Principal	1	1	1	1	1	1
	Abrir-Puerta-Cuarto	1	1	1	1	1	1
	Abrir-Cortina-Sala	1	1	1	1	1	1
	Abrir-Cortina-Cuarto	1	1	1	1	1	1
19/06/2023	Encender-Luz-Sala	1	1	1	1	1	1
	Encender-Ventilador	1	1	1	1	1	0
	Encender-Cocineta	1	1	1	1	1	1
Datos Fallidos		2	2	3	1	2	2

Fuente: Elaborado por Investigador.

En la tabla 18 se puede visualizar los resultados obtenidos de la prueba de funcionamiento del sistema de reconocimiento de voz entrenado.

Tabla 18.- Resultados de las pruebas realizados para el reconocimiento de voz.

Número de Comandos de Voz	Porcentaje (%)
Total	210
Comandos No Reconocidos	12
Comandos Reconocidos	198

Fuente: Elaborado por Investigador.

Una vez realizadas las pruebas de funcionamiento del sistema domótico los resultados obtenidos muestran que el proyecto de investigación tiene un rendimiento del 94,29% lo que representa que el sistema tiene una fiabilidad bastante alta, el margen de error es de 5,71% y esto es debido a muchos factores como el ruido producido por el micrófono o la mala pronunciación de las palabras por el usuario.

3.14. Presupuesto del prototipo

Para realizar el cálculo del presupuesto en la implementación del sistema domótico para personas con discapacidad visual mediante inteligencia artificial y reconocimiento de voz se tomó en cuenta el presupuesto de construcción y presupuesto de diseño. Para obtener el presupuesto de diseño se consideró el número de horas empleadas en el diseño y construcción como también el salario de un Ingeniero en Telecomunicaciones que de acuerdo con el Ministerio de Trabajo es de 858 dólares americanos. Considerando un promedio de 20 días laborables y utilizando la siguiente ecuación se calculó el salario por día [21].

$$\text{Salario}_{diario} = \frac{\text{Salario}_{mensual}}{\text{Días}_{laborables}}$$

$$\text{Salario}_{diario} = \frac{858}{20}$$

$$\text{Salario}_{diario} = \$42.9$$

El Ministerio de Trabajo reglamenta que por día son ocho horas laborables y aplicadas en la siguiente ecuación se obtiene el valor en dólares por hora de trabajo.

$$\text{Salario}_{hora} = \frac{\text{Salario}_{diario}}{\text{Horas}_{laborables}}$$

$$\text{Salario}_{\text{hora}} = \frac{42.9}{8}$$

$$\text{Salario}_{\text{hora}} = \$5.36$$

Se evalúa unas 310 horas de investigación utilizadas en el diseño, simulación, construcción y pruebas de funcionamiento: mediante la siguiente ecuación se obtiene el presupuesto de diseño del presente proyecto.

$$\text{Presupuesto}_{\text{diseño}} = \text{Horas}_{\text{investigación}} * \text{Salario}_{\text{hora}}$$

$$\text{Presupuesto}_{\text{diseño}} = 310 * 5.36$$

$$\text{Presupuesto}_{\text{diseño}} = \$1661,6$$

En la tabla 19 se muestra un análisis del presupuesto del proyecto donde se detalla el costo de cada uno de los componentes y elementos que se utilizó en la elaboración del sistema domótico para personas con discapacidad visual.

Con el uso de la siguiente ecuación se obtuvo el valor del presupuesto total del presente proyecto de investigación.

$$\text{Presupuesto}_{\text{total}} = \text{Presupuesto}_{\text{diseño}} * \text{Presupuesto}_{\text{construcción}}$$

$$\text{Presupuesto}_{\text{total}} = \$1661,6 + \$433,15$$

$$\text{Presupuesto}_{\text{total}} = \$2094,75$$

Tabla 19.- Presupuesto.

Ítem	Descripción	Unidad	Cantidad	Valor Unitario	Valor Total
1	Raspberry Pi 4	c/u	1	\$ 200	\$ 200
2	Micrófono Bluetooth	c/u	1	\$ 30	\$ 30
3	Transistores 2n2222	c/u	5	\$ 1	\$ 5
4	Servomotor Sg90	c/u	2	\$ 4	\$ 8
5	Fuente de Alimentación de 5V	c/u	2	\$ 10	\$ 20
6	Resistencias 220 ohmios	c/u	4	\$ 0,25	\$ 1,25
7	Resistencias de 10 K ohmios	c/u	10	\$ 0,25	\$ 2,50
8	Diodos 1n4007	c/u	2	\$ 0,50	\$ 1
9	Diodos Led	c/u	10	\$ 0,30	\$ 3
10	Baquelita	c/u	2	\$ 5	\$ 10
11	Cloruro Férrico	c/u	3	\$ 3	\$ 9
12	Cables	m	2	\$ 5	\$ 10
13	Maqueta	c/u	1	\$ 100	\$ 100
14	Case para la placa	c/u	1	\$15	\$15
15	Modulo relé de 5v de 2 vías	c/u	2	\$5	\$10
16	Relés 5V	c/u	4	\$ 0,85	\$ 3,40
17	Ventilador de 5V	c/u	1	\$5	\$5
Presupuesto					\$433,15

Fuente: Elaborado por Investigador

El presupuesto total del presente proyecto de investigación es de DOS MIL NOVENTA Y CUATRO DORALES CON SETENTA Y CINCO CENTAVOS, este dinero fue financiado por el investigador.

CAPÍTULO IV

Una vez finalizado el presente trabajo de investigación se presentan las siguientes conclusiones y recomendaciones a las que se ha llegado:

5.1. Conclusiones

- Se identificó el estado actual de las viviendas de las personas con discapacidad visual donde se logró verificar que la mayor parte de personas cuentan con vivienda propia y requieren de la domótica para mejorar su calidad de vida dentro del hogar.
- Se diseñó el prototipo del sistema domótico para personas con discapacidad visual mediante Inteligencia Artificial y reconocimiento de voz, el cual genera seguridad debido a que el entrenamiento de la Red Neuronal autentica y encripta la información, y permite que solo el usuario con el que se entrenó la Red Neuronal haga uso del sistema domótico.
- Se evaluó al sistema domótico para personas con discapacidad visual mediante inteligencia artificial y reconocimiento de voz realizando pruebas de funcionamiento en días diferentes donde el sistema ofrece un elevado porcentaje del 94,29% de efectividad, esto se debe al constante entrenamiento de la Red Neuronal para el reconocimiento de voz del usuario que va a controlar el sistema, como también al Micrófono Fifine M6 de alto rendimiento usado para el envío de los comandos, el cual reduce el ruido del sistema y hace que el programa sea estable para el reconocimiento de la voz.
- El sistema domótico para personas con discapacidad visual al estar programado en código abierto no necesita de licencias, es así como el proyecto puede seguir ampliándose, dependiendo de las necesidades de las personas, es decir si una persona con discapacidad visual desea controlar una vivienda de mayor número de plantas, habitaciones, etc., se puede implementar aumentando puertos con otro controlador para satisfacer la necesidad del inmueble que necesite controlar la persona.

5.2. Recomendaciones

- Para evitar problemas con el sistema de reconocimiento de voz en la Raspberry PI 4, es importante pronunciar cada palabra de forma clara y correctamente para que los comandos de voz enviados concuerden con los datos almacenados en la base de datos del sistema.
- Se recomienda utilizar un repetidor de señal Bluetooth para aumentar el alcance de micrófono como el 1Mii B06Pro, cubriendo así todos los puntos de la casa, para aumentar el rango de comunicación inalámbrica entre el micrófono y la tarjeta Raspberry Pi4.
- Para el acoplamiento del Sistema Domótico en otras casas, se recomienda realizar un análisis de las principales áreas a controlar en la vivienda, basándose principalmente en las necesidades de la persona con discapacidad visual; además se debe entrenar nuevamente el modelo para que reconozca la voz de esa persona.

REFERENCIAS BIBLIOGRÁFICAS

- [1] OMS, «La OMS presenta el primer Informe mundial sobre la visión,» Ginebra, 2019.
- [2] D. Rodríguez , «Tecnologías de información y comunicación para el turismo inclusivo,» *Revista Facultad de Ciencias Económicas*, vol. 26, n° 1, pp. 125-146, 2018.
- [3] D. Sierra , «Funcionalidad del sistema de señalización de calles y la orientación espacial de personas con discapacidad visual en la ciudad de Ambato,» Universidad Técnica de Ambato, 2022.
- [4] J. Pérez , *Navegación web dirigida por comandos de voz*, Chile: Universidad Nacional de La Plata, 2017.
- [5] L. Pascual, «Contribución al desarrollo de modelos de control domotico,» Madrid, 2022.
- [6] D. Chaparro, *Desarrollo de una aplicación móvil para la comunicación de personas con discapacidad con asistentes virtuales inteligentes de dispositivos IoT*, Universidad de Alicante, 2021.
- [7] A. Cahuana, «Sistema domótico con la tecnología arduino a través de un aplicativo móvil para el control corporal,» Tigua, 2020.
- [8] M. Carvalho, «Arquitectura de servicios B.Live para sistemas domoticos,» Madrid, 2019.
- [9] D. Valencia y L. Gonzales , «Diseño e implementación de un prototipo de robot asistente para personas con discapacidad motriz y adultos mayores, basado en Inteligencia Artificial,» UPS, Ambato, 2019.
- [10] K. Barrios , J. López , S. Mendieta y R. Benavides , *Sistema de reconocimiento de voz: un enlace en la comunicación hombre-máquina*, Panamá, 2018.
- [11] A. Inés , *La inteligencia Artificial desde una perspectiva Ética*, Universidad Católica de Córdoba, 2022.
- [12] L. Holguin, *Hogar inteligente utilizando internet de las cosas*, Guayaquil: Universidad de Guayaquil, 2018.

- [13] P. Nandankar y A. Zade, «Home Automation System Using Android Mobile Phone,» *Internation Journal of Innovative Reserach in Advance Engineering*, vol. 5, n° 04, pp. 149-153, 2018.
- [14] A. Ashraf, M. Faisal y K. Parvin, «Air Conditioning for Smart Home Energy Management System,» *International Journal of Engineering & Technology*, vol. 7, pp. 487-490, 2018.
- [15] E. Díaz, DESARROLLO DE UN SISTEMA DOMÓTICO BASADO EN IOT PARA LA SEGURIDAD RESIDENCIAL Y MEJORAMIENTO DEL CONSUMO ENERGÉTICO, APLICANDO CONCEPTOS DE BIG DATA, Bogotá: Universidad Militar Nueva Granada , 2019.
- [16] R. Gahona y A. Gavilema, DISEÑO DE LA RED INTERNET DE LAS COSAS (IOT) PARA EL EDIFICIO DE LA EMPRESA CONSEL, Quito: Universidad Politécnica Salesiana , 2020.
- [17] F. Sánchez, «Análisis de la eficiencia del control de un motor asincrónico mediante SVPWM en lazo abierto,» Universidad Técnica de Cotopaxi, 2022.
- [18] V. Marcial y L. Esteves , «Impacto de la Inteligencia Artificial en el comportamiento informacional,» *Dialnet*, vol. 18, n° 3, pp. 94-106, 2022.
- [19] A. Abeliuk y C. Gutiérrez, *Historia y evolución de la Inteligencia Artificial*, Chile, 2021, pp. 1-8.
- [20] C. Fajardo de Andara, *Marvin Lee Minsky: pionero en la investigación de la Inteligencia Artificial (1927-2016)*, Venezuela, 2021.
- [21] V. Áreas, J. Salazar, C. Graciano , C. Julio , R. Rojas y V. Pirela , *Una introducción a las aplicaciones de la*, Revista Latinoamericana de Hipertensión.
- [22] E. Bernal y R. Cardona , *Videojuegos: Avances tecnológicos en aplicación de física e Inteligencia Artificial*, 2018.
- [23] I. Jara y J. Ochoa , *Usos y efectos de la Inteligencia Artificial en educación*, 2020.
- [24] L. Cotino , *RIESGOS E IMPACTOS DEL BIG DATA, LA INTELIGENCIA ARTIFICIAL Y LA ROBÓTICA. ENFOQUES, MODELOS Y PRINCIPIOS DE LA RESPUESTA DEL DERECHO*, Valencia : Constitucional de la Universitat de Valencia, 2019.

- [25] B. Mahesh, «Machine Learning Algorithms - A Review,» *International Journal of Science and Research*, pp. 1-6, 2018.
- [26] E. Bautista, J. Quirama y E. Bautista, «Modelo predictivo del progreso en el aprendizaje de los estudiantes de uniminuto aplicando técnicas de machine learning,» 2019.
- [27] J. Dueñas , «Aplicación de técnicas de machine learning a la ciberseguridad: Aprendizaje supervisado para la detección de amenazas web mediante clasificación basada en árboles de decisión,» Universidad Oberia de Catalunya, 2020.
- [28] E. Bajaña y R. Loja , «Desarrollo de modelo predictivo basado en algoritmos de aprendizaje supervisado de Machine Learning, para el análisis de datos en pacientes con la enfermedad de Crohn.,» Universidad de Guayaquil, 2022.
- [29] S. Valverde, «Aplicaciones de la Inteligencia Artificial en la empresa,» Universidad de Cantabria, 2019.
- [30] L. Rouhiainen, «La Inteligencia Artificial,» 2018.
- [31] K. Barrios , J. López , S. Mendieta , R. Benavides y Y. Sáez , «Sistema de reconocimiento de voz: un enlace en la comunicación hombre-máquina,» *Iniciación Científica*, vol. 4, nº 2, pp. 1-4, 2018.
- [32] A. Araujo , J. Pérez y W. Rodríguez , *Aplicacion de una Red Neuronal Convolutiva para el Reconocimiento de Personas a Traves de la Voz*, Venezuela: Universidad de Los Andes.
- [33] C. Delgado, J. Sandoval y W. Arteaga, *Blockly Voice: un entorno de programación guiado por voz*, 2018.
- [34] J. Rojas , E. Abello , A. Simanca y F. Garrido , «EAPP: Plataforma tecnológica para la traducción de voz a texto como apoyo a la educación inclusiva en el proceso de enseñanza en la educación superior,» *AVENIR*, vol. 3, nº 1, pp. 1-5, 2019.
- [35] A. Jiménez , *Informe Olivenza 2018, sobre la situación general de la discapacidad en España*, 2018.
- [36] S. Denis, *Clasificación Internacional del Funcionamiento, de la Discapacidad y de la Salud*, 2018.

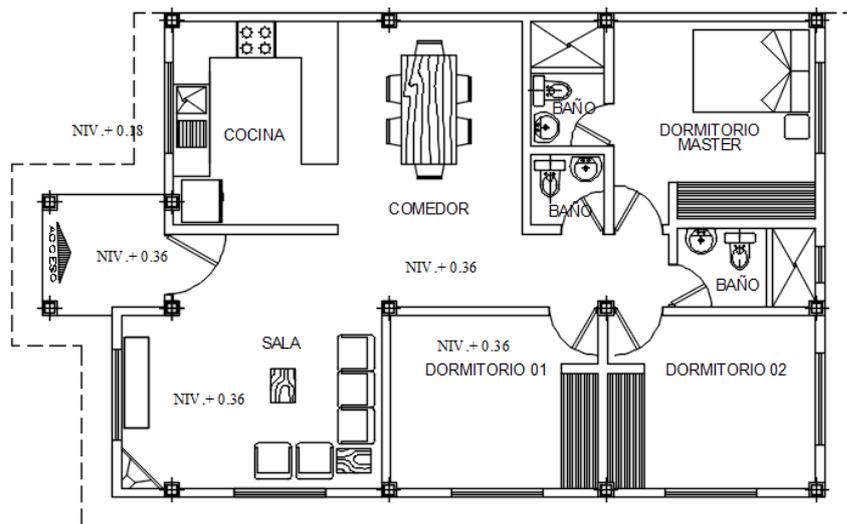
- [47] D. Aguilar, R. León y J. Alvarez, «Evaluation and analysis of a ptoyotype wireless sensor for monitoring critical infrastures in bridges,» *Internation Conference on Automatica* , 2016.

ANEXOS

ANEXO 1

PLANO DE LA VIVENDA DE LA PERSONA CON DISCAPACIDAD VISUAL CON LA CUAL SE REALIZÓ EL ENTRENAMIENTO DE LA RED NEURONAL.

En el Software de AUTOCAD se realizó el plano de la vivienda de la persona con discapacidad visual con la cual se entrenó el modelo de Red Neuronal para el control del prototipo del sistema domótico, la cual está ubicada en el Barrio La Joya parroquia Huachi Chico del Cantón Ambato.



ANEXO 2

ANÁLISIS DE FIABILIDAD

En este anexo se detalla el análisis de fiabilidad de las encuestas realizadas a las personas con discapacidad visual, la cual se realizó con ayuda del programa Excel.

Escala: All variables

Resumen de procesamiento de casos

		N	%
Caso	Válido	21	100
	Excluido	0	0
	Total	21	100

Estadísticas de fiabilidad

<u>Alfa de Cronbach</u>	<u>N de elementos</u>
0,854	5

Estadísticos

	¿La vivienda en la que usted habita es?	¿Cómo considera usted la accesibilidad con la que cuenta una persona con discapacidad visual en su vivienda?	¿Las medidas de seguridad con las que cuenta su vivienda para una persona con discapacidad visual son?	¿Cómo estima usted la inclusión de la domótica en una vivienda para mejorar la calidad de vida de personas con discapacidad visual?	¿Qué aspectos tecnológicos considera usted que deben solventarse para mejorar la calidad de vida de las personas con discapacidad visual dentro del hogar?
N	21	21	21	21	21
Válido					
N	0	0	0	0	0
Perdidos					

Como se puede observar en los resultados estadísticos el análisis de fiabilidad de las encuestas realizadas nos brinda un Alfa de Cronbach 0,854. En consecuencia, el análisis de la encuesta planteada presenta que las preguntas de la encuesta aplicada tienen una gran relación entre ellas y ningún dato fue perdido.

ANEXO 3

ENCUESTA



UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE TELECOMUNICACIONES

La presente encuesta se realiza para el proyecto de investigación “Sistema domótico para personas con discapacidad visual mediante Inteligencia Artificial y reconocimiento de voz”, con el fin de identificar el estado actual de las viviendas para personas con discapacidad visual.

Instrucciones:

- Responder a las preguntas honestamente según su criterio.
- Si alguna pregunta no es clara, siéntase libre de preguntarle al investigador.

1.- ¿La vivienda en la que usted habita es?

- a) Propia
- b) Arrendada

2.- ¿Cómo considera usted la accesibilidad con la que cuenta una persona con discapacidad visual en su vivienda?

- a) Excelente
- b) Buena
- c) Regular
- d) Mala

3.- ¿Las medidas de seguridad con las que cuenta su vivienda para una persona con discapacidad visual son?

- a) Excelentes

- b) Buena
- c) Regular
- d) Mala

4.- ¿Cómo estima usted la inclusión de la domótica en una vivienda para mejorar la calidad de vida de personas con discapacidad visual?

- a) Excelente
- b) Buena
- c) Regular
- d) Mala

5.- ¿Qué aspectos tecnológicos considera usted que deben solventarse para mejorar la calidad de vida de las personas con discapacidad visual dentro del hogar?

- a) Control de iluminación
- b) Seguridad
- c) Control de dispositivos y artefactos electrónicos
- d) Acceso
- e) Todos

GRACIAS POR SU ATENCIÓN

ANEXO 4

RECEPCIÓN DE LA SOLICITUD PARA LA REALIZACIÓN DE LA ENCUESTA

 UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE TELECOMUNICACIONES

FORMATO DE SOLICITUD

Ambato, 09/06/2023

Mg. Johan Serrano
Líder Educativo de la Escuela de Educación Básica especializada "Julius Doephner"

Dr. Freddy Geovanny Benalcázar Palacios

Presente

Estimado Mg. Johan Serrano me pongo en contacto con usted, para solicitar de manera más comedida se permita a mi tutorado el Sr. Dennis Andrés Lascano Solís estudiante de la Universidad Técnica de Ambato, de la Facultad de Ingeniería en Sistemas Electrónica e Industrial, de noveno semestre de la carrera de Ingeniería en Telecomunicaciones, realizar una encuesta a los estudiante de su institución educativa con el objetivo de "Identificar el estado actual de las viviendas de las personas con discapacidad visual", ya que mi tutorado se encuentra realizando un proyecto de investigación con el tema: "Sistema domótico para personas con discapacidad visual mediante inteligencia artificial y reconocimiento de voz" y para ello necesitamos de su colaboración para continuar con dicho proyecto.

De antemano agradezco su colaboración y su tiempo prestado.

Atentamente.


.....

Dr. Freddy Geovanny Benalcázar Palacios
C.I. 1802535888
Cel. 0984026615
Correo: fg.benalcazar@uta.edu.ec


Rely
2023-06-12

ANEXO 5

FOTOGRAFÍAS QUE EVIDENCIAN LA APLICACIÓN DE LA ENCUESTA



ANEXO 6

CÓDIGO DE PROGRAMACIÓN DEL ENTRENADOR DE LA RED DEL SISTEMA DOMÓTICO

```
1.
2. import os
3.
4. from IPython import display
5. import matplotlib.pyplot as plt
6. import numpy as np
7. import pandas as pd
8.
9. import tensorflow as tf
10.     import tensorflow_hub as hub
11.     import tensorflow_io as tfio
12.
13.
14.     yamnet_model_handle
    = 'https://tfhub.dev/google/yamnet/1'
15.     yamnet_model = hub.load(yamnet_model_handle)
16.
17.
18.     testing_wav_file_name =
    tf.keras.utils.get_file('abrir_puerta_principal.wav',
19.                             'pi
    /Desktop/audios',
20.                             cac
    he_dir='./',
21.                             cac
    he_subdir='test_data')
22.
23.     print(testing_wav_file_name)
24.
25.
26.
27.     @tf.function
28.     def load_wav_16k_mono(filename):
29.
30.         file_contents = tf.io.read_file(filename)
31.         wav, sample_rate = tf.audio.decode_wav(
32.             file_contents,
33.             desired_channels=1)
34.         wav = tf.squeeze(wav, axis=-1)
35.         sample_rate = tf.cast(sample_rate,
    dtype=tf.int64)
36.         wav = tfio.audio.resample(wav,
    rate_in=sample_rate, rate_out=16000)
37.         return wav
38.
39.     testing_wav_data =
    load_wav_16k_mono(testing_wav_file_name)
```

```

40.
41.     _ = plt.plot(testing_wav_data)
42.
43.
44.     display.Audio(testing_wav_data, rate=16000)
45.
46.
47.
48.     class_map_path =
yamnet_model.class_map_path().numpy().decode('utf-8')
49.     class_names
= list(pd.read_csv(class_map_path) ['display_name'])
50.
51.     for name in class_names[:20]:
52.         print(name)
53.     print('...')
54.
55.
56.
57.     scores, embeddings, spectrogram =
yamnet_model(testing_wav_data)
58.     class_scores = tf.reduce_mean(scores, axis=0)
59.     top_class = tf.math.argmax(class_scores)
60.     inferred_class = class_names[top_class]
61.
62.     print(f'The main sound is: {inferred_class}')
63.     print(f'The embeddings shape: {embeddings.shape}')
64.
65.
66.     _ = tf.keras.utils.get_file('esc-50.zip',
67.                                 'https://github.com/karoldv
1/ESC-50/archive/master.zip',
68.                                 cache_dir='./',
69.                                 cache_subdir='datasets',
70.                                 extract=True)
71.
72.
73.
74.     esc50_csv = './datasets/ESC-50-
master/meta/esc50.csv'
75.     base_data_path = './datasets/ESC-50-master/audio/'
76.
77.     pd_data = pd.read_csv(esc50_csv)
78.     pd_data.head()
79.
80.
81.     filenames = filtered_pd['filename']
82.     targets = filtered_pd['target']
83.     folds = filtered_pd['fold']
84.
85.     main_ds =
tf.data.Dataset.from_tensor_slices((filenames, targets,
folds))

```

```

86.     main_ds.element_spec
87.
88.     def load_wav_for_map(filename, label, fold):
89.         return load_wav_16k_mono(filename), label, fold
90.
91.     main_ds = main_ds.map(load_wav_for_map)
92.     main_ds.element_spec
93.
94.     # applies the embedding extraction model to a wav
    data
95.     def extract_embedding(wav_data, label, fold):
96.         ''' run YAMNet to extract embedding from the wav
    data '''
97.         scores, embeddings, spectrogram =
    yamnet_model(wav_data)
98.         num_embeddings = tf.shape(embeddings)[0]
99.         return (embeddings,
100.                tf.repeat(label, num_embeddings),
101.                tf.repeat(fold, num_embeddings))
102.
103.     main_ds = main_ds.map(extract_embedding).unbatch()
104.     main_ds.element_spec
105.
106.
107.
108.     cached_ds = main_ds.cache()
109.     train_ds = cached_ds.filter(lambda embedding,
    label, fold: fold < 4)
110.     val_ds = cached_ds.filter(lambda embedding,
    label, fold: fold == 4)
111.     test_ds = cached_ds.filter(lambda embedding,
    label, fold: fold == 5)
112.
113.     # remove the folds column now that it's not needed
    anymore
114.     remove_fold_column = lambda embedding,
    label, fold: (embedding, label)
115.
116.     train_ds = train_ds.map(remove_fold_column)
117.     val_ds = val_ds.map(remove_fold_column)
118.     test_ds = test_ds.map(remove_fold_column)
119.
120.     train_ds =
    train_ds.cache().shuffle(1000).batch(32).prefetch(tf.data
    a.AUTOTUNE)
121.     val_ds =
    val_ds.cache().batch(32).prefetch(tf.data.AUTOTUNE)
122.     test_ds =
    test_ds.cache().batch(32).prefetch(tf.data.AUTOTUNE)
123.
124.
125.     my_model = tf.keras.Sequential([

```

```

126.         tf.keras.layers.Input(shape=(1024),
127.             dtype=tf.float32,
128.             name='input_embedding'),
129.         tf.keras.layers.Dense(512, activation='relu'),
130.         tf.keras.layers.Dense(len(my_classes))
131.     ], name='my_model')
132.     my_model.summary()
133.
134.     my_model.compile(loss=tf.keras.losses.SparseCatego
135.         ricalCrossentropy(from_logits=True),
136.                       optimizer="adam",
137.                       metrics=['accuracy'])
138.     callback =
139.         tf.keras.callbacks.EarlyStopping(monitor='loss',
140.             e=3,
141.             _best_weights=True)
142.     history = my_model.fit(train_ds,
143.                             epochs=20,
144.                             validation_data=val_ds,
145.                             callbacks=callback)
146.
147.
148.     loss, accuracy = my_model.evaluate(test_ds)
149.
150.     print("Loss: ", loss)
151.     print("Accuracy: ", accuracy)
152.
153.     scores, embeddings, spectrogram =
154.         yamnet_model(testing_wav_data)
155.     result = my_model(embeddings).numpy()
156.     inferred_class =
157.         my_classes[result.mean(axis=0).argmax()]
158.     print(f'The main sound is: {inferred_class}')
159.
160.     class ReduceMeanLayer(tf.keras.layers.Layer):
161.         def __init__(self, axis=0, **kwargs):
162.             super(ReduceMeanLayer, self).__init__(**kwargs)
163.
164.             self.axis = axis
165.
166.         def call(self, input):
167.             return tf.math.reduce_mean(input,
168.                 axis=self.axis)

```

```

169.     input_segment = tf.keras.layers.Input(shape=(),
        dtype=tf.float32, name='audio')
170.     embedding_extraction_layer =
        hub.KerasLayer(yamnet_model_handle,
171.                                                              trainab
        le=False, name='yamnet')
172.     _, embeddings_output, _ =
        embedding_extraction_layer(input_segment)
173.     serving_outputs = my_model(embeddings_output)
174.     serving_outputs = ReduceMeanLayer(axis=0,
        name='classifier')(serving_outputs)
175.     serving_model = tf.keras.Model(input_segment,
        serving_outputs)
176.     serving_model.save(saved_model_path,
        include_optimizer=False)
177.
178.     tf.keras.utils.plot_model(serving_model)
179.
180.
181.     reloaded_results = reloaded_model(testing_wav_data)
182.     cat_or_dog =
        my_classes[tf.math.argmax(reloaded_results)]
183.     print(f'The main sound is: {cat_or_dog}')
184.
185.
186.     serving_results =
        reloaded_model.signatures['serving_default'](testing_wav
        _data)
187.     cat_or_dog =
        my_classes[tf.math.argmax(serving_results['classifier'])
        ]
188.     print(f'The main sound is: {cat_or_dog}')
189.
190.
191.
192.     test_pd = filtered_pd.loc[filtered_pd['fold'] == 5]
193.     row = test_pd.sample(1)
194.     filename = row['filename'].item()
195.     print(filename)
196.     waveform = load_wav_16k_mono(filename)
197.     print(f'Waveform values: {waveform}')
198.     _ = plt.plot(waveform)
199.
200.     display.Audio(waveform, rate=16000)
201.
202.     scores, embeddings, spectrogram =
        yamnet_model(waveform)
203.     class_scores = tf.reduce_mean(scores, axis=0)
204.     top_class = tf.math.argmax(class_scores)
205.     inferred_class = class_names[top_class]
206.     top_score = class_scores[top_class]

```

```

207.     print(f'[YAMNet] The main sound is:
        {inferred_class} ({top_score})')
208.
209.     reloaded_results = reloaded_model(waveform)
210.     your_top_class = tf.math.argmax(reloaded_results)
211.     your_inferred_class = my_classes[your_top_class]
212.     class_probabilities =
        tf.nn.softmax(reloaded_results, axis=-1)
213.     your_top_score =
        class_probabilities[your_top_class]
214.     print(f'[Your model] The main sound is:
        {your_inferred_class} ({your_top_score})')

```

Config.inc.php

```

1. <?php
2.
3.     define( "DIR_RAIZ", $_SERVER['DOCUMENT_ROOT'] );
4.     define( "DIR_INC", DIR_RAIZ . "/includes/" );
5.
6.     define( "HOST", "http://" .
    $_SERVER['HTTP_HOST'] );
7.     define( "DIR_IMG", HOST . "/images/" );
8.
9.     define( "MYSQL_HOST", "localhost" );
10.    define( "MYSQL_USER", "andres" );
11.    define( "MYSQL_PASSWD", "Azcarate3489" );
12.    define( "MYSQL_DBNAME", "basededatos" );
13.    define( "MYSQL_DEBUG", "false" );
14.
15.    define( "TMPDIR", DIR_RAIZ . "/tmp" );
16.    session_save_path( TMPDIR );
17.
18.    /*echo $_SERVER['DOCUMENT_ROOT'] . "<br>";
19.    echo DIR_RAIZ . "<br>";
20.    echo TMPDIR . "<br>";
21.    echo HOST . "<br>";
22.    echo DIR_INC . "<br>";
23.    echo DIR_IMG . "<br>";
24.    echo MYSQL_HOST;
25. */
26. ?>

```

CÓDIGO DE CONTROL DE ENTRADAS Y SALIDAS

```
1. import RPi.GPIO as GPIO
2. import pyaudio
3. import random
4. import os
5. import shutil
6. import torch
7. import json
8. import time
9. import array
10.     import subprocess
11.     import numpy
12.     from voice_util import detect_audio_segment
    , read_voice_file, total_energy_list,
    convert2wav, trim_audio_with_sox
13.     from train_model import mfcc_from_file,
    similarity_mfcc, mfcc_from_audio
14.     from gmm_model import load_gmm_from_npz
15.     import webrtcvad
16.     import librosa
17.     #Iniciamos las variables para salidas de
    las raspberry
18.     GPIO.cleanup()
19.     GPIO.setmode(GPIO.BCM)
20.
21.     #Creamos funciones para convertir la
    posición en ángulo para el movimiento de
    servomotores.
22.     def angulo(ang):
23.         return float(pos)/10.+5.
24.     def angulo2(ang2):
25.         return float(pos2)/10.+5.
26.     #Declaramos variables para las cortinas y
    sus pines de salida
27.     Cortina_cuarto = 12
28.     GPIO.setup(Cortina_cuarto, GPIO.OUT)
29.     pwm = GPIO.PWM(Cortina_cuarto,100)
30.     Definimos posiciones partida y
    llegada del servomotor
31.     depart = 0
32.     arrive = 65
33.     delay = 0.1
34.     incStep = 5
```

```

35.     pos = depart
36.     Cortina_sala = 13
37.     GPIO.setup(Cortina_sala, GPIO.OUT)
38.     pwm2 = GPIO.PWM(Cortina_sala,100)
39.     depart2 = 0
40.     arrive2 = 65
41.     pos2 = depart
42.     Definimos variables para las luces led
43.     luzcuarto = 2
44.     luzsala = 3
45.     luzcocina = 17
46.     luzbano = 4
47.     Definimos variables para las puertas
48.     puerta_principal = 23
49.     puerta_cuarto = 24
50.     Definimos variables para los dispositivos
    AC
51.     ventilador = 27
52.     cocineta = 22
53.     #Las variables son declaradas salidas
54.     GPIO.setup(luzsala, GPIO.OUT)
55.     GPIO.setup(luzcuarto, GPIO.OUT)
56.     GPIO.setup(luzcocina, GPIO.OUT)
57.     GPIO.setup(luzbano, GPIO.OUT)
58.     GPIO.setup(puerta_principal, GPIO.OUT)
59.     GPIO.setup(puerta_cuarto, GPIO.OUT)
60.     GPIO.setup(ventilador, GPIO.OUT)
61.     GPIO.setup(cocineta, GPIO.OUT)
62.     #Realizamos varias funciones para activar o
    desactivar los pines de salida
63.     def encender_luz_cuarto():
64.         GPIO.output(luzcuarto, GPIO.HIGH)
65.         print("Encendido luz cuarto")
66.     def apagar_luz_cuarto():
67.         GPIO.output(luzcuarto, GPIO.LOW)
68.         print("Apagando luz de cuarto")
69.     def encender_luz_sala():
70.         GPIO.output(luzsala, GPIO.HIGH)
71.         print("Encendido luz sala")
72.     def apagar_luz_sala():
73.         GPIO.output(luzsala, GPIO.LOW)
74.         print("Apagando luz de sala")
75.     def encender_luz_cocina():
76.         GPIO.output(luzcocina, GPIO.HIGH)
77.         print("Encendiendo luz de cocina")

```

```

78.     def apagar_luz_cocina():
79.         GPIO.output(luzcocina, GPIO.LOW)
80.         print("Apagando luz de cocina")
81.     def encender_luz_bano():
82.         GPIO.output(luzbano, GPIO.HIGH)
83.         print("Encendiendo luz de baño")
84.     def apagar_luz_bano():
85.         GPIO.output(luzbano, GPIO.LOW)
86.         print("Apagando luz de baño")
87.     def abrir_puerta_principal():
88.         GPIO.output(puerta_principal,
89.                    GPIO.HIGH)
90.         print("Puerta principal abierta")
91.     def cerrar_puerta_principal():
92.         GPIO.output(puerta_principal, GPIO.LOW)
93.         print("Puerta principal cerrada")
94.     def abrir_puerta_cuarto():
95.         GPIO.output(puerta_cuarto, GPIO.HIGH)
96.         print("Puerta del cuarto abierta")
97.     def cerrar_puerta_cuarto():
98.         GPIO.output(puerta_cuarto, GPIO.LOW)
99.         print("Puerta del cuarto cerrada")
100.    def encender_ventilador():
101.        GPIO.output(ventilador, GPIO.HIGH)
102.        print("Ventilador encendido")
103.    def apagar_ventilador():
104.        GPIO.output(ventilador, GPIO.LOW)
105.        print("Ventilador apagado")
106.    def encender_cocineta():
107.        GPIO.output(cocineta, GPIO.HIGH)
108.        print("Cocineta encendida")
109.    def apagar_cocineta():
110.        GPIO.output(cocineta, GPIO.LOW)
111.        print("Cocineta apagada")
112.    #Convertimos los diferentes tipos de a solo
113.    formato
114.    def vad_audio_segments(audio_file,
115.                           sample_rate=16000):
116.        #sample_rate = 16000
117.        frame_win = sample_rate*0.05
118.        frame_step = sample_rate * 0.05
119.        tr = 0.0001
120.        wav_data = read_voice_file(audio_file,
121.                                   sample_rate=sample_rate)

```

```

118.         engery_list =
            total_energy_list(wav_data, sample_rate,
                               frame_win, frame_step)
119.         audio_segs, sil_segs =
            detect_audio_segment(engery_list, 0.05, tr)
120.         return audio_segs
121.     Realizamos una función para encontrar
        el audio y compararlo con la voz en vivo
122.     def find_command(cmd_data, seg_st, seg_end,
                          sample_rate):
123.         voz, cmdEngine.max_time =
            cmdEngine.find_cmd(cmd_data, seg_st, seg_end,
                               sample_rate)
124.     #Luego de filtrar el audio y compararlo se
        compara con el comando y se envia a llamar a la
        función de activación correspondiente
125.         if voz:
126.             print("cmd : {}".format(voz))
127.             if voz == "encender_luz_cuarto":
128.                 encender_luz_cuarto()
129.             if voz == "encender_luz_sala":
130.                 encender_luz_sala()
131.             if voz == "encender_luz_cocina":
132.                 encender_luz_cocina()
133.             if voz == "encender_luz_baño":
134.                 encender_luz_bano()
135.             if voz == "encender_ventilador":
136.                 encender_ventilador()
137.             if voz == "apagar_luz_cuarto":
138.                 apagar_luz_cuarto()
139.             if voz == "apagar_luz_sala":
140.                 apagar_luz_sala()
141.             if voz == "apagar_luz_cocina":
142.                 apagar_luz_cocina()
143.             if voz == "apagar_luz_baño":
144.                 apagar_luz_bano()
145.             if voz == "apagar_ventilador":
146.                 apagar_ventilador()
147.     #Se realiza un ciclo PWM para controlar los
        2 servomotores
148.         if voz == "abrir_cortina_cuarto":
149.             pwm.start(angulo(arrive))
150.             for pos in range(arrive,
                                depart, -incStep):
151.                 duty = angulo(pos)

```

```

152.             pwm.ChangeDutyCycle(duty)
153.             print(duty)
154.             time.sleep(delay)
155.         if voz == "cerrar_cortina_cuarto":
156.             pwm.start(angulo(depart))
157.             for pos in range(depart,
arrive, incStep):
158.                 duty = angulo(pos)
159.                 pwm.ChangeDutyCycle(duty)
160.                 print(duty)
161.                 time.sleep(delay)
162.         if voz == "abrir_cortina_sala":
163.             pwm2.start(angulo(arrive2))
164.             for pos2 in range(arrive2,
depart2, -incStep):
165.                 duty = angulo2(pos2)
166.                 pwm2.ChangeDutyCycle(duty)
167.                 print(duty)
168.                 time.sleep(delay)
169.
170.         if voz == "cerrar_cortina_sala":
171.             pwm2.start(angulo(depart2))
172.             for pos2 in range(depart2,
arrive2, incStep):
173.                 duty = angulo2(pos2)
174.                 pwm2.ChangeDutyCycle(duty)
175.                 print(duty)
176.                 time.sleep(delay)
177.         if voz == "encender_cocineta":
178.             encender_cocineta()
179.         if voz == "apagar_cocineta":
180.             apagar_cocineta()
181.         if voz == "abrir_puerta_principal":
182.             abrir_puerta_principal()
183.         if voz
== "cerrar_puerta_principal":
184.             cerrar_puerta_principal()
185.         if voz == "abrir_puerta_cuarto":
186.             abrir_puerta_cuarto()
187.         if voz == "cerrar_puerta_cuarto":
188.             cerrar_puerta_cuarto()
189.         #Aquí se realiza el cálculo y
comparación con el modelo
190.         for cmd_name in same_cmd:

```

```

191.         for idx in range(len(same_cmd[cmd_
            name]))):
192.             mfcc_feat, _, _ =
                mfcc_from_file(same_cmd[cmd_name][idx][0],
                    sample_rate=sample_rate)
193.             speech_segs =
                vad_audio_segments(same_cmd[cmd_name][idx][0],
                    sample_rate=sample_rate)
194.             assert speech_segs
195.             seg_st
                = int(speech_segs[0][0] / 0.01)
196.             seg_end = int(speech_segs[-
                1][1] / 0.01)
197.             mfcc_feat = mfcc_feat[:,
                seg_st:seg_end]
198.             same_cmd[cmd_name][idx].append(
                mfcc_feat)
199.             same_cmd[cmd_name][idx].append(
                speech_segs[-1][1] - speech_segs[0][0])
200.         return same_cmd
201.     class StreamingVAD:
202.         def __init__(self, sample_rate,
            seg_sil_time):
203.             self.proc_time = 0
204.             self.seg_sil_time = seg_sil_time
205.             self.sample_rate = sample_rate
206.             self.vad_data = []
207.             self.temp_data = []
208.             self.energy_list = []
209.             self.vad_segment = []
210.             self.vad = webrtcvad.Vad(3)
211.         def get_floatdata(self, frame):
212.             short_array = array.array('h',
                frame)
213.             float_array = []
214.             for sample in short_array:
215.                 if sample < 0:
216.                     float_array.append(float(sa
                mple / 32768.0))
217.                 else:
218.                     float_array.append(float(sa
                mple / 32767.0))
219.             return float_array
220.         def update_vad(self, float_data):

```

```

221.         # update audio_data
222.         frame_win = 0.05
223.         vad_tr = 0.0001
224.         self.temp_data += float_data
225.         vad_samples
    = int(self.sample_rate*frame_win)
226.         if len(self.temp_data) > vad_sampl
    es*5:
227.             frames
    = len(self.temp_data) // vad_samples
228.             self.vad_data += self.temp_dat
    a[:int(frames*vad_samples)]
229.             self.energy_list +=
    total_energy_list(self.temp_data[:int(frames*va
    d_samples)], self.sample_rate,
230.                    vad_samples, vad_samples).tolist()
231.             self.temp_data = self.temp_dat
    a[int(frames*vad_samples):]
232.             self.proc_time +=
    frames * frame_win
233.             audio_segs, sil_segs =
    detect_audio_segment(numpy.array(self.energy_li
    st), frame_win, vad_tr)
234.             if audio_segs and audio_segs[-
    1][1] + self.seg_sil_time < self.proc_time:
235.                 self.vad_segment = [audio_s
    egs[0][0], audio_segs[-1][1]]
236.                 #print("vad segment: [{}
    ,
    {}]".format(self.vad_segment[0],
    self.vad_segment[1]))
237.     #Se realiza la comparación del modelo y el
    audio
238.     def proc_frame(self, frame):
239.         float_data
    = self.get_floatdata(frame)
240.         self.update_vad(float_data)
241.         return self.vad_segment
242.     # Esta función resetea la variable que
    escucha el audio para una próxima lectura
243.     def reset_vad_seg(self):
244.         self.vad_segment = []
245.         self.vad_data = []
246.         self.energy_list = []

```

```

247.         self.proc_time = 0
248.     #Esta función realiza la adquisición de
        datos
249.         def get_command_data(self):
250.             cmd_data = []
251.             if self.vad_segment:
252.                 seg_st_idx
                = int(self.sample_rate * self.vad_segment[0])
253.                 seg_end_idx
                = int(self.sample_rate * self.vad_segment[1])
254.                 cmd_data
                = self.vad_data[seg_st_idx:seg_end_idx]
255.             return cmd_data
256.     class VoiceCmdEngine():
257.         def __init__(self, voice_cmds,
        same_model, diff_model, stream_vad):
258.             self.stream_vad = stream_vad
259.             self.cur_time = 0
260.             self.max_time = 0
261.             self.cmd_num = 0
262.             self.voice_cmds = voice_cmds
263.             self.same_model = same_model
264.             self.diff_model = diff_model
265.     # función para realizar el reconocimiento
266.         def find_cmd(self, speech_data, seg_st,
        seg_end, sample_rate):
267.             max_cmd_time = 0
268.             for cmd_name in self.voice_cmds:
269.                 for idx in range(len(self.voic
        e_cmds[cmd_name])):
270.                     cmd_feat
                = self.voice_cmds[cmd_name][idx][1]
271.                     cmd_len
                = self.voice_cmds[cmd_name][idx][2]
272.                     diff_time = abs(cmd_len
                - (seg_end-seg_st))
273.                     if max_cmd_time < cmd_len:
274.                         max_cmd_time = cmd_len
275.                     if diff_time < 0.3:
276.                         seg_st_idx
                = int(seg_st/0.01)
277.                         seg_end_idx
                = int(seg_end/0.01)

```

```

278.             similarity =
                similarity_mfcc(cmd_feat, seg_feat)
279.             similarity =
                numpy.array(similarity)
280.             cmd_score1
                = self.same_model.score(similarity) - self.diff
                _model.score(similarity)
281.             cmd_score2 = -1.000
282.             if cmd_score1 > 0.1:
283.                 similarity =
                    similarity_mfcc(seg_feat, cmd_feat)
284.                 similarity =
                    numpy.array(similarity)
285.                 cmd_score2
                    = self.same_model.score(similarity) - self.diff
                    _model.score(similarity)
286.                 if cmd_score2 > 0.1:
287.                     #if cmd_score1 > 0.1:
288.                         recog_cmd =
                            cmd_name
289.                         return recog_cmd,
                            max_cmd_time
290.                         #recog_cmd =
                            cmd_name
291.                         #break
292.                 return recog_cmd, max_cmd_time
293.         def get_command(self, frame):
294.             # vad segment
295.             vad_segment
                = self.stream_vad.proc_frame(frame)
296.             sample_rate
                = self.stream_vad.sample_rate
297.             # check vad segment
298.             if vad_segment:
299.                 cmd_data
                    = self.stream_vad.get_command_data()
300.                 seg_st, seg_end = vad_segment
301.                 self.stream_vad.reset_vad_seg()
302.                 print("detected seg: [{},"
                        "{}]".format(seg_st, seg_end))
303.                 #cmd_name, self.max_time =
                    self.find_cmd(cmd_data, seg_st, seg_end,
                    sample_rate)
304.                 #if cmd_name:

```

```

305.         cmdEngineThread =
           Thread(target=find_command, args=(cmd_data,
           seg_st, seg_end, sample_rate,))
306.         cmdEngineThread.start()
307.         elif self.stream_vad.proc_time > 5:
308.             self.stream_vad.reset_vad_seg()
309.
310.             stt_res = {"partial" : ""}
311.             return json.dumps(stt_res)
312.     #detecta la voz realiza la comparación con
           el modelo y las voces previamente grabadas
313.     def detect_cmd(voice_cmds, same_model,
           diff_model):
314.         sample_rate = 8000
315.         FORMAT = pyaudio.paInt16
316.         CHANNELS = 1
317.         SAMPLE_RATE = sample_rate
318.         CHUNK = int(SAMPLE_RATE / 10)
319.         audio = pyaudio.PyAudio()
320.         stream = audio.open(format=FORMAT,
321.                             channels=CHANNELS,
322.                             rate=SAMPLE_RATE,
323.                             input=True,
324.                             frames_per_buffer=CH
           UNK)
325.         sil_seg_time = 0.7
326.         global cmdEngine
327.         stream_vad = StreamingVAD(sample_rate,
           sil_seg_time)
328.         cmdEngine = VoiceCmdEngine(voice_cmds,
           same_model, diff_model, stream_vad)
329.     #Funcion principal que imprimeel resultado
           y llama a las diferentes funciones declaradas
           anteriormente
330.         while True:
331.             data = stream.read(CHUNK)
332.             if len(data) == 0:
333.                 break
334.             cmdEngine.get_command(data)
335.     if __name__ == "__main__":
336.         # load classification model
337.         same_model =
           load_gmm_from_npz("models/same_model.npz")
338.         diff_model =
           load_gmm_from_npz("models/diff_model.npz")

```

```
339.     cur_time= time.time()
340.     # load command directory
341.     cmd_dir = "./register"
342.     #cmd_dir = "./data/debug"
343.     voice_cmds = voice_cmd_files(cmd_dir)
344.     print("load time: {}".format(time.time()
    - cur_time))
```