



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL
FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA ELECTRÓNICA Y AUTOMATIZACIÓN**

TEMA:

Desarrollo de una propuesta de diseño para un sistema de automatización
de una casa inteligente

AUTOR:

Piedrahita Morán, Jandry Jossue

**Trabajo de Integración Curricular previo a la obtención del título de
INGENIERO EN ELECTRÓNICA Y AUTOMATIZACIÓN**

TUTOR:

Ing. Vega Ureta, Nino Tello. M.Sc.

Guayaquil, Ecuador

4 de septiembre del 2024



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

**FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA ELECTRÓNICA Y AUTOMATIZACIÓN**

CERTIFICACIÓN

Certificamos que el presente Trabajo de Integración Curricular fue realizado en su totalidad por el Sr. Piedrahita Morán, Jandry Jossue, como requerimiento para la obtención del título de INGENIERO EN ELECTRÓNICA Y AUTOMATIZACIÓN.

TUTOR

Ing. Vega Ureta, Nino Tello. M.Sc.

DIRECTOR DE CARRERA

Ing. Bohórquez Escobar, Celso Bayardo. PHD.

Guayaquil, 4 de septiembre del 2024



UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA ELECTRÓNICA Y AUTOMATIZACIÓN

DECLARACIÓN DE RESPONSABILIDAD

Yo, Piedrahita Morán, Jandry Jossue

DECLARO QUE:

El trabajo de Integración Curricular **“Desarrollo de una propuesta de diseño para un sistema de automatización de una casa inteligente”** previo a la obtención del Título de **Ingeniero en Electrónica y Automatización**, ha sido desarrollado respetando derechos intelectuales de terceros conforme las citas que constan en el documento, cuyas fuentes se incorporan en las referencias o bibliografías. Consecuentemente este trabajo es de mi total autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance del Trabajo de Integración Curricular referido.

Guayaquil, 4 de septiembre del 2024

EL AUTOR

Piedrahita Morán, Jandry Jossue



UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA ELECTRÓNICA Y AUTOMATIZACIÓN

AUTORIZACIÓN

Yo, Piedrahita Morán, Jandry Jossue

Autorizo a la Universidad Católica de Santiago de Guayaquil, la publicación, en la biblioteca de la institución del Trabajo de Integración Curricular: **“Desarrollo de una propuesta de diseño para un sistema de automatización de una casa inteligente”**, cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y total autoría.

Guayaquil, 4 de septiembre del 2024

EL AUTOR

Piedrahita Morán, Jandry Jossue



UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA ELECTRÓNICA Y AUTOMATIZACIÓN

REPORTE DE COMPILATIO



Reporte COMPILATIO del estudiante Piedrahita Morán, Jandry Jossue de la CARRERA DE INGENIERÍA ELECTRÓNICA Y AUTOMATIZACIÓN con el tema **“Desarrollo de una propuesta de diseño para un sistema de automatización de una casa inteligente”**, mismos que se encuentra al 4% de coincidencias.


Ing. Vega Ureta, Nino Tello. M.Sc.
DOCENTE-TUTOR

AGRADECIMIENTO

Quisiera brindar mi profundo agradecimiento a Dios y a mi familia por la ayuda durante toda mi carrera universitaria, a la Universidad Católica por darme la oportunidad de formarme profesionalmente, a toda la Facultad Técnica por confiar en mí y abrirme las puertas de la asombrosa carrera de la ingeniería electrónica y automatización

Mis agradecimientos a mi tutor, fue una pena no poder conocerlo durante los semestres anteriores, pero tuve la dicha de que al final fuera mi tutor del trabajo de integración curricular, siempre estaba dispuesto a corregirme y a pesar de todo a ayudarme con las ideas para mi proyecto, gracias por brindarme de su tiempo y orientarme en el trabajo.

Por último, expreso mi más sincero agradecimiento a cada uno de los ingenieros que me dieron clases durante toda la carrera, no solo compartiendo sus conocimientos sino también a madurar como persona, ser una persona de bien, responsable y valorar el sacrificio de mis padres, gracias, Ing. Bayardo, Ing. Zamora e Ing. Vega.

DEDICATORIA

Quiero empezar mi dedicatoria primeramente con Dios porque sin la ayuda de Él no sería todo esto posible, por haberme permitido llegar al final de una de mis metas académicas y continuar superándome, a mis padres por apoyarme y amarme siempre, ayudarme en perseguir mis sueños, en nunca quedarme con la duda sino en avanzar y superarme día a día, por trabajar duro y darme lo mejor de las herencias, que es el estudio, por decirme que todo lo que me proponga podría lograrlo, por los llamados de atención porque siempre ayudaran a formar poco a poco mi carácter y responsabilidad, tengo mucho que agradecerles.

A mis hermanos por ser parte de mi vida y estar conmigo siempre, a mi enamorada Doménica que es muy importante para mí, por siempre mostrarme su amor y apoyo incondicional, con sus palabras, risas y abrazos que me ayudaron mucho a seguir adelante. A toda mi familia por sus alegrías, apoyo, oraciones y consejos en jamás rendirme y mire el sacrificio que hacen mis padres para poder estudiar, por mis abuelitos y familiares que ya no están conmigo, pero siempre los recordare y llevare conmigo.

Finalmente, a mis amigos cercanos y no cercanos, por los que siguen y se fueron, gracias por brindarme buenos momentos y muchas risas, a los amigos de a u que conocí en los primeros semestres, en la mitad y al final de la carrera fue un gusto compartir clases con ustedes.



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

**FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA ELECTRÓNICA Y AUTOMATIZACIÓN**

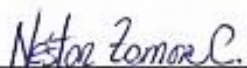
TRIBUNAL DE SUSTENTACIÓN

f. 

ING. BOHÓRQUEZ ESCOBAR, CELSO BAYARDO. PHD.
DIRECTOR DE CARRERA

f. 

ING. UBILLA GONZÁLEZ, RICARDO XAVIER, MSC.
COORDINADOR DE ÁREA

f. 

ING. ZAMORA CEDEÑO, NÉSTOR ARMANDO, MSC.
OPONENTE

ÍNDICE GENERAL

ÍNDICE DE FIGURAS	XII
INDICE DE TABLAS	XV
Resumen	XVI
Capítulo 1: Descripción general del trabajo de titulación	2
1.1 Introduucción	2
1.2 Antecedentes	2
1.3 Definición del Problema	3
1.4 Justificación del Problema	4
1.5 Objetivos del Problema de Investigación.....	4
1.5.1 Objetivo general	4
1.5.2 Objetivos específicos.....	5
1.6 Hipótesis	5
1.7 Metodología de Investigación	5
Capítulo 2: Fundamentación Teórica	7
2.1 Raspberry Pi 5	7
2.1.1 Diferencia de la Raspberry Pi 5 de los modelos anteriores	8
2.1.2 Principales usos de la Raspberry Pi 5	10
2.1.3 Diferentes modelos de Raspberry Pi 5	12
2.2 Controlador de motor L298N.....	14
2.3 Características del sensor TSL2561	16
2.4 Sensor de movimiento PIR HC-SR501	18
2.5 Especificaciones del sensor DHT22	20
2.6 Especificaciones del sensor de gas MQ-2	21

2.7 Tecnología utilizada en la domótica.....	23
2.7.1 Domótica en el ahorro de energía	26
Capítulo 3: Aportes de la investigación.....	29
3.1 Descripción de los elementos del sistema propuesto de la casa inteligente	32
3.1.1 Funciones y modelos de los componentes utilizados en el proyecto.....	35
3.2 Gestión de la ubicación de los elementos en el sistema propuesto	37
3.2.1 Descripción de la integración y comunicación de los sensores con los actuadores en la casa inteligente	37
3.2.1.1 Diagrama operacional del sensor de luz	37
3.2.1.2 Diagrama operacional del sensor de gas	38
3.2.1.3 Diagrama operacional del sensor de movimiento	40
3.2.1.4 Diagrama operacional del sensor de temperatura y humedad	41
3.3 Conexiones empleados en cada uno de los elementos del sistema de casa inteligente	42
3.3.1 Descripción de los voltajes y pines asignados de los elementos conectados a la Raspberry pi 5	44
3.4 Programación de la Raspberry Pi 5 en Python 3	46
3.5 Presupuesto y cronograma general del proyecto propuesto.....	57
3.5.1 Listado de costos del sistema de casa inteligente.....	57
3.5.2 Calendario de planificación de actividades para la instalación del sistema	58
Capítulo 4: Conclusiones y recomendaciones	60
4.1 Conclusiones	60

4.2 Recomendaciones	61
Bibliografías	62
Anexo 1.....	70
Anexo 2.....	77

ÍNDICE DE FIGURAS

Capítulo 2

Figura 2.1: Modelo básico de Raspberry Pi 5.	7
Figura 2.2: Alta demanda de fabricación de la Raspberry Pi 5.	8
Figura 2.3: Las señales basadas en redes neuronales artificiales (ANN) empleadas en la Raspberry Pi.	9
Figura 2.4: Partes de una Raspberry pi 3 modelo tipo B.	10
Figura 2.5: Lenguajes de programación más utilizados hasta el 2021.	11
Figura 2.6: Pruebas de rendimiento de la Raspberry pi 5 vs 4.	12
Figura 2.7: Puerto de salida de tipo RCA en la Raspberry pi.	13
Figura 2.8: Dimensiones de la Raspberry pi 5.	14
Figura 2.9: Módulo L298N controlador de motores	15
Figura 2.10: Representación de un PWM.	16
Figura 2.11: Sensor de luz digital TSL2561.	17
Figura 2.12: Protocolo de comunicación I2C.	18
Figura 2.13: Sensor de movimiento PIR HC-SR501.	19
Figura 2.14: Elementos del sensor de movimiento PIR HC-SR501.	20
Figura 2.15: Pines del sensor de temperatura DHT-22.	21
Figura 2.16: Funcionamiento del sensor de gas MQ-2.	22
Figura 2.17: Pines del sensor de gas MQ-2.	23
Figura 2.18: Tipos de enfoques de una casa inteligente.	24
Figura 2.19: Control de aparatos en una casa inteligente.	25
Figura 2.20: Interruptor wifi usado para el control de encendido y apagado de dispositivos eléctricos y electrónicos.	26

Figura 2.21: Incremento y uso de la domótica en los hogares a nivel mundial desde el 2017 hasta el 2025.	27
--	----

Capítulo 3

Figura 3.1: Plano del domicilio de 1 piso en AutoCAD.....	29
Figura 3.2: Diagrama de flujo del funcionamiento del equipo propuesta de casa inteligente con Raspberry pi 5.	33
Figura 3.3: Plano de ubicación de los elementos propuestos en AutoCAD.	37
Figura 3.4: Diagrama de flujo del funcionamiento del sensor de luz.	38
Figura 3.5: Diagrama de flujo del funcionamiento del sensor de gas.....	39
Figura 3.6: Diagrama de flujo del funcionamiento del sensor de movimiento.	40
Figura 3.7: Diagrama de flujo del funcionamiento del sensor de temperatura y humedad relativa.	41
Figura 3.8: Diagrama de conexiones de la Raspberry pi 5 con los sensores y actuadores para la casa inteligente.	42
Figura 3.9: Librerías utilizadas para la programación de los sensores y actuadores utilizados en el entorno de Arduino IDE.....	46
Figura 3.10: Programación de relés de control de luz y gas.	47
Figura 3.11: Programación de relés de control de movimiento, ventilación y ajustes del controlador del motor.	48
Figura 3.12: Programación de umbral de detección del sensor de luz y del sensor de gas.....	49
Figura 3.13: Programación del sensor de movimiento, de temperatura y humedad relativa.	50

Figura 3.14: Programación de apertura, cierre de cortinas del motor y control del relé de luz.	51
Figura 3.15: Programación del relé de gas, ventilación y alarma.....	51
Figura 3.16: Programación de estructura de envío de alertas por correo. ...	52
Figura 3.17: Programación de mensajes del sensor de luz.	53
Figura 3.18: Programación de mensajes del sensor de gas.	54
Figura 3.19: Programación de mensajes del sensor de movimiento.....	55
Figura 3.22: Programación de mensajes del sensor de temperatura y humedad relativa.	56
Figura 3.21: Programación de estructura de cierre y reset del código.....	56

INDICE DE TABLAS

Capítulo 3

Tabla 3.1: Funciones y nombres de los elementos utilizados para la propuesta de diseño de la casa inteligente.	36
Tabla 3.2: Voltajes y corrientes de operación de los elementos del proyecto.	45
Tabla 3.3: Tipos de pines utilizados en la Raspberry pi 5.	45
Tabla 3.4: Costo de los elementos para la propuesta de diseño.....	57
Tabla 3.5: Cronograma de estimación de actividades realizadas en 2 semanas para la instalación del sistema propuesto.....	58

Resumen

En el presente trabajo de integración curricular se basa en el diseño de un sistema de control de casa inteligente utilizando Raspberry Pi 5 y Python 3, integrando diversos sensores y actuadores para mejorar la seguridad, eficiencia energética y comodidad del hogar. En el objetivo se centra en establecer un sistema de monitoreo y control para dispositivos eléctricos mediante el uso de relés y sensores, elaborar la programación de Python 3 implementada para el funcionamiento del sistema inteligente y en base a esto diseñar el diagrama de conexiones de conexiones y flujo de funcionamiento del dispositivo. Con la presentación de este proyecto se pretende obtener el diseño de un sistema de control inteligente de un hogar, para la automatización de dispositivos electrónicos mediante la gestión de la Raspberry pi 5, para mejorar el confort de los residentes, a su vez obtener un ahorro de energía y mejorar la seguridad del domicilio a través del uso de sensores y actuadores. Se utiliza la metodología bibliográfica para la consultas teóricas de los distintos sensores a utilizar tales como sensores de movimiento, gas, temperatura y humedad, funcionamiento de actuadores como relés en la automatización, control de dispositivos, las características y funciones de las Raspberry pi 5 como dispositivo de control principal para la gestión eficiente de dispositivos.

Palabras claves: Raspberry pi 5, DHT-22, controlador L298, sensor TSL2561, sensor de gas MQ-2, sensor de movimiento PIR HC-SR501.

ABSTRACT

The present curricular integration work is based on the design of a smart home control system using Raspberry Pi 5 and Python 3, integrating various sensors and actuators to improve the security, energy efficiency and comfort of the home. The objective is focused on establishing a monitoring and control system for electrical devices through the use of relays and sensors, developing the Python 3 programming implemented for the operation of the intelligent system and based on this designing the connection diagram and operating flow of the device. With the presentation of this project, it is intended to obtain the design of an intelligent control system for a home, for the automation of electronic devices through the management of the Raspberry pi 5, to improve the comfort of the residents, in turn obtain energy savings and improve home security through the use of sensors and actuators. The bibliographic methodology is used for theoretical consultations of the different sensors to be used such as motion, gas, temperature and humidity sensors, operation of actuators such as relays in automation, device control, the characteristics and functions of the Raspberry Pi 5 as the main control device for efficient device management.

Keywords: Raspberry pi 5, DHT-22, L298 controller, TSL2561 sensor, MQ-2 gas sensor, HC-SR501 PIR motion sensor.

Capítulo 1: Descripción general del trabajo de titulación

1.1 Introducción

La evolución continua de la tecnología ha posibilitado la creación de sistemas de automatización en casa, que generalmente se conocen como casas inteligentes. Estos sistemas posibilitan el control y el monitoreo de diversas actividades del hogar, brindando una alta comodidad, protección y eficiencia energética. La Raspberry Pi 5, un computador económico con grande capacidad de procesamiento, mezclado con el idioma de programación Python 3, pone a disposición de los usuarios una superficie ideal para concebir un sistema de administración doméstica inteligente de gran utilidad.

1.2 Antecedentes

Un estudio abogó por un sistema de automatización basado en Raspberry Pi como la unidad central de control. Los sensores y actuadores se implementan para controlar los dispositivos eléctricos, y el trabajo se realizó para aumentar la eficiencia y la seguridad. La computación en el borde se implementó para garantizar que los datos recopilados se almacenen en el lugar del mono, lo que protege los datos privados (Stolojescu et al., 2021).

Una tesis proporcionó que los distintos tipos de sensores tales como de temperatura, humedad y movimiento, junto con actuadores como relé, pueden implementarse para reducir el consumo de energía y aumentar la

seguridad para los hogares. La programación se realizó en Python para adaptarse al control de los dispositivos presentes (Raju et al., 2021).

Un trabajo basado en Raspberry Pi incluye la monitorización de gas, temperatura, humedad y gestión de dispositivos eléctricos a través del relé. Se implementaron capacidades remotas y de automatización dentro del sistema para aumentar la seguridad y la comodidad. Un uso de tecnologías IoT y computación mediante Raspberry Pi y sensores para garantizar la seguridad y eficiencia energética del hogar, lo que reduce respectivamente el gasto de transmisión y almacenamiento de datos (Yar et al., 2021).

Una gestión de energía de la ciudad se propuso por un estudio a través del sistema de automatización basado en IoT en los hogares. Los sensores y actuadores se implementan para aumentar el monitoreo y control del consumo energético dentro de los hogares, que contribuyeron a la reducción del consumo y la demanda de energía máxima. Se propone un sistema de control del hogar que utiliza sistemas de IoT e implementa sensores de movimiento y gas para protección. Facilita el uso de Raspberry Pi como el controlador maestro para monitorear los dispositivos presentes (Ghoul & Naifar, 2024).

1.3 Definición del Problema

El eje principal del problema es la demandante necesidad de un sistema de administración de casa inteligente que sea mensurable, escalable y personalizable, utilizando tecnología de bajo costo. Muchos sistemas de comercio existentes son costosos y cerrados, esto limita la posibilidad de

personalización y crecimiento en función de las necesidades particulares del usuario.

1.4 Justificación del Problema

La justificación para este sistema de control de casa inteligente se apoya en múltiples beneficios fundamentales que contiene. En primer lugar, se relaciona con su viabilidad para combatir la ineficiencia energética mediante el control del apagado y encendido de los dispositivos electrónicos y electrodomésticos del hogar, lo que conduce a una disminución significativa en el consumo de energía y, en consecuencia, los costos. En segundo lugar, es su viabilidad para brindar seguridad en el domicilio y la capacidad de utilizar mensajes de alarma instantáneos en casos extremos, tales como un incendio, el ingreso no autorizado de desconocidos al domicilio, etc. En tercer lugar, proporciona una gran comodidad para el usuario, con la capacidad de reducir la carga de varios trabajos durante el día. Al mismo tiempo, dada la rentabilidad y la capacidad de adaptación a casi todos los aspectos de la vida cotidiana, constituye una idea absolutamente rentable y eficaz debido a su bajo costo para un gran grupo de personas.

1.5 Objetivos del Problema de Investigación

1.5.1 Objetivo general

Diseñar un sistema de control de casa inteligente utilizando Raspberry Pi 5 y Python 3, integrando diversos sensores y actuadores para mejorar la seguridad, eficiencia energética y comodidad del hogar.

1.5.2 Objetivos específicos

- Establecer un sistema de monitoreo y control para dispositivos eléctricos mediante el uso de relés y sensores.
- Definir la programación de Python 3 implementada para el funcionamiento del sistema de monitoreo y control de casa inteligente.
- Desarrollar el diagrama de conexiones de conexiones y flujo de funcionamiento del dispositivo.

1.6 Hipótesis

La implementación de un sistema de automatización basado en Raspberry Pi 5, integrado con sensores de luz, gas, movimiento, temperatura y humedad, permitirá optimizar el consumo energético del domicilio y aumentar la seguridad de sus ocupantes. Este sistema, al automatizar las tareas, como el control de iluminación, ventilación, alarmas y enviar alertas en tiempo real, conseguirá una gestión más eficiente de los recursos domésticos y una respuesta rápida ante posibles situaciones de riesgo, mejorando así la comodidad y seguridad del hogar inteligente.

1.7 Metodología de Investigación

Para este trabajo de integración curricular, se llevarán a cabo las distintas metodologías que contribuirán en el desarrollo del trabajo, tales como:

- La metodología bibliográfica para la consultas teóricas de los distintos sensores a utilizar tales como sensores de movimiento, gas, temperatura y humedad, funcionamiento de actuadores como relés en

la automatización, control de dispositivos, las características y funciones de las Raspberry pi 5 como dispositivo de control principal para la gestión eficiente de dispositivos (Castelo & Ramos, 2020).

- La metodología experimental en el desarrollo del diseño de las conexiones y funcionamiento de los distintos componentes a utilizar en el sistema de control para la casa inteligente (Loor, 2020).
- Se lleva a cabo el diseño del sistema basado en un modelo modular que integre hardware y software. En esta etapa, se define cómo los sensores y actuadores se interconectan con la Raspberry Pi 5 y cómo esta gestionará las diferentes funciones automatizadas de la casa.
- Análisis de costos y resultados para determinar la factibilidad económica y viabilidad del sistema para su posible implementación para el control de dispositivos electrónicos en un domicilio.

Capítulo 2: Fundamentación Teórica

2.1 Raspberry Pi 5

La última adición a la serie de minicomputadoras de placa única es un hito en el ámbito de la tecnología de computadoras de alto rendimiento y bajo costo. Contrariamente a sus predecesoras, esta versión no se limita a una actualización menor, sino que representa un adelanto significativo en términos de capacidad y eficiencia. Dotada de un innovador procesador de alta velocidad y GPU VideoCore VII, las Raspberry Pi 5 brinda excelentes prestaciones computacionales como se ve en la figura 2.1. Entre los ejemplos destacados de la aplicación de MS se incluyen el aprendizaje automático y la automatización doméstica, que requieren una capacidad gráfica adicional (Halfacree, 2023).

Figura 2.1: Modelo básico de Raspberry Pi 5.



Nota: Dispositivo con integración de PCIe 2.0. para mejora de rendimiento en proyecto de renderizado. Autor: (Roth, 2023)

Además, la potencia de visualización de la pantalla dual 4Kp60 también se sitúa entre los aspectos más notables. Otros ejemplos son la capacidad de

aplicaciones multimedia y educativas de alta definición. La eficiencia energética mejorada también hace que Raspberry Pi 5 sea atractiva para profesionales y educadores, este aspecto es especial. Entonces, al igual que con otros dispositivos similares, la Raspberry Pi 5 tiene una oportunidad significativa de convertirse en una innovadora herramienta la cual ha tenido una alta demanda en su producción, ya que inspira a profesionales en campos tan diversos como la ciberseguridad y la robótica a actuar con creatividad (Fezari & Dahoud, 2023).

Figura 2.2: Alta demanda de fabricación de la Raspberry Pi 5.



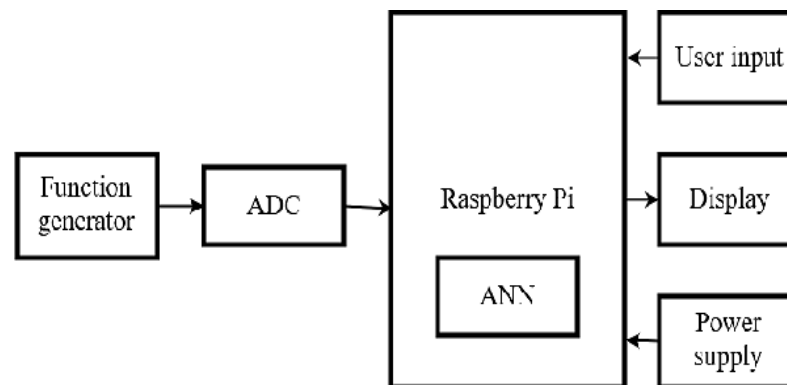
Nota: Fabricación de 90000 dispositivos, por las altas demandas de los consumidores. Autor: (Pastor, 2024)

2.1.1 Diferencia de la Raspberry Pi 5 de los modelos anteriores

Por otro lado, la Raspberry Pi 5 también se distingue significativamente de sus predecesoras por su ANN avanzada como se observa en la figura 2.3. En contraste con los modelos de la Raspberry Pi que utilizaban solo modelos RGB simples para el procesamiento de imágenes, el modelo 5 cuenta con una ANN de cinco clases. Como resultado, tiene la función de realizar tareas computacionales comparativamente más complejas con un nivel mucho más

alto de precisión. Esto amplía significativamente las posibilidades de aplicación del modelo actual. Para empezar, la ANN más compleja en el Raspberry Pi 5 es un gran paso adelante en términos de la ANN del Raspberry Pi 3 Modelo B (Shahi, 2024).

Figura 2.3: Las señales basadas en redes neuronales artificiales (ANN) empleadas en la Raspberry Pi.

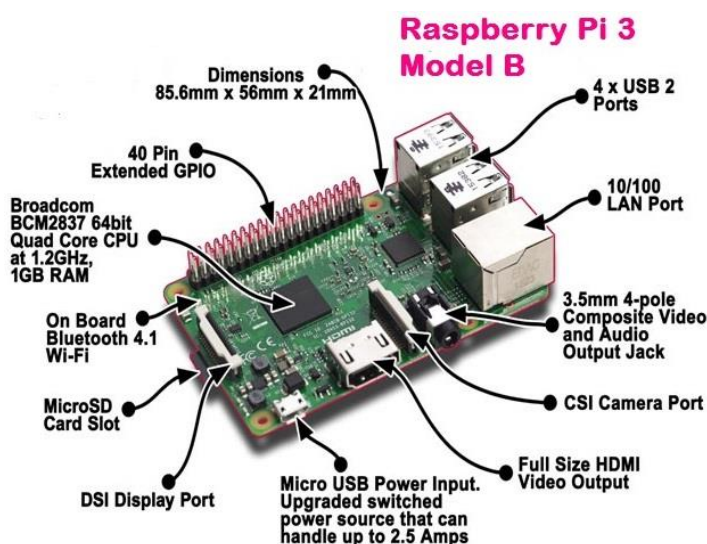


Nota: Algoritmos de inteligencia artificial utilizados en procesamiento de las Raspberry Pi. Autor: (Bharadwaja et al., 2018)

Es decir, el modelo 3 de tipo B era popular y versátil como se observa en la figura 2.4, se usó a menudo en numerosas aplicaciones, pero sus oportunidades de motores de aprendizaje y de anillador de imágenes eran deficientes desde el punto de vista en comparación con los del modelo actual. Por ejemplo, la ANN incorporada más compleja y eficiente en el modelo actual no suma solo el trabajo que el modelo previo podía hacer, sino también la eficiencia de ese trabajo. Y esto se debe a la arquitectura avanzada del proceso en el que se administra algoritmos más complejos y manejo de datos. En otras palabras, la ANN es súper rápida y ágil. Por lo tanto, es ampliamente utilizado en inteligencia artificial, aprendizaje profundizado y minería de datos.

Por lo tanto, esta actualización de hardware y algoritmo nuevamente hace que la Octava sea virtualmente perfecta para tareas de datos y analítica avanzada (Papakyriakou & Barbounakis, 2023).

Figura 2.4: Partes de una Raspberry pi 3 modelo tipo B.



Nota: Elementos y características técnicas de la Raspberry pi 3 tipo B.

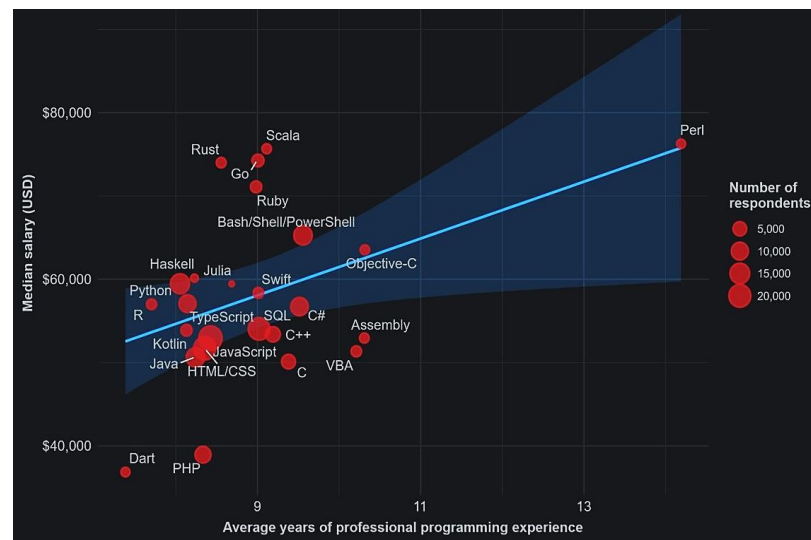
Autor: (Lokhande, 2020)

2.1.2 Principales usos de la Raspberry Pi 5

La Raspberry Pi 5 ha encontrado numerosas aplicaciones en una variedad de dominios debido a sus amplios conjuntos de características y capacidades. Uno de sus principales usos en la educación, donde se emplea como plataforma barata y de fácil acceso para enseñar a estudiantes de programas y electrónica. La capacidad del dispositivo para ejecutar una variedad de lenguajes de programación, como Python, Java, C++, entre otros como se observa en la figura 2.5, y para soportar numerosos programas educativos, hace que sea una opción lucrativa para los novatos y estudiantes experimentados. Los profesores suelen utilizar Raspberry Pi 5 para organizar

lecciones prácticas en las que los estudiantes aprovechan la oportunidad para probar el hardware y el software en ambientes seguros y experimentales (Smashing, 2023).

Figura 2.5: Lenguajes de programación más utilizados hasta el 2021.

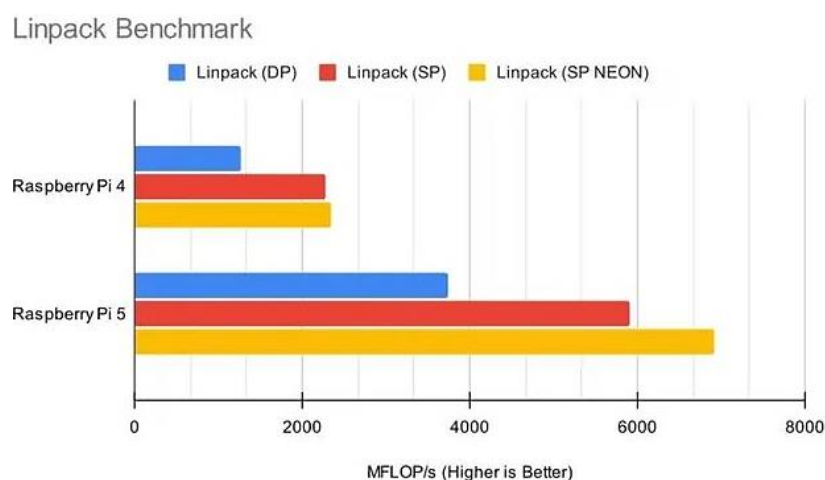


Nota: Rentabilidad y utilidad de los lenguajes de programación en la actualidad hasta el 2021. Autor: (Pastor, 2021)

Además, cada vez se utiliza más en el desarrollo de proyectos de Internet de las cosas. debido a su tamaño reducido, bajo consumo de energía y buena capacidad de procesamiento. Así, ayuda a aceptar y transmitir información a través de una red de sensores y actuadores que resulta crucial en aplicaciones que requieren desde la automatización del hogar hasta la vigilancia de los parámetros ambientales. Dada la posibilidad de conectar varios sensores y dispositivos, Raspberry Pi 5 se convierte en una plataforma de elección para el desarrollo de distintos tipos de proyectos centrados en IoT lo cual permite la implementación de soluciones inteligentes y eficientes (Lauer, 2023).

La Raspberry Pi 5 también se puede utilizar para aplicaciones de medios multimedia gracias a sus capacidades de procesamiento y descripciones de imágenes mejoradas. Adecuada para el ámbito digital, estaciones de medios y videojuegos ligeros, su capacidad para mostrar contenidos de alta definición y ejecutar aplicaciones de medios sofisticadas ofrece la posibilidad de ser implementado en entornos diversos, incluyendo kioscos interactivos y sistemas de entretenimiento en el hogar. La capacidad de salida de la pantalla dual 4K a 60 Hz mejora significativamente la experiencia de usuario final. Es vital con ellas la adopción de software y herramientas de gestión como APT que facilitan la instalación, actualización y mantenimiento de los paquetes de software (Bauduin, 2024).

Figura 2.6: Pruebas de rendimiento de la Raspberry pi 5 vs 4.



Nota: Pruebas de potencia en velocidad de procesamiento de la Raspberry pi 5 vs su modelo anterior. Autor: (González, 2023)

2.1.3 Diferentes modelos de Raspberry Pi 5

La gama de modelos Raspberry Pi 5 cubre muchas necesidades, con cada modelo que presenta sus características y capacidades según el tipo de

proyecto. Raspberry Pi Modelo B es uno de los modelos destacados, ya que presenta una salida RCA 1 como se observa en la figura 2.7. En consecuencia, se considera idóneo para proyectos con salida de vídeo simple, como las consolas de juegos retro o centros multimedia simples. Asimismo, es fácil de conectar con visualizadores antiguos y sistemas de entretenimiento sencillos y, por lo tanto, es rentable y eficaz para aplicaciones sin una alta resolución . Además de su salida RCA, el Modelo B es común en los proyectos de desarrollo debido a su versatilidad (Farnell, 2022).

Figura 2.7: Puerto de salida de tipo RCA en la Raspberry pi.



Nota: Salida de audio bidireccional y video entregado por el puerto RCA de Raspberry pi. Autor: (Wikilibros, 2020)

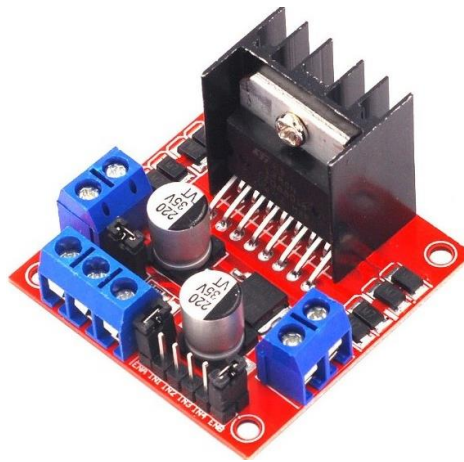
Conecta varios tipos de entradas y conectores, lo que significa que se utiliza en diferentes contextos. Esta versatilidad es el motivo de que sea común y les permite a los profesionales o entusiastas de la tecnología trabajar con mucho hardware y configuraciones. Otro aspecto común de algunos de los modelos de Raspberry Pi 5 los cuales mantiene las mismas dimensiones de tamaño como se observa en la figura 2.8 y solo varia en su memoria RAM, pueden admitir más datos de entrada de imágenes. Se planearon algunos aspectos con avanzados para proyectos de visión por computadora e



El controlador de motor L298N como se observa en la figura 2.9 es

La posibilidad más destacada del L298N es que se puede controlar con señales PWM, lo que da una amplia gama de control para la velocidad y par del motor, cuando se ajusta el ciclo de trabajo de la señal de entrada. Esta característica es crucial en sistemas como la administración de motores eléctricos en los robots donde se necesita un rendimiento constante del motor para movimientos precisos concretos (Triacs, 2021).

Figura 2.9: Módulo L298N controlador de motores .

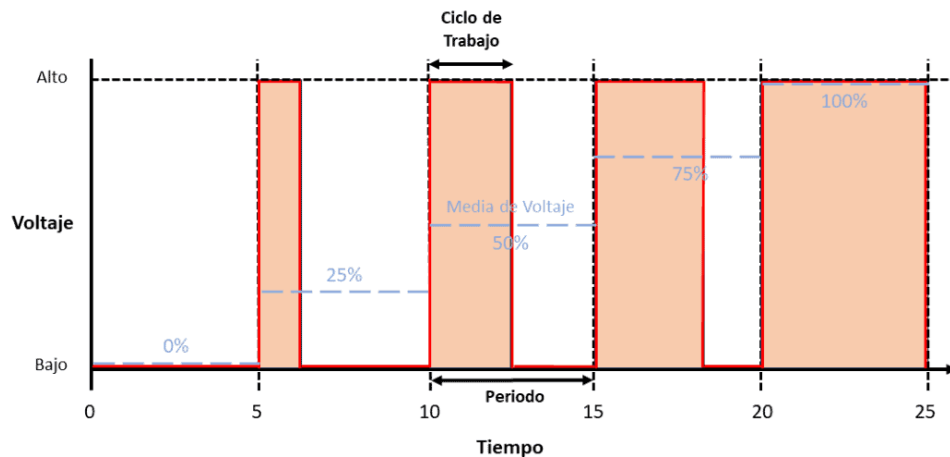


Nota: Driver de motores y servomotores con rango de operación de 5V a 35V en DC. Autor: (Terce, 2020)

También es seleccionado para tales aplicaciones, como manejar motores a través del PWM como se observa en la figura 2.10 porque puede satisfacer los requisitos eléctricos. Este está controlado por un transistor Darlington, lo que permite a los pines de entrada del controlador cumplir con los requisitos eléctricos necesarios para la compatibilidad y es de eficiencia. Esto incluye la capacidad para soportar los niveles de corriente y voltaje para asegurar eficiencia y larga vida del motor. Además, para el L298N, hay un amplio Vs de suministro, que es un Vs máximo de 35 V a un mínimo de 5 V,

lo que relaja la capacidad de utilizar diversas fuentes de alimentación en muchas aplicaciones para garantizar que satisfaga los límites electrónicos en el sistema (Gustama et al., 2024).

Figura 2.10: Representación de un PWM.



Nota: Principio del funcionamiento de la modulación de ancho de pulso en los dispositivos. Autor: (Mariete, 2023)

2.3 Características del sensor TSL2561

El sensor TSL2561 como se observa en la figura 2.11 es un sensor avanzado de luz digital, reconocido por su versatilidad y precisión al medir la luz ambiental. Su sistema de detección de luz dual incorpora dos sensores independientes de luz visible e infrarroja. Esto permite capacidades exhaustivas de medición ya que el sensor TSL2561 es ideal para detectar cualquier tipo de radiación luminosa. Esta multifuncionalidad de detección se justifica en la respuesta plana del sensor a la luz visible con detección doble. Por tanto, el sensor puede leer al máximo hasta 0.1 Lux y superar los 40,000 Lux. Esta capacidad abarca tanto aplicaciones de consumo, como la regularización de la retroiluminación automática de los dispositivos móviles, hasta aplicaciones industriales, donde se demanda con fines de regulación y

seguimiento de las condiciones de iluminación. En ambos casos, un rango de lectura de luz tan amplio permite una fácil adaptación del sensor a diversos entornos de operación (Kuswanto & Mulaab, 2023).

Figura 2.11: Sensor de luz digital TSL2561.

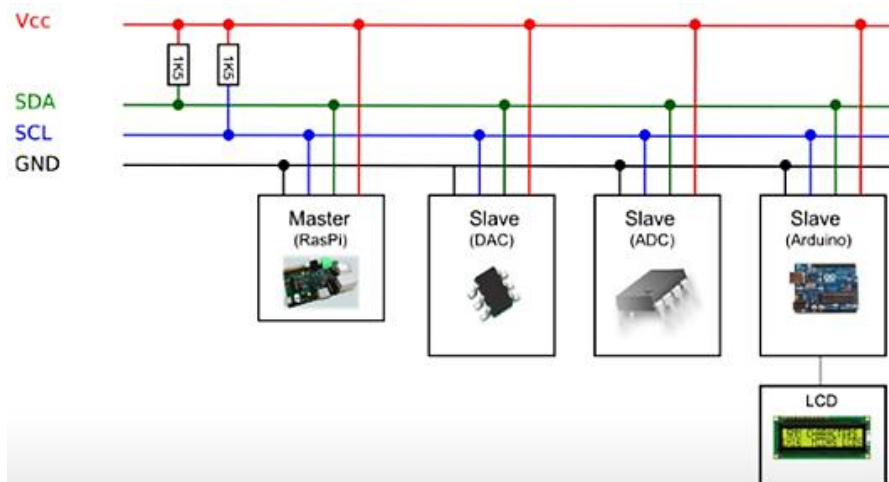


Nota: Sensor de luz con rango de funcionamiento de 3.3V con sensibilidad máxima de 40000 lux. Autor: (Ada, 2020)

Además, el sensor TSL2561 está provisto de un puerto de comunicaciones I2C como se observa en la figura 2.12 que se utiliza y facilita la integración en sistemas incorporados y plataformas de microcontroladores. La interfaz digital I2C reduce el esfuerzo de conexión y permite múltiples direcciones, de hasta tres sensores conectados simultáneamente. Esto es especialmente relevante para las aplicaciones que requieren la monitorización de varias áreas o condiciones luminosas a la vez. Por último, la alta precisión del sensor, respaldada por la configuración de ganancia y tiempos de adquisición, permite una adaptación aún mayor. Los ajustes en estos parámetros permiten a los desarrolladores personalizar el proceso de medición y optimizarlo para sus aplicaciones específicas. En conjunto, todas

las capacidades de detección mencionadas convierten al sensor TSL2561 en una excelente opción para quienes buscan un sensor de luz adaptable y preciso que pueda operar en condiciones cambiantes (Lean et al., 2024).

Figura 2.12: Protocolo de comunicación I2C.



Nota: Método de comunicación serial utilizado entre microcontroladores para la lectura y escritura de datos. Autor: (Murky, 2020)

2.4 Sensor de movimiento PIR HC-SR501

El HC-SR501 como se observa en la figura 2.13 es un sensor de movimiento infrarrojo pasivo más popular. Estas especificaciones generan la mayor parte del interés cuando se busca un sensor de movimiento. Tal característica del sensor se justifica concentrada de este modo, ya que un ángulo de detección de aproximadamente 110 grados es suficiente para cubrir un área considerable. Es decir, el HC-SR501 tiene una amplia cobertura, lo que demuestra lo adecuado que es para sistemas de iluminación automática y de seguridad, donde hay un espacio suficientemente grande que cubrir. Además, su rango de detección es de unos 7 metros, lo que contribuye a una cobertura más amplia que en los sensores HC-SR505 y AM312. Un rango de

detección tan amplio es críticamente importante en escenarios donde el sensor debe cubrir un amplio espacio o un edificio grande (Mendoza & Quevedo, 2023)

Figura 2.13: Sensor de movimiento PIR HC-SR501.

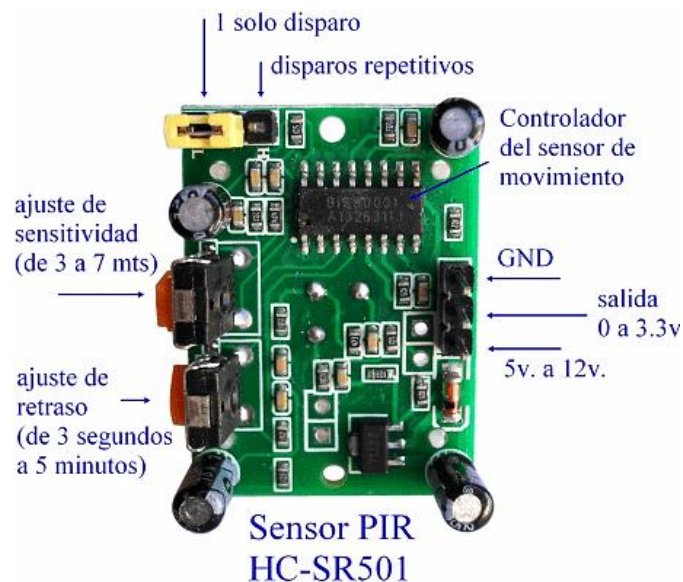


Nota: Sensor de movimiento con rango de detección hasta de 5 metros con un rango de detección de 120 grados. Autor: (Velasco, 2022)

Es similarmente útil ya que puede ajustar la sensibilidad y tomar el tiempo para lo suficiente. En este sentido, el retardo de tiempo se puede establecer de 1 a 300 segundos a través del giro de los tornillos que posee el dispositivo los cuales gradúan el tiempo o retardo en cada rango de detección que perciba el dispositivo como se observa en la figura 2.14, y la recomendación es mantenerlo en 3 segundos para evitar activaciones falsas y asegurar el funcionamiento estable de los sistemas . Del mismo modo, el potenciómetro de sensibilidad del HC-SR501 se puede ajustar a cualquier posición que prefiera el usuario o detenerse en el punto medio si no está seguro. En este caso, se puede obtener un equilibrio en el nivel de sensibilidad. Además, para garantizar el funcionamiento adecuado del dispositivo en escenarios donde es vital detectar el movimiento varias veces, el puente debe colocarse en la posición H, cabe destacar que el dispositivo

opera entre distintos rangos de voltaje de 5 a 20V, lo que lo hace ideal para cualquier tipo de microcontrolador al tener un gran rango de funcionamiento por su voltaje (Ventre et al., 2024).

Figura 2.14: Elementos del sensor de movimiento PIR HC-SR501.



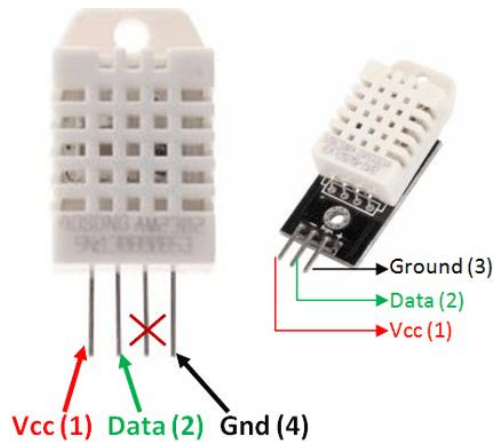
Nota: Configuración de retardo de sensado y sensibilidad del sensor de movimiento. Autor: (330ohms, 2024)

2.5 Especificaciones del sensor DHT22

En el ámbito de la monitorización ambiental, el sensor DHT22, o AM2302 como se observa en la figura 2.15, también se destaca por su capacidad para medir con precisión tanto la temperatura como la humedad relativa que detecta de su entorno. El DHT22 utiliza un sensor de humedad capacitivo y un termistor, lo que hace que el sensor combine estos dos componentes para brindar lecturas fiables y detalladas. Este sensor es beneficioso al capturar datos precisos, que son críticos para proyectos que requieren monitoreo ambiental preciso. El sensor DHT22, por otro lado, tiene la capacidad de almacenar esta salida digital en el pin de datos. La ventaja de

esta salida digital es que simplifica la interfaz del sensor con el microcontrolador (Belford, 2023).

Figura 2.15: Pines del sensor de temperatura DHT-22.



Nota: Reducción de los 4 pines del sensor de temperatura a 3 pines con la integración de su módulo digital. Autor: (Mutou, 2022)

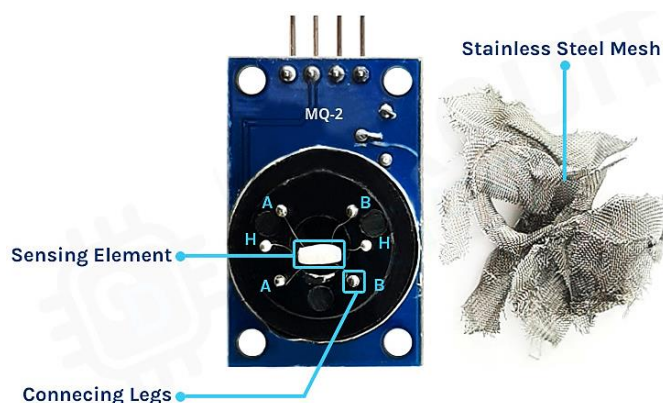
Además, la salida digital puede entrar sin entradas analógicas. El DHT22 es una mejor versión del DHT11, ya que brinda al usuario mayor resolución y precisión, lo que lo convierte en la mejor opción para proyectos que requieren mediciones más integrales y adecuadas. Ambos sensores son rentables, ya que se pueden comprar a un bajo costo y tienen diferentes aplicaciones que pueden beneficiarse de sus capacidades de medición precisa y en tiempo real. No obstante, la velocidad con la que el DHT22 puede adquirir estos datos puede ser una limitación (Yida, 2020).

2.6 Especificaciones del sensor de gas MQ-2

El sensor de gas MQ-2 es un dispositivo ampliamente utilizado y altamente versátil para detectar gases inflamables y peligrosos utilizado como componente en un sistema de seguridad, como alarma de fugas de gas para

el hogar y un detector industrial. El sensor MQ-2 es un sensor de gas MQ-2 con un mecanismo de funcionamiento electroquímico que emplea un material de detección compuesto principalmente de dióxido de estaño SnO_2 , que se prepara en capas sobre un microtubo de alúmina Al_2O_3 . El MQ-2 detecta varios gases peligrosos e inflamables, entre ellos. La respuesta se basa en la variación de la conductividad eléctrica del dióxido de estaño en presencia de diferentes gases. El detector se basa en una variación de la resistencia en función del gas. Bajo la influencia del gas inflamable y peligroso como GLP gas licuado de petróleo, i-butano, propano, metano, alcohol, hidrógeno y humo, la resistencia del detector está afectada (Made et al., 2024).

Figura 2.16: Funcionamiento del sensor de gas MQ-2.

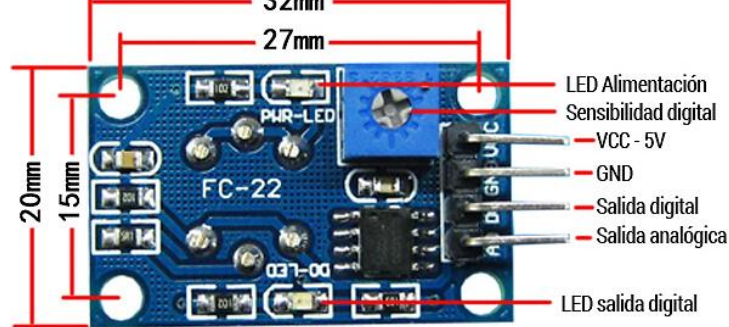


Nota: Elementos que permiten al sensor de gas detectar los diferentes tipos de gases del entorno que monitorea. Autor: (Ecuarobot, 2022)

Bajo la influencia de los gases anteriores, la resistencia disminuye y, por lo tanto, la capacidad de conducción aumenta. El circuito en el sensor que convierte la variación en datos utilizables mide este cambio. La variación puede medirse en un rango de concentración desde 300 hasta 10,000 ppm. Para garantizar que los componentes expuestos de sensibilidad funcionen en



22mm



la comodidad en el hogar, lo que puede ser esencial en la gestión y la organización residencial como se observa en la figura 2.18. El sistema obtiene información de la recopilación en tiempo real de sensores dispersos por toda la casa. Los monitores de sensores incluyen variaciones de temperatura a nivel de sala, fugas de agua y gas, etc. (Palacios, 2020).

Figura 2.18: Tipos de enfoques de una casa inteligente.



Nota: Diferentes tipos de características que suele poseer un casa o domicilio inteligente. Autor: (Sarachu, 2024)

La recopilación de esta información ofrece una descripción general del sistema que emplea el sistema para tomar decisiones automáticas en función de la situación actual. Por ejemplo, si el sistema detecta una fuga, el agua puede cerrar automáticamente las válvulas de líquido y notificar al propietario a través de su dispositivo como se observa en la figura 2.19. También asegura que el desperdicio no ocurra en caso de incidente. La integración de este sistema a través de varios dispositivos ofrece la oportunidad de una regulación óptima del consumo de energía. La interconectividad es, por lo tanto, esencial para el funcionamiento del sistema porque la comunicación en tiempo real

entre dispositivos permite que los dispositivos accionen y actúen en este sistema. El sistema se basa en varias conexiones de plataformas, posiblemente incluyendo Internet, Bluetooth y radiofrecuencia, para el control y la activación remotos. Esto proporciona una oportunidad para que el usuario supervise su hogar, incluso cuando no está en casa. La integración de la conectividad a esta red amplía la personalización y la capacidad del usuario para el control remoto (Quishpe & Hernández, 2023).

Figura 2.19: Control de aparatos en una casa inteligente.

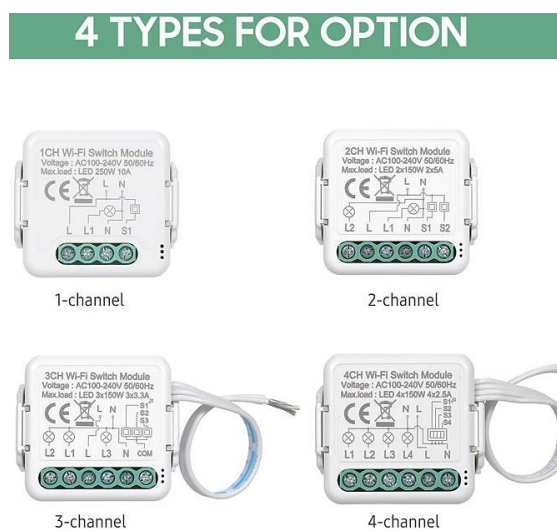


Nota: Distintos eléctricos y electrónicos que se suelen usar en el control de una casa inteligente. Autor: (Acosta, 2023)

Además, las tecnologías avanzadas recientes también han integrado esta iniciativa aún más para los hogares. Suponiendo que el propietario adquiere un dispositivo avanzado, como un televisor inteligente o dispositivos de entretenimiento, el uso del sistema de control centralizado puede ser posible para integrar su uso en cualquier lugar sin la necesidad de otros controles como se observa en la figura 2.20. Además, la funcionalidad incorporada proporciona una energía generalizada eficiente en el hogar a

través de la centralización del sistema. Por lo tanto, la tecnología avanzada utilizada en esta situación implica que la infraestructura como la red de comunicación esté a nivel para garantizar la estabilidad de funcionamiento en el hogar automatizado. La implementación exitosa de los hogares automatizados implica configurar una red de comunicaciones fuerte y de alta velocidad para hacer el trabajo y en caso de problemas para corregir el problema antes de que ocurra (López, 2023).

Figura 2.20: Interruptor wifi usado para el control de encendido y apagado de dispositivos eléctricos y electrónicos.



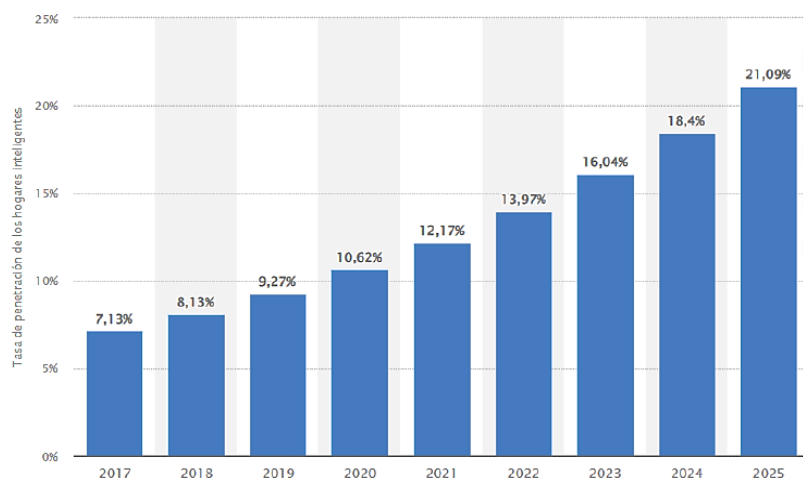
Nota: Módulo wifi compatible con entradas de 100V a 240V con una capacidad máxima de 10A. Autor: (Aurrera, 2024)

2.7.1 Domótica en el ahorro de energía

Un papel crucial en la promoción de la eficiencia energética cumple la domótica, que permite el control preciso y automatizado de los electrodomésticos y los sistemas de la casa. En otras palabras, la domótica no exige a las personas reemplazar los electrodomésticos actuales, sino optimizar su consumo de energía. La automatización brinda una consciencia

y un control aumentados sobre la electricidad utilizada, lo que da como resultado una disminución en su consumo y en los costos asociados con los que usan, haciendo que se use cada vez estos sistemas como se observa en la figura 2.21. Otro enfoque principal de la eficiencia energética es la racionalización de las cargas eléctricas. Por lo tanto, basándose en cuánta electricidad están tomando ciertas áreas de la casa, la domótica tiene la capacidad de apagar automáticamente equipos no esenciales. Por ejemplo, si el sistema descubre que ciertos electrodomésticos no se utilizan o si el crecimiento de la demanda de electricidad ha alcanzado cierto punto, dicho dispositivo los desactivará para evitar que consuman energía. Esto no solo reduce el uso, sino que, de hecho, hace que los aparatos duren más al evitar que estén encendidos a lo largo de los días de la semana (Santiago, 2021).

Figura 2.21: Incremento y uso de la domótica en los hogares a nivel mundial desde el 2017 hasta el 2025.



Nota: Crecimiento del uso de los sistemas inteligentes utilizados en los hogares por parte de sus residentes. Autor: (Yoigo, 2022)

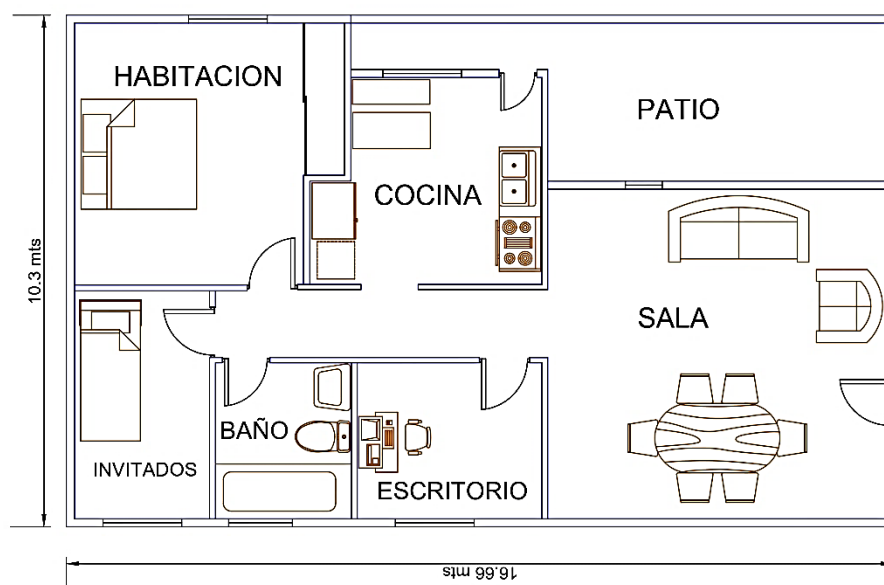
La automatización programada y zonificada de sistemas de climatización es otro caso de cómo la domótica conduce a una disminución

en el uso de la energía. Limita la refrigeración y la calefacción a áreas específicas que necesitan realmente a través de la ciencia como la ocupación o la temperatura, lo que reduce la necesidad planta para la comodidad. El uso reducido se extiende pues optimiza la utilidad de los sistemas de calefacción y refrigeración para mejorar la comodidad de. ayudar en la reducción del consumo de energía. Por ejemplo, la domótica ahorra el consumo de electricidad. Estos sistemas ajustan la luz utilizada en momentos designados a según la disponibilidad de luz solar. Los días soleados pueden provocar la reducción de alimentación. La combinación de todos estos sistemas y muchos otros desarrollados en la domótica lista para usar, como la optimización mediante el control del gasto y la combinación de energías renovables e iluminación, hace que los hogares son más eficientes energéticos. Todos estos ayudan disminuir las facturas de los hogares y en términos generales la comodidad de incluso por. Además, la domótica apoya la integración de tecnologías sostenibles y el aumento del uso a través del uso personal (Benítez, 2022).

Capítulo 3: Aportes de la investigación

En el presente capítulo se basa en el análisis y diseño de una casa inteligente con Raspberry Pi 5, se incluye una extensa investigación y desarrollo en lo que respecta a una propuesta de diseño de una solución tecnológica avanzada que busca automatizar y mejorar un hogar moderno existente como se observa en la figura 3.1 y anexos. Los componentes esenciales que se seleccionan con cuidado para realizar esta tarea incluyen la Raspberry Pi 5, que se desempeña como núcleo del sistema.

Figura 3.1: Plano del domicilio de 1 piso en AutoCAD.



Elaborada por: Autor

Como se muestra en la figura A1 de los anexos el diagrama unifilar en AutoCAD de la vivienda. Este diagrama proporciona detalles generales sobre la parte eléctrica de una casa individual de un piso. Cada sección del gráfico ejemplifica un componente del sistema eléctrico:

- **C1 Alumbrado:** Circuito destinado a la iluminación general de la casa.

- **C2 T. Corriente:** Circuito de corriente general para uso en enchufes y aparatos eléctricos.
- **C3 Cocina, Horno:** Circuito específico para la cocina, incluyendo el horno.
- **C4 Lavadora - Patio:** Circuito para la lavadora y el área del patio.
- **C5 Baño y Escritorio:** Circuito que abarca tanto el baño como el área del escritorio.

Los detalles técnicos del diagrama incluyen:

- **2*1.5+T:** Indica un cable de sección 1.5 mm² con una protección tipo T.
- **2*2.5+T:** Cable de sección 2.5 mm² con protección tipo T.
- **2*6+T:** Cable de sección 6 mm² con protección tipo T.
- **10A, 16A, 25A, 20A, 20A:** Los distintos tipos de amperajes de las protecciones de cada área.
- **25A /30mA:** Son los interruptores diferenciales de 25A que tiene una sensibilidad de 30mA.
- **IG 25A:** Interruptor general de 25A.

En la figura A2 de los anexos se observa es el diagrama de Circuitos de iluminación y enchufes de la vivienda de un solo piso en AutoCAD. Muestra la forma en que los circuitos están distribuidos en diferentes áreas de la casa, las cuales son:

- **Habitación:** Circuitos para iluminación y enchufes en las habitaciones.
- **Patio:** Circuito para el área exterior, incluyendo luces y enchufes.
- **Cocina:** Circuito específico para electrodomésticos de cocina, como el horno.

- Sala: Circuito para iluminación y enchufes en la sala de estar.
- Invitados: Circuito para áreas de descanso o habitaciones para invitados.
- Baño: Circuito específico para el baño.
- Escritorio: Circuito para iluminación y enchufes en el área del escritorio.

El capítulo también se aborda la configuración física y eléctrica detallada que se realiza para asegurarse de que cada componente este debidamente conectado a la Raspberry Pi 5 mediante los pines GPIO correctamente asignados. Esto garantiza la comunicación segura entre los dispositivos y el sistema central, del que dependen para su funcionalidad.

Adicionalmente en la figura A2 y A3 de los anexos, se aprecia respectivamente la ubicación del router y repetidor para la señal wifi del domicilio, la cual es necesaria ya que el sistema domótico a proponer depende de la conexión a la red wifi para él envío de notificaciones del usuario en cual parte del mundo y a sus vez se proporciona un sistema de cámaras inteligentes para la vigilancia del entorno del domicilio, con el fin de mejorar la seguridad de la casa, ante cualquier caso de intromisión o robo por parte de personas desconocidas.

Además, se desarrolla el software personalizado escrito en el lenguaje de programación Python 3 para permitir el control remoto de estos dispositivos y el monitoreo en la Raspberry Pi 5. Puede inferir en la automatización de datos por medio de los sensores, el envío de correos electrónicos de alerta y

finalmente utiliza su dirección IP para comunicarse con la interfaz, lo cual se describe a continuación.

3.1 Descripción de los elementos del sistema propuesto de la casa inteligente

A continuación, se presenta una descripción de los elementos empleados en el sistema de la casa inteligente que están en funcionamiento, en la figura 3.2 se presenta el diagrama de flujo.

Raspberry Pi 5:

- **Función:** Actuar como la unidad de control central donde se integran lecturas de sensores y actuadores.
- **Descripción:** Es una mini PC que proporciona potencia de procesamiento y capacidades de conectividad para el sistema. Este componente incluye un puerto de propósito general de E/S que además se diseña para interconectar el sistema a través del puerto GPIO, el cual se puede conectar a varios sensores y controladores.

Sensor De Luz TSL2561:

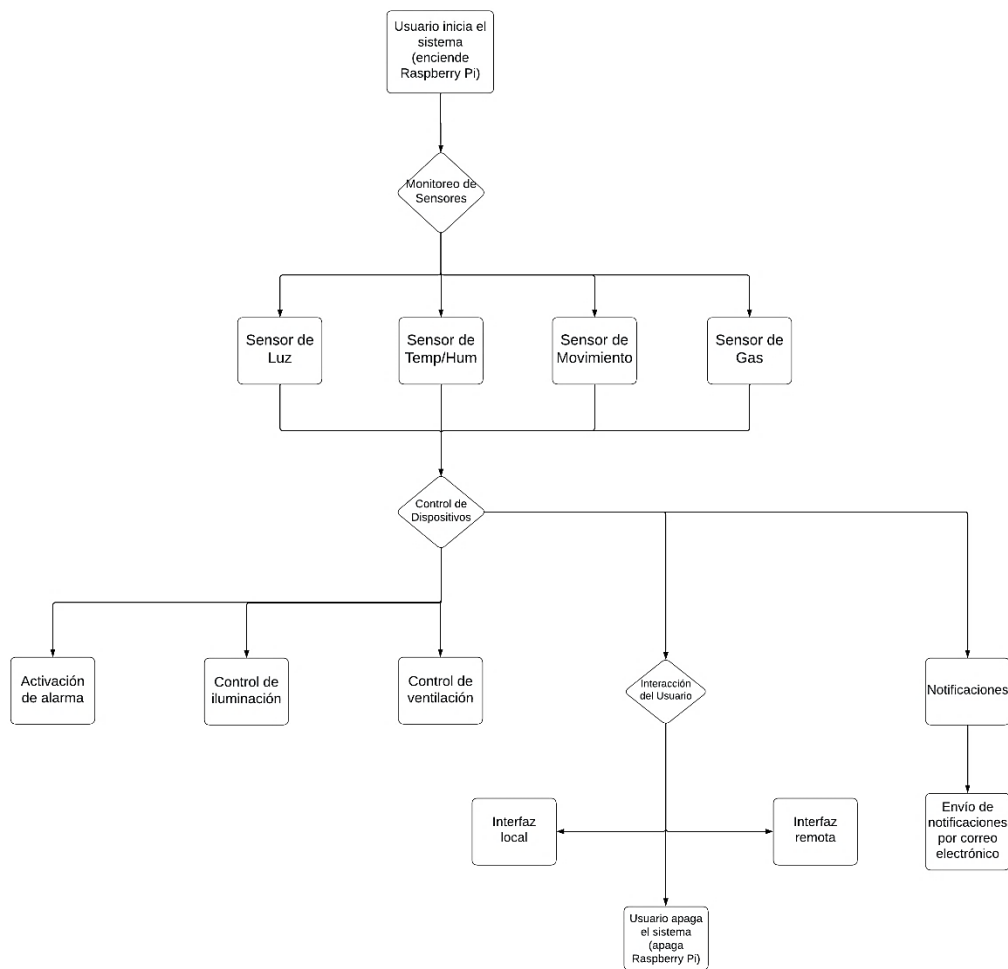
- **Función:** Detección del nivel de luz ambiental.
- **Descripción:** Un sensor digital que mide la intensidad de la luz, la cual se aplica para ajustar la iluminación artificial interior y cerrar y abrir las cortinas de la ventana en caso de necesitar claridad basada en la cantidad de luz detectada por el sensor.

Sensor De Gas MQ-2:

- **Función:** Detección de concentración de gas y humo.

- **Descripción:** Un sensor de gas digital que detecta gas y humo que se acciona por el sensor para que suene la alarma y notifique al usuario en caso de humo o gas, cual, asimismo, mejorará la seguridad doméstica.

Figura 3.2: Diagrama de flujo del funcionamiento del equipo propuesta de casa inteligente con Raspberry pi 5.



Elaborada por: Autor

Sensor de Movimiento PIR HC-SR501

- **Función:** Detección de movimientos.
- **Descripción:** un sensor infrarrojo pasivo para la detección del movimiento de las personas u objetos detectados en su rango. Su

función es, activar la alarma y notificar al usuario en caso de haber detectado un movimiento violento.

- **Sensor de Humedad y Temperatura Dht22**

- **Función:** Mide la temperatura y la humedad ambiental.
- **Descripción:** Se utiliza para controlar la ventilación en función de la temperatura ambiente, activando un ventilador cuando la temperatura supera un umbral específico.

Driver L298N

- **Función:** Controlar motores.
- **Descripción:** Gestiona la dirección y velocidad de los motores de corriente continua. Maneja los motores que abren y cierran las cortinas según la lectura del sensor de luz.

Relés:

- **Función:** Interruptor electromecánico.
- **Descripción:** Se utiliza para controlar el encendido y apagado de dispositivos de consumo elevado, como las luces, las alarmas y los ventiladores.

Conexión GPIO:

- **Función:** Sirve como una interfaz entre la Raspberry Pi 5 y los sensores y actuadores.
- **Descripción:** Son pines de propósito general que se configuran para recabar datos de sensores y mandar señales de control a los actuadores.

I2C Conexión:

- **Función:** Un protocolo de comunicación que se conecta la Raspberry Pi a sensores digitales como el TSL2561.
- **Descripción:** Es un protocolo en serie que permite la conectividad de múltiples dispositivos con dos cables SDA y SCL; es apropiado para sensores digitales.

Conexión a la red y envío de correos electrónicos:

- **Función:** Envía al usuario las alertas por correo electrónico.
- **Descripción:** Utiliza la conectividad de red de la Raspberry Pi para enviar correos a través del servidor SMTP en perspectiva de eventos críticos como la detección de humo o cuando el sensor detecte movimiento.

3.1.1 Funciones y modelos de los componentes utilizados en el proyecto

Los elementos utilizados para la implementación del sistema de casa inteligente con Raspberry Pi 5 se enumeran en la Tabla 3.1. Cada entrada de la tabla contiene la información sobre el elemento apropiado, incluida la instancia específica, la función que se utiliza en el sistema, si es digital o analógico. Cada elemento se utiliza para automatizar y mejorar la funcionalidad del hogar inteligente. Todos los elementos se seleccionan y se utilizan de tal forma que permiten la integración óptica y el funcionamiento digital dentro del marco de la automatización modificada del hogar, lo que permite al beneficio del usuario final en una experiencia mejorada.

Tabla 3.1: Funciones y nombres de los elementos utilizados para la propuesta de diseño de la casa inteligente.

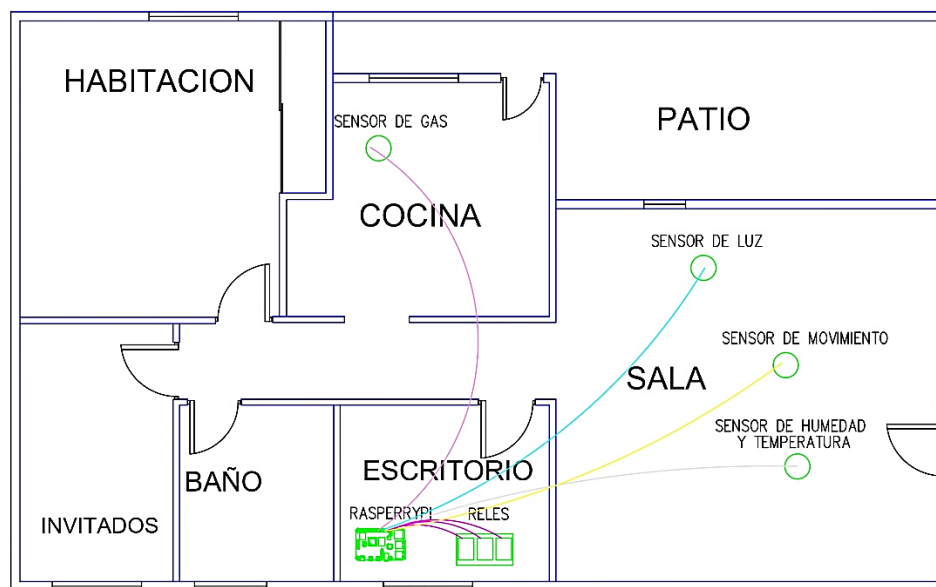
Elemento	Modelo	Función	Tipo
Raspberry Pi	Raspberry Pi 5	Placa base para control y procesamiento	Computadora
Sensor de luz	TSL2561	Detecta niveles de luz para control de iluminación	Digital
Sensor de gas	MQ-2	Detecta concentraciones de gas (humo)	Digital
Sensor de movimiento	PIR HC-SR501	Detecta movimiento para activar sistemas de seguridad	Digital
Sensor de temperatura y humedad	DHT22	Mide temperatura y humedad ambiente	Digital
Driver para cortinas	L298N	Controla motores para abrir y cerrar cortinas	Controlador
Relé para luz	1 canal	Controla el encendido/apagado de luces	Actuador
Relé para gas	1 canal	Controla la activación de sistemas de alarma y rociador de agua.	Actuador
Relé para alarma	1 canal	Activa/desactiva la alarma ante detección de movimiento o gas	Actuador
Relé para ventilación	1 canal	Enciende/apaga el ventilador ante altas temperaturas	Actuador

Fuente: Autor

3.2 Gestión de la ubicación de los elementos en el sistema propuesto

En la figura 3.3, se observa la ubicación propuesta de los elementos en la casa inteligente. La imagen muestra cómo se instalan los diferentes elementos tecnológicos en la vivienda para facilitar su control e integración con el entorno y otros dispositivos disponibles. Cada elemento se coloca en su posición estructurada para mejorar su rendimiento y cooperación en el sistema de gestión de la casa inteligente.

Figura 3.3: Plano de ubicación de los elementos propuestos en AutoCAD.



Elaborada por: Autor

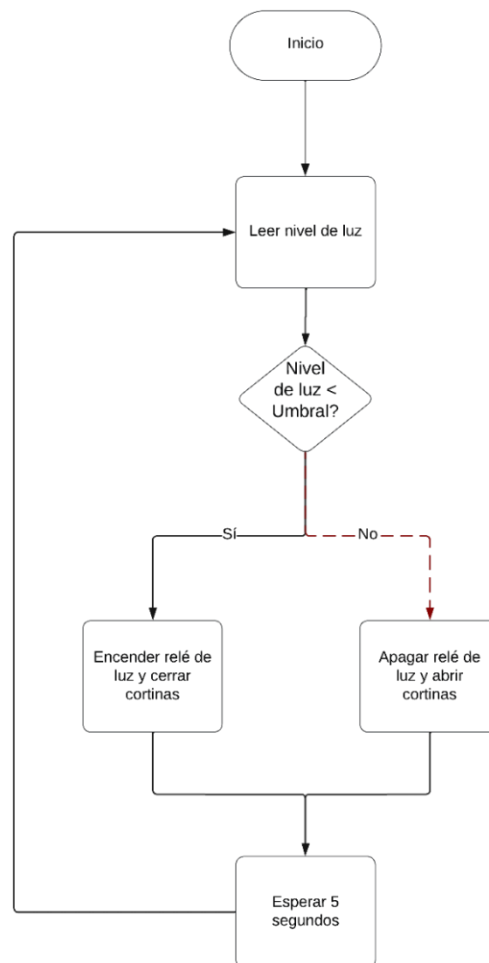
3.2.1 Descripción de la integración y comunicación de los sensores con los actuadores en la casa inteligente

3.2.1.1 Diagrama operacional del sensor de luz

El diagrama de flujo para el sensor de luz TSL2561 de la casa inteligente como se observa en la figura 3.4 empieza luego de que se inicializa el sistema y los componentes utilizados. La Raspberry Pi lee el valor del nivel de luz, en lux, desde el sensor y evalúa si el valor de lux es menor o igual a

un umbral. Si la afirmación es verdadera, la luz del relé grande se enciende para encender la luz y el motor L298N se cierra cerrando las cortinas; después de lo cual se apaga la luz del relé grande y el motor L298N se activa para abrir las cortinas. Después de cada uno de estos pasos, el sistema espera 5 segundos para reiniciar el Sistema y repetir todo el proceso nuevamente.

Figura 3.4: Diagrama de flujo del funcionamiento del sensor de luz.



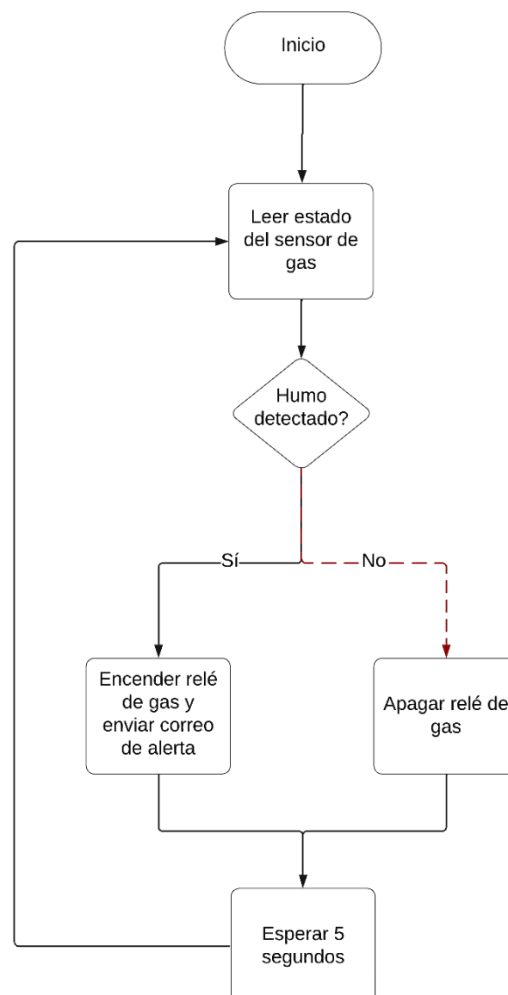
Elaborada por: Autor

3.2.1.2 Diagrama operacional del sensor de gas

El flujo del diagrama de flujo para el sensor de gas MQ-2 como se observa en la figura 3.5 comienza con la inicialización del sistema y todos los

componentes relacionados con el sensor. Luego, la Raspberry Pi lee el valor de humo del sensor para detectar si hay humo o no. Si el sensor detecta humo, dos acciones se activan: encender el relé en la que activa el dispositivo de seguridad, como alarma o ventilador, y enviar un correo electrónico al dueño de la casa con el tiempo y la fecha detectar humo. Si no se detecta humo, se apagará el relé de gas. Luego, el sistema esperará 5 segundos y continuará repitiendo el proceso nuevamente.

Figura 3.5: Diagrama de flujo del funcionamiento del sensor de gas.

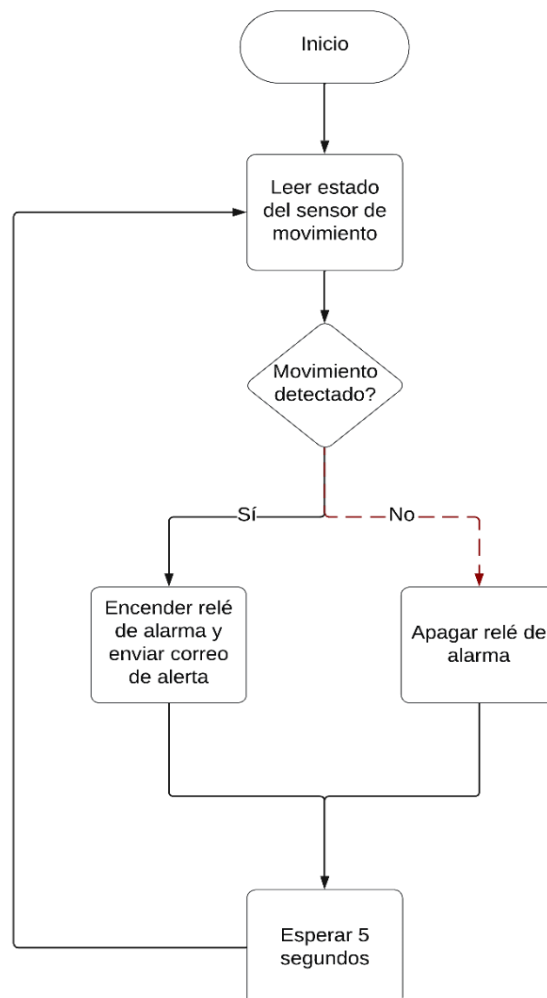


Elaborada por: Autor

3.2.1.3 Diagrama operacional del sensor de movimiento

El diagrama de flujo para el sensor de movimiento PIR HC-SR501 como se observa en la figura 3.6 comienza con la inicialización del sistema y de sus componentes. La Raspberry Pi lee el estado de funcionamiento. En caso de detectar movimiento, se activan acciones. La primera enciende el relé que prende la alarma de la casa. La segunda, envía un correo electrónico al dueño de la casa para notificarle el momento en que se le prendió la alarma. En caso de no detectarse movimiento, se apaga el relé. El sistema espera 5 segundos y repite el proceso.

Figura 3.6: Diagrama de flujo del funcionamiento del sensor de movimiento.

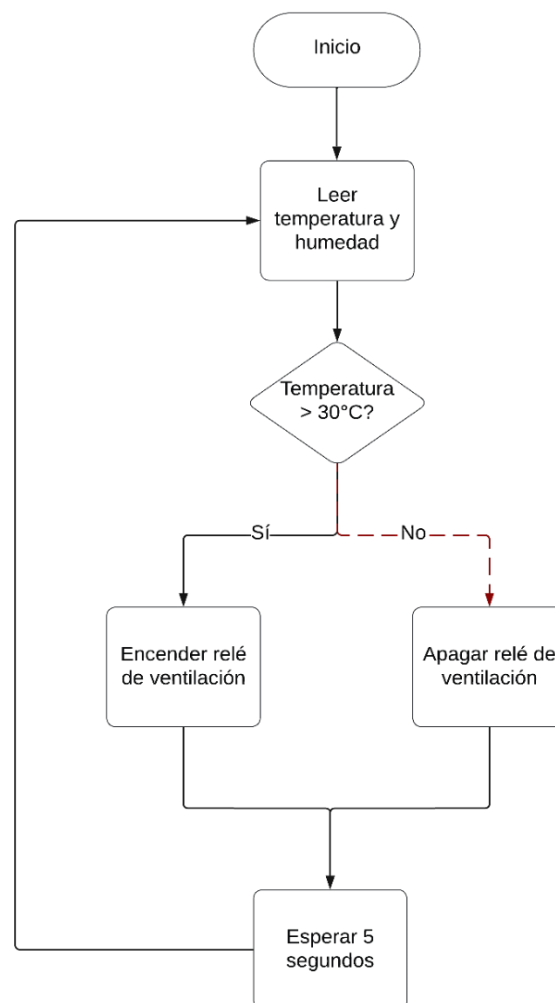


Elaborada por: Autor

3.2.1.4 Diagrama operacional del sensor de temperatura y humedad

El diagrama de flujo del sensor de temperatura DHT22 y humedad como se observa en la figura 3.7 comienza con la inicialización del sistema y los componentes del sistema. El sistema lee los valores de la temperatura y la humedad desde el sensor DHT22 a través de la Raspberry Pi. Se evalúa si la temperatura leída es mayor a 30 grados, si la aseveración es verdadera, se activa el relé del ventilador y si la aseveración es falsa, el relé se apaga. El sistema espera 5 segundos y repite el proceso.

Figura 3.7: Diagrama de flujo del funcionamiento del sensor de temperatura y humedad relativa.

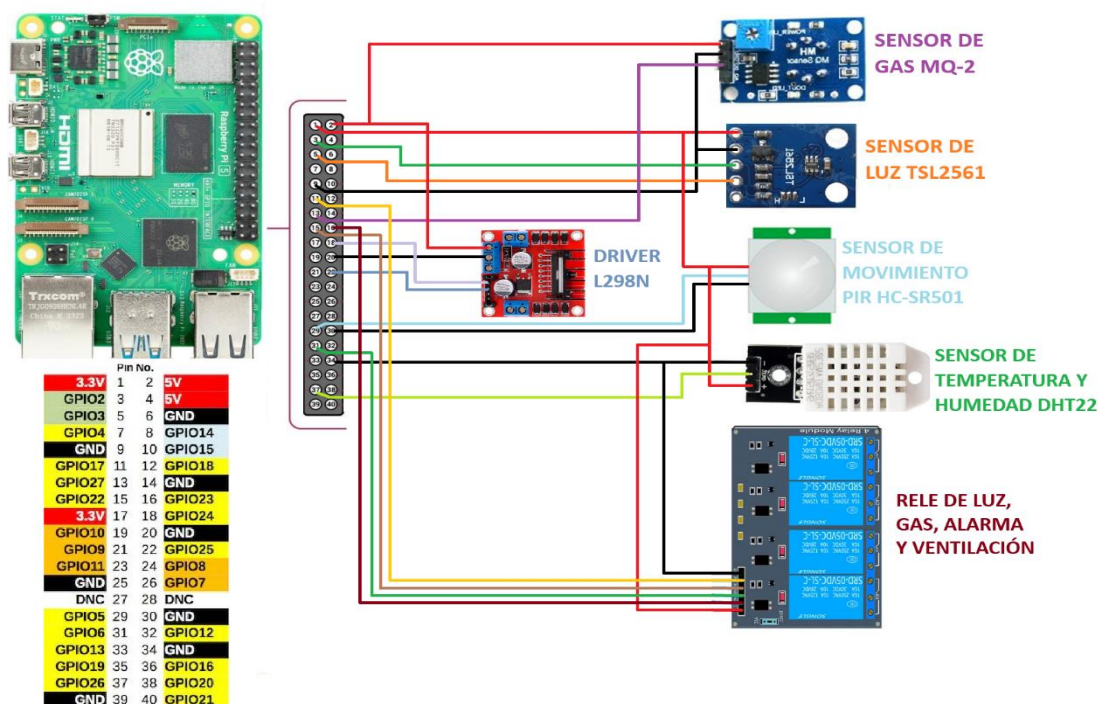


Elaborada por: Autor

3.3 Conexiones empleados en cada uno de los elementos del sistema de casa inteligente

A continuación se muestran, los voltajes de operación, las conexiones de los numerosos componentes y módulos que se emplean para el presente proyecto como se observa en la figura 3.8. Estos elementos son cruciales para garantizar la operación segura y eficaz del sistema de la casa inteligente con Raspberry Pi 5.

Figura 3.8: Diagrama de conexiones de la Raspberry pi 5 con los sensores y actuadores para la casa inteligente.



Elaborada por: Autor

1. Raspberry Pi 5

- Alimentación: 5V, 2,5A.

2. Sensor de Luz TSL2561

- VCC: 3.3V (pin 1 en la Raspberry Pi)

- **GND:** Negativo (pin 9 en la Raspberry Pi)
- **SDA:** GPIO 2 (pin 3 en la Raspberry Pi)
- **SCL:** GPIO 3 (pin 5 en la Raspberry Pi)

3. Sensor de Gas MQ-2

- **VCC:** 5V (pin 2 o 4 en la Raspberry Pi)
- **GND:** Negativo (pin 6 en la Raspberry Pi)
- **DO (Salida Digital):** GPIO 27 (pin 13 en la Raspberry Pi)

4. Sensor de Movimiento PIR HC-SR501

- **VCC:** 5V (pin 2 o 4 en la Raspberry Pi)
- **GND:** Negativo (pin 6 en la Raspberry Pi)
- **OUT (Salida):** GPIO 5 (pin 29 en la Raspberry Pi)

5. Sensor de Temperatura y Humedad DHT22

- **VCC:** 3.3V o 5V (pin 1 o 2 en la Raspberry Pi)
- **GND:** Negativo (pin 6 en la Raspberry Pi)
- **DATA:** Negativo 26 (pin 37 en la Raspberry Pi)

6. Driver L298N (para control de cortinas)

- **IN1:** GPIO 24 (pin 18 en la Raspberry Pi)
- **IN2:** GPIO 25 (pin 22 en la Raspberry Pi)
- **VCC:** 5V (pin 2 o 4 en la Raspberry Pi)
- **GND:** Negativo (pin 6 en la Raspberry Pi)

7. Relé para Luz

- **VCC:** 5V (pin 2 o 4 en la Raspberry Pi)
- **GND:** Negativo (pin 6 en la Raspberry Pi)
- **IN (Control):** GPIO 17 (pin 11 en la Raspberry Pi)

8. Relé para Alarma de Humo

- **VCC:** 5V (pin 2 o 4 en la Raspberry Pi)
- **GND:** Negativo (pin 6 en la Raspberry Pi)
- **IN (Control):** GPIO 22 (pin 15 en la Raspberry Pi)

9. Relé para Alarma de Movimiento

- **VCC:** 5V (pin 2 o 4 en la Raspberry Pi)
- **GND:** Negativo (pin 6 en la Raspberry Pi)
- **IN (Control):** GPIO 6 (pin 31 en la Raspberry Pi)

10. Relé para Ventilación

- **VCC:** 5V (pin 2 o 4 en la Raspberry Pi)
- **GND:** Negativo (pin 6 en la Raspberry Pi)
- **IN (Control):** GPIO 23 (pin 16 en la Raspberry Pi)

3.3.1 Descripción de los voltajes y pines asignados de los elementos conectados a la Raspberry pi 5

En la tabla 3.2 se aprecia los elementos utilizados para la propuesta de diseño, en los cuales se detalla los voltajes y corrientes que necesitan cada uno de ellos, para su respectivo funcionamiento.

Tabla 3.2: Voltajes y corrientes de operación de los elementos del proyecto.

Elemento	Voltaje de funcionamiento	Corriente de funcionamiento
Raspberry Pi 5	5V (alimentación)	~2.5A máx.
Sensor de luz TSL2561	3.3V / 5V	0.5 mA (en reposo), 0.95 mA (en detección)
Sensor de gas MQ-2	5V	150 mA
Sensor de movimiento PIR HC-SR501	5V	50 μ A (en reposo), 15 mA (en detección)
Sensor DHT22	3.3V / 5V	1-2.5 mA (en funcionamiento)
Driver L298N	5V	Depende del motor
Relé	5V	70-100 mA (activación)

Fuente: Autor

En la tabla 3.3 se visualiza la descripción del tipo y número de pin de la raspberry pi 5, que van a utilizar cada uno de los sensores y actuadores para su respectiva conexión y funcionamiento.

Tabla 3.3: Tipos de pines utilizados en la Raspberry pi 5.

Elemento	Tipo	Pin (Raspberry Pi 5)
Sensor de luz TSL2561	I2C	GPIO 2 (SDA) , GPIO 3 (SCL)
Sensor de gas MQ-2	Digital	GPIO 27
Sensor de movimiento PIR HC-SR501	Digital	GPIO 5
Sensor de temperatura y humedad DHT22	Digital	GPIO 26
Driver L298N	Controlador Motor	GPIO 24 y GPIO 25
Relé para luz	Actuador	GPIO 17
Relé para gas	Actuador	GPIO 22
Relé para alarma	Actuador	GPIO 6
Relé para ventilación	Actuador	GPIO 23

Fuente: Autor

3.4 Programación de la Raspberry Pi 5 en Python 3

En la figura 3.9 se observa las bibliotecas esenciales para controlar y leer sensores y actuadores; se utiliza `time` para manejar el tiempo para pausas y tiempos de retraso, `board` y `busio` son para la configuración de los pines y la comunicación I2C, respectivamente, sin los cuales el sensor de luz TSL2561 no funcionaría, `digitalio` se requiere para el relé para controlar los pines digitales, `smtplib` es para enviar correos electrónicos y `email.mime.text` es para dar formato al contenido del correo.

`adafruit_tsl2561.TSL2561` se utiliza para interactuar con el sensor de luz TSL2561. `RPi.GPIO` se utiliza para controlar los pines GPIO y conectar sensores y actuadores a la pi. `Adafruit_DHT` es para leer del sensor de temperatura y humedad DHT22. Con estas importaciones, se crea un ambiente para la supervisión, control y notificación basada en eventos para tus casos de uso de automatización Raspberry Pi.

Figura 3.9: Librerías utilizadas para la programación de los sensores y actuadores utilizados en el entorno de Arduino IDE.

```
import time
import board
import busio
import digitalio
import smtplib
from email.mime.text import MIMEText
from adafruit_tsl2561 import TSL2561
import RPi.GPIO as GPIO
import Adafruit_DHT
```

Elaborada por: Autor

En la figura 3.10 se observa la configuración de dos relés para dos sensores distintos. Primero, `rele_luz_pin` se declara el número de pin GPIO

al que está conectado el relé de el sensor de luz, 17. Utilizando “digitalio.DigitalInOut(led)” se genera un objeto “rele_luz” que representa dicho pin, su dirección se establece como salida. Luego, se define “rele_gas_pin” como el pin GPIO del sensor de gas, 22, y se configura con “GPIO.setmode(GPIO.BCM)” el modo de numeración BCM de los pines del GPIO a utilizar en la Raspberry Pi.

A continuación, “GPIO.setup(rele_gas_pin, GPIO.OUT)” declara este pin como salida para ser conectado al relé que activará un dispositivo de seguridad tras la detección de gas. Con estas configuraciones, las Raspberry Pi pueden controlar los relés condicionados a lo que los sensores identifiquen que los autoriza a hacer, facilitando la automatización de eventos como encendido de luces o seguridad en base a está identificación.

Figura 3.10: Programación de relés de control de luz y gas.

```
# Configuración del relé para el sensor de luz
rele_luz_pin = 17
rele_luz = digitalio.DigitalInOut(board.D17)
rele_luz.direction = digitalio.Direction.OUTPUT

# Configuración del relé para el sensor de gas
rele_gas_pin = 22
GPIO.setmode(GPIO.BCM)
GPIO.setup(rele_gas_pin, GPIO.OUT)
```

Elaborada por: Autor

En la figura 3.11 se observa la configuración de múltiples pines GPIO que se utilizan para controlar varios actuadores en el sistema de domótica casero. Primero, se establece el “rele_alarma_pin” como un pin de salida GPIO 6 que controlará el relé utilizado para activar una alarma en caso de que

el sensor correspondiente detecte movimiento. El “rele_ventilacion_pin” se define a continuación, que es el pin GPIO 23 que activará / desactivará el relé para encender y apagar el ventilador de ventilación, según la temperatura registrada por el sensor relevante.

Por último, se establecen “pin_motor1_a” y “pin_motor1_b”, los pines GPIO 24 y 25 que, respectivamente, se usan para controlar el motor a través del L298N, que controla el mecanismo de las cortinas. Con estas configuraciones, la Raspberry Pi puede automatizar claramente numerosas funciones en el hogar, que incluyen seguridad contra el movimiento, regulación del clima y control del ambiente luz natural.

Figura 3.11: Programación de relés de control de movimiento, ventilación y ajustes del controlador del motor.

```
# Configuración del relé para el sensor de movimiento
rele_alarma_pin = 6
GPIO.setup(rele_alarma_pin, GPIO.OUT)

# Configuración del relé para el ventilador de ventilación
rele_ventilacion_pin = 23
GPIO.setup(rele_ventilacion_pin, GPIO.OUT)

# Configuración del driver L298N para control de cortinas
pin_motor1_a = 24
pin_motor1_b = 25
GPIO.setup(pin_motor1_a, GPIO.OUT)
GPIO.setup(pin_motor1_b, GPIO.OUT)
```

Elaborada por: Autor

En la figura 3.12 se observa la creación de la comunicación i2c y el sensor de luz TSL2561. El código conceptual de la línea “i2c = busio.I2C (board.SCL, board.SDA) ” proporciona un bus i2c inicializado utilizando los pines SCL (reloj) y SDA (datos) de la Raspberry Pi. Esto es esencial para la comunicación con el sensor de luz. Además, “sensor_luz = TSL2561(i2c) ” da

como resultado la creación de un objeto para el sensor TSL2561, permitiendo la lectura de los niveles de luz. Además, “umbral_luz = 50” define la luminosidad; un ambiente mayor de 50 es claro y menor es oscuro.

Finalmente, se genera gas = 27 y el pin de GPIO como entrada para el sensor mq2. La Raspberry Pi puede leer si el atmosférico tiene gas o humo con el siguiente código “GPIO.setup(mq2, GPIO.IN). Entrenan a la Raspberry Pi para monitorear la luminosidad y leer la presencia de gas y actuar según las definiciones del umbral.

Figura 3.12: Programación de umbral de detección del sensor de luz y del sensor de gas.

```
# Configuración del bus I2C y del sensor TSL2561
i2c = busio.I2C(board.SCL, board.SDA)
sensor_luz = TSL2561(i2c)

# Umbral para detectar si está claro u oscuro
umbral_luz = 50

# Configuración del sensor de gas MQ2
mq2_pin = 27
GPIO.setup(mq2_pin, GPIO.IN)
```

Elaborada por: Autor

En la figura 3.13 se observa la configuración de dos sensores y la definición de una función para leer los datos de un sensor. Primero, “pir_pin = 5” configura el pin GPIO como 5 una entrada para permitir que el sensor de movimiento PIR HC-SR501 detecte movimiento en una Raspberry Pi. Posteriormente, “dht22_pin = 26” informa al pin GPIO como 26 para el DHT22 para leer la temperatura y la humedad.

Finalmente, se define una función llamada “leer_luz()”, que, cuando se llama, devuelve la luminosidad medida por el sensor de luz TSL2561 leyendo a través de “sensor_luz.lux”. Primero, esto permite que la Raspberry Pi monitoree las condiciones del sensor de luz, como el movimiento, la temperatura, la humedad y la luz, y este último le permite al sistema en sí tomar decisiones basadas en estas lecturas.

Figura 3.13: Programación del sensor de movimiento, de temperatura y humedad relativa.

```
# Configuración del sensor de movimiento PIR HC-SR501
pir_pin = 5
GPIO.setup(pir_pin, GPIO.IN)

# Configuración del sensor de temperatura y humedad DHT22
dht22_pin = 26

def leer_luz():
    return sensor_luz.lux
```

Elaborada por: Autor

En la figura 3.14 se observa la configuración de tres funciones que se utilizan para controlar el estado de las cortinas o el relé de luz a través de Raspberry Pi. La función “abrir_cortinas()” abre las cortinas activando el motor cuyos pines son altos y bajos. El, “pin_motor1_a” es alto y “pin_motor1_b” es bajo. La función “cerrar_cortinas” hace exactamente lo opuesto abriendo el “pin_motor1_a que es bajo y cerrando el “pin_motor1_b que es alto.

La función toma un parámetro, estado, que apaga y enciende el relé de luz. Si estado es True, el relé LDR se enciende “rele_luz.value = True”, y si estado es False, el relé se apaga. Con estas funciones, se automatiza el

proceso del control de las cortinas y las luces basándonos en los datos del sensor, lo que mejora la eficiencia del sistema domótico.

Figura 3.14: Programación de apertura, cierre de cortinas del motor y control del relé de luz.

```
def abrir_cortinas():
    GPIO.output(pin_motor1_a, GPIO.HIGH)
    GPIO.output(pin_motor1_b, GPIO.LOW)

def cerrar_cortinas():
    GPIO.output(pin_motor1_a, GPIO.LOW)
    GPIO.output(pin_motor1_b, GPIO.HIGH)

def controlar_rele_luz(estado):
    if estado:
        rele_luz.value = True # Encender relé
    else:
        rele_luz.value = False # Apagar relé
```

Elaborada por: Autor

En la figura 3.15 se observa la configuración de tres funciones que controlarán los relés asociados a tres sistemas de la vivienda inteligente usando la Raspberry Pi. La función “controlar_rele_gas establece el relé del sensor de gas si el valor booleano de estado es principalmente True o False. Lo mismo resulta para la función “controlar_rele_alarma que establece el relé de alarma de movimiento en específico para el valor de estado si es True o False. Por otro lado, la función “controlar_rele_ventilacion regula el relé del sistema de ventilación si el valor es True o simplemente False.

Figura 3.15: Programación del relé de gas, ventilación y alarma.

```
def controlar_rele_gas(estado):
    if estado:
        GPIO.output(rele_gas_pin, GPIO.HIGH) # Encender relé
    else:
        GPIO.output(rele_gas_pin, GPIO.LOW) # Apagar relé

def controlar_rele_alarma(estado):
    if estado:
        GPIO.output(rele_alarma_pin, GPIO.HIGH) # Encender relé
    else:
        GPIO.output(rele_alarma_pin, GPIO.LOW) # Apagar relé

def controlar_rele_ventilacion(estado):
    if estado:
        GPIO.output(rele_ventilacion_pin, GPIO.HIGH) # Encender relé
    else:
        GPIO.output(rele_ventilacion_pin, GPIO.LOW) # Apagar relé
```

Elaborada por: Autor

En la figura 3.16 se observa la función “enviar_correo_alerta(asunto, cuerpo)”, que se encarga de enviar un correo electrónico de alerta. Primero, construye un objeto encabezado de correo y se escriben en él el remitente y el destinatario del correo. Posteriormente, el mensaje de correo a enviar, construido a partir del “cuerpo y el “asunto proporcionados se crea mediante “MIMEText”. Por último, se establece el remitente y el destinatario mediante “message['From']” y “message['To']” respectivamente.

Figura 3.16: Programación de estructura de envío de alertas por correo.

```
def enviar_correo_alerta(asunto, cuerpo):
    remitente = "tucorreo@gmail.com"
    destinatario = "destinatario@gmail.com"

    msg = MIMEText(cuerpo)
    msg['Subject'] = asunto
    msg['From'] = remitente
    msg['To'] = destinatario

    try:
        server = smtplib.SMTP('smtp.gmail.com', 587)
        server.starttls()
        server.login(remitente, "tu_contraseña_de_correo")
        server.sendmail(remitente, destinatario, msg.as_string())
        server.quit()
        print("Correo de alerta enviado.")
    except Exception as e:
        print(f"Error al enviar el correo: {e}")
```

Elaborada por: Autor

Cerrando el bloque try, El segundo paso es configurar la conexión SMTP con el servidor Gmail en el puerto 587, iniciar una conexión SSL segura con “starttls()”, autenticar utilizando el correo y la contraseña del remitente y enviar el correo utilizando “sendmail()”. Por último, se cierra la conexión con “quit()” e imprime un mensaje de confirmación. En el caso de un error, una excepción es capturada y se imprime un mensaje de error.

“enviar_correo_alerta” permite enviar alertas automáticas por correo en respuesta a eventos de dada por los sensores del sistema domótico.

En la figura 3.17 se observa el bloque try, e invoca el while True, lo que significa que el script ejecutará una y otra vez. Dentro del bucle, el indicador de función “leer_luz()” se activa para recopilar el valor de luz en lux, que imprime en la pantalla. Si el umbral de luz es mayor que el llamante léxico, imprime un mensaje diciendo que es oscuro y llama al método “cerrar_cortinas()”, que activa el indicador y activa el dispositivo, cerrando las persianas.

Figura 3.17: Programación de mensajes del sensor de luz.

```
try:
    while True:
        # Sensor de luz
        luz = leer_luz()
        print(f"Nivel de luz: {luz} lux")

        if luz < umbral_luz:
            print("Está oscuro, cerrando cortinas.")
            cerrar_cortinas()
        else:
            print("Está claro, abriendo cortinas.")
            abrir_cortinas()
```

Elaborada por: Autor

Si el umbral de luz es igual o mayor que el umbral de luz imprime un mensaje indicando que es claro, y el llamante léxico invoca la función “abrir_cortinas()”, que abre las persianas. El propósito final de este script es ajustar las persianas de la casa automáticamente según el umbral de nivel de luz. Ofrece un producto eficiente energéticamente y optimiza el confort.

En la figura 3.18 se observa el código para el monitoreo del sensor de gas MQ-2 para determinar si está presente humo. La variable “humo_detectado” recibe una nueva asignación leyendo el estado del pin mq2_pin configurado para el sensor de gas mediante “GPIO.input(mq2_pin)”, un not se aplica después para invertir el valor debido a que el sensor devuelva un valor bajo cuando se detecta humo. Si “humo_detectado” es diferente de cero, entonces, se emite un mensaje diciendo que se ha detectado humo, se activa el relé del sistema de gas a través del método “controlar_rele_gas(True)” y se envía un correo electrónico mediante “enviar_correo_alerta()” que incluye la fecha y hora actuales formateadas con “time.strftime(“%Y-%m-%d %H:%M:%S”)”.

Figura 3.18: Programación de mensajes del sensor de gas.

```
# Sensor de gas
humo_detectado = not GPIO.input(mq2_pin)

if humo_detectado:
    print("Humo detectado, encendiendo alarma y enviando correo.")
    controlar_rele_gas(True)
    enviar_correo_alerta(
        "Alerta de humo detectado",
        f"Humo detectado en casa a las {time.strftime('%Y-%m-%d %H:%M:%S')}."
    )
else:
    controlar_rele_gas(False)
```

Elaborada por: Autor

Si no hay humo, se apaga el relé del sistema de gas a través de “controlar_rele_gas(False)”. Una vez más, el código es independiente de cualquier sistema operativo y puede aceptar métodos y funciones con argumentos o de retorno requeridos sin cambios.

En la figura 3.19 se observa el código para el sensor del movimiento PIR HC-SR501. La variable “movimiento_detectado” se actualiza observando el estado del pin, configurado para el sensor de movimiento, a través de “GPIO.input()”. Si “movimiento_detectado” es verdadero, emitimos un mensaje de movimiento detectado, activamos el relé de la alarma a través de “controlar_rele_alarma(True)” y enviamos un correo electrónico con una alerta a través de “enviar_correo_alerta()” con un mensaje que incluye la fecha y hora actuales, formateada con “time.strftime('%Y-%m-%d %H:%M:%S')”. Si no se detecta movimiento, apagamos el relé de la alarma con “controlar_rele_alarma(False)”.

Figura 3.19: Programación de mensajes del sensor de movimiento.

```
# Sensor de movimiento
movimiento_detectado = GPIO.input(pir_pin)

if movimiento_detectado:
    print("Movimiento detectado, encendiendo alarma y enviando correo.")
    controlar_rele_alarma(True)
    enviar_correo_alerta(
        "Alerta de movimiento detectado",
        f"Movimiento detectado en casa a las {time.strftime('%Y-%m-%d %H:%M:%S')}."
    )
else:
    controlar_rele_alarma(False)
```

Elaborada por: Autor

En la figura 3.20 se observa el código que monitorea el sensor de temperatura y humedad DHT22. Primero, utiliza la función “Adafruit_DHT.read_retry()” que lee los valores de humedad y temperatura del sensor conectado a través del pin “dht22_pin”. Luego, si se obtiene un valor válido de la temperatura más que 30°C, se imprime un mensaje de temperatura alta y se activa un relé de ventilación llamando a “controlar_rele_ventilacion(True)”. Si la temperatura no supera los 30°C o la

lectura es inválida, entonces se desactiva el relé de ventilación con la ayuda de “controlar_rele_ventilacion(False)”. Además, el código también duerme cinco segundos antes de leer la próxima vez con “time.sleep(5)”.

Figura 3.22: Programación de mensajes del sensor de temperatura y humedad relativa.

```
# Sensor de temperatura y humedad
humedad, temperatura = Adafruit_DHT.read_retry(Adafruit_DHT.DHT22, dht22_pin)

if temperatura is not None and temperatura > 30:
    print(f"Temperatura alta ({temperatura}°C), encendiendo ventilación.")
    controlar_rele_ventilacion(True)
else:
    controlar_rele_ventilacion(False)

time.sleep(5) # Espera 5 segundos antes de la siguiente lectura
```

Elaborada por: Autor

En la figura 3.21 se observa el bloque de código que maneja la excepción “KeyboardInterrupt”. Esta se ejecuta cuando el usuario interrumpe el programa manualmente, por lo general, al presionar “Ctrl-C”. Se imprime una notificación por pantalla de que el programa ha finalizado. A continuación, se desactivan los relés mediante “control_rele_luz” o “rele_control(False)” control_relay_gas(False), “control_rele_alarma(False), “control_rele_vent(False)” y vía “rele_control v=false”. De este modo, se asegura de que las conexiones conectadas se desactiven de forma segura.

Figura 3.21: Programación de estructura de cierre y reset del código.

```
except KeyboardInterrupt:
    print("Programa terminado.")
    controlar_rele_luz(False)
    controlar_rele_gas(False)
    controlar_rele_alarma(False)
    controlar_rele_ventilacion(False)
    GPIO.cleanup()
```

Elaborada por: Autor

Se llama a la librería de “cleanup” general de librería “GPIO.cleanup()”. De este modo se ponen todos los pines GPIO en el estado en el que estaban antes de comenzar el programa, proporcionando un punto de cierre y liberación de posibles pines en uso. Dado que “GPIO.cleanup()” no borra las configuraciones de un pin asignadas mediante “GPIO.setup()” anteriormente, esta función se puede utilizar de manera segura al final de un programa para dejar el sistema en un estado seguro y limpio para futuras transacciones.

3.5 Presupuesto y cronograma general del proyecto propuesto

3.5.1 Listado de costos del sistema de casa inteligente

La Tabla 3.4 se muestra el desglose detallado de los costos de los componentes utilizados en el sistema de domótica inteligente implementado. Esto cubre la versión en formato kit de la Raspberry Pi 5 utilizada, que cuesta 550 USD por unidad; el sensor de luz TSL2561, el sensor de gas MQ-2, el sensor de movimiento PIR HC-SR501, el sensor de temperatura y humedad DHT22, el driver de motor L298N, relés, etc. Todos los elementos se muestran con la cantidad y el precio en USD, dándonos un total de 599 USD.

Tabla 3.4: Costo de los elementos para la propuesta de diseño.

Componente	Modelo/Descripción	Cantidad	Precio Unitario (USD)	Total (USD)
Raspberry Pi	Versión 5 kit	1	550	550
Sensor de luz TSL2561	TSL2561	1	5	5
Sensor de gas	MQ-2	1	3	3

Sensor de movimiento PIR	PIR HC-SR501	1	2	2
Sensor de temperatura y humedad	Sensor DHT22	1	8	8
Driver motor	L298N	1	10	10
Relés	1 canal	4	1.5	6
Otros componentes	Resistencias, cables, etc.	1	5	5
Fuente de Voltaje	3.3./5V a 5A	1	1	20
Total				599

Fuente: Autor

3.5.2 Calendario de planificación de actividades para la instalación del sistema

En el siguiente tabla 3.5 se presenta el cronograma de Instalación del Sistema Propuesto, mostrando mediante semanas el plan de acción para la aplicación de dicho sistema en la vivienda. A continuación, se ofrecerá de forma clara las actividades principales a realizar y la fecha de ejecución de cada una. Este plan de trabajo garantiza un proceso ordenado y eficiente en la instalación del proyecto desde su planificación hasta la entrega final y capacitación al usuario.

Tabla 3.5: Cronograma de estimación de actividades realizadas en 2 semanas para la instalación del sistema propuesto.

Semana	Actividad	Descripción
Semana 1	Planificación del Proyecto	Definición de requisitos y diseño del sistema.
Semana 1	Adquisición de Equipos	Compra de hardware y software necesario.

Semana 2	Preparación del Sitio	Inspección y preparación de áreas para instalación.
Semana 2	Instalación de Sensores	Montaje y configuración de sensores (luz, gas, movimiento, temperatura y humedad).
Semana 2	Instalación de Raspberry Pi y Relés	Configuración del Raspberry Pi 5 y relés en el área de escritorio.
Semana 2	Instalación del Driver para Cortinas	Montaje y configuración del driver para cortinas en la Sala.
Semana 2	Integración y Pruebas	Verificación de la integración y pruebas de funcionamiento.
Semana 2	Formación y Entrega	Capacitación para el usuario y entrega del sistema completo.

Fuente: Autor

Capítulo 4: Conclusiones y recomendaciones

4.1 Conclusiones

- Se estableció un sistema de control y monitoreo adecuado para la gestión de los dispositivos electrónicos mediante el uso de sensores y actuadores, con un control mediante el uso de actuadores, se implementó la Raspberry pi 5 para su uso y control de los dispositivos logrando la conexión a cada uno de ellos. Se ahorro del consumo eléctrico ya que los dispositivos se encienden o apagan dependiendo del entorno donde se encuentre el propietario de la residencia para su confort.
- La utilización de Python 3 como lenguaje de programación permitió un mejor control de cada uno de los elementos del sistema ya que este lenguaje se adapta muy bien a la configuración, permitiendo un significativo mejor desarrollo y una implementación adecuada en su automatización respectiva.
- Se diseñaron diferentes diagramas de flujo de cada uno de los sensores y actuadores utilizados en el proyecto los cuales, que explican el funcionamiento de cada uno y se diseñó el diagrama de conexiones, en el cual se explica cómo se conectaron los diferentes sensores y relés a los pines de la Raspberry pi 5 para tal conectividad.

4.2 Recomendaciones

- Se sugiere diseñar un plan de mantenimiento preventivo a futuro de este sistema, con el fin de conservar y extender la duración de vida útil de cada uno de los elementos electrónicos.
- Se sugiere que, para la creación de un sistema con mayor número de elementos controlados, usar la configuración de esclavo maestro con distintas Raspberry pi 5 o si se desea reducir el costo con el uso de Arduino Uno como esclavo para tener un sistema de mayor tamaño que mantenga un rendimiento correcto.
- Se recomienda impartir la guía de uso y configuración del sistema a la gente de la residencia para una mejor utilización.

Bibliografías

330ohms. (2024, julio 30). *¿Qué es un sensor PIR?* 330ohms.

<https://www.330ohms.com/blogs/blog/que-es-un-sensor-pir>

Acosta, C. (2023). *GoConqr - Actividad 1: Domotica*. GoConqr.

<https://www.goconqr.com/ficha/2011821/actividad-1-domotica>

Ada, Lady. (2020). *Overview | TSL2561 Luminosity Sensor | Adafruit*

Learning System. <https://learn.adafruit.com/tsl2561/overview>

Aurrera. (2024). *Módulo de interruptor de luz inteligente WiFi APP Control*

remoto Control de voz | Bodega Aurrera en línea. Walmart | Tu tienda

en línea. [https://www.bodegaurrera.com.mx/ip/casa-](https://www.bodegaurrera.com.mx/ip/casa-inteligente/automatizacion/asistentes-de-voz/modulo-de-interruptor-de-luz-inteligente-wifi-app-control-remoto-control-de-voz/00077472855877)

[inteligente/automatizacion/asistentes-de-voz/modulo-de-interruptor-](https://www.bodegaurrera.com.mx/ip/casa-inteligente/automatizacion/asistentes-de-voz/modulo-de-interruptor-de-luz-inteligente-wifi-app-control-remoto-control-de-voz/00077472855877)

[de-luz-inteligente-wifi-app-control-remoto-control-de-](https://www.bodegaurrera.com.mx/ip/casa-inteligente/automatizacion/asistentes-de-voz/modulo-de-interruptor-de-luz-inteligente-wifi-app-control-remoto-control-de-voz/00077472855877)

[voz/00077472855877](https://www.bodegaurrera.com.mx/ip/casa-inteligente/automatizacion/asistentes-de-voz/modulo-de-interruptor-de-luz-inteligente-wifi-app-control-remoto-control-de-voz/00077472855877)

Bauduin, S. (2024, marzo 6). *Raspberry Pi 5: 6 great tips, tricks & uses for*

the single-board PC. PCWorld.

[https://www.pcworld.com/article/2250605/raspberry-pi-5-the-6-best-](https://www.pcworld.com/article/2250605/raspberry-pi-5-the-6-best-tips-tricks-applications-for-the-single-board-pc.html)

[tips-tricks-applications-for-the-single-board-pc.html](https://www.pcworld.com/article/2250605/raspberry-pi-5-the-6-best-tips-tricks-applications-for-the-single-board-pc.html)

Belford, R. (2023, marzo 15). *1: DHT22 Temperature and Humidity Sensor*.

Chemistry LibreTexts.

[https://chem.libretexts.org/Courses/University_of_Arkansas_Little_Roc](https://chem.libretexts.org/Courses/University_of_Arkansas_Little_Rock/IOST_Library/09%3A_Sensor_Book/1%3A_Temperature_Sensors/01%3A_DHT22_Temperature_Sensor)

[k/IOST_Library/09%3A_Sensor_Book/1%3A_Temperature_Sensors/0](https://chem.libretexts.org/Courses/University_of_Arkansas_Little_Rock/IOST_Library/09%3A_Sensor_Book/1%3A_Temperature_Sensors/01%3A_DHT22_Temperature_Sensor)

[1%3A_DHT22_Temperature_Sensor](https://chem.libretexts.org/Courses/University_of_Arkansas_Little_Rock/IOST_Library/09%3A_Sensor_Book/1%3A_Temperature_Sensors/01%3A_DHT22_Temperature_Sensor)

Benítez, I. F. (2022). *Diseño e implementación de un sistema de control*

domótico residencial. Latin American Developments in Energy

Engineering, 3(1), Article 1.

<https://doi.org/10.17981/ladee.03.01.2022.3>

Bharadwaja, V., Ananmy, R., Nikhil, S., Vineetha, K. V., Shah, J., & Kurup, D. G. (2018). Implementation of Artificial Neural Network on Raspberry Pi for Signal Processing Applications. *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 1488-1491. 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI).
<https://doi.org/10.1109/ICACCI.2018.8554887>

Castelo, J. C., & Ramos, J. M. P. (2020). Estudio de viabilidad técnica del Raspberry Pi para su uso en sistemas de visión artificial. *E-IDEA 4.0 Revista Multidisciplinar*, 2(4), Article 4.

Ecuarobot. (2022, abril 14). ¿Cómo funciona el sensor de humo y gases inflamables MQ-2 con Arduino? *EcuaRobot*.
<https://ecuarobot.com/2022/04/14/como-funciona-el-sensor-de-humo-y-gases-inflamables-mq-2-con-arduino/>

Farnell. (2022). *RPI5-8GB-SINGLE - SBC, Raspberry Pi5 8GB, BCM2712, Arm Cortex-A76, 8GB RAM, MicroSD, Wifi, HDMI, Power button*.
<https://il.farnell.com/raspberry-pi/rpi5-8gb-single/raspberry-pi-5-model-b-8gb-2-4ghz/dp/4256000>

Fezari, M., & Dahoud, A. (2023). *Raspberry Pi 5: The new Raspberry Pi family with more computation power and AI integration*.
<https://doi.org/10.13140/RG.2.2.13547.52009>

- Ghoul, Y., & Naifar, O. (2024). IoT based applications for healthcare and home automation. *Multimedia Tools and Applications*, 83(10), 29945-29967. <https://doi.org/10.1007/s11042-023-16774-z>
- González, O. (2023, septiembre 28). *Análisis a fondo de la Raspberry Pi 5: La nueva bestia* | BricoGeek.com. Blog BricoGeek.com. <https://blog.bricogeek.com/noticias/raspberry-pi/analisis-a-fondo-de-la-raspberry-pi-5-la-nueva-bestia/>
- Gustama, A., Hadikusuma, R. S., & Hidayat, R. (2024). Utilization of PWM Signals in Automatic Parking Gate Speed Control Using PID System Modeling. *JTEV (Jurnal Teknik Elektro Dan Vokasional)*, 10(1), Article 1. <https://doi.org/10.24036/jtev.v9i2.115893>
- Halfacree, G. (2023). *The Official Raspberry Pi Beginner's Guide: How to use your new computer*. Raspberry Pi Press.
- Kuswanto, D. & Mulaab. (2023). Smart laboratory prototype with the internet of things concept using Raspberry Pi. *AIP Conference Proceedings*, 2706(1), 020084. <https://doi.org/10.1063/5.0120327>
- Lauer, R. (2023, diciembre 7). *Hands-On with Cellular IoT on the Raspberry Pi 5*. Hackster.io. <https://www.hackster.io/rob-lauer/hands-on-with-cellular-iot-on-the-raspberry-pi-5-8c9e44>
- Lean, C. P., Krishnan, G., Li, C., Yuan, K. F., Kiat, N. P., & Khan, M. R. B. (2024). A Raspberry Pi-Powered IoT Smart Farming System for Efficient Water Irrigation and Crop Monitoring. *Malaysian Journal of Science and Advanced Technology*, 149-158. <https://doi.org/10.56532/mjsat.v4i2.295>

- Lokhande, U. (2020). Introduction of Raspberry Pi 3 Model B.
BINARYUPDATES.COM. <https://binaryupdates.com/introduction-of-raspberry-pi-3-model-b/>
- Loor, C. (2020). *Sistema inteligente para el reconocimiento de figuras geométricas basado en Python Raspberry PI* [bachelorThesis].
<https://dspace.uniandes.edu.ec/handle/123456789/13193>
- López, F. E. V. (2023). Análisis de soluciones debido a la generación de información por dispositivos inteligentes de domótica en el internet de las cosas. *Centro Universitario Tecnológico CEUTEC*.
<https://repositorio.unitec.edu/xmlui/handle/123456789/11704>
- Made, S. W. I., Gunawan, A. A. N., Ketut, P. I., & Ketut, S. I. (2024). Design and Manufacture of LPG Gas Leak Detection Based on Arduino Uno Using MQ-2 Sensor. *Asian Journal of Research and Reviews in Physics*, 8(1), 27-32. <https://doi.org/10.9734/ajr2p/2024/v8i1155>
- Mariete. (2023, noviembre 1). *¿Qué es una señal PWM (Pulse Width Modulation o Modulación por Anchura de Pulsos)?* eMariete.
<https://emariete.com/senal-pwm-pulse-width-modulation/>
- Mendoza, E., & Quevedo, A. (2023). *Diseño e Implementación de un banco de pruebas empleando el NodeMCU (ESP8266) para aplicaciones domoticas con IoT y APPLE HOMEKIT* [bachelorThesis].
<http://dspace.ups.edu.ec/handle/123456789/24050>
- Murky. (2020). *I2C: Protocolo de Comunicación Serial*.
<http://www.murkyrobot.com/guias/comunicacion/i2c>
- Mutou. (2022, septiembre 20). *DHT22 pin arrangement: Su principio de funcionamiento y su*. Servicios llave en mano de fabricación de placas

de circuito y montaje de placas de circuito impreso - WellPCB.

<https://placapcb.com/DHT22-pin-arrangement.html>

Palacios, J. L. O. (2020). La protección del usuario en la domótica y las facilidades que ofrece. *Revista Científica Sinapsis*, 1(16), Article 16. <https://doi.org/10.37117/s.v1i16.276>

Papakyriakou, D., & Barbounakis, I. (2023). Benchmarking and Review of Raspberry Pi (RPi) 2B vs RPi 3B vs RPi 3B+ vs RPi 4B (8GB). *International Journal of Computer Applications*, 185, 975-8887. <https://doi.org/10.5120/ijca2023922693>

Pastor, J. (2021, marzo 1). *Si quieres aprender a programar, éstos son los lenguajes más populares y prometedores en 2021*. Xataka. <https://www.xataka.com/aplicaciones/quieres-aprender-a-programar-estos-lenguajes-populares-prometedores-2021>

Pastor, J. (2024, enero 15). *La demanda de las Raspberry Pi 5 era un problema, pero ya hay solución contundente: Fabricarlas a lo bestia*. Xataka. <https://www.xataka.com/ordenadores/demanda-raspberry-pi-5-era-problema-ha-solucion-contundente-fabricarlas-a-bestia>

Quishpe, L. F., & Hernández, K. D. (2023). *Sistema de seguridad autónomo para la detección de fugas de gas GLP en un área residencial* [bachelorThesis]. <http://dspace.ups.edu.ec/handle/123456789/24503>

Raju, L., Sowmya, G., Srividhya, S., Surabhi, S., Retika, M. K., & Reshmika Janani, M. (2021). IoT-Based Home Automation Using Raspberry Pi. En R. Seyezhai, S. Karuppuchamy, & L. Ashok Kumar (Eds.), *Recent Trends in Renewable Energy Sources and Power Conversion* (pp. 155-161). Springer. https://doi.org/10.1007/978-981-16-0669-4_12

Raspberry Pi. (2024). *Raspberry Pi hardware—Raspberry Pi Documentation*.

<https://www.raspberrypi.com/documentation/computers/raspberry-pi.html>

Roth, E. (2023, septiembre 28). *The Raspberry Pi 5 is finally here*. The

Verge. <https://www.theverge.com/2023/9/28/23889238/raspberry-pi-5-specs-availability-pricing>

Santiago, F. F. (2021). *Domótica e inmótica para el ahorro de energía*.

<https://repositorio.unicach.mx/handle/20.500.12753/4154>

Sarachu, E. (2024, marzo 9). *Domótica ⇒ ¿Qué es la domótica? ¿Cómo*

funciona? - 2024. <https://e-ficiencia.com/domotica-que-es-y-como-funciona/>

Shahi, J. (2024). *Detection of Rail Clip with YOLO on Raspberry Pi*.

<https://urn.kb.se/resolve?urn=urn:nbn:se:miun:diva-51680>

Smashing. (2023, octubre 2). *5 Raspberry Pi 5 Programming Languages You*

Should Learn—SmashingApps.com.

<https://www.smashingapps.com/raspberry-pi-5-programming-languages/>

Stolojescu, C., Crisan, C., & Butunoi, B.-P. (2021). An IoT-Based Smart

Home Automation System. *Sensors*, 21(11), Article 11.

<https://doi.org/10.3390/s21113784>

Terce. (2020). *ARD-L298N*. TERCE ECUATORIANA S.A.

<https://www.terce.com.ec/es/inicio/362-ard-l298n.html>

Tiendatec. (2021). *SENSOR MQ-2—DETECTOR GAS COMBUSTIBLE*

AMPLIO RANGO. tiendatec.es. <https://www.tiendatec.es/maker->

zone/modulos/1570-sensor-mq-2-detector-gas-combustible-amplio-rango-8472496020930.html

Triacs. (2021, junio 25). *Controlador de Motor L298N Driver*. Electrónica

Triacs. <https://triacs.cl/modulos/205-controlador-de-motor-l298n-.html>

Velasco. (2022). *SENSOR DE MOVIMIENTO PIR HC-SR501*. VELASCO

STORE. <https://velascostore.com/sensores/5418-sensor-de-movimiento-pir-hc-sr501.html>

Ventre, L. O., Micolini, O., Ludemann, M., Carranza, A., Andrea, D., &

Candotti, E. (2024). *Caso de estudio: Metodología para el diseño y*

desarrollo de sistemas embebidos distribuidos. XXIX Congreso

Argentino de Ciencias de la Computación (CACIC) (Luján, 9 al 12 de

octubre de 2023). <http://sedici.unlp.edu.ar/handle/10915/164820>

Vilros. (2021). *Raspberry Pi 5 Pinout*. Vilros.Com.

<https://vilros.com/pages/raspberry-pi-5-pinout>

Wahid, M. Z., & Octaviano, A. (2023). Rancang Bangun Alat Pendeteksi

Kebocoran Tabung Gas Dengan Menggunakan Sensor Mq-2 Dan

Flame Detector. *LOGIC : Jurnal Ilmu Komputer Dan Pendidikan*, 1(5),

Article 5.

Wikilibros. (2020, octubre 16). *Archivo:RCA to 3.5mm video cable.jpg*.

https://commons.wikimedia.org/wiki/File:RCA_to_3.5mm_video_cable.jpg

Yar, H., Imran, A. S., Khan, Z. A., Sajjad, M., & Kastrati, Z. (2021). Towards

Smart Home Automation Using IoT-Enabled Edge-Computing

Paradigm. *Sensors*, 21(14), Article 14.

<https://doi.org/10.3390/s21144932>

Yida. (2020, abril 20). DHT11 vs DHT22 – Which Temperature and Humidity Sensor Should You Use? *Latest Open Tech From Seeed*.

<https://www.seeedstudio.com/blog/2020/04/20/dht11-vs-dht22-am2302-which-temperature-humidity-sensor-should-you-use/>

Yoigo. (2022, febrero 22). *Cómo ahorrar energía con la domótica: Trucos y consejos*. Blog EnergyGO. <https://blog.energygo.es/como-ahorrar-energia-con-la-domotica-trucos-y-consejos/>

Anexo 1

Código empleado para la programación de la Raspberry pi 5

```
import time

import board

import busio

import digitalio

import smtplib

from email.mime.text import MIMEText

from adafruit_tsl2561 import TSL2561

import RPi.GPIO as GPIO

import Adafruit_DHT


# Configuración del relé para el sensor de luz

relé_luz_pin = 17

relé_luz = digitalio.DigitalInOut(board.D17)

relé_luz.direction = digitalio.Direction.OUTPUT


# Configuración del relé para el sensor de gas

relé_gas_pin = 22

GPIO.setmode(GPIO.BCM)

GPIO.setup(relé_gas_pin, GPIO.OUT)


# Configuración del relé para el sensor de movimiento
```

```
rele_alarma_pin = 6
GPIO.setup(rele_alarma_pin, GPIO.OUT)

# Configuración del relé para el ventilador de ventilación
rele_ventilacion_pin = 23
GPIO.setup(rele_ventilacion_pin, GPIO.OUT)

# Configuración del driver L298N para control de cortinas
pin_motor1_a = 24
pin_motor1_b = 25
GPIO.setup(pin_motor1_a, GPIO.OUT)
GPIO.setup(pin_motor1_b, GPIO.OUT)

# Configuración del bus I2C y del sensor TSL2561
i2c = busio.I2C(board.SCL, board.SDA)
sensor_luz = TSL2561(i2c)

# Umbral para detectar si está claro u oscuro
umbral_luz = 50

# Configuración del sensor de gas MQ2
mq2_pin = 27
GPIO.setup(mq2_pin, GPIO.IN)

# Configuración del sensor de movimiento PIR HC-SR501
```

```
pir_pin = 5

GPIO.setup(pir_pin, GPIO.IN)


# Configuración del sensor de temperatura y humedad DHT22

dht22_pin = 26


def leer_luz():

    return sensor_luz.lux


def abrir_cortinas():

    GPIO.output(pin_motor1_a, GPIO.HIGH)

    GPIO.output(pin_motor1_b, GPIO.LOW)


def cerrar_cortinas():

    GPIO.output(pin_motor1_a, GPIO.LOW)

    GPIO.output(pin_motor1_b, GPIO.HIGH)


def controlar_rele_luz(estado):

    if estado:

        rele_luz.value = True # Encender relé

    else:

        rele_luz.value = False # Apagar relé


def controlar_rele_gas(estado):

    if estado:
```

```

        GPIO.output(rele_gas_pin, GPIO.HIGH) # Encender relé

    else:

        GPIO.output(rele_gas_pin, GPIO.LOW) # Apagar relé


def controlar_rele_alarma(estado):

    if estado:

        GPIO.output(rele_alarma_pin, GPIO.HIGH) # Encender relé

    else:

        GPIO.output(rele_alarma_pin, GPIO.LOW) # Apagar relé


def controlar_rele_ventilacion(estado):

    if estado:

        GPIO.output(rele_ventilacion_pin, GPIO.HIGH) # Encender relé

    else:

        GPIO.output(rele_ventilacion_pin, GPIO.LOW) # Apagar relé


def enviar_correo_alerta(asunto, cuerpo):

    remitente = "tucorreo@gmail.com"

    destinatario = "destinatario@gmail.com"


    msg = MIMEText(cuerpo)

    msg['Subject'] = asunto

    msg['From'] = remitente

    msg['To'] = destinatario

```

```

try:

    server = smtplib.SMTP('smtp.gmail.com', 587)

    server.starttls()

    server.login(remitente, "tu_contraseña_de_correo")

    server.sendmail(remitente, destinatario, msg.as_string())

    server.quit()

    print("Correo de alerta enviado.")

except Exception as e:

    print(f"Error al enviar el correo: {e}")

```

```

try:

    while True:

        # Sensor de luz

        luz = leer_luz()

        print(f"Nivel de luz: {luz} lux")

        if luz < umbral_luz:

            print("Está oscuro, cerrando cortinas.")

            cerrar_cortinas()

        else:

            print("Está claro, abriendo cortinas.")

            abrir_cortinas()

        # Sensor de gas

        humo_detectado = not GPIO.input(mq2_pin)

```

```

if humo_detectado:

    print("Humo detectado, encendiendo alarma y enviando correo.")

    controlar_rele_gas(True)

    enviar_correo_alerta(

        "Alerta de humo detectado",

        f"Humo detectado en casa a las {time.strftime('%Y-%m-%d
%H:%M:%S')}')."

    )

else:

    controlar_rele_gas(False)

# Sensor de movimiento

movimiento_detectado = GPIO.input(pir_pin)

if movimiento_detectado:

    print("Movimiento detectado, encendiendo alarma y enviando
correo.")

    controlar_rele_alarma(True)

    enviar_correo_alerta(

        "Alerta de movimiento detectado",

        f"Movimiento detectado en casa a las {time.strftime('%Y-%m-%d
%H:%M:%S')}')."

    )

else:

```

```

        controlar_rele_alarma(False)

    # Sensor de temperatura y humedad

    humedad, temperatura =
Adafruit_DHT.read_retry(Adafruit_DHT.DHT22, dht22_pin)

    if temperatura is not None and temperatura > 30:

        print(f"Temperatura alta ({temperatura}°C), encendiendo
ventilación.")

        controlar_rele_ventilacion(True)

    else:

        controlar_rele_ventilacion(False)

    time.sleep(5) # Espera 5 segundos antes de la siguiente lectura

except KeyboardInterrupt:

    print("Programa terminado.")

    controlar_rele_luz(False)

    controlar_rele_gas(False)

    controlar_rele_alarma(False)

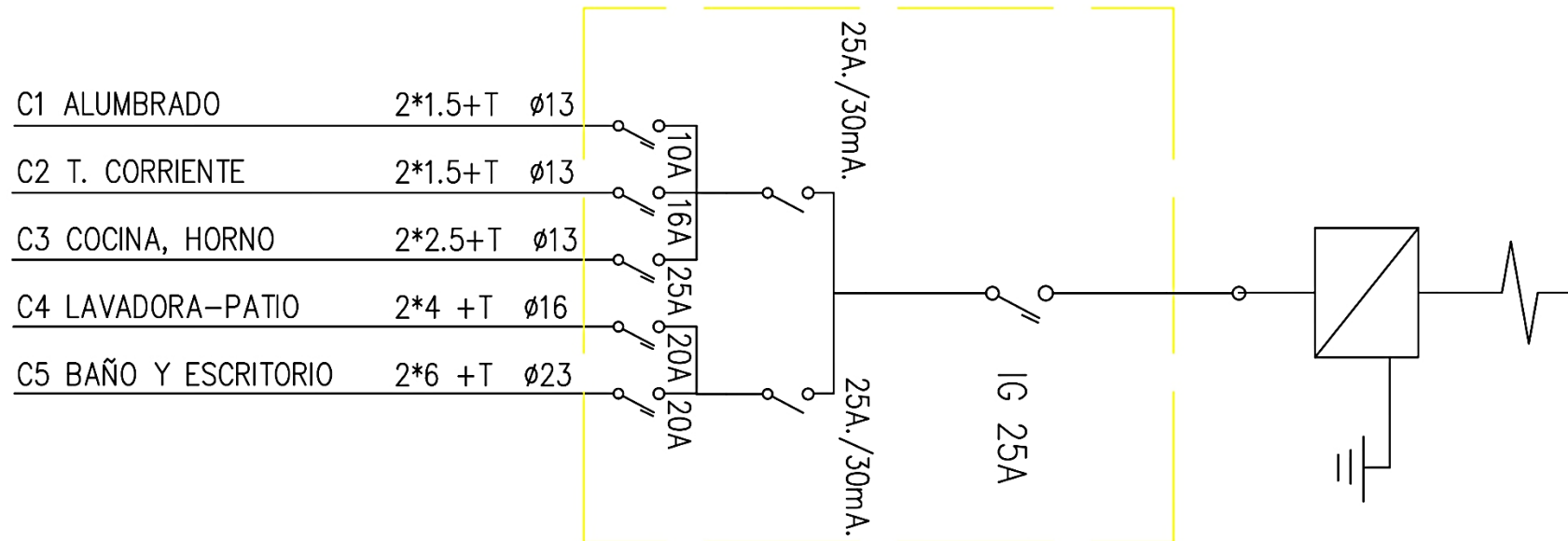
    controlar_rele_ventilacion(False)

    GPIO.cleanup()

```

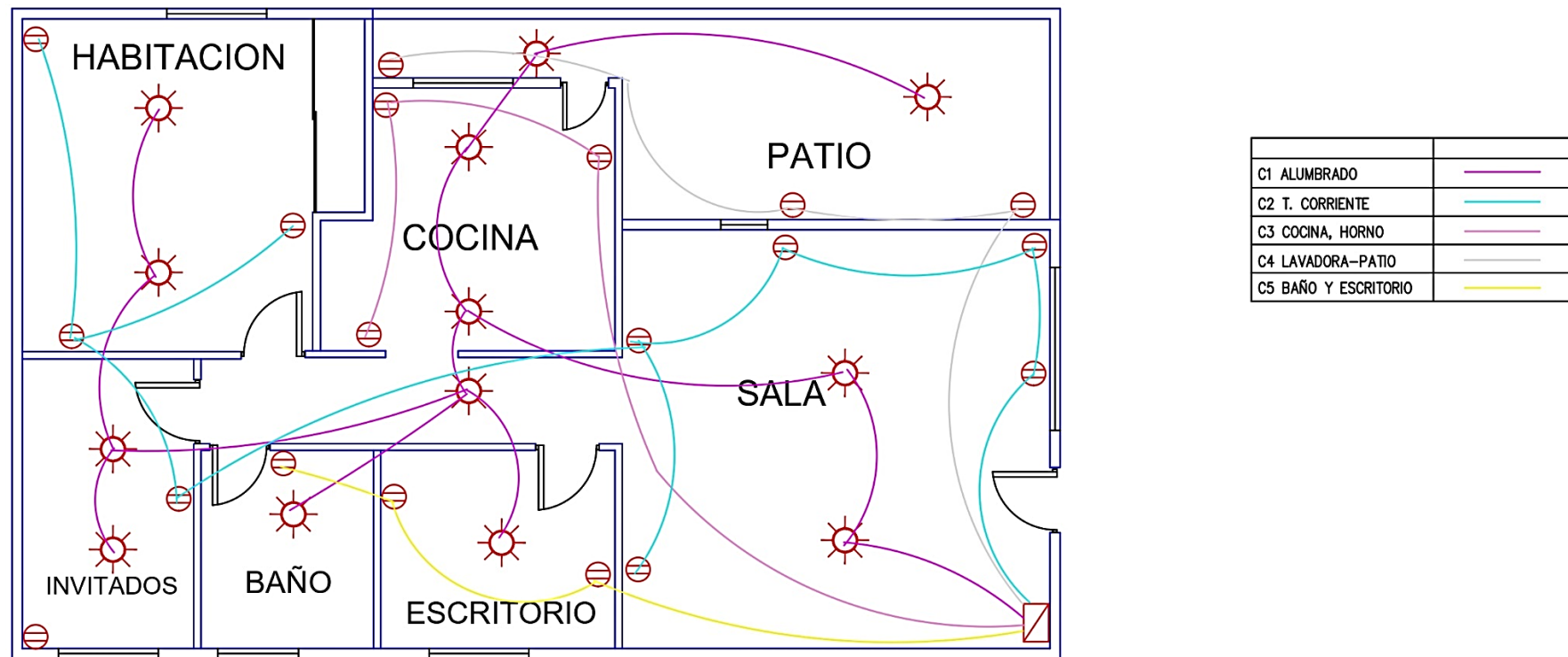
Anexo 2

Figura A1: Diagrama unifilar de la casa de un piso en AutoCAD.



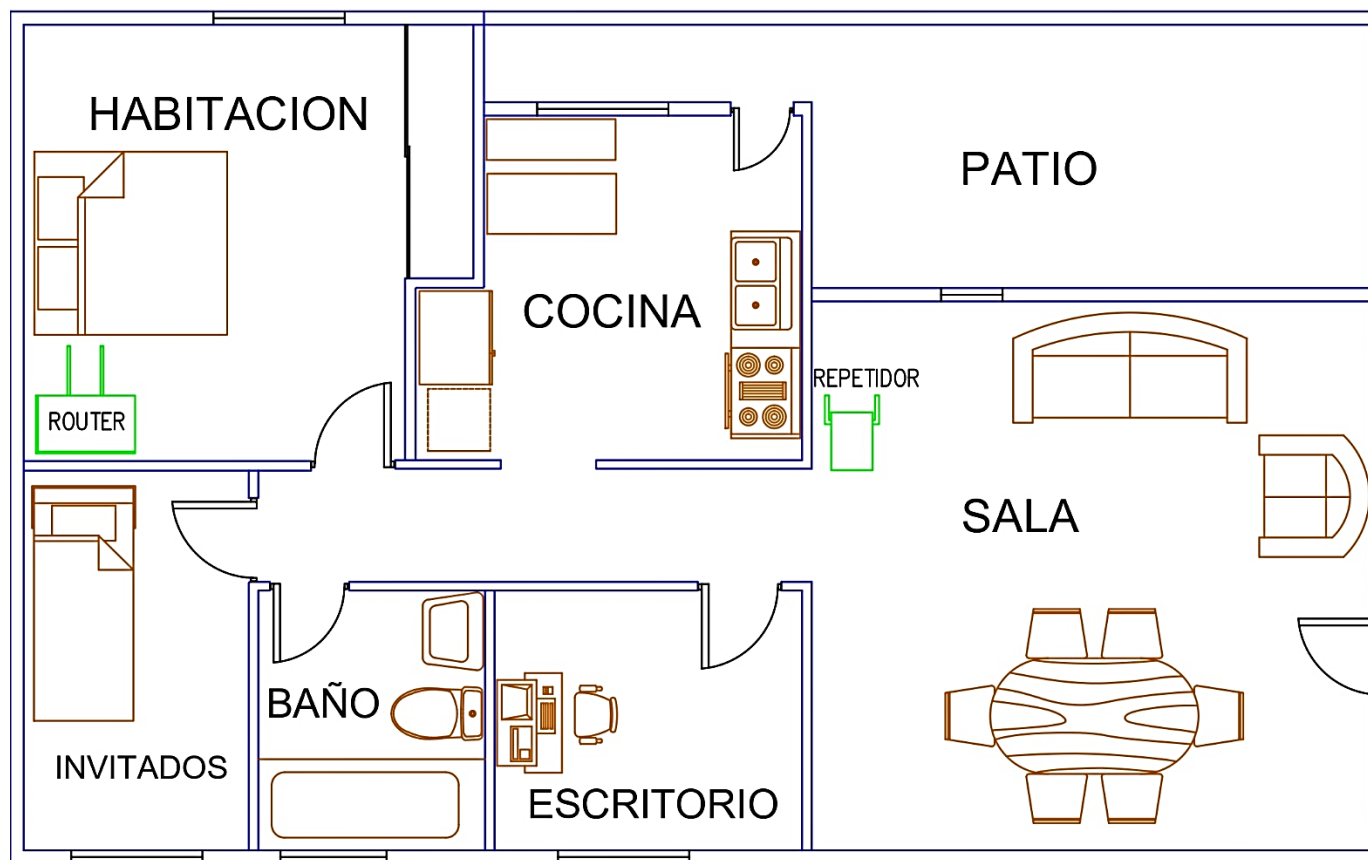
Elaborada por: Autor

Figura A2: Diagrama de circuito electrónico e iluminación de la casa de un piso en AutoCAD.



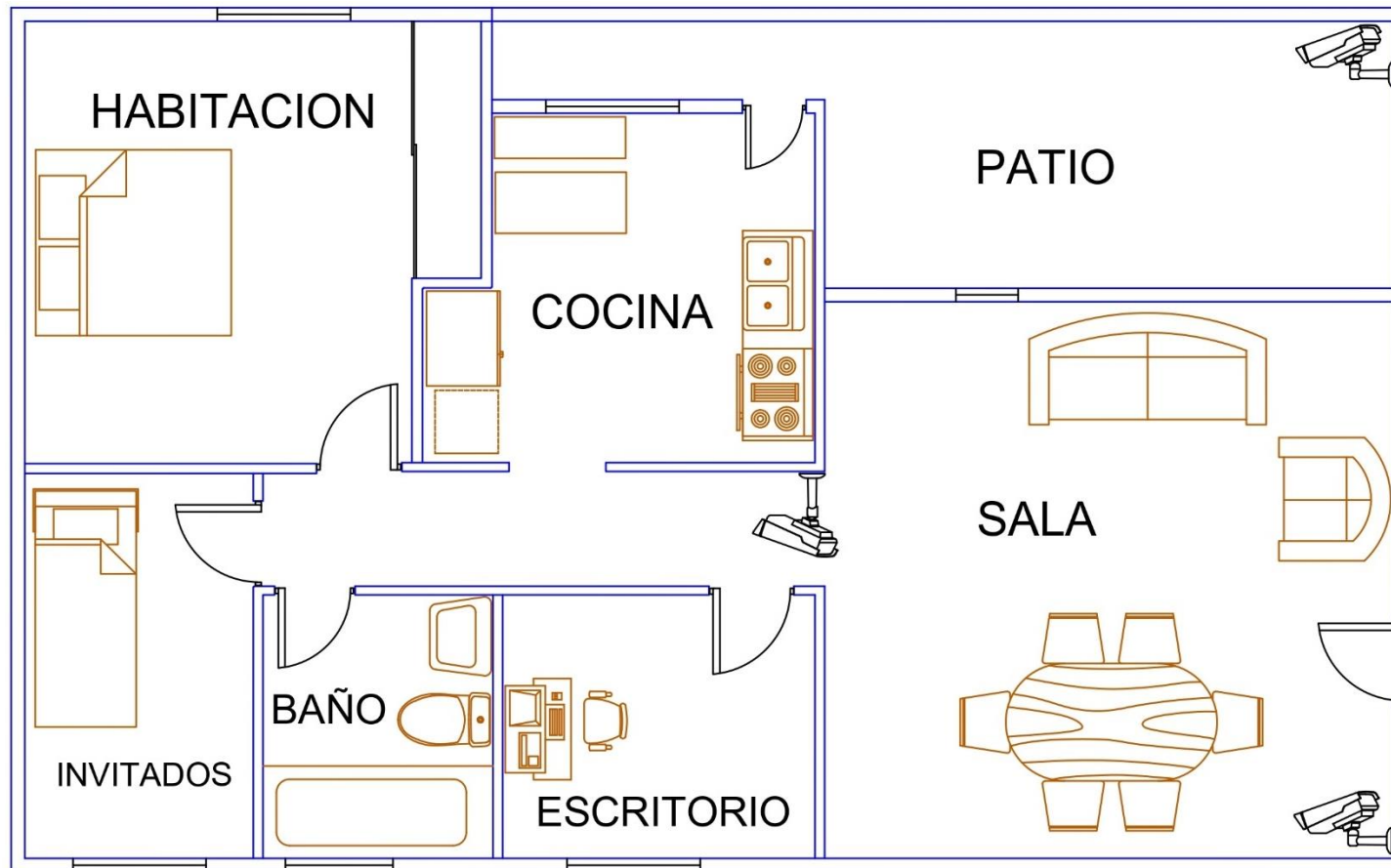
Elaborada por: Autor

Figura A3: Diagrama de instalación de red wifi para cobertura total del domicilio en AutoCAD.



Elaborada por: Autor

Figura A4: Diagrama de ubicación del sistema de cámaras de vigilancia para la casa de un piso en AutoCAD.



Elaborada por: Autor



Presidencia
de la República
del Ecuador



Plan Nacional
de Ciencia, Tecnología,
Innovación y Saberes



SENESCYT
Secretaría Nacional de Educación Superior,
Ciencia, Tecnología e Innovación

DECLARACIÓN Y AUTORIZACIÓN

Yo, **Piedrahita Morán, Jandry Jossue** con C.C: **0941751398** autor del Trabajo de Integración Curricular: **Desarrollo de una propuesta de diseño para un sistema de automatización de una casa inteligente**, previo a la obtención del título de **Ingeniero en Electrónica y Automatización**, en la Universidad Católica de Santiago de Guayaquil.

1.- Declaro tener pleno conocimiento de la obligación que tienen las instituciones de educación superior, de conformidad con el Artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de integración curricular para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la SENESCYT a tener una copia del referido trabajo de integración curricular, con el propósito de generar un repositorio que democratice la información, respetando las políticas de propiedad intelectual vigentes.

Guayaquil, 4 de septiembre del 2024

Piedrahita Morán, Jandry Jossue

C.C: 0941751398



**Presidencia
de la República
del Ecuador**



**Plan Nacional
de Ciencia, Tecnología,
Innovación y Saberes**



SENESCYT
Secretaría Nacional de Educación Superior,
Ciencia, Tecnología e Innovación

REPOSITORIO NACIONAL EN CIENCIA Y TECNOLOGÍA			
FICHA DE REGISTRO DE TESIS/TRABAJO DE INTEGRACIÓN CURRICULAR			
TÍTULO Y SUBTÍTULO:	Desarrollo de una propuesta de diseño para un sistema de automatización de una casa inteligente.		
AUTOR(ES)	Piedrahita Morán, Jandry Jossue		
REVISOR(ES)/TUTOR(ES)	Ing. Vega Ureta, Nino Tello. M.Sc.		
INSTITUCIÓN:	Universidad Católica de Santiago de Guayaquil.		
FACULTAD:	Facultad de Educación Técnica para el Desarrollo		
CARRERA:	Ingeniería en Electrónica y automatización		
TÍTULO OBTENIDO:	Ingeniero en Electrónica y Automatización		
FECHA DE PUBLICACIÓN:	4 de septiembre del 2024	No. DE PÁGINAS:	80
ÁREAS TEMÁTICAS:	Automatización, hogar inteligente, monitoreo de entorno.		
PALABRAS CLAVES/ KEYWORDS:	Raspberry pi 5, DHT-22, controlador L298, sensor TSL2561, sensor de gas MQ-2, sensor de movimiento PIR HC-SR501.		
En el presente trabajo de integración curricular se basa en el diseño de un sistema de control de casa inteligente utilizando Raspberry Pi 5 y Python 3, integrando diversos sensores y actuadores para mejorar la seguridad, eficiencia energética y comodidad del hogar. En el objetivo se centra en establecer un sistema de monitoreo y control para dispositivos eléctricos mediante el uso de relés y sensores, elaborar la programación de Python 3 implementada para el funcionamiento del sistema inteligente y en base a esto diseñar el diagrama de conexiones de conexiones y flujo de funcionamiento del dispositivo. Con la presentación de este proyecto se pretende obtener el diseño de un sistema de control inteligente de un hogar, para la automatización de dispositivos electrónicos mediante la gestión de la Raspberry pi 5, para mejorar el confort de los residentes, a su vez obtener un ahorro de energía y mejorar la seguridad del domicilio a través del uso de sensores y actuadores. Se utiliza la metodología bibliográfica para la consultas teóricas de los distintos sensores a utilizar tales como sensores de movimiento, gas, temperatura y humedad, funcionamiento de actuadores como relés en la automatización, control de dispositivos, las características y funciones de las Raspberry pi 5 como dispositivo de control principal para la gestión eficiente de dispositivos.			
ADJUNTO PDF:	☒ SI	☐ NO	
CONTACTO CON AUTOR/ES:	Teléfono: +593-941751398	E-mail: jandrypiedra@hotmail.com	
CONTACTO CON LA INSTITUCIÓN: COORDINADOR DEL PROCESO DE UTE	Nombre: Ricardo Xavier Ubilla González		
	Teléfono: +593999528515		
	E-mail: Ricardo.ubilla@cu.ucsg.edu.ec		
SECCIÓN PARA USO DE BIBLIOTECA			
Nº. DE REGISTRO (en base a datos):			
Nº. DE CLASIFICACIÓN:			
DIRECCIÓN URL (tesis en la web):			