

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

DESARROLLO DE UN PROTOTIPO DE IoT BASADO EN EL MÓDULO WIFI Y SERVIDOR WEB LOCAL EN LA PLATAFORMA RASPBERRY-PI ENFOCADO PARA PERSONAS CON CAPACIDADES ESPECIALES DE MOVILIDAD

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIERO EN ELECTRÓNICA Y TELECOMUNICACIONES**

ESPINOZA ALTAMIRANO ANTHONY DANILO

DIRECTOR: MSc. RAMIRO EDUARDO MOREJÓN TOBAR

Quito, agosto 2021

AVAL

Certifico que el presente trabajo fue desarrollado por el Sr. Anthony Danilo Espinoza Altamirano, bajo mi supervisión

MSc. RAMIRO MOREJÓN
DIRECTOR DEL TRABAJO DE TITULACIÓN

DECLARACIÓN DE AUTORÍA

Yo Anthony Danilo Espinoza Altamirano, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente

ANTHONY DANILO ESPINOZA ALTAMIRANO

DEDICATORIA

A la prestigiosa Escuela Politécnica Nacional, reconocida Institución de Educación Superior, por haberme concedido el honor y privilegio de estudiar en sus aulas, hasta convertirme en un profesional politécnico.

Al señor Msc. Ramiro Morejón, distinguido catedrático y noble Tutor de este Trabajo de Titulación previo a la obtención del título de Ingeniero en Electrónica y Telecomunicaciones, por su sabia conducción y apoyo, quien ha hecho posible la culminación con éxito de este complejo y esforzado Trabajo.

A los distinguidos maestros y catedráticos de la legendaria Facultad de Ingeniería Eléctrica y Electrónica por sus enseñanzas e invaluables conocimientos que me brindaron.

A mis compañeros, inolvidables seres, que me enseñaron el valor de la amistad.

AGRADECIMIENTO

Dedico este proyecto de titulación:

A mis abuelitos Manuel y Evita, mi gratitud por su entrega y amor desde mi más tierna infancia, por su maravilloso ejemplo de valores morales y de amor infinito.

A mi padre Danilo, quien me supo inculcar desde niño ese amor por abrazar una profesión guiando mi existencia por el sendero del estudio y la superación, por toda la dedicación que puso en mi formación humana y personal.

A mi madre Lorena, que me inculco el valor de la responsabilidad y el trabajo, convirtiéndose en un ejemplo de constancia, perseverancia y abnegación en cada dificultad a lo largo de mi camino.

A mis hermanos André, Paulina y Martin, que me llenaron de alegría y fortaleza, constituyéndose en los pilares fundamentales de mi existencia.

A mi tía Toa, a mi abuelita Elvita y a mis primos que no permitieron que decaiga en los momentos difíciles con su permanente motivación.

Al amor de mi vida Samantha quien sabe que, con dedicación y constancia todas nuestras metas son alcanzables.

Cada uno de ellos, se convirtió en el motor y apoyo incondicional de mi vida en este proceso de lucha constante, que parecía un sueño y que hoy es una feliz realidad.

ÍNDICE DE CONTENIDO

AVAL	I
DECLARACIÓN DE AUTORÍA.....	II
DEDICATORIA.....	III
ÍNDICE DE CONTENIDO.....	V
RESUMEN	VII
ABSTRACT	VIII
1. INTRODUCCIÓN.....	1
1.1. OBJETIVOS	2
1.2. ALCANCE	2
1.3. MARCO TEÓRICO.....	3
1.3.1. DISCAPACIDADES	3
1.3.1.1. Discapacidades motrices.....	3
1.3.1.2. Personas con discapacidad en el Ecuador.....	4
1.3.2. DOMÓTICA	5
1.3.3. IoT	5
1.3.4. TARJETAS DE DESARROLLO	5
1.3.5. RASPBERRY PI	5
1.3.6. RDUINO	8
1.3.7. ARDUINO IDE	8
1.3.8. WIFI ESP8266.....	9
1.3.9. SENSORES Y ACTUADORES.....	10
1.3.10. APLICACIONES WEB	17
1.3.11. SERVIDORES	17
1.3.11.1. Servidor LAMPP	17
1.3.12. PHP	18
1.3.12.1. Librería PDO	18
1.3.13. JAVASCRIPT.....	18
1.3.14. BASE DE DATOS.....	19
1.3.15. MARIA DB	19
1.3.16. VISUAL STUDIO CODE	19
2. METODOLOGÍA.....	21
2.1. TÉCNICAS DE RECOLECCIÓN DE INFORMACIÓN.....	21

2.1.1.	TÉCNICA DE RECOLECCIÓN DE FUENTES VIRTUALES	21
2.2.	METODOLOGÍA DEL PROCESO INVESTIGATIVO	22
2.3.	DESARROLLO.....	23
2.3.1.	SERVIDOR.....	24
2.3.2.	SISTEMA OPERATIVO	24
2.3.3.	SERVIDOR VNC	25
2.3.4.	SERVIDOR LAMPP	25
2.3.5.	BASE DE DATOS.....	26
2.3.6.	API.....	28
2.3.7.	DISPOSITIVOS ELÉCTRICOS.....	31
2.3.8.	CONTROL DE LÁMPARA	34
2.3.9.	CONTROL DE CORTINA	36
2.3.10.	CONTROL DE ILUMINACIÓN LED.....	39
2.3.11.	USUARIO	42
3.	RESULTADOS Y DISCUSIÓN	44
3.1.	FUNCIONALIDAD	44
3.2.	ESCALABILIDAD	44
3.3.	ROBUSTEZ.....	45
3.4.	SEGURIDAD	46
3.5.	EXPERIENCIA DE USUARIO.....	47
4.	CONCLUSIONES Y RECOMENDACIONES.....	48
4.1.	CONCLUSIONES.....	48
4.2.	RECOMENDACIONES	48
5.	REFERENCIAS BIBLIOGRÁFICAS	50
	ANEXOS	53

RESUMEN

En el presente trabajo de titulación se reporta el diseño y la implementación de un prototipo de un sistema domótico basado en Internet de las cosas, IoT (Internet of the Things), enfocado en ayudar a realizar actividades cotidianas a personas con capacidades especiales o discapacidad motriz (mayor incidencia en personas con discapacidad en Ecuador), las acciones fueron determinadas a partir de un análisis de las rutinas cotidianas que más afectan a estas personas, siendo las principales: alcanzar a encender dispositivos.

vos eléctricos de iluminación (lámpara común e iluminación led) y el control de apertura y cierre de las cortinas de una habitación.

El sistema utiliza una tarjeta de desarrollo Raspberry Pi4 que, configurado como servidor permite la interconectividad de los dispositivos. Para dotar de conectividad a los dispositivos a controlar, se utilizó tarjetas de desarrollo WiFi ESP8266 y como actuadores, tarjetas de relé, que permiten la integración del mundo análogo con el mundo digital. Por otra parte, para la interacción del usuario con el sistema, se desarrolló una aplicación web basada en PHP y javascript, pensada para facilitar la interacción de personas con capacidades especiales de movilidad.

PALABRAS CLAVE: IoT, Raspberry Pi, aplicación web.

ABSTRACT

This degree work presents the design, development and implementation of a prototype of a home automation system based on the Internet of things, IoT (Internet of the Things), focused on helping people with special abilities to carry out daily activities. Mobility (higher incidence in people with disabilities in Ecuador), these actions were determined from an analysis of the daily routines that most affect this type of people, the main ones being: reaching to turn on electrical lighting devices (lamp common and led lighting) and the control of the opening and closing of the curtains in a room.

The system uses a Raspberry Pi4 development card that, configured as a server, allows the interconnectivity of the devices. To provide the devices to be controlled with internet, ESP8266 WiFi development cards were used and, as actuators, relay cards, which allow the integration of the analog world with the digital world. On the other hand, for user interaction with the system, a web application based on PHP and javascript was developed, designed to facilitate interaction with people with special mobility abilities.

KEYWORDS: IoT, Raspberry Pi, web application.

1. INTRODUCCIÓN

La Organización Mundial de la Salud (OMS) señala que, más de mil millones de personas viven con algún tipo de discapacidad motriz, y se cree que esta cifra ira en aumento. La prevalencia de la discapacidad debido al incremento de las enfermedades consideradas crónicas como: el cáncer, la diabetes, enfermedades cardio-vasculares, entre otras; limita la movilidad según el Informe Mundial sobre la Discapacidad [1].

En Ecuador el Consejo Nacional para la Igualdad de Discapacidades, en su informe actualizado al 2020 reporta los datos presentados en la Figura 1.1.

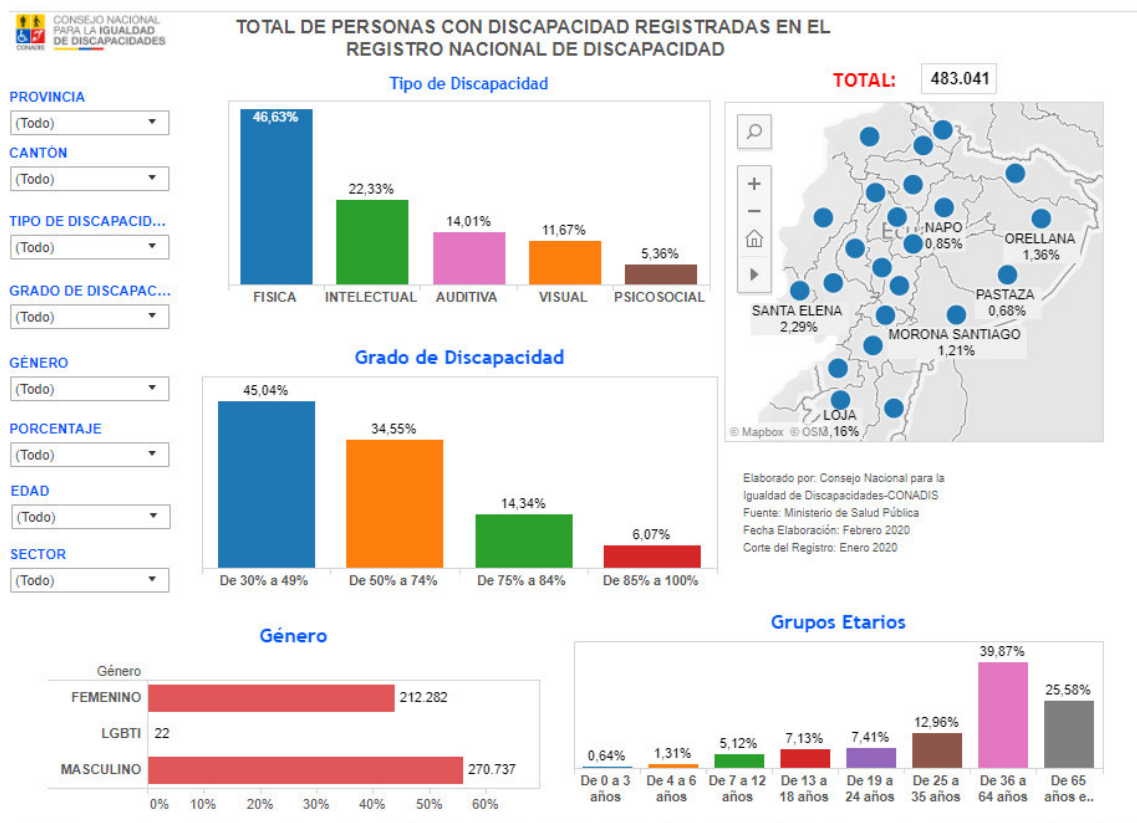


Figura 1.1 Estadísticas de discapacidad en el Ecuador 2020 [1]

Según el informe, en el Ecuador existen más de 480 mil personas que registran algún tipo de discapacidad, de estas los impedimentos físicos representan un 46,63%, con un grado de discapacidad del 45,04%. La edad es un factor determinante por lo que el grupo mayoritario correspondiente al 65.45% supera los 36 años, 39.87% de personas entre 36 a 64 años y 25.58% de personas mayores a 65 años [2].

Si bien es cierto, las políticas a nivel mundial y local se deben enfocar en reducir las causas de las discapacidades, lastimosamente es imposible pensar en un futuro sin personas con discapacidad, ya que no son únicamente enfermedades las responsables de la discapacidad, sino también en muchos casos se debe a accidentes, catástrofes o incluso a causas genéticas, por lo cual es necesario pensar en elaborar servicios y productos tecnológicos que permitan a las personas con discapacidad gozar de una mejor calidad de vida.

Entre otros, un problema que dicho escenario conlleva es la dificultad de que personas con movilidad reducida puedan realizar tareas tales como: abrir puertas, cortinas, encender luces y otros artefactos eléctricos del hogar. Por esto se han planteado muchos trabajos que abordan este problema, sin embargo, esta propuesta quiere incluir un sistema embebido que tiene capacidades de comunicaciones, proceso y sobre todo crecimiento a través de una estructura modular del código que permita que continúe este siempre necesario desarrollo. Para esto se propone el uso de una plataforma de bajo costo y de gran versatilidad.

1.1. OBJETIVOS

El objetivo general de este Proyecto Integrador es: Desarrollar un prototipo de IoT basado en sistemas embebidos como Raspberry PI, para asistir a personas con movilidad reducida en tareas cotidianas.

Los objetivos específicos de este Proyecto Integrador son:

- Estudiar las características del módulo ESP8266, las herramientas, procedimientos de configuración y el sistema embebido Raspberry PI.
- Diseñar la circuitería de soporte como: la de alimentación y fuerza para los dispositivos IoT.
- Implementar una aplicación web en lenguaje PHP con una interfaz amigable para la comunicación y monitoreo del módulo IoT.
- Evaluar el módulo IoT y la interacción con la aplicación web.

1.2. ALCANCE

El módulo ESP8266 es una plataforma reducida que ha sido concebida como un complemento de comunicación WIFI de sistemas embebidos y particularmente con las tarjetas de desarrollo de WIFI ESP8266 y RASPBERRY PI. El módulo actúa como un transceptor entre la red inalámbrica WIFI (802.11) y los MCU utilizando control remoto

UART / SDIO / SPI / I2C / I2S y otros. En este trabajo de titulación se configurará el procesador Tensilica L106 de 32 bits para hacer uso del reducido número de pines disponibles, que permitirá accionar una cortina motorizada.

Para conseguir el funcionamiento adecuado, se requiere manejar los niveles de potencia y voltajes de polarización de cada uno de los componentes por lo que la solución contempla el diseño de acuerdo con los requerimientos de cada uno de ellos.

La aplicación que se comunicará con el dispositivo desarrollado debe presentar un ambiente amigable por lo que será codificada en PHP y tendrá las funcionalidades básicas de abrir y cerrar la cortina. Otras funcionalidades deben ser consideradas para versiones comerciales futuras, ya que el prototipo es una prueba de concepto.

El trabajo tiene un producto final demostrable que consiste en una versión básica del prototipo.

1.3. MARCO TEÓRICO

1.3.1. DISCAPACIDADES

Las discapacidades son condiciones que afectan a las personas, dichas condiciones hacen que se les dificulte el llevar a cabo una acción o tarea, las dificultades pueden variar por varios factores tanto particulares de las personas como con las condiciones sociales del medio en el que la persona viva, para una persona que no pueda caminar o esté en silla de ruedas es más complicado vivir en una zona rural que en una urbana [3].

El término discapacidad engloba varias limitaciones funcionales que tienen las personas en distintas poblaciones al rededor del mundo, pueden ser limitaciones físicas, sensoriales o quizás una dolencia que necesita ser atendida por médicos, las cuales pueden ser temporales como por ejemplo un brazo roto que necesite ser inmovilizado hasta que sane o permanentes como una ceguera de la que no se pueda recuperar, las discapacidades como se explicó en el párrafo anterior son heterogéneas, es decir serán distintas dependiendo de las condiciones que rodeen a las personas [2].

1.3.1.1. Discapacidades motrices

La definición de la RAE de motricidad es: “*Capacidad del sistema nervioso central de producir la contracción de un músculo*” [4]. Unido con el significado de discapacidad se obtiene que una discapacidad motora es una limitación de una persona para realizar movimientos.

Las personas que poseen una discapacidad motora tienen dificultad para realizar actividades, ingresar a lugares o interactuar con objetos, un ejemplo es una persona que no tenga movilidad en las piernas, esto hará que se le dificulte llegar a un segundo piso si no existen rampas o ascensores que le permitan trasladarse, ir al baño también será una tarea complicada si estos no están adecuados para que los usen personas en sillas de ruedas. En conclusión, la discapacidad motriz es limitada por las barreras arquitectónicas complicando su desplazamiento e independencia.

1.3.1.2. Personas con discapacidad en el Ecuador

Según el último censo, realizado en el 2010, en Ecuador aproximadamente el 5,6% de la población sufre de algún tipo de discapacidad, este fue uno de los motivos para que el gobierno de la época se enfoque en brindar atención a las personas con discapacidad, no solo enfocándose en la salud sino también con asistencia social, reconociendo a las personas con discapacidad como un grupo prioritario de atención, por ello el “Plan Nacional del Buen Vivir” contemplaba entre otras cosas mejoras a la calidad de vida para estas personas, incluso se firmaron compromisos con la ONU y la Convención Interamericana para eliminar la discriminación hacia las personas que tengan discapacidad [5].

En la última década se han realizado avances en lo que se refiere a la inclusión de personas con discapacidad en Ecuador, un claro ejemplo de ello es la inserción de estas en el ámbito laboral, apoyado por la constitución del Ecuador creada en el 2008, en el artículo 330 y 334 se habla sobre la necesidad de que las personas con discapacidad tengan garantizada la accesibilidad al trabajo remunerado, no se les deberá reducir el sueldo y el estado incentivará un régimen laboral que vele por las necesidades y el cuidado de las personas con discapacidad, además por ley desde el año 2006 las empresas públicas y privadas que tengan al menos 25 trabajadores deberán contar con un mínimo del 4% que serán personas con discapacidades, a pesar de lo mencionado aún faltan corregir muchos aspectos legales y culturales para lograr una inclusión de las personas con discapacidad [6].

1.3.2. DOMÓTICA

Los orígenes de la domótica se remontan a década de los 70 en Estados Unidos, con la implementación de sistemas de calefacción en edificios los cuales contribuían al ahorro de energía utilizando termostatos para controlar su funcionamiento, aunque en ese tiempo aún no se empleaba el término en sí, ya se comenzó a abrir paso la instalación de sistemas automáticos para domicilios [7].

Actualmente se usa el término de domótica para referirse a automatizaciones dentro de una casa para aumentar el confort de las personas que habitan en ella, ahorrar energía, incluso para aumentar la seguridad del hogar, este tipo de viviendas se denominan casas inteligentes, las cuales mejoran la calidad de vida de las personas minimizando su interacción con los objetos del hogar, ya que cuando se automatizan estos cumplirán sus funciones automáticamente y más [8].

1.3.3. IoT

El Internet de las cosas, más conocido como IoT por sus siglas en inglés (Internet of Things) hace referencia a objetos, con conexión a internet, que se interconectan a través del mismo, teniendo al Internet como una base de datos para tomar decisiones autónomas para funcionar o realizar una u otra acción [9].

Una de las características principales del Internet de las cosas es la integración de sensores y dispositivos en cosas que se utilizan a diario para que estas se controlen automáticamente, siempre que se las haya programado previamente, y cuenten con acceso a una red con Internet para que tengan un lugar de donde obtener la información que ayude a las acciones que tiene que realizar [10].

1.3.4. TARJETAS DE DESARROLLO

Una tarjeta de desarrollo es un hardware que se utiliza frecuentemente en proyectos electrónicos, la tarjeta más conocida es Arduino, la cual es muy utilizada en la enseñanza de ingeniería, para realizar proyectos que interactúen con el ambiente mediante sensores y actuadores, otra tarjeta conocida es la Raspberry Pi que es esencialmente una CPU muy reducida, en el cual se puede instalar un sistema operativo y manejarlo con un mouse, monitor y teclado, en estas dos tarjetas de desarrollo se enfoca este trabajo.

1.3.5. RASPBERRY PI

La creación de Raspberry se remonta al año 2006 cuando la universidad de Cambridge comenzó a darse cuenta de que existía menos interés de estudiar ciencias de la computación y quienes estaban interesados cada vez presentaban menos destreza, así que comenzaron a desarrollar la Raspberry Pi, objetivo que alcanzaron para el año 2011,

aunque su comercialización inició el año siguiente, teniendo gran aceptación del público al punto de colapsar los servidores encargados de la preventa [11].

Raspberry Pi es un ordenador de placa pequeña, el cual cuenta con un hardware potente que puede trabajar con una distribución ligera de Linux, mediante esta placa es más fácil utilizar la programación clásica y la comunicación del Protocolo de control de transmisión/Protocolo de Internet (TCP/IP), además Raspbian puede ser convertido en un sistema operativo, lo que contribuye a la utilización de esta placa, sabiendo que Raspbian es específicamente creado para Raspberry Pi [12].

Raspberry Pi está pensada especialmente para proyectos electrónicos por lo que es importante contar con los puertos que necesite el desarrollo de los diversos proyectos, entre los más importantes se destacan el puerto USB que se usa para instalar el sistema operativo, además al funcionar como un mini CPU también se usará el puerto para conectar teclado y mouse, es necesario una entrada para conectar el monitor para el cual se utiliza la entrada HDMI, para este y muchos otros proyectos se necesita la conexión a Internet por lo que contiene los puertos de comunicaciones TCP/IP, slot para tarjeta SD que almacenará la información requerida, entre otros que se especifican en la Figura 1.2 Puertos de la Raspberry Pi 4.

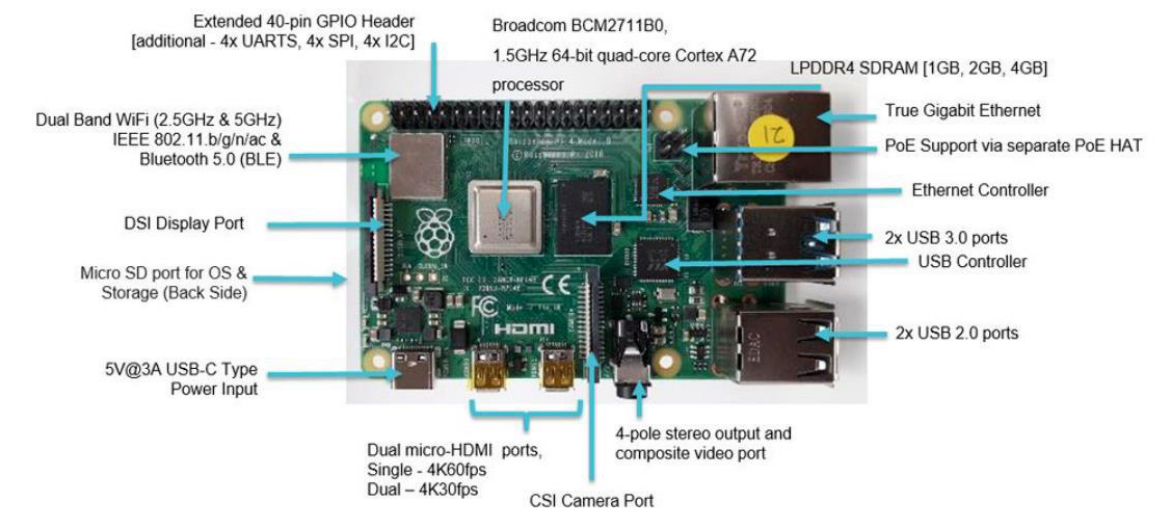


Figura 1.2 Puertos de la Raspberry Pi 4 [13]

El sistema operativo que se utilizará para el desarrollo de este proyecto es Raspbian, un sistema operativo optimizado por Raspberry Pi, para la instalación de este sistema operativo, según [14] se deberá seguir los siguientes pasos:

- Descargar del internet: Win32DiskManager, XLaunch Configuration, Raspbian Wheezy.
- Introducir una tarjeta SD en la computadora que tienen los programas previamente descargados.
- Abrir Win32DiskManager donde se seleccionará la unidad de memoria SD, se da clic en búsqueda de archivos.
- Se busca el archivo de Raspbian Wheezy y se oprime en el botón write, con esto la tarjeta SD está lista para la instalación de Raspbian en la Raspberry PI.
- Se conecta a la Raspberry PI el monitor, teclado, mouse, una fuente de energía y se lo conecta a Internet, es importante que esté conectada a la red para el proceso de instalación.
- Se inserta la tarjeta SD en la Raspberry y se la inicia dando clic en el ícono con XLaunch.
- En la pestaña que se despliega se pone la contraseña, la cual viene configurada por defecto como: Raspberry.

Después de seguir los pasos mencionados se verá una pantalla de escritorio como la que se muestra en la Figura 1.3 Pantalla de escritorio del sistema operativo raspbian, para ingresar a la configuración básica del sistema operativo se debe dar clic en el ícono que tiene el nombre de LXTerminal y se escribe el comando: `sudo raspi-config`, con esto se pueden fijar los parámetros que se utilizarán.

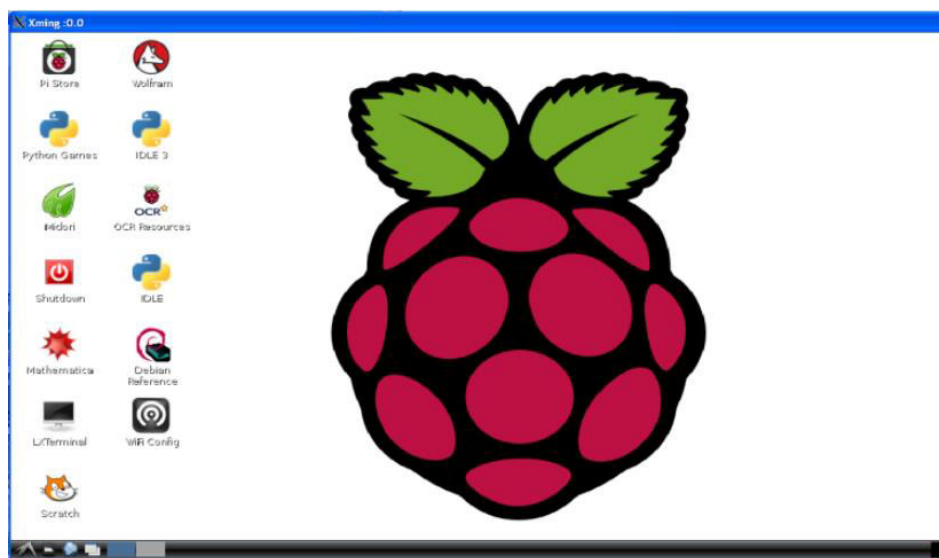


Figura 1.3 Pantalla de escritorio del sistema operativo raspbian [12]

1.3.6. RDUINO

Arduino es un microcontrolador que se utiliza para crear objetos o espacios interactivos, puede percibir las condiciones del entorno mediante sensores y artefactos parecidos, pero también puede intervenir en el entorno mediante motores, luces, entre otras cosas, esta placa puede adaptarse a las necesidades del usuario, el hardware es barato y el software que utiliza se lo puede descargar gratis, esta es una de las principales ventajas. El microcontrolador se programa con el “Arduino Programming Lenguaje” algo que no es difícil de entender [15].

Esta placa es de las más utilizadas en la enseñanza de ingeniería, ya que aparte de la ventaja del costo, tiene una gran comunidad, es decir, existen muchos usuarios de Arduino lo que facilita la búsqueda de información acerca de este microcontrolador, se pueden encontrar varios tutoriales que expliquen su funcionamiento o como programar alguna función específica que necesite el usuario [12].

1.3.7. ARDUINO IDE

Con respecto al entorno de desarrollo, se utiliza Arduino IDE (Integrated Development Enviroment), este fue desarrollado en Java, la interfaz es simple como se demuestra en la Figura 1.4 Arduino IDE, cuenta con diversos botones entre los cuales los más importantes son el verify, que permite identificar problemas en el programa, open que abre archivos, save que abre archivos, entre otros; todo esto ayuda a que sea un entorno de fácil comprensión y que no se necesiten conocimientos avanzados de software para trabajar en este entorno, el lenguaje utilizado básicamente es inspirado en C++ por lo que se tienen las mismas palabras reservadas, las cuales serán resaltadas a lo largo del programa [16].



Figura 1.4 Arduino IDE [16]

1.3.8. WIFI ESP8266

Wifi, es una tecnología que permite las comunicaciones inalámbricas de dispositivos electrónicos, es parecido a una red Ethernet, pero a diferencia de esta, no necesita cables para que se conecten los dispositivos, sino que la conexión se realiza mediante radiofrecuencias o infrarrojos [17].

El ESP8266 forma parte de una generación de computadores embebidos, es un microcontrolador que consta de una antena integrada creada por Espressif, lo que facilita la conexión inalámbrica mediante Wifi, además consta con un software para la programación de aplicaciones para mayor facilidad al conectarse con plataformas del IoT, como es un microcontrolador cumple con las funciones de monitoreo, control y automatización, además tiene puertos de comunicación para facilitar aún más la conexión a Internet con por medio del Wifi [18].

El chip integrado ESP8266 ha sido uno de los más utilizados, para proyectos IoT, es de los más recomendables, es compatible con el protocolo TCP/IP, a pesar de que se lo pueda

conectar a otros controladores, cuenta con, entre otros, puertos GPIO (los que se observan en la Figura 1.5 Módulo ESP8266 y sus puertos) para conectar sensores, por lo que también se puede utilizar sin un controlador y programarlo independientemente, lleva una CPU de 32 bits, 8 pines digitales y uno analógico, además soporta otros protocolos de comunicación como: SPI, I2C, UART; es un módulo económico que ronda los \$4 aproximadamente, además que consume poca energía, lo convierte en uno de los módulos preferidos para el desarrollo de proyectos IoT [19].

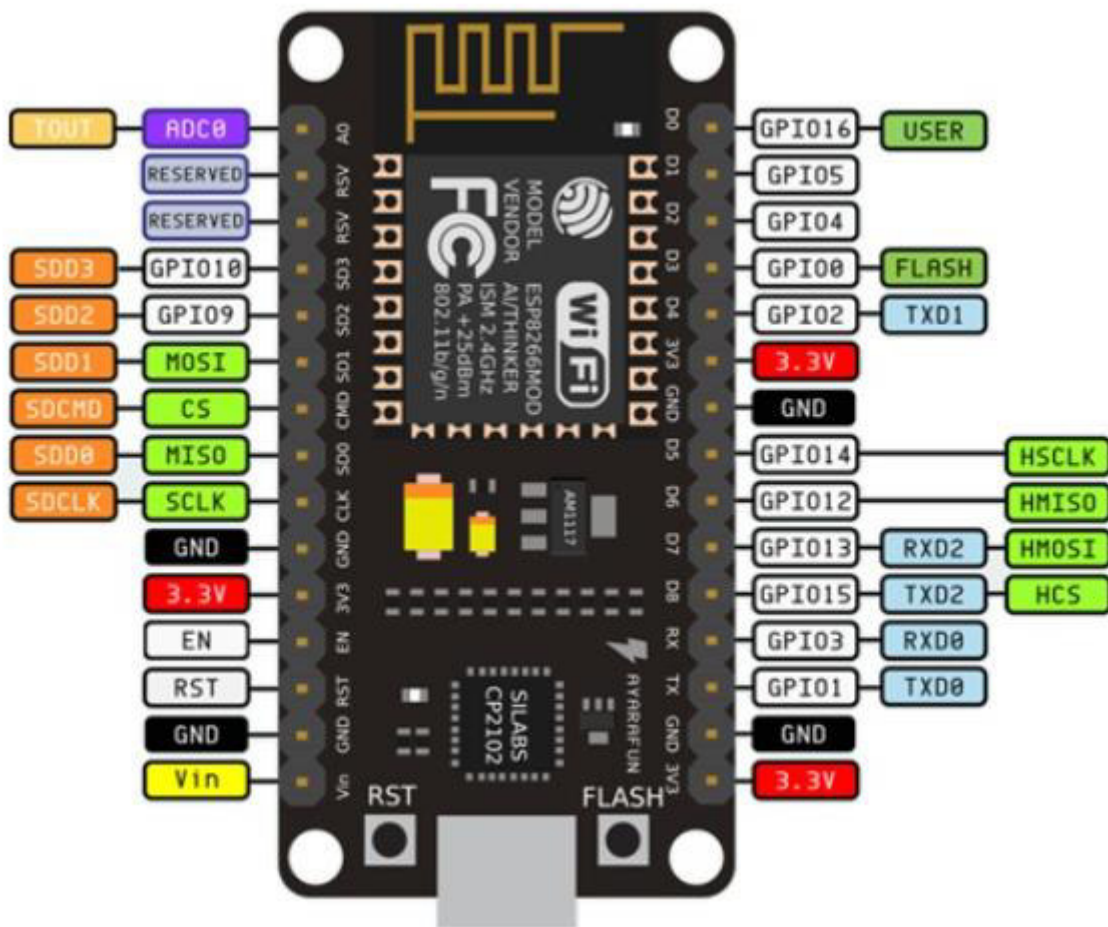


Figura 1.5 Módulo ESP8266 y sus puertos [20]

1.3.9. SENSORES Y ACTUADORES

Un sensor es un dispositivo que interactúa con el medio externo que lo rodea, es como un transductor que toma medidas de diferentes variables físicas, como la temperatura, humedad, iluminación, entre otros, las mediciones de estas variables son traducidas a impulsos eléctricos que servirán para tomar decisiones o simplemente monitorear la variable física que se está midiendo [21].

Los sensores son importantes para proyectos electrónicos en los que interviene el medio en el que se desenvuelve el sistema, porque son los que se encargan de recolectar la información necesaria para determinar las acciones a realizar. Existen dos formas en las que se puede dar la transducción: activa, cuando la magnitud física que se deberá detectar proporciona la energía suficiente al sensor para generar la señal eléctrica y pasiva, cuando la magnitud física que se deberá detectar modificará parámetros del sensor como la resistencia [18].

Los actuadores son dispositivos que tienen la capacidad de transformar energía para ejercer una fuerza, realizar un cambio de velocidad o modificar el estado de un elemento mecánico, en general existen dos clasificaciones de los actuadores: por el tipo de energía que utiliza (neumática, hidráulica, térmica y eléctrica) y por el movimiento que generan (lineal o rotatorio), para el desarrollo de este proyecto los tipos de actuadores son eléctricos [21].

Los actuadores eléctricos son los más utilizados porque esta forma de energía es de fácil acceso, los actuadores de este tipo transforman la energía eléctrica en energía mecánica; la forma en la que trabajan los actuadores eléctricos se basa en el principio físico que indica que si un filamento conductor por el que está pasando corriente se encuentra en la línea de acción de un campo magnético el filamento estará expuesto a una fuerza electromagnética lo que hace que se desplace, los actuadores tienen varios filamentos conductores (espiras), cuando hay corriente estas espiras consiguen propiedades magnéticas, por su forma de funcionar también se los pueden llamar actuadores electromagnéticos [21].

- **Relés**

Un relé utiliza un efecto de electroimán para funcionar, este efecto lo consigue cuando hay corriente circulando por una bobina enrollada a un núcleo de hierro, el objetivo de un relé es que las corrientes de gran valor puedan circular en circuitos de forma controlada, con un circuito de mando de corriente baja, esto se consigue con dos circuitos como se observa en la Figura 1.6 Esquema de funcionamiento de relé, el primero es un circuito de potencia en donde circulará la corriente de alto valor y el otro es un circuito de mando por donde estará circulando la corriente de bajo valor [18].

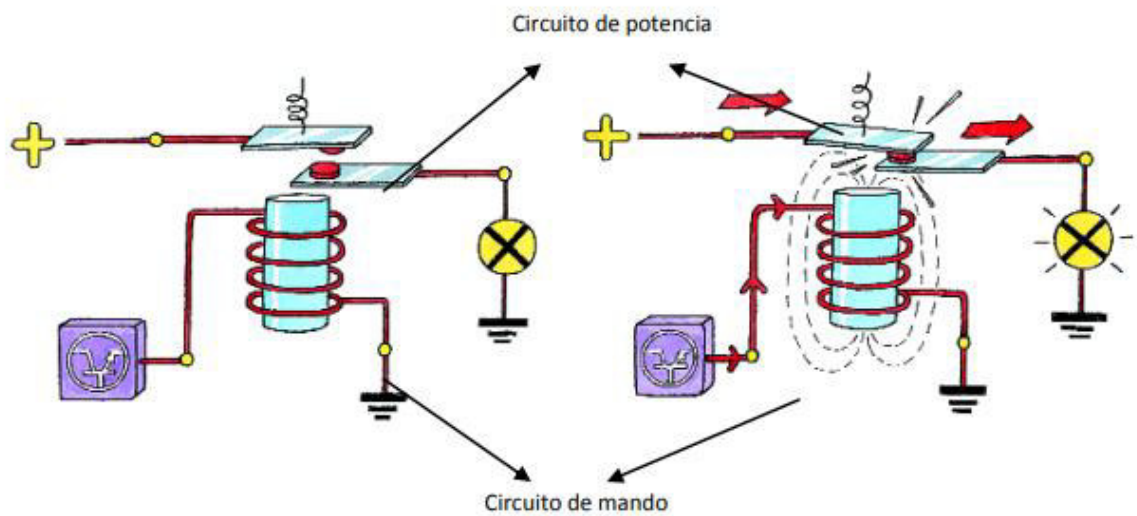


Figura 1.6 Esquema de funcionamiento de relé [23]

- **Válvulas de inyección**

También se las conoce como electroválvulas, son dispositivos electromagnéticos que se encargan de abrir y cerrar el circuito de presión, consta de una bobina ubicada en el cuerpo de la válvula y una aguja de inyección que se mantiene en reposo, lo que impide el paso de combustible. Cuando la válvula recibe un impulso de la unidad de control se producirá un efecto de electroimán con la bobina y se levantará la aguja permitiendo el paso de combustible, la cantidad de combustible que suministra la válvula depende enteramente del tiempo que estuvo alzada la aguja, en la Figura 1.7 Esquema de funcionamiento de una electroválvula se puede ver un esquema de funcionamiento de estas electroválvulas [22].

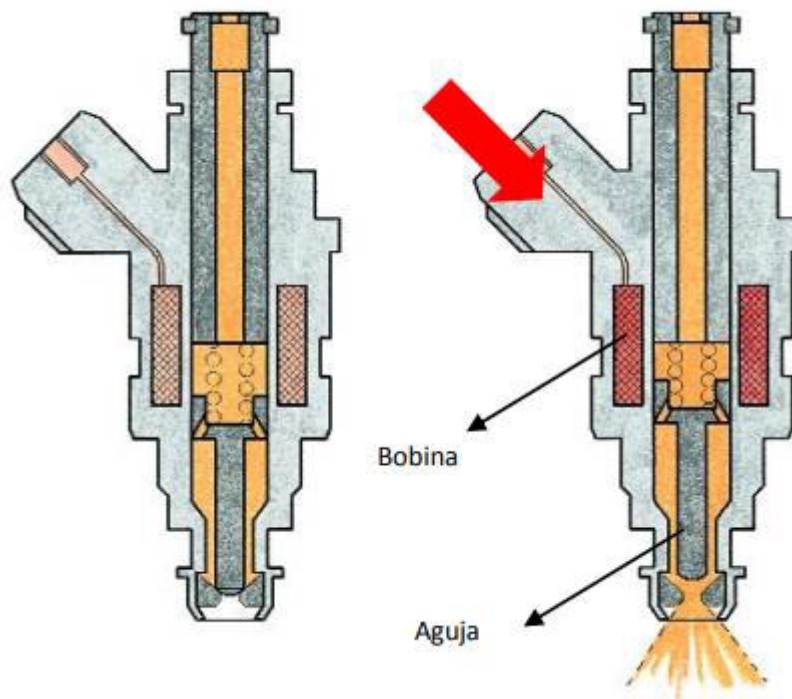


Figura 1.7 Esquema de funcionamiento de una electroválvula [22]

1.3.5.1 Tipos de controles

La característica esencial de los sistemas automáticos es que pueden iniciar o detener procesos sin necesidad de la intervención de las personas, esto lo logran con la ayuda de los sensores que reciben la información del medio en que se encuentra el sistema, el cual puede tener diversas respuestas según la información obtenida, esto es un sistema de control digital [24].

- **Control PID**

Es un sistema de control que a través de un mecanismo de retroalimentación identifica diferencias entre el valor deseado de la variable y el valor real que está teniendo dicha variable, es llamado PID por sus tres parámetros esenciales: Proporcional, Integral y Derivativo, el parámetro proporcional es el que mide la diferencia porcentual entre el valor actual y el deseado, el parámetro integral hace referencia al tiempo que se demora la corrección del sistema, para esto se debe tener en cuenta que entre menos tiempo se utilice más inestable será el sistema y el parámetro derivativo cumple la función de predicción, anticipa posibles errores y los corrige antes de que sean errores graves que tengan consecuencias en los sistemas, en la Figura 1.8 Esquema de control PID se observa un esquema de funcionamiento de estos controles [19].

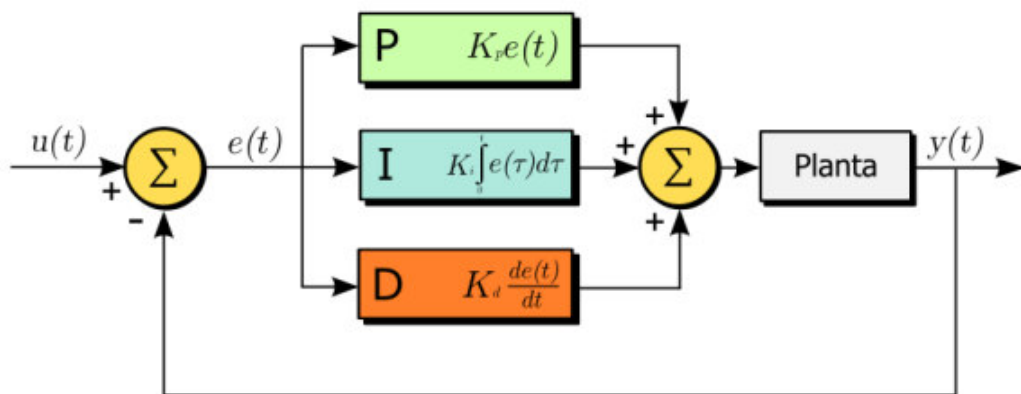


Figura 1.8 Esquema de control PID [24]

- **Control PWM**

El control por modulador de anchura de pulsos conocido como PWM por sus siglas en inglés (pulse-width modulation) es un sistema que consiste en restar a una señal moduladora de forma sinusoidal una señal triangular que varía entre -1 y 1 (voltios) a una frecuencia de 3420 Hz, esta señal es la que controlará la conmutación de interruptores, cuando la señal moduladora sea mayor a la triangular devolverá un 1 mientras que cuando la señal moduladora sea menor que la triangular se obtendrá un -1, así se obtendrá una señal cuadrada con máximo 1 (voltios) y mínimo -1 (voltios) como se observa en la Figura 1.9 Modulación de anchura de pulsos, “ma” es la señal moduladora y “ua” la señal cuadrada [25].

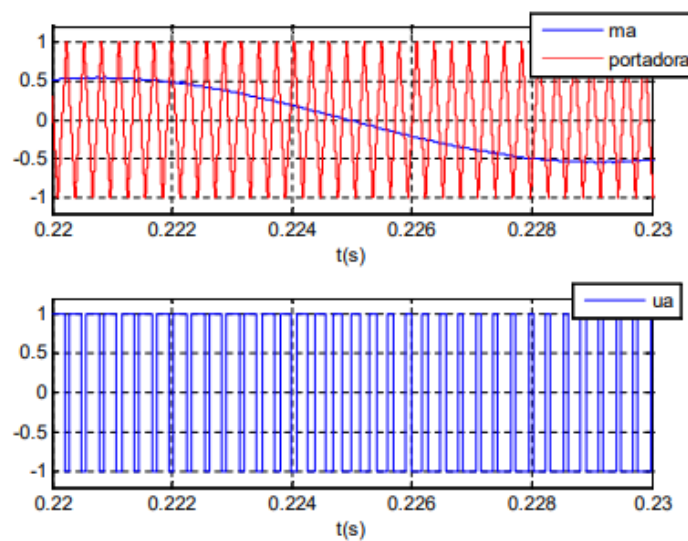


Figura 1.9 Modulación de anchura de pulsos, “ma” es la señal moduladora y “ua” la señal cuadrada [24]

- **Control de histéresis**

Este control utiliza señales V_{in} (Voltaje de entrada) para su funcionamiento, es parecido al PWM, se tendrán valores referenciales a h y V ambos en positivo y negativo para la conmutación de interruptores, cuando la señal V_{in} llegue al valor de $-h$ se tendrá el valor de $-V$, hasta que la señal alcance un valor por encima de $+h$ en cuyo caso la señal pasará a tener el valor de $+V$, en la Figura 1.10 Control por histéresis se observa más didácticamente cómo funciona la señal y cuando ocurren los cambios de valores [19].

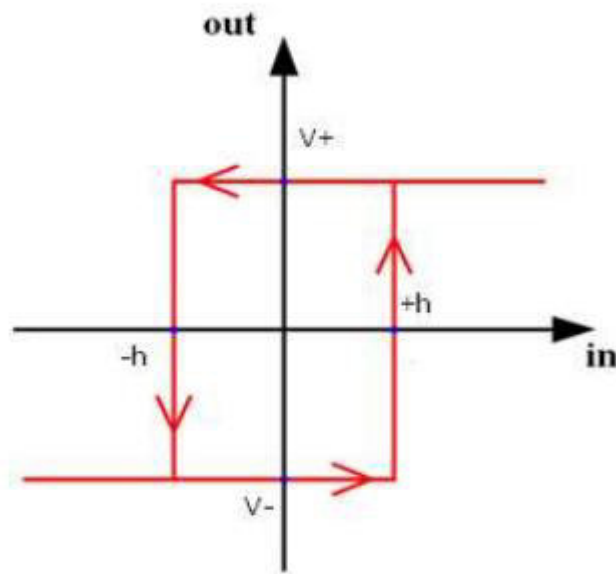


Figura 1.10 Control por histéresis [24]

- **Control ON/OFF**

Este tipo de control tiene la característica de que la señal de comando tiene dos estados, el encendido y apagado, en sistemas con este tipo de elemento es común ver oscilaciones las que se denominan ciclo límite, los parámetros dependerán de las condiciones para las que fue diseñado. Un ejemplo es en el sector agroindustrial para los sistemas de refrigeración, cuando la temperatura sobrepase un límite este se dará inicio al compresor que disminuye la temperatura cuando este bajo el límite se apagará el compresor, encendiéndolo o apagándolo, en la Figura 1.11 Esquema de control ON/OFF se encontrará un esquema del funcionamiento de este control con sus dos estados [26].

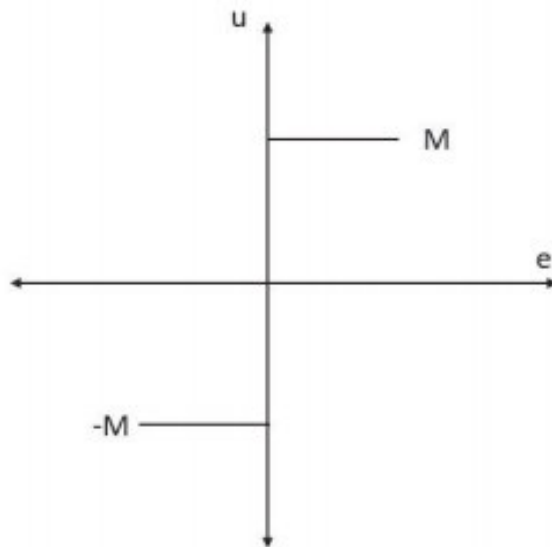


Figura 1.11 Esquema de control ON/OFF [26]

- **Control a lazo abierto**

Este busca que la variable que interviene en el proceso alcance valores deseados según instrucciones dadas al inicio del proceso, es de fácil implementación, aunque no es recomendable para la industria, ya que se deberá intervenir manualmente si detecta acciones externas, las cuales pueden ser ruidos o disturbios, algo que está siempre presente en las industrias, en la Figura 1.12 Esquema de control de lazo abierto encontramos un esquema de cómo funciona el control de lazo abierto [26].

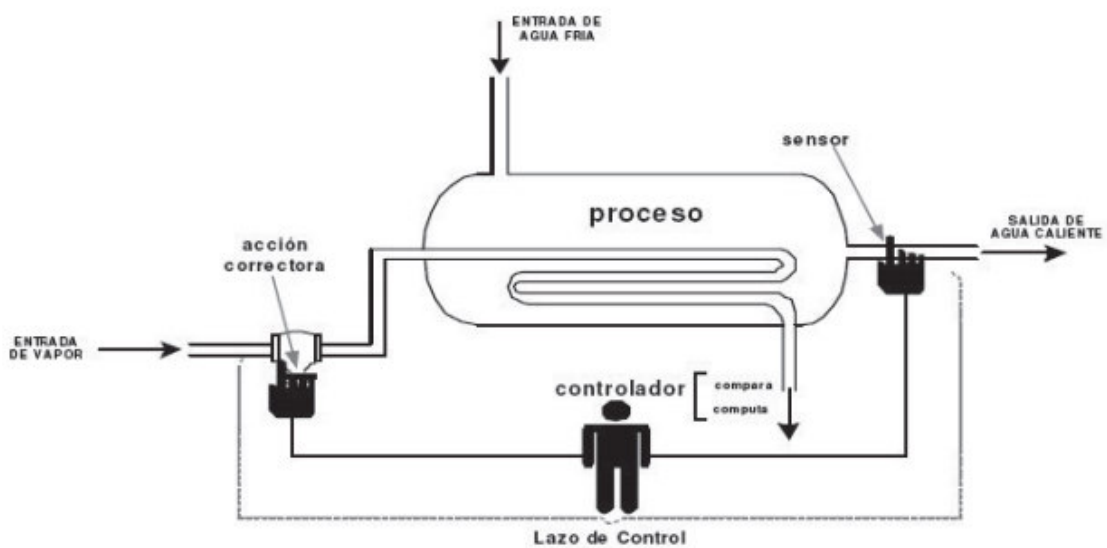


Figura 1.12 Esquema de control de lazo abierto [26]

- **Control a lazo cerrado**

Busca anomalías en la variable cuando sale mediante la retroalimentación, así se mantendrá controlada automáticamente la variable con la que se esté trabajando, a diferencia del sistema a lazo abierto este si puede corregir acciones externas, aunque solo lo podrá hacer después del impulso con desviaciones, ya que este volverá al sistema de control para su monitoreo, en la Figura 1.13 Esquema de control a lazo cerrado se muestra un esquema de cómo funciona este control [26].

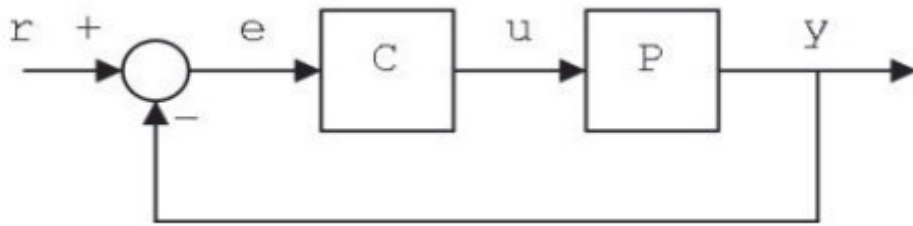


Figura 1.13 Esquema de control a lazo cerrado [26]

1.3.10. APLICACIONES WEB

Las aplicaciones web son aplicaciones del tipo cliente/servidor, el cliente puede ser el navegador o un explorador que usa un usuario, lo que es un servidor se explicará en la siguiente sección, el programador encargado de desarrollar la aplicación no creará ni el cliente, ni servidor, ni el protocolo que utilizarán estos dos para comunicarse, estos tres elementos están estandarizados, y el protocolo utilizado es el HTTP [27].

1.3.11. SERVIDORES

Los servidores son los encargados de responder la solicitud que hace el cliente mediante el protocolo HTTP para contactarse, son ordenadores de gran potencia [16], que almacenan la información que requiere el cliente, los servidores web en cambio son programas, aunque realizan la misma tarea, es decir se mantienen esperando solicitudes de clientes para realizar la conexión.

1.3.11.1. Servidor LAMPP

Este servidor está conformado por cuatro tecnologías: Linux, Apache, MySQL y PHP, las iniciales de cada una forman LAMP, es útil para crear sitios web, aplicaciones entre otras cosas. Es gracias al conjunto de servicios y tecnologías que se logra conformar la plataforma que se busca, además es de las más utilizadas, ya que es gratuito, fácil de instalar y tiene gran estabilidad para montar las aplicaciones web. Linux es el sistema operativo pilar de la plataforma, se puede usar cualquier sistema Gnu/Linux. Apache

Permite alojar y despachar páginas web ejecutadas con PHP. MySQL es la base de datos que se utiliza, es de la más utilizada en el mundo, permite almacenar datos y despacharlos. PHP es el lenguaje de programación que se emplea en la mayoría de páginas web, es con este lenguaje que se desarrolla la plataforma requerida [28]. Se puede instalar en casi cualquier computador, debido a que los requerimientos de hardware son mínimos, lo único que se necesita es: 128 Mb de RAM, CPU de 800 MHz, 1 GB de disco duro. Es útil para sitios de pruebas o temporales del propio ordenador, si se recibirán miles de usuarios diariamente o para plataformas de streaming o almacenar video se requerirán otras plataformas adecuadas.

1.3.12. PHP

PHP es un lenguaje de programación con sintaxis parecida a java o C++, el término PHP viene del inglés Hipertext Preprocessor, con este lenguaje de programación se pueden desarrollar todo tipo de programas, pero lo más común es usarlo para la creación de páginas web, incluso suele estar incluido en páginas HTML, quien se encarga de ejecutar los programas creados con PHP para páginas web es el mismo servidor web [29].

La primera versión de PHP fue creada por Rasmus Lerdfor quien lanzó esta versión en 1995, desde entonces tuvo gran acogida por ser un lenguaje simple y completo para desarrollar aplicaciones y páginas web, actualmente la implementación está a cargo de PHP Group, quien lanzó la más reciente versión que es PHP 7. Es un lenguaje de libre acceso ayudando a los desarrolladores seguir extendiendo sus proyectos [26].

1.3.12.1. Librería PDO

En el lenguaje nativo de PHP, para el acceso a distintas bases de datos se utilizan distintas palabras, lo que dificulta el conmutar entre diversas bases de datos al momento de programar una página web, para solucionar este problema PHP ofrece una extensión para poder conmutar libremente por diversas bases de datos, esta solución es la librería PDO (Objetos de datos de PHP), la cual trabaja con conjuntos estándar de funciones para la comunicación con las bases de datos; esto hace referencia a que se podrá realizar consultas, obtener datos y manejar errores con mayor facilidad [27].

1.3.13. JAVASCRIPT

Es un lenguaje de programación especializado para aplicaciones web, ya que permite la implementación de funciones sofisticadas dentro de sistemas web, el lenguaje javascript permite la correcta interacción entre diversos lenguajes de programación como PHP, y lenguajes de maquetación como HTML o CSS.

La principal fortaleza de javascript es que permite crear contenido de actualización dinámica.

1.3.14. BASE DE DATOS

Una base de datos es una memoria externa en la que se guardará gran cantidad de información, esta base de datos se define y crea solamente una vez, las bases de datos son diseñadas para satisfacer las necesidades de las organizaciones [27], es decir que existen bases de datos especializadas para hospitales, empresas, universidades, entre otros; estos datos estarán compartidos con todos los miembros de la organización para la que se asignó la base de datos, las más utilizadas son: Oracle, MariaDB, SQL-Server, MySQL, DB2, etc.

1.3.15. MARIA DB

Es una base de datos creada por Michael “Monty” Widenius, actualmente es la base de datos más utilizada principalmente porque es de software libre, su principal competencia es MySQL, una base de datos que fue comprada en 2008 por Sun Microsystems [32], la cual a su vez compró Oracle en 2010 por lo que actualmente no es un software totalmente libre, ya que algunos plugins de MySQL son de propiedad exclusiva de Oracle. Por este motivo Maria DB ha tenido gran acogida dentro de los desarrolladores debido a que además tiene gran similitud con MySQL, algo que es de esperarse si es creada por el fundador de MySQL Michel Widenius.

1.3.16. VISUAL STUDIO CODE

Es un editor de texto desarrollado por Microsoft, compatible con Windows, Linux y macOS, ayuda al programador facilitando la depuración y el control del programa, también proporciona gran ayuda visual como se demuestra en la Figura 1.14 resaltando con distintos colores las palabras reservadas y creando espacio entre las sentencias de control, esto facilita entender hasta donde se extiende cada bloque de control del programa creado [33].

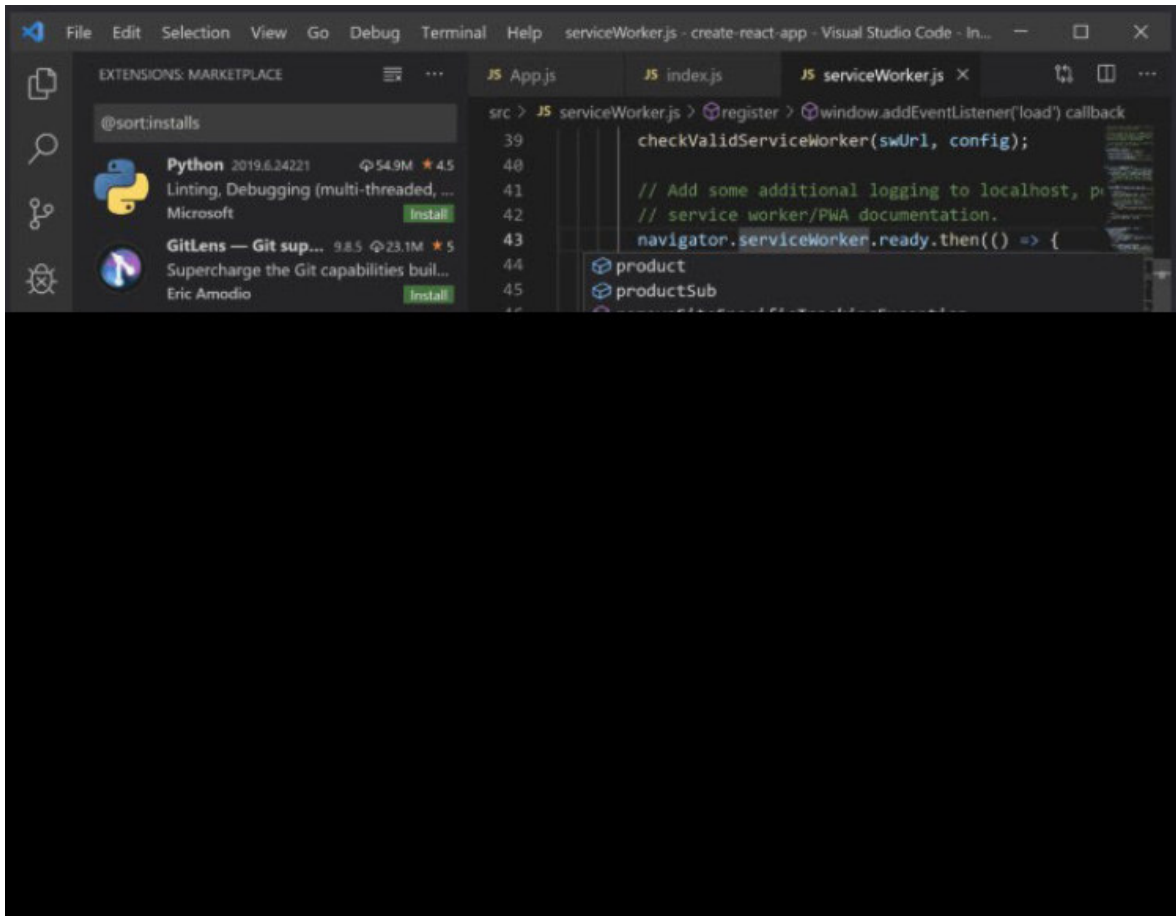


Figura 1.14 Código escrito en Visual Studio Code [15]

2. METODOLOGÍA

El presente trabajo de titulación se lo ha realizado bajo una metodología analítica – experimental, con el objetivo de diseñar y desarrollar un prototipo de IoT basado en sistemas embebidos tales como Raspberry PI y que será entregado a la sociedad con la finalidad de asistir a personas con movilidad reducida en tareas cotidianas como la apertura de cortinas o el encendido y apagado de aparatos eléctricos como lámparas e iluminación.

Se emplea en un inicio una metodología analítica para extraer información de fuentes fidedignas, pudiendo extrapolar las características y funcionalidades necesarias para cumplir con el objetivo del proyecto además de lograr una línea base que sea reproducible para la comunidad científica permitiendo de esta manera beneficiar a futuro a muchas más personas con movilidad reducida. De la misma manera se habla de una metodología experimental, ya que el desarrollo del prototipo involucra manufactura de circuitos y programación de tarjetas de desarrollo, que nos permita mediante la experimentación de prueba y error, entregar un prototipo con la menor cantidad de errores posible, y la mayor cantidad de ventajas competitivas frente a productos comerciales del mercado.

Los recursos ocupados en la realización del presente proyecto de titulación son, a nivel físico, placas de desarrollo como Arduino, Raspberry Pi, y módulos WiFi ESP8266, mecanismos de control como relés y puentes H. A nivel lógico se puede mencionar la utilización de entornos de desarrollo como el IDE de Arduino, y diversos lenguajes de programación como C (para tarjetas de desarrollo) y PHP (para el desarrollo del software que controlará el prototipo).

2.1. TÉCNICAS DE RECOLECCIÓN DE INFORMACIÓN

La información que será recopilada para la realización del presente proyecto de titulación es de vital importancia, por tal motivo su correcta selección garantizará la procedencia y veracidad de la información mostrada. Debido a las circunstancias actuales en el mundo se priorizará la obtención de información de fuentes virtuales, especialmente la información mostrada en portales reconocidos a nivel mundial como universidades, fabricantes de la industria tecnológica, e información de entidades gubernamentales y no gubernamentales, entre otros. Sin embargo, la información de fuentes tradicionales será también considerada sobre todo la existente en la biblioteca de la institución.

2.1.1. TÉCNICA DE RECOLECCIÓN DE FUENTES VIRTUALES

Las fuentes virtuales representan la mayor cantidad de información existente en la actualidad, gracias a estas es posible conocer de manera inmediata alcances y desarrollos

logrados en diversos ámbitos por parte de la comunidad científica alrededor del mundo, sin embargo, es necesario que esta información sea filtrada de manera adecuada, por tal motivo se aplicarán los siguientes parámetros antes de su utilización.

- **Validez científica del documento.** – Se descartarán fuentes que no sean verificables, se hará énfasis en tomar información de publicaciones, libros, papers, etc., de entidades científicamente reconocidas a nivel mundial, como universidades o empresas.
- **Pertinencia del documento.** – Toda la información mostrada en el presente proyecto de titulación tendrá relación directa y se habrá utilizado para el desarrollo del mismo, evitando información no trascendental, mostrando la información precisa y necesaria para el lector.
- **Soporte Bibliográfico.** - No se considerará información que no cuente con un sustento de responsabilidad, sin importar que se encuentre en repositorios conocidos o no.

2.2. METODOLOGÍA DEL PROCESO INVESTIGATIVO

Para definir el método investigativo a emplear se detalla a continuación la Tabla 2.1.

Tabla 2.1 Metodología del proceso investigativo [Fuente: Elaborado por el autor]

Etapa de investigación	Métodos			Técnicas
	Empíricos	Teóricos	Matemáticos	
Fundamentación Teórica		Analítico		Revisión bibliográfica
Marco Metodológico	Recopilación de información		Diseño matemático	Revisión de documentos y diseños similares publicados
Propuesta		Analítico		
Implementación	Metodología de prueba y error		Cálculos de sintonización	

2.3. DESARROLLO

Se desarrolló un prototipo domótico basado en IoT para personas con capacidades especiales de movilidad. Un diagrama general del proyecto es presentado a continuación en la Figura 2.1.

Como se puede apreciar en la Figura 2.1, la intención del proyecto fue lograr el control de uno o varios dispositivos a través de la red, lo cual permitirá que el proyecto sea escalable y pueda ser montado en un servidor en Internet, logrando un IoT que permita la interconexión entre los equipos y el controlador sin importar la distancia entre ellos.

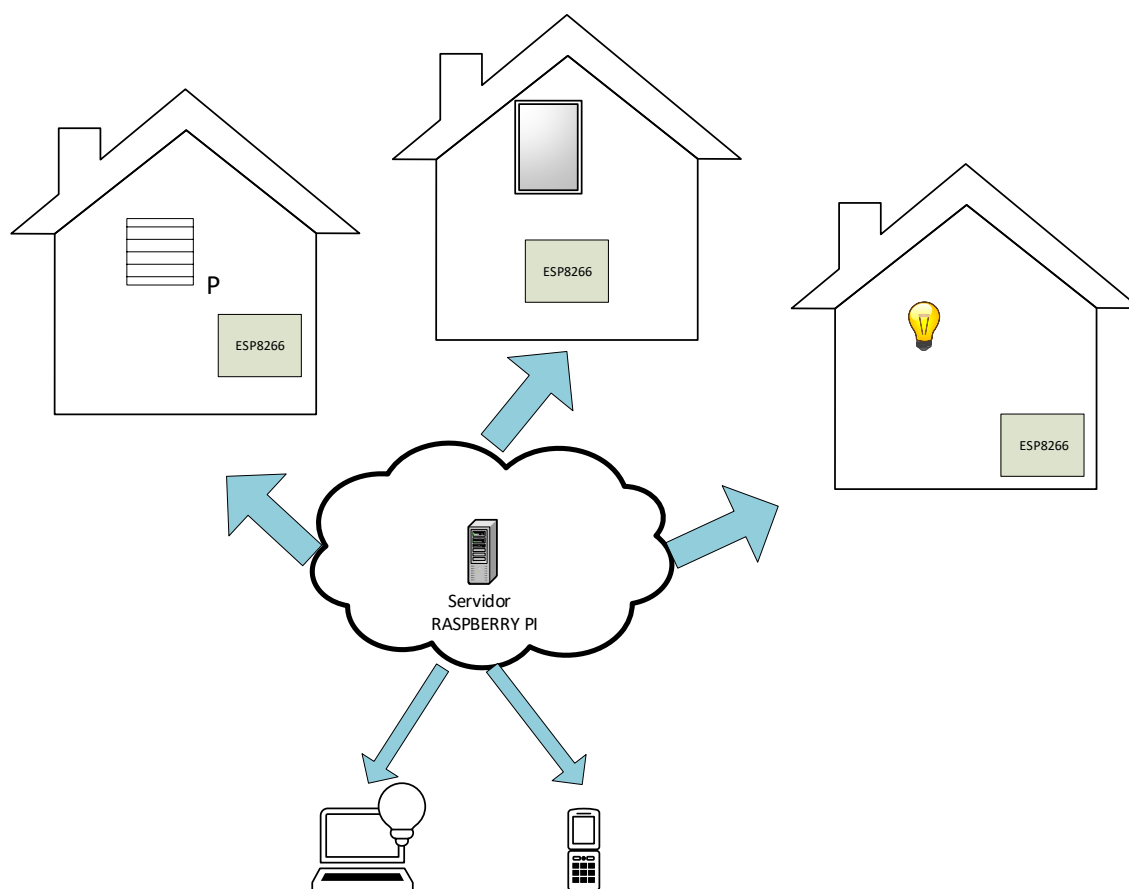


Figura 2.1 Diagrama general del proyecto [Fuente: Elaborado por el autor]

Para ejemplificar de mejor manera la funcionalidad del proyecto se basará en el diagrama de la Figura 2.1, donde se observa que los equipos eléctricos a ser controlados (luz, cortinas, iluminación) estarán conectados a la red mediante una tarjeta WiFi ESP8266, mientras que en la red estará funcionando un servidor Lampp, montado en una tarjeta de desarrollo Raspberry Pi, en este dispositivo se almacenan las órdenes a ejecutar, la base de datos y la interfaz que podrán ver los equipos finales. Siendo los equipos finales, computadoras, celulares, o cualquier dispositivo que soporten aplicativos web.

En otras palabras, las personas mediante sus dispositivos móviles, podrán controlar sus aparatos eléctricos, mediante un aplicativo web que se encontrará alojado en el servidor Raspberry Pi.

Como se observa en la Figura 2.1 existen tres etapas o niveles donde el proyecto fue desarrollado, que son:

- Dispositivos eléctricos
- Servidor
- Usuarios

A continuación, en la Figura 2.2 se ejemplifica los niveles mencionados y su relación entre ellos, donde queda claro que el núcleo central es el servidor, este se relacionará tanto con los dispositivos eléctricos como con el usuario, sin embargo, los niveles del usuario y los dispositivos eléctricos no interactúan entre sí de manera directa.

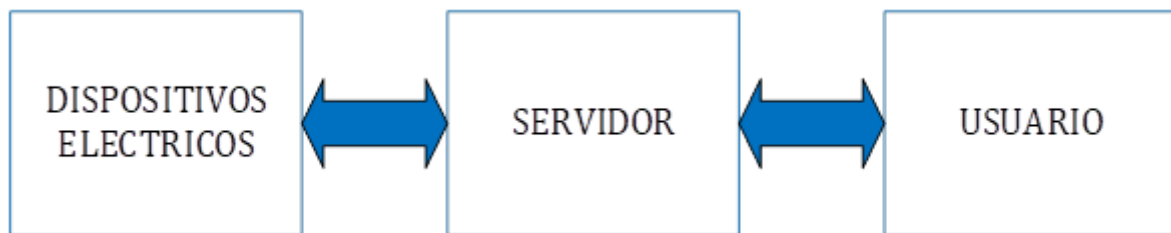


Figura 2.2 Etapas de funcionamiento del prototipo[Fuente: Elaborado por el autor]

2.3.1. SERVIDOR

Como servidor se empleó el dispositivo Raspberry Pi 4, cuyas características se encuentran resumidas en el apartado 1.3.5.1, una de las principales ventajas de su utilización como servidor de este dispositivo es su costo bajo y su versatilidad debido a la gran cantidad de desarrolladores que utilizan este equipo.

2.3.2. SISTEMA OPERATIVO

La Raspberry Pi 4, acepta varias y distintas distribuciones para utilizar como sistema operativo, entre las principales se pueden nombrar: Raspbian (Sistema basado en Linux optimizado para Raspberry), Kali Linux (utilizado principalmente en el mundo del hacking por sus herramientas), RetroPie (sistema operativo para juegos retro), e incluso existe un sistema operativo basado en Windows como es Windows 10 IoT Core, lastimosamente este último no incluye interfaz gráfica y dificulta su utilización [22].

Por tal motivo para el desarrollo del prototipo se emplea el Sistema Operativo Raspbian, que es el recomendado por el fabricante, cuya interfaz y entorno gráfico se presenta a

continuación en la Figura 2.3. Como se encuentra explicado en el apartado 1.3.5.1, la tarjeta no cuenta con dispositivos periféricos.

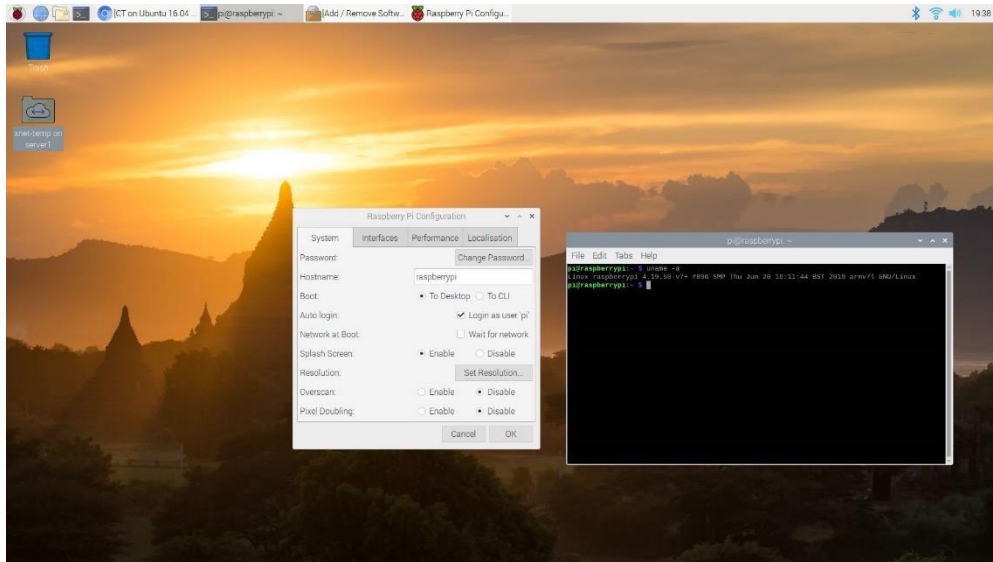


Figura 2.3 Entorno de Raspbian [Fuente: Elaborado por el autor]

2.3.3. SERVIDOR VNC

Una vez que la Raspberry Pi 4 tiene instalado el sistema operativo, se procede a la instalación de los paquetes y aplicativos que permitirán cumplir con el desarrollo del proyecto.

En primera instancia se instala en la Raspberry un servidor VNC (Virtual Network Computing) que permite el control remoto del dispositivo Raspberry siempre y cuando el computador se encuentre en la misma red. Cabe mencionar que este servidor se lo instala para la etapa de desarrollo del prototipo, ya que brinda facilidades de acceso y evita la utilización de periféricos lo cual resulta molesto, siendo la Raspberry un equipo de dimensiones tan pequeñas, Para la etapa de producción se puede prescindir de este servidor y bastará con copiar el archivo completo de lo desarrollado en otro dispositivo Raspberry y funcionará normalmente, haciendo que replicar el proyecto sea relativamente sencillo.

2.3.4. SERVIDOR LAMPP

A continuación, se requiere que el servidor tenga un servidor Apache, que permitirá que la Raspberry pueda funcionar como servidor y que los demás dispositivos conectados a la red puedan ingresar a los aplicativos web instalados en la Raspberry. También se necesita instalar el servidor PHP, para el presente proyecto se trabaja con PHP versión 7 debido a

que es la versión más estable y actual que se encuentra en el mercado. En la sección 1, se detalla que ambos aplicativos se encuentran embebidos en y basta con instalar un servidor LAMPP, se procede con dicha instalación y se comprueba su correcto funcionamiento.

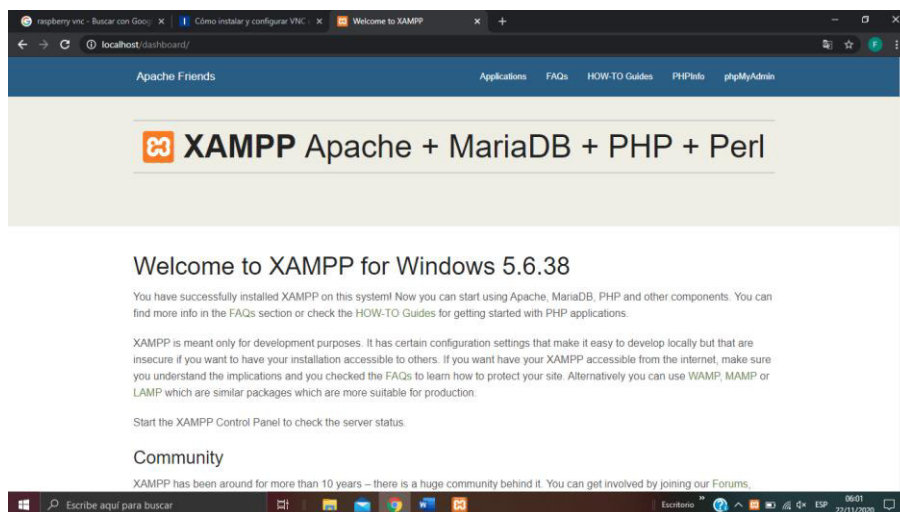


Figura 2.4 Verificación de Raspberry funcionando como servidor [28]

Como se puede observar en la Figura 2.4, desde el navegador, haciendo referencia al localhost se podrá verificar que se encuentre levantado el servidor, de esta manera, para ingresar a aplicativos mediante dispositivos finales como celulares o computadores bastará con colocar la IP del servidor (Raspberry) en el navegador del dispositivo y se podrá acceder al aplicativo web desarrollado.

2.3.5. BASE DE DATOS

En la sección 1 se expone de mejor manera la necesidad y los tipos de bases de datos existentes, si bien es cierto en el paquete LAMPP ya cuenta con una base de datos MySQL, debido a que este tipo de base de datos dejó de tener una versión gratuita a partir del presente año, su utilización se ve restringida a diversas funciones, por tal motivo se instala en el servidor un nuevo gestor de base de datos llamada MariaDB, que si se mantiene con una versión community (gratuita) permite acceder a todo el funcionamiento del gestor sin restricciones, además que el kernel (núcleo) del gestor es basado en SQL lo cual nos garantiza un acceso libre, pero seguro de los datos que se guardarán en nuestra base de datos.

Una vez instalado la base de datos y el gestor, se crea la base de datos, con la estructura mostrada de manera seguida en la Figura 2.5.

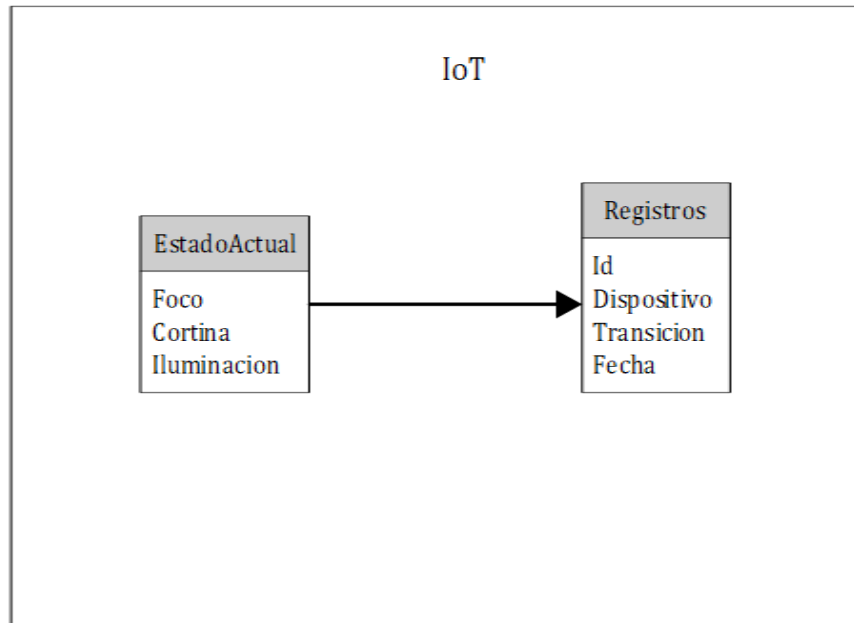


Figura 2.5 Estructura de la base de datos [Fuente: Elaborado por el autor]

Como se observa en la Figura 2.5, la base de datos tiene un nombre de IoT, y cuenta con dos tablas, “EstadoActual” y “Registros”.

La primera tabla “EstadoActual” será utilizada para setear el estado en el que debe colocarse el dispositivo, es decir tiene tres columnas que corresponden a los tres dispositivos que se manejan en el presente prototipo, y tendrá un solo registro que se irá actualizando según las peticiones del usuario, en otras palabras, cada vez que el usuario desee activar o desactivar un dispositivo eléctrico, lo que se hará es modificar el único registro de la columna correspondiente al dispositivo que se desea manipular.

En la segunda tabla “Registros” se almacenará los cambios realizados a los dispositivos, cuenta con un campo “id”, “dispositivo”, “transición” y “fecha” que guardará un nuevo registro cada vez que el usuario ordene un cambio de estado en la primera tabla “EstadoActual”.

Tabla 2.2 Estructura y funcionalidad de tabla "Registros" [Fuente: Elaborado por el autor]

Funcionalidad tabla “Registro”

Id	Registro auto incremental único para almacenar cada fila en la tabla
Dispositivo	Hará referencia a que dispositivo eléctrico (foco, cortina, iluminación) cambió de estado

Transición	Se almacenará la información de que tipo de cambio de estado se realizó, si fue de encendido pasó hacia apagado o viceversa
Fecha	Se almacenará con fecha y hora el cambio realizado

Esta Tabla 2.2 a diferencia de la primera, constará con n registros, iguales a las veces que el usuario ha manipulado el estado actual de los dispositivos eléctricos.

Una vez con la base de datos funcionando y estructurada, es necesario el desarrollo de funciones y medios que accedan a las bases de datos, sea para leer o para modificarlas, este proceso es recomendable realizarlo mediante API (Application Programming Interface).

2.3.6. API

En sí, la Interfaz de Aplicaciones de Programación (API), no es más que un conjunto de definiciones y protocolos que se utilizan para desarrollar e integrar software, siendo su utilización uno de los métodos más seguros para leer y modificar las bases de datos de los aplicativos [35]. Es de esta manera que se puede considerar que un api es el núcleo en conjunto del conocido backend (parte que se encarga de la lógica y es oculta para el usuario en un aplicativo)

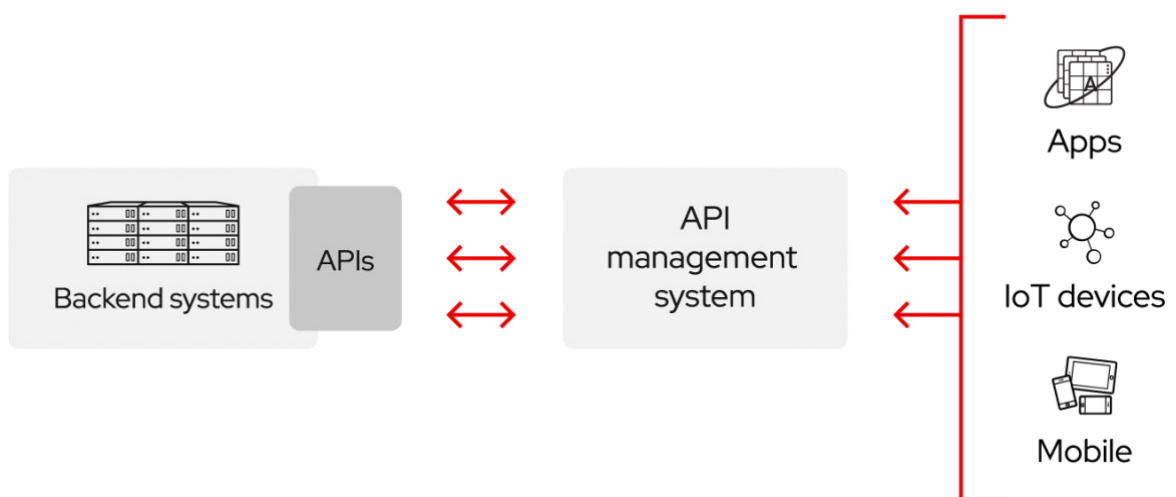


Figura 2.6 Aplicabilidad de API [Fuente: Elaborado por el autor]

El api desarrollado para el presente proyecto son un conjunto de funciones y scripts realizados en PHP que permiten ejecutar lógica de bajo nivel, como se detallaba en el apartado anterior, cada vez que el usuario solicite un cambio de estado de los dispositivos eléctricos, la tabla “EstadoActual” va a cambiar, y se crearan nuevos registros en la tabla “Registros”, este proceso lo realizará el api, para evitar que las credenciales de la base de datos deban ser compartidas con el usuario final, el api se convierte en una especie de mediador entre la base de datos y el usuario final, de esta manera se robustece el sistema domótico y lo hace menos vulnerable a ataques y fallas en su funcionamiento.

En la Figura 2.7 se presenta la estructura utilizada en el api desarrollado para el presente proyecto de titulación.

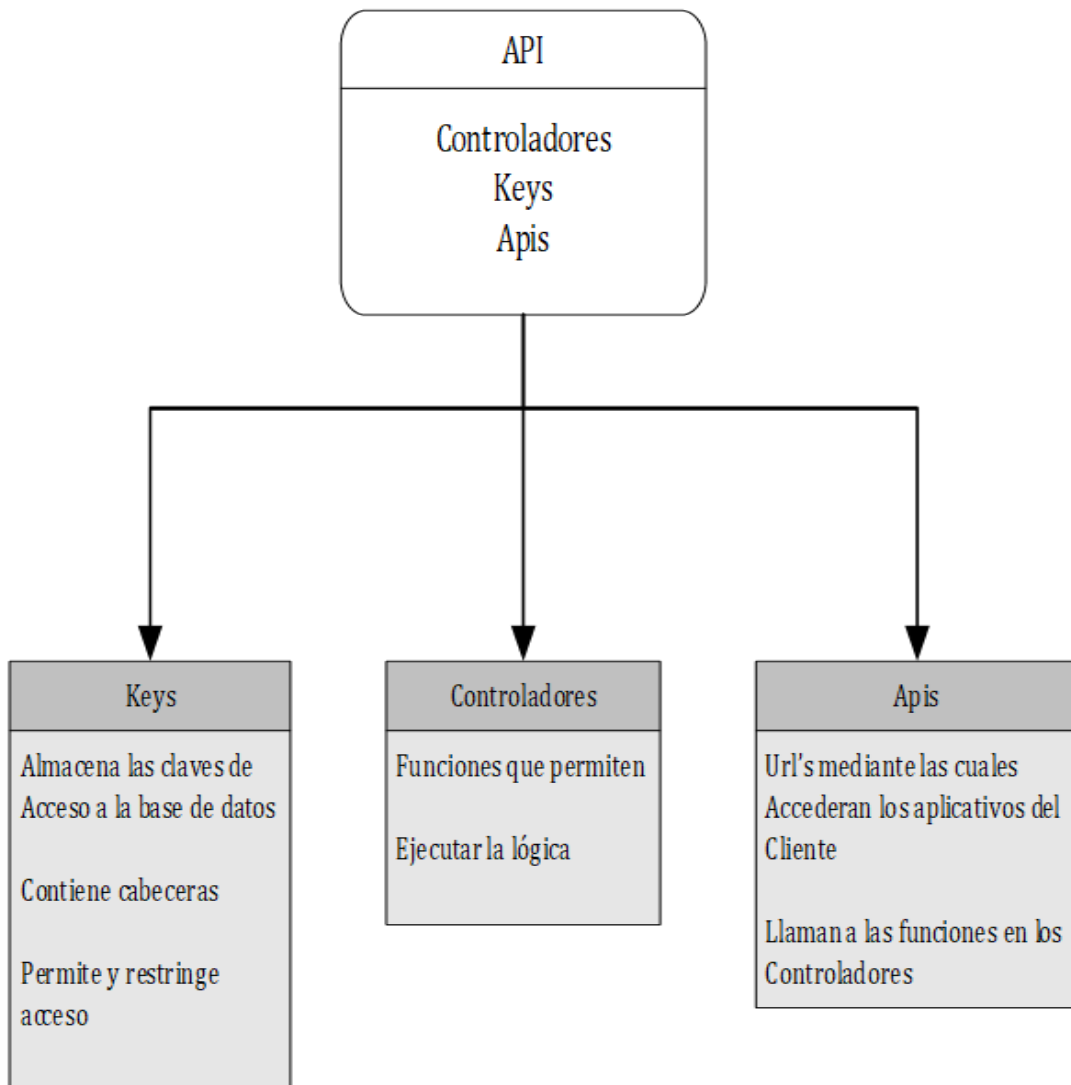


Figura 2.7 Estructura Api [Fuente: Elaborado por el autor]

Como se muestra en la Figura 2.8, el api cuenta esencialmente con tres módulos para lograr su objetivo que son:

- **Keys.** - En este módulo se almacenan las credenciales de acceso, lectura y escritura de la base de datos. Además de las cabeceras que permitirán restringir o permitir el acceso al aplicativo para determinadas IP's.
- **Controladores.** - Consta básicamente de un compendio de funciones que sirven para manipular tanto la base de datos como para crear las secuencias y lógica necesaria para el programa.
- **Apis.** - Es el código en PHP que se ejecutará (llamando a las funciones del controlador, y leyendo las credenciales de Keys) cuando un usuario realice una petición al servidor. También se podría explicar el módulo llamado apis, entendiéndose como las url's, a las cuales la interfaz del usuario va a acceder para solicitar información o escribir la misma en la base de datos.

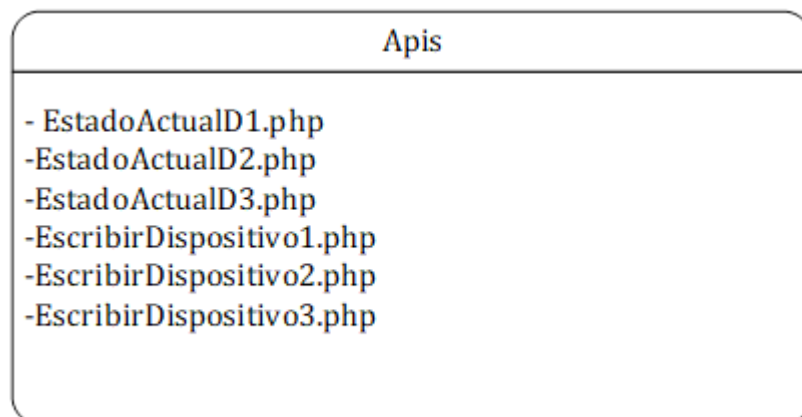


Figura 2.8 Scripts en módulo Apis [Fuente: Elaborado por el autor]

Como se observa en la Figura 2.8, dentro del módulo "Apis" se encuentran programas desarrollados en PHP, que permiten conocer el estado actual de determinado dispositivo eléctrico, y también programas que nos permitirán modificar el estado de dicho elemento.

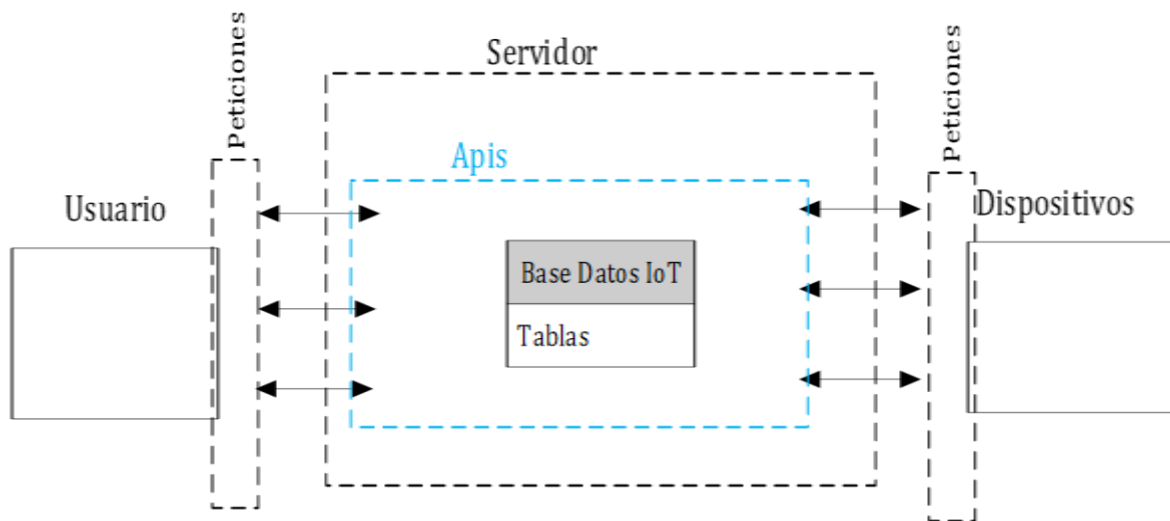


Figura 2.9 Funcionalidad de apis dentro del servidor [Fuente: Elaborado por el autor]

En la Figura 2.9 se aprecia la funcionalidad de las apis, que es en resumen: por parte del usuario se va a realizar peticiones al servidor, peticiones de lectura para mostrar al usuario final como se encuentra el dispositivo consultado (EstadoActualD1.php, EstadoActualD2.php, y EstadoActualD2.php), y también peticiones de escritura (EscribirDispositivo1.php, EscribirDispositivo2.php, y EscribirDispositivo3.php), ya que el usuario decidirá en qué momento desea encender o apagar determinado dispositivo.

Mientras que, de igual manera por el lado de los dispositivos eléctricos, también se comunicarán con el servidor mediante peticiones que responderán las apis, en el caso particular de los dispositivos eléctricos, las apis que se consumirán serán únicamente las de lectura (EstadoActualD1.php, EstadoActualD2.php, y EstadoActualD2.php), debido a que los dispositivos no cuentan con el poder de decisión para encenderse o apagarse, esta decisión la corresponde exclusivamente al usuario final.

2.3.7. DISPOSITIVOS ELÉCTRICOS

Continuando con la explicación del prototipo desarrollado y haciendo alusión a la definición de IoT, que es básicamente dotar de un control inteligente a través de internet a las “cosas” u objetos de uso cotidiano, para la elaboración del proyecto se consideró las necesidades específicas de las personas con movilidad reducida, determinando que los principales objetos a controlar en los que las personas tienen dificultad son las lámparas o focos de 110 V, comunes y corrientes, que se encuentran en la mayoría de los hogares ecuatorianos; el control de la apertura y cierre de las cortinas, ya que por obvias razones a las personas que utilizan sillas de ruedas se les dificulta alcanzar alturas considerables como el caso de las cortinas; y finalmente el control de la iluminación LED.

De manera general se puede decir que el control del dispositivo será realizado mediante un actuador, y la etapa de control previa a la conexión del dispositivo específico será muy similar para todos los dispositivos, esto se presenta en la Figura 2.10.

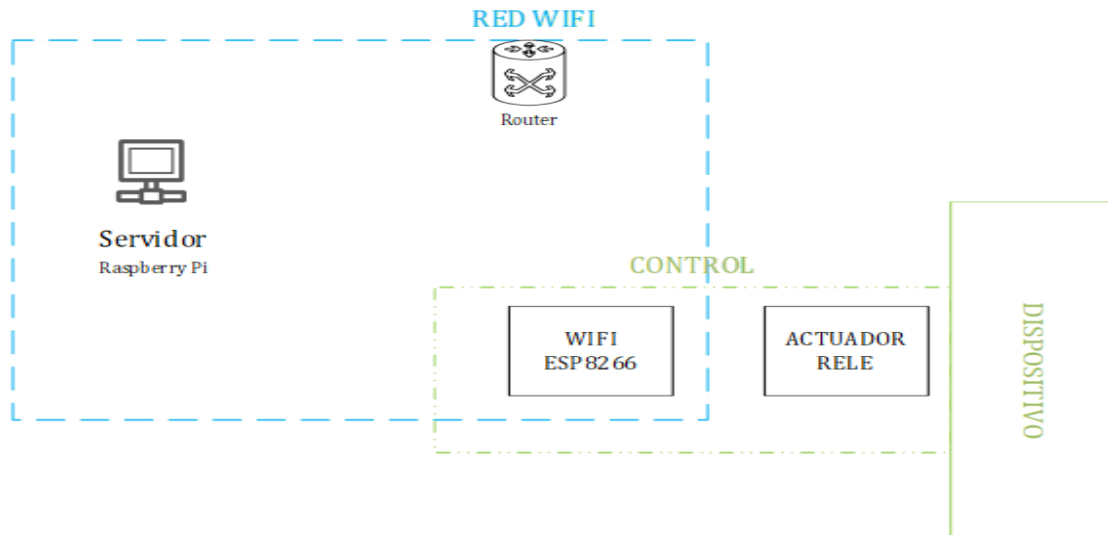


Figura 2.10 Diagrama de bloques de conexión [Fuente: Elaborado por el autor]

En la Figura 2.10 se verifica que la etapa de control hacia el dispositivo eléctrico será llevada a cabo mediante la tarjeta de WiFi ESP8266 cuyas características se detalla en el apartado 1.

En tal virtud se analizará el control ejercido sobre los actuadores por la tarjeta WiFi ESP8266. El software desarrollado para ejercer el control presenta el diagrama de flujo de manera seguida en la Figura 2.11.

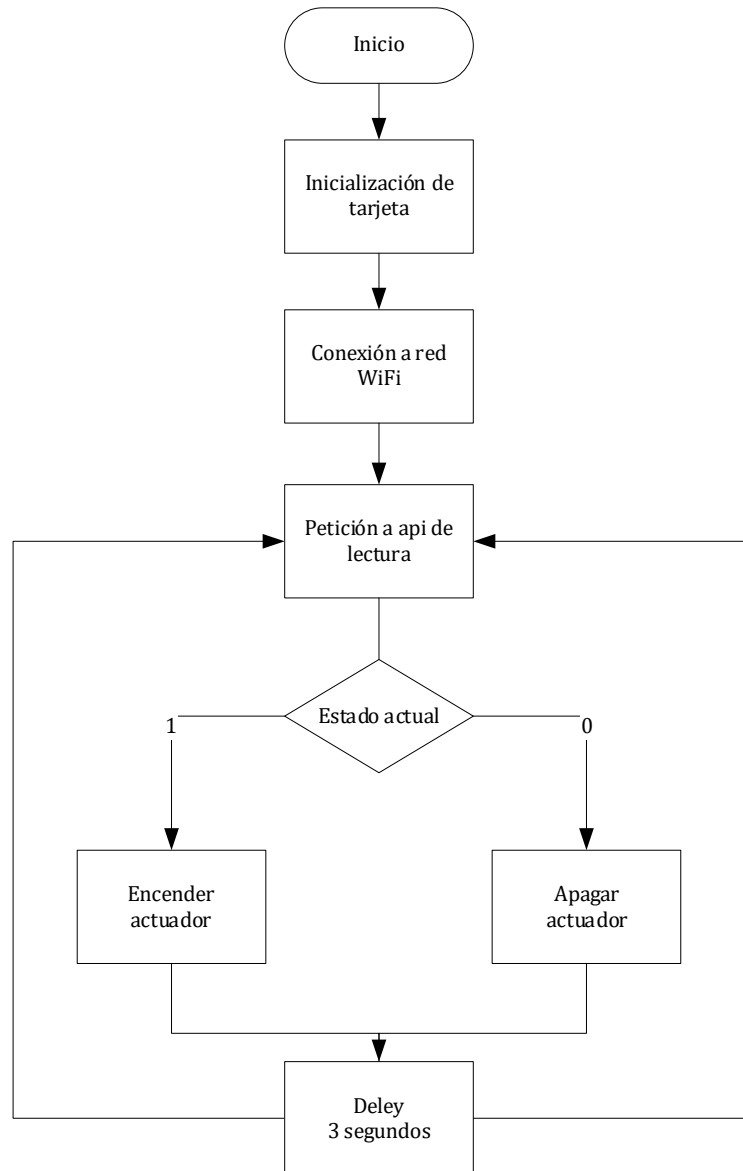


Figura 2.11 Diagrama de flujo del software en ESP8266 [Fuente: Elaborado por el autor]

De manera resumida en la Figura 2.11, se ejemplifica la lógica del programa desarrollado para la tarjeta WiFi ESP8266, donde se observa que el primer paso es inicializar la tarjeta, seteando parámetros como su velocidad de transmisión (115200 baudios), las credenciales para el acceso a la red wifi, y fijando el tipo de cabeceras que el dispositivo va a utilizar para realizar las peticiones al servidor ("Content-Type", "text/plain").

Una vez inicializado y conectado a la red wifi, la tarjeta ESP8266 enviará una petición cada tres segundos al servidor para comprobar el estado en el cual debe estar el dispositivo o actuador que le corresponde controlar, es decir va a leer el api que se encuentra en <http://direccionServidor/api/estadoActualDx.php>, este api como se explicó en el apartado anterior, consultará la base de datos IoT y mostrará el estado en el que el usuario desea

colocar el dispositivo, mostrará un 1 si debe estar encendido, o un 0 si debe estar apagado. Dependiendo de esta respuesta la tarjeta ESP8266 activará o no el actuador. Cabe mencionar que la tarjeta únicamente consulta al api, y no tiene acceso directo a la base de datos lo cual permite una velocidad de ejecución sumamente rápida, y podría estar leyendo la respuesta cada milésima de segundo, pero para el objetivo de control sería un despropósito de uso de recursos, por tal motivo se realiza que la tarjeta ESP8266 lea el estado solo cada 3 segundos, es decir que en el peor de los casos, el usuario tendrá que esperar tres segundos para ver reflejados sus acciones sobre los dispositivos, este tiempo es ínfimo y se lo considera despreciable, por ende se puede determinar como un control en tiempo real.

2.3.8. CONTROL DE LÁMPARA

Dentro del presente proyecto de titulación se controla una lámpara cuya potencia de trabajo es de 100 watts.

El control realizado sobre el dispositivo es de tipo ON/OFF basado en las lecturas obtenidas por la tarjeta de desarrollo ESP8266, cuyo código de ejecución se presenta a continuación:

```
#include <ESP8266HTTPClient.h>
#include <ESP8266WiFi.h>

void setup() {
  pinMode(LED_BUILTIN, OUTPUT);
  pinMode(12, OUTPUT);
  pinMode(13, OUTPUT);
  digitalWrite(12, HIGH);
  digitalWrite(13, HIGH);

  Serial.begin(115200); //Serial
connection

  WiFi.begin("Anthony", "uja86Sasda13"); //WiFi

  while (WiFi.status() != WL_CONNECTED) { //Esperar connection
completion

    delay(500);

    Serial.println("Waiting for connection");
```

```

}

}

void loop() {

  if(WiFi.status() == WL_CONNECTED){

    HTTPClient http;

    http.begin("http://192.168.1.14/api_tesis/api/estadoActual/estadoActualD1.php");
    http.addHeader("Content-Type", "text/plain");

    int httpCode = http.POST("Message from ESP8266");
    String payload = http.getString();

    if (payload.equals("0")){
      Serial.println("encendido");
      digitalWrite(LED_BUILTIN, LOW);
      digitalWrite(12, HIGH);
      digitalWrite(13, HIGH);
    }

    if (payload.equals("3")){
      Serial.println("encendido");
      digitalWrite(LED_BUILTIN, LOW);
      digitalWrite(12, HIGH);
      digitalWrite(13, HIGH);
    }

    if (payload.equals("1")){
      Serial.println("apagado");
      digitalWrite(LED_BUILTIN, HIGH);
    }
  }
}

```

```

digitalWrite(13, HIGH);
digitalWrite(12, LOW);

}
if (payload.equals("2")){
  Serial.println("apagado");
  digitalWrite(LED_BUILTIN, HIGH);

  digitalWrite(12, HIGH);
  digitalWrite(13, LOW);

}

Serial.println(httpCode);
Serial.println(payload);

http.end();

}else{

  Serial.println("Error WiFi connection");

}

delay(3000); //Hacer una petición cada 3 segundos
}

```

2.3.9. CONTROL DE CORTINA

En el presente proyecto de titulación se controla dos cortinas con motor cuya potencia de trabajo es de 50 watts.

El control realizado sobre cada dispositivo es un control ON/OFF basado en las lecturas obtenidas por la tarjeta de desarrollo ESP8266, cuyo código de ejecución se presenta a continuación:

```
#include <ESP8266HTTPClient.h>
#include <ESP8266WiFi.h>

void setup() {
  pinMode(LED_BUILTIN, OUTPUT);
  pinMode(12, OUTPUT);
  pinMode(13, OUTPUT);
  digitalWrite(12, HIGH);
  digitalWrite(13, HIGH);

  Serial.begin(115200); //Serial
connection

  WiFi.begin("Anthony", "uja86Sasda13"); //WiFi

  while (WiFi.status() != WL_CONNECTED) { //Esperar connection
completion

    delay(500);
    Serial.println("Waiting for connection");

  }
}

void loop() {

  if(WiFi.status() == WL_CONNECTED) {

    HTTPClient http;
```

```

http.begin("http://192.168.1.14/api_tesis/api/estadoActual/estadoActualD1.php");
http.addHeader("Content-Type", "text/plain");

int httpCode = http.POST("Message from ESP8266");
String payload = http.getString();

if (payload.equals("0")){
  Serial.println("encendido");
  digitalWrite(LED_BUILTIN, LOW);
  digitalWrite(12, HIGH);
  digitalWrite(13, HIGH);
}
if (payload.equals("3")){
  Serial.println("encendido");
  digitalWrite(LED_BUILTIN, LOW);
  digitalWrite(12, HIGH);
  digitalWrite(13, HIGH);
}
if (payload.equals("0")){
digitalWrite(LED_BUILTIN, HIGH);

digitalWrite(13, HIGH);
digitalWrite(12, HIGH);

}
elseif (payload.equals("3")){
digitalWrite(LED_BUILTIN, HIGH);

digitalWrite(12, HIGH);
digitalWrite(13, HIGH);

}
elseif (payload.equals("1")){

```

```

digitalWrite(LED_BUILTIN, HIGH);

digitalWrite(12, HIGH);
digitalWrite(13, LOW);

}
elseif (payload.equals("2")){
digitalWrite(LED_BUILTIN, HIGH);

digitalWrite(12, LOW);
digitalWrite(13, HIGH);

}
Serial.println(httpCode);
Serial.println(payload);

http.end();

}else{

Serial.println("Error WiFi connection");

}

delay(3000); //Hacer una petición cada 3 segundos
}

```

2.3.10. CONTROL DE ILUMINACIÓN LED

En el presente proyecto de titulación se controla un dispositivo de iluminación led, cuya potencia de trabajo es de 20 watts,

El control realizado sobre el dispositivo es un control ON/OFF basado en las lecturas obtenidas por la tarjeta de desarrollo ESP8266, cuyo código de ejecución se presenta a continuación:


```

#include <ESP8266HTTPClient.h>
#include <ESP8266WiFi.h>

void setup() {
  pinMode(LED_BUILTIN, OUTPUT);
  pinMode(12, OUTPUT);
  pinMode(13, OUTPUT);
  digitalWrite(12, HIGH);
  digitalWrite(13, HIGH);

  Serial.begin(115200); //Serial
connection

  WiFi.begin("Anthony", "uja86Sasdal3"); //WiFi

  while (WiFi.status() != WL_CONNECTED) { //Esperar connection
completion

    delay(500);
    Serial.println("Waiting for connection");
  }
}

void loop() {

  if(WiFi.status() == WL_CONNECTED){

    HTTPClient http;

    http.begin("http://192.168.1.14/api_tesis/api/estadoActual/estadoActua
lD1.php");
    http.addHeader("Content-Type", "text/plain");

    int httpCode = http.POST("Message from ESP8266");

```

```
String payload = http.getString();

if (payload.equals("0")){
    Serial.println("encendido");
    digitalWrite(LED_BUILTIN, LOW);
    digitalWrite(12, HIGH);
    digitalWrite(13, HIGH);
}

if (payload.equals("3")){
    Serial.println("encendido");
    digitalWrite(LED_BUILTIN, LOW);
    digitalWrite(12, HIGH);
    digitalWrite(13, HIGH);
}

if (payload.equals("1")){
    Serial.println("apagado");
    digitalWrite(LED_BUILTIN, HIGH);

    digitalWrite(13, HIGH);
    digitalWrite(12, LOW);

}

if (payload.equals("2")){
    Serial.println("apagado");
    digitalWrite(LED_BUILTIN, HIGH);

    digitalWrite(12, HIGH);
    digitalWrite(13, LOW);

}

Serial.println(httpCode);
Serial.println(payload);
```

```
    http.end();

}else{

    Serial.println("Error WiFi connection");

}

delay(3000); //Hacer una petición cada 3 segundos

}
```

2.3.11. USUARIO

Finalmente se desarrolló el aplicativo web que permitirá el control del usuario, pensando en las necesidades especiales de las personas con movilidad reducida se optó por desarrollar una aplicación sencilla, pero elegante que cumpla con el objetivo de encender, apagar, abrir o cerrar los dispositivos a controlar, se evitó el uso innecesario de imágenes y logotipos para no sobrecargar la pantalla y que el aplicativo cumpla su función de la manera más limpia posible.

Cabe mencionar que el aplicativo web desarrollado se encuentra alojado en el servidor general del proyecto es decir en la Raspberry Pi, sin embargo, como se explicó en apartados anteriores, el único segmento con acceso a la base de datos son los apis, en tal virtud se podría colocar el aplicativo en cualquier servidor externo y continuaría funcionando de la misma manera.

El aplicativo desarrollado es responsive, lo cual quiere decir que se adaptará a cualquier medida de pantalla que disponga el usuario, de tal manera que podrá ser utilizado sin importar que el usuario ingrese desde un celular, una tableta o un computador.

El aplicativo está conformado por varias pantallas, cada una con un controlador distinto para un dispositivo distinto, pero vinculadas entre sí para que la experiencia del usuario sea agradable.

A continuación, en la Figura 2.12 se presenta la interfaz web desarrollada para que el usuario realice el control de los dispositivos.

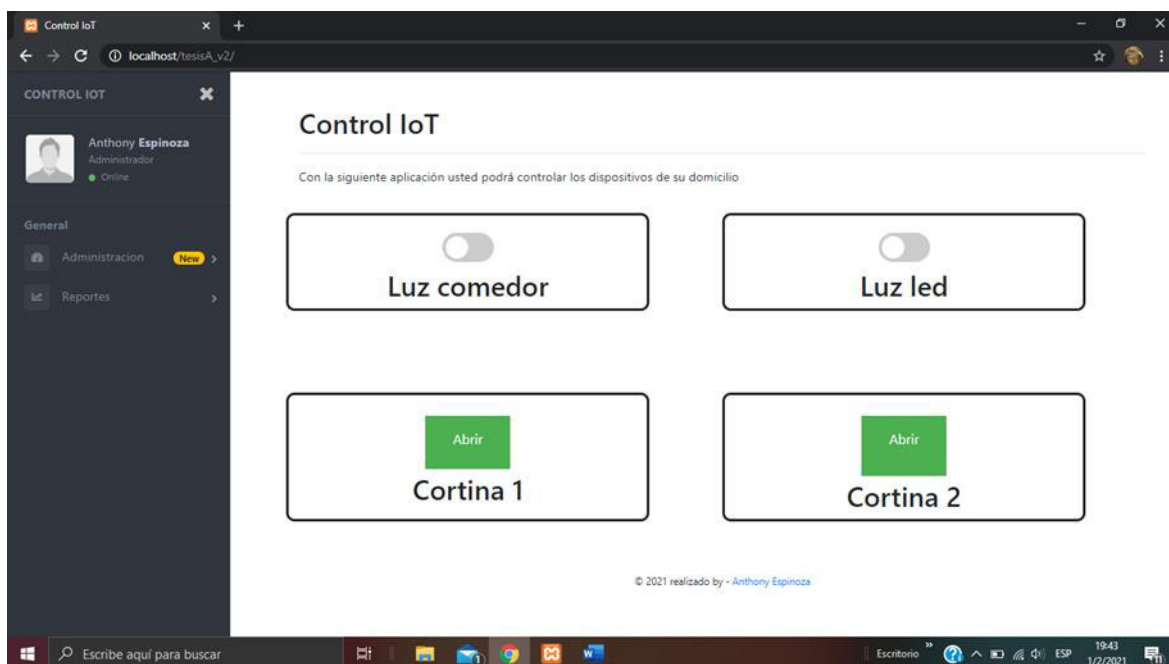


Figura 2.12 Interfaz para control de lámpara [Fuente: Elaborado por el autor]

En el

ANEXO se encuentra el código correspondiente a las interfaces mostradas con anterioridad.

3. RESULTADOS Y DISCUSIÓN

Es necesario considerar varios aspectos para poder hablar de los resultados obtenidos a lo largo del desarrollo del presente proyecto de titulación, debido a que una evaluación de: funciona o no funciona, sería un despropósito, por tal motivo se va a analizar los resultados basándose en parámetros de funcionalidad, escalabilidad, robustez, seguridad, experiencia de usuario, e integraciones futuras.

3.1. FUNCIONALIDAD

El objetivo general del proyecto era el desarrollar un prototipo que basado en IoT permita el control de dispositivos de difícil acceso para personas con capacidades especiales de movilidad, bajo esta sentencia, el sistema desarrollado cumple con todos los parámetros impuestos en el objetivo.

El dispositivo Raspberry Pi se configuró como servidor y encarga de gestionar el control de los dispositivos seleccionados a través de una red, que en este caso particular es una red wifi doméstica, que se enmarca dentro del IoT.

Adicionalmente los dispositivos se enganchan a la red wifi del domicilio, mediante una tarjeta WiFi ESP8266, permitiendo que todo el sistema sea inalámbrico y no requiera de conexiones cableadas.

Finalmente, los usuarios cuentan con un aplicativo web para el control de los dispositivos eléctricos, se realizan las pruebas de funcionamiento donde se verifica el correcto accionar y funcionamiento de los equipos mediante el sistema de prototipo desarrollado.

3.2. ESCALABILIDAD

Si bien es cierto, para el alcance del presente proyecto, las funcionalidades desarrolladas cumplen a satisfacción los requerimientos establecidos, gracias al modelado empleado tanto en software como en hardware, el sistema podría ser llevado a producción e incluso comercializado y/o donado para que mejore la situación de innumerables personas en el mundo. Se dice que para determinar la escalabilidad de un sistema es imprescindible que sea desarrollado en módulos, para que los mismos puedan ser reutilizados cuando la cantidad de usuarios de la aplicación crezca, y es así que el sistema desarrollado si cumple con este requisito, ya que en primera instancia el servidor puede ser cambiado por un servidor de producción, es decir un servidor que se encuentre en el Internet trabajando las 24 horas del día, todos los días del año, bastaría con copiar el código en el servidor profesional y seguiría funcionando de una manera adecuada, teniendo la ventaja que con

el mencionado servidor el sistema ya estaría en la Internet y el control de objetos o “cosas” se lo podría realizar de igual manera en cualquier parte del planeta.

Cabe mencionar que al haber seleccionado como lenguaje de desarrollo PHP, le otorga al sistema una ventaja de escalabilidad, ya que en la actualidad PHP es uno de los lenguajes más utilizados en el planeta, y sería muy difícil que se quede sin soporte sus versiones.

Por otra parte, al no permitir que el aplicativo del usuario acceda directamente a la base de datos, sino mediante el Api, al momento de salir con el sistema a un ambiente de producción o comercial, bastará con cambiar las interfaces por donde acceden los usuarios, más no se deberá cambiar la lógica que ejecuta la escritura ni la lectura de la base de datos, es decir, el mismo api servirá para un ambiente con más usuarios.

3.3. ROBUSTEZ

Aquí se expresa la necesidad de cuantificar el soporte del sistema a múltiples usuarios sin presentar fallos en su ejecución. Por tal motivo se realizan pruebas de conexión al aplicativo web con varios dispositivos de manera simultánea, por razones de disponibilidad de equipos, se probó con 10 equipos utilizando la aplicación a la vez, y se verifica su correcto funcionamiento. Si bien es muy difícil que una persona dentro de una sola casa desee controlar un dispositivo eléctrico desde más de 10 equipos terminales (celulares, tablets, computadores, etc.) en un mismo instante de tiempo, esta prueba ayuda a determinar que es factible que varios equipos terminales consuman la aplicación sin ningún problema.

Por otra parte, al hablar de robustez también se la puede interpretar como la capacidad del sistema en soportar una gran cantidad de peticiones en un periodo corto de tiempo, es así que se realiza una prueba de peticiones mediante el uso del comando ping -t o conocido como ping sostenido, que no es más que realizar peticiones a una determinada página, servidor o aplicativo en intervalos milésimas de segundos

```
Administrador: Símbolo del sistema - ping -t 192.168.1.1
Respuesta desde 192.168.1.1: bytes=32 tiempo=2ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=4ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=3ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=23ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=2ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=3ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=2ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=2ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=3ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=3ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=11ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=2ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=3ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=2ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=4ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=3ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=5ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=2ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=2ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=2ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=3ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=3ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=3ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=42ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=5ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=2ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=2ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=3ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=2ms TTL=64
Respuesta desde 192.168.1.1: bytes=32 tiempo=2ms TTL=64
```

Figura 3.1 Prueba de robustez mediante ping -t [Fuente: Elaborado por el autor]

Como se observa en la Figura 3.1 el sistema no presenta ningún tipo de inconveniente frente a peticiones repetitivas, se lo probó por varios minutos, en distintas horas del día, y su resultado fue el mismo, un sistema robusto que no presentó ningún error en la respuesta de peticiones.

3.4. SEGURIDAD

En la actualidad, hablar de seguridad en los aplicativos es muy relativo pues es de dominio público el aumento en problemas de hacking hacia servidores de empresas mundialmente conocidas y con todo un equipo encargado de velar por la seguridad de sus sistemas, como bancos, entidades gubernamentales y redes sociales, pero lastimosamente no se puede considerar ningún aplicativo 100% seguro, al menos en los niveles que se está trabajando para el presente proyecto. Por tal motivo lo que se ha realizado para aumentar las seguridades del sistema son seguir las buenas prácticas de programación, que entre sus innumerables recomendaciones rescata, no compartir las credenciales de las bases de datos en los aplicativos de los usuarios finales, manejar las bases de datos mediante interfaces de programación (Apis), evitar desborde de memoria y depurar el código que será utilizado por los usuarios finales. Todas estas recomendaciones las cumple el sistema del presente proyecto, por lo cual se puede decir que el uso del aplicativo y la reserva de la información manejada dentro de la misma es segura a un nivel básico pero suficiente para los requerimientos establecidos.

3.5. EXPERIENCIA DE USUARIO

Este apartado se puede considerar uno de los puntos más complejos a ser evaluados en los resultados, debido a que cada usuario puede y va a tener una perspectiva distinta acorde a sus gustos, experiencias previas e incluso emociones, es así que para evaluar los resultados en esta área se realizó una encuesta (Ver ANEXO IV) a personas con capacidades especiales de movilidad que utilizaron el prototipo desarrollado.

Debido a las circunstancias especiales que se viven en todo el mundo debido a la pandemia del COVID-19, encontrar personas con movilidad reducida que permitan el ingreso a sus hogares para una realizar pruebas de experiencia de usuario del prototipo desarrollado ha sido prácticamente imposible, por tal razón se optó en acudir a un hospicio en la ciudad de Quito llamado “Hospice San Camilo” el día sábado 7 de noviembre del 2020 de 9 am a 2 pm, el hospicio se encuentra ubicado en las calles Manuel Zambrano E1-218 & Galo Plaza Lasso, a este lugar concurren varios pacientes en sillas de ruedas a los cuales se les mostró el funcionamiento de prototipo y se les solicitó que interactuaran con el aplicativo para controlar los dispositivos eléctricos, obteniendo los siguientes resultados. Los parámetros a evaluar y sus resultados se muestran a continuación en la Tabla 3.1

Tabla 3.1 Evaluación de experiencia de usuario [Fuente: Realizado por el autor]

Parámetro	Calificación promedio (Sobre 5)
Funcionalidad	5
Contenido	4
Sistema sencillo de utilizar	5
Diseño	3

Con base en las aportaciones recibidas en la encuesta realizada, se plantea la futura integración a dispositivos de acceso como las puertas y dispositivos de entretenimiento como radios y televisores.

4. CONCLUSIONES Y RECOMENDACIONES

4.1. CONCLUSIONES

- Se desarrolló un prototipo de IoT basado en sistemas embebidos como Raspberry Pi para asistir a personas con movilidad reducida en tareas cotidianas de apertura y cierre de cortinas, y encendido y apagado de iluminación.
- El prototipo de IoT se lo realizó con hardware modular, y software libre, que permiten futuras reproducciones y mejoras del actual proyecto.
- El prototipo obtenido con la finalización del presente proyecto de titulación abre un campo de posibilidades para lograr la ayuda a personas con diversos tipos de capacidades especiales.
- Se estudió y analizó las características del módulo ESP8266, las herramientas, procedimientos de configuración y del sistema embebido Raspberry Pi.
- Se diseñó la circuitería de soporte como la de alimentación y fuerza para los dispositivos IoT.
- Se utilizó bases de datos libres y para mejorar la seguridad de la información almacenada se utilizó una API de conexión en el *backend*.
- Se desarrolló e implementó una aplicación web en lenguaje PHP con una interfaz amigable para la comunicación y monitoreo del módulo IoT.
- Se evaluó mediante una encuesta a 5 personas con movilidad reducida sobre el funcionamiento del sistema y su grado de experiencia al usuario con la aplicación web.
- Se desarrolló una interfaz de experiencia amigable con el usuario para lograr la fácil interacción con la persona con capacidades especiales de movilidad.

4.2. RECOMENDACIONES

- Para el trabajo con la tarjeta de desarrollo Raspberry Pi, se recomienda la instalación de software VNC, ya que brinda facilidades de acceso al dispositivo y elimina la conexión de dispositivos periféricos en dicha tarjeta.
- En la tarjeta de desarrollo WIFI ESP8266, se recomienda configurarlo para programar con el IDE de Arduino, caso contrario se lo debería hacer con comandos AT.

- Se recomienda aparte de separar el *backend* con el *frontend*, utilizar librerías especializadas para el manejo de información como PDO de PHP para un correcto resguardo de datos.
- Se debe recordar que en la actualidad la mayoría de personas ingresan a la web por sus dispositivos celulares, por lo cual los aplicativos deberán contar con vistas adecuadas a equipos móviles.

5. REFERENCIAS BIBLIOGRÁFICAS

- [1] Consejo Nacional para la igualdad de las Capacidades, “Total de personas con discapacidad registradas en el registro nacional de Discapacidad,” 2021.
- [2] A. Padilla, “Inclusión educativa de personas con discapacidad,” *rev.colomb.psiqiatr*, nº 4, pp. pp.670-699, 2011.
- [3] A. Padilla, “Discapacidad: contexto, concepto y modelos.,” *Revista Colombiana de Psiquiatria*, nº 16, pp. 381-414, 2010.
- [4] Real Academia Española RAE, “Real Academia Española,” 2014.
- [5] C. V. y. M. E. Bernal, “Institucionalidad y marco legislativo de la discapacidad en el Ecuador.,” 2016.
- [6] C. Ordoñez, “Breve análisis de la inserción laboral de personas con discapacidad en el Ecuador.,” *Alteridad*, vol. 6, nº 1, pp. 145-147, 2011.
- [7] L. Carvajal, “Metodología de la Investigación Científica. Curso general y aplicado,” p. 139, 2006.
- [8] L. F. Herrera Quintero, “Viviendas inteligentes (domótica),” *Ingeniería e investigación*, vol. 25, nº 2, pp. 47-53, 2005.
- [9] D. González Rodríguez, “González, D. R. (2013). Arquitectura y Gestión de la IoT.,” *Telemática*, vol. 12, nº 3, pp. 49-60, 2013.
- [10] E. A. Quiroga Montoya, S. F. Jaramillo Colorado, W. Y. Campo Muñoz y G. E. Chanchí Golondrino, “Propuesta de una arquitectura para agricultura de precisión soportada en IoT,” *RISTI-Revista Ibérica de Sistemas e Tecnologias de Informação*, nº 24, pp. 39-56, 2017.
- [11] M. L. Salcedo Tovar, “ Minicomputador educacional de bajo costo raspberry pi: primera parte.,” *Revista Ethos Venezolana*, vol. 7, nº 1, pp. 28-45, 2015.
- [12] C. Catalán Cantero y A. Blesa Gascón, “Enseñanza de sistemas empuotrados: de Arduino a Raspberry Pi,” de *Actas de las XXII JENUI*, Almería, 2016.

- [13] Omniretro, "Como instalar y configurar RetroPie en Raspberry Pi," 2019.
- [14] E. X. Herrera Acosta, "Estudio, diseño e implementación de un prototipo de Entrenador," *Universidad Tecnológica Israel*, 2014.
- [15] R. Herrador Enríquez, Guía de usuario de Arduino, Universidad de Córdoba, 2009.
- [16] A. Pedrera Caicedo, Arduino para Principiantes: 2ª Edición., IT Campus Academy., 2017.
- [17] J. Ceja, R. Renteria, R. Ruelas y G. Ochoa, "Módulo ESP8266 y sus aplicaciones en el internet de las cosas," *Revista de Ingeniería Eléctrica*, vol. 1, nº 2, pp. 24-36, 2017.
- [18] J. Valderrama y E. Brea, "ESP8266: Un microcontrolador para el Internet de las Cosas," *JIFI 2018 Jornadas de investigación Encuentro académico industrial*, vol. 8, 2018.
- [19] A. M. Muñoz, "Despliegue de una red de sensores basada en chips ESP-8266.," 2020.
- [20] Naylampmechatronics, "USANDO ESP8266 CON EL IDE DE ARDUINO," 2019.
- [21] L. G. Ramírez Corona y G. S. Abarca Jiménez, Sensores y actuadores., Grupo Editorial Patria, 2014.
- [22] J. E. Guarella, J. P. Herdia, L. Rodríguez y I. Bagatto, Sensores y actuadores en motores, Universidad Nacional de la PLata: Buenos Aires, 2011.
- [23] AreaTecnología, "¿Qué es un Relé?," 2013.
- [24] J. L. Torres Capuz y S. X. Chanoluisa Martínez, "Diseño e implementación de un sistema de control en una planta de llenado de tanques a escala usando un transmisor de nivel y algoritmo PID para pruebas en el laboratorio de automatización industrial," *Universidad Politécnica Salesiana sede Guayaquil*, 2015.
- [25] C. Ortiz Santos y J. Torres Martinez, "Estudio comparativo de tres controles digitales para inversores fotovoltaicos trifásicos conectados a la red eléctrica comercial," 2011.
- [26] A. A. Pardo Figueroa, "Diseño y simulación de control avanzado para un sistema de refrigeración aplicado a la industria agroalimentaria," *Universidad de Piura*, 2017.

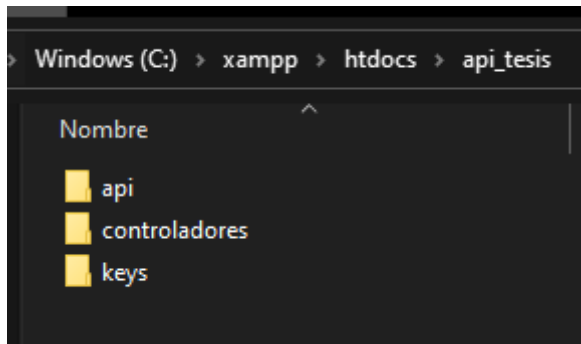
- [27] S. Luján Mora, Programación de aplicaciones web: historia, principios básicos y clientes web, Editorial Club Universitario, 2002.
- [28] S. Borges, "Servidor LAMP," infranetworking, Junio 2020. [En línea]. Available: <https://blog.infranetworking.com/servidor-lamp/>. [Último acceso: Noviembre 2020].
- [29] D. Guevara y L. Quirola Valarezo, "Explotación de vulnerabilidades en un sistema de gestión médica empleando LAMP.," 2019.
- [30] M. Olsson, PHP 7 Quick Scripting Reference, Apress, 2016.
- [31] V. J. Muñoz Eslava, El nuevo PHP. Conceptos avanzados, 2013.
- [32] R. D. Chávez Cabrera, "Análisis del Framework Django para implementar aplicaciones web con base de datos Mariadb y metodología de desarrollo Scrum. Aplicativo: Aplicación web para mantenimiento mecánico en industrias cárnicas para la Empresa Pública Municipal de Faenamamiento," *Universidad Técnica del Norte*, 2016.
- [33] A. Márquez Gómez y M. Rodríguez Couñago, "Aula Virtual de SQL/PSM," *Universidad Complutense de Madrid*, 2019.
- [34] Raspberry Pi, "Raspberry Pi OS," 2019.
- [35] Red Hat, "Qué son las API y para qué sirven," 2016.
- [36] Conectronica, "TR-069 y la gestión de miles de millones de dispositivos," 2012.
- [37] Flacso, "Estudios industriales de la micro, pequeña y mediana empresa," *FlacsoAndes*, 2013.
- [38] "Elaborado por el autor," 2020.
- [39] INEC, "Directorio de Empresas y Establecimientos," p. 15, 2019.
- [40] USRobotics, "Configuración TR-069," 2007.
- [41] Pandorafms, "Administración de Configuración de Servidores (ACS), en profundidad," 2018.

ANEXOS

ANEXO A

Desarrollo de programación en Api

- Estructura del Api



- Controladores:

```
<?php
//Abrir conexion a la base de datos
function connect($db)
{
    try {
        $conn = new PDO("mysql:host={$db['host']};dbname={$db['db']}", $db
['username'], $db['password']);

        // set the PDO error mode to exception
        $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);

        return $conn;
    } catch (PDOException $exception) {
        exit($exception->getMessage());
    }
}
```

```

    }
}

function estadoActualD1($dbConn){
    $sql = $dbConn-
>prepare("SELECT lampara from estadoActual where id = 0");

    $sql->execute();
    $a = json_encode($sql->fetchAll(PDO::FETCH_ASSOC));

    return $a;
}

function estadoActualD2($dbConn){
    $sql = $dbConn-
>prepare("SELECT led from estadoActual where id = 0");

    $sql->execute();
    $a = json_encode($sql->fetchAll(PDO::FETCH_ASSOC));

    return $a;
}

function estadoActualD3($dbConn){
    $sql = $dbConn-
>prepare("SELECT cortina1 from estadoActual where id = 0");

```

```

        $sql->execute();

        $a = json_encode($sql->fetchAll(PDO::FETCH_ASSOC));

        return $a;
    }

    function estadoActualD4($dbConn){
        $sql = $dbConn-
>prepare("SELECT cortina2 from estadoActual where id = 0");

        $sql->execute();

        $a = json_encode($sql->fetchAll(PDO::FETCH_ASSOC));

        return $a;
    }

    function cambiarEstado($dbConn, $dispositivo, $estado){
        $fecha = date('Y-m-d H:i:s');

        $sql = $dbConn-
>prepare("INSERT INTO registro (dispositivo, fecha, estado) VALUES (:dispos
itivo, :fecha, :estado)");

        $sql->bindValue(':dispositivo', $dispositivo);

        $sql->bindValue(':fecha', $fecha);

        $sql->bindValue(':estado', $estado);

        $a= $sql->execute();

        return $a;
    }

```



```

}

function actualizarEstadoD1($dbConn, $estado){

    $sql = $dbConn-
>prepare("UPDATE estadoActual SET lampara= :estado WHERE id = 0");

    $sql->bindValue(':estado', $estado);

    $a = $sql->execute();

    return $a;
}

function actualizarEstadoD2($dbConn, $estado){

    $sql = $dbConn-
>prepare("UPDATE estadoActual SET led = :estado WHERE id = 0");

    $sql->bindValue(':estado', $estado);

    $a = $sql->execute();

    return $a;
}

function actualizarEstadoD3($dbConn, $estado){

    $sql = $dbConn-
>prepare("UPDATE estadoActual SET cortina1 = :estado WHERE id = 0");

    $sql->bindValue(':estado', $estado);

    $a = $sql->execute();
}

```

```

    return $a;
}

function actualizarEstadoD4($dbConn, $estado){

    $sql = $dbConn-
>prepare("UPDATE estadoActual SET cortina2= :estado WHERE id = 0");

    $sql->bindValue(':estado', $estado);

    $a = $sql->execute();

    return $a;
}
?>

```

- Api de lectura de estado actual

```

<?php

include '../..//keys/config.php';
include "../..//controladores/controladores.php";

$dbConn = connect($db);

$resp = estadoActualD4($dbConn);
$resp1 = json_decode($resp);
echo($resp1[0]->cortina2);

?>

```

- Api de escritura

```
<?php
include '../..../keys/config.php';
include "../..../controladores/controladores.php";

$dbConn = connect($db);
$dispositivo= $_GET['id'];

// Aqui leo el actual estado, para determinar a que cambiar
if ($dispositivo == '1') {
    $auxEstado = estadoActualD1($dbConn);
    $resp1 = json_decode($auxEstado);
    $estadoActual= $resp1[0]->lampara;
}elseif ($dispositivo == '2') {
    $auxEstado = estadoActualD2($dbConn);
    $resp1 = json_decode($auxEstado);
    $estadoActual= $resp1[0]->led;
}elseif ($dispositivo == '3') {
    $auxEstado = estadoActualD3($dbConn);
    $resp1 = json_decode($auxEstado);
    $estadoActual= $resp1[0]->cortina1;
}elseif ($dispositivo == '4') {
    $auxEstado = estadoActualD4($dbConn);
    $resp1 = json_decode($auxEstado);
    $estadoActual= $resp1[0]->cortina2;
}

// Dependiendo el estado y el dispositivo cambio
```

```

if ($dispositivo == '1' || $dispositivo == '2') {
    if ($estadoActual== '0') {
        $estado = '1';
    }else{
        $estado = '0';
    }
}

}elseif ($dispositivo == '3' || $dispositivo == '4') {
    if ($estadoActual== '0') {
        $estado = '1';
    }elseif ($estadoActual== '1') {
        $estado = '3';
    }elseif ($estadoActual== '3') {
        $estado = '2';
    }elseif ($estadoActual== '2') {
        $estado = '0';
    }
}
}

// Actualizo la tabla estadoActual
if ($dispositivo == '1') {
    actualizarEstadoD1($dbConn, $estado);
}elseif ($dispositivo == '2') {
    actualizarEstadoD2($dbConn, $estado);
}elseif ($dispositivo == '3') {
    actualizarEstadoD3($dbConn, $estado);
}elseif ($dispositivo == '4') {
    actualizarEstadoD4($dbConn, $estado);
}
}

```

```
// echo($estadoActual);  
echo($estado);  
$resp = cambiarEstado($dbConn, $dispositivo, $estado);  
  
?>
```

ANEXO B

Software en la tarjeta WiFi ESP8266

```
#include <ESP8266HTTPClient.h>
#include <ESP8266WiFi.h>

void setup() {
    pinMode(LED_BUILTIN, OUTPUT);
    pinMode(12, OUTPUT);
    pinMode(13, OUTPUT);
    digitalWrite(12, HIGH);
    digitalWrite(13, HIGH);

    Serial.begin(115200); //Serial
connection

    WiFi.begin("Anthony", "uja86Sasda13"); //WiFi

    while (WiFi.status() != WL_CONNECTED) { //Esperar connection
completion

        delay(500);
        Serial.println("Waiting for connection");

    }

}

void loop() {

    if(WiFi.status() == WL_CONNECTED){
```

```

HTTPClient http;

http.begin("http://192.168.1.14/api_tesis/api/estadoActual/estadoActua
lD1.php");
http.addHeader("Content-Type", "text/plain");

int httpCode = http.POST("Message from ESP8266");
String payload = http.getString();

if (payload.equals("0")){
    Serial.println("encendido");
    digitalWrite(LED_BUILTIN, LOW);
    digitalWrite(12, HIGH);
    digitalWrite(13, HIGH);
}
if (payload.equals("3")){
    Serial.println("encendido");
    digitalWrite(LED_BUILTIN, LOW);
    digitalWrite(12, HIGH);
    digitalWrite(13, HIGH);
}
if (payload.equals("1")){
    Serial.println("apagado");
    digitalWrite(LED_BUILTIN, HIGH);

    digitalWrite(13, HIGH);
    digitalWrite(12, LOW);

}
if (payload.equals("2")){
    Serial.println("apagado");
    digitalWrite(LED_BUILTIN, HIGH);
}

```

```
digitalWrite(12, HIGH);  
digitalWrite(13, LOW);  
  
}  
  
Serial.println(httpCode);  
Serial.println(payload);  
  
http.end();  
  
}else{  
  
    Serial.println("Error WiFi connection");  
  
}  
  
delay(3000); //Hacer una petición cada 3 segundos  
  
}
```


ANEXO C

Aplicación web – Control IoT – index.php

```
<?php
$estado1="abierto";
?>

<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <meta name="description" content="Responsive sidebar template with sliding effect and dropdown menu based on bootstrap 3">
  <title>Control IoT</title>
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css" integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm"
    crossorigin="anonymous">
  <link href="https://use.fontawesome.com/releases/v5.0.6/css/all.css" rel="stylesheet">
  <link href="//maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css" rel="stylesheet" id="bootstrap-css">
  <script src="//maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js">
  </script>
  <script src="//cdnjs.cloudflare.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
  <script src="logica.js"></script>
  <link rel="stylesheet" href="estilos.css">

</head>
```

```

<body>
<div class="page-wrapper chiller-theme toggled">
  <a id="show-sidebar" class="btn btn-sm btn-dark" href="#">
    <i class="fas fa-bars"></i>
  </a>
  <nav id="sidebar" class="sidebar-wrapper">
    <div class="sidebar-content">
      <div class="sidebar-brand">
        <a href="#">Control IoT</a>
        <div id="close-sidebar">
          <i class="fas fa-times"></i>
        </div>
      </div>
      <div class="sidebar-header">
        <div class="user-pic">
          
        </div>
        <div class="user-info">
          <span class="user-name">Anthony
            <strong>Espinoza</strong>
          </span>
          <span class="user-role">Administrador</span>
          <span class="user-status">
            <i class="fa fa-circle"></i>
            <span>Online</span>
          </span>
        </div>
      </div>
    </div>
  </div>

```

```

<!-- sidebar-search -->
<div class="sidebar-menu">
  <ul>
    <li class="header-menu">
      <span>General</span>
    </li>
    <li class="sidebar-dropdown">
      <a href="#">
        <i class="fa fa-tachometer-alt"></i>
        <span>Administracion</span>
        <span class="badge badge-pill badge-warning">New</span>
      </a>
      <div class="sidebar-submenu">
        <ul>
          <li>
            <a href="#">Dashboard 1
              <span class="badge badge-pill badge-success">Pro</span>
            </a>
          </li>
          <li>
            <a href="#">Dashboard 2</a>
          </li>
          <li>
            <a href="#">Dashboard 3</a>
          </li>
        </ul>
      </div>
    </li>
  </ul>
</div>

```

```

<li class="sidebar-dropdown">
  <a href="#">
    <i class="fa fa-chart-line"></i>
    <span>Reportes</span>
  </a>
  <div class="sidebar-submenu">
    <ul>
      <li>
        <a href="#">Tiempo de encendido</a>
      </li>
      <li>
        <a href="#">Reportes Energéticos</a>
      </li>
    </ul>
  </div>
</li>

</ul>
</div>
<!-- sidebar-menu -->
</div>
<!-- sidebar-content -->
<!-- <div class="sidebar-footer">
  <a href="#">
    <i class="fa fa-bell"></i>
    <span class="badge badge-pill badge-warning notification">3</span>

```

```

</a>
<a href="#">
  <i class="fa fa-envelope"></i>
  <span class="badge badge-pill badge-success notification">7</span>
</a>
<a href="#">
  <i class="fa fa-cog"></i>
  <span class="badge-sonar"></span>
</a>
<a href="#">
  <i class="fa fa-power-off"></i>
</a>
</div> -->
</nav>
<!-- sidebar-wrapper -->
<main class="page-content">
  <div class="container-fluid">
    <h2>Control IoT</h2>
    <hr>
    <div class="row">
      <div class="form-group col-md-12">
        <p>Con la siguiente aplicación usted podrá controlar los dispositi
vos de su domicilio</p>
      </div>

      <div class="form-group col-md-5 luz-comedor">

        <!-- Rounded switch -->
        <label class="switch">
          <input type="checkbox" id="switch">

```

```

        <span class="slider round"></span>
    </label>
    <br>
    <h2>Luz comedor</h2>
</div>
<div class="form-group col-md-1"></div>
<div class="form-group col-md-5 luz-led">
    <!-- Rounded switch -->
    <label class="switch">
        <input type="checkbox" id="switch">
        <span class="slider round"></span>
    </label>
    <br>
    <h2>Luz led</h2>
</div>
<div class="form-group col-md-1"></div>
<div class="form-group col-md-12">
    <br><br><br>
</div>
<div class="form-group col-md-5 cortina1">
    <button class="button button1" id="btnCortina1"> <label id="cortina1
Label">Abrir</label> </button>
    <h2>Cortina 1</h2>
</div>
<div class="form-group col-md-1"></div>

```

```

    <div class="form-group col-md-5 cortina2">

        <button class="button button2" id="btnCortina2"><p id="cortina2Label
">Abrir</p> </button>

        <h2>Cortina 2</h2>

    </div>

    <div class="form-group col-md-1"></div>

</div>

<footer class="text-center">
    <br> <br>
    <div class="mb-2">
        <small>
            © 2021 realizado by - <a target="_blank" rel="noopener noreferer" href="https://azouaoui.netlify.com">
                Anthony Espinoza
            </a>
        </small>
    </div>

</footer>

</div>
</main>
<!-- page-content -->
</div>
<!-- page-wrapper -->

    <script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>

```

```

    <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd
/popper.min.js" integrity="sha384-
ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusvfa0b4Q"
        crossorigin="anonymous"></script>

    <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstra
p.min.js" integrity="sha384-
JZR6Spejh4U02d8j0t6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyjUar5+76PVCmY1"
        crossorigin="anonymous"></script>

</body>

</html>

```

- Hoja de estilos utilizados para el aplicativos – estilos.css

```

@keyframes swing {
  0% {
    transform: rotate(0deg);
  }
  10% {
    transform: rotate(10deg);
  }
  30% {
    transform: rotate(0deg);
  }
  40% {
    transform: rotate(-10deg);
  }
  50% {
    transform: rotate(0deg);
  }
  60% {

```



```

    transform: rotate(5deg);
  }
  70% {
    transform: rotate(0deg);
  }
  80% {
    transform: rotate(-5deg);
  }
  100% {
    transform: rotate(0deg);
  }
}

@keyframes sonar {
  0% {
    transform: scale(0.9);
    opacity: 1;
  }
  100% {
    transform: scale(2);
    opacity: 0;
  }
}

body {
  font-size: 0.9rem;
}

.page-wrapper .sidebar-wrapper,
.sidebar-wrapper .sidebar-brand > a,
.sidebar-wrapper .sidebar-dropdown > a:after,
.sidebar-wrapper .sidebar-menu .sidebar-dropdown .sidebar-
submenu li a:before,

```

```

.sidebar-wrapper ul li a i,
.page-wrapper .page-content,
.sidebar-wrapper .sidebar-search input.search-menu,
.sidebar-wrapper .sidebar-search .input-group-text,
.sidebar-wrapper .sidebar-menu ul li a,
#show-sidebar,
#close-sidebar {
  -webkit-transition: all 0.3s ease;
  -moz-transition: all 0.3s ease;
  -ms-transition: all 0.3s ease;
  -o-transition: all 0.3s ease;
  transition: all 0.3s ease;
}

/*-----page-wrapper-----*/

.page-wrapper {
  height: 100vh;
}

.page-wrapper .theme {
  width: 40px;
  height: 40px;
  display: inline-block;
  border-radius: 4px;
  margin: 2px;
}

.page-wrapper .theme.chiller-theme {
  background: #1e2229;
}

```

```

}

/*-----toggel sidebar-----*/

.page-wrapper.toggled .sidebar-wrapper {
  left: 0px;
}

@media screen and (min-width: 768px) {
  .page-wrapper.toggled .page-content {
    padding-left: 300px;
  }
}

/*-----show sidebar button-----*/

#show-sidebar {
  position: fixed;
  left: 0;
  top: 10px;
  border-radius: 0 4px 4px 0px;
  width: 35px;
  transition-delay: 0.3s;
}

.page-wrapper.toggled #show-sidebar {
  left: -40px;
}

/*-----sidebar-wrapper-----*/

.sidebar-wrapper {
  width: 260px;
  height: 100%;
}

```

```
max-height: 100%;
position: fixed;
top: 0;
left: -300px;
z-index: 999;
}

.sidebar-wrapper ul {
  list-style-type: none;
  padding: 0;
  margin: 0;
}

.sidebar-wrapper a {
  text-decoration: none;
}

/*-----sidebar-content-----*/

.sidebar-content {
  max-height: calc(100% - 30px);
  height: calc(100% - 30px);
  overflow-y: auto;
  position: relative;
}

.sidebar-content.desktop {
  overflow-y: hidden;
}
```

```

/*-----sidebar-brand-----*/

.sidebar-wrapper .sidebar-brand {
  padding: 10px 20px;
  display: flex;
  align-items: center;
}

.sidebar-wrapper .sidebar-brand > a {
  text-transform: uppercase;
  font-weight: bold;
  flex-grow: 1;
}

.sidebar-wrapper .sidebar-brand #close-sidebar {
  cursor: pointer;
  font-size: 20px;
}

/*-----sidebar-header-----*/

.sidebar-wrapper .sidebar-header {
  padding: 20px;
  overflow: hidden;
}

.sidebar-wrapper .sidebar-header .user-pic {
  float: left;
  width: 60px;
  padding: 2px;
  border-radius: 12px;
}

```

```
margin-right: 15px;
overflow: hidden;
}

.sidebar-wrapper .sidebar-header .user-pic img {
  object-fit: cover;
  height: 100%;
  width: 100%;
}

.sidebar-wrapper .sidebar-header .user-info {
  float: left;
}

.sidebar-wrapper .sidebar-header .user-info > span {
  display: block;
}

.sidebar-wrapper .sidebar-header .user-info .user-role {
  font-size: 12px;
}

.sidebar-wrapper .sidebar-header .user-info .user-status {
  font-size: 11px;
  margin-top: 4px;
}

.sidebar-wrapper .sidebar-header .user-info .user-status i {
  font-size: 8px;
  margin-right: 4px;
}
```

```
    color: #5cb85c;
}

/*-----sidebar-search-----*/

.sidebar-wrapper .sidebar-search > div {
    padding: 10px 20px;
}

/*-----sidebar-menu-----*/

.sidebar-wrapper .sidebar-menu {
    padding-bottom: 10px;
}

.sidebar-wrapper .sidebar-menu .header-menu span {
    font-weight: bold;
    font-size: 14px;
    padding: 15px 20px 5px 20px;
    display: inline-block;
}

.sidebar-wrapper .sidebar-menu ul li a {
    display: inline-block;
    width: 100%;
    text-decoration: none;
    position: relative;
    padding: 8px 30px 8px 20px;
}
```

```

.sidebar-wrapper .sidebar-menu ul li a i {
  margin-right: 10px;
  font-size: 12px;
  width: 30px;
  height: 30px;
  line-height: 30px;
  text-align: center;
  border-radius: 4px;
}

.sidebar-wrapper .sidebar-menu ul li a:hover > i::before {
  display: inline-block;
  animation: swing ease-in-out 0.5s 1 alternate;
}

.sidebar-wrapper .sidebar-menu .sidebar-dropdown > a:after {
  font-family: "Font Awesome 5 Free";
  font-weight: 900;
  content: "\f105";
  font-style: normal;
  display: inline-block;
  font-style: normal;
  font-variant: normal;
  text-rendering: auto;
  -webkit-font-smoothing: antialiased;
  -moz-osx-font-smoothing: grayscale;
  text-align: center;
  background: 0 0;
  position: absolute;
  right: 15px;
}

```



```

top: 14px;
}

.sidebar-wrapper .sidebar-menu .sidebar-dropdown .sidebar-submenu ul {
padding: 5px 0;
}

.sidebar-wrapper .sidebar-menu .sidebar-dropdown .sidebar-submenu li {
padding-left: 25px;
font-size: 13px;
}

.sidebar-wrapper .sidebar-menu .sidebar-dropdown .sidebar-
submenu li a:before {
content: "\f111";
font-family: "Font Awesome 5 Free";
font-weight: 400;
font-style: normal;
display: inline-block;
text-align: center;
text-decoration: none;
-webkit-font-smoothing: antialiased;
-moz-osx-font-smoothing: grayscale;
margin-right: 10px;
font-size: 8px;
}

.sidebar-wrapper .sidebar-menu ul li a span.label,
.sidebar-wrapper .sidebar-menu ul li a span.badge {
float: right;
margin-top: 8px;
}

```

```

margin-left: 5px;
}

.sidebar-wrapper .sidebar-menu .sidebar-dropdown .sidebar-
submenu li a .badge,
.sidebar-wrapper .sidebar-menu .sidebar-dropdown .sidebar-
submenu li a .label {
  float: right;
  margin-top: 0px;
}

.sidebar-wrapper .sidebar-menu .sidebar-submenu {
  display: none;
}

.sidebar-wrapper .sidebar-menu .sidebar-dropdown.active > a:after {
  transform: rotate(90deg);
  right: 17px;
}

/*-----side-footer-----*/

.sidebar-footer {
  position: absolute;
  width: 100%;
  bottom: 0;
  display: flex;
}

.sidebar-footer > a {
  flex-grow: 1;
}

```

```
text-align: center;
height: 30px;
line-height: 30px;
position: relative;
}

.sidebar-footer > a .notification {
  position: absolute;
  top: 0;
}

.badge-sonar {
  display: inline-block;
  background: #980303;
  border-radius: 50%;
  height: 8px;
  width: 8px;
  position: absolute;
  top: 0;
}

.badge-sonar:after {
  content: "";
  position: absolute;
  top: 0;
  left: 0;
  border: 2px solid #980303;
  opacity: 0;
  border-radius: 50%;
  width: 100%;
```

```

height: 100%;
animation: sonar 1.5s infinite;
}

/*-----page-content-----*/

.page-wrapper .page-content {
  display: inline-block;
  width: 100%;
  padding-left: 0px;
  padding-top: 20px;
}

.page-wrapper .page-content > div {
  padding: 20px 40px;
}

.page-wrapper .page-content {
  overflow-x: hidden;
}

/*-----scroll bar-----*/

::-webkit-scrollbar {
  width: 5px;
  height: 7px;
}

::-webkit-scrollbar-button {
  width: 0px;
  height: 0px;
}

```

```

}
::-webkit-scrollbar-thumb {
  background: #525965;
  border: 0px none #ffffff;
  border-radius: 0px;
}
::-webkit-scrollbar-thumb:hover {
  background: #525965;
}
::-webkit-scrollbar-thumb:active {
  background: #525965;
}
::-webkit-scrollbar-track {
  background: transparent;
  border: 0px none #ffffff;
  border-radius: 50px;
}
::-webkit-scrollbar-track:hover {
  background: transparent;
}
::-webkit-scrollbar-track:active {
  background: transparent;
}
::-webkit-scrollbar-corner {
  background: transparent;
}

/*-----chiller-theme-----*/
-----*/

.chiller-theme .sidebar-wrapper {

```

```

background: #31353D;
}

.chiller-theme .sidebar-wrapper .sidebar-header,
.chiller-theme .sidebar-wrapper .sidebar-search,
.chiller-theme .sidebar-wrapper .sidebar-menu {
  border-top: 1px solid #3a3f48;
}

.chiller-theme .sidebar-wrapper .sidebar-search input.search-menu,
.chiller-theme .sidebar-wrapper .sidebar-search .input-group-text {
  border-color: transparent;
  box-shadow: none;
}

.chiller-theme .sidebar-wrapper .sidebar-header .user-info .user-role,
.chiller-theme .sidebar-wrapper .sidebar-header .user-info .user-status,
.chiller-theme .sidebar-wrapper .sidebar-search input.search-menu,
.chiller-theme .sidebar-wrapper .sidebar-search .input-group-text,
.chiller-theme .sidebar-wrapper .sidebar-brand>a,
.chiller-theme .sidebar-wrapper .sidebar-menu ul li a,
.chiller-theme .sidebar-footer>a {
  color: #818896;
}

.chiller-theme .sidebar-wrapper .sidebar-menu ul li:hover>a,
.chiller-theme .sidebar-wrapper .sidebar-menu .sidebar-dropdown.active>a,
.chiller-theme .sidebar-wrapper .sidebar-header .user-info,
.chiller-theme .sidebar-wrapper .sidebar-brand>a:hover,
.chiller-theme .sidebar-footer>a:hover i {

```

```

    color: #b8bfce;
}

.page-wrapper.chiller-theme.toggled #close-sidebar {
    color: #bdbdbd;
}

.page-wrapper.chiller-theme.toggled #close-sidebar:hover {
    color: #ffffff;
}

.chiller-theme .sidebar-wrapper ul li:hover a i,
.chiller-theme .sidebar-wrapper .sidebar-dropdown .sidebar-
submenu li a:hover:before,
.chiller-theme .sidebar-wrapper .sidebar-search input.search-
menu:focus+span,
.chiller-theme .sidebar-wrapper .sidebar-menu .sidebar-
dropdown.active a i {
    color: #16c7ff;
    text-shadow:0px 0px 10px rgba(22, 199, 255, 0.5);
}

.chiller-theme .sidebar-wrapper .sidebar-menu ul li a i,
.chiller-theme .sidebar-wrapper .sidebar-menu .sidebar-dropdown div,
.chiller-theme .sidebar-wrapper .sidebar-search input.search-menu,
.chiller-theme .sidebar-wrapper .sidebar-search .input-group-text {
    background: #3a3f48;
}

.chiller-theme .sidebar-wrapper .sidebar-menu .header-menu span {
    color: #6c7b88;
}

```

```

}

.chiller-theme .sidebar-footer {
    background: #3a3f48;
    box-shadow: 0px -1px 5px #282c33;
    border-top: 1px solid #464a52;
}

.chiller-theme .sidebar-footer>a:first-child {
    border-left: none;
}

.chiller-theme .sidebar-footer>a:last-child {
    border-right: none;
}

/* The switch - the box around the slider */
.switch {
    position: relative;
    display: inline-block;
    width: 60px;
    height: 34px;
}

/* Hide default HTML checkbox */
.switch input {
    opacity: 0;
    width: 0;
    height: 0;
}

```



```
/* The slider */
.slider {
  position: absolute;
  cursor: pointer;
  top: 0;
  left: 0;
  right: 0;
  bottom: 0;
  background-color: #ccc;
  -webkit-transition: .4s;
  transition: .4s;
}

.slider:before {
  position: absolute;
  content: "";
  height: 26px;
  width: 26px;
  left: 4px;
  bottom: 4px;
  background-color: white;
  -webkit-transition: .4s;
  transition: .4s;
}

input:checked + .slider {
  background-color: #2196F3;
}

input:focus + .slider {
```

```

    box-shadow: 0 0 1px #2196F3;
}

input:checked + .slider:before {
    -webkit-transform: translateX(26px);
    -ms-transform: translateX(26px);
    transform: translateX(26px);
}

/* Rounded sliders */
.slider.round {
    border-radius: 34px;
}

.slider.round:before {
    border-radius: 50%;
}

.luz-comedor{
    border-style: solid;
    border-radius: 10px;
    text-align: center;
    padding-top: 20px;
}

.luz-led{
    border-style: solid;
    border-radius: 10px;
    text-align: center;
    padding-top: 20px;
}

```

```

.cortina1{
  border-style: solid;
  border-radius: 10px;
  text-align: center;
  padding-top: 20px;
}

.cortina2{
  border-style: solid;
  border-radius: 10px;
  text-align: center;
  padding-top: 20px;
}

.button {
  background-color: #4CAF50; /* Green */
  border: none;
  color: white;
  padding: 15px 32px;
  text-align: center;
  text-decoration: none;
  display: inline-block;
  font-size: 16px;
  margin: 4px 2px;
  cursor: pointer;
  -webkit-transition-duration: 0.4s; /* Safari */
  transition-duration: 0.4s;
}

.button2:hover {
  box-
shadow: 0 12px 16px 0 rgba(0,0,0,0.24),0 17px 50px 0 rgba(0,0,0,0.19);
}

```

```

}

.button1:hover {
    box-
shadow: 0 12px 16px 0 rgba(0,0,0,0.24),0 17px 50px 0 rgba(0,0,0,0.19);
}

```

- Comandos realizados en javascript para lograr dinamismo en la aplicación – logica.js

```

jQuery(function ($) {

    var tiempo = 6000;

    $(".sidebar-dropdown > a").click(function() {
    $(".sidebar-submenu").slideUp(200);
    if (
        $(this)
            .parent()
            .hasClass("active")
    ) {
        $(".sidebar-dropdown").removeClass("active");
        $(this)
            .parent()
            .removeClass("active");
    } else {
        $(".sidebar-dropdown").removeClass("active");
        $(this)
            .next(".sidebar-submenu")
            .slideDown(200);
        $(this)
            .parent()
            .addClass("active");
    }
    }
    });

```

```

}
});

$("#close-sidebar").click(function() {
$(".page-wrapper").removeClass("toggled");
});
$("#show-sidebar").click(function() {
$(".page-wrapper").addClass("toggled");
});

$("#switch").click(function() {
$.get("http://localhost/api_tesis/api/cambiarEstado/cambiarEstado.php?id=1", function(data, status){
    // alert("Data: " + data + "\nStatus: " + status);
});

});

$("#btnCortina1").click(function() {

$.get("http://localhost/api_tesis/api/cambiarEstado/cambiarEstado.php?id=3", function(data, status){

    $("#btnCortina1").attr('disabled', 'disabled');
    document.getElementById('btnCortina1').style.background='#989898';
    document.getElementById('cortina1Label').innerHTML='Activando...'
});
});

```

```

        setTimeout(sample, tiempo);

    });

    function sample() {

        $.get("http://localhost/api_tesis/api/cambiarEstado/cambiarEstado.php?id
=3", function(data, status){

            console.log(data);
            $("#btnCortina1").attr('disabled', false);
            document.getElementById('btnCortina1').style.background='#4CAF50
';

            if(data=='3' || data=='1'){
                document.getElementById('cortina1Label').innerHTML='Abrir';
            }

            if(data=='0' || data=='2'){
                document.getElementById('cortina1Label').innerHTML='Cerrar';
            }

        });
    }

    $("#btnCortina2").click(function() {

```

```

$.get("http://localhost/api_tesis/api/cambiarEstado/cambiarEstado.php?id=4", function(data, status){

    $("#btnCortina2").attr('disabled', 'disabled');
    document.getElementById('btnCortina2').style.background='#989898';
    document.getElementById('cortina2Label').innerHTML='Activando...';

});

setTimeout(sample1, tiempo);

});

function sample1() {

    $.get("http://localhost/api_tesis/api/cambiarEstado/cambiarEstado.php?id=4", function(data, status){

        console.log(data);
        $("#btnCortina2").attr('disabled', false);
        document.getElementById('btnCortina2').style.background='#4CAF50';
        if(data=='3' || data=='1'){
            document.getElementById('cortina2Label').innerHTML='Abrir';
        }
        if(data=='0' || data=='2'){
            document.getElementById('cortina2Label').innerHTML='Cerrar';
        }
    }
}

```

```
});
```

```
}
```

```
});
```


ANEXO D

Encuesta realizada a personas con movilidad reducida que probaron el sistema

Nombre:

Edad:

1.- Que tipo de discapacidad motriz tiene usted?

Total

Parcial

Temporal

Definitiva

2.- Usted tiene algún cuidador a tiempo completo? ¿Sea familiar o persona aparte que lo acompañe la mayoría del tiempo?

Si

No

2.- Considera usted que sistema que probó le ayudaría en el desarrollo de sus actividades cotidianas?

No

Poco

Medio

Bastante

3.- Considera usted que la aplicación que permite el control del sistema es amigable con el usuario y sus necesidades?

Si

No

4.- Cuanto estaría dispuesto a pagar por el sistema para tenerlo instalado en su casa?

- \$1 a \$50 dólares
- \$51 a \$100 dólares
- \$101 a \$200 dólares
- \$201 a \$400 dólares
- más de \$400 dólares

5.- Existe alguna otra funcionalidad que le gustaría que esté integrado en el sistema?

- No
- Si y cuál _____

ORDEN DE EMPASTADO