

ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE FORMACIÓN DE TECNÓLOGOS

IMPLEMENTACIÓN DE UN SISTEMA DE CONTROL DOMÓTICO ORIENTADO A PERSONAS CON DISCAPACIDAD MOTRIZ

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

Lema Coronel Pablo Miguel

pablo.lema@epn.edu.ec

DIRECTOR: ING. CATALINA ELIZABETH ARMAS FREIRE, MSC

elizabeth.armas@epn.edu.ec

CODIRECTOR: ING. ROBIN GERARDO ÁLVAREZ RUEDA, PHD

robin.alvarez@epn.edu.ec

Quito, julio 2022

CERTIFICACIÓN

Certificamos que el presente trabajo fue desarrollado por el Sr. Lema Coronel Pablo Miguel como requerimiento parcial a la obtención del título de TECNÓLOGO EN ELECTRÓNICA Y TELECOMUNICACIONES, bajo nuestra supervisión:



Ing. Catalina Elizabeth Armas Freire, MSC

DIRECTORA DEL PROYECTO



Ing. Robin Gerardo Álvarez Rueda, PHD

CODIRECTOR DEL PROYECTO

DECLARACIÓN

Yo, Lema Coronel Pablo Miguel con CI: 172364337-3, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y que he consultado las referencias bibliográficas que se incluyen en este documento.

Sin perjuicio de los derechos reconocidos en el primer párrafo del artículo 144 del Código Orgánico de la Economía Social de los Conocimientos, Creatividad e Innovación – COESC-, soy titular de la obra en mención y otorgo una licencia gratuita, intransferible y no exclusiva de uso con fines académicos a la Escuela Politécnica Nacional.

Entrego toda la información técnica pertinente, en caso de que hubiese una explotación comercial de la obra por parte de la EPN, se negociará los porcentajes de los beneficios conforme lo establece la normativa nacional vigente.



Pablo Miguel Lema Coronel

CI: 172364337-3

Teléfono: 0998243487

Correo: pablo.lesa@epn.edu.ec

DEDICATORIA

Primero a Dios y a mi madre Rosario que nunca dejó de creer en mí y a mi padre Miguel Lema, a mi hermanos mayores Vero y Fher, Gaby y Edy, mis hermanos menores Juan y Nico, Jimmy y Anita, y a los pequeños de la casa Daniel, Damaris, Ruth, a mis sobrinos Nando, Mathy, Antho, la pequeña Sary.

AGRADECIMIENTO

A Ing. Ely que con su paciencia supo alentarme a la culminación de este proyecto. A los docentes de la EPN que con sus conocimientos aportaron a mi formación profesional, en especial a Ing. Brito, Ing Pablo Proaño, Ing. Fanny Flores e Ing. Alan Cuenca de quienes he aprendido mucho no solo para mi vida profesional sino también para la vida. También un agradecimiento especial a Stefy Plúa que cada semestre me ayudo en los trámites de manera cordial y con carisma, a Ing. Mary Rosero, a mis amigos de toda la vida, Michu, Tia Mary, Joss, Jairo y a Daya que ha sido ese sustento y apoyo incondicional.

ÍNDICE DE CONTENIDOS

1	Introducción.....	1
1.1	Objetivo general.....	2
1.2	Objetivos específicos.....	2
1.3	Fundamentos.....	2
2	Metodología.....	7
3	Resultados y discusión.....	8
3.1	Tecnologías para adquisición de ondas cerebrales.....	9
3.2	Algoritmos de funcionamiento del sistema.....	12
3.3	Niveles de atención e identificación de parpadeos para el control <i>on/off</i>	16
3.4	Sistema de control domótico basado en los niveles de atención e identificación de parpadeos.....	27
3.5	Pruebas de funcionamiento del sistema de control domótico implementado	34
3.6	Manual de funcionamiento.....	40
4	Conclusiones y recomendaciones.....	42
4.1	Conclusiones.....	42
4.2	Recomendaciones.....	43
5	Referencias bibliográficas.....	44
	ANEXOS.....	46
	Anexo 1: certificado de funcionamiento.....	47
	Anexo 2: código para crear la librería <i>brain.h</i>	49
	Anexo 3: código en Python para identificación de parpadeos.....	51
	Anexo 4: código de programación del Arduino maestro.....	53
	Anexo 5: código de programación del Arduino esclavo.....	55
	Anexo 6: código de bloques de estado de cada botón de la app.....	57
	Anexo 7: código para asignar una dirección IP estática a un módulo Esp-01.....	59
	Anexo 8: código para asignar IP estática a Esp32 Cam.....	61

ÍNDICE DE FIGURAS

Figura 1.1 Estructura del sistema nervioso	3
Figura 1.2 Ondas cerebrales en electroencefalograma	3
Figura 1.3 Modelo funcional BCI	5
Figura 1.4 Entorno de App Inventor	5
Figura 1.5 Interfaz gráfica de App diseñada	6
Figura 3.1 Etapas del sistema de control	8
Figura 3.2 Método invasivo	10
Figura 3.3 Método no invasivo	10
Figura 3.4 Partes Mindwave Mobile	12
Figura 3.5 Primera etapa del sistema	12
Figura 3.6 ThinkGear ASIC Module	13
Figura 3.7 Diagrama de flujo de librería Brain.h	14
Figura 3.8 Diagrama de flujo de algoritmo para identificar parpadeo	15
Figura 3.9 Diagrama de flujo de la comunicación entre Arduino maestro y Arduino esclavo	16
Figura 3.10 Esquema Arduino maestro	17
Figura 3.11 Esquema Arduino esclavo	18
Figura 3.12 Esquema Arduino maestro- Arduino Esclavo	18
Figura 3.13 Diagrama de flujo de App diseñada	19
Figura 3.14 Barra de estado	20
Figura 3.15 Receptor de datos recibiendo datos	20
Figura 3.16 Receptor de datos modo reposo	20
Figura 3.17 Diagrama de flujo del estado de conexión	21
Figura 3.18 Botón menú editar	21
Figura 3.19 Modo edición activado	22
Figura 3.20 Ingreso de caracteres alfanuméricos	22
Figura 3.21 Barra de botones principales	23
Figura 3.22 Botón Luces activado	23
Figura 3.23 Botón Puerta activado	23
Figura 3.24 Botón Ambiente activado	24
Figura 3.25 Visor de cámara	27
Figura 3.26 Interruptor On/off de diadema Mindwave Mobile	28
Figura 3.27 Clip presionando el lóbulo de la oreja	28
Figura 3.28 Contacto con la piel y electrodo	28

Figura 3.29 LED rojo indica que no hay conexión	29
Figura 3.30 Indicador de conexión sin recepción de datos	29
Figura 3.31 Indicador de recepción de datos.....	30
Figura 3.32 Comunicación serial Maestro-Esclavo	30
Figura 3.33 Línea de código Python para imprimir carácter.....	31
Figura 3.34 Línea de código Arduino para recibir caracter desde Python	31
Figura 3.35 Led indicador de parpadeo	31
Figura 3.36 Diagrama de flujo para enviar dato a App	32
Figura 3.37 Barra de estado de conexión	32
Figura 3.38 Propiedades del router.....	33
Figura 3.39 Conexión a la red WiFi.....	33
Figura 3.40 Líneas de código para conectarse a red WiFi.....	34
Figura 3.41 Verificación de encendido de luminaria, modo manual.....	34
Figura 3.42 Diagrama de la comunicación del sistema	35
Figura 3.43 Representación física de la instalación de un punto inalámbrico	35
Figura 3.44 Prueba de funcionamiento de iluminación 1	37
Figura 3.45 Prueba de funcionamiento de iluminación 2	37
Figura 3.46 Prueba de funcionamiento de iluminación 3	38
Figura 3.47 Nivel de atención respecto al tiempo de respuesta	39
Figura 3.49 Código QR de manual de funcionamiento del Sistema implementado	41

ÍNDICE DE TABLAS

Tabla 1.1 Tipos de ondas cerebrales	4
Tabla 1.2 Características del módulo Bluetooth HC-05	6
Tabla 3.1 Características del sistema	9
Tabla 3.2 Principales medidores de ondas cerebrales	11
Tabla 3.3 Valores de datos del chip ThinkGear	13
Tabla 3.4 Características del módulo ESP-01	24
Tabla 3.5 Direcciones IP asignadas a las luminarias	25
Tabla 3.6 Dirección IP estática asignada a chapa eléctrica	26
Tabla 3.7 Direcciones IP estáticas asignadas a las cargas	26
Tabla 3.8 Características del módulo ESP-32 Cam	27
Tabla 3.9 Pruebas de comunicación entre diadema y Arduino maestro	36
Tabla 3.10 Pruebas de comunicación entre ordenador y Arduino esclavo	36
Tabla 3.11 Prueba de funcionamiento de iluminación	38
Tabla 3.12 Datos en relación a la edad y tiempo para alcanzar un nivel de atención ..	39
Tabla 3.13 Pruebas de funcionamiento	40

RESUMEN

El presente documento se basa en el diseño, implementación y pruebas de funcionamiento de un sistema de control domótico orientado a personas con discapacidad motriz, mediante niveles de atención y parpadeos.

El primer capítulo detalla la problemática encontrada en las personas con discapacidad motriz al momento de desenvolverse en el hogar y realizar actividades como encender o apagar luces, atender a la puerta o controlar dispositivos que requieran desplazamiento físico de la persona. Se plantea una solución a través de tecnologías inalámbricas, tecnologías de adquisición de señales cerebrales y detección de parpadeos. Además, se detalla los fundamentos teóricos que complementan el contenido de este trabajo.

El segundo capítulo presenta la metodología aplicada en el proyecto, donde se utilizó los métodos: exploratorio, analítico y aplicado, porque la implementación del sistema se realizó mediante el uso de tecnologías existentes.

El tercer capítulo detalla los resultados y discusión; iniciando con un estudio de tecnologías y algoritmos para adquisición de señales cerebrales e identificación de parpadeos, seguido por el diseño de una App y la implementación del sistema. Finalmente, se muestran las pruebas realizadas para determinar el funcionamiento total del sistema.

En la cuarta y quinta sección se muestra las conclusiones y recomendaciones obtenidas tras la implementación del sistema, y la bibliografía que sustenta el marco teórico.

En la sexta sección se presenta los anexos necesarios con la documentación para la implementación del sistema como: códigos de programación para Arduino, Python y App Inventor.

PALABRAS CLAVE: domótica, *Mindwave Mobile*, red LAN.

ABSTRACT

This document is based on the design, implementation and functional tests of a home automation control system aimed at people with motor disabilities, through levels of attention and blinks.

The first chapter details the problems found in people with motor disabilities at the time of getting around at home and carrying out activities such as turning lights on or off, answering the door or controlling devices that require physical movement of the person. A solution is proposed through wireless technologies, brain signal acquisition technologies and blink detection. In addition, the theoretical foundations that complement the content of this work are detailed.

The second chapter presents the methodology applied in the project, where the methods were used: exploratory, analytical and applied, because the implementation of the system was carried out through the use of existing technologies.

The third chapter details the results and discussion; starting with a study of technologies and algorithms for the acquisition of brain signals and identification of blinks, followed by the design of an App and the implementation of the system. Finally, the tests carried out to determine the total operation of the system are shown.

The fourth and fifth section shows the conclusions and recommendations obtained after the implementation of the system, and the bibliography that supports the theoretical framework.

In the sixth section, the necessary annexes are presented with the documentation for the implementation of the system, such as: programming codes for Arduino, Python and App Inventor.

KEYWORDS: *home automation, Mindwave Mobile, LAN network.*

1 Introducción

En su informe mundial sobre la discapacidad, la Organización Mundial de la Salud indica que en todo el mundo viven más de mil millones de personas con alguna forma de discapacidad; de ellas, doscientos millones experimentan dificultades considerables en su funcionamiento motriz o mental. En una proyección a futuro, la discapacidad será un preocupante mayor, pues su prevalencia está en aumento. Esto se debe a que la población está envejeciendo y el riesgo de discapacidad es mayor entre los adultos mayores, así también el aumento de enfermedades como la diabetes, cáncer, enfermedades cardiovasculares y trastornos de salud mental [1].

Alrededor del mundo, las personas con algún tipo de discapacidad tienen una menor participación económica, tasas de pobreza más altas respecto a las personas sin discapacidad y menos oportunidades académicas o laborales. En parte, esto es consecuencia de las limitaciones que entorpecen al acceso de las personas con discapacidad a varios servicios que se consideran obvios, como la educación, salud, transporte, empleo o información. La visión, es un mundo inclusivo donde todos puedan vivir con salud, felicidad y dignidad.

En el Ecuador, el Consejo Nacional para Igualdad de Discapacidades (CONADIS) registra en su página oficial que hay un total de 481 392 personas con discapacidad motriz, de las cuales 226 356 personas tienen algún tipo de discapacidad física. Respecto a la ciudad de Quito hay 5 337 personas con el 75% al 100% de discapacidad físico-motriz, que por su condición requieren de terceras personas para su desempeño diario [2].

Para las personas con limitaciones motrices, sea en extremidades superiores, inferiores o ambas, resulta difícil desenvolverse en el hogar y tener control sobre dispositivos en casa y realizar actividades como encender o apagar luces, atender a la puerta o controlar cualquier dispositivo que requiera desplazamiento, lo que genera la dependencia directa de un tercero [2]. Muchas familias que tienen miembros con algún tipo de limitación física no cuentan con la estabilidad económica o recursos extra para contratar personal capacitado que cuide de la persona con discapacidad motriz o a su vez, que algún familiar disponga el tiempo para prestar los cuidados que ésta pueda requerir, lo que da lugar a que permanezcan sin supervisión y exponiendo totalmente su integridad.

Expuestas estas necesidades, se empatizó con estos grupos vulnerables y crear un sistema de automatización mediante niveles de atención, parpadeos y una aplicación móvil.

En el Anexo 1, se adjunta el certificado de funcionamiento, el cual avala que este proyecto se encuentra operando de forma correcta.

1.1 Objetivo general

Implementar un sistema de control domótico orientado a personas con discapacidad motriz.

1.2 Objetivos específicos

- Estudiar las tecnologías para adquisición de ondas cerebrales.
- Establecer los algoritmos de funcionamiento del sistema.
- Aplicar los niveles de atención e identificación de parpadeos para el control *on/off*, orientado a personas con discapacidad.
- Implementar el sistema de control domótico basado en los niveles de atención e identificación de parpadeos.
- Desarrollar pruebas de funcionamiento del sistema de control domótico implementado.

1.3 Fundamentos

Sistema nervioso

Es un sistema complejo que se encarga de dirigir, monitorear y controlar todas las funciones y actividades de los órganos y el cuerpo en su conjunto. La función de este sistema es combinar el movimiento con los estímulos del cuerpo. De esta manera, los humanos y otros animales controlan sus emociones, actitudes y acciones [3].

El sistema nervioso, como se muestra en la Figura 1.1, está dividido en dos partes: el Sistema Nervioso Periférico (SNP), constituido por los nervios craneales y espinales, y órganos sensoriales; y el Sistema Nervioso Central (SNC), que está formado por el encéfalo y la médula espinal.

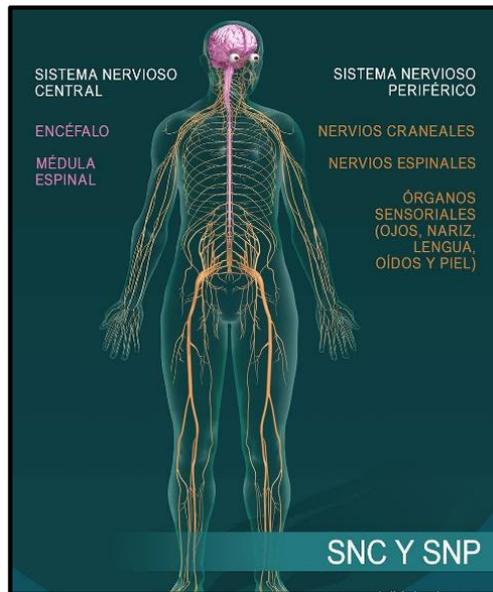


Figura 1.1 Estructura del sistema nervioso [3]

El sistema nervioso tiene subunidades estructurales conocidas como neuronas; las mismas que se comunican entre sí por medio de pequeños impulsos eléctricos llamados ondas cerebrales, que son visibles cuando se las particionan con un filtro. Para ello, se necesita crear un registro de electroencefalograma (EEG) de la actividad eléctrica en el cerebro, como se observa en la Figura 1.2, utilizando sensores (electrodos) conectados al cuero cabelludo para ver los impulsos eléctricos del cerebro en modo de onda [4].

La frecuencia de estas ondas se las mide en Hertz (Hz) o ciclos por segundo.

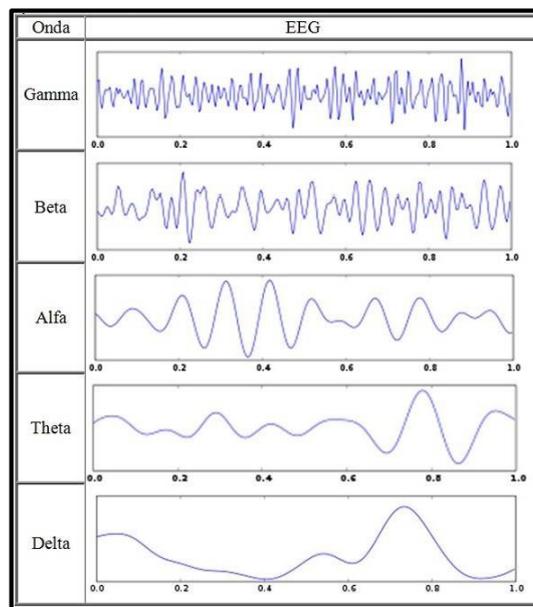


Figura 1.2 Ondas cerebrales en electroencefalograma [4]

Tipos de ondas cerebrales [5]

El cerebro tiene activo cinco tipos de ondas cerebrales. Dependiendo que actividad se realice a cada instante, las ondas muestran mayor actividad en diferente área del cerebro y otras con menor intensidad en diferentes zonas.

En la Tabla 1.1, se encuentra una sinopsis de las frecuencias que tienden a generarse en el cerebro humano.

Tabla 1.1 Tipos de ondas cerebrales [5]

Onda cerebral	Característica
Gamma [40-100 (Hz)]	Aparecen por situación de alto nivel de estrés o bien, cuando partimos de ondas alfa, ante estados de máxima concentración. Actividad sincronizada y superior de muchos grupos neuronales
Beta [15-18 (Hz)]	En el extremo más bajo corresponden a estados de acción, trabajo y concentración. Mente alerta, enfocada en el mundo externo
Alfa [8-13 (Hz)]	Aparecen cuando nos sentimos relajados, la atención se desconecta del exterior y se conecta con el interior. Mente tranquila y clara
Theta [3-7 (Hz)]	Del sueño superficial al sueño profundo. Pueden ser buenas o malas. Reactividad emocional, cognitiva
Delta [0-2 (Hz)]	Son de gran amplitud y suceden cuando se está dormido sin soñar. Pleno rendimiento de partes más profundas del cerebro

Interfaz cerebro computador (BCI)

Es una tecnología basada en la adquisición de ondas cerebrales, que posteriormente serán procesadas por un ordenador donde se aprovecharán sus características; consecutivamente se controlará algún dispositivo mediante comandos [6]. En la Figura 1.3, se observa el principio de funcionamiento básico de las interfaces cerebro ordenador.

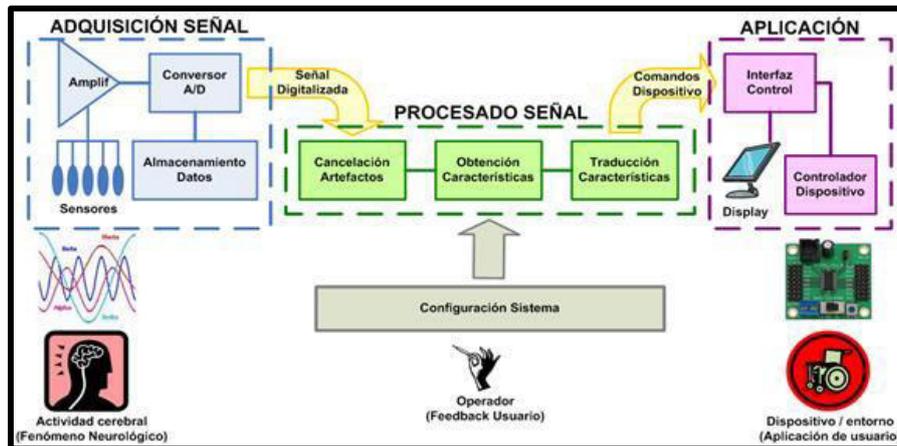


Figura 1.3 Modelo funcional BCI [6]

Atención: es un proceso fundamental y esencial para la ejecución de cualquier acción, es decir, permite al individuo orientarse hacia cualquier estímulo y procesarlo, para responder en consecuencia [7].

MIT App Inventor

Es un entorno de desarrollo de software creado por Google para crear aplicaciones para el sistema operativo *Android*. Este lenguaje es gratuito codificado por bloques y está disponible en línea, donde las aplicaciones creadas con Inventor cumplen algunos requisitos básicos en dispositivos móviles, pero se limitan fácilmente [8].

Dentro del entorno de App Inventor se trabaja en modo Diseño y en modo Bloques. La Figura 1.4, muestra el entorno Modo Diseñador.

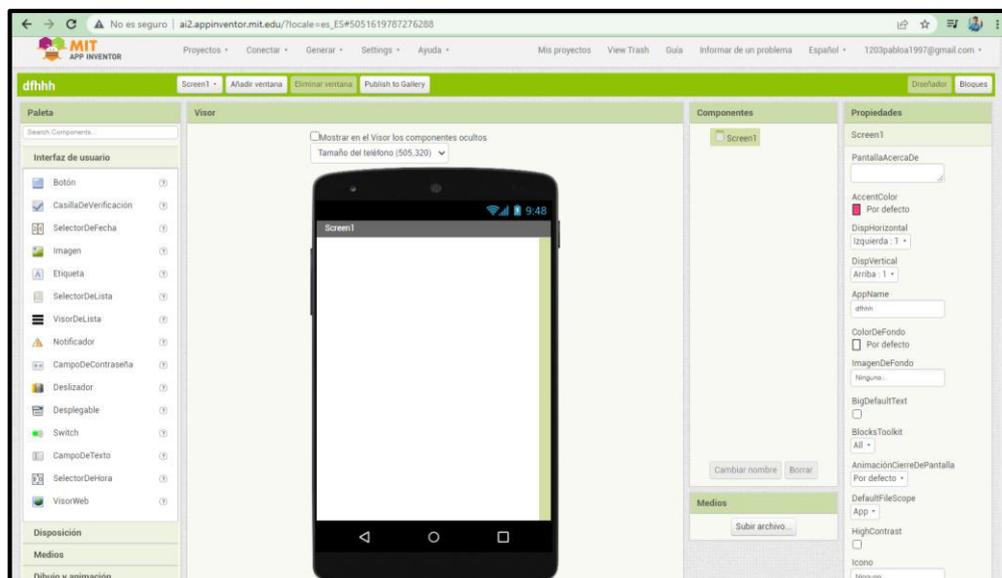


Figura 1.4 Entorno de *App Inventor*

La Figura 1.5, muestra el entorno de App Inventor y el desarrollo de la App.

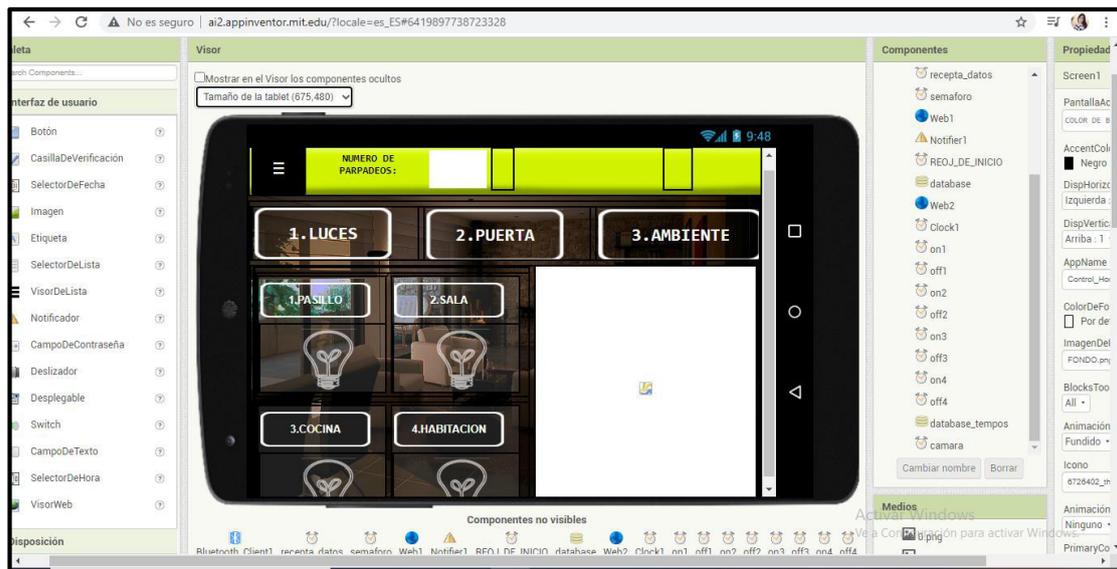


Figura 1.5 Interfaz gráfica de App diseñada

Modulo Bluetooth HC-05

En la Tabla 1.2, se observa las características principales del módulo *Bluetooth* HC-05.

Tabla 1.2 Características del módulo *Bluetooth* HC-05

Características	Descripción
Voltaje de operación	3.6- 6 (V)
Consumo de corriente	50 (mA)
Alcance	10 (m)
Bluetooth	V2.0
Pin	Función
State	Para conectar un led de salida y visualizar comunicación
Rx	Recepción de datos a un voltaje de 3.3 (V)
Tx	Transmisión de datos
GND	Tierra del módulo
+5 (V)	Alimentación del módulo
EN	En nivel alto, está en modo configuración

2 Metodología

Para el desarrollo del proyecto se empleó una investigación exploratoria, analítica y aplicada porque la implementación del sistema se realizó mediante el uso de tecnologías existentes. Para la fase inicial del proyecto se realizó una investigación exploratoria de las tecnologías vigentes para la adquisición de ondas cerebrales, ya que se quiso determinar la opción comercial de tecnología no invasiva adecuada, de esta manera, procesar la información de la actividad cerebral de la persona con discapacidad motriz. Una vez obtenida la información, mediante el método analítico se establecieron los algoritmos de funcionamiento del sistema, estos incluyeron: el algoritmo que permitió clasificar los niveles de atención, para posteriormente aplicarlos en el control *on/off* de cargas orientado a personas con discapacidad. El segundo es un algoritmo que permitió la identificación e inspección de objetos, determinación de posición de objetos en el espacio o determinación de coordenadas importantes de algún elemento; estas cualidades se aprovecharon para la identificación de rostro humano, y ese fue el punto de partida para identificar parpadeos de la persona con discapacidad.

En base a la metodología aplicada se implementó la BCI y se desarrolló una aplicación (App) para sistema *Android* que recepta información (número de parpadeos) desde el Arduino maestro para encender y apagar las cargas seleccionadas por la persona con discapacidad, mediante la interfaz gráfica de la App desarrollada. En la fase de implementación se desarrollaron los códigos necesarios para la comunicación entre los diferentes dispositivos. Se consideró la sincronización de la velocidad de señal para la comunicación *Bluetooth* entre dispositivo de adquisición de señales cerebrales, Arduino y App, para el envío de datos desde la diadema al Arduino maestro y desde el Arduino maestro a la App para confirmación de parpadeos. Se instalaron cargas *ON/OFF* y una cámara en la puerta principal; con el fin de que el usuario pueda controlarlas en el sistema domótico sin necesidad de tocar ningún dispositivo o elemento.

Una vez finalizada la implementación, se ejecutó una evaluación de todas las configuraciones y del funcionamiento del sistema. De esta manera se realizaron pruebas para verificar la correcta comunicación entre cada uno de los dispositivos del sistema. Primero se evaluó que las comunicaciones entre el dispositivo de adquisición de señales cerebrales, Arduino maestro, Arduino esclavo y computador sean exitosas. La segunda prueba, fue establecer el nivel de atención promedio para activar el sistema, considerando que cada persona tiene diferente capacidad de enfoque. Por último, se probó la comunicación en red entre la App diseñada, las cargas e iluminación.

3 Resultados y discusión

El sistema de control implementado está constituido por dos etapas. La primera es una BCI, conformada por una diadema *Mindwave Mobile*, un Arduino maestro, un Arduino esclavo, y un ordenador. La segunda, es un sistema de control inalámbrico entre una App y los puntos inalámbricos de control de cargas y luminarias, mediante WiFi y *Bluetooth*. En la Figura 3.1 se observa el diagrama donde se contempla las etapas del sistema.

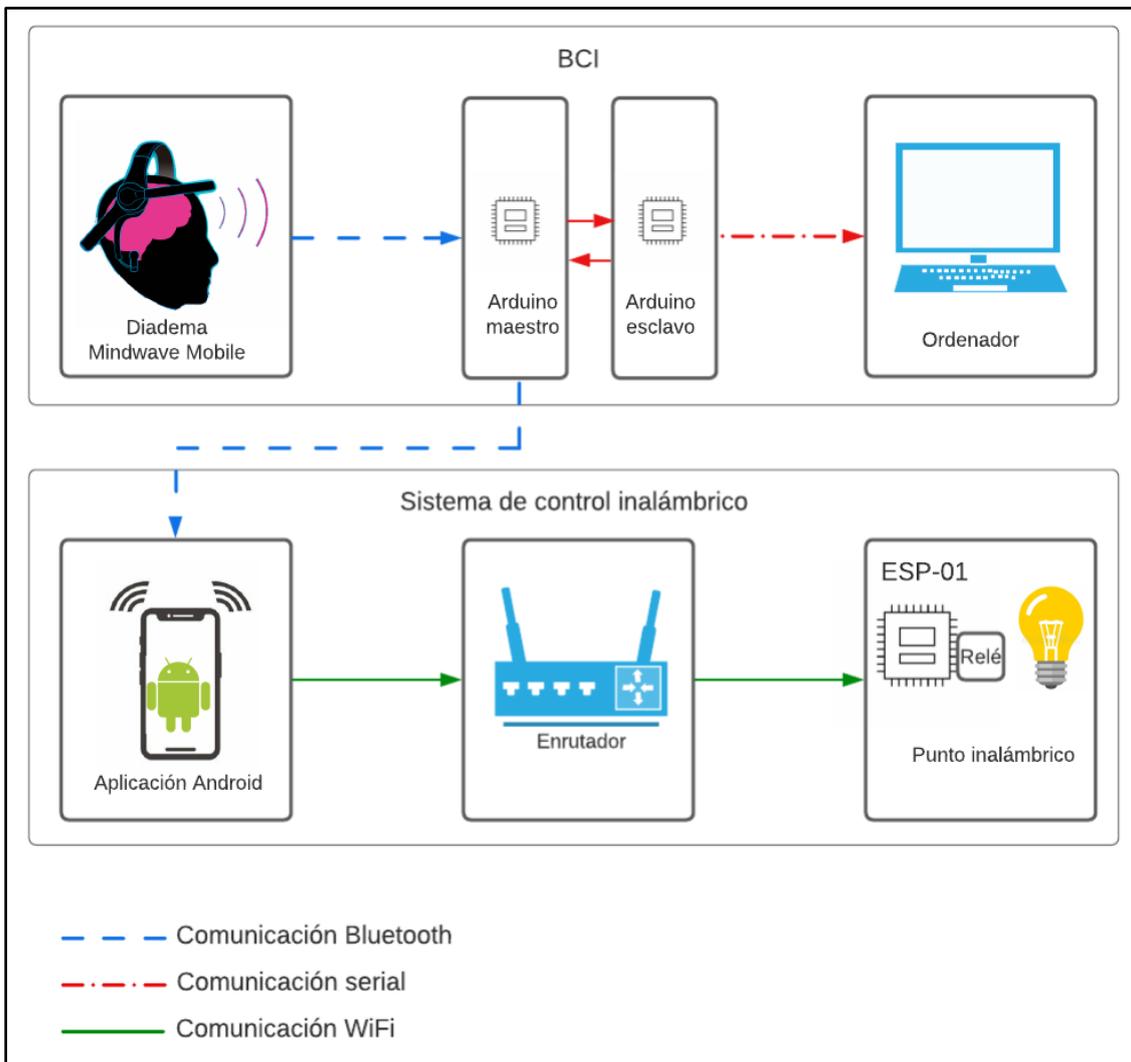


Figura 3.1 Etapas del sistema de control

Tabla 3.1 Características del sistema

CARACTERISTICAS	REQUERIMIENTOS DEL SISTEMA
a) Comunicación inalámbrica	Que se recepte y emita los datos mediante <i>Bluetooth</i> y <i>WiFi</i>
b) Comunicación bidireccional por puerto serial entre la computadora, Arduino maestro y esclavo.	Que se reciba la confirmación del parpadeo desde la computadora y se envíe al Arduino esclavo, para posteriormente ser enviadas a través del Arduino maestro a una aplicación <i>Android</i>
c) Interfaz de usuario	Que la aplicación permita la comunicación con cada una de las cargas, a través de <i>WiFi</i>
d) Facilidad de instalación	Que las señales lleguen a las cargas sin alterar las instalaciones ya existentes en la vivienda
e) Hardware fiable	Que se descarte fallas en la implementación del sistema e instalación de puntos inalámbricos
f) Sistema de control manual e inalámbrico.	Que sirva tanto para la persona con discapacidad, como para el resto de las personas que cohabitan en casa
g) Adaptabilidad	Que el sistema se pueda adaptar a cualquier vivienda, sin importar la estructura o características de la red de Internet, aprovechando que el sistema es netamente inalámbrico

3.1 Tecnologías para adquisición de ondas cerebrales

Los sistemas o métodos de adquisición de datos son de dos tipos:

Métodos invasivos: se incrustan quirúrgicamente en el cerebro humano de manera invasiva, de tal manera que pueden detectar señales a través de electrodos. Un ejemplo es la electrocorticografía (ECoG), que registra la actividad del cerebro a través de una red de electrodos [9]. En la Figura 3.2, se observa el método invasivo.



Figura 3.2 Método invasivo [9]

Métodos No invasivos: los electrodos son ubicados fuera del cráneo sobre el cuero cabelludo, para medir las señales del cerebro. La relación señal a ruido (SNR) es baja, debido a que se colocan los electrodos fuera del cráneo, y esto es un factor que afecta la calidad de la señal. Este tipo de método se observa en la Figura 3.3.



Figura 3.3 Método no invasivo [9]

El método no invasivo es el adecuado para trabajar en el sistema domótico, ya que no requiere una intervención quirúrgica ni implante de un electrodo en el cerebro humano, además ayuda a personas que han sufrido algún tipo de parálisis, a controlar las prótesis con las señales eléctricas del cerebro.

Medidores de ondas cerebrales comerciales no invasivos

En la Tabla 3.2, se observan los principales medidores de ondas cerebrales disponibles en el mercado.

Tabla 3.2 Principales medidores de ondas cerebrales

Emotiv – Epoc Flex	Emotiv- Epoc	Ifocus band- Focus band
		
Emotiv- Insight	Interaxon- Muse	Neurosky- Mindwave Mobile
		

Diadema *Mindwave Mobile*

Es una diadema que mide y transmite datos del espectro de potencia de las ondas de electroencefalograma (ondas alfa, beta, gamma, theta) de forma segura mediante *Bluetooth* a una computadora o dispositivo *Android*; donde posteriormente, se observa cómo cambian las ondas cerebrales en tiempo real, controlar los niveles de atención y relajación e incluso aprender cómo reacciona el cerebro estímulos [10].

Especificaciones técnicas

- Emparejamiento inalámbrico automático
- Batería AAA individual (no incluida)
- 8 horas de tiempo de funcionamiento de la batería
- Módulo de modo dual *Bluetooth* (BT)/ *Bluetooth Low Energy* (BLE) (rango de 10 m).
- ID de auriculares estáticos
- Plataformas compatibles: iOS (IOS 8 o posterior), *Windows* (XP / 7/8/10), Mac (OSX 10.8 o posterior) y *Android* (*Android* 2.3 o posterior).
- Velocidad de señal de 9600 baudios [10].

Las partes de la diadema *Mindwave Mobile* se aprecian en la Figura 3.4.

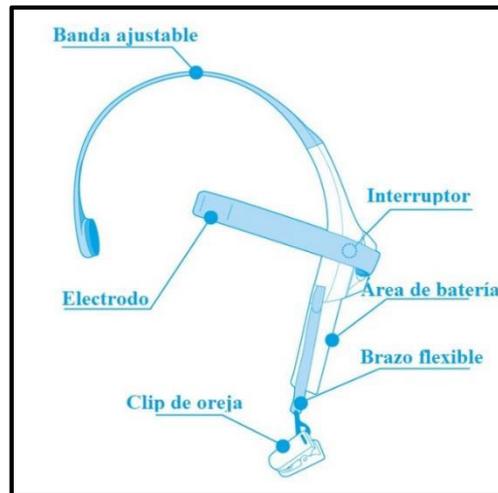


Figura 3.4 Partes *Mindwave Mobile* [15]

Se selecciona este dispositivo debido a su costo (económico) a pesar que su SNR es bajo ya que está orientado a un módulo didáctico.

3.2 Algoritmos de funcionamiento del sistema

En esta sección se resuelve la primera parte del sistema para lo cual se describe el algoritmo para clasificación de los niveles de las señales sensadas, niveles de atención y el algoritmo para identificación de parpadeos. En la Figura 3.5 se indica la etapa mencionada.

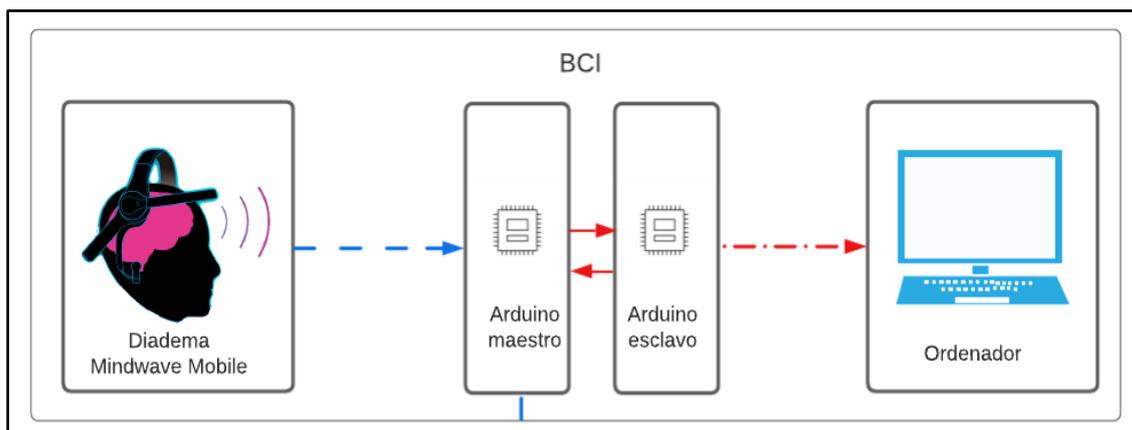


Figura 3.5 Primera etapa del sistema

Algoritmo para clasificación de los niveles de señales sensadas y niveles de atención

El algoritmo para clasificación de los niveles de señales sensadas y niveles de atención es elaborado en Arduino, para lo que se utiliza el protocolo de información del chip *ThinkGear* que es parte de la diadema *Mindwave*.

El chip *ThinkGear* se encarga de atenuar el ruido ambiental para realizar mediciones en áreas no aisladas eléctricamente. El electrodo de la diadema *Mindwave*, sensa las señales, las envía al chip y este las procesa y las convierte en una secuencia de datos digitales [11].

En la Figura 3.6, se observa el aspecto físico de un *ThinkGear ASIC Module*.

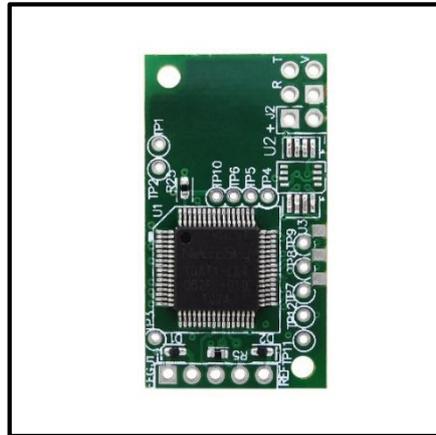


Figura 3.6 *ThinkGear ASIC Module*

Dichos datos digitales procesados por el chip *ThinkGear*, son enviados mediante *Bluetooth*, para ser aprovechados y clasificados de acuerdo a la aplicación.

En la Tabla 3.3, se observan los valores de datos procesados por el chip *ThinkGear* y su detalle correspondiente.

Tabla 3.3 Valores de datos del chip *ThinkGear*

Byte	Valor	Explicación
0	0XAA	Dato para sincronizar
1	0XAA	Dato para sincronizar
2	0X08	Longitud de carga útil
3	0X02	Calidad de la señal. Valor que oscila entre 0 a 255
4	0X20	Detección de señal deficiente
5	0X01	Nivel de batería
6	0X7E	Nivel de batería
7	0X04	Atención
8	0X12	Nivel de atención
9	0X05	Meditación
10	0X60	Nivel de meditación
11	0XE3	Checksum

De los valores mostrados se utilizaron los siguientes:

- calidad de la señal (0X02)
- nivel de atención (0X12).

En el Anexo 2 se observa el código en Arduino para crear la librería y asignar variables a los diferentes bytes que envía el chip *ThinkGear*.

Las variables a usarse son:

- `readSignalQuality()` para la calidad de la señal.
- `read_Attention()` para el nivel de atención.

Los datos de estas variables se envían mediante *Bluetooth* al Arduino maestro.

En la Figura 3.7 se muestra el diagrama de la librería creada para asignar nombres a las variables de Calidad de la Señal y Atención.

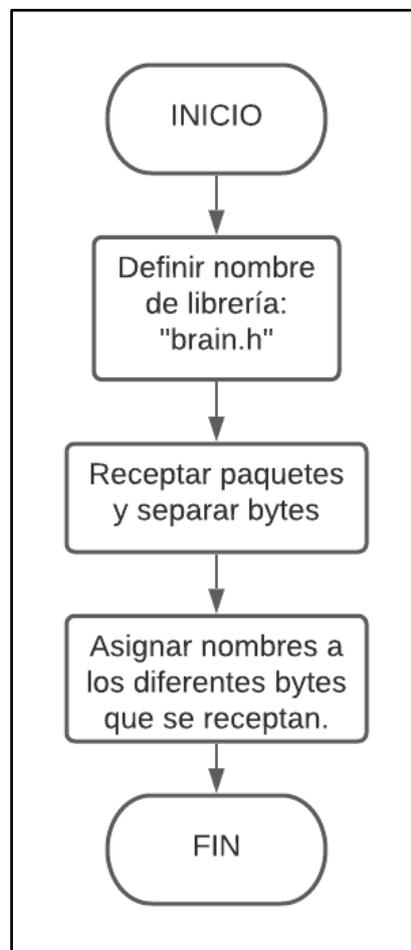


Figura 3.7 Diagrama de flujo de librería Brain.h

Algoritmo para identificación de parpadeos

Para la identificación de parpadeos se utilizó inteligencia artificial para escanear el rostro y detectar cuando la persona está parpadeando. Para esta sección se utilizó una biblioteca de visión artificial para Python (*Open Source Computer Vision*).

OPEN CV

Open Source Computer Vision (OpenCV) es una biblioteca de visión artificial de código abierto, análisis de imágenes y aprendizaje automático. Por ello, existen un gran número de algoritmos que permiten identificar rostro, reconocer objetos, detectar movimientos de manos entre otras, con unas pocas líneas de código [12].

Posterior al reconocimiento facial y detección de parpadeos, se utilizó una librería para habilitar el puerto serial y tener comunicación con el Arduino esclavo e indicar cuando ha ocurrido el parpadeo.

En la Figura 3.8, se muestra el diagrama de flujo del algoritmo para identificación de parpadeos.

En el Anexo 3, se observa el código escrito en Python.

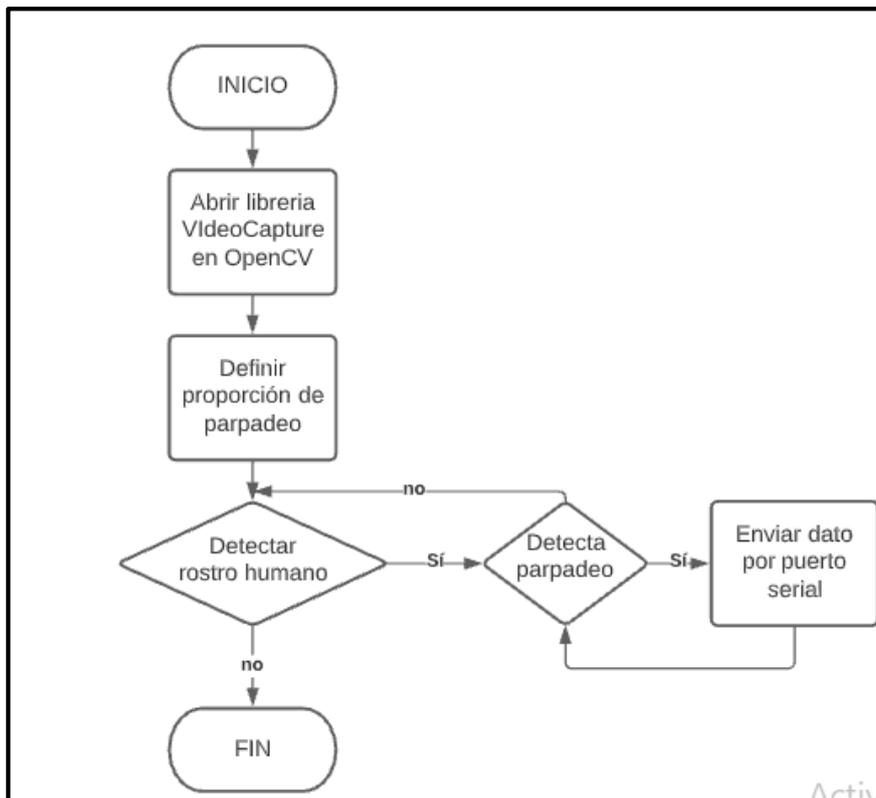


Figura 3.8 Diagrama de flujo de algoritmo para identificar parpadeo

En esta fase el Arduino esclavo envía el indicador que hubo el evento de parpadeo al Arduino maestro para que se comunique con la App.

3.3 Niveles de atención e identificación de parpadeos para el control *on/off*

Arduino maestro

El Arduino maestro se ha programado de tal manera que, mediante el *Bluetooth* HC-05 reciba los datos y con ayuda de la librería *Brain.h*, se trabaje solamente con los valores de atención y de calidad de la señal. El diagrama de flujo de la comunicación entre el Arduino maestro y esclavo se aprecia en la Figura 3.9.

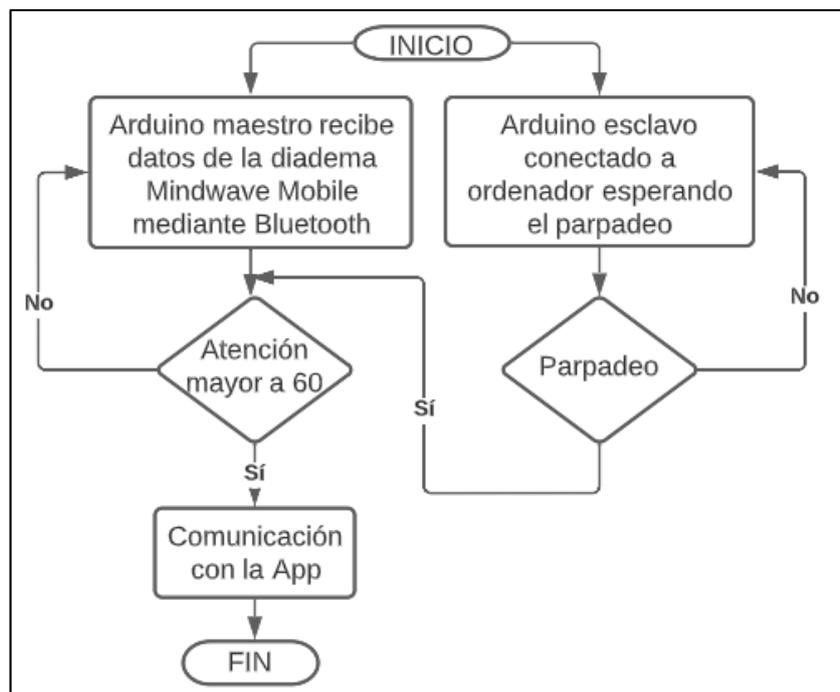


Figura 3.9 Diagrama de flujo de la comunicación entre Arduino maestro y Arduino esclavo

En el Anexo 4, se observa el código escrito en Arduino IDE para recepción de datos y comunicación con la App. Adicional se muestra a través de diodos LED (*Light Emitting Diode*) el nivel de atención y la calidad de la señal. En la Figura 3.10, se muestra el diagrama del Arduino Maestro, el cual tiene dos módulos *Bluetooth* conectados. El primero recibe los datos que envía la diadema *Mindwave Mobile*; mientras que el segundo envía datos a la App. Esto es posible gracias a la librería *SoftwareSerial.h* que permite configurar más pines del Arduino para tener más de una comunicación serial al mismo tiempo. El Arduino tiene configurado los pines 1 y 2 para transmisión y recepción

los cuales fueron usados para conectar el primer módulo *Bluetooth* para recibir los datos desde la diadema. Con la librería *SoftwareSerial.h* se configuró los pines 10 y 11 para transmisión y recepción de datos y conectar el segundo módulo *Bluetooth* para tener comunicación con la App.

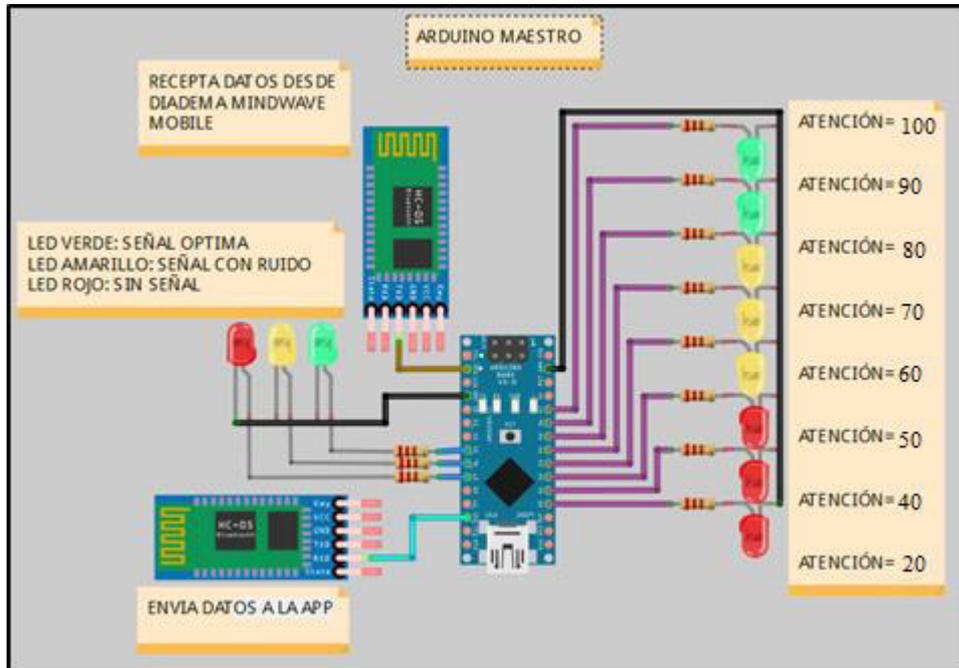


Figura 3.10 Esquema Arduino maestro

Para que el sistema funcione debe estar encendido el led verde, que indica que la señal es óptima y tiene buena calidad. Esto se determina por medio del valor de `readSignalQuality ()`. Si la variable de la calidad de señal tiene el valor de 0, indica que está óptima para trabajar; si tiene el valor de 200 indica que la diadema no está enviando señal o datos; por último, si tiene cualquier otro valor es porque la señal receptada esta con ruido externo.

Se ha programado para que a partir de un nivel de atención mayor a 60 el sistema está listo para conectarse con la App. En este estado el Arduino maestro solo espera la confirmación desde el Arduino esclavo que hubo el evento de parpadeo.

Arduino Esclavo

El esquema del Arduino esclavo se muestra en la Figura 3.11, donde se observa que el Arduino está conectado a un computador mediante el puerto Serial.

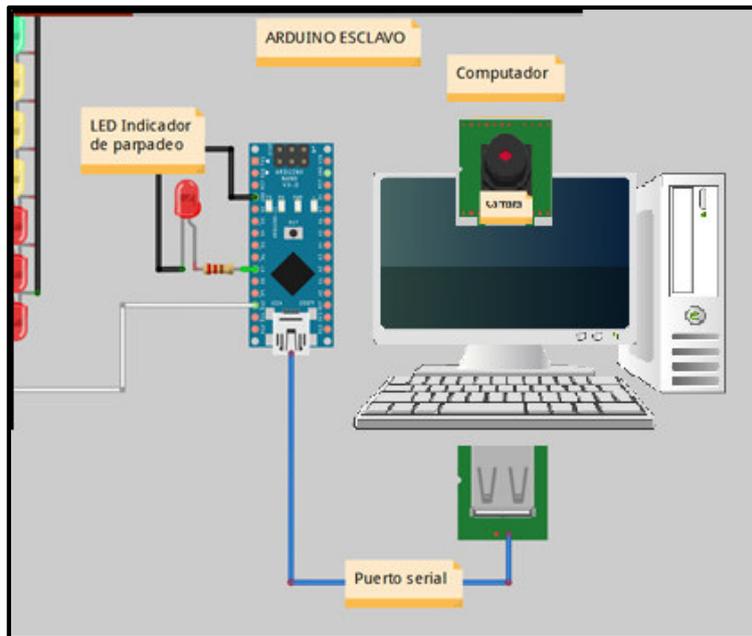


Figura 3.11 Esquema Arduino esclavo

En esta fase la cámara de la computadora es usada para la identificación de parpadeos. Una vez detectado el parpadeo, se envía una orden al Arduino esclavo que ha ocurrido el evento de parpadeo. El Arduino esclavo y el Arduino maestro están conectados a través de pines seriales de transmisión (Tx) y recepción (Rx). En este caso se le ha asignado los pines 10 y 11 para Rx y Tx, respectivamente. En el Anexo 4 se observa el código de programación usado para el Arduino esclavo. La Figura 3.12, muestra el esquema en conjunto de la comunicación entre Arduino maestro y esclavo.

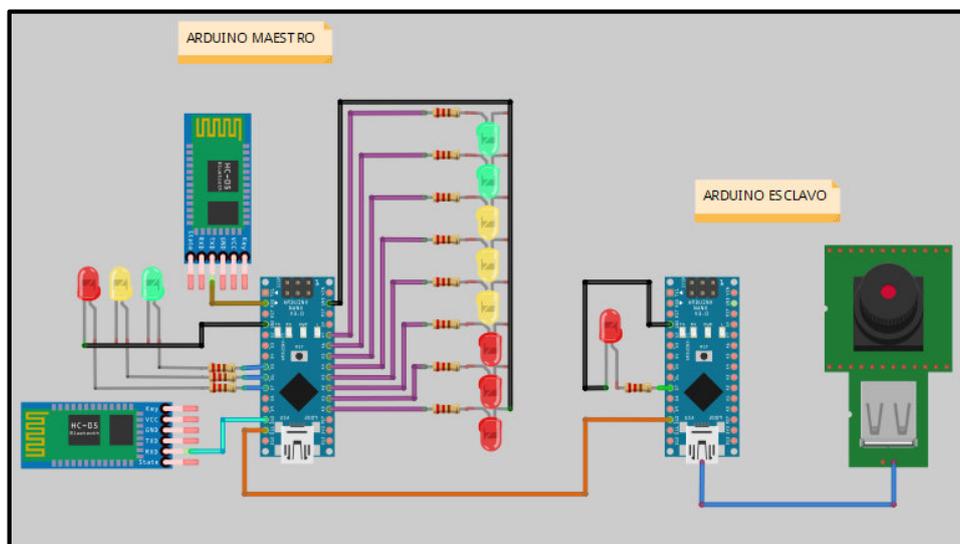


Figura 3.12 Esquema Arduino maestro- Arduino Esclavo

App para sistema *Android*

Para el sistema domótico, la aplicación para sistema *Android* (App) se desarrolló en *App Inventor*, la misma que va a utilizar el *Bluetooth* del celular para receptor los datos provenientes desde el Arduino maestro. En la Figura 3.13 se observa el diagrama de flujo de la App diseñada.

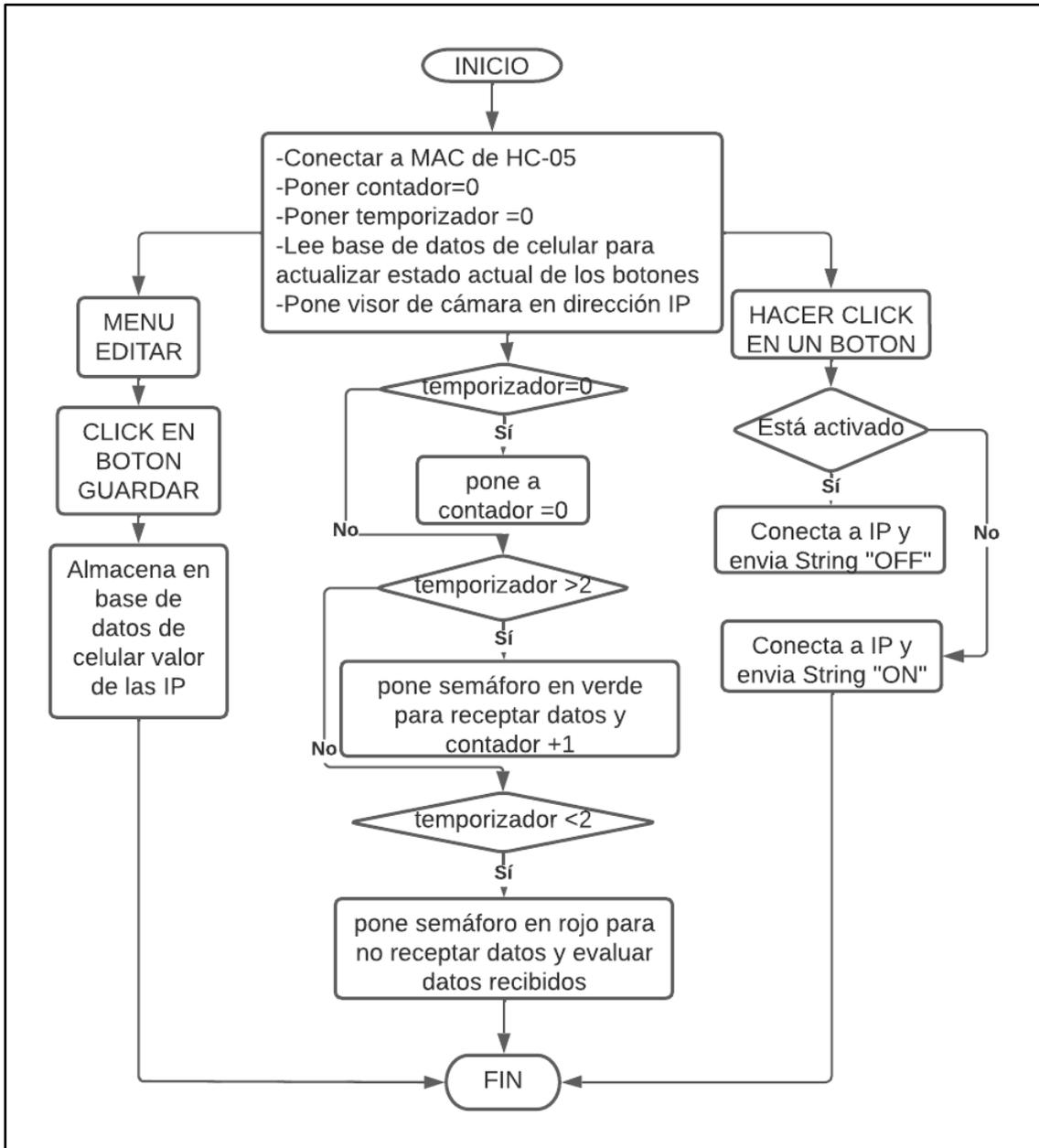


Figura 3.13 Diagrama de flujo de App diseñada

Partes de la App diseñada

La interfaz gráfica de la App consta de las siguientes cinco partes:

- barra de estado.
- menú editar.
- barra de botones principales.
- menú de botones secundarios.
- visor de cámara.

Barra de estado

La Figura 3.14, muestra la barra de estado donde se visualiza 3 indicadores: número de parpadeos, receptor de datos y estado de conexión.



Figura 3.14 Barra de estado

- **Número de parpadeos:** indica los parpadeos que la persona con discapacidad realiza si tiene una atención mayor que 60. Este dato lo envía el Arduino maestro mediante conexión *Bluetooth* a la App.
- **Receptor de datos:** este indicador estará activo durante 7 segundos recibiendo datos. En ese intervalo de tiempo, cada vez que el Arduino maestro envía un dato a la App, el contador de parpadeos se incrementa en uno. Pasado ese intervalo de tiempo este indicador se torna de color rojo, que es un modo reposo para poner el contador en cero. Este proceso se realiza porque para moverse sobre los diferentes menús de la App, se requiere un número determinado de parpadeos para acceder a una acción determinada.

En la Figura 3.15 y la Figura 3.16 se muestra el receptor de datos, recibiendo datos y en modo reposo, respectivamente.



Figura 3.15 Receptor de datos recibiendo datos



Figura 3.16 Receptor de datos modo reposo

La programación de bloques de los estados de cada botón se observa en el Anexo 7, donde se evalúa la posición de cada botón en función del parpadeo.

- **Estado de conexión:** muestra si el dispositivo *Android* está conectado al Arduino maestro y está listo para la comunicación. En esta sección se ha programado la App para que, al momento de inicializar la aplicación se conecte de manera automática al módulo *Bluetooth* HC-05, mediante la dirección Mac. En la Figura 3.17, se observa el diagrama de flujo del estado de conexión.



Figura 3.17 Diagrama de flujo del estado de conexión

Menú editar

Para acceder al menú editar se presiona el botón que se muestra en la Figura 3.18.



Figura 3.18 Botón menú editar

Dentro del menú editar, se ingresa de manera manual los nombres de los dispositivos que se van a controlar y las direcciones IP correspondientes. En la Figura 3.19 se

muestra el modo de edición activado, y en la Figura 3.20 se observa como ingresar de manera manual los caracteres que son alfanuméricos.



Figura 3.19 Modo edición activado

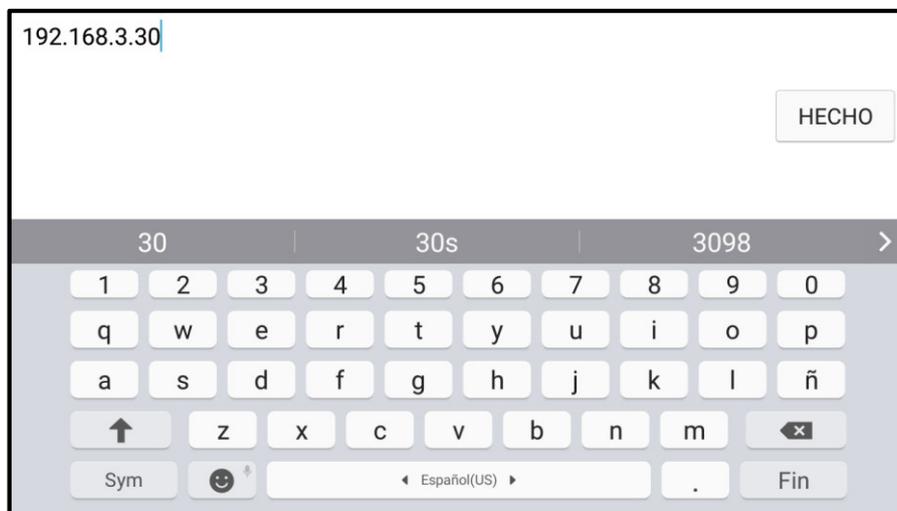


Figura 3.20 Ingreso de caracteres alfanuméricos

Barra de botones principales

En esta barra se encuentran tres botones que contienen los 3 menús principales.

- Menú luces.
- Menú puerta.
- Menú ambiente.

En la Figura 3.21 se observa la barra de botones principales.



Figura 3.21 Barra de botones principales

Cuando el usuario se encuentra en cualquiera de los 3 menús, el botón estará con un fondo de color naranja para indicar el menú actual activado. La Figura 3.22, Figura 3.23 y Figura 3.24 indican los respectivos menús y botones activados.



Figura 3.22 Botón Luces activado



Figura 3.23 Botón Puerta activado



Figura 3.24 Botón Ambiente activado

En la primera etapa se parpadea una vez para ir a menú luces, dos veces para ir al menú puerta y tres veces para ir al menú ambiente. Luego de esta acción se espera dos segundos para que el contador regrese a cero y parpadear nuevamente dentro del menú escogido para acceder al dispositivo a controlar.

Menú de Botones secundarios

Cada botón secundario de la App mantiene una comunicación constante con los relés que se encargan de encender o apagar las luminarias o cargas. Para este proceso se realizó la comunicación utilizando el Protocolo de transferencia de hipertexto (http) para enviar un *String* o texto a la placa ESP-01, para que se encargue de accionar el relé [13].

En la Tabla 3.4, se observa las características del módulo ESP-01.

Tabla 3.4 Características del módulo ESP-01

Característica	Detalle
Tensión de alimentación	3.3 (V)
Protocolo soportado	802.11 b/g/n
Soporte de red	2.4 (GHz)
Banda	2400 (MHz)
Wi-Fi Direct	Soft Access Point
Potencia de salida	0.15 (W)
Procesador integrado	32 bits

De la misma manera se tiene tres submenús asociados a cada menú principal.

- **Menú Luces:** en este menú se encuentran cuatro botones para controlar el encendido y apagado de luminarias ubicadas en cuatro lugares diferentes de la casa. Para acceder a cada botón se parpadea de acuerdo al numeral que está asociado, es decir, si se quiere encender o apagar la luminaria del pasillo se parpadea una vez. Si la luminaria está apagada al parpadear una vez más la luminaria se enciende, caso contrario se parpadea dos veces más y se apaga. Después de realizar la acción de encendido o apagado, el contador regresa a la primera etapa, donde se escogerá nuevamente el menú principal que se desee.

Cada botón está programado para que se muestre de diferente color cuando está encendida o no la luminaria. Si esta de color amarillo indica que la luminaria está encendida, mientras que si esta de color blanco, indica que la luminaria está apagada.

A cada punto de conexión se le asigna una dirección IP para gestionar la conexión entre el dispositivo y la App. Si se asigna una dirección IP dinámica, éstas cambiarán cada cierto tiempo y por consiguiente se debe cambiar las direcciones en la App, como se muestra en la Figura 3.20. Por esta razón se asigna direcciones IP estáticas para mantener siempre los dispositivos conectados en la red, ofreciendo una conexión más estable y un control exclusivo.

En el Anexo 5, se observa el código utilizado para asignar una dirección IP estática a los módulos ESP-01.

Cada punto inalámbrico (rele+ESP-01) tiene su diferente dirección IP que sirve para asociar cada botón de la App indistintamente a esa dirección IP. Desde la App se envía un texto acorde a la acción que se realiza. Si se apaga una luminaria o carga la App envía un "OFF", mientras que si se enciende, envía un "ON". Cuando el ESP-01 recibe el texto "ON" u "OFF", activa o desactiva el relé, respectivamente a través del pin 0, que es el pin donde está conectado el relé. A continuación en la Tabla 3.5 se observa las diferentes direcciones IP usadas en esta sección.

Tabla 3.5 Direcciones IP asignadas a las luminarias

LUMINARIA	DIRECCIÓN IP
Pasillo	192.168.3.30
Sala	192.168.3.31
Cocina	192.168.3.32
Habitación	192.168.3.33

- **Menú puerta:** este menú contiene el botón que servirá para accionar la chapa eléctrica que está instalada en la puerta principal de la casa. De la misma manera el relé encargado de abrir la puerta tiene una dirección IP estática, el cual al recibir la orden desde la App, se acciona y abre la puerta. En la Tabla 3.6, se observa la dirección IP estática asignada a la chapa eléctrica.

Tabla 3.6 Dirección IP estática asignada a chapa eléctrica

PUERTA	DIRECCIÓN IP
Chapa eléctrica	192.168.3.34

- **Menú ambiente:** se disponen de dos botones para ventilación y calefacción que están asociados a un relé y una dirección IP estática. Tiene el mismo principio de funcionamiento que las luminarias, ya que se acciona un relé controlando un ventilador y un calefactor. Las direcciones IP asignadas a estas cargas se las observa en la Tabla 3.7

Tabla 3.7 Direcciones IP estáticas asignadas a las cargas

CARGA	DIRECCIÓN IP
Ventilación	192.168.3.35
Calefacción	192.168.3.36

Visor de cámara

Es un espacio que siempre se muestra en la App, y refleja la imagen de la cámara instalada en la puerta principal, como se observa en la Figura 3.25. La cámara que se ha instalado es la ESP32 cam, a la cual se le ha asignado una dirección IP estática mediante un código escrito en Arduino IDE. El código se observa en el Anexo 8. La dirección IP utilizada es 192.168.3.37. En la Tabla 3.8 se observa las características del módulo ESP-32 Cam.



Figura 3.25 Visor de cámara

Tabla 3.8 Características del módulo ESP-32 Cam

Característica	Detalle
Voltaje de alimentación	5 (V).
Módulo Wi-Fi BT	802.11b/g/n
Tipo de cámara	OV2640 2MP
Velocidad de reloj	160 (MHz)
Frecuencia principal	240 (MHz)
Interfaces	UART / SPI / I2C / PWM / ADC / DAC
Modos de operación	STA / AP / STA+AP

3.4 Sistema de control domótico basado en los niveles de atención e identificación de parpadeos

Para la implementación de sistema es necesario verificar todas las comunicaciones entre cada uno de los dispositivos.

Comunicación entre dispositivo de adquisición de señales cerebrales con el Arduino maestro mediante *Bluetooth*.

Primero se identifica el interruptor *On/off* de la diadema *Mindwave Mobile* y accionarlo como se muestra en la Figura 3.26



Figura 3.26 Interruptor *On/off* de diadema *Mindwave Mobile*

Posterior a esto se coloca la diadema en la cabeza, asegurándose que el clip este presionando el lóbulo de la oreja y el electrodo tenga contacto con la piel de la frente como se muestra en la Figura 3.27 y Figura 3.28.

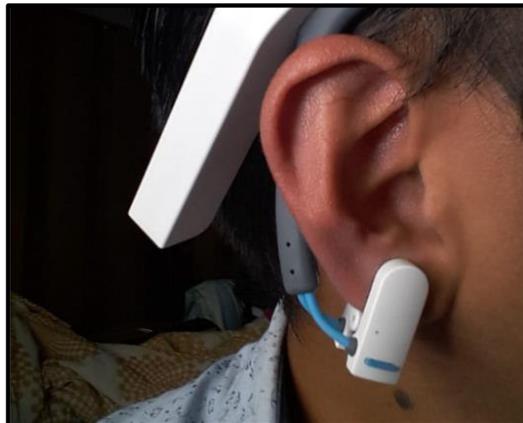


Figura 3.27 Clip presionando el lóbulo de la oreja



Figura 3.28 Contacto con la piel y electrodo

Mientras el diodo indicador de señal está de color rojo, muestra que no hay conexión entre el Arduino maestro con la diadema *Mindwave Mobile*, como se indica en la Figura 3.29.



Figura 3.29 LED rojo indica que no hay conexión

Después de encender la diadema *Mindwave Mobile*, se observa que se enciende el diodo amarillo para indicar que ya hay una conexión entre el Arduino maestro y la diadema, pero aún no hay recepción de datos, como se indica en la Figura 3.30.



Figura 3.30 Indicador de conexión sin recepción de datos

Luego, hay que esperar que se establezca la señal por unos segundos hasta que el led indicador que se encienda, sea el led verde como se muestra en la Figura 3.31.

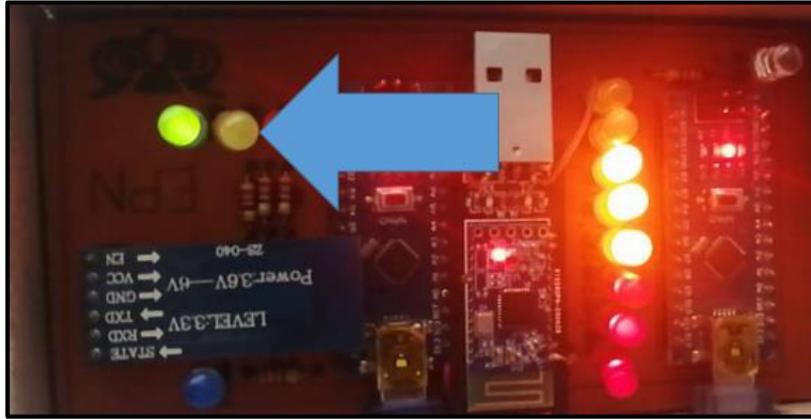


Figura 3.31 Indicador de recepción de datos

Los leds indicadores de nivel de atención también se encenderán de acuerdo al valor de atención que se tenga en ese instante. Estos valores se observan en la Figura 3.10.

Comunicación entre Arduino maestro con el Arduino esclavo

Tanto Arduino maestro y esclavo tiene configurados los pines 10 y 11 como Rx y Tx respectivamente a una velocidad de señal de 9600 baudios mediante el puerto serial. Esta configuración se observa en el Anexo 4 y Anexo 5.

La comunicación serial entre el Arduino maestro y el Arduino esclavo se muestra en la Figura 3.32, mediante la placa impresa utilizada en el proyecto.

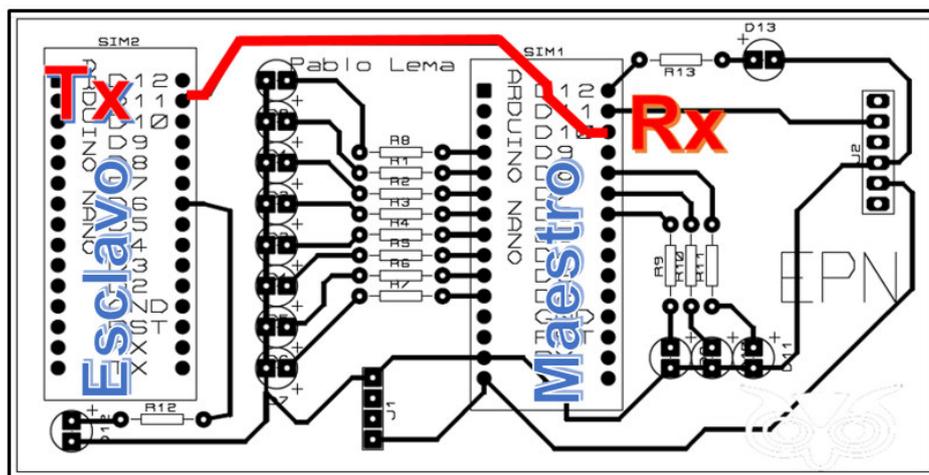


Figura 3.32 Comunicación serial Maestro-Esclavo

Comunicación entre Arduino esclavo con el computador

Se usa el puerto serial para la comunicación entre el Arduino esclavo y el computador. El programa escrito en Python usa la cámara del computador como recurso para escanear el rostro humano y determinar si el individuo realiza parpadeos. A través del

puerto serial y mediante una librería de Python, se envía un dato de tipo *string* al Arduino esclavo cada vez que ocurre el evento de parpadeo. Se observa en las líneas de código de la Figura 3.33 y Figura 3.34 como se envía desde el computador el dato del parpadeo y cuando el Arduino esclavo recibe dicha información, respectivamente.

```
#Imprime carácter por el puerto serial
time.sleep(0.5)
arduino.write(str.encode('a'))
```

Figura 3.33 Línea de código Python para imprimir carácter

```
{
  if (Serial.read() == 'a') { //Condicional si hay evento de parpadeo
```

Figura 3.34 Línea de código Arduino para recibir caracter desde Python
Cada vez que ocurre el parpadeo, el Arduino esclavo enciende un led por 500 milisegundos, como se muestra en la Figura 3.35.

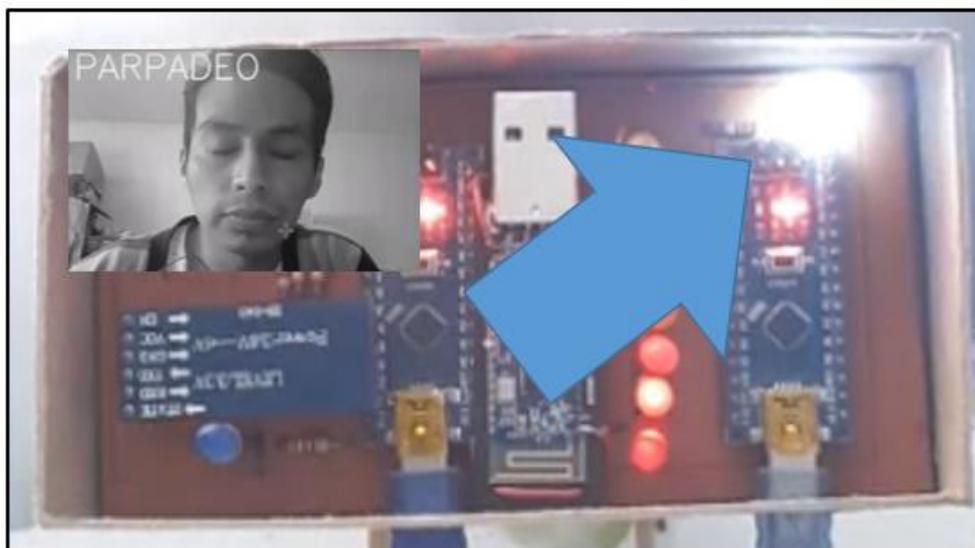


Figura 3.35 Led indicador de parpadeo

Comunicación del Arduino maestro con la App mediante *Bluetooth*

El Arduino maestro se comunica con la App para que muestre en el indicador de parpadeos, el número de parpadeos que recibe desde el Arduino esclavo. Este dato el Arduino maestro lo envía a la App mediante la comunicación *Bluetooth* siempre y cuando el nivel de atención sea mayor a 60.

En la Figura 3.36 se muestra el diagrama de flujo que envía el dato del parpadeo a la App mediante *Bluetooth*. Cuando la App está conectada al Arduino maestro imprime un mensaje en la barra estado de conexión de la App, como se muestra en la Figura 3.37.

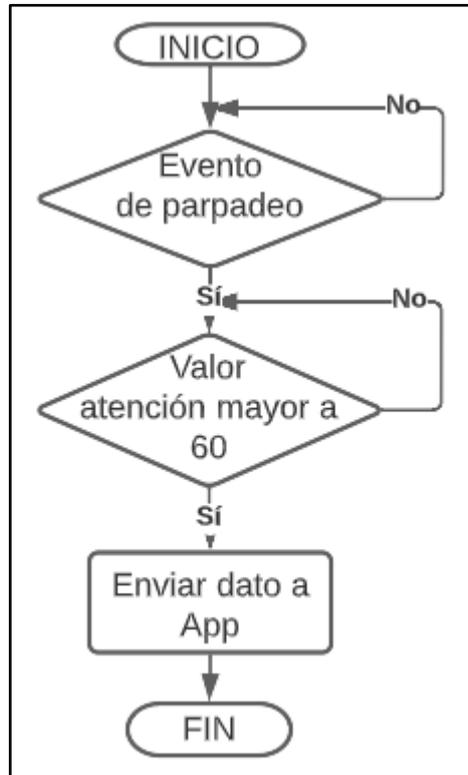


Figura 3.36 Diagrama de flujo para enviar dato a App



Figura 3.37 Barra de estado de conexión

Comunicación entre la App con cada punto inalámbrico para control

La App está programada de tal manera que se pueda controlar las cargas o luminarias de manera manual y comprobar la comunicación entre la App y los puntos inalámbricos. A través de WiFi se ha establecido la comunicación entre los puntos inalámbricos y la App. El proceso para crear la red entre la App y cada punto es el siguiente:

1. Asignar una dirección IP al router: para este proyecto se asignó la dirección 192.168.3.1, como se muestra en la Figura 3.38.

nombre del producto	MIK-013					
Tiempo de actividad	3 20:56:09					
Fecha/Hora	Sun Jan 4 20:56:09 1970					
Configuración De LAN						
Dirección IP	192.168.3.1					
Máscara de subred	255.255.255.0					
Servidor DHCP	Activar					
Dirección MAC	90:8D:78:CE:54:25					
Configuración WLAN						
inalámbrico	Activado					
Modo	AP					
SSID	JEREMY					
Cifrado	WPA/WPA2 Mixed					
Canal	6					
SSID de difusión	Activado					
WPS	Desactivado					
Estado del repetidor	Desconectado					
Configuración De WAN						
Interfaz	Protocolo	Dirección IP	Puerta de enlace	DNS	Dirección MAC	Estado
WAN	DHCP	192.168.100.10	192.168.100.1	192.168.100.1	90:8D:78:CE:54:26	Conexión activa(DHCP Client)
<input type="button" value="Actualizar"/>						

Figura 3.38 Propiedades del router

2. Conectar el dispositivo *Android* a la misma red WiFi, como se muestra en la Figura 3.39. Cada dispositivo está conectado a la misma red como se muestra en las líneas de código de la Figura 3.40.

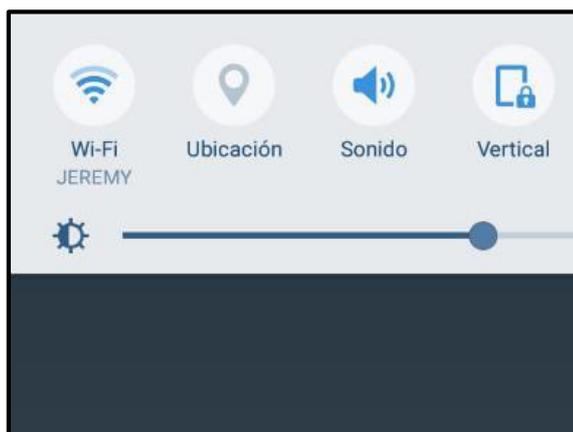


Figura 3.39 Conexión a la red WiFi

```
#include <ESP8266WiFi.h>

//Ingresar datos de red WiFi

const char* ssid = "JEREMY";

const char* password = "jeremy93";
```

Figura 3.40 Líneas de código para conectarse a red WiFi

Cuando se presiona un botón manualmente, verificar si se enciende la carga o luminaria correspondiente, como se muestra en la Figura 3.41.



Figura 3.41 Verificación de encendido de luminaria, modo manual

3.5 Pruebas de funcionamiento del sistema de control domótico implementado

Para esta fase del proyecto se comprobó que haya conexión en todo el sistema, desde la recepción de datos hasta el encendido de una luminaria o carga siguiendo el diagrama que se muestra en la Figura 3.42. Además, se observa el diagrama de instalación física de un punto inalámbrico en la Figura 3.43.

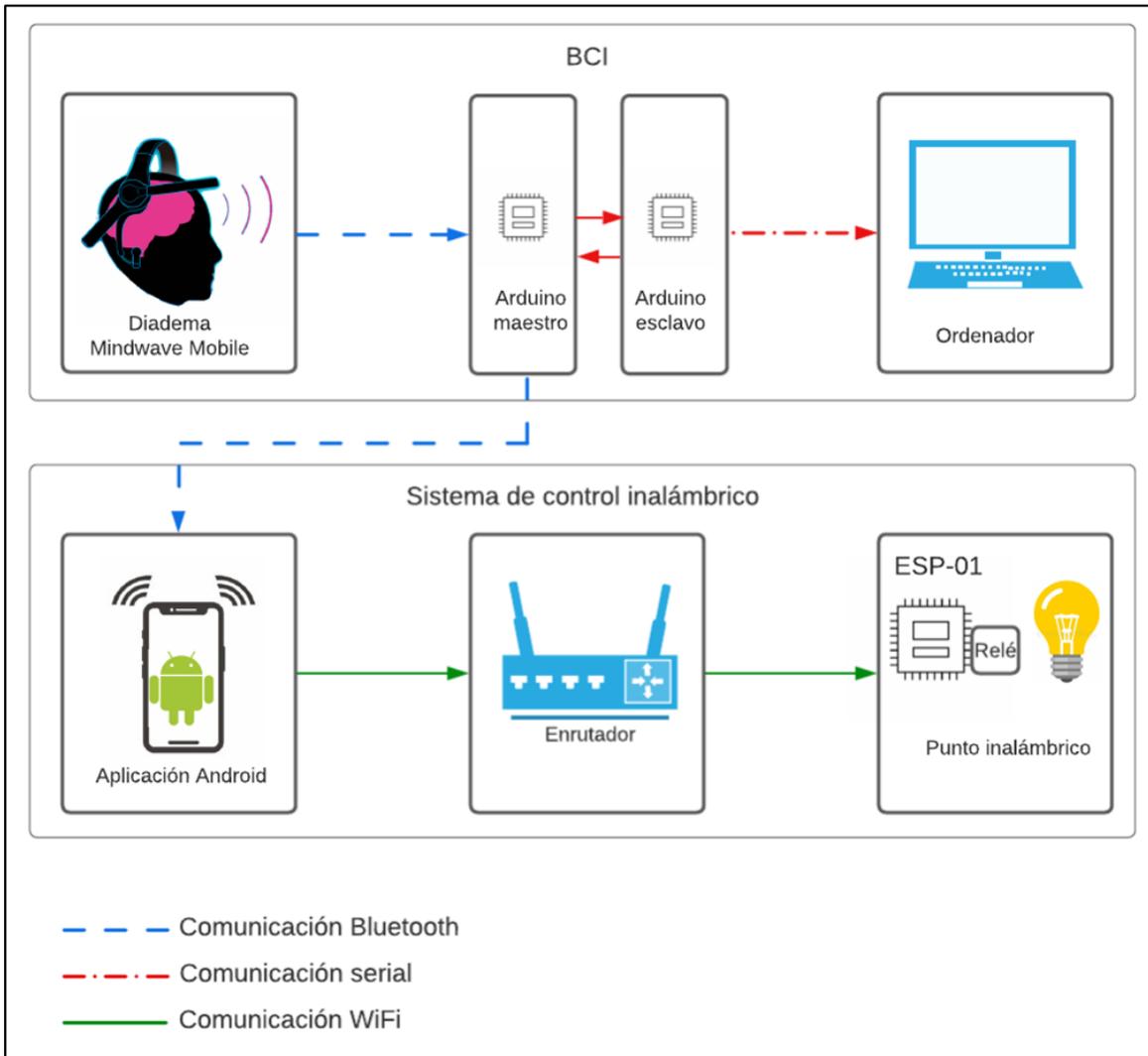


Figura 3.42 Diagrama de la comunicación del sistema



Figura 3.43 Representación física de la instalación de un punto inalámbrico

Prueba de comunicación entre la diadema *Mindwave Mobile* y el Arduino maestro

El enfoque se dió al led indicador de color verde, ya que este led al estar encendido, confirma que hay envío y recepción de datos desde la diadema hasta el Arduino maestro. En la Tabla 3.9 se observa cinco pruebas realizadas y el tiempo en segundos que demora la diadema en comunicarse y enviar datos al Arduino maestro.

Tabla 3.9 Pruebas de comunicación entre diadema y Arduino maestro

Prueba	Led verde encendido	Tiempo (s)
1	Ok	35
2	Ok	37
3	Ok	38
4	Ok	35
5	Ok	40

Como se observa el tiempo promedio que demora la diadema en enviar datos al Arduino es de 37 segundos. Esto quiere decir que el sistema no tarda mucho tiempo en funcionar.

Prueba de comunicación entre el Arduino esclavo y ordenador

En esta prueba se detectó el parpadeo a través de la cámara del ordenador y se observó que el led de color blanco se encienda. En la Tabla 3.10 se observa cinco pruebas realizadas y la confirmación del encendido del led blanco.

Tabla 3.10 Pruebas de comunicación entre ordenador y Arduino esclavo

Prueba	Led blanco encendido
1	Ok
2	Ok
3	Fallido
4	Ok
5	Ok

Como se observa hay una prueba fallida y esto se debió a que la iluminación de la habitación donde se realizó la prueba era escasa. Esto quiere decir que, si no se tiene iluminación adecuada, la lectura de parpadeos que hará la cámara del ordenador no será exitosa.

Prueba de funcionamiento de iluminación

Para esta fase de prueba se contó con la colaboración de una persona, como se observa en la Figura 3.44. La prueba consistía en encender una luminaria, y de esta manera se probó indirectamente la comunicación entre el Arduino maestro y la App.



Figura 3.44 Prueba de funcionamiento de iluminación 1

El objetivo es encender la luminaria de la sala. Primero, para acceder al menú LUCES se parpadea una vez, como se muestra en la Figura 3.45.



Figura 3.45 Prueba de funcionamiento de iluminación 2

Luego, se parpadea 2 veces para acceder al botón SALA, y encender la luminaria correspondiente, como se muestra en la Figura 3.46.

Se le pidió a la persona que encienda y apague la misma luminaria repitiendo este proceso cinco veces.

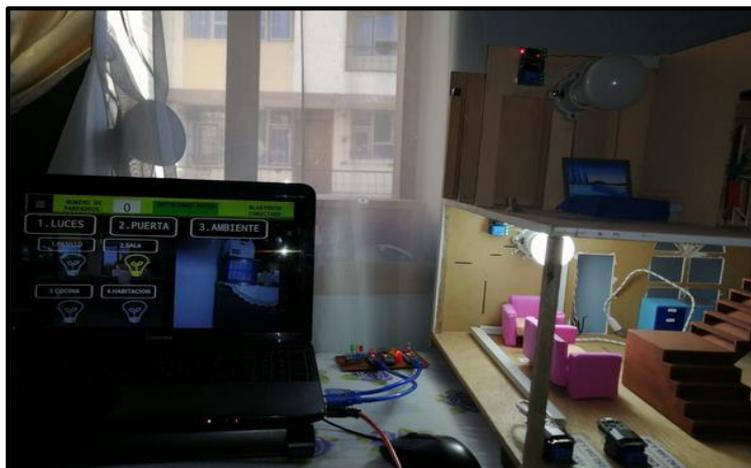


Figura 3.46 Prueba de funcionamiento de iluminación 3

Tabla 3.11 Prueba de funcionamiento de iluminación

Prueba	Resultado
1	Ok
2	Ok
3	Ok
4	Fallido
5	Ok

La Tabla 3.11 muestra las cinco pruebas realizadas y el resultado obtenido de la prueba. Como se observa hay una prueba fallida debido que la persona pierde el nivel de atención requerido. Para esta prueba, se observó que el nivel de atención influye en la comunicación exitosa del sistema, para lo cual se analizó un valor promedio que le facilite a cualquier persona la utilización del sistema.

Prueba para determinar nivel de atención

Para esta fase se ha realizado diferentes pruebas con 15 personas de diferente edad, para determinar un promedio en el nivel de atención y establecerlo en el sistema doméstico, porque cada persona tiene diferente capacidad de enfoque.

Según las estadísticas de discapacidad del Consejo Nacional para la Igualdad de Discapacidades, personas de 25 años en adelante representan el 76.69% del total de personas con discapacidad en la ciudad de Quito, Pichincha [2]. Por esa razón, se ha

realizado las pruebas en personas con esas edades. Se tomó referencia las personas mayores de 20 años, ya que representan el 65.03% del total de la población con discapacidad física. Según se muestra en la Figura 3.47, es difícil que la persona alcance valores altos de atención y mantenga un enfoque constante.

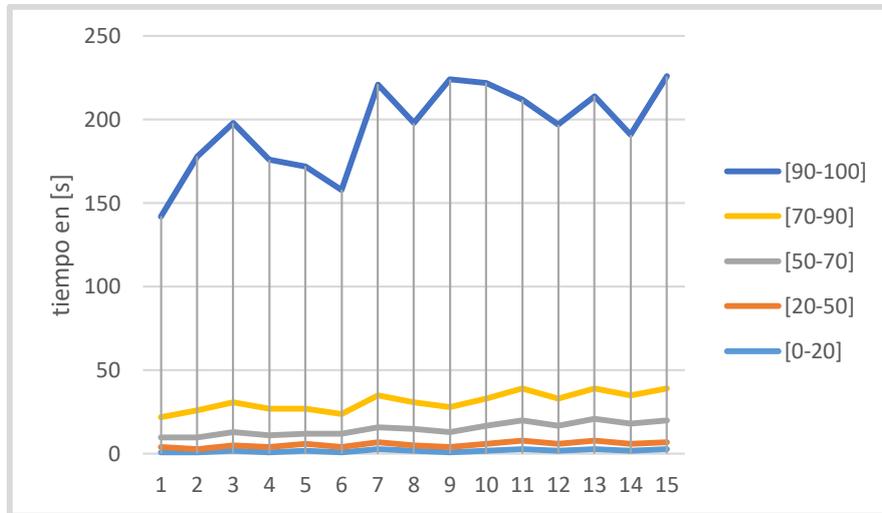


Figura 3.47 Nivel de atención respecto al tiempo de respuesta

Tabla 3.12 Datos en relación a la edad y tiempo para alcanzar un nivel de atención

Persona N°	Edad	Tiempo que demora en llegar a nivel de atención				
		[0-20]	[20-50]	[50-70]	[70-90]	[90-100]
1	28 años	1 [s]	3 [s]	6 [s]	12 [s]	120 [s]
2	30 años	1 [s]	2 [s]	7 [s]	16 [s]	152 [s]
3	31 años	2 [s]	3 [s]	8 [s]	18 [s]	167 [s]
4	32 años	1 [s]	3 [s]	7 [s]	16 [s]	149 [s]
5	34 años	2 [s]	4 [s]	6 [s]	15 [s]	145 [s]
6	35 años	1 [s]	3 [s]	8 [s]	12 [s]	134 [s]
7	37 años	3 [s]	4 [s]	9 [s]	19 [s]	186 [s]
8	40 años	2 [s]	3 [s]	10 [s]	16 [s]	167 [s]
9	42 años	1 [s]	3 [s]	9 [s]	15 [s]	196 [s]
10	45 años	2 [s]	4 [s]	11 [s]	16 [s]	189 [s]
11	48 años	3 [s]	5 [s]	12 [s]	19 [s]	173 [s]
12	50 años	2 [s]	4 [s]	11 [s]	16 [s]	164 [s]
13	55 años	3 [s]	5 [s]	13 [s]	18 [s]	175 [s]
14	58 años	2 [s]	4 [s]	12 [s]	17 [s]	156 [s]
15	60 años	3 [s]	4 [s]	13 [s]	19 [s]	187 [s]

De acuerdo a la Tabla 3.12 un valor adecuado de atención está entre los 50 y 70, ya que a valores altos de atención se requiere tiempos más extensos para llegar a dicho nivel de atención. Por esa razón se ha tomado de referencia el nivel de atención igual a 60 que es un promedio entre los valores de 50 y 70, considerando el tiempo que le toma a la persona en llegar a ese nivel.

Cabe recalcar que esta prueba no está limitada a la edad de una persona sino al tiempo que una persona puede llegar a un nivel de atención determinado.

Prueba de funcionamiento del sistema

En la Tabla 3.13 se observan las pruebas realizadas con cada uno de los puntos inalámbricos.

Tabla 3.13 Pruebas de funcionamiento

Prueba	Resultado	Observación
Acceso al menú LUCES	Ok	1 parpadeo
Acceso a PASILLO	Ok	1 parpadeo
Acceso a SALA	Ok	2 parpadeos
Acceso a HABITACIÓN	Ok	3 parpadeos
Acceso a COCINA	Ok	4 parpadeos
Acceso al menú PUERTA	Ok	2 parpadeos
Acceso a ABRIR PUERTA	Ok	1 parpadeo
Acceso al menú AMBIENTE	Ok	3 parpadeos
Acceso a VENTILADOR	Ok	1 parpadeo
Acceso a CALEFACTOR	Ok	2 parpadeos

De acuerdo a la Tabla 3.13 todas las pruebas fueron exitosas, ya que se consideraron las deficiencias en las pruebas anteriores y se rectificó para evitar pruebas fallidas, tales como iluminación deficiente en el ambiente para lectura de parpadeos y nivel de atención adecuado.

3.6 Manual de funcionamiento

En la Figura 3.48, se muestra el código QR del video donde está el manual de funcionamiento del sistema.



Figura 3.48 Código QR de manual de funcionamiento del Sistema implementado

4 Conclusiones y recomendaciones

4.1 Conclusiones

- La fiabilidad y la posibilidad de adquisición de señales depende de la calidad del electrodo en contacto con la piel y la correcta ubicación sobre la frente de la persona con discapacidad.
- Los algoritmos de funcionamiento del sistema son: algoritmo para clasificación de los niveles de señales sensadas y niveles de atención, y algoritmo para identificación de parpadeos.
- A través de la diadema *Mindwave* se especifica un valor determinado de atención que, mediante las pruebas realizadas en 15 personas se determinó que el 60% son adultos mayores de 36 años, los cuales tienen un tiempo de respuesta de 17.22 (s) hasta alcanzar un nivel de atención mayor a 60. Debajo de este nivel de atención se evitó que el sistema funcione con parpadeo involuntario.
- Para la identificación de parpadeos se utilizó la visión artificial mediante la cámara del ordenador, para detectar los parpadeos (algoritmo) mediante un código escrito en Python a través de la librería de cv2 de OpenCv.
- Las pruebas realizadas evidenciaron la comunicación entre cada uno de los dispositivos en cada fase del sistema. La prueba de comunicación entre la diadema *Mindwave* y el Arduino maestro fue exitosa, con un tiempo promedio de 37 (s) para tener una comunicación total. En la prueba de comunicación entre el ordenador y el Arduino esclavo, la iluminación deficiente del ambiente provocó que el 80% de las pruebas sean exitosas. La prueba de funcionamiento de luminarias tuvo el mismo resultado que la prueba anterior ya que la iluminación deficiente del ambiente evita la lectura y detección del parpadeo y en consecuencia, la comunicación con las demás etapas del sistema. La prueba de funcionamiento del sistema fue exitosa ya que se modificó los diferentes parámetros que evitaban que las pruebas anteriores se cumplan al 100%, tal como el valor de atención adecuado y nivel de iluminación apropiado.
- En la primera etapa de la BCI se utilizó el protocolo 802.15.1 (*Bluetooth*), el cual envía tramas de 8 bytes, utilizando empaquetamiento de datos a una velocidad de 9600 baudios para la comunicación entre Arduino esclavo, Arduino maestro y computador, de esa manera evitar pérdida de información, colisiones o retrasos.

- La comunicación entre la App diseñada y los puntos inalámbricos, utiliza la tecnología 802.11n (WiFi), que al ser una tecnología de red inalámbrica permite la interacción entre dispositivos sin necesidad de cableado eléctrico.

4.2 Recomendaciones

- Cambiar las baterías AAA a la diadema *Mindwave Mobile* cada 8 horas, de esa manera se evita problemas en el envío de datos.
- Ubicar la computadora en un lugar bien iluminado para evitar errores en la lectura de los parpadeos ya que, al utilizar la cámara de la computadora, con menos luminosidad aumenta la probabilidad de error de lectura.
- Verificar la dirección IP del router, para poder comunicar la App y puntos inalámbricos, ya que en base a la dirección del router se asignan las direcciones IP a los puntos.
- Estudiar las nuevas tecnologías y versiones de la diadema *Mindwave Mobile*, ya que vienen incorporadas nuevas características como las EEG (Electroencefalograma) y EOG (Electrooculograma), y se las puede implementar en el sistema, evitando el uso de programas de visión artificial.
- Utilizar la App en dispositivos que tengan una pantalla amplia donde se pueda visualizar la interfaz gráfica y de esa manera poder usar el sistema de manera eficaz. Si no se tiene un dispositivo con tal característica se recomienda instalar la aplicación AnyDesk y proyectar la pantalla del dispositivo *Android* en la computadora.
- Implementar el sistema en una casa real.

5 Referencias bibliográficas

- [1] O. M. d. I. Salud, "OMS," [Online]. Available: https://www.who.int/disabilities/world_report/2011/es/. [Accessed 02 05 2021].
- [2] C. N. p. I. I. d. Discapacidades, "Consejo Nacional para la Igualdad de Discapacidades," [Online]. Available: <https://www.consejodiscapacidades.gob.ec/estadisticas-de-discapacidad/>. [Accessed 02 05 2021].
- [3] Significados, "Significados," [Online]. Available: <https://www.significados.com/sistema-nervioso/#:~:text=El%20sistema%20nervioso%20es%20un,seres%20humanos%2C%20poseen%20sistemas%20nerviosos..> [Accessed 02 05 2021].
- [4] N. F. Back, "NeuroFeedBack," [Online]. Available: <https://www.neurofeedback.cat/que-son-las-ondas-cerebrales/>. [Accessed 03 05 2021].
- [5] L. m. e. maravillosa, "La mente es maravillosa," 12 03 2019. [Online]. Available: <https://lamenteesmaravillosa.com/tipos-de-ondas-cerebrales/>.
- [6] S. Elo, "SciElo," 09 10 2008. [Online]. Available: http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S1909-97622008000200005.
- [7] COGNIFIT, "CogniFit," [Online]. Available: <https://www.cognifit.com/es/atencion>. [Accessed 04 05 2021].
- [8] P. E. Sum, "Programo Ergo Sum," [Online]. Available: <https://www.programoergosum.com/cursos-online/appinventor/27-curso-de-programacion-con-app-inventor/primeros-pasos>. [Accessed 04 05 2021].
- [9] BlogThinking, "BlogThinking," 08 02 2018. [Online]. Available: <https://blogthinkbig.com/el-futuro-de-la-tecnologia-brain-computer/#:~:text=BCI%20invasivo%20se%20plantea%20como,humano%2C%20como%20electrodos%20colocados%20quir%C3%BAArgicamente.&text=Para%>

20permitir%20esta%20haza%C3%B1a%2C%20cada,transmitir%20se%C3%B1ales. [Accessed 04 05 2021].

- [10] SanDoRobotics, "SanDoRobotics," [Online]. Available: <https://sandorobotics.com/producto/sen-14758/#:~:text=Este%20es%20el%20MindWave%20Mobile,%2C%20iOS%2C%20o%20dispositivo%20Android..> [Accessed 04 05 2021].
- [11] Neurosky, "Neurosky Inc," [Online]. Available: http://developer.neurosky.com/docs/doku.php?id=thinkgear_communications_protocol. [Accessed 04 05 2021].
- [12] RoboLogs, "RoboLogs," [Online]. Available: [https://robologs.net/tutoriales/tutoriales-opencv/#:~:text=OpenCV%20\(Open%20Source%20Computer%20Vision,de%20imagen%20y%20aprendizaje%20autom%C3%A1tico.&text=OpenCV%20es%20una%20librer%C3%ADa%20multiplataforma,%2C%20Python%2C%20Java%20y%20Matlab..](https://robologs.net/tutoriales/tutoriales-opencv/#:~:text=OpenCV%20(Open%20Source%20Computer%20Vision,de%20imagen%20y%20aprendizaje%20autom%C3%A1tico.&text=OpenCV%20es%20una%20librer%C3%ADa%20multiplataforma,%2C%20Python%2C%20Java%20y%20Matlab..) [Accessed 05 05 2021].
- [13] Interpolados, "Interpolados," [Online]. Available: <https://interpolados.wordpress.com/tag/puerto-80/>. [Accessed 06 05 2021].
- [14] M. Clinic, "MAYO CLINIC," [Online]. Available: <https://www.mayoclinic.org/es-es/tests-procedures/eeg/about/pac-20393875>. [Accessed 03 05 2021].
- [15] P. E. Sum, "Programo Ergo Sum," [Online]. Available: <https://www.programoergosum.com/cursos-online/arduino/253-curso-de-iniciacion-a-arduino/software-arduino-ide>. [Accessed 04 05 2021].

ANEXOS

Anexo 1: certificado de funcionamiento



ESCUELA POLITECNICA NACIONAL

Campus Politécnico "J. Rubén Orellana R

Quito, 13 de julio de 2022

CERTIFICADO DE FUNCIONAMIENTO DE PROYECTO DE TITULACIÓN

Yo, Catalina Elizabeth Armas Freire, docente a tiempo completo de la Escuela Politécnica Nacional y como director de este trabajo de titulación, certifico que he constatado el correcto funcionamiento del sistema de control domótico orientado a personas con discapacidad motriz, el cual fue implementado por el estudiante Lema Coronel Pablo Miguel.

El proyecto cumple con los requerimientos de diseño y parámetros necesarios para que los usuarios puedan usar el sistema de control.

DIRECTOR

Ing. Catalina Elizabeth Armas Freire, Msc.

Ladrón de Guevara E11-253, Escuela de Formación de Tecnólogos, Oficina 01. EXT: 2732

email: elizabeth.arms@epn.edu.ec

Quito-Ecuador

Anexo 2: código de la librería brain.h

Por la extensión del código, en el siguiente link se encuentra un archivo que contiene el código en su totalidad

[Código para la librería Brain.docx](#)

Anexo 3: código en Python para identificación de parpadeos

Por la extensión del código, en el siguiente link se encuentra un archivo que contiene el código en su totalidad

[Código en Python para identificación de parpadeos.docx](#)

Anexo 4: código de programación del Arduino maestro

Por la extensión del código, en el siguiente link se encuentra un archivo que contiene el código en su totalidad

[Código Arduino Maestro.docx](#)

Anexo 5: código de programación del Arduino esclavo

Por la extensión del código, en el siguiente link se encuentra un archivo que contiene el código en su totalidad

[Código de programación Arduino esclavo.docx](#)

Anexo 6: código de bloques de estado de cada botón de la app

Por la extensión del código, en el siguiente link se encuentra un archivo que contiene el código en su totalidad

[Código de bloque de botones de estado de la App.docx](#)

**Anexo 7: código para asignar una dirección IP estática a un
módulo Esp-01**

Por la extensión del código, en el siguiente link se encuentra un archivo que contiene el código en su totalidad

[Código de programación para asignar dirección IP a punto inalámbrico.docx](#)

Anexo 8: código para asignar IP estática a Esp32 Cam

Por la extensión del código, en el siguiente link se encuentra un archivo que contiene el código en su totalidad

[Código de programación para asignar dirección IP a cámara.docx](#)